ember

Application Framework API Reference

March 23 2011

120-3023-000B

Ember Corporation
25 Thomson Place
Boston, MA 02210
+1 (617) 951-0200
www.ember.com

wireless semiconductor solutions

Copyright © 2011 by Ember Corporation

All rights reserved.

The information in this document is subject to change without notice. The statements, configurations, technical data, and recommendations in this document are believed to be accurate and reliable but are presented without express or implied warranty. Users must take full responsibility for their applications of any products specified in this document. The information in this document is the property of Ember Corporation.

Title, ownership, and all rights in copyrights, patents, trademarks, trade secrets and other intellectual property rights in the Ember Proprietary Products and any copy, portion, or modification thereof, shall not transfer to Purchaser or its customers and shall remain in Ember and its licensors.

No source code rights are granted to Purchaser or its customers with respect to all Ember Application Software. Purchaser agrees not to copy, modify, alter, translate, decompile, disassemble, or reverse engineer the Ember Hardware (including without limitation any embedded software) or attempt to disable any security devices or codes incorporated in the Ember Hardware. Purchaser shall not alter, remove, or obscure any printed or displayed legal notices contained on or in the Ember Hardware.

Ember, Ember Enabled, EmberZNet, InSight, and the Ember logo are trademarks of Ember Corporation.

All other trademarks are the property of their respective holders.

About this Guide

Purpose

This document is a unified collection of API reference documentation covering EmberZNet 4.x. Application Framework. Ember recommends that you use this document as a searchable reference.

It includes all of the information contained in the html version of these materials that are provided as an online reference for developers of EmberZNet-based ZigBee wireless applications. There are three key advantages that this document provides over the online html versions:

- Everything is contained in this single document.
- This document is fully searchable using the Adobe Acrobat search engine that is part of the free Acrobat Reader (available from www.adobe.com).
- This document can be easily printed.

Audience

This document is intended for use by programmers and designers developing ZigBee wireless networking products based on the EmberZNet Stack Software 4.x.

This document assumes that the reader has a solid understanding of embedded systems design and programming in the C language. Experience with networking and radio frequency systems is useful but not expected.

Getting Help

Developer kit customers are eligible for training and technical support. You can use the Ember web site www.ember.com to obtain information about all Ember products and services, and to sign up for product support.

You can also contact Ember technical support at http://www.ember.com/support_index.html.

Application Framework API Reference

- Introduction
- Modules
  - Ember Application Framework API reference
    - General Application Framework interface
    - Application Framework types
    - Application Framework callback interface
    - Application Framework command buffer loading interface
  - Ember Application Framework Command Line Interface (CLI)
  - Servicediscovery
- Data Structures
  - _BlockThreshold
  - _ChatParticipant
  - _ChatRoom
  - _ConfigureReportingRecord
  - _ConfigureReportingStatusRecord
  - _DiscoverAttributesInfoRecord
  - _NeighborInfo
  - _NodeInformation
  - _Notification
  - _ReadAttributeStatusRecord
  - _ReadReportingConfigurationAttributeRecord
  - _ReadReportingConfigurationRecord
  - _ReportAttributeRecord
  - _SceneExtensionAttributeInfo
  - _SceneExtensionFieldSet
  - _WriteAttributeRecord
  - _WriteAttributeStatusRecord
  - EmAfOtaStorageLinuxConfig
  - EmberAfAllowedInterPanMessage
  - EmberAfAttributeMetadata
  - EmberAfAttributeMinMaxValue
  - EmberAfAttributeSearchRecord
  - EmberAfCluster
  - EmberAfClusterCommand
  - EmberAfDefaultAttributeValue
  - EmberAfDefaultOrMinMaxAttributeValue
  - EmberAfDefinedEndpoint
  - EmberAfEndpointList
  - EmberAfEndpointType
  - EmberAfEventContext
  - EmberAfIasZoneTableEntry
  - EmberAfIncomingMessage
  - EmberAfInterpanHeader

EmberAfLoadControlEvent
EmberAfManufacturerCodeEntry
EmberAfMessagingServerMessage
EmberAfOtaHeader
EmberAfOtaImageId
EmberAfReportEntry
EmberAfSceneTableEntry
EmberAfScheduledPrice
EmberAfServiceDiscoveryResult
EmberAfTagData
Data Fields
File List
_AF_API.top
af-structs.h
af-types.h
af.h
attribute-id.h
attribute-size.h
attribute-storage.h
attribute-type.h
bacnet-protocol-tunnel.h
basic-callback.h
basic.h
call-command-handler.h
callback.h
client-api.h
client-command-macro.h
cluster-id.h
color-control.h
command-id.h
core-cli.h
custom-cli.h
debug-printing-test.h
debug-printing.h
debug.h
demand-response-load-control.h
drlc-callback.h
drlc-server.h
enums.h
generic-tunnel.h
groups-server.h
ias-ace-server.h
ias-wd-server.h
ias-zone-table.h
identify.h

interpan.h
key-establishment-storage.h
key-establishment.h
level-control.h
load-control-event-table.h
messaging-client-callback.h
messaging-client.h
messaging-server.h
network-cli.h
on-off.h
ota-bootload-ncp.h
ota-bootload-xmodem.h
ota-cli.h
ota-client-page-request-test.h
ota-client-page-request.h
ota-client-policy.h
ota-client-signature-verify.h
ota-client.h
ota-server-policy.h
ota-server.h
ota-static-file-data.h
ota-storage-linux.h
ota-storage-simple-custom.h
ota-storage-simple-driver.h
ota-storage.h
ota.h
partner-link-key-exchange.h
plugin-cli.h
price-client-callback.h
price-client.h
price-server.h
print-cluster.h
psl-commissioning-callback.h
psl-commissioning.h
scenes.h
security-cli.h
simple-metering-client-callback.h
simple-metering-test.h
smart-energy-registration.h
stack-handlers.h
temperature-measurement.h
test-callback.h
thermostat.h
time-client.h

- time-server.h
- trust-center-keepalive.h
- tunneling-client-callback.h
- tunneling-client.h
- tunneling-server-callback.h
- tunneling-server.h
- zcl-cli.h
- zdo-cli.h
- Directory Hierarchy
  - app
    - framework
      - cli
      - gen
      - include
      - plugin
        - bacnet-protocol-tunnel
        - basic
          - gen
        - color-control
        - drlc
          - gen
        - drlc-server
        - generic-tunnel
        - groups-server
        - ias-ace-zone
        - ias-wd-server
        - identify
        - interpan
        - key-establishment
        - level-control
        - messaging-client
          - gen
        - messaging-server
        - on-off
        - ota-bootload
        - ota-client
        - ota-client-policy
        - ota-common
        - ota-server
        - ota-server-policy
        - ota-storage-common
        - ota-storage-posix-filesystem
        - ota-storage-simple
        - ota-storage-simple-ram
        - partner-link-key-exchange

- price-client
  - gen
- price-server
- psl-commissioning
  - gen
- scenes
- simple-metering-client
  - gen
- simple-metering-server
- smart-energy-registration
- temperature-measurement
- test
  - gen
- thermostat
- time-client
- time-server
- trust-center-keepalive
- tunneling-client
  - gen
- tunneling-server
  - gen
- util

Globals

Introduction

Application Framework V2

The Ember Application Framework V2 API Reference documentation includes the following API sets:

- **Ember Application Framework API reference**

Modules

Here is a list of all modules:

- Ember Application Framework API reference
  - General Application Framework interface
  - Application Framework types
  - Application Framework callback interface
  - Application Framework command buffer loading interface
- Ember Application Framework Command Line Interface (CLI)
- Servicediscovery

Ember Application Framework API reference

Modules

General Application Framework interface
Application Framework types
Application Framework callback interface
Application Framework command buffer loading interface

General Application Framework interface

[Ember Application Framework API reference]

Defines

#define	**CONFIGURATION_HEADER**

Attribute Storage

enum	{ **EMBER_AF_DATA_TYPE_ANALOG**, **EMBER_AF_DATA_TYPE_DISCRETE**, **EMBER_AF_DATA_TYPE_NONE** }
EmberAfDefinedEndpoint	**endpoints** []
EmberAfAttributeMetadata *	**emberAfLocateAttributeMetadata** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attribute, int8u mask, int16u manufacturerCode)
boolean	**emberAfContainsCluster** (int8u endpoint, **EmberAfClusterId** clusterId)
boolean	**emberAfContainsServer** (int8u endpoint, **EmberAfClusterId** clusterId)
boolean	**emberAfContainsClient** (int8u endpoint, **EmberAfClusterId** clusterId)
boolean	**emberAfReadOrUpdateAttribute** (**EmberAfAttributeSearchRecord** *attRecord, **EmberAfAttributeMetadata** **metadata, int8u *buffer, boolean write)
EmberAfStatus	**emberAfWriteAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int8u mask, int8u *dataPtr, int8u dataType)
EmberAfStatus	**emberAfWriteServerAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int8u *dataPtr, int8u dataType)
EmberAfStatus	**emberAfWriteClientAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int8u *dataPtr, int8u dataType)
EmberAfStatus	**emberAfWriteManufacturerSpecificServerAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int16u manufacturerCode, int8u *dataPtr, int8u dataType)
EmberAfStatus	**emberAfWriteManufacturerSpecificClientAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int16u manufacturerCode, int8u *dataPtr, int8u dataType)
EmberAfStatus	**emberAfVerifyAttributeWrite** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int8u mask, int16u manufacturerCode, int8u *buffer, int8u dataType)
EmberAfStatus	**emberAfReadAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int8u mask, int8u *dataPtr, int8u maxLength, int8u *dataType)
EmberAfStatus	**emberAfReadServerAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int8u *dataPtr, int8u maxLength)
EmberAfStatus	**emberAfReadClientAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int8u *dataPtr, int8u maxLength)
EmberAfStatus	**emberAfReadManufacturerSpecificServerAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int16u manufacturerCode, int8u *dataPtr, int8u maxLength)
EmberAfStatus	**emberAfReadManufacturerSpecificClientAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int16u manufacturerCode, int8u *dataPtr, int8u maxLength)
int8u	**emberAfGetDataSize** (int8u dataType)
int8u	**emberAfIndexFromEndpoint** (int8u endpoint)
int8u	**emberAfEndpointCount** (void)
int8u	**emberAfGetAttributeAnalogOrDiscreteType** (int8u dataType)
boolean	**emberAfIsTypeSigned** (int8u dataType)
int32u	**emberAfGetInt32u** (const int8u *message, int16u currentIndex, int16u msgLen)
int32u	**emberAfGetInt24u** (const int8u *message, int16u currentIndex, int16u msgLen)

int16u	**emberAfGetInt16u** (const int8u *message, int16u currentIndex, int16u msgLen)
int8u *	**emberAfGetString** (int8u *message, int16u currentIndex, int16u msgLen)
void	**emberAfCopyInt16u** (int8u *data, int16u index, int16u x)
void	**emberAfCopyInt24u** (int8u *data, int16u index, int32u x)
void	**emberAfCopyInt32u** (int8u *data, int16u index, int32u x)
void	**emberAfCopyString** (int8u *dest, int8u *src, int8u size)
int8u	**emberAfStringLength** (const int8u *buffer)
#define	**emberAfClusterIsManufacturerSpecific**(cluster)
#define	**emberAfAttributeIsReadOnly**(metadata)
#define	**emberAfAttributeIsClient**(metadata)
#define	**emberAfAttributeIsTokenized**(metadata)
#define	**emberAfAttributeIsSingleton**(metadata)
#define	**emberAfAttributeIsManufacturerSpecific**(metadata)
#define	**emberAfAttributeSize**(metadata)
#define	**emberAfEndpointFromIndex**(index)
#define	**emberAfProfileIdFromIndex**(index)
#define	**emberAfDeviceIdFromIndex**(index)
#define	**emberAfDeviceVersionFromIndex**(index)
#define	**emberAfPrimaryProfileId**()
#define	**emberAfPrimaryEndpoint**()
#define	**emberAfGetInt8u**(message, currentIndex, msgLen)
#define	**emberAfCopyInt8u**(data, index, x)

Hibernation

enum	{ **EMBER_AF_WAITING_FOR_DATA_ACK**, **EMBER_AF_LAST_POLL_GOT_DATA**, **EMBER_AF_WAITING_FOR_SERVICE_DISCOVERY**, **EMBER_AF_WAITING_FOR_ZDO_RESPONSE**, **EMBER_AF_WAITING_FOR_ZCL_RESPONSE**, **EMBER_AF_WAITING_FOR_REGISTRATION**, **EMBER_AF_WAITING_FOR_PARTNER_LINK_KEY_EXCHANGE** }
int32u	**emberAfHibernateDuration**
int32u	**emAfCurrentAppTasks**
EmberAfEventSleepControl	**emAfDefaultSleepControl**
void	**emberAfAddToCurrentAppTasks** (int32u mask)
void	**emberAfRemoveFromCurrentAppTasks** (int32u mask)
boolean	**emberAfOkToHibernate** (void)
void	**emberAfRunEvents** (void)
boolean	**emberAfStartMove** (void)
EmberStatus	**emberAfScheduleClusterTick** (int8u endpoint, int16u clusterId, boolean isClient, int32u timeMs, **EmberAfEventSleepControl** sleepControl)
EmberStatus	**emberAfDeactivateClusterTick** (int8u endpoint, int16u clusterId, boolean isClient)
EmberStatus	**emberAfEventControlSetDelay** (EmberEventControl *eventControl, int32u timeMs)
int32u	**emberAfMsToNextEvent** (int32u maxMs)
EmberAfEventSleepControl	**emberAfGetCurrentSleepControl** (void)
void	**emberAfSetDefaultSleepControl** (**EmberAfEventSleepControl** control)
#define	**MILLISECOND_TICKS_PER_QUARTERSECOND**
#define	**MILLISECOND_TICKS_PER_MINUTE**
#define	**MILLISECOND_TICKS_PER_HOUR**
#define	**EMBER_AF_SET_HIBERNATE_DURATION**(newValue)
#define	**EMBER_AF_SET_NAP_DURATION**(newValue)
#define	**EMBER_AF_WAKE_TIMEOUT**
#define	**EMBER_AF_WAKE_TIMEOUT_MASK**
#define	**emberAfOkToNap**()

#define	**emberAfCurrentAppTasks**()
#define	**EMBER_AF_CLIENT_CLUSTER_TICK**
#define	**EMBER_AF_SERVER_CLUSTER_TICK**
#define	**emberAfQSToNextEvent**(maxQS)
#define	**emberAfGetDefaultSleepControl**()

Miscellaneous

PGM **EmberAfOtaImageId**	**emberAfInvalidImageId**
boolean	**emberAfIsThisDataTypeAStringType** (int8u dataType)
int8u	**emberAfNextSequence** (void)
int8s	**emberAfCompareValues** (int8u *val1, int8u *val2, int8u len)
void	**emberAfGetEui64** (EmberEUI64 returnEui64)
EmberNodeId	**emberAfGetNodeId** (void)
#define	**EMBER_AF_NEW_IMAGE_VERIFICATION**
#define	**EMBER_AF_CONTINUE_IMAGE_VERIFY**

Messaging

boolean	**emberAfResponseWaitingFlag**
EmberAfClusterCommand *	**emAfCurrentCommand**
EmberStatus	**emberAfSendResponse** (void)
EmberStatus	**emberAfSendRequest** (void)
EmberStatus	**emberAfSendMulticast** (EmberMulticastId multicastId, EmberApsFrame *apsFrame, int16u messageLength, int8u *message)
EmberStatus	**emberAfSendBroadcast** (EmberNodeId destination, EmberApsFrame *apsFrame, int16u messageLength, int8u *message)
EmberStatus	**emberAfSendUnicast** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u messageLength, int8u *message)
EmberStatus	**emberAfSendInterPan** (EmberPanId panId, const EmberEUI64 destinationLongId, EmberNodeId destinationShortId, EmberMulticastId multicastId, **EmberAfClusterId** clusterId, **EmberAfProfileId** profileId, int16u messageLength, int8u *messageBytes)
EmberStatus	**emberAfSendEndDeviceBind** (int8u endpoint)
EmberStatus	**emberAfSendCommandMulticast** (EmberMulticastId multicastId)
EmberStatus	**emberAfSendCommandUnicast** (EmberOutgoingMessageType type, int16u indexOrDestination)
EmberStatus	**emberAfSendCommandBroadcast** (EmberNodeId destination)
EmberStatus	**emberAfSendCommandInterPan** (EmberPanId panId, const EmberEUI64 destinationLongId, EmberNodeId destinationShortId, EmberMulticastId multicastId, **EmberAfProfileId** profileId)
EmberStatus	**emberAfSendDefaultResponse** (const **EmberAfClusterCommand** *cmd, **EmberAfStatus** status)
int8u	**emberAfMaximumApsPayloadLength** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame)
EmberApsFrame *	**emberAfGetCommandApsFrame** (void)
EmberStatus	**emberAfFindDevicesSupportingCluster** (EmberNodeId target, **EmberAfClusterId** clusterId, boolean serverCluster, **EmberAfServiceDiscoveryCallback** *callback)
EmberStatus	**emberAfFindDevicesByProfileAndCluster** (EmberNodeId target, **EmberAfProfileId** profileId, **EmberAfClusterId** clusterId, boolean serverCluster, **EmberAfServiceDiscoveryCallback** *callback)
EmberStatus	**emberAfFindIeeeAddress** (EmberNodeId shortAddress, **EmberAfServiceDiscoveryCallback** *callback)
EmberStatus	**emberAfFindNodeId** (EmberEUI64 longAddress, **EmberAfServiceDiscoveryCallback** *callback)
int8u	**emberAfAddAddressTableEntry** (EmberEUI64 longId, EmberNodeId shortId)
EmberStatus	**emberAfSetAddressTableEntry** (int8u index, EmberEUI64 longId, EmberNodeId shortId)
EmberStatus	**emberAfInitiateKeyEstablishment** (EmberNodeId nodeId, int8u endpoint)
EmberStatus	**emberAfInitiateInterPanKeyEstablishment** (EmberPanId panId, const EmberEUI64 eui64)

boolean	**emberAfPerformingKeyEstablishment** (void)
EmberStatus	**emberAfInitiatePartnerLinkKeyExchange** (EmberNodeId target, int8u endpoint, **EmberAfPartnerLinkKeyExchangeCallback** *callback)
#define	**emberAfSetResponseWaiting**()
#define	**emberAfSendInterpanUnicast**(destinationShortAddress,destinationPanId, destinationAppProfileId,destinationClusterId,multicastIdOrZero,longAddressOrNull, messageLength,messageBytes)
#define	**emberAfSendCommandInterpanUnicast**(eui64,nodeId,panId,multicastId, profileId)
#define	**EMBER_AF_CLIENT_CLUSTER_DISCOVERY**
#define	**EMBER_AF_SERVER_CLUSTER_DISCOVERY**
#define	**emberAfCurrentCommand**()
#define	**emberAfCurrentEndpoint**()

Device Control

boolean	**emberAfIsDeviceEnabled** (int8u endpoint)
void	**emberAfSetDeviceEnabled** (int8u endpoint, boolean enabled)

Printing

boolean	**emberAfPrintEnabled** (int16u functionality)
void	**emberAfPrintBuffer** (int16u area, const int8u *buffer, int16u bufferLen, boolean withSpaces)
void	**emberAfPrintString** (int16u area, const int8u *buffer)
void	**emberAfPrint** (int16u functionality, PGM_P formatString,...)
void	**emberAfPrintln** (int16u functionality, PGM_P formatString,...)
void	**emberAfFlush** (int16u functionality)
void	**emberAfPrintOn** (int16u functionality)
void	**emberAfPrintOff** (int16u functionality)
void	**emberAfPrintAllOn** (void)
void	**emberAfPrintAllOff** (void)
void	**emberAfPrintStatus** (void)
void	**emberAfPrintLittleEndianEui64** (const EmberEUI64 eui64)
void	**emberAfPrintBigEndianEui64** (const EmberEUI64 eui64)
void	**emberAfPrintMessageData** (int8u *data, int16u length)
#define	**emberAfGuaranteedPrint**(...)
#define	**emberAfGuaranteedPrintln**(...)
#define	**emberAfGuaranteedPrintBuffer**(buffer, len, withSpace)
#define	**emberAfGuaranteedPrintString**(buffer)
#define	**emberAfGuaranteedFlush**()

Time

int32u	**emberAfGetCurrentTime** (void)
void	**emberAfSetTime** (int32u utcTime)
void	**emberAfPrintTime** (int32u utcTime)

Stack handler utility functions

void	**emberAfPollCompleteHandler** (EmberStatus status)
void	**emberAfEzspPollCompleteHandler** (EmberStatus status)

Network utility functions

EmberStatus	**emberAfFormNetwork** (EmberNetworkParameters *parameters)
EmberStatus	**emberAfJoinNetwork** (EmberNetworkParameters *parameters)
EmberStatus	**emberAfFindUnusedPanIdAndForm** (void)
EmberStatus	**emberAfStartSearchForJoinableNetwork** (void)

ZCL macros

#define	**ZCL_FRAME_CONTROL_FRAME_TYPE_MASK**
#define	**ZCL_CLUSTER_SPECIFIC_COMMAND**
#define	**ZCL_PROFILE_WIDE_COMMAND**
#define	**ZCL_MANUFACTURER_SPECIFIC_MASK**
#define	**ZCL_FRAME_CONTROL_DIRECTION_MASK**
#define	**ZCL_FRAME_CONTROL_SERVER_TO_CLIENT**
#define	**ZCL_FRAME_CONTROL_CLIENT_TO_SERVER**
#define	**ZCL_DISABLE_DEFAULT_RESPONSE_MASK**
#define	**EMBER_AF_ZCL_OVERHEAD**
#define	**EMBER_AF_ZCL_MANUFACTURER_SPECIFIC_OVERHEAD**

Detailed Description

This documentation describes the application programming interface (API) for the Ember Application Framework V2. The file **af.h** is the master include file for the Application Framework V2 modules.

Define Documentation

#define CONFIGURATION_HEADER

Definition at line **24** of file **af.h**.

#define emberAfClusterIsManufacturerSpecific (cluster)

macro that returns true if the cluster is in the manufacturer specific range

Parameters:

cluster EmberAfCluster* to consider

Definition at line **378** of file **af.h**.

#define emberAfAttributeIsReadOnly (metadata)

macro that returns true if attribute is read only.

Parameters:

metadata EmberAfAttributeMetadata* to consider.

Definition at line **386** of file **af.h**.

#define emberAfAttributeIsClient (metadata)

macro that returns true if this is a client attribute, and false if it is server

Parameters:

metadata EmberAfAttributeMetadata* to consider.

Definition at line **393** of file **af.h**.

#define emberAfAttributeIsTokenized (metadata)

macro that returns true if attribute is saved to token.

Parameters:

metadata EmberAfAttributeMetadata* to consider.

Definition at line **400** of file **af.h**.

#define emberAfAttributeIsSingleton (metadata)

macro that returns true if attribute is a singleton

Parameters:

metadata EmberAfAttributeMetadata* to consider.

Definition at line **407** of file **af.h**.

#define emberAfAttributeIsManufacturerSpecific (metadata)

macro that returns true if attribute is manufacturer specific

Parameters:

metadata EmberAfAttributeMetadata* to consider.

Definition at line **414** of file **af.h**.

#define emberAfAttributeSize (metadata)

macro that returns size of attribute in bytes.

Parameters:

metadata EmberAfAttributeMetadata* to consider.

Definition at line **422** of file **af.h**.

#define emberAfEndpointFromIndex (index)

Macro that takes index of endpoint, and returns Zigbee endpoint.

Definition at line **430** of file **af.h**.

#define emberAfProfileIdFromIndex (index)

Macro that takes index of endpoint, and returns profile Id for it.

Definition at line **440** of file **af.h**.

#define emberAfDeviceIdFromIndex (index)

Macro that takes index of endpoint, and returns device Id for it.

Definition at line **445** of file **af.h**.

#define emberAfDeviceVersionFromIndex (index)

Macro that takes index of endpoint, and returns device version for it.

Definition at line **450** of file **af.h**.

#define emberAfPrimaryProfileId ( )

Macro that returns primary profile ID.

Primary profile is the profile of a primary endpoint as defined in the appbuilder.

Definition at line **458** of file **af.h**.

#define emberAfPrimaryEndpoint ( )

Macro that returns the primary endpoint.

Definition at line **463** of file **af.h**.

#define emberAfGetInt8u (message,
currentIndex,
msgLen)

Macro for consistency, that extracts single byte out of the message.

Definition at line **512** of file **af.h**.

#define emberAfCopyInt8u (data,
index,
x)

Macro for consistency that copies an int8u from variable into buffer.

Definition at line **517** of file **af.h**.

#define EMBER_AF_NEW_IMAGE_VERIFICATION

This indicates a new image verification is taking place.

Definition at line **575** of file **af.h**.

#define EMBER_AF_CONTINUE_IMAGE_VERIFY

This indicates the continuation of an image verification already in progress.

Definition at line **581** of file **af.h**.

#define emberAfGuaranteedPrint (...)

Print that can't be turned off.

Definition at line **643** of file **af.h**.

#define emberAfGuaranteedPrintln (...)

Println that can't be turned off.

Definition at line **648** of file **af.h**.

#define emberAfGuaranteedPrintBuffer (buffer, len, withSpace)

Buffer print that can't be turned off.

Definition at line **653** of file **af.h**.

#define emberAfGuaranteedPrintString (buffer)

String print that can't be turned off.

Definition at line **658** of file **af.h**.

#define emberAfGuaranteedFlush ( )

Buffer flush for **emberAfGuaranteedPrint()**, **emberAfGuaranteedPrintln()**, **emberAfGuaranteedPrintBuffer()**, and **emberAfGuaranteedPrintString()**.

Definition at line **664** of file **af.h**.

#define MILLISECOND_TICKS_PER_QUARTERSECOND

Macro for setting hibernate duration at run time.

Parameters:

newValue The value to which the hibernate duration should be set

Definition at line **749** of file **af.h**.

#define MILLISECOND_TICKS_PER_MINUTE

Macro for setting hibernate duration at run time.

Parameters:

newValue The value to which the hibernate duration should be set

Definition at line **750** of file **af.h**.

#define MILLISECOND_TICKS_PER_HOUR

Macro for setting hibernate duration at run time.

Parameters:

newValue The value to which the hibernate duration should be set

Definition at line **751** of file **af.h**.

#define EMBER_AF_SET_HIBERNATE_DURATION (newValue)

Macro for setting hibernate duration at run time.

Parameters:

newValue The value to which the hibernate duration should be set

Definition at line **761** of file **af.h**.

#define EMBER_AF_SET_NAP_DURATION (newValue)

Macro for setting nap duration at run time.

Parameters:

newValue The value to which the hibernate duration should be set

Definition at line **770** of file **af.h**.

#define EMBER_AF_WAKE_TIMEOUT

The amount of time in quarter seconds that a task which has an optional response can keep the device awake for. Curently defaults to 3 seconds.

Definition at line **817** of file **af.h**.

#define EMBER_AF_WAKE_TIMEOUT_MASK

A mask of all app tasks that currently can be governed by a timeout. Any task which does not necessarily have a deterministic response which can trigger an end to the wake cycle must be included in this mask so that it can be timed out.

Definition at line **828** of file **af.h**.

#define emberAfOkToNap ( )

A macro used to check if it is ok for the application to nap. It is not ok for the applicaton to nap if any processes are going on that require the device to stay awake. This macro only checks that current active events are not holding the device out of nap.

Definition at line **867** of file **af.h**.

#define emberAfCurrentAppTasks ( )

A macro used to retrieve the bitmask of all application frameowrk tasks currently in progress. This can be useful for debugging if some task is holding the device out of hibernation.

Definition at line **877** of file **af.h**.

#define EMBER_AF_CLIENT_CLUSTER_TICK

Friendly define for use in the scheduling or canceling client events with **emberAfScheduleClusterTick()** and **emberAfDeactivateClusterTick()**.

Definition at line **906** of file **af.h**.

#define EMBER_AF_SERVER_CLUSTER_TICK

Friendly define for use in the scheduling or canceling server events with **emberAfScheduleClusterTick()** and **emberAfDeactivateClusterTick()**.

Definition at line **912** of file **af.h**.

#define emberAfQSToNextEvent (maxQS)

A function used to retrieve the number of quarter seconds until the next event scheduled in the application framework's event mechanism. This function will round down and will return 0 if the next event must fire within a quarter second.

Parameters:

maxQS,the maximum number of quarter seconds until the next event.

Returns:

The number of quarter seconds until the next event or maxQS if no event is scheduled before then.

Definition at line **1014** of file **af.h**.

#define emberAfGetDefaultSleepControl ( )

Macro for setting hibernate duration at run time.

Parameters:

newValue The value to which the hibernate duration should be set

Definition at line **1054** of file **af.h**.

#define emberAfSetResponseWaiting ( )

DEPRECATED: Use emberAfSendResponse to send the response immediately or any of the emberAfSend or emberAfSendCommand functions to send delayed responses.

Definition at line **1129** of file **af.h**.

```
#define emberAfSendInterpanUnicast ( destinationShortAddress,
                                     destinationPanId,
                                     destinationAppProfileId,
                                     destinationClusterId,
                                     multicastIdOrZero,
                                     longAddressOrNull,
                                     messageLength,
                                     messageBytes              )
```

DEPRECATED: Use emberAfSendResponse to send the response immediately or any of the emberAfSend or emberAfSendCommand functions to send delayed responses.

Definition at line **1206** of file **af.h**.

```
#define emberAfSendCommandInterpanUnicast ( eui64,
                                            nodeId,
                                            panId,
                                            multicastId,
                                            profileId      )
```

DEPRECATED: Use emberAfSendResponse to send the response immediately or any of the emberAfSend or emberAfSendCommand functions to send delayed responses.

Definition at line **1269** of file **af.h**.

#define EMBER_AF_CLIENT_CLUSTER_DISCOVERY

Friendly define for use in discovering client clusters with **emberAfFindDevicesSupportingCluster()** or **emberAfFindDevicesByProfileAndCluster()**.

Definition at line **1320** of file **af.h**.

#define EMBER_AF_SERVER_CLUSTER_DISCOVERY

Friendly define for use in discovering server clusters with **emberAfFindDevicesSupportingCluster()** or **emberAfFindDevicesByProfileAndCluster()**.

Definition at line **1327** of file **af.h**.

#define emberAfCurrentCommand ( )

Use this macro to retrive the current command. This macro may only be used within the command parsing context. For instance Any of the command handling callbacks may use this macro. If this macro is used outside the command context, the returned **EmberAfClusterCommand** pointer will be null.

Definition at line **1462** of file **af.h**.

#define emberAfCurrentEndpoint ( )

returns the current endpoint that is being served.

The purpose of this macro is mostly to access endpoint that is being served in the command callbacks.

Definition at line **1471** of file **af.h**.

#define ZCL_FRAME_CONTROL_FRAME_TYPE_MASK

Definition at line **1533** of file **af.h**.

#define ZCL_CLUSTER_SPECIFIC_COMMAND

Definition at line **1534** of file **af.h**.

#define ZCL_PROFILE_WIDE_COMMAND

Definition at line **1535** of file **af.h**.

#define ZCL_MANUFACTURER_SPECIFIC_MASK

Definition at line **1537** of file **af.h**.

#define ZCL_FRAME_CONTROL_DIRECTION_MASK

Definition at line **1539** of file **af.h**.

#define ZCL_FRAME_CONTROL_SERVER_TO_CLIENT

Definition at line **1540** of file **af.h**.

#define ZCL_FRAME_CONTROL_CLIENT_TO_SERVER

Definition at line **1541** of file **af.h**.

#define ZCL_DISABLE_DEFAULT_RESPONSE_MASK

Definition at line **1543** of file **af.h**.

#define EMBER_AF_ZCL_OVERHEAD

Definition at line **1550** of file **af.h**.

#define EMBER_AF_ZCL_MANUFACTURER_SPECIFIC_OVERHEAD

Definition at line **1551** of file **af.h**.

Enumeration Type Documentation

anonymous enum

Data types are either analog or discrete. This makes a difference for some of the ZCL global commands

Enumerator:

EMBER_AF_DATA_TYPE_ANALOG
EMBER_AF_DATA_TYPE_DISCRETE
EMBER_AF_DATA_TYPE_NONE

Definition at line **475** of file **af.h**.

anonymous enum

An enum used to track the tasks that the Application framework cares about. These are inteneded to be tasks that should keep the device out of hibernation like an application level request / response. If the response does not come in as a data ack, then the application will need to stay out of hibernation to wait and poll for it.

Of coures some tasks do not necessarily have a response. For instance, a ZDO request may or may not have a response. In this case, the application framework cannot rely on the fact that a response will come in to end the wake cycle, so the Application framework must timeout the wake cycle if no expected response is received or no other event can be relied upon to end the wake cycle.

Tasks of this type should be added to the EMBER_AF_WAKE_TIMEOUT_MASK so that they can be governed by a timeout instead of a request / response

the register emAfCurrentAppTasks is an int32u bitmask used to track which tasks are active at any given time. The bottom 16 bits, values 0x01 - 0x8000 are reserved for Ember's use. The top 16 bits are reserved for the customer, values 0x8000 - 0x80000000

Enumerator:

EMBER_AF_WAITING_FOR_DATA_ACK
EMBER_AF_LAST_POLL_GOT_DATA
EMBER_AF_WAITING_FOR_SERVICE_DISCOVERY
EMBER_AF_WAITING_FOR_ZDO_RESPONSE
EMBER_AF_WAITING_FOR_ZCL_RESPONSE
EMBER_AF_WAITING_FOR_REGISTRATION
EMBER_AF_WAITING_FOR_PARTNER_LINK_KEY_EXCHANGE

Definition at line **799** of file **af.h**.

Function Documentation

```
EmberAfAttributeMetadata* emberAfLocateAttributeMetadata ( int8u              endpoint,
                                                           EmberAfClusterId   cluster,
                                                           EmberAfAttributeId attribute,
                                                           int8u              mask,
                                                           int16u             manufacturerCode
                                                         )
```

locate attribute metadata

Function returns pointer to the attribute metadata structure, or NULL if attribute was not found.

Parameters:

endpoint Zigbee endpoint number.
cluster Cluster ID of the sought cluster.
attribute Attribute ID of the sought attribute.
mask CLUSTER_MASK_SERVER or CLUSTER_MASK_CLIENT

Returns:
Returns pointer to the attribute metadata location.

```
boolean emberAfContainsCluster ( int8u            endpoint,
                                 EmberAfClusterId clusterId
                               )
```

Returns TRUE If endpoint contains cluster.

This function returns true regardless of whether the endpoint contains server, client or both.

```
boolean emberAfContainsServer ( int8u            endpoint,
                                EmberAfClusterId clusterId
                              )
```

Returns TRUE If endpoint contains cluster server.

This function returns true if the endpoint contains server of a given cluster.

```
boolean emberAfContainsClient ( int8u            endpoint,
                                EmberAfClusterId clusterId
                              )
```

Returns TRUE If endpoint contains cluster client.

This function returns true if the endpoint contains client of a given cluster.

```
boolean emberAfReadOrUpdateAttribute ( EmberAfAttributeSearchRecord * attRecord,
                                       EmberAfAttributeMetadata **    metadata,
                                       int8u *                        buffer,
                                       boolean                        write
                                     )
```

Main function for manipulating attributes.

This function locates the attribute and returns true upon succes and false if attribute was not found. if metadata is non-NULL, then it is populated. if buffer is non-NULL, then it is populated with attribute data.

Note, that this function DOES NOT perform any read-only or type checking. This is just a low-level interface to the attribute table, and it is up to the called to ensure permissions and type checking.

Use other attribute functions for higher-level access to attribute table.

Parameters:
attRecord* Pointer to an instance of **EmberAfAttributeSearchRecord which contains everything needed to find the unique attribute requested in memory.
metadata Location where the attribute metadata pointer will be copied into. It can be NULL, in which case metadata pointer is not retrieved.
buffer Location where attribute will be written into or read from, depending if this is read or write operation. If it is NULL, then it will not be used, and data can't be read or written.
write If it is TRUE, then this is write operation, otherwise it is a read operation.

Returns:
TRUE if operation was succesfull, FALSE otherwise.

```
EmberAfStatus emberAfWriteAttribute ( int8u              endpoint,
                                      EmberAfClusterId   cluster,
                                      EmberAfAttributeId attributeID,
                                      int8u              mask,
                                      int8u *            dataPtr,
                                      int8u              dataType
                                    )
```

write an attribute, performing all the checks.

This function will attempt to write the attribute value from the provided pointer. This function will only check that the attribute exists. If it does it will write the value into the attribute table for the given attribute.

This funciton will not check to see if the attribute is writable since the read only / writable characteristic of an attribute only pertains to external devices writing over the air. Because this function is being called locally it assumes that the device knows what it is doing and has permission to perform the given operation.

See also:
emberAfWriteClientAttribute,
emberAfWriteServerAttribute,
emberAfWriteManufacturerSpecificClientAttribute, emberAfWriteManufacturerSpecificServerAttribute

```
EmberAfStatus emberAfWriteServerAttribute ( int8u              endpoint,
                                            EmberAfClusterId   cluster,
                                            EmberAfAttributeId attributeID,
                                            int8u *            dataPtr,
                                            int8u              dataType
                                          )
```

write a cluster server attribute.

This function is the same as emberAfWriteAttribute except that it saves having to pass the cluster mask. this is useful for code savings since write attribute is used frequently throughout the framework

See also:
emberAfWriteClientAttribute,
emberAfWriteManufacturerSpecificClientAttribute, emberAfWriteManufacturerSpecificServerAttribute

```
EmberAfStatus emberAfWriteClientAttribute ( int8u              endpoint,
                                            EmberAfClusterId   cluster,
                                            EmberAfAttributeId attributeID,
                                            int8u *            dataPtr,
                                            int8u              dataType
                                          )
```

write an cluster client attribute.

This function is the same as emberAfWriteAttribute except that it saves having to pass the cluster mask. this is useful for code savings since write attribute is used frequently throughout the framework

See also:
emberAfWriteServerAttribute,
emberAfWriteManufacturerSpecificClientAttribute, emberAfWriteManufacturerSpecificServerAttribute

```
EmberAfStatus emberAfWriteManufacturerSpecificServerAttribute ( int8u              endpoint,
                                                                EmberAfClusterId   cluster,
                                                                EmberAfAttributeId attributeID,
                                                                int16u             manufacturerCode,
                                                                int8u *            dataPtr,
                                                                int8u              dataType
                                                              )
```

write an manufacturer specific server attribute.

This function is the same as emberAfWriteAttribute except that it saves having to pass the cluster mask. this is useful for code savings since write attribute is used frequently throughout the framework

See also:
emberAfWriteClientAttribute, emberAfWriteServerAttribute, emberAfWriteManufacturerSpecificClientAttribute

```
EmberAfStatus emberAfWriteManufacturerSpecificClientAttribute ( int8u              endpoint,
                                                                EmberAfClusterId   cluster,
                                                                EmberAfAttributeId attributeID,
                                                                int16u             manufacturerCode,
                                                                int8u *            dataPtr,
                                                                int8u              dataType
                                                              )
```

write an manufacturer specific client attribute.

This function is the same as emberAfWriteAttribute except that it saves having to pass the cluster mask. this is useful for code savings since write attribute is used frequently throughout the framework

See also:
emberAfWriteClientAttribute, emberAfWriteServerAttribute, emberAfWriteManufacturerSpecificServerAttribute

```
EmberAfStatus emberAfVerifyAttributeWrite ( int8u              endpoint,
                                            EmberAfClusterId   cluster,
                                            EmberAfAttributeId attributeID,
                                            int8u              mask,
                                            int16u             manufacturerCode,
                                            int8u *            buffer,
                                            int8u              dataType
                                          )
```

Function that test the success of attribute write.

This function returns success if attribute write would be succesfull. It does not actuall write anything, just validates for read-only and data-type.

Parameters:

endpoint	Zigbee endpoint number
cluster	Cluster ID of the sought cluster.
attribute	Attribute ID of the sought attribute.
mask	CLUSTER_MASK_SERVER or CLUSTER_MASK_CLIENT
buffer	Location where attribute will be written from.

dataType ZCL attribute type.

```
EmberAfStatus emberAfReadAttribute ( int8u              endpoint,
                                     EmberAfClusterId   cluster,
                                     EmberAfAttributeId attributeID,
                                     int8u              mask,
                                     int8u *            dataPtr,
                                     int8u              maxLength,
                                     int8u *            dataType
                                   )
```

Read the attribute value, performing all the checks.

This function will attempt to read the attribute and store it into the pointer. It will also read the data type.
Both dataPtr and dataType may be NULL, signifying that either value or type is not desired.

See also:
emberAfReadClientAttribute,
emberAfReadServerAttribute,
emberAfReadManufacturerSpecificClientAttribute, emberAfReadManufacturerSpecificServerAttribute

```
EmberAfStatus emberAfReadServerAttribute ( int8u              endpoint,
                                           EmberAfClusterId   cluster,
                                           EmberAfAttributeId attributeID,
                                           int8u *            dataPtr,
                                           int8u              maxLength
                                         )
```

Read the server attribute value, performing all the checks.

This function will attempt to read the attribute and store it into the pointer. It will also read the data type.
Both dataPtr and dataType may be NULL, signifying that either value or type is not desired.

See also:
emberAfReadClientAttribute,
emberAfReadManufacturerSpecificClientAttribute, emberAfReadManufacturerSpecificServerAttribute

```
EmberAfStatus emberAfReadClientAttribute ( int8u              endpoint,
                                           EmberAfClusterId   cluster,
                                           EmberAfAttributeId attributeID,
                                           int8u *            dataPtr,
                                           int8u              maxLength
                                         )
```

Read the attribute value, performing all the checks.

This function will attempt to read the attribute and store it into the pointer. It will also read the data type.
Both dataPtr and dataType may be NULL, signifying that either value or type is not desired.

See also:
emberAfReadServerAttribute,
emberAfReadManufacturerSpecificClientAttribute, emberAfReadManufacturerSpecificServerAttribute

```
EmberAfStatus emberAfReadManufacturerSpecificServerAttribute ( int8u              endpoint,
                                                               EmberAfClusterId   cluster,
```

```
                                                EmberAfAttributeId attributeID,
                                                int16u             manufacturerCode,
                                                int8u *            dataPtr,
                                                int8u              maxLength
                                              )
```

Read the server attribute value, performing all the checks.

This function will attempt to read the attribute and store it into the pointer. It will also read the data type.
Both dataPtr and dataType may be NULL, signifying that either value or type is not desired.

See also:
emberAfReadClientAttribute,
emberAfReadServerAttribute, emberAfReadManufacturerSpecificClientAttribute

```
EmberAfStatus emberAfReadManufacturerSpecificClientAttribute ( int8u              endpoint,
                                                               EmberAfClusterId   cluster,
                                                               EmberAfAttributeId attributeID,
                                                               int16u             manufacturerCode,
                                                               int8u *            dataPtr,
                                                               int8u              maxLength
                                                             )
```

Read the attribute value, performing all the checks.

This function will attempt to read the attribute and store it into the pointer. It will also read the data type.
Both dataPtr and dataType may be NULL, signifying that either value or type is not desired.

See also:
emberAfReadClientAttribute,
emberAfReadServerAttribute, emberAfReadManufacturerSpecificServerAttribute

int8u emberAfGetDataSize (int8u dataType)

this function returns the size of the ZCL data in bytes.

Parameters:
dataType Zcl data type

Returns:
size in bytes or 0 if invalid data type

int8u emberAfIndexFromEndpoint (int8u endpoint)

Returns the index of a given endpoint

int8u emberAfEndpointCount (void)

Returns the number of endpoints.

int8u emberAfGetAttributeAnalogOrDiscreteType (int8u dataType)

Returns the type of the attribute, either ANALOG, DISCRETE or NONE.

```
boolean emberAfIsTypeSigned ( int8u dataType )
```

Returns true if type is signed, false otherwise.

```
int32u emberAfGetInt32u ( const int8u * message,
                          int16u        currentIndex,
                          int16u        msgLen
                        )
```

Function that extracts a 32-bit integer from the message buffer.

```
int32u emberAfGetInt24u ( const int8u * message,
                          int16u        currentIndex,
                          int16u        msgLen
                        )
```

Function that extracts a 24-bit integer from the message buffer.

```
int16u emberAfGetInt16u ( const int8u * message,
                          int16u        currentIndex,
                          int16u        msgLen
                        )
```

Function that extracts a 16-bit integer from the message buffer.

```
int8u* emberAfGetString ( int8u * message,
                          int16u  currentIndex,
                          int16u  msgLen
                        )
```

Function that extracts a ZCL string from the message buffer.

```
void emberAfCopyInt16u ( int8u * data,
                         int16u  index,
                         int16u  x
                       )
```

function that copies an in16u value into a buffer

```
void emberAfCopyInt24u ( int8u * data,
                         int16u  index,
                         int32u  x
                       )
```

function that copies an in24u value into a buffer

```
void emberAfCopyInt32u ( int8u * data,
                         int16u  index,
                         int32u  x
                       )
```

function that copies an in32u value into a buffer

void emberAfCopyString (int8u * dest, int8u * src, int8u size)

macro that returns true if the cluster is in the manufacturer specific range

Parameters:

cluster EmberAfCluster* to consider

int8u emberAfStringLength (const int8u * buffer)

macro that returns true if the cluster is in the manufacturer specific range

Parameters:

cluster EmberAfCluster* to consider

boolean emberAfIsDeviceEnabled (int8u endpoint)

Function that checks if endpoint is enabled.

This function returns true if device at a given endpoint is enabled. At startup all endpoints are enabled.

Parameters:

endpoint Zigbee endpoint number

void emberAfSetDeviceEnabled (int8u endpoint, boolean enabled)

Function that enables or disables an endpoint.

By calling this function, you turn off all processing of incoming traffic for a given endpoint.

Parameters:

endpoint Zigbee endpoint number

boolean emberAfIsThisDataTypeAStringType (int8u dataType)

Returns true if a given ZCL data type is a string type.

You should use this function if you need to perform a different memory operation on a certain attribute because it is a string type. Since ZCL strings carry length as the first byte, it is often required to treat them differently than regular data types.

Returns:

true if data type is a string.

int8u emberAfNextSequence (void)

Returns a next sequence number for the outgoing ZCL message.

Whenever sending ZCL messages you need a sequence. This function gives one.

Returns:
Next usable sequence number.

```
int8s emberAfCompareValues ( int8u *  val1,
                             int8u *  val2,
                             int8u    len
                           )
```

Simple integer comparison function. Compares two values of a known length as integers. The integers are in native endianess.

Returns:
-1 if val1 is smaller, 0 if they are the same or 1 if val2 is smaller.

```
void emberAfGetEui64 ( EmberEUI64  returnEui64 )
```

populates the eui64 with the local eui64.

```
EmberNodeId emberAfGetNodeId ( void )
```

Returns the node ID of the local node.

```
boolean emberAfPrintEnabled ( int16u  functionality )
```

returns true if certain debug area is enabled.

```
void emberAfPrintBuffer ( int16u        area,
                          const int8u *  buffer,
                          int16u        bufferLen,
                          boolean       withSpaces
                        )
```

Useful function to print a buffer.

```
void emberAfPrintString ( int16u        area,
                          const int8u *  buffer
                        )
```

Useful function to print characters strings. The first byte of the buffer specifies the length of the string.

```
void emberAfPrint ( int16u  functionality,
                    PGM_P   formatString,
                            ...
                  )
```

prints the specified text if certain debug are is enabled

Parameters:

functionality,: one of the EMBER_AF_PRINT_xxx macros as defined by appbuilder

formatString,: formatString for varargs

```
void emberAfPrintln ( int16u  functionality,
                      PGM_P  formatString,
                             ...
                     )
```

prints the specified text if certain debug are is enabled. Print-out will include the newline character at the end.

Parameters:

functionality,: one of the EMBER_AF_PRINT_xxx macros as defined by appbuilder

formatString,: formatString for varargs

```
void emberAfFlush ( int16u  functionality )
```

buffer flush

```
void emberAfPrintOn ( int16u  functionality )
```

turns on debugging for certain functional area

```
void emberAfPrintOff ( int16u  functionality )
```

turns off debugging for certain functional area

```
void emberAfPrintAllOn ( void )
```

turns on debugging for all functional areas

```
void emberAfPrintAllOff ( void )
```

turns off debugging for all functional areas

```
void emberAfPrintStatus ( void )
```

prints current status of functional areas

```
void emberAfPrintLittleEndianEui64 ( const EmberEUI64  eui64 )
```

prints eui64 stored in little endian format

```
void emberAfPrintBigEndianEui64 ( const EmberEUI64  eui64 )
```

prints eui64 stored in big endian format

```
void emberAfPrintMessageData ( int8u *  data,
                               int16u  length
                              )
```

prints all message data in message format

```
void emberAfAddToCurrentAppTasks ( int32u  mask )
```

A function used to add a task to the task register.

void emberAfRemoveFromCurrentAppTasks (int32u mask)

A function used to remove a task from the task register.

boolean emberAfOkToHibernate (void)

A function used to check if it is ok to hibernate. It is not ok to hibernate if any app tasks are currently in progress, so this macro just checks that there are no current app tasks registered. If a task is in progress but has timedout it will be removed before a determination concerning hibernation is made. This function also checks the current active events to see if any are holding the device out of hibernation.

void emberAfRunEvents (void)

a function used to run the applcation framework's event mechanism. This function passes the application framework's event tables to the ember stack's event processing code.

boolean emberAfStartMove (void)

This function schedules an event to call emberFindAndRejoinNetwork. the first call to emberFindAndRejoinNetwork will happen on the current channel with security enabled. After that, the event will be rescheduled every 10 seconds across all 15.4 channels. This event will be rescheduled until EMBER_AF_REJOIN_ATTEMPTS_MAX is reached or a suitable network is joined.
If EMBER_AF_REJOIN_ATTEMPTS_MAX is set to 0xff, the event will be rescheduled indefinitely until a suitable network is found and joined.

Returns:
Returns true if the device is able to start a move, false if a a move event is already scheduled.

```
EmberStatus emberAfScheduleClusterTick ( int8u                    endpoint,
                                         int16u                   clusterId,
                                         boolean                  isClient,
                                         int32u                   timeMs,
                                         EmberAfEventSleepControl sleepControl
                                       )
```

This function is used to schedule a cluster related event inside the application framework's event mechanism. This function provides a wrapper for the ember stack event mechanism which allows the cluster code to access its events by their endpoint, cluster id and client/server identity. The passed sleepControl value provides the same functionality as adding an event to the current app tasks. By passing this value, the cluster is able to indicate the priority of the event that is scheduled and what type of sleeping the processor should be able to do while the event being scheduled is active. The sleepControl value for each active event is checked inside the emberAfOkToHibernate and emberAfOkToNap functions.

Parameters:

endpoint	the endpoint of the event to be scheduled
clusterId	the clusterId of the event to be scheduled
isClient	TRUE if the event to be scheduled is associated with a client cluster, FALSE otherwise.
time	the number of milliseconds till the event should be called.
sleepControl	the priority of the event, what the processor should be allowed to do in terms of sleeping while the event is active.

Returns:
If the event in question was found in the event table and scheduled, this function will return EMBER_SUCCESS,
If the event could not be found it will return EMBER_BAD_ARGUMENT

```
EmberStatus emberAfDeactivateClusterTick ( int8u    endpoint,
```

```
                                    int16u    clusterId,
                                    boolean   isClient
                                  )
```

A function used to deactivate a cluster related event. This function provides a wrapper for the ember stack's event mechanism which allows an event to be accessed by its endpoint, cluster id and client/server identity.

Parameters:

endpoint the endpoint of the event to be deactivated.
clusterId the clusterId of the event to be deactivated.
isClient TRUE if the event to be deactivate is a client cluster, FALSE otherwise.

Returns:

If the event in question was found in the event table and scheduled, this function will return EMBER_SUCCESS, If the event could not be found it will return EMBER_BAD_ARGUMENT

```
EmberStatus emberAfEventControlSetDelay ( EmberEventControl *  eventControl,
                                          int32u               timeMs
                                        )
```

A convenience function used to schedule an event in the Ember Event Control API depending on how large the passed timeMs value is. The Ember Event Control API uses three functions for the scheduling of events which indicate the time increment in milliseconds, quarter seconds, or seconds. This function wraps that API and provides a single function for the scheduling of all events based on a passed int32u value indicating the number of milliseconds until the next event. This function is used by both the cluster scheduling as well as the application framework's own internal event scheduling.

Parameters:

eventControl A pointer to the ember event control value which is referred to in the event table.
timeMs the number of milliseconds until the next event.

Returns:

If the passed milliseconds can be transformed into a proper event control time increment: milliseconds, quarter seconds or minutes, that can be represented in a int16u, this function will schedule the event and return EMBER_SUCCESS. Otherwise it will return EMBER_BAD_ARGUMENT.

```
int32u emberAfMsToNextEvent ( int32u  maxMs )
```

A function used to retrieve the number of milliseconds until the next event scheduled in the application framework's event mechanism.

Parameters:

maxMs,the maximum number of milliseconds until the next event.

Returns:

The number of milliseconds until the next event or maxMs if no event is scheduled before then.

```
EmberAfEventSleepControl emberAfGetCurrentSleepControl ( void )
```

A function for retrieving the most restrictive sleep control value for all scheduled events. This function is used by emberAfOkToNap and emberAfOkToHibernate to makes sure that there are no events scheduled which will keep the device from hibernating or napping.

Returns:

The most restrictive sleep control value for all scheduled events or the value returned by **emberAfGetDefaultSleepControl()** if no events are currently scheduled. The default sleep control value is initialized to EMBER_AF_OK_TO_HIBERNATE but can be changed at any time using

the **emberAfSetDefaultSleepControl()** function.

void emberAfSetDefaultSleepControl (EmberAfEventSleepControl control)

A function for setting the default sleep control value against which all scheduled event sleep control values will be evaluated. This can be used to keep a device awake for an extended period of time by setting the default to EMBER_AF_STAY_AWAKE and then resetting the value to EMBER_AF_OK_TO_HIBERNATE once the wake period is complete.

int32u emberAfGetCurrentTime (void)

Retrieves current time.

Convienience function for retrieving the current time. If the time server cluster is implemented, then the time is retrieved from that cluster's time attribute. Otherwise, the emberAfGetCurrentTimeCallback is called. In the case of the stub, the current hal time is returned.

A real time is expected to in the ZigBee time format, the number of seconds since the year 2000.

void emberAfSetTime (int32u utcTime)

Sets current time. Convenience function for setting the time to a value. If the time server cluster is implemented on this device, then this call will be passed along to the time cluster server which will update the time. Otherwise the emberAfSetTimeCallback is called, which in the case of the stub does nothing.

Parameters:

utcTime,: A ZigBee time, the number of seconds since the year 2000.

void emberAfPrintTime (int32u utcTime)

Prints time.

Convenience function for all clusters to print time. This function expects to be passed a ZigBee time which is the number of seconds since the year 2000. If EMBER_AF_PRINT_CORE is defined, this function will print a human readable time from the passed value. If not, this function will print nothing.

Parameters:

utcTime,: A ZigBee time, the number of seconds since the year 2000.

EmberStatus emberAfSendResponse (void)

This function sends a ZCL response, based on the information that is currently in the outgoing buffer. It is expected that a complete ZCL message is present, including header. The application may use this method directly from within the message handling function and associated callbacks. However this will result in the response being sent before the APS Ack is sent which is not ideal.

NOTE: This will overwrite the ZCL sequence number of the message to use the LAST received sequence number.

EmberStatus emberAfSendRequest (void)

DEPRECATED: Use emberAfSendCommandUnicast instead to unicast the message to a specific node or binding or address table index. If using the emberAfFillCommand macros to construct the outgoing message, the sequence number will be set automatically. Otherwise, emberAfNextSequence can be used to retrieve a new sequence number for the message.

This function sends a ZCL request, based on the information that is currently in the outgoing buffer. It

is expected that a complete ZCL mesasge is present, including header.

NOTE: This will overwrite the ZCL sequence number of the message to use the NEXT sequence number of the local device.

```
EmberStatus emberAfSendMulticast ( EmberMulticastId  multicastId,
                                   EmberApsFrame *   apsFrame,
                                   int16u            messageLength,
                                   int8u *           message
                                 )
```

Sends multicast.

```
EmberStatus emberAfSendBroadcast ( EmberNodeId       destination,
                                   EmberApsFrame *   apsFrame,
                                   int16u            messageLength,
                                   int8u *           message
                                 )
```

Sends broadcast.

```
EmberStatus emberAfSendUnicast ( EmberOutgoingMessageType  type,
                                 int16u                    indexOrDestination,
                                 EmberApsFrame *           apsFrame,
                                 int16u                    messageLength,
                                 int8u *                   message
                               )
```

Sends unicast.

```
EmberStatus emberAfSendInterPan ( EmberPanId          panId,
                                  const EmberEUI64    destinationLongId,
                                  EmberNodeId         destinationShortId,
                                  EmberMulticastId    multicastId,
                                  EmberAfClusterId    clusterId,
                                  EmberAfProfileId    profileId,
                                  int16u              messageLength,
                                  int8u *             messageBytes
                                )
```

Sends interpan message.

```
EmberStatus emberAfSendEndDeviceBind ( int8u  endpoint )
```

Sends end device binding request.

```
EmberStatus emberAfSendCommandMulticast ( EmberMulticastId  multicastId )
```

Sends the command prepared with emberAfFill.... macro.

This function is used to send a command that was previously prepared using the emberAfFill... macros from the client command API. It will be sent as multicast.

```
EmberStatus emberAfSendCommandUnicast ( EmberOutgoingMessageType type,
                                        int16u                   indexOrDestination
                                      )
```

Sends the command prepared with emberAfFill.... macro.

This function is used to send a command that was previously prepared using the emberAfFill... macros from the client command API. It will be sent as unicast.

```
EmberStatus emberAfSendCommandBroadcast ( EmberNodeId destination )
```

Sends the command prepared with emberAfFill.... macro.

This function is used to send a command that was previously prepared using the emberAfFill... macros from the client command API.

```
EmberStatus emberAfSendCommandInterPan ( EmberPanId          panId,
                                         const EmberEUI64    destinationLongId,
                                         EmberNodeId         destinationShortId,
                                         EmberMulticastId    multicastId,
                                         EmberAfProfileId    profileId
                                       )
```

Sends the command prepared with emberAfFill.... macro.

This function is used to send a command that was previously prepared using the emberAfFill... macros from the client command API. It will be sent via inter-PAN. If destinationLongId is not NULL, the message will be sent to that long address using long addressing mode; otherwise, the message will be sent to destinationShortId using short address mode. IF multicastId is not zero, the message will be sent using multicast mode.

```
EmberStatus emberAfSendDefaultResponse ( const EmberAfClusterCommand * cmd,
                                         EmberAfStatus                 status
                                       )
```

Sends a default response to a cluster command.

This function is used to prepare and send a default response to a cluster command.

Parameters:

cmd The cluster command to which to respond.

status Status code for the default response command.

Returns:

An EmberStatus value that indicates the success or failure of sending the response.

```
int8u emberAfMaximumApsPayloadLength ( EmberOutgoingMessageType type,
                                       int16u                   indexOrDestination,
                                       EmberApsFrame *          apsFrame
                                     )
```

Returns the maximum size of the payload that the Application Support sub-layer will accept for the given message type, destination, and APS frame.

The size depends on multiple factors, including the security level in use and additional information added to the message to support the various options.

Parameters:

type	The outgoing message type.
indexOrDestination	Depending on the message type, this is either the EmberNodeId of the destination, an index into the address table, an index into the binding table, the multicast identifier, or a broadcast address.
apsFrame	The APS frame for the message.

Returns:
The maximum APS payload length for the given message.

EmberApsFrame* emberAfGetCommandApsFrame (void)

Access to client API aps frame.

```
EmberStatus emberAfFindDevicesSupportingCluster ( EmberNodeId                    target,
                                                  EmberAfClusterId               clusterId,
                                                  boolean                        serverCluster,
                                                  EmberAfServiceDiscoveryCallback * callback
                                                )
```

Find devices in the network that support a given cluster. This function assumes that the application profile for the cluster being discovered is that of the primary endpoint configuration on the local device. If you have multiple endpoints with different profiles, this may not be a valid assumption. For a more flexible interface that supports specifying a profile ID, see **emberAfFindDevicesByProfileAndCluster()**. may either be a specific device, or the broadcast.

With this function a service discovery is initiated and received responses are returned by executing the callback function passed in. For unicast discoveries, the callback will be executed only once. Either the target will return a result or a timeout will occur. For broadcast discoveries, the callback may be called multiple times and after a period of time the discovery will be finished with a final call to the callback.

This API has been superseded by **emberAfFindDevicesByProfileAndCluster()** but remains as a macro for convenience and backwards compatibility. It may also be useful in cases where the calling function can't easily determine the application profile (by looking at the description of a relevant endpoint).

Parameters:

target	The destination node ID for the discovery; either a specific node's ID or EMBER_RX_ON_WHEN_IDLE_BROADCAST_ADDRESS.
clusterId	The cluster being discovered. (Assumed to be from Primary endpoint's application profile.)
serverCluster	EMBER_AF_SERVER_CLUSTER_DISCOVERY (TRUE) if discovering servers for the target cluster; EMBER_AF_CLIENT_CLUSTER_DISCOVERY (FALSE) if discovering clients for that cluster.
callback	Function pointer for the callback function triggered when a match is discovered. (For broadcast discoveries, this is called once per matching node, even if a node has multiple matching endpoints.)

```
EmberStatus emberAfFindDevicesByProfileAndCluster ( EmberNodeId                    target,
                                                    EmberAfProfileId               profileId,
                                                    EmberAfClusterId               clusterId,
                                                    boolean                        serverCluster,
                                                    EmberAfServiceDiscoveryCallback * callback
                                                  )
```

Use this function to find devices in the network with endpoints matching a given profile ID and cluster ID in their descriptors. Target may either be a specific device, or the broadcast address EMBER_RX_ON_WHEN_IDLE_BROADCAST_ADDRESS.

With this function a service discovery is initiated and received responses are returned by executing the callback function passed in. For unicast discoveries, the callback will be executed only once. Either the target will return a result or a timeout will occur. For broadcast discoveries, the callback may be called multiple times and after a period of time the discovery will be finished with a final call to the callback.

Parameters:

target	The destination node ID for the discovery; either a specific node's ID or EMBER_RX_ON_WHEN_IDLE_BROADCAST_ADDRESS.
profileId	The application profile for the cluster being discovered.
clusterId	The cluster being discovered.
serverCluster	EMBER_AF_SERVER_CLUSTER_DISCOVERY (TRUE) if discovering servers for the target cluster; EMBER_AF_CLIENT_CLUSTER_DISCOVERY (FALSE) if discovering clients for that cluster.
callback	Function pointer for the callback function triggered when a match is discovered. (For broadcast discoveries, this is called once per matching node, even if a node has multiple matching endpoints.)

```
EmberStatus emberAfFindIeeeAddress ( EmberNodeId                    shortAddress,
                                     EmberAfServiceDiscoveryCallback * callback
                                   )
```

Use this function to initiate a discovery for the IEEE address of the specified node id. This will send a unicast sent to the target node ID.

```
EmberStatus emberAfFindNodeId ( EmberEUI64                    longAddress,
                                EmberAfServiceDiscoveryCallback * callback
                              )
```

Use this function to initiate a discovery for the short ID of the specified long address. This will send a broadcast to all rx-on-when-idle devices (non-sleepies).

```
int8u emberAfAddAddressTableEntry ( EmberEUI64   longId,
                                    EmberNodeId  shortId
                                  )
```

Use this function to add an entry for a remote device to the address table.

Parameters:

longId	The EUI64 of the remote device.
shortId	The node id of the remote device or EMBER_UNKNOWN_NODE_ID if the node id is currently unknown.

Returns:

The index of the address table entry for this remove device or EMBER_NULL_ADDRESS_TABLE_INDEX if an error occurred (e.g., the address table is full).

```
EmberStatus emberAfSetAddressTableEntry ( int8u          index,
                                          EmberEUI64     longId,
                                          EmberNodeId    shortId
                                        )
```

Use this function to add an entry for a remote device to the address table at a specific location.

Parameters:

index	The index of the address table entry.
longId	The EUI64 of the remote device.
shortId	The node id of the remote device or EMBER_UNKNOWN_NODE_ID if the node id is currently unknown.

Returns:

EMBER_SUCCESS if the address table entry was successfully set, EMBER_ADDRESS_TABLE_ENTRY_IS_ACTIVE if any messages are being sent using the existing entry at that index, or EMBER_ADDRESS_TABLE_INDEX_OUT_OF_RANGE if the index is out of range.

EmberStatus emberAfInitiateKeyEstablishment (EmberNodeId nodeId, int8u endpoint)

Use this function to initiate key establishment with a remote node. **emberAfKeyEstablishmentCallback** will be called as events occur and when key establishment completes.

Parameters:

nodeId The node id of the remote device.
endpoint The endpoint on the remote device.

Returns:

EMBER_SUCCESS if key establishment was initiated successfully

EmberStatus emberAfInitiateInterPanKeyEstablishment (EmberPanId panId, const EmberEUI64 eui64)

Use this function to initiate key establishment wih a remote node on a different PAN. **emberAfInterPanKeyEstablishmentCallback** will be called as events occur and when key establishment completes.

Parameters:

panId The PAN id of the remote device.
eui64 The EUI64 of the remote device.

Returns:

EMBER_SUCCESS if key establishment was initiated successfully

boolean emberAfPerformingKeyEstablishment (void)

Use this function to tell if the device is in the process of performing key establishment.

Returns:

TRUE if key establishment is in progress.

EmberStatus emberAfInitiatePartnerLinkKeyExchange (EmberNodeId target, int8u endpoint, EmberAfPartnerLinkKeyExchangeCallback * callback)

Use this function to initiate partner link key exchange with a remote node.

Parameters:

target The node id of the remote device.
endpoint The key establishment endpoint of the remote device.
callback The callback that should be called when the partner link key exchange completes.

Returns:

EMBER_SUCCESS if the partner link key exchange was initiated successfully.

void emberAfPollCompleteHandler (EmberStatus status)

that function contains the functionality that app framework should execute during the poll complete handler. If you implement your own callback, you should call this function within.

void emberAfEzspPollCompleteHandler (EmberStatus status)

that function contains the functionality that app framework should execute during the EZSP poll complete handler. If you implement your own callback, you should call this function within.

EmberStatus emberAfFormNetwork (EmberNetworkParameters * parameters)

Use this function to form a new network using the specified network parameters.

Parameters:

parameters Specification of the new network.

Returns:

An EmberStatus value that indicates either the successful formation of the new network or the reason that the network formation failed.

EmberStatus emberAfJoinNetwork (EmberNetworkParameters * parameters)

Use this function to associate with the network using the specified network parameters.

Parameters:

parameters Specification of the network with which the node should associate.

Returns:

An EmberStatus value that indicates either that the association process began successfully or the reason for failure.

EmberStatus emberAfFindUnusedPanIdAndForm (void)

Use this function to find an unused PAN id and form a new network.

Returns:

An EmberStatus value that indicates either the process begin successfully or the reason for failure.

EmberStatus emberAfStartSearchForJoinableNetwork (void)

Use this function to find a joinable network and join it.

Returns:

An EmberStatus value that indicates either the process begin successfully or the reason for failure.

Variable Documentation

EmberAfDefinedEndpoint endpoints[]

macro that returns true if the cluster is in the manufacturer specific range

Parameters:

cluster EmberAfCluster* to consider

PGM EmberAfOtaImageId emberAfInvalidImageId

This variable defines an invalid image id. It is used to determine if a returned **EmberAfOtaImageId** is valid or not. This is done by passing the data to the function **emberAfIsOtaImageIdValid()**.

int32u emberAfHibernateDuration

Macro for setting hibernate duration at run time.

Parameters:

newValue The value to which the hibernate duration should be set

int32u emAfCurrentAppTasks

externed value used in task functions and macros which is declared inside app/framework/util/util.c

EmberAfEventSleepControl emAfDefaultSleepControl

A function used to retrive the default sleep control against which all event sleep control values are evaluated. The default sleep control value is initialized to EMBER_AF_OK_TO_HIBERNATE but can be changed by the application at any time using the **emberAfSetDefaultSleepControl()** function.

Returns:

The current default sleep control value.

boolean emberAfResponseWaitingFlag

DEPRECATED: Use emberAfSendResponse to send the response immediately or any of the emberAfSend or emberAfSendCommand functions to send delayed responses.

This macro prepares the response to be sent once the application exits the context of the incomingMessageHandler. This ensures that the message response will be sent back after the APS Ack is sent for the incoming message.

EmberAfClusterCommand* emAfCurrentCommand

DEPRECATED: Use emberAfSendResponse to send the response immediately or any of the emberAfSend or emberAfSendCommand functions to send delayed responses.

Application Framework types

[Ember Application Framework API reference]

Data Structures

union	**EmberAfDefaultAttributeValue** Type for default values. More...
struct	**EmberAfAttributeMinMaxValue** Type describing the attribute default, min and max values. More...
union	**EmberAfDefaultOrMinMaxAttributeValue** Union describing the attribute default/min/max values. More...
struct	**EmberAfAttributeMetadata** Each attribute has it's metadata stored in such struct. More...
struct	**EmberAfCluster** Struct describing cluster. More...
struct	**EmberAfAttributeSearchRecord** Struct used to find an attribute in storage. Together the elements in this search record constitute the "primary key" used to identify a unique attribute value in attribute storage. More...
struct	**EmberAfManufacturerCodeEntry**
struct	**EmberAfIncomingMessage** a struct containing the superset of values passed to both emberIncomingMessageHandler on the SOC and ezspIncomingMessageHandler on the host. More...
struct	**EmberAfInterpanHeader** Interpan header used for sending and receiving interpan messages. More...
struct	**EmberAfAllowedInterPanMessage** This structure is used define an interpan message that will be accepted by the interpan filters. More...
struct	**EmberAfClusterCommand** The EmberAFClusterCommand is a struct wrapper for all the data pertaining to a command which comes in over the air. This enables struct is used to encapsulate a command in a single place on the stack and pass a pointer to that location around during command processing. More...
struct	**EmberAfEndpointType** Endpoint type struct describes clusters that are on the endpoint. More...
struct	**EmberAfDefinedEndpoint** Struct that maps actual endpoint type, onto a specific endpoint. More...
struct	**EmberAfLoadControlEvent** Struct that describes a load control event. More...

Defines

#define **EMBER_AF_NULL_MANUFACTURER_CODE**
#define **EMBER_AF_INTER_PAN_UNICAST**
#define **EMBER_AF_INTER_PAN_BROADCAST**
#define **EMBER_AF_INTER_PAN_MULTICAST**
#define **INTER_PAN_UNICAST**
#define **INTER_PAN_BROADCAST**
#define **INTER_PAN_MULTICAST**
#define **EMBER_AF_INTERPAN_OPTION_NONE**
#define **EMBER_AF_INTERPAN_OPTION_APS_ENCRYPT**
#define **EMBER_AF_INTERPAN_OPTION_MAC_HAS_LONG_ADDRESS**
#define **InterPanHeader**
#define **EMBER_AF_INTERPAN_DIRECTION_CLIENT_TO_SERVER**
#define **EMBER_AF_INTERPAN_DIRECTION_SERVER_TO_CLIENT**
#define **EMBER_AF_INTERPAN_DIRECTION_BOTH**
#define **EMBER_AF_INTERPAN_GLOBAL_COMMAND**
#define **EMBER_AF_INTERPAN_MANUFACTURER_SPECIFIC**

Typedefs

typedef int16u **EmberAfProfileId**
typedef int16u **EmberAfAttributeId**
typedef int16u **EmberAfClusterId**
typedef int8u **EmberAfAttributeType**
typedef int8u **EmberAfClusterMask**
typedef int8u **EmberAfAttributeMask**
typedef void(* **EmberAfGenericClusterFunction**)(void)
typedef int8u **EmberAfInterpanMessageType**
typedef int16u **EmberAfInterpanOptions**
typedef int8u **EmberAfAllowedInterpanOptions**

Detailed Description

This documentation describes the types used by the Ember Application Framework V2.

Define Documentation

#define EMBER_AF_NULL_MANUFACTURER_CODE

A distinguished manufacturer code that is used to indicate the absence of a manufacturer-specific profile, cluster, command, or attribute.

Definition at line **64** of file **af-types.h**.

#define EMBER_AF_INTER_PAN_UNICAST

Definition at line **303** of file **af-types.h**.

#define EMBER_AF_INTER_PAN_BROADCAST

Definition at line **304** of file **af-types.h**.

#define EMBER_AF_INTER_PAN_MULTICAST

Definition at line **305** of file **af-types.h**.

#define INTER_PAN_UNICAST

Definition at line **308** of file **af-types.h**.

#define INTER_PAN_BROADCAST

Definition at line **309** of file **af-types.h**.

#define INTER_PAN_MULTICAST

Definition at line **310** of file **af-types.h**.

#define EMBER_AF_INTERPAN_OPTION_NONE

Definition at line **313** of file **af-types.h**.

#define EMBER_AF_INTERPAN_OPTION_APS_ENCRYPT

Definition at line **314** of file **af-types.h**.

#define EMBER_AF_INTERPAN_OPTION_MAC_HAS_LONG_ADDRESS

Definition at line **315** of file **af-types.h**.

#define InterPanHeader

Definition at line **352** of file **af-types.h**.

#define EMBER_AF_INTERPAN_DIRECTION_CLIENT_TO_SERVER

Definition at line **359** of file **af-types.h**.

#define EMBER_AF_INTERPAN_DIRECTION_SERVER_TO_CLIENT

Definition at line **360** of file **af-types.h**.

#define EMBER_AF_INTERPAN_DIRECTION_BOTH

Definition at line **361** of file **af-types.h**.

#define EMBER_AF_INTERPAN_GLOBAL_COMMAND

Definition at line **362** of file **af-types.h**.

#define EMBER_AF_INTERPAN_MANUFACTURER_SPECIFIC

Definition at line **363** of file **af-types.h**.

Typedef Documentation

typedef int16u EmberAfProfileId

Type for refering to ZigBee application profile id.

Definition at line **25** of file **af-types.h**.

typedef int16u EmberAfAttributeId

Type for refering to ZCL attribute id.

Definition at line **30** of file **af-types.h**.

typedef int16u EmberAfClusterId

Type for refering to ZCL cluster id.

Definition at line **35** of file **af-types.h**.

typedef int8u EmberAfAttributeType

Type for refering to ZCL attribute type.

Definition at line **40** of file **af-types.h**.

typedef int8u EmberAfClusterMask

Type for the cluster mask.

Definition at line **45** of file **af-types.h**.

typedef int8u EmberAfAttributeMask

Type for the attribute mask.

Definition at line **50** of file **af-types.h**.

typedef void(* EmberAfGenericClusterFunction)(void)

Generic function type, used for either of the cluster function.

This type is used for the array of the cluster functions, and should always be cast into one of the specific functions before being called.

Definition at line **58** of file **af-types.h**.

typedef int8u EmberAfInterpanMessageType

Interpan Message type: unicast, broadcast, or multicast.

Definition at line **302** of file **af-types.h**.

typedef int16u EmberAfInterpanOptions

The options for sending/receiving interpan messages.

Definition at line **320** of file **af-types.h**.

typedef int8u EmberAfAllowedInterpanOptions

The options for what interpan messages are allowed.

Definition at line **357** of file **af-types.h**.

Application Framework callback interface

[Ember Application Framework API reference]

Functions

void	**emberAfPluginBasicResetToFactoryDefaultsCallback** (void)
boolean	**emberAfPluginDrlcEventActionCallback** (**EmberAfLoadControlEvent** *loadControlEvent, **EmberAfAmiEventStatus** eventStatus, int8u sequenceNumber)
void	**emberAfPluginMessagingClientDisplayMessageCallback** (EmberAfPluginMessagingClientMessage *message)
void	**emberAfPluginMessagingClientCancelMessageCallback** (EmberAfPluginMessagingClientMessage *message)
void	**emberAfPluginPriceClientPriceStartedCallback** (EmberAfPluginPriceClientPrice *price)
void	**emberAfPluginPriceClientPriceExpiredCallback** (EmberAfPluginPriceClientPrice *price)
void	**emberAfPluginPslCommissioningTouchLinkCompleteCallback** (const EmberPslNetwork *networkInfo, int8u deviceInformationRecordCount, const EmberPslDeviceInfoRecord *deviceInformationRecordList)
void	**emberAfPluginPslCommissioningTouchLinkFailedCallback** (void)
int8u	**emberAfPluginPslCommissioningGroupIdentifierCountCallback** (int8u endpoint)
boolean	**emberAfPluginPslCommissioningGroupIdentifierCallback** (int8u endpoint, int8u index, EmberAfPluginPslCommissioningGroupInformationRecord *record)
int8u	**emberAfPluginPslCommissioningEndpointInformationCountCallback** (int8u endpoint)
boolean	**emberAfPluginPslCommissioningEndpointInformationCallback** (int8u endpoint, int8u index, EmberAfPluginPslCommissioningEndpointInformationRecord *record)
void	**emberAfPluginPslCommissioningIdentifyCallback** (int16u duration)
int16u	**emberAfPluginSimpleMeteringClientRequestMirrorCallback** (EmberEUI64 requestingDeviceIeeeAddress)
int16u	**emberAfPluginSimpleMeteringClientRemoveMirrorCallback** (EmberEUI64 requestingDeviceIeeeAddress)
boolean	**emberAfPluginTestExampleCustomerFunctionCallback** (int8u arg1, int8u *arg2, boolean arg3)
void	**emberAfPluginTunnelingClientTunnelOpenedCallback** (int16u tunnelId)
void	**emberAfPluginTunnelingClientDataReceivedCallback** (int16u tunnelId, int8u *data, int16u dataLen)
void	**emberAfPluginTunnelingClientDataErrorCallback** (int16u tunnelId, **EmberAfTunnelingTransferDataStatus** transferDataStatus)
boolean	**emberAfPluginTunnelingServerIsProtocolSupportedCallback** (int8u protocolId, int16u manufacturerCode)
void	**emberAfPluginTunnelingServerTunnelOpenedCallback** (int16u tunnelId, int8u protocolId, int16u manufacturerCode)
void	**emberAfPluginTunnelingServerDataReceivedCallback** (int16u tunnelId, int8u *data, int16u dataLen)
void	**emberAfPluginTunnelingServerDataErrorCallback** (int16u tunnelId, **EmberAfTunnelingTransferDataStatus** transferDataStatus)
void	**emberAfPluginTunnelingServerTunnelClosedCallback** (int16u tunnelId, boolean clientInitiated)

Non-Cluster Related Callbacks

boolean	**emberAfReadAttributesResponseCallback** (**EmberAfClusterId** clusterId, int8u *buffer, int16u bufLen)
boolean	**emberAfWriteAttributesResponseCallback** (**EmberAfClusterId** clusterId, int8u *buffer, int16u bufLen)
boolean	**emberAfConfigureReportingResponseCallback** (**EmberAfClusterId** clusterId, int8u *buffer, int16u bufLen)

boolean	**emberAfReadReportingConfigurationResponseCallback** (**EmberAfClusterId** clusterId, int8u *buffer, int16u bufLen)
boolean	**emberAfReportAttributesCallback** (**EmberAfClusterId** clusterId, int8u *buffer, int16u bufLen)
boolean	**emberAfDefaultResponseCallback** (**EmberAfClusterId** clusterId, int8u commandId, **EmberAfStatus** status)
boolean	**emberAfDiscoverAttributesResponseCallback** (**EmberAfClusterId** clusterId, boolean discoveryComplete, int8u *buffer, int16u bufLen)
EmberAfAttributeWritePermission	**emberAfAllowNetworkWriteAttributeCallback** (int8u endpoint, **EmberAfClusterId** clusterId, **EmberAfAttributeId** attributeId, int8u mask, int16u manufacturerCode, int8u *value, int8u type)
void	**emberAfPreAttributeChangeCallback** (int8u endpoint, **EmberAfClusterId** clusterId, **EmberAfAttributeId** attributeId, int8u mask, int16u manufacturerCode, int8u type, int8u size, int8u *value)
void	**emberAfPostAttributeChangeCallback** (int8u endpoint, **EmberAfClusterId** clusterId, **EmberAfAttributeId** attributeId, int8u mask, int16u manufacturerCode, int8u type, int8u size, int8u *value)
boolean	**emberAfPreMessageReceivedCallback** (**EmberAfIncomingMessage** *incomingMessage)
boolean	**emberAfPreCommandReceivedCallback** (**EmberAfClusterCommand** *cmd)
boolean	**emberAfPreZDOMessageReceivedCallback** (EmberNodeId emberNodeId, EmberApsFrame *apsFrame, int8u *message, int16u length)
boolean	**emberAfMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
boolean	**emberAfStackStatusCallback** (EmberStatus status)
void	**emberAfNcpInitCallback** (void)
void	**emberAfMainInitCallback** (void)
void	**emberAfClusterInitCallback** (int8u endpoint, **EmberAfClusterId** clusterId)
boolean	**emberAfPreGoToSleepCallback** (int32u sleepDurationAttempt)
void	**emberAfPostWakeUpCallback** (int32u sleepDuration)
void	**emberAfMainTickCallback** (void)
boolean	**emberAfExternalAttributeWriteCallback** (int8u endpoint, **EmberAfClusterId** clusterId, **EmberAfAttributeMetadata** *attributeMetadata, int16u manufacturerCode, int8u *buffer)
boolean	**emberAfExternalAttributeReadCallback** (int8u endpoint, **EmberAfClusterId** clusterId, **EmberAfAttributeMetadata** *attributeMetadata, int16u manufacturerCode, int8u *buffer)
EmberAfStatus	**emberAfReportEntryConfiguredCallback** (**EmberAfReportEntry** *entry)
void	**emberAfTrustCenterJoinCallback** (EmberNodeId newNodeId, EmberEUI64 newNodeEui64, EmberNodeId parentOfNewNode, EmberDeviceUpdate status, EmberJoinDecision decision)
int32u	**emberAfGetCurrentTimeCallback** (void)
void	**emberAfSetTimeCallback** (int32u utcTime)
EmberStatus	**emberAfInitiateKeyEstablishmentCallback** (EmberNodeId nodeId, int8u endpoint)
EmberStatus	**emberAfInitiateInterPanKeyEstablishmentCallback** (EmberPanId panId, const EmberEUI64 eui64)
boolean	**emberAfKeyEstablishmentCallback** (**EmberAfKeyEstablishmentNotifyMessage** status, boolean amInitiator, EmberNodeId partnerShortId, int8u delayInSeconds)
boolean	**emberAfInterPanKeyEstablishmentCallback** (**EmberAfKeyEstablishmentNotifyMessage** status, boolean amInitiator, EmberPanId panId, const EmberEUI64 eui64, int8u delayInSeconds)
boolean	**emberAfPerformingKeyEstablishmentCallback** (void)
EmberStatus	**emberAfInitiatePartnerLinkKeyExchangeCallback** (EmberNodeId target, int8u endpoint, **EmberAfPartnerLinkKeyExchangeCallback** *callback)
EmberStatus	**emberAfPartnerLinkKeyExchangeRequestCallback** (EmberEUI64 partner)
void	**emberAfPartnerLinkKeyExchangeResponseCallback** (EmberNodeId sender, EmberZdoStatus status)
EmberStatus	**emberAfRegistrationStartCallback** (void)
void	**emberAfRegistrationAbortCallback** (void)

void	**emberAfRegistrationCallback** (boolean success)
void	**emberAfTrustCenterKeepaliveUpdateCallback** (boolean registrationComplete)
void	**emberAfTrustCenterKeepaliveAbortCallback** (void)
void	**emberAfCustomCommandLineCallback** (void)
boolean	**emberAfOtaServerIncomingMessageRawCallback** (**EmberAfClusterCommand** *message)
boolean	**emberAfOtaServerSendImageNotifyCallback** (EmberNodeId dest, int8u endpoint, int8u payloadType, int8u queryJitter, const **EmberAfOtaImageId** *id)
boolean	**emberAfOtaStorageDriverInitCallback** (void)
boolean	**emberAfOtaStorageDriverReadCallback** (int32u offset, int32u length, int8u *returnData)
boolean	**emberAfOtaStorageDriverWriteCallback** (const int8u *dataToWrite, int32u offset, int32u length)
int8u	**emberAfOtaBootloadCallback** (const **EmberAfOtaImageId** *id, int16u ncpUpgradeTagId)
EmberAfOtaStorageStatus	**emberAfOtaStorageInitCallback** (void)
void	**emberAfOtaStorageCloseCallback** (void)
int8u	**emberAfOtaStorageGetCountCallback** (void)
EmberAfOtaImageId	**emberAfOtaStorageSearchCallback** (int16u manufacturerId, int16u imageTypeId, const int16u *hardwareVersion)
int32u	**emberAfOtaStorageGetTotalImageSizeCallback** (const **EmberAfOtaImageId** *id)
EmberAfOtaImageId	**emberAfOtaStorageIteratorFirstCallback** (void)
EmberAfOtaImageId	**emberAfOtaStorageIteratorNextCallback** (void)
EmberAfOtaStorageStatus	**emberAfOtaStorageGetFullHeaderCallback** (const **EmberAfOtaImageId** *id, **EmberAfOtaHeader** *returnData)
EmberAfOtaStorageStatus	**emberAfOtaStorageReadImageDataCallback** (const **EmberAfOtaImageId** *id, int32u offset, int32u length, int8u *returnData, int32u *returnedLength)
EmberAfOtaStorageStatus	**emberAfOtaStorageClearTempDataCallback** (void)
EmberAfOtaStorageStatus	**emberAfOtaStorageWriteTempDataCallback** (int32u offset, int32u length, const int8u *data)
EmberAfOtaStorageStatus	**emberAfOtaStorageCheckTempDataCallback** (int32u *currentOffset, int32u *totalImageSize, **EmberAfOtaImageId** *newFileInfo)
EmberAfOtaStorageStatus	**emberAfOtaStorageFinishDownloadCallback** (int32u offset)
boolean	**emberAfOtaClientIncomingMessageRawCallback** (**EmberAfClusterCommand** *message)
void	**emberAfOtaClientStartCallback** (void)
void	**emberAfOtaClientVersionInfoCallback** (**EmberAfOtaImageId** *currentImageInfo, int16u *hardwareVersion)
EmberAfImageVerifyStatus	**emberAfOtaClientCustomVerifyCallback** (boolean newVerification, const **EmberAfOtaImageId** *id)
boolean	**emberAfOtaClientDownloadCompleteCallback** (boolean success, const **EmberAfOtaImageId** *id)
void	**emberAfOtaClientBootloadCallback** (const **EmberAfOtaImageId** *id)
int8u	**emberAfOtaServerQueryCallback** (const **EmberAfOtaImageId** *currentImageId, int16u *hardwareVersion, **EmberAfOtaImageId** *nextUpgradeImageId)
int8u	**emberAfOtaServerBlockSizeCallback** (EmberNodeId clientNodeId)
int8u	**emberAfOtaServerImageBlockRequestCallback** (const **EmberAfOtaImageId** *id, int32u offset, int8u maxDataSize, int32u *waitTimeMinutes)
boolean	**emberAfOtaServerUpgradeEndRequestCallback** (EmberNodeId source, int8u status, int32u *returnValue, const **EmberAfOtaImageId** *imageId)
int8u	**emberAfOtaPageRequestServerPolicyCallback** (void)
EmberStatus	**emberAfInterpanSendMessageCallback** (**EmberAfInterpanHeader** *header, int16u messageLength, int8u *message)
void	**emberAfInterpanInitCallback** (void)
void	**emberAfUnusedPanIdFoundCallback** (EmberPanId panId, int8u channel)
void	**emberAfJoinableNetworkFoundCallback** (EmberZigbeeNetwork *networkFound, int8u lqi, int8s rssi)

void	**emberAfScanErrorCallback** (EmberStatus status)
EmberStatus	**emberAfFindUnusedPanIdAndFormCallback** (void)
EmberStatus	**emberAfStartSearchForJoinableNetworkCallback** (void)
void	**emberAfContinueJoinableNetworkSearchCallback** (void)
void	**emberAfNetworkFindOperationCompleteCallback** (void)

Basic Cluster Callbacks

boolean	**emberAfBasicClusterResetToFactoryDefaultsCallback** (void)
void	**emberAfBasicClusterServerInitCallback** (int8u endpoint)
void	**emberAfBasicClusterClientInitCallback** (int8u endpoint)
void	**emberAfBasicClusterServerTickCallback** (int8u endpoint)
void	**emberAfBasicClusterClientTickCallback** (int8u endpoint)
void	**emberAfBasicClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfBasicClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfBasicClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfBasicClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfBasicClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfBasicClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfBasicClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfBasicClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Power Configuration Cluster Callbacks

void	**emberAfPowerConfigClusterServerInitCallback** (int8u endpoint)
void	**emberAfPowerConfigClusterClientInitCallback** (int8u endpoint)
void	**emberAfPowerConfigClusterServerTickCallback** (int8u endpoint)
void	**emberAfPowerConfigClusterClientTickCallback** (int8u endpoint)
void	**emberAfPowerConfigClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfPowerConfigClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfPowerConfigClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPowerConfigClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)

void	**emberAfPowerConfigClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPowerConfigClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPowerConfigClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfPowerConfigClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Device Temperature Configuration Cluster Callbacks

void	**emberAfDeviceTempClusterServerInitCallback** (int8u endpoint)
void	**emberAfDeviceTempClusterClientInitCallback** (int8u endpoint)
void	**emberAfDeviceTempClusterServerTickCallback** (int8u endpoint)
void	**emberAfDeviceTempClusterClientTickCallback** (int8u endpoint)
void	**emberAfDeviceTempClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfDeviceTempClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfDeviceTempClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfDeviceTempClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfDeviceTempClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfDeviceTempClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfDeviceTempClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfDeviceTempClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Identify Cluster Callbacks

boolean	**emberAfIdentifyClusterIdentifyCallback** (int16u identifyTime)
boolean	**emberAfIdentifyClusterIdentifyQueryCallback** (void)
boolean	**emberAfIdentifyClusterIdentifyQueryResponseCallback** (int16u timeout)
void	**emberAfIdentifyClusterServerInitCallback** (int8u endpoint)
void	**emberAfIdentifyClusterClientInitCallback** (int8u endpoint)
void	**emberAfIdentifyClusterServerTickCallback** (int8u endpoint)
void	**emberAfIdentifyClusterClientTickCallback** (int8u endpoint)
void	**emberAfIdentifyClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIdentifyClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)

void	**emberAfIdentifyClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIdentifyClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIdentifyClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIdentifyClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIdentifyClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfIdentifyClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Groups Cluster Callbacks

boolean	**emberAfGroupsClusterEndpointInGroupCallback** (int8u endpoint, int16u groupId)
void	**emberAfGroupsClusterClearGroupTableCallback** (void)
boolean	**emberAfGroupsClusterAddGroupCallback** (int16u groupId, int8u *groupName)
boolean	**emberAfGroupsClusterViewGroupCallback** (int16u groupId)
boolean	**emberAfGroupsClusterGetGroupMembershipCallback** (int8u groupCount, int8u *groupList)
boolean	**emberAfGroupsClusterRemoveGroupCallback** (int16u groupId)
boolean	**emberAfGroupsClusterRemoveAllGroupsCallback** (void)
boolean	**emberAfGroupsClusterAddGroupIfIdentifyingCallback** (int16u groupId, int8u *groupName)
boolean	**emberAfGroupsClusterAddGroupResponseCallback** (int8u status, int16u groupId)
boolean	**emberAfGroupsClusterViewGroupResponseCallback** (int8u status, int16u groupId, int8u *groupName)
boolean	**emberAfGroupsClusterGetGroupMembershipResponseCallback** (int8u capacity, int8u groupCount, int8u *groupList)
boolean	**emberAfGroupsClusterRemoveGroupResponseCallback** (int8u status, int16u groupId)
void	**emberAfGroupsClusterServerInitCallback** (int8u endpoint)
void	**emberAfGroupsClusterClientInitCallback** (int8u endpoint)
void	**emberAfGroupsClusterServerTickCallback** (int8u endpoint)
void	**emberAfGroupsClusterClientTickCallback** (int8u endpoint)
void	**emberAfGroupsClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfGroupsClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfGroupsClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfGroupsClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfGroupsClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfGroupsClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfGroupsClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

void	**emberAfGroupsClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Scenes Cluster Callbacks

EmberAfStatus	**emberAfScenesClusterStoreCurrentSceneCallback** (int8u endpoint, int16u groupId, int8u sceneId)
EmberAfStatus	**emberAfScenesClusterRecallSavedSceneCallback** (int8u endpoint, int16u groupId, int8u sceneId)
void	**emberAfScenesClusterClearSceneTableCallback** (void)
boolean	**emberAfScenesClusterAddSceneCallback** (int16u groupId, int8u sceneId, int16u transitionTime, int8u *sceneName, int8u *extensionFieldSets)
boolean	**emberAfScenesClusterViewSceneCallback** (int16u groupId, int8u sceneId)
boolean	**emberAfScenesClusterRemoveSceneCallback** (int16u groupId, int8u sceneId)
boolean	**emberAfScenesClusterRemoveAllScenesCallback** (int16u groupId)
boolean	**emberAfScenesClusterStoreSceneCallback** (int16u groupId, int8u sceneId)
boolean	**emberAfScenesClusterRecallSceneCallback** (int16u groupId, int8u sceneId)
boolean	**emberAfScenesClusterGetSceneMembershipCallback** (int16u groupId)
boolean	**emberAfScenesClusterAddSceneResponseCallback** (int8u status, int16u groupId, int8u sceneId)
boolean	**emberAfScenesClusterViewSceneResponseCallback** (int8u status, int16u groupId, int8u sceneId, int16u transitionTime, int8u *sceneName, int8u *extensionFieldSets)
boolean	**emberAfScenesClusterRemoveSceneResponseCallback** (int8u status, int16u groupId, int8u sceneId)
boolean	**emberAfScenesClusterRemoveAllScenesResponseCallback** (int8u status, int16u groupId)
boolean	**emberAfScenesClusterStoreSceneResponseCallback** (int8u status, int16u groupId, int8u sceneId)
boolean	**emberAfScenesClusterGetSceneMembershipResponseCallback** (int8u status, int8u capacity, int16u groupId, int8u sceneCount, int8u *sceneList)
void	**emberAfScenesClusterServerInitCallback** (int8u endpoint)
void	**emberAfScenesClusterClientInitCallback** (int8u endpoint)
void	**emberAfScenesClusterServerTickCallback** (int8u endpoint)
void	**emberAfScenesClusterClientTickCallback** (int8u endpoint)
void	**emberAfScenesClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfScenesClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfScenesClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfScenesClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfScenesClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfScenesClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfScenesClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfScenesClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

On/off Cluster Callbacks

boolean	**emberAfOnOffClusterOffCallback** (void)
boolean	**emberAfOnOffClusterOnCallback** (void)
boolean	**emberAfOnOffClusterToggleCallback** (void)
void	**emberAfOnOffClusterServerInitCallback** (int8u endpoint)
void	**emberAfOnOffClusterClientInitCallback** (int8u endpoint)
void	**emberAfOnOffClusterServerTickCallback** (int8u endpoint)
void	**emberAfOnOffClusterClientTickCallback** (int8u endpoint)
void	**emberAfOnOffClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOnOffClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOnOffClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOnOffClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOnOffClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOnOffClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOnOffClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfOnOffClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
boolean	**emberAfOnOffClusterSampleMfgSpecificOffWithTransitionCallback** (void)
boolean	**emberAfOnOffClusterSampleMfgSpecificOnWithTransitionCallback** (void)
boolean	**emberAfOnOffClusterSampleMfgSpecificToggleWithTransitionCallback** (void)

On/off Switch Configuration Cluster Callbacks

void	**emberAfOnOffSwitchConfigClusterServerInitCallback** (int8u endpoint)
void	**emberAfOnOffSwitchConfigClusterClientInitCallback** (int8u endpoint)
void	**emberAfOnOffSwitchConfigClusterServerTickCallback** (int8u endpoint)
void	**emberAfOnOffSwitchConfigClusterClientTickCallback** (int8u endpoint)
void	**emberAfOnOffSwitchConfigClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOnOffSwitchConfigClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOnOffSwitchConfigClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOnOffSwitchConfigClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOnOffSwitchConfigClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOnOffSwitchConfigClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)

void	**emberAfOnOffSwitchConfigClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfOnOffSwitchConfigClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Level Control Cluster Callbacks

boolean	**emberAfLevelControlClusterMoveToLevelCallback** (int8u level, int16u transitionTime)
boolean	**emberAfLevelControlClusterMoveCallback** (int8u moveMode, int8u rate)
boolean	**emberAfLevelControlClusterStepCallback** (int8u stepMode, int8u stepSize, int16u transitionTime)
boolean	**emberAfLevelControlClusterStopCallback** (void)
boolean	**emberAfLevelControlClusterMoveToLevelWithOnOffCallback** (int8u level, int16u transitionTime)
boolean	**emberAfLevelControlClusterMoveWithOnOffCallback** (int8u moveMode, int8u rate)
boolean	**emberAfLevelControlClusterStepWithOnOffCallback** (int8u stepMode, int8u stepSize, int16u transitionTime)
boolean	**emberAfLevelControlClusterStopWithOnOffCallback** (void)
void	**emberAfLevelControlClusterServerInitCallback** (int8u endpoint)
void	**emberAfLevelControlClusterClientInitCallback** (int8u endpoint)
void	**emberAfLevelControlClusterServerTickCallback** (int8u endpoint)
void	**emberAfLevelControlClusterClientTickCallback** (int8u endpoint)
void	**emberAfLevelControlClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfLevelControlClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfLevelControlClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfLevelControlClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfLevelControlClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfLevelControlClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfLevelControlClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfLevelControlClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Alarms Cluster Callbacks

boolean	**emberAfAlarmClusterResetAlarmCallback** (int8u alarmCode, int16u clusterId)
boolean	**emberAfAlarmClusterResetAllAlarmsCallback** (void)
boolean	**emberAfAlarmClusterGetAlarmCallback** (void)
boolean	**emberAfAlarmClusterResetAlarmLogCallback** (void)
boolean	**emberAfAlarmClusterAlarmCallback** (int8u alarmCode, int16u clusterId)

boolean	**emberAfAlarmClusterGetAlarmResponseCallback** (int8u status, int8u alarmCode, int16u clusterId, int32u timeStamp)
void	**emberAfAlarmClusterServerInitCallback** (int8u endpoint)
void	**emberAfAlarmClusterClientInitCallback** (int8u endpoint)
void	**emberAfAlarmClusterServerTickCallback** (int8u endpoint)
void	**emberAfAlarmClusterClientTickCallback** (int8u endpoint)
void	**emberAfAlarmClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfAlarmClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfAlarmClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfAlarmClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfAlarmClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfAlarmClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfAlarmClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfAlarmClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Time Cluster Callbacks

void	**emberAfTimeClusterServerInitCallback** (int8u endpoint)
void	**emberAfTimeClusterClientInitCallback** (int8u endpoint)
void	**emberAfTimeClusterServerTickCallback** (int8u endpoint)
void	**emberAfTimeClusterClientTickCallback** (int8u endpoint)
void	**emberAfTimeClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfTimeClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfTimeClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfTimeClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfTimeClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfTimeClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfTimeClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfTimeClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

RSSI Location Cluster Callbacks

boolean	**emberAfRssiLocationClusterSetAbsoluteLocationCallback** (int16s coordinate1, int16s coordinate2, int16s coordinate3, int16s power, int16u pathLossExponent)
boolean	**emberAfRssiLocationClusterSetDeviceConfigurationCallback** (int16s power, int16u pathLossExponent, int16u calculationPeriod, int8u numberRssiMeasurements, int16u reportingPeriod)
boolean	**emberAfRssiLocationClusterGetDeviceConfigurationCallback** (int8u *targetAddress)
boolean	**emberAfRssiLocationClusterGetLocationDataCallback** (int8u flags, int8u numberResponses, int8u *targetAddress)
boolean	**emberAfRssiLocationClusterRssiResponseCallback** (int8u *replyingDevice, int16s coordinate1, int16s coordinate2, int16s coordinate3, int8s rssi, int8u numberRssiMeasurements)
boolean	**emberAfRssiLocationClusterSendPingsCallback** (int8u *targetAddress, int8u numberRssiMeasurements, int16u calculationPeriod)
boolean	**emberAfRssiLocationClusterAnchorNodeAnnounceCallback** (int8u *anchorNodeIeeeAddress, int16s coordinate1, int16s coordinate2, int16s coordinate3)
boolean	**emberAfRssiLocationClusterDeviceConfigurationResponseCallback** (int8u status, int16s power, int16u pathLossExponent, int16u calculationPeriod, int8u numberRssiMeasurements, int16u reportingPeriod)
boolean	**emberAfRssiLocationClusterLocationDataResponseCallback** (int8u status, int8u locationType, int16s coordinate1, int16s coordinate2, int16s coordinate3, int16s power, int16u pathLossExponent, int8u locationMethod, int8u qualityMeasure, int16u locationAge)
boolean	**emberAfRssiLocationClusterLocationDataNotificationCallback** (int8u locationType, int16s coordinate1, int16s coordinate2, int16s coordinate3, int16s power, int16u pathLossExponent, int8u locationMethod, int8u qualityMeasure, int16u locationAge)
boolean	**emberAfRssiLocationClusterCompactLocationDataNotificationCallback** (int8u locationType, int16s coordinate1, int16s coordinate2, int16s coordinate3, int8u qualityMeasure, int16u locationAge)
boolean	**emberAfRssiLocationClusterRssiPingCallback** (int8u locationType)
boolean	**emberAfRssiLocationClusterRssiRequestCallback** (void)
boolean	**emberAfRssiLocationClusterReportRssiMeasurementsCallback** (int8u *measuringDevice, int8u neighbors, int8u *neighborsInfo)
boolean	**emberAfRssiLocationClusterRequestOwnLocationCallback** (int8u *blindNode)
void	**emberAfRssiLocationClusterServerInitCallback** (int8u endpoint)
void	**emberAfRssiLocationClusterClientInitCallback** (int8u endpoint)
void	**emberAfRssiLocationClusterServerTickCallback** (int8u endpoint)
void	**emberAfRssiLocationClusterClientTickCallback** (int8u endpoint)
void	**emberAfRssiLocationClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfRssiLocationClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfRssiLocationClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfRssiLocationClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfRssiLocationClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfRssiLocationClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfRssiLocationClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfRssiLocationClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Binary Input (Basic) Cluster Callbacks

void	**emberAfBinaryInputBasicClusterServerInitCallback** (int8u endpoint)
void	**emberAfBinaryInputBasicClusterClientInitCallback** (int8u endpoint)
void	**emberAfBinaryInputBasicClusterServerTickCallback** (int8u endpoint)
void	**emberAfBinaryInputBasicClusterClientTickCallback** (int8u endpoint)
void	**emberAfBinaryInputBasicClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfBinaryInputBasicClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfBinaryInputBasicClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfBinaryInputBasicClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfBinaryInputBasicClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfBinaryInputBasicClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfBinaryInputBasicClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfBinaryInputBasicClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Commissioning Cluster Callbacks

boolean	**emberAfCommissioningClusterRestartDeviceCallback** (int8u options, int8u delay, int8u jitter)
boolean	**emberAfCommissioningClusterSaveStartupParametersCallback** (int8u options, int8u index)
boolean	**emberAfCommissioningClusterRestoreStartupParametersCallback** (int8u options, int8u index)
boolean	**emberAfCommissioningClusterResetStartupParametersCallback** (int8u options, int8u index)
boolean	**emberAfCommissioningClusterRestartDeviceResponseCallback** (int8u status)
boolean	**emberAfCommissioningClusterSaveStartupParametersResponseCallback** (int8u status)
boolean	**emberAfCommissioningClusterRestoreStartupParametersResponseCallback** (int8u status)
boolean	**emberAfCommissioningClusterResetStartupParametersResponseCallback** (int8u status)
void	**emberAfCommissioningClusterServerInitCallback** (int8u endpoint)
void	**emberAfCommissioningClusterClientInitCallback** (int8u endpoint)
void	**emberAfCommissioningClusterServerTickCallback** (int8u endpoint)
void	**emberAfCommissioningClusterClientTickCallback** (int8u endpoint)
void	**emberAfCommissioningClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfCommissioningClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)

void	**emberAfCommissioningClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfCommissioningClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfCommissioningClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfCommissioningClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfCommissioningClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfCommissioningClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Shade Configuration Cluster Callbacks

void	**emberAfShadeConfigClusterServerInitCallback** (int8u endpoint)
void	**emberAfShadeConfigClusterClientInitCallback** (int8u endpoint)
void	**emberAfShadeConfigClusterServerTickCallback** (int8u endpoint)
void	**emberAfShadeConfigClusterClientTickCallback** (int8u endpoint)
void	**emberAfShadeConfigClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfShadeConfigClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfShadeConfigClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfShadeConfigClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfShadeConfigClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfShadeConfigClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfShadeConfigClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfShadeConfigClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Pump Configuration and Control Cluster Callbacks

void	**emberAfPumpConfigControlClusterServerInitCallback** (int8u endpoint)
void	**emberAfPumpConfigControlClusterClientInitCallback** (int8u endpoint)
void	**emberAfPumpConfigControlClusterServerTickCallback** (int8u endpoint)
void	**emberAfPumpConfigControlClusterClientTickCallback** (int8u endpoint)
void	**emberAfPumpConfigControlClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)

void	**emberAfPumpConfigControlClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfPumpConfigControlClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPumpConfigControlClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPumpConfigControlClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPumpConfigControlClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPumpConfigControlClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfPumpConfigControlClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Thermostat Cluster Callbacks

boolean	**emberAfThermostatClusterSetpointRaiseLowerCallback** (int8u mode, int8s amount)
void	**emberAfThermostatClusterServerInitCallback** (int8u endpoint)
void	**emberAfThermostatClusterClientInitCallback** (int8u endpoint)
void	**emberAfThermostatClusterServerTickCallback** (int8u endpoint)
void	**emberAfThermostatClusterClientTickCallback** (int8u endpoint)
void	**emberAfThermostatClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfThermostatClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfThermostatClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfThermostatClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfThermostatClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfThermostatClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfThermostatClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfThermostatClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Fan Control Cluster Callbacks

void	**emberAfFanControlClusterServerInitCallback** (int8u endpoint)
void	**emberAfFanControlClusterClientInitCallback** (int8u endpoint)
void	**emberAfFanControlClusterServerTickCallback** (int8u endpoint)

	void **emberAfFanControlClusterClientTickCallback** (int8u endpoint)
	void **emberAfFanControlClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfFanControlClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfFanControlClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfFanControlClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfFanControlClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfFanControlClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfFanControlClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
	void **emberAfFanControlClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Dehumidification Control Cluster Callbacks

	void **emberAfDehumidControlClusterServerInitCallback** (int8u endpoint)
	void **emberAfDehumidControlClusterClientInitCallback** (int8u endpoint)
	void **emberAfDehumidControlClusterServerTickCallback** (int8u endpoint)
	void **emberAfDehumidControlClusterClientTickCallback** (int8u endpoint)
	void **emberAfDehumidControlClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfDehumidControlClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfDehumidControlClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfDehumidControlClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfDehumidControlClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfDehumidControlClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfDehumidControlClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
	void **emberAfDehumidControlClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Thermostat User Interface Configuration Cluster Callbacks

	void **emberAfThermostatUiConfigClusterServerInitCallback** (int8u endpoint)

void	**emberAfThermostatUiConfigClusterClientInitCallback** (int8u endpoint)
void	**emberAfThermostatUiConfigClusterServerTickCallback** (int8u endpoint)
void	**emberAfThermostatUiConfigClusterClientTickCallback** (int8u endpoint)
void	**emberAfThermostatUiConfigClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfThermostatUiConfigClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfThermostatUiConfigClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfThermostatUiConfigClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfThermostatUiConfigClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfThermostatUiConfigClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfThermostatUiConfigClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfThermostatUiConfigClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Color Control Cluster Callbacks

boolean	**emberAfColorControlClusterMoveToHueCallback** (int8u hue, int8u direction, int16u transitionTime)
boolean	**emberAfColorControlClusterMoveHueCallback** (int8u moveMode, int8u rate)
boolean	**emberAfColorControlClusterStepHueCallback** (int8u stepMode, int8u stepSize, int8u transitionTime)
boolean	**emberAfColorControlClusterMoveToSaturationCallback** (int8u saturation, int16u transitionTime)
boolean	**emberAfColorControlClusterMoveSaturationCallback** (int8u moveMode, int8u rate)
boolean	**emberAfColorControlClusterStepSaturationCallback** (int8u stepMode, int8u stepSize, int8u transitionTime)
boolean	**emberAfColorControlClusterMoveToHueAndSaturationCallback** (int8u hue, int8u saturation, int16u transitionTime)
boolean	**emberAfColorControlClusterMoveToColorCallback** (int16u colorX, int16u colorY, int16u transitionTime)
boolean	**emberAfColorControlClusterMoveColorCallback** (int16s rateX, int16s rateY)
boolean	**emberAfColorControlClusterStepColorCallback** (int16s stepX, int16s stepY, int16u transitionTime)
boolean	**emberAfColorControlClusterMoveToColorTemperatureCallback** (int16u colorTemperature, int16u transitionTime)
void	**emberAfColorControlClusterServerInitCallback** (int8u endpoint)
void	**emberAfColorControlClusterClientInitCallback** (int8u endpoint)
void	**emberAfColorControlClusterServerTickCallback** (int8u endpoint)
void	**emberAfColorControlClusterClientTickCallback** (int8u endpoint)
void	**emberAfColorControlClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfColorControlClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfColorControlClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)

	void **emberAfColorControlClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfColorControlClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfColorControlClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfColorControlClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
	void **emberAfColorControlClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Ballast Configuration Cluster Callbacks

	void **emberAfBallastConfigurationClusterServerInitCallback** (int8u endpoint)
	void **emberAfBallastConfigurationClusterClientInitCallback** (int8u endpoint)
	void **emberAfBallastConfigurationClusterServerTickCallback** (int8u endpoint)
	void **emberAfBallastConfigurationClusterClientTickCallback** (int8u endpoint)
	void **emberAfBallastConfigurationClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfBallastConfigurationClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfBallastConfigurationClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfBallastConfigurationClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfBallastConfigurationClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfBallastConfigurationClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfBallastConfigurationClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
	void **emberAfBallastConfigurationClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Illuminance Measurement Cluster Callbacks

	void **emberAfIllumMeasurementClusterServerInitCallback** (int8u endpoint)
	void **emberAfIllumMeasurementClusterClientInitCallback** (int8u endpoint)
	void **emberAfIllumMeasurementClusterServerTickCallback** (int8u endpoint)
	void **emberAfIllumMeasurementClusterClientTickCallback** (int8u endpoint)
	void **emberAfIllumMeasurementClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)

void	**emberAfIllumMeasurementClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIllumMeasurementClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIllumMeasurementClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIllumMeasurementClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIllumMeasurementClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIllumMeasurementClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfIllumMeasurementClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Illuminance Level Sensing Cluster Callbacks

void	**emberAfIllumLevelSensingClusterServerInitCallback** (int8u endpoint)
void	**emberAfIllumLevelSensingClusterClientInitCallback** (int8u endpoint)
void	**emberAfIllumLevelSensingClusterServerTickCallback** (int8u endpoint)
void	**emberAfIllumLevelSensingClusterClientTickCallback** (int8u endpoint)
void	**emberAfIllumLevelSensingClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIllumLevelSensingClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIllumLevelSensingClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIllumLevelSensingClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIllumLevelSensingClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIllumLevelSensingClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIllumLevelSensingClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfIllumLevelSensingClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Temperature Measurement Cluster Callbacks

int32s	**emberAfTempMeasurementClusterConvertVoltsToCelsiusCallback** (int16u voltage)
void	**emberAfTempMeasurementClusterServerInitCallback** (int8u endpoint)
void	**emberAfTempMeasurementClusterClientInitCallback** (int8u endpoint)
void	**emberAfTempMeasurementClusterServerTickCallback** (int8u endpoint)

	void **emberAfTempMeasurementClusterClientTickCallback** (int8u endpoint)
	void **emberAfTempMeasurementClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfTempMeasurementClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfTempMeasurementClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfTempMeasurementClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfTempMeasurementClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfTempMeasurementClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfTempMeasurementClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
	void **emberAfTempMeasurementClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Pressure Measurement Cluster Callbacks

	void **emberAfPressureMeasurementClusterServerInitCallback** (int8u endpoint)
	void **emberAfPressureMeasurementClusterClientInitCallback** (int8u endpoint)
	void **emberAfPressureMeasurementClusterServerTickCallback** (int8u endpoint)
	void **emberAfPressureMeasurementClusterClientTickCallback** (int8u endpoint)
	void **emberAfPressureMeasurementClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfPressureMeasurementClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfPressureMeasurementClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfPressureMeasurementClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfPressureMeasurementClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfPressureMeasurementClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfPressureMeasurementClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
	void **emberAfPressureMeasurementClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Flow Measurement Cluster Callbacks

	void **emberAfFlowMeasurementClusterServerInitCallback** (int8u endpoint)
	void **emberAfFlowMeasurementClusterClientInitCallback** (int8u endpoint)
	void **emberAfFlowMeasurementClusterServerTickCallback** (int8u endpoint)
	void **emberAfFlowMeasurementClusterClientTickCallback** (int8u endpoint)
	void **emberAfFlowMeasurementClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfFlowMeasurementClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfFlowMeasurementClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfFlowMeasurementClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfFlowMeasurementClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfFlowMeasurementClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfFlowMeasurementClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
	void **emberAfFlowMeasurementClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Relative Humidity Measurement Cluster Callbacks

	void **emberAfRelativeHumidityMeasurementClusterServerInitCallback** (int8u endpoint)
	void **emberAfRelativeHumidityMeasurementClusterClientInitCallback** (int8u endpoint)
	void **emberAfRelativeHumidityMeasurementClusterServerTickCallback** (int8u endpoint)
	void **emberAfRelativeHumidityMeasurementClusterClientTickCallback** (int8u endpoint)
	void **emberAfRelativeHumidityMeasurementClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfRelativeHumidityMeasurementClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfRelativeHumidityMeasurementClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfRelativeHumidityMeasurementClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfRelativeHumidityMeasurementClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfRelativeHumidityMeasurementClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfRelativeHumidityMeasurementClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
	void **emberAfRelativeHumidityMeasurementClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Occupancy Sensing Cluster Callbacks

void	**emberAfOccupancySensingClusterServerInitCallback** (int8u endpoint)
void	**emberAfOccupancySensingClusterClientInitCallback** (int8u endpoint)
void	**emberAfOccupancySensingClusterServerTickCallback** (int8u endpoint)
void	**emberAfOccupancySensingClusterClientTickCallback** (int8u endpoint)
void	**emberAfOccupancySensingClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOccupancySensingClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOccupancySensingClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOccupancySensingClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOccupancySensingClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOccupancySensingClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOccupancySensingClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfOccupancySensingClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

IAS Zone Cluster Callbacks

boolean	**emberAfIasZoneClusterZoneEnrollResponseCallback** (int8u enrollResponseCode, int8u zoneId)
boolean	**emberAfIasZoneClusterZoneStatusChangeNotificationCallback** (int16u zoneStatus, int8u extendedStatus)
boolean	**emberAfIasZoneClusterZoneEnrollRequestCallback** (int16u zoneType, int16u manufacturerCode)
void	**emberAfIasZoneClusterServerInitCallback** (int8u endpoint)
void	**emberAfIasZoneClusterClientInitCallback** (int8u endpoint)
void	**emberAfIasZoneClusterServerTickCallback** (int8u endpoint)
void	**emberAfIasZoneClusterClientTickCallback** (int8u endpoint)
void	**emberAfIasZoneClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIasZoneClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIasZoneClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIasZoneClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIasZoneClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIasZoneClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)

void	**emberAfIasZoneClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfIasZoneClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

IAS ACE Cluster Callbacks

boolean	**emberAfIasAceClusterArmCallback** (int8u armMode)
boolean	**emberAfIasAceClusterBypassCallback** (int8u numberOfZones, int8u *zoneIds)
boolean	**emberAfIasAceClusterEmergencyCallback** (void)
boolean	**emberAfIasAceClusterFireCallback** (void)
boolean	**emberAfIasAceClusterPanicCallback** (void)
boolean	**emberAfIasAceClusterGetZoneIdMapCallback** (void)
boolean	**emberAfIasAceClusterGetZoneInformationCallback** (int8u zoneId)
boolean	**emberAfIasAceClusterArmResponseCallback** (int8u armNotification)
boolean	**emberAfIasAceClusterGetZoneIdMapResponseCallback** (int16u section0, int16u section1, int16u section2, int16u section3, int16u section4, int16u section5, int16u section6, int16u section7, int16u section8, int16u section9, int16u section10, int16u section11, int16u section12, int16u section13, int16u section14, int16u section15)
boolean	**emberAfIasAceClusterGetZoneInformationResponseCallback** (int8u zoneId, int16u zoneType, int8u *ieeeAddress)
void	**emberAfIasAceClusterServerInitCallback** (int8u endpoint)
void	**emberAfIasAceClusterClientInitCallback** (int8u endpoint)
void	**emberAfIasAceClusterServerTickCallback** (int8u endpoint)
void	**emberAfIasAceClusterClientTickCallback** (int8u endpoint)
void	**emberAfIasAceClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIasAceClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIasAceClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIasAceClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIasAceClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIasAceClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIasAceClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfIasAceClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

IAS WD Cluster Callbacks

boolean	**emberAfIasWdClusterStartWarningCallback** (int8u warningInfo, int16u warningDuration)
boolean	**emberAfIasWdClusterSquawkCallback** (int8u squawkInfo)

void	**emberAfIasWdClusterServerInitCallback** (int8u endpoint)
void	**emberAfIasWdClusterClientInitCallback** (int8u endpoint)
void	**emberAfIasWdClusterServerTickCallback** (int8u endpoint)
void	**emberAfIasWdClusterClientTickCallback** (int8u endpoint)
void	**emberAfIasWdClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIasWdClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIasWdClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIasWdClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIasWdClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIasWdClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIasWdClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfIasWdClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Door Lock Cluster Callbacks

boolean	**emberAfDoorLockClusterLockDoorCallback** (void)
boolean	**emberAfDoorLockClusterUnlockDoorCallback** (void)
boolean	**emberAfDoorLockClusterLockDoorResponseCallback** (int8u status)
boolean	**emberAfDoorLockClusterUnlockDoorResponseCallback** (int8u status)
void	**emberAfDoorLockClusterServerInitCallback** (int8u endpoint)
void	**emberAfDoorLockClusterClientInitCallback** (int8u endpoint)
void	**emberAfDoorLockClusterServerTickCallback** (int8u endpoint)
void	**emberAfDoorLockClusterClientTickCallback** (int8u endpoint)
void	**emberAfDoorLockClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfDoorLockClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfDoorLockClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfDoorLockClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfDoorLockClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfDoorLockClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfDoorLockClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfDoorLockClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Window Covering Cluster Callbacks

boolean	**emberAfWindowCoveringClusterWindowCoveringUpOpenCallback** (void)
boolean	**emberAfWindowCoveringClusterWindowCoveringDownCloseCallback** (void)
boolean	**emberAfWindowCoveringClusterWindowCoveringStopCallback** (void)
boolean	**emberAfWindowCoveringClusterWindowCoveringGoToLiftSetpointCallback** (int8u indexOfLiftSetpoint)
boolean	**emberAfWindowCoveringClusterWindowCoveringGoToLiftValueCallback** (int16u liftValue)
boolean	**emberAfWindowCoveringClusterWindowCoveringGoToLiftPercentageCallback** (int8u percentageLiftValue)
boolean	**emberAfWindowCoveringClusterWindowCoveringGoToTiltSetpointCallback** (int8u indexOfTiltSetpoint)
boolean	**emberAfWindowCoveringClusterWindowCoveringGoToTiltValueCallback** (int16u tiltValue)
boolean	**emberAfWindowCoveringClusterWindowCoveringGoToTiltPercentageCallback** (int8u percentageTiltValue)
boolean	**emberAfWindowCoveringClusterWindowCoveringGoToProgramSetpointCallback** (int8u setpointType, int8u setpointIndex, int16u setpointValue)
void	**emberAfWindowCoveringClusterServerInitCallback** (int8u endpoint)
void	**emberAfWindowCoveringClusterClientInitCallback** (int8u endpoint)
void	**emberAfWindowCoveringClusterServerTickCallback** (int8u endpoint)
void	**emberAfWindowCoveringClusterClientTickCallback** (int8u endpoint)
void	**emberAfWindowCoveringClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfWindowCoveringClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfWindowCoveringClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfWindowCoveringClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfWindowCoveringClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfWindowCoveringClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfWindowCoveringClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfWindowCoveringClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Generic Tunnel Cluster Callbacks

boolean	**emberAfGenericTunnelClusterMatchProtocolAddressCallback** (int8u *protocolAddress)
boolean	**emberAfGenericTunnelClusterMatchProtocolAddressResponseCallback** (void)
boolean	**emberAfGenericTunnelClusterAdvertiseProtocolAddressCallback** (int8u *protocolAddress)
void	**emberAfGenericTunnelClusterServerInitCallback** (int8u endpoint)
void	**emberAfGenericTunnelClusterClientInitCallback** (int8u endpoint)
void	**emberAfGenericTunnelClusterServerTickCallback** (int8u endpoint)

void	**emberAfGenericTunnelClusterClientTickCallback** (int8u endpoint)
void	**emberAfGenericTunnelClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfGenericTunnelClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfGenericTunnelClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfGenericTunnelClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfGenericTunnelClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfGenericTunnelClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfGenericTunnelClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfGenericTunnelClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

BACnet Protocol Tunnel Cluster Callbacks

boolean	**emberAfBacnetProtocolTunnelClusterTransferNpduCallback** (int8u *NPDU)
void	**emberAfBacnetProtocolTunnelClusterServerInitCallback** (int8u endpoint)
void	**emberAfBacnetProtocolTunnelClusterClientInitCallback** (int8u endpoint)
void	**emberAfBacnetProtocolTunnelClusterServerTickCallback** (int8u endpoint)
void	**emberAfBacnetProtocolTunnelClusterClientTickCallback** (int8u endpoint)
void	**emberAfBacnetProtocolTunnelClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfBacnetProtocolTunnelClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfBacnetProtocolTunnelClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfBacnetProtocolTunnelClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfBacnetProtocolTunnelClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfBacnetProtocolTunnelClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfBacnetProtocolTunnelClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfBacnetProtocolTunnelClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Over the Air Bootloading Cluster Cluster Callbacks

boolean	**emberAfOtaBootloadClusterImageNotifyCallback** (int8u payloadType, int8u queryJitter, int16u manufacturerId, int16u imageType, int32u newFileVersion)
boolean	**emberAfOtaBootloadClusterQueryNextImageRequestCallback** (int8u fieldControl, int16u manufacturerId, int16u imageType, int32u currentFileVersion, int16u hardwareVersion)
boolean	**emberAfOtaBootloadClusterQueryNextImageResponseCallback** (int8u status, int16u manufacturerId, int16u imageType, int32u fileVersion, int32u imageSize)
boolean	**emberAfOtaBootloadClusterImageBlockRequestCallback** (int8u fieldControl, int16u manufacturerId, int16u imageType, int32u fileVersion, int32u fileOffset, int8u maxDataSize, int8u *requestNodeAddress)
boolean	**emberAfOtaBootloadClusterImagePageRequestCallback** (int8u fieldControl, int16u manufacturerId, int16u imageType, int32u fileVersion, int32u fileOffset, int8u maxDataSize, int16u pageSize, int16u responseSpacing, int8u *requestNodeAddress)
boolean	**emberAfOtaBootloadClusterImageBlockResponseCallback** (int8u status, int16u manufacturerId, int16u imageType, int32u fileVersion, int32u fileOffset, int8u dataSize, int8u *imageData)
boolean	**emberAfOtaBootloadClusterUpgradeEndRequestCallback** (int8u status, int16u manufacturerId, int16u imageType, int32u fileVersion)
boolean	**emberAfOtaBootloadClusterUpgradeEndResponseCallback** (int16u manufacturerId, int16u imageType, int32u fileVersion, int32u currentTime, int32u upgradeTime)
boolean	**emberAfOtaBootloadClusterQuerySpecificFileRequestCallback** (int8u *requestNodeAddress, int16u manufacturerId, int16u imageType, int32u fileVersion, int16u currentZigbeeStackVersion)
boolean	**emberAfOtaBootloadClusterQuerySpecificFileResponseCallback** (int8u status, int16u manufacturerId, int16u imageType, int32u fileVersion, int32u imageSize)
void	**emberAfOtaBootloadClusterServerInitCallback** (int8u endpoint)
void	**emberAfOtaBootloadClusterClientInitCallback** (int8u endpoint)
void	**emberAfOtaBootloadClusterServerTickCallback** (int8u endpoint)
void	**emberAfOtaBootloadClusterClientTickCallback** (int8u endpoint)
void	**emberAfOtaBootloadClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOtaBootloadClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOtaBootloadClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOtaBootloadClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOtaBootloadClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOtaBootloadClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOtaBootloadClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfOtaBootloadClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Demand Response and Load Control Cluster Callbacks

boolean	**emberAfDemandResponseLoadControlClusterLoadControlEventCallback** (int32u issuerEventId, int16u deviceClass, int8u utilityEnrollmentGroup, int32u startTime, int16u durationInMinutes, int8u criticalityLevel, int8u coolingTemperatureOffset, int8u heatingTemperatureOffset, int16s coolingTemperatureSetPoint, int16s heatingTemperatureSetPoint, int8s averageLoadAdjustmentPercentage, int8u dutyCycle, int8u eventControl)
boolean	**emberAfDemandResponseLoadControlClusterCancelLoadControlEventCallback** (int32u issuerEventId, int16u deviceClass, int8u utilityEnrollmentGroup, int8u cancelControl, int32u effectiveTime)
boolean	**emberAfDemandResponseLoadControlClusterCancelAllLoadControlEventsCallback** (int8u cancelControl)
boolean	**emberAfDemandResponseLoadControlClusterReportEventStatusCallback** (int32u issuerEventId, int8u eventStatus, int32u eventStatusTime, int8u criticalityLevelApplied, int16s coolingTemperatureSetPointApplied, int16s heatingTemperatureSetPointApplied, int8s averageLoadAdjustmentPercentageApplied, int8u dutyCycleApplied, int8u eventControl, int8u signatureType, int8u *signature)
boolean	**emberAfDemandResponseLoadControlClusterGetScheduledEventsCallback** (int32u startTime, int8u numberOfEvents)
void	**emberAfDemandResponseLoadControlClusterServerInitCallback** (int8u endpoint)
void	**emberAfDemandResponseLoadControlClusterClientInitCallback** (int8u endpoint)
void	**emberAfDemandResponseLoadControlClusterServerTickCallback** (int8u endpoint)
void	**emberAfDemandResponseLoadControlClusterClientTickCallback** (int8u endpoint)
void	**emberAfDemandResponseLoadControlClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfDemandResponseLoadControlClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfDemandResponseLoadControlClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfDemandResponseLoadControlClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfDemandResponseLoadControlClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfDemandResponseLoadControlClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfDemandResponseLoadControlClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfDemandResponseLoadControlClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Simple Metering Cluster Callbacks

boolean	**emberAfSimpleMeteringClusterGetProfileResponseCallback** (int32u endTime, int8u status, int8u profileIntervalPeriod, int8u numberOfPeriodsDelivered, int8u *intervals)
boolean	**emberAfSimpleMeteringClusterRequestMirrorCallback** (void)
boolean	**emberAfSimpleMeteringClusterRemoveMirrorCallback** (void)
boolean	**emberAfSimpleMeteringClusterRequestFastPollModeResponseCallback** (int8u appliedUpdatePeriod, int32u fastPollModeEndtime)

boolean	**emberAfSimpleMeteringClusterGetProfileCallback** (int8u intervalChannel, int32u endTime, int8u numberOfPeriods)
boolean	**emberAfSimpleMeteringClusterRequestMirrorResponseCallback** (int16u endpointId)
boolean	**emberAfSimpleMeteringClusterMirrorRemovedCallback** (int16u endpointId)
boolean	**emberAfSimpleMeteringClusterRequestFastPollModeCallback** (int8u fastPollUpdatePeriod, int8u duration)
void	**emberAfSimpleMeteringClusterServerInitCallback** (int8u endpoint)
void	**emberAfSimpleMeteringClusterClientInitCallback** (int8u endpoint)
void	**emberAfSimpleMeteringClusterServerTickCallback** (int8u endpoint)
void	**emberAfSimpleMeteringClusterClientTickCallback** (int8u endpoint)
void	**emberAfSimpleMeteringClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfSimpleMeteringClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfSimpleMeteringClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfSimpleMeteringClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfSimpleMeteringClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfSimpleMeteringClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfSimpleMeteringClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfSimpleMeteringClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Price Cluster Callbacks

boolean	**emberAfPriceClusterPublishPriceCallback** (int32u providerId, int8u *rateLabel, int32u issuerEventId, int32u currentTime, int8u unitOfMeasure, int16u currency, int8u priceTrailingDigitAndPriceTier, int8u numberOfPriceTiersAndRegisterTier, int32u startTime, int16u durationInMinutes, int32u price, int8u priceRatio, int32u generationPrice, int8u generationPriceRatio, int32u alternateCostDelivered, int8u alternateCostUnit, int8u alternateCostTrailingDigit, int8u numberOfBlockThresholds, int8u priceControl)
boolean	**emberAfPriceClusterPublishBlockPeriodCallback** (int32u providerId, int32u issuerEventId, int32u blockPeriodStartTime, int32u blockPeriodDurationInMinutes, int8u numberOfPriceTiersAndNumberOfBlockThresholds, int8u blockPeriodControl)
boolean	**emberAfPriceClusterGetCurrentPriceCallback** (int8u commandOptions)
boolean	**emberAfPriceClusterGetScheduledPricesCallback** (int32u startTime, int8u numberOfEvents)
boolean	**emberAfPriceClusterPriceAcknowledgementCallback** (int32u providerId, int32u issuerEventId, int32u priceAckTime, int8u control)
boolean	**emberAfPriceClusterGetBlockPeriodsCallback** (int32u startTime, int8u numberOfEvents)
void	**emberAfPriceClusterServerInitCallback** (int8u endpoint)
void	**emberAfPriceClusterClientInitCallback** (int8u endpoint)
void	**emberAfPriceClusterServerTickCallback** (int8u endpoint)
void	**emberAfPriceClusterClientTickCallback** (int8u endpoint)
void	**emberAfPriceClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)

void	**emberAfPriceClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfPriceClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPriceClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPriceClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPriceClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPriceClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfPriceClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Messaging Cluster Callbacks

boolean	**emberAfMessagingClusterDisplayMessageCallback** (int32u messageId, int8u messageControl, int32u startTime, int16u durationInMinutes, int8u *message)
boolean	**emberAfMessagingClusterCancelMessageCallback** (int32u messageId, int8u messageControl)
boolean	**emberAfMessagingClusterGetLastMessageCallback** (void)
boolean	**emberAfMessagingClusterMessageConfirmationCallback** (int32u messageId, int32u confirmationTime)
void	**emberAfMessagingClusterServerInitCallback** (int8u endpoint)
void	**emberAfMessagingClusterClientInitCallback** (int8u endpoint)
void	**emberAfMessagingClusterServerTickCallback** (int8u endpoint)
void	**emberAfMessagingClusterClientTickCallback** (int8u endpoint)
void	**emberAfMessagingClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfMessagingClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfMessagingClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfMessagingClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfMessagingClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfMessagingClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfMessagingClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfMessagingClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Tunneling Cluster Callbacks

boolean	**emberAfTunnelingClusterRequestTunnelCallback** (int8u protocolId, int16u manufacturerCode, int8u flowControlSupport)
boolean	**emberAfTunnelingClusterCloseTunnelCallback** (int16u tunnelId)
boolean	**emberAfTunnelingClusterTransferDataClientToServerCallback** (int16u tunnelId, int8u *data)
boolean	**emberAfTunnelingClusterTransferDataErrorClientToServerCallback** (int16u tunnelId, int8u transferDataStatus)
boolean	**emberAfTunnelingClusterAckTransferDataClientToServerCallback** (int16u tunnelId, int16u numberOfBytesLeft)
boolean	**emberAfTunnelingClusterReadyDataClientToServerCallback** (int16u tunnelId, int16u numberOfOctetsLeft)
boolean	**emberAfTunnelingClusterRequestTunnelResponseCallback** (int16u tunnelId, int8u tunnelStatus)
boolean	**emberAfTunnelingClusterTransferDataServerToClientCallback** (int16u tunnelId, int8u *data)
boolean	**emberAfTunnelingClusterTransferDataErrorServerToClientCallback** (int16u tunnelId, int8u transferDataStatus)
boolean	**emberAfTunnelingClusterAckTransferDataServerToClientCallback** (int16u tunnelId, int16u numberOfBytesLeft)
boolean	**emberAfTunnelingClusterReadyDataServerToClientCallback** (int16u tunnelId, int16u numberOfOctetsLeft)
void	**emberAfTunnelingClusterServerInitCallback** (int8u endpoint)
void	**emberAfTunnelingClusterClientInitCallback** (int8u endpoint)
void	**emberAfTunnelingClusterServerTickCallback** (int8u endpoint)
void	**emberAfTunnelingClusterClientTickCallback** (int8u endpoint)
void	**emberAfTunnelingClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfTunnelingClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfTunnelingClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfTunnelingClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfTunnelingClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfTunnelingClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfTunnelingClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfTunnelingClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Prepayment Cluster Callbacks

boolean	**emberAfPrepaymentClusterSelectAvailableEmergencyCreditCallback** (int32u commandDateTime, int8u originatingDevice, int8u *siteId, int8u *meterSerialNumber)
boolean	**emberAfPrepaymentClusterChangeSupplyCallback** (int32u providerId, int32u requestDateTime, int8u *siteId, int8u *meterSerialNumber, int32u implementationDateTime, int8u proposedSupplyStatus, int8u originatorIdSupplyControlBits)
boolean	**emberAfPrepaymentClusterSupplyStatusResponseCallback** (int32u providerId, int32u implementationtDateTime, int8u supplyStatus)

void	**emberAfPrepaymentClusterServerInitCallback** (int8u endpoint)
void	**emberAfPrepaymentClusterClientInitCallback** (int8u endpoint)
void	**emberAfPrepaymentClusterServerTickCallback** (int8u endpoint)
void	**emberAfPrepaymentClusterClientTickCallback** (int8u endpoint)
void	**emberAfPrepaymentClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfPrepaymentClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfPrepaymentClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPrepaymentClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPrepaymentClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPrepaymentClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPrepaymentClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfPrepaymentClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Key establishment Cluster Callbacks

boolean	**emberAfKeyEstablishmentClusterClientCommandReceivedCallback** (**EmberAfClusterCommand** *cmd)
boolean	**emberAfKeyEstablishmentClusterServerCommandReceivedCallback** (**EmberAfClusterCommand** *cmd)
boolean	**emberAfKeyEstablishmentClusterInitiateKeyEstablishmentRequestCallback** (int16u keyEstablishmentSuite, int8u ephemeralDataGenerateTime, int8u confirmKeyGenerateTime, int8u *identity)
boolean	**emberAfKeyEstablishmentClusterEphemeralDataRequestCallback** (int8u *ephemeralData)
boolean	**emberAfKeyEstablishmentClusterConfirmKeyDataRequestCallback** (int8u *secureMessageAuthenticationCode)
boolean	**emberAfKeyEstablishmentClusterTerminateKeyEstablishmentCallback** (int8u statusCode, int8u waitTime, int16u keyEstablishmentSuite)
boolean	**emberAfKeyEstablishmentClusterInitiateKeyEstablishmentResponseCallback** (int16u requestedKeyEstablishmentSuite, int8u ephemeralDataGenerateTime, int8u confirmKeyGenerateTime, int8u *identity)
boolean	**emberAfKeyEstablishmentClusterEphemeralDataResponseCallback** (int8u *ephemeralData)
boolean	**emberAfKeyEstablishmentClusterConfirmKeyDataResponseCallback** (int8u *secureMessageAuthenticationCode)
void	**emberAfKeyEstablishmentClusterServerInitCallback** (int8u endpoint)
void	**emberAfKeyEstablishmentClusterClientInitCallback** (int8u endpoint)
void	**emberAfKeyEstablishmentClusterServerTickCallback** (int8u endpoint)
void	**emberAfKeyEstablishmentClusterClientTickCallback** (int8u endpoint)
void	**emberAfKeyEstablishmentClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)

void	**emberAfKeyEstablishmentClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfKeyEstablishmentClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfKeyEstablishmentClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfKeyEstablishmentClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfKeyEstablishmentClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfKeyEstablishmentClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfKeyEstablishmentClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Sample Mfg Specific Cluster Cluster Callbacks

boolean	**emberAfSampleMfgSpecificClusterCommandOneCallback** (int8u argOne)
void	**emberAfSampleMfgSpecificClusterServerInitCallback** (int8u endpoint)
void	**emberAfSampleMfgSpecificClusterClientInitCallback** (int8u endpoint)
void	**emberAfSampleMfgSpecificClusterServerTickCallback** (int8u endpoint)
void	**emberAfSampleMfgSpecificClusterClientTickCallback** (int8u endpoint)
void	**emberAfSampleMfgSpecificClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfSampleMfgSpecificClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfSampleMfgSpecificClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfSampleMfgSpecificClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfSampleMfgSpecificClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfSampleMfgSpecificClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfSampleMfgSpecificClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfSampleMfgSpecificClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Detailed Description

This header provides callback function prototypes to interface the developer's application code with the Ember Application Framework.

This header provides callback function prototypes for the Basic server cluster application framework plugin.

This header provides callback function prototypes for the Demand response load control cluster client application framework plugin.

This header provides callback function prototypes for the Messaging client cluster application framework plugin.

This header provides callback function prototypes for the Price client plugin application framework plugin.

This header provides callback function prototypes for the PSL Commissioning application framework plugin.

This header provides callback function prototypes for the Simple metering client application framework plugin.

This header provides callback function prototypes for the Test plugin application framework plugin.

This header provides callback function prototypes for the Tunneling client cluster application framework plugin.

This header provides callback function prototypes for the Tunneling server cluster application framework plugin.

Function Documentation

```
boolean emberAfReadAttributesResponseCallback ( EmberAfClusterId clusterId,
                                                int8u *           buffer,
                                                int16u            bufLen
                                              )
```

Read Attributes Response.

This function is called by the application framework when a Read Attributes Response command is received from an external device. The application should return TRUE if the message was processed or FALSE if it was not.

Parameters:

clusterId	The cluster identifier of this response. Ver.: always
buffer	Buffer containing the list of read attribute status records. Ver.: always
bufLen	The length in bytes of the list. Ver.: always

```
boolean emberAfWriteAttributesResponseCallback ( EmberAfClusterId clusterId,
                                                 int8u *           buffer,
                                                 int16u            bufLen
                                               )
```

Write Attributes Response.

This function is called by the application framework when a Write Attributes Response command is received

from an external device. The application should return TRUE if the message was processed or FALSE if it was not.

Parameters:

clusterId	The cluster identifier of this response. Ver.: always
buffer	Buffer containing the list of write attribute status records. Ver.: always
bufLen	The length in bytes of the list. Ver.: always

```
boolean emberAfConfigureReportingResponseCallback ( EmberAfClusterId clusterId,
                                                    int8u *          buffer,
                                                    int16u           bufLen
                                                  )
```

Configure Reporting Response.

This function is called by the application framework when a Configure Reporting Response command is received from an external device. The application should return TRUE if the message was processed or FALSE if it was not.

Parameters:

clusterId	The cluster identifier of this response. Ver.: always
buffer	Buffer containing the list of attribute status records. Ver.: always
bufLen	The length in bytes of the list. Ver.: always

```
boolean emberAfReadReportingConfigurationResponseCallback ( EmberAfClusterId clusterId,
                                                            int8u *          buffer,
                                                            int16u           bufLen
                                                          )
```

Read Reporting Configuration Response.

This function is called by the application framework when a Read Reporting Configuration Response command is received from an external device. The application should return TRUE if the message was processed or FALSE if it was not.

Parameters:

clusterId	The cluster identifier of this response. Ver.: always
buffer	Buffer containing the list of attribute reporting configuration records. Ver.: always
bufLen	The length in bytes of the list. Ver.: always

```
boolean emberAfReportAttributesCallback ( EmberAfClusterId clusterId,
                                          int8u *          buffer,
                                          int16u           bufLen
```

```
)
```

Report Attributes.

This function is called by the application framework when a Report Attributes command is received from an external device. The application should return TRUE if the message was processed or FALSE if it was not.

Parameters:

clusterId	The cluster identifier of this command. Ver.: always
buffer	Buffer containing the list of attribute report records. Ver.: always
bufLen	The length in bytes of the list. Ver.: always

```
boolean emberAfDefaultResponseCallback ( EmberAfClusterId clusterId,
                                         int8u            commandId,
                                         EmberAfStatus    status
                                       )
```

Default Response.

This function is called by the application framework when a Default Response command is received from an external device. The application should return TRUE if the message was processed or FALSE if it was not.

Parameters:

clusterId	The cluster identifier of this response. Ver.: always
commandId	The command identifier to which this is a response. Ver.: always
status	Specifies either SUCCESS or the nature of the error that was detected in the received command. Ver.: always

```
boolean emberAfDiscoverAttributesResponseCallback ( EmberAfClusterId clusterId,
                                                    boolean          discoveryComplete,
                                                    int8u *          buffer,
                                                    int16u           bufLen
                                                  )
```

Discover Attributes Response.

This function is called by the application framework when a Discover Attributes Response command is received from an external device. The Discover Attributes Response command contains a boolean indicating if discovery is complete and a list of zero or more attribute identifier/type records. The application should return TRUE if the message was processed or FALSE if it was not.

Parameters:

clusterId	The cluster identifier of this response. Ver.: always
discoveryComplete	Indicates whether there are more attributes to be discovered. TRUE if there are no more attributes to be discovered. Ver.: always

buffer	Buffer containing the list of attribute identifier/type records. Ver.: always
bufLen	The length in bytes of the list. Ver.: always

```
EmberAfAttributeWritePermission emberAfAllowNetworkWriteAttributeCallback ( int8u              endpoint,
                                                                            EmberAfClusterId    clusterId,
                                                                            EmberAfAttributeId  attributeId,
                                                                            int8u               mask,
                                                                            int16u              manufacturerCode,
                                                                            int8u *             value,
                                                                            int8u               type
                                                                          )
```

Allow Network Write Attribute.

This function is called by the application framework before it writes an attribute in response to a write attribute request from an external device. The value passed into this callback is the value to which the attribute is to be set by the framework. The application framework expects one of three possible values 0=Do not allow write, 1=Allow write normal (attribute must be writeable), 2=Allow write for a read only attribute.

Parameters:

endpoint	Ver.: always
clusterId	Ver.: always
attributeId	Ver.: always
mask	Ver.: always
manufacturerCode	Ver.: always
value	Ver.: always
type	Ver.: always

```
void emberAfPreAttributeChangeCallback ( int8u              endpoint,
                                         EmberAfClusterId   clusterId,
                                         EmberAfAttributeId attributeId,
                                         int8u              mask,
                                         int16u             manufacturerCode,
                                         int8u              type,
                                         int8u              size,
                                         int8u *            value
                                       )
```

Pre Attribute Change.

This function is called by the application framework before it changes an attribute value. The value passed into this callback is the value to which the attribute is to be set by the framework.

Parameters:

endpoint	Ver.: always
clusterId	Ver.: always
attributeId	Ver.: always
mask	Ver.: always
manufacturerCode	Ver.: always
type	Ver.: always
size	Ver.: always
value	Ver.: always

```
void emberAfPostAttributeChangeCallback ( int8u              endpoint,
                                          EmberAfClusterId   clusterId,
                                          EmberAfAttributeId attributeId,
                                          int8u              mask,
                                          int16u             manufacturerCode,
                                          int8u              type,
                                          int8u              size,
                                          int8u *            value
                                        )
```

Post Attribute Change.

This function is called by the application framework after it changes an attribute value. The value passed into this callback is the value to which the attribute was set by the framework.

Parameters:

endpoint	Ver.: always
clusterId	Ver.: always
attributeId	Ver.: always
mask	Ver.: always

manufacturerCode Ver.: always
type Ver.: always
size Ver.: always
value Ver.: always

boolean emberAfPreMessageReceivedCallback (EmberAfIncomingMessage * incomingMessage)

Pre Message Received.

This function is called by the application framework when a raw message is received but has not yet been parsed by the application framework. This function should return a boolean value indicating whether the message has been handled by the user application code and should be ignored by the AppFramework. By default, this callback returns FALSE indicating that the message has not been handled and should be handled by the AppFramework. This function receives a pointer to a struct containing the super set of arguments sent to the emberIncomingMessageHandler and the ezspIncomingMessageHandler.

Parameters:

incomingMessage Ver.: always

boolean emberAfPreCommandReceivedCallback (EmberAfClusterCommand * cmd)

Pre Command Received.

This function is called by the application framework when a command is received but has yet to be handled by the framework code. This function should return a boolean value indicating whether the command has been handled by the user application code and should be ignored by the AppFramework. By default, this callback returns FALSE indicating that the message has not been handled and should be handled by the

Parameters:

cmd Ver.: always

boolean emberAfPreZDOMessageReceivedCallback (EmberNodeId emberNodeId, EmberApsFrame * apsFrame, int8u * message, int16u length)

Pre ZDO Message Received.

This function passes the application an incoming ZDO message and gives the appictation the opportunity to handle it. By default, this callback returns FALSE indicating that the incoming ZDO message has not

been handled and should be handled by the Application Framework.

Parameters:

emberNodeId	Ver.: always
apsFrame	Ver.: always
message	Ver.: always
length	Ver.: always

```
boolean emberAfMessageSentCallback ( EmberOutgoingMessageType type,
                                     int16u                   indexOrDestination,
                                     EmberApsFrame *          apsFrame,
                                     int16u                   msgLen,
                                     int8u *                  message,
                                     EmberStatus              status
                                   )
```

Message Sent.

This function is called by the application framework from the message sent handler, when it is informed by the stack regarding the message sent status. All of the values passed to the emberMessageSentHandler are passed on to this callback. This provides an opportunity for the application to verify that its message has been sent successfully and take the appropriate action. This callback should return a boolean value of TRUE or FALSE. A value of TRUE indicates that the message sent notification has been handled and should not be handled by the application framework.

Parameters:

type	Ver.: always
indexOrDestination	Ver.: always
apsFrame	Ver.: always
msgLen	Ver.: always
message	Ver.: always
status	Ver.: always

```
boolean emberAfStackStatusCallback ( EmberStatus status )
```

Stack Status.

This function is called by the application framework from the stack status handler. This callbacks

provides applications an opportunity to be notified of changes to the stack status and take appropriate action. The application should return TRUE if the status has been handled and should not be handled by the application framework.

Parameters:

status Ver.: always

void emberAfNcpInitCallback (void)

Ncp Init.

This callback is used for initializing the network coprocessor either at startup, or upon reset. It should be used to perform any kind of EZSP initialization, as it will be called every time when NCP is reset. In an SOC environment, this callbacks is not used.

void emberAfMainInitCallback (void)

Main Init.

This is the main application initialization callback. It is called prior to application initialization and gives the application a chance to take care of any of its own initialization procedures.

void emberAfClusterInitCallback (int8u endpoint, EmberAfClusterId clusterId)

Cluster Init.

This function is called when a specific cluster is initialized. It gives the application an opportunity to take care of cluster initialization procedures. It is called exactly once for each endpoint where cluster is present.

Parameters:

endpoint Ver.: always

clusterId Ver.: always

boolean emberAfPreGoToSleepCallback (int32u sleepDurationAttempt)

Pre Go To Sleep.

Called directly before a device goes to sleep. This function is passed the number of milliseconds the device is attempting to go to sleep for. This function returns a boolean indicating if sleep should be canceled. A returned value of TRUE indicates that sleep should be canceled. A return value of FALSE (default) indicates that sleep my continue as expected.

Parameters:

sleepDurationAttempt	Ver.: always

void emberAfPostWakeUpCallback (int32u sleepDuration)

Post Wake Up.

Called directly after a device wakes up from sleep. This function is passed the number of milliseconds the device actually slept for.

Parameters:

sleepDuration	Ver.: always

void emberAfMainTickCallback (void)

Main Tick.

Whenever main application tick is called, this callback will be called at the end of the main tick execution.

boolean emberAfExternalAttributeWriteCallback (int8u endpoint, EmberAfClusterId clusterId, EmberAfAttributeMetadata * attributeMetadata, int16u manufacturerCode, int8u * buffer)

External Attribute Write.

This callback is called for whenever an attribute is written, if this attribute was tagged to be stored externally. Function should return true if write was successful and false if not. Upon successful exit, the data from the passed buffer should be stored as the current value of the attribute. Important to note is, that if this method ever returns false, then it is possible to break contract of the write-undivided-attributes global command. That command checks for ranges and attribute existence, but if this method returns false, when the attribute is finally written, then the contract of writing all or no attributes will be broken.

Parameters:

endpoint	Ver.: always
clusterId	Ver.: always
attributeMetadata	Ver.: always
manufacturerCode	Ver.: always

buffer	Ver.: always

```
boolean emberAfExternalAttributeReadCallback ( int8u                    endpoint,
                                               EmberAfClusterId         clusterId,
                                               EmberAfAttributeMetadata * attributeMetadata,
                                               int16u                   manufacturerCode,
                                               int8u *                  buffer
                                             )
```

External Attribute Read.

If an attribute is tagged to be stored externally, this function will be called whenever that attribute is read. This function should return TRUE if read was successful and FALSE if not. The passed buffer should be populated with the associated attributes value.

Parameters:

endpoint	Ver.: always
clusterId	Ver.: always
attributeMetadata	Ver.: always
manufacturerCode	Ver.: always
buffer	Ver.: always

```
EmberAfStatus emberAfReportEntryConfiguredCallback ( EmberAfReportEntry * entry )
```

Report Entry Configured.

This callback is called whenever a reporting entry is configured, including when entries are deleted or updated. The application can use this callback for scheduling readings or measurements based on the minimum and maximum reporting interval for the entry. The application should return EMBER_ZCL_STATUS_SUCCESS if it can support the configuration or an error status otherwise. Note: attribute reporting is required for many clusters and attributes, so rejecting a reporting configuration may violate ZigBee specifications.

Parameters:

entry	Ver.: always

```
void emberAfTrustCenterJoinCallback ( EmberNodeId    newNodeId,
                                      EmberEUI64     newNodeEui64,
                                      EmberNodeId    parentOfNewNode,
```

```
                                    EmberDeviceUpdate status,
                                    EmberJoinDecision decision
                                  )
```

Trust Center Join.

This callback is called from within the application framework's implementation of emberTrustCenterJoinHandler or ezspTrustCenterJoinHandler. This callback provides the same arguments passed to the TrustCenterJoinHandler. For more information about the TrustCenterJoinHandler please see documentation included in stack/include/trust-center.h.

Parameters:

newNodeId	Ver.: always
newNodeEui64	Ver.: always
parentOfNewNode	Ver.: always
status	Ver.: always
decision	Ver.: always

int32u emberAfGetCurrentTimeCallback (void)

Get Current Time.

This callback is called when device attempts to get current time from the hardware. If this device has means to retrieve exact time, then this method should implement it. If the callback can't provide the exact time it should return 0 to indicate failure. Default action is to return 0, which indicates that device does not have access to real time.

void emberAfSetTimeCallback (int32u utcTime)

Set Time.

This callback should be implemented, if the device has access to real time clock, and has an ability to update that clock. The application framework expects to be passed the utcTime which is the number of seconds since the year 2000. Default implementation does nothing. Note: This function used to take time in year, month, day, hour, min, sec. We have changed this to utcTime in order to conserve code space.

Parameters:

utcTime	Ver.: always

EmberStatus emberAfInitiateKeyEstablishmentCallback (EmberNodeId nodeId, int8u endpoint

```
)
```

Initiate Key Establishment.

This function is called by the framework to initiate key establishment with a remote device. The application should return EMBER_SUCCESS if key establishment was initiated successfully. The application should call **emberAfKeyEstablishmentCallback** as events occur.

Parameters:

nodeId The node id of the remote device. Ver.: always

endpoint The endpoint on the remote device. Ver.: always

```
EmberStatus emberAfInitiateInterPanKeyEstablishmentCallback ( EmberPanId          panId,
                                                              const EmberEUI64    eui64
                                                            )
```

Initiate Inter Pan Key Establishment.

This function is called by the framework to initiate key establishment with a remote device on a different PAN. The application should return EMBER_SUCCESS if key establishment was initiated successfully. The application should call **emberAfInterPanKeyEstablishmentCallback** as events occur.

Parameters:

panId The PAN id of the remote device. Ver.: always

eui64 The EUI64 of the remote device. Ver.: always

```
boolean emberAfKeyEstablishmentCallback ( EmberAfKeyEstablishmentNotifyMessage status,
                                          boolean                              amInitiator,
                                          EmberNodeId                          partnerShortId,
                                          int8u                                delayInSeconds
                                        )
```

Key Establishment.

A callback by the key-establishment code to indicate an event has occurred. For error codes this is purely a notification. For non-error status codes (besides LINK_KEY_ESTABLISHED), it is the application's chance to allow or disallow the operation. If the application returns TRUE then the key establishment is allowed to proceed. If it returns FALSE, then key establishment is aborted. LINK_KEY_ESTABLISHED is a notification of success.

Parameters:

status Ver.: always

amInitiator	Ver.: always
partnerShortId	Ver.: always
delayInSeconds	Ver.: always

```
boolean emberAfInterPanKeyEstablishmentCallback ( EmberAfKeyEstablishmentNotifyMessage status,
                                                  boolean                              amInitiator,
                                                  EmberPanId                           panId,
                                                  const EmberEUI64                     eui64,
                                                  int8u                                delayInSeconds
                                                )
```

Inter Pan Key Establishment.

A callback by the key-establishment code to indicate an event has occurred. For error codes this is purely a notification. For non-error status codes (besides LINK_KEY_ESTABLISHED), it is the application's chance to allow or disallow the operation. If the application returns TRUE then the key establishment is allowed to proceed. If it returns FALSE, then key establishment is aborted. LINK_KEY_ESTABLISHED is a notification of success.

Parameters:

status	Ver.: always
amInitiator	Ver.: always
panId	Ver.: always
eui64	Ver.: always
delayInSeconds	Ver.: always

```
boolean emberAfPerformingKeyEstablishmentCallback ( void )
```

Performing Key Establishment.

This function is called by the framework to determine if the device is performing key establishment. The application should return TRUE if key establishment is in progress.

```
EmberStatus emberAfInitiatePartnerLinkKeyExchangeCallback ( EmberNodeId                                target,
                                                            int8u                                      endpoint,
                                                            EmberAfPartnerLinkKeyExchangeCallback *    callback
                                                          )
```

Initiate Partner Link Key Exchange.

This function is called by the framework to initiate a partner link key exchange with a remote device. The application should return EMBER_SUCCESS if the partner link key exchange was initiated successfully. When the partner link key exchange completes, the application should call the given callback.

Parameters:

target	The node id of the remote device. Ver.: always
endpoint	The key establishment endpoint of the remote device. Ver.: always
callback	The callback that should be called when the partner link key exchange completse. Ver.: always

EmberStatus emberAfPartnerLinkKeyExchangeRequestCallback (EmberEUI64 partner)

Partner Link Key Exchange Request.

This function is called by the framework on SOC platforms when a remote node requests a partner link key exchange. The application should return EMBER_SUCCESS to accept the request or any other status to reject it. On network coprocessor platforms, this function will not be called because the NCP handles partner link key exchange requests based on the binding policy.

Parameters:

partner	The EUI of the remote node. Ver.: always

void emberAfPartnerLinkKeyExchangeResponseCallback (EmberNodeId sender, EmberZdoStatus status)

Partner Link Key Exchange Response.

This function is called by the framework when a remote node requests a partner link key exchange. The application should return TRUE to accept the request or FALSE to reject it. On network coprocessor platforms, this function will not be called because the NCP handles partner link key exchange requests based on the binding policy.

Parameters:

sender	The EUI of the remote node. Ver.: always
status	The ZDO response status. Ver.: always

EmberStatus emberAfRegistrationStartCallback (void)

Registration Start.

This callback is called when the device joins a network and the registration process should begin. The application should return EMBER_SUCCESS if the registration process started successfully. When registration

is complete, the application should call emberAfRegistrationCallback with an indication of success or failure.

void emberAfRegistrationAbortCallback (void)

Registration Abort.

This callback is called when the device should abort the registration process.

void emberAfRegistrationCallback (boolean success)

Registration.

This callback is called when the device joins a network and the process of registration is complete. This callback provides a success value of TRUE if the registration process was successful and a value of FALSE if registration failed.

Parameters:

success TRUE if registration succeeded, FALSE otherwise. Ver.: always

void emberAfTrustCenterKeepaliveUpdateCallback (boolean registrationComplete)

Trust Center Keepalive Update.

This callback is called when the device finishes registration (successfully or otherwise) and the trust center keepalive process must be updated. If the keepalive process has not been started, then it is started. Otherwise if the keepalive is in the process of searching for the TC, it will process the result of that Trust Center search operation.

Parameters:

registrationComplete Ver.: always

void emberAfTrustCenterKeepaliveAbortCallback (void)

Trust Center Keepalive Abort.

This callback is called when the device should abort the trust center keepalive process.

void emberAfCustomCommandLineCallback (void)

Custom Command Line.

If you enable this callback, then it will be called when 'custom' CLI command is executed. It is responsible for reading it's own arguments.

boolean emberAfOtaServerIncomingMessageRawCallback (EmberAfClusterCommand * message)

Ota Server Incoming Message Raw.

This callback is for processing incoming messages for the Over-the-air bootload cluster server. ZCL will not process the message and instead hand the raw over the air data to the callback for its own processing.

Parameters:

message A pointer to the structure containing the message buffer and other information about it. Ver.: always

boolean emberAfOtaServerSendImageNotifyCallback (EmberNodeId dest, int8u endpoint, int8u payloadType, int8u queryJitter, const EmberAfOtaImageId * id)

Ota Server Send Image Notify.

This callback is an indication to the OTA server that it should send out notification about an OTA file that is available for download.

Parameters:

dest The destination of the image notify message. May be a broadcast address. Ver.: always

endpoint The destination endpoint of the image notify message. May be a broadcast endpoint. Ver.: always

payloadType The type of data the image notify message will contain. 0 = no data. 1 = Manufacturer ID. 2 = Manufacturer ID and the image type ID. 3 = Manufacturer ID, image type ID, and firmware version. Ver.: always

queryJitter The percentage of nodes that should respond to this message, from 1-100. On receipt of this message, each recipient will randomly choose a percentage and only query the server if their percentage is below this value. Ver.: always

id The image information that will be put in the message. The data within this struct that will be appended to the message is determined by the previous 'payloadType' argument. Ver.: always

boolean emberAfOtaStorageDriverInitCallback (void)

Ota Storage Driver Init.

The initialization code for the OTA storage driver.

boolean emberAfOtaStorageDriverReadCallback (int32u offset, int32u length, int8u * returnData)

Ota Storage Driver Read.

This callback defines the low-level means by which a device reads from the OTA storage device.

Parameters:

offset	The address offset from the start of the storage device where data is to be read. Ver.: always
length	Ver.: always
returnData	A pointer where the data read from the device should be written to. Ver.: always

```
boolean emberAfOtaStorageDriverWriteCallback ( const int8u *  dataToWrite,
                                               int32u         offset,
                                               int32u         length
                                             )
```

Ota Storage Driver Write.

This callback defines the low-level means by which a device reads from the OTA storage device.

Parameters:

dataToWrite	A pointer to the data that will be written to the storage device. Ver.: always
offset	The address offset from the start of the storage device where data will be written. Ver.: always
length	The length of the data to be written to the storage device. Ver.: always

```
int8u emberAfOtaBootloadCallback ( const EmberAfOtaImageId *  id,
                                   int16u                     ncpUpgradeTagId
                                 )
```

Ota Bootload.

The platform specific routine to bootload the device from a ZigBee over-the-air upgrade file.

Parameters:

id	A pointer to the structure that contains the information about what OTA image to bootload. Ver.: always
ncpUpgradeTagId	The tag ID of the upgrade data that will be used to bootload the device. Ver.: always

```
EmberAfOtaStorageStatus emberAfOtaStorageInitCallback ( void )
```

Ota Storage Init.

This callback initializes the ZigBee Over-the-air storage module.

void emberAfOtaStorageCloseCallback (void)

Ota Storage Close.

This callback shuts down the ZigBee Over-the-air storage module.

int8u emberAfOtaStorageGetCountCallback (void)

Ota Storage Get Count.

This callback returns the total number of ZigBee Over-the-air upgrade images stored in the storage module.

EmberAfOtaImageId emberAfOtaStorageSearchCallback (int16u manufacturerId, int16u imageTypeId, const int16u * hardwareVersion)

Ota Storage Search.

This callback searches through the list of all images for one that matches the passed parameters. On success an image identifier is returned with a matching image. On failure emberAfInvalidImageId is returned.

Parameters:

manufacturerId	The ZigBee assigned identifier of the manufacturer contained in the OTA image being searched for. Ver.: always
imageTypeId	The image type identifier contained in the OTA image being searched for. Ver.: always
hardwareVersion	This is a pointer to the hardware version that will be used in the search. If the pointer is NULL, hardware version will not be considered when searching for matching images. If it points to a value, the search will only consider images where that value falls between the minimum and maxmimum hardware version specified in the OTA file. If no hardware version is present in an OTA file but the other parameters match, the file will be considered a match Ver.: always

int32u emberAfOtaStorageGetTotalImageSizeCallback (const EmberAfOtaImageId * id)

Ota Storage Get Total Image Size.

This function returns the total size of the ZigBee Over-the-air file with the passed parameters. If no file is found with those parameters, 0 is returned.

Parameters:

id	A pointer to the image identifier for the OTA file to retrieve information for. Ver.: always

EmberAfOtaImageId emberAfOtaStorageIteratorFirstCallback (void)

Ota Storage Iterator First.

This callback lets you walk through the list of all OTA files by jumping to the first file in the list maintained by the storage module. If there is no file then emberAfOtaInvalidImageId is returned.

EmberAfOtaImageId emberAfOtaStorageIteratorNextCallback (void)

Ota Storage Iterator Next.

This callback lets you walk through the list of all OTA files by jumping to the next file in the list maintained by the storage module. If there is no next file then emberAfOtaInvalidImageId is returned.

EmberAfOtaStorageStatus emberAfOtaStorageGetFullHeaderCallback (const EmberAfOtaImageId * id, EmberAfOtaHeader * returnData)

Ota Storage Get Full Header.

This callback populates the **EmberAfOtaHeader** structure pointed to by the returnData with data about the OTA file stored in the storage module.

Parameters:

id	This is a pointer to the image id for the OTA file to retrieve information about. Ver.: always
returnData	This is a pointer to the location of the structure that will be populated with data. Ver.: always

EmberAfOtaStorageStatus emberAfOtaStorageReadImageDataCallback (const EmberAfOtaImageId * id, int32u offset, int32u length, int8u * returnData, int32u * returnedLength)

Ota Storage Read Image Data.

This callback reads data from the specified OTA file and returns that data to the caller.

Parameters:

id	This is a pointer to the image id for the OTA file to retrieve data from. Ver.: always
offset	This is the offset relative to the start of the image where the data should be read from. Ver.: always
length	This is the length of data that will be read. Ver.: always
returnData	This is a pointer to where the data read out of the file will be written to Ver.: always
returnedLength	This is a pointer to a variable where the actual length of data read will be written to. A short read may occur if the end of file was reached. Ver.: always

EmberAfOtaStorageStatus emberAfOtaStorageClearTempDataCallback (void)

Ota Storage Clear Temp Data.

This function clears any existing temp data that was downloaed. It is used immediately prior to downloading a raw image over the air.

EmberAfOtaStorageStatus emberAfOtaStorageWriteTempDataCallback (int32u offset, int32u length, const int8u * data)

Ota Storage Write Temp Data.

This function writes to the temporary data in the storage device at the specified offset. It is used when downloading a raw image over the air.

Parameters:

offset	The location within the download image file where to write the data. Ver.: always
length	The length of data to write. Ver.: always
data	A pointer to the temporary data that will be written to the storage device. Ver.: always

EmberAfOtaStorageStatus emberAfOtaStorageCheckTempDataCallback (int32u * currentOffset, int32u * totalImageSize, EmberAfOtaImageId * newFileInfo)

Ota Storage Check Temp Data.

This callback will validate temporary data in the storage device to determine whether it is a complete file, a partially downloaded file, or there is no file present. When a complete or partial file is found it will return EMBER_AF_OTA_STORAGE_SUCCESS or EMBER_AF_OTA_STORAGE_PARTIAL_FILE_FOUND, respectively. In that case, the currentOffset, totalImageSize, and newFileInfo will be populated with data. When EMBER_AF_OTA_STORAGE_ERROR is returned, no temporary data is present.

Parameters:

currentOffset	A pointer to a value that will be written with the offset within the total file size that has been successfully stored in the storage device. This will indicate how much data has been currently dowloaded. Ver.: always
totalImageSize	A pointer to a value that will be written with the total image size of the OTA file when a download has completed. This does not indicate how much data has actually been downloaded currently. Ver.: always
newFileInfo	This is the image id of the temporary file data stored in the storage device. Ver.: always

EmberAfOtaStorageStatus emberAfOtaStorageFinishDownloadCallback (int32u offset)

Ota Storage Finish Download.

This function indicates to the storage module that the download has finished.

Parameters:

offset The final offset of the downloaded file (i.e. the total size) Ver.: always

boolean emberAfOtaClientIncomingMessageRawCallback (EmberAfClusterCommand * message)

Ota Client Incoming Message Raw.

This callback is for processing incoming messages for the Over-the-air bootload cluster client. ZCL will not process the message and instead hand the raw over the air data to the callback for its own processing.

Parameters:

message A pointer to the structure containing the message buffer and other information about it. Ver.: always

void emberAfOtaClientStartCallback (void)

Ota Client Start.

This callback should be called when the profile specific registration has completed successfully. It will start the client's state machine that will find the OTA server, query it for the next image, download the image, wait for the bootload message, and kick off the bootload.

void emberAfOtaClientVersionInfoCallback (EmberAfOtaImageId * currentImageInfo, int16u * hardwareVersion)

Ota Client Version Info.

This function is called by the OTA client when a new query will occur to the server asking what the next version of firmware is. The client can inform the cluster software as to what information to use in the query (and subsequent download).

Parameters:

currentImageInfo This is the information to use in the next query by the client cluster code. It contains the manufacturer ID, image type ID, and the firmware version to be specified in the query message sent to the server. Ver.: always

hardwareVersion This is a pointer to the hardware version to use in the query. If no hardware version should be used, then EMBER_AF_INVALID_HARDWARE_VERSION should be used. Ver.: always

EmberAfImageVerifyStatus emberAfOtaClientCustomVerifyCallback (boolean newVerification,

```
                                                const EmberAfOtaImageId *  id
                                              )
```

Ota Client Custom Verify.

This callback is executed by the OTA client after the signature verification has successfully completed. It allows the device to do its own custom verification of the image (such as verifying that the EBL is intact).

Parameters:

newVerification	This indicates if a new verification should be started. Ver.: always
id	This is ID of the image to be verified. Ver.: always

```
boolean emberAfOtaClientDownloadCompleteCallback ( boolean                    success,
                                                   const EmberAfOtaImageId *  id
                                                 )
```

Ota Client Download Complete.

This callback indicates that the OTA client has completed the download of a file. If the file has been completely downloaded and cryptographic checks have been turned on, then those will be performed prior to this callback and that outcome included in the 'success' result. On failure, this callback is merely informative, and the return type is ignored. On succesful download, this callback allows the client to perform any additional verification of the downloaded image and return that result to the OTA server.

Parameters:

success	This indicates the success or failure of the download and cryptographic verification process (if applicable). Ver.: always
id	This is the image identifier information that corresponds to the download result. Ver.: always

```
void emberAfOtaClientBootloadCallback ( const EmberAfOtaImageId *  id  )
```

Ota Client Bootload.

This callback is fired when the OTA Client recevies a command to bootload the newly downloaded OTA image. This callback will perform the platform specific to bootload their device.

Parameters:

id	This is the identifier relating to the image that has been downloaded and is ready for bootload. Ver.: always

```
int8u emberAfOtaServerQueryCallback ( const EmberAfOtaImageId *  currentImageId,
                                      int16u *                   hardwareVersion,
                                      EmberAfOtaImageId *        nextUpgradeImageId
                                    )
```

Ota Server Query.

This callback is fired when the OTA server receives a query request by the client. The callback lets the server application indicate to the client what the 'next' version of software is for the device, or if there is not one available.

Parameters:

currentImageId	This is the current software image that the client hase. Ver.: always
hardwareVersion	If this value is non-NULL, it indicates the hardware version of the client device. If NULL, the client did not specify a hardware version. Ver.: always
nextUpgradeImageId	This is a pointer to a data structure containing the 'next' software version for the client to download. Ver.: always

int8u emberAfOtaServerBlockSizeCallback (EmberNodeId clientNodeId)

Ota Server Block Size.

This function provides a way for the server to adjust the block size of its response to an Image block request by a client.

Parameters:

clientNodeId	The node Id of OTA client making an image block request. Ver.: always

```
int8u emberAfOtaServerImageBlockRequestCallback ( const EmberAfOtaImageId *  id,
                                                  int32u                     offset,
                                                  int8u                      maxDataSize,
                                                  int32u *                   waitTimeMinutes
                                                )
```

Ota Server Image Block Request.

This function is called when the server application receives an image block request by a client.

Parameters:

id	This structure contains the image Id that the client is requesting a block from. Ver.: always
offset	This is the offset within the OTA file that the client is requesting. Ver.: always
maxDataSize	This is the maximum size that the client has requested for the data block sent in the response. Ver.: always
waitTimeMinutes	If the function returns EMBER_ZCL_STATUS_WAIT_FOR_DATA, then this is a pointer to the delay time in minutes that will be sent back to the client. Ver.: always

```
boolean emberAfOtaServerUpgradeEndRequestCallback ( EmberNodeId    source,
                                                    int8u          status,
                                                    int32u *       returnValue,
```

```
                                    const EmberAfOtaImageId * imageId
                                    )
```

Ota Server Upgrade End Request.

This function is called when the OTA server receives a request an upgrade end request. If the request indicated a successful download by the client, the server must tell the client when and if to upgrade to the downloaded image.

Parameters:

source	The node ID of the device that sent the upgrade end request. Ver.: always
status	This is the ZCL status sent by the client indicating the result of its attempt to download the new upgrade image. If the status is not EMBER_ZCL_STATUS_SUCCESS then this callback is merely informative and no response mesasge will be generated by the server. Ver.: always
returnValue	If the server returns TRUE indicating that the client should apply the upgrade, this time value indicates when in the future the client should apply the upgrade. Ver.: always
imageId	This variable indicates the software version that the client successfully downloaded and is asking to upgrade to. Ver.: always

int8u emberAfOtaPageRequestServerPolicyCallback (void)

Ota Page Request Server Policy.

This callback is called by the OTA server page request code when it wants to determine if it is allowed for an OTA client to make a page request. It is only called if page request support has been enabled on the server. It should return EMBER_ZCL_STATUS_SUCCESS if it allows the page request, and EMBER_ZCL_STATUS_UNSUP_CLUSTER_COMMAND if it does not want to allow it.

```
EmberStatus emberAfInterpanSendMessageCallback ( EmberAfInterpanHeader * header,
                                                 int16u                  messageLength,
                                                 int8u *                 message
                                               )
```

Interpan Send Message.

This function will send a raw MAC message with interpan frame format using the passed parameters.

Parameters:

header	Interpan header info Ver.: always
messageLength	The length of the message received or to send Ver.: always
message	The message data received or to send. Ver.: always

void emberAfInterpanInitCallback (void)

Interpan Init.

Initializes the interpan functionality.

```
void emberAfUnusedPanIdFoundCallback ( EmberPanId  panId,
                                       int8u       channel
                                     )
```

Unused Pan Id Found.

This is called by the framework on behalf of the form-and-join library to notify the application of the PAN id and channel found following a call to emberScanForUnusedPanId(). See form-and-join documentation for more information.

Parameters:

panId	Ver.: always
channel	Ver.: always

```
void emberAfJoinableNetworkFoundCallback ( EmberZigbeeNetwork *  networkFound,
                                           int8u                 lqi,
                                           int8s                 rssi
                                         )
```

Joinable Network Found.

This is called by the framework on behalf of the form-and-join library to notify the application of the network found after a call to emberScanForJoinableNetwork() or emberScanForNextJoinableNetwork(). See form-and-join documentation for more information.

Parameters:

networkFound	Ver.: always
lqi	Ver.: always
rssi	Ver.: always

```
void emberAfScanErrorCallback ( EmberStatus  status )
```

Scan Error.

This is called by the framework on behalf of the form-and-join library to notify the application if an error occurs while scanning. See form-and-join documentation for more information.

Parameters:

status The status of the scan. Ver.: always

EmberStatus emberAfFindUnusedPanIdAndFormCallback (void)

Find Unused Pan Id And Form.

This function is called by the framework to search for an unused PAN id and form a new network. The application should return EMBER_SUCCESS if the operation was initiated successfully.

EmberStatus emberAfStartSearchForJoinableNetworkCallback (void)

Start Search For Joinable Network.

This function is called by the framework to search for joinable networks and join a network. The application should return EMBER_SUCCESS if the operation was initiated successfully.

void emberAfContinueJoinableNetworkSearchCallback (void)

Continue Joinable Network Search.

This is called by the framework to instruct the application to continue its joinable network search.

void emberAfNetworkFindOperationCompleteCallback (void)

Network Find Operation Complete.

This is called by the framework when the network find operation is complete.

boolean emberAfBasicClusterResetToFactoryDefaultsCallback (void)

Reset To Factory Defaults.

void emberAfBasicClusterServerInitCallback (int8u endpoint)

Server Init.

Basic cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfBasicClusterClientInitCallback (int8u endpoint)

Client Init.

Basic cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfBasicClusterServerTickCallback (int8u endpoint)

Server Tick.

Basic cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfBasicClusterClientTickCallback (int8u endpoint)

Client Tick.

Basic cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfBasicClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Basic cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfBasicClusterClientAttributeChangedCallback (int8u endpoint,

```
                                    EmberAfAttributeId attributeId
                                  )
```

Client Attribute Changed.

Basic cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfBasicClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                             EmberAfAttributeId attributeId,
                                                                             int16u             manufacturerCode
                                                                           )
```

Server Manufacturer Specific Attribute Changed.

Basic cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfBasicClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                             EmberAfAttributeId attributeId,
                                                                             int16u             manufacturerCode
                                                                           )
```

Client Manufacturer Specific Attribute Changed.

Basic cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfBasicClusterServerDefaultResponseCallback ( int8u    endpoint,
                                                        int8u    commandId,
```

```
                                                EmberAfStatus status
                                              )
```

Server Default Response.

Basic cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfBasicClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                        int8u          commandId,
                                                        EmberAfStatus  status
                                                      )
```

Client Default Response.

Basic cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfBasicClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                    int16u                    indexOrDestination,
                                                    EmberApsFrame *           apsFrame,
                                                    int16u                    msgLen,
                                                    int8u *                   message,
                                                    EmberStatus               status
                                                  )
```

Server Message Sent.

Basic cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always

msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfBasicClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                    int16u                   indexOrDestination,
                                                    EmberApsFrame *          apsFrame,
                                                    int16u                   msgLen,
                                                    int8u *                  message,
                                                    EmberStatus              status
                                                  )
```

Client Message Sent.

Basic cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfPowerConfigClusterServerInitCallback ( int8u endpoint )
```

Server Init.

Power Configuration cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

```
void emberAfPowerConfigClusterClientInitCallback ( int8u endpoint )
```

Client Init.

Power Configuration cluster, Client Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfPowerConfigClusterServerTickCallback (int8u endpoint)

Server Tick.

Power Configuration cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfPowerConfigClusterClientTickCallback (int8u endpoint)

Client Tick.

Power Configuration cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfPowerConfigClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Power Configuration cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfPowerConfigClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Power Configuration cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

```
void emberAfPowerConfigClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                   EmberAfAttributeId attributeId,
                                                                                   int16u             manufacturerCode
                                                                                 )
```

Server Manufacturer Specific Attribute Changed.

Power Configuration cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfPowerConfigClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                   EmberAfAttributeId attributeId,
                                                                                   int16u             manufacturerCode
                                                                                 )
```

Client Manufacturer Specific Attribute Changed.

Power Configuration cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfPowerConfigClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                              int8u          commandId,
                                                              EmberAfStatus  status
                                                            )
```

Server Default Response.

Power Configuration cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always

status	Status in default response Ver.: always

```
void emberAfPowerConfigClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                              int8u          commandId,
                                                              EmberAfStatus  status
                                                            )
```

Client Default Response.

Power Configuration cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfPowerConfigClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                          int16u                    indexOrDestination,
                                                          EmberApsFrame *           apsFrame,
                                                          int16u                    msgLen,
                                                          int8u *                   message,
                                                          EmberStatus               status
                                                        )
```

Server Message Sent.

Power Configuration cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfPowerConfigClusterClientMessageSentCallback ( EmberOutgoingMessageType  type,
                                                          int16u                    indexOrDestination,
                                                          EmberApsFrame *           apsFrame,
                                                          int16u                    msgLen,
                                                          int8u *                   message,
```

EmberStatus status
)

Client Message Sent.

Power Configuration cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

void emberAfDeviceTempClusterServerInitCallback (int8u endpoint)

Server Init.

Device Temperature Configuration cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfDeviceTempClusterClientInitCallback (int8u endpoint)

Client Init.

Device Temperature Configuration cluster, Client Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfDeviceTempClusterServerTickCallback (int8u endpoint)

Server Tick.

Device Temperature Configuration cluster, Server Tick

Parameters:

endpoint	Endpoint that is being served Ver.: always

void emberAfDeviceTempClusterClientTickCallback (int8u endpoint)

Client Tick.

Device Temperature Configuration cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfDeviceTempClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Device Temperature Configuration cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfDeviceTempClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Device Temperature Configuration cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfDeviceTempClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Server Manufacturer Specific Attribute Changed.

Device Temperature Configuration cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfDeviceTempClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                  EmberAfAttributeId attributeId,
                                                                                  int16u             manufacturerCode
                                                                                )
```

Client Manufacturer Specific Attribute Changed.

Device Temperature Configuration cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfDeviceTempClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                             int8u          commandId,
                                                             EmberAfStatus  status
                                                           )
```

Server Default Response.

Device Temperature Configuration cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfDeviceTempClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                             int8u          commandId,
                                                             EmberAfStatus  status
                                                           )
```

Client Default Response.

Device Temperature Configuration cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfDeviceTempClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                         int16u                    indexOrDestination,
                                                         EmberApsFrame *           apsFrame,
                                                         int16u                    msgLen,
                                                         int8u *                   message,
                                                         EmberStatus               status
                                                       )
```

Server Message Sent.

Device Temperature Configuration cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfDeviceTempClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                         int16u                    indexOrDestination,
                                                         EmberApsFrame *           apsFrame,
                                                         int16u                    msgLen,
                                                         int8u *                   message,
                                                         EmberStatus               status
                                                       )
```

Client Message Sent.

Device Temperature Configuration cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always

apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

boolean emberAfIdentifyClusterIdentifyCallback (int16u identifyTime)

Identify.

Parameters:

identifyTime Ver.: always

boolean emberAfIdentifyClusterIdentifyQueryCallback (void)

Identify Query.

boolean emberAfIdentifyClusterIdentifyQueryResponseCallback (int16u timeout)

Identify Query Response.

Parameters:

timeout Ver.: always

void emberAfIdentifyClusterServerInitCallback (int8u endpoint)

Server Init.

Identify cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfIdentifyClusterClientInitCallback (int8u endpoint)

Client Init.

Identify cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfIdentifyClusterServerTickCallback (int8u endpoint)

Server Tick.

Identify cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfIdentifyClusterClientTickCallback (int8u endpoint)

Client Tick.

Identify cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfIdentifyClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Identify cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfIdentifyClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Identify cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

```
void emberAfIdentifyClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                EmberAfAttributeId attributeId,
                                                                                int16u             manufacturerCode
                                                                              )
```

Server Manufacturer Specific Attribute Changed.

Identify cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfIdentifyClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                EmberAfAttributeId attributeId,
                                                                                int16u             manufacturerCode
                                                                              )
```

Client Manufacturer Specific Attribute Changed.

Identify cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfIdentifyClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                           int8u          commandId,
                                                           EmberAfStatus  status
                                                         )
```

Server Default Response.

Identify cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always

status	Status in default response Ver.: always

```
void emberAfIdentifyClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                           int8u          commandId,
                                                           EmberAfStatus  status
                                                         )
```

Client Default Response.

Identify cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfIdentifyClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                       int16u                    indexOrDestination,
                                                       EmberApsFrame *           apsFrame,
                                                       int16u                    msgLen,
                                                       int8u *                   message,
                                                       EmberStatus               status
                                                     )
```

Server Message Sent.

Identify cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfIdentifyClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                       int16u                    indexOrDestination,
                                                       EmberApsFrame *           apsFrame,
                                                       int16u                    msgLen,
                                                       int8u *                   message,
```

```
                                                EmberStatus          status
                                              )
```

Client Message Sent.

Identify cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfGroupsClusterEndpointInGroupCallback ( int8u   endpoint,
                                                      int16u  groupId
                                                    )
```

Endpoint In Group.

This function is called by the framework when it needs to determine if an endpoint is a member of a group.
The application should return TRUE if the endpoint is a member of the group and FALSE otherwise.

Parameters:

endpoint	The endpoint. Ver.: always
groupId	The group identifier. Ver.: always

```
void emberAfGroupsClusterClearGroupTableCallback ( void   )
```

Clear Group Table.

This function is called by the framework when the application should clear the group table.

```
boolean emberAfGroupsClusterAddGroupCallback ( int16u  groupId,
                                               int8u *  groupName
                                             )
```

Add Group.

Parameters:

groupId	Ver.: always

groupName Ver.: always

boolean emberAfGroupsClusterViewGroupCallback (int16u groupId)

View Group.

Parameters:

groupId Ver.: always

boolean emberAfGroupsClusterGetGroupMembershipCallback (int8u groupCount, int8u * groupList)

Get Group Membership.

Parameters:

groupCount Ver.: always

groupList Ver.: always

boolean emberAfGroupsClusterRemoveGroupCallback (int16u groupId)

Remove Group.

Parameters:

groupId Ver.: always

boolean emberAfGroupsClusterRemoveAllGroupsCallback (void)

Remove All Groups.

boolean emberAfGroupsClusterAddGroupIfIdentifyingCallback (int16u groupId, int8u * groupName)

Add Group If Identifying.

Parameters:

groupId Ver.: always

groupName Ver.: always

```
boolean emberAfGroupsClusterAddGroupResponseCallback ( int8u   status,
                                                       int16u  groupId
                                                     )
```

Add Group Response.

Parameters:

status Ver.: always

groupId Ver.: always

```
boolean emberAfGroupsClusterViewGroupResponseCallback ( int8u   status,
                                                        int16u  groupId,
                                                        int8u * groupName
                                                      )
```

View Group Response.

Parameters:

status Ver.: always

groupId Ver.: always

groupName Ver.: always

```
boolean emberAfGroupsClusterGetGroupMembershipResponseCallback ( int8u   capacity,
                                                                 int8u   groupCount,
                                                                 int8u * groupList
                                                               )
```

Get Group Membership Response.

Parameters:

capacity Ver.: always

groupCount Ver.: always

groupList Ver.: always

boolean emberAfGroupsClusterRemoveGroupResponseCallback (int8u status, int16u groupId)

Remove Group Response.

Parameters:

status Ver.: always

groupId Ver.: always

void emberAfGroupsClusterServerInitCallback (int8u endpoint)

Server Init.

Groups cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfGroupsClusterClientInitCallback (int8u endpoint)

Client Init.

Groups cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfGroupsClusterServerTickCallback (int8u endpoint)

Server Tick.

Groups cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfGroupsClusterClientTickCallback (int8u endpoint)

Client Tick.

Groups cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

```
void emberAfGroupsClusterServerAttributeChangedCallback ( int8u              endpoint,
                                                          EmberAfAttributeId attributeId
                                                        )
```

Server Attribute Changed.

Groups cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

```
void emberAfGroupsClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                          EmberAfAttributeId attributeId
                                                        )
```

Client Attribute Changed.

Groups cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

```
void emberAfGroupsClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                              EmberAfAttributeId attributeId,
                                                                              int16u             manufacturerCode
                                                                            )
```

Server Manufacturer Specific Attribute Changed.

Groups cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfGroupsClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                              EmberAfAttributeId attributeId,
                                                                              int16u             manufacturerCode
                                                                            )
```

Client Manufacturer Specific Attribute Changed.

Groups cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfGroupsClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                         int8u          commandId,
                                                         EmberAfStatus  status
                                                       )
```

Server Default Response.

Groups cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfGroupsClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                         int8u          commandId,
                                                         EmberAfStatus  status
                                                       )
```

Client Default Response.

Groups cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfGroupsClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                     int16u                    indexOrDestination,
                                                     EmberApsFrame *           apsFrame,
                                                     int16u                    msgLen,
                                                     int8u *                   message,
                                                     EmberStatus               status
                                                   )
```

Server Message Sent.

Groups cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfGroupsClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                     int16u                    indexOrDestination,
                                                     EmberApsFrame *           apsFrame,
                                                     int16u                    msgLen,
                                                     int8u *                   message,
                                                     EmberStatus               status
                                                   )
```

Client Message Sent.

Groups cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always

status	The status of the sent message Ver.: always

EmberAfStatus emberAfScenesClusterStoreCurrentSceneCallback (int8u endpoint, int16u groupId, int8u sceneId)

Store Current Scene.

This function is called by the framework when the application should store the current scene. If an entry already exists in the scene table with the same scene and group ids, the application should update the entry with the current scene. Otherwise, a new entry should be adde to the scene table, if possible.

Parameters:

endpoint	The endpoint. Ver.: always
groupId	The group identifier. Ver.: always
sceneId	The scene identifier. Ver.: always

EmberAfStatus emberAfScenesClusterRecallSavedSceneCallback (int8u endpoint, int16u groupId, int8u sceneId)

Recall Saved Scene.

This function is called by the framework when the application should recall a saved scene.

Parameters:

endpoint	The endpoint. Ver.: always
groupId	The group identifier. Ver.: always
sceneId	The scene identifier. Ver.: always

void emberAfScenesClusterClearSceneTableCallback (void)

ClearSceneTable.

This function is called by the framework when the application should clear the scene table.

boolean emberAfScenesClusterAddSceneCallback (int16u groupId, int8u sceneId,

```
                                                    int16u  transitionTime,
                                                    int8u * sceneName,
                                                    int8u * extensionFieldSets
                                                  )
```

Add Scene.

Parameters:

groupId	Ver.: always
sceneId	Ver.: always
transitionTime	Ver.: always
sceneName	Ver.: always
extensionFieldSets	Ver.: always

```
boolean emberAfScenesClusterViewSceneCallback ( int16u  groupId,
                                                int8u   sceneId
                                              )
```

View Scene.

Parameters:

groupId	Ver.: always
sceneId	Ver.: always

```
boolean emberAfScenesClusterRemoveSceneCallback ( int16u  groupId,
                                                  int8u   sceneId
                                                )
```

Remove Scene.

Parameters:

groupId	Ver.: always
sceneId	Ver.: always

```
boolean emberAfScenesClusterRemoveAllScenesCallback ( int16u  groupId )
```

Remove All Scenes.

Parameters:

groupId Ver.: always

```
boolean emberAfScenesClusterStoreSceneCallback ( int16u groupId,
                                                 int8u  sceneId
                                               )
```

Store Scene.

Parameters:

groupId Ver.: always

sceneId Ver.: always

```
boolean emberAfScenesClusterRecallSceneCallback ( int16u groupId,
                                                  int8u  sceneId
                                                )
```

Recall Scene.

Parameters:

groupId Ver.: always

sceneId Ver.: always

```
boolean emberAfScenesClusterGetSceneMembershipCallback ( int16u groupId )
```

Get Scene Membership.

Parameters:

groupId Ver.: always

```
boolean emberAfScenesClusterAddSceneResponseCallback ( int8u  status,
                                                       int16u groupId,
                                                       int8u  sceneId
                                                     )
```

Add Scene Response.

Parameters:

status	Ver.: always
groupId	Ver.: always
sceneId	Ver.: always

```
boolean emberAfScenesClusterViewSceneResponseCallback ( int8u   status,
                                                        int16u  groupId,
                                                        int8u   sceneId,
                                                        int16u  transitionTime,
                                                        int8u * sceneName,
                                                        int8u * extensionFieldSets
                                                      )
```

View Scene Response.

Parameters:

status	Ver.: always
groupId	Ver.: always
sceneId	Ver.: always
transitionTime	Ver.: always
sceneName	Ver.: always
extensionFieldSets	Ver.: always

```
boolean emberAfScenesClusterRemoveSceneResponseCallback ( int8u   status,
                                                          int16u  groupId,
                                                          int8u   sceneId
                                                        )
```

Remove Scene Response.

Parameters:

status	Ver.: always
groupId	Ver.: always
sceneId	Ver.: always

```
boolean emberAfScenesClusterRemoveAllScenesResponseCallback ( int8u   status,
                                                              int16u  groupId
                                                            )
```

Remove All Scenes Response.

Parameters:

status	Ver.: always
groupId	Ver.: always

```
boolean emberAfScenesClusterStoreSceneResponseCallback ( int8u   status,
                                                         int16u  groupId,
                                                         int8u   sceneId
                                                       )
```

Store Scene Response.

Parameters:

status	Ver.: always
groupId	Ver.: always
sceneId	Ver.: always

```
boolean emberAfScenesClusterGetSceneMembershipResponseCallback ( int8u   status,
                                                                 int8u   capacity,
                                                                 int16u  groupId,
                                                                 int8u   sceneCount,
                                                                 int8u * sceneList
                                                               )
```

Get Scene Membership Response.

Parameters:

status	Ver.: always
capacity	Ver.: always
groupId	Ver.: always
sceneCount	Ver.: always

sceneList Ver.: always

void emberAfScenesClusterServerInitCallback (int8u endpoint)

Server Init.

Scenes cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfScenesClusterClientInitCallback (int8u endpoint)

Client Init.

Scenes cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfScenesClusterServerTickCallback (int8u endpoint)

Server Tick.

Scenes cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfScenesClusterClientTickCallback (int8u endpoint)

Client Tick.

Scenes cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfScenesClusterServerAttributeChangedCallback (int8u endpoint,

```
                                    EmberAfAttributeId attributeId
                                  )
```

Server Attribute Changed.

Scenes cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfScenesClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                          EmberAfAttributeId attributeId
                                                        )
```

Client Attribute Changed.

Scenes cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfScenesClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                              EmberAfAttributeId attributeId,
                                                                              int16u             manufacturerCode
                                                                            )
```

Server Manufacturer Specific Attribute Changed.

Scenes cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfScenesClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                              EmberAfAttributeId attributeId,
                                                                              int16u             manufacturerCode
                                                                            )
```

Client Manufacturer Specific Attribute Changed.

Scenes cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfScenesClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                         int8u          commandId,
                                                         EmberAfStatus  status
                                                       )
```

Server Default Response.

Scenes cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfScenesClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                         int8u          commandId,
                                                         EmberAfStatus  status
                                                       )
```

Client Default Response.

Scenes cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfScenesClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                     int16u                     indexOrDestination,
                                                     EmberApsFrame *            apsFrame,
                                                     int16u                     msgLen,
```

```
                                                    int8u *                    message,
                                                    EmberStatus                status
                                                  )
```

Server Message Sent.

Scenes cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfScenesClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                     int16u                     indexOrDestination,
                                                     EmberApsFrame *            apsFrame,
                                                     int16u                     msgLen,
                                                     int8u *                    message,
                                                     EmberStatus                status
                                                   )
```

Client Message Sent.

Scenes cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

boolean emberAfOnOffClusterOffCallback (void)

Off.

boolean emberAfOnOffClusterOnCallback (void)

On.

boolean emberAfOnOffClusterToggleCallback (void)

Toggle.

void emberAfOnOffClusterServerInitCallback (int8u endpoint)

Server Init.

On/off cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfOnOffClusterClientInitCallback (int8u endpoint)

Client Init.

On/off cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfOnOffClusterServerTickCallback (int8u endpoint)

Server Tick.

On/off cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfOnOffClusterClientTickCallback (int8u endpoint)

Client Tick.

On/off cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

```
void emberAfOnOffClusterServerAttributeChangedCallback ( int8u              endpoint,
                                                         EmberAfAttributeId attributeId
                                                       )
```

Server Attribute Changed.

On/off cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfOnOffClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                         EmberAfAttributeId attributeId
                                                       )
```

Client Attribute Changed.

On/off cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfOnOffClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                             EmberAfAttributeId attributeId,
                                                                             int16u             manufacturerCode
                                                                           )
```

Server Manufacturer Specific Attribute Changed.

On/off cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfOnOffClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                             EmberAfAttributeId attributeId,
```

```
                                                    int16u          manufacturerCode
                                                  )
```

Client Manufacturer Specific Attribute Changed.

On/off cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfOnOffClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                        int8u          commandId,
                                                        EmberAfStatus  status
                                                      )
```

Server Default Response.

On/off cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfOnOffClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                        int8u          commandId,
                                                        EmberAfStatus  status
                                                      )
```

Client Default Response.

On/off cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfOnOffClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
```

```
                                                int16u                  indexOrDestination,
                                                EmberApsFrame *         apsFrame,
                                                int16u                  msgLen,
                                                int8u *                 message,
                                                EmberStatus             status
                                              )
```

Server Message Sent.

On/off cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfOnOffClusterClientMessageSentCallback ( EmberOutgoingMessageType  type,
                                                    int16u                    indexOrDestination,
                                                    EmberApsFrame *           apsFrame,
                                                    int16u                    msgLen,
                                                    int8u *                   message,
                                                    EmberStatus               status
                                                  )
```

Client Message Sent.

On/off cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfOnOffClusterSampleMfgSpecificOffWithTransitionCallback ( void )
```

Sample Mfg Specific Off With Transition.

boolean emberAfOnOffClusterSampleMfgSpecificOnWithTransitionCallback (void)

Sample Mfg Specific On With Transition.

boolean emberAfOnOffClusterSampleMfgSpecificToggleWithTransitionCallback (void)

Sample Mfg Specific Toggle With Transition.

void emberAfOnOffSwitchConfigClusterServerInitCallback (int8u endpoint)

Server Init.

On/off Switch Configuration cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfOnOffSwitchConfigClusterClientInitCallback (int8u endpoint)

Client Init.

On/off Switch Configuration cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfOnOffSwitchConfigClusterServerTickCallback (int8u endpoint)

Server Tick.

On/off Switch Configuration cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfOnOffSwitchConfigClusterClientTickCallback (int8u endpoint)

Client Tick.

On/off Switch Configuration cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

---

void emberAfOnOffSwitchConfigClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

On/off Switch Configuration cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

---

void emberAfOnOffSwitchConfigClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

On/off Switch Configuration cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

---

void emberAfOnOffSwitchConfigClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Server Manufacturer Specific Attribute Changed.

On/off Switch Configuration cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

manufacturerCode Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfOnOffSwitchConfigClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                          EmberAfAttributeId attributeId,
                                                                                          int16u             manufacturerCode
                                                                                        )
```

Client Manufacturer Specific Attribute Changed.

On/off Switch Configuration cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfOnOffSwitchConfigClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                     int8u          commandId,
                                                                     EmberAfStatus  status
                                                                   )
```

Server Default Response.

On/off Switch Configuration cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfOnOffSwitchConfigClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                     int8u          commandId,
                                                                     EmberAfStatus  status
                                                                   )
```

Client Default Response.

On/off Switch Configuration cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfOnOffSwitchConfigClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                                int16u                     indexOrDestination,
                                                                EmberApsFrame *            apsFrame,
                                                                int16u                     msgLen,
                                                                int8u *                    message,
                                                                EmberStatus                status
                                                              )
```

Server Message Sent.

On/off Switch Configuration cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfOnOffSwitchConfigClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                                int16u                     indexOrDestination,
                                                                EmberApsFrame *            apsFrame,
                                                                int16u                     msgLen,
                                                                int8u *                    message,
                                                                EmberStatus                status
                                                              )
```

Client Message Sent.

On/off Switch Configuration cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfLevelControlClusterMoveToLevelCallback ( int8u   level,
                                                        int16u  transitionTime
```

```
)
```

Move To Level.

Parameters:

level	Ver.: always
transitionTime	Ver.: always

```
boolean emberAfLevelControlClusterMoveCallback ( int8u  moveMode,
                                                 int8u  rate
                                               )
```

Move.

Parameters:

moveMode	Ver.: always
rate	Ver.: always

```
boolean emberAfLevelControlClusterStepCallback ( int8u   stepMode,
                                                 int8u   stepSize,
                                                 int16u  transitionTime
                                               )
```

Step.

Parameters:

stepMode	Ver.: always
stepSize	Ver.: always
transitionTime	Ver.: always

```
boolean emberAfLevelControlClusterStopCallback ( void   )
```

Stop.

```
boolean emberAfLevelControlClusterMoveToLevelWithOnOffCallback ( int8u   level,
                                                                 int16u  transitionTime
                                                               )
```

Move To Level With On Off.

Parameters:

level	Ver.: always
transitionTime	Ver.: always

boolean emberAfLevelControlClusterMoveWithOnOffCallback (int8u moveMode, int8u rate)

Move With On Off.

Parameters:

moveMode	Ver.: always
rate	Ver.: always

boolean emberAfLevelControlClusterStepWithOnOffCallback (int8u stepMode, int8u stepSize, int16u transitionTime)

Step With On Off.

Parameters:

stepMode	Ver.: always
stepSize	Ver.: always
transitionTime	Ver.: always

boolean emberAfLevelControlClusterStopWithOnOffCallback (void)

Stop With On Off.

void emberAfLevelControlClusterServerInitCallback (int8u endpoint)

Server Init.

Level Control cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfLevelControlClusterClientInitCallback (int8u endpoint)

Client Init.

Level Control cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfLevelControlClusterServerTickCallback (int8u endpoint)

Server Tick.

Level Control cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfLevelControlClusterClientTickCallback (int8u endpoint)

Client Tick.

Level Control cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfLevelControlClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Level Control cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

```
void emberAfLevelControlClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                                EmberAfAttributeId attributeId
                                                              )
```

Client Attribute Changed.

Level Control cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

```
void emberAfLevelControlClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                     EmberAfAttributeId attributeId,
                                                                                     int16u             manufacturerCode
                                                                                   )
```

Server Manufacturer Specific Attribute Changed.

Level Control cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

manufacturerCode Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfLevelControlClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                    EmberAfAttributeId attributeId,
                                                                                    int16u             manufacturerCode
                                                                                  )
```

Client Manufacturer Specific Attribute Changed.

Level Control cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

manufacturerCode Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfLevelControlClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                               int8u          commandId,
                                                               EmberAfStatus  status
                                                             )
```

Server Default Response.

Level Control cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfLevelControlClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                               int8u          commandId,
                                                               EmberAfStatus  status
                                                             )
```

Client Default Response.

Level Control cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfLevelControlClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                           int16u                    indexOrDestination,
                                                           EmberApsFrame *           apsFrame,
                                                           int16u                    msgLen,
                                                           int8u *                   message,
                                                           EmberStatus               status
                                                         )
```

Server Message Sent.

Level Control cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always

indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfLevelControlClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                           int16u                    indexOrDestination,
                                                           EmberApsFrame *           apsFrame,
                                                           int16u                    msgLen,
                                                           int8u *                   message,
                                                           EmberStatus               status
                                                         )
```

Client Message Sent.

Level Control cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfAlarmClusterResetAlarmCallback ( int8u  alarmCode,
                                                int16u clusterId
                                              )
```

Reset Alarm.

Parameters:

alarmCode	Ver.: always
clusterId	Ver.: always

```
boolean emberAfAlarmClusterResetAllAlarmsCallback ( void )
```

Reset All Alarms.

boolean emberAfAlarmClusterGetAlarmCallback (void)

Get Alarm.

boolean emberAfAlarmClusterResetAlarmLogCallback (void)

Reset Alarm Log.

boolean emberAfAlarmClusterAlarmCallback (int8u alarmCode, int16u clusterId)

Alarm.

Parameters:

alarmCode	Ver.: always
clusterId	Ver.: always

boolean emberAfAlarmClusterGetAlarmResponseCallback (int8u status, int8u alarmCode, int16u clusterId, int32u timeStamp)

Get Alarm Response.

Parameters:

status	Ver.: always
alarmCode	Ver.: always
clusterId	Ver.: always
timeStamp	Ver.: always

void emberAfAlarmClusterServerInitCallback (int8u endpoint)

Server Init.

Alarms cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfAlarmClusterClientInitCallback (int8u endpoint)

Client Init.

Alarms cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfAlarmClusterServerTickCallback (int8u endpoint)

Server Tick.

Alarms cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfAlarmClusterClientTickCallback (int8u endpoint)

Client Tick.

Alarms cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfAlarmClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Alarms cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

```
void emberAfAlarmClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                         EmberAfAttributeId attributeId
                                                       )
```

Client Attribute Changed.

Alarms cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

```
void emberAfAlarmClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                             EmberAfAttributeId attributeId,
                                                                             int16u             manufacturerCode
                                                                           )
```

Server Manufacturer Specific Attribute Changed.

Alarms cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

manufacturerCode Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfAlarmClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                             EmberAfAttributeId attributeId,
                                                                             int16u             manufacturerCode
                                                                           )
```

Client Manufacturer Specific Attribute Changed.

Alarms cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

manufacturerCode Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfAlarmClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                        int8u          commandId,
                                                        EmberAfStatus  status
                                                      )
```

Server Default Response.

Alarms cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfAlarmClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                        int8u          commandId,
                                                        EmberAfStatus  status
                                                      )
```

Client Default Response.

Alarms cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfAlarmClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                    int16u                   indexOrDestination,
                                                    EmberApsFrame *          apsFrame,
                                                    int16u                   msgLen,
                                                    int8u *                  message,
                                                    EmberStatus              status
                                                  )
```

Server Message Sent.

Alarms cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always

indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

void emberAfAlarmClusterClientMessageSentCallback (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame * apsFrame, int16u msgLen, int8u * message, EmberStatus status)

Client Message Sent.

Alarms cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

void emberAfTimeClusterServerInitCallback (int8u endpoint)

Server Init.

Time cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfTimeClusterClientInitCallback (int8u endpoint)

Client Init.

Time cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfTimeClusterServerTickCallback (int8u endpoint)

Server Tick.

Time cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfTimeClusterClientTickCallback (int8u endpoint)

Client Tick.

Time cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfTimeClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Time cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfTimeClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Time cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfTimeClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                            EmberAfAttributeId attributeId,
                                                                            int16u             manufacturerCode
                                                                          )
```

Server Manufacturer Specific Attribute Changed.

Time cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfTimeClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                            EmberAfAttributeId attributeId,
                                                                            int16u             manufacturerCode
                                                                          )
```

Client Manufacturer Specific Attribute Changed.

Time cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfTimeClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                       int8u          commandId,
                                                       EmberAfStatus  status
                                                     )
```

Server Default Response.

Time cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfTimeClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                       int8u          commandId,
                                                       EmberAfStatus  status
                                                     )
```

Client Default Response.

Time cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfTimeClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                   int16u                    indexOrDestination,
                                                   EmberApsFrame *           apsFrame,
                                                   int16u                    msgLen,
                                                   int8u *                   message,
                                                   EmberStatus               status
                                                 )
```

Server Message Sent.

Time cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfTimeClusterClientMessageSentCallback ( EmberOutgoingMessageType  type,
                                                   int16u                    indexOrDestination,
                                                   EmberApsFrame *           apsFrame,
                                                   int16u                    msgLen,
```

```
                                                int8u *                    message,
                                                EmberStatus                status
                                              )
```

Client Message Sent.

Time cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfRssiLocationClusterSetAbsoluteLocationCallback ( int16s  coordinate1,
                                                                int16s  coordinate2,
                                                                int16s  coordinate3,
                                                                int16s  power,
                                                                int16u  pathLossExponent
                                                              )
```

Set Absolute Location.

Parameters:

coordinate1	Ver.: always
coordinate2	Ver.: always
coordinate3	Ver.: always
power	Ver.: always
pathLossExponent	Ver.: always

```
boolean emberAfRssiLocationClusterSetDeviceConfigurationCallback ( int16s  power,
                                                                   int16u  pathLossExponent,
                                                                   int16u  calculationPeriod,
                                                                   int8u   numberRssiMeasurements,
                                                                   int16u  reportingPeriod
                                                                 )
```

Set Device Configuration.

Parameters:

power	Ver.: always
pathLossExponent	Ver.: always
calculationPeriod	Ver.: always
numberRssiMeasurements	Ver.: always
reportingPeriod	Ver.: always

boolean emberAfRssiLocationClusterGetDeviceConfigurationCallback (int8u * targetAddress)

Get Device Configuration.

Parameters:

targetAddress	Ver.: always

boolean emberAfRssiLocationClusterGetLocationDataCallback (int8u flags, int8u numberResponses, int8u * targetAddress)

Get Location Data.

Parameters:

flags	Ver.: always
numberResponses	Ver.: always
targetAddress	Ver.: always

boolean emberAfRssiLocationClusterRssiResponseCallback (int8u * replyingDevice, int16s coordinate1, int16s coordinate2, int16s coordinate3, int8s rssi, int8u numberRssiMeasurements)

Rssi Response.

Parameters:

replyingDevice	Ver.: always
coordinate1	Ver.: always
coordinate2	Ver.: always
coordinate3	Ver.: always
rssi	Ver.: always
numberRssiMeasurements	Ver.: always

```
boolean emberAfRssiLocationClusterSendPingsCallback ( int8u *  targetAddress,
                                                      int8u    numberRssiMeasurements,
                                                      int16u   calculationPeriod
                                                    )
```

Send Pings.

Parameters:

targetAddress	Ver.: always
numberRssiMeasurements	Ver.: always
calculationPeriod	Ver.: always

```
boolean emberAfRssiLocationClusterAnchorNodeAnnounceCallback ( int8u *  anchorNodeIeeeAddress,
                                                               int16s   coordinate1,
                                                               int16s   coordinate2,
                                                               int16s   coordinate3
                                                             )
```

Anchor Node Announce.

Parameters:

anchorNodeIeeeAddress	Ver.: always
coordinate1	Ver.: always
coordinate2	Ver.: always
coordinate3	Ver.: always

```
boolean emberAfRssiLocationClusterDeviceConfigurationResponseCallback ( int8u   status,
                                                                        int16s  power,
                                                                        int16u  pathLossExponent,
                                                                        int16u  calculationPeriod,
                                                                        int8u   numberRssiMeasurements,
                                                                        int16u  reportingPeriod
                                                                      )
```

Device Configuration Response.

Parameters:

status	Ver.: always
power	Ver.: always
pathLossExponent	Ver.: always
calculationPeriod	Ver.: always
numberRssiMeasurements	Ver.: always
reportingPeriod	Ver.: always

```
boolean emberAfRssiLocationClusterLocationDataResponseCallback ( int8u   status,
                                                                 int8u   locationType,
                                                                 int16s  coordinate1,
                                                                 int16s  coordinate2,
                                                                 int16s  coordinate3,
                                                                 int16s  power,
                                                                 int16u  pathLossExponent,
                                                                 int8u   locationMethod,
                                                                 int8u   qualityMeasure,
                                                                 int16u  locationAge
                                                               )
```

Location Data Response.

Parameters:

status	Ver.: always
locationType	Ver.: always
coordinate1	Ver.: always

coordinate2	Ver.: always
coordinate3	Ver.: always
power	Ver.: always
pathLossExponent	Ver.: always
locationMethod	Ver.: always
qualityMeasure	Ver.: always
locationAge	Ver.: always

```
boolean emberAfRssiLocationClusterLocationDataNotificationCallback ( int8u   locationType,
                                                                      int16s  coordinate1,
                                                                      int16s  coordinate2,
                                                                      int16s  coordinate3,
                                                                      int16s  power,
                                                                      int16u  pathLossExponent,
                                                                      int8u   locationMethod,
                                                                      int8u   qualityMeasure,
                                                                      int16u  locationAge
                                                                    )
```

Location Data Notification.

Parameters:

locationType	Ver.: always
coordinate1	Ver.: always
coordinate2	Ver.: always
coordinate3	Ver.: always
power	Ver.: always
pathLossExponent	Ver.: always
locationMethod	Ver.: always
qualityMeasure	Ver.: always
locationAge	Ver.: always

```
boolean emberAfRssiLocationClusterCompactLocationDataNotificationCallback ( int8u   locationType,
                                                                            int16s  coordinate1,
                                                                            int16s  coordinate2,
                                                                            int16s  coordinate3,
                                                                            int8u   qualityMeasure,
                                                                            int16u  locationAge
                                                                          )
```

Compact Location Data Notification.

Parameters:

locationType	Ver.: always
coordinate1	Ver.: always
coordinate2	Ver.: always
coordinate3	Ver.: always
qualityMeasure	Ver.: always
locationAge	Ver.: always

```
boolean emberAfRssiLocationClusterRssiPingCallback ( int8u  locationType  )
```

Rssi Ping.

Parameters:

locationType	Ver.: always

```
boolean emberAfRssiLocationClusterRssiRequestCallback ( void   )
```

Rssi Request.

```
boolean emberAfRssiLocationClusterReportRssiMeasurementsCallback ( int8u *  measuringDevice,
                                                                    int8u    neighbors,
                                                                    int8u *  neighborsInfo
                                                                  )
```

Report Rssi Measurements.

Parameters:

measuringDevice Ver.: always
neighbors Ver.: always
neighborsInfo Ver.: always

boolean emberAfRssiLocationClusterRequestOwnLocationCallback (int8u * blindNode)

Request Own Location.

Parameters:
blindNode Ver.: always

void emberAfRssiLocationClusterServerInitCallback (int8u endpoint)

Server Init.

RSSI Location cluster, Server Init

Parameters:
endpoint Endpoint that is being initialized Ver.: always

void emberAfRssiLocationClusterClientInitCallback (int8u endpoint)

Client Init.

RSSI Location cluster, Client Init

Parameters:
endpoint Endpoint that is being initialized Ver.: always

void emberAfRssiLocationClusterServerTickCallback (int8u endpoint)

Server Tick.

RSSI Location cluster, Server Tick

Parameters:
endpoint Endpoint that is being served Ver.: always

void emberAfRssiLocationClusterClientTickCallback (int8u endpoint)

Client Tick.

RSSI Location cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfRssiLocationClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

RSSI Location cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfRssiLocationClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

RSSI Location cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfRssiLocationClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Server Manufacturer Specific Attribute Changed.

RSSI Location cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfRssiLocationClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                     EmberAfAttributeId attributeId,
                                                                                     int16u             manufacturerCode
                                                                                   )
```

Client Manufacturer Specific Attribute Changed.

RSSI Location cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfRssiLocationClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                               int8u          commandId,
                                                               EmberAfStatus  status
                                                             )
```

Server Default Response.

RSSI Location cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfRssiLocationClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                               int8u          commandId,
                                                               EmberAfStatus  status
                                                             )
```

Client Default Response.

RSSI Location cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfRssiLocationClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                           int16u                    indexOrDestination,
                                                           EmberApsFrame *           apsFrame,
                                                           int16u                    msgLen,
                                                           int8u *                   message,
                                                           EmberStatus               status
                                                         )
```

Server Message Sent.

RSSI Location cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfRssiLocationClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                           int16u                    indexOrDestination,
                                                           EmberApsFrame *           apsFrame,
                                                           int16u                    msgLen,
                                                           int8u *                   message,
                                                           EmberStatus               status
                                                         )
```

Client Message Sent.

RSSI Location cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always

message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

void emberAfBinaryInputBasicClusterServerInitCallback (int8u endpoint)

Server Init.

Binary Input (Basic) cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfBinaryInputBasicClusterClientInitCallback (int8u endpoint)

Client Init.

Binary Input (Basic) cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfBinaryInputBasicClusterServerTickCallback (int8u endpoint)

Server Tick.

Binary Input (Basic) cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfBinaryInputBasicClusterClientTickCallback (int8u endpoint)

Client Tick.

Binary Input (Basic) cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfBinaryInputBasicClusterServerAttributeChangedCallback (int8u endpoint,

```
                                                EmberAfAttributeId attributeId
                                              )
```

Server Attribute Changed.

Binary Input (Basic) cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfBinaryInputBasicClusterClientAttributeChangedCallback ( int8u            endpoint,
                                                                    EmberAfAttributeId attributeId
                                                                  )
```

Client Attribute Changed.

Binary Input (Basic) cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfBinaryInputBasicClusterServerManufacturerSpecificAttributeChangedCallback ( int8u            endpoint,
                                                                                        EmberAfAttributeId attributeId,
                                                                                        int16u            manufacturerCode
                                                                                      )
```

Server Manufacturer Specific Attribute Changed.

Binary Input (Basic) cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfBinaryInputBasicClusterClientManufacturerSpecificAttributeChangedCallback ( int8u            endpoint,
                                                                                        EmberAfAttributeId attributeId,
                                                                                        int16u            manufacturerCode
                                                                                      )
```

Client Manufacturer Specific Attribute Changed.

Binary Input (Basic) cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfBinaryInputBasicClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                   int8u          commandId,
                                                                   EmberAfStatus  status
                                                                 )
```

Server Default Response.

Binary Input (Basic) cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfBinaryInputBasicClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                   int8u          commandId,
                                                                   EmberAfStatus  status
                                                                 )
```

Client Default Response.

Binary Input (Basic) cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfBinaryInputBasicClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                               int16u                   indexOrDestination,
                                                               EmberApsFrame *          apsFrame,
                                                               int16u                   msgLen,
```

```
                                                int8u *                 message,
                                                EmberStatus             status
                                              )
```

Server Message Sent.

Binary Input (Basic) cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfBinaryInputBasicClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                               int16u                    indexOrDestination,
                                                               EmberApsFrame *           apsFrame,
                                                               int16u                    msgLen,
                                                               int8u *                   message,
                                                               EmberStatus               status
                                                             )
```

Client Message Sent.

Binary Input (Basic) cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfCommissioningClusterRestartDeviceCallback ( int8u options,
                                                           int8u delay,
                                                           int8u jitter
                                                         )
```

Restart Device.

Parameters:

options Ver.: always

delay Ver.: always

jitter Ver.: always

boolean emberAfCommissioningClusterSaveStartupParametersCallback (int8u options, int8u index)

Save Startup Parameters.

Parameters:

options Ver.: always

index Ver.: always

boolean emberAfCommissioningClusterRestoreStartupParametersCallback (int8u options, int8u index)

Restore Startup Parameters.

Parameters:

options Ver.: always

index Ver.: always

boolean emberAfCommissioningClusterResetStartupParametersCallback (int8u options, int8u index)

Reset Startup Parameters.

Parameters:

options Ver.: always

index Ver.: always

boolean emberAfCommissioningClusterRestartDeviceResponseCallback (int8u status)

Restart Device Response.

Parameters:

status Ver.:
always

boolean emberAfCommissioningClusterSaveStartupParametersResponseCallback (int8u status)

Save Startup Parameters Response.

Parameters:

status Ver.:
always

boolean emberAfCommissioningClusterRestoreStartupParametersResponseCallback (int8u status)

Restore Startup Parameters Response.

Parameters:

status Ver.:
always

boolean emberAfCommissioningClusterResetStartupParametersResponseCallback (int8u status)

Reset Startup Parameters Response.

Parameters:

status Ver.:
always

void emberAfCommissioningClusterServerInitCallback (int8u endpoint)

Server Init.

Commissioning cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.:
always

void emberAfCommissioningClusterClientInitCallback (int8u endpoint)

Client Init.

Commissioning cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfCommissioningClusterServerTickCallback (int8u endpoint)

Server Tick.

Commissioning cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfCommissioningClusterClientTickCallback (int8u endpoint)

Client Tick.

Commissioning cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfCommissioningClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Commissioning cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfCommissioningClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Commissioning cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfCommissioningClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                      EmberAfAttributeId attributeId,
                                                                                      int16u             manufacturerCode
                                                                                    )
```

Server Manufacturer Specific Attribute Changed.

Commissioning cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfCommissioningClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                      EmberAfAttributeId attributeId,
                                                                                      int16u             manufacturerCode
                                                                                    )
```

Client Manufacturer Specific Attribute Changed.

Commissioning cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfCommissioningClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                int8u          commandId,
                                                                EmberAfStatus  status
                                                              )
```

Server Default Response.

Commissioning cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfCommissioningClusterClientDefaultResponseCallback ( int8u           endpoint,
                                                                int8u           commandId,
                                                                EmberAfStatus   status
                                                              )
```

Client Default Response.

Commissioning cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfCommissioningClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                            int16u                     indexOrDestination,
                                                            EmberApsFrame *            apsFrame,
                                                            int16u                     msgLen,
                                                            int8u *                    message,
                                                            EmberStatus                status
                                                          )
```

Server Message Sent.

Commissioning cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfCommissioningClusterClientMessageSentCallback ( EmberOutgoingMessageType  type,
```

```
                                                int16u                  indexOrDestination,
                                                EmberApsFrame *         apsFrame,
                                                int16u                  msgLen,
                                                int8u *                 message,
                                                EmberStatus             status
                                              )
```

Client Message Sent.

Commissioning cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

void emberAfShadeConfigClusterServerInitCallback (int8u endpoint)

Server Init.

Shade Configuration cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfShadeConfigClusterClientInitCallback (int8u endpoint)

Client Init.

Shade Configuration cluster, Client Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfShadeConfigClusterServerTickCallback (int8u endpoint)

Server Tick.

Shade Configuration cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfShadeConfigClusterClientTickCallback (int8u endpoint)

Client Tick.

Shade Configuration cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfShadeConfigClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Shade Configuration cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfShadeConfigClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Shade Configuration cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfShadeConfigClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode

```
)
```

Server Manufacturer Specific Attribute Changed.

Shade Configuration cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfShadeConfigClusterClientManufacturerSpecificAttributeChangedCallback ( int8u                endpoint,
                                                                                    EmberAfAttributeId attributeId,
                                                                                    int16u             manufacturerCode
                                                                                  )
```

Client Manufacturer Specific Attribute Changed.

Shade Configuration cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfShadeConfigClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                              int8u          commandId,
                                                              EmberAfStatus  status
                                                            )
```

Server Default Response.

Shade Configuration cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfShadeConfigClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                              int8u          commandId,
```

```
                                                EmberAfStatus status
                                              )
```

Client Default Response.

Shade Configuration cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfShadeConfigClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                          int16u                    indexOrDestination,
                                                          EmberApsFrame *           apsFrame,
                                                          int16u                    msgLen,
                                                          int8u *                   message,
                                                          EmberStatus               status
                                                        )
```

Server Message Sent.

Shade Configuration cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfShadeConfigClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                          int16u                    indexOrDestination,
                                                          EmberApsFrame *           apsFrame,
                                                          int16u                    msgLen,
                                                          int8u *                   message,
                                                          EmberStatus               status
                                                        )
```

Client Message Sent.

Shade Configuration cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

void emberAfPumpConfigControlClusterServerInitCallback (int8u endpoint)

Server Init.

Pump Configuration and Control cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfPumpConfigControlClusterClientInitCallback (int8u endpoint)

Client Init.

Pump Configuration and Control cluster, Client Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfPumpConfigControlClusterServerTickCallback (int8u endpoint)

Server Tick.

Pump Configuration and Control cluster, Server Tick

Parameters:

endpoint	Endpoint that is being served Ver.: always

void emberAfPumpConfigControlClusterClientTickCallback (int8u endpoint)

Client Tick.

Pump Configuration and Control cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

---

void emberAfPumpConfigControlClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Pump Configuration and Control cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

---

void emberAfPumpConfigControlClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Pump Configuration and Control cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

---

void emberAfPumpConfigControlClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Server Manufacturer Specific Attribute Changed.

Pump Configuration and Control cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

manufacturerCode Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfPumpConfigControlClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                          EmberAfAttributeId attributeId,
                                                                                          int16u             manufacturerCode
                                                                                        )
```

Client Manufacturer Specific Attribute Changed.

Pump Configuration and Control cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfPumpConfigControlClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                     int8u          commandId,
                                                                     EmberAfStatus  status
                                                                   )
```

Server Default Response.

Pump Configuration and Control cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfPumpConfigControlClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                     int8u          commandId,
                                                                     EmberAfStatus  status
                                                                   )
```

Client Default Response.

Pump Configuration and Control cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfPumpConfigControlClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                                int16u                     indexOrDestination,
                                                                EmberApsFrame *            apsFrame,
                                                                int16u                     msgLen,
                                                                int8u *                    message,
                                                                EmberStatus                status
                                                              )
```

Server Message Sent.

Pump Configuration and Control cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfPumpConfigControlClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                                int16u                     indexOrDestination,
                                                                EmberApsFrame *            apsFrame,
                                                                int16u                     msgLen,
                                                                int8u *                    message,
                                                                EmberStatus                status
                                                              )
```

Client Message Sent.

Pump Configuration and Control cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfThermostatClusterSetpointRaiseLowerCallback ( int8u mode,
                                                             int8s amount
```

)

Setpoint Raise Lower.

Parameters:

mode Ver.: always

amount Ver.: always

void emberAfThermostatClusterServerInitCallback (int8u endpoint)

Server Init.

Thermostat cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfThermostatClusterClientInitCallback (int8u endpoint)

Client Init.

Thermostat cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfThermostatClusterServerTickCallback (int8u endpoint)

Server Tick.

Thermostat cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfThermostatClusterClientTickCallback (int8u endpoint)

Client Tick.

Thermostat cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfThermostatClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Thermostat cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfThermostatClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Thermostat cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfThermostatClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Server Manufacturer Specific Attribute Changed.

Thermostat cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

manufacturerCode Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfThermostatClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                  EmberAfAttributeId attributeId,
                                                                                  int16u             manufacturerCode
                                                                                )
```

Client Manufacturer Specific Attribute Changed.

Thermostat cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfThermostatClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                             int8u          commandId,
                                                             EmberAfStatus  status
                                                           )
```

Server Default Response.

Thermostat cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfThermostatClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                             int8u          commandId,
                                                             EmberAfStatus  status
                                                           )
```

Client Default Response.

Thermostat cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always

status	Status in default response Ver.: always

```
void emberAfThermostatClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                         int16u                    indexOrDestination,
                                                         EmberApsFrame *           apsFrame,
                                                         int16u                    msgLen,
                                                         int8u *                   message,
                                                         EmberStatus               status
                                                       )
```

Server Message Sent.

Thermostat cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfThermostatClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                         int16u                    indexOrDestination,
                                                         EmberApsFrame *           apsFrame,
                                                         int16u                    msgLen,
                                                         int8u *                   message,
                                                         EmberStatus               status
                                                       )
```

Client Message Sent.

Thermostat cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

void emberAfFanControlClusterServerInitCallback (int8u endpoint)

Server Init.

Fan Control cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfFanControlClusterClientInitCallback (int8u endpoint)

Client Init.

Fan Control cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfFanControlClusterServerTickCallback (int8u endpoint)

Server Tick.

Fan Control cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfFanControlClusterClientTickCallback (int8u endpoint)

Client Tick.

Fan Control cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfFanControlClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Fan Control cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfFanControlClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                              EmberAfAttributeId attributeId
                                                            )
```

Client Attribute Changed.

Fan Control cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfFanControlClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                  EmberAfAttributeId attributeId,
                                                                                  int16u             manufacturerCode
                                                                                )
```

Server Manufacturer Specific Attribute Changed.

Fan Control cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfFanControlClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                  EmberAfAttributeId attributeId,
                                                                                  int16u             manufacturerCode
                                                                                )
```

Client Manufacturer Specific Attribute Changed.

Fan Control cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfFanControlClusterServerDefaultResponseCallback ( int8u           endpoint,
                                                             int8u           commandId,
                                                             EmberAfStatus   status
                                                           )
```

Server Default Response.

Fan Control cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfFanControlClusterClientDefaultResponseCallback ( int8u           endpoint,
                                                             int8u           commandId,
                                                             EmberAfStatus   status
                                                           )
```

Client Default Response.

Fan Control cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfFanControlClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                         int16u                    indexOrDestination,
                                                         EmberApsFrame *           apsFrame,
                                                         int16u                    msgLen,
                                                         int8u *                   message,
                                                         EmberStatus               status
                                                       )
```

Server Message Sent.

Fan Control cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfFanControlClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                         int16u                    indexOrDestination,
                                                         EmberApsFrame *           apsFrame,
                                                         int16u                    msgLen,
                                                         int8u *                   message,
                                                         EmberStatus               status
                                                       )
```

Client Message Sent.

Fan Control cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfDehumidControlClusterServerInitCallback ( int8u endpoint )
```

Server Init.

Dehumidification Control cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

```
void emberAfDehumidControlClusterClientInitCallback ( int8u endpoint )
```

Client Init.

Dehumidification Control cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfDehumidControlClusterServerTickCallback (int8u endpoint)

Server Tick.

Dehumidification Control cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfDehumidControlClusterClientTickCallback (int8u endpoint)

Client Tick.

Dehumidification Control cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfDehumidControlClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Dehumidification Control cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfDehumidControlClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Dehumidification Control cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfDehumidControlClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                       EmberAfAttributeId attributeId,
                                                                                       int16u             manufacturerCode
                                                                                     )
```

Server Manufacturer Specific Attribute Changed.

Dehumidification Control cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfDehumidControlClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                       EmberAfAttributeId attributeId,
                                                                                       int16u             manufacturerCode
                                                                                     )
```

Client Manufacturer Specific Attribute Changed.

Dehumidification Control cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfDehumidControlClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                  int8u          commandId,
                                                                  EmberAfStatus  status
                                                                )
```

Server Default Response.

Dehumidification Control cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfDehumidControlClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                 int8u          commandId,
                                                                 EmberAfStatus  status
                                                               )
```

Client Default Response.

Dehumidification Control cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfDehumidControlClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                             int16u                    indexOrDestination,
                                                             EmberApsFrame *           apsFrame,
                                                             int16u                    msgLen,
                                                             int8u *                   message,
                                                             EmberStatus               status
                                                           )
```

Server Message Sent.

Dehumidification Control cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfDehumidControlClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                             int16u                    indexOrDestination,
                                                             EmberApsFrame *           apsFrame,
                                                             int16u                    msgLen,
                                                             int8u *                   message,
                                                             EmberStatus               status
                                                           )
```

Client Message Sent.

Dehumidification Control cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfThermostatUiConfigClusterServerInitCallback ( int8u endpoint )
```

Server Init.

Thermostat User Interface Configuration cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

```
void emberAfThermostatUiConfigClusterClientInitCallback ( int8u endpoint )
```

Client Init.

Thermostat User Interface Configuration cluster, Client Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

```
void emberAfThermostatUiConfigClusterServerTickCallback ( int8u endpoint )
```

Server Tick.

Thermostat User Interface Configuration cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfThermostatUiConfigClusterClientTickCallback (int8u endpoint)

Client Tick.

Thermostat User Interface Configuration cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfThermostatUiConfigClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Thermostat User Interface Configuration cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfThermostatUiConfigClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Thermostat User Interface Configuration cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfThermostatUiConfigClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint,

```
                                                EmberAfAttributeId attributeId,
                                                int16u             manufacturerCode
                                              )
```

Server Manufacturer Specific Attribute Changed.

Thermostat User Interface Configuration cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfThermostatUiConfigClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                          EmberAfAttributeId attributeId,
                                                                                          int16u             manufacturerCode
                                                                                        )
```

Client Manufacturer Specific Attribute Changed.

Thermostat User Interface Configuration cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfThermostatUiConfigClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                      int8u          commandId,
                                                                      EmberAfStatus  status
                                                                    )
```

Server Default Response.

Thermostat User Interface Configuration cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfThermostatUiConfigClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                     int8u          commandId,
                                                                     EmberAfStatus  status
                                                                   )
```

Client Default Response.

Thermostat User Interface Configuration cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfThermostatUiConfigClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                                 int16u                    indexOrDestination,
                                                                 EmberApsFrame *           apsFrame,
                                                                 int16u                    msgLen,
                                                                 int8u *                   message,
                                                                 EmberStatus               status
                                                               )
```

Server Message Sent.

Thermostat User Interface Configuration cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfThermostatUiConfigClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                                 int16u                    indexOrDestination,
                                                                 EmberApsFrame *           apsFrame,
                                                                 int16u                    msgLen,
                                                                 int8u *                   message,
                                                                 EmberStatus               status
                                                               )
```

Client Message Sent.

Thermostat User Interface Configuration cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfColorControlClusterMoveToHueCallback ( int8u   hue,
                                                      int8u   direction,
                                                      int16u  transitionTime
                                                    )
```

Move To Hue.

Parameters:

hue	Ver.: always
direction	Ver.: always
transitionTime	Ver.: always

```
boolean emberAfColorControlClusterMoveHueCallback ( int8u  moveMode,
                                                    int8u  rate
                                                  )
```

Move Hue.

Parameters:

moveMode	Ver.: always
rate	Ver.: always

```
boolean emberAfColorControlClusterStepHueCallback ( int8u  stepMode,
                                                    int8u  stepSize,
                                                    int8u  transitionTime
                                                  )
```

Step Hue.

Parameters:

stepMode	Ver.: always
stepSize	Ver.: always
transitionTime	Ver.: always

```
boolean emberAfColorControlClusterMoveToSaturationCallback ( int8u   saturation,
                                                              int16u  transitionTime
                                                            )
```

Move To Saturation.

Parameters:

saturation	Ver.: always
transitionTime	Ver.: always

```
boolean emberAfColorControlClusterMoveSaturationCallback ( int8u  moveMode,
                                                           int8u  rate
                                                         )
```

Move Saturation.

Parameters:

moveMode	Ver.: always
rate	Ver.: always

```
boolean emberAfColorControlClusterStepSaturationCallback ( int8u  stepMode,
                                                           int8u  stepSize,
                                                           int8u  transitionTime
                                                         )
```

Step Saturation.

Parameters:

stepMode	Ver.: always
stepSize	Ver.: always

transitionTime Ver.: always

```
boolean emberAfColorControlClusterMoveToHueAndSaturationCallback ( int8u   hue,
                                                                    int8u   saturation,
                                                                    int16u  transitionTime
                                                                  )
```

Move To Hue And Saturation.

Parameters:

hue Ver.: always

saturation Ver.: always

transitionTime Ver.: always

```
boolean emberAfColorControlClusterMoveToColorCallback ( int16u  colorX,
                                                        int16u  colorY,
                                                        int16u  transitionTime
                                                      )
```

Move To Color.

Parameters:

colorX Ver.: always

colorY Ver.: always

transitionTime Ver.: always

```
boolean emberAfColorControlClusterMoveColorCallback ( int16s  rateX,
                                                      int16s  rateY
                                                    )
```

Move Color.

Parameters:

rateX Ver.: always

rateY Ver.: always

```
boolean emberAfColorControlClusterStepColorCallback ( int16s stepX,
                                                      int16s stepY,
                                                      int16u transitionTime
                                                    )
```

Step Color.

Parameters:

stepX	Ver.: always
stepY	Ver.: always
transitionTime	Ver.: always

```
boolean emberAfColorControlClusterMoveToColorTemperatureCallback ( int16u colorTemperature,
                                                                   int16u transitionTime
                                                                 )
```

Move To Color Temperature.

Parameters:

colorTemperature	Ver.: always
transitionTime	Ver.: always

```
void emberAfColorControlClusterServerInitCallback ( int8u endpoint )
```

Server Init.

Color Control cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

```
void emberAfColorControlClusterClientInitCallback ( int8u endpoint )
```

Client Init.

Color Control cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfColorControlClusterServerTickCallback (int8u endpoint)

Server Tick.

Color Control cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfColorControlClusterClientTickCallback (int8u endpoint)

Client Tick.

Color Control cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfColorControlClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Color Control cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfColorControlClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Color Control cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfColorControlClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                     EmberAfAttributeId attributeId,
                                                                                     int16u             manufacturerCode
                                                                                   )
```

Server Manufacturer Specific Attribute Changed.

Color Control cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfColorControlClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                     EmberAfAttributeId attributeId,
                                                                                     int16u             manufacturerCode
                                                                                   )
```

Client Manufacturer Specific Attribute Changed.

Color Control cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfColorControlClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                               int8u          commandId,
                                                               EmberAfStatus  status
                                                             )
```

Server Default Response.

Color Control cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfColorControlClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                               int8u          commandId,
                                                               EmberAfStatus  status
                                                             )
```

Client Default Response.

Color Control cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfColorControlClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                           int16u                    indexOrDestination,
                                                           EmberApsFrame *           apsFrame,
                                                           int16u                    msgLen,
                                                           int8u *                   message,
                                                           EmberStatus               status
                                                         )
```

Server Message Sent.

Color Control cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfColorControlClusterClientMessageSentCallback ( EmberOutgoingMessageType  type,
                                                           int16u                    indexOrDestination,
                                                           EmberApsFrame *           apsFrame,
                                                           int16u                    msgLen,
```

```
int8u *          message,
EmberStatus      status
)
```

Client Message Sent.

Color Control cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

void emberAfBallastConfigurationClusterServerInitCallback (int8u endpoint)

Server Init.

Ballast Configuration cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfBallastConfigurationClusterClientInitCallback (int8u endpoint)

Client Init.

Ballast Configuration cluster, Client Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfBallastConfigurationClusterServerTickCallback (int8u endpoint)

Server Tick.

Ballast Configuration cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfBallastConfigurationClusterClientTickCallback (int8u endpoint)

Client Tick.

Ballast Configuration cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfBallastConfigurationClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Ballast Configuration cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfBallastConfigurationClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Ballast Configuration cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfBallastConfigurationClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Server Manufacturer Specific Attribute Changed.

Ballast Configuration cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfBallastConfigurationClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                            EmberAfAttributeId attributeId,
                                                                                            int16u             manufacturerCode
                                                                                          )
```

Client Manufacturer Specific Attribute Changed.

Ballast Configuration cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfBallastConfigurationClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                        int8u          commandId,
                                                                        EmberAfStatus  status
                                                                      )
```

Server Default Response.

Ballast Configuration cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfBallastConfigurationClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                        int8u          commandId,
                                                                        EmberAfStatus  status
                                                                      )
```

Client Default Response.

Ballast Configuration cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfBallastConfigurationClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                                   int16u                    indexOrDestination,
                                                                   EmberApsFrame *           apsFrame,
                                                                   int16u                    msgLen,
                                                                   int8u *                   message,
                                                                   EmberStatus               status
                                                                 )
```

Server Message Sent.

Ballast Configuration cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfBallastConfigurationClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                                   int16u                    indexOrDestination,
                                                                   EmberApsFrame *           apsFrame,
                                                                   int16u                    msgLen,
                                                                   int8u *                   message,
                                                                   EmberStatus               status
                                                                 )
```

Client Message Sent.

Ballast Configuration cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always

indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

void emberAfIllumMeasurementClusterServerInitCallback (int8u endpoint)

Server Init.

Illuminance Measurement cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfIllumMeasurementClusterClientInitCallback (int8u endpoint)

Client Init.

Illuminance Measurement cluster, Client Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfIllumMeasurementClusterServerTickCallback (int8u endpoint)

Server Tick.

Illuminance Measurement cluster, Server Tick

Parameters:

endpoint	Endpoint that is being served Ver.: always

void emberAfIllumMeasurementClusterClientTickCallback (int8u endpoint)

Client Tick.

Illuminance Measurement cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfIllumMeasurementClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Illuminance Measurement cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfIllumMeasurementClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Illuminance Measurement cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfIllumMeasurementClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Server Manufacturer Specific Attribute Changed.

Illuminance Measurement cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

manufacturerCode Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfIllumMeasurementClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                         EmberAfAttributeId attributeId,
                                                                                         int16u             manufacturerCode
                                                                                       )
```

Client Manufacturer Specific Attribute Changed.

Illuminance Measurement cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfIllumMeasurementClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                   int8u          commandId,
                                                                   EmberAfStatus  status
                                                                 )
```

Server Default Response.

Illuminance Measurement cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfIllumMeasurementClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                   int8u          commandId,
                                                                   EmberAfStatus  status
                                                                 )
```

Client Default Response.

Illuminance Measurement cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfIllumMeasurementClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                               int16u                    indexOrDestination,
                                                               EmberApsFrame *           apsFrame,
                                                               int16u                    msgLen,
                                                               int8u *                   message,
                                                               EmberStatus               status
                                                             )
```

Server Message Sent.

Illuminance Measurement cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfIllumMeasurementClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                               int16u                    indexOrDestination,
                                                               EmberApsFrame *           apsFrame,
                                                               int16u                    msgLen,
                                                               int8u *                   message,
                                                               EmberStatus               status
                                                             )
```

Client Message Sent.

Illuminance Measurement cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfIllumLevelSensingClusterServerInitCallback ( int8u endpoint )
```

Server Init.

Illuminance Level Sensing cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfIllumLevelSensingClusterClientInitCallback (int8u endpoint)

Client Init.

Illuminance Level Sensing cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfIllumLevelSensingClusterServerTickCallback (int8u endpoint)

Server Tick.

Illuminance Level Sensing cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfIllumLevelSensingClusterClientTickCallback (int8u endpoint)

Client Tick.

Illuminance Level Sensing cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfIllumLevelSensingClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Illuminance Level Sensing cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfIllumLevelSensingClusterClientAttributeChangedCallback ( int8u            endpoint,
                                                                     EmberAfAttributeId attributeId
                                                                   )
```

Client Attribute Changed.

Illuminance Level Sensing cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfIllumLevelSensingClusterServerManufacturerSpecificAttributeChangedCallback ( int8u            endpoint,
                                                                                          EmberAfAttributeId attributeId,
                                                                                          int16u            manufacturerCode
                                                                                        )
```

Server Manufacturer Specific Attribute Changed.

Illuminance Level Sensing cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfIllumLevelSensingClusterClientManufacturerSpecificAttributeChangedCallback ( int8u            endpoint,
                                                                                         EmberAfAttributeId attributeId,
                                                                                         int16u            manufacturerCode
                                                                                       )
```

Client Manufacturer Specific Attribute Changed.

Illuminance Level Sensing cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfIllumLevelSensingClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                    int8u          commandId,
                                                                    EmberAfStatus  status
                                                                  )
```

Server Default Response.

Illuminance Level Sensing cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfIllumLevelSensingClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                    int8u          commandId,
                                                                    EmberAfStatus  status
                                                                  )
```

Client Default Response.

Illuminance Level Sensing cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfIllumLevelSensingClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                                int16u                    indexOrDestination,
                                                                EmberApsFrame *           apsFrame,
                                                                int16u                    msgLen,
                                                                int8u *                   message,
                                                                EmberStatus               status
                                                              )
```

Server Message Sent.

Illuminance Level Sensing cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfIllumLevelSensingClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                                int16u indexOrDestination,
                                                                EmberApsFrame * apsFrame,
                                                                int16u msgLen,
                                                                int8u * message,
                                                                EmberStatus status
                                                              )
```

Client Message Sent.

Illuminance Level Sensing cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
int32s emberAfTempMeasurementClusterConvertVoltsToCelsiusCallback ( int16u voltage )
```

Convert Volts To Celsius.

Function that converts volts on the ADC line into celsius.

Parameters:

voltage	Voltage from ADC Ver.: always

```
void emberAfTempMeasurementClusterServerInitCallback ( int8u endpoint )
```

Server Init.

Temperature Measurement cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfTempMeasurementClusterClientInitCallback (int8u endpoint)

Client Init.

Temperature Measurement cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfTempMeasurementClusterServerTickCallback (int8u endpoint)

Server Tick.

Temperature Measurement cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfTempMeasurementClusterClientTickCallback (int8u endpoint)

Client Tick.

Temperature Measurement cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfTempMeasurementClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Temperature Measurement cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfTempMeasurementClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                                   EmberAfAttributeId attributeId
                                                                 )
```

Client Attribute Changed.

Temperature Measurement cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfTempMeasurementClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                       EmberAfAttributeId attributeId,
                                                                                       int16u             manufacturerCode
                                                                                     )
```

Server Manufacturer Specific Attribute Changed.

Temperature Measurement cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfTempMeasurementClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                       EmberAfAttributeId attributeId,
                                                                                       int16u             manufacturerCode
                                                                                     )
```

Client Manufacturer Specific Attribute Changed.

Temperature Measurement cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfTempMeasurementClusterServerDefaultResponseCallback ( int8u           endpoint,
                                                                  int8u           commandId,
                                                                  EmberAfStatus   status
                                                                )
```

Server Default Response.

Temperature Measurement cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfTempMeasurementClusterClientDefaultResponseCallback ( int8u           endpoint,
                                                                  int8u           commandId,
                                                                  EmberAfStatus   status
                                                                )
```

Client Default Response.

Temperature Measurement cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfTempMeasurementClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                              int16u                     indexOrDestination,
                                                              EmberApsFrame *            apsFrame,
                                                              int16u                     msgLen,
                                                              int8u *                    message,
                                                              EmberStatus                status
                                                            )
```

Server Message Sent.

Temperature Measurement cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfTempMeasurementClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                              int16u indexOrDestination,
                                                              EmberApsFrame * apsFrame,
                                                              int16u msgLen,
                                                              int8u * message,
                                                              EmberStatus status
                                                            )
```

Client Message Sent.

Temperature Measurement cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfPressureMeasurementClusterServerInitCallback ( int8u endpoint )
```

Server Init.

Pressure Measurement cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

```
void emberAfPressureMeasurementClusterClientInitCallback ( int8u endpoint )
```

Client Init.

Pressure Measurement cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfPressureMeasurementClusterServerTickCallback (int8u endpoint)

Server Tick.

Pressure Measurement cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfPressureMeasurementClusterClientTickCallback (int8u endpoint)

Client Tick.

Pressure Measurement cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfPressureMeasurementClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Pressure Measurement cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfPressureMeasurementClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Pressure Measurement cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfPressureMeasurementClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                            EmberAfAttributeId attributeId,
                                                                                            int16u             manufacturerCode
                                                                                          )
```

Server Manufacturer Specific Attribute Changed.

Pressure Measurement cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfPressureMeasurementClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                            EmberAfAttributeId attributeId,
                                                                                            int16u             manufacturerCode
                                                                                          )
```

Client Manufacturer Specific Attribute Changed.

Pressure Measurement cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfPressureMeasurementClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                      int8u          commandId,
                                                                      EmberAfStatus  status
                                                                    )
```

Server Default Response.

Pressure Measurement cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfPressureMeasurementClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                      int8u          commandId,
                                                                      EmberAfStatus  status
                                                                    )
```

Client Default Response.

Pressure Measurement cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfPressureMeasurementClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                                  int16u                    indexOrDestination,
                                                                  EmberApsFrame *           apsFrame,
                                                                  int16u                    msgLen,
                                                                  int8u *                   message,
                                                                  EmberStatus               status
                                                                )
```

Server Message Sent.

Pressure Measurement cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfPressureMeasurementClusterClientMessageSentCallback ( EmberOutgoingMessageType  type,
                                                                  int16u                    indexOrDestination,
```

```
                                    EmberApsFrame *     apsFrame,
                                    int16u              msgLen,
                                    int8u *             message,
                                    EmberStatus         status
                                )
```

Client Message Sent.

Pressure Measurement cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

void emberAfFlowMeasurementClusterServerInitCallback (int8u endpoint)

Server Init.

Flow Measurement cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfFlowMeasurementClusterClientInitCallback (int8u endpoint)

Client Init.

Flow Measurement cluster, Client Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfFlowMeasurementClusterServerTickCallback (int8u endpoint)

Server Tick.

Flow Measurement cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfFlowMeasurementClusterClientTickCallback (int8u endpoint)

Client Tick.

Flow Measurement cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfFlowMeasurementClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Flow Measurement cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfFlowMeasurementClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Flow Measurement cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfFlowMeasurementClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Server Manufacturer Specific Attribute Changed.

Flow Measurement cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfFlowMeasurementClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                       EmberAfAttributeId attributeId,
                                                                                       int16u             manufacturerCode
                                                                                     )
```

Client Manufacturer Specific Attribute Changed.

Flow Measurement cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfFlowMeasurementClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                  int8u          commandId,
                                                                  EmberAfStatus  status
                                                                )
```

Server Default Response.

Flow Measurement cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfFlowMeasurementClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                  int8u          commandId,
                                                                  EmberAfStatus  status
                                                                )
```

Client Default Response.

Flow Measurement cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfFlowMeasurementClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                              int16u                    indexOrDestination,
                                                              EmberApsFrame *           apsFrame,
                                                              int16u                    msgLen,
                                                              int8u *                   message,
                                                              EmberStatus               status
                                                            )
```

Server Message Sent.

Flow Measurement cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfFlowMeasurementClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                              int16u                    indexOrDestination,
                                                              EmberApsFrame *           apsFrame,
                                                              int16u                    msgLen,
                                                              int8u *                   message,
                                                              EmberStatus               status
                                                            )
```

Client Message Sent.

Flow Measurement cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

void emberAfRelativeHumidityMeasurementClusterServerInitCallback (int8u endpoint)

Server Init.

Relative Humidity Measurement cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfRelativeHumidityMeasurementClusterClientInitCallback (int8u endpoint)

Client Init.

Relative Humidity Measurement cluster, Client Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfRelativeHumidityMeasurementClusterServerTickCallback (int8u endpoint)

Server Tick.

Relative Humidity Measurement cluster, Server Tick

Parameters:

endpoint	Endpoint that is being served Ver.: always

void emberAfRelativeHumidityMeasurementClusterClientTickCallback (int8u endpoint)

Client Tick.

Relative Humidity Measurement cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfRelativeHumidityMeasurementClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Relative Humidity Measurement cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfRelativeHumidityMeasurementClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Relative Humidity Measurement cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfRelativeHumidityMeasurementClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Server Manufacturer Specific Attribute Changed.

Relative Humidity Measurement cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

manufacturerCode Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfRelativeHumidityMeasurementClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                                    EmberAfAttributeId attributeId,
                                                                                                    int16u             manufacturerCode
                                                                                                  )
```

Client Manufacturer Specific Attribute Changed.

Relative Humidity Measurement cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfRelativeHumidityMeasurementClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                              int8u          commandId,
                                                                              EmberAfStatus  status
                                                                            )
```

Server Default Response.

Relative Humidity Measurement cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfRelativeHumidityMeasurementClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                              int8u          commandId,
                                                                              EmberAfStatus  status
                                                                            )
```

Client Default Response.

Relative Humidity Measurement cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfRelativeHumidityMeasurementClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                                          int16u indexOrDestination,
                                                                          EmberApsFrame * apsFrame,
                                                                          int16u msgLen,
                                                                          int8u * message,
                                                                          EmberStatus status
                                                                        )
```

Server Message Sent.

Relative Humidity Measurement cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfRelativeHumidityMeasurementClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                                          int16u indexOrDestination,
                                                                          EmberApsFrame * apsFrame,
                                                                          int16u msgLen,
                                                                          int8u * message,
                                                                          EmberStatus status
                                                                        )
```

Client Message Sent.

Relative Humidity Measurement cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfOccupancySensingClusterServerInitCallback ( int8u endpoint )
```

Server Init.

Occupancy Sensing cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfOccupancySensingClusterClientInitCallback (int8u endpoint)

Client Init.

Occupancy Sensing cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfOccupancySensingClusterServerTickCallback (int8u endpoint)

Server Tick.

Occupancy Sensing cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfOccupancySensingClusterClientTickCallback (int8u endpoint)

Client Tick.

Occupancy Sensing cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfOccupancySensingClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Occupancy Sensing cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfOccupancySensingClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                                    EmberAfAttributeId  attributeId
                                                                  )
```

Client Attribute Changed.

Occupancy Sensing cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfOccupancySensingClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                        EmberAfAttributeId  attributeId,
                                                                                        int16u              manufacturerCode
                                                                                      )
```

Server Manufacturer Specific Attribute Changed.

Occupancy Sensing cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfOccupancySensingClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                        EmberAfAttributeId  attributeId,
                                                                                        int16u              manufacturerCode
                                                                                      )
```

Client Manufacturer Specific Attribute Changed.

Occupancy Sensing cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfOccupancySensingClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                    int8u          commandId,
                                                                    EmberAfStatus  status
                                                                  )
```

Server Default Response.

Occupancy Sensing cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfOccupancySensingClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                   int8u          commandId,
                                                                   EmberAfStatus  status
                                                                 )
```

Client Default Response.

Occupancy Sensing cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfOccupancySensingClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                               int16u                    indexOrDestination,
                                                               EmberApsFrame *           apsFrame,
                                                               int16u                    msgLen,
                                                               int8u *                   message,
                                                               EmberStatus               status
                                                             )
```

Server Message Sent.

Occupancy Sensing cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfOccupancySensingClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                               int16u                    indexOrDestination,
                                                               EmberApsFrame *           apsFrame,
                                                               int16u                    msgLen,
                                                               int8u *                   message,
                                                               EmberStatus               status
                                                             )
```

Client Message Sent.

Occupancy Sensing cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfIasZoneClusterZoneEnrollResponseCallback ( int8u enrollResponseCode,
                                                          int8u zoneId
                                                        )
```

Zone Enroll Response.

Parameters:

enrollResponseCode	Ver.: always
zoneId	Ver.: always

```
boolean emberAfIasZoneClusterZoneStatusChangeNotificationCallback ( int16u zoneStatus,
```

```
                                                    int8u   extendedStatus
                                                )
```

Zone Status Change Notification.

Parameters:

zoneStatus	Ver.: always
extendedStatus	Ver.: always

boolean emberAfIasZoneClusterZoneEnrollRequestCallback (int16u zoneType, int16u manufacturerCode)

Zone Enroll Request.

Parameters:

zoneType	Ver.: always
manufacturerCode	Ver.: always

void emberAfIasZoneClusterServerInitCallback (int8u endpoint)

Server Init.

IAS Zone cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfIasZoneClusterClientInitCallback (int8u endpoint)

Client Init.

IAS Zone cluster, Client Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfIasZoneClusterServerTickCallback (int8u endpoint)

Server Tick.

IAS Zone cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfIasZoneClusterClientTickCallback (int8u endpoint)

Client Tick.

IAS Zone cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfIasZoneClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

IAS Zone cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfIasZoneClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

IAS Zone cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

```
void emberAfIasZoneClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                               EmberAfAttributeId attributeId,
                                                                               int16u             manufacturerCode
                                                                             )
```

Server Manufacturer Specific Attribute Changed.

IAS Zone cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfIasZoneClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                               EmberAfAttributeId attributeId,
                                                                               int16u             manufacturerCode
                                                                             )
```

Client Manufacturer Specific Attribute Changed.

IAS Zone cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfIasZoneClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                          int8u          commandId,
                                                          EmberAfStatus  status
                                                        )
```

Server Default Response.

IAS Zone cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfIasZoneClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                          int8u          commandId,
                                                          EmberAfStatus  status
                                                        )
```

Client Default Response.

IAS Zone cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfIasZoneClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                      int16u                     indexOrDestination,
                                                      EmberApsFrame *            apsFrame,
                                                      int16u                     msgLen,
                                                      int8u *                    message,
                                                      EmberStatus                status
                                                    )
```

Server Message Sent.

IAS Zone cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfIasZoneClusterClientMessageSentCallback ( EmberOutgoingMessageType  type,
                                                      int16u                     indexOrDestination,
                                                      EmberApsFrame *            apsFrame,
                                                      int16u                     msgLen,
                                                      int8u *                    message,
                                                      EmberStatus                status
                                                    )
```

Client Message Sent.

IAS Zone cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

boolean emberAfIasAceClusterArmCallback (int8u armMode)

Arm.

Parameters:

armMode	Ver.: always

boolean emberAfIasAceClusterBypassCallback (int8u numberOfZones, int8u * zoneIds)

Bypass.

Parameters:

numberOfZones	Ver.: always
zoneIds	Ver.: always

boolean emberAfIasAceClusterEmergencyCallback (void)

Emergency.

boolean emberAfIasAceClusterFireCallback (void)

Fire.

boolean emberAfIasAceClusterPanicCallback (void)

Panic.

boolean emberAfIasAceClusterGetZoneIdMapCallback (void)

Get Zone Id Map.

boolean emberAfIasAceClusterGetZoneInformationCallback (int8u zoneId)

Get Zone Information.

Parameters:

zoneId Ver.: always

boolean emberAfIasAceClusterArmResponseCallback (int8u armNotification)

Arm Response.

Parameters:

armNotification Ver.: always

```
boolean emberAfIasAceClusterGetZoneIdMapResponseCallback ( int16u section0,
                                                           int16u section1,
                                                           int16u section2,
                                                           int16u section3,
                                                           int16u section4,
                                                           int16u section5,
                                                           int16u section6,
                                                           int16u section7,
                                                           int16u section8,
                                                           int16u section9,
                                                           int16u section10,
                                                           int16u section11,
                                                           int16u section12,
                                                           int16u section13,
                                                           int16u section14,
                                                           int16u section15
                                                         )
```

Get Zone Id Map Response.

Parameters:

section0 Ver.: always

section1 Ver.: always
section2 Ver.: always
section3 Ver.: always
section4 Ver.: always
section5 Ver.: always
section6 Ver.: always
section7 Ver.: always
section8 Ver.: always
section9 Ver.: always
section10 Ver.: always
section11 Ver.: always
section12 Ver.: always
section13 Ver.: always
section14 Ver.: always
section15 Ver.: always

boolean emberAfIasAceClusterGetZoneInformationResponseCallback (int8u zoneId, int16u zoneType, int8u * ieeeAddress)

Get Zone Information Response.

Parameters:

zoneId Ver.: always
zoneType Ver.: always
ieeeAddress Ver.: always

void emberAfIasAceClusterServerInitCallback (int8u endpoint)

Server Init.

IAS ACE cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfIasAceClusterClientInitCallback (int8u endpoint)

Client Init.

IAS ACE cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfIasAceClusterServerTickCallback (int8u endpoint)

Server Tick.

IAS ACE cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfIasAceClusterClientTickCallback (int8u endpoint)

Client Tick.

IAS ACE cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfIasAceClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

IAS ACE cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfIasAceClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                          EmberAfAttributeId attributeId
                                                        )
```

Client Attribute Changed.

IAS ACE cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfIasAceClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                              EmberAfAttributeId attributeId,
                                                                              int16u             manufacturerCode
                                                                            )
```

Server Manufacturer Specific Attribute Changed.

IAS ACE cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfIasAceClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                              EmberAfAttributeId attributeId,
                                                                              int16u             manufacturerCode
                                                                            )
```

Client Manufacturer Specific Attribute Changed.

IAS ACE cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfIasAceClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                         int8u          commandId,
                                                         EmberAfStatus  status
                                                       )
```

Server Default Response.

IAS ACE cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfIasAceClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                         int8u          commandId,
                                                         EmberAfStatus  status
                                                       )
```

Client Default Response.

IAS ACE cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfIasAceClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                     int16u                     indexOrDestination,
                                                     EmberApsFrame *            apsFrame,
                                                     int16u                     msgLen,
                                                     int8u *                    message,
                                                     EmberStatus                status
                                                   )
```

Server Message Sent.

IAS ACE cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfIasAceClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                     int16u                    indexOrDestination,
                                                     EmberApsFrame *           apsFrame,
                                                     int16u                    msgLen,
                                                     int8u *                   message,
                                                     EmberStatus               status
                                                   )
```

Client Message Sent.

IAS ACE cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfIasWdClusterStartWarningCallback ( int8u   warningInfo,
                                                  int16u  warningDuration
                                                )
```

Start Warning.

Parameters:

warningInfo	Ver.: always
warningDuration	Ver.: always

```
boolean emberAfIasWdClusterSquawkCallback ( int8u squawkInfo )
```

Squawk.

Parameters:

squawkInfo Ver.: always

void emberAfIasWdClusterServerInitCallback (int8u endpoint)

Server Init.

IAS WD cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfIasWdClusterClientInitCallback (int8u endpoint)

Client Init.

IAS WD cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfIasWdClusterServerTickCallback (int8u endpoint)

Server Tick.

IAS WD cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfIasWdClusterClientTickCallback (int8u endpoint)

Client Tick.

IAS WD cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

```
void emberAfIasWdClusterServerAttributeChangedCallback ( int8u              endpoint,
                                                         EmberAfAttributeId attributeId
                                                       )
```

Server Attribute Changed.

IAS WD cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfIasWdClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                         EmberAfAttributeId attributeId
                                                       )
```

Client Attribute Changed.

IAS WD cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfIasWdClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                             EmberAfAttributeId attributeId,
                                                                             int16u             manufacturerCode
                                                                           )
```

Server Manufacturer Specific Attribute Changed.

IAS WD cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfIasWdClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                             EmberAfAttributeId attributeId,
```

```
                                                    int16u          manufacturerCode
                                                )
```

Client Manufacturer Specific Attribute Changed.

IAS WD cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfIasWdClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                        int8u          commandId,
                                                        EmberAfStatus  status
                                                      )
```

Server Default Response.

IAS WD cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfIasWdClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                        int8u          commandId,
                                                        EmberAfStatus  status
                                                      )
```

Client Default Response.

IAS WD cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfIasWdClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
```

```
                                                int16u                  indexOrDestination,
                                                EmberApsFrame *         apsFrame,
                                                int16u                  msgLen,
                                                int8u *                 message,
                                                EmberStatus             status
                                              )
```

Server Message Sent.

IAS WD cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfIasWdClusterClientMessageSentCallback ( EmberOutgoingMessageType  type,
                                                    int16u                    indexOrDestination,
                                                    EmberApsFrame *           apsFrame,
                                                    int16u                    msgLen,
                                                    int8u *                   message,
                                                    EmberStatus               status
                                                  )
```

Client Message Sent.

IAS WD cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfDoorLockClusterLockDoorCallback ( void   )
```

Lock Door.

boolean emberAfDoorLockClusterUnlockDoorCallback (void)

Unlock Door.

boolean emberAfDoorLockClusterLockDoorResponseCallback (int8u status)

Lock Door Response.

Parameters:

status Ver.: always

boolean emberAfDoorLockClusterUnlockDoorResponseCallback (int8u status)

Unlock Door Response.

Parameters:

status Ver.: always

void emberAfDoorLockClusterServerInitCallback (int8u endpoint)

Server Init.

Door Lock cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfDoorLockClusterClientInitCallback (int8u endpoint)

Client Init.

Door Lock cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfDoorLockClusterServerTickCallback (int8u endpoint)

Server Tick.

Door Lock cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfDoorLockClusterClientTickCallback (int8u endpoint)

Client Tick.

Door Lock cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfDoorLockClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Door Lock cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfDoorLockClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Door Lock cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfDoorLockClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId,

```
                                                                    int16u          manufacturerCode
                                                                  )
```

Server Manufacturer Specific Attribute Changed.

Door Lock cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfDoorLockClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                EmberAfAttributeId  attributeId,
                                                                                int16u              manufacturerCode
                                                                              )
```

Client Manufacturer Specific Attribute Changed.

Door Lock cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfDoorLockClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                           int8u          commandId,
                                                           EmberAfStatus  status
                                                         )
```

Server Default Response.

Door Lock cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfDoorLockClusterClientDefaultResponseCallback ( int8u          endpoint,
```

```
                                                    int8u          commandId,
                                                    EmberAfStatus  status
                                                  )
```

Client Default Response.

Door Lock cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfDoorLockClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                       int16u                    indexOrDestination,
                                                       EmberApsFrame *           apsFrame,
                                                       int16u                    msgLen,
                                                       int8u *                   message,
                                                       EmberStatus               status
                                                     )
```

Server Message Sent.

Door Lock cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfDoorLockClusterClientMessageSentCallback ( EmberOutgoingMessageType  type,
                                                       int16u                    indexOrDestination,
                                                       EmberApsFrame *           apsFrame,
                                                       int16u                    msgLen,
                                                       int8u *                   message,
                                                       EmberStatus               status
                                                     )
```

Client Message Sent.

Door Lock cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

boolean emberAfWindowCoveringClusterWindowCoveringUpOpenCallback (void)

Window Covering Up Open.

boolean emberAfWindowCoveringClusterWindowCoveringDownCloseCallback (void)

Window Covering Down Close.

boolean emberAfWindowCoveringClusterWindowCoveringStopCallback (void)

Window Covering Stop.

boolean emberAfWindowCoveringClusterWindowCoveringGoToLiftSetpointCallback (int8u indexOfLiftSetpoint)

Window Covering Go To Lift Setpoint.

Parameters:

indexOfLiftSetpoint	Ver.: always

boolean emberAfWindowCoveringClusterWindowCoveringGoToLiftValueCallback (int16u liftValue)

Window Covering Go To Lift Value.

Parameters:

liftValue	Ver.: always

boolean emberAfWindowCoveringClusterWindowCoveringGoToLiftPercentageCallback (int8u percentageLiftValue)

Window Covering Go To Lift Percentage.

Parameters:

percentageLiftValue Ver.: always

boolean emberAfWindowCoveringClusterWindowCoveringGoToTiltSetpointCallback (int8u indexOfTiltSetpoint)

Window Covering Go To Tilt Setpoint.

Parameters:

indexOfTiltSetpoint Ver.: always

boolean emberAfWindowCoveringClusterWindowCoveringGoToTiltValueCallback (int16u tiltValue)

Window Covering Go To Tilt Value.

Parameters:

tiltValue Ver.: always

boolean emberAfWindowCoveringClusterWindowCoveringGoToTiltPercentageCallback (int8u percentageTiltValue)

Window Covering Go To Tilt Percentage.

Parameters:

percentageTiltValue Ver.: always

boolean emberAfWindowCoveringClusterWindowCoveringGoToProgramSetpointCallback (int8u setpointType, int8u setpointIndex, int16u setpointValue)

Window Covering Go To Program Setpoint.

Parameters:

setpointType Ver.: always

setpointIndex Ver.: always

setpointValue Ver.: always

void emberAfWindowCoveringClusterServerInitCallback (int8u endpoint)

Server Init.

Window Covering cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfWindowCoveringClusterClientInitCallback (int8u endpoint)

Client Init.

Window Covering cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfWindowCoveringClusterServerTickCallback (int8u endpoint)

Server Tick.

Window Covering cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfWindowCoveringClusterClientTickCallback (int8u endpoint)

Client Tick.

Window Covering cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfWindowCoveringClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Window Covering cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfWindowCoveringClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                                  EmberAfAttributeId attributeId
                                                                )
```

Client Attribute Changed.

Window Covering cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfWindowCoveringClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                       EmberAfAttributeId attributeId,
                                                                                       int16u             manufacturerCode
                                                                                     )
```

Server Manufacturer Specific Attribute Changed.

Window Covering cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfWindowCoveringClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                      EmberAfAttributeId attributeId,
                                                                                      int16u             manufacturerCode
                                                                                    )
```

Client Manufacturer Specific Attribute Changed.

Window Covering cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfWindowCoveringClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                 int8u          commandId,
                                                                 EmberAfStatus  status
                                                               )
```

Server Default Response.

Window Covering cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfWindowCoveringClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                 int8u          commandId,
                                                                 EmberAfStatus  status
                                                               )
```

Client Default Response.

Window Covering cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfWindowCoveringClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                             int16u                    indexOrDestination,
                                                             EmberApsFrame *           apsFrame,
                                                             int16u                    msgLen,
                                                             int8u *                   message,
                                                             EmberStatus               status
                                                           )
```

Server Message Sent.

Window Covering cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfWindowCoveringClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                             int16u indexOrDestination,
                                                             EmberApsFrame * apsFrame,
                                                             int16u msgLen,
                                                             int8u * message,
                                                             EmberStatus status
                                                           )
```

Client Message Sent.

Window Covering cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfGenericTunnelClusterMatchProtocolAddressCallback ( int8u * protocolAddress )
```

Match Protocol Address.

Parameters:

protocolAddress	Ver.: always

```
boolean emberAfGenericTunnelClusterMatchProtocolAddressResponseCallback ( void )
```

Match Protocol Address Response.

boolean emberAfGenericTunnelClusterAdvertiseProtocolAddressCallback (int8u * protocolAddress)

Advertise Protocol Address.

Parameters:

protocolAddress Ver.: always

void emberAfGenericTunnelClusterServerInitCallback (int8u endpoint)

Server Init.

Generic Tunnel cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfGenericTunnelClusterClientInitCallback (int8u endpoint)

Client Init.

Generic Tunnel cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfGenericTunnelClusterServerTickCallback (int8u endpoint)

Server Tick.

Generic Tunnel cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfGenericTunnelClusterClientTickCallback (int8u endpoint)

Client Tick.

Generic Tunnel cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

```
void emberAfGenericTunnelClusterServerAttributeChangedCallback ( int8u              endpoint,
                                                                 EmberAfAttributeId attributeId
                                                               )
```

Server Attribute Changed.

Generic Tunnel cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

```
void emberAfGenericTunnelClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                                 EmberAfAttributeId attributeId
                                                               )
```

Client Attribute Changed.

Generic Tunnel cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

```
void emberAfGenericTunnelClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                     EmberAfAttributeId attributeId,
                                                                                     int16u             manufacturerCode
                                                                                   )
```

Server Manufacturer Specific Attribute Changed.

Generic Tunnel cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

manufacturerCode Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfGenericTunnelClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                      EmberAfAttributeId attributeId,
                                                                                      int16u             manufacturerCode
                                                                                    )
```

Client Manufacturer Specific Attribute Changed.

Generic Tunnel cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfGenericTunnelClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                int8u          commandId,
                                                                EmberAfStatus  status
                                                              )
```

Server Default Response.

Generic Tunnel cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfGenericTunnelClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                int8u          commandId,
                                                                EmberAfStatus  status
                                                              )
```

Client Default Response.

Generic Tunnel cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfGenericTunnelClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                            int16u                   indexOrDestination,
                                                            EmberApsFrame *          apsFrame,
                                                            int16u                   msgLen,
                                                            int8u *                  message,
                                                            EmberStatus              status
                                                          )
```

Server Message Sent.

Generic Tunnel cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfGenericTunnelClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                            int16u                   indexOrDestination,
                                                            EmberApsFrame *          apsFrame,
                                                            int16u                   msgLen,
                                                            int8u *                  message,
                                                            EmberStatus              status
                                                          )
```

Client Message Sent.

Generic Tunnel cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfBacnetProtocolTunnelClusterTransferNpduCallback ( int8u * NPDU )
```

Transfer Npdu.

Parameters:

NPDU Ver.:
always

void emberAfBacnetProtocolTunnelClusterServerInitCallback (int8u endpoint)

Server Init.

BACnet Protocol Tunnel cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.:
always

void emberAfBacnetProtocolTunnelClusterClientInitCallback (int8u endpoint)

Client Init.

BACnet Protocol Tunnel cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.:
always

void emberAfBacnetProtocolTunnelClusterServerTickCallback (int8u endpoint)

Server Tick.

BACnet Protocol Tunnel cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.:
always

void emberAfBacnetProtocolTunnelClusterClientTickCallback (int8u endpoint)

Client Tick.

BACnet Protocol Tunnel cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.:
always

```
void emberAfBacnetProtocolTunnelClusterServerAttributeChangedCallback ( int8u            endpoint,
                                                                        EmberAfAttributeId attributeId
                                                                      )
```

Server Attribute Changed.

BACnet Protocol Tunnel cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfBacnetProtocolTunnelClusterClientAttributeChangedCallback ( int8u            endpoint,
                                                                        EmberAfAttributeId attributeId
                                                                      )
```

Client Attribute Changed.

BACnet Protocol Tunnel cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfBacnetProtocolTunnelClusterServerManufacturerSpecificAttributeChangedCallback ( int8u            endpoint,
                                                                                            EmberAfAttributeId attributeId,
                                                                                            int16u            manufacturerCode
                                                                                          )
```

Server Manufacturer Specific Attribute Changed.

BACnet Protocol Tunnel cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfBacnetProtocolTunnelClusterClientManufacturerSpecificAttributeChangedCallback ( int8u            endpoint,
                                                                                            EmberAfAttributeId attributeId,
                                                                                            int16u            manufacturerCode
```

```
)
```

Client Manufacturer Specific Attribute Changed.

BACnet Protocol Tunnel cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfBacnetProtocolTunnelClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                        int8u          commandId,
                                                                        EmberAfStatus  status
                                                                      )
```

Server Default Response.

BACnet Protocol Tunnel cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfBacnetProtocolTunnelClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                       int8u          commandId,
                                                                       EmberAfStatus  status
                                                                     )
```

Client Default Response.

BACnet Protocol Tunnel cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfBacnetProtocolTunnelClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                                   int16u                    indexOrDestination,
```

```
                                                EmberApsFrame *        apsFrame,
                                                int16u                 msgLen,
                                                int8u *                message,
                                                EmberStatus            status
                                              )
```

Server Message Sent.

BACnet Protocol Tunnel cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfBacnetProtocolTunnelClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                                   int16u                    indexOrDestination,
                                                                   EmberApsFrame *           apsFrame,
                                                                   int16u                    msgLen,
                                                                   int8u *                   message,
                                                                   EmberStatus               status
                                                                 )
```

Client Message Sent.

BACnet Protocol Tunnel cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfOtaBootloadClusterImageNotifyCallback ( int8u   payloadType,
                                                       int8u   queryJitter,
                                                       int16u  manufacturerId,
                                                       int16u  imageType,
                                                       int32u  newFileVersion
```

```
)
```

Image Notify.

Parameters:

payloadType	Ver.: always
queryJitter	Ver.: always
manufacturerId	Ver.: always
imageType	Ver.: always
newFileVersion	Ver.: always

```
boolean emberAfOtaBootloadClusterQueryNextImageRequestCallback ( int8u   fieldControl,
                                                                 int16u  manufacturerId,
                                                                 int16u  imageType,
                                                                 int32u  currentFileVersion,
                                                                 int16u  hardwareVersion
                                                               )
```

Query Next Image Request.

Parameters:

fieldControl	Ver.: always
manufacturerId	Ver.: always
imageType	Ver.: always
currentFileVersion	Ver.: always
hardwareVersion	Ver.: always

```
boolean emberAfOtaBootloadClusterQueryNextImageResponseCallback ( int8u   status,
                                                                  int16u  manufacturerId,
                                                                  int16u  imageType,
                                                                  int32u  fileVersion,
                                                                  int32u  imageSize
                                                                )
```

Query Next Image Response.

Parameters:

status	Ver.: always
manufacturerId	Ver.: always
imageType	Ver.: always
fileVersion	Ver.: always
imageSize	Ver.: always

```
boolean emberAfOtaBootloadClusterImageBlockRequestCallback ( int8u    fieldControl,
                                                             int16u   manufacturerId,
                                                             int16u   imageType,
                                                             int32u   fileVersion,
                                                             int32u   fileOffset,
                                                             int8u    maxDataSize,
                                                             int8u *  requestNodeAddress
                                                           )
```

Image Block Request.

Parameters:

fieldControl	Ver.: always
manufacturerId	Ver.: always
imageType	Ver.: always
fileVersion	Ver.: always
fileOffset	Ver.: always
maxDataSize	Ver.: always
requestNodeAddress	Ver.: always

```
boolean emberAfOtaBootloadClusterImagePageRequestCallback ( int8u    fieldControl,
                                                            int16u   manufacturerId,
                                                            int16u   imageType,
                                                            int32u   fileVersion,
                                                            int32u   fileOffset,
                                                            int8u    maxDataSize,
                                                            int16u   pageSize,
                                                            int16u   responseSpacing,
```

```
                                                    int8u *  requestNodeAddress
                                                  )
```

Image Page Request.

Parameters:

fieldControl	Ver.: always
manufacturerId	Ver.: always
imageType	Ver.: always
fileVersion	Ver.: always
fileOffset	Ver.: always
maxDataSize	Ver.: always
pageSize	Ver.: always
responseSpacing	Ver.: always
requestNodeAddress	Ver.: always

```
boolean emberAfOtaBootloadClusterImageBlockResponseCallback ( int8u    status,
                                                              int16u   manufacturerId,
                                                              int16u   imageType,
                                                              int32u   fileVersion,
                                                              int32u   fileOffset,
                                                              int8u    dataSize,
                                                              int8u *  imageData
                                                            )
```

Image Block Response.

Parameters:

status	Ver.: always
manufacturerId	Ver.: always
imageType	Ver.: always
fileVersion	Ver.: always
fileOffset	Ver.: always

dataSize	Ver.: always
imageData	Ver.: always

```
boolean emberAfOtaBootloadClusterUpgradeEndRequestCallback ( int8u   status,
                                                             int16u  manufacturerId,
                                                             int16u  imageType,
                                                             int32u  fileVersion
                                                           )
```

Upgrade End Request.

Parameters:

status	Ver.: always
manufacturerId	Ver.: always
imageType	Ver.: always
fileVersion	Ver.: always

```
boolean emberAfOtaBootloadClusterUpgradeEndResponseCallback ( int16u  manufacturerId,
                                                              int16u  imageType,
                                                              int32u  fileVersion,
                                                              int32u  currentTime,
                                                              int32u  upgradeTime
                                                            )
```

Upgrade End Response.

Parameters:

manufacturerId	Ver.: always
imageType	Ver.: always
fileVersion	Ver.: always
currentTime	Ver.: always
upgradeTime	Ver.: always

```
boolean emberAfOtaBootloadClusterQuerySpecificFileRequestCallback ( int8u *  requestNodeAddress,
                                                                    int16u   manufacturerId,
```

```
                                                    int16u  imageType,
                                                    int32u  fileVersion,
                                                    int16u  currentZigbeeStackVersion
                                                  )
```

Query Specific File Request.

Parameters:

requestNodeAddress	Ver.: always
manufacturerId	Ver.: always
imageType	Ver.: always
fileVersion	Ver.: always
currentZigbeeStackVersion	Ver.: always

```
boolean emberAfOtaBootloadClusterQuerySpecificFileResponseCallback ( int8u   status,
                                                                      int16u  manufacturerId,
                                                                      int16u  imageType,
                                                                      int32u  fileVersion,
                                                                      int32u  imageSize
                                                                    )
```

Query Specific File Response.

Parameters:

status	Ver.: always
manufacturerId	Ver.: always
imageType	Ver.: always
fileVersion	Ver.: always
imageSize	Ver.: always

```
void emberAfOtaBootloadClusterServerInitCallback ( int8u  endpoint )
```

Server Init.

Over the Air Bootloading Cluster cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfOtaBootloadClusterClientInitCallback (int8u endpoint)

Client Init.

Over the Air Bootloading Cluster cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfOtaBootloadClusterServerTickCallback (int8u endpoint)

Server Tick.

Over the Air Bootloading Cluster cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfOtaBootloadClusterClientTickCallback (int8u endpoint)

Client Tick.

Over the Air Bootloading Cluster cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfOtaBootloadClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Over the Air Bootloading Cluster cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfOtaBootloadClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Over the Air Bootloading Cluster cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfOtaBootloadClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Server Manufacturer Specific Attribute Changed.

Over the Air Bootloading Cluster cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

manufacturerCode Manufacturer Code of the attribute that changed Ver.: always

void emberAfOtaBootloadClusterClientManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Client Manufacturer Specific Attribute Changed.

Over the Air Bootloading Cluster cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

manufacturerCode Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfOtaBootloadClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                              int8u          commandId,
                                                              EmberAfStatus  status
                                                            )
```

Server Default Response.

Over the Air Bootloading Cluster cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfOtaBootloadClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                              int8u          commandId,
                                                              EmberAfStatus  status
                                                            )
```

Client Default Response.

Over the Air Bootloading Cluster cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfOtaBootloadClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                          int16u                    indexOrDestination,
                                                          EmberApsFrame *           apsFrame,
                                                          int16u                    msgLen,
                                                          int8u *                   message,
                                                          EmberStatus               status
                                                        )
```

Server Message Sent.

Over the Air Bootloading Cluster cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always

indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfOtaBootloadClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                          int16u                    indexOrDestination,
                                                          EmberApsFrame *           apsFrame,
                                                          int16u                    msgLen,
                                                          int8u *                   message,
                                                          EmberStatus               status
                                                        )
```

Client Message Sent.

Over the Air Bootloading Cluster cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfDemandResponseLoadControlClusterLoadControlEventCallback ( int32u issuerEventId,
                                                                          int16u deviceClass,
                                                                          int8u  utilityEnrollmentGroup,
                                                                          int32u startTime,
                                                                          int16u durationInMinutes,
                                                                          int8u  criticalityLevel,
                                                                          int8u  coolingTemperatureOffset,
                                                                          int8u  heatingTemperatureOffset,
                                                                          int16s coolingTemperatureSetPoint,
                                                                          int16s heatingTemperatureSetPoint,
                                                                          int8s  averageLoadAdjustmentPercentage,
                                                                          int8u  dutyCycle,
                                                                          int8u  eventControl
                                                                        )
```

Load Control Event.

Parameters:

issuerEventId	Ver.: always
deviceClass	Ver.: always
utilityEnrollmentGroup	Ver.: always
startTime	Ver.: always
durationInMinutes	Ver.: always
criticalityLevel	Ver.: always
coolingTemperatureOffset	Ver.: always
heatingTemperatureOffset	Ver.: always
coolingTemperatureSetPoint	Ver.: always
heatingTemperatureSetPoint	Ver.: always
averageLoadAdjustmentPercentage	Ver.: always
dutyCycle	Ver.: always
eventControl	Ver.: always

```
boolean emberAfDemandResponseLoadControlClusterCancelLoadControlEventCallback ( int32u  issuerEventId,
                                                                                 int16u  deviceClass,
                                                                                 int8u   utilityEnrollmentGroup,
                                                                                 int8u   cancelControl,
                                                                                 int32u  effectiveTime
                                                                               )
```

Cancel Load Control Event.

Parameters:

issuerEventId	Ver.: always
deviceClass	Ver.: always
utilityEnrollmentGroup	Ver.: always
cancelControl	Ver.: always
effectiveTime	Ver.: always

boolean emberAfDemandResponseLoadControlClusterCancelAllLoadControlEventsCallback (int8u cancelControl)

Cancel All Load Control Events.

Parameters:

cancelControl	Ver.: always

boolean emberAfDemandResponseLoadControlClusterReportEventStatusCallback (int32u issuerEventId, int8u eventStatus, int32u eventStatusTime, int8u criticalityLevelApplied, int16s coolingTemperatureSetPointApplied, int16s heatingTemperatureSetPointApplied, int8s averageLoadAdjustmentPercentageApplied, int8u dutyCycleApplied, int8u eventControl, int8u signatureType, int8u * signature)

Report Event Status.

Parameters:

issuerEventId	Ver.: always
eventStatus	Ver.: always
eventStatusTime	Ver.: always
criticalityLevelApplied	Ver.: always
coolingTemperatureSetPointApplied	Ver.: always
heatingTemperatureSetPointApplied	Ver.: always
averageLoadAdjustmentPercentageApplied	Ver.: always
dutyCycleApplied	Ver.: always
eventControl	Ver.: always
signatureType	Ver.: always
signature	Ver.: until SE_1.1, 105638r08

boolean emberAfDemandResponseLoadControlClusterGetScheduledEventsCallback (int32u startTime, int8u numberOfEvents)

Get Scheduled Events.

Parameters:

startTime Ver.: always
numberOfEvents Ver.: always

void emberAfDemandResponseLoadControlClusterServerInitCallback (int8u endpoint)

Server Init.

Demand Response and Load Control cluster, Server Init

Parameters:
endpoint Endpoint that is being initialized Ver.: always

void emberAfDemandResponseLoadControlClusterClientInitCallback (int8u endpoint)

Client Init.

Demand Response and Load Control cluster, Client Init

Parameters:
endpoint Endpoint that is being initialized Ver.: always

void emberAfDemandResponseLoadControlClusterServerTickCallback (int8u endpoint)

Server Tick.

Demand Response and Load Control cluster, Server Tick

Parameters:
endpoint Endpoint that is being served Ver.: always

void emberAfDemandResponseLoadControlClusterClientTickCallback (int8u endpoint)

Client Tick.

Demand Response and Load Control cluster, Client Tick

Parameters:
endpoint Endpoint that is being served Ver.: always

```
void emberAfDemandResponseLoadControlClusterServerAttributeChangedCallback ( int8u              endpoint,
                                                                              EmberAfAttributeId attributeId
                                                                            )
```

Server Attribute Changed.

Demand Response and Load Control cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfDemandResponseLoadControlClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                                              EmberAfAttributeId attributeId
                                                                            )
```

Client Attribute Changed.

Demand Response and Load Control cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfDemandResponseLoadControlClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                                  EmberAfAttributeId attributeId,
                                                                                                  int16u             manufacturerCode
                                                                                                )
```

Server Manufacturer Specific Attribute Changed.

Demand Response and Load Control cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfDemandResponseLoadControlClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                                  EmberAfAttributeId attributeId,
                                                                                                  int16u             manufacturerCode
```

```
)
```

Client Manufacturer Specific Attribute Changed.

Demand Response and Load Control cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfDemandResponseLoadControlClusterServerDefaultResponseCallback ( int8u            endpoint,
                                                                            int8u            commandId,
                                                                            EmberAfStatus    status
                                                                          )
```

Server Default Response.

Demand Response and Load Control cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfDemandResponseLoadControlClusterClientDefaultResponseCallback ( int8u            endpoint,
                                                                            int8u            commandId,
                                                                            EmberAfStatus    status
                                                                          )
```

Client Default Response.

Demand Response and Load Control cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfDemandResponseLoadControlClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                                        int16u                     indexOrDestination,
```

```
                                                EmberApsFrame *        apsFrame,
                                                int16u                 msgLen,
                                                int8u *                message,
                                                EmberStatus            status
                                              )
```

Server Message Sent.

Demand Response and Load Control cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfDemandResponseLoadControlClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                                         int16u                 indexOrDestination,
                                                                         EmberApsFrame *        apsFrame,
                                                                         int16u                 msgLen,
                                                                         int8u *                message,
                                                                         EmberStatus            status
                                                                       )
```

Client Message Sent.

Demand Response and Load Control cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfSimpleMeteringClusterGetProfileResponseCallback ( int32u  endTime,
                                                                 int8u   status,
                                                                 int8u   profileIntervalPeriod,
                                                                 int8u   numberOfPeriodsDelivered,
                                                                 int8u * intervals
```

)

Get Profile Response.

Parameters:

endTime	Ver.: always
status	Ver.: always
profileIntervalPeriod	Ver.: always
numberOfPeriodsDelivered	Ver.: always
intervals	Ver.: always

boolean emberAfSimpleMeteringClusterRequestMirrorCallback (void)

Request Mirror.

boolean emberAfSimpleMeteringClusterRemoveMirrorCallback (void)

Remove Mirror.

boolean emberAfSimpleMeteringClusterRequestFastPollModeResponseCallback (int8u appliedUpdatePeriod, int32u fastPollModeEndtime)

Request Fast Poll Mode Response.

Parameters:

appliedUpdatePeriod	Ver.: always
fastPollModeEndtime	Ver.: always

boolean emberAfSimpleMeteringClusterGetProfileCallback (int8u intervalChannel, int32u endTime, int8u numberOfPeriods)

Get Profile.

Parameters:

intervalChannel Ver.: always
endTime Ver.: always
numberOfPeriods Ver.: always

boolean emberAfSimpleMeteringClusterRequestMirrorResponseCallback (int16u endpointId)

Request Mirror Response.

Parameters:
endpointId Ver.: always

boolean emberAfSimpleMeteringClusterMirrorRemovedCallback (int16u endpointId)

Mirror Removed.

Parameters:
endpointId Ver.: always

boolean emberAfSimpleMeteringClusterRequestFastPollModeCallback (int8u fastPollUpdatePeriod, int8u duration)

Request Fast Poll Mode.

Parameters:
fastPollUpdatePeriod Ver.: always
duration Ver.: always

void emberAfSimpleMeteringClusterServerInitCallback (int8u endpoint)

Server Init.

Simple Metering cluster, Server Init

Parameters:
endpoint Endpoint that is being initialized Ver.: always

void emberAfSimpleMeteringClusterClientInitCallback (int8u endpoint)

Client Init.

Simple Metering cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfSimpleMeteringClusterServerTickCallback (int8u endpoint)

Server Tick.

Simple Metering cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfSimpleMeteringClusterClientTickCallback (int8u endpoint)

Client Tick.

Simple Metering cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfSimpleMeteringClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Simple Metering cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

void emberAfSimpleMeteringClusterClientAttributeChangedCallback (int8u endpoint,

```
                                                EmberAfAttributeId attributeId
                                              )
```

Client Attribute Changed.

Simple Metering cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfSimpleMeteringClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                       EmberAfAttributeId attributeId,
                                                                                       int16u             manufacturerCode
                                                                                     )
```

Server Manufacturer Specific Attribute Changed.

Simple Metering cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfSimpleMeteringClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                       EmberAfAttributeId attributeId,
                                                                                       int16u             manufacturerCode
                                                                                     )
```

Client Manufacturer Specific Attribute Changed.

Simple Metering cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfSimpleMeteringClusterServerDefaultResponseCallback ( int8u    endpoint,
                                                                 int8u    commandId,
```

```
                                                EmberAfStatus status
                                              )
```

Server Default Response.

Simple Metering cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfSimpleMeteringClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                 int8u          commandId,
                                                                 EmberAfStatus  status
                                                               )
```

Client Default Response.

Simple Metering cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfSimpleMeteringClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                             int16u                    indexOrDestination,
                                                             EmberApsFrame *           apsFrame,
                                                             int16u                    msgLen,
                                                             int8u *                   message,
                                                             EmberStatus               status
                                                           )
```

Server Message Sent.

Simple Metering cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always

msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfSimpleMeteringClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                             int16u                    indexOrDestination,
                                                             EmberApsFrame *           apsFrame,
                                                             int16u                    msgLen,
                                                             int8u *                   message,
                                                             EmberStatus               status
                                                           )
```

Client Message Sent.

Simple Metering cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfPriceClusterPublishPriceCallback ( int32u  providerId,
                                                  int8u *  rateLabel,
                                                  int32u  issuerEventId,
                                                  int32u  currentTime,
                                                  int8u   unitOfMeasure,
                                                  int16u  currency,
                                                  int8u   priceTrailingDigitAndPriceTier,
                                                  int8u   numberOfPriceTiersAndRegisterTier,
                                                  int32u  startTime,
                                                  int16u  durationInMinutes,
                                                  int32u  price,
                                                  int8u   priceRatio,
                                                  int32u  generationPrice,
                                                  int8u   generationPriceRatio,
                                                  int32u  alternateCostDelivered,
                                                  int8u   alternateCostUnit,
                                                  int8u   alternateCostTrailingDigit,
                                                  int8u   numberOfBlockThresholds,
                                                  int8u   priceControl
```

```
)
```

Publish Price.

Parameters:

providerId	Ver.: always
rateLabel	Ver.: always
issuerEventId	Ver.: always
currentTime	Ver.: always
unitOfMeasure	Ver.: always
currency	Ver.: always
priceTrailingDigitAndPriceTier	Ver.: always
numberOfPriceTiersAndRegisterTier	Ver.: always
startTime	Ver.: always
durationInMinutes	Ver.: always
price	Ver.: always
priceRatio	Ver.: always
generationPrice	Ver.: always
generationPriceRatio	Ver.: always
alternateCostDelivered	Ver.: since SE_1.0, 075356r15
alternateCostUnit	Ver.: since SE_1.0, 075356r15
alternateCostTrailingDigit	Ver.: since SE_1.0, 075356r15
numberOfBlockThresholds	Ver.: since SE_1.1, 105638r09
priceControl	Ver.: since SE_1.1, 105638r09

```
boolean emberAfPriceClusterPublishBlockPeriodCallback ( int32u  providerId,
                                                        int32u  issuerEventId,
                                                        int32u  blockPeriodStartTime,
                                                        int32u  blockPeriodDurationInMinutes,
                                                        int8u   numberOfPriceTiersAndNumberOfBlockThresholds,
                                                        int8u   blockPeriodControl
                                                      )
```

Publish Block Period.

Parameters:

providerId	Ver.: always
issuerEventId	Ver.: always
blockPeriodStartTime	Ver.: always

blockPeriodDurationInMinutes	Ver.: always
numberOfPriceTiersAndNumberOfBlockThresholds	Ver.: always
blockPeriodControl	Ver.: always

boolean emberAfPriceClusterGetCurrentPriceCallback (int8u commandOptions)

Get Current Price.

Parameters:

commandOptions	Ver.: always

boolean emberAfPriceClusterGetScheduledPricesCallback (int32u startTime, int8u numberOfEvents)

Get Scheduled Prices.

Parameters:

startTime	Ver.: always
numberOfEvents	Ver.: always

boolean emberAfPriceClusterPriceAcknowledgementCallback (int32u providerId, int32u issuerEventId, int32u priceAckTime, int8u control)

Price Acknowledgement.

Parameters:

providerId	Ver.: always
issuerEventId	Ver.: always
priceAckTime	Ver.: always
control	Ver.: always

boolean emberAfPriceClusterGetBlockPeriodsCallback (int32u startTime, int8u numberOfEvents)

Get Block Periods.

Parameters:

startTime	Ver.: always
numberOfEvents	Ver.: always

void emberAfPriceClusterServerInitCallback (int8u endpoint)

Server Init.

Price cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfPriceClusterClientInitCallback (int8u endpoint)

Client Init.

Price cluster, Client Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfPriceClusterServerTickCallback (int8u endpoint)

Server Tick.

Price cluster, Server Tick

Parameters:

endpoint	Endpoint that is being served Ver.: always

void emberAfPriceClusterClientTickCallback (int8u endpoint)

Client Tick.

Price cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

---

void emberAfPriceClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Price cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

---

void emberAfPriceClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Price cluster, Client Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

---

void emberAfPriceClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Server Manufacturer Specific Attribute Changed.

Price cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfPriceClusterClientManufacturerSpecificAttributeChangedCallback ( int8u               endpoint,
                                                                             EmberAfAttributeId  attributeId,
                                                                             int16u              manufacturerCode
                                                                           )
```

Client Manufacturer Specific Attribute Changed.

Price cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfPriceClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                        int8u          commandId,
                                                        EmberAfStatus  status
                                                      )
```

Server Default Response.

Price cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfPriceClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                        int8u          commandId,
                                                        EmberAfStatus  status
                                                      )
```

Client Default Response.

Price cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfPriceClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                    int16u                   indexOrDestination,
                                                    EmberApsFrame *          apsFrame,
                                                    int16u                   msgLen,
                                                    int8u *                  message,
                                                    EmberStatus              status
                                                  )
```

Server Message Sent.

Price cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfPriceClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                    int16u                   indexOrDestination,
                                                    EmberApsFrame *          apsFrame,
                                                    int16u                   msgLen,
                                                    int8u *                  message,
                                                    EmberStatus              status
                                                  )
```

Client Message Sent.

Price cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always

status	The status of the sent message Ver.: always

```
boolean emberAfMessagingClusterDisplayMessageCallback ( int32u  messageId,
                                                        int8u   messageControl,
                                                        int32u  startTime,
                                                        int16u  durationInMinutes,
                                                        int8u * message
                                                      )
```

Display Message.

Parameters:

messageId	Ver.: always
messageControl	Ver.: always
startTime	Ver.: always
durationInMinutes	Ver.: always
message	Ver.: always

```
boolean emberAfMessagingClusterCancelMessageCallback ( int32u  messageId,
                                                       int8u   messageControl
                                                     )
```

Cancel Message.

Parameters:

messageId	Ver.: always
messageControl	Ver.: always

```
boolean emberAfMessagingClusterGetLastMessageCallback ( void  )
```

Get Last Message.

```
boolean emberAfMessagingClusterMessageConfirmationCallback ( int32u  messageId,
                                                             int32u  confirmationTime
                                                           )
```

Message Confirmation.

Parameters:

messageId	Ver.: always
confirmationTime	Ver.: always

void emberAfMessagingClusterServerInitCallback (int8u endpoint)

Server Init.

Messaging cluster, Server Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfMessagingClusterClientInitCallback (int8u endpoint)

Client Init.

Messaging cluster, Client Init

Parameters:

endpoint	Endpoint that is being initialized Ver.: always

void emberAfMessagingClusterServerTickCallback (int8u endpoint)

Server Tick.

Messaging cluster, Server Tick

Parameters:

endpoint	Endpoint that is being served Ver.: always

void emberAfMessagingClusterClientTickCallback (int8u endpoint)

Client Tick.

Messaging cluster, Client Tick

Parameters:

endpoint	Endpoint that is being served Ver.: always

```
void emberAfMessagingClusterServerAttributeChangedCallback ( int8u              endpoint,
                                                             EmberAfAttributeId attributeId
                                                           )
```

Server Attribute Changed.

Messaging cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfMessagingClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                             EmberAfAttributeId attributeId
                                                           )
```

Client Attribute Changed.

Messaging cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfMessagingClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                  EmberAfAttributeId attributeId,
                                                                                  int16u             manufacturerCode
                                                                                )
```

Server Manufacturer Specific Attribute Changed.

Messaging cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfMessagingClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                  EmberAfAttributeId attributeId,
```

```
                                                                    int16u          manufacturerCode
                                                                  )
```

Client Manufacturer Specific Attribute Changed.

Messaging cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfMessagingClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                            int8u          commandId,
                                                            EmberAfStatus  status
                                                          )
```

Server Default Response.

Messaging cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfMessagingClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                            int8u          commandId,
                                                            EmberAfStatus  status
                                                          )
```

Client Default Response.

Messaging cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfMessagingClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
```

```
                                                int16u                  indexOrDestination,
                                                EmberApsFrame *         apsFrame,
                                                int16u                  msgLen,
                                                int8u *                 message,
                                                EmberStatus             status
                                              )
```

Server Message Sent.

Messaging cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfMessagingClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                        int16u                   indexOrDestination,
                                                        EmberApsFrame *          apsFrame,
                                                        int16u                   msgLen,
                                                        int8u *                  message,
                                                        EmberStatus              status
                                                      )
```

Client Message Sent.

Messaging cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfTunnelingClusterRequestTunnelCallback ( int8u   protocolId,
                                                       int16u  manufacturerCode,
                                                       int8u   flowControlSupport
                                                     )
```

Request Tunnel.

Parameters:

protocolId	Ver.: always
manufacturerCode	Ver.: always
flowControlSupport	Ver.: always

boolean emberAfTunnelingClusterCloseTunnelCallback (int16u tunnelId)

Close Tunnel.

Parameters:

tunnelId	Ver.: always

boolean emberAfTunnelingClusterTransferDataClientToServerCallback (int16u tunnelId, int8u * data)

Transfer Data Client To Server.

Parameters:

tunnelId	Ver.: always
data	Ver.: always

boolean emberAfTunnelingClusterTransferDataErrorClientToServerCallback (int16u tunnelId, int8u transferDataStatus)

Transfer Data Error Client To Server.

Parameters:

tunnelId	Ver.: always
transferDataStatus	Ver.: always

boolean emberAfTunnelingClusterAckTransferDataClientToServerCallback (int16u tunnelId, int16u numberOfBytesLeft

```
)
```

Ack Transfer Data Client To Server.

Parameters:

tunnelId	Ver.: always
numberOfBytesLeft	Ver.: always

```
boolean emberAfTunnelingClusterReadyDataClientToServerCallback ( int16u  tunnelId,
                                                                 int16u  numberOfOctetsLeft
                                                               )
```

Ready Data Client To Server.

Parameters:

tunnelId	Ver.: always
numberOfOctetsLeft	Ver.: always

```
boolean emberAfTunnelingClusterRequestTunnelResponseCallback ( int16u  tunnelId,
                                                               int8u   tunnelStatus
                                                             )
```

Request Tunnel Response.

Parameters:

tunnelId	Ver.: always
tunnelStatus	Ver.: always

```
boolean emberAfTunnelingClusterTransferDataServerToClientCallback ( int16u   tunnelId,
                                                                    int8u *  data
                                                                  )
```

Transfer Data Server To Client.

Parameters:

tunnelId	Ver.: always
data	Ver.: always

boolean emberAfTunnelingClusterTransferDataErrorServerToClientCallback (int16u tunnelId, int8u transferDataStatus)

Transfer Data Error Server To Client.

Parameters:

tunnelId	Ver.: always
transferDataStatus	Ver.: always

boolean emberAfTunnelingClusterAckTransferDataServerToClientCallback (int16u tunnelId, int16u numberOfBytesLeft)

Ack Transfer Data Server To Client.

Parameters:

tunnelId	Ver.: always
numberOfBytesLeft	Ver.: always

boolean emberAfTunnelingClusterReadyDataServerToClientCallback (int16u tunnelId, int16u numberOfOctetsLeft)

Ready Data Server To Client.

Parameters:

tunnelId	Ver.: always
numberOfOctetsLeft	Ver.: always

void emberAfTunnelingClusterServerInitCallback (int8u endpoint)

Server Init.

Tunneling cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfTunnelingClusterClientInitCallback (int8u endpoint)

Client Init.

Tunneling cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfTunnelingClusterServerTickCallback (int8u endpoint)

Server Tick.

Tunneling cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfTunnelingClusterClientTickCallback (int8u endpoint)

Client Tick.

Tunneling cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfTunnelingClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Tunneling cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always

attributeId Attribute that changed Ver.: always

```
void emberAfTunnelingClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                             EmberAfAttributeId attributeId
                                                           )
```

Client Attribute Changed.

Tunneling cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfTunnelingClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                  EmberAfAttributeId attributeId,
                                                                                  int16u             manufacturerCode
                                                                                )
```

Server Manufacturer Specific Attribute Changed.

Tunneling cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfTunnelingClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                 EmberAfAttributeId attributeId,
                                                                                 int16u             manufacturerCode
                                                                               )
```

Client Manufacturer Specific Attribute Changed.

Tunneling cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfTunnelingClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                            int8u          commandId,
                                                            EmberAfStatus  status
                                                          )
```

Server Default Response.

Tunneling cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfTunnelingClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                            int8u          commandId,
                                                            EmberAfStatus  status
                                                          )
```

Client Default Response.

Tunneling cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfTunnelingClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                        int16u                    indexOrDestination,
                                                        EmberApsFrame *           apsFrame,
                                                        int16u                    msgLen,
                                                        int8u *                   message,
                                                        EmberStatus               status
                                                      )
```

Server Message Sent.

Tunneling cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always

indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfTunnelingClusterClientMessageSentCallback ( EmberOutgoingMessageType  type,
                                                        int16u                    indexOrDestination,
                                                        EmberApsFrame *           apsFrame,
                                                        int16u                    msgLen,
                                                        int8u *                   message,
                                                        EmberStatus               status
                                                      )
```

Client Message Sent.

Tunneling cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfPrepaymentClusterSelectAvailableEmergencyCreditCallback ( int32u  commandDateTime,
                                                                          int8u   originatingDevice,
                                                                          int8u * siteId,
                                                                          int8u * meterSerialNumber
                                                                        )
```

Select Available Emergency Credit.

Parameters:

commandDateTime	Ver.: always
originatingDevice	Ver.: always
siteId	Ver.: always
meterSerialNumber	Ver.: always

```
boolean emberAfPrepaymentClusterChangeSupplyCallback ( int32u  providerId,
                                                       int32u  requestDateTime,
                                                       int8u * siteId,
                                                       int8u * meterSerialNumber,
                                                       int32u  implementationDateTime,
                                                       int8u   proposedSupplyStatus,
                                                       int8u   originatorIdSupplyControlBits
                                                     )
```

Change Supply.

Parameters:

providerId	Ver.: always
requestDateTime	Ver.: always
siteId	Ver.: always
meterSerialNumber	Ver.: always
implementationDateTime	Ver.: always
proposedSupplyStatus	Ver.: always
originatorIdSupplyControlBits	Ver.: always

```
boolean emberAfPrepaymentClusterSupplyStatusResponseCallback ( int32u providerId,
                                                               int32u implementationtDateTime,
                                                               int8u  supplyStatus
                                                             )
```

Supply Status Response.

Parameters:

providerId	Ver.: always
implementationtDateTime	Ver.: always
supplyStatus	Ver.: always

```
void emberAfPrepaymentClusterServerInitCallback ( int8u endpoint )
```

Server Init.

Prepayment cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfPrepaymentClusterClientInitCallback (int8u endpoint)

Client Init.

Prepayment cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfPrepaymentClusterServerTickCallback (int8u endpoint)

Server Tick.

Prepayment cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfPrepaymentClusterClientTickCallback (int8u endpoint)

Client Tick.

Prepayment cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfPrepaymentClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Prepayment cluster, Server Attribute Changed

Parameters:

endpoint Endpoint that is being initialized Ver.: always
attributeId Attribute that changed Ver.: always

void emberAfPrepaymentClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Client Attribute Changed.

Prepayment cluster, Client Attribute Changed

Parameters:
endpoint Endpoint that is being initialized Ver.: always
attributeId Attribute that changed Ver.: always

void emberAfPrepaymentClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Server Manufacturer Specific Attribute Changed.

Prepayment cluster, Server Manufacturer Specific Attribute Changed

Parameters:
endpoint Endpoint that is being initialized Ver.: always
attributeId Attribute that changed Ver.: always
manufacturerCode Manufacturer Code of the attribute that changed Ver.: always

void emberAfPrepaymentClusterClientManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Client Manufacturer Specific Attribute Changed.

Prepayment cluster, Client Manufacturer Specific Attribute Changed

Parameters:
endpoint Endpoint that is being initialized Ver.: always
attributeId Attribute that changed Ver.: always

manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfPrepaymentClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                             int8u          commandId,
                                                             EmberAfStatus  status
                                                           )
```

Server Default Response.

Prepayment cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfPrepaymentClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                             int8u          commandId,
                                                             EmberAfStatus  status
                                                           )
```

Client Default Response.

Prepayment cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfPrepaymentClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                         int16u                     indexOrDestination,
                                                         EmberApsFrame *            apsFrame,
                                                         int16u                     msgLen,
                                                         int8u *                    message,
                                                         EmberStatus                status
                                                       )
```

Server Message Sent.

Prepayment cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfPrepaymentClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                         int16u                    indexOrDestination,
                                                         EmberApsFrame *           apsFrame,
                                                         int16u                    msgLen,
                                                         int8u *                   message,
                                                         EmberStatus               status
                                                       )
```

Client Message Sent.

Prepayment cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfKeyEstablishmentClusterClientCommandReceivedCallback ( EmberAfClusterCommand * cmd )
```

Client Command Received.

This function is called by the application framework when a server-to-client key establishment command is received but has yet to be handled by the framework code. This function should return a boolean value indicating whether the command has been handled by the application code and should not be further processed by the framework.

Parameters:

cmd Ver.: always

```
boolean emberAfKeyEstablishmentClusterServerCommandReceivedCallback ( EmberAfClusterCommand * cmd )
```

Server Command Received.

This function is called by the application framework when a client-to-server key establishment command is received but has yet to be handled by the framework code. This function should return a boolean value indicating whether the command has been handled by the application code and should not be further processed by the framework.

Parameters:

cmd Ver.: always

boolean emberAfKeyEstablishmentClusterInitiateKeyEstablishmentRequestCallback (int16u keyEstablishmentSuite, int8u ephemeralDataGenerateTime, int8u confirmKeyGenerateTime, int8u * identity)

Initiate Key Establishment Request.

Parameters:

keyEstablishmentSuite	Ver.: always
ephemeralDataGenerateTime	Ver.: always
confirmKeyGenerateTime	Ver.: always
identity	Ver.: always

boolean emberAfKeyEstablishmentClusterEphemeralDataRequestCallback (int8u * ephemeralData)

Ephemeral Data Request.

Parameters:

ephemeralData Ver.: always

boolean emberAfKeyEstablishmentClusterConfirmKeyDataRequestCallback (int8u * secureMessageAuthenticationCode)

Confirm Key Data Request.

Parameters:

secureMessageAuthenticationCode Ver.: always

boolean emberAfKeyEstablishmentClusterTerminateKeyEstablishmentCallback (int8u statusCode,

```
                                                                    int8u   waitTime,
                                                                    int16u  keyEstablishmentSuite
                                                                  )
```

Terminate Key Establishment.

Parameters:

statusCode	Ver.: always
waitTime	Ver.: always
keyEstablishmentSuite	Ver.: always

```
boolean emberAfKeyEstablishmentClusterInitiateKeyEstablishmentResponseCallback ( int16u  requestedKeyEstablishmentSuite,
                                                                                  int8u   ephemeralDataGenerateTime,
                                                                                  int8u   confirmKeyGenerateTime,
                                                                                  int8u * identity
                                                                                )
```

Initiate Key Establishment Response.

Parameters:

requestedKeyEstablishmentSuite	Ver.: always
ephemeralDataGenerateTime	Ver.: always
confirmKeyGenerateTime	Ver.: always
identity	Ver.: always

```
boolean emberAfKeyEstablishmentClusterEphemeralDataResponseCallback ( int8u * ephemeralData )
```

Ephemeral Data Response.

Parameters:

ephemeralData	Ver.: always

```
boolean emberAfKeyEstablishmentClusterConfirmKeyDataResponseCallback ( int8u * secureMessageAuthenticationCode )
```

Confirm Key Data Response.

Parameters:

secureMessageAuthenticationCode Ver.: always

void emberAfKeyEstablishmentClusterServerInitCallback (int8u endpoint)

Server Init.

Key establishment cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfKeyEstablishmentClusterClientInitCallback (int8u endpoint)

Client Init.

Key establishment cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfKeyEstablishmentClusterServerTickCallback (int8u endpoint)

Server Tick.

Key establishment cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfKeyEstablishmentClusterClientTickCallback (int8u endpoint)

Client Tick.

Key establishment cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfKeyEstablishmentClusterServerAttributeChangedCallback (int8u endpoint,

```
                                                EmberAfAttributeId attributeId
                                              )
```

Server Attribute Changed.

Key establishment cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfKeyEstablishmentClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                                    EmberAfAttributeId attributeId
                                                                  )
```

Client Attribute Changed.

Key establishment cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfKeyEstablishmentClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                        EmberAfAttributeId attributeId,
                                                                                        int16u             manufacturerCode
                                                                                      )
```

Server Manufacturer Specific Attribute Changed.

Key establishment cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfKeyEstablishmentClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                        EmberAfAttributeId attributeId,
                                                                                        int16u             manufacturerCode
                                                                                      )
```

Client Manufacturer Specific Attribute Changed.

Key establishment cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfKeyEstablishmentClusterServerDefaultResponseCallback ( int8u          endpoint,
                                                                   int8u          commandId,
                                                                   EmberAfStatus  status
                                                                 )
```

Server Default Response.

Key establishment cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfKeyEstablishmentClusterClientDefaultResponseCallback ( int8u          endpoint,
                                                                   int8u          commandId,
                                                                   EmberAfStatus  status
                                                                 )
```

Client Default Response.

Key establishment cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfKeyEstablishmentClusterServerMessageSentCallback ( EmberOutgoingMessageType  type,
                                                               int16u                    indexOrDestination,
                                                               EmberApsFrame *           apsFrame,
                                                               int16u                    msgLen,
```

```
                                                    int8u *                 message,
                                                    EmberStatus             status
                                                  )
```

Server Message Sent.

Key establishment cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfKeyEstablishmentClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                               int16u                     indexOrDestination,
                                                               EmberApsFrame *            apsFrame,
                                                               int16u                     msgLen,
                                                               int8u *                    message,
                                                               EmberStatus                status
                                                             )
```

Client Message Sent.

Key establishment cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
boolean emberAfSampleMfgSpecificClusterCommandOneCallback ( int8u argOne )
```

Command One.

Parameters:

argOne	Ver.: always

void emberAfSampleMfgSpecificClusterServerInitCallback (int8u endpoint)

Server Init.

Sample Mfg Specific Cluster cluster, Server Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfSampleMfgSpecificClusterClientInitCallback (int8u endpoint)

Client Init.

Sample Mfg Specific Cluster cluster, Client Init

Parameters:

endpoint Endpoint that is being initialized Ver.: always

void emberAfSampleMfgSpecificClusterServerTickCallback (int8u endpoint)

Server Tick.

Sample Mfg Specific Cluster cluster, Server Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfSampleMfgSpecificClusterClientTickCallback (int8u endpoint)

Client Tick.

Sample Mfg Specific Cluster cluster, Client Tick

Parameters:

endpoint Endpoint that is being served Ver.: always

void emberAfSampleMfgSpecificClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)

Server Attribute Changed.

Sample Mfg Specific Cluster cluster, Server Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfSampleMfgSpecificClusterClientAttributeChangedCallback ( int8u              endpoint,
                                                                      EmberAfAttributeId attributeId
                                                                    )
```

Client Attribute Changed.

Sample Mfg Specific Cluster cluster, Client Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always

```
void emberAfSampleMfgSpecificClusterServerManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                          EmberAfAttributeId attributeId,
                                                                                          int16u             manufacturerCode
                                                                                        )
```

Server Manufacturer Specific Attribute Changed.

Sample Mfg Specific Cluster cluster, Server Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfSampleMfgSpecificClusterClientManufacturerSpecificAttributeChangedCallback ( int8u              endpoint,
                                                                                          EmberAfAttributeId attributeId,
                                                                                          int16u             manufacturerCode
                                                                                        )
```

Client Manufacturer Specific Attribute Changed.

Sample Mfg Specific Cluster cluster, Client Manufacturer Specific Attribute Changed

Parameters:

endpoint	Endpoint that is being initialized Ver.: always
attributeId	Attribute that changed Ver.: always
manufacturerCode	Manufacturer Code of the attribute that changed Ver.: always

```
void emberAfSampleMfgSpecificClusterServerDefaultResponseCallback ( int8u           endpoint,
                                                                     int8u           commandId,
                                                                     EmberAfStatus   status
                                                                   )
```

Server Default Response.

Sample Mfg Specific Cluster cluster, Server Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfSampleMfgSpecificClusterClientDefaultResponseCallback ( int8u           endpoint,
                                                                     int8u           commandId,
                                                                     EmberAfStatus   status
                                                                   )
```

Client Default Response.

Sample Mfg Specific Cluster cluster, Client Default Response

Parameters:

endpoint	Destination endpoint Ver.: always
commandId	Command id Ver.: always
status	Status in default response Ver.: always

```
void emberAfSampleMfgSpecificClusterServerMessageSentCallback ( EmberOutgoingMessageType type,
                                                                int16u                    indexOrDestination,
                                                                EmberApsFrame *           apsFrame,
                                                                int16u                    msgLen,
                                                                int8u *                   message,
                                                                EmberStatus               status
                                                              )
```

Server Message Sent.

Sample Mfg Specific Cluster cluster, Server Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfSampleMfgSpecificClusterClientMessageSentCallback ( EmberOutgoingMessageType type,
                                                                int16u                     indexOrDestination,
                                                                EmberApsFrame *            apsFrame,
                                                                int16u                     msgLen,
                                                                int8u *                    message,
                                                                EmberStatus                status
                                                              )
```

Client Message Sent.

Sample Mfg Specific Cluster cluster, Client Message Sent

Parameters:

type	The type of message sent Ver.: always
indexOrDestination	The destination or address to which the message was sent Ver.: always
apsFrame	The APS frame for the message Ver.: always
msgLen	The length of the message Ver.: always
message	The message that was sent Ver.: always
status	The status of the sent message Ver.: always

```
void emberAfPluginBasicResetToFactoryDefaultsCallback ( void )
```

Reset To Factory Defaults.

This function is called by the Basic server plugin when a request to reset to factory defaults is received. The plugin will reset attributes managed by the framework to their default values. The application should preform any other necessary reset-related operations in this callback, including resetting any externally-stored attributes.

```
boolean emberAfPluginDrlcEventActionCallback ( EmberAfLoadControlEvent * loadControlEvent,
```

```
                                    EmberAfAmiEventStatus    eventStatus,
                                    int8u                    sequenceNumber
                                )
```

Event Action.

This function is called by the demand response and load control client plugin whenever an event status changes within the DRLC event table. The list of possible event status values is defined by the ZCL spec and is listed in the Application Framework's generated enums located at **app/framework/gen/enums.h**.
For example, an event status may be: AMI_EVENT_STATUS_LOAD_CONTROL_EVENT_COMMAND_RX indicating that a properly formatted event was received; AMI_EVENT_STATUS_EVENT_STARTED indicating that an event has started; AMI_EVENT_STATUS_THE_EVENT_HAS_BEEN_CANCELED, indicating that the event was canceled. This callback is intended to give the device an opportunity to take action on the event in question. For instance if an event starts, the device should take the appropriate event action on the hardware. This callback returns a boolean, if returned value is true, then a notification will be send over the air automatically to the originator of the event. If it is false, then nothing will be sent back to the originator of the event. Please note that in order for your application to be ZigBee compliant, a notification must be sent over the air to the originator of the event, so a value of FALSE should only be returned if your application code takes care of sending this message or there is some other reason a message does not need to be sent by the framework.

Parameters:

loadControlEvent	Actual event Ver.: always
eventStatus	Status of event Ver.: always
sequenceNumber	Sequence number Ver.: always

void emberAfPluginMessagingClientDisplayMessageCallback (EmberAfPluginMessagingClientMessage * message)

Display Message.

This function is called by the Messaging client plugin whenever the application should display a message.

Parameters:

message	The message that should be displayed. Ver.: always

void emberAfPluginMessagingClientCancelMessageCallback (EmberAfPluginMessagingClientMessage * message)

Cancel Message.

This function is called by the Messaging client plugin whenever the application should stop displaying a message.

Parameters:

message	The message that should no longer be displayed. Ver.: always

void emberAfPluginPriceClientPriceStartedCallback (EmberAfPluginPriceClientPrice * price)

Price Started.

This function is called by the Price client plugin whenever a price starts.

Parameters:

price The price that has started. Ver.: always

void emberAfPluginPriceClientPriceExpiredCallback (EmberAfPluginPriceClientPrice * price)

Price Expired.

This function is called by the Price client plugin whenever a price expires.

Parameters:

price The price that has expired. Ver.: always

void emberAfPluginPslCommissioningTouchLinkCompleteCallback (const EmberPslNetwork * networkInfo, int8u deviceInformationRecordCount, const EmberPslDeviceInfoRecord * deviceInformationRecordList)

Touch Link Complete.

This function is called by the PSL Commissioning plugin when touch linking completes.

Parameters:

networkInfo	The ZigBee and PSL-specific information about the network and target. Ver.: always
deviceInformationRecordCount	The number of sub-device information records for the target. Ver.: always
deviceInformationRecordList	The list of sub-device information records for the target. Ver.: always

void emberAfPluginPslCommissioningTouchLinkFailedCallback (void)

Touch Link Failed.

This function is called by the PSL Commissioning plugin if touch linking fails.

int8u emberAfPluginPslCommissioningGroupIdentifierCountCallback (int8u endpoint)

Group Identifier Count.

This function is called by the PSL Commissioning plugin to determine the number of group identifiers in use by a specific endpoint on the device. The total number of group identifiers on the device, which are shared by all endpoints, is defined by EMBER_PSL_GROUP_ADDRESSES.

Parameters:

endpoint The endpoint for which the group identifier count is requested. Ver.: always

```
boolean emberAfPluginPslCommissioningGroupIdentifierCallback ( int8u                                          endpoint,
                                                               int8u                                          index,
                                                               EmberAfPluginPslCommissioningGroupInformationRecord * record
                                                             )
```

Group Identifier.

This function is called by the PSL Commissioning plugin to obtain information about the group identifiers in use by a specific endpoint on the device. The application should populate the record with information about the group identifier and return TRUE. If no information is available for the given endpoint and index, the application should return FALSE.

Parameters:

endpoint The endpoint for which the group identifier is requested. Ver.: always

index The index of the group on the endpoint. Ver.: always

record The group information record. Ver.: always

```
int8u emberAfPluginPslCommissioningEndpointInformationCountCallback ( int8u endpoint )
```

Endpoint Information Count.

This function is called by the PSL Commissioning plugin to determine the number of remote endpoints controlled by a specific endpoint on the local device.

Parameters:

endpoint The local endpoint for which the remote endpoint information count is requested. Ver.: always

```
boolean emberAfPluginPslCommissioningEndpointInformationCallback ( int8u                                          endpoint,
                                                                   int8u                                          index,
                                                                   EmberAfPluginPslCommissioningEndpointInformationRecord * record
                                                                 )
```

Endpoint Information.

This function is called by the PSL Commissioning plugin to obtain information about the remote endpoints controlled by a specific endpoint on the local device. The application should populate the record

with information about the remote endpoint and return TRUE. If no information is available for the given endpoint and index, the application should return FALSE.

Parameters:

endpoint The local endpoint for which the remote endpoint information is requested. Ver.: always

index The index of the remote endpoint information on the local endpoint. Ver.: always

record The endpoint information record. Ver.: always

void emberAfPluginPslCommissioningIdentifyCallback (int16u duration)

Identify.

This function is called by the PSL Commissioning plugin to notify the application that it should take an action to identify itself. This typically occurs when an Identify Request is received via inter-PAN messaging.

Parameters:

duration If the duration is zero, the device should exit identify mode. If the duration is 0xFFFF, the device should remain in identify mode for the default time. Otherwise, the duration specifies the length of time in tenths of a second that the device should remain in identify mode. Ver.: always

int16u emberAfPluginSimpleMeteringClientRequestMirrorCallback (EmberEUI64 requestingDeviceIeeeAddress)

Request Mirror.

This function is called by the Simple Metering client plugin whenever a Request Mirror command is received. The application should return the endpoint to which the mirror has been assigned. If no mirror could be assigned, the application should return 0xFFFF.

Parameters:

requestingDeviceIeeeAddress Ver.: always

int16u emberAfPluginSimpleMeteringClientRemoveMirrorCallback (EmberEUI64 requestingDeviceIeeeAddress)

Remove Mirror.

This function is called by the Simple Metering client plugin whenever a Remove Mirror command is received. The application should return the endpoint on which the mirror has been removed. If the mirror could not be removed, the application should return 0xFFFF.

Parameters:

requestingDeviceIeeeAddress Ver.: always

boolean emberAfPluginTestExampleCustomerFunctionCallback (int8u arg1,

```
                                                    int8u *   arg2,
                                                    boolean   arg3
                                                  )
```

Example Customer Function.

Test callback for the plugin.

Parameters:

arg1 Example argument 1. Ver.: always

arg2 Example argument 2. Ver.: always

arg3 Example argument 3. Ver.: always

```
void emberAfPluginTunnelingClientTunnelOpenedCallback ( int16u  tunnelId )
```

Tunnel Opened.

This function is called by the Tunneling client plugin whenever a tunnel is opened. Clients may open tunnels by sending a Request Tunnel command.

Parameters:

tunnelId The identifier of the tunnel that has been opened. Ver.: always

```
void emberAfPluginTunnelingClientDataReceivedCallback ( int16u   tunnelId,
                                                        int8u *  data,
                                                        int16u   dataLen
                                                      )
```

Data Received.

This function is called by the Tunneling client plugin whenever data is received from a server through a tunnel.

Parameters:

tunnelId The identifier of the tunnel through which the data was received. Ver.: always

data Buffer containing the raw octets of the data. Ver.: always

dataLen The length in octets of the data. Ver.: always

```
void emberAfPluginTunnelingClientDataErrorCallback ( int16u                                tunnelId,
                                                     EmberAfTunnelingTransferDataStatus    transferDataStatus
                                                   )
```

Data Error.

This function is called by the Tunneling client plugin whenever a data error occurs on a tunnel. Errors occur if a device attempts to send data on tunnel that is no longer active or if the tunneling does not belong to the device.

Parameters:

tunnelId	The identifier of the tunnel on which this data error occurred. Ver.: always
transferDataStatus	The error that occurred. Ver.: always

```
boolean emberAfPluginTunnelingServerIsProtocolSupportedCallback ( int8u   protocolId,
                                                                  int16u  manufacturerCode
                                                                )
```

Is Protocol Supported.

This function is called by the Tunneling server plugin whenever a Request Tunnel command is received. The application should return TRUE if the protocol is supported and FALSE otherwise.

Parameters:

protocolId	The identifier of the metering communication protocol for which the tunnel is requested. Ver.: always
manufacturerCode	The manufacturer code for manufacturer-defined protocols or 0xFFFF in unused. Ver.: always

```
void emberAfPluginTunnelingServerTunnelOpenedCallback ( int16u  tunnelId,
                                                        int8u   protocolId,
                                                        int16u  manufacturerCode
                                                      )
```

Tunnel Opened.

This function is called by the Tunneling server plugin whenever a tunnel is opened. Clients may open tunnels by sending a Request Tunnel command.

Parameters:

tunnelId	The identifier of the tunnel that has been opened. Ver.: always
protocolId	The identifier of the metering communication protocol for the tunnel. Ver.: always
manufacturerCode	The manufacturer code for manufacturer-defined protocols or 0xFFFF in unused. Ver.: always

```
void emberAfPluginTunnelingServerDataReceivedCallback ( int16u  tunnelId,
                                                        int8u *  data,
                                                        int16u  dataLen
                                                      )
```

Data Received.

This function is called by the Tunneling server plugin whenever data is received from a client through a tunnel.

Parameters:

tunnelId	The identifier of the tunnel through which the data was received. Ver.: always
data	Buffer containing the raw octets of the data. Ver.: always
dataLen	The length in octets of the data. Ver.: always

```
void emberAfPluginTunnelingServerDataErrorCallback ( int16u                          tunnelId,
                                                     EmberAfTunnelingTransferDataStatus transferDataStatus
                                                   )
```

Data Error.

This function is called by the Tunneling server plugin whenever a data error occurs on a tunnel. Errors occur if a device attempts to send data on tunnel that is no longer active or if the tunneling does not belong to the device.

Parameters:

tunnelId	The identifier of the tunnel on which this data error occurred. Ver.: always
transferDataStatus	The error that occurred. Ver.: always

```
void emberAfPluginTunnelingServerTunnelClosedCallback ( int16u   tunnelId,
                                                        boolean  clientInitiated
                                                      )
```

Tunnel Closed.

This function is called by the Tunneling server plugin whenever a tunnel is closed. Clients may close tunnels by sending a Close Tunnel command. The server can preemptively close inactive tunnels after a timeout.

Parameters:

tunnelId	The identifier of the tunnel that has been closed. Ver.: always
clientInitiated	TRUE if the client initiated the closing of the tunnel or FALSE if the server closed the tunnel due to inactivity. Ver.: always

Application Framework command buffer loading interface

[Ember Application Framework API reference]

Global Commands

```
#define emberAfFillCommandGlobalServerToClientReadAttributes(clusterId, attributeIds, attributeIdsLen)
#define emberAfFillCommandGlobalClientToServerReadAttributes(clusterId, attributeIds, attributeIdsLen)
#define emberAfFillCommandGlobalServerToClientReadAttributesResponse(clusterId,
        readAttributeStatusRecords, readAttributeStatusRecordsLen)
#define emberAfFillCommandGlobalClientToServerReadAttributesResponse(clusterId,
        readAttributeStatusRecords, readAttributeStatusRecordsLen)
#define emberAfFillCommandGlobalServerToClientWriteAttributes(clusterId,
        writeAttributeRecords, writeAttributeRecordsLen)
#define emberAfFillCommandGlobalClientToServerWriteAttributes(clusterId,
        writeAttributeRecords, writeAttributeRecordsLen)
#define emberAfFillCommandGlobalServerToClientWriteAttributesUndivided(clusterId,
        writeAttributeRecords, writeAttributeRecordsLen)
#define emberAfFillCommandGlobalClientToServerWriteAttributesUndivided(clusterId,
        writeAttributeRecords, writeAttributeRecordsLen)
#define emberAfFillCommandGlobalServerToClientWriteAttributesResponse
        (clusterId, writeAttributeStatusRecords, writeAttributeStatusRecordsLen)
#define emberAfFillCommandGlobalClientToServerWriteAttributesResponse
        (clusterId, writeAttributeStatusRecords, writeAttributeStatusRecordsLen)
#define emberAfFillCommandGlobalServerToClientWriteAttributesNoResponse(clusterId,
        writeAttributeRecords, writeAttributeRecordsLen)
#define emberAfFillCommandGlobalClientToServerWriteAttributesNoResponse(clusterId,
        writeAttributeRecords, writeAttributeRecordsLen)
#define emberAfFillCommandGlobalServerToClientConfigureReporting(clusterId,
        configureReportingRecords, configureReportingRecordsLen)
#define emberAfFillCommandGlobalClientToServerConfigureReporting(clusterId,
        configureReportingRecords, configureReportingRecordsLen)
#define emberAfFillCommandGlobalServerToClientConfigureReportingResponse
        (clusterId, configureReportingStatusRecords, configureReportingStatusRecordsLen)
#define emberAfFillCommandGlobalClientToServerConfigureReportingResponse
        (clusterId, configureReportingStatusRecords, configureReportingStatusRecordsLen)
#define emberAfFillCommandGlobalServerToClientReadReportingConfiguration
        (clusterId, readReportingConfigurationAttributeRecords, readReportingConfigurationAttributeRecordsLen)
#define emberAfFillCommandGlobalClientToServerReadReportingConfiguration
        (clusterId, readReportingConfigurationAttributeRecords, readReportingConfigurationAttributeRecordsLen)
#define emberAfFillCommandGlobalServerToClientReadReportingConfigurationResponse
        (clusterId, readReportingConfigurationRecords, readReportingConfigurationRecordsLen)
```

```
#define emberAfFillCommandGlobalClientToServerReadReportingConfigurationResponse
        (clusterId, readReportingConfigurationRecords, readReportingConfigurationRecordsLen)
#define emberAfFillCommandGlobalServerToClientReportAttributes(clusterId,
        reportAttributeRecords, reportAttributeRecordsLen)
#define emberAfFillCommandGlobalClientToServerReportAttributes(clusterId,
        reportAttributeRecords, reportAttributeRecordsLen)
#define emberAfFillCommandGlobalServerToClientDefaultResponse(clusterId, commandId, status)
#define emberAfFillCommandGlobalClientToServerDefaultResponse(clusterId, commandId, status)
#define emberAfFillCommandGlobalServerToClientDiscoverAttributes(clusterId, startId, maxAttributeIds)
#define emberAfFillCommandGlobalClientToServerDiscoverAttributes(clusterId, startId, maxAttributeIds)
#define emberAfFillCommandGlobalServerToClientDiscoverAttributesResponse(clusterId,
        discoveryComplete, discoverAttributesInfoRecords, discoverAttributesInfoRecordsLen)
#define emberAfFillCommandGlobalClientToServerDiscoverAttributesResponse(clusterId,
        discoveryComplete, discoverAttributesInfoRecords, discoverAttributesInfoRecordsLen)
```

Basic Commands

```
#define emberAfFillCommandBasicClusterResetToFactoryDefaults()
```

Identify Commands

```
#define emberAfFillCommandIdentifyClusterIdentify(identifyTime)
#define emberAfFillCommandIdentifyClusterIdentifyQuery()
#define emberAfFillCommandIdentifyClusterIdentifyQueryResponse(timeout)
```

Groups Commands

```
#define emberAfFillCommandGroupsClusterAddGroup(groupId, groupName)
#define emberAfFillCommandGroupsClusterViewGroup(groupId)
#define emberAfFillCommandGroupsClusterGetGroupMembership(groupCount, groupList, groupListLen)
#define emberAfFillCommandGroupsClusterRemoveGroup(groupId)
#define emberAfFillCommandGroupsClusterRemoveAllGroups()
#define emberAfFillCommandGroupsClusterAddGroupIfIdentifying(groupId, groupName)
#define emberAfFillCommandGroupsClusterAddGroupResponse(status, groupId)
#define emberAfFillCommandGroupsClusterViewGroupResponse(status, groupId, groupName)
#define emberAfFillCommandGroupsClusterGetGroupMembershipResponse(capacity, groupCount,
        groupList, groupListLen)
#define emberAfFillCommandGroupsClusterRemoveGroupResponse(status, groupId)
```

Scenes Commands

```
#define emberAfFillCommandScenesClusterAddScene(groupId, sceneId, transitionTime,
        sceneName, extensionFieldSets, extensionFieldSetsLen)
#define emberAfFillCommandScenesClusterViewScene(groupId, sceneId)
#define emberAfFillCommandScenesClusterRemoveScene(groupId, sceneId)
#define emberAfFillCommandScenesClusterRemoveAllScenes(groupId)
#define emberAfFillCommandScenesClusterStoreScene(groupId, sceneId)
#define emberAfFillCommandScenesClusterRecallScene(groupId, sceneId)
#define emberAfFillCommandScenesClusterGetSceneMembership(groupId)
#define emberAfFillCommandScenesClusterAddSceneResponse(status, groupId, sceneId)
#define emberAfFillCommandScenesClusterViewSceneResponse(status, groupId, sceneId,
        transitionTime, sceneName, extensionFieldSets, extensionFieldSetsLen)
#define emberAfFillCommandScenesClusterRemoveSceneResponse(status, groupId, sceneId)
#define emberAfFillCommandScenesClusterRemoveAllScenesResponse(status, groupId)
#define emberAfFillCommandScenesClusterStoreSceneResponse(status, groupId, sceneId)
#define emberAfFillCommandScenesClusterGetSceneMembershipResponse(status, capacity, groupId,
        sceneCount, sceneList, sceneListLen)
```

On/off Commands

```
#define emberAfFillCommandOnOffClusterOff()
#define emberAfFillCommandOnOffClusterOn()
#define emberAfFillCommandOnOffClusterToggle()
#define emberAfFillCommandOnOffClusterSampleMfgSpecificOffWithTransition()
#define emberAfFillCommandOnOffClusterSampleMfgSpecificOnWithTransition()
#define emberAfFillCommandOnOffClusterSampleMfgSpecificToggleWithTransition()
```

Level Control Commands

```
#define emberAfFillCommandLevelControlClusterMoveToLevel(level, transitionTime)
#define emberAfFillCommandLevelControlClusterMove(moveMode, rate)
#define emberAfFillCommandLevelControlClusterStep(stepMode, stepSize, transitionTime)
#define emberAfFillCommandLevelControlClusterStop()
#define emberAfFillCommandLevelControlClusterMoveToLevelWithOnOff(level, transitionTime)
#define emberAfFillCommandLevelControlClusterMoveWithOnOff(moveMode, rate)
#define emberAfFillCommandLevelControlClusterStepWithOnOff(stepMode, stepSize, transitionTime)
```

#define **emberAfFillCommandLevelControlClusterStopWithOnOff**()

Alarms Commands

#define **emberAfFillCommandAlarmClusterResetAlarm**(alarmCode, clusterId)
#define **emberAfFillCommandAlarmClusterResetAllAlarms**()
#define **emberAfFillCommandAlarmClusterGetAlarm**()
#define **emberAfFillCommandAlarmClusterResetAlarmLog**()
#define **emberAfFillCommandAlarmClusterAlarm**(alarmCode, clusterId)
#define **emberAfFillCommandAlarmClusterGetAlarmResponse**(status, alarmCode, clusterId, timeStamp)

RSSI Location Commands

#define **emberAfFillCommandRssiLocationClusterSetAbsoluteLocation**(coordinate1, coordinate2, coordinate3, power, pathLossExponent)
#define **emberAfFillCommandRssiLocationClusterSetDeviceConfiguration**(power, pathLossExponent, calculationPeriod, numberRssiMeasurements, reportingPeriod)
#define **emberAfFillCommandRssiLocationClusterGetDeviceConfiguration**(targetAddress)
#define **emberAfFillCommandRssiLocationClusterGetLocationData**(flags, numberResponses, targetAddress)
#define **emberAfFillCommandRssiLocationClusterRssiResponse**(replyingDevice, coordinate1, coordinate2, coordinate3, rssi, numberRssiMeasurements)
#define **emberAfFillCommandRssiLocationClusterSendPings**(targetAddress, numberRssiMeasurements, calculationPeriod)
#define **emberAfFillCommandRssiLocationClusterAnchorNodeAnnounce**(anchorNodeIeeeAddress, coordinate1, coordinate2, coordinate3)
#define **emberAfFillCommandRssiLocationClusterDeviceConfigurationResponse**(status, power, pathLossExponent, calculationPeriod, numberRssiMeasurements, reportingPeriod)
#define **emberAfFillCommandRssiLocationClusterLocationDataResponse**(status, locationType, coordinate1, coordinate2, coordinate3, power, pathLossExponent, locationMethod, qualityMeasure, locationAge)
#define **emberAfFillCommandRssiLocationClusterLocationDataNotification**(locationType, coordinate1, coordinate2, coordinate3, power, pathLossExponent, locationMethod, qualityMeasure, locationAge)
#define **emberAfFillCommandRssiLocationClusterCompactLocationDataNotification**(locationType, coordinate1, coordinate2, coordinate3, qualityMeasure, locationAge)
#define **emberAfFillCommandRssiLocationClusterRssiPing**(locationType)
#define **emberAfFillCommandRssiLocationClusterRssiRequest**()
#define **emberAfFillCommandRssiLocationClusterReportRssiMeasurements**(measuringDevice, neighbors, neighborsInfo, neighborsInfoLen)
#define **emberAfFillCommandRssiLocationClusterRequestOwnLocation**(blindNode)

Commissioning Commands

```
#define emberAfFillCommandCommissioningClusterRestartDevice(options, delay, jitter)
#define emberAfFillCommandCommissioningClusterSaveStartupParameters(options, index)
#define emberAfFillCommandCommissioningClusterRestoreStartupParameters(options, index)
#define emberAfFillCommandCommissioningClusterResetStartupParameters(options, index)
#define emberAfFillCommandCommissioningClusterRestartDeviceResponse(status)
#define emberAfFillCommandCommissioningClusterSaveStartupParametersResponse(status)
#define emberAfFillCommandCommissioningClusterRestoreStartupParametersResponse(status)
#define emberAfFillCommandCommissioningClusterResetStartupParametersResponse(status)
```

Thermostat Commands

```
#define emberAfFillCommandThermostatClusterSetpointRaiseLower(mode, amount)
```

Color Control Commands

```
#define emberAfFillCommandColorControlClusterMoveToHue(hue, direction, transitionTime)
#define emberAfFillCommandColorControlClusterMoveHue(moveMode, rate)
#define emberAfFillCommandColorControlClusterStepHue(stepMode, stepSize, transitionTime)
#define emberAfFillCommandColorControlClusterMoveToSaturation(saturation, transitionTime)
#define emberAfFillCommandColorControlClusterMoveSaturation(moveMode, rate)
#define emberAfFillCommandColorControlClusterStepSaturation(stepMode, stepSize, transitionTime)
#define emberAfFillCommandColorControlClusterMoveToHueAndSaturation(hue, saturation, transitionTime)
#define emberAfFillCommandColorControlClusterMoveToColor(colorX, colorY, transitionTime)
#define emberAfFillCommandColorControlClusterMoveColor(rateX, rateY)
#define emberAfFillCommandColorControlClusterStepColor(stepX, stepY, transitionTime)
#define emberAfFillCommandColorControlClusterMoveToColorTemperature(colorTemperature, transitionTime)
```

IAS Zone Commands

```
#define emberAfFillCommandIasZoneClusterZoneEnrollResponse(enrollResponseCode, zoneId)
#define emberAfFillCommandIasZoneClusterZoneStatusChangeNotification(zoneStatus, extendedStatus)
#define emberAfFillCommandIasZoneClusterZoneEnrollRequest(zoneType, manufacturerCode)
```

IAS ACE Commands

```
#define emberAfFillCommandIasAceClusterArm(armMode)
#define emberAfFillCommandIasAceClusterBypass(numberOfZones, zoneIds, zoneIdsLen)
#define emberAfFillCommandIasAceClusterEmergency()
#define emberAfFillCommandIasAceClusterFire()
#define emberAfFillCommandIasAceClusterPanic()
#define emberAfFillCommandIasAceClusterGetZoneIdMap()
#define emberAfFillCommandIasAceClusterGetZoneInformation(zoneId)
#define emberAfFillCommandIasAceClusterArmResponse(armNotification)
#define emberAfFillCommandIasAceClusterGetZoneIdMapResponse(section0, section1, section2,
        section3, section4, section5, section6, section7, section8, section9, section10, section11, section12,
        section13, section14, section15)
#define emberAfFillCommandIasAceClusterGetZoneInformationResponse(zoneId, zoneType, ieeeAddress)
```

IAS WD Commands

```
#define emberAfFillCommandIasWdClusterStartWarning(warningInfo, warningDuration)
#define emberAfFillCommandIasWdClusterSquawk(squawkInfo)
```

Door Lock Commands

```
#define emberAfFillCommandDoorLockClusterLockDoor()
#define emberAfFillCommandDoorLockClusterUnlockDoor()
#define emberAfFillCommandDoorLockClusterLockDoorResponse(status)
#define emberAfFillCommandDoorLockClusterUnlockDoorResponse(status)
```

Window Covering Commands

```
#define emberAfFillCommandWindowCoveringClusterWindowCoveringUpOpen()
#define emberAfFillCommandWindowCoveringClusterWindowCoveringDownClose()
#define emberAfFillCommandWindowCoveringClusterWindowCoveringStop()
#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftSetpoint(indexOfLiftSetpoint)
#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftValue(liftValue)
#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftPercentage(percentageLiftValue)
#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltSetpoint(indexOfTiltSetpoint)
#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltValue(tiltValue)
#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltPercentage(percentageTiltValue)
#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToProgramSetpoint
        (setpointType, setpointIndex, setpointValue)
```

Generic Tunnel Commands

```
#define emberAfFillCommandGenericTunnelClusterMatchProtocolAddress(protocolAddress)
#define emberAfFillCommandGenericTunnelClusterMatchProtocolAddressResponse()
#define emberAfFillCommandGenericTunnelClusterAdvertiseProtocolAddress(protocolAddress)
```

BACnet Protocol Tunnel Commands

```
#define emberAfFillCommandBacnetProtocolTunnelClusterTransferNpdu(NPDU)
```

Over the Air Bootloading Cluster Commands

```
#define emberAfFillCommandOtaBootloadClusterImageNotify(payloadType, queryJitter,
        manufacturerId, imageType, newFileVersion)
#define emberAfFillCommandOtaBootloadClusterQueryNextImageRequest(fieldControl,
        manufacturerId, imageType, currentFileVersion, hardwareVersion)
#define emberAfFillCommandOtaBootloadClusterQueryNextImageResponse(status, manufacturerId,
        imageType, fileVersion, imageSize)
#define emberAfFillCommandOtaBootloadClusterImageBlockRequest(fieldControl, manufacturerId,
        imageType, fileVersion, fileOffset, maxDataSize, requestNodeAddress)
#define emberAfFillCommandOtaBootloadClusterImagePageRequest(fieldControl, manufacturerId,
        imageType, fileVersion, fileOffset, maxDataSize, pageSize, responseSpacing, requestNodeAddress)
#define emberAfFillCommandOtaBootloadClusterImageBlockResponse(status, manufacturerId,
        imageType, fileVersion, fileOffset, dataSize, imageData)
#define emberAfFillCommandOtaBootloadClusterUpgradeEndRequest(status, manufacturerId,
        imageType, fileVersion)
#define emberAfFillCommandOtaBootloadClusterUpgradeEndResponse(manufacturerId, imageType,
        fileVersion, currentTime, upgradeTime)
#define emberAfFillCommandOtaBootloadClusterQuerySpecificFileRequest(requestNodeAddress,
        manufacturerId, imageType, fileVersion, currentZigbeeStackVersion)
#define emberAfFillCommandOtaBootloadClusterQuerySpecificFileResponse(status, manufacturerId,
        imageType, fileVersion, imageSize)
```

Demand Response and Load Control Commands

#define	**emberAfFillCommandDemandResponseLoadControlClusterLoadControlEvent**(issuerEventId, deviceClass, utilityEnrollmentGroup, startTime, durationInMinutes, criticalityLevel, coolingTemperatureOffset, heatingTemperatureOffset, coolingTemperatureSetPoint, heatingTemperatureSetPoint, averageLoadAdjustmentPercentage, dutyCycle, eventControl)
#define	**emberAfFillCommandDemandResponseLoadControlClusterCancelLoadControlEvent**(issuerEventId, deviceClass, utilityEnrollmentGroup, cancelControl, effectiveTime)
#define	**emberAfFillCommandDemandResponseLoadControlClusterCancelAllLoadControlEvents**(cancelControl)
#define	**emberAfFillCommandDemandResponseLoadControlClusterReportEventStatus**(issuerEventId, eventStatus, eventStatusTime, criticalityLevelApplied, coolingTemperatureSetPointApplied, heatingTemperatureSetPointApplied, averageLoadAdjustmentPercentageApplied, dutyCycleApplied, eventControl, signatureType, signature)
#define	**emberAfFillCommandDemandResponseLoadControlClusterGetScheduledEvents**(startTime, numberOfEvents)

Simple Metering Commands

#define	**emberAfFillCommandSimpleMeteringClusterGetProfileResponse**(endTime, status, profileIntervalPeriod, numberOfPeriodsDelivered, intervals, intervalsLen)
#define	**emberAfFillCommandSimpleMeteringClusterRequestMirror**()
#define	**emberAfFillCommandSimpleMeteringClusterRemoveMirror**()
#define	**emberAfFillCommandSimpleMeteringClusterRequestFastPollModeResponse**(appliedUpdatePeriod, fastPollModeEndtime)
#define	**emberAfFillCommandSimpleMeteringClusterGetProfile**(intervalChannel, endTime, numberOfPeriods)
#define	**emberAfFillCommandSimpleMeteringClusterRequestMirrorResponse**(endpointId)
#define	**emberAfFillCommandSimpleMeteringClusterMirrorRemoved**(endpointId)
#define	**emberAfFillCommandSimpleMeteringClusterRequestFastPollMode**(fastPollUpdatePeriod, duration)

Price Commands

#define	**emberAfFillCommandPriceClusterPublishPrice**(providerId, rateLabel, issuerEventId, currentTime, unitOfMeasure, currency, priceTrailingDigitAndPriceTier, numberOfPriceTiersAndRegisterTier, startTime, durationInMinutes, price, priceRatio, generationPrice, generationPriceRatio, alternateCostDelivered, alternateCostUnit, alternateCostTrailingDigit, numberOfBlockThresholds, priceControl)
#define	**emberAfFillCommandPriceClusterPublishBlockPeriod**(providerId, issuerEventId, blockPeriodStartTime, blockPeriodDurationInMinutes, numberOfPriceTiersAndNumberOfBlockThresholds, blockPeriodControl)
#define	**emberAfFillCommandPriceClusterGetCurrentPrice**(commandOptions)
#define	**emberAfFillCommandPriceClusterGetScheduledPrices**(startTime, numberOfEvents)
#define	**emberAfFillCommandPriceClusterPriceAcknowledgement**(providerId, issuerEventId, priceAckTime, control)
#define	**emberAfFillCommandPriceClusterGetBlockPeriods**(startTime, numberOfEvents)

Messaging Commands

```
#define emberAfFillCommandMessagingClusterDisplayMessage(messageId, messageControl,
        startTime, durationInMinutes, message)
#define emberAfFillCommandMessagingClusterCancelMessage(messageId, messageControl)
#define emberAfFillCommandMessagingClusterGetLastMessage()
#define emberAfFillCommandMessagingClusterMessageConfirmation(messageId, confirmationTime)
```

Tunneling Commands

```
#define emberAfFillCommandTunnelingClusterRequestTunnel(protocolId, manufacturerCode, flowControlSupport)
#define emberAfFillCommandTunnelingClusterCloseTunnel(tunnelId)
#define emberAfFillCommandTunnelingClusterTransferDataClientToServer(tunnelId, data, dataLen)
#define emberAfFillCommandTunnelingClusterTransferDataErrorClientToServer(tunnelId, transferDataStatus)
#define emberAfFillCommandTunnelingClusterAckTransferDataClientToServer(tunnelId, numberOfBytesLeft)
#define emberAfFillCommandTunnelingClusterReadyDataClientToServer(tunnelId, numberOfOctetsLeft)
#define emberAfFillCommandTunnelingClusterRequestTunnelResponse(tunnelId, tunnelStatus)
#define emberAfFillCommandTunnelingClusterTransferDataServerToClient(tunnelId, data, dataLen)
#define emberAfFillCommandTunnelingClusterTransferDataErrorServerToClient(tunnelId, transferDataStatus)
#define emberAfFillCommandTunnelingClusterAckTransferDataServerToClient(tunnelId, numberOfBytesLeft)
#define emberAfFillCommandTunnelingClusterReadyDataServerToClient(tunnelId, numberOfOctetsLeft)
```

Prepayment Commands

```
#define emberAfFillCommandPrepaymentClusterSelectAvailableEmergencyCredit
        (commandDateTime, originatingDevice, siteId, meterSerialNumber)
#define emberAfFillCommandPrepaymentClusterChangeSupply(providerId, requestDateTime,
        siteId, meterSerialNumber, implementationDateTime, proposedSupplyStatus, originatorIdSupplyControlBits)
#define emberAfFillCommandPrepaymentClusterSupplyStatusResponse(providerId,
        implementationtDateTime, supplyStatus)
```

Key establishment Commands

```
#define emberAfFillCommandKeyEstablishmentClusterInitiateKeyEstablishmentRequest
        (keyEstablishmentSuite, ephemeralDataGenerateTime, confirmKeyGenerateTime, identity)
#define emberAfFillCommandKeyEstablishmentClusterEphemeralDataRequest(ephemeralData)
```

#define **emberAfFillCommandKeyEstablishmentClusterConfirmKeyDataRequest**(secureMessageAuthenticationCode)
#define **emberAfFillCommandKeyEstablishmentClusterServerToClientTerminateKeyEstablishment** (statusCode, waitTime, keyEstablishmentSuite)
#define **emberAfFillCommandKeyEstablishmentClusterClientToServerTerminateKeyEstablishment** (statusCode, waitTime, keyEstablishmentSuite)
#define **emberAfFillCommandKeyEstablishmentClusterInitiateKeyEstablishmentResponse** (requestedKeyEstablishmentSuite, ephemeralDataGenerateTime, confirmKeyGenerateTime, identity)
#define **emberAfFillCommandKeyEstablishmentClusterEphemeralDataResponse**(ephemeralData)
#define **emberAfFillCommandKeyEstablishmentClusterConfirmKeyDataResponse**(secureMessageAuthenticationCode)

Sample Mfg Specific Cluster Commands

#define **emberAfFillCommandSampleMfgSpecificClusterCommandOne**(argOne)

Detailed Description

This document describes the ZCL command interface used by the Ember Application Framework for filling ZCL command buffers.

Define Documentation

#define emberAfFillCommandGlobalServerToClientReadAttributes (clusterId, attributeIds, attributeIdsLen)

Command description for ReadAttributes.

Command: ReadAttributes

Parameters:

clusterId	EmberAfClusterId
attributeIds	int8u*
attributeIdsLen	int8u

Definition at line **33** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalClientToServerReadAttributes (clusterId, attributeIds, attributeIdsLen)

Command description for ReadAttributes.

Command: ReadAttributes

Parameters:

clusterId	EmberAfClusterId
attributeIds	int8u*
attributeIdsLen	int8u

Definition at line **50** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalServerToClientReadAttributesResponse (clusterId, readAttributeStatusRecords, readAttributeStatusRecordsLen)

Command description for ReadAttributesResponse.

Command: ReadAttributesResponse

Parameters:

clusterId	EmberAfClusterId
readAttributeStatusRecords	int8u*
readAttributeStatusRecordsLen	int8u

Definition at line **67** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalClientToServerReadAttributesResponse (clusterId, readAttributeStatusRecords, readAttributeStatusRecordsLen)

Command description for ReadAttributesResponse.

Command: ReadAttributesResponse

Parameters:

clusterId	EmberAfClusterId
readAttributeStatusRecords	int8u*
readAttributeStatusRecordsLen	int8u

Definition at line **84** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalServerToClientWriteAttributes (clusterId, writeAttributeRecords, writeAttributeRecordsLen)

Command description for WriteAttributes.

Command: WriteAttributes

Parameters:

clusterId	EmberAfClusterId
writeAttributeRecords	int8u*
writeAttributeRecordsLen	int8u

Definition at line **101** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalClientToServerWriteAttributes (clusterId, writeAttributeRecords, writeAttributeRecordsLen)

Command description for WriteAttributes.

Command: WriteAttributes

Parameters:

clusterId	EmberAfClusterId
writeAttributeRecords	int8u*
writeAttributeRecordsLen	int8u

Definition at line **118** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalServerToClientWriteAttributesUndivided (clusterId, writeAttributeRecords, writeAttributeRecordsLen)

Command description for WriteAttributesUndivided.

Command: WriteAttributesUndivided

Parameters:

clusterId	EmberAfClusterId
writeAttributeRecords	int8u*

writeAttributeRecordsLen int8u

Definition at line **135** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalClientToServerWriteAttributesUndivided (clusterId, writeAttributeRecords, writeAttributeRecordsLen)

Command description for WriteAttributesUndivided.

Command: WriteAttributesUndivided

Parameters:

clusterId	EmberAfClusterId
writeAttributeRecords	int8u*
writeAttributeRecordsLen	int8u

Definition at line **152** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalServerToClientWriteAttributesResponse (clusterId, writeAttributeStatusRecords, writeAttributeStatusRecordsLen)

Command description for WriteAttributesResponse.

Command: WriteAttributesResponse

Parameters:

clusterId	EmberAfClusterId
writeAttributeStatusRecords	int8u*
writeAttributeStatusRecordsLen	int8u

Definition at line **169** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalClientToServerWriteAttributesResponse (clusterId, writeAttributeStatusRecords, writeAttributeStatusRecordsLen)

Command description for WriteAttributesResponse.

Command: WriteAttributesResponse

Parameters:

clusterId	EmberAfClusterId
writeAttributeStatusRecords	int8u*
writeAttributeStatusRecordsLen	int8u

Definition at line **186** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalServerToClientWriteAttributesNoResponse (clusterId, writeAttributeRecords, writeAttributeRecordsLen)

Command description for WriteAttributesNoResponse.

Command: WriteAttributesNoResponse

Parameters:

clusterId	EmberAfClusterId
writeAttributeRecords	int8u*
writeAttributeRecordsLen	int8u

Definition at line **203** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalClientToServerWriteAttributesNoResponse (clusterId, writeAttributeRecords, writeAttributeRecordsLen)

Command description for WriteAttributesNoResponse.

Command: WriteAttributesNoResponse

Parameters:

clusterId	EmberAfClusterId
writeAttributeRecords	int8u*
writeAttributeRecordsLen	int8u

Definition at line **220** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalServerToClientConfigureReporting (clusterId, configureReportingRecords, configureReportingRecordsLen)

Command description for ConfigureReporting.

Command: ConfigureReporting

Parameters:

clusterId	EmberAfClusterId
configureReportingRecords	int8u*
configureReportingRecordsLen	int8u

Definition at line **237** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalClientToServerConfigureReporting (clusterId, configureReportingRecords, configureReportingRecordsLen)

Command description for ConfigureReporting.

Command: ConfigureReporting

Parameters:

clusterId	EmberAfClusterId
configureReportingRecords	int8u*
configureReportingRecordsLen	int8u

Definition at line **254** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalServerToClientConfigureReportingResponse (clusterId, configureReportingStatusRecords, configureReportingStatusRecordsLen)

Command description for ConfigureReportingResponse.

Command: ConfigureReportingResponse

Parameters:

clusterId	EmberAfClusterId
configureReportingStatusRecords	int8u*
configureReportingStatusRecordsLen	int8u

Definition at line **271** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalClientToServerConfigureReportingResponse (clusterId, configureReportingStatusRecords, configureReportingStatusRecordsLen)

Command description for ConfigureReportingResponse.

Command: ConfigureReportingResponse

Parameters:

clusterId	EmberAfClusterId
configureReportingStatusRecords	int8u*
configureReportingStatusRecordsLen	int8u

Definition at line **288** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalServerToClientReadReportingConfiguration (clusterId, readReportingConfigurationAttributeRecords, readReportingConfigurationAttributeRecordsLen)

Command description for ReadReportingConfiguration.

Command: ReadReportingConfiguration

Parameters:

clusterId	EmberAfClusterId
readReportingConfigurationAttributeRecords	int8u*
readReportingConfigurationAttributeRecordsLen	int8u

Definition at line **305** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalClientToServerReadReportingConfiguration (clusterId, readReportingConfigurationAttributeRecords, readReportingConfigurationAttributeRecordsLen)

Command description for ReadReportingConfiguration.

Command: ReadReportingConfiguration

Parameters:

clusterId	EmberAfClusterId
readReportingConfigurationAttributeRecords	int8u*

readReportingConfigurationAttributeRecordsLen int8u

Definition at line **322** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalServerToClientReadReportingConfigurationResponse (clusterId, readReportingConfigurationRecords, readReportingConfigurationRecordsLen)

Command description for ReadReportingConfigurationResponse.

Command: ReadReportingConfigurationResponse

Parameters:

clusterId	EmberAfClusterId
readReportingConfigurationRecords	int8u*
readReportingConfigurationRecordsLen	int8u

Definition at line **339** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalClientToServerReadReportingConfigurationResponse (clusterId, readReportingConfigurationRecords, readReportingConfigurationRecordsLen)

Command description for ReadReportingConfigurationResponse.

Command: ReadReportingConfigurationResponse

Parameters:

clusterId	EmberAfClusterId
readReportingConfigurationRecords	int8u*
readReportingConfigurationRecordsLen	int8u

Definition at line **356** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalServerToClientReportAttributes (clusterId, reportAttributeRecords, reportAttributeRecordsLen)

Command description for ReportAttributes.

Command: ReportAttributes

Parameters:

clusterId	EmberAfClusterId
reportAttributeRecords	int8u*
reportAttributeRecordsLen	int8u

Definition at line **373** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalClientToServerReportAttributes (clusterId, reportAttributeRecords, reportAttributeRecordsLen)

Command description for ReportAttributes.

Command: ReportAttributes

Parameters:

clusterId	EmberAfClusterId
reportAttributeRecords	int8u*
reportAttributeRecordsLen	int8u

Definition at line **390** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalServerToClientDefaultResponse (clusterId, commandId, status)

Command description for DefaultResponse.

Command: DefaultResponse

Parameters:

clusterId	EmberAfClusterId
commandId	int8u
status	int8u

Definition at line **407** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalClientToServerDefaultResponse (clusterId, commandId, status)

Command description for DefaultResponse.

Command: DefaultResponse

Parameters:

clusterId	EmberAfClusterId
commandId	int8u
status	int8u

Definition at line **424** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalServerToClientDiscoverAttributes (clusterId, startId, maxAttributeIds)

Command description for DiscoverAttributes.

Command: DiscoverAttributes

Parameters:

clusterId	EmberAfClusterId
startId	int16u
maxAttributeIds	int8u

Definition at line **441** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalClientToServerDiscoverAttributes (clusterId, startId, maxAttributeIds)

Command description for DiscoverAttributes.

Command: DiscoverAttributes

Parameters:

clusterId	EmberAfClusterId
startId	int16u
maxAttributeIds	int8u

Definition at line **458** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalServerToClientDiscoverAttributesResponse (clusterId, discoveryComplete, discoverAttributesInfoRecords, discoverAttributesInfoRecordsLen)

Command description for DiscoverAttributesResponse.

Command: DiscoverAttributesResponse

Parameters:

clusterId	EmberAfClusterId
discoveryComplete	int8u
discoverAttributesInfoRecords	int8u*
discoverAttributesInfoRecordsLen	int8u

Definition at line **476** of file **client-command-macro.h**.

#define emberAfFillCommandGlobalClientToServerDiscoverAttributesResponse (clusterId, discoveryComplete, discoverAttributesInfoRecords, discoverAttributesInfoRecordsLen)

Command description for DiscoverAttributesResponse.

Command: DiscoverAttributesResponse

Parameters:

clusterId	EmberAfClusterId
discoveryComplete	int8u
discoverAttributesInfoRecords	int8u*
discoverAttributesInfoRecordsLen	int8u

Definition at line **495** of file **client-command-macro.h**.

#define emberAfFillCommandBasicClusterResetToFactoryDefaults ( )

Command that resets all attribute values to factory default.

Cluster: Basic, Attributes for determining basic information about a device, setting user device information such as location, and enabling a device. Command: ResetToFactoryDefaults

Definition at line **517** of file **client-command-macro.h**.

#define emberAfFillCommandIdentifyClusterIdentify (identifyTime)

Command description for Identify.

Cluster: Identify, Attributes and commands for putting a device into Identification mode (e.g. flashing a light). Command: Identify

Parameters:

identifyTime int16u

Definition at line **546** of file **client-command-macro.h**.

#define emberAfFillCommandIdentifyClusterIdentifyQuery ( )

Command description for IdentifyQuery.

Cluster: Identify, Attributes and commands for putting a device into Identification mode (e.g. flashing a light). Command: IdentifyQuery

Definition at line **560** of file **client-command-macro.h**.

#define emberAfFillCommandIdentifyClusterIdentifyQueryResponse (timeout)

Command description for IdentifyQueryResponse.

Cluster: Identify, Attributes and commands for putting a device into Identification mode (e.g. flashing a light). Command: IdentifyQueryResponse

Parameters:

timeout int16u

Definition at line **574** of file **client-command-macro.h**.

#define emberAfFillCommandGroupsClusterAddGroup (groupId, groupName)

Command description for AddGroup.

Cluster: Groups, Attributes and commands for group configuration and manipulation. Command: AddGroup

Parameters:

groupId int16u
groupName int8u*

Definition at line **595** of file **client-command-macro.h**.

#define emberAfFillCommandGroupsClusterViewGroup (groupId)

Command description for ViewGroup.

Cluster: Groups, Attributes and commands for group configuration and manipulation. Command: ViewGroup

Parameters:

groupId int16u

Definition at line **611** of file **client-command-macro.h**.

#define emberAfFillCommandGroupsClusterGetGroupMembership (groupCount, groupList, groupListLen)

Command description for GetGroupMembership.

Cluster: Groups, Attributes and commands for group configuration and manipulation.
Command: GetGroupMembership

Parameters:

groupCount int8u
groupList int8u*
groupListLen int8u

Definition at line **628** of file **client-command-macro.h**.

#define emberAfFillCommandGroupsClusterRemoveGroup (groupId)

Command description for RemoveGroup.

Cluster: Groups, Attributes and commands for group configuration and manipulation. Command: RemoveGroup

Parameters:

groupId int16u

Definition at line **645** of file **client-command-macro.h**.

#define emberAfFillCommandGroupsClusterRemoveAllGroups ( )

Command description for RemoveAllGroups.

Cluster: Groups, Attributes and commands for group configuration and manipulation.
Command: RemoveAllGroups

Definition at line **659** of file **client-command-macro.h**.

#define emberAfFillCommandGroupsClusterAddGroupIfIdentifying (groupId, groupName)

Command description for AddGroupIfIdentifying.

Cluster: Groups, Attributes and commands for group configuration and manipulation.
Command: AddGroupIfIdentifying

Parameters:

groupId int16u
groupName int8u*

Definition at line **674** of file **client-command-macro.h**.

#define emberAfFillCommandGroupsClusterAddGroupResponse (status, groupId)

Command description for AddGroupResponse.

Cluster: Groups, Attributes and commands for group configuration and manipulation.
Command: AddGroupResponse

Parameters:

status int8u
groupId int16u

Definition at line **691** of file **client-command-macro.h**.

#define emberAfFillCommandGroupsClusterViewGroupResponse (status, groupId, groupName)

Command description for ViewGroupResponse.

Cluster: Groups, Attributes and commands for group configuration and manipulation.
Command: ViewGroupResponse

Parameters:

status	int8u
groupId	int16u
groupName	int8u*

Definition at line **709** of file **client-command-macro.h**.

#define emberAfFillCommandGroupsClusterGetGroupMembershipResponse (capacity, groupCount, groupList, groupListLen)

Command description for GetGroupMembershipResponse.

Cluster: Groups, Attributes and commands for group configuration and manipulation.
Command: GetGroupMembershipResponse

Parameters:

capacity	int8u
groupCount	int8u
groupList	int8u*
groupListLen	int8u

Definition at line **729** of file **client-command-macro.h**.

#define emberAfFillCommandGroupsClusterRemoveGroupResponse (status, groupId)

Command description for RemoveGroupResponse.

Cluster: Groups, Attributes and commands for group configuration and manipulation.

Command: RemoveGroupResponse

Parameters:

status	int8u
groupId	int16u

Definition at line **748** of file **client-command-macro.h**.

#define emberAfFillCommandScenesClusterAddScene (groupId, sceneId, transitionTime, sceneName, extensionFieldSets, extensionFieldSetsLen)

Command description for AddScene.

Cluster: Scenes, Attributes and commands for scene configuration and manipulation. Command: AddScene

Parameters:

groupId	int16u
sceneId	int8u
transitionTime	int16u
sceneName	int8u*
extensionFieldSets	int8u*
extensionFieldSetsLen	int8u

Definition at line **774** of file **client-command-macro.h**.

#define emberAfFillCommandScenesClusterViewScene (groupId, sceneId)

Command description for ViewScene.

Cluster: Scenes, Attributes and commands for scene configuration and manipulation. Command: ViewScene

Parameters:

groupId	int16u
sceneId	int8u

Definition at line **795** of file **client-command-macro.h**.

#define emberAfFillCommandScenesClusterRemoveScene (groupId, sceneId)

Command description for RemoveScene.

Cluster: Scenes, Attributes and commands for scene configuration and manipulation. Command: RemoveScene

Parameters:

groupId int16u
sceneId int8u

Definition at line **812** of file **client-command-macro.h**.

#define emberAfFillCommandScenesClusterRemoveAllScenes (groupId)

Command description for RemoveAllScenes.

Cluster: Scenes, Attributes and commands for scene configuration and manipulation.
Command: RemoveAllScenes

Parameters:

groupId int16u

Definition at line **828** of file **client-command-macro.h**.

#define emberAfFillCommandScenesClusterStoreScene (groupId, sceneId)

Command description for StoreScene.

Cluster: Scenes, Attributes and commands for scene configuration and manipulation. Command: StoreScene

Parameters:

groupId int16u
sceneId int8u

Definition at line **844** of file **client-command-macro.h**.

#define emberAfFillCommandScenesClusterRecallScene (groupId, sceneId)

Command description for RecallScene.

Cluster: Scenes, Attributes and commands for scene configuration and manipulation. Command: RecallScene

Parameters:

groupId int16u
sceneId int8u

Definition at line **861** of file **client-command-macro.h**.

#define emberAfFillCommandScenesClusterGetSceneMembership (groupId)

Command description for GetSceneMembership.

Cluster: Scenes, Attributes and commands for scene configuration and manipulation.
Command: GetSceneMembership

Parameters:

groupId int16u

Definition at line **877** of file **client-command-macro.h**.

#define emberAfFillCommandScenesClusterAddSceneResponse (status, groupId, sceneId)

Command description for AddSceneResponse.

Cluster: Scenes, Attributes and commands for scene configuration and manipulation.
Command: AddSceneResponse

Parameters:

status int8u
groupId int16u
sceneId int8u

Definition at line **894** of file **client-command-macro.h**.

#define emberAfFillCommandScenesClusterViewSceneResponse (status, groupId, sceneId,

transitionTime,
sceneName,
extensionFieldSets,
extensionFieldSetsLen)

Command description for ViewSceneResponse.

Cluster: Scenes, Attributes and commands for scene configuration and manipulation.
Command: ViewSceneResponse

Parameters:

status	int8u
groupId	int16u
sceneId	int8u
transitionTime	int16u
sceneName	int8u*
extensionFieldSets	int8u*
extensionFieldSetsLen	int8u

Definition at line **917** of file **client-command-macro.h**.

#define emberAfFillCommandScenesClusterRemoveSceneResponse (status,
groupId,
sceneId)

Command description for RemoveSceneResponse.

Cluster: Scenes, Attributes and commands for scene configuration and manipulation.
Command: RemoveSceneResponse

Parameters:

status	int8u
groupId	int16u
sceneId	int8u

Definition at line **940** of file **client-command-macro.h**.

#define emberAfFillCommandScenesClusterRemoveAllScenesResponse (status,
groupId)

Command description for RemoveAllScenesResponse.

Cluster: Scenes, Attributes and commands for scene configuration and manipulation.
Command: RemoveAllScenesResponse

Parameters:

status int8u
groupId int16u

Definition at line **958** of file **client-command-macro.h**.

#define emberAfFillCommandScenesClusterStoreSceneResponse (status, groupId, sceneId)

Command description for StoreSceneResponse.

Cluster: Scenes, Attributes and commands for scene configuration and manipulation.
Command: StoreSceneResponse

Parameters:

status int8u
groupId int16u
sceneId int8u

Definition at line **976** of file **client-command-macro.h**.

#define emberAfFillCommandScenesClusterGetSceneMembershipResponse (status, capacity, groupId, sceneCount, sceneList, sceneListLen)

Command description for GetSceneMembershipResponse.

Cluster: Scenes, Attributes and commands for scene configuration and manipulation.
Command: GetSceneMembershipResponse

Parameters:

status int8u
capacity int8u
groupId int16u
sceneCount int8u

sceneList	int8u*
sceneListLen	int8u

Definition at line **998** of file **client-command-macro.h**.

#define emberAfFillCommandOnOffClusterOff ( )

Command description for Off.

Cluster: On/off, Attributes and commands for switching devices between 'On' and 'Off' states. Command: Off

Definition at line **1022** of file **client-command-macro.h**.

#define emberAfFillCommandOnOffClusterOn ( )

Command description for On.

Cluster: On/off, Attributes and commands for switching devices between 'On' and 'Off' states. Command: On

Definition at line **1035** of file **client-command-macro.h**.

#define emberAfFillCommandOnOffClusterToggle ( )

Command description for Toggle.

Cluster: On/off, Attributes and commands for switching devices between 'On' and 'Off' states. Command: Toggle

Definition at line **1048** of file **client-command-macro.h**.

#define emberAfFillCommandOnOffClusterSampleMfgSpecificOffWithTransition ( )

Client command that turns the device off with a transition given by the transition time in the Ember Sample transition time attribute.

Cluster: On/off, Attributes and commands for switching devices between 'On' and 'Off' states.
Command: SampleMfgSpecificOffWithTransition

Definition at line **1062** of file **client-command-macro.h**.

#define emberAfFillCommandOnOffClusterSampleMfgSpecificOnWithTransition ( )

Client command that turns the device on with a transition given by the transition time in the Ember Sample transition time attribute.

Cluster: On/off, Attributes and commands for switching devices between 'On' and 'Off' states.
Command: SampleMfgSpecificOnWithTransition

Definition at line **1077** of file **client-command-macro.h**.

#define emberAfFillCommandOnOffClusterSampleMfgSpecificToggleWithTransition ( )

Client command that toggles the device with a transition given by the transition time in the Ember Sample transition time attribute.

Cluster: On/off, Attributes and commands for switching devices between 'On' and 'Off' states.
Command: SampleMfgSpecificToggleWithTransition

Definition at line **1092** of file **client-command-macro.h**.

#define emberAfFillCommandLevelControlClusterMoveToLevel (level, transitionTime)

Command description for MoveToLevel.

Cluster: Level Control, Attributes and commands for controlling devices that can be set to a level between fully 'On' and fully 'Off.' Command: MoveToLevel

Parameters:

level	int8u
transitionTime	int16u

Definition at line **1118** of file **client-command-macro.h**.

#define emberAfFillCommandLevelControlClusterMove (moveMode, rate)

Command description for Move.

Cluster: Level Control, Attributes and commands for controlling devices that can be set to a level between fully 'On' and fully 'Off.' Command: Move

Parameters:

moveMode int8u
rate int8u

Definition at line **1135** of file **client-command-macro.h**.

#define emberAfFillCommandLevelControlClusterStep (stepMode, stepSize, transitionTime)

Command description for Step.

Cluster: Level Control, Attributes and commands for controlling devices that can be set to a level between fully 'On' and fully 'Off.' Command: Step

Parameters:

stepMode int8u
stepSize int8u
transitionTime int16u

Definition at line **1153** of file **client-command-macro.h**.

#define emberAfFillCommandLevelControlClusterStop ( )

Command description for Stop.

Cluster: Level Control, Attributes and commands for controlling devices that can be set to a level between fully 'On' and fully 'Off.' Command: Stop

Definition at line **1169** of file **client-command-macro.h**.

#define emberAfFillCommandLevelControlClusterMoveToLevelWithOnOff (level, transitionTime)

Command description for MoveToLevelWithOnOff.

Cluster: Level Control, Attributes and commands for controlling devices that can be set to a level between fully 'On' and fully 'Off.' Command: MoveToLevelWithOnOff

Parameters:

level	int8u
transitionTime	int16u

Definition at line **1184** of file **client-command-macro.h**.

#define emberAfFillCommandLevelControlClusterMoveWithOnOff (moveMode, rate)

Command description for MoveWithOnOff.

Cluster: Level Control, Attributes and commands for controlling devices that can be set to a level between fully 'On' and fully 'Off.' Command: MoveWithOnOff

Parameters:

moveMode	int8u
rate	int8u

Definition at line **1201** of file **client-command-macro.h**.

#define emberAfFillCommandLevelControlClusterStepWithOnOff (stepMode, stepSize, transitionTime)

Command description for StepWithOnOff.

Cluster: Level Control, Attributes and commands for controlling devices that can be set to a level between fully 'On' and fully 'Off.' Command: StepWithOnOff

Parameters:

stepMode	int8u
stepSize	int8u
transitionTime	int16u

Definition at line **1219** of file **client-command-macro.h**.

#define emberAfFillCommandLevelControlClusterStopWithOnOff ( )

Command description for StopWithOnOff.

Cluster: Level Control, Attributes and commands for controlling devices that can be set to a level between fully 'On' and fully 'Off.' Command: StopWithOnOff

Definition at line **1235** of file **client-command-macro.h**.

#define emberAfFillCommandAlarmClusterResetAlarm (alarmCode, clusterId)

Command description for ResetAlarm.

Cluster: Alarms, Attributes and commands for sending notifications and configuring alarm functionality. Command: ResetAlarm

Parameters:

alarmCode int8u
clusterId int16u

Definition at line **1255** of file **client-command-macro.h**.

#define emberAfFillCommandAlarmClusterResetAllAlarms ( )

Command description for ResetAllAlarms.

Cluster: Alarms, Attributes and commands for sending notifications and configuring alarm functionality. Command: ResetAllAlarms

Definition at line **1270** of file **client-command-macro.h**.

#define emberAfFillCommandAlarmClusterGetAlarm ( )

Command description for GetAlarm.

Cluster: Alarms, Attributes and commands for sending notifications and configuring alarm functionality. Command: GetAlarm

Definition at line **1283** of file **client-command-macro.h**.

#define emberAfFillCommandAlarmClusterResetAlarmLog ( )

Command description for ResetAlarmLog.

Cluster: Alarms, Attributes and commands for sending notifications and configuring alarm functionality. Command: ResetAlarmLog

Definition at line **1296** of file **client-command-macro.h**.

#define emberAfFillCommandAlarmClusterAlarm (alarmCode, clusterId)

Command description for Alarm.

Cluster: Alarms, Attributes and commands for sending notifications and configuring alarm functionality. Command: Alarm

Parameters:

alarmCode int8u
clusterId int16u

Definition at line **1311** of file **client-command-macro.h**.

#define emberAfFillCommandAlarmClusterGetAlarmResponse (status, alarmCode, clusterId, timeStamp)

Command description for GetAlarmResponse.

Cluster: Alarms, Attributes and commands for sending notifications and configuring alarm functionality. Command: GetAlarmResponse

Parameters:

status int8u
alarmCode int8u
clusterId int16u
timeStamp int32u

Definition at line **1330** of file **client-command-macro.h**.

#define emberAfFillCommandRssiLocationClusterSetAbsoluteLocation (coordinate1, coordinate2, coordinate3, power, pathLossExponent)

Command description for SetAbsoluteLocation.

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: SetAbsoluteLocation

Parameters:

coordinate1	int16s
coordinate2	int16s
coordinate3	int16s
power	int16s
pathLossExponent	int16u

Definition at line **1362** of file **client-command-macro.h**.

#define emberAfFillCommandRssiLocationClusterSetDeviceConfiguration (power, pathLossExponent, calculationPeriod, numberRssiMeasurements, reportingPeriod)

Command description for SetDeviceConfiguration.

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: SetDeviceConfiguration

Parameters:

power	int16s
pathLossExponent	int16u
calculationPeriod	int16u
numberRssiMeasurements	int8u
reportingPeriod	int16u

Definition at line **1385** of file **client-command-macro.h**.

#define emberAfFillCommandRssiLocationClusterGetDeviceConfiguration (targetAddress)

Command description for GetDeviceConfiguration.

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: GetDeviceConfiguration

Parameters:

targetAddress int8u*

Definition at line **1404** of file **client-command-macro.h**.

#define emberAfFillCommandRssiLocationClusterGetLocationData (flags, numberResponses, targetAddress)

Command description for GetLocationData.

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: GetLocationData

Parameters:

flags	int8u
numberResponses	int8u
targetAddress	int8u*

Definition at line **1421** of file **client-command-macro.h**.

#define emberAfFillCommandRssiLocationClusterRssiResponse (replyingDevice, coordinate1, coordinate2, coordinate3, rssi, numberRssiMeasurements)

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: RssiResponse.

Parameters:

replyingDevice	int8u*
coordinate1	int16s
coordinate2	int16s
coordinate3	int16s
rssi	int8s

numberRssiMeasurements int8u

Definition at line **1443** of file **client-command-macro.h**.

#define emberAfFillCommandRssiLocationClusterSendPings (targetAddress, numberRssiMeasurements, calculationPeriod)

Command description for SendPings.

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: SendPings

Parameters:

targetAddress int8u*
numberRssiMeasurements int8u
calculationPeriod int16u

Definition at line **1465** of file **client-command-macro.h**.

#define emberAfFillCommandRssiLocationClusterAnchorNodeAnnounce (anchorNodeIeeeAddress, coordinate1, coordinate2, coordinate3)

Command description for AnchorNodeAnnounce.

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: AnchorNodeAnnounce

Parameters:

anchorNodeIeeeAddress int8u*
coordinate1 int16s
coordinate2 int16s
coordinate3 int16s

Definition at line **1485** of file **client-command-macro.h**.

#define emberAfFillCommandRssiLocationClusterDeviceConfigurationResponse (status, power, pathLossExponent,

```
                                                                    calculationPeriod,
                                                                    numberRssiMeasurements,
                                                                    reportingPeriod              )
```

Command description for DeviceConfigurationResponse.

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: DeviceConfigurationResponse

Parameters:

status	int8u
power	int16s
pathLossExponent	int16u
calculationPeriod	int16u
numberRssiMeasurements	int8u
reportingPeriod	int16u

Definition at line **1508** of file **client-command-macro.h**.

```
#define emberAfFillCommandRssiLocationClusterLocationDataResponse ( status,
                                                                    locationType,
                                                                    coordinate1,
                                                                    coordinate2,
                                                                    coordinate3,
                                                                    power,
                                                                    pathLossExponent,
                                                                    locationMethod,
                                                                    qualityMeasure,
                                                                    locationAge          )
```

Command description for LocationDataResponse.

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: LocationDataResponse

Parameters:

status	int8u
locationType	int8u
coordinate1	int16s
coordinate2	int16s
coordinate3	int16s
power	int16s

pathLossExponent	int16u
locationMethod	int8u
qualityMeasure	int8u
locationAge	int16u

Definition at line **1537** of file **client-command-macro.h**.

```
#define emberAfFillCommandRssiLocationClusterLocationDataNotification ( locationType,
                                                                        coordinate1,
                                                                        coordinate2,
                                                                        coordinate3,
                                                                        power,
                                                                        pathLossExponent,
                                                                        locationMethod,
                                                                        qualityMeasure,
                                                                        locationAge          )
```

Command description for LocationDataNotification.

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: LocationDataNotification

Parameters:

locationType	int8u
coordinate1	int16s
coordinate2	int16s
coordinate3	int16s
power	int16s
pathLossExponent	int16u
locationMethod	int8u
qualityMeasure	int8u
locationAge	int16u

Definition at line **1569** of file **client-command-macro.h**.

```
#define emberAfFillCommandRssiLocationClusterCompactLocationDataNotification ( locationType,
                                                                               coordinate1,
                                                                               coordinate2,
                                                                               coordinate3,
                                                                               qualityMeasure,
                                                                               locationAge       )
```

Command description for CompactLocationDataNotification.

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: CompactLocationDataNotification

Parameters:

locationType	int8u
coordinate1	int16s
coordinate2	int16s
coordinate3	int16s
qualityMeasure	int8u
locationAge	int16u

Definition at line **1597** of file **client-command-macro.h**.

#define emberAfFillCommandRssiLocationClusterRssiPing (locationType)

Command description for RssiPing.

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: RssiPing

Parameters:

locationType int8u

Definition at line **1617** of file **client-command-macro.h**.

#define emberAfFillCommandRssiLocationClusterRssiRequest ( )

Command description for RssiRequest.

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: RssiRequest

Definition at line **1631** of file **client-command-macro.h**.

#define emberAfFillCommandRssiLocationClusterReportRssiMeasurements (measuringDevice, neighbors, neighborsInfo, neighborsInfoLen)

Command description for ReportRssiMeasurements.

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: ReportRssiMeasurements

Parameters:

measuringDevice	int8u*
neighbors	int8u
neighborsInfo	int8u*
neighborsInfoLen	int8u

Definition at line **1648** of file **client-command-macro.h**.

#define emberAfFillCommandRssiLocationClusterRequestOwnLocation (blindNode)

Command description for RequestOwnLocation.

Cluster: RSSI Location, Attributes and commands that provide a means for exchanging location information and channel parameters among devices. Command: RequestOwnLocation

Parameters:

blindNode	int8u*

Definition at line **1666** of file **client-command-macro.h**.

#define emberAfFillCommandCommissioningClusterRestartDevice (options, delay, jitter)

Command description for RestartDevice.

Cluster: Commissioning, Attributes and commands for commissioning and managing a ZigBee device. Command: RestartDevice

Parameters:

options	int8u
delay	int8u
jitter	int8u

Definition at line **1693** of file **client-command-macro.h**.

#define emberAfFillCommandCommissioningClusterSaveStartupParameters (options, index)

Command description for SaveStartupParameters.

Cluster: Commissioning, Attributes and commands for commissioning and managing a ZigBee device.
Command: SaveStartupParameters

Parameters:

options int8u
index int8u

Definition at line **1711** of file **client-command-macro.h**.

#define emberAfFillCommandCommissioningClusterRestoreStartupParameters (options, index)

Command description for RestoreStartupParameters.

Cluster: Commissioning, Attributes and commands for commissioning and managing a ZigBee device.
Command: RestoreStartupParameters

Parameters:

options int8u
index int8u

Definition at line **1728** of file **client-command-macro.h**.

#define emberAfFillCommandCommissioningClusterResetStartupParameters (options, index)

Command description for ResetStartupParameters.

Cluster: Commissioning, Attributes and commands for commissioning and managing a ZigBee device.
Command: ResetStartupParameters

Parameters:

options int8u
index int8u

Definition at line **1745** of file **client-command-macro.h**.

#define emberAfFillCommandCommissioningClusterRestartDeviceResponse (status)

Command description for RestartDeviceResponse.

Cluster: Commissioning, Attributes and commands for commissioning and managing a ZigBee device.
Command: RestartDeviceResponse

Parameters:

status int8u

Definition at line **1761** of file **client-command-macro.h**.

#define emberAfFillCommandCommissioningClusterSaveStartupParametersResponse (status)

Command description for SaveStartupParametersResponse.

Cluster: Commissioning, Attributes and commands for commissioning and managing a ZigBee device.
Command: SaveStartupParametersResponse

Parameters:

status int8u

Definition at line **1776** of file **client-command-macro.h**.

#define emberAfFillCommandCommissioningClusterRestoreStartupParametersResponse (status)

Command description for RestoreStartupParametersResponse.

Cluster: Commissioning, Attributes and commands for commissioning and managing a ZigBee device.
Command: RestoreStartupParametersResponse

Parameters:

status int8u

Definition at line **1791** of file **client-command-macro.h**.

#define emberAfFillCommandCommissioningClusterResetStartupParametersResponse (status)

Command description for ResetStartupParametersResponse.

Cluster: Commissioning, Attributes and commands for commissioning and managing a ZigBee device.

Command: ResetStartupParametersResponse

Parameters:

status int8u

Definition at line **1806** of file **client-command-macro.h**.

#define emberAfFillCommandThermostatClusterSetpointRaiseLower (mode, amount)

Command description for SetpointRaiseLower.

Cluster: Thermostat, An interface for configuring and controlling the functionality of a thermostat.
Command: SetpointRaiseLower

Parameters:

mode int8u
amount int8s

Definition at line **1837** of file **client-command-macro.h**.

#define emberAfFillCommandColorControlClusterMoveToHue (hue, direction, transitionTime)

Command description for MoveToHue.

Cluster: Color Control, Attributes and commands for controlling the color properties of a color-capable light. Command: MoveToHue

Parameters:

hue int8u
direction int8u
transitionTime int16u

Definition at line **1875** of file **client-command-macro.h**.

#define emberAfFillCommandColorControlClusterMoveHue (moveMode, rate)

Command description for MoveHue.

Cluster: Color Control, Attributes and commands for controlling the color properties of a color-capable light. Command: MoveHue

Parameters:

moveMode int8u
rate int8u

Definition at line **1893** of file **client-command-macro.h**.

#define emberAfFillCommandColorControlClusterStepHue (stepMode, stepSize, transitionTime)

Command description for StepHue.

Cluster: Color Control, Attributes and commands for controlling the color properties of a color-capable light. Command: StepHue

Parameters:

stepMode int8u
stepSize int8u
transitionTime int8u

Definition at line **1911** of file **client-command-macro.h**.

#define emberAfFillCommandColorControlClusterMoveToSaturation (saturation, transitionTime)

Command description for MoveToSaturation.

Cluster: Color Control, Attributes and commands for controlling the color properties of a color-capable light. Command: MoveToSaturation

Parameters:

saturation int8u
transitionTime int16u

Definition at line **1929** of file **client-command-macro.h**.

#define emberAfFillCommandColorControlClusterMoveSaturation (moveMode, rate)

Command description for MoveSaturation.

Cluster: Color Control, Attributes and commands for controlling the color properties of a color-capable light. Command: MoveSaturation

Parameters:

moveMode	int8u
rate	int8u

Definition at line **1946** of file **client-command-macro.h**.

#define emberAfFillCommandColorControlClusterStepSaturation (stepMode, stepSize, transitionTime)

Command description for StepSaturation.

Cluster: Color Control, Attributes and commands for controlling the color properties of a color-capable light. Command: StepSaturation

Parameters:

stepMode	int8u
stepSize	int8u
transitionTime	int8u

Definition at line **1964** of file **client-command-macro.h**.

#define emberAfFillCommandColorControlClusterMoveToHueAndSaturation (hue, saturation, transitionTime)

Command description for MoveToHueAndSaturation.

Cluster: Color Control, Attributes and commands for controlling the color properties of a color-capable light. Command: MoveToHueAndSaturation

Parameters:

hue	int8u
saturation	int8u
transitionTime	int16u

Definition at line **1983** of file **client-command-macro.h**.

#define emberAfFillCommandColorControlClusterMoveToColor (colorX, colorY, transitionTime)

Moves the lighting to a specific color.

Cluster: Color Control, Attributes and commands for controlling the color properties of a color-capable light. Command: MoveToColor

Parameters:

colorX	int16u
colorY	int16u
transitionTime	int16u

Definition at line **2002** of file **client-command-macro.h**.

#define emberAfFillCommandColorControlClusterMoveColor (rateX, rateY)

Moves the color.

Cluster: Color Control, Attributes and commands for controlling the color properties of a color-capable light. Command: MoveColor

Parameters:

rateX	int16s
rateY	int16s

Definition at line **2020** of file **client-command-macro.h**.

#define emberAfFillCommandColorControlClusterStepColor (stepX, stepY, transitionTime)

Steps the lighting to a specific color.

Cluster: Color Control, Attributes and commands for controlling the color properties of a color-capable light. Command: StepColor

Parameters:

stepX int16s
stepY int16s
transitionTime int16u

Definition at line **2038** of file **client-command-macro.h**.

#define emberAfFillCommandColorControlClusterMoveToColorTemperature (colorTemperature, transitionTime)

Moves the lighting to a specific color temperature.

Cluster: Color Control, Attributes and commands for controlling the color properties of a color-capable light. Command: MoveToColorTemperature

Parameters:
colorTemperature int16u
transitionTime int16u

Definition at line **2056** of file **client-command-macro.h**.

#define emberAfFillCommandIasZoneClusterZoneEnrollResponse (enrollResponseCode, zoneId)

Command description for zoneEnrollResponse.

Cluster: IAS Zone, Attributes and commands for IAS security zone devices. Command: ZoneEnrollResponse

Parameters:
enrollResponseCode int8u
zoneId int8u

Definition at line **2118** of file **client-command-macro.h**.

#define emberAfFillCommandIasZoneClusterZoneStatusChangeNotification (zoneStatus, extendedStatus)

Command description for zoneStatusChangeNotification.

Cluster: IAS Zone, Attributes and commands for IAS security zone devices.
Command: ZoneStatusChangeNotification

Parameters:

zoneStatus int16u
extendedStatus int8u

Definition at line **2135** of file **client-command-macro.h**.

#define emberAfFillCommandIasZoneClusterZoneEnrollRequest (zoneType, manufacturerCode)

Command description for zoneEnrollRequest.

Cluster: IAS Zone, Attributes and commands for IAS security zone devices. Command: ZoneEnrollRequest

Parameters:

zoneType int16u
manufacturerCode int16u

Definition at line **2152** of file **client-command-macro.h**.

#define emberAfFillCommandIasAceClusterArm (armMode)

Command description for Arm.

Cluster: IAS ACE, Attributes and commands for IAS Ancillary Control Equipment. Command: Arm

Parameters:

armMode int8u

Definition at line **2173** of file **client-command-macro.h**.

#define emberAfFillCommandIasAceClusterBypass (numberOfZones, zoneIds, zoneIdsLen)

Command description for Bypass.

Cluster: IAS ACE, Attributes and commands for IAS Ancillary Control Equipment. Command: Bypass

Parameters:

numberOfZones int8u
zoneIds int8u*

zoneIdsLen int8u

Definition at line **2190** of file **client-command-macro.h**.

#define emberAfFillCommandIasAceClusterEmergency ( )

Command description for Emergency.

Cluster: IAS ACE, Attributes and commands for IAS Ancillary Control Equipment. Command: Emergency

Definition at line **2206** of file **client-command-macro.h**.

#define emberAfFillCommandIasAceClusterFire ( )

Command description for Fire.

Cluster: IAS ACE, Attributes and commands for IAS Ancillary Control Equipment. Command: Fire

Definition at line **2219** of file **client-command-macro.h**.

#define emberAfFillCommandIasAceClusterPanic ( )

Command description for Panic.

Cluster: IAS ACE, Attributes and commands for IAS Ancillary Control Equipment. Command: Panic

Definition at line **2232** of file **client-command-macro.h**.

#define emberAfFillCommandIasAceClusterGetZoneIdMap ( )

Command description for GetZoneIdMap.

Cluster: IAS ACE, Attributes and commands for IAS Ancillary Control Equipment. Command: GetZoneIdMap

Definition at line **2245** of file **client-command-macro.h**.

#define emberAfFillCommandIasAceClusterGetZoneInformation (zoneId)

Command description for GetZoneInformation.

Cluster: IAS ACE, Attributes and commands for IAS Ancillary Control Equipment. Command: GetZoneInformation

Parameters:

zoneId int8u

Definition at line **2259** of file **client-command-macro.h**.

#define emberAfFillCommandIasAceClusterArmResponse (armNotification)

Command description for ArmResponse.

Cluster: IAS ACE, Attributes and commands for IAS Ancillary Control Equipment. Command: ArmResponse

Parameters:

armNotification int8u

Definition at line **2274** of file **client-command-macro.h**.

```
#define emberAfFillCommandIasAceClusterGetZoneIdMapResponse ( section0,
                                                              section1,
                                                              section2,
                                                              section3,
                                                              section4,
                                                              section5,
                                                              section6,
                                                              section7,
                                                              section8,
                                                              section9,
                                                              section10,
                                                              section11,
                                                              section12,
                                                              section13,
                                                              section14,
                                                              section15 )
```

Command description for GetZoneIdMapResponse.

Cluster: IAS ACE, Attributes and commands for IAS Ancillary Control Equipment.
Command: GetZoneIdMapResponse

Parameters:

section0	int16u
section1	int16u
section2	int16u
section3	int16u
section4	int16u
section5	int16u
section6	int16u
section7	int16u
section8	int16u
section9	int16u
section10	int16u
section11	int16u
section12	int16u
section13	int16u
section14	int16u
section15	int16u

Definition at line **2304** of file **client-command-macro.h**.

#define emberAfFillCommandIasAceClusterGetZoneInformationResponse (zoneId, zoneType, ieeeAddress)

Command description for GetZoneInformationResponse.

Cluster: IAS ACE, Attributes and commands for IAS Ancillary Control Equipment.
Command: GetZoneInformationResponse

Parameters:

zoneId	int8u
zoneType	int16u
ieeeAddress	int8u*

Definition at line **2336** of file **client-command-macro.h**.

#define emberAfFillCommandIasWdClusterStartWarning (warningInfo, warningDuration)

Command description for StartWarning.

Cluster: IAS WD, Attributes and commands for IAS Warning Devices. Command: StartWarning

Parameters:

warningInfo int8u
warningDuration int16u

Definition at line **2359** of file **client-command-macro.h**.

#define emberAfFillCommandIasWdClusterSquawk (squawkInfo)

Command description for Squawk.

Cluster: IAS WD, Attributes and commands for IAS Warning Devices. Command: Squawk

Parameters:

squawkInfo int8u

Definition at line **2375** of file **client-command-macro.h**.

#define emberAfFillCommandDoorLockClusterLockDoor ( )

Locks the door.

Cluster: Door Lock, Provides an interface into a generic way to secure a door. Command: LockDoor

Definition at line **2394** of file **client-command-macro.h**.

#define emberAfFillCommandDoorLockClusterUnlockDoor ( )

Unlocks the door.

Cluster: Door Lock, Provides an interface into a generic way to secure a door. Command: UnlockDoor

Definition at line **2407** of file **client-command-macro.h**.

#define emberAfFillCommandDoorLockClusterLockDoorResponse (status)

Indicates lock success or failure.

Cluster: Door Lock, Provides an interface into a generic way to secure a door. Command: LockDoorResponse

Parameters:

status int8u

Definition at line **2421** of file **client-command-macro.h**.

#define emberAfFillCommandDoorLockClusterUnlockDoorResponse (status)

Indicates unlock success or failure.

Cluster: Door Lock, Provides an interface into a generic way to secure a door. Command: UnlockDoorResponse

Parameters:

status int8u

Definition at line **2436** of file **client-command-macro.h**.

#define emberAfFillCommandWindowCoveringClusterWindowCoveringUpOpen ( )

Moves window covering to InstalledOpenLimit - Lift and InstalledOpenLimit - Tilt.

Cluster: Window Covering, Provides an interface for controlling and adjusting automatic window coverings. Command: WindowCoveringUpOpen

Definition at line **2455** of file **client-command-macro.h**.

#define emberAfFillCommandWindowCoveringClusterWindowCoveringDownClose ( )

Moves window covering to InstalledClosedLimit - Lift and InstalledCloseLimit - Tilt.

Cluster: Window Covering, Provides an interface for controlling and adjusting automatic window coverings. Command: WindowCoveringDownClose

Definition at line **2468** of file **client-command-macro.h**.

#define emberAfFillCommandWindowCoveringClusterWindowCoveringStop ( )

Stop any adjusting of window covering.

Cluster: Window Covering, Provides an interface for controlling and adjusting automatic window coverings. Command: WindowCoveringStop

Definition at line **2481** of file **client-command-macro.h**.

#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftSetpoint (indexOfLiftSetpoint)

Goto lift to value specified.

Cluster: Window Covering, Provides an interface for controlling and adjusting automatic window coverings. Command: WindowCoveringGoToLiftSetpoint

Parameters:

indexOfLiftSetpoint int8u

Definition at line **2495** of file **client-command-macro.h**.

#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftValue (liftValue)

Goto lift value specified.

Cluster: Window Covering, Provides an interface for controlling and adjusting automatic window coverings. Command: WindowCoveringGoToLiftValue

Parameters:

liftValue int16u

Definition at line **2510** of file **client-command-macro.h**.

#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftPercentage (percentageLiftValue)

Goto lift percentage specifiec.

Cluster: Window Covering, Provides an interface for controlling and adjusting automatic window coverings. Command: WindowCoveringGoToLiftPercentage

Parameters:

percentageLiftValue int8u

Definition at line **2525** of file **client-command-macro.h**.

#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltSetpoint (indexOfTiltSetpoint)

Goto tilt setpoint.

Cluster: Window Covering, Provides an interface for controlling and adjusting automatic window coverings. Command: WindowCoveringGoToTiltSetpoint

Parameters:

indexOfTiltSetpoint int8u

Definition at line **2540** of file **client-command-macro.h**.

#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltValue (tiltValue)

Goto tilt value specified.

Cluster: Window Covering, Provides an interface for controlling and adjusting automatic window coverings. Command: WindowCoveringGoToTiltValue

Parameters:

tiltValue int16u

Definition at line **2555** of file **client-command-macro.h**.

#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltPercentage (percentageTiltValue)

Goto tilt percentage specified.

Cluster: Window Covering, Provides an interface for controlling and adjusting automatic window coverings. Command: WindowCoveringGoToTiltPercentage

Parameters:

percentageTiltValue int8u

Definition at line **2570** of file **client-command-macro.h**.

#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToProgramSetpoint (setpointType,

setpointIndex,
setpointValue)

Cluster: Window Covering, Provides an interface for controlling and adjusting automatic window coverings. Command: WindowCoveringGoToProgramSetpoint.

Parameters:

setpointType int8u
setpointIndex int8u
setpointValue int16u

Definition at line **2587** of file **client-command-macro.h**.

#define emberAfFillCommandGenericTunnelClusterMatchProtocolAddress (protocolAddress)

This command is generated when an application wishes to find the ZigBee address (node, endpoint) of the Generic Tunnel server cluster with a given ProtocolAddress attribute. The command is typically multicast to a group of inter-communicating Generic Tunnel clusters.

Cluster: Generic Tunnel, The minimum common commands and attributes required to tunnel any protocol. Command: MatchProtocolAddress

Parameters:

protocolAddress int8u*

Definition at line **2609** of file **client-command-macro.h**.

#define emberAfFillCommandGenericTunnelClusterMatchProtocolAddressResponse ( )

This command is generated upon receipt of a Match Protocol Address command to indicate that the Protocol Address was successfully matched.

Cluster: Generic Tunnel, The minimum common commands and attributes required to tunnel any protocol. Command: MatchProtocolAddressResponse

Definition at line **2623** of file **client-command-macro.h**.

#define emberAfFillCommandGenericTunnelClusterAdvertiseProtocolAddress (protocolAddress)

This command is typically sent upon startup, and whenever the ProtocolAddress attribute changes. It is typically multicast to a group of inter-communicating Generic Tunnel clusters.

Cluster: Generic Tunnel, The minimum common commands and attributes required to tunnel any protocol. Command: AdvertiseProtocolAddress

Parameters:

protocolAddress int8u*

Definition at line **2637** of file **client-command-macro.h**.

#define emberAfFillCommandBacnetProtocolTunnelClusterTransferNpdu (NPDU)

This command is generated when a BACnet network layer wishes to transfer a BACnet NPDU across a ZigBee tunnel to another BACnet network layer.

Cluster: BACnet Protocol Tunnel, Commands and attributes required to tunnel the BACnet protocol. Command: TransferNpdu

Parameters:

NPDU int8u*

Definition at line **2657** of file **client-command-macro.h**.

#define emberAfFillCommandOtaBootloadClusterImageNotify (payloadType, queryJitter, manufacturerId, imageType, newFileVersion)

This command is generated when the upgrade server wishes to notify the clients of the available OTA upgrade image. The command can be sent as unicast which provides a way for the server to force the upgrade on the client. The command can also be sent as broadcast or multicast to certain class of clients (for example, the ones that have matching manufacturing and device ids).

Cluster: Over the Air Bootloading Cluster, Zigbee Over-the-air Bootloading cluster Command: ImageNotify

Parameters:

payloadType	int8u
queryJitter	int8u
manufacturerId	int16u
imageType	int16u
newFileVersion	int32u

Definition at line **2681** of file **client-command-macro.h**.

#define emberAfFillCommandOtaBootloadClusterQueryNextImageRequest (fieldControl, manufacturerId, imageType, currentFileVersion, hardwareVersion)

This command is generated upon receipt of an Image Notify command to indicate that the client is looking for the next firmware image to upgrade to. The client may also choose to send the command periodically to the server.

Cluster: Over the Air Bootloading Cluster, Zigbee Over-the-air Bootloading cluster
Command: QueryNextImageRequest

Parameters:

fieldControl	int8u
manufacturerId	int16u
imageType	int16u
currentFileVersion	int32u
hardwareVersion	int16u

Definition at line **2704** of file **client-command-macro.h**.

#define emberAfFillCommandOtaBootloadClusterQueryNextImageResponse (status, manufacturerId, imageType, fileVersion, imageSize)

This command is generated upon receipt of an QueryNextImageRequest command to response whether the server has a valid OTA upgrade image for the client or not. If the server has the file, information regarding the file and OTA upgrade process will be included in the command.

Cluster: Over the Air Bootloading Cluster, Zigbee Over-the-air Bootloading cluster
Command: QueryNextImageResponse

Parameters:

status	int8u
manufacturerId	int16u
imageType	int16u
fileVersion	int32u
imageSize	int32u

Definition at line **2727** of file **client-command-macro.h**.

```
#define emberAfFillCommandOtaBootloadClusterImageBlockRequest ( fieldControl,
                                                                manufacturerId,
                                                                imageType,
                                                                fileVersion,
                                                                fileOffset,
                                                                maxDataSize,
                                                                requestNodeAddress )
```

This command is generated by the client to request blocks of OTA upgrade file data.

Cluster: Over the Air Bootloading Cluster, Zigbee Over-the-air Bootloading cluster Command: ImageBlockRequest

Parameters:

fieldControl	int8u
manufacturerId	int16u
imageType	int16u
fileVersion	int32u
fileOffset	int32u
maxDataSize	int8u
requestNodeAddress	int8u*

Definition at line **2752** of file **client-command-macro.h**.

```
#define emberAfFillCommandOtaBootloadClusterImagePageRequest ( fieldControl,
                                                               manufacturerId,
                                                               imageType,
                                                               fileVersion,
                                                               fileOffset,
                                                               maxDataSize,
                                                               pageSize,
                                                               responseSpacing,
                                                               requestNodeAddress )
```

This command is generated by the client to request pages of OTA upgrade file data. A page would contain multiple blocks of data.

Cluster: Over the Air Bootloading Cluster, Zigbee Over-the-air Bootloading cluster Command: ImagePageRequest

Parameters:

fieldControl	int8u
manufacturerId	int16u
imageType	int16u
fileVersion	int32u
fileOffset	int32u
maxDataSize	int8u
pageSize	int16u
responseSpacing	int16u
requestNodeAddress	int8u*

Definition at line **2781** of file **client-command-macro.h**.

#define emberAfFillCommandOtaBootloadClusterImageBlockResponse (status, manufacturerId, imageType, fileVersion, fileOffset, dataSize, imageData)

This command is generated by the server in response to the block or page request command. If the server has the data available, it will reply back with a SUCCESS status. For other error cases, it may reply with status WAIT_FOR_DATA (server does not have the data available yet) or ABORT (invalid requested parameters or other failure cases).

Cluster: Over the Air Bootloading Cluster, Zigbee Over-the-air Bootloading cluster
Command: ImageBlockResponse

Parameters:

status	int8u
manufacturerId	int16u
imageType	int16u
fileVersion	int32u
fileOffset	int32u
dataSize	int8u
imageData	int8u*

Definition at line **2810** of file **client-command-macro.h**.

#define emberAfFillCommandOtaBootloadClusterUpgradeEndRequest (status, manufacturerId, imageType,

fileVersion)

This command is generated by the client to notify the server of the end of the upgrade process. The process may end with success or error status.

Cluster: Over the Air Bootloading Cluster, Zigbee Over-the-air Bootloading cluster
Command: UpgradeEndRequest

Parameters:

status	int8u
manufacturerId	int16u
imageType	int16u
fileVersion	int32u

Definition at line **2834** of file **client-command-macro.h**.

#define emberAfFillCommandOtaBootloadClusterUpgradeEndResponse (manufacturerId, imageType, fileVersion, currentTime, upgradeTime)

This command is generated by the server in response to the upgrade request in order to let the client know when to upgrade to running new firmware image.

Cluster: Over the Air Bootloading Cluster, Zigbee Over-the-air Bootloading cluster
Command: UpgradeEndResponse

Parameters:

manufacturerId	int16u
imageType	int16u
fileVersion	int32u
currentTime	int32u
upgradeTime	int32u

Definition at line **2856** of file **client-command-macro.h**.

#define emberAfFillCommandOtaBootloadClusterQuerySpecificFileRequest (requestNodeAddress, manufacturerId, imageType, fileVersion, currentZigbeeStackVersion)

This command is generated by the client to request a file that is specific to itself. The intention is to provide a way for the client to request non-OTA upgrade file.

Cluster: Over the Air Bootloading Cluster, Zigbee Over-the-air Bootloading cluster
Command: QuerySpecificFileRequest

Parameters:

requestNodeAddress	int8u*
manufacturerId	int16u
imageType	int16u
fileVersion	int32u
currentZigbeeStackVersion	int16u

Definition at line **2879** of file **client-command-macro.h**.

#define emberAfFillCommandOtaBootloadClusterQuerySpecificFileResponse (status, manufacturerId, imageType, fileVersion, imageSize)

This command is generated upon receipt of an QuerySpecificFileRequest command to response whether the server has a valid file for the client or not. If the server has the file, information regarding the file and OTA process will be included in the command.

Cluster: Over the Air Bootloading Cluster, Zigbee Over-the-air Bootloading cluster
Command: QuerySpecificFileResponse

Parameters:

status	int8u
manufacturerId	int16u
imageType	int16u
fileVersion	int32u
imageSize	int32u

Definition at line **2902** of file **client-command-macro.h**.

#define emberAfFillCommandDemandResponseLoadControlClusterLoadControlEvent (issuerEventId, deviceClass, utilityEnrollmentGroup, startTime, durationInMinutes,

```
criticalityLevel,
coolingTemperatureOffset,
heatingTemperatureOffset,
coolingTemperatureSetPoint,
heatingTemperatureSetPoint,
averageLoadAdjustmentPercentage,
dutyCycle,
eventControl )
```

Command description for LoadControlEvent.

Cluster: Demand Response and Load Control, This cluster provides an interface to the functionality of Smart Energy Demand Response and Load Control. Devices targeted by this cluster include thermostats and devices that support load control. Command: LoadControlEvent

Parameters:

issuerEventId	int32u
deviceClass	int16u
utilityEnrollmentGroup	int8u
startTime	int32u
durationInMinutes	int16u
criticalityLevel	int8u
coolingTemperatureOffset	int8u
heatingTemperatureOffset	int8u
coolingTemperatureSetPoint	int16s
heatingTemperatureSetPoint	int16s
averageLoadAdjustmentPercentage	int8s
dutyCycle	int8u
eventControl	int8u

Definition at line **2938** of file **client-command-macro.h**.

```
#define emberAfFillCommandDemandResponseLoadControlClusterCancelLoadControlEvent ( issuerEventId,
deviceClass,
utilityEnrollmentGroup,
cancelControl,
effectiveTime )
```

Command description for CancelLoadControlEvent.

Cluster: Demand Response and Load Control, This cluster provides an interface to the functionality of Smart Energy Demand Response and Load Control. Devices targeted by this cluster include thermostats and devices that support load control. Command: CancelLoadControlEvent

Parameters:

issuerEventId	int32u
deviceClass	int16u
utilityEnrollmentGroup	int8u
cancelControl	int8u
effectiveTime	int32u

Definition at line **2969** of file **client-command-macro.h**.

#define emberAfFillCommandDemandResponseLoadControlClusterCancelAllLoadControlEvents (cancelControl)

Command description for CancelAllLoadControlEvents.

Cluster: Demand Response and Load Control, This cluster provides an interface to the functionality of Smart Energy Demand Response and Load Control. Devices targeted by this cluster include thermostats and devices that support load control. Command: CancelAllLoadControlEvents

Parameters:

cancelControl int8u

Definition at line **2988** of file **client-command-macro.h**.

#define emberAfFillCommandDemandResponseLoadControlClusterReportEventStatus (issuerEventId, eventStatus, eventStatusTime, criticalityLevelApplied, coolingTemperatureSetPointApplied, heatingTemperatureSetPointApplied, averageLoadAdjustmentPercentageApplied, dutyCycleApplied, eventControl, signatureType, signature)

Command description for ReportEventStatus.

Cluster: Demand Response and Load Control, This cluster provides an interface to the functionality of Smart Energy Demand Response and Load Control. Devices targeted by this cluster include thermostats and devices that support load control. Command: ReportEventStatus

Parameters:

issuerEventId	int32u
eventStatus	int8u
eventStatusTime	int32u
criticalityLevelApplied	int8u
coolingTemperatureSetPointApplied	int16s
heatingTemperatureSetPointApplied	int16s
averageLoadAdjustmentPercentageApplied	int8s
dutyCycleApplied	int8u
eventControl	int8u
signatureType	int8u
signature	int8u*

Definition at line **3013** of file **client-command-macro.h**.

#define emberAfFillCommandDemandResponseLoadControlClusterGetScheduledEvents (startTime, numberOfEvents)

Command description for GetScheduledEvents.

Cluster: Demand Response and Load Control, This cluster provides an interface to the functionality of Smart Energy Demand Response and Load Control. Devices targeted by this cluster include thermostats and devices that support load control. Command: GetScheduledEvents

Parameters:

startTime	int32u
numberOfEvents	int8u

Definition at line **3040** of file **client-command-macro.h**.

#define emberAfFillCommandSimpleMeteringClusterGetProfileResponse (endTime, status, profileIntervalPeriod, numberOfPeriodsDelivered, intervals, intervalsLen)

Command description for GetProfileResponse.

Cluster: Simple Metering, The Metering Cluster provides a mechanism to retrieve usage information from Electric, Gas, Water, and potentially Thermal metering devices. Command: GetProfileResponse

Parameters:

endTime	int32u
status	int8u
profileIntervalPeriod	int8u
numberOfPeriodsDelivered	int8u
intervals	int8u*
intervalsLen	int8u

Definition at line **3066** of file **client-command-macro.h**.

#define emberAfFillCommandSimpleMeteringClusterRequestMirror ( )

Command description for RequestMirror.

Cluster: Simple Metering, The Metering Cluster provides a mechanism to retrieve usage information from Electric, Gas, Water, and potentially Thermal metering devices. Command: RequestMirror

Definition at line **3085** of file **client-command-macro.h**.

#define emberAfFillCommandSimpleMeteringClusterRemoveMirror ( )

Command description for RemoveMirror.

Cluster: Simple Metering, The Metering Cluster provides a mechanism to retrieve usage information from Electric, Gas, Water, and potentially Thermal metering devices. Command: RemoveMirror

Definition at line **3098** of file **client-command-macro.h**.

#define emberAfFillCommandSimpleMeteringClusterRequestFastPollModeResponse (appliedUpdatePeriod, fastPollModeEndtime)

Command sent as a response to fast poll request.

Cluster: Simple Metering, The Metering Cluster provides a mechanism to retrieve usage information from Electric, Gas, Water, and potentially Thermal metering devices. Command: RequestFastPollModeResponse

Parameters:

appliedUpdatePeriod	int8u
fastPollModeEndtime	int32u

Definition at line **3113** of file **client-command-macro.h**.

#define emberAfFillCommandSimpleMeteringClusterGetProfile (intervalChannel, endTime, numberOfPeriods)

Command description for GetProfile.

Cluster: Simple Metering, The Metering Cluster provides a mechanism to retrieve usage information from Electric, Gas, Water, and potentially Thermal metering devices. Command: GetProfile

Parameters:

intervalChannel int8u
endTime int32u
numberOfPeriods int8u

Definition at line **3131** of file **client-command-macro.h**.

#define emberAfFillCommandSimpleMeteringClusterRequestMirrorResponse (endpointId)

Command description for RequestMirrorResponse.

Cluster: Simple Metering, The Metering Cluster provides a mechanism to retrieve usage information from Electric, Gas, Water, and potentially Thermal metering devices. Command: RequestMirrorResponse

Parameters:

endpointId int16u

Definition at line **3148** of file **client-command-macro.h**.

#define emberAfFillCommandSimpleMeteringClusterMirrorRemoved (endpointId)

Command description for MirrorRemoved.

Cluster: Simple Metering, The Metering Cluster provides a mechanism to retrieve usage information from Electric, Gas, Water, and potentially Thermal metering devices. Command: MirrorRemoved

Parameters:

endpointId int16u

Definition at line **3163** of file **client-command-macro.h**.

```
#define emberAfFillCommandSimpleMeteringClusterRequestFastPollMode ( fastPollUpdatePeriod,
                                                                     duration              )
```

Command to request fast poll mode.

Cluster: Simple Metering, The Metering Cluster provides a mechanism to retrieve usage information from Electric, Gas, Water, and potentially Thermal metering devices. Command: RequestFastPollMode

Parameters:

fastPollUpdatePeriod	int8u
duration	int8u

Definition at line **3179** of file **client-command-macro.h**.

```
#define emberAfFillCommandPriceClusterPublishPrice ( providerId,
                                                     rateLabel,
                                                     issuerEventId,
                                                     currentTime,
                                                     unitOfMeasure,
                                                     currency,
                                                     priceTrailingDigitAndPriceTier,
                                                     numberOfPriceTiersAndRegisterTier,
                                                     startTime,
                                                     durationInMinutes,
                                                     price,
                                                     priceRatio,
                                                     generationPrice,
                                                     generationPriceRatio,
                                                     alternateCostDelivered,
                                                     alternateCostUnit,
                                                     alternateCostTrailingDigit,
                                                     numberOfBlockThresholds,
                                                     priceControl                 )
```

Command description for PublishPrice.

Cluster: Price, The Price Cluster provides the mechanism for communicating Gas, Energy, or Water pricing information within the premise. Command: PublishPrice

Parameters:

providerId	int32u

rateLabel	int8u*
issuerEventId	int32u
currentTime	int32u
unitOfMeasure	int8u
currency	int16u
priceTrailingDigitAndPriceTier	int8u
numberOfPriceTiersAndRegisterTier	int8u
startTime	int32u
durationInMinutes	int16u
price	int32u
priceRatio	int8u
generationPrice	int32u
generationPriceRatio	int8u
alternateCostDelivered	int32u
alternateCostUnit	int8u
alternateCostTrailingDigit	int8u
numberOfBlockThresholds	int8u
priceControl	int8u

Definition at line **3218** of file **client-command-macro.h**.

#define emberAfFillCommandPriceClusterPublishBlockPeriod (providerId, issuerEventId, blockPeriodStartTime, blockPeriodDurationInMinutes, numberOfPriceTiersAndNumberOfBlockThresholds, blockPeriodControl)

Command description for PublishBlockPeriod.

Cluster: Price, The Price Cluster provides the mechanism for communicating Gas, Energy, or Water pricing information within the premise. Command: PublishBlockPeriod

Parameters:

providerId	int32u
issuerEventId	int32u
blockPeriodStartTime	int32u
blockPeriodDurationInMinutes	int32u
numberOfPriceTiersAndNumberOfBlockThresholds	int8u
blockPeriodControl	int8u

Definition at line **3256** of file **client-command-macro.h**.

#define emberAfFillCommandPriceClusterGetCurrentPrice (commandOptions)

Command description for GetCurrentPrice.

Cluster: Price, The Price Cluster provides the mechanism for communicating Gas, Energy, or Water pricing information within the premise. Command: GetCurrentPrice

Parameters:

commandOptions int8u

Definition at line **3276** of file **client-command-macro.h**.

#define emberAfFillCommandPriceClusterGetScheduledPrices (startTime, numberOfEvents)

Command description for GetScheduledPrices.

Cluster: Price, The Price Cluster provides the mechanism for communicating Gas, Energy, or Water pricing information within the premise. Command: GetScheduledPrices

Parameters:

startTime int32u
numberOfEvents int8u

Definition at line **3292** of file **client-command-macro.h**.

#define emberAfFillCommandPriceClusterPriceAcknowledgement (providerId, issuerEventId, priceAckTime, control)

Command description for PriceAcknowledgement.

Cluster: Price, The Price Cluster provides the mechanism for communicating Gas, Energy, or Water pricing information within the premise. Command: PriceAcknowledgement

Parameters:

providerId int32u
issuerEventId int32u
priceAckTime int32u
control int8u

Definition at line **3311** of file **client-command-macro.h**.

#define emberAfFillCommandPriceClusterGetBlockPeriods (startTime, numberOfEvents)

Command description for GetBlockPeriod(s).

Cluster: Price, The Price Cluster provides the mechanism for communicating Gas, Energy, or Water pricing information within the premise. Command: GetBlockPeriods

Parameters:

startTime	int32u
numberOfEvents	int8u

Definition at line **3330** of file **client-command-macro.h**.

#define emberAfFillCommandMessagingClusterDisplayMessage (messageId, messageControl, startTime, durationInMinutes, message)

Command description for DisplayMessage.

Cluster: Messaging, This cluster provides an interface for passing text messages between SE devices. Command: DisplayMessage

Parameters:

messageId	int32u
messageControl	int8u
startTime	int32u
durationInMinutes	int16u
message	int8u*

Definition at line **3355** of file **client-command-macro.h**.

#define emberAfFillCommandMessagingClusterCancelMessage (messageId, messageControl)

Command description for CancelMessage.

Cluster: Messaging, This cluster provides an interface for passing text messages between SE devices.
Command: CancelMessage

Parameters:

messageId int32u
messageControl int8u

Definition at line **3375** of file **client-command-macro.h**.

#define emberAfFillCommandMessagingClusterGetLastMessage ( )

Command description for GetLastMessage.

Cluster: Messaging, This cluster provides an interface for passing text messages between SE devices.
Command: GetLastMessage

Definition at line **3390** of file **client-command-macro.h**.

#define emberAfFillCommandMessagingClusterMessageConfirmation (messageId, confirmationTime)

Command description for MessageConfirmation.

Cluster: Messaging, This cluster provides an interface for passing text messages between SE devices.
Command: MessageConfirmation

Parameters:

messageId int32u
confirmationTime int32u

Definition at line **3405** of file **client-command-macro.h**.

#define emberAfFillCommandTunnelingClusterRequestTunnel (protocolId, manufacturerCode, flowControlSupport)

Command description for RequestTunnel.

Cluster: Tunneling, The tunneling cluster provides an interface for tunneling protocols. Command: RequestTunnel

Parameters:

protocolId int8u
manufacturerCode int16u
flowControlSupport int8u

Definition at line **3428** of file **client-command-macro.h**.

#define emberAfFillCommandTunnelingClusterCloseTunnel (tunnelId)

Command description for CloseTunnel.

Cluster: Tunneling, The tunneling cluster provides an interface for tunneling protocols. Command: CloseTunnel

Parameters:
tunnelId int16u

Definition at line **3445** of file **client-command-macro.h**.

#define emberAfFillCommandTunnelingClusterTransferDataClientToServer (tunnelId, data, dataLen)

Command description for TransferData.

Cluster: Tunneling, The tunneling cluster provides an interface for tunneling protocols.
Command: TransferDataClientToServer

Parameters:
tunnelId int16u
data int8u*
dataLen int8u

Definition at line **3462** of file **client-command-macro.h**.

#define emberAfFillCommandTunnelingClusterTransferDataErrorClientToServer (tunnelId, transferDataStatus)

Command description for TransferDataError.

Cluster: Tunneling, The tunneling cluster provides an interface for tunneling protocols.
Command: TransferDataErrorClientToServer

Parameters:

tunnelId	int16u
transferDataStatus	int8u

Definition at line **3480** of file **client-command-macro.h**.

#define emberAfFillCommandTunnelingClusterAckTransferDataClientToServer (tunnelId, numberOfBytesLeft)

Command description for AckTransferData.

Cluster: Tunneling, The tunneling cluster provides an interface for tunneling protocols.
Command: AckTransferDataClientToServer

Parameters:

tunnelId	int16u
numberOfBytesLeft	int16u

Definition at line **3497** of file **client-command-macro.h**.

#define emberAfFillCommandTunnelingClusterReadyDataClientToServer (tunnelId, numberOfOctetsLeft)

Command description for ReadyData.

Cluster: Tunneling, The tunneling cluster provides an interface for tunneling protocols.
Command: ReadyDataClientToServer

Parameters:

tunnelId	int16u
numberOfOctetsLeft	int16u

Definition at line **3514** of file **client-command-macro.h**.

#define emberAfFillCommandTunnelingClusterRequestTunnelResponse (tunnelId, tunnelStatus)

Command description for RequestTunnelResponse.

Cluster: Tunneling, The tunneling cluster provides an interface for tunneling protocols.
Command: RequestTunnelResponse

Parameters:

tunnelId int16u
tunnelStatus int8u

Definition at line **3531** of file **client-command-macro.h**.

#define emberAfFillCommandTunnelingClusterTransferDataServerToClient (tunnelId, data, dataLen)

Command description for TransferData.

Cluster: Tunneling, The tunneling cluster provides an interface for tunneling protocols.
Command: TransferDataServerToClient

Parameters:

tunnelId int16u
data int8u*
dataLen int8u

Definition at line **3549** of file **client-command-macro.h**.

#define emberAfFillCommandTunnelingClusterTransferDataErrorServerToClient (tunnelId, transferDataStatus)

Command description for TransferDataError.

Cluster: Tunneling, The tunneling cluster provides an interface for tunneling protocols.
Command: TransferDataErrorServerToClient

Parameters:

tunnelId int16u
transferDataStatus int8u

Definition at line **3567** of file **client-command-macro.h**.

#define emberAfFillCommandTunnelingClusterAckTransferDataServerToClient (tunnelId, numberOfBytesLeft)

Command description for AckTransferData.

Cluster: Tunneling, The tunneling cluster provides an interface for tunneling protocols.
Command: AckTransferDataServerToClient

Parameters:

tunnelId	int16u
numberOfBytesLeft	int16u

Definition at line **3584** of file **client-command-macro.h**.

#define emberAfFillCommandTunnelingClusterReadyDataServerToClient (tunnelId, numberOfOctetsLeft)

Command description for ReadyData.

Cluster: Tunneling, The tunneling cluster provides an interface for tunneling protocols.
Command: ReadyDataServerToClient

Parameters:

tunnelId	int16u
numberOfOctetsLeft	int16u

Definition at line **3601** of file **client-command-macro.h**.

#define emberAfFillCommandPrepaymentClusterSelectAvailableEmergencyCredit (commandDateTime, originatingDevice, siteId, meterSerialNumber)

Command description for SelectAvailableEmergencyCredit.

Cluster: Prepayment, The Prepayment Cluster provides the facility to pass messages relating to prepayment between devices on the HAN. Command: SelectAvailableEmergencyCredit

Parameters:

commandDateTime	int32u
originatingDevice	int8u
siteId	int8u*
meterSerialNumber	int8u*

Definition at line **3625** of file **client-command-macro.h**.

#define emberAfFillCommandPrepaymentClusterChangeSupply (providerId,

```
requestDateTime,
siteId,
meterSerialNumber,
implementationDateTime,
proposedSupplyStatus,
originatorIdSupplyControlBits )
```

Command description for ChangeSupply.

Cluster: Prepayment, The Prepayment Cluster provides the facility to pass messages relating to prepayment between devices on the HAN. Command: ChangeSupply

Parameters:

providerId	int32u
requestDateTime	int32u
siteId	int8u*
meterSerialNumber	int8u*
implementationDateTime	int32u
proposedSupplyStatus	int8u
originatorIdSupplyControlBits	int8u

Definition at line **3649** of file **client-command-macro.h**.

```
#define emberAfFillCommandPrepaymentClusterSupplyStatusResponse ( providerId,
                                                                  implementationtDateTime,
                                                                  supplyStatus )
```

Command description for SupplyStatusResponse.

Cluster: Prepayment, The Prepayment Cluster provides the facility to pass messages relating to prepayment between devices on the HAN. Command: SupplyStatusResponse

Parameters:

providerId	int32u
implementationtDateTime	int32u
supplyStatus	int8u

Definition at line **3672** of file **client-command-macro.h**.

```
#define emberAfFillCommandKeyEstablishmentClusterInitiateKeyEstablishmentRequest ( keyEstablishmentSuite,
                                                                                   ephemeralDataGenerateTime,
```

confirmKeyGenerateTime,
identity)

Command description for InitiateKeyEstablishmentRequest.

Cluster: Key establishment, Key establishment cluster Command: InitiateKeyEstablishmentRequest

Parameters:

keyEstablishmentSuite	int16u
ephemeralDataGenerateTime	int8u
confirmKeyGenerateTime	int8u
identity	int8u*

Definition at line **3697** of file **client-command-macro.h**.

#define emberAfFillCommandKeyEstablishmentClusterEphemeralDataRequest (ephemeralData)

Command description for EphemeralDataRequest.

Cluster: Key establishment, Key establishment cluster Command: EphemeralDataRequest

Parameters:

ephemeralData int8u*

Definition at line **3716** of file **client-command-macro.h**.

#define emberAfFillCommandKeyEstablishmentClusterConfirmKeyDataRequest (secureMessageAuthenticationCode)

Command description for ConfirmKeyDataRequest.

Cluster: Key establishment, Key establishment cluster Command: ConfirmKeyDataRequest

Parameters:

secureMessageAuthenticationCode int8u*

Definition at line **3732** of file **client-command-macro.h**.

#define emberAfFillCommandKeyEstablishmentClusterServerToClientTerminateKeyEstablishment (statusCode, waitTime, keyEstablishmentSuite)

Command description for TerminateKeyEstablishment.

Cluster: Key establishment, Key establishment cluster Command: TerminateKeyEstablishment

Parameters:

statusCode	int8u
waitTime	int8u
keyEstablishmentSuite	int16u

Definition at line **3749** of file **client-command-macro.h**.

#define emberAfFillCommandKeyEstablishmentClusterClientToServerTerminateKeyEstablishment (statusCode, waitTime, keyEstablishmentSuite)

Command description for TerminateKeyEstablishment.

Cluster: Key establishment, Key establishment cluster Command: TerminateKeyEstablishment

Parameters:

statusCode	int8u
waitTime	int8u
keyEstablishmentSuite	int16u

Definition at line **3768** of file **client-command-macro.h**.

#define emberAfFillCommandKeyEstablishmentClusterInitiateKeyEstablishmentResponse (requestedKeyEstablishmentSuite, ephemeralDataGenerateTime, confirmKeyGenerateTime, identity)

Command description for InitiateKeyEstablishmentResponse.

Cluster: Key establishment, Key establishment cluster Command: InitiateKeyEstablishmentResponse

Parameters:

requestedKeyEstablishmentSuite	int16u
ephemeralDataGenerateTime	int8u
confirmKeyGenerateTime	int8u
identity	int8u*

Definition at line **3788** of file **client-command-macro.h**.

#define emberAfFillCommandKeyEstablishmentClusterEphemeralDataResponse (ephemeralData)

Command description for EphemeralDataResponse.

Cluster: Key establishment, Key establishment cluster Command: EphemeralDataResponse

Parameters:

ephemeralData int8u*

Definition at line **3807** of file **client-command-macro.h**.

#define emberAfFillCommandKeyEstablishmentClusterConfirmKeyDataResponse (secureMessageAuthenticationCode)

Command description for ConfirmKeyDataResponse.

Cluster: Key establishment, Key establishment cluster Command: ConfirmKeyDataResponse

Parameters:

secureMessageAuthenticationCode int8u*

Definition at line **3823** of file **client-command-macro.h**.

#define emberAfFillCommandSampleMfgSpecificClusterCommandOne (argOne)

A sample manufacturer specific command within the sample manufacturer specific cluster.

Cluster: Sample Mfg Specific Cluster, This cluster provides an example of how the Application Framework can be extended to include manufacturer specific clusters. Command: CommandOne

Parameters:

argOne int8u

Definition at line **3845** of file **client-command-macro.h**.

Ember Application Framework Command Line Interface (CLI)

Servicediscovery

Data Structures

Here are the data structures with brief descriptions:

_BlockThreshold	
_ChatParticipant	
_ChatRoom	
_ConfigureReportingRecord	
_ConfigureReportingStatusRecord	
_DiscoverAttributesInfoRecord	
_NeighborInfo	
_NodeInformation	
_Notification	
_ReadAttributeStatusRecord	
_ReadReportingConfigurationAttributeRecord	
_ReadReportingConfigurationRecord	
_ReportAttributeRecord	
_SceneExtensionAttributeInfo	
_SceneExtensionFieldSet	
_WriteAttributeRecord	
_WriteAttributeStatusRecord	
EmAfOtaStorageLinuxConfig	
EmberAfAllowedInterPanMessage	This structure is used define an interpan message that will be accepted by the interpan filters
EmberAfAttributeMetadata	Each attribute has it's metadata stored in such struct
EmberAfAttributeMinMaxValue	Type describing the attribute default, min and max values
EmberAfAttributeSearchRecord	Struct used to find an attribute in storage. Together the elements in this search record constitute the "primary key" used to identify a unique attribute value in attribute storage
EmberAfCluster	Struct describing cluster
EmberAfClusterCommand	The EmberAFClusterCommand is a struct wrapper for all the data pertaining to a command which comes in over the air. This enables struct is used to encapsulate a command in a single place on the stack and pass a pointer to that location around during command processing
EmberAfDefaultAttributeValue	Type for default values
EmberAfDefaultOrMinMaxAttributeValue	Union describing the attribute default/min/max values
EmberAfDefinedEndpoint	Struct that maps actual endpoint type, onto a specific endpoint
EmberAfEndpointList	A list of endpoints received during a service discovery attempt. This will be returned for a match descriptor request and a active endpoint request
EmberAfEndpointType	Endpoint type struct describes clusters that are on the endpoint
EmberAfEventContext	Structure used to keep track of cluster related events and their sleep control values. The cluster code will not know at runtime all of the events that it has access to in the event table This structure is stored by the application framework in an event context table which allong with helper functions allows the cluster code to schedule and deactivate its associated events

EmberAfIasZoneTableEntry	
EmberAfIncomingMessage	Struct containing the superset of values passed to both emberIncomingMessageHandler on the SOC and ezspIncomingMessageHandler on the host
EmberAfInterpanHeader	Interpan header used for sending and receiving interpan messages
EmberAfLoadControlEvent	Struct that describes a load control event
EmberAfManufacturerCodeEntry	
EmberAfMessagingServerMessage	
EmberAfOtaHeader	This structure is an in-memory representation of the Over-the-air header data that resides on disk. It is not a byte-for-byte copy
EmberAfOtaImageId	This is a unique identifier for referencing ZigBee Over-the-air upgrade images. It is used by the OTA plugins when passing around information about an upgrade file
EmberAfReportEntry	A structure used to reporting configurations
EmberAfSceneTableEntry	A structure used to store scene table entries in RAM or in tokens, depending on a plugin setting. If endpoint field is **EMBER_AF_SCENE_TABLE_UNUSED_ENDPOINT_ID**, the entry is unused
EmberAfScheduledPrice	
EmberAfServiceDiscoveryResult	A structure containing general information about the service discovery
EmberAfTagData	This structure contains information about a tag that resides within an Over-the-air bootload file

_BlockThreshold Struct Reference

```
#include <af-structs.h>
```

Data Fields

int8u	**blockThreshold**
int8u	**priceControl**
int32u	**blockPeriodStartTime**
int32u	**blockPeriodDurationMinutes**
int8u	**fuelType**
int32u	**standingCharge**

Detailed Description

Definition at line **110** of file **af-structs.h**.

Field Documentation

int8u _BlockThreshold::blockThreshold

Definition at line **111** of file **af-structs.h**.

int8u _BlockThreshold::priceControl

Definition at line **112** of file **af-structs.h**.

int32u _BlockThreshold::blockPeriodStartTime

Definition at line **113** of file **af-structs.h**.

int32u _BlockThreshold::blockPeriodDurationMinutes

Definition at line **114** of file **af-structs.h**.

int8u _BlockThreshold::fuelType

Definition at line **115** of file **af-structs.h**.

int32u _BlockThreshold::standingCharge

Definition at line **116** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_ChatParticipant Struct Reference

```
#include <af-structs.h>
```

Data Fields

int16u	**uid**
int8u *	**nickname**

Detailed Description

Definition at line **139** of file **af-structs.h**.

Field Documentation

int16u _ChatParticipant::uid

Definition at line **140** of file **af-structs.h**.

int8u* _ChatParticipant::nickname

Definition at line **141** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_ChatRoom Struct Reference

```
#include <af-structs.h>
```

Data Fields

int16u **cid**
int8u * **name**

Detailed Description

Definition at line **146** of file **af-structs.h**.

Field Documentation

int16u _ChatRoom::cid

Definition at line **147** of file **af-structs.h**.

int8u* _ChatRoom::name

Definition at line **148** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_ConfigureReportingRecord Struct Reference

```
#include <af-structs.h>
```

Data Fields

int8u	**direction**
int16u	**attributeId**
int8u	**attributeType**
int16u	**minimumReportingInterval**
int16u	**maximumReportingInterval**
int8u	**reportableChangeLocation**
int16u	**timeoutPeriod**

Detailed Description

Definition at line **40** of file **af-structs.h**.

Field Documentation

int8u _ConfigureReportingRecord::direction

Definition at line **41** of file **af-structs.h**.

int16u _ConfigureReportingRecord::attributeId

Definition at line **42** of file **af-structs.h**.

int8u _ConfigureReportingRecord::attributeType

Definition at line **43** of file **af-structs.h**.

int16u _ConfigureReportingRecord::minimumReportingInterval

Definition at line **44** of file **af-structs.h**.

int16u _ConfigureReportingRecord::maximumReportingInterval

Definition at line **45** of file **af-structs.h**.

int8u _ConfigureReportingRecord::reportableChangeLocation

Definition at line **46** of file **af-structs.h**.

int16u _ConfigureReportingRecord::timeoutPeriod

Definition at line **47** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_ConfigureReportingStatusRecord Struct Reference

```
#include <af-structs.h>
```

Data Fields

int8u	**status**
int8u	**direction**
int16u	**attributeId**

Detailed Description

Definition at line **52** of file **af-structs.h**.

Field Documentation

int8u _ConfigureReportingStatusRecord::status

Definition at line **53** of file **af-structs.h**.

int8u _ConfigureReportingStatusRecord::direction

Definition at line **54** of file **af-structs.h**.

int16u _ConfigureReportingStatusRecord::attributeId

Definition at line **55** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_DiscoverAttributesInfoRecord Struct Reference

```
#include <af-structs.h>
```

Data Fields

int16u	**attributeId**
int8u	**attributeType**

Detailed Description

Definition at line **88** of file **af-structs.h**.

Field Documentation

int16u _DiscoverAttributesInfoRecord::attributeId

Definition at line **89** of file **af-structs.h**.

int8u _DiscoverAttributesInfoRecord::attributeType

Definition at line **90** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_NeighborInfo Struct Reference

```
#include <af-structs.h>
```

Data Fields

int8u *	**neighbor**
int16s	**x**
int16s	**y**
int16s	**z**
int8s	**rssi**
int8u	**numberRssiMeasurements**

Detailed Description

Definition at line **128** of file **af-structs.h**.

Field Documentation

int8u* _NeighborInfo::neighbor

Definition at line **129** of file **af-structs.h**.

int16s _NeighborInfo::x

Definition at line **130** of file **af-structs.h**.

int16s _NeighborInfo::y

Definition at line **131** of file **af-structs.h**.

int16s _NeighborInfo::z

Definition at line **132** of file **af-structs.h**.

int8s _NeighborInfo::rssi

Definition at line **133** of file **af-structs.h**.

int8u _NeighborInfo::numberRssiMeasurements

Definition at line **134** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_NodeInformation Struct Reference

```
#include <af-structs.h>
```

Data Fields

int16u	**uid**
int16u	**address**
int8u	**endpoint**
int8u *	**nickname**

Detailed Description

Definition at line **153** of file **af-structs.h**.

Field Documentation

int16u _NodeInformation::uid

Definition at line **154** of file **af-structs.h**.

int16u _NodeInformation::address

Definition at line **155** of file **af-structs.h**.

int8u _NodeInformation::endpoint

Definition at line **156** of file **af-structs.h**.

int8u* _NodeInformation::nickname

Definition at line **157** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_Notification Struct Reference

```
#include <af-structs.h>
```

Data Fields

int16u	**contentId**
int8u	**statusFeedback**

Detailed Description

Definition at line **121** of file **af-structs.h**.

Field Documentation

int16u _Notification::contentId

Definition at line **122** of file **af-structs.h**.

int8u _Notification::statusFeedback

Definition at line **123** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_ReadAttributeStatusRecord Struct Reference

```
#include <af-structs.h>
```

Data Fields

int16u	**attributeId**
int8u	**status**
int8u	**attributeType**
int8u	**attributeLocation**

Detailed Description

Definition at line **16** of file **af-structs.h**.

Field Documentation

int16u _ReadAttributeStatusRecord::attributeId

Definition at line **17** of file **af-structs.h**.

int8u _ReadAttributeStatusRecord::status

Definition at line **18** of file **af-structs.h**.

int8u _ReadAttributeStatusRecord::attributeType

Definition at line **19** of file **af-structs.h**.

int8u _ReadAttributeStatusRecord::attributeLocation

Definition at line **20** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_ReadReportingConfigurationAttributeRecord Struct Reference

```
#include <af-structs.h>
```

Data Fields

int8u **direction**
int16u **attributeId**

Detailed Description

Definition at line **73** of file **af-structs.h**.

Field Documentation

int8u _ReadReportingConfigurationAttributeRecord::direction

Definition at line **74** of file **af-structs.h**.

int16u _ReadReportingConfigurationAttributeRecord::attributeId

Definition at line **75** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_ReadReportingConfigurationRecord Struct Reference

```
#include <af-structs.h>
```

Data Fields

int8u	**status**
int8u	**direction**
int16u	**attributeId**
int8u	**attributeType**
int16u	**minimumReportingInterval**
int16u	**maximumReportingInterval**
int8u	**reportableChangeLocation**
int16u	**timeoutPeriod**

Detailed Description

Definition at line **60** of file **af-structs.h**.

Field Documentation

int8u _ReadReportingConfigurationRecord::status

Definition at line **61** of file **af-structs.h**.

int8u _ReadReportingConfigurationRecord::direction

Definition at line **62** of file **af-structs.h**.

int16u _ReadReportingConfigurationRecord::attributeId

Definition at line **63** of file **af-structs.h**.

int8u _ReadReportingConfigurationRecord::attributeType

Definition at line **64** of file **af-structs.h**.

int16u _ReadReportingConfigurationRecord::minimumReportingInterval

Definition at line **65** of file **af-structs.h**.

int16u _ReadReportingConfigurationRecord::maximumReportingInterval

Definition at line **66** of file **af-structs.h**.

int8u _ReadReportingConfigurationRecord::reportableChangeLocation

Definition at line **67** of file **af-structs.h**.

int16u _ReadReportingConfigurationRecord::timeoutPeriod

Definition at line **68** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_ReportAttributeRecord Struct Reference

```
#include <af-structs.h>
```

Data Fields

int16u	**attributeId**
int8u	**attributeType**
int8u	**attributeLocation**

Detailed Description

Definition at line **80** of file **af-structs.h**.

Field Documentation

int16u _ReportAttributeRecord::attributeId

Definition at line **81** of file **af-structs.h**.

int8u _ReportAttributeRecord::attributeType

Definition at line **82** of file **af-structs.h**.

int8u _ReportAttributeRecord::attributeLocation

Definition at line **83** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_SceneExtensionAttributeInfo Struct Reference

```
#include <af-structs.h>
```

Data Fields

int8u	**attributeType**
int8u	**attributeLocation**

Detailed Description

Definition at line **95** of file **af-structs.h**.

Field Documentation

int8u _SceneExtensionAttributeInfo::attributeType

Definition at line **96** of file **af-structs.h**.

int8u _SceneExtensionAttributeInfo::attributeLocation

Definition at line **97** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_SceneExtensionFieldSet Struct Reference

```
#include <af-structs.h>
```

Data Fields

int16u	**clusterId**
int8u	**length**
int8u	**value**

Detailed Description

Definition at line **102** of file **af-structs.h**.

Field Documentation

int16u _SceneExtensionFieldSet::clusterId

Definition at line **103** of file **af-structs.h**.

int8u _SceneExtensionFieldSet::length

Definition at line **104** of file **af-structs.h**.

int8u _SceneExtensionFieldSet::value

Definition at line **105** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_WriteAttributeRecord Struct Reference

```
#include <af-structs.h>
```

Data Fields

int16u	**attributeId**
int8u	**attributeType**
int8u	**attributeLocation**

Detailed Description

Definition at line **25** of file **af-structs.h**.

Field Documentation

int16u _WriteAttributeRecord::attributeId

Definition at line **26** of file **af-structs.h**.

int8u _WriteAttributeRecord::attributeType

Definition at line **27** of file **af-structs.h**.

int8u _WriteAttributeRecord::attributeLocation

Definition at line **28** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

_WriteAttributeStatusRecord Struct Reference

```
#include <af-structs.h>
```

Data Fields

int8u	**status**
int16u	**attributeId**

Detailed Description

Definition at line **33** of file **af-structs.h**.

Field Documentation

int8u _WriteAttributeStatusRecord::status

Definition at line **34** of file **af-structs.h**.

int16u _WriteAttributeStatusRecord::attributeId

Definition at line **35** of file **af-structs.h**.

The documentation for this struct was generated from the following file:

- **af-structs.h**

EmAfOtaStorageLinuxConfig Struct Reference

```
#include <ota-storage-linux.h>
```

Data Fields

boolean	**memoryDebug**
boolean	**fileDebug**
boolean	**fieldDebug**
boolean	**ignoreFilesWithUnderscorePrefix**
boolean	**printFileDiscoveryOrRemoval**
EmAfOtaStorageFileAddedHandler *	**fileAddedHandler**

Detailed Description

Definition at line **5** of file **ota-storage-linux.h**.

Field Documentation

boolean EmAfOtaStorageLinuxConfig::memoryDebug

Definition at line **6** of file **ota-storage-linux.h**.

boolean EmAfOtaStorageLinuxConfig::fileDebug

Definition at line **7** of file **ota-storage-linux.h**.

boolean EmAfOtaStorageLinuxConfig::fieldDebug

Definition at line **8** of file **ota-storage-linux.h**.

boolean EmAfOtaStorageLinuxConfig::ignoreFilesWithUnderscorePrefix

Definition at line **9** of file **ota-storage-linux.h**.

boolean EmAfOtaStorageLinuxConfig::printFileDiscoveryOrRemoval

Definition at line **10** of file **ota-storage-linux.h**.

EmAfOtaStorageFileAddedHandler* EmAfOtaStorageLinuxConfig::fileAddedHandler

Definition at line **11** of file **ota-storage-linux.h**.

The documentation for this struct was generated from the following file:

- **ota-storage-linux.h**

EmberAfAllowedInterPanMessage Struct Reference

[Application Framework types]

This structure is used define an interpan message that will be accepted by the interpan filters. More...

```
#include <af-types.h>
```

Data Fields

EmberAfProfileId	profileId
EmberAfClusterId	clusterId
int8u	commandId
EmberAfAllowedInterpanOptions	options

Detailed Description

This structure is used define an interpan message that will be accepted by the interpan filters.

Definition at line 369 of file af-types.h.

Field Documentation

EmberAfProfileId EmberAfAllowedInterPanMessage::profileId

Definition at line 370 of file af-types.h.

EmberAfClusterId EmberAfAllowedInterPanMessage::clusterId

Definition at line 371 of file af-types.h.

int8u EmberAfAllowedInterPanMessage::commandId

Definition at line **372** of file **af-types.h**.

EmberAfAllowedInterpanOptions EmberAfAllowedInterPanMessage::options

Definition at line **373** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfAttributeMetadata Struct Reference

[Application Framework types]

Each attribute has it's metadata stored in such struct. More...

```
#include <af-types.h>
```

Data Fields

EmberAfAttributeId	attributeId
EmberAfAttributeType	attributeType
int8u	size
EmberAfAttributeMask	mask
EmberAfDefaultOrMinMaxAttributeValue	defaultValue

Detailed Description

Each attribute has it's metadata stored in such struct.

There is only one of these per attribute across all endpoints.

Definition at line 134 of file af-types.h.

Field Documentation

EmberAfAttributeId EmberAfAttributeMetadata::attributeId

Attribute ID, according to ZCL specs.

Definition at line 138 of file af-types.h.

EmberAfAttributeType EmberAfAttributeMetadata::attributeType

Attribute type, according to ZCL specs.

Definition at line 142 of file af-types.h.

int8u EmberAfAttributeMetadata::size

Size of this attribute in bytes.

Definition at line **146** of file **af-types.h**.

EmberAfAttributeMask EmberAfAttributeMetadata::mask

Attribute mask, tagging attribute with specific functionality. See ATTRIBUTE_MASK_ macros defined in **app/framework/gen/attribute-storage.h**.

Definition at line **152** of file **af-types.h**.

EmberAfDefaultOrMinMaxAttributeValue EmberAfAttributeMetadata::defaultValue

Pointer to the default value union. Actual value stored depends on the mask.

Definition at line **157** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfAttributeMinMaxValue Struct Reference

[Application Framework types]

Type describing the attribute default, min and max values. More...

```
#include <af-types.h>
```

Data Fields

EmberAfDefaultAttributeValue defaultValue
EmberAfDefaultAttributeValue minValue
EmberAfDefaultAttributeValue maxValue

Detailed Description

Type describing the attribute default, min and max values.

This struct is required if the attribute mask specifies that this attribute has a known min and max values.

Definition at line **92** of file **af-types.h**.

Field Documentation

EmberAfDefaultAttributeValue EmberAfAttributeMinMaxValue::defaultValue

Default value of the attribute.

Definition at line **96** of file **af-types.h**.

EmberAfDefaultAttributeValue EmberAfAttributeMinMaxValue::minValue

Minimum allowed value

Definition at line **100** of file **af-types.h**.

EmberAfDefaultAttributeValue EmberAfAttributeMinMaxValue::maxValue

Maximum allowed value.

Definition at line **104** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfAttributeSearchRecord Struct Reference

[Application Framework types]

Struct used to find an attribute in storage. Together the elements in this search record constitute the "primary key" used to identify a unique attribute value in attribute storage. More...

```
#include <af-types.h>
```

Data Fields

int8u	**endpoint**
EmberAfClusterId	**clusterId**
EmberAfClusterMask	**clusterMask**
EmberAfAttributeId	**attributeId**
int16u	**manufacturerCode**

Detailed Description

Struct used to find an attribute in storage. Together the elements in this search record constitute the "primary key" used to identify a unique attribute value in attribute storage.

Definition at line **200** of file **af-types.h**.

Field Documentation

int8u EmberAfAttributeSearchRecord::endpoint

Endpoint that the attribute is located on

Definition at line **205** of file **af-types.h**.

EmberAfClusterId EmberAfAttributeSearchRecord::clusterId

Cluster that the attribute is located on. If the cluster id is inside the manufacturer specific

range, 0xfc00 - 0xffff, The manufacturer code should also be set to the code associated with the manufacturer specific cluster.

Definition at line **213** of file **af-types.h**.

EmberAfClusterMask EmberAfAttributeSearchRecord::clusterMask

Cluster mask for the cluster, used to determine if it is the server or client version of the cluster. See CLUSTER_MASK_ macros defined in **app/framework/gen/attribute-storage.h**

Definition at line **220** of file **af-types.h**.

EmberAfAttributeId EmberAfAttributeSearchRecord::attributeId

The two byte identifier for the attribute. If the cluster id is inside the manufacturer specific range 0xfc00 - 0xffff, or the manufacturer code is NOT 0, the attribute is assumed to be manufacturer specific.

Definition at line **227** of file **af-types.h**.

int16u EmberAfAttributeSearchRecord::manufacturerCode

Manufacturer Code associated with the cluster and or attribute. If the cluster id is inside the manufacturer specific range, this value should indicate the manufacturer code for the manufacturer specific cluster. Otherwise if this value is non zero, and the cluster id is a standard ZCL cluster, it is assumed that the attribute being sought is a manufacturer specific extension to the standard ZCL cluster indicated by the cluster id.

Definition at line **238** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfCluster Struct Reference

[Application Framework types]

Struct describing cluster. More...

```
#include <af-types.h>
```

Data Fields

EmberAfClusterId	clusterId
EmberAfAttributeMetadata *	attributes
int16u	attributeCount
int16u	clusterSize
EmberAfClusterMask	mask
const EmberAfGenericClusterFunction *	functions

Detailed Description

Struct describing cluster.

Definition at line 163 of file af-types.h.

Field Documentation

EmberAfClusterId EmberAfCluster::clusterId

ID of cluster according to ZCL spec

Definition at line 167 of file af-types.h.

EmberAfAttributeMetadata* EmberAfCluster::attributes

Pointer to attribute metadata array for this cluster.

Definition at line **171** of file **af-types.h**.

int16u EmberAfCluster::attributeCount

Total number of attributes

Definition at line **175** of file **af-types.h**.

int16u EmberAfCluster::clusterSize

Total size of non-external, non-singleton attribute for this cluster.

Definition at line **179** of file **af-types.h**.

EmberAfClusterMask EmberAfCluster::mask

Mask with additional functionality for cluster. See CLUSTER_MASK macros.

Definition at line **184** of file **af-types.h**.

const EmberAfGenericClusterFunction* EmberAfCluster::functions

An array into the cluster functions. The length of the array is determined by the function bits in mask. This may be null if this cluster has no functions.

Definition at line **191** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfClusterCommand Struct Reference

[Application Framework types]

The EmberAFClusterCommand is a struct wrapper for all the data pertaining to a command which comes in over the air. This enables struct is used to encapsulate a command in a single place on the stack and pass a pointer to that location around during command processing. More...

```
#include <af-types.h>
```

Data Fields

EmberApsFrame *	**apsFrame**
EmberIncomingMessageType	**type**
EmberNodeId	**source**
int8u *	**buffer**
int16u	**bufLen**
boolean	**clusterSpecific**
boolean	**mfgSpecific**
int16u	**mfgCode**
int8u	**seqNum**
int8u	**commandId**
int8u	**payloadStartIndex**
int8u	**direction**
EmberAfInterpanHeader *	**interPanHeader**

Detailed Description

The EmberAFClusterCommand is a struct wrapper for all the data pertaining to a command which comes in over the air. This enables struct is used to encapsulate a command in a single place on the stack and pass a pointer to that location around during command processing.

Definition at line **385** of file **af-types.h**.

Field Documentation

EmberApsFrame* EmberAfClusterCommand::apsFrame

Aps frame for the incoming message

Definition at line **389** of file **af-types.h**.

EmberIncomingMessageType EmberAfClusterCommand::type

Definition at line **390** of file **af-types.h**.

EmberNodeId EmberAfClusterCommand::source

Definition at line **391** of file **af-types.h**.

int8u* EmberAfClusterCommand::buffer

Definition at line **392** of file **af-types.h**.

int16u EmberAfClusterCommand::bufLen

Definition at line **393** of file **af-types.h**.

boolean EmberAfClusterCommand::clusterSpecific

Definition at line **394** of file **af-types.h**.

boolean EmberAfClusterCommand::mfgSpecific

Definition at line **395** of file **af-types.h**.

int16u EmberAfClusterCommand::mfgCode

Definition at line **396** of file **af-types.h**.

int8u EmberAfClusterCommand::seqNum

Definition at line **397** of file **af-types.h**.

int8u EmberAfClusterCommand::commandId

Definition at line **398** of file **af-types.h**.

int8u EmberAfClusterCommand::payloadStartIndex

Definition at line **399** of file **af-types.h**.

int8u EmberAfClusterCommand::direction

Definition at line **400** of file **af-types.h**.

EmberAfInterpanHeader* EmberAfClusterCommand::interPanHeader

Definition at line **401** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfDefaultAttributeValue Union Reference

[Application Framework types]

Type for default values. More...

```
#include <af-types.h>
```

Data Fields

int8u *	**ptrToDefaultValue**
int16u	**defaultValue**

Detailed Description

Type for default values.

Default value is either a value itself, if it is 2 bytes or less, or a pointer to the value itself, if attribute type is longer than 2 bytes.

Definition at line **73** of file **af-types.h**.

Field Documentation

int8u* EmberAfDefaultAttributeValue::ptrToDefaultValue

Points to data if size is more than 2 bytes. If size is more than 2 bytes, and this value is NULL, then the default value is all zeroes.

Definition at line **79** of file **af-types.h**.

int16u EmberAfDefaultAttributeValue::defaultValue

Actual default value if the attribute size is 2 bytes or less.

Definition at line **83** of file **af-types.h**.

The documentation for this union was generated from the following file:

- **af-types.h**

EmberAfDefaultOrMinMaxAttributeValue Union Reference

[Application Framework types]

Union describing the attribute default/min/max values. More...

```
#include <af-types.h>
```

Data Fields

int8u *	**ptrToDefaultValue**
int16u	**defaultValue**
EmberAfAttributeMinMaxValue *	**ptrToMinMaxValue**

Detailed Description

Union describing the attribute default/min/max values.

Definition at line **110** of file **af-types.h**.

Field Documentation

int8u* EmberAfDefaultOrMinMaxAttributeValue::ptrToDefaultValue

Points to data if size is more than 2 bytes. If size is more than 2 bytes, and this value is NULL, then the default value is all zeroes.

Definition at line **116** of file **af-types.h**.

int16u EmberAfDefaultOrMinMaxAttributeValue::defaultValue

Actual default value if the attribute size is 2 bytes or less.

Definition at line **120** of file **af-types.h**.

EmberAfAttributeMinMaxValue* EmberAfDefaultOrMinMaxAttributeValue::ptrToMinMaxValue

Points to the min max attribute value structure, if min/max is supported for this attribute.

Definition at line **125** of file **af-types.h**.

The documentation for this union was generated from the following file:

- **af-types.h**

EmberAfDefinedEndpoint Struct Reference

[Application Framework types]

Struct that maps actual endpoint type, onto a specific endpoint. More...

```
#include <af-types.h>
```

Data Fields

int8u	**endpoint**
EmberAfProfileId	**profileId**
int16u	**deviceId**
int8u	**deviceVersion**
EmberAfEndpointType *	**endpointType**

Detailed Description

Struct that maps actual endpoint type, onto a specific endpoint.

Definition at line **426** of file **af-types.h**.

Field Documentation

int8u EmberAfDefinedEndpoint::endpoint

Actual ZigBee endpoint number.

Definition at line **430** of file **af-types.h**.

EmberAfProfileId EmberAfDefinedEndpoint::profileId

Profile ID of the device on this endpoint.

Definition at line **434** of file **af-types.h**.

int16u EmberAfDefinedEndpoint::deviceId

Device ID of the device on this endpoint.

Definition at line **438** of file **af-types.h**.

int8u EmberAfDefinedEndpoint::deviceVersion

Version of the device.

Definition at line **442** of file **af-types.h**.

EmberAfEndpointType* EmberAfDefinedEndpoint::endpointType

Endpoint type for this endpoint.

Definition at line **446** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfEndpointList Struct Reference

A list of endpoints received during a service discovery attempt. This will be returned for a match descriptor request and a active endpoint request. More...

```
#include <af-types.h>
```

Data Fields

int8u	**count**
const int8u *	**list**

Detailed Description

A list of endpoints received during a service discovery attempt. This will be returned for a match descriptor request and a active endpoint request.

Definition at line **554** of file **af-types.h**.

Field Documentation

int8u EmberAfEndpointList::count

Definition at line **555** of file **af-types.h**.

const int8u* EmberAfEndpointList::list

Definition at line **556** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfEndpointType Struct Reference

[Application Framework types]

Endpoint type struct describes clusters that are on the endpoint. More...

```
#include <af-types.h>
```

Data Fields

EmberAfCluster *	cluster
int8u	clusterCount
int16u	endpointSize

Detailed Description

Endpoint type struct describes clusters that are on the endpoint.

Definition at line 407 of file af-types.h.

Field Documentation

EmberAfCluster* EmberAfEndpointType::cluster

Pointer to the cluster structs, describing clusters on this endpoint type.

Definition at line 412 of file af-types.h.

int8u EmberAfEndpointType::clusterCount

Number of clusters in this endpoint type.

Definition at line 416 of file af-types.h.

int16u EmberAfEndpointType::endpointSize

Size of all non-external, non-singlet attribute in this endpoint type.

Definition at line **420** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfEventContext Struct Reference

a structure used to keep track of cluster related events and their sleep control values. The cluster code will not know at runtime all of the events that it has access to in the event table This structure is stored by the application framework in an event context table which allong with helper functions allows the cluster code to schedule and deactivate its associated events. More...

```
#include <af-types.h>
```

Data Fields

int8u	**endpoint**
int16u	**clusterId**
boolean	**isClient**
EmberAfEventSleepControl	**sleepControl**
EmberEventControl *	**eventControl**

Detailed Description

a structure used to keep track of cluster related events and their sleep control values. The cluster code will not know at runtime all of the events that it has access to in the event table This structure is stored by the application framework in an event context table which allong with helper functions allows the cluster code to schedule and deactivate its associated events.

Definition at line **607** of file **af-types.h**.

Field Documentation

int8u EmberAfEventContext::endpoint

The endpoint of the associated cluster event.

Definition at line **611** of file **af-types.h**.

int16u EmberAfEventContext::clusterId

The cluster id of the associated cluster event.

Definition at line **615** of file **af-types.h**.

boolean EmberAfEventContext::isClient

The server/client identity of the associated cluster event.

Definition at line **619** of file **af-types.h**.

EmberAfEventSleepControl EmberAfEventContext::sleepControl

A sleep control value used to control the processor's sleep behavior while the event is active..

Definition at line **624** of file **af-types.h**.

EmberEventControl* EmberAfEventContext::eventControl

A pointer to the event control value which is stored in the event table and is used to actually schedule the event.

Definition at line **629** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfIasZoneTableEntry Struct Reference

```
#include <ias-zone-table.h>
```

Data Fields

int16u **zoneType**
int8u **ieeeAdd** [8]

Detailed Description

Definition at line **26** of file **ias-zone-table.h**.

Field Documentation

int16u EmberAfIasZoneTableEntry::zoneType

Definition at line **27** of file **ias-zone-table.h**.

int8u EmberAfIasZoneTableEntry::ieeeAdd[8]

Definition at line **28** of file **ias-zone-table.h**.

The documentation for this struct was generated from the following file:

- **ias-zone-table.h**

EmberAfIncomingMessage Struct Reference

[Application Framework types]

a struct containing the superset of values passed to both emberIncomingMessageHandler on the SOC and ezspIncomingMessageHandler on the host. More...

```
#include <af-types.h>
```

Data Fields

EmberIncomingMessageType	**type**
EmberApsFrame *	**apsFrame**
int8u *	**message**
int16u	**msgLen**
int16u	**source**
int8u	**lastHopLqi**
int8s	**lastHopRssi**
int8u	**bindingTableIndex**
int8u	**addressTableIndex**

Detailed Description

a struct containing the superset of values passed to both emberIncomingMessageHandler on the SOC and ezspIncomingMessageHandler on the host.

Definition at line **255** of file **af-types.h**.

Field Documentation

EmberIncomingMessageType EmberAfIncomingMessage::type

The type of the incoming message

Definition at line **259** of file **af-types.h**.

EmberApsFrame* EmberAfIncomingMessage::apsFrame

Aps frame for the incoming message

Definition at line **263** of file **af-types.h**.

int8u* EmberAfIncomingMessage::message

The message copied into a flat buffer

Definition at line **267** of file **af-types.h**.

int16u EmberAfIncomingMessage::msgLen

Length of the incoming message

Definition at line **271** of file **af-types.h**.

int16u EmberAfIncomingMessage::source

Two byte node id of the sending node.

Definition at line **275** of file **af-types.h**.

int8u EmberAfIncomingMessage::lastHopLqi

Link quality from the node that last relayed the message.

Definition at line **280** of file **af-types.h**.

int8s EmberAfIncomingMessage::lastHopRssi

The energy level (in units of dBm) observed during the reception.

Definition at line **284** of file **af-types.h**.

int8u EmberAfIncomingMessage::bindingTableIndex

The index of a binding that matches the message or 0xFF if there is no matching binding.

Definition at line **289** of file **af-types.h**.

int8u EmberAfIncomingMessage::addressTableIndex

The index of the entry in the address table that matches the sender of the message or 0xFF if there is no matching entry.

Definition at line **295** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfInterpanHeader Struct Reference

[Application Framework types]

Interpan header used for sending and receiving interpan messages. More...

```
#include <af-types.h>
```

Data Fields

EmberAfInterpanMessageType	messageType
EmberEUI64	longAddress
EmberNodeId	shortAddress
EmberPanId	panId
EmberAfProfileId	profileId
EmberAfClusterId	clusterId
EmberMulticastId	groupId
EmberAfInterpanOptions	options

Detailed Description

Interpan header used for sending and receiving interpan messages.

Definition at line 326 of file af-types.h.

Field Documentation

EmberAfInterpanMessageType EmberAfInterpanHeader::messageType

Definition at line 327 of file af-types.h.

EmberEUI64 EmberAfInterpanHeader::longAddress

MAC addressing For outgoing messages this is the destination. For incoming messages it is the source, which always has a long address.

Definition at line **334** of file **af-types.h**.

EmberNodeId EmberAfInterpanHeader::shortAddress

Definition at line **335** of file **af-types.h**.

EmberPanId EmberAfInterpanHeader::panId

Definition at line **336** of file **af-types.h**.

EmberAfProfileId EmberAfInterpanHeader::profileId

APS data

Definition at line **341** of file **af-types.h**.

EmberAfClusterId EmberAfInterpanHeader::clusterId

Definition at line **342** of file **af-types.h**.

EmberMulticastId EmberAfInterpanHeader::groupId

The groupId is only used for EMBER_AF_INTERPAN_MULTICAST

Definition at line **347** of file **af-types.h**.

EmberAfInterpanOptions EmberAfInterpanHeader::options

Definition at line **348** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfLoadControlEvent Struct Reference

[Application Framework types]

Struct that describes a load control event. More...

```
#include <af-types.h>
```

Data Fields

int32u	**eventId**
EmberEUI64	**source**
int8u	**sourceEndpoint**
int8u	**destinationEndpoint**
int16u	**deviceClass**
int8u	**utilityEnrollmentGroup**
int32u	**startTime**
int16u	**duration**
int8u	**criticalityLevel**
int8u	**coolingTempOffset**
int8u	**heatingTempOffset**
int16s	**coolingTempSetPoint**
int16s	**heatingTempSetPoint**
int8s	**avgLoadPercentage**
int8u	**dutyCycle**
int8u	**eventControl**
int16u	**startRand**
int16u	**endRand**
int8u	**optionControl**

Detailed Description

Struct that describes a load control event.

This is used in the load control event callback and within the demand response load control cluster code.

Definition at line **458** of file **af-types.h**.

Field Documentation

int32u EmberAfLoadControlEvent::eventId

Definition at line **459** of file **af-types.h**.

EmberEUI64 EmberAfLoadControlEvent::source

Definition at line **460** of file **af-types.h**.

int8u EmberAfLoadControlEvent::sourceEndpoint

Definition at line **461** of file **af-types.h**.

int8u EmberAfLoadControlEvent::destinationEndpoint

Definition at line **462** of file **af-types.h**.

int16u EmberAfLoadControlEvent::deviceClass

Definition at line **463** of file **af-types.h**.

int8u EmberAfLoadControlEvent::utilityEnrollmentGroup

Definition at line **464** of file **af-types.h**.

int32u EmberAfLoadControlEvent::startTime

Start time in seconds

Definition at line **468** of file **af-types.h**.

int16u EmberAfLoadControlEvent::duration

Duration in minutes

Definition at line **472** of file **af-types.h**.

int8u EmberAfLoadControlEvent::criticalityLevel

Definition at line **473** of file **af-types.h**.

int8u EmberAfLoadControlEvent::coolingTempOffset

Definition at line **474** of file **af-types.h**.

int8u EmberAfLoadControlEvent::heatingTempOffset

Definition at line **475** of file **af-types.h**.

int16s EmberAfLoadControlEvent::coolingTempSetPoint

Definition at line **476** of file **af-types.h**.

int16s EmberAfLoadControlEvent::heatingTempSetPoint

Definition at line **477** of file **af-types.h**.

int8s EmberAfLoadControlEvent::avgLoadPercentage

Definition at line **478** of file **af-types.h**.

int8u EmberAfLoadControlEvent::dutyCycle

Definition at line **479** of file **af-types.h**.

int8u EmberAfLoadControlEvent::eventControl

Definition at line **480** of file **af-types.h**.

int16u EmberAfLoadControlEvent::startRand

Definition at line **481** of file **af-types.h**.

int16u EmberAfLoadControlEvent::endRand

Definition at line **482** of file **af-types.h**.

int8u EmberAfLoadControlEvent::optionControl

Definition at line **483** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfManufacturerCodeEntry Struct Reference

[Application Framework types]

```
#include <af-types.h>
```

Data Fields

int16u	**index**
int16u	**manufacturerCode**

Detailed Description

A struct used to construct a table of manufacturer codes for manufacturer specfic attributes and clusters.

Definition at line **245** of file **af-types.h**.

Field Documentation

int16u EmberAfManufacturerCodeEntry::index

Definition at line **246** of file **af-types.h**.

int16u EmberAfManufacturerCodeEntry::manufacturerCode

Definition at line **247** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfMessagingServerMessage Struct Reference

```
#include <messaging-server.h>
```

Data Fields

boolean	**valid**
int32u	**messageId**
int8u	**messageControl**
int32u	**startTime**
int16u	**durationInMinutes**
int8u	**message** [EMBER_AF_PLUGIN_MESSAGING_SERVER_MESSAGE_SIZE+1]

Detailed Description

Definition at line **20** of file **messaging-server.h**.

Field Documentation

boolean EmberAfMessagingServerMessage::valid

Definition at line **21** of file **messaging-server.h**.

int32u EmberAfMessagingServerMessage::messageId

Definition at line **22** of file **messaging-server.h**.

int8u EmberAfMessagingServerMessage::messageControl

Definition at line **23** of file **messaging-server.h**.

int32u EmberAfMessagingServerMessage::startTime

Definition at line **24** of file **messaging-server.h**.

int16u EmberAfMessagingServerMessage::durationInMinutes

Definition at line **25** of file **messaging-server.h**.

int8u EmberAfMessagingServerMessage::message[EMBER_AF_PLUGIN_MESSAGING_SERVER_MESSAGE_SIZE+1]

Definition at line **26** of file **messaging-server.h**.

The documentation for this struct was generated from the following file:

- **messaging-server.h**

EmberAfOtaHeader Struct Reference

This structure is an in-memory representation of the Over-the-air header data that resides on disk. It is not a byte-for-byte copy. More...

```
#include <af-types.h>
```

Data Fields

int16u	**headerVersion**
int16u	**headerLength**
int16u	**fieldControl**
int16u	**manufacturerId**
int16u	**imageTypeId**
int32u	**firmwareVersion**
int16u	**zigbeeStackVersion**
int8u	**headerString** [EMBER_AF_OTA_MAX_HEADER_STRING_LENGTH+1]
int32u	**imageSize**
int8u	**securityCredentials**
int8u	**upgradeFileDestination** [EUI64_SIZE]
int16u	**minimumHardwareVersion**
int16u	**maximumHardwareVersion**

Detailed Description

This structure is an in-memory representation of the Over-the-air header data that resides on disk. It is not a byte-for-byte copy.

Definition at line **915** of file **af-types.h**.

Field Documentation

int16u EmberAfOtaHeader::headerVersion

Definition at line **917** of file **af-types.h**.

int16u EmberAfOtaHeader::headerLength

Definition at line **918** of file **af-types.h**.

int16u EmberAfOtaHeader::fieldControl

Definition at line **919** of file **af-types.h**.

int16u EmberAfOtaHeader::manufacturerId

Definition at line **920** of file **af-types.h**.

int16u EmberAfOtaHeader::imageTypeId

Definition at line **921** of file **af-types.h**.

int32u EmberAfOtaHeader::firmwareVersion

Definition at line **922** of file **af-types.h**.

int16u EmberAfOtaHeader::zigbeeStackVersion

Definition at line **923** of file **af-types.h**.

int8u EmberAfOtaHeader::headerString[EMBER_AF_OTA_MAX_HEADER_STRING_LENGTH+1]

The spec. does NOT require that the string be NULL terminated in the header stored on disk. Therefore we make sure we can support a 32-character string without a NULL terminator by adding +1 in the data structure.

Definition at line **931** of file **af-types.h**.

int32u EmberAfOtaHeader::imageSize

When reading the header this will be the complete length of the file. When writing the header, this must be set to the length of the MFG image data portion including all tags.

Definition at line **938** of file **af-types.h**.

int8u EmberAfOtaHeader::securityCredentials

The remaining four fields are optional. The field control should be checked to determine if their values are valid.

Definition at line **944** of file **af-types.h**.

int8u EmberAfOtaHeader::upgradeFileDestination[EUI64_SIZE]

Definition at line **945** of file **af-types.h**.

int16u EmberAfOtaHeader::minimumHardwareVersion

Definition at line **946** of file **af-types.h**.

int16u EmberAfOtaHeader::maximumHardwareVersion

Definition at line **947** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfOtaImageId Struct Reference

This is a unique identifier for referencing ZigBee Over-the-air upgrade images. It is used by the OTA plugins when passing around information about an upgrade file. More...

```
#include <af-types.h>
```

Data Fields

int16u	**manufacturerId**
int16u	**imageTypeId**
int32u	**firmwareVersion**
int8u	**deviceSpecificFileEui64** [EUI64_SIZE]

Detailed Description

This is a unique identifier for referencing ZigBee Over-the-air upgrade images. It is used by the OTA plugins when passing around information about an upgrade file.

Definition at line **879** of file **af-types.h**.

Field Documentation

int16u EmberAfOtaImageId::manufacturerId

Definition at line **880** of file **af-types.h**.

int16u EmberAfOtaImageId::imageTypeId

Definition at line **881** of file **af-types.h**.

int32u EmberAfOtaImageId::firmwareVersion

Definition at line **882** of file **af-types.h**.

int8u EmberAfOtaImageId::deviceSpecificFileEui64[EUI64_SIZE]

This is only used for device specific files. It will be set to all 0's when the image does not have an upgrade destination field in it. Little endian format.

Definition at line **890** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfReportEntry Struct Reference

A structure used to reporting configurations. More...

```
#include <af-types.h>
```

Data Fields

boolean	**inUse**
EmberAfReportingDirection	**direction**
EmberNodeId	**nodeId**
int8u	**sourceEndpoint**
int8u	**destinationEndpoint**
EmberAfClusterId	**clusterId**
EmberAfAttributeId	**attributeId**
int8u	**mask**
int16u	**manufacturerCode**
int16u	**minInterval**
int16u	**maxInterval**
int16u	**currentInterval**
int32u	**reportableChange**
int32u	**lastReportValue**
int16u	**timeout**
int8u	**failCount**

Detailed Description

A structure used to reporting configurations.

Definition at line **635** of file **af-types.h**.

Field Documentation

boolean EmberAfReportEntry::inUse

TRUE if the entry is in use; otherwise, FALSE.

Definition at line **639** of file **af-types.h**.

EmberAfReportingDirection EmberAfReportEntry::direction

EMBER_ZCL_REPORTING_DIRECTION_REPORTED for reports sent from the local device or EMBER_ZCL_REPORTING_DIRECTION_RECEIVED for reports received from a remote device.

Definition at line **645** of file **af-types.h**.

EmberNodeId EmberAfReportEntry::nodeId

The node id of the remote device. For reported attributes, this is the node that will receive the report from us. For received attributes, this is the node that will send the report to us.

Definition at line **651** of file **af-types.h**.

int8u EmberAfReportEntry::sourceEndpoint

The endpoint on which the reported attribute is located. For reported attributes, this is the endpoint on the local device. For received attributes, it is the endpoint on the remote device.

Definition at line **657** of file **af-types.h**.

int8u EmberAfReportEntry::destinationEndpoint

The endpoint to which the report attribute is sent. For reported attributes, this is the endpoint on the remote device. For received attributes, it is the endpoint on the local device.

Definition at line **663** of file **af-types.h**.

EmberAfClusterId EmberAfReportEntry::clusterId

The cluster where the attribute is located.

Definition at line **667** of file **af-types.h**.

EmberAfAttributeId EmberAfReportEntry::attributeId

The id of the attribute being reported or received.

Definition at line **671** of file **af-types.h**.

int8u EmberAfReportEntry::mask

CLUSTER_MASK_SERVER for server-side attributes or CLUSTER_MASK_CLIENT for client-side attributes.

Definition at line **676** of file **af-types.h**.

int16u EmberAfReportEntry::manufacturerCode

Manufacturer code associated with the cluster and/or attribute. If the cluster id is inside the manufacturer-specific range, this value indicates the manufacturer code for the cluster. Otherwise, if this value is non- zero and the cluster id is a standard ZCL cluster, it indicates the manufacturer code for attribute.

Definition at line **684** of file **af-types.h**.

int16u EmberAfReportEntry::minInterval

The minimum reporting interval, measured in seconds. This field is ignored for received reports.

Definition at line **689** of file **af-types.h**.

int16u EmberAfReportEntry::maxInterval

The maximum reporting interval, measured in seconds. This field is ignored for received reports.

Definition at line **694** of file **af-types.h**.

int16u EmberAfReportEntry::currentInterval

The elapsed interval in seconds since the last report. This field is ignored for received reports.

Definition at line **699** of file **af-types.h**.

int32u EmberAfReportEntry::reportableChange

The minimum change to the attribute that will result in a report being sent. This field is ignored for received reports and for discrete data types.

Definition at line **705** of file **af-types.h**.

int32u EmberAfReportEntry::lastReportValue

The value of the attribute that was sent in the last report. This field is ignored for received reports.

Definition at line **710** of file **af-types.h**.

int16u EmberAfReportEntry::timeout

The maximum expected time between reports. This field is ignored for reported reports.

Definition at line **715** of file **af-types.h**.

int8u EmberAfReportEntry::failCount

The number of times that reporting this attribute has failed. This field is ignored for received reports.

Definition at line **720** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfSceneTableEntry Struct Reference

A structure used to store scene table entries in RAM or in tokens, depending on a plugin setting. If endpoint field is **EMBER_AF_SCENE_TABLE_UNUSED_ENDPOINT_ID**, the entry is unused. More...

```
#include <af-types.h>
```

Data Fields

int8u	**endpoint**
int16u	**groupId**
int8u	**sceneId**
int16u	**transitionTime**

Detailed Description

A structure used to store scene table entries in RAM or in tokens, depending on a plugin setting. If endpoint field is **EMBER_AF_SCENE_TABLE_UNUSED_ENDPOINT_ID**, the entry is unused.

Definition at line **774** of file **af-types.h**.

Field Documentation

int8u EmberAfSceneTableEntry::endpoint

Definition at line **775** of file **af-types.h**.

int16u EmberAfSceneTableEntry::groupId

Definition at line **776** of file **af-types.h**.

int8u EmberAfSceneTableEntry::sceneId

Definition at line **777** of file **af-types.h**.

int16u EmberAfSceneTableEntry::transitionTime

Definition at line **781** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfScheduledPrice Struct Reference

```
#include <price-server.h>
```

Data Fields

int32u	**providerId**
int8u	**rateLabel** [ZCL_PRICE_CLUSTER_MAXIMUM_RATE_LABEL_LENGTH+1]
int32u	**issuerEventID**
int8u	**unitOfMeasure**
int16u	**currency**
int8u	**priceTrailingDigitAndTier**
int8u	**numberOfPriceTiersAndTier**
int32u	**startTime**
int16u	**duration**
int32u	**price**
int8u	**priceRatio**
int32u	**generationPrice**
int8u	**generationPriceRatio**
int32u	**alternateCostDelivered**
int8u	**alternateCostUnit**
int8u	**alternateCostTrailingDigit**
int8u	**numberOfBlockThresholds**
int8u	**priceControl**
boolean	**valid**

Detailed Description

Definition at line **20** of file **price-server.h**.

Field Documentation

int32u EmberAfScheduledPrice::providerId

Definition at line **21** of file **price-server.h**.

int8u EmberAfScheduledPrice::rateLabel[ZCL_PRICE_CLUSTER_MAXIMUM_RATE_LABEL_LENGTH+1]

Definition at line **22** of file **price-server.h**.

int32u EmberAfScheduledPrice::issuerEventID

Definition at line **23** of file **price-server.h**.

int8u EmberAfScheduledPrice::unitOfMeasure

Definition at line **24** of file **price-server.h**.

int16u EmberAfScheduledPrice::currency

Definition at line **25** of file **price-server.h**.

int8u EmberAfScheduledPrice::priceTrailingDigitAndTier

Definition at line **26** of file **price-server.h**.

int8u EmberAfScheduledPrice::numberOfPriceTiersAndTier

Definition at line **27** of file **price-server.h**.

int32u EmberAfScheduledPrice::startTime

Definition at line **28** of file **price-server.h**.

int16u EmberAfScheduledPrice::duration

Definition at line **29** of file **price-server.h**.

int32u EmberAfScheduledPrice::price

Definition at line **30** of file **price-server.h**.

int8u EmberAfScheduledPrice::priceRatio

Definition at line **31** of file **price-server.h**.

int32u EmberAfScheduledPrice::generationPrice

Definition at line **32** of file **price-server.h**.

int8u EmberAfScheduledPrice::generationPriceRatio

Definition at line **33** of file **price-server.h**.

int32u EmberAfScheduledPrice::alternateCostDelivered

Definition at line **34** of file **price-server.h**.

int8u EmberAfScheduledPrice::alternateCostUnit

Definition at line 35 of file price-server.h.

int8u EmberAfScheduledPrice::alternateCostTrailingDigit

Definition at line 36 of file price-server.h.

int8u EmberAfScheduledPrice::numberOfBlockThresholds

Definition at line 37 of file price-server.h.

int8u EmberAfScheduledPrice::priceControl

Definition at line 38 of file price-server.h.

boolean EmberAfScheduledPrice::valid

Definition at line 39 of file price-server.h.

The documentation for this struct was generated from the following file:

- price-server.h

EmberAfServiceDiscoveryResult Struct Reference

A structure containing general information about the service discovery. More...

```
#include <af-types.h>
```

Data Fields

EmberAfServiceDiscoveryStatus	**status**
int16u	**zdoRequestClusterId**
EmberNodeId	**matchAddress**
const void *	**responseData**

Detailed Description

A structure containing general information about the service discovery.

Definition at line **515** of file **af-types.h**.

Field Documentation

EmberAfServiceDiscoveryStatus EmberAfServiceDiscoveryResult::status

The status indicates both the type of request (broadcast or unicast) and whether a response has been received.

Definition at line **520** of file **af-types.h**.

int16u EmberAfServiceDiscoveryResult::zdoRequestClusterId

This indicates what ZDO request cluster was associated with the request. It is helpful for a callback that may be used for multiple ZDO request types to determine the type of data returned. This will be based on the ZDO cluster values defined in ember-types.h.

Definition at line **528** of file **af-types.h**.

EmberNodeId EmberAfServiceDiscoveryResult::matchAddress

This is the address of the device that matched the request, which may be different than the device that *actually* is responding. This occurs when parents respond on behalf of their children.

Definition at line **535** of file **af-types.h**.

const void* EmberAfServiceDiscoveryResult::responseData

Only if the status code indicates a response will this data be non-NULL. When there is data, the type is according to the ZDO cluster ID sent out. For NETWORK_ADDRESS_REQUEST or IEEE_ADDRESS_REQUEST, the long ID will be contained in the responseData, so it will be a value of type EmberEUI64. The short ID will be in the matchAddress parameter field. For the MATCH_DESCRIPTORS_REQUEST the responseData will point to an **EmberAfEndpointList** structure.

Definition at line **546** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

EmberAfTagData Struct Reference

This structure contains information about a tag that resides within an Over-the-air bootload file. More...

```
#include <af-types.h>
```

Data Fields

int16u **id**
int32u **length**

Detailed Description

This structure contains information about a tag that resides within an Over-the-air bootload file.

Definition at line **954** of file **af-types.h**.

Field Documentation

int16u EmberAfTagData::id

Definition at line **955** of file **af-types.h**.

int32u EmberAfTagData::length

Definition at line **956** of file **af-types.h**.

The documentation for this struct was generated from the following file:

- **af-types.h**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- a -

- address : **_NodeInformation**
- addressTableIndex : **EmberAfIncomingMessage**
- alternateCostDelivered : **EmberAfScheduledPrice**
- alternateCostTrailingDigit : **EmberAfScheduledPrice**
- alternateCostUnit : **EmberAfScheduledPrice**
- apsFrame : **EmberAfIncomingMessage** , **EmberAfClusterCommand**
- attributeCount : **EmberAfCluster**
- attributeId : **_ConfigureReportingStatusRecord** , **_ReadReportingConfigurationRecord** , **EmberAfAttributeMetadata** , **_ReadReportingConfigurationAttributeRecord** , **_ReportAttributeRecord** , **EmberAfAttributeSearchRecord** , **_DiscoverAttributesInfoRecord** , **EmberAfReportEntry** , **_ReadAttributeStatusRecord** , **_WriteAttributeRecord** , **_WriteAttributeStatusRecord** , **_ConfigureReportingRecord**

- attributeLocation : **_ReadAttributeStatusRecord** , **_ReportAttributeRecord** , **_WriteAttributeRecord** , **_SceneExtensionAttributeInfo**
- attributes : **EmberAfCluster**
- attributeType : **_ReadAttributeStatusRecord** , **_SceneExtensionAttributeInfo** , **_ConfigureReportingRecord** , **_ReportAttributeRecord** , **_DiscoverAttributesInfoRecord** , **_ReadReportingConfigurationRecord** , **_WriteAttributeRecord** , **EmberAfAttributeMetadata**
- avgLoadPercentage : **EmberAfLoadControlEvent**

File List

Here is a list of all files with brief descriptions:

_AF_API.top [code]	
af-structs.h [code]	
af-types.h [code]	The include file for all the types for Ember ApplicationFramework
af.h [code]	The master include file for the Ember ApplicationFramework API
attribute-id.h [code]	
attribute-size.h [code]	
attribute-storage.h [code]	
attribute-type.h [code]	
bacnet-protocol-tunnel.h [code]	
basic-callback.h [code]	
basic.h [code]	
call-command-handler.h [code]	
callback.h [code]	
client-api.h [code]	API for generating command buffer
client-command-macro.h [code]	
cluster-id.h [code]	
color-control.h [code]	
command-id.h [code]	
core-cli.h [code]	
custom-cli.h [code]	
debug-printing-test.h [code]	
debug-printing.h [code]	
debug.h [code]	
demand-response-load-control.h [code]	
drlc-callback.h [code]	
drlc-server.h [code]	
enums.h [code]	

generic-tunnel.h [code]	
groups-server.h [code]	
ias-ace-server.h [code]	
ias-wd-server.h [code]	
ias-zone-table.h [code]	
identify.h [code]	
interpan.h [code]	Plugin for receiving InterPAN messages. See message for documentation
key-establishment-storage.h [code]	
key-establishment.h [code]	
level-control.h [code]	
load-control-event-table.h [code]	
messaging-client-callback.h [code]	
messaging-client.h [code]	
messaging-server.h [code]	
network-cli.h [code]	
on-off.h [code]	
ota-bootload-ncp.h [code]	
ota-bootload-xmodem.h [code]	
ota-cli.h [code]	
ota-client-page-request-test.h [code]	
ota-client-page-request.h [code]	
ota-client-policy.h [code]	
ota-client-signature-verify.h [code]	
ota-client.h [code]	
ota-server-policy.h [code]	
ota-server.h [code]	
ota-static-file-data.h [code]	
ota-storage-linux.h [code]	
ota-storage-simple-custom.h [code]	
ota-storage-simple-driver.h [code]	
ota-storage.h [code]	
ota.h [code]	

partner-link-key-exchange.h [code]	
plugin-cli.h [code]	
price-client-callback.h [code]	
price-client.h [code]	
price-server.h [code]	
print-cluster.h [code]	
psl-commissioning-callback.h [code]	
psl-commissioning.h [code]	
scenes.h [code]	
security-cli.h [code]	
simple-metering-client-callback.h [code]	
simple-metering-test.h [code]	
smart-energy-registration.h [code]	
stack-handlers.h [code]	
temperature-measurement.h [code]	
test-callback.h [code]	
thermostat.h [code]	
time-client.h [code]	
time-server.h [code]	
trust-center-keepalive.h [code]	
tunneling-client-callback.h [code]	
tunneling-client.h [code]	
tunneling-server-callback.h [code]	
tunneling-server.h [code]	
zcl-cli.h [code]	
zdo-cli.h [code]	

_AF_API.top File Reference

Go to the source code of this file.

---

af-structs.h File Reference

Go to the source code of this file.

Data Structures

struct	_ReadAttributeStatusRecord
struct	_WriteAttributeRecord
struct	_WriteAttributeStatusRecord
struct	_ConfigureReportingRecord
struct	_ConfigureReportingStatusRecord
struct	_ReadReportingConfigurationRecord
struct	_ReadReportingConfigurationAttributeRecord
struct	_ReportAttributeRecord
struct	_DiscoverAttributesInfoRecord
struct	_SceneExtensionAttributeInfo
struct	_SceneExtensionFieldSet
struct	_BlockThreshold
struct	_Notification
struct	_NeighborInfo
struct	_ChatParticipant
struct	_ChatRoom
struct	_NodeInformation

Typedefs

typedef struct _ReadAttributeStatusRecord	ReadAttributeStatusRecord
typedef struct _WriteAttributeRecord	WriteAttributeRecord
typedef struct _WriteAttributeStatusRecord	WriteAttributeStatusRecord
typedef struct _ConfigureReportingRecord	ConfigureReportingRecord
typedef struct _ConfigureReportingStatusRecord	ConfigureReportingStatusRecord
typedef struct _ReadReportingConfigurationRecord	ReadReportingConfigurationRecord
typedef struct _ReadReportingConfigurationAttributeRecord	ReadReportingConfigurationAttributeRecord
typedef struct _ReportAttributeRecord	ReportAttributeRecord
typedef struct _DiscoverAttributesInfoRecord	DiscoverAttributesInfoRecord
typedef struct _SceneExtensionAttributeInfo	SceneExtensionAttributeInfo
typedef struct _SceneExtensionFieldSet	SceneExtensionFieldSet
typedef struct _BlockThreshold	BlockThreshold
typedef struct _Notification	Notification
typedef struct _NeighborInfo	NeighborInfo
typedef struct _ChatParticipant	ChatParticipant
typedef struct _ChatRoom	ChatRoom

typedef struct _NodeInformation	NodeInformation
typedef int8u	Identity
typedef int8u	EphemeralData
typedef int8u	Smac
typedef int8u	Signature

Typedef Documentation

typedef struct _ReadAttributeStatusRecord ReadAttributeStatusRecord

typedef struct _WriteAttributeRecord WriteAttributeRecord

typedef struct _WriteAttributeStatusRecord WriteAttributeStatusRecord

typedef struct _ConfigureReportingRecord ConfigureReportingRecord

typedef struct _ConfigureReportingStatusRecord ConfigureReportingStatusRecord

typedef struct _ReadReportingConfigurationRecord ReadReportingConfigurationRecord

typedef struct _ReadReportingConfigurationAttributeRecord ReadReportingConfigurationAttributeRecord

typedef struct _ReportAttributeRecord ReportAttributeRecord

typedef struct _DiscoverAttributesInfoRecord DiscoverAttributesInfoRecord

typedef struct _SceneExtensionAttributeInfo SceneExtensionAttributeInfo

typedef struct _SceneExtensionFieldSet SceneExtensionFieldSet

typedef struct _BlockThreshold BlockThreshold

typedef struct _Notification Notification

typedef struct _NeighborInfo NeighborInfo

typedef struct _ChatParticipant ChatParticipant

typedef struct _ChatRoom ChatRoom

typedef struct _NodeInformation NodeInformation

typedef int8u Identity

Definition at line **163** of file **af-structs.h**.

typedef int8u EphemeralData

Definition at line **168** of file **af-structs.h**.

typedef int8u Smac

Definition at line **173** of file **af-structs.h**.

typedef int8u Signature

Definition at line **178** of file **af-structs.h**.

af-types.h File Reference

The include file for all the types for Ember ApplicationFramework. More...

```
#include "app/framework/gen/enums.h"
```

Go to the source code of this file.

Data Structures

union	**EmberAfDefaultAttributeValue** Type for default values. More...
struct	**EmberAfAttributeMinMaxValue** Type describing the attribute default, min and max values. More...
union	**EmberAfDefaultOrMinMaxAttributeValue** Union describing the attribute default/min/max values. More...
struct	**EmberAfAttributeMetadata** Each attribute has it's metadata stored in such struct. More...
struct	**EmberAfCluster** Struct describing cluster. More...
struct	**EmberAfAttributeSearchRecord** Struct used to find an attribute in storage. Together the elements in this search record constitute the "primary key" used to identify a unique attribute value in attribute storage. More...
struct	**EmberAfManufacturerCodeEntry**
struct	**EmberAfIncomingMessage** a struct containing the superset of values passed to both emberIncomingMessageHandler on the SOC and ezspIncomingMessageHandler on the host. More...
struct	**EmberAfInterpanHeader** Interpan header used for sending and receiving interpan messages. More...
struct	**EmberAfAllowedInterPanMessage** This structure is used define an interpan message that will be accepted by the interpan filters. More...
struct	**EmberAfClusterCommand** The EmberAFClusterCommand is a struct wrapper for all the data pertaining to a command which comes in over the air. This enables struct is used to encapsulate a command in a single place on the stack and pass a pointer to that location around during command processing. More...
struct	**EmberAfEndpointType** Endpoint type struct describes clusters that are on the endpoint. More...
struct	**EmberAfDefinedEndpoint** Struct that maps actual endpoint type, onto a specific endpoint. More...
struct	**EmberAfLoadControlEvent** Struct that describes a load control event. More...
struct	**EmberAfServiceDiscoveryResult** A structure containing general information about the service discovery. More...
struct	**EmberAfEndpointList** A list of endpoints received during a service discovery attempt. This will be returned for a match descriptor request and a active endpoint request. More...
struct	**EmberAfEventContext** a structure used to keep track of cluster related events and their sleep control values. The cluster code will not know at runtime all of the events that it has access to in the event table This structure is stored by the application framework in an event context table which allong with helper functions allows the cluster code to schedule and deactivate its associated events. More...
struct	**EmberAfReportEntry** A structure used to reporting configurations. More...
struct	**EmberAfSceneTableEntry** A structure used to store scene table entries in RAM or in tokens, depending on a plugin setting. If endpoint field is **EMBER_AF_SCENE_TABLE_UNUSED_ENDPOINT_ID**, the entry is unused. More...
struct	**EmberAfOtaImageId**

	This is a unique identifier for referencing ZigBee Over-the-air upgrade images. It is used by the OTA plugins when passing around information about an upgrade file. More...
struct	**EmberAfOtaHeader** This structure is an in-memory representation of the Over-the-air header data that resides on disk. It is not a byte-for-byte copy. More...
struct	**EmberAfTagData** This structure contains information about a tag that resides within an Over-the-air bootload file. More...

Defines

#define	**EMBER_AF_NULL_MANUFACTURER_CODE**
#define	**EMBER_AF_INTER_PAN_UNICAST**
#define	**EMBER_AF_INTER_PAN_BROADCAST**
#define	**EMBER_AF_INTER_PAN_MULTICAST**
#define	**INTER_PAN_UNICAST**
#define	**INTER_PAN_BROADCAST**
#define	**INTER_PAN_MULTICAST**
#define	**EMBER_AF_INTERPAN_OPTION_NONE**
#define	**EMBER_AF_INTERPAN_OPTION_APS_ENCRYPT**
#define	**EMBER_AF_INTERPAN_OPTION_MAC_HAS_LONG_ADDRESS**
#define	**InterPanHeader**
#define	**EMBER_AF_INTERPAN_DIRECTION_CLIENT_TO_SERVER**
#define	**EMBER_AF_INTERPAN_DIRECTION_SERVER_TO_CLIENT**
#define	**EMBER_AF_INTERPAN_DIRECTION_BOTH**
#define	**EMBER_AF_INTERPAN_GLOBAL_COMMAND**
#define	**EMBER_AF_INTERPAN_MANUFACTURER_SPECIFIC**
#define	**EM_AF_DISCOVERY_RESPONSE_MASK**
#define	**emberAfHaveDiscoveryResponseStatus**(status)
#define	**EMBER_AF_SCENE_TABLE_NULL_INDEX**
#define	**EMBER_AF_SCENE_TABLE_UNUSED_ENDPOINT_ID**
#define	**ZCL_SCENES_CLUSTER_MAXIMUM_NAME_LENGTH**
#define	**ZCL_SCENES_GLOBAL_SCENE_GROUP_ID**
#define	**ZCL_PRICE_CLUSTER_MAXIMUM_RATE_LABEL_LENGTH**
#define	**EMBER_AF_OTA_MAX_HEADER_STRING_LENGTH**
#define	**APP_NOTIFY_ERROR_CODE_START**
#define	**APP_NOTIFY_MESSAGE_TEXT**

Typedefs

typedef int16u	**EmberAfProfileId**
typedef int16u	**EmberAfAttributeId**
typedef int16u	**EmberAfClusterId**
typedef int8u	**EmberAfAttributeType**
typedef int8u	**EmberAfClusterMask**
typedef int8u	**EmberAfAttributeMask**
typedef void(*	**EmberAfGenericClusterFunction**)(void)
typedef int8u	**EmberAfInterpanMessageType**
typedef int16u	**EmberAfInterpanOptions**
typedef int8u	**EmberAfAllowedInterpanOptions**
typedef void(	**EmberAfServiceDiscoveryCallback**)(const **EmberAfServiceDiscoveryResult** *result)
typedef void(	**EmberAfPartnerLinkKeyExchangeCallback**)(boolean success)
typedef void(*	**EmberAfTickFunction**)(int8u endpoint)
typedef void(*	**EmberAfInitFunction**)(int8u endpoint)
typedef void(*	**EmberAfClusterAttributeChangedCallback**)(int8u endpoint, **EmberAfAttributeId** attributeId)
typedef void(*	**EmberAfManufacturerSpecificClusterAttributeChangedCallback**)(int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
typedef void(*	**EmberAfDefaultResponseFunction**)(int8u endpoint, int8u commandId, **EmberAfStatus** status)
typedef void(*	**EmberAfMessageSentFunction**)(EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Enumerations

```
enum EmberAfServiceDiscoveryStatus {
    EMBER_AF_BROADCAST_SERVICE_DISCOVERY_COMPLETE,
    EMBER_AF_BROADCAST_SERVICE_DISCOVERY_RESPONSE_RECEIVED,
    EMBER_AF_UNICAST_SERVICE_DISCOVERY_TIMEOUT,
    EMBER_AF_UNICAST_SERVICE_DISCOVERY_COMPLETE_WITH_RESPONSE
}
enum EmberAfEventSleepControl {
    EMBER_AF_OK_TO_HIBERNATE,
    EMBER_AF_OK_TO_NAP,
    EMBER_AF_STAY_AWAKE
}
enum EmberAfOtaStorageStatus {
    EMBER_AF_OTA_STORAGE_SUCCESS,
    EMBER_AF_OTA_STORAGE_ERROR,
    EMBER_AF_OTA_STORAGE_RETURN_DATA_TOO_LONG,
    EMBER_AF_OTA_STORAGE_PARTIAL_FILE_FOUND
}
enum EmberAfKeyEstablishmentNotifyMessage {
    NO_APP_MESSAGE,
    RECEIVED_PARTNER_CERTIFICATE,
    GENERATING_EPHEMERAL_KEYS,
    GENERATING_SHARED_SECRET,
    KEY_GENERATION_DONE,
    GENERATE_SHARED_SECRET_DONE,
    LINK_KEY_ESTABLISHED,
    NO_LOCAL_RESOURCES,
    PARTNER_NO_RESOURCES,
    TIMEOUT_OCCURRED,
    INVALID_APP_COMMAND,
    MESSAGE_SEND_FAILURE,
    PARTNER_SENT_TERMINATE,
    INVALID_PARTNER_MESSAGE,
    BAD_CERTIFICATE_ISSUER,
    KEY_CONFIRM_FAILURE,
    BAD_KEY_ESTABLISHMENT_SUITE,
    KEY_TABLE_FULL,
    NO_ESTABLISHMENT_ALLOWED
}
enum EmberAfImageVerifyStatus {
    EMBER_AF_IMAGE_GOOD,
    EMBER_AF_IMAGE_BAD,
    EMBER_AF_IMAGE_VERIFY_IN_PROGRESS
}
```

Detailed Description

The include file for all the types for Ember ApplicationFramework.

Definition in file **af-types.h**.

Define Documentation

#define EM_AF_DISCOVERY_RESPONSE_MASK

Definition at line **504** of file **af-types.h**.

#define emberAfHaveDiscoveryResponseStatus (status)

A simple way to determine if the service discovery callback has a response.

Definition at line **510** of file **af-types.h**.

#define EMBER_AF_SCENE_TABLE_NULL_INDEX

Indicates the absence of a Scene table entry.

Definition at line **755** of file **af-types.h**.

#define EMBER_AF_SCENE_TABLE_UNUSED_ENDPOINT_ID

Value used when setting or getting the endpoint in a Scene table entry. It indicates that the entry is not in use.

Definition at line **760** of file **af-types.h**.

#define ZCL_SCENES_CLUSTER_MAXIMUM_NAME_LENGTH

Maximum length of Scene names, not including the length byte.

Definition at line **764** of file **af-types.h**.

#define ZCL_SCENES_GLOBAL_SCENE_GROUP_ID

The group identifier for the global scene.

Definition at line **768** of file **af-types.h**.

#define ZCL_PRICE_CLUSTER_MAXIMUM_RATE_LABEL_LENGTH

Definition at line **831** of file **af-types.h**.

#define EMBER_AF_OTA_MAX_HEADER_STRING_LENGTH

The maximum size of the string that is present in the header of the ZigBee Over-the-air file format.

Definition at line **908** of file **af-types.h**.

#define APP_NOTIFY_ERROR_CODE_START

Definition at line **1004** of file **af-types.h**.

#define APP_NOTIFY_MESSAGE_TEXT

Definition at line **1005** of file **af-types.h**.

Typedef Documentation

typedef void(EmberAfServiceDiscoveryCallback)(const EmberAfServiceDiscoveryResult *result)

This defines a callback where a code element or cluster can be informed as to the result of a service discovery they have requested. For each match, the callback is fired with all the resulting matches from that source. If the discovery was unicast to a specific device, then the callback will only be fired once with either MATCH_FOUND or COMPLETE (no matches found). If the discovery is broadcast then multiple callbacks may be fired with EMBER_AF_SERVICE_DISCOVERY_RESPONSE_RECEIVED. After a couple seconds the callback will then be fired with EMBER_AF_SERVICE_DISCOVERY_COMPLETE as the result.

Definition at line **570** of file **af-types.h**.

typedef void(EmberAfPartnerLinkKeyExchangeCallback)(boolean success)

This defines a callback where a code element or cluster can be informed as to the result of a request to initiate a partner link key exchange. The callback will be triggered with success equal to TRUE if the exchange completed successfully.

Definition at line **578** of file **af-types.h**.

typedef void(* EmberAfTickFunction)(int8u endpoint)

Type for refering to the tick callback for cluster.

Tick function will be called once for each tick for each endpoint in the cluster. The rate of tick is determined by the metadata of the cluster.

Definition at line **1057** of file **af-types.h**.

typedef void(* EmberAfInitFunction)(int8u endpoint)

Type for refering to the init callback for cluster.

Init function is called when the application starts up, once for each cluster/endpoint combination.

Definition at line **1066** of file **af-types.h**.

typedef void(* EmberAfClusterAttributeChangedCallback)(int8u endpoint, EmberAfAttributeId attributeId)

Type for refering to the attribute changed callback function.

This function is called just after an attribute changes.

Definition at line **1073** of file **af-types.h**.

typedef void(* EmberAfManufacturerSpecificClusterAttributeChangedCallback)(int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

Type for refering to the manufacturer specific attribute changed callback function.

This function is called just after a manufacturer specific attribute changes.

Definition at line **1082** of file **af-types.h**.

typedef void(* EmberAfDefaultResponseFunction)(int8u endpoint, int8u commandId, EmberAfStatus status)

Type for refering to the default response callback function.

This function is called when default response is received, before the global callback. Global callback is called immediatelly afterwards.

Definition at line **1093** of file **af-types.h**.

typedef void(* EmberAfMessageSentFunction)(EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Type for refering to the message sent callback function.

This function is called when a message is sent.

Definition at line **1102** of file **af-types.h**.

Enumeration Type Documentation

enum EmberAfServiceDiscoveryStatus

This is an enum used to indicate the result of the service discovery. Unicast discoveries are completed as soon as a response is received. Broadcast discoveries wait a period of time for multiple responses to be received.

Enumerator:

EMBER_AF_BROADCAST_SERVICE_DISCOVERY_COMPLETE
EMBER_AF_BROADCAST_SERVICE_DISCOVERY_RESPONSE_RECEIVED
EMBER_AF_UNICAST_SERVICE_DISCOVERY_TIMEOUT
EMBER_AF_UNICAST_SERVICE_DISCOVERY_COMPLETE_WITH_RESPONSE

Definition at line **497** of file **af-types.h**.

enum EmberAfEventSleepControl

This is an enum used to control how the device will sleep for a given active cluster related event. When the event is scheduled, the scheduling code can pass a sleep control value which will be stored along with the event. The processor is only allowed to sleep according to the most restrictive sleep control value for any active event. For instance, if two events are active, one with EMBER_AF_OK_TO_HIBERNATE and the other with EMBER_AF_OK_TO_NAP, then the processor will only be allowed to nap until the second event is deactivated.

Enumerator:

EMBER_AF_OK_TO_HIBERNATE
EMBER_AF_OK_TO_NAP
EMBER_AF_STAY_AWAKE

Definition at line **593** of file **af-types.h**.

enum EmberAfOtaStorageStatus

This status contains the success or error code of an OTA storage device operation.

Enumerator:

EMBER_AF_OTA_STORAGE_SUCCESS
EMBER_AF_OTA_STORAGE_ERROR
EMBER_AF_OTA_STORAGE_RETURN_DATA_TOO_LONG
EMBER_AF_OTA_STORAGE_PARTIAL_FILE_FOUND

Definition at line **897** of file **af-types.h**.

enum EmberAfKeyEstablishmentNotifyMessage

Enumerator:

NO_APP_MESSAGE	
RECEIVED_PARTNER_CERTIFICATE	
GENERATING_EPHEMERAL_KEYS	
GENERATING_SHARED_SECRET	
KEY_GENERATION_DONE	
GENERATE_SHARED_SECRET_DONE	
LINK_KEY_ESTABLISHED	LINK_KEY_ESTABLISHED indicates Success, key establishment done.
NO_LOCAL_RESOURCES	Error codes: Transient failures where Key Establishment could be retried
PARTNER_NO_RESOURCES	
TIMEOUT_OCCURRED	
INVALID_APP_COMMAND	
MESSAGE_SEND_FAILURE	
PARTNER_SENT_TERMINATE	
INVALID_PARTNER_MESSAGE	
BAD_CERTIFICATE_ISSUER	Fatal Errors: These results are not worh retrying because the outcome will not change
KEY_CONFIRM_FAILURE	
BAD_KEY_ESTABLISHMENT_SUITE	
KEY_TABLE_FULL	
NO_ESTABLISHMENT_ALLOWED	Neither initiator nor responder is an ESP/TC so the key establishment is not allowed per the spec.

Definition at line **959** of file **af-types.h**.

enum EmberAfImageVerifyStatus

This enumeration is used to indicate the state of an OTA bootload image undergoing verification. This is used both for cryptographic verification and manufacturer specific verification.

Enumerator:

EMBER_AF_IMAGE_GOOD
EMBER_AF_IMAGE_BAD
EMBER_AF_IMAGE_VERIFY_IN_PROGRESS

Definition at line **1036** of file **af-types.h**.

af.h File Reference

The master include file for the Ember ApplicationFramework API. More...

```
#include "app/framework/util/config.h"
#include <PLATFORM_HEADER>
#include "stack/include/ember.h"
#include "hal/hal.h"
#include "app/util/serial/serial.h"
#include "stack/include/event.h"
#include "stack/include/error.h"
#include "af-types.h"
#include "app/framework/util/print.h"
#include "app/framework/gen/af-structs.h"
#include "app/framework/gen/attribute-id.h"
#include "app/framework/gen/attribute-storage.h"
#include "app/framework/gen/attribute-type.h"
#include "app/framework/gen/call-command-handler.h"
#include "app/framework/gen/callback.h"
#include "app/framework/gen/client-command-macro.h"
#include "app/framework/gen/cluster-id.h"
#include "app/framework/gen/command-id.h"
#include "app/framework/gen/debug-printing.h"
#include "app/framework/gen/enums.h"
#include "app/framework/gen/print-cluster.h"
#include "app/framework/gen/stack-handlers.h"
#include "app/framework/util/client-api.h"
```

Go to the source code of this file.

Defines

	#define **CONFIGURATION_HEADER**

Attribute Storage

	#define **emberAfClusterIsManufacturerSpecific**(cluster)
	#define **emberAfAttributeIsReadOnly**(metadata)
	#define **emberAfAttributeIsClient**(metadata)
	#define **emberAfAttributeIsTokenized**(metadata)
	#define **emberAfAttributeIsSingleton**(metadata)
	#define **emberAfAttributeIsManufacturerSpecific**(metadata)
	#define **emberAfAttributeSize**(metadata)
	#define **emberAfEndpointFromIndex**(index)
	#define **emberAfProfileIdFromIndex**(index)
	#define **emberAfDeviceIdFromIndex**(index)
	#define **emberAfDeviceVersionFromIndex**(index)

	#define **emberAfPrimaryProfileId**()
	#define **emberAfPrimaryEndpoint**()
	#define **emberAfGetInt8u**(message, currentIndex, msgLen)
	#define **emberAfCopyInt8u**(data, index, x)
	enum { **EMBER_AF_DATA_TYPE_ANALOG**, **EMBER_AF_DATA_TYPE_DISCRETE**, **EMBER_AF_DATA_TYPE_NONE** }
EmberAfDefinedEndpoint	**endpoints** []
EmberAfAttributeMetadata *	**emberAfLocateAttributeMetadata** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attribute, int8u mask, int16u manufacturerCode)
boolean	**emberAfContainsCluster** (int8u endpoint, **EmberAfClusterId** clusterId)
boolean	**emberAfContainsServer** (int8u endpoint, **EmberAfClusterId** clusterId)
boolean	**emberAfContainsClient** (int8u endpoint, **EmberAfClusterId** clusterId)
boolean	**emberAfReadOrUpdateAttribute** (**EmberAfAttributeSearchRecord** *attRecord, **EmberAfAttributeMetadata** **metadata, int8u *buffer, boolean write)
EmberAfStatus	**emberAfWriteAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int8u mask, int8u *dataPtr, int8u dataType)
EmberAfStatus	**emberAfWriteServerAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int8u *dataPtr, int8u dataType)
EmberAfStatus	**emberAfWriteClientAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int8u *dataPtr, int8u dataType)
EmberAfStatus	**emberAfWriteManufacturerSpecificServerAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int16u manufacturerCode, int8u *dataPtr, int8u dataType)
EmberAfStatus	**emberAfWriteManufacturerSpecificClientAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int16u manufacturerCode, int8u *dataPtr, int8u dataType)
EmberAfStatus	**emberAfVerifyAttributeWrite** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int8u mask, int16u manufacturerCode, int8u *buffer, int8u dataType)
EmberAfStatus	**emberAfReadAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int8u mask, int8u *dataPtr, int8u maxLength, int8u *dataType)
EmberAfStatus	**emberAfReadServerAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int8u *dataPtr, int8u maxLength)
EmberAfStatus	**emberAfReadClientAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int8u *dataPtr, int8u maxLength)
EmberAfStatus	**emberAfReadManufacturerSpecificServerAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int16u manufacturerCode, int8u *dataPtr, int8u maxLength)
EmberAfStatus	**emberAfReadManufacturerSpecificClientAttribute** (int8u endpoint, **EmberAfClusterId** cluster, **EmberAfAttributeId** attributeID, int16u manufacturerCode, int8u *dataPtr, int8u maxLength)
int8u	**emberAfGetDataSize** (int8u dataType)
int8u	**emberAfIndexFromEndpoint** (int8u endpoint)
int8u	**emberAfEndpointCount** (void)
int8u	**emberAfGetAttributeAnalogOrDiscreteType** (int8u dataType)

boolean	**emberAfIsTypeSigned** (int8u dataType)
int32u	**emberAfGetInt32u** (const int8u *message, int16u currentIndex, int16u msgLen)
int32u	**emberAfGetInt24u** (const int8u *message, int16u currentIndex, int16u msgLen)
int16u	**emberAfGetInt16u** (const int8u *message, int16u currentIndex, int16u msgLen)
int8u *	**emberAfGetString** (int8u *message, int16u currentIndex, int16u msgLen)
void	**emberAfCopyInt16u** (int8u *data, int16u index, int16u x)
void	**emberAfCopyInt24u** (int8u *data, int16u index, int32u x)
void	**emberAfCopyInt32u** (int8u *data, int16u index, int32u x)
void	**emberAfCopyString** (int8u *dest, int8u *src, int8u size)
int8u	**emberAfStringLength** (const int8u *buffer)

Miscellaneous

#define	**EMBER_AF_NEW_IMAGE_VERIFICATION**
#define	**EMBER_AF_CONTINUE_IMAGE_VERIFY**
PGM **EmberAfOtaImageId**	**emberAfInvalidImageId**
boolean	**emberAfIsThisDataTypeAStringType** (int8u dataType)
int8u	**emberAfNextSequence** (void)
int8s	**emberAfCompareValues** (int8u *val1, int8u *val2, int8u len)
void	**emberAfGetEui64** (EmberEUI64 returnEui64)
EmberNodeId	**emberAfGetNodeId** (void)

Printing

#define	**emberAfGuaranteedPrint**(...)
#define	**emberAfGuaranteedPrintln**(...)
#define	**emberAfGuaranteedPrintBuffer**(buffer, len, withSpace)
#define	**emberAfGuaranteedPrintString**(buffer)
#define	**emberAfGuaranteedFlush**()
boolean	**emberAfPrintEnabled** (int16u functionality)
void	**emberAfPrintBuffer** (int16u area, const int8u *buffer, int16u bufferLen, boolean withSpaces)
void	**emberAfPrintString** (int16u area, const int8u *buffer)
void	**emberAfPrint** (int16u functionality, PGM_P formatString,...)
void	**emberAfPrintln** (int16u functionality, PGM_P formatString,...)
void	**emberAfFlush** (int16u functionality)
void	**emberAfPrintOn** (int16u functionality)
void	**emberAfPrintOff** (int16u functionality)
void	**emberAfPrintAllOn** (void)
void	**emberAfPrintAllOff** (void)
void	**emberAfPrintStatus** (void)
void	**emberAfPrintLittleEndianEui64** (const EmberEUI64 eui64)
void	**emberAfPrintBigEndianEui64** (const EmberEUI64 eui64)
void	**emberAfPrintMessageData** (int8u *data, int16u length)

Hibernation

#define	**MILLISECOND_TICKS_PER_QUARTERSECOND**
#define	**MILLISECOND_TICKS_PER_MINUTE**
#define	**MILLISECOND_TICKS_PER_HOUR**
#define	**EMBER_AF_SET_HIBERNATE_DURATION**(newValue)
#define	**EMBER_AF_SET_NAP_DURATION**(newValue)
#define	**EMBER_AF_WAKE_TIMEOUT**
#define	**EMBER_AF_WAKE_TIMEOUT_MASK**
#define	**emberAfOkToNap**()
#define	**emberAfCurrentAppTasks**()
#define	**EMBER_AF_CLIENT_CLUSTER_TICK**
#define	**EMBER_AF_SERVER_CLUSTER_TICK**
#define	**emberAfQSToNextEvent**(maxQS)
#define	**emberAfGetDefaultSleepControl**()
enum	{ **EMBER_AF_WAITING_FOR_DATA_ACK**, **EMBER_AF_LAST_POLL_GOT_DATA**, **EMBER_AF_WAITING_FOR_SERVICE_DISCOVERY**, **EMBER_AF_WAITING_FOR_ZDO_RESPONSE**, **EMBER_AF_WAITING_FOR_ZCL_RESPONSE**, **EMBER_AF_WAITING_FOR_REGISTRATION**, **EMBER_AF_WAITING_FOR_PARTNER_LINK_KEY_EXCHANGE** }
int32u	**emberAfHibernateDuration**
int32u	**emAfCurrentAppTasks**
EmberAfEventSleepControl	**emAfDefaultSleepControl**
void	**emberAfAddToCurrentAppTasks** (int32u mask)
void	**emberAfRemoveFromCurrentAppTasks** (int32u mask)
boolean	**emberAfOkToHibernate** (void)
void	**emberAfRunEvents** (void)
boolean	**emberAfStartMove** (void)
EmberStatus	**emberAfScheduleClusterTick** (int8u endpoint, int16u clusterId, boolean isClient, int32u timeMs, **EmberAfEventSleepControl** sleepControl)
EmberStatus	**emberAfDeactivateClusterTick** (int8u endpoint, int16u clusterId, boolean isClient)
EmberStatus	**emberAfEventControlSetDelay** (EmberEventControl *eventControl, int32u timeMs)
int32u	**emberAfMsToNextEvent** (int32u maxMs)
EmberAfEventSleepControl	**emberAfGetCurrentSleepControl** (void)
void	**emberAfSetDefaultSleepControl** (**EmberAfEventSleepControl** control)

Messaging

#define	**emberAfSetResponseWaiting**()

#define	**emberAfSendInterpanUnicast**(destinationShortAddress,destinationPanId, destinationAppProfileId,destinationClusterId,multicastIdOrZero, longAddressOrNull,messageLength,messageBytes)
#define	**emberAfSendCommandInterpanUnicast**(eui64,nodeId,panId,multicastId, profileId)
#define	**EMBER_AF_CLIENT_CLUSTER_DISCOVERY**
#define	**EMBER_AF_SERVER_CLUSTER_DISCOVERY**
#define	**emberAfCurrentCommand**()
#define	**emberAfCurrentEndpoint**()
boolean	**emberAfResponseWaitingFlag**
EmberAfClusterCommand *	**emAfCurrentCommand**
EmberStatus	**emberAfSendResponse** (void)
EmberStatus	**emberAfSendRequest** (void)
EmberStatus	**emberAfSendMulticast** (EmberMulticastId multicastId, EmberApsFrame *apsFrame, int16u messageLength, int8u *message)
EmberStatus	**emberAfSendBroadcast** (EmberNodeId destination, EmberApsFrame *apsFrame, int16u messageLength, int8u *message)
EmberStatus	**emberAfSendUnicast** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u messageLength, int8u *message)
EmberStatus	**emberAfSendInterPan** (EmberPanId panId, const EmberEUI64 destinationLongId, EmberNodeId destinationShortId, EmberMulticastId multicastId, **EmberAfClusterId** clusterId, **EmberAfProfileId** profileId, int16u messageLength, int8u *messageBytes)
EmberStatus	**emberAfSendEndDeviceBind** (int8u endpoint)
EmberStatus	**emberAfSendCommandMulticast** (EmberMulticastId multicastId)
EmberStatus	**emberAfSendCommandUnicast** (EmberOutgoingMessageType type, int16u indexOrDestination)
EmberStatus	**emberAfSendCommandBroadcast** (EmberNodeId destination)
EmberStatus	**emberAfSendCommandInterPan** (EmberPanId panId, const EmberEUI64 destinationLongId, EmberNodeId destinationShortId, EmberMulticastId multicastId, **EmberAfProfileId** profileId)
EmberStatus	**emberAfSendDefaultResponse** (const **EmberAfClusterCommand** *cmd, **EmberAfStatus** status)
int8u	**emberAfMaximumApsPayloadLength** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame)
EmberApsFrame *	**emberAfGetCommandApsFrame** (void)
EmberStatus	**emberAfFindDevicesSupportingCluster** (EmberNodeId target, **EmberAfClusterId** clusterId, boolean serverCluster, **EmberAfServiceDiscoveryCallback** *callback)
EmberStatus	**emberAfFindDevicesByProfileAndCluster** (EmberNodeId target, **EmberAfProfileId** profileId, **EmberAfClusterId** clusterId, boolean serverCluster, **EmberAfServiceDiscoveryCallback** *callback)
EmberStatus	**emberAfFindIeeeAddress** (EmberNodeId shortAddress, **EmberAfServiceDiscoveryCallback** *callback)
EmberStatus	**emberAfFindNodeId** (EmberEUI64 longAddress, **EmberAfServiceDiscoveryCallback** *callback)
int8u	**emberAfAddAddressTableEntry** (EmberEUI64 longId, EmberNodeId shortId)
EmberStatus	**emberAfSetAddressTableEntry** (int8u index, EmberEUI64 longId, EmberNodeId shortId)
EmberStatus	**emberAfInitiateKeyEstablishment** (EmberNodeId nodeId, int8u endpoint)

EmberStatus	**emberAfInitiateInterPanKeyEstablishment** (EmberPanId panId, const EmberEUI64 eui64)
boolean	**emberAfPerformingKeyEstablishment** (void)
EmberStatus	**emberAfInitiatePartnerLinkKeyExchange** (EmberNodeId target, int8u endpoint, **EmberAfPartnerLinkKeyExchangeCallback** *callback)

ZCL macros

#define	**ZCL_FRAME_CONTROL_FRAME_TYPE_MASK**
#define	**ZCL_CLUSTER_SPECIFIC_COMMAND**
#define	**ZCL_PROFILE_WIDE_COMMAND**
#define	**ZCL_MANUFACTURER_SPECIFIC_MASK**
#define	**ZCL_FRAME_CONTROL_DIRECTION_MASK**
#define	**ZCL_FRAME_CONTROL_SERVER_TO_CLIENT**
#define	**ZCL_FRAME_CONTROL_CLIENT_TO_SERVER**
#define	**ZCL_DISABLE_DEFAULT_RESPONSE_MASK**
#define	**EMBER_AF_ZCL_OVERHEAD**
#define	**EMBER_AF_ZCL_MANUFACTURER_SPECIFIC_OVERHEAD**

Device Control

boolean	**emberAfIsDeviceEnabled** (int8u endpoint)
void	**emberAfSetDeviceEnabled** (int8u endpoint, boolean enabled)

Time

int32u	**emberAfGetCurrentTime** (void)
void	**emberAfSetTime** (int32u utcTime)
void	**emberAfPrintTime** (int32u utcTime)

Stack handler utility functions

void	**emberAfPollCompleteHandler** (EmberStatus status)
void	**emberAfEzspPollCompleteHandler** (EmberStatus status)

Network utility functions

EmberStatus	**emberAfFormNetwork** (EmberNetworkParameters *parameters)
EmberStatus	**emberAfJoinNetwork** (EmberNetworkParameters *parameters)
EmberStatus	**emberAfFindUnusedPanIdAndForm** (void)
EmberStatus	**emberAfStartSearchForJoinableNetwork** (void)

Detailed Description

The master include file for the Ember ApplicationFramework API.

Definition in file **af.h**.

---

attribute-id.h File Reference

Go to the source code of this file.

Defines

```
#define ZCL_VERSION_ATTRIBUTE_ID
#define ZCL_APPLICATION_VERSION_ATTRIBUTE_ID
#define ZCL_STACK_VERSION_ATTRIBUTE_ID
#define ZCL_HW_VERSION_ATTRIBUTE_ID
#define ZCL_MANUFACTURER_NAME_ATTRIBUTE_ID
#define ZCL_MODEL_IDENTIFIER_ATTRIBUTE_ID
#define ZCL_DATE_CODE_ATTRIBUTE_ID
#define ZCL_POWER_SOURCE_ATTRIBUTE_ID
#define ZCL_LOCATION_DESCRIPTION_ATTRIBUTE_ID
#define ZCL_PHYSICAL_ENVIRONMENT_ATTRIBUTE_ID
#define ZCL_DEVICE_ENABLED_ATTRIBUTE_ID
#define ZCL_ALARM_MASK_ATTRIBUTE_ID
#define ZCL_DISABLE_LOCAL_CONFIG_ATTRIBUTE_ID
#define ZCL_MAINS_VOLTAGE_ATTRIBUTE_ID
#define ZCL_MAINS_FREQUENCY_ATTRIBUTE_ID
#define ZCL_MAINS_ALARM_MASK_ATTRIBUTE_ID
#define ZCL_MAINS_VOLTAGE_MIN_THRESHOLD_ATTRIBUTE_ID
#define ZCL_MAINS_VOLTAGE_MAX_THRESHOLD_ATTRIBUTE_ID
#define ZCL_MAINS_VOLTAGE_DWELL_TRIP_POINT_ATTRIBUTE_ID
#define ZCL_BATTERY_VOLTAGE_ATTRIBUTE_ID
#define ZCL_BATTERY_MANUFACTURER_ATTRIBUTE_ID
#define ZCL_BATTERY_SIZE_ATTRIBUTE_ID
#define ZCL_BATTERY_AHR_RATING_ATTRIBUTE_ID
#define ZCL_BATTERY_QUANTITY_ATTRIBUTE_ID
#define ZCL_BATTERY_RATED_VOLTAGE_ATTRIBUTE_ID
#define ZCL_BATTERY_ALARM_MASK_ATTRIBUTE_ID
#define ZCL_BATTERY_VOLTAGE_MIN_THRESHOLD_ATTRIBUTE_ID
#define ZCL_CURRENT_TEMPERATURE_ATTRIBUTE_ID
#define ZCL_MIN_TEMP_EXPERIENCED_ATTRIBUTE_ID
#define ZCL_MAX_TEMP_EXPERIENCED_ATTRIBUTE_ID
#define ZCL_OVER_TEMP_TOTAL_DWELL_ATTRIBUTE_ID
#define ZCL_DEVICE_TEMP_ALARM_MASK_ATTRIBUTE_ID
#define ZCL_LOW_TEMP_THRESHOLD_ATTRIBUTE_ID
#define ZCL_HIGH_TEMP_THRESHOLD_ATTRIBUTE_ID
#define ZCL_LOW_TEMP_DWELL_TRIP_POINT_ATTRIBUTE_ID
#define ZCL_HIGH_TEMP_DWELL_TRIP_POINT_ATTRIBUTE_ID
#define ZCL_IDENTIFY_TIME_ATTRIBUTE_ID
#define ZCL_GROUP_NAME_SUPPORT_ATTRIBUTE_ID
#define ZCL_SCENE_COUNT_ATTRIBUTE_ID
#define ZCL_CURRENT_SCENE_ATTRIBUTE_ID
#define ZCL_CURRENT_GROUP_ATTRIBUTE_ID
#define ZCL_SCENE_VALID_ATTRIBUTE_ID
```

```
#define ZCL_SCENE_NAME_SUPPORT_ATTRIBUTE_ID
#define ZCL_LAST_CONFIGURED_BY_ATTRIBUTE_ID
#define ZCL_ON_OFF_ATTRIBUTE_ID
#define ZCL_SAMPLE_MFG_SPECIFIC_TRANSITION_TIME_ATTRIBUTE_ID
#define ZCL_SWITCH_TYPE_ATTRIBUTE_ID
#define ZCL_SWITCH_ACTIONS_ATTRIBUTE_ID
#define ZCL_CURRENT_LEVEL_ATTRIBUTE_ID
#define ZCL_LEVEL_CONTROL_REMAINING_TIME_ATTRIBUTE_ID
#define ZCL_ON_OFF_TRANSITION_TIME_ATTRIBUTE_ID
#define ZCL_ON_LEVEL_ATTRIBUTE_ID
#define ZCL_ALARM_COUNT_ATTRIBUTE_ID
#define ZCL_TIME_ATTRIBUTE_ID
#define ZCL_TIME_STATUS_ATTRIBUTE_ID
#define ZCL_TIME_ZONE_ATTRIBUTE_ID
#define ZCL_DST_START_ATTRIBUTE_ID
#define ZCL_DST_END_ATTRIBUTE_ID
#define ZCL_DST_SHIFT_ATTRIBUTE_ID
#define ZCL_STANDARD_TIME_ATTRIBUTE_ID
#define ZCL_LOCAL_TIME_ATTRIBUTE_ID
#define ZCL_LAST_SET_TIME_ATTRIBUTE_ID
#define ZCL_VALID_UNTIL_TIME_ATTRIBUTE_ID
#define ZCL_LOCATION_TYPE_ATTRIBUTE_ID
#define ZCL_LOCATION_METHOD_ATTRIBUTE_ID
#define ZCL_LOCATION_AGE_ATTRIBUTE_ID
#define ZCL_QUALITY_MEASURE_ATTRIBUTE_ID
#define ZCL_NUMBER_OF_DEVICES_ATTRIBUTE_ID
#define ZCL_COORDINATE1_ATTRIBUTE_ID
#define ZCL_COORDINATE2_ATTRIBUTE_ID
#define ZCL_COORDINATE3_ATTRIBUTE_ID
#define ZCL_POWER_ATTRIBUTE_ID
#define ZCL_PATH_LOSS_EXPONENT_ATTRIBUTE_ID
#define ZCL_REPORTING_PERIOD_ATTRIBUTE_ID
#define ZCL_CALCULATION_PERIOD_ATTRIBUTE_ID
#define ZCL_NUMBER_RSSI_MEASUREMENTS_ATTRIBUTE_ID
#define ZCL_ACTIVE_TEXT_ATTRIBUTE_ID
#define ZCL_DESCRIPTION_ATTRIBUTE_ID
#define ZCL_INACTIVE_TEXT_ATTRIBUTE_ID
#define ZCL_OUT_OF_SERVICE_ATTRIBUTE_ID
#define ZCL_POLARITY_ATTRIBUTE_ID
#define ZCL_PRESENT_VALUE_ATTRIBUTE_ID
#define ZCL_RELIABILITY_ATTRIBUTE_ID
#define ZCL_STATUS_FLAGS_ATTRIBUTE_ID
#define ZCL_APPLICATION_TYPE_ATTRIBUTE_ID
#define ZCL_SHORT_ADDRESS_ATTRIBUTE_ID
#define ZCL_EXTENDED_PAN_ID_ATTRIBUTE_ID
#define ZCL_PAN_ID_ATTRIBUTE_ID
#define ZCL_CHANNEL_MASK_ATTRIBUTE_ID
#define ZCL_PROTOCOL_VERSION_ATTRIBUTE_ID
#define ZCL_STACK_PROFILE_ATTRIBUTE_ID
#define ZCL_STARTUP_CONTROL_ATTRIBUTE_ID
#define ZCL_TRUST_CENTER_ADDRESS_ATTRIBUTE_ID
#define ZCL_TRUST_CENTER_MASTER_KEY_ATTRIBUTE_ID
```

```
#define ZCL_NETWORK_KEY_ATTRIBUTE_ID
#define ZCL_USE_INSECURE_JOIN_ATTRIBUTE_ID
#define ZCL_PRECONFIGURED_LINK_KEY_ATTRIBUTE_ID
#define ZCL_NETWORK_KEY_SEQUENCE_NUMBER_ATTRIBUTE_ID
#define ZCL_NETWORK_KEY_TYPE_ATTRIBUTE_ID
#define ZCL_NETWORK_MANAGER_ADDRESS_ATTRIBUTE_ID
#define ZCL_SCAN_ATTEMPTS_ATTRIBUTE_ID
#define ZCL_TIME_BETWEEN_SCANS_ATTRIBUTE_ID
#define ZCL_REJOIN_INTERVAL_ATTRIBUTE_ID
#define ZCL_MAX_REJOIN_INTERVAL_ATTRIBUTE_ID
#define ZCL_INDERECT_POLL_RATE_ATTRIBUTE_ID
#define ZCL_PARENT_RETRY_THRESHOLD_ATTRIBUTE_ID
#define ZCL_CONCENTRATOR_FLAG_ATTRIBUTE_ID
#define ZCL_CONCENTRATOR_RADIUS_ATTRIBUTE_ID
#define ZCL_CONCENTRATOR_DISCOVERY_TIME_ATTRIBUTE_ID
#define ZCL_SHADE_CONFIG_PHYSICAL_CLOSED_LIMIT_ATTRIBUTE_ID
#define ZCL_SHADE_CONFIG_MOTOR_STEP_SIZE_ATTRIBUTE_ID
#define ZCL_SHADE_CONFIG_STATUS_ATTRIBUTE_ID
#define ZCL_SHADE_CONFIG_CLOSED_LIMIT_ATTRIBUTE_ID
#define ZCL_SHADE_CONFIG_MODE_ATTRIBUTE_ID
#define ZCL_MAX_PRESSURE_ATTRIBUTE_ID
#define ZCL_MAX_SPEED_ATTRIBUTE_ID
#define ZCL_MAX_FLOW_ATTRIBUTE_ID
#define ZCL_MIN_CONST_PRESSURE_ATTRIBUTE_ID
#define ZCL_MAX_CONST_PRESSURE_ATTRIBUTE_ID
#define ZCL_MIN_COMP_PRESSURE_ATTRIBUTE_ID
#define ZCL_MAX_COMP_PRESSURE_ATTRIBUTE_ID
#define ZCL_MIN_CONST_SPEED_ATTRIBUTE_ID
#define ZCL_MAX_CONST_SPEED_ATTRIBUTE_ID
#define ZCL_MIN_CONST_FLOW_ATTRIBUTE_ID
#define ZCL_MAX_CONST_FLOW_ATTRIBUTE_ID
#define ZCL_MIN_CONST_TEMP_ATTRIBUTE_ID
#define ZCL_MAX_CONST_TEMP_ATTRIBUTE_ID
#define ZCL_PUMP_STATUS_ATTRIBUTE_ID
#define ZCL_EFFECTIVE_OPERATION_MODE_ATTRIBUTE_ID
#define ZCL_EFFECTIVE_CONTROL_MODE_ATTRIBUTE_ID
#define ZCL_CAPACITY_ATTRIBUTE_ID
#define ZCL_SPEED_ATTRIBUTE_ID
#define ZCL_LIFETIME_RUNNING_HOURS_ATTRIBUTE_ID
#define ZCL_PUMP_POWER_ATTRIBUTE_ID
#define ZCL_LIFETIME_ENERGY_CONSUMED_ATTRIBUTE_ID
#define ZCL_OPERATION_MODE_ATTRIBUTE_ID
#define ZCL_CONTROL_MODE_ATTRIBUTE_ID
#define ZCL_PUMP_ALARM_MASK_ATTRIBUTE_ID
#define ZCL_LOCAL_TEMPERATURE_ATTRIBUTE_ID
#define ZCL_OUTDOOR_TEMPERATURE_ATTRIBUTE_ID
#define ZCL_THERMOSTAT_OCCUPANCY_ATTRIBUTE_ID
#define ZCL_ABS_MIN_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID
#define ZCL_ABS_MAX_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID
#define ZCL_ABS_MIN_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID
#define ZCL_ABS_MAX_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID
#define ZCL_PI_COOLING_DEMAND_ATTRIBUTE_ID
```

#define **ZCL_PI_HEATING_DEMAND_ATTRIBUTE_ID**
#define **ZCL_LOCAL_TEMPERATURE_CALIBRATION_ATTRIBUTE_ID**
#define **ZCL_OCCUPIED_COOLING_SETPOINT_ATTRIBUTE_ID**
#define **ZCL_OCCUPIED_HEATING_SETPOINT_ATTRIBUTE_ID**
#define **ZCL_UNOCCUPIED_COOLING_SETPOINT_ATTRIBUTE_ID**
#define **ZCL_UNOCCUPIED_HEATING_SETPOINT_ATTRIBUTE_ID**
#define **ZCL_MIN_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID**
#define **ZCL_MAX_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID**
#define **ZCL_MIN_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID**
#define **ZCL_MAX_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID**
#define **ZCL_MIN_SETPOINT_DEAD_BAND_ATTRIBUTE_ID**
#define **ZCL_REMOTE_SENSING_ATTRIBUTE_ID**
#define **ZCL_CONTROL_SEQUENCE_OF_OPERATION_ATTRIBUTE_ID**
#define **ZCL_SYSTEM_MODE_ATTRIBUTE_ID**
#define **ZCL_THERMOSTAT_ALARM_MASK_ATTRIBUTE_ID**
#define **ZCL_FAN_CONTROL_FAN_MODE_ATTRIBUTE_ID**
#define **ZCL_FAN_CONTROL_FAN_MODE_SEQUENCE_ATTRIBUTE_ID**
#define **ZCL_RELATIVE_HUMIDITY_ATTRIBUTE_ID**
#define **ZCL_DEHUMIDIFICATION_COOLING_ATTRIBUTE_ID**
#define **ZCL_RH_DEHUMIDIFICATION_SETPOINT_ATTRIBUTE_ID**
#define **ZCL_RELATIVE_HUMIDITY_MODE_ATTRIBUTE_ID**
#define **ZCL_DEHUMIDIFICATION_LOCKOUT_ATTRIBUTE_ID**
#define **ZCL_DEHUMIDIFICATION_HYSTERESIS_ATTRIBUTE_ID**
#define **ZCL_DEHUMIDIFICATION_MAX_COOL_ATTRIBUTE_ID**
#define **ZCL_RELATIVE_HUMIDITY_DISPLAY_ATTRIBUTE_ID**
#define **ZCL_TEMPERATURE_DISPLAY_MODE_ATTRIBUTE_ID**
#define **ZCL_KEYPAD_LOCKOUT_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_CURRENT_HUE_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_CURRENT_SATURATION_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_REMAINING_TIME_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_CURRENT_X_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_CURRENT_Y_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_DRIFT_COMPENSATION_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_COMPENSATION_TEXT_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_COLOR_TEMPERATURE_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_COLOR_MODE_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_NUMBER_OF_PRIMARIES_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_1_X_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_1_Y_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_1_INTENSITY_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_2_X_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_2_Y_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_2_INTENSITY_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_3_X_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_3_Y_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_3_INTENSITY_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_4_X_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_4_Y_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_4_INTENSITY_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_5_X_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_5_Y_ATTRIBUTE_ID**
#define **ZCL_COLOR_CONTROL_PRIMARY_5_INTENSITY_ATTRIBUTE_ID**

```
#define ZCL_COLOR_CONTROL_PRIMARY_6_X_ATTRIBUTE_ID
#define ZCL_COLOR_CONTROL_PRIMARY_6_Y_ATTRIBUTE_ID
#define ZCL_COLOR_CONTROL_PRIMARY_6_INTENSITY_ATTRIBUTE_ID
#define ZCL_COLOR_CONTROL_WHITE_POINT_X_ATTRIBUTE_ID
#define ZCL_COLOR_CONTROL_WHITE_POINT_Y_ATTRIBUTE_ID
#define ZCL_COLOR_CONTROL_COLOR_POINT_R_X_ATTRIBUTE_ID
#define ZCL_COLOR_CONTROL_COLOR_POINT_R_Y_ATTRIBUTE_ID
#define ZCL_COLOR_CONTROL_COLOR_POINT_R_INTENSITY_ATTRIBUTE_ID
#define ZCL_COLOR_CONTROL_COLOR_POINT_G_X_ATTRIBUTE_ID
#define ZCL_COLOR_CONTROL_COLOR_POINT_G_Y_ATTRIBUTE_ID
#define ZCL_COLOR_CONTROL_COLOR_POINT_G_INTENSITY_ATTRIBUTE_ID
#define ZCL_COLOR_CONTROL_COLOR_POINT_B_X_ATTRIBUTE_ID
#define ZCL_COLOR_CONTROL_COLOR_POINT_B_Y_ATTRIBUTE_ID
#define ZCL_COLOR_CONTROL_COLOR_POINT_B_INTENSITY_ATTRIBUTE_ID
#define ZCL_PHYSICAL_MIN_LEVEL_ATTRIBUTE_ID
#define ZCL_PHYSICAL_MAX_LEVEL_ATTRIBUTE_ID
#define ZCL_BALLAST_STATUS_ATTRIBUTE_ID
#define ZCL_MIN_LEVEL_ATTRIBUTE_ID
#define ZCL_MAX_LEVEL_ATTRIBUTE_ID
#define ZCL_POWER_ON_LEVEL_ATTRIBUTE_ID
#define ZCL_POWER_ON_FADE_TIME_ATTRIBUTE_ID
#define ZCL_INTRINSIC_BALLAST_FACTOR_ATTRIBUTE_ID
#define ZCL_BALLAST_FACTOR_ADJUSTMENT_ATTRIBUTE_ID
#define ZCL_LAMP_QUALITY_ATTRIBUTE_ID
#define ZCL_LAMP_TYPE_ATTRIBUTE_ID
#define ZCL_LAMP_MANUFACTURER_ATTRIBUTE_ID
#define ZCL_LAMP_RATED_HOURS_ATTRIBUTE_ID
#define ZCL_LAMP_BURN_HOURS_ATTRIBUTE_ID
#define ZCL_LAMP_ALARM_MODE_ATTRIBUTE_ID
#define ZCL_LAMP_BURN_HOURS_TRIP_POINT_ATTRIBUTE_ID
#define ZCL_ILLUM_MEASURED_VALUE_ATTRIBUTE_ID
#define ZCL_ILLUM_MIN_MEASURED_VALUE_ATTRIBUTE_ID
#define ZCL_ILLUM_MAX_MEASURED_VALUE_ATTRIBUTE_ID
#define ZCL_ILLUM_TOLERANCE_ATTRIBUTE_ID
#define ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_ATTRIBUTE_ID
#define ZCL_LEVEL_STATUS_ATTRIBUTE_ID
#define ZCL_SENSING_LIGHT_SENSOR_TYPE_ATTRIBUTE_ID
#define ZCL_ILLUMINANCE_TARGET_LEVEL_ATTRIBUTE_ID
#define ZCL_TEMP_MEASURED_VALUE_ATTRIBUTE_ID
#define ZCL_TEMP_MIN_MEASURED_VALUE_ATTRIBUTE_ID
#define ZCL_TEMP_MAX_MEASURED_VALUE_ATTRIBUTE_ID
#define ZCL_TEMP_TOLERANCE_ATTRIBUTE_ID
#define ZCL_PRESSURE_MEASURED_VALUE_ATTRIBUTE_ID
#define ZCL_PRESSURE_MIN_MEASURED_VALUE_ATTRIBUTE_ID
#define ZCL_PRESSURE_MAX_MEASURED_VALUE_ATTRIBUTE_ID
#define ZCL_PRESSURE_TOLERANCE_ATTRIBUTE_ID
#define ZCL_PRESSURE_SCALED_VALUE_ATTRIBUTE_ID
#define ZCL_PRESSURE_MIN_SCALED_VALUE_ATTRIBUTE_ID
#define ZCL_PRESSURE_MAX_SCALED_VALUE_ATTRIBUTE_ID
#define ZCL_PRESSURE_SCALED_TOLERANCE_ATTRIBUTE_ID
#define ZCL_PRESSURE_SCALE_ATTRIBUTE_ID
#define ZCL_FLOW_MEASURED_VALUE_ATTRIBUTE_ID
```

```
#define ZCL_FLOW_MIN_MEASURED_VALUE_ATTRIBUTE_ID
#define ZCL_FLOW_MAX_MEASURED_VALUE_ATTRIBUTE_ID
#define ZCL_FLOW_TOLERANCE_ATTRIBUTE_ID
#define ZCL_RELATIVE_HUMIDITY_MEASURED_VALUE_ATTRIBUTE_ID
#define ZCL_RELATIVE_HUMIDITY_MIN_MEASURED_VALUE_ATTRIBUTE_ID
#define ZCL_RELATIVE_HUMIDITY_MAX_MEASURED_VALUE_ATTRIBUTE_ID
#define ZCL_RELATIVE_HUMIDITY_TOLERANCE_ATTRIBUTE_ID
#define ZCL_OCCUPANCY_ATTRIBUTE_ID
#define ZCL_OCCUPANCY_SENSOR_TYPE_ATTRIBUTE_ID
#define ZCL_PIR_OCCUPIED_TO_UNOCCUPIED_DELAY_ATTRIBUTE_ID
#define ZCL_PIR_UNOCCUPIED_TO_OCCUPIED_DELAY_ATTRIBUTE_ID
#define ZCL_PIR_UNOCCUPIED_TO_OCCUPIED_THRESHOLD_ATTRIBUTE_ID
#define ZCL_ULTRASONIC_OCCUPIED_TO_UNOCCUPIED_DELAY_ATTRIBUTE_ID
#define ZCL_ULTRASONIC_UNOCCUPIED_TO_OCCUPIED_DELAY_ATTRIBUTE_ID
#define ZCL_ULTRASONIC_UNOCCUPIED_TO_OCCUPIED_THRESHOLD_ATTRIBUTE_ID
#define ZCL_ZONE_STATE_ATTRIBUTE_ID
#define ZCL_ZONE_TYPE_ATTRIBUTE_ID
#define ZCL_ZONE_STATUS_ATTRIBUTE_ID
#define ZCL_IAS_CIE_ADDRESS_ATTRIBUTE_ID
#define ZCL_MAX_DURATION_ATTRIBUTE_ID
#define ZCL_LOCK_STATE_ATTRIBUTE_ID
#define ZCL_LOCK_TYPE_ATTRIBUTE_ID
#define ZCL_ACTUATOR_ENABLED_ATTRIBUTE_ID
#define ZCL_DOOR_STATE_ATTRIBUTE_ID
#define ZCL_DOOR_OPEN_EVENTS_ATTRIBUTE_ID
#define ZCL_DOOR_CLOSED_EVENTS_ATTRIBUTE_ID
#define ZCL_OPEN_PERIOD_ATTRIBUTE_ID
#define ZCL_COVERING_TYPE_ATTRIBUTE_ID
#define ZCL_LIMIT_LIFT_ATTRIBUTE_ID
#define ZCL_LIMIT_TILT_ATTRIBUTE_ID
#define ZCL_CURRENT_LIFT_ATTRIBUTE_ID
#define ZCL_CURRENT_TILT_ATTRIBUTE_ID
#define ZCL_NUMBER_LIFT_ATTRIBUTE_ID
#define ZCL_NUMBER_TILT_ATTRIBUTE_ID
#define ZCL_CONFIG_STATUS_ATTRIBUTE_ID
#define ZCL_CURRENT_LIFT_PERCENTAGE_ATTRIBUTE_ID
#define ZCL_CURRENT_TILT_PERCENTAGE_ATTRIBUTE_ID
#define ZCL_OPEN_LIMIT_LIFT_ATTRIBUTE_ID
#define ZCL_CLOSED_LIMIT_LIFT_ATTRIBUTE_ID
#define ZCL_OPEN_LIMIT_TILT_ATTRIBUTE_ID
#define ZCL_CLOSED_LIMIT_TILT_ATTRIBUTE_ID
#define ZCL_VELOCITY_LIFT_ATTRIBUTE_ID
#define ZCL_ACCELERATION_LIFT_ATTRIBUTE_ID
#define ZCL_DECELERATION_LIFT_ATTRIBUTE_ID
#define ZCL_MODE_ATTRIBUTE_ID
#define ZCL_SETPOINTS_LIFT_ATTRIBUTE_ID
#define ZCL_SETPOINTS_TILT_ATTRIBUTE_ID
#define ZCL_MAXIMUM_INCOMING_TRANSFER_SIZE_ATTRIBUTE_ID
#define ZCL_MAXIMUM_OUTGOING_TRANSFER_SIZE_ATTRIBUTE_ID
#define ZCL_PROTOCOL_ADDRESS_ATTRIBUTE_ID
#define ZCL_UPGRADE_SERVER_ID_ATTRIBUTE_ID
#define ZCL_FILE_OFFSET_ATTRIBUTE_ID
```

```
#define ZCL_CURRENT_FILE_VERSION_ATTRIBUTE_ID
#define ZCL_CURRENT_ZIGBEE_STACK_VERSION_ATTRIBUTE_ID
#define ZCL_DOWNLOADED_FILE_VERSION_ATTRIBUTE_ID
#define ZCL_DOWNLOADED_ZIGBEE_STACK_VERSION_ATTRIBUTE_ID
#define ZCL_IMAGE_UPGRADE_STATUS_ATTRIBUTE_ID
#define ZCL_MANUFACTURER_ID_ATTRIBUTE_ID
#define ZCL_IMAGE_TYPE_ID_ATTRIBUTE_ID
#define ZCL_UTILITY_ENROLLMENT_GROUP_ATTRIBUTE_ID
#define ZCL_START_RANDOMIZE_MINUTES_ATTRIBUTE_ID
#define ZCL_STOP_RANDOMIZE_MINUTES_ATTRIBUTE_ID
#define ZCL_DEVICE_CLASS_VALUE_ATTRIBUTE_ID
#define ZCL_CURRENT_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_SUMMATION_RECEIVED_ATTRIBUTE_ID
#define ZCL_CURRENT_MAX_DEMAND_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_MAX_DEMAND_RECEIVED_ATTRIBUTE_ID
#define ZCL_DFT_SUMMATION_ATTRIBUTE_ID
#define ZCL_DAILY_FREEZE_TIME_ATTRIBUTE_ID
#define ZCL_POWER_FACTOR_ATTRIBUTE_ID
#define ZCL_READING_SNAP_SHOT_TIME_ATTRIBUTE_ID
#define ZCL_CURRENT_MAX_DEMAND_DELIVERED_TIME_ATTRIBUTE_ID
#define ZCL_CURRENT_MAX_DEMAND_RECEIVED_TIME_ATTRIBUTE_ID
#define ZCL_DEFAULT_UPDATE_PERIOD_ATTRIBUTE_ID
#define ZCL_FAST_POLL_UPDATE_PERIOD_ATTRIBUTE_ID
#define ZCL_CURRENT_BLOCK_PERIOD_CONSUMPTION_DELIVERED_ATTRIBUTE_ID
#define ZCL_DAILY_CONSUMPTION_TARGET_ATTRIBUTE_ID
#define ZCL_CURRENT_BLOCK_ATTRIBUTE_ID
#define ZCL_PROFILE_INTERVAL_PERIOD_ATTRIBUTE_ID
#define ZCL_INTERVAL_READ_REPORTING_PERIOD_ATTRIBUTE_ID
#define ZCL_PRESET_READING_TIME_ATTRIBUTE_ID
#define ZCL_VOLUME_PER_REPORT_ATTRIBUTE_ID
#define ZCL_FLOW_RESTRICTION_ATTRIBUTE_ID
#define ZCL_SUPPLY_STATUS_ATTRIBUTE_ID
#define ZCL_CURRENT_INLET_ENERGY_CARRIER_SUMMATION_ATTRIBUTE_ID
#define ZCL_CURRENT_OUTLET_ENERGY_CARRIER_SUMMATION_ATTRIBUTE_ID
#define ZCL_INLET_TEMPERATURE_ATTRIBUTE_ID
#define ZCL_OUTLET_TEMPERATURE_ATTRIBUTE_ID
#define ZCL_CONTROL_TEMPERATURE_ATTRIBUTE_ID
#define ZCL_CURRENT_INLET_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID
#define ZCL_CURRENT_OUTLET_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_SUMMATION_RECEIVED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_SUMMATION_RECEIVED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_SUMMATION_RECEIVED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_SUMMATION_RECEIVED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_SUMMATION_RECEIVED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_SUMMATION_RECEIVED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_SUMMATION_DELIVERED_ATTRIBUTE_ID
```

#define **ZCL_CURRENT_TIER7_SUMMATION_RECEIVED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER8_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER8_SUMMATION_RECEIVED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER9_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER9_SUMMATION_RECEIVED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER10_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER10_SUMMATION_RECEIVED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER11_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER11_SUMMATION_RECEIVED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER12_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER12_SUMMATION_RECEIVED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER13_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER13_SUMMATION_RECEIVED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER14_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER14_SUMMATION_RECEIVED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER15_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER15_SUMMATION_RECEIVED_ATTRIBUTE_ID**
#define **ZCL_STATUS_ATTRIBUTE_ID**
#define **ZCL_REMAINING_BATTERY_LIFE_ATTRIBUTE_ID**
#define **ZCL_HOURS_IN_OPERATION_ATTRIBUTE_ID**
#define **ZCL_HOURS_IN_FAULT_ATTRIBUTE_ID**
#define **ZCL_UNIT_OF_MEASURE_ATTRIBUTE_ID**
#define **ZCL_MULTIPLIER_ATTRIBUTE_ID**
#define **ZCL_DIVISOR_ATTRIBUTE_ID**
#define **ZCL_SUMMATION_FORMATTING_ATTRIBUTE_ID**
#define **ZCL_DEMAND_FORMATTING_ATTRIBUTE_ID**
#define **ZCL_HISTORICAL_CONSUMPTION_FORMATTING_ATTRIBUTE_ID**
#define **ZCL_METERING_DEVICE_TYPE_ATTRIBUTE_ID**
#define **ZCL_SITE_ID_ATTRIBUTE_ID**
#define **ZCL_METER_SERIAL_NUMBER_ATTRIBUTE_ID**
#define **ZCL_ENERGY_CARRIER_UNIT_OF_MEASURE_ATTRIBUTE_ID**
#define **ZCL_ENERGY_CARRIER_SUMMATION_FORMATTING_ATTRIBUTE_ID**
#define **ZCL_ENERGY_CARRIER_DEMAND_FORMATTING_ATTRIBUTE_ID**
#define **ZCL_TEMPERATURE_UNIT_OF_MEASURE_ATTRIBUTE_ID**
#define **ZCL_TEMPERATURE_FORMATTING_ATTRIBUTE_ID**
#define **ZCL_INSTANTANEOUS_DEMAND_ATTRIBUTE_ID**
#define **ZCL_CURRENT_DAY_CONSUMPTION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_DAY_CONSUMPTION_RECEIVED_ATTRIBUTE_ID**
#define **ZCL_PREVIOUS_DAY_CONSUMPTION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_PREVIOUS_DAY_CONSUMPTION_RECEIVED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_START_TIME_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_START_TIME_RECEIVED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_VALUE_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_VALUE_RECEIVED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_DAY_MAX_PRESSURE_ATTRIBUTE_ID**
#define **ZCL_CURRENT_DAY_MIN_PRESSURE_ATTRIBUTE_ID**
#define **ZCL_PREVIOUS_DAY_MAX_PRESSURE_ATTRIBUTE_ID**
#define **ZCL_PREVIOUS_DAY_MIN_PRESSURE_ATTRIBUTE_ID**
#define **ZCL_CURRENT_DAY_MAX_DEMAND_ATTRIBUTE_ID**
#define **ZCL_PREVIOUS_DAY_MAX_DEMAND_ATTRIBUTE_ID**
#define **ZCL_CURRENT_MONTH_MAX_DEMAND_ATTRIBUTE_ID**
#define **ZCL_CURRENT_YEAR_MAX_DEMAND_ATTRIBUTE_ID**

```
#define ZCL_CURRENT_DAY_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID
#define ZCL_PREVIOUS_DAY_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID
#define ZCL_CURRENT_MONTH_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID
#define ZCL_CURRENT_MONTH_MIN_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID
#define ZCL_CURRENT_YEAR_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID
#define ZCL_CURRENT_YEAR_MIN_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID
#define ZCL_MAX_NUMBER_OF_PERIODS_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_DEMAND_DELIVERED_ATTRIBUTE_ID
#define ZCL_DEMAND_LIMIT_ATTRIBUTE_ID
#define ZCL_DEMAND_INTEGRATION_PERIOD_ATTRIBUTE_ID
#define ZCL_NUMBER_OF_DEMAND_SUBINTERVALS_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_NO_TIER_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER1_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
```

```
#define ZCL_CURRENT_TIER2_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER2_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER3_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER4_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
```

```
#define ZCL_CURRENT_TIER5_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER5_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER6_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER7_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER8_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
```

```
#define ZCL_CURRENT_TIER9_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER9_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER10_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER11_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
```

```
#define ZCL_CURRENT_TIER12_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER12_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER13_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER14_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER15_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER15_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER15_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER15_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER15_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER15_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER15_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER15_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID
#define ZCL_CURRENT_TIER15_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID
```

#define **ZCL_CURRENT_TIER15_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER15_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER15_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER15_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER15_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER15_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_CURRENT_TIER15_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID**
#define **ZCL_GENERIC_ALARM_MASK_ATTRIBUTE_ID**
#define **ZCL_ELECTRICITY_ALARM_MASK_ATTRIBUTE_ID**
#define **ZCL_GENERIC_FLOW_PRESSURE_ALARM_MASK_ATTRIBUTE_ID**
#define **ZCL_WATER_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID**
#define **ZCL_HEAT_AND_COOLING_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID**
#define **ZCL_GAS_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID**
#define **ZCL_MIRROR_IEEE_ADDRESS_ATTRIBUTE_ID**
#define **ZCL_MIRROR_SYNC_PERIOD_ATTRIBUTE_ID**
#define **ZCL_MIRROR_LAST_SYNC_TIME_ATTRIBUTE_ID**
#define **ZCL_PRICE_INCREASE_RANDOMIZE_MINUTES_ATTRIBUTE_ID**
#define **ZCL_PRICE_DECREASE_RANDOMIZE_MINUTES_ATTRIBUTE_ID**
#define **ZCL_COMMODITY_TYPE_CLIENT_ATTRIBUTE_ID**
#define **ZCL_TIER1_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_TIER2_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_TIER3_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_TIER4_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_TIER5_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_TIER6_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_TIER7_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_TIER8_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_TIER9_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_TIER10_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_TIER11_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_TIER12_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_TIER13_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_TIER14_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_TIER15_PRICE_LABEL_ATTRIBUTE_ID**
#define **ZCL_BLOCK1_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_BLOCK2_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_BLOCK3_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_BLOCK4_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_BLOCK5_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_BLOCK6_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_BLOCK7_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_BLOCK8_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_BLOCK9_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_BLOCK10_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_BLOCK11_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_BLOCK12_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_BLOCK13_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_BLOCK14_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_BLOCK15_THRESHOLD_ATTRIBUTE_ID**
#define **ZCL_START_OF_BLOCK_PERIOD_ATTRIBUTE_ID**
#define **ZCL_BLOCK_PERIOD_DURATION_MINUTES_ATTRIBUTE_ID**
#define **ZCL_THRESHOLD_MULTIPLIER_ATTRIBUTE_ID**

#define **ZCL_THRESHOLD_DIVISOR_ATTRIBUTE_ID**
#define **ZCL_COMMODITY_TYPE_SERVER_ATTRIBUTE_ID**
#define **ZCL_STANDING_CHARGE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK1_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK2_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK3_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK4_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK5_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK6_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK7_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK8_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK9_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK10_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK11_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK12_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK13_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK14_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK15_PRICE_ATTRIBUTE_ID**
#define **ZCL_NO_TIER_BLOCK16_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK1_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK2_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK3_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK4_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK5_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK6_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK7_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK8_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK9_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK10_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK11_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK12_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK13_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK14_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK15_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER1_BLOCK16_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK1_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK2_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK3_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK4_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK5_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK6_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK7_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK8_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK9_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK10_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK11_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK12_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK13_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK14_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK15_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER2_BLOCK16_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER3_BLOCK1_PRICE_ATTRIBUTE_ID**

```
#define ZCL_TIER3_BLOCK2_PRICE_ATTRIBUTE_ID
#define ZCL_TIER3_BLOCK3_PRICE_ATTRIBUTE_ID
#define ZCL_TIER3_BLOCK4_PRICE_ATTRIBUTE_ID
#define ZCL_TIER3_BLOCK5_PRICE_ATTRIBUTE_ID
#define ZCL_TIER3_BLOCK6_PRICE_ATTRIBUTE_ID
#define ZCL_TIER3_BLOCK7_PRICE_ATTRIBUTE_ID
#define ZCL_TIER3_BLOCK8_PRICE_ATTRIBUTE_ID
#define ZCL_TIER3_BLOCK9_PRICE_ATTRIBUTE_ID
#define ZCL_TIER3_BLOCK10_PRICE_ATTRIBUTE_ID
#define ZCL_TIER3_BLOCK11_PRICE_ATTRIBUTE_ID
#define ZCL_TIER3_BLOCK12_PRICE_ATTRIBUTE_ID
#define ZCL_TIER3_BLOCK13_PRICE_ATTRIBUTE_ID
#define ZCL_TIER3_BLOCK14_PRICE_ATTRIBUTE_ID
#define ZCL_TIER3_BLOCK15_PRICE_ATTRIBUTE_ID
#define ZCL_TIER3_BLOCK16_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK1_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK2_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK3_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK4_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK5_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK6_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK7_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK8_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK9_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK10_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK11_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK12_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK13_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK14_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK15_PRICE_ATTRIBUTE_ID
#define ZCL_TIER4_BLOCK16_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK1_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK2_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK3_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK4_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK5_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK6_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK7_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK8_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK9_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK10_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK11_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK12_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK13_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK14_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK15_PRICE_ATTRIBUTE_ID
#define ZCL_TIER5_BLOCK16_PRICE_ATTRIBUTE_ID
#define ZCL_TIER6_BLOCK1_PRICE_ATTRIBUTE_ID
#define ZCL_TIER6_BLOCK2_PRICE_ATTRIBUTE_ID
#define ZCL_TIER6_BLOCK3_PRICE_ATTRIBUTE_ID
#define ZCL_TIER6_BLOCK4_PRICE_ATTRIBUTE_ID
#define ZCL_TIER6_BLOCK5_PRICE_ATTRIBUTE_ID
```

#define **ZCL_TIER6_BLOCK6_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER6_BLOCK7_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER6_BLOCK8_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER6_BLOCK9_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER6_BLOCK10_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER6_BLOCK11_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER6_BLOCK12_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER6_BLOCK13_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER6_BLOCK14_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER6_BLOCK15_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER6_BLOCK16_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK1_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK2_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK3_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK4_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK5_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK6_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK7_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK8_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK9_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK10_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK11_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK12_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK13_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK14_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK15_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER7_BLOCK16_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK1_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK2_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK3_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK4_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK5_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK6_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK7_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK8_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK9_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK10_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK11_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK12_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK13_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK14_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK15_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER8_BLOCK16_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER9_BLOCK1_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER9_BLOCK2_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER9_BLOCK3_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER9_BLOCK4_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER9_BLOCK5_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER9_BLOCK6_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER9_BLOCK7_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER9_BLOCK8_PRICE_ATTRIBUTE_ID**
#define **ZCL_TIER9_BLOCK9_PRICE_ATTRIBUTE_ID**

```
#define ZCL_TIER9_BLOCK10_PRICE_ATTRIBUTE_ID
#define ZCL_TIER9_BLOCK11_PRICE_ATTRIBUTE_ID
#define ZCL_TIER9_BLOCK12_PRICE_ATTRIBUTE_ID
#define ZCL_TIER9_BLOCK13_PRICE_ATTRIBUTE_ID
#define ZCL_TIER9_BLOCK14_PRICE_ATTRIBUTE_ID
#define ZCL_TIER9_BLOCK15_PRICE_ATTRIBUTE_ID
#define ZCL_TIER9_BLOCK16_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK1_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK2_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK3_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK4_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK5_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK6_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK7_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK8_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK9_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK10_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK11_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK12_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK13_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK14_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK15_PRICE_ATTRIBUTE_ID
#define ZCL_TIER10_BLOCK16_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK1_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK2_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK3_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK4_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK5_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK6_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK7_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK8_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK9_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK10_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK11_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK12_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK13_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK14_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK15_PRICE_ATTRIBUTE_ID
#define ZCL_TIER11_BLOCK16_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK1_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK2_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK3_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK4_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK5_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK6_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK7_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK8_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK9_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK10_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK11_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK12_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK13_PRICE_ATTRIBUTE_ID
```

```
#define ZCL_TIER12_BLOCK14_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK15_PRICE_ATTRIBUTE_ID
#define ZCL_TIER12_BLOCK16_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK1_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK2_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK3_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK4_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK5_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK6_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK7_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK8_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK9_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK10_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK11_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK12_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK13_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK14_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK15_PRICE_ATTRIBUTE_ID
#define ZCL_TIER13_BLOCK16_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK1_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK2_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK3_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK4_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK5_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK6_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK7_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK8_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK9_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK10_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK11_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK12_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK13_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK14_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK15_PRICE_ATTRIBUTE_ID
#define ZCL_TIER14_BLOCK16_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK1_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK2_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK3_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK4_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK5_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK6_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK7_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK8_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK9_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK10_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK11_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK12_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK13_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK14_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK15_PRICE_ATTRIBUTE_ID
#define ZCL_TIER15_BLOCK16_PRICE_ATTRIBUTE_ID
#define ZCL_CLOSE_TUNNEL_TIMEOUT_ATTRIBUTE_ID
```

#define **ZCL_PAYMENT_CONTROL_ATTRIBUTE_ID**
#define **ZCL_CREDIT_REMAINING_ATTRIBUTE_ID**
#define **ZCL_EMERGENCY_CREDIT_REMAINING_ATTRIBUTE_ID**
#define **ZCL_CREDIT_STATUS_ATTRIBUTE_ID**
#define **ZCL_TOP_UP_DATE_TIME_1_ATTRIBUTE_ID**
#define **ZCL_TOP_UP_AMOUNT_1_ATTRIBUTE_ID**
#define **ZCL_ORIGINATING_DEVICE_1_ATTRIBUTE_ID**
#define **ZCL_TOP_UP_DATE_TIME_2_ATTRIBUTE_ID**
#define **ZCL_TOP_UP_AMOUNT_2_ATTRIBUTE_ID**
#define **ZCL_ORIGINATING_DEVICE_2_ATTRIBUTE_ID**
#define **ZCL_TOP_UP_DATE_TIME_3_ATTRIBUTE_ID**
#define **ZCL_TOP_UP_AMOUNT_3_ATTRIBUTE_ID**
#define **ZCL_ORIGINATING_DEVICE_3_ATTRIBUTE_ID**
#define **ZCL_TOP_UP_DATE_TIME_4_ATTRIBUTE_ID**
#define **ZCL_TOP_UP_AMOUNT_4_ATTRIBUTE_ID**
#define **ZCL_ORIGINATING_DEVICE_4_ATTRIBUTE_ID**
#define **ZCL_TOP_UP_DATE_TIME_5_ATTRIBUTE_ID**
#define **ZCL_TOP_UP_AMOUNT_5_ATTRIBUTE_ID**
#define **ZCL_ORIGINATING_DEVICE_5_ATTRIBUTE_ID**
#define **ZCL_FUEL_DEBT_REMAINING_ATTRIBUTE_ID**
#define **ZCL_FUEL_DEBT_RECOVERY_RATE_ATTRIBUTE_ID**
#define **ZCL_FUEL_DEBT_RECOVERY_PERIOD_ATTRIBUTE_ID**
#define **ZCL_NON_FUEL_DEBT_REMAINING_ATTRIBUTE_ID**
#define **ZCL_NON_FUEL_DEBT_RECOVERY_RATE_ATTRIBUTE_ID**
#define **ZCL_NON_FUEL_DEBT_RECOVERY_PERIOD_ATTRIBUTE_ID**
#define **ZCL_PROPOSED_CHANGE_PROVIDER_ID_ATTRIBUTE_ID**
#define **ZCL_PROPOSED_CHANGE_IMPLEMENTATION_TIME_ATTRIBUTE_ID**
#define **ZCL_PROPOSED_CHANGE_SUPPLY_STATUS_ATTRIBUTE_ID**
#define **ZCL_DELAYED_SUPPLY_INTERRUPT_VALUE_REMAINING_ATTRIBUTE_ID**
#define **ZCL_DELAYED_SUPPLY_INTERRUPT_VALUE_TYPE_ATTRIBUTE_ID**
#define **ZCL_KEY_ESTABLISHMENT_SUITE_CLIENT_ATTRIBUTE_ID**
#define **ZCL_KEY_ESTABLISHMENT_SUITE_SERVER_ATTRIBUTE_ID**
#define **ZCL_ATTRIBUTE_ONE_ATTRIBUTE_ID**

Define Documentation

#define ZCL_VERSION_ATTRIBUTE_ID

Definition at line **18** of file **attribute-id.h**.

#define ZCL_APPLICATION_VERSION_ATTRIBUTE_ID

Definition at line **19** of file **attribute-id.h**.

#define ZCL_STACK_VERSION_ATTRIBUTE_ID

Definition at line **20** of file **attribute-id.h**.

#define ZCL_HW_VERSION_ATTRIBUTE_ID

Definition at line **21** of file **attribute-id.h**.

#define ZCL_MANUFACTURER_NAME_ATTRIBUTE_ID

Definition at line **22** of file **attribute-id.h**.

#define ZCL_MODEL_IDENTIFIER_ATTRIBUTE_ID

Definition at line **23** of file **attribute-id.h**.

#define ZCL_DATE_CODE_ATTRIBUTE_ID

Definition at line **24** of file **attribute-id.h**.

#define ZCL_POWER_SOURCE_ATTRIBUTE_ID

Definition at line **25** of file **attribute-id.h**.

#define ZCL_LOCATION_DESCRIPTION_ATTRIBUTE_ID

Definition at line **26** of file **attribute-id.h**.

#define ZCL_PHYSICAL_ENVIRONMENT_ATTRIBUTE_ID

Definition at line **27** of file **attribute-id.h**.

#define ZCL_DEVICE_ENABLED_ATTRIBUTE_ID

Definition at line **28** of file **attribute-id.h**.

#define ZCL_ALARM_MASK_ATTRIBUTE_ID

Definition at line **29** of file **attribute-id.h**.

#define ZCL_DISABLE_LOCAL_CONFIG_ATTRIBUTE_ID

Definition at line **30** of file **attribute-id.h**.

#define ZCL_MAINS_VOLTAGE_ATTRIBUTE_ID

Definition at line **37** of file **attribute-id.h**.

#define ZCL_MAINS_FREQUENCY_ATTRIBUTE_ID

Definition at line **38** of file **attribute-id.h**.

#define ZCL_MAINS_ALARM_MASK_ATTRIBUTE_ID

Definition at line **39** of file **attribute-id.h**.

#define ZCL_MAINS_VOLTAGE_MIN_THRESHOLD_ATTRIBUTE_ID

Definition at line **40** of file **attribute-id.h**.

#define ZCL_MAINS_VOLTAGE_MAX_THRESHOLD_ATTRIBUTE_ID

Definition at line **41** of file **attribute-id.h**.

#define ZCL_MAINS_VOLTAGE_DWELL_TRIP_POINT_ATTRIBUTE_ID

Definition at line **42** of file **attribute-id.h**.

#define ZCL_BATTERY_VOLTAGE_ATTRIBUTE_ID

Definition at line **43** of file **attribute-id.h**.

#define ZCL_BATTERY_MANUFACTURER_ATTRIBUTE_ID

Definition at line **44** of file **attribute-id.h**.

#define ZCL_BATTERY_SIZE_ATTRIBUTE_ID

Definition at line **45** of file **attribute-id.h**.

#define ZCL_BATTERY_AHR_RATING_ATTRIBUTE_ID

Definition at line **46** of file **attribute-id.h**.

#define ZCL_BATTERY_QUANTITY_ATTRIBUTE_ID

Definition at line **47** of file **attribute-id.h**.

#define ZCL_BATTERY_RATED_VOLTAGE_ATTRIBUTE_ID

Definition at line **48** of file **attribute-id.h**.

#define ZCL_BATTERY_ALARM_MASK_ATTRIBUTE_ID

Definition at line **49** of file **attribute-id.h**.

#define ZCL_BATTERY_VOLTAGE_MIN_THRESHOLD_ATTRIBUTE_ID

Definition at line **50** of file **attribute-id.h**.

#define ZCL_CURRENT_TEMPERATURE_ATTRIBUTE_ID

Definition at line **57** of file **attribute-id.h**.

#define ZCL_MIN_TEMP_EXPERIENCED_ATTRIBUTE_ID

Definition at line **58** of file **attribute-id.h**.

#define ZCL_MAX_TEMP_EXPERIENCED_ATTRIBUTE_ID

Definition at line **59** of file **attribute-id.h**.

#define ZCL_OVER_TEMP_TOTAL_DWELL_ATTRIBUTE_ID

Definition at line **60** of file **attribute-id.h**.

#define ZCL_DEVICE_TEMP_ALARM_MASK_ATTRIBUTE_ID

Definition at line **61** of file **attribute-id.h**.

#define ZCL_LOW_TEMP_THRESHOLD_ATTRIBUTE_ID

Definition at line **62** of file **attribute-id.h**.

#define ZCL_HIGH_TEMP_THRESHOLD_ATTRIBUTE_ID

Definition at line **63** of file **attribute-id.h**.

#define ZCL_LOW_TEMP_DWELL_TRIP_POINT_ATTRIBUTE_ID

Definition at line **64** of file **attribute-id.h**.

#define ZCL_HIGH_TEMP_DWELL_TRIP_POINT_ATTRIBUTE_ID

Definition at line **65** of file **attribute-id.h**.

#define ZCL_IDENTIFY_TIME_ATTRIBUTE_ID

Definition at line **72** of file **attribute-id.h**.

#define ZCL_GROUP_NAME_SUPPORT_ATTRIBUTE_ID

Definition at line **79** of file **attribute-id.h**.

#define ZCL_SCENE_COUNT_ATTRIBUTE_ID

Definition at line **86** of file **attribute-id.h**.

#define ZCL_CURRENT_SCENE_ATTRIBUTE_ID

Definition at line **87** of file **attribute-id.h**.

#define ZCL_CURRENT_GROUP_ATTRIBUTE_ID

Definition at line **88** of file **attribute-id.h**.

#define ZCL_SCENE_VALID_ATTRIBUTE_ID

Definition at line **89** of file **attribute-id.h**.

#define ZCL_SCENE_NAME_SUPPORT_ATTRIBUTE_ID

Definition at line **90** of file **attribute-id.h**.

#define ZCL_LAST_CONFIGURED_BY_ATTRIBUTE_ID

Definition at line **91** of file **attribute-id.h**.

#define ZCL_ON_OFF_ATTRIBUTE_ID

Definition at line **98** of file **attribute-id.h**.

#define ZCL_SAMPLE_MFG_SPECIFIC_TRANSITION_TIME_ATTRIBUTE_ID

Definition at line **99** of file **attribute-id.h**.

#define ZCL_SWITCH_TYPE_ATTRIBUTE_ID

Definition at line **106** of file **attribute-id.h**.

#define ZCL_SWITCH_ACTIONS_ATTRIBUTE_ID

Definition at line **107** of file **attribute-id.h**.

#define ZCL_CURRENT_LEVEL_ATTRIBUTE_ID

Definition at line **114** of file **attribute-id.h**.

#define ZCL_LEVEL_CONTROL_REMAINING_TIME_ATTRIBUTE_ID

Definition at line **115** of file **attribute-id.h**.

#define ZCL_ON_OFF_TRANSITION_TIME_ATTRIBUTE_ID

Definition at line **116** of file **attribute-id.h**.

#define ZCL_ON_LEVEL_ATTRIBUTE_ID

Definition at line **117** of file **attribute-id.h**.

#define ZCL_ALARM_COUNT_ATTRIBUTE_ID

Definition at line **124** of file **attribute-id.h**.

#define ZCL_TIME_ATTRIBUTE_ID

Definition at line **131** of file **attribute-id.h**.

#define ZCL_TIME_STATUS_ATTRIBUTE_ID

Definition at line **132** of file **attribute-id.h**.

#define ZCL_TIME_ZONE_ATTRIBUTE_ID

Definition at line **133** of file **attribute-id.h**.

#define ZCL_DST_START_ATTRIBUTE_ID

Definition at line **134** of file **attribute-id.h**.

#define ZCL_DST_END_ATTRIBUTE_ID

Definition at line **135** of file **attribute-id.h**.

#define ZCL_DST_SHIFT_ATTRIBUTE_ID

Definition at line **136** of file **attribute-id.h**.

#define ZCL_STANDARD_TIME_ATTRIBUTE_ID

Definition at line **137** of file **attribute-id.h**.

#define ZCL_LOCAL_TIME_ATTRIBUTE_ID

Definition at line **138** of file **attribute-id.h**.

#define ZCL_LAST_SET_TIME_ATTRIBUTE_ID

Definition at line **139** of file **attribute-id.h**.

#define ZCL_VALID_UNTIL_TIME_ATTRIBUTE_ID

Definition at line **140** of file **attribute-id.h**.

#define ZCL_LOCATION_TYPE_ATTRIBUTE_ID

Definition at line **147** of file **attribute-id.h**.

#define ZCL_LOCATION_METHOD_ATTRIBUTE_ID

Definition at line **148** of file **attribute-id.h**.

#define ZCL_LOCATION_AGE_ATTRIBUTE_ID

Definition at line **149** of file **attribute-id.h**.

#define ZCL_QUALITY_MEASURE_ATTRIBUTE_ID

Definition at line **150** of file **attribute-id.h**.

#define ZCL_NUMBER_OF_DEVICES_ATTRIBUTE_ID

Definition at line **151** of file **attribute-id.h**.

#define ZCL_COORDINATE1_ATTRIBUTE_ID

Definition at line **152** of file **attribute-id.h**.

#define ZCL_COORDINATE2_ATTRIBUTE_ID

Definition at line **153** of file **attribute-id.h**.

#define ZCL_COORDINATE3_ATTRIBUTE_ID

Definition at line **154** of file **attribute-id.h**.

#define ZCL_POWER_ATTRIBUTE_ID

Definition at line **155** of file **attribute-id.h**.

#define ZCL_PATH_LOSS_EXPONENT_ATTRIBUTE_ID

Definition at line **156** of file **attribute-id.h**.

#define ZCL_REPORTING_PERIOD_ATTRIBUTE_ID

Definition at line **157** of file **attribute-id.h**.

#define ZCL_CALCULATION_PERIOD_ATTRIBUTE_ID

Definition at line **158** of file **attribute-id.h**.

#define ZCL_NUMBER_RSSI_MEASUREMENTS_ATTRIBUTE_ID

Definition at line **159** of file **attribute-id.h**.

#define ZCL_ACTIVE_TEXT_ATTRIBUTE_ID

Definition at line **166** of file **attribute-id.h**.

#define ZCL_DESCRIPTION_ATTRIBUTE_ID

Definition at line **167** of file **attribute-id.h**.

#define ZCL_INACTIVE_TEXT_ATTRIBUTE_ID

Definition at line **168** of file **attribute-id.h**.

#define ZCL_OUT_OF_SERVICE_ATTRIBUTE_ID

Definition at line **169** of file **attribute-id.h**.

#define ZCL_POLARITY_ATTRIBUTE_ID

Definition at line **170** of file **attribute-id.h**.

#define ZCL_PRESENT_VALUE_ATTRIBUTE_ID

Definition at line **171** of file **attribute-id.h**.

#define ZCL_RELIABILITY_ATTRIBUTE_ID

Definition at line **172** of file **attribute-id.h**.

#define ZCL_STATUS_FLAGS_ATTRIBUTE_ID

Definition at line **173** of file **attribute-id.h**.

#define ZCL_APPLICATION_TYPE_ATTRIBUTE_ID

Definition at line **174** of file **attribute-id.h**.

#define ZCL_SHORT_ADDRESS_ATTRIBUTE_ID

Definition at line **181** of file **attribute-id.h**.

#define ZCL_EXTENDED_PAN_ID_ATTRIBUTE_ID

Definition at line **182** of file **attribute-id.h**.

#define ZCL_PAN_ID_ATTRIBUTE_ID

Definition at line **183** of file **attribute-id.h**.

#define ZCL_CHANNEL_MASK_ATTRIBUTE_ID

Definition at line **184** of file **attribute-id.h**.

#define ZCL_PROTOCOL_VERSION_ATTRIBUTE_ID

Definition at line **185** of file **attribute-id.h**.

#define ZCL_STACK_PROFILE_ATTRIBUTE_ID

Definition at line **186** of file **attribute-id.h**.

#define ZCL_STARTUP_CONTROL_ATTRIBUTE_ID

Definition at line **187** of file **attribute-id.h**.

#define ZCL_TRUST_CENTER_ADDRESS_ATTRIBUTE_ID

Definition at line **188** of file **attribute-id.h**.

#define ZCL_TRUST_CENTER_MASTER_KEY_ATTRIBUTE_ID

Definition at line **189** of file **attribute-id.h**.

#define ZCL_NETWORK_KEY_ATTRIBUTE_ID

Definition at line **190** of file **attribute-id.h**.

#define ZCL_USE_INSECURE_JOIN_ATTRIBUTE_ID

Definition at line **191** of file **attribute-id.h**.

#define ZCL_PRECONFIGURED_LINK_KEY_ATTRIBUTE_ID

Definition at line **192** of file **attribute-id.h**.

#define ZCL_NETWORK_KEY_SEQUENCE_NUMBER_ATTRIBUTE_ID

Definition at line **193** of file **attribute-id.h**.

#define ZCL_NETWORK_KEY_TYPE_ATTRIBUTE_ID

Definition at line **194** of file **attribute-id.h**.

#define ZCL_NETWORK_MANAGER_ADDRESS_ATTRIBUTE_ID

Definition at line **195** of file **attribute-id.h**.

#define ZCL_SCAN_ATTEMPTS_ATTRIBUTE_ID

Definition at line **196** of file **attribute-id.h**.

#define ZCL_TIME_BETWEEN_SCANS_ATTRIBUTE_ID

Definition at line **197** of file **attribute-id.h**.

#define ZCL_REJOIN_INTERVAL_ATTRIBUTE_ID

Definition at line **198** of file **attribute-id.h**.

#define ZCL_MAX_REJOIN_INTERVAL_ATTRIBUTE_ID

Definition at line **199** of file **attribute-id.h**.

#define ZCL_INDERECT_POLL_RATE_ATTRIBUTE_ID

Definition at line **200** of file **attribute-id.h**.

#define ZCL_PARENT_RETRY_THRESHOLD_ATTRIBUTE_ID

Definition at line **201** of file **attribute-id.h**.

#define ZCL_CONCENTRATOR_FLAG_ATTRIBUTE_ID

Definition at line **202** of file **attribute-id.h**.

#define ZCL_CONCENTRATOR_RADIUS_ATTRIBUTE_ID

Definition at line **203** of file **attribute-id.h**.

#define ZCL_CONCENTRATOR_DISCOVERY_TIME_ATTRIBUTE_ID

Definition at line **204** of file **attribute-id.h**.

#define ZCL_SHADE_CONFIG_PHYSICAL_CLOSED_LIMIT_ATTRIBUTE_ID

Definition at line **211** of file **attribute-id.h**.

#define ZCL_SHADE_CONFIG_MOTOR_STEP_SIZE_ATTRIBUTE_ID

Definition at line **212** of file **attribute-id.h**.

#define ZCL_SHADE_CONFIG_STATUS_ATTRIBUTE_ID

Definition at line **213** of file **attribute-id.h**.

#define ZCL_SHADE_CONFIG_CLOSED_LIMIT_ATTRIBUTE_ID

Definition at line **214** of file **attribute-id.h**.

#define ZCL_SHADE_CONFIG_MODE_ATTRIBUTE_ID

Definition at line **215** of file **attribute-id.h**.

#define ZCL_MAX_PRESSURE_ATTRIBUTE_ID

Definition at line **222** of file **attribute-id.h**.

#define ZCL_MAX_SPEED_ATTRIBUTE_ID

Definition at line **223** of file **attribute-id.h**.

#define ZCL_MAX_FLOW_ATTRIBUTE_ID

Definition at line **224** of file **attribute-id.h**.

#define ZCL_MIN_CONST_PRESSURE_ATTRIBUTE_ID

Definition at line **225** of file **attribute-id.h**.

#define ZCL_MAX_CONST_PRESSURE_ATTRIBUTE_ID

Definition at line **226** of file **attribute-id.h**.

#define ZCL_MIN_COMP_PRESSURE_ATTRIBUTE_ID

Definition at line **227** of file **attribute-id.h**.

#define ZCL_MAX_COMP_PRESSURE_ATTRIBUTE_ID

Definition at line **228** of file **attribute-id.h**.

#define ZCL_MIN_CONST_SPEED_ATTRIBUTE_ID

Definition at line **229** of file **attribute-id.h**.

#define ZCL_MAX_CONST_SPEED_ATTRIBUTE_ID

Definition at line **230** of file **attribute-id.h**.

#define ZCL_MIN_CONST_FLOW_ATTRIBUTE_ID

Definition at line **231** of file **attribute-id.h**.

#define ZCL_MAX_CONST_FLOW_ATTRIBUTE_ID

Definition at line **232** of file **attribute-id.h**.

#define ZCL_MIN_CONST_TEMP_ATTRIBUTE_ID

Definition at line **233** of file **attribute-id.h**.

#define ZCL_MAX_CONST_TEMP_ATTRIBUTE_ID

Definition at line **234** of file **attribute-id.h**.

#define ZCL_PUMP_STATUS_ATTRIBUTE_ID

Definition at line **235** of file **attribute-id.h**.

#define ZCL_EFFECTIVE_OPERATION_MODE_ATTRIBUTE_ID

Definition at line **236** of file **attribute-id.h**.

#define ZCL_EFFECTIVE_CONTROL_MODE_ATTRIBUTE_ID

Definition at line **237** of file **attribute-id.h**.

#define ZCL_CAPACITY_ATTRIBUTE_ID

Definition at line **238** of file **attribute-id.h**.

#define ZCL_SPEED_ATTRIBUTE_ID

Definition at line **239** of file **attribute-id.h**.

#define ZCL_LIFETIME_RUNNING_HOURS_ATTRIBUTE_ID

Definition at line **240** of file **attribute-id.h**.

#define ZCL_PUMP_POWER_ATTRIBUTE_ID

Definition at line **241** of file **attribute-id.h**.

#define ZCL_LIFETIME_ENERGY_CONSUMED_ATTRIBUTE_ID

Definition at line **242** of file **attribute-id.h**.

#define ZCL_OPERATION_MODE_ATTRIBUTE_ID

Definition at line **243** of file **attribute-id.h**.

#define ZCL_CONTROL_MODE_ATTRIBUTE_ID

Definition at line **244** of file **attribute-id.h**.

#define ZCL_PUMP_ALARM_MASK_ATTRIBUTE_ID

Definition at line **245** of file **attribute-id.h**.

#define ZCL_LOCAL_TEMPERATURE_ATTRIBUTE_ID

Definition at line **252** of file **attribute-id.h**.

#define ZCL_OUTDOOR_TEMPERATURE_ATTRIBUTE_ID

Definition at line **253** of file **attribute-id.h**.

#define ZCL_THERMOSTAT_OCCUPANCY_ATTRIBUTE_ID

Definition at line **254** of file **attribute-id.h**.

#define ZCL_ABS_MIN_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID

Definition at line **255** of file **attribute-id.h**.

#define ZCL_ABS_MAX_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID

Definition at line **256** of file **attribute-id.h**.

#define ZCL_ABS_MIN_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID

Definition at line **257** of file **attribute-id.h**.

#define ZCL_ABS_MAX_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID

Definition at line **258** of file **attribute-id.h**.

#define ZCL_PI_COOLING_DEMAND_ATTRIBUTE_ID

Definition at line **259** of file **attribute-id.h**.

#define ZCL_PI_HEATING_DEMAND_ATTRIBUTE_ID

Definition at line **260** of file **attribute-id.h**.

#define ZCL_LOCAL_TEMPERATURE_CALIBRATION_ATTRIBUTE_ID

Definition at line **261** of file **attribute-id.h**.

#define ZCL_OCCUPIED_COOLING_SETPOINT_ATTRIBUTE_ID

Definition at line **262** of file **attribute-id.h**.

#define ZCL_OCCUPIED_HEATING_SETPOINT_ATTRIBUTE_ID

Definition at line **263** of file **attribute-id.h**.

#define ZCL_UNOCCUPIED_COOLING_SETPOINT_ATTRIBUTE_ID

Definition at line **264** of file **attribute-id.h**.

#define ZCL_UNOCCUPIED_HEATING_SETPOINT_ATTRIBUTE_ID

Definition at line **265** of file **attribute-id.h**.

#define ZCL_MIN_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID

Definition at line **266** of file **attribute-id.h**.

#define ZCL_MAX_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID

Definition at line **267** of file **attribute-id.h**.

#define ZCL_MIN_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID

Definition at line **268** of file **attribute-id.h**.

#define ZCL_MAX_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID

Definition at line **269** of file **attribute-id.h**.

#define ZCL_MIN_SETPOINT_DEAD_BAND_ATTRIBUTE_ID

Definition at line **270** of file **attribute-id.h**.

#define ZCL_REMOTE_SENSING_ATTRIBUTE_ID

Definition at line **271** of file **attribute-id.h**.

#define ZCL_CONTROL_SEQUENCE_OF_OPERATION_ATTRIBUTE_ID

Definition at line **272** of file **attribute-id.h**.

#define ZCL_SYSTEM_MODE_ATTRIBUTE_ID

Definition at line **273** of file **attribute-id.h**.

#define ZCL_THERMOSTAT_ALARM_MASK_ATTRIBUTE_ID

Definition at line **274** of file **attribute-id.h**.

#define ZCL_FAN_CONTROL_FAN_MODE_ATTRIBUTE_ID

Definition at line **281** of file **attribute-id.h**.

#define ZCL_FAN_CONTROL_FAN_MODE_SEQUENCE_ATTRIBUTE_ID

Definition at line **282** of file **attribute-id.h**.

#define ZCL_RELATIVE_HUMIDITY_ATTRIBUTE_ID

Definition at line **289** of file **attribute-id.h**.

#define ZCL_DEHUMIDIFICATION_COOLING_ATTRIBUTE_ID

Definition at line **290** of file **attribute-id.h**.

#define ZCL_RH_DEHUMIDIFICATION_SETPOINT_ATTRIBUTE_ID

Definition at line **291** of file **attribute-id.h**.

#define ZCL_RELATIVE_HUMIDITY_MODE_ATTRIBUTE_ID

Definition at line **292** of file **attribute-id.h**.

#define ZCL_DEHUMIDIFICATION_LOCKOUT_ATTRIBUTE_ID

Definition at line **293** of file **attribute-id.h**.

#define ZCL_DEHUMIDIFICATION_HYSTERESIS_ATTRIBUTE_ID

Definition at line **294** of file **attribute-id.h**.

#define ZCL_DEHUMIDIFICATION_MAX_COOL_ATTRIBUTE_ID

Definition at line **295** of file **attribute-id.h**.

#define ZCL_RELATIVE_HUMIDITY_DISPLAY_ATTRIBUTE_ID

Definition at line **296** of file **attribute-id.h**.

#define ZCL_TEMPERATURE_DISPLAY_MODE_ATTRIBUTE_ID

Definition at line **303** of file **attribute-id.h**.

#define ZCL_KEYPAD_LOCKOUT_ATTRIBUTE_ID

Definition at line **304** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_CURRENT_HUE_ATTRIBUTE_ID

Definition at line **311** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_CURRENT_SATURATION_ATTRIBUTE_ID

Definition at line **312** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_REMAINING_TIME_ATTRIBUTE_ID

Definition at line **313** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_CURRENT_X_ATTRIBUTE_ID

Definition at line **314** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_CURRENT_Y_ATTRIBUTE_ID

Definition at line **315** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_DRIFT_COMPENSATION_ATTRIBUTE_ID

Definition at line **316** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_COMPENSATION_TEXT_ATTRIBUTE_ID

Definition at line **317** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_COLOR_TEMPERATURE_ATTRIBUTE_ID

Definition at line **318** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_COLOR_MODE_ATTRIBUTE_ID

Definition at line **319** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_NUMBER_OF_PRIMARIES_ATTRIBUTE_ID

Definition at line **320** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_1_X_ATTRIBUTE_ID

Definition at line **321** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_1_Y_ATTRIBUTE_ID

Definition at line **322** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_1_INTENSITY_ATTRIBUTE_ID

Definition at line **323** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_2_X_ATTRIBUTE_ID

Definition at line **324** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_2_Y_ATTRIBUTE_ID

Definition at line **325** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_2_INTENSITY_ATTRIBUTE_ID

Definition at line **326** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_3_X_ATTRIBUTE_ID

Definition at line **327** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_3_Y_ATTRIBUTE_ID

Definition at line **328** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_3_INTENSITY_ATTRIBUTE_ID

Definition at line **329** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_4_X_ATTRIBUTE_ID

Definition at line **330** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_4_Y_ATTRIBUTE_ID

Definition at line **331** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_4_INTENSITY_ATTRIBUTE_ID

Definition at line **332** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_5_X_ATTRIBUTE_ID

Definition at line **333** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_5_Y_ATTRIBUTE_ID

Definition at line **334** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_5_INTENSITY_ATTRIBUTE_ID

Definition at line **335** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_6_X_ATTRIBUTE_ID

Definition at line **336** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_6_Y_ATTRIBUTE_ID

Definition at line **337** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_PRIMARY_6_INTENSITY_ATTRIBUTE_ID

Definition at line **338** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_WHITE_POINT_X_ATTRIBUTE_ID

Definition at line **339** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_WHITE_POINT_Y_ATTRIBUTE_ID

Definition at line **340** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_COLOR_POINT_R_X_ATTRIBUTE_ID

Definition at line **341** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_COLOR_POINT_R_Y_ATTRIBUTE_ID

Definition at line **342** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_COLOR_POINT_R_INTENSITY_ATTRIBUTE_ID

Definition at line **343** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_COLOR_POINT_G_X_ATTRIBUTE_ID

Definition at line **344** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_COLOR_POINT_G_Y_ATTRIBUTE_ID

Definition at line **345** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_COLOR_POINT_G_INTENSITY_ATTRIBUTE_ID

Definition at line **346** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_COLOR_POINT_B_X_ATTRIBUTE_ID

Definition at line **347** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_COLOR_POINT_B_Y_ATTRIBUTE_ID

Definition at line **348** of file **attribute-id.h**.

#define ZCL_COLOR_CONTROL_COLOR_POINT_B_INTENSITY_ATTRIBUTE_ID

Definition at line **349** of file **attribute-id.h**.

#define ZCL_PHYSICAL_MIN_LEVEL_ATTRIBUTE_ID

Definition at line **356** of file **attribute-id.h**.

#define ZCL_PHYSICAL_MAX_LEVEL_ATTRIBUTE_ID

Definition at line **357** of file **attribute-id.h**.

#define ZCL_BALLAST_STATUS_ATTRIBUTE_ID

Definition at line **358** of file **attribute-id.h**.

#define ZCL_MIN_LEVEL_ATTRIBUTE_ID

Definition at line **359** of file **attribute-id.h**.

#define ZCL_MAX_LEVEL_ATTRIBUTE_ID

Definition at line **360** of file **attribute-id.h**.

#define ZCL_POWER_ON_LEVEL_ATTRIBUTE_ID

Definition at line **361** of file **attribute-id.h**.

#define ZCL_POWER_ON_FADE_TIME_ATTRIBUTE_ID

Definition at line **362** of file **attribute-id.h**.

#define ZCL_INTRINSIC_BALLAST_FACTOR_ATTRIBUTE_ID

Definition at line **363** of file **attribute-id.h**.

#define ZCL_BALLAST_FACTOR_ADJUSTMENT_ATTRIBUTE_ID

Definition at line **364** of file **attribute-id.h**.

#define ZCL_LAMP_QUALITY_ATTRIBUTE_ID

Definition at line **365** of file **attribute-id.h**.

#define ZCL_LAMP_TYPE_ATTRIBUTE_ID

Definition at line **366** of file **attribute-id.h**.

#define ZCL_LAMP_MANUFACTURER_ATTRIBUTE_ID

Definition at line **367** of file **attribute-id.h**.

#define ZCL_LAMP_RATED_HOURS_ATTRIBUTE_ID

Definition at line **368** of file **attribute-id.h**.

#define ZCL_LAMP_BURN_HOURS_ATTRIBUTE_ID

Definition at line **369** of file **attribute-id.h**.

#define ZCL_LAMP_ALARM_MODE_ATTRIBUTE_ID

Definition at line **370** of file **attribute-id.h**.

#define ZCL_LAMP_BURN_HOURS_TRIP_POINT_ATTRIBUTE_ID

Definition at line **371** of file **attribute-id.h**.

#define ZCL_ILLUM_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **378** of file **attribute-id.h**.

#define ZCL_ILLUM_MIN_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **379** of file **attribute-id.h**.

#define ZCL_ILLUM_MAX_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **380** of file **attribute-id.h**.

#define ZCL_ILLUM_TOLERANCE_ATTRIBUTE_ID

Definition at line **381** of file **attribute-id.h**.

#define ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_ATTRIBUTE_ID

Definition at line **382** of file **attribute-id.h**.

#define ZCL_LEVEL_STATUS_ATTRIBUTE_ID

Definition at line **389** of file **attribute-id.h**.

#define ZCL_SENSING_LIGHT_SENSOR_TYPE_ATTRIBUTE_ID

Definition at line **390** of file **attribute-id.h**.

#define ZCL_ILLUMINANCE_TARGET_LEVEL_ATTRIBUTE_ID

Definition at line **391** of file **attribute-id.h**.

#define ZCL_TEMP_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **398** of file **attribute-id.h**.

#define ZCL_TEMP_MIN_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **399** of file **attribute-id.h**.

#define ZCL_TEMP_MAX_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **400** of file **attribute-id.h**.

#define ZCL_TEMP_TOLERANCE_ATTRIBUTE_ID

Definition at line **401** of file **attribute-id.h**.

#define ZCL_PRESSURE_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **408** of file **attribute-id.h**.

#define ZCL_PRESSURE_MIN_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **409** of file **attribute-id.h**.

#define ZCL_PRESSURE_MAX_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **410** of file **attribute-id.h**.

#define ZCL_PRESSURE_TOLERANCE_ATTRIBUTE_ID

Definition at line **411** of file **attribute-id.h**.

#define ZCL_PRESSURE_SCALED_VALUE_ATTRIBUTE_ID

Definition at line **412** of file **attribute-id.h**.

#define ZCL_PRESSURE_MIN_SCALED_VALUE_ATTRIBUTE_ID

Definition at line **413** of file **attribute-id.h**.

#define ZCL_PRESSURE_MAX_SCALED_VALUE_ATTRIBUTE_ID

Definition at line **414** of file **attribute-id.h**.

#define ZCL_PRESSURE_SCALED_TOLERANCE_ATTRIBUTE_ID

Definition at line **415** of file **attribute-id.h**.

#define ZCL_PRESSURE_SCALE_ATTRIBUTE_ID

Definition at line **416** of file **attribute-id.h**.

#define ZCL_FLOW_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **423** of file **attribute-id.h**.

#define ZCL_FLOW_MIN_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **424** of file **attribute-id.h**.

#define ZCL_FLOW_MAX_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **425** of file **attribute-id.h**.

#define ZCL_FLOW_TOLERANCE_ATTRIBUTE_ID

Definition at line **426** of file **attribute-id.h**.

#define ZCL_RELATIVE_HUMIDITY_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **433** of file **attribute-id.h**.

#define ZCL_RELATIVE_HUMIDITY_MIN_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **434** of file **attribute-id.h**.

#define ZCL_RELATIVE_HUMIDITY_MAX_MEASURED_VALUE_ATTRIBUTE_ID

Definition at line **435** of file **attribute-id.h**.

#define ZCL_RELATIVE_HUMIDITY_TOLERANCE_ATTRIBUTE_ID

Definition at line **436** of file **attribute-id.h**.

#define ZCL_OCCUPANCY_ATTRIBUTE_ID

Definition at line **443** of file **attribute-id.h**.

#define ZCL_OCCUPANCY_SENSOR_TYPE_ATTRIBUTE_ID

Definition at line **444** of file **attribute-id.h**.

#define ZCL_PIR_OCCUPIED_TO_UNOCCUPIED_DELAY_ATTRIBUTE_ID

Definition at line **445** of file **attribute-id.h**.

#define ZCL_PIR_UNOCCUPIED_TO_OCCUPIED_DELAY_ATTRIBUTE_ID

Definition at line **446** of file **attribute-id.h**.

#define ZCL_PIR_UNOCCUPIED_TO_OCCUPIED_THRESHOLD_ATTRIBUTE_ID

Definition at line **447** of file **attribute-id.h**.

#define ZCL_ULTRASONIC_OCCUPIED_TO_UNOCCUPIED_DELAY_ATTRIBUTE_ID

Definition at line **448** of file **attribute-id.h**.

#define ZCL_ULTRASONIC_UNOCCUPIED_TO_OCCUPIED_DELAY_ATTRIBUTE_ID

Definition at line **449** of file **attribute-id.h**.

#define ZCL_ULTRASONIC_UNOCCUPIED_TO_OCCUPIED_THRESHOLD_ATTRIBUTE_ID

Definition at line **450** of file **attribute-id.h**.

#define ZCL_ZONE_STATE_ATTRIBUTE_ID

Definition at line **457** of file **attribute-id.h**.

#define ZCL_ZONE_TYPE_ATTRIBUTE_ID

Definition at line **458** of file **attribute-id.h**.

#define ZCL_ZONE_STATUS_ATTRIBUTE_ID

Definition at line **459** of file **attribute-id.h**.

#define ZCL_IAS_CIE_ADDRESS_ATTRIBUTE_ID

Definition at line **460** of file **attribute-id.h**.

#define ZCL_MAX_DURATION_ATTRIBUTE_ID

Definition at line **467** of file **attribute-id.h**.

#define ZCL_LOCK_STATE_ATTRIBUTE_ID

Definition at line **474** of file **attribute-id.h**.

#define ZCL_LOCK_TYPE_ATTRIBUTE_ID

Definition at line **475** of file **attribute-id.h**.

#define ZCL_ACTUATOR_ENABLED_ATTRIBUTE_ID

Definition at line **476** of file **attribute-id.h**.

#define ZCL_DOOR_STATE_ATTRIBUTE_ID

Definition at line **477** of file **attribute-id.h**.

#define ZCL_DOOR_OPEN_EVENTS_ATTRIBUTE_ID

Definition at line **478** of file **attribute-id.h**.

#define ZCL_DOOR_CLOSED_EVENTS_ATTRIBUTE_ID

Definition at line **479** of file **attribute-id.h**.

#define ZCL_OPEN_PERIOD_ATTRIBUTE_ID

Definition at line **480** of file **attribute-id.h**.

#define ZCL_COVERING_TYPE_ATTRIBUTE_ID

Definition at line **487** of file **attribute-id.h**.

#define ZCL_LIMIT_LIFT_ATTRIBUTE_ID

Definition at line **488** of file **attribute-id.h**.

#define ZCL_LIMIT_TILT_ATTRIBUTE_ID

Definition at line **489** of file **attribute-id.h**.

#define ZCL_CURRENT_LIFT_ATTRIBUTE_ID

Definition at line **490** of file **attribute-id.h**.

#define ZCL_CURRENT_TILT_ATTRIBUTE_ID

Definition at line **491** of file **attribute-id.h**.

#define ZCL_NUMBER_LIFT_ATTRIBUTE_ID

Definition at line **492** of file **attribute-id.h**.

#define ZCL_NUMBER_TILT_ATTRIBUTE_ID

Definition at line **493** of file **attribute-id.h**.

#define ZCL_CONFIG_STATUS_ATTRIBUTE_ID

Definition at line **494** of file **attribute-id.h**.

#define ZCL_CURRENT_LIFT_PERCENTAGE_ATTRIBUTE_ID

Definition at line **495** of file **attribute-id.h**.

#define ZCL_CURRENT_TILT_PERCENTAGE_ATTRIBUTE_ID

Definition at line **496** of file **attribute-id.h**.

#define ZCL_OPEN_LIMIT_LIFT_ATTRIBUTE_ID

Definition at line **497** of file **attribute-id.h**.

#define ZCL_CLOSED_LIMIT_LIFT_ATTRIBUTE_ID

Definition at line **498** of file **attribute-id.h**.

#define ZCL_OPEN_LIMIT_TILT_ATTRIBUTE_ID

Definition at line **499** of file **attribute-id.h**.

#define ZCL_CLOSED_LIMIT_TILT_ATTRIBUTE_ID

Definition at line **500** of file **attribute-id.h**.

#define ZCL_VELOCITY_LIFT_ATTRIBUTE_ID

Definition at line **501** of file **attribute-id.h**.

#define ZCL_ACCELERATION_LIFT_ATTRIBUTE_ID

Definition at line **502** of file **attribute-id.h**.

#define ZCL_DECELERATION_LIFT_ATTRIBUTE_ID

Definition at line **503** of file **attribute-id.h**.

#define ZCL_MODE_ATTRIBUTE_ID

Definition at line **504** of file **attribute-id.h**.

#define ZCL_SETPOINTS_LIFT_ATTRIBUTE_ID

Definition at line **505** of file **attribute-id.h**.

#define ZCL_SETPOINTS_TILT_ATTRIBUTE_ID

Definition at line **506** of file **attribute-id.h**.

#define ZCL_MAXIMUM_INCOMING_TRANSFER_SIZE_ATTRIBUTE_ID

Definition at line **513** of file **attribute-id.h**.

#define ZCL_MAXIMUM_OUTGOING_TRANSFER_SIZE_ATTRIBUTE_ID

Definition at line **514** of file **attribute-id.h**.

#define ZCL_PROTOCOL_ADDRESS_ATTRIBUTE_ID

Definition at line **515** of file **attribute-id.h**.

#define ZCL_UPGRADE_SERVER_ID_ATTRIBUTE_ID

Definition at line **522** of file **attribute-id.h**.

#define ZCL_FILE_OFFSET_ATTRIBUTE_ID

Definition at line **523** of file **attribute-id.h**.

#define ZCL_CURRENT_FILE_VERSION_ATTRIBUTE_ID

Definition at line **524** of file **attribute-id.h**.

#define ZCL_CURRENT_ZIGBEE_STACK_VERSION_ATTRIBUTE_ID

Definition at line **525** of file **attribute-id.h**.

#define ZCL_DOWNLOADED_FILE_VERSION_ATTRIBUTE_ID

Definition at line **526** of file **attribute-id.h**.

#define ZCL_DOWNLOADED_ZIGBEE_STACK_VERSION_ATTRIBUTE_ID

Definition at line **527** of file **attribute-id.h**.

#define ZCL_IMAGE_UPGRADE_STATUS_ATTRIBUTE_ID

Definition at line **528** of file **attribute-id.h**.

#define ZCL_MANUFACTURER_ID_ATTRIBUTE_ID

Definition at line **529** of file **attribute-id.h**.

#define ZCL_IMAGE_TYPE_ID_ATTRIBUTE_ID

Definition at line **530** of file **attribute-id.h**.

#define ZCL_UTILITY_ENROLLMENT_GROUP_ATTRIBUTE_ID

Definition at line **537** of file **attribute-id.h**.

#define ZCL_START_RANDOMIZE_MINUTES_ATTRIBUTE_ID

Definition at line **538** of file **attribute-id.h**.

#define ZCL_STOP_RANDOMIZE_MINUTES_ATTRIBUTE_ID

Definition at line **539** of file **attribute-id.h**.

#define ZCL_DEVICE_CLASS_VALUE_ATTRIBUTE_ID

Definition at line **540** of file **attribute-id.h**.

#define ZCL_CURRENT_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **547** of file **attribute-id.h**.

#define ZCL_CURRENT_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **548** of file **attribute-id.h**.

#define ZCL_CURRENT_MAX_DEMAND_DELIVERED_ATTRIBUTE_ID

Definition at line **549** of file **attribute-id.h**.

#define ZCL_CURRENT_MAX_DEMAND_RECEIVED_ATTRIBUTE_ID

Definition at line **550** of file **attribute-id.h**.

#define ZCL_DFT_SUMMATION_ATTRIBUTE_ID

Definition at line **551** of file **attribute-id.h**.

#define ZCL_DAILY_FREEZE_TIME_ATTRIBUTE_ID

Definition at line **552** of file **attribute-id.h**.

#define ZCL_POWER_FACTOR_ATTRIBUTE_ID

Definition at line **553** of file **attribute-id.h**.

#define ZCL_READING_SNAP_SHOT_TIME_ATTRIBUTE_ID

Definition at line **554** of file **attribute-id.h**.

#define ZCL_CURRENT_MAX_DEMAND_DELIVERED_TIME_ATTRIBUTE_ID

Definition at line **555** of file **attribute-id.h**.

#define ZCL_CURRENT_MAX_DEMAND_RECEIVED_TIME_ATTRIBUTE_ID

Definition at line **556** of file **attribute-id.h**.

#define ZCL_DEFAULT_UPDATE_PERIOD_ATTRIBUTE_ID

Definition at line **557** of file **attribute-id.h**.

#define ZCL_FAST_POLL_UPDATE_PERIOD_ATTRIBUTE_ID

Definition at line **558** of file **attribute-id.h**.

#define ZCL_CURRENT_BLOCK_PERIOD_CONSUMPTION_DELIVERED_ATTRIBUTE_ID

Definition at line **559** of file **attribute-id.h**.

#define ZCL_DAILY_CONSUMPTION_TARGET_ATTRIBUTE_ID

Definition at line **560** of file **attribute-id.h**.

#define ZCL_CURRENT_BLOCK_ATTRIBUTE_ID

Definition at line **561** of file **attribute-id.h**.

#define ZCL_PROFILE_INTERVAL_PERIOD_ATTRIBUTE_ID

Definition at line **562** of file **attribute-id.h**.

#define ZCL_INTERVAL_READ_REPORTING_PERIOD_ATTRIBUTE_ID

Definition at line **563** of file **attribute-id.h**.

#define ZCL_PRESET_READING_TIME_ATTRIBUTE_ID

Definition at line **564** of file **attribute-id.h**.

#define ZCL_VOLUME_PER_REPORT_ATTRIBUTE_ID

Definition at line **565** of file **attribute-id.h**.

#define ZCL_FLOW_RESTRICTION_ATTRIBUTE_ID

Definition at line **566** of file **attribute-id.h**.

#define ZCL_SUPPLY_STATUS_ATTRIBUTE_ID

Definition at line **567** of file **attribute-id.h**.

#define ZCL_CURRENT_INLET_ENERGY_CARRIER_SUMMATION_ATTRIBUTE_ID

Definition at line **568** of file **attribute-id.h**.

#define ZCL_CURRENT_OUTLET_ENERGY_CARRIER_SUMMATION_ATTRIBUTE_ID

Definition at line **569** of file **attribute-id.h**.

#define ZCL_INLET_TEMPERATURE_ATTRIBUTE_ID

Definition at line **570** of file **attribute-id.h**.

#define ZCL_OUTLET_TEMPERATURE_ATTRIBUTE_ID

Definition at line **571** of file **attribute-id.h**.

#define ZCL_CONTROL_TEMPERATURE_ATTRIBUTE_ID

Definition at line **572** of file **attribute-id.h**.

#define ZCL_CURRENT_INLET_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID

Definition at line **573** of file **attribute-id.h**.

#define ZCL_CURRENT_OUTLET_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID

Definition at line **574** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **575** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **576** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **577** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **578** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **579** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **580** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **581** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **582** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **583** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **584** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **585** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **586** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **587** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **588** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **589** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **590** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **591** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **592** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **593** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **594** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **595** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **596** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **597** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **598** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **599** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **600** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **601** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **602** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **603** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_SUMMATION_RECEIVED_ATTRIBUTE_ID

Definition at line **604** of file **attribute-id.h**.

#define ZCL_STATUS_ATTRIBUTE_ID

Definition at line **605** of file **attribute-id.h**.

#define ZCL_REMAINING_BATTERY_LIFE_ATTRIBUTE_ID

Definition at line **606** of file **attribute-id.h**.

#define ZCL_HOURS_IN_OPERATION_ATTRIBUTE_ID

Definition at line **607** of file **attribute-id.h**.

#define ZCL_HOURS_IN_FAULT_ATTRIBUTE_ID

Definition at line **608** of file **attribute-id.h**.

#define ZCL_UNIT_OF_MEASURE_ATTRIBUTE_ID

Definition at line **609** of file **attribute-id.h**.

#define ZCL_MULTIPLIER_ATTRIBUTE_ID

Definition at line **610** of file **attribute-id.h**.

#define ZCL_DIVISOR_ATTRIBUTE_ID

Definition at line **611** of file **attribute-id.h**.

#define ZCL_SUMMATION_FORMATTING_ATTRIBUTE_ID

Definition at line **612** of file **attribute-id.h**.

#define ZCL_DEMAND_FORMATTING_ATTRIBUTE_ID

Definition at line **613** of file **attribute-id.h**.

#define ZCL_HISTORICAL_CONSUMPTION_FORMATTING_ATTRIBUTE_ID

Definition at line **614** of file **attribute-id.h**.

#define ZCL_METERING_DEVICE_TYPE_ATTRIBUTE_ID

Definition at line **615** of file **attribute-id.h**.

#define ZCL_SITE_ID_ATTRIBUTE_ID

Definition at line **616** of file **attribute-id.h**.

#define ZCL_METER_SERIAL_NUMBER_ATTRIBUTE_ID

Definition at line **617** of file **attribute-id.h**.

#define ZCL_ENERGY_CARRIER_UNIT_OF_MEASURE_ATTRIBUTE_ID

Definition at line **618** of file **attribute-id.h**.

#define ZCL_ENERGY_CARRIER_SUMMATION_FORMATTING_ATTRIBUTE_ID

Definition at line **619** of file **attribute-id.h**.

#define ZCL_ENERGY_CARRIER_DEMAND_FORMATTING_ATTRIBUTE_ID

Definition at line **620** of file **attribute-id.h**.

#define ZCL_TEMPERATURE_UNIT_OF_MEASURE_ATTRIBUTE_ID

Definition at line **621** of file **attribute-id.h**.

#define ZCL_TEMPERATURE_FORMATTING_ATTRIBUTE_ID

Definition at line **622** of file **attribute-id.h**.

#define ZCL_INSTANTANEOUS_DEMAND_ATTRIBUTE_ID

Definition at line **623** of file **attribute-id.h**.

#define ZCL_CURRENT_DAY_CONSUMPTION_DELIVERED_ATTRIBUTE_ID

Definition at line **624** of file **attribute-id.h**.

#define ZCL_CURRENT_DAY_CONSUMPTION_RECEIVED_ATTRIBUTE_ID

Definition at line **625** of file **attribute-id.h**.

#define ZCL_PREVIOUS_DAY_CONSUMPTION_DELIVERED_ATTRIBUTE_ID

Definition at line **626** of file **attribute-id.h**.

#define ZCL_PREVIOUS_DAY_CONSUMPTION_RECEIVED_ATTRIBUTE_ID

Definition at line **627** of file **attribute-id.h**.

#define ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_START_TIME_DELIVERED_ATTRIBUTE_ID

Definition at line **628** of file **attribute-id.h**.

#define ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_START_TIME_RECEIVED_ATTRIBUTE_ID

Definition at line **629** of file **attribute-id.h**.

#define ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_VALUE_DELIVERED_ATTRIBUTE_ID

Definition at line **630** of file **attribute-id.h**.

#define ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_VALUE_RECEIVED_ATTRIBUTE_ID

Definition at line **631** of file **attribute-id.h**.

#define ZCL_CURRENT_DAY_MAX_PRESSURE_ATTRIBUTE_ID

Definition at line **632** of file **attribute-id.h**.

#define ZCL_CURRENT_DAY_MIN_PRESSURE_ATTRIBUTE_ID

Definition at line **633** of file **attribute-id.h**.

#define ZCL_PREVIOUS_DAY_MAX_PRESSURE_ATTRIBUTE_ID

Definition at line **634** of file **attribute-id.h**.

#define ZCL_PREVIOUS_DAY_MIN_PRESSURE_ATTRIBUTE_ID

Definition at line **635** of file **attribute-id.h**.

#define ZCL_CURRENT_DAY_MAX_DEMAND_ATTRIBUTE_ID

Definition at line **636** of file **attribute-id.h**.

#define ZCL_PREVIOUS_DAY_MAX_DEMAND_ATTRIBUTE_ID

Definition at line **637** of file **attribute-id.h**.

#define ZCL_CURRENT_MONTH_MAX_DEMAND_ATTRIBUTE_ID

Definition at line **638** of file **attribute-id.h**.

#define ZCL_CURRENT_YEAR_MAX_DEMAND_ATTRIBUTE_ID

Definition at line **639** of file **attribute-id.h**.

#define ZCL_CURRENT_DAY_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID

Definition at line **640** of file **attribute-id.h**.

#define ZCL_PREVIOUS_DAY_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID

Definition at line **641** of file **attribute-id.h**.

#define ZCL_CURRENT_MONTH_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID

Definition at line **642** of file **attribute-id.h**.

#define ZCL_CURRENT_MONTH_MIN_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID

Definition at line **643** of file **attribute-id.h**.

#define ZCL_CURRENT_YEAR_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID

Definition at line **644** of file **attribute-id.h**.

#define ZCL_CURRENT_YEAR_MIN_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID

Definition at line **645** of file **attribute-id.h**.

#define ZCL_MAX_NUMBER_OF_PERIODS_DELIVERED_ATTRIBUTE_ID

Definition at line **646** of file **attribute-id.h**.

#define ZCL_CURRENT_DEMAND_DELIVERED_ATTRIBUTE_ID

Definition at line **647** of file **attribute-id.h**.

#define ZCL_DEMAND_LIMIT_ATTRIBUTE_ID

Definition at line **648** of file **attribute-id.h**.

#define ZCL_DEMAND_INTEGRATION_PERIOD_ATTRIBUTE_ID

Definition at line **649** of file **attribute-id.h**.

#define ZCL_NUMBER_OF_DEMAND_SUBINTERVALS_ATTRIBUTE_ID

Definition at line **650** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **651** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **652** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **653** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **654** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **655** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **656** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **657** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **658** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **659** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **660** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **661** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **662** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **663** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **664** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **665** of file **attribute-id.h**.

#define ZCL_CURRENT_NO_TIER_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **666** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **667** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **668** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **669** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **670** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **671** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **672** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **673** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **674** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **675** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **676** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **677** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **678** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **679** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **680** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **681** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER1_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **682** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **683** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **684** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **685** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **686** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **687** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **688** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **689** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **690** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **691** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **692** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **693** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **694** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **695** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **696** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **697** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER2_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **698** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **699** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **700** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **701** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **702** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **703** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **704** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **705** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **706** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **707** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **708** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **709** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **710** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **711** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **712** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **713** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER3_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **714** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **715** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **716** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **717** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **718** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **719** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **720** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **721** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **722** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **723** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **724** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **725** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **726** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **727** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **728** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **729** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER4_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **730** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **731** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **732** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **733** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **734** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **735** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **736** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **737** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **738** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **739** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **740** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **741** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **742** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **743** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **744** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **745** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER5_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **746** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **747** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **748** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **749** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **750** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **751** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **752** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **753** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **754** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **755** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **756** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **757** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **758** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **759** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **760** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **761** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER6_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **762** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **763** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **764** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **765** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **766** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **767** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **768** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **769** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **770** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **771** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **772** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **773** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **774** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **775** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **776** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **777** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER7_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **778** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **779** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **780** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **781** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **782** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **783** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **784** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **785** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **786** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **787** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **788** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **789** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **790** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **791** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **792** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **793** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER8_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **794** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **795** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **796** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **797** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **798** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **799** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **800** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **801** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **802** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **803** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **804** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **805** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **806** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **807** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **808** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **809** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER9_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **810** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **811** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **812** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **813** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **814** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **815** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **816** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **817** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **818** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **819** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **820** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **821** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **822** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **823** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **824** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **825** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER10_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **826** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **827** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **828** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **829** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **830** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **831** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **832** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **833** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **834** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **835** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **836** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **837** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **838** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **839** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **840** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **841** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER11_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **842** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **843** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **844** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **845** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **846** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **847** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **848** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **849** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **850** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **851** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **852** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **853** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **854** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **855** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **856** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **857** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER12_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **858** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **859** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **860** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **861** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **862** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **863** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **864** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **865** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **866** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **867** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **868** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **869** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **870** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **871** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **872** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **873** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER13_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **874** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **875** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **876** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **877** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **878** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **879** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **880** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **881** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **882** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **883** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **884** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **885** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **886** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **887** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **888** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **889** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER14_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **890** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **891** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **892** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **893** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **894** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **895** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **896** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **897** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **898** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **899** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **900** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **901** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **902** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **903** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **904** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **905** of file **attribute-id.h**.

#define ZCL_CURRENT_TIER15_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID

Definition at line **906** of file **attribute-id.h**.

#define ZCL_GENERIC_ALARM_MASK_ATTRIBUTE_ID

Definition at line **907** of file **attribute-id.h**.

#define ZCL_ELECTRICITY_ALARM_MASK_ATTRIBUTE_ID

Definition at line **908** of file **attribute-id.h**.

#define ZCL_GENERIC_FLOW_PRESSURE_ALARM_MASK_ATTRIBUTE_ID

Definition at line **909** of file **attribute-id.h**.

#define ZCL_WATER_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID

Definition at line **910** of file **attribute-id.h**.

#define ZCL_HEAT_AND_COOLING_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID

Definition at line **911** of file **attribute-id.h**.

#define ZCL_GAS_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID

Definition at line **912** of file **attribute-id.h**.

#define ZCL_MIRROR_IEEE_ADDRESS_ATTRIBUTE_ID

Definition at line **913** of file **attribute-id.h**.

#define ZCL_MIRROR_SYNC_PERIOD_ATTRIBUTE_ID

Definition at line **914** of file **attribute-id.h**.

#define ZCL_MIRROR_LAST_SYNC_TIME_ATTRIBUTE_ID

Definition at line **915** of file **attribute-id.h**.

#define ZCL_PRICE_INCREASE_RANDOMIZE_MINUTES_ATTRIBUTE_ID

Definition at line **922** of file **attribute-id.h**.

#define ZCL_PRICE_DECREASE_RANDOMIZE_MINUTES_ATTRIBUTE_ID

Definition at line **923** of file **attribute-id.h**.

#define ZCL_COMMODITY_TYPE_CLIENT_ATTRIBUTE_ID

Definition at line **924** of file **attribute-id.h**.

#define ZCL_TIER1_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **927** of file **attribute-id.h**.

#define ZCL_TIER2_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **928** of file **attribute-id.h**.

#define ZCL_TIER3_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **929** of file **attribute-id.h**.

#define ZCL_TIER4_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **930** of file **attribute-id.h**.

#define ZCL_TIER5_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **931** of file **attribute-id.h**.

#define ZCL_TIER6_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **932** of file **attribute-id.h**.

#define ZCL_TIER7_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **933** of file **attribute-id.h**.

#define ZCL_TIER8_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **934** of file **attribute-id.h**.

#define ZCL_TIER9_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **935** of file **attribute-id.h**.

#define ZCL_TIER10_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **936** of file **attribute-id.h**.

#define ZCL_TIER11_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **937** of file **attribute-id.h**.

#define ZCL_TIER12_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **938** of file **attribute-id.h**.

#define ZCL_TIER13_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **939** of file **attribute-id.h**.

#define ZCL_TIER14_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **940** of file **attribute-id.h**.

#define ZCL_TIER15_PRICE_LABEL_ATTRIBUTE_ID

Definition at line **941** of file **attribute-id.h**.

#define ZCL_BLOCK1_THRESHOLD_ATTRIBUTE_ID

Definition at line **942** of file **attribute-id.h**.

#define ZCL_BLOCK2_THRESHOLD_ATTRIBUTE_ID

Definition at line **943** of file **attribute-id.h**.

#define ZCL_BLOCK3_THRESHOLD_ATTRIBUTE_ID

Definition at line **944** of file **attribute-id.h**.

#define ZCL_BLOCK4_THRESHOLD_ATTRIBUTE_ID

Definition at line **945** of file **attribute-id.h**.

#define ZCL_BLOCK5_THRESHOLD_ATTRIBUTE_ID

Definition at line **946** of file **attribute-id.h**.

#define ZCL_BLOCK6_THRESHOLD_ATTRIBUTE_ID

Definition at line **947** of file **attribute-id.h**.

#define ZCL_BLOCK7_THRESHOLD_ATTRIBUTE_ID

Definition at line **948** of file **attribute-id.h**.

#define ZCL_BLOCK8_THRESHOLD_ATTRIBUTE_ID

Definition at line **949** of file **attribute-id.h**.

#define ZCL_BLOCK9_THRESHOLD_ATTRIBUTE_ID

Definition at line **950** of file **attribute-id.h**.

#define ZCL_BLOCK10_THRESHOLD_ATTRIBUTE_ID

Definition at line **951** of file **attribute-id.h**.

#define ZCL_BLOCK11_THRESHOLD_ATTRIBUTE_ID

Definition at line **952** of file **attribute-id.h**.

#define ZCL_BLOCK12_THRESHOLD_ATTRIBUTE_ID

Definition at line **953** of file **attribute-id.h**.

#define ZCL_BLOCK13_THRESHOLD_ATTRIBUTE_ID

Definition at line **954** of file **attribute-id.h**.

#define ZCL_BLOCK14_THRESHOLD_ATTRIBUTE_ID

Definition at line **955** of file **attribute-id.h**.

#define ZCL_BLOCK15_THRESHOLD_ATTRIBUTE_ID

Definition at line **956** of file **attribute-id.h**.

#define ZCL_START_OF_BLOCK_PERIOD_ATTRIBUTE_ID

Definition at line **957** of file **attribute-id.h**.

#define ZCL_BLOCK_PERIOD_DURATION_MINUTES_ATTRIBUTE_ID

Definition at line **958** of file **attribute-id.h**.

#define ZCL_THRESHOLD_MULTIPLIER_ATTRIBUTE_ID

Definition at line **959** of file **attribute-id.h**.

#define ZCL_THRESHOLD_DIVISOR_ATTRIBUTE_ID

Definition at line **960** of file **attribute-id.h**.

#define ZCL_COMMODITY_TYPE_SERVER_ATTRIBUTE_ID

Definition at line **961** of file **attribute-id.h**.

#define ZCL_STANDING_CHARGE_ATTRIBUTE_ID

Definition at line **962** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **963** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **964** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **965** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **966** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **967** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **968** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **969** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **970** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **971** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **972** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **973** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **974** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **975** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **976** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **977** of file **attribute-id.h**.

#define ZCL_NO_TIER_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **978** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **979** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **980** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **981** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **982** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **983** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **984** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **985** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **986** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **987** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **988** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **989** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **990** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **991** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **992** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **993** of file **attribute-id.h**.

#define ZCL_TIER1_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **994** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **995** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **996** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **997** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **998** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **999** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **1000** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **1001** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **1002** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **1003** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **1004** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **1005** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **1006** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **1007** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **1008** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **1009** of file **attribute-id.h**.

#define ZCL_TIER2_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **1010** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **1011** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **1012** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **1013** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **1014** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **1015** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **1016** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **1017** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **1018** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **1019** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **1020** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **1021** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **1022** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **1023** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **1024** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **1025** of file **attribute-id.h**.

#define ZCL_TIER3_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **1026** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **1027** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **1028** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **1029** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **1030** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **1031** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **1032** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **1033** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **1034** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **1035** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **1036** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **1037** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **1038** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **1039** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **1040** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **1041** of file **attribute-id.h**.

#define ZCL_TIER4_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **1042** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **1043** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **1044** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **1045** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **1046** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **1047** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **1048** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **1049** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **1050** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **1051** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **1052** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **1053** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **1054** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **1055** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **1056** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **1057** of file **attribute-id.h**.

#define ZCL_TIER5_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **1058** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **1059** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **1060** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **1061** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **1062** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **1063** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **1064** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **1065** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **1066** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **1067** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **1068** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **1069** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **1070** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **1071** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **1072** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **1073** of file **attribute-id.h**.

#define ZCL_TIER6_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **1074** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **1075** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **1076** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **1077** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **1078** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **1079** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **1080** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **1081** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **1082** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **1083** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **1084** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **1085** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **1086** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **1087** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **1088** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **1089** of file **attribute-id.h**.

#define ZCL_TIER7_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **1090** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **1091** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **1092** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **1093** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **1094** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **1095** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **1096** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **1097** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **1098** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **1099** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **1100** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **1101** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **1102** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **1103** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **1104** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **1105** of file **attribute-id.h**.

#define ZCL_TIER8_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **1106** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **1107** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **1108** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **1109** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **1110** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **1111** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **1112** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **1113** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **1114** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **1115** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **1116** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **1117** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **1118** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **1119** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **1120** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **1121** of file **attribute-id.h**.

#define ZCL_TIER9_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **1122** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **1123** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **1124** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **1125** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **1126** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **1127** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **1128** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **1129** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **1130** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **1131** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **1132** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **1133** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **1134** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **1135** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **1136** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **1137** of file **attribute-id.h**.

#define ZCL_TIER10_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **1138** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **1139** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **1140** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **1141** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **1142** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **1143** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **1144** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **1145** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **1146** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **1147** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **1148** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **1149** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **1150** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **1151** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **1152** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **1153** of file **attribute-id.h**.

#define ZCL_TIER11_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **1154** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **1155** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **1156** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **1157** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **1158** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **1159** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **1160** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **1161** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **1162** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **1163** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **1164** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **1165** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **1166** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **1167** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **1168** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **1169** of file **attribute-id.h**.

#define ZCL_TIER12_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **1170** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **1171** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **1172** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **1173** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **1174** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **1175** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **1176** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **1177** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **1178** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **1179** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **1180** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **1181** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **1182** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **1183** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **1184** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **1185** of file **attribute-id.h**.

#define ZCL_TIER13_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **1186** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **1187** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **1188** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **1189** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **1190** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **1191** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **1192** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **1193** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **1194** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **1195** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **1196** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **1197** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **1198** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **1199** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **1200** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **1201** of file **attribute-id.h**.

#define ZCL_TIER14_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **1202** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK1_PRICE_ATTRIBUTE_ID

Definition at line **1203** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK2_PRICE_ATTRIBUTE_ID

Definition at line **1204** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK3_PRICE_ATTRIBUTE_ID

Definition at line **1205** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK4_PRICE_ATTRIBUTE_ID

Definition at line **1206** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK5_PRICE_ATTRIBUTE_ID

Definition at line **1207** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK6_PRICE_ATTRIBUTE_ID

Definition at line **1208** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK7_PRICE_ATTRIBUTE_ID

Definition at line **1209** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK8_PRICE_ATTRIBUTE_ID

Definition at line **1210** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK9_PRICE_ATTRIBUTE_ID

Definition at line **1211** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK10_PRICE_ATTRIBUTE_ID

Definition at line **1212** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK11_PRICE_ATTRIBUTE_ID

Definition at line **1213** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK12_PRICE_ATTRIBUTE_ID

Definition at line **1214** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK13_PRICE_ATTRIBUTE_ID

Definition at line **1215** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK14_PRICE_ATTRIBUTE_ID

Definition at line **1216** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK15_PRICE_ATTRIBUTE_ID

Definition at line **1217** of file **attribute-id.h**.

#define ZCL_TIER15_BLOCK16_PRICE_ATTRIBUTE_ID

Definition at line **1218** of file **attribute-id.h**.

#define ZCL_CLOSE_TUNNEL_TIMEOUT_ATTRIBUTE_ID

Definition at line **1225** of file **attribute-id.h**.

#define ZCL_PAYMENT_CONTROL_ATTRIBUTE_ID

Definition at line **1232** of file **attribute-id.h**.

#define ZCL_CREDIT_REMAINING_ATTRIBUTE_ID

Definition at line **1233** of file **attribute-id.h**.

#define ZCL_EMERGENCY_CREDIT_REMAINING_ATTRIBUTE_ID

Definition at line **1234** of file **attribute-id.h**.

#define ZCL_CREDIT_STATUS_ATTRIBUTE_ID

Definition at line **1235** of file **attribute-id.h**.

#define ZCL_TOP_UP_DATE_TIME_1_ATTRIBUTE_ID

Definition at line **1236** of file **attribute-id.h**.

#define ZCL_TOP_UP_AMOUNT_1_ATTRIBUTE_ID

Definition at line **1237** of file **attribute-id.h**.

#define ZCL_ORIGINATING_DEVICE_1_ATTRIBUTE_ID

Definition at line **1238** of file **attribute-id.h**.

#define ZCL_TOP_UP_DATE_TIME_2_ATTRIBUTE_ID

Definition at line **1239** of file **attribute-id.h**.

#define ZCL_TOP_UP_AMOUNT_2_ATTRIBUTE_ID

Definition at line **1240** of file **attribute-id.h**.

#define ZCL_ORIGINATING_DEVICE_2_ATTRIBUTE_ID

Definition at line **1241** of file **attribute-id.h**.

#define ZCL_TOP_UP_DATE_TIME_3_ATTRIBUTE_ID

Definition at line **1242** of file **attribute-id.h**.

#define ZCL_TOP_UP_AMOUNT_3_ATTRIBUTE_ID

Definition at line **1243** of file **attribute-id.h**.

#define ZCL_ORIGINATING_DEVICE_3_ATTRIBUTE_ID

Definition at line **1244** of file **attribute-id.h**.

#define ZCL_TOP_UP_DATE_TIME_4_ATTRIBUTE_ID

Definition at line **1245** of file **attribute-id.h**.

#define ZCL_TOP_UP_AMOUNT_4_ATTRIBUTE_ID

Definition at line **1246** of file **attribute-id.h**.

#define ZCL_ORIGINATING_DEVICE_4_ATTRIBUTE_ID

Definition at line **1247** of file **attribute-id.h**.

#define ZCL_TOP_UP_DATE_TIME_5_ATTRIBUTE_ID

Definition at line **1248** of file **attribute-id.h**.

#define ZCL_TOP_UP_AMOUNT_5_ATTRIBUTE_ID

Definition at line **1249** of file **attribute-id.h**.

#define ZCL_ORIGINATING_DEVICE_5_ATTRIBUTE_ID

Definition at line **1250** of file **attribute-id.h**.

#define ZCL_FUEL_DEBT_REMAINING_ATTRIBUTE_ID

Definition at line **1251** of file **attribute-id.h**.

#define ZCL_FUEL_DEBT_RECOVERY_RATE_ATTRIBUTE_ID

Definition at line **1252** of file **attribute-id.h**.

#define ZCL_FUEL_DEBT_RECOVERY_PERIOD_ATTRIBUTE_ID

Definition at line **1253** of file **attribute-id.h**.

#define ZCL_NON_FUEL_DEBT_REMAINING_ATTRIBUTE_ID

Definition at line **1254** of file **attribute-id.h**.

#define ZCL_NON_FUEL_DEBT_RECOVERY_RATE_ATTRIBUTE_ID

Definition at line **1255** of file **attribute-id.h**.

#define ZCL_NON_FUEL_DEBT_RECOVERY_PERIOD_ATTRIBUTE_ID

Definition at line **1256** of file **attribute-id.h**.

#define ZCL_PROPOSED_CHANGE_PROVIDER_ID_ATTRIBUTE_ID

Definition at line **1257** of file **attribute-id.h**.

#define ZCL_PROPOSED_CHANGE_IMPLEMENTATION_TIME_ATTRIBUTE_ID

Definition at line **1258** of file **attribute-id.h**.

#define ZCL_PROPOSED_CHANGE_SUPPLY_STATUS_ATTRIBUTE_ID

Definition at line **1259** of file **attribute-id.h**.

#define ZCL_DELAYED_SUPPLY_INTERRUPT_VALUE_REMAINING_ATTRIBUTE_ID

Definition at line **1260** of file **attribute-id.h**.

#define ZCL_DELAYED_SUPPLY_INTERRUPT_VALUE_TYPE_ATTRIBUTE_ID

Definition at line **1261** of file **attribute-id.h**.

#define ZCL_KEY_ESTABLISHMENT_SUITE_CLIENT_ATTRIBUTE_ID

Definition at line **1268** of file **attribute-id.h**.

#define ZCL_KEY_ESTABLISHMENT_SUITE_SERVER_ATTRIBUTE_ID

Definition at line **1271** of file **attribute-id.h**.

#define ZCL_ATTRIBUTE_ONE_ATTRIBUTE_ID

Definition at line **1278** of file **attribute-id.h**.

---

attribute-size.h File Reference

Go to the source code of this file.

Variables

ZCL_INT8U_ATTRIBUTE_TYPE
ZCL_CHAR_STRING_ATTRIBUTE_TYPE
ZCL_ENUM8_ATTRIBUTE_TYPE
ZCL_BOOLEAN_ATTRIBUTE_TYPE
ZCL_BITMAP8_ATTRIBUTE_TYPE
ZCL_INT16U_ATTRIBUTE_TYPE
ZCL_INT16S_ATTRIBUTE_TYPE
ZCL_INT24U_ATTRIBUTE_TYPE
ZCL_IEEE_ADDRESS_ATTRIBUTE_TYPE
ZCL_UTC_TIME_ATTRIBUTE_TYPE
ZCL_INT32S_ATTRIBUTE_TYPE
ZCL_INT32U_ATTRIBUTE_TYPE
ZCL_DATA8_ATTRIBUTE_TYPE
ZCL_BITMAP32_ATTRIBUTE_TYPE
ZCL_SECURITY_KEY_ATTRIBUTE_TYPE
ZCL_BITMAP16_ATTRIBUTE_TYPE
ZCL_INT8S_ATTRIBUTE_TYPE
ZCL_ENUM16_ATTRIBUTE_TYPE
ZCL_OCTET_STRING_ATTRIBUTE_TYPE
ZCL_INT48U_ATTRIBUTE_TYPE
ZCL_INT24S_ATTRIBUTE_TYPE

Variable Documentation

ZCL_INT8U_ATTRIBUTE_TYPE

Definition at line 10 of file attribute-size.h.

ZCL_CHAR_STRING_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_ENUM8_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_BOOLEAN_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_BITMAP8_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_INT16U_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_INT16S_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_INT24U_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_IEEE_ADDRESS_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_UTC_TIME_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_INT32S_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_INT32U_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_DATA8_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_BITMAP32_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_SECURITY_KEY_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_BITMAP16_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_INT8S_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_ENUM16_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_OCTET_STRING_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_INT48U_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

ZCL_INT24S_ATTRIBUTE_TYPE

Definition at line **10** of file **attribute-size.h**.

---

attribute-storage.h File Reference

Go to the source code of this file.

Defines

#define ATTRIBUTE_MASK_WRITABLE
#define ATTRIBUTE_MASK_TOKENIZE
#define ATTRIBUTE_MASK_MIN_MAX
#define ATTRIBUTE_MASK_MANUFACTURER_SPECIFIC
#define ATTRIBUTE_MASK_EXTERNAL_STORAGE
#define ATTRIBUTE_MASK_SINGLETON
#define ATTRIBUTE_MASK_CLIENT
#define CLUSTER_MASK_INIT_FUNCTION
#define CLUSTER_MASK_ATTRIBUTE_CHANGED_FUNCTION
#define CLUSTER_MASK_DEFAULT_RESPONSE_FUNCTION
#define CLUSTER_MASK_MESSAGE_SENT_FUNCTION
#define CLUSTER_MASK_MANUFACTURER_SPECIFIC_ATTRIBUTE_CHANGED_FUNCTION
#define CLUSTER_MASK_SERVER
#define CLUSTER_MASK_CLIENT

Define Documentation

#define ATTRIBUTE_MASK_WRITABLE

Definition at line 17 of file attribute-storage.h.

#define ATTRIBUTE_MASK_TOKENIZE

Definition at line 19 of file attribute-storage.h.

#define ATTRIBUTE_MASK_MIN_MAX

Definition at line 21 of file attribute-storage.h.

#define ATTRIBUTE_MASK_MANUFACTURER_SPECIFIC

Definition at line 23 of file attribute-storage.h.

#define ATTRIBUTE_MASK_EXTERNAL_STORAGE

Definition at line **25** of file **attribute-storage.h**.

#define ATTRIBUTE_MASK_SINGLETON

Definition at line **27** of file **attribute-storage.h**.

#define ATTRIBUTE_MASK_CLIENT

Definition at line **29** of file **attribute-storage.h**.

#define CLUSTER_MASK_INIT_FUNCTION

Definition at line **34** of file **attribute-storage.h**.

#define CLUSTER_MASK_ATTRIBUTE_CHANGED_FUNCTION

Definition at line **36** of file **attribute-storage.h**.

#define CLUSTER_MASK_DEFAULT_RESPONSE_FUNCTION

Definition at line **38** of file **attribute-storage.h**.

#define CLUSTER_MASK_MESSAGE_SENT_FUNCTION

Definition at line **40** of file **attribute-storage.h**.

#define CLUSTER_MASK_MANUFACTURER_SPECIFIC_ATTRIBUTE_CHANGED_FUNCTION

Definition at line **42** of file **attribute-storage.h**.

#define CLUSTER_MASK_SERVER

Definition at line **44** of file **attribute-storage.h**.

#define CLUSTER_MASK_CLIENT

Definition at line **46** of file **attribute-storage.h**.

---

attribute-type.h File Reference

Go to the source code of this file.

Enumerations

```
enum {
        ZCL_NO_DATA_ATTRIBUTE_TYPE,
        ZCL_DATA8_ATTRIBUTE_TYPE,
        ZCL_DATA16_ATTRIBUTE_TYPE,
        ZCL_DATA24_ATTRIBUTE_TYPE,
        ZCL_DATA32_ATTRIBUTE_TYPE,
        ZCL_DATA40_ATTRIBUTE_TYPE,
        ZCL_DATA48_ATTRIBUTE_TYPE,
        ZCL_DATA56_ATTRIBUTE_TYPE,
        ZCL_DATA64_ATTRIBUTE_TYPE,
        ZCL_BOOLEAN_ATTRIBUTE_TYPE,
        ZCL_BITMAP8_ATTRIBUTE_TYPE,
        ZCL_BITMAP16_ATTRIBUTE_TYPE,
        ZCL_BITMAP24_ATTRIBUTE_TYPE,
        ZCL_BITMAP32_ATTRIBUTE_TYPE,
        ZCL_BITMAP40_ATTRIBUTE_TYPE,
        ZCL_BITMAP48_ATTRIBUTE_TYPE,
        ZCL_BITMAP56_ATTRIBUTE_TYPE,
        ZCL_BITMAP64_ATTRIBUTE_TYPE,
        ZCL_INT8U_ATTRIBUTE_TYPE,
        ZCL_INT16U_ATTRIBUTE_TYPE,
        ZCL_INT24U_ATTRIBUTE_TYPE,
        ZCL_INT32U_ATTRIBUTE_TYPE,
        ZCL_INT40U_ATTRIBUTE_TYPE,
        ZCL_INT48U_ATTRIBUTE_TYPE,
        ZCL_INT56U_ATTRIBUTE_TYPE,
        ZCL_INT64U_ATTRIBUTE_TYPE,
        ZCL_INT8S_ATTRIBUTE_TYPE,
        ZCL_INT16S_ATTRIBUTE_TYPE,
        ZCL_INT24S_ATTRIBUTE_TYPE,
        ZCL_INT32S_ATTRIBUTE_TYPE,
        ZCL_INT40S_ATTRIBUTE_TYPE,
        ZCL_INT48S_ATTRIBUTE_TYPE,
        ZCL_INT56S_ATTRIBUTE_TYPE,
```

```
        ZCL_INT64S_ATTRIBUTE_TYPE,
        ZCL_ENUM8_ATTRIBUTE_TYPE,
        ZCL_ENUM16_ATTRIBUTE_TYPE,
        ZCL_FLOAT_SEMI_ATTRIBUTE_TYPE,
        ZCL_FLOAT_SINGLE_ATTRIBUTE_TYPE,
        ZCL_FLOAT_DOUBLE_ATTRIBUTE_TYPE,
        ZCL_OCTET_STRING_ATTRIBUTE_TYPE,
        ZCL_CHAR_STRING_ATTRIBUTE_TYPE,
        ZCL_LONG_OCTET_STRING_ATTRIBUTE_TYPE,
        ZCL_LONG_CHAR_STRING_ATTRIBUTE_TYPE,
        ZCL_ARRAY_ATTRIBUTE_TYPE,
        ZCL_STRUCT_ATTRIBUTE_TYPE,
        ZCL_SET_ATTRIBUTE_TYPE,
        ZCL_BAG_ATTRIBUTE_TYPE,
        ZCL_TIME_OF_DAY_ATTRIBUTE_TYPE,
        ZCL_DATE_ATTRIBUTE_TYPE,
        ZCL_UTC_TIME_ATTRIBUTE_TYPE,
        ZCL_CLUSTER_ID_ATTRIBUTE_TYPE,
        ZCL_ATTRIBUTE_ID_ATTRIBUTE_TYPE,
        ZCL_BACNET_OID_ATTRIBUTE_TYPE,
        ZCL_IEEE_ADDRESS_ATTRIBUTE_TYPE,
        ZCL_SECURITY_KEY_ATTRIBUTE_TYPE,
        ZCL_UNKNOWN_ATTRIBUTE_TYPE
      }
```

Enumeration Type Documentation

anonymous enum

Enumerator:

ZCL_NO_DATA_ATTRIBUTE_TYPE
ZCL_DATA8_ATTRIBUTE_TYPE
ZCL_DATA16_ATTRIBUTE_TYPE
ZCL_DATA24_ATTRIBUTE_TYPE
ZCL_DATA32_ATTRIBUTE_TYPE
ZCL_DATA40_ATTRIBUTE_TYPE
ZCL_DATA48_ATTRIBUTE_TYPE
ZCL_DATA56_ATTRIBUTE_TYPE
ZCL_DATA64_ATTRIBUTE_TYPE
ZCL_BOOLEAN_ATTRIBUTE_TYPE
ZCL_BITMAP8_ATTRIBUTE_TYPE
ZCL_BITMAP16_ATTRIBUTE_TYPE

ZCL_BITMAP24_ATTRIBUTE_TYPE
ZCL_BITMAP32_ATTRIBUTE_TYPE
ZCL_BITMAP40_ATTRIBUTE_TYPE
ZCL_BITMAP48_ATTRIBUTE_TYPE
ZCL_BITMAP56_ATTRIBUTE_TYPE
ZCL_BITMAP64_ATTRIBUTE_TYPE
ZCL_INT8U_ATTRIBUTE_TYPE
ZCL_INT16U_ATTRIBUTE_TYPE
ZCL_INT24U_ATTRIBUTE_TYPE
ZCL_INT32U_ATTRIBUTE_TYPE
ZCL_INT40U_ATTRIBUTE_TYPE
ZCL_INT48U_ATTRIBUTE_TYPE
ZCL_INT56U_ATTRIBUTE_TYPE
ZCL_INT64U_ATTRIBUTE_TYPE
ZCL_INT8S_ATTRIBUTE_TYPE
ZCL_INT16S_ATTRIBUTE_TYPE
ZCL_INT24S_ATTRIBUTE_TYPE
ZCL_INT32S_ATTRIBUTE_TYPE
ZCL_INT40S_ATTRIBUTE_TYPE
ZCL_INT48S_ATTRIBUTE_TYPE
ZCL_INT56S_ATTRIBUTE_TYPE
ZCL_INT64S_ATTRIBUTE_TYPE
ZCL_ENUM8_ATTRIBUTE_TYPE
ZCL_ENUM16_ATTRIBUTE_TYPE
ZCL_FLOAT_SEMI_ATTRIBUTE_TYPE
ZCL_FLOAT_SINGLE_ATTRIBUTE_TYPE
ZCL_FLOAT_DOUBLE_ATTRIBUTE_TYPE
ZCL_OCTET_STRING_ATTRIBUTE_TYPE
ZCL_CHAR_STRING_ATTRIBUTE_TYPE
ZCL_LONG_OCTET_STRING_ATTRIBUTE_TYPE
ZCL_LONG_CHAR_STRING_ATTRIBUTE_TYPE
ZCL_ARRAY_ATTRIBUTE_TYPE
ZCL_STRUCT_ATTRIBUTE_TYPE
ZCL_SET_ATTRIBUTE_TYPE
ZCL_BAG_ATTRIBUTE_TYPE
ZCL_TIME_OF_DAY_ATTRIBUTE_TYPE
ZCL_DATE_ATTRIBUTE_TYPE
ZCL_UTC_TIME_ATTRIBUTE_TYPE
ZCL_CLUSTER_ID_ATTRIBUTE_TYPE
ZCL_ATTRIBUTE_ID_ATTRIBUTE_TYPE
ZCL_BACNET_OID_ATTRIBUTE_TYPE
ZCL_IEEE_ADDRESS_ATTRIBUTE_TYPE
ZCL_SECURITY_KEY_ATTRIBUTE_TYPE
ZCL_UNKNOWN_ATTRIBUTE_TYPE

Definition at line **15** of file **attribute-type.h**.

---

bacnet-protocol-tunnel.h File Reference

Go to the source code of this file.

Defines

```
#define BACNET_PROTOCOL_CLUSTER_MAX_NPDU_SIZE
```

Functions

```
void bacnetProtocolTxTransferNpdu (int16u npduLength, int8u *npdu, int16u
    destShortAddress, int8u destEndpoint, int8u sourceEndpoint)
```

Define Documentation

#define BACNET_PROTOCOL_CLUSTER_MAX_NPDU_SIZE

Definition at line **15** of file **bacnet-protocol-tunnel.h**.

Function Documentation

```
void bacnetProtocolTxTransferNpdu ( int16u  npduLength,
                                    int8u * npdu,
                                    int16u  destShortAddress,
                                    int8u   destEndpoint,
                                    int8u   sourceEndpoint
                                  )
```

basic-callback.h File Reference

Go to the source code of this file.

Functions

void **emberAfPluginBasicResetToFactoryDefaultsCallback** (void)

basic.h File Reference

Go to the source code of this file.

---

call-command-handler.h File Reference

Go to the source code of this file.

Functions

EmberAfStatus emberAfBasicClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfIdentifyClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfIdentifyClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfGroupsClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfGroupsClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfScenesClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfScenesClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfOnOffClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfLevelControlClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfAlarmClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfAlarmClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfRssiLocationClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfRssiLocationClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfCommissioningClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfCommissioningClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfThermostatClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfColorControlClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfIasZoneClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfIasZoneClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfIasAceClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfIasAceClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfIasWdClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfDoorLockClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfDoorLockClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfWindowCoveringClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfGenericTunnelClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfGenericTunnelClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfBacnetProtocolTunnelClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfDemandResponseLoadControlClusterClientCommandParse (**EmberAfClusterCommand** *cmd)

EmberAfStatus emberAfDemandResponseLoadControlClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfSimpleMeteringClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfSimpleMeteringClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfPriceClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfPriceClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfMessagingClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfMessagingClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfTunnelingClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfTunnelingClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfPrepaymentClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfPrepaymentClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfKeyEstablishmentClusterClientCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfKeyEstablishmentClusterServerCommandParse (**EmberAfClusterCommand** *cmd)
EmberAfStatus emberAfSampleMfgSpecificClusterServerCommandParse (**EmberAfClusterCommand** *cmd)

Function Documentation

EmberAfStatus emberAfBasicClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfIdentifyClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfIdentifyClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfGroupsClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfGroupsClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfScenesClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfScenesClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfOnOffClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfLevelControlClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfAlarmClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfAlarmClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfRssiLocationClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfRssiLocationClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfCommissioningClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfCommissioningClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfThermostatClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfColorControlClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfIasZoneClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfIasZoneClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfIasAceClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfIasAceClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfIasWdClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfDoorLockClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfDoorLockClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfWindowCoveringClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfGenericTunnelClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfGenericTunnelClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfBacnetProtocolTunnelClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfDemandResponseLoadControlClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfDemandResponseLoadControlClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfSimpleMeteringClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfSimpleMeteringClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfPriceClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfPriceClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfMessagingClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfMessagingClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfTunnelingClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfTunnelingClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfPrepaymentClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfPrepaymentClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfKeyEstablishmentClusterClientCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfKeyEstablishmentClusterServerCommandParse (EmberAfClusterCommand * cmd)

EmberAfStatus emberAfSampleMfgSpecificClusterServerCommandParse (EmberAfClusterCommand * cmd)

---

callback.h File Reference

#include "app/framework/include/af-types.h"

Go to the source code of this file.

Non-Cluster Related Callbacks

boolean	**emberAfReadAttributesResponseCallback** (**EmberAfClusterId** clusterId, int8u *buffer, int16u bufLen)
boolean	**emberAfWriteAttributesResponseCallback** (**EmberAfClusterId** clusterId, int8u *buffer, int16u bufLen)
boolean	**emberAfConfigureReportingResponseCallback** (**EmberAfClusterId** clusterId, int8u *buffer, int16u bufLen)
boolean	**emberAfReadReportingConfigurationResponseCallback** (**EmberAfClusterId** clusterId, int8u *buffer, int16u bufLen)
boolean	**emberAfReportAttributesCallback** (**EmberAfClusterId** clusterId, int8u *buffer, int16u bufLen)
boolean	**emberAfDefaultResponseCallback** (**EmberAfClusterId** clusterId, int8u commandId, **EmberAfStatus** status)
boolean	**emberAfDiscoverAttributesResponseCallback** (**EmberAfClusterId** clusterId, boolean discoveryComplete, int8u *buffer, int16u bufLen)
EmberAfAttributeWritePermission	**emberAfAllowNetworkWriteAttributeCallback** (int8u endpoint, **EmberAfClusterId** clusterId, **EmberAfAttributeId** attributeId, int8u mask, int16u manufacturerCode, int8u *value, int8u type)
void	**emberAfPreAttributeChangeCallback** (int8u endpoint, **EmberAfClusterId** clusterId, **EmberAfAttributeId** attributeId, int8u mask, int16u manufacturerCode, int8u type, int8u size, int8u *value)
void	**emberAfPostAttributeChangeCallback** (int8u endpoint, **EmberAfClusterId** clusterId, **EmberAfAttributeId** attributeId, int8u mask, int16u manufacturerCode, int8u type, int8u size, int8u *value)
boolean	**emberAfPreMessageReceivedCallback** (**EmberAfIncomingMessage** *incomingMessage)
boolean	**emberAfPreCommandReceivedCallback** (**EmberAfClusterCommand** *cmd)
boolean	**emberAfPreZDOMessageReceivedCallback** (EmberNodeId emberNodeId, EmberApsFrame *apsFrame, int8u *message, int16u length)
boolean	**emberAfMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
boolean	**emberAfStackStatusCallback** (EmberStatus status)
void	**emberAfNcpInitCallback** (void)
void	**emberAfMainInitCallback** (void)
void	**emberAfClusterInitCallback** (int8u endpoint, **EmberAfClusterId** clusterId)
boolean	**emberAfPreGoToSleepCallback** (int32u sleepDurationAttempt)
void	**emberAfPostWakeUpCallback** (int32u sleepDuration)
void	**emberAfMainTickCallback** (void)
boolean	**emberAfExternalAttributeWriteCallback** (int8u endpoint, **EmberAfClusterId** clusterId, **EmberAfAttributeMetadata** *attributeMetadata, int16u manufacturerCode, int8u *buffer)

boolean	**emberAfExternalAttributeReadCallback** (int8u endpoint, **EmberAfClusterId** clusterId, **EmberAfAttributeMetadata** *attributeMetadata, int16u manufacturerCode, int8u *buffer)
EmberAfStatus	**emberAfReportEntryConfiguredCallback** (**EmberAfReportEntry** *entry)
void	**emberAfTrustCenterJoinCallback** (EmberNodeId newNodeId, EmberEUI64 newNodeEui64, EmberNodeId parentOfNewNode, EmberDeviceUpdate status, EmberJoinDecision decision)
int32u	**emberAfGetCurrentTimeCallback** (void)
void	**emberAfSetTimeCallback** (int32u utcTime)
EmberStatus	**emberAfInitiateKeyEstablishmentCallback** (EmberNodeId nodeId, int8u endpoint)
EmberStatus	**emberAfInitiateInterPanKeyEstablishmentCallback** (EmberPanId panId, const EmberEUI64 eui64)
boolean	**emberAfKeyEstablishmentCallback** (**EmberAfKeyEstablishmentNotifyMessage** status, boolean amInitiator, EmberNodeId partnerShortId, int8u delayInSeconds)
boolean	**emberAfInterPanKeyEstablishmentCallback** (**EmberAfKeyEstablishmentNotifyMessage** status, boolean amInitiator, EmberPanId panId, const EmberEUI64 eui64, int8u delayInSeconds)
boolean	**emberAfPerformingKeyEstablishmentCallback** (void)
EmberStatus	**emberAfInitiatePartnerLinkKeyExchangeCallback** (EmberNodeId target, int8u endpoint, **EmberAfPartnerLinkKeyExchangeCallback** *callback)
EmberStatus	**emberAfPartnerLinkKeyExchangeRequestCallback** (EmberEUI64 partner)
void	**emberAfPartnerLinkKeyExchangeResponseCallback** (EmberNodeId sender, EmberZdoStatus status)
EmberStatus	**emberAfRegistrationStartCallback** (void)
void	**emberAfRegistrationAbortCallback** (void)
void	**emberAfRegistrationCallback** (boolean success)
void	**emberAfTrustCenterKeepaliveUpdateCallback** (boolean registrationComplete)
void	**emberAfTrustCenterKeepaliveAbortCallback** (void)
void	**emberAfCustomCommandLineCallback** (void)
boolean	**emberAfOtaServerIncomingMessageRawCallback** (**EmberAfClusterCommand** *message)
boolean	**emberAfOtaServerSendImageNotifyCallback** (EmberNodeId dest, int8u endpoint, int8u payloadType, int8u queryJitter, const **EmberAfOtaImageId** *id)
boolean	**emberAfOtaStorageDriverInitCallback** (void)
boolean	**emberAfOtaStorageDriverReadCallback** (int32u offset, int32u length, int8u *returnData)
boolean	**emberAfOtaStorageDriverWriteCallback** (const int8u *dataToWrite, int32u offset, int32u length)
int8u	**emberAfOtaBootloadCallback** (const **EmberAfOtaImageId** *id, int16u ncpUpgradeTagId)
EmberAfOtaStorageStatus	**emberAfOtaStorageInitCallback** (void)
void	**emberAfOtaStorageCloseCallback** (void)
int8u	**emberAfOtaStorageGetCountCallback** (void)
EmberAfOtaImageId	**emberAfOtaStorageSearchCallback** (int16u manufacturerId, int16u imageTypeId, const int16u *hardwareVersion)
int32u	**emberAfOtaStorageGetTotalImageSizeCallback** (const **EmberAfOtaImageId** *id)
EmberAfOtaImageId	**emberAfOtaStorageIteratorFirstCallback** (void)
EmberAfOtaImageId	**emberAfOtaStorageIteratorNextCallback** (void)
EmberAfOtaStorageStatus	**emberAfOtaStorageGetFullHeaderCallback** (const **EmberAfOtaImageId** *id, **EmberAfOtaHeader** *returnData)
EmberAfOtaStorageStatus	**emberAfOtaStorageReadImageDataCallback** (const **EmberAfOtaImageId** *id, int32u offset, int32u length, int8u *returnData, int32u *returnedLength)
EmberAfOtaStorageStatus	**emberAfOtaStorageClearTempDataCallback** (void)

EmberAfOtaStorageStatus	emberAfOtaStorageWriteTempDataCallback (int32u offset, int32u length, const int8u *data)
EmberAfOtaStorageStatus	emberAfOtaStorageCheckTempDataCallback (int32u *currentOffset, int32u *totalImageSize, EmberAfOtaImageId *newFileInfo)
EmberAfOtaStorageStatus	emberAfOtaStorageFinishDownloadCallback (int32u offset)
boolean	emberAfOtaClientIncomingMessageRawCallback (EmberAfClusterCommand *message)
void	emberAfOtaClientStartCallback (void)
void	emberAfOtaClientVersionInfoCallback (EmberAfOtaImageId *currentImageInfo, int16u *hardwareVersion)
EmberAfImageVerifyStatus	emberAfOtaClientCustomVerifyCallback (boolean newVerification, const EmberAfOtaImageId *id)
boolean	emberAfOtaClientDownloadCompleteCallback (boolean success, const EmberAfOtaImageId *id)
void	emberAfOtaClientBootloadCallback (const EmberAfOtaImageId *id)
int8u	emberAfOtaServerQueryCallback (const EmberAfOtaImageId *currentImageId, int16u *hardwareVersion, EmberAfOtaImageId *nextUpgradeImageId)
int8u	emberAfOtaServerBlockSizeCallback (EmberNodeId clientNodeId)
int8u	emberAfOtaServerImageBlockRequestCallback (const EmberAfOtaImageId *id, int32u offset, int8u maxDataSize, int32u *waitTimeMinutes)
boolean	emberAfOtaServerUpgradeEndRequestCallback (EmberNodeId source, int8u status, int32u *returnValue, const EmberAfOtaImageId *imageId)
int8u	emberAfOtaPageRequestServerPolicyCallback (void)
EmberStatus	emberAfInterpanSendMessageCallback (EmberAfInterpanHeader *header, int16u messageLength, int8u *message)
void	emberAfInterpanInitCallback (void)
void	emberAfUnusedPanIdFoundCallback (EmberPanId panId, int8u channel)
void	emberAfJoinableNetworkFoundCallback (EmberZigbeeNetwork *networkFound, int8u lqi, int8s rssi)
void	emberAfScanErrorCallback (EmberStatus status)
EmberStatus	emberAfFindUnusedPanIdAndFormCallback (void)
EmberStatus	emberAfStartSearchForJoinableNetworkCallback (void)
void	emberAfContinueJoinableNetworkSearchCallback (void)
void	emberAfNetworkFindOperationCompleteCallback (void)

Basic Cluster Callbacks

boolean	emberAfBasicClusterResetToFactoryDefaultsCallback (void)
void	emberAfBasicClusterServerInitCallback (int8u endpoint)
void	emberAfBasicClusterClientInitCallback (int8u endpoint)
void	emberAfBasicClusterServerTickCallback (int8u endpoint)
void	emberAfBasicClusterClientTickCallback (int8u endpoint)
void	emberAfBasicClusterServerAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)
void	emberAfBasicClusterClientAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId)
void	emberAfBasicClusterServerManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)
void	emberAfBasicClusterClientManufacturerSpecificAttributeChangedCallback (int8u endpoint, EmberAfAttributeId attributeId, int16u manufacturerCode)

	void **emberAfBasicClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfBasicClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfBasicClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
	void **emberAfBasicClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Power Configuration Cluster Callbacks

	void **emberAfPowerConfigClusterServerInitCallback** (int8u endpoint)
	void **emberAfPowerConfigClusterClientInitCallback** (int8u endpoint)
	void **emberAfPowerConfigClusterServerTickCallback** (int8u endpoint)
	void **emberAfPowerConfigClusterClientTickCallback** (int8u endpoint)
	void **emberAfPowerConfigClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfPowerConfigClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
	void **emberAfPowerConfigClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfPowerConfigClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
	void **emberAfPowerConfigClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfPowerConfigClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
	void **emberAfPowerConfigClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
	void **emberAfPowerConfigClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Device Temperature Configuration Cluster Callbacks

	void **emberAfDeviceTempClusterServerInitCallback** (int8u endpoint)
	void **emberAfDeviceTempClusterClientInitCallback** (int8u endpoint)
	void **emberAfDeviceTempClusterServerTickCallback** (int8u endpoint)
	void **emberAfDeviceTempClusterClientTickCallback** (int8u endpoint)
	void **emberAfDeviceTempClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)

void	**emberAfDeviceTempClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfDeviceTempClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfDeviceTempClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfDeviceTempClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfDeviceTempClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfDeviceTempClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfDeviceTempClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Identify Cluster Callbacks

boolean	**emberAfIdentifyClusterIdentifyCallback** (int16u identifyTime)
boolean	**emberAfIdentifyClusterIdentifyQueryCallback** (void)
boolean	**emberAfIdentifyClusterIdentifyQueryResponseCallback** (int16u timeout)
void	**emberAfIdentifyClusterServerInitCallback** (int8u endpoint)
void	**emberAfIdentifyClusterClientInitCallback** (int8u endpoint)
void	**emberAfIdentifyClusterServerTickCallback** (int8u endpoint)
void	**emberAfIdentifyClusterClientTickCallback** (int8u endpoint)
void	**emberAfIdentifyClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIdentifyClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIdentifyClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIdentifyClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIdentifyClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIdentifyClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIdentifyClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfIdentifyClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Groups Cluster Callbacks

boolean	**emberAfGroupsClusterEndpointInGroupCallback** (int8u endpoint, int16u groupId)
void	**emberAfGroupsClusterClearGroupTableCallback** (void)
boolean	**emberAfGroupsClusterAddGroupCallback** (int16u groupId, int8u *groupName)
boolean	**emberAfGroupsClusterViewGroupCallback** (int16u groupId)
boolean	**emberAfGroupsClusterGetGroupMembershipCallback** (int8u groupCount, int8u *groupList)
boolean	**emberAfGroupsClusterRemoveGroupCallback** (int16u groupId)
boolean	**emberAfGroupsClusterRemoveAllGroupsCallback** (void)
boolean	**emberAfGroupsClusterAddGroupIfIdentifyingCallback** (int16u groupId, int8u *groupName)
boolean	**emberAfGroupsClusterAddGroupResponseCallback** (int8u status, int16u groupId)
boolean	**emberAfGroupsClusterViewGroupResponseCallback** (int8u status, int16u groupId, int8u *groupName)
boolean	**emberAfGroupsClusterGetGroupMembershipResponseCallback** (int8u capacity, int8u groupCount, int8u *groupList)
boolean	**emberAfGroupsClusterRemoveGroupResponseCallback** (int8u status, int16u groupId)
void	**emberAfGroupsClusterServerInitCallback** (int8u endpoint)
void	**emberAfGroupsClusterClientInitCallback** (int8u endpoint)
void	**emberAfGroupsClusterServerTickCallback** (int8u endpoint)
void	**emberAfGroupsClusterClientTickCallback** (int8u endpoint)
void	**emberAfGroupsClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfGroupsClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfGroupsClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfGroupsClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfGroupsClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfGroupsClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfGroupsClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfGroupsClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Scenes Cluster Callbacks

EmberAfStatus	**emberAfScenesClusterStoreCurrentSceneCallback** (int8u endpoint, int16u groupId, int8u sceneId)
EmberAfStatus	**emberAfScenesClusterRecallSavedSceneCallback** (int8u endpoint, int16u groupId, int8u sceneId)
void	**emberAfScenesClusterClearSceneTableCallback** (void)
boolean	**emberAfScenesClusterAddSceneCallback** (int16u groupId, int8u sceneId, int16u transitionTime, int8u *sceneName, int8u *extensionFieldSets)
boolean	**emberAfScenesClusterViewSceneCallback** (int16u groupId, int8u sceneId)
boolean	**emberAfScenesClusterRemoveSceneCallback** (int16u groupId, int8u sceneId)
boolean	**emberAfScenesClusterRemoveAllScenesCallback** (int16u groupId)

boolean	**emberAfScenesClusterStoreSceneCallback** (int16u groupId, int8u sceneId)
boolean	**emberAfScenesClusterRecallSceneCallback** (int16u groupId, int8u sceneId)
boolean	**emberAfScenesClusterGetSceneMembershipCallback** (int16u groupId)
boolean	**emberAfScenesClusterAddSceneResponseCallback** (int8u status, int16u groupId, int8u sceneId)
boolean	**emberAfScenesClusterViewSceneResponseCallback** (int8u status, int16u groupId, int8u sceneId, int16u transitionTime, int8u *sceneName, int8u *extensionFieldSets)
boolean	**emberAfScenesClusterRemoveSceneResponseCallback** (int8u status, int16u groupId, int8u sceneId)
boolean	**emberAfScenesClusterRemoveAllScenesResponseCallback** (int8u status, int16u groupId)
boolean	**emberAfScenesClusterStoreSceneResponseCallback** (int8u status, int16u groupId, int8u sceneId)
boolean	**emberAfScenesClusterGetSceneMembershipResponseCallback** (int8u status, int8u capacity, int16u groupId, int8u sceneCount, int8u *sceneList)
void	**emberAfScenesClusterServerInitCallback** (int8u endpoint)
void	**emberAfScenesClusterClientInitCallback** (int8u endpoint)
void	**emberAfScenesClusterServerTickCallback** (int8u endpoint)
void	**emberAfScenesClusterClientTickCallback** (int8u endpoint)
void	**emberAfScenesClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfScenesClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfScenesClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfScenesClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfScenesClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfScenesClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfScenesClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfScenesClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

On/off Cluster Callbacks

boolean	**emberAfOnOffClusterOffCallback** (void)
boolean	**emberAfOnOffClusterOnCallback** (void)
boolean	**emberAfOnOffClusterToggleCallback** (void)
void	**emberAfOnOffClusterServerInitCallback** (int8u endpoint)
void	**emberAfOnOffClusterClientInitCallback** (int8u endpoint)
void	**emberAfOnOffClusterServerTickCallback** (int8u endpoint)
void	**emberAfOnOffClusterClientTickCallback** (int8u endpoint)
void	**emberAfOnOffClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOnOffClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)

void	**emberAfOnOffClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOnOffClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOnOffClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOnOffClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOnOffClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfOnOffClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
boolean	**emberAfOnOffClusterSampleMfgSpecificOffWithTransitionCallback** (void)
boolean	**emberAfOnOffClusterSampleMfgSpecificOnWithTransitionCallback** (void)
boolean	**emberAfOnOffClusterSampleMfgSpecificToggleWithTransitionCallback** (void)

On/off Switch Configuration Cluster Callbacks

void	**emberAfOnOffSwitchConfigClusterServerInitCallback** (int8u endpoint)
void	**emberAfOnOffSwitchConfigClusterClientInitCallback** (int8u endpoint)
void	**emberAfOnOffSwitchConfigClusterServerTickCallback** (int8u endpoint)
void	**emberAfOnOffSwitchConfigClusterClientTickCallback** (int8u endpoint)
void	**emberAfOnOffSwitchConfigClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOnOffSwitchConfigClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOnOffSwitchConfigClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOnOffSwitchConfigClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOnOffSwitchConfigClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOnOffSwitchConfigClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOnOffSwitchConfigClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfOnOffSwitchConfigClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Level Control Cluster Callbacks

boolean	**emberAfLevelControlClusterMoveToLevelCallback** (int8u level, int16u transitionTime)
boolean	**emberAfLevelControlClusterMoveCallback** (int8u moveMode, int8u rate)
boolean	**emberAfLevelControlClusterStepCallback** (int8u stepMode, int8u stepSize, int16u transitionTime)
boolean	**emberAfLevelControlClusterStopCallback** (void)
boolean	**emberAfLevelControlClusterMoveToLevelWithOnOffCallback** (int8u level, int16u transitionTime)
boolean	**emberAfLevelControlClusterMoveWithOnOffCallback** (int8u moveMode, int8u rate)
boolean	**emberAfLevelControlClusterStepWithOnOffCallback** (int8u stepMode, int8u stepSize, int16u transitionTime)
boolean	**emberAfLevelControlClusterStopWithOnOffCallback** (void)
void	**emberAfLevelControlClusterServerInitCallback** (int8u endpoint)
void	**emberAfLevelControlClusterClientInitCallback** (int8u endpoint)
void	**emberAfLevelControlClusterServerTickCallback** (int8u endpoint)
void	**emberAfLevelControlClusterClientTickCallback** (int8u endpoint)
void	**emberAfLevelControlClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfLevelControlClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfLevelControlClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfLevelControlClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfLevelControlClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfLevelControlClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfLevelControlClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfLevelControlClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Alarms Cluster Callbacks

boolean	**emberAfAlarmClusterResetAlarmCallback** (int8u alarmCode, int16u clusterId)
boolean	**emberAfAlarmClusterResetAllAlarmsCallback** (void)
boolean	**emberAfAlarmClusterGetAlarmCallback** (void)
boolean	**emberAfAlarmClusterResetAlarmLogCallback** (void)
boolean	**emberAfAlarmClusterAlarmCallback** (int8u alarmCode, int16u clusterId)
boolean	**emberAfAlarmClusterGetAlarmResponseCallback** (int8u status, int8u alarmCode, int16u clusterId, int32u timeStamp)
void	**emberAfAlarmClusterServerInitCallback** (int8u endpoint)
void	**emberAfAlarmClusterClientInitCallback** (int8u endpoint)
void	**emberAfAlarmClusterServerTickCallback** (int8u endpoint)
void	**emberAfAlarmClusterClientTickCallback** (int8u endpoint)

void	**emberAfAlarmClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfAlarmClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfAlarmClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfAlarmClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfAlarmClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfAlarmClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfAlarmClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfAlarmClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Time Cluster Callbacks

void	**emberAfTimeClusterServerInitCallback** (int8u endpoint)
void	**emberAfTimeClusterClientInitCallback** (int8u endpoint)
void	**emberAfTimeClusterServerTickCallback** (int8u endpoint)
void	**emberAfTimeClusterClientTickCallback** (int8u endpoint)
void	**emberAfTimeClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfTimeClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfTimeClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfTimeClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfTimeClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfTimeClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfTimeClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfTimeClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

RSSI Location Cluster Callbacks

boolean	**emberAfRssiLocationClusterSetAbsoluteLocationCallback** (int16s coordinate1, int16s coordinate2, int16s coordinate3, int16s power, int16u pathLossExponent)

boolean	**emberAfRssiLocationClusterSetDeviceConfigurationCallback** (int16s power, int16u pathLossExponent, int16u calculationPeriod, int8u numberRssiMeasurements, int16u reportingPeriod)
boolean	**emberAfRssiLocationClusterGetDeviceConfigurationCallback** (int8u *targetAddress)
boolean	**emberAfRssiLocationClusterGetLocationDataCallback** (int8u flags, int8u numberResponses, int8u *targetAddress)
boolean	**emberAfRssiLocationClusterRssiResponseCallback** (int8u *replyingDevice, int16s coordinate1, int16s coordinate2, int16s coordinate3, int8s rssi, int8u numberRssiMeasurements)
boolean	**emberAfRssiLocationClusterSendPingsCallback** (int8u *targetAddress, int8u numberRssiMeasurements, int16u calculationPeriod)
boolean	**emberAfRssiLocationClusterAnchorNodeAnnounceCallback** (int8u *anchorNodeIeeeAddress, int16s coordinate1, int16s coordinate2, int16s coordinate3)
boolean	**emberAfRssiLocationClusterDeviceConfigurationResponseCallback** (int8u status, int16s power, int16u pathLossExponent, int16u calculationPeriod, int8u numberRssiMeasurements, int16u reportingPeriod)
boolean	**emberAfRssiLocationClusterLocationDataResponseCallback** (int8u status, int8u locationType, int16s coordinate1, int16s coordinate2, int16s coordinate3, int16s power, int16u pathLossExponent, int8u locationMethod, int8u qualityMeasure, int16u locationAge)
boolean	**emberAfRssiLocationClusterLocationDataNotificationCallback** (int8u locationType, int16s coordinate1, int16s coordinate2, int16s coordinate3, int16s power, int16u pathLossExponent, int8u locationMethod, int8u qualityMeasure, int16u locationAge)
boolean	**emberAfRssiLocationClusterCompactLocationDataNotificationCallback** (int8u locationType, int16s coordinate1, int16s coordinate2, int16s coordinate3, int8u qualityMeasure, int16u locationAge)
boolean	**emberAfRssiLocationClusterRssiPingCallback** (int8u locationType)
boolean	**emberAfRssiLocationClusterRssiRequestCallback** (void)
boolean	**emberAfRssiLocationClusterReportRssiMeasurementsCallback** (int8u *measuringDevice, int8u neighbors, int8u *neighborsInfo)
boolean	**emberAfRssiLocationClusterRequestOwnLocationCallback** (int8u *blindNode)
void	**emberAfRssiLocationClusterServerInitCallback** (int8u endpoint)
void	**emberAfRssiLocationClusterClientInitCallback** (int8u endpoint)
void	**emberAfRssiLocationClusterServerTickCallback** (int8u endpoint)
void	**emberAfRssiLocationClusterClientTickCallback** (int8u endpoint)
void	**emberAfRssiLocationClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfRssiLocationClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfRssiLocationClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfRssiLocationClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfRssiLocationClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfRssiLocationClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfRssiLocationClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfRssiLocationClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Binary Input (Basic) Cluster Callbacks

void	**emberAfBinaryInputBasicClusterServerInitCallback** (int8u endpoint)
void	**emberAfBinaryInputBasicClusterClientInitCallback** (int8u endpoint)
void	**emberAfBinaryInputBasicClusterServerTickCallback** (int8u endpoint)
void	**emberAfBinaryInputBasicClusterClientTickCallback** (int8u endpoint)
void	**emberAfBinaryInputBasicClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfBinaryInputBasicClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfBinaryInputBasicClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfBinaryInputBasicClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfBinaryInputBasicClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfBinaryInputBasicClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfBinaryInputBasicClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfBinaryInputBasicClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Commissioning Cluster Callbacks

boolean	**emberAfCommissioningClusterRestartDeviceCallback** (int8u options, int8u delay, int8u jitter)
boolean	**emberAfCommissioningClusterSaveStartupParametersCallback** (int8u options, int8u index)
boolean	**emberAfCommissioningClusterRestoreStartupParametersCallback** (int8u options, int8u index)
boolean	**emberAfCommissioningClusterResetStartupParametersCallback** (int8u options, int8u index)
boolean	**emberAfCommissioningClusterRestartDeviceResponseCallback** (int8u status)
boolean	**emberAfCommissioningClusterSaveStartupParametersResponseCallback** (int8u status)
boolean	**emberAfCommissioningClusterRestoreStartupParametersResponseCallback** (int8u status)
boolean	**emberAfCommissioningClusterResetStartupParametersResponseCallback** (int8u status)
void	**emberAfCommissioningClusterServerInitCallback** (int8u endpoint)
void	**emberAfCommissioningClusterClientInitCallback** (int8u endpoint)
void	**emberAfCommissioningClusterServerTickCallback** (int8u endpoint)
void	**emberAfCommissioningClusterClientTickCallback** (int8u endpoint)
void	**emberAfCommissioningClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)

void	**emberAfCommissioningClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfCommissioningClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfCommissioningClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfCommissioningClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfCommissioningClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfCommissioningClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfCommissioningClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Shade Configuration Cluster Callbacks

void	**emberAfShadeConfigClusterServerInitCallback** (int8u endpoint)
void	**emberAfShadeConfigClusterClientInitCallback** (int8u endpoint)
void	**emberAfShadeConfigClusterServerTickCallback** (int8u endpoint)
void	**emberAfShadeConfigClusterClientTickCallback** (int8u endpoint)
void	**emberAfShadeConfigClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfShadeConfigClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfShadeConfigClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfShadeConfigClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfShadeConfigClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfShadeConfigClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfShadeConfigClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfShadeConfigClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Pump Configuration and Control Cluster Callbacks

void	**emberAfPumpConfigControlClusterServerInitCallback** (int8u endpoint)

void	**emberAfPumpConfigControlClusterClientInitCallback** (int8u endpoint)
void	**emberAfPumpConfigControlClusterServerTickCallback** (int8u endpoint)
void	**emberAfPumpConfigControlClusterClientTickCallback** (int8u endpoint)
void	**emberAfPumpConfigControlClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfPumpConfigControlClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfPumpConfigControlClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPumpConfigControlClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPumpConfigControlClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPumpConfigControlClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPumpConfigControlClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfPumpConfigControlClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Thermostat Cluster Callbacks

boolean	**emberAfThermostatClusterSetpointRaiseLowerCallback** (int8u mode, int8s amount)
void	**emberAfThermostatClusterServerInitCallback** (int8u endpoint)
void	**emberAfThermostatClusterClientInitCallback** (int8u endpoint)
void	**emberAfThermostatClusterServerTickCallback** (int8u endpoint)
void	**emberAfThermostatClusterClientTickCallback** (int8u endpoint)
void	**emberAfThermostatClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfThermostatClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfThermostatClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfThermostatClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfThermostatClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfThermostatClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfThermostatClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfThermostatClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Fan Control Cluster Callbacks

void	**emberAfFanControlClusterServerInitCallback** (int8u endpoint)
void	**emberAfFanControlClusterClientInitCallback** (int8u endpoint)
void	**emberAfFanControlClusterServerTickCallback** (int8u endpoint)
void	**emberAfFanControlClusterClientTickCallback** (int8u endpoint)
void	**emberAfFanControlClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfFanControlClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfFanControlClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfFanControlClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfFanControlClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfFanControlClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfFanControlClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfFanControlClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Dehumidification Control Cluster Callbacks

void	**emberAfDehumidControlClusterServerInitCallback** (int8u endpoint)
void	**emberAfDehumidControlClusterClientInitCallback** (int8u endpoint)
void	**emberAfDehumidControlClusterServerTickCallback** (int8u endpoint)
void	**emberAfDehumidControlClusterClientTickCallback** (int8u endpoint)
void	**emberAfDehumidControlClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfDehumidControlClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfDehumidControlClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfDehumidControlClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfDehumidControlClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)

void	**emberAfDehumidControlClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfDehumidControlClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfDehumidControlClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Thermostat User Interface Configuration Cluster Callbacks

void	**emberAfThermostatUiConfigClusterServerInitCallback** (int8u endpoint)
void	**emberAfThermostatUiConfigClusterClientInitCallback** (int8u endpoint)
void	**emberAfThermostatUiConfigClusterServerTickCallback** (int8u endpoint)
void	**emberAfThermostatUiConfigClusterClientTickCallback** (int8u endpoint)
void	**emberAfThermostatUiConfigClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfThermostatUiConfigClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfThermostatUiConfigClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfThermostatUiConfigClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfThermostatUiConfigClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfThermostatUiConfigClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfThermostatUiConfigClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfThermostatUiConfigClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Color Control Cluster Callbacks

boolean	**emberAfColorControlClusterMoveToHueCallback** (int8u hue, int8u direction, int16u transitionTime)
boolean	**emberAfColorControlClusterMoveHueCallback** (int8u moveMode, int8u rate)
boolean	**emberAfColorControlClusterStepHueCallback** (int8u stepMode, int8u stepSize, int8u transitionTime)
boolean	**emberAfColorControlClusterMoveToSaturationCallback** (int8u saturation, int16u transitionTime)
boolean	**emberAfColorControlClusterMoveSaturationCallback** (int8u moveMode, int8u rate)
boolean	**emberAfColorControlClusterStepSaturationCallback** (int8u stepMode, int8u stepSize, int8u transitionTime)
boolean	**emberAfColorControlClusterMoveToHueAndSaturationCallback** (int8u hue, int8u saturation, int16u transitionTime)
boolean	**emberAfColorControlClusterMoveToColorCallback** (int16u colorX, int16u colorY, int16u transitionTime)

boolean	**emberAfColorControlClusterMoveColorCallback** (int16s rateX, int16s rateY)
boolean	**emberAfColorControlClusterStepColorCallback** (int16s stepX, int16s stepY, int16u transitionTime)
boolean	**emberAfColorControlClusterMoveToColorTemperatureCallback** (int16u colorTemperature, int16u transitionTime)
void	**emberAfColorControlClusterServerInitCallback** (int8u endpoint)
void	**emberAfColorControlClusterClientInitCallback** (int8u endpoint)
void	**emberAfColorControlClusterServerTickCallback** (int8u endpoint)
void	**emberAfColorControlClusterClientTickCallback** (int8u endpoint)
void	**emberAfColorControlClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfColorControlClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfColorControlClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfColorControlClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfColorControlClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfColorControlClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfColorControlClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfColorControlClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Ballast Configuration Cluster Callbacks

void	**emberAfBallastConfigurationClusterServerInitCallback** (int8u endpoint)
void	**emberAfBallastConfigurationClusterClientInitCallback** (int8u endpoint)
void	**emberAfBallastConfigurationClusterServerTickCallback** (int8u endpoint)
void	**emberAfBallastConfigurationClusterClientTickCallback** (int8u endpoint)
void	**emberAfBallastConfigurationClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfBallastConfigurationClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfBallastConfigurationClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfBallastConfigurationClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfBallastConfigurationClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfBallastConfigurationClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)

void	**emberAfBallastConfigurationClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfBallastConfigurationClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Illuminance Measurement Cluster Callbacks

void	**emberAfIlluminanceMeasurementClusterServerInitCallback** (int8u endpoint)
void	**emberAfIlluminanceMeasurementClusterClientInitCallback** (int8u endpoint)
void	**emberAfIlluminanceMeasurementClusterServerTickCallback** (int8u endpoint)
void	**emberAfIlluminanceMeasurementClusterClientTickCallback** (int8u endpoint)
void	**emberAfIlluminanceMeasurementClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIlluminanceMeasurementClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIlluminanceMeasurementClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIlluminanceMeasurementClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIlluminanceMeasurementClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIlluminanceMeasurementClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIlluminanceMeasurementClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfIlluminanceMeasurementClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Illuminance Level Sensing Cluster Callbacks

void	**emberAfIlluminanceLevelSensingClusterServerInitCallback** (int8u endpoint)
void	**emberAfIlluminanceLevelSensingClusterClientInitCallback** (int8u endpoint)
void	**emberAfIlluminanceLevelSensingClusterServerTickCallback** (int8u endpoint)
void	**emberAfIlluminanceLevelSensingClusterClientTickCallback** (int8u endpoint)
void	**emberAfIlluminanceLevelSensingClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIlluminanceLevelSensingClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIlluminanceLevelSensingClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)

void	**emberAfIlluminationLevelSensingClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIlluminationLevelSensingClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIlluminationLevelSensingClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIlluminationLevelSensingClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfIlluminationLevelSensingClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Temperature Measurement Cluster Callbacks

int32s	**emberAfTempMeasurementClusterConvertVoltsToCelsiusCallback** (int16u voltage)
void	**emberAfTempMeasurementClusterServerInitCallback** (int8u endpoint)
void	**emberAfTempMeasurementClusterClientInitCallback** (int8u endpoint)
void	**emberAfTempMeasurementClusterServerTickCallback** (int8u endpoint)
void	**emberAfTempMeasurementClusterClientTickCallback** (int8u endpoint)
void	**emberAfTempMeasurementClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfTempMeasurementClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfTempMeasurementClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfTempMeasurementClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfTempMeasurementClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfTempMeasurementClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfTempMeasurementClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfTempMeasurementClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Pressure Measurement Cluster Callbacks

void	**emberAfPressureMeasurementClusterServerInitCallback** (int8u endpoint)
void	**emberAfPressureMeasurementClusterClientInitCallback** (int8u endpoint)
void	**emberAfPressureMeasurementClusterServerTickCallback** (int8u endpoint)
void	**emberAfPressureMeasurementClusterClientTickCallback** (int8u endpoint)

void	**emberAfPressureMeasurementClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfPressureMeasurementClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfPressureMeasurementClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPressureMeasurementClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPressureMeasurementClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPressureMeasurementClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPressureMeasurementClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfPressureMeasurementClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Flow Measurement Cluster Callbacks

void	**emberAfFlowMeasurementClusterServerInitCallback** (int8u endpoint)
void	**emberAfFlowMeasurementClusterClientInitCallback** (int8u endpoint)
void	**emberAfFlowMeasurementClusterServerTickCallback** (int8u endpoint)
void	**emberAfFlowMeasurementClusterClientTickCallback** (int8u endpoint)
void	**emberAfFlowMeasurementClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfFlowMeasurementClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfFlowMeasurementClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfFlowMeasurementClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfFlowMeasurementClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfFlowMeasurementClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfFlowMeasurementClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfFlowMeasurementClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Relative Humidity Measurement Cluster Callbacks

void	**emberAfRelativeHumidityMeasurementClusterServerInitCallback** (int8u endpoint)
void	**emberAfRelativeHumidityMeasurementClusterClientInitCallback** (int8u endpoint)
void	**emberAfRelativeHumidityMeasurementClusterServerTickCallback** (int8u endpoint)
void	**emberAfRelativeHumidityMeasurementClusterClientTickCallback** (int8u endpoint)
void	**emberAfRelativeHumidityMeasurementClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfRelativeHumidityMeasurementClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfRelativeHumidityMeasurementClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfRelativeHumidityMeasurementClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfRelativeHumidityMeasurementClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfRelativeHumidityMeasurementClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfRelativeHumidityMeasurementClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfRelativeHumidityMeasurementClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Occupancy Sensing Cluster Callbacks

void	**emberAfOccupancySensingClusterServerInitCallback** (int8u endpoint)
void	**emberAfOccupancySensingClusterClientInitCallback** (int8u endpoint)
void	**emberAfOccupancySensingClusterServerTickCallback** (int8u endpoint)
void	**emberAfOccupancySensingClusterClientTickCallback** (int8u endpoint)
void	**emberAfOccupancySensingClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOccupancySensingClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOccupancySensingClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOccupancySensingClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)

void	**emberAfOccupancySensingClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOccupancySensingClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOccupancySensingClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfOccupancySensingClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

IAS Zone Cluster Callbacks

boolean	**emberAfIasZoneClusterZoneEnrollResponseCallback** (int8u enrollResponseCode, int8u zoneId)
boolean	**emberAfIasZoneClusterZoneStatusChangeNotificationCallback** (int16u zoneStatus, int8u extendedStatus)
boolean	**emberAfIasZoneClusterZoneEnrollRequestCallback** (int16u zoneType, int16u manufacturerCode)
void	**emberAfIasZoneClusterServerInitCallback** (int8u endpoint)
void	**emberAfIasZoneClusterClientInitCallback** (int8u endpoint)
void	**emberAfIasZoneClusterServerTickCallback** (int8u endpoint)
void	**emberAfIasZoneClusterClientTickCallback** (int8u endpoint)
void	**emberAfIasZoneClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIasZoneClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIasZoneClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIasZoneClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIasZoneClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIasZoneClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIasZoneClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfIasZoneClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

IAS ACE Cluster Callbacks

boolean	**emberAfIasAceClusterArmCallback** (int8u armMode)
boolean	**emberAfIasAceClusterBypassCallback** (int8u numberOfZones, int8u *zoneIds)
boolean	**emberAfIasAceClusterEmergencyCallback** (void)
boolean	**emberAfIasAceClusterFireCallback** (void)
boolean	**emberAfIasAceClusterPanicCallback** (void)
boolean	**emberAfIasAceClusterGetZoneIdMapCallback** (void)

boolean	**emberAfIasAceClusterGetZoneInformationCallback** (int8u zoneId)
boolean	**emberAfIasAceClusterArmResponseCallback** (int8u armNotification)
boolean	**emberAfIasAceClusterGetZoneIdMapResponseCallback** (int16u section0, int16u section1, int16u section2, int16u section3, int16u section4, int16u section5, int16u section6, int16u section7, int16u section8, int16u section9, int16u section10, int16u section11, int16u section12, int16u section13, int16u section14, int16u section15)
boolean	**emberAfIasAceClusterGetZoneInformationResponseCallback** (int8u zoneId, int16u zoneType, int8u *ieeeAddress)
void	**emberAfIasAceClusterServerInitCallback** (int8u endpoint)
void	**emberAfIasAceClusterClientInitCallback** (int8u endpoint)
void	**emberAfIasAceClusterServerTickCallback** (int8u endpoint)
void	**emberAfIasAceClusterClientTickCallback** (int8u endpoint)
void	**emberAfIasAceClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIasAceClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIasAceClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIasAceClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIasAceClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIasAceClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIasAceClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfIasAceClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

IAS WD Cluster Callbacks

boolean	**emberAfIasWdClusterStartWarningCallback** (int8u warningInfo, int16u warningDuration)
boolean	**emberAfIasWdClusterSquawkCallback** (int8u squawkInfo)
void	**emberAfIasWdClusterServerInitCallback** (int8u endpoint)
void	**emberAfIasWdClusterClientInitCallback** (int8u endpoint)
void	**emberAfIasWdClusterServerTickCallback** (int8u endpoint)
void	**emberAfIasWdClusterClientTickCallback** (int8u endpoint)
void	**emberAfIasWdClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIasWdClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfIasWdClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIasWdClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfIasWdClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)

void	**emberAfIasWdClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfIasWdClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfIasWdClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Door Lock Cluster Callbacks

boolean	**emberAfDoorLockClusterLockDoorCallback** (void)
boolean	**emberAfDoorLockClusterUnlockDoorCallback** (void)
boolean	**emberAfDoorLockClusterLockDoorResponseCallback** (int8u status)
boolean	**emberAfDoorLockClusterUnlockDoorResponseCallback** (int8u status)
void	**emberAfDoorLockClusterServerInitCallback** (int8u endpoint)
void	**emberAfDoorLockClusterClientInitCallback** (int8u endpoint)
void	**emberAfDoorLockClusterServerTickCallback** (int8u endpoint)
void	**emberAfDoorLockClusterClientTickCallback** (int8u endpoint)
void	**emberAfDoorLockClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfDoorLockClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfDoorLockClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfDoorLockClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfDoorLockClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfDoorLockClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfDoorLockClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfDoorLockClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Window Covering Cluster Callbacks

boolean	**emberAfWindowCoveringClusterWindowCoveringUpOpenCallback** (void)
boolean	**emberAfWindowCoveringClusterWindowCoveringDownCloseCallback** (void)
boolean	**emberAfWindowCoveringClusterWindowCoveringStopCallback** (void)
boolean	**emberAfWindowCoveringClusterWindowCoveringGoToLiftSetpointCallback** (int8u indexOfLiftSetpoint)
boolean	**emberAfWindowCoveringClusterWindowCoveringGoToLiftValueCallback** (int16u liftValue)

boolean	**emberAfWindowCoveringClusterWindowCoveringGoToLiftPercentageCallback** (int8u percentageLiftValue)
boolean	**emberAfWindowCoveringClusterWindowCoveringGoToTiltSetpointCallback** (int8u indexOfTiltSetpoint)
boolean	**emberAfWindowCoveringClusterWindowCoveringGoToTiltValueCallback** (int16u tiltValue)
boolean	**emberAfWindowCoveringClusterWindowCoveringGoToTiltPercentageCallback** (int8u percentageTiltValue)
boolean	**emberAfWindowCoveringClusterWindowCoveringGoToProgramSetpointCallback** (int8u setpointType, int8u setpointIndex, int16u setpointValue)
void	**emberAfWindowCoveringClusterServerInitCallback** (int8u endpoint)
void	**emberAfWindowCoveringClusterClientInitCallback** (int8u endpoint)
void	**emberAfWindowCoveringClusterServerTickCallback** (int8u endpoint)
void	**emberAfWindowCoveringClusterClientTickCallback** (int8u endpoint)
void	**emberAfWindowCoveringClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfWindowCoveringClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfWindowCoveringClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfWindowCoveringClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfWindowCoveringClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfWindowCoveringClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfWindowCoveringClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfWindowCoveringClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Generic Tunnel Cluster Callbacks

boolean	**emberAfGenericTunnelClusterMatchProtocolAddressCallback** (int8u *protocolAddress)
boolean	**emberAfGenericTunnelClusterMatchProtocolAddressResponseCallback** (void)
boolean	**emberAfGenericTunnelClusterAdvertiseProtocolAddressCallback** (int8u *protocolAddress)
void	**emberAfGenericTunnelClusterServerInitCallback** (int8u endpoint)
void	**emberAfGenericTunnelClusterClientInitCallback** (int8u endpoint)
void	**emberAfGenericTunnelClusterServerTickCallback** (int8u endpoint)
void	**emberAfGenericTunnelClusterClientTickCallback** (int8u endpoint)
void	**emberAfGenericTunnelClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfGenericTunnelClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfGenericTunnelClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)

void	**emberAfGenericTunnelClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfGenericTunnelClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfGenericTunnelClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfGenericTunnelClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfGenericTunnelClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

BACnet Protocol Tunnel Cluster Callbacks

boolean	**emberAfBacnetProtocolTunnelClusterTransferNpduCallback** (int8u *NPDU)
void	**emberAfBacnetProtocolTunnelClusterServerInitCallback** (int8u endpoint)
void	**emberAfBacnetProtocolTunnelClusterClientInitCallback** (int8u endpoint)
void	**emberAfBacnetProtocolTunnelClusterServerTickCallback** (int8u endpoint)
void	**emberAfBacnetProtocolTunnelClusterClientTickCallback** (int8u endpoint)
void	**emberAfBacnetProtocolTunnelClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfBacnetProtocolTunnelClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfBacnetProtocolTunnelClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfBacnetProtocolTunnelClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfBacnetProtocolTunnelClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfBacnetProtocolTunnelClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfBacnetProtocolTunnelClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfBacnetProtocolTunnelClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Over the Air Bootloading Cluster Cluster Callbacks

boolean	**emberAfOtaBootloadClusterImageNotifyCallback** (int8u payloadType, int8u queryJitter, int16u manufacturerId, int16u imageType, int32u newFileVersion)

boolean	**emberAfOtaBootloadClusterQueryNextImageRequestCallback** (int8u fieldControl, int16u manufacturerId, int16u imageType, int32u currentFileVersion, int16u hardwareVersion)
boolean	**emberAfOtaBootloadClusterQueryNextImageResponseCallback** (int8u status, int16u manufacturerId, int16u imageType, int32u fileVersion, int32u imageSize)
boolean	**emberAfOtaBootloadClusterImageBlockRequestCallback** (int8u fieldControl, int16u manufacturerId, int16u imageType, int32u fileVersion, int32u fileOffset, int8u maxDataSize, int8u *requestNodeAddress)
boolean	**emberAfOtaBootloadClusterImagePageRequestCallback** (int8u fieldControl, int16u manufacturerId, int16u imageType, int32u fileVersion, int32u fileOffset, int8u maxDataSize, int16u pageSize, int16u responseSpacing, int8u *requestNodeAddress)
boolean	**emberAfOtaBootloadClusterImageBlockResponseCallback** (int8u status, int16u manufacturerId, int16u imageType, int32u fileVersion, int32u fileOffset, int8u dataSize, int8u *imageData)
boolean	**emberAfOtaBootloadClusterUpgradeEndRequestCallback** (int8u status, int16u manufacturerId, int16u imageType, int32u fileVersion)
boolean	**emberAfOtaBootloadClusterUpgradeEndResponseCallback** (int16u manufacturerId, int16u imageType, int32u fileVersion, int32u currentTime, int32u upgradeTime)
boolean	**emberAfOtaBootloadClusterQuerySpecificFileRequestCallback** (int8u *requestNodeAddress, int16u manufacturerId, int16u imageType, int32u fileVersion, int16u currentZigbeeStackVersion)
boolean	**emberAfOtaBootloadClusterQuerySpecificFileResponseCallback** (int8u status, int16u manufacturerId, int16u imageType, int32u fileVersion, int32u imageSize)
void	**emberAfOtaBootloadClusterServerInitCallback** (int8u endpoint)
void	**emberAfOtaBootloadClusterClientInitCallback** (int8u endpoint)
void	**emberAfOtaBootloadClusterServerTickCallback** (int8u endpoint)
void	**emberAfOtaBootloadClusterClientTickCallback** (int8u endpoint)
void	**emberAfOtaBootloadClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOtaBootloadClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfOtaBootloadClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOtaBootloadClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfOtaBootloadClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOtaBootloadClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfOtaBootloadClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfOtaBootloadClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Demand Response and Load Control Cluster Callbacks

boolean	**emberAfDemandResponseLoadControlClusterLoadControlEventCallback** (int32u issuerEventId, int16u deviceClass, int8u utilityEnrollmentGroup, int32u startTime, int16u durationInMinutes, int8u criticalityLevel, int8u coolingTemperatureOffset, int8u heatingTemperatureOffset, int16s coolingTemperatureSetPoint, int16s heatingTemperatureSetPoint, int8s averageLoadAdjustmentPercentage, int8u dutyCycle, int8u eventControl)
boolean	**emberAfDemandResponseLoadControlClusterCancelLoadControlEventCallback** (int32u issuerEventId, int16u deviceClass, int8u utilityEnrollmentGroup, int8u cancelControl, int32u effectiveTime)
boolean	**emberAfDemandResponseLoadControlClusterCancelAllLoadControlEventsCallback** (int8u cancelControl)
boolean	**emberAfDemandResponseLoadControlClusterReportEventStatusCallback** (int32u issuerEventId, int8u eventStatus, int32u eventStatusTime, int8u criticalityLevelApplied, int16s coolingTemperatureSetPointApplied, int16s heatingTemperatureSetPointApplied, int8s averageLoadAdjustmentPercentageApplied, int8u dutyCycleApplied, int8u eventControl, int8u signatureType, int8u *signature)
boolean	**emberAfDemandResponseLoadControlClusterGetScheduledEventsCallback** (int32u startTime, int8u numberOfEvents)
void	**emberAfDemandResponseLoadControlClusterServerInitCallback** (int8u endpoint)
void	**emberAfDemandResponseLoadControlClusterClientInitCallback** (int8u endpoint)
void	**emberAfDemandResponseLoadControlClusterServerTickCallback** (int8u endpoint)
void	**emberAfDemandResponseLoadControlClusterClientTickCallback** (int8u endpoint)
void	**emberAfDemandResponseLoadControlClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfDemandResponseLoadControlClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfDemandResponseLoadControlClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfDemandResponseLoadControlClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfDemandResponseLoadControlClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfDemandResponseLoadControlClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfDemandResponseLoadControlClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfDemandResponseLoadControlClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Simple Metering Cluster Callbacks

boolean	**emberAfSimpleMeteringClusterGetProfileResponseCallback** (int32u endTime, int8u status, int8u profileIntervalPeriod, int8u numberOfPeriodsDelivered, int8u *intervals)
boolean	**emberAfSimpleMeteringClusterRequestMirrorCallback** (void)
boolean	**emberAfSimpleMeteringClusterRemoveMirrorCallback** (void)

boolean	**emberAfSimpleMeteringClusterRequestFastPollModeResponseCallback** (int8u appliedUpdatePeriod, int32u fastPollModeEndtime)
boolean	**emberAfSimpleMeteringClusterGetProfileCallback** (int8u intervalChannel, int32u endTime, int8u numberOfPeriods)
boolean	**emberAfSimpleMeteringClusterRequestMirrorResponseCallback** (int16u endpointId)
boolean	**emberAfSimpleMeteringClusterMirrorRemovedCallback** (int16u endpointId)
boolean	**emberAfSimpleMeteringClusterRequestFastPollModeCallback** (int8u fastPollUpdatePeriod, int8u duration)
void	**emberAfSimpleMeteringClusterServerInitCallback** (int8u endpoint)
void	**emberAfSimpleMeteringClusterClientInitCallback** (int8u endpoint)
void	**emberAfSimpleMeteringClusterServerTickCallback** (int8u endpoint)
void	**emberAfSimpleMeteringClusterClientTickCallback** (int8u endpoint)
void	**emberAfSimpleMeteringClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfSimpleMeteringClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfSimpleMeteringClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfSimpleMeteringClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfSimpleMeteringClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfSimpleMeteringClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfSimpleMeteringClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfSimpleMeteringClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Price Cluster Callbacks

boolean	**emberAfPriceClusterPublishPriceCallback** (int32u providerId, int8u *rateLabel, int32u issuerEventId, int32u currentTime, int8u unitOfMeasure, int16u currency, int8u priceTrailingDigitAndPriceTier, int8u numberOfPriceTiersAndRegisterTier, int32u startTime, int16u durationInMinutes, int32u price, int8u priceRatio, int32u generationPrice, int8u generationPriceRatio, int32u alternateCostDelivered, int8u alternateCostUnit, int8u alternateCostTrailingDigit, int8u numberOfBlockThresholds, int8u priceControl)
boolean	**emberAfPriceClusterPublishBlockPeriodCallback** (int32u providerId, int32u issuerEventId, int32u blockPeriodStartTime, int32u blockPeriodDurationInMinutes, int8u numberOfPriceTiersAndNumberOfBlockThresholds, int8u blockPeriodControl)
boolean	**emberAfPriceClusterGetCurrentPriceCallback** (int8u commandOptions)
boolean	**emberAfPriceClusterGetScheduledPricesCallback** (int32u startTime, int8u numberOfEvents)
boolean	**emberAfPriceClusterPriceAcknowledgementCallback** (int32u providerId, int32u issuerEventId, int32u priceAckTime, int8u control)
boolean	**emberAfPriceClusterGetBlockPeriodsCallback** (int32u startTime, int8u numberOfEvents)
void	**emberAfPriceClusterServerInitCallback** (int8u endpoint)

void	**emberAfPriceClusterClientInitCallback** (int8u endpoint)
void	**emberAfPriceClusterServerTickCallback** (int8u endpoint)
void	**emberAfPriceClusterClientTickCallback** (int8u endpoint)
void	**emberAfPriceClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfPriceClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfPriceClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPriceClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPriceClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPriceClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPriceClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfPriceClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Messaging Cluster Callbacks

boolean	**emberAfMessagingClusterDisplayMessageCallback** (int32u messageId, int8u messageControl, int32u startTime, int16u durationInMinutes, int8u *message)
boolean	**emberAfMessagingClusterCancelMessageCallback** (int32u messageId, int8u messageControl)
boolean	**emberAfMessagingClusterGetLastMessageCallback** (void)
boolean	**emberAfMessagingClusterMessageConfirmationCallback** (int32u messageId, int32u confirmationTime)
void	**emberAfMessagingClusterServerInitCallback** (int8u endpoint)
void	**emberAfMessagingClusterClientInitCallback** (int8u endpoint)
void	**emberAfMessagingClusterServerTickCallback** (int8u endpoint)
void	**emberAfMessagingClusterClientTickCallback** (int8u endpoint)
void	**emberAfMessagingClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfMessagingClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfMessagingClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfMessagingClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfMessagingClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfMessagingClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfMessagingClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

void	**emberAfMessagingClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Tunneling Cluster Callbacks

boolean	**emberAfTunnelingClusterRequestTunnelCallback** (int8u protocolId, int16u manufacturerCode, int8u flowControlSupport)
boolean	**emberAfTunnelingClusterCloseTunnelCallback** (int16u tunnelId)
boolean	**emberAfTunnelingClusterTransferDataClientToServerCallback** (int16u tunnelId, int8u *data)
boolean	**emberAfTunnelingClusterTransferDataErrorClientToServerCallback** (int16u tunnelId, int8u transferDataStatus)
boolean	**emberAfTunnelingClusterAckTransferDataClientToServerCallback** (int16u tunnelId, int16u numberOfBytesLeft)
boolean	**emberAfTunnelingClusterReadyDataClientToServerCallback** (int16u tunnelId, int16u numberOfOctetsLeft)
boolean	**emberAfTunnelingClusterRequestTunnelResponseCallback** (int16u tunnelId, int8u tunnelStatus)
boolean	**emberAfTunnelingClusterTransferDataServerToClientCallback** (int16u tunnelId, int8u *data)
boolean	**emberAfTunnelingClusterTransferDataErrorServerToClientCallback** (int16u tunnelId, int8u transferDataStatus)
boolean	**emberAfTunnelingClusterAckTransferDataServerToClientCallback** (int16u tunnelId, int16u numberOfBytesLeft)
boolean	**emberAfTunnelingClusterReadyDataServerToClientCallback** (int16u tunnelId, int16u numberOfOctetsLeft)
void	**emberAfTunnelingClusterServerInitCallback** (int8u endpoint)
void	**emberAfTunnelingClusterClientInitCallback** (int8u endpoint)
void	**emberAfTunnelingClusterServerTickCallback** (int8u endpoint)
void	**emberAfTunnelingClusterClientTickCallback** (int8u endpoint)
void	**emberAfTunnelingClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfTunnelingClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfTunnelingClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfTunnelingClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfTunnelingClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfTunnelingClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfTunnelingClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfTunnelingClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Prepayment Cluster Callbacks

boolean	**emberAfPrepaymentClusterSelectAvailableEmergencyCreditCallback** (int32u commandDateTime, int8u originatingDevice, int8u *siteId, int8u *meterSerialNumber)
boolean	**emberAfPrepaymentClusterChangeSupplyCallback** (int32u providerId, int32u requestDateTime, int8u *siteId, int8u *meterSerialNumber, int32u implementationDateTime, int8u proposedSupplyStatus, int8u originatorIdSupplyControlBits)
boolean	**emberAfPrepaymentClusterSupplyStatusResponseCallback** (int32u providerId, int32u implementationtDateTime, int8u supplyStatus)
void	**emberAfPrepaymentClusterServerInitCallback** (int8u endpoint)
void	**emberAfPrepaymentClusterClientInitCallback** (int8u endpoint)
void	**emberAfPrepaymentClusterServerTickCallback** (int8u endpoint)
void	**emberAfPrepaymentClusterClientTickCallback** (int8u endpoint)
void	**emberAfPrepaymentClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfPrepaymentClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfPrepaymentClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPrepaymentClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfPrepaymentClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPrepaymentClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfPrepaymentClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfPrepaymentClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Key establishment Cluster Callbacks

boolean	**emberAfKeyEstablishmentClusterClientCommandReceivedCallback** (**EmberAfClusterCommand** *cmd)
boolean	**emberAfKeyEstablishmentClusterServerCommandReceivedCallback** (**EmberAfClusterCommand** *cmd)
boolean	**emberAfKeyEstablishmentClusterInitiateKeyEstablishmentRequestCallback** (int16u keyEstablishmentSuite, int8u ephemeralDataGenerateTime, int8u confirmKeyGenerateTime, int8u *identity)
boolean	**emberAfKeyEstablishmentClusterEphemeralDataRequestCallback** (int8u *ephemeralData)
boolean	**emberAfKeyEstablishmentClusterConfirmKeyDataRequestCallback** (int8u *secureMessageAuthenticationCode)

boolean	**emberAfKeyEstablishmentClusterTerminateKeyEstablishmentCallback** (int8u statusCode, int8u waitTime, int16u keyEstablishmentSuite)
boolean	**emberAfKeyEstablishmentClusterInitiateKeyEstablishmentResponseCallback** (int16u requestedKeyEstablishmentSuite, int8u ephemeralDataGenerateTime, int8u confirmKeyGenerateTime, int8u *identity)
boolean	**emberAfKeyEstablishmentClusterEphemeralDataResponseCallback** (int8u *ephemeralData)
boolean	**emberAfKeyEstablishmentClusterConfirmKeyDataResponseCallback** (int8u *secureMessageAuthenticationCode)
void	**emberAfKeyEstablishmentClusterServerInitCallback** (int8u endpoint)
void	**emberAfKeyEstablishmentClusterClientInitCallback** (int8u endpoint)
void	**emberAfKeyEstablishmentClusterServerTickCallback** (int8u endpoint)
void	**emberAfKeyEstablishmentClusterClientTickCallback** (int8u endpoint)
void	**emberAfKeyEstablishmentClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfKeyEstablishmentClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfKeyEstablishmentClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfKeyEstablishmentClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfKeyEstablishmentClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfKeyEstablishmentClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfKeyEstablishmentClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfKeyEstablishmentClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

Sample Mfg Specific Cluster Cluster Callbacks

boolean	**emberAfSampleMfgSpecificClusterCommandOneCallback** (int8u argOne)
void	**emberAfSampleMfgSpecificClusterServerInitCallback** (int8u endpoint)
void	**emberAfSampleMfgSpecificClusterClientInitCallback** (int8u endpoint)
void	**emberAfSampleMfgSpecificClusterServerTickCallback** (int8u endpoint)
void	**emberAfSampleMfgSpecificClusterClientTickCallback** (int8u endpoint)
void	**emberAfSampleMfgSpecificClusterServerAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfSampleMfgSpecificClusterClientAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId)
void	**emberAfSampleMfgSpecificClusterServerManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)

void	**emberAfSampleMfgSpecificClusterClientManufacturerSpecificAttributeChangedCallback** (int8u endpoint, **EmberAfAttributeId** attributeId, int16u manufacturerCode)
void	**emberAfSampleMfgSpecificClusterServerDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfSampleMfgSpecificClusterClientDefaultResponseCallback** (int8u endpoint, int8u commandId, **EmberAfStatus** status)
void	**emberAfSampleMfgSpecificClusterServerMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)
void	**emberAfSampleMfgSpecificClusterClientMessageSentCallback** (EmberOutgoingMessageType type, int16u indexOrDestination, EmberApsFrame *apsFrame, int16u msgLen, int8u *message, EmberStatus status)

client-api.h File Reference

API for generating command buffer. More...

#include "**../gen/client-command-macro.h**"

Go to the source code of this file.

Client API functions

int16u	**emberAfFillExternalBuffer** (int8u frameControl, **EmberAfClusterId** clusterId, int8u commandId, PGM_P format,...)
int16u	**emberAfFillExternalManufacturerSpecificBuffer** (int8u frameControl, **EmberAfClusterId** clusterId, int16u manufacturerCode, int8u commandId, PGM_P format,...)
void	**emberAfSetExternalBuffer** (int8u *buffer, int16u bufferLen, int16u *responseLengthPtr, EmberApsFrame *apsFramePtr)
int16u	**emberAfFillBuffer** (int8u *buffer, int16u bufferLen, int8u frameControl, int8u commandId, PGM_P format,...)

Detailed Description

API for generating command buffer.

Definition in file **client-api.h**.

Function Documentation

```
int16u emberAfFillExternalBuffer ( int8u            frameControl,
                                   EmberAfClusterId clusterId,
                                   int8u            commandId,
                                   PGM_P            format,
                                                    ...
                                 )
```

Function that fills in the buffer with command.

Fills the outgoing zcl buffer and returns the number of bytes used in a buffer. The buffers used are the ones that were previously set with emberAfSetExternalBuffer.

Parameters:

frameControl	byte used for frame control
clusterId	cluster id of message

commandId	command id of message
format	String format for further arguments to the function. Format values are: ○ '1' - '9' and 'A' - 'G': 1-16 byte long generic blobs ○ 'u': int8u ○ 'v': int16u ○ 'w': int32u ○ 's': pointer to int8u, first byte contains remaining length ○ 'S': one argument is int8u*, next argument is int8u length. Buffer is filled with length byte first and then the rest of data. ○ 'b': one argument is int8u*, next argument is int8u length. Buffer is filled only with the data in pointer, length is used, but not written as a first byte.

```
int16u emberAfFillExternalManufacturerSpecificBuffer ( int8u            frameControl,
                                                       EmberAfClusterId clusterId,
                                                       int16u           manufacturerCode,
                                                       int8u            commandId,
                                                       PGM_P            format,
                                                       ...
                                                     )
```

Function that fills in the buffer with manufacturer-specifc command.

Fills the outgoing zcl buffer and returns the number of bytes used in a buffer. The buffers used are the ones that were previously set with emberAfSetExternalBuffer.

Parameters:

frameControl	byte used for frame control
clusterId	cluster id of message
manufacturerCode	manufacturer code of message
commandId	command id of message
format	String format for further arguments to the function. Format values are: ○ '1' - '9' and 'A' - 'G': 1-16 byte long generic blobs ○ 'u': int8u ○ 'v': int16u ○ 'w': int32u ○ 's': pointer to int8u, first byte contains remaining length ○ 'S': one argument is int8u*, next argument is int8u length. Buffer is filled with length byte first and then the rest of data. ○ 'b': one argument is int8u*, next argument is int8u length. Buffer is filled only with the data in pointer, length is used, but not written as a first byte.

```
void emberAfSetExternalBuffer ( int8u *          buffer,
                                int16u           bufferLen,
                                int16u *         responseLengthPtr,
                                EmberApsFrame *  apsFramePtr
                              )
```

Function that registers the buffer to use with the fill function. Registers the buffer for use with the emberAfFillExternalBuffer function.

Parameters:

buffer	Main buffer for constructing message.

bufferLen	Available length of buffer.
responseLengthPtr	location where length of message will be written into.
apsFramePtr	location where APS frame data will be written.

```
int16u emberAfFillBuffer ( int8u *  buffer,
                           int16u   bufferLen,
                           int8u    frameControl,
                           int8u    commandId,
                           PGM_P    format,
                                    ...
                         )
```

Stateless function that fills the passed buffer with command data.

Stateless method, that fill the passed buffer. This method is used internally by emberAfFillExternalBuffer, but can be used for generic buffer filling.

---

client-command-macro.h File Reference

Go to the source code of this file.

Global Commands

#define	**emberAfFillCommandGlobalServerToClientReadAttributes**(clusterId, attributeIds, attributeIdsLen)
#define	**emberAfFillCommandGlobalClientToServerReadAttributes**(clusterId, attributeIds, attributeIdsLen)
#define	**emberAfFillCommandGlobalServerToClientReadAttributesResponse**(clusterId, readAttributeStatusRecords, readAttributeStatusRecordsLen)
#define	**emberAfFillCommandGlobalClientToServerReadAttributesResponse**(clusterId, readAttributeStatusRecords, readAttributeStatusRecordsLen)
#define	**emberAfFillCommandGlobalServerToClientWriteAttributes**(clusterId, writeAttributeRecords, writeAttributeRecordsLen)
#define	**emberAfFillCommandGlobalClientToServerWriteAttributes**(clusterId, writeAttributeRecords, writeAttributeRecordsLen)
#define	**emberAfFillCommandGlobalServerToClientWriteAttributesUndivided**(clusterId, writeAttributeRecords, writeAttributeRecordsLen)
#define	**emberAfFillCommandGlobalClientToServerWriteAttributesUndivided**(clusterId, writeAttributeRecords, writeAttributeRecordsLen)
#define	**emberAfFillCommandGlobalServerToClientWriteAttributesResponse**(clusterId, writeAttributeStatusRecords, writeAttributeStatusRecordsLen)
#define	**emberAfFillCommandGlobalClientToServerWriteAttributesResponse**(clusterId, writeAttributeStatusRecords, writeAttributeStatusRecordsLen)
#define	**emberAfFillCommandGlobalServerToClientWriteAttributesNoResponse**(clusterId, writeAttributeRecords, writeAttributeRecordsLen)
#define	**emberAfFillCommandGlobalClientToServerWriteAttributesNoResponse**(clusterId, writeAttributeRecords, writeAttributeRecordsLen)
#define	**emberAfFillCommandGlobalServerToClientConfigureReporting**(clusterId, configureReportingRecords, configureReportingRecordsLen)
#define	**emberAfFillCommandGlobalClientToServerConfigureReporting**(clusterId, configureReportingRecords, configureReportingRecordsLen)
#define	**emberAfFillCommandGlobalServerToClientConfigureReportingResponse**(clusterId, configureReportingStatusRecords, configureReportingStatusRecordsLen)
#define	**emberAfFillCommandGlobalClientToServerConfigureReportingResponse**(clusterId, configureReportingStatusRecords, configureReportingStatusRecordsLen)
#define	**emberAfFillCommandGlobalServerToClientReadReportingConfiguration**(clusterId, readReportingConfigurationAttributeRecords, readReportingConfigurationAttributeRecordsLen)
#define	**emberAfFillCommandGlobalClientToServerReadReportingConfiguration**(clusterId, readReportingConfigurationAttributeRecords, readReportingConfigurationAttributeRecordsLen)
#define	**emberAfFillCommandGlobalServerToClientReadReportingConfigurationResponse**(clusterId, readReportingConfigurationRecords, readReportingConfigurationRecordsLen)
#define	**emberAfFillCommandGlobalClientToServerReadReportingConfigurationResponse**(clusterId, readReportingConfigurationRecords, readReportingConfigurationRecordsLen)
#define	**emberAfFillCommandGlobalServerToClientReportAttributes**(clusterId, reportAttributeRecords, reportAttributeRecordsLen)
#define	**emberAfFillCommandGlobalClientToServerReportAttributes**(clusterId, reportAttributeRecords, reportAttributeRecordsLen)
#define	**emberAfFillCommandGlobalServerToClientDefaultResponse**(clusterId, commandId, status)
#define	**emberAfFillCommandGlobalClientToServerDefaultResponse**(clusterId, commandId, status)
#define	**emberAfFillCommandGlobalServerToClientDiscoverAttributes**(clusterId, startId, maxAttributeIds)

#define **emberAfFillCommandGlobalClientToServerDiscoverAttributes**(clusterId, startId, maxAttributeIds)
#define **emberAfFillCommandGlobalServerToClientDiscoverAttributesResponse**(clusterId, discoveryComplete, discoverAttributesInfoRecords, discoverAttributesInfoRecordsLen)
#define **emberAfFillCommandGlobalClientToServerDiscoverAttributesResponse**(clusterId, discoveryComplete, discoverAttributesInfoRecords, discoverAttributesInfoRecordsLen)

Basic Commands

#define **emberAfFillCommandBasicClusterResetToFactoryDefaults**()

Identify Commands

#define **emberAfFillCommandIdentifyClusterIdentify**(identifyTime)
#define **emberAfFillCommandIdentifyClusterIdentifyQuery**()
#define **emberAfFillCommandIdentifyClusterIdentifyQueryResponse**(timeout)

Groups Commands

#define **emberAfFillCommandGroupsClusterAddGroup**(groupId, groupName)
#define **emberAfFillCommandGroupsClusterViewGroup**(groupId)
#define **emberAfFillCommandGroupsClusterGetGroupMembership**(groupCount, groupList, groupListLen)
#define **emberAfFillCommandGroupsClusterRemoveGroup**(groupId)
#define **emberAfFillCommandGroupsClusterRemoveAllGroups**()
#define **emberAfFillCommandGroupsClusterAddGroupIfIdentifying**(groupId, groupName)
#define **emberAfFillCommandGroupsClusterAddGroupResponse**(status, groupId)
#define **emberAfFillCommandGroupsClusterViewGroupResponse**(status, groupId, groupName)
#define **emberAfFillCommandGroupsClusterGetGroupMembershipResponse**(capacity, groupCount, groupList, groupListLen)
#define **emberAfFillCommandGroupsClusterRemoveGroupResponse**(status, groupId)

Scenes Commands

#define **emberAfFillCommandScenesClusterAddScene**(groupId, sceneId, transitionTime, sceneName, extensionFieldSets, extensionFieldSetsLen)
#define **emberAfFillCommandScenesClusterViewScene**(groupId, sceneId)
#define **emberAfFillCommandScenesClusterRemoveScene**(groupId, sceneId)
#define **emberAfFillCommandScenesClusterRemoveAllScenes**(groupId)
#define **emberAfFillCommandScenesClusterStoreScene**(groupId, sceneId)
#define **emberAfFillCommandScenesClusterRecallScene**(groupId, sceneId)
#define **emberAfFillCommandScenesClusterGetSceneMembership**(groupId)
#define **emberAfFillCommandScenesClusterAddSceneResponse**(status, groupId, sceneId)
#define **emberAfFillCommandScenesClusterViewSceneResponse**(status, groupId, sceneId, transitionTime, sceneName, extensionFieldSets, extensionFieldSetsLen)
#define **emberAfFillCommandScenesClusterRemoveSceneResponse**(status, groupId, sceneId)
#define **emberAfFillCommandScenesClusterRemoveAllScenesResponse**(status, groupId)
#define **emberAfFillCommandScenesClusterStoreSceneResponse**(status, groupId, sceneId)

#define **emberAfFillCommandScenesClusterGetSceneMembershipResponse**(status, capacity, groupId, sceneCount, sceneList, sceneListLen)

On/off Commands

#define **emberAfFillCommandOnOffClusterOff**()
#define **emberAfFillCommandOnOffClusterOn**()
#define **emberAfFillCommandOnOffClusterToggle**()
#define **emberAfFillCommandOnOffClusterSampleMfgSpecificOffWithTransition**()
#define **emberAfFillCommandOnOffClusterSampleMfgSpecificOnWithTransition**()
#define **emberAfFillCommandOnOffClusterSampleMfgSpecificToggleWithTransition**()

Level Control Commands

#define **emberAfFillCommandLevelControlClusterMoveToLevel**(level, transitionTime)
#define **emberAfFillCommandLevelControlClusterMove**(moveMode, rate)
#define **emberAfFillCommandLevelControlClusterStep**(stepMode, stepSize, transitionTime)
#define **emberAfFillCommandLevelControlClusterStop**()
#define **emberAfFillCommandLevelControlClusterMoveToLevelWithOnOff**(level, transitionTime)
#define **emberAfFillCommandLevelControlClusterMoveWithOnOff**(moveMode, rate)
#define **emberAfFillCommandLevelControlClusterStepWithOnOff**(stepMode, stepSize, transitionTime)
#define **emberAfFillCommandLevelControlClusterStopWithOnOff**()

Alarms Commands

#define **emberAfFillCommandAlarmClusterResetAlarm**(alarmCode, clusterId)
#define **emberAfFillCommandAlarmClusterResetAllAlarms**()
#define **emberAfFillCommandAlarmClusterGetAlarm**()
#define **emberAfFillCommandAlarmClusterResetAlarmLog**()
#define **emberAfFillCommandAlarmClusterAlarm**(alarmCode, clusterId)
#define **emberAfFillCommandAlarmClusterGetAlarmResponse**(status, alarmCode, clusterId, timeStamp)

RSSI Location Commands

#define **emberAfFillCommandRssiLocationClusterSetAbsoluteLocation**(coordinate1, coordinate2, coordinate3, power, pathLossExponent)
#define **emberAfFillCommandRssiLocationClusterSetDeviceConfiguration**(power, pathLossExponent, calculationPeriod, numberRssiMeasurements, reportingPeriod)
#define **emberAfFillCommandRssiLocationClusterGetDeviceConfiguration**(targetAddress)
#define **emberAfFillCommandRssiLocationClusterGetLocationData**(flags, numberResponses, targetAddress)
#define **emberAfFillCommandRssiLocationClusterRssiResponse**(replyingDevice, coordinate1, coordinate2, coordinate3, rssi, numberRssiMeasurements)
#define **emberAfFillCommandRssiLocationClusterSendPings**(targetAddress, numberRssiMeasurements, calculationPeriod)
#define **emberAfFillCommandRssiLocationClusterAnchorNodeAnnounce**(anchorNodeIeeeAddress, coordinate1, coordinate2, coordinate3)

#define	**emberAfFillCommandRssiLocationClusterDeviceConfigurationResponse**(status, power, pathLossExponent, calculationPeriod, numberRssiMeasurements, reportingPeriod)
#define	**emberAfFillCommandRssiLocationClusterLocationDataResponse**(status, locationType, coordinate1, coordinate2, coordinate3, power, pathLossExponent, locationMethod, qualityMeasure, locationAge)
#define	**emberAfFillCommandRssiLocationClusterLocationDataNotification**(locationType, coordinate1, coordinate2, coordinate3, power, pathLossExponent, locationMethod, qualityMeasure, locationAge)
#define	**emberAfFillCommandRssiLocationClusterCompactLocationDataNotification**(locationType, coordinate1, coordinate2, coordinate3, qualityMeasure, locationAge)
#define	**emberAfFillCommandRssiLocationClusterRssiPing**(locationType)
#define	**emberAfFillCommandRssiLocationClusterRssiRequest**()
#define	**emberAfFillCommandRssiLocationClusterReportRssiMeasurements**(measuringDevice, neighbors, neighborsInfo, neighborsInfoLen)
#define	**emberAfFillCommandRssiLocationClusterRequestOwnLocation**(blindNode)

Commissioning Commands

#define	**emberAfFillCommandCommissioningClusterRestartDevice**(options, delay, jitter)
#define	**emberAfFillCommandCommissioningClusterSaveStartupParameters**(options, index)
#define	**emberAfFillCommandCommissioningClusterRestoreStartupParameters**(options, index)
#define	**emberAfFillCommandCommissioningClusterResetStartupParameters**(options, index)
#define	**emberAfFillCommandCommissioningClusterRestartDeviceResponse**(status)
#define	**emberAfFillCommandCommissioningClusterSaveStartupParametersResponse**(status)
#define	**emberAfFillCommandCommissioningClusterRestoreStartupParametersResponse**(status)
#define	**emberAfFillCommandCommissioningClusterResetStartupParametersResponse**(status)

Thermostat Commands

#define	**emberAfFillCommandThermostatClusterSetpointRaiseLower**(mode, amount)

Color Control Commands

#define	**emberAfFillCommandColorControlClusterMoveToHue**(hue, direction, transitionTime)
#define	**emberAfFillCommandColorControlClusterMoveHue**(moveMode, rate)
#define	**emberAfFillCommandColorControlClusterStepHue**(stepMode, stepSize, transitionTime)
#define	**emberAfFillCommandColorControlClusterMoveToSaturation**(saturation, transitionTime)
#define	**emberAfFillCommandColorControlClusterMoveSaturation**(moveMode, rate)
#define	**emberAfFillCommandColorControlClusterStepSaturation**(stepMode, stepSize, transitionTime)
#define	**emberAfFillCommandColorControlClusterMoveToHueAndSaturation**(hue, saturation, transitionTime)
#define	**emberAfFillCommandColorControlClusterMoveToColor**(colorX, colorY, transitionTime)
#define	**emberAfFillCommandColorControlClusterMoveColor**(rateX, rateY)
#define	**emberAfFillCommandColorControlClusterStepColor**(stepX, stepY, transitionTime)
#define	**emberAfFillCommandColorControlClusterMoveToColorTemperature**(colorTemperature, transitionTime)

IAS Zone Commands

#define	**emberAfFillCommandIasZoneClusterZoneEnrollResponse**(enrollResponseCode, zoneId)

#define **emberAfFillCommandIasZoneClusterZoneStatusChangeNotification**(zoneStatus, extendedStatus)
#define **emberAfFillCommandIasZoneClusterZoneEnrollRequest**(zoneType, manufacturerCode)

IAS ACE Commands

#define **emberAfFillCommandIasAceClusterArm**(armMode)
#define **emberAfFillCommandIasAceClusterBypass**(numberOfZones, zoneIds, zoneIdsLen)
#define **emberAfFillCommandIasAceClusterEmergency**()
#define **emberAfFillCommandIasAceClusterFire**()
#define **emberAfFillCommandIasAceClusterPanic**()
#define **emberAfFillCommandIasAceClusterGetZoneIdMap**()
#define **emberAfFillCommandIasAceClusterGetZoneInformation**(zoneId)
#define **emberAfFillCommandIasAceClusterArmResponse**(armNotification)
#define **emberAfFillCommandIasAceClusterGetZoneIdMapResponse**(section0, section1, section2, section3, section4, section5, section6, section7, section8, section9, section10, section11, section12, section13, section14, section15)
#define **emberAfFillCommandIasAceClusterGetZoneInformationResponse**(zoneId, zoneType, ieeeAddress)

IAS WD Commands

#define **emberAfFillCommandIasWdClusterStartWarning**(warningInfo, warningDuration)
#define **emberAfFillCommandIasWdClusterSquawk**(squawkInfo)

Door Lock Commands

#define **emberAfFillCommandDoorLockClusterLockDoor**()
#define **emberAfFillCommandDoorLockClusterUnlockDoor**()
#define **emberAfFillCommandDoorLockClusterLockDoorResponse**(status)
#define **emberAfFillCommandDoorLockClusterUnlockDoorResponse**(status)

Window Covering Commands

#define **emberAfFillCommandWindowCoveringClusterWindowCoveringUpOpen**()
#define **emberAfFillCommandWindowCoveringClusterWindowCoveringDownClose**()
#define **emberAfFillCommandWindowCoveringClusterWindowCoveringStop**()
#define **emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftSetpoint** (indexOfLiftSetpoint)
#define **emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftValue**(liftValue)
#define **emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftPercentage** (percentageLiftValue)
#define **emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltSetpoint** (indexOfTiltSetpoint)
#define **emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltValue**(tiltValue)
#define **emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltPercentage** (percentageTiltValue)
#define **emberAfFillCommandWindowCoveringClusterWindowCoveringGoToProgramSetpoint** (setpointType, setpointIndex, setpointValue)

Generic Tunnel Commands

```
#define emberAfFillCommandGenericTunnelClusterMatchProtocolAddress(protocolAddress)
#define emberAfFillCommandGenericTunnelClusterMatchProtocolAddressResponse()
#define emberAfFillCommandGenericTunnelClusterAdvertiseProtocolAddress(protocolAddress)
```

BACnet Protocol Tunnel Commands

```
#define emberAfFillCommandBacnetProtocolTunnelClusterTransferNpdu(NPDU)
```

Over the Air Bootloading Cluster Commands

```
#define emberAfFillCommandOtaBootloadClusterImageNotify(payloadType, queryJitter, manufacturerId,
        imageType, newFileVersion)
#define emberAfFillCommandOtaBootloadClusterQueryNextImageRequest(fieldControl, manufacturerId,
        imageType, currentFileVersion, hardwareVersion)
#define emberAfFillCommandOtaBootloadClusterQueryNextImageResponse(status, manufacturerId,
        imageType, fileVersion, imageSize)
#define emberAfFillCommandOtaBootloadClusterImageBlockRequest(fieldControl, manufacturerId,
        imageType, fileVersion, fileOffset, maxDataSize, requestNodeAddress)
#define emberAfFillCommandOtaBootloadClusterImagePageRequest(fieldControl, manufacturerId,
        imageType, fileVersion, fileOffset, maxDataSize, pageSize, responseSpacing, requestNodeAddress)
#define emberAfFillCommandOtaBootloadClusterImageBlockResponse(status, manufacturerId, imageType,
        fileVersion, fileOffset, dataSize, imageData)
#define emberAfFillCommandOtaBootloadClusterUpgradeEndRequest(status, manufacturerId, imageType,
        fileVersion)
#define emberAfFillCommandOtaBootloadClusterUpgradeEndResponse(manufacturerId, imageType,
        fileVersion, currentTime, upgradeTime)
#define emberAfFillCommandOtaBootloadClusterQuerySpecificFileRequest(requestNodeAddress,
        manufacturerId, imageType, fileVersion, currentZigbeeStackVersion)
#define emberAfFillCommandOtaBootloadClusterQuerySpecificFileResponse(status, manufacturerId,
        imageType, fileVersion, imageSize)
```

Demand Response and Load Control Commands

```
#define emberAfFillCommandDemandResponseLoadControlClusterLoadControlEvent(issuerEventId,
        deviceClass, utilityEnrollmentGroup, startTime, durationInMinutes, criticalityLevel,
        coolingTemperatureOffset, heatingTemperatureOffset, coolingTemperatureSetPoint,
        heatingTemperatureSetPoint, averageLoadAdjustmentPercentage, dutyCycle, eventControl)
#define emberAfFillCommandDemandResponseLoadControlClusterCancelLoadControlEvent(issuerEventId,
        deviceClass, utilityEnrollmentGroup, cancelControl, effectiveTime)
#define emberAfFillCommandDemandResponseLoadControlClusterCancelAllLoadControlEvents
        (cancelControl)
#define emberAfFillCommandDemandResponseLoadControlClusterReportEventStatus(issuerEventId,
        eventStatus, eventStatusTime, criticalityLevelApplied, coolingTemperatureSetPointApplied,
        heatingTemperatureSetPointApplied, averageLoadAdjustmentPercentageApplied, dutyCycleApplied,
        eventControl, signatureType, signature)
```

#define	**emberAfFillCommandDemandResponseLoadControlClusterGetScheduledEvents**(startTime, numberOfEvents)

Simple Metering Commands

#define	**emberAfFillCommandSimpleMeteringClusterGetProfileResponse**(endTime, status, profileIntervalPeriod, numberOfPeriodsDelivered, intervals, intervalsLen)
#define	**emberAfFillCommandSimpleMeteringClusterRequestMirror**()
#define	**emberAfFillCommandSimpleMeteringClusterRemoveMirror**()
#define	**emberAfFillCommandSimpleMeteringClusterRequestFastPollModeResponse**(appliedUpdatePeriod, fastPollModeEndtime)
#define	**emberAfFillCommandSimpleMeteringClusterGetProfile**(intervalChannel, endTime, numberOfPeriods)
#define	**emberAfFillCommandSimpleMeteringClusterRequestMirrorResponse**(endpointId)
#define	**emberAfFillCommandSimpleMeteringClusterMirrorRemoved**(endpointId)
#define	**emberAfFillCommandSimpleMeteringClusterRequestFastPollMode**(fastPollUpdatePeriod, duration)

Price Commands

#define	**emberAfFillCommandPriceClusterPublishPrice**(providerId, rateLabel, issuerEventId, currentTime, unitOfMeasure, currency, priceTrailingDigitAndPriceTier, numberOfPriceTiersAndRegisterTier, startTime, durationInMinutes, price, priceRatio, generationPrice, generationPriceRatio, alternateCostDelivered, alternateCostUnit, alternateCostTrailingDigit, numberOfBlockThresholds, priceControl)
#define	**emberAfFillCommandPriceClusterPublishBlockPeriod**(providerId, issuerEventId, blockPeriodStartTime, blockPeriodDurationInMinutes, numberOfPriceTiersAndNumberOfBlockThresholds, blockPeriodControl)
#define	**emberAfFillCommandPriceClusterGetCurrentPrice**(commandOptions)
#define	**emberAfFillCommandPriceClusterGetScheduledPrices**(startTime, numberOfEvents)
#define	**emberAfFillCommandPriceClusterPriceAcknowledgement**(providerId, issuerEventId, priceAckTime, control)
#define	**emberAfFillCommandPriceClusterGetBlockPeriods**(startTime, numberOfEvents)

Messaging Commands

#define	**emberAfFillCommandMessagingClusterDisplayMessage**(messageId, messageControl, startTime, durationInMinutes, message)
#define	**emberAfFillCommandMessagingClusterCancelMessage**(messageId, messageControl)
#define	**emberAfFillCommandMessagingClusterGetLastMessage**()
#define	**emberAfFillCommandMessagingClusterMessageConfirmation**(messageId, confirmationTime)

Tunneling Commands

#define	**emberAfFillCommandTunnelingClusterRequestTunnel**(protocolId, manufacturerCode, flowControlSupport)
#define	**emberAfFillCommandTunnelingClusterCloseTunnel**(tunnelId)
#define	**emberAfFillCommandTunnelingClusterTransferDataClientToServer**(tunnelId, data, dataLen)
#define	**emberAfFillCommandTunnelingClusterTransferDataErrorClientToServer**(tunnelId, transferDataStatus)
#define	**emberAfFillCommandTunnelingClusterAckTransferDataClientToServer**(tunnelId, numberOfBytesLeft)

#define **emberAfFillCommandTunnelingClusterReadyDataClientToServer**(tunnelId, numberOfOctetsLeft)

#define **emberAfFillCommandTunnelingClusterRequestTunnelResponse**(tunnelId, tunnelStatus)

#define **emberAfFillCommandTunnelingClusterTransferDataServerToClient**(tunnelId, data, dataLen)

#define **emberAfFillCommandTunnelingClusterTransferDataErrorServerToClient**(tunnelId, transferDataStatus)

#define **emberAfFillCommandTunnelingClusterAckTransferDataServerToClient**(tunnelId, numberOfBytesLeft)

#define **emberAfFillCommandTunnelingClusterReadyDataServerToClient**(tunnelId, numberOfOctetsLeft)

Prepayment Commands

#define **emberAfFillCommandPrepaymentClusterSelectAvailableEmergencyCredit**(commandDateTime, originatingDevice, siteId, meterSerialNumber)

#define **emberAfFillCommandPrepaymentClusterChangeSupply**(providerId, requestDateTime, siteId, meterSerialNumber, implementationDateTime, proposedSupplyStatus, originatorIdSupplyControlBits)

#define **emberAfFillCommandPrepaymentClusterSupplyStatusResponse**(providerId, implementationtDateTime, supplyStatus)

Key establishment Commands

#define **emberAfFillCommandKeyEstablishmentClusterInitiateKeyEstablishmentRequest**(keyEstablishmentSuite, ephemeralDataGenerateTime, confirmKeyGenerateTime, identity)

#define **emberAfFillCommandKeyEstablishmentClusterEphemeralDataRequest**(ephemeralData)

#define **emberAfFillCommandKeyEstablishmentClusterConfirmKeyDataRequest**(secureMessageAuthenticationCode)

#define **emberAfFillCommandKeyEstablishmentClusterServerToClientTerminateKeyEstablishment**(statusCode, waitTime, keyEstablishmentSuite)

#define **emberAfFillCommandKeyEstablishmentClusterClientToServerTerminateKeyEstablishment**(statusCode, waitTime, keyEstablishmentSuite)

#define **emberAfFillCommandKeyEstablishmentClusterInitiateKeyEstablishmentResponse**(requestedKeyEstablishmentSuite, ephemeralDataGenerateTime, confirmKeyGenerateTime, identity)

#define **emberAfFillCommandKeyEstablishmentClusterEphemeralDataResponse**(ephemeralData)

#define **emberAfFillCommandKeyEstablishmentClusterConfirmKeyDataResponse**(secureMessageAuthenticationCode)

Sample Mfg Specific Cluster Commands

#define **emberAfFillCommandSampleMfgSpecificClusterCommandOne**(argOne)

cluster-id.h File Reference

Go to the source code of this file.

Defines

```
#define ZCL_BASIC_CLUSTER_ID
#define ZCL_POWER_CONFIG_CLUSTER_ID
#define ZCL_DEVICE_TEMP_CLUSTER_ID
#define ZCL_IDENTIFY_CLUSTER_ID
#define ZCL_GROUPS_CLUSTER_ID
#define ZCL_SCENES_CLUSTER_ID
#define ZCL_ON_OFF_CLUSTER_ID
#define ZCL_ON_OFF_SWITCH_CONFIG_CLUSTER_ID
#define ZCL_LEVEL_CONTROL_CLUSTER_ID
#define ZCL_ALARM_CLUSTER_ID
#define ZCL_TIME_CLUSTER_ID
#define ZCL_RSSI_LOCATION_CLUSTER_ID
#define ZCL_BINARY_INPUT_BASIC_CLUSTER_ID
#define ZCL_COMMISSIONING_CLUSTER_ID
#define ZCL_SHADE_CONFIG_CLUSTER_ID
#define ZCL_PUMP_CONFIG_CONTROL_CLUSTER_ID
#define ZCL_THERMOSTAT_CLUSTER_ID
#define ZCL_FAN_CONTROL_CLUSTER_ID
#define ZCL_DEHUMID_CONTROL_CLUSTER_ID
#define ZCL_THERMOSTAT_UI_CONFIG_CLUSTER_ID
#define ZCL_COLOR_CONTROL_CLUSTER_ID
#define ZCL_BALLAST_CONFIGURATION_CLUSTER_ID
#define ZCL_ILLUM_MEASUREMENT_CLUSTER_ID
#define ZCL_ILLUM_LEVEL_SENSING_CLUSTER_ID
#define ZCL_TEMP_MEASUREMENT_CLUSTER_ID
#define ZCL_PRESSURE_MEASUREMENT_CLUSTER_ID
#define ZCL_FLOW_MEASUREMENT_CLUSTER_ID
#define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER_ID
#define ZCL_OCCUPANCY_SENSING_CLUSTER_ID
#define ZCL_IAS_ZONE_CLUSTER_ID
#define ZCL_IAS_ACE_CLUSTER_ID
#define ZCL_IAS_WD_CLUSTER_ID
```

#define **ZCL_DOOR_LOCK_CLUSTER_ID**
#define **ZCL_WINDOW_COVERING_CLUSTER_ID**
#define **ZCL_GENERIC_TUNNEL_CLUSTER_ID**
#define **ZCL_BACNET_PROTOCOL_TUNNEL_CLUSTER_ID**
#define **ZCL_OTA_BOOTLOAD_CLUSTER_ID**
#define **ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID**
#define **ZCL_SIMPLE_METERING_CLUSTER_ID**
#define **ZCL_PRICE_CLUSTER_ID**
#define **ZCL_MESSAGING_CLUSTER_ID**
#define **ZCL_TUNNELING_CLUSTER_ID**
#define **ZCL_PREPAYMENT_CLUSTER_ID**
#define **ZCL_KEY_ESTABLISHMENT_CLUSTER_ID**
#define **ZCL_SAMPLE_MFG_SPECIFIC_CLUSTER_ID**

Define Documentation

#define ZCL_BASIC_CLUSTER_ID

Definition at line **25** of file **cluster-id.h**.

#define ZCL_POWER_CONFIG_CLUSTER_ID

Definition at line **26** of file **cluster-id.h**.

#define ZCL_DEVICE_TEMP_CLUSTER_ID

Definition at line **27** of file **cluster-id.h**.

#define ZCL_IDENTIFY_CLUSTER_ID

Definition at line **28** of file **cluster-id.h**.

#define ZCL_GROUPS_CLUSTER_ID

Definition at line **29** of file **cluster-id.h**.

#define ZCL_SCENES_CLUSTER_ID

Definition at line **30** of file **cluster-id.h**.

#define ZCL_ON_OFF_CLUSTER_ID

Definition at line **31** of file **cluster-id.h**.

#define ZCL_ON_OFF_SWITCH_CONFIG_CLUSTER_ID

Definition at line **32** of file **cluster-id.h**.

#define ZCL_LEVEL_CONTROL_CLUSTER_ID

Definition at line **33** of file **cluster-id.h**.

#define ZCL_ALARM_CLUSTER_ID

Definition at line **34** of file **cluster-id.h**.

#define ZCL_TIME_CLUSTER_ID

Definition at line **35** of file **cluster-id.h**.

#define ZCL_RSSI_LOCATION_CLUSTER_ID

Definition at line **36** of file **cluster-id.h**.

#define ZCL_BINARY_INPUT_BASIC_CLUSTER_ID

Definition at line **37** of file **cluster-id.h**.

#define ZCL_COMMISSIONING_CLUSTER_ID

Definition at line **38** of file **cluster-id.h**.

#define ZCL_SHADE_CONFIG_CLUSTER_ID

Definition at line **39** of file **cluster-id.h**.

#define ZCL_PUMP_CONFIG_CONTROL_CLUSTER_ID

Definition at line **40** of file **cluster-id.h**.

#define ZCL_THERMOSTAT_CLUSTER_ID

Definition at line **41** of file **cluster-id.h**.

#define ZCL_FAN_CONTROL_CLUSTER_ID

Definition at line **42** of file **cluster-id.h**.

#define ZCL_DEHUMID_CONTROL_CLUSTER_ID

Definition at line **43** of file **cluster-id.h**.

#define ZCL_THERMOSTAT_UI_CONFIG_CLUSTER_ID

Definition at line **44** of file **cluster-id.h**.

#define ZCL_COLOR_CONTROL_CLUSTER_ID

Definition at line **45** of file **cluster-id.h**.

#define ZCL_BALLAST_CONFIGURATION_CLUSTER_ID

Definition at line **46** of file **cluster-id.h**.

#define ZCL_ILLUM_MEASUREMENT_CLUSTER_ID

Definition at line **47** of file **cluster-id.h**.

#define ZCL_ILLUM_LEVEL_SENSING_CLUSTER_ID

Definition at line **48** of file **cluster-id.h**.

#define ZCL_TEMP_MEASUREMENT_CLUSTER_ID

Definition at line **49** of file **cluster-id.h**.

#define ZCL_PRESSURE_MEASUREMENT_CLUSTER_ID

Definition at line **50** of file **cluster-id.h**.

#define ZCL_FLOW_MEASUREMENT_CLUSTER_ID

Definition at line **51** of file **cluster-id.h**.

#define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER_ID

Definition at line **52** of file **cluster-id.h**.

#define ZCL_OCCUPANCY_SENSING_CLUSTER_ID

Definition at line **53** of file **cluster-id.h**.

#define ZCL_IAS_ZONE_CLUSTER_ID

Definition at line **54** of file **cluster-id.h**.

#define ZCL_IAS_ACE_CLUSTER_ID

Definition at line **55** of file **cluster-id.h**.

#define ZCL_IAS_WD_CLUSTER_ID

Definition at line **56** of file **cluster-id.h**.

#define ZCL_DOOR_LOCK_CLUSTER_ID

Definition at line **57** of file **cluster-id.h**.

#define ZCL_WINDOW_COVERING_CLUSTER_ID

Definition at line **58** of file **cluster-id.h**.

#define ZCL_GENERIC_TUNNEL_CLUSTER_ID

Definition at line **59** of file **cluster-id.h**.

#define ZCL_BACNET_PROTOCOL_TUNNEL_CLUSTER_ID

Definition at line **60** of file **cluster-id.h**.

#define ZCL_OTA_BOOTLOAD_CLUSTER_ID

Definition at line **61** of file **cluster-id.h**.

#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID

Definition at line **62** of file **cluster-id.h**.

#define ZCL_SIMPLE_METERING_CLUSTER_ID

Definition at line **63** of file **cluster-id.h**.

#define ZCL_PRICE_CLUSTER_ID

Definition at line **64** of file **cluster-id.h**.

#define ZCL_MESSAGING_CLUSTER_ID

Definition at line **65** of file **cluster-id.h**.

#define ZCL_TUNNELING_CLUSTER_ID

Definition at line **66** of file **cluster-id.h**.

#define ZCL_PREPAYMENT_CLUSTER_ID

Definition at line **67** of file **cluster-id.h**.

#define ZCL_KEY_ESTABLISHMENT_CLUSTER_ID

Definition at line **68** of file **cluster-id.h**.

#define ZCL_SAMPLE_MFG_SPECIFIC_CLUSTER_ID

Definition at line **69** of file **cluster-id.h**.

---

color-control.h File Reference

Go to the source code of this file.

Defines

#define ZCL_COLOR_CONTROL_HUE_STEP_UNIT
#define ZCL_COLOR_CONTROL_SATURATION_STEP_UNIT
#define ZCL_INVALID_COLOR_CONTROL_COMMAND_ID

Define Documentation

#define ZCL_COLOR_CONTROL_HUE_STEP_UNIT

Definition at line 15 of file color-control.h.

#define ZCL_COLOR_CONTROL_SATURATION_STEP_UNIT

Definition at line 16 of file color-control.h.

#define ZCL_INVALID_COLOR_CONTROL_COMMAND_ID

Definition at line 17 of file color-control.h.

---

command-id.h File Reference

Go to the source code of this file.

Defines

```
#define ZCL_READ_ATTRIBUTES_COMMAND_ID
#define ZCL_READ_ATTRIBUTES_RESPONSE_COMMAND_ID
#define ZCL_WRITE_ATTRIBUTES_COMMAND_ID
#define ZCL_WRITE_ATTRIBUTES_UNDIVIDED_COMMAND_ID
#define ZCL_WRITE_ATTRIBUTES_RESPONSE_COMMAND_ID
#define ZCL_WRITE_ATTRIBUTES_NO_RESPONSE_COMMAND_ID
#define ZCL_CONFIGURE_REPORTING_COMMAND_ID
#define ZCL_CONFIGURE_REPORTING_RESPONSE_COMMAND_ID
#define ZCL_READ_REPORTING_CONFIGURATION_COMMAND_ID
#define ZCL_READ_REPORTING_CONFIGURATION_RESPONSE_COMMAND_ID
#define ZCL_REPORT_ATTRIBUTES_COMMAND_ID
#define ZCL_DEFAULT_RESPONSE_COMMAND_ID
#define ZCL_DISCOVER_ATTRIBUTES_COMMAND_ID
#define ZCL_DISCOVER_ATTRIBUTES_RESPONSE_COMMAND_ID
#define ZCL_RESET_TO_FACTORY_DEFAULTS_COMMAND_ID
#define ZCL_IDENTIFY_QUERY_RESPONSE_COMMAND_ID
#define ZCL_IDENTIFY_COMMAND_ID
#define ZCL_IDENTIFY_QUERY_COMMAND_ID
#define ZCL_ADD_GROUP_RESPONSE_COMMAND_ID
#define ZCL_VIEW_GROUP_RESPONSE_COMMAND_ID
#define ZCL_GET_GROUP_MEMBERSHIP_RESPONSE_COMMAND_ID
#define ZCL_REMOVE_GROUP_RESPONSE_COMMAND_ID
#define ZCL_ADD_GROUP_COMMAND_ID
#define ZCL_VIEW_GROUP_COMMAND_ID
#define ZCL_GET_GROUP_MEMBERSHIP_COMMAND_ID
#define ZCL_REMOVE_GROUP_COMMAND_ID
#define ZCL_REMOVE_ALL_GROUPS_COMMAND_ID
#define ZCL_ADD_GROUP_IF_IDENTIFYING_COMMAND_ID
#define ZCL_ADD_SCENE_RESPONSE_COMMAND_ID
#define ZCL_VIEW_SCENE_RESPONSE_COMMAND_ID
#define ZCL_REMOVE_SCENE_RESPONSE_COMMAND_ID
#define ZCL_REMOVE_ALL_SCENES_RESPONSE_COMMAND_ID
#define ZCL_STORE_SCENE_RESPONSE_COMMAND_ID
#define ZCL_GET_SCENE_MEMBERSHIP_RESPONSE_COMMAND_ID
#define ZCL_ADD_SCENE_COMMAND_ID
#define ZCL_VIEW_SCENE_COMMAND_ID
```

```
#define ZCL_REMOVE_SCENE_COMMAND_ID
#define ZCL_REMOVE_ALL_SCENES_COMMAND_ID
#define ZCL_STORE_SCENE_COMMAND_ID
#define ZCL_RECALL_SCENE_COMMAND_ID
#define ZCL_GET_SCENE_MEMBERSHIP_COMMAND_ID
#define ZCL_OFF_COMMAND_ID
#define ZCL_ON_COMMAND_ID
#define ZCL_TOGGLE_COMMAND_ID
#define ZCL_SAMPLE_MFG_SPECIFIC_OFF_WITH_TRANSITION_COMMAND_ID
#define ZCL_SAMPLE_MFG_SPECIFIC_ON_WITH_TRANSITION_COMMAND_ID
#define ZCL_SAMPLE_MFG_SPECIFIC_TOGGLE_WITH_TRANSITION_COMMAND_ID
#define ZCL_MOVE_TO_LEVEL_COMMAND_ID
#define ZCL_MOVE_COMMAND_ID
#define ZCL_STEP_COMMAND_ID
#define ZCL_STOP_COMMAND_ID
#define ZCL_MOVE_TO_LEVEL_WITH_ON_OFF_COMMAND_ID
#define ZCL_MOVE_WITH_ON_OFF_COMMAND_ID
#define ZCL_STEP_WITH_ON_OFF_COMMAND_ID
#define ZCL_STOP_WITH_ON_OFF_COMMAND_ID
#define ZCL_ALARM_COMMAND_ID
#define ZCL_GET_ALARM_RESPONSE_COMMAND_ID
#define ZCL_RESET_ALARM_COMMAND_ID
#define ZCL_RESET_ALL_ALARMS_COMMAND_ID
#define ZCL_GET_ALARM_COMMAND_ID
#define ZCL_RESET_ALARM_LOG_COMMAND_ID
#define ZCL_DEVICE_CONFIGURATION_RESPONSE_COMMAND_ID
#define ZCL_LOCATION_DATA_RESPONSE_COMMAND_ID
#define ZCL_LOCATION_DATA_NOTIFICATION_COMMAND_ID
#define ZCL_COMPACT_LOCATION_DATA_NOTIFICATION_COMMAND_ID
#define ZCL_RSSI_PING_COMMAND_ID
#define ZCL_RSSI_REQUEST_COMMAND_ID
#define ZCL_REPORT_RSSI_MEASUREMENTS_COMMAND_ID
#define ZCL_REQUEST_OWN_LOCATION_COMMAND_ID
#define ZCL_SET_ABSOLUTE_LOCATION_COMMAND_ID
#define ZCL_SET_DEVICE_CONFIGURATION_COMMAND_ID
#define ZCL_GET_DEVICE_CONFIGURATION_COMMAND_ID
#define ZCL_GET_LOCATION_DATA_COMMAND_ID
#define ZCL_RSSI_RESPONSE_COMMAND_ID
#define ZCL_SEND_PINGS_COMMAND_ID
#define ZCL_ANCHOR_NODE_ANNOUNCE_COMMAND_ID
#define ZCL_RESTART_DEVICE_RESPONSE_COMMAND_ID
#define ZCL_SAVE_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID
#define ZCL_RESTORE_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID
#define ZCL_RESET_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID
#define ZCL_RESTART_DEVICE_COMMAND_ID
```

```
#define ZCL_SAVE_STARTUP_PARAMETERS_COMMAND_ID
#define ZCL_RESTORE_STARTUP_PARAMETERS_COMMAND_ID
#define ZCL_RESET_STARTUP_PARAMETERS_COMMAND_ID
#define ZCL_SETPOINT_RAISE_LOWER_COMMAND_ID
#define ZCL_MOVE_TO_HUE_COMMAND_ID
#define ZCL_MOVE_HUE_COMMAND_ID
#define ZCL_STEP_HUE_COMMAND_ID
#define ZCL_MOVE_TO_SATURATION_COMMAND_ID
#define ZCL_MOVE_SATURATION_COMMAND_ID
#define ZCL_STEP_SATURATION_COMMAND_ID
#define ZCL_MOVE_TO_HUE_AND_SATURATION_COMMAND_ID
#define ZCL_MOVE_TO_COLOR_COMMAND_ID
#define ZCL_MOVE_COLOR_COMMAND_ID
#define ZCL_STEP_COLOR_COMMAND_ID
#define ZCL_MOVE_TO_COLOR_TEMPERATURE_COMMAND_ID
#define ZCL_ZONE_STATUS_CHANGE_NOTIFICATION_COMMAND_ID
#define ZCL_ZONE_ENROLL_REQUEST_COMMAND_ID
#define ZCL_ZONE_ENROLL_RESPONSE_COMMAND_ID
#define ZCL_ARM_RESPONSE_COMMAND_ID
#define ZCL_GET_ZONE_ID_MAP_RESPONSE_COMMAND_ID
#define ZCL_GET_ZONE_INFORMATION_RESPONSE_COMMAND_ID
#define ZCL_ARM_COMMAND_ID
#define ZCL_BYPASS_COMMAND_ID
#define ZCL_EMERGENCY_COMMAND_ID
#define ZCL_FIRE_COMMAND_ID
#define ZCL_PANIC_COMMAND_ID
#define ZCL_GET_ZONE_ID_MAP_COMMAND_ID
#define ZCL_GET_ZONE_INFORMATION_COMMAND_ID
#define ZCL_START_WARNING_COMMAND_ID
#define ZCL_SQUAWK_COMMAND_ID
#define ZCL_LOCK_DOOR_RESPONSE_COMMAND_ID
#define ZCL_UNLOCK_DOOR_RESPONSE_COMMAND_ID
#define ZCL_LOCK_DOOR_COMMAND_ID
#define ZCL_UNLOCK_DOOR_COMMAND_ID
#define ZCL_WINDOW_COVERING_UP_OPEN_COMMAND_ID
#define ZCL_WINDOW_COVERING_DOWN_CLOSE_COMMAND_ID
#define ZCL_WINDOW_COVERING_STOP_COMMAND_ID
#define ZCL_WINDOW_COVERING_GO_TO_LIFT_SETPOINT_COMMAND_ID
#define ZCL_WINDOW_COVERING_GO_TO_LIFT_VALUE_COMMAND_ID
#define ZCL_WINDOW_COVERING_GO_TO_LIFT_PERCENTAGE_COMMAND_ID
#define ZCL_WINDOW_COVERING_GO_TO_TILT_SETPOINT_COMMAND_ID
#define ZCL_WINDOW_COVERING_GO_TO_TILT_VALUE_COMMAND_ID
#define ZCL_WINDOW_COVERING_GO_TO_TILT_PERCENTAGE_COMMAND_ID
#define ZCL_WINDOW_COVERING_GO_TO_PROGRAM_SETPOINT_COMMAND_ID
#define ZCL_MATCH_PROTOCOL_ADDRESS_RESPONSE_COMMAND_ID
```

```
#define ZCL_ADVERTISE_PROTOCOL_ADDRESS_COMMAND_ID
#define ZCL_MATCH_PROTOCOL_ADDRESS_COMMAND_ID
#define ZCL_TRANSFER_NPDU_COMMAND_ID
#define ZCL_IMAGE_NOTIFY_COMMAND_ID
#define ZCL_QUERY_NEXT_IMAGE_RESPONSE_COMMAND_ID
#define ZCL_IMAGE_BLOCK_RESPONSE_COMMAND_ID
#define ZCL_UPGRADE_END_RESPONSE_COMMAND_ID
#define ZCL_QUERY_SPECIFIC_FILE_RESPONSE_COMMAND_ID
#define ZCL_QUERY_NEXT_IMAGE_REQUEST_COMMAND_ID
#define ZCL_IMAGE_BLOCK_REQUEST_COMMAND_ID
#define ZCL_IMAGE_PAGE_REQUEST_COMMAND_ID
#define ZCL_UPGRADE_END_REQUEST_COMMAND_ID
#define ZCL_QUERY_SPECIFIC_FILE_REQUEST_COMMAND_ID
#define ZCL_LOAD_CONTROL_EVENT_COMMAND_ID
#define ZCL_CANCEL_LOAD_CONTROL_EVENT_COMMAND_ID
#define ZCL_CANCEL_ALL_LOAD_CONTROL_EVENTS_COMMAND_ID
#define ZCL_REPORT_EVENT_STATUS_COMMAND_ID
#define ZCL_GET_SCHEDULED_EVENTS_COMMAND_ID
#define ZCL_GET_PROFILE_RESPONSE_COMMAND_ID
#define ZCL_REQUEST_MIRROR_COMMAND_ID
#define ZCL_REMOVE_MIRROR_COMMAND_ID
#define ZCL_REQUEST_FAST_POLL_MODE_RESPONSE_COMMAND_ID
#define ZCL_GET_PROFILE_COMMAND_ID
#define ZCL_REQUEST_MIRROR_RESPONSE_COMMAND_ID
#define ZCL_MIRROR_REMOVED_COMMAND_ID
#define ZCL_REQUEST_FAST_POLL_MODE_COMMAND_ID
#define ZCL_PUBLISH_PRICE_COMMAND_ID
#define ZCL_PUBLISH_BLOCK_PERIOD_COMMAND_ID
#define ZCL_GET_CURRENT_PRICE_COMMAND_ID
#define ZCL_GET_SCHEDULED_PRICES_COMMAND_ID
#define ZCL_PRICE_ACKNOWLEDGEMENT_COMMAND_ID
#define ZCL_GET_BLOCK_PERIODS_COMMAND_ID
#define ZCL_DISPLAY_MESSAGE_COMMAND_ID
#define ZCL_CANCEL_MESSAGE_COMMAND_ID
#define ZCL_GET_LAST_MESSAGE_COMMAND_ID
#define ZCL_MESSAGE_CONFIRMATION_COMMAND_ID
#define ZCL_REQUEST_TUNNEL_RESPONSE_COMMAND_ID
#define ZCL_TRANSFER_DATA_SERVER_TO_CLIENT_COMMAND_ID
#define ZCL_TRANSFER_DATA_ERROR_SERVER_TO_CLIENT_COMMAND_ID
#define ZCL_ACK_TRANSFER_DATA_SERVER_TO_CLIENT_COMMAND_ID
#define ZCL_READY_DATA_SERVER_TO_CLIENT_COMMAND_ID
#define ZCL_REQUEST_TUNNEL_COMMAND_ID
#define ZCL_CLOSE_TUNNEL_COMMAND_ID
#define ZCL_TRANSFER_DATA_CLIENT_TO_SERVER_COMMAND_ID
#define ZCL_TRANSFER_DATA_ERROR_CLIENT_TO_SERVER_COMMAND_ID
#define ZCL_ACK_TRANSFER_DATA_CLIENT_TO_SERVER_COMMAND_ID
```

#define **ZCL_READY_DATA_CLIENT_TO_SERVER_COMMAND_ID**
#define **ZCL_SUPPLY_STATUS_RESPONSE_COMMAND_ID**
#define **ZCL_SELECT_AVAILABLE_EMERGENCY_CREDIT_COMMAND_ID**
#define **ZCL_CHANGE_SUPPLY_COMMAND_ID**
#define **ZCL_INITIATE_KEY_ESTABLISHMENT_RESPONSE_COMMAND_ID**
#define **ZCL_EPHEMERAL_DATA_RESPONSE_COMMAND_ID**
#define **ZCL_CONFIRM_KEY_DATA_RESPONSE_COMMAND_ID**
#define **ZCL_INITIATE_KEY_ESTABLISHMENT_REQUEST_COMMAND_ID**
#define **ZCL_EPHEMERAL_DATA_REQUEST_COMMAND_ID**
#define **ZCL_CONFIRM_KEY_DATA_REQUEST_COMMAND_ID**
#define **ZCL_TERMINATE_KEY_ESTABLISHMENT_COMMAND_ID**
#define **ZCL_COMMAND_ONE_COMMAND_ID**

Define Documentation

#define ZCL_READ_ATTRIBUTES_COMMAND_ID

Definition at line **17** of file **command-id.h**.

#define ZCL_READ_ATTRIBUTES_RESPONSE_COMMAND_ID

Definition at line **18** of file **command-id.h**.

#define ZCL_WRITE_ATTRIBUTES_COMMAND_ID

Definition at line **19** of file **command-id.h**.

#define ZCL_WRITE_ATTRIBUTES_UNDIVIDED_COMMAND_ID

Definition at line **20** of file **command-id.h**.

#define ZCL_WRITE_ATTRIBUTES_RESPONSE_COMMAND_ID

Definition at line **21** of file **command-id.h**.

#define ZCL_WRITE_ATTRIBUTES_NO_RESPONSE_COMMAND_ID

Definition at line **22** of file **command-id.h**.

#define ZCL_CONFIGURE_REPORTING_COMMAND_ID

Definition at line **23** of file **command-id.h**.

#define ZCL_CONFIGURE_REPORTING_RESPONSE_COMMAND_ID

Definition at line **24** of file **command-id.h**.

#define ZCL_READ_REPORTING_CONFIGURATION_COMMAND_ID

Definition at line **25** of file **command-id.h**.

#define ZCL_READ_REPORTING_CONFIGURATION_RESPONSE_COMMAND_ID

Definition at line **26** of file **command-id.h**.

#define ZCL_REPORT_ATTRIBUTES_COMMAND_ID

Definition at line **27** of file **command-id.h**.

#define ZCL_DEFAULT_RESPONSE_COMMAND_ID

Definition at line **28** of file **command-id.h**.

#define ZCL_DISCOVER_ATTRIBUTES_COMMAND_ID

Definition at line **29** of file **command-id.h**.

#define ZCL_DISCOVER_ATTRIBUTES_RESPONSE_COMMAND_ID

Definition at line **30** of file **command-id.h**.

#define ZCL_RESET_TO_FACTORY_DEFAULTS_COMMAND_ID

Definition at line **35** of file **command-id.h**.

#define ZCL_IDENTIFY_QUERY_RESPONSE_COMMAND_ID

Definition at line **42** of file **command-id.h**.

#define ZCL_IDENTIFY_COMMAND_ID

Definition at line **45** of file **command-id.h**.

#define ZCL_IDENTIFY_QUERY_COMMAND_ID

Definition at line **46** of file **command-id.h**.

#define ZCL_ADD_GROUP_RESPONSE_COMMAND_ID

Definition at line **53** of file **command-id.h**.

#define ZCL_VIEW_GROUP_RESPONSE_COMMAND_ID

Definition at line **54** of file **command-id.h**.

#define ZCL_GET_GROUP_MEMBERSHIP_RESPONSE_COMMAND_ID

Definition at line **55** of file **command-id.h**.

#define ZCL_REMOVE_GROUP_RESPONSE_COMMAND_ID

Definition at line **56** of file **command-id.h**.

#define ZCL_ADD_GROUP_COMMAND_ID

Definition at line **59** of file **command-id.h**.

#define ZCL_VIEW_GROUP_COMMAND_ID

Definition at line **60** of file **command-id.h**.

#define ZCL_GET_GROUP_MEMBERSHIP_COMMAND_ID

Definition at line **61** of file **command-id.h**.

#define ZCL_REMOVE_GROUP_COMMAND_ID

Definition at line **62** of file **command-id.h**.

#define ZCL_REMOVE_ALL_GROUPS_COMMAND_ID

Definition at line **63** of file **command-id.h**.

#define ZCL_ADD_GROUP_IF_IDENTIFYING_COMMAND_ID

Definition at line **64** of file **command-id.h**.

#define ZCL_ADD_SCENE_RESPONSE_COMMAND_ID

Definition at line **71** of file **command-id.h**.

#define ZCL_VIEW_SCENE_RESPONSE_COMMAND_ID

Definition at line **72** of file **command-id.h**.

#define ZCL_REMOVE_SCENE_RESPONSE_COMMAND_ID

Definition at line **73** of file **command-id.h**.

#define ZCL_REMOVE_ALL_SCENES_RESPONSE_COMMAND_ID

Definition at line **74** of file **command-id.h**.

#define ZCL_STORE_SCENE_RESPONSE_COMMAND_ID

Definition at line **75** of file **command-id.h**.

#define ZCL_GET_SCENE_MEMBERSHIP_RESPONSE_COMMAND_ID

Definition at line **76** of file **command-id.h**.

#define ZCL_ADD_SCENE_COMMAND_ID

Definition at line **79** of file **command-id.h**.

#define ZCL_VIEW_SCENE_COMMAND_ID

Definition at line **80** of file **command-id.h**.

#define ZCL_REMOVE_SCENE_COMMAND_ID

Definition at line **81** of file **command-id.h**.

#define ZCL_REMOVE_ALL_SCENES_COMMAND_ID

Definition at line **82** of file **command-id.h**.

#define ZCL_STORE_SCENE_COMMAND_ID

Definition at line **83** of file **command-id.h**.

#define ZCL_RECALL_SCENE_COMMAND_ID

Definition at line **84** of file **command-id.h**.

#define ZCL_GET_SCENE_MEMBERSHIP_COMMAND_ID

Definition at line **85** of file **command-id.h**.

#define ZCL_OFF_COMMAND_ID

Definition at line **92** of file **command-id.h**.

#define ZCL_ON_COMMAND_ID

Definition at line **93** of file **command-id.h**.

#define ZCL_TOGGLE_COMMAND_ID

Definition at line **94** of file **command-id.h**.

#define ZCL_SAMPLE_MFG_SPECIFIC_OFF_WITH_TRANSITION_COMMAND_ID

Definition at line **95** of file **command-id.h**.

#define ZCL_SAMPLE_MFG_SPECIFIC_ON_WITH_TRANSITION_COMMAND_ID

Definition at line **96** of file **command-id.h**.

#define ZCL_SAMPLE_MFG_SPECIFIC_TOGGLE_WITH_TRANSITION_COMMAND_ID

Definition at line **97** of file **command-id.h**.

#define ZCL_MOVE_TO_LEVEL_COMMAND_ID

Definition at line **104** of file **command-id.h**.

#define ZCL_MOVE_COMMAND_ID

Definition at line **105** of file **command-id.h**.

#define ZCL_STEP_COMMAND_ID

Definition at line **106** of file **command-id.h**.

#define ZCL_STOP_COMMAND_ID

Definition at line **107** of file **command-id.h**.

#define ZCL_MOVE_TO_LEVEL_WITH_ON_OFF_COMMAND_ID

Definition at line **108** of file **command-id.h**.

#define ZCL_MOVE_WITH_ON_OFF_COMMAND_ID

Definition at line **109** of file **command-id.h**.

#define ZCL_STEP_WITH_ON_OFF_COMMAND_ID

Definition at line **110** of file **command-id.h**.

#define ZCL_STOP_WITH_ON_OFF_COMMAND_ID

Definition at line **111** of file **command-id.h**.

#define ZCL_ALARM_COMMAND_ID

Definition at line **118** of file **command-id.h**.

#define ZCL_GET_ALARM_RESPONSE_COMMAND_ID

Definition at line **119** of file **command-id.h**.

#define ZCL_RESET_ALARM_COMMAND_ID

Definition at line **122** of file **command-id.h**.

#define ZCL_RESET_ALL_ALARMS_COMMAND_ID

Definition at line **123** of file **command-id.h**.

#define ZCL_GET_ALARM_COMMAND_ID

Definition at line **124** of file **command-id.h**.

#define ZCL_RESET_ALARM_LOG_COMMAND_ID

Definition at line **125** of file **command-id.h**.

#define ZCL_DEVICE_CONFIGURATION_RESPONSE_COMMAND_ID

Definition at line **132** of file **command-id.h**.

#define ZCL_LOCATION_DATA_RESPONSE_COMMAND_ID

Definition at line **133** of file **command-id.h**.

#define ZCL_LOCATION_DATA_NOTIFICATION_COMMAND_ID

Definition at line **134** of file **command-id.h**.

#define ZCL_COMPACT_LOCATION_DATA_NOTIFICATION_COMMAND_ID

Definition at line **135** of file **command-id.h**.

#define ZCL_RSSI_PING_COMMAND_ID

Definition at line **136** of file **command-id.h**.

#define ZCL_RSSI_REQUEST_COMMAND_ID

Definition at line **137** of file **command-id.h**.

#define ZCL_REPORT_RSSI_MEASUREMENTS_COMMAND_ID

Definition at line **138** of file **command-id.h**.

#define ZCL_REQUEST_OWN_LOCATION_COMMAND_ID

Definition at line **139** of file **command-id.h**.

#define ZCL_SET_ABSOLUTE_LOCATION_COMMAND_ID

Definition at line **142** of file **command-id.h**.

#define ZCL_SET_DEVICE_CONFIGURATION_COMMAND_ID

Definition at line **143** of file **command-id.h**.

#define ZCL_GET_DEVICE_CONFIGURATION_COMMAND_ID

Definition at line **144** of file **command-id.h**.

#define ZCL_GET_LOCATION_DATA_COMMAND_ID

Definition at line **145** of file **command-id.h**.

#define ZCL_RSSI_RESPONSE_COMMAND_ID

Definition at line **146** of file **command-id.h**.

#define ZCL_SEND_PINGS_COMMAND_ID

Definition at line **147** of file **command-id.h**.

#define ZCL_ANCHOR_NODE_ANNOUNCE_COMMAND_ID

Definition at line **148** of file **command-id.h**.

#define ZCL_RESTART_DEVICE_RESPONSE_COMMAND_ID

Definition at line **155** of file **command-id.h**.

#define ZCL_SAVE_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID

Definition at line **156** of file **command-id.h**.

#define ZCL_RESTORE_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID

Definition at line **157** of file **command-id.h**.

#define ZCL_RESET_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID

Definition at line **158** of file **command-id.h**.

#define ZCL_RESTART_DEVICE_COMMAND_ID

Definition at line **161** of file **command-id.h**.

#define ZCL_SAVE_STARTUP_PARAMETERS_COMMAND_ID

Definition at line **162** of file **command-id.h**.

#define ZCL_RESTORE_STARTUP_PARAMETERS_COMMAND_ID

Definition at line **163** of file **command-id.h**.

#define ZCL_RESET_STARTUP_PARAMETERS_COMMAND_ID

Definition at line **164** of file **command-id.h**.

#define ZCL_SETPOINT_RAISE_LOWER_COMMAND_ID

Definition at line **171** of file **command-id.h**.

#define ZCL_MOVE_TO_HUE_COMMAND_ID

Definition at line **178** of file **command-id.h**.

#define ZCL_MOVE_HUE_COMMAND_ID

Definition at line **179** of file **command-id.h**.

#define ZCL_STEP_HUE_COMMAND_ID

Definition at line **180** of file **command-id.h**.

#define ZCL_MOVE_TO_SATURATION_COMMAND_ID

Definition at line **181** of file **command-id.h**.

#define ZCL_MOVE_SATURATION_COMMAND_ID

Definition at line **182** of file **command-id.h**.

#define ZCL_STEP_SATURATION_COMMAND_ID

Definition at line **183** of file **command-id.h**.

#define ZCL_MOVE_TO_HUE_AND_SATURATION_COMMAND_ID

Definition at line **184** of file **command-id.h**.

#define ZCL_MOVE_TO_COLOR_COMMAND_ID

Definition at line **185** of file **command-id.h**.

#define ZCL_MOVE_COLOR_COMMAND_ID

Definition at line **186** of file **command-id.h**.

#define ZCL_STEP_COLOR_COMMAND_ID

Definition at line **187** of file **command-id.h**.

#define ZCL_MOVE_TO_COLOR_TEMPERATURE_COMMAND_ID

Definition at line **188** of file **command-id.h**.

#define ZCL_ZONE_STATUS_CHANGE_NOTIFICATION_COMMAND_ID

Definition at line **195** of file **command-id.h**.

#define ZCL_ZONE_ENROLL_REQUEST_COMMAND_ID

Definition at line **196** of file **command-id.h**.

#define ZCL_ZONE_ENROLL_RESPONSE_COMMAND_ID

Definition at line **199** of file **command-id.h**.

#define ZCL_ARM_RESPONSE_COMMAND_ID

Definition at line **206** of file **command-id.h**.

#define ZCL_GET_ZONE_ID_MAP_RESPONSE_COMMAND_ID

Definition at line **207** of file **command-id.h**.

#define ZCL_GET_ZONE_INFORMATION_RESPONSE_COMMAND_ID

Definition at line **208** of file **command-id.h**.

#define ZCL_ARM_COMMAND_ID

Definition at line **211** of file **command-id.h**.

#define ZCL_BYPASS_COMMAND_ID

Definition at line **212** of file **command-id.h**.

#define ZCL_EMERGENCY_COMMAND_ID

Definition at line **213** of file **command-id.h**.

#define ZCL_FIRE_COMMAND_ID

Definition at line **214** of file **command-id.h**.

#define ZCL_PANIC_COMMAND_ID

Definition at line **215** of file **command-id.h**.

#define ZCL_GET_ZONE_ID_MAP_COMMAND_ID

Definition at line **216** of file **command-id.h**.

#define ZCL_GET_ZONE_INFORMATION_COMMAND_ID

Definition at line **217** of file **command-id.h**.

#define ZCL_START_WARNING_COMMAND_ID

Definition at line **224** of file **command-id.h**.

#define ZCL_SQUAWK_COMMAND_ID

Definition at line **225** of file **command-id.h**.

#define ZCL_LOCK_DOOR_RESPONSE_COMMAND_ID

Definition at line **232** of file **command-id.h**.

#define ZCL_UNLOCK_DOOR_RESPONSE_COMMAND_ID

Definition at line **233** of file **command-id.h**.

#define ZCL_LOCK_DOOR_COMMAND_ID

Definition at line **236** of file **command-id.h**.

#define ZCL_UNLOCK_DOOR_COMMAND_ID

Definition at line **237** of file **command-id.h**.

#define ZCL_WINDOW_COVERING_UP_OPEN_COMMAND_ID

Definition at line **244** of file **command-id.h**.

#define ZCL_WINDOW_COVERING_DOWN_CLOSE_COMMAND_ID

Definition at line **245** of file **command-id.h**.

#define ZCL_WINDOW_COVERING_STOP_COMMAND_ID

Definition at line **246** of file **command-id.h**.

#define ZCL_WINDOW_COVERING_GO_TO_LIFT_SETPOINT_COMMAND_ID

Definition at line **247** of file **command-id.h**.

#define ZCL_WINDOW_COVERING_GO_TO_LIFT_VALUE_COMMAND_ID

Definition at line **248** of file **command-id.h**.

#define ZCL_WINDOW_COVERING_GO_TO_LIFT_PERCENTAGE_COMMAND_ID

Definition at line **249** of file **command-id.h**.

#define ZCL_WINDOW_COVERING_GO_TO_TILT_SETPOINT_COMMAND_ID

Definition at line **250** of file **command-id.h**.

#define ZCL_WINDOW_COVERING_GO_TO_TILT_VALUE_COMMAND_ID

Definition at line **251** of file **command-id.h**.

#define ZCL_WINDOW_COVERING_GO_TO_TILT_PERCENTAGE_COMMAND_ID

Definition at line **252** of file **command-id.h**.

#define ZCL_WINDOW_COVERING_GO_TO_PROGRAM_SETPOINT_COMMAND_ID

Definition at line **253** of file **command-id.h**.

#define ZCL_MATCH_PROTOCOL_ADDRESS_RESPONSE_COMMAND_ID

Definition at line **260** of file **command-id.h**.

#define ZCL_ADVERTISE_PROTOCOL_ADDRESS_COMMAND_ID

Definition at line **261** of file **command-id.h**.

#define ZCL_MATCH_PROTOCOL_ADDRESS_COMMAND_ID

Definition at line **264** of file **command-id.h**.

#define ZCL_TRANSFER_NPDU_COMMAND_ID

Definition at line **271** of file **command-id.h**.

#define ZCL_IMAGE_NOTIFY_COMMAND_ID

Definition at line **278** of file **command-id.h**.

#define ZCL_QUERY_NEXT_IMAGE_RESPONSE_COMMAND_ID

Definition at line **279** of file **command-id.h**.

#define ZCL_IMAGE_BLOCK_RESPONSE_COMMAND_ID

Definition at line **280** of file **command-id.h**.

#define ZCL_UPGRADE_END_RESPONSE_COMMAND_ID

Definition at line **281** of file **command-id.h**.

#define ZCL_QUERY_SPECIFIC_FILE_RESPONSE_COMMAND_ID

Definition at line **282** of file **command-id.h**.

#define ZCL_QUERY_NEXT_IMAGE_REQUEST_COMMAND_ID

Definition at line **285** of file **command-id.h**.

#define ZCL_IMAGE_BLOCK_REQUEST_COMMAND_ID

Definition at line **286** of file **command-id.h**.

#define ZCL_IMAGE_PAGE_REQUEST_COMMAND_ID

Definition at line **287** of file **command-id.h**.

#define ZCL_UPGRADE_END_REQUEST_COMMAND_ID

Definition at line **288** of file **command-id.h**.

#define ZCL_QUERY_SPECIFIC_FILE_REQUEST_COMMAND_ID

Definition at line **289** of file **command-id.h**.

#define ZCL_LOAD_CONTROL_EVENT_COMMAND_ID

Definition at line **296** of file **command-id.h**.

#define ZCL_CANCEL_LOAD_CONTROL_EVENT_COMMAND_ID

Definition at line **297** of file **command-id.h**.

#define ZCL_CANCEL_ALL_LOAD_CONTROL_EVENTS_COMMAND_ID

Definition at line **298** of file **command-id.h**.

#define ZCL_REPORT_EVENT_STATUS_COMMAND_ID

Definition at line **301** of file **command-id.h**.

#define ZCL_GET_SCHEDULED_EVENTS_COMMAND_ID

Definition at line **302** of file **command-id.h**.

#define ZCL_GET_PROFILE_RESPONSE_COMMAND_ID

Definition at line **309** of file **command-id.h**.

#define ZCL_REQUEST_MIRROR_COMMAND_ID

Definition at line **310** of file **command-id.h**.

#define ZCL_REMOVE_MIRROR_COMMAND_ID

Definition at line **311** of file **command-id.h**.

#define ZCL_REQUEST_FAST_POLL_MODE_RESPONSE_COMMAND_ID

Definition at line **312** of file **command-id.h**.

#define ZCL_GET_PROFILE_COMMAND_ID

Definition at line **315** of file **command-id.h**.

#define ZCL_REQUEST_MIRROR_RESPONSE_COMMAND_ID

Definition at line **316** of file **command-id.h**.

#define ZCL_MIRROR_REMOVED_COMMAND_ID

Definition at line **317** of file **command-id.h**.

#define ZCL_REQUEST_FAST_POLL_MODE_COMMAND_ID

Definition at line **318** of file **command-id.h**.

#define ZCL_PUBLISH_PRICE_COMMAND_ID

Definition at line **325** of file **command-id.h**.

#define ZCL_PUBLISH_BLOCK_PERIOD_COMMAND_ID

Definition at line **326** of file **command-id.h**.

#define ZCL_GET_CURRENT_PRICE_COMMAND_ID

Definition at line **329** of file **command-id.h**.

#define ZCL_GET_SCHEDULED_PRICES_COMMAND_ID

Definition at line **330** of file **command-id.h**.

#define ZCL_PRICE_ACKNOWLEDGEMENT_COMMAND_ID

Definition at line **331** of file **command-id.h**.

#define ZCL_GET_BLOCK_PERIODS_COMMAND_ID

Definition at line **332** of file **command-id.h**.

#define ZCL_DISPLAY_MESSAGE_COMMAND_ID

Definition at line **339** of file **command-id.h**.

#define ZCL_CANCEL_MESSAGE_COMMAND_ID

Definition at line **340** of file **command-id.h**.

#define ZCL_GET_LAST_MESSAGE_COMMAND_ID

Definition at line **343** of file **command-id.h**.

#define ZCL_MESSAGE_CONFIRMATION_COMMAND_ID

Definition at line **344** of file **command-id.h**.

#define ZCL_REQUEST_TUNNEL_RESPONSE_COMMAND_ID

Definition at line **351** of file **command-id.h**.

#define ZCL_TRANSFER_DATA_SERVER_TO_CLIENT_COMMAND_ID

Definition at line **352** of file **command-id.h**.

#define ZCL_TRANSFER_DATA_ERROR_SERVER_TO_CLIENT_COMMAND_ID

Definition at line **353** of file **command-id.h**.

#define ZCL_ACK_TRANSFER_DATA_SERVER_TO_CLIENT_COMMAND_ID

Definition at line **354** of file **command-id.h**.

#define ZCL_READY_DATA_SERVER_TO_CLIENT_COMMAND_ID

Definition at line **355** of file **command-id.h**.

#define ZCL_REQUEST_TUNNEL_COMMAND_ID

Definition at line **358** of file **command-id.h**.

#define ZCL_CLOSE_TUNNEL_COMMAND_ID

Definition at line **359** of file **command-id.h**.

#define ZCL_TRANSFER_DATA_CLIENT_TO_SERVER_COMMAND_ID

Definition at line **360** of file **command-id.h**.

#define ZCL_TRANSFER_DATA_ERROR_CLIENT_TO_SERVER_COMMAND_ID

Definition at line **361** of file **command-id.h**.

#define ZCL_ACK_TRANSFER_DATA_CLIENT_TO_SERVER_COMMAND_ID

Definition at line **362** of file **command-id.h**.

#define ZCL_READY_DATA_CLIENT_TO_SERVER_COMMAND_ID

Definition at line **363** of file **command-id.h**.

#define ZCL_SUPPLY_STATUS_RESPONSE_COMMAND_ID

Definition at line **370** of file **command-id.h**.

#define ZCL_SELECT_AVAILABLE_EMERGENCY_CREDIT_COMMAND_ID

Definition at line **373** of file **command-id.h**.

#define ZCL_CHANGE_SUPPLY_COMMAND_ID

Definition at line **374** of file **command-id.h**.

#define ZCL_INITIATE_KEY_ESTABLISHMENT_RESPONSE_COMMAND_ID

Definition at line **381** of file **command-id.h**.

#define ZCL_EPHEMERAL_DATA_RESPONSE_COMMAND_ID

Definition at line **382** of file **command-id.h**.

#define ZCL_CONFIRM_KEY_DATA_RESPONSE_COMMAND_ID

Definition at line **383** of file **command-id.h**.

#define ZCL_INITIATE_KEY_ESTABLISHMENT_REQUEST_COMMAND_ID

Definition at line **386** of file **command-id.h**.

#define ZCL_EPHEMERAL_DATA_REQUEST_COMMAND_ID

Definition at line **387** of file **command-id.h**.

#define ZCL_CONFIRM_KEY_DATA_REQUEST_COMMAND_ID

Definition at line **388** of file **command-id.h**.

#define ZCL_TERMINATE_KEY_ESTABLISHMENT_COMMAND_ID

Definition at line **391** of file **command-id.h**.

#define ZCL_COMMAND_ONE_COMMAND_ID

Definition at line **398** of file **command-id.h**.

---

core-cli.h File Reference

Go to the source code of this file.

Defines

#define **EMBER_AF_CORE_COMMANDS**
#define **EMBER_AF_INFO_COMMAND**

Functions

void **emAfCliCountersCommand** (void)
void **emAfCliHelpCommand** (void)
void **emAfCliInfoCommand** (void)
void **emAfCliResetCommand** (void)
void **emAfCliEchoCommand** (void)

Variables

EmberCommandEntry **debugPrintCommands** []

Define Documentation

#define EMBER_AF_CORE_COMMANDS

Definition at line **16** of file **core-cli.h**.

#define EMBER_AF_INFO_COMMAND

Definition at line **22** of file **core-cli.h**.

Function Documentation

void emAfCliCountersCommand (void)

void emAfCliHelpCommand (void)

void emAfCliInfoCommand (void)

void emAfCliResetCommand (void)

void emAfCliEchoCommand (void)

Variable Documentation

EmberCommandEntry debugPrintCommands[]

---

custom-cli.h File Reference

Go to the source code of this file.

Defines

```
#define CUSTOM_COMMANDS
```

Define Documentation

#define CUSTOM_COMMANDS

Definition at line 7 of file custom-cli.h.

debug-printing-test.h File Reference

Go to the source code of this file.

Defines

```
#define EMBER_AF_PRINT_ENABLE
#define EMBER_AF_PRINT_BASIC_CLUSTER
#define EMBER_AF_PRINT_POWER_CONFIG_CLUSTER
#define EMBER_AF_PRINT_DEVICE_TEMP_CLUSTER
#define EMBER_AF_PRINT_IDENTIFY_CLUSTER
#define EMBER_AF_PRINT_GROUPS_CLUSTER
#define EMBER_AF_PRINT_SCENES_CLUSTER
#define EMBER_AF_PRINT_ON_OFF_CLUSTER
#define EMBER_AF_PRINT_ON_OFF_SWITCH_CONFIG_CLUSTER
#define EMBER_AF_PRINT_LEVEL_CONTROL_CLUSTER
#define EMBER_AF_PRINT_ALARM_CLUSTER
#define EMBER_AF_PRINT_TIME_CLUSTER
#define EMBER_AF_PRINT_RSSI_LOCATION_CLUSTER
#define EMBER_AF_PRINT_BINARY_INPUT_BASIC_CLUSTER
#define EMBER_AF_PRINT_COMMISSIONING_CLUSTER
#define EMBER_AF_PRINT_SHADE_CONFIG_CLUSTER
#define EMBER_AF_PRINT_PUMP_CONFIG_CONTROL_CLUSTER
#define EMBER_AF_PRINT_THERMOSTAT_CLUSTER
#define EMBER_AF_PRINT_FAN_CONTROL_CLUSTER
#define EMBER_AF_PRINT_DEHUMID_CONTROL_CLUSTER
#define EMBER_AF_PRINT_THERMOSTAT_UI_CONFIG_CLUSTER
#define EMBER_AF_PRINT_COLOR_CONTROL_CLUSTER
#define EMBER_AF_PRINT_BALLAST_CONFIGURATION_CLUSTER
#define EMBER_AF_PRINT_ILLUM_MEASUREMENT_CLUSTER
#define EMBER_AF_PRINT_ILLUM_LEVEL_SENSING_CLUSTER
#define EMBER_AF_PRINT_TEMP_MEASUREMENT_CLUSTER
#define EMBER_AF_PRINT_PRESSURE_MEASUREMENT_CLUSTER
#define EMBER_AF_PRINT_FLOW_MEASUREMENT_CLUSTER
#define EMBER_AF_PRINT_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER
#define EMBER_AF_PRINT_OCCUPANCY_SENSING_CLUSTER
#define EMBER_AF_PRINT_IAS_ZONE_CLUSTER
#define EMBER_AF_PRINT_IAS_ACE_CLUSTER
```

#define **EMBER_AF_PRINT_IAS_WD_CLUSTER**
#define **EMBER_AF_PRINT_DOOR_LOCK_CLUSTER**
#define **EMBER_AF_PRINT_WINDOW_COVERING_CLUSTER**
#define **EMBER_AF_PRINT_GENERIC_TUNNEL_CLUSTER**
#define **EMBER_AF_PRINT_BACNET_PROTOCOL_TUNNEL_CLUSTER**
#define **EMBER_AF_PRINT_OTA_BOOTLOAD_CLUSTER**
#define **EMBER_AF_PRINT_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER**
#define **EMBER_AF_PRINT_SIMPLE_METERING_CLUSTER**
#define **EMBER_AF_PRINT_PRICE_CLUSTER**
#define **EMBER_AF_PRINT_MESSAGING_CLUSTER**
#define **EMBER_AF_PRINT_TUNNELING_CLUSTER**
#define **EMBER_AF_PRINT_PREPAYMENT_CLUSTER**
#define **EMBER_AF_PRINT_KEY_ESTABLISHMENT_CLUSTER**
#define **EMBER_AF_PRINT_SAMPLE_MFG_SPECIFIC_CLUSTER**
#define **EMBER_AF_PRINT_CORE**
#define **EMBER_AF_PRINT_DEBUG**
#define **EMBER_AF_PRINT_APP**
#define **EMBER_AF_PRINT_SECURITY**
#define **EMBER_AF_PRINT_ATTRIBUTES**
#define **EMBER_AF_PRINT_REPORTING**
#define **EMBER_AF_PRINT_SERVICE_DISCOVERY**
#define **EMBER_AF_PRINT_REGISTRATION**
#define **EMBER_AF_PRINT_ZDO**
#define **EMBER_AF_PRINT_CUSTOM1**
#define **EMBER_AF_PRINT_CUSTOM2**
#define **EMBER_AF_PRINT_CUSTOM3**
#define **EMBER_AF_PRINT_OUTPUT**
#define **EMBER_AF_PRINT_NAMES**
#define **EMBER_AF_PRINT_NAME_NUMBER**
#define **EMBER_AF_PRINT_BITS**

Define Documentation

#define EMBER_AF_PRINT_ENABLE

Definition at line **15** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_BASIC_CLUSTER

Definition at line **16** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_POWER_CONFIG_CLUSTER

Definition at line **17** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_DEVICE_TEMP_CLUSTER

Definition at line **18** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_IDENTIFY_CLUSTER

Definition at line **19** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_GROUPS_CLUSTER

Definition at line **20** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_SCENES_CLUSTER

Definition at line **21** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_ON_OFF_CLUSTER

Definition at line **22** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_ON_OFF_SWITCH_CONFIG_CLUSTER

Definition at line **23** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_LEVEL_CONTROL_CLUSTER

Definition at line **24** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_ALARM_CLUSTER

Definition at line **25** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_TIME_CLUSTER

Definition at line **26** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_RSSI_LOCATION_CLUSTER

Definition at line **27** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_BINARY_INPUT_BASIC_CLUSTER

Definition at line **28** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_COMMISSIONING_CLUSTER

Definition at line **29** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_SHADE_CONFIG_CLUSTER

Definition at line **30** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_PUMP_CONFIG_CONTROL_CLUSTER

Definition at line **31** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_THERMOSTAT_CLUSTER

Definition at line **32** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_FAN_CONTROL_CLUSTER

Definition at line **33** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_DEHUMID_CONTROL_CLUSTER

Definition at line **34** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_THERMOSTAT_UI_CONFIG_CLUSTER

Definition at line **35** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_COLOR_CONTROL_CLUSTER

Definition at line **36** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_BALLAST_CONFIGURATION_CLUSTER

Definition at line **37** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_ILLUM_MEASUREMENT_CLUSTER

Definition at line **38** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_ILLUM_LEVEL_SENSING_CLUSTER

Definition at line **39** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_TEMP_MEASUREMENT_CLUSTER

Definition at line **40** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_PRESSURE_MEASUREMENT_CLUSTER

Definition at line **41** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_FLOW_MEASUREMENT_CLUSTER

Definition at line **42** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER

Definition at line **43** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_OCCUPANCY_SENSING_CLUSTER

Definition at line **44** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_IAS_ZONE_CLUSTER

Definition at line **45** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_IAS_ACE_CLUSTER

Definition at line **46** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_IAS_WD_CLUSTER

Definition at line **47** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_DOOR_LOCK_CLUSTER

Definition at line **48** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_WINDOW_COVERING_CLUSTER

Definition at line **49** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_GENERIC_TUNNEL_CLUSTER

Definition at line **50** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_BACNET_PROTOCOL_TUNNEL_CLUSTER

Definition at line **51** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_OTA_BOOTLOAD_CLUSTER

Definition at line **52** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER

Definition at line **53** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_SIMPLE_METERING_CLUSTER

Definition at line **54** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_PRICE_CLUSTER

Definition at line **55** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_MESSAGING_CLUSTER

Definition at line **56** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_TUNNELING_CLUSTER

Definition at line **57** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_PREPAYMENT_CLUSTER

Definition at line **58** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_KEY_ESTABLISHMENT_CLUSTER

Definition at line **59** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_SAMPLE_MFG_SPECIFIC_CLUSTER

Definition at line **60** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_CORE

Definition at line **61** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_DEBUG

Definition at line **62** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_APP

Definition at line **63** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_SECURITY

Definition at line **64** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_ATTRIBUTES

Definition at line **65** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_REPORTING

Definition at line **66** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_SERVICE_DISCOVERY

Definition at line **67** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_REGISTRATION

Definition at line **68** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_ZDO

Definition at line **69** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_CUSTOM1

Definition at line **70** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_CUSTOM2

Definition at line **71** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_CUSTOM3

Definition at line **72** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_OUTPUT

Definition at line **75** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_NAMES

Definition at line **78** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_NAME_NUMBER

Definition at line **139** of file **debug-printing-test.h**.

#define EMBER_AF_PRINT_BITS

Definition at line **140** of file **debug-printing-test.h**.

---

debug-printing.h File Reference

Go to the source code of this file.

Defines

#define **emberAfBasicClusterPrint**(...)
#define **emberAfBasicClusterPrintln**(...)
#define **emberAfBasicClusterFlush**()
#define **emberAfBasicClusterDebugExec**(x)
#define **emberAfBasicClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfBasicClusterPrintString**(buffer)
#define **emberAfPowerConfigClusterPrint**(...)
#define **emberAfPowerConfigClusterPrintln**(...)
#define **emberAfPowerConfigClusterFlush**()
#define **emberAfPowerConfigClusterDebugExec**(x)
#define **emberAfPowerConfigClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfPowerConfigClusterPrintString**(buffer)
#define **emberAfDeviceTempClusterPrint**(...)
#define **emberAfDeviceTempClusterPrintln**(...)
#define **emberAfDeviceTempClusterFlush**()
#define **emberAfDeviceTempClusterDebugExec**(x)
#define **emberAfDeviceTempClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfDeviceTempClusterPrintString**(buffer)
#define **emberAfIdentifyClusterPrint**(...)
#define **emberAfIdentifyClusterPrintln**(...)
#define **emberAfIdentifyClusterFlush**()
#define **emberAfIdentifyClusterDebugExec**(x)
#define **emberAfIdentifyClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfIdentifyClusterPrintString**(buffer)
#define **emberAfGroupsClusterPrint**(...)
#define **emberAfGroupsClusterPrintln**(...)
#define **emberAfGroupsClusterFlush**()
#define **emberAfGroupsClusterDebugExec**(x)
#define **emberAfGroupsClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfGroupsClusterPrintString**(buffer)
#define **emberAfScenesClusterPrint**(...)
#define **emberAfScenesClusterPrintln**(...)
#define **emberAfScenesClusterFlush**()
#define **emberAfScenesClusterDebugExec**(x)

#define **emberAfScenesClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfScenesClusterPrintString**(buffer)
#define **emberAfOnOffClusterPrint**(...)
#define **emberAfOnOffClusterPrintln**(...)
#define **emberAfOnOffClusterFlush**()
#define **emberAfOnOffClusterDebugExec**(x)
#define **emberAfOnOffClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfOnOffClusterPrintString**(buffer)
#define **emberAfOnOffSwitchConfigClusterPrint**(...)
#define **emberAfOnOffSwitchConfigClusterPrintln**(...)
#define **emberAfOnOffSwitchConfigClusterFlush**()
#define **emberAfOnOffSwitchConfigClusterDebugExec**(x)
#define **emberAfOnOffSwitchConfigClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfOnOffSwitchConfigClusterPrintString**(buffer)
#define **emberAfLevelControlClusterPrint**(...)
#define **emberAfLevelControlClusterPrintln**(...)
#define **emberAfLevelControlClusterFlush**()
#define **emberAfLevelControlClusterDebugExec**(x)
#define **emberAfLevelControlClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfLevelControlClusterPrintString**(buffer)
#define **emberAfAlarmClusterPrint**(...)
#define **emberAfAlarmClusterPrintln**(...)
#define **emberAfAlarmClusterFlush**()
#define **emberAfAlarmClusterDebugExec**(x)
#define **emberAfAlarmClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfAlarmClusterPrintString**(buffer)
#define **emberAfTimeClusterPrint**(...)
#define **emberAfTimeClusterPrintln**(...)
#define **emberAfTimeClusterFlush**()
#define **emberAfTimeClusterDebugExec**(x)
#define **emberAfTimeClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfTimeClusterPrintString**(buffer)
#define **emberAfRssiLocationClusterPrint**(...)
#define **emberAfRssiLocationClusterPrintln**(...)
#define **emberAfRssiLocationClusterFlush**()
#define **emberAfRssiLocationClusterDebugExec**(x)
#define **emberAfRssiLocationClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfRssiLocationClusterPrintString**(buffer)
#define **emberAfBinaryInputBasicClusterPrint**(...)
#define **emberAfBinaryInputBasicClusterPrintln**(...)
#define **emberAfBinaryInputBasicClusterFlush**()
#define **emberAfBinaryInputBasicClusterDebugExec**(x)
#define **emberAfBinaryInputBasicClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfBinaryInputBasicClusterPrintString**(buffer)

```
#define emberAfCommissioningClusterPrint(...)
#define emberAfCommissioningClusterPrintln(...)
#define emberAfCommissioningClusterFlush()
#define emberAfCommissioningClusterDebugExec(x)
#define emberAfCommissioningClusterPrintBuffer(buffer, len, withSpace)
#define emberAfCommissioningClusterPrintString(buffer)
#define emberAfShadeConfigClusterPrint(...)
#define emberAfShadeConfigClusterPrintln(...)
#define emberAfShadeConfigClusterFlush()
#define emberAfShadeConfigClusterDebugExec(x)
#define emberAfShadeConfigClusterPrintBuffer(buffer, len, withSpace)
#define emberAfShadeConfigClusterPrintString(buffer)
#define emberAfPumpConfigControlClusterPrint(...)
#define emberAfPumpConfigControlClusterPrintln(...)
#define emberAfPumpConfigControlClusterFlush()
#define emberAfPumpConfigControlClusterDebugExec(x)
#define emberAfPumpConfigControlClusterPrintBuffer(buffer, len, withSpace)
#define emberAfPumpConfigControlClusterPrintString(buffer)
#define emberAfThermostatClusterPrint(...)
#define emberAfThermostatClusterPrintln(...)
#define emberAfThermostatClusterFlush()
#define emberAfThermostatClusterDebugExec(x)
#define emberAfThermostatClusterPrintBuffer(buffer, len, withSpace)
#define emberAfThermostatClusterPrintString(buffer)
#define emberAfFanControlClusterPrint(...)
#define emberAfFanControlClusterPrintln(...)
#define emberAfFanControlClusterFlush()
#define emberAfFanControlClusterDebugExec(x)
#define emberAfFanControlClusterPrintBuffer(buffer, len, withSpace)
#define emberAfFanControlClusterPrintString(buffer)
#define emberAfDehumidControlClusterPrint(...)
#define emberAfDehumidControlClusterPrintln(...)
#define emberAfDehumidControlClusterFlush()
#define emberAfDehumidControlClusterDebugExec(x)
#define emberAfDehumidControlClusterPrintBuffer(buffer, len, withSpace)
#define emberAfDehumidControlClusterPrintString(buffer)
#define emberAfThermostatUiConfigClusterPrint(...)
#define emberAfThermostatUiConfigClusterPrintln(...)
#define emberAfThermostatUiConfigClusterFlush()
#define emberAfThermostatUiConfigClusterDebugExec(x)
#define emberAfThermostatUiConfigClusterPrintBuffer(buffer, len, withSpace)
#define emberAfThermostatUiConfigClusterPrintString(buffer)
#define emberAfColorControlClusterPrint(...)
#define emberAfColorControlClusterPrintln(...)
```

```
#define emberAfColorControlClusterFlush()
#define emberAfColorControlClusterDebugExec(x)
#define emberAfColorControlClusterPrintBuffer(buffer, len, withSpace)
#define emberAfColorControlClusterPrintString(buffer)
#define emberAfBallastConfigurationClusterPrint(...)
#define emberAfBallastConfigurationClusterPrintln(...)
#define emberAfBallastConfigurationClusterFlush()
#define emberAfBallastConfigurationClusterDebugExec(x)
#define emberAfBallastConfigurationClusterPrintBuffer(buffer, len, withSpace)
#define emberAfBallastConfigurationClusterPrintString(buffer)
#define emberAfIllumMeasurementClusterPrint(...)
#define emberAfIllumMeasurementClusterPrintln(...)
#define emberAfIllumMeasurementClusterFlush()
#define emberAfIllumMeasurementClusterDebugExec(x)
#define emberAfIllumMeasurementClusterPrintBuffer(buffer, len, withSpace)
#define emberAfIllumMeasurementClusterPrintString(buffer)
#define emberAfIllumLevelSensingClusterPrint(...)
#define emberAfIllumLevelSensingClusterPrintln(...)
#define emberAfIllumLevelSensingClusterFlush()
#define emberAfIllumLevelSensingClusterDebugExec(x)
#define emberAfIllumLevelSensingClusterPrintBuffer(buffer, len, withSpace)
#define emberAfIllumLevelSensingClusterPrintString(buffer)
#define emberAfTempMeasurementClusterPrint(...)
#define emberAfTempMeasurementClusterPrintln(...)
#define emberAfTempMeasurementClusterFlush()
#define emberAfTempMeasurementClusterDebugExec(x)
#define emberAfTempMeasurementClusterPrintBuffer(buffer, len, withSpace)
#define emberAfTempMeasurementClusterPrintString(buffer)
#define emberAfPressureMeasurementClusterPrint(...)
#define emberAfPressureMeasurementClusterPrintln(...)
#define emberAfPressureMeasurementClusterFlush()
#define emberAfPressureMeasurementClusterDebugExec(x)
#define emberAfPressureMeasurementClusterPrintBuffer(buffer, len, withSpace)
#define emberAfPressureMeasurementClusterPrintString(buffer)
#define emberAfFlowMeasurementClusterPrint(...)
#define emberAfFlowMeasurementClusterPrintln(...)
#define emberAfFlowMeasurementClusterFlush()
#define emberAfFlowMeasurementClusterDebugExec(x)
#define emberAfFlowMeasurementClusterPrintBuffer(buffer, len, withSpace)
#define emberAfFlowMeasurementClusterPrintString(buffer)
#define emberAfRelativeHumidityMeasurementClusterPrint(...)
#define emberAfRelativeHumidityMeasurementClusterPrintln(...)
#define emberAfRelativeHumidityMeasurementClusterFlush()
#define emberAfRelativeHumidityMeasurementClusterDebugExec(x)
```

#define **emberAfRelativeHumidityMeasurementClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfRelativeHumidityMeasurementClusterPrintString**(buffer)
#define **emberAfOccupancySensingClusterPrint**(...)
#define **emberAfOccupancySensingClusterPrintln**(...)
#define **emberAfOccupancySensingClusterFlush**()
#define **emberAfOccupancySensingClusterDebugExec**(x)
#define **emberAfOccupancySensingClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfOccupancySensingClusterPrintString**(buffer)
#define **emberAfIasZoneClusterPrint**(...)
#define **emberAfIasZoneClusterPrintln**(...)
#define **emberAfIasZoneClusterFlush**()
#define **emberAfIasZoneClusterDebugExec**(x)
#define **emberAfIasZoneClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfIasZoneClusterPrintString**(buffer)
#define **emberAfIasAceClusterPrint**(...)
#define **emberAfIasAceClusterPrintln**(...)
#define **emberAfIasAceClusterFlush**()
#define **emberAfIasAceClusterDebugExec**(x)
#define **emberAfIasAceClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfIasAceClusterPrintString**(buffer)
#define **emberAfIasWdClusterPrint**(...)
#define **emberAfIasWdClusterPrintln**(...)
#define **emberAfIasWdClusterFlush**()
#define **emberAfIasWdClusterDebugExec**(x)
#define **emberAfIasWdClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfIasWdClusterPrintString**(buffer)
#define **emberAfDoorLockClusterPrint**(...)
#define **emberAfDoorLockClusterPrintln**(...)
#define **emberAfDoorLockClusterFlush**()
#define **emberAfDoorLockClusterDebugExec**(x)
#define **emberAfDoorLockClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfDoorLockClusterPrintString**(buffer)
#define **emberAfWindowCoveringClusterPrint**(...)
#define **emberAfWindowCoveringClusterPrintln**(...)
#define **emberAfWindowCoveringClusterFlush**()
#define **emberAfWindowCoveringClusterDebugExec**(x)
#define **emberAfWindowCoveringClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfWindowCoveringClusterPrintString**(buffer)
#define **emberAfGenericTunnelClusterPrint**(...)
#define **emberAfGenericTunnelClusterPrintln**(...)
#define **emberAfGenericTunnelClusterFlush**()
#define **emberAfGenericTunnelClusterDebugExec**(x)
#define **emberAfGenericTunnelClusterPrintBuffer**(buffer, len, withSpace)
#define **emberAfGenericTunnelClusterPrintString**(buffer)

```
#define emberAfBacnetProtocolTunnelClusterPrint(...)
#define emberAfBacnetProtocolTunnelClusterPrintln(...)
#define emberAfBacnetProtocolTunnelClusterFlush()
#define emberAfBacnetProtocolTunnelClusterDebugExec(x)
#define emberAfBacnetProtocolTunnelClusterPrintBuffer(buffer, len, withSpace)
#define emberAfBacnetProtocolTunnelClusterPrintString(buffer)
#define emberAfOtaBootloadClusterPrint(...)
#define emberAfOtaBootloadClusterPrintln(...)
#define emberAfOtaBootloadClusterFlush()
#define emberAfOtaBootloadClusterDebugExec(x)
#define emberAfOtaBootloadClusterPrintBuffer(buffer, len, withSpace)
#define emberAfOtaBootloadClusterPrintString(buffer)
#define emberAfDemandResponseLoadControlClusterPrint(...)
#define emberAfDemandResponseLoadControlClusterPrintln(...)
#define emberAfDemandResponseLoadControlClusterFlush()
#define emberAfDemandResponseLoadControlClusterDebugExec(x)
#define emberAfDemandResponseLoadControlClusterPrintBuffer(buffer, len, withSpace)
#define emberAfDemandResponseLoadControlClusterPrintString(buffer)
#define emberAfSimpleMeteringClusterPrint(...)
#define emberAfSimpleMeteringClusterPrintln(...)
#define emberAfSimpleMeteringClusterFlush()
#define emberAfSimpleMeteringClusterDebugExec(x)
#define emberAfSimpleMeteringClusterPrintBuffer(buffer, len, withSpace)
#define emberAfSimpleMeteringClusterPrintString(buffer)
#define emberAfPriceClusterPrint(...)
#define emberAfPriceClusterPrintln(...)
#define emberAfPriceClusterFlush()
#define emberAfPriceClusterDebugExec(x)
#define emberAfPriceClusterPrintBuffer(buffer, len, withSpace)
#define emberAfPriceClusterPrintString(buffer)
#define emberAfMessagingClusterPrint(...)
#define emberAfMessagingClusterPrintln(...)
#define emberAfMessagingClusterFlush()
#define emberAfMessagingClusterDebugExec(x)
#define emberAfMessagingClusterPrintBuffer(buffer, len, withSpace)
#define emberAfMessagingClusterPrintString(buffer)
#define emberAfTunnelingClusterPrint(...)
#define emberAfTunnelingClusterPrintln(...)
#define emberAfTunnelingClusterFlush()
#define emberAfTunnelingClusterDebugExec(x)
#define emberAfTunnelingClusterPrintBuffer(buffer, len, withSpace)
#define emberAfTunnelingClusterPrintString(buffer)
#define emberAfPrepaymentClusterPrint(...)
#define emberAfPrepaymentClusterPrintln(...)
```

```
#define emberAfPrepaymentClusterFlush()
#define emberAfPrepaymentClusterDebugExec(x)
#define emberAfPrepaymentClusterPrintBuffer(buffer, len, withSpace)
#define emberAfPrepaymentClusterPrintString(buffer)
#define emberAfKeyEstablishmentClusterPrint(...)
#define emberAfKeyEstablishmentClusterPrintln(...)
#define emberAfKeyEstablishmentClusterFlush()
#define emberAfKeyEstablishmentClusterDebugExec(x)
#define emberAfKeyEstablishmentClusterPrintBuffer(buffer, len, withSpace)
#define emberAfKeyEstablishmentClusterPrintString(buffer)
#define emberAfSampleMfgSpecificClusterPrint(...)
#define emberAfSampleMfgSpecificClusterPrintln(...)
#define emberAfSampleMfgSpecificClusterFlush()
#define emberAfSampleMfgSpecificClusterDebugExec(x)
#define emberAfSampleMfgSpecificClusterPrintBuffer(buffer, len, withSpace)
#define emberAfSampleMfgSpecificClusterPrintString(buffer)
#define emberAfCorePrint(...)
#define emberAfCorePrintln(...)
#define emberAfCoreFlush()
#define emberAfCoreDebugExec(x)
#define emberAfCorePrintBuffer(buffer, len, withSpace)
#define emberAfCorePrintString(buffer)
#define emberAfDebugPrint(...)
#define emberAfDebugPrintln(...)
#define emberAfDebugFlush()
#define emberAfDebugDebugExec(x)
#define emberAfDebugPrintBuffer(buffer, len, withSpace)
#define emberAfDebugPrintString(buffer)
#define emberAfAppPrint(...)
#define emberAfAppPrintln(...)
#define emberAfAppFlush()
#define emberAfAppDebugExec(x)
#define emberAfAppPrintBuffer(buffer, len, withSpace)
#define emberAfAppPrintString(buffer)
#define emberAfSecurityPrint(...)
#define emberAfSecurityPrintln(...)
#define emberAfSecurityFlush()
#define emberAfSecurityDebugExec(x)
#define emberAfSecurityPrintBuffer(buffer, len, withSpace)
#define emberAfSecurityPrintString(buffer)
#define emberAfAttributesPrint(...)
#define emberAfAttributesPrintln(...)
#define emberAfAttributesFlush()
#define emberAfAttributesDebugExec(x)
```

```
#define emberAfAttributesPrintBuffer(buffer, len, withSpace)
#define emberAfAttributesPrintString(buffer)
#define emberAfReportingPrint(...)
#define emberAfReportingPrintln(...)
#define emberAfReportingFlush()
#define emberAfReportingDebugExec(x)
#define emberAfReportingPrintBuffer(buffer, len, withSpace)
#define emberAfReportingPrintString(buffer)
#define emberAfServiceDiscoveryPrint(...)
#define emberAfServiceDiscoveryPrintln(...)
#define emberAfServiceDiscoveryFlush()
#define emberAfServiceDiscoveryDebugExec(x)
#define emberAfServiceDiscoveryPrintBuffer(buffer, len, withSpace)
#define emberAfServiceDiscoveryPrintString(buffer)
#define emberAfRegistrationPrint(...)
#define emberAfRegistrationPrintln(...)
#define emberAfRegistrationFlush()
#define emberAfRegistrationDebugExec(x)
#define emberAfRegistrationPrintBuffer(buffer, len, withSpace)
#define emberAfRegistrationPrintString(buffer)
#define emberAfZdoPrint(...)
#define emberAfZdoPrintln(...)
#define emberAfZdoFlush()
#define emberAfZdoDebugExec(x)
#define emberAfZdoPrintBuffer(buffer, len, withSpace)
#define emberAfZdoPrintString(buffer)
#define emberAfCustom1Print(...)
#define emberAfCustom1Println(...)
#define emberAfCustom1Flush()
#define emberAfCustom1DebugExec(x)
#define emberAfCustom1PrintBuffer(buffer, len, withSpace)
#define emberAfCustom1PrintString(buffer)
#define emberAfCustom2Print(...)
#define emberAfCustom2Println(...)
#define emberAfCustom2Flush()
#define emberAfCustom2DebugExec(x)
#define emberAfCustom2PrintBuffer(buffer, len, withSpace)
#define emberAfCustom2PrintString(buffer)
#define emberAfCustom3Print(...)
#define emberAfCustom3Println(...)
#define emberAfCustom3Flush()
#define emberAfCustom3DebugExec(x)
#define emberAfCustom3PrintBuffer(buffer, len, withSpace)
#define emberAfCustom3PrintString(buffer)
```

Define Documentation

#define emberAfBasicClusterPrint (...)

Definition at line **18** of file **debug-printing.h**.

#define emberAfBasicClusterPrintln (...)

Definition at line **19** of file **debug-printing.h**.

#define emberAfBasicClusterFlush ( )

Definition at line **20** of file **debug-printing.h**.

#define emberAfBasicClusterDebugExec (x)

Definition at line **21** of file **debug-printing.h**.

#define emberAfBasicClusterPrintBuffer (buffer,
len,
withSpace)

Definition at line **22** of file **debug-printing.h**.

#define emberAfBasicClusterPrintString (buffer)

Definition at line **23** of file **debug-printing.h**.

#define emberAfPowerConfigClusterPrint (...)

Definition at line **36** of file **debug-printing.h**.

#define emberAfPowerConfigClusterPrintln (...)

Definition at line **37** of file **debug-printing.h**.

#define emberAfPowerConfigClusterFlush ( )

Definition at line **38** of file **debug-printing.h**.

#define emberAfPowerConfigClusterDebugExec (x)

Definition at line **39** of file **debug-printing.h**.

#define emberAfPowerConfigClusterPrintBuffer (buffer,
len,
withSpace)

Definition at line **40** of file **debug-printing.h**.

#define emberAfPowerConfigClusterPrintString (buffer)

Definition at line **41** of file **debug-printing.h**.

#define emberAfDeviceTempClusterPrint (...)

Definition at line **54** of file **debug-printing.h**.

#define emberAfDeviceTempClusterPrintln (...)

Definition at line **55** of file **debug-printing.h**.

#define emberAfDeviceTempClusterFlush ( )

Definition at line **56** of file **debug-printing.h**.

#define emberAfDeviceTempClusterDebugExec (x)

Definition at line **57** of file **debug-printing.h**.

#define emberAfDeviceTempClusterPrintBuffer (buffer,
len,

withSpace)

Definition at line **58** of file **debug-printing.h**.

#define emberAfDeviceTempClusterPrintString (buffer)

Definition at line **59** of file **debug-printing.h**.

#define emberAfIdentifyClusterPrint (...)

Definition at line **72** of file **debug-printing.h**.

#define emberAfIdentifyClusterPrintln (...)

Definition at line **73** of file **debug-printing.h**.

#define emberAfIdentifyClusterFlush ( )

Definition at line **74** of file **debug-printing.h**.

#define emberAfIdentifyClusterDebugExec (x)

Definition at line **75** of file **debug-printing.h**.

#define emberAfIdentifyClusterPrintBuffer (buffer,
len,
withSpace)

Definition at line **76** of file **debug-printing.h**.

#define emberAfIdentifyClusterPrintString (buffer)

Definition at line **77** of file **debug-printing.h**.

#define emberAfGroupsClusterPrint (...)

Definition at line **90** of file **debug-printing.h**.

#define emberAfGroupsClusterPrintln (...)

Definition at line **91** of file **debug-printing.h**.

#define emberAfGroupsClusterFlush ( )

Definition at line **92** of file **debug-printing.h**.

#define emberAfGroupsClusterDebugExec (x)

Definition at line **93** of file **debug-printing.h**.

#define emberAfGroupsClusterPrintBuffer (buffer, len, withSpace)

Definition at line **94** of file **debug-printing.h**.

#define emberAfGroupsClusterPrintString (buffer)

Definition at line **95** of file **debug-printing.h**.

#define emberAfScenesClusterPrint (...)

Definition at line **108** of file **debug-printing.h**.

#define emberAfScenesClusterPrintln (...)

Definition at line **109** of file **debug-printing.h**.

#define emberAfScenesClusterFlush ( )

Definition at line **110** of file **debug-printing.h**.

#define emberAfScenesClusterDebugExec (x)

Definition at line **111** of file **debug-printing.h**.

#define emberAfScenesClusterPrintBuffer (buffer, len, withSpace)

Definition at line **112** of file **debug-printing.h**.

#define emberAfScenesClusterPrintString (buffer)

Definition at line **113** of file **debug-printing.h**.

#define emberAfOnOffClusterPrint (...)

Definition at line **126** of file **debug-printing.h**.

#define emberAfOnOffClusterPrintln (...)

Definition at line **127** of file **debug-printing.h**.

#define emberAfOnOffClusterFlush ( )

Definition at line **128** of file **debug-printing.h**.

#define emberAfOnOffClusterDebugExec (x)

Definition at line **129** of file **debug-printing.h**.

#define emberAfOnOffClusterPrintBuffer (buffer, len, withSpace)

Definition at line **130** of file **debug-printing.h**.

#define emberAfOnOffClusterPrintString (buffer)

Definition at line **131** of file **debug-printing.h**.

#define emberAfOnOffSwitchConfigClusterPrint (...)

Definition at line **144** of file **debug-printing.h**.

#define emberAfOnOffSwitchConfigClusterPrintln (...)

Definition at line **145** of file **debug-printing.h**.

#define emberAfOnOffSwitchConfigClusterFlush ( )

Definition at line **146** of file **debug-printing.h**.

#define emberAfOnOffSwitchConfigClusterDebugExec (x)

Definition at line **147** of file **debug-printing.h**.

#define emberAfOnOffSwitchConfigClusterPrintBuffer (buffer, len, withSpace)

Definition at line **148** of file **debug-printing.h**.

#define emberAfOnOffSwitchConfigClusterPrintString (buffer)

Definition at line **149** of file **debug-printing.h**.

#define emberAfLevelControlClusterPrint (...)

Definition at line **162** of file **debug-printing.h**.

#define emberAfLevelControlClusterPrintln (...)

Definition at line **163** of file **debug-printing.h**.

#define emberAfLevelControlClusterFlush ( )

Definition at line **164** of file **debug-printing.h**.

#define emberAfLevelControlClusterDebugExec (x)

Definition at line **165** of file **debug-printing.h**.

#define emberAfLevelControlClusterPrintBuffer (buffer, len, withSpace)

Definition at line **166** of file **debug-printing.h**.

#define emberAfLevelControlClusterPrintString (buffer)

Definition at line **167** of file **debug-printing.h**.

#define emberAfAlarmClusterPrint (...)

Definition at line **180** of file **debug-printing.h**.

#define emberAfAlarmClusterPrintln (...)

Definition at line **181** of file **debug-printing.h**.

#define emberAfAlarmClusterFlush ( )

Definition at line **182** of file **debug-printing.h**.

#define emberAfAlarmClusterDebugExec (x)

Definition at line **183** of file **debug-printing.h**.

#define emberAfAlarmClusterPrintBuffer (buffer, len, withSpace)

Definition at line **184** of file **debug-printing.h**.

#define emberAfAlarmClusterPrintString (buffer)

Definition at line **185** of file **debug-printing.h**.

#define emberAfTimeClusterPrint (...)

Definition at line **198** of file **debug-printing.h**.

#define emberAfTimeClusterPrintln (...)

Definition at line **199** of file **debug-printing.h**.

#define emberAfTimeClusterFlush ( )

Definition at line **200** of file **debug-printing.h**.

#define emberAfTimeClusterDebugExec (x)

Definition at line **201** of file **debug-printing.h**.

#define emberAfTimeClusterPrintBuffer (buffer,
len,
withSpace)

Definition at line **202** of file **debug-printing.h**.

#define emberAfTimeClusterPrintString (buffer)

Definition at line **203** of file **debug-printing.h**.

#define emberAfRssiLocationClusterPrint (...)

Definition at line **216** of file **debug-printing.h**.

#define emberAfRssiLocationClusterPrintln (...)

Definition at line **217** of file **debug-printing.h**.

#define emberAfRssiLocationClusterFlush ( )

Definition at line **218** of file **debug-printing.h**.

#define emberAfRssiLocationClusterDebugExec (x)

Definition at line **219** of file **debug-printing.h**.

#define emberAfRssiLocationClusterPrintBuffer (buffer, len, withSpace)

Definition at line **220** of file **debug-printing.h**.

#define emberAfRssiLocationClusterPrintString (buffer)

Definition at line **221** of file **debug-printing.h**.

#define emberAfBinaryInputBasicClusterPrint (...)

Definition at line **234** of file **debug-printing.h**.

#define emberAfBinaryInputBasicClusterPrintln (...)

Definition at line **235** of file **debug-printing.h**.

#define emberAfBinaryInputBasicClusterFlush ( )

Definition at line **236** of file **debug-printing.h**.

#define emberAfBinaryInputBasicClusterDebugExec (x)

Definition at line **237** of file **debug-printing.h**.

#define emberAfBinaryInputBasicClusterPrintBuffer (buffer,

```
                                                    len,
                                                    withSpace    )
```

Definition at line **238** of file **debug-printing.h**.

```
#define emberAfBinaryInputBasicClusterPrintString ( buffer    )
```

Definition at line **239** of file **debug-printing.h**.

```
#define emberAfCommissioningClusterPrint ( ...    )
```

Definition at line **252** of file **debug-printing.h**.

```
#define emberAfCommissioningClusterPrintln ( ...    )
```

Definition at line **253** of file **debug-printing.h**.

```
#define emberAfCommissioningClusterFlush (    )
```

Definition at line **254** of file **debug-printing.h**.

```
#define emberAfCommissioningClusterDebugExec ( x    )
```

Definition at line **255** of file **debug-printing.h**.

```
#define emberAfCommissioningClusterPrintBuffer ( buffer,
                                                 len,
                                                 withSpace    )
```

Definition at line **256** of file **debug-printing.h**.

```
#define emberAfCommissioningClusterPrintString ( buffer    )
```

Definition at line **257** of file **debug-printing.h**.

```
#define emberAfShadeConfigClusterPrint ( ...    )
```

Definition at line **270** of file **debug-printing.h**.

#define emberAfShadeConfigClusterPrintln (...)

Definition at line **271** of file **debug-printing.h**.

#define emberAfShadeConfigClusterFlush ( )

Definition at line **272** of file **debug-printing.h**.

#define emberAfShadeConfigClusterDebugExec (x)

Definition at line **273** of file **debug-printing.h**.

#define emberAfShadeConfigClusterPrintBuffer (buffer,
len,
withSpace)

Definition at line **274** of file **debug-printing.h**.

#define emberAfShadeConfigClusterPrintString (buffer)

Definition at line **275** of file **debug-printing.h**.

#define emberAfPumpConfigControlClusterPrint (...)

Definition at line **288** of file **debug-printing.h**.

#define emberAfPumpConfigControlClusterPrintln (...)

Definition at line **289** of file **debug-printing.h**.

#define emberAfPumpConfigControlClusterFlush ( )

Definition at line **290** of file **debug-printing.h**.

#define emberAfPumpConfigControlClusterDebugExec (x)

Definition at line **291** of file **debug-printing.h**.

#define emberAfPumpConfigControlClusterPrintBuffer (buffer, len, withSpace)

Definition at line **292** of file **debug-printing.h**.

#define emberAfPumpConfigControlClusterPrintString (buffer)

Definition at line **293** of file **debug-printing.h**.

#define emberAfThermostatClusterPrint (...)

Definition at line **306** of file **debug-printing.h**.

#define emberAfThermostatClusterPrintln (...)

Definition at line **307** of file **debug-printing.h**.

#define emberAfThermostatClusterFlush ( )

Definition at line **308** of file **debug-printing.h**.

#define emberAfThermostatClusterDebugExec (x)

Definition at line **309** of file **debug-printing.h**.

#define emberAfThermostatClusterPrintBuffer (buffer, len, withSpace)

Definition at line **310** of file **debug-printing.h**.

#define emberAfThermostatClusterPrintString (buffer)

Definition at line **311** of file **debug-printing.h**.

#define emberAfFanControlClusterPrint (...)

Definition at line **324** of file **debug-printing.h**.

#define emberAfFanControlClusterPrintln (...)

Definition at line **325** of file **debug-printing.h**.

#define emberAfFanControlClusterFlush ( )

Definition at line **326** of file **debug-printing.h**.

#define emberAfFanControlClusterDebugExec (x)

Definition at line **327** of file **debug-printing.h**.

#define emberAfFanControlClusterPrintBuffer (buffer, len, withSpace)

Definition at line **328** of file **debug-printing.h**.

#define emberAfFanControlClusterPrintString (buffer)

Definition at line **329** of file **debug-printing.h**.

#define emberAfDehumidControlClusterPrint (...)

Definition at line **342** of file **debug-printing.h**.

#define emberAfDehumidControlClusterPrintln (...)

Definition at line **343** of file **debug-printing.h**.

#define emberAfDehumidControlClusterFlush ( )

Definition at line **344** of file **debug-printing.h**.

#define emberAfDehumidControlClusterDebugExec (x)

Definition at line **345** of file **debug-printing.h**.

#define emberAfDehumidControlClusterPrintBuffer (buffer, len, withSpace)

Definition at line **346** of file **debug-printing.h**.

#define emberAfDehumidControlClusterPrintString (buffer)

Definition at line **347** of file **debug-printing.h**.

#define emberAfThermostatUiConfigClusterPrint (...)

Definition at line **360** of file **debug-printing.h**.

#define emberAfThermostatUiConfigClusterPrintln (...)

Definition at line **361** of file **debug-printing.h**.

#define emberAfThermostatUiConfigClusterFlush ( )

Definition at line **362** of file **debug-printing.h**.

#define emberAfThermostatUiConfigClusterDebugExec (x)

Definition at line **363** of file **debug-printing.h**.

#define emberAfThermostatUiConfigClusterPrintBuffer (buffer, len, withSpace)

Definition at line **364** of file **debug-printing.h**.

#define emberAfThermostatUiConfigClusterPrintString (buffer)

Definition at line **365** of file **debug-printing.h**.

#define emberAfColorControlClusterPrint (...)

Definition at line **378** of file **debug-printing.h**.

#define emberAfColorControlClusterPrintln (...)

Definition at line **379** of file **debug-printing.h**.

#define emberAfColorControlClusterFlush ( )

Definition at line **380** of file **debug-printing.h**.

#define emberAfColorControlClusterDebugExec (x)

Definition at line **381** of file **debug-printing.h**.

#define emberAfColorControlClusterPrintBuffer (buffer,
len,
withSpace)

Definition at line **382** of file **debug-printing.h**.

#define emberAfColorControlClusterPrintString (buffer)

Definition at line **383** of file **debug-printing.h**.

#define emberAfBallastConfigurationClusterPrint (...)

Definition at line **396** of file **debug-printing.h**.

#define emberAfBallastConfigurationClusterPrintln (...)

Definition at line **397** of file **debug-printing.h**.

#define emberAfBallastConfigurationClusterFlush ( )

Definition at line **398** of file **debug-printing.h**.

#define emberAfBallastConfigurationClusterDebugExec (x)

Definition at line **399** of file **debug-printing.h**.

#define emberAfBallastConfigurationClusterPrintBuffer (buffer, len, withSpace)

Definition at line **400** of file **debug-printing.h**.

#define emberAfBallastConfigurationClusterPrintString (buffer)

Definition at line **401** of file **debug-printing.h**.

#define emberAfIllumMeasurementClusterPrint (...)

Definition at line **414** of file **debug-printing.h**.

#define emberAfIllumMeasurementClusterPrintln (...)

Definition at line **415** of file **debug-printing.h**.

#define emberAfIllumMeasurementClusterFlush ( )

Definition at line **416** of file **debug-printing.h**.

#define emberAfIllumMeasurementClusterDebugExec (x)

Definition at line **417** of file **debug-printing.h**.

#define emberAfIllumMeasurementClusterPrintBuffer (buffer, len, withSpace)

Definition at line **418** of file **debug-printing.h**.

#define emberAfIllumMeasurementClusterPrintString (buffer)

Definition at line **419** of file **debug-printing.h**.

#define emberAfIllumLevelSensingClusterPrint (...)

Definition at line **432** of file **debug-printing.h**.

#define emberAfIllumLevelSensingClusterPrintln (...)

Definition at line **433** of file **debug-printing.h**.

#define emberAfIllumLevelSensingClusterFlush ( )

Definition at line **434** of file **debug-printing.h**.

#define emberAfIllumLevelSensingClusterDebugExec (x)

Definition at line **435** of file **debug-printing.h**.

#define emberAfIllumLevelSensingClusterPrintBuffer (buffer, len, withSpace)

Definition at line **436** of file **debug-printing.h**.

#define emberAfIllumLevelSensingClusterPrintString (buffer)

Definition at line **437** of file **debug-printing.h**.

#define emberAfTempMeasurementClusterPrint (...)

Definition at line **450** of file **debug-printing.h**.

#define emberAfTempMeasurementClusterPrintln (...)

Definition at line **451** of file **debug-printing.h**.

#define emberAfTempMeasurementClusterFlush ( )

Definition at line **452** of file **debug-printing.h**.

#define emberAfTempMeasurementClusterDebugExec (x)

Definition at line **453** of file **debug-printing.h**.

#define emberAfTempMeasurementClusterPrintBuffer (buffer, len, withSpace)

Definition at line **454** of file **debug-printing.h**.

#define emberAfTempMeasurementClusterPrintString (buffer)

Definition at line **455** of file **debug-printing.h**.

#define emberAfPressureMeasurementClusterPrint (...)

Definition at line **468** of file **debug-printing.h**.

#define emberAfPressureMeasurementClusterPrintln (...)

Definition at line **469** of file **debug-printing.h**.

#define emberAfPressureMeasurementClusterFlush ( )

Definition at line **470** of file **debug-printing.h**.

#define emberAfPressureMeasurementClusterDebugExec (x)

Definition at line **471** of file **debug-printing.h**.

#define emberAfPressureMeasurementClusterPrintBuffer (buffer, len, withSpace)

Definition at line **472** of file **debug-printing.h**.

#define emberAfPressureMeasurementClusterPrintString (buffer)

Definition at line **473** of file **debug-printing.h**.

#define emberAfFlowMeasurementClusterPrint (...)

Definition at line **486** of file **debug-printing.h**.

#define emberAfFlowMeasurementClusterPrintln (...)

Definition at line **487** of file **debug-printing.h**.

#define emberAfFlowMeasurementClusterFlush ( )

Definition at line **488** of file **debug-printing.h**.

#define emberAfFlowMeasurementClusterDebugExec (x)

Definition at line **489** of file **debug-printing.h**.

#define emberAfFlowMeasurementClusterPrintBuffer (buffer, len, withSpace)

Definition at line **490** of file **debug-printing.h**.

#define emberAfFlowMeasurementClusterPrintString (buffer)

Definition at line **491** of file **debug-printing.h**.

#define emberAfRelativeHumidityMeasurementClusterPrint (...)

Definition at line **504** of file **debug-printing.h**.

#define emberAfRelativeHumidityMeasurementClusterPrintln (...)

Definition at line **505** of file **debug-printing.h**.

#define emberAfRelativeHumidityMeasurementClusterFlush ( )

Definition at line **506** of file **debug-printing.h**.

#define emberAfRelativeHumidityMeasurementClusterDebugExec (x)

Definition at line **507** of file **debug-printing.h**.

#define emberAfRelativeHumidityMeasurementClusterPrintBuffer (buffer, len, withSpace)

Definition at line **508** of file **debug-printing.h**.

#define emberAfRelativeHumidityMeasurementClusterPrintString (buffer)

Definition at line **509** of file **debug-printing.h**.

#define emberAfOccupancySensingClusterPrint (...)

Definition at line **522** of file **debug-printing.h**.

#define emberAfOccupancySensingClusterPrintln (...)

Definition at line **523** of file **debug-printing.h**.

#define emberAfOccupancySensingClusterFlush ( )

Definition at line **524** of file **debug-printing.h**.

#define emberAfOccupancySensingClusterDebugExec (x)

Definition at line **525** of file **debug-printing.h**.

#define emberAfOccupancySensingClusterPrintBuffer (buffer, len, withSpace)

Definition at line **526** of file **debug-printing.h**.

#define emberAfOccupancySensingClusterPrintString (buffer)

Definition at line **527** of file **debug-printing.h**.

#define emberAfIasZoneClusterPrint (...)

Definition at line **540** of file **debug-printing.h**.

#define emberAfIasZoneClusterPrintln (...)

Definition at line **541** of file **debug-printing.h**.

#define emberAfIasZoneClusterFlush ( )

Definition at line **542** of file **debug-printing.h**.

#define emberAfIasZoneClusterDebugExec (x)

Definition at line **543** of file **debug-printing.h**.

#define emberAfIasZoneClusterPrintBuffer (buffer,

```
                                                    len,
                                                    withSpace    )
```

Definition at line **544** of file **debug-printing.h**.

```
#define emberAfIasZoneClusterPrintString ( buffer    )
```

Definition at line **545** of file **debug-printing.h**.

```
#define emberAfIasAceClusterPrint ( ...    )
```

Definition at line **558** of file **debug-printing.h**.

```
#define emberAfIasAceClusterPrintln ( ...    )
```

Definition at line **559** of file **debug-printing.h**.

```
#define emberAfIasAceClusterFlush (    )
```

Definition at line **560** of file **debug-printing.h**.

```
#define emberAfIasAceClusterDebugExec ( x    )
```

Definition at line **561** of file **debug-printing.h**.

```
#define emberAfIasAceClusterPrintBuffer ( buffer,
                                          len,
                                          withSpace    )
```

Definition at line **562** of file **debug-printing.h**.

```
#define emberAfIasAceClusterPrintString ( buffer    )
```

Definition at line **563** of file **debug-printing.h**.

```
#define emberAfIasWdClusterPrint ( ...    )
```

Definition at line **576** of file **debug-printing.h**.

#define emberAfIasWdClusterPrintln (...)

Definition at line **577** of file **debug-printing.h**.

#define emberAfIasWdClusterFlush ( )

Definition at line **578** of file **debug-printing.h**.

#define emberAfIasWdClusterDebugExec (x)

Definition at line **579** of file **debug-printing.h**.

#define emberAfIasWdClusterPrintBuffer (buffer,
len,
withSpace)

Definition at line **580** of file **debug-printing.h**.

#define emberAfIasWdClusterPrintString (buffer)

Definition at line **581** of file **debug-printing.h**.

#define emberAfDoorLockClusterPrint (...)

Definition at line **594** of file **debug-printing.h**.

#define emberAfDoorLockClusterPrintln (...)

Definition at line **595** of file **debug-printing.h**.

#define emberAfDoorLockClusterFlush ( )

Definition at line **596** of file **debug-printing.h**.

#define emberAfDoorLockClusterDebugExec (x)

Definition at line **597** of file **debug-printing.h**.

#define emberAfDoorLockClusterPrintBuffer (buffer, len, withSpace)

Definition at line **598** of file **debug-printing.h**.

#define emberAfDoorLockClusterPrintString (buffer)

Definition at line **599** of file **debug-printing.h**.

#define emberAfWindowCoveringClusterPrint (...)

Definition at line **612** of file **debug-printing.h**.

#define emberAfWindowCoveringClusterPrintln (...)

Definition at line **613** of file **debug-printing.h**.

#define emberAfWindowCoveringClusterFlush ( )

Definition at line **614** of file **debug-printing.h**.

#define emberAfWindowCoveringClusterDebugExec (x)

Definition at line **615** of file **debug-printing.h**.

#define emberAfWindowCoveringClusterPrintBuffer (buffer, len, withSpace)

Definition at line **616** of file **debug-printing.h**.

#define emberAfWindowCoveringClusterPrintString (buffer)

Definition at line **617** of file **debug-printing.h**.

#define emberAfGenericTunnelClusterPrint (...)

Definition at line **630** of file **debug-printing.h**.

#define emberAfGenericTunnelClusterPrintln (...)

Definition at line **631** of file **debug-printing.h**.

#define emberAfGenericTunnelClusterFlush ( )

Definition at line **632** of file **debug-printing.h**.

#define emberAfGenericTunnelClusterDebugExec (x)

Definition at line **633** of file **debug-printing.h**.

#define emberAfGenericTunnelClusterPrintBuffer (buffer,
len,
withSpace)

Definition at line **634** of file **debug-printing.h**.

#define emberAfGenericTunnelClusterPrintString (buffer)

Definition at line **635** of file **debug-printing.h**.

#define emberAfBacnetProtocolTunnelClusterPrint (...)

Definition at line **648** of file **debug-printing.h**.

#define emberAfBacnetProtocolTunnelClusterPrintln (...)

Definition at line **649** of file **debug-printing.h**.

#define emberAfBacnetProtocolTunnelClusterFlush ( )

Definition at line **650** of file **debug-printing.h**.

#define emberAfBacnetProtocolTunnelClusterDebugExec (x)

Definition at line **651** of file **debug-printing.h**.

#define emberAfBacnetProtocolTunnelClusterPrintBuffer (buffer, len, withSpace)

Definition at line **652** of file **debug-printing.h**.

#define emberAfBacnetProtocolTunnelClusterPrintString (buffer)

Definition at line **653** of file **debug-printing.h**.

#define emberAfOtaBootloadClusterPrint (...)

Definition at line **666** of file **debug-printing.h**.

#define emberAfOtaBootloadClusterPrintln (...)

Definition at line **667** of file **debug-printing.h**.

#define emberAfOtaBootloadClusterFlush ( )

Definition at line **668** of file **debug-printing.h**.

#define emberAfOtaBootloadClusterDebugExec (x)

Definition at line **669** of file **debug-printing.h**.

#define emberAfOtaBootloadClusterPrintBuffer (buffer, len, withSpace)

Definition at line **670** of file **debug-printing.h**.

#define emberAfOtaBootloadClusterPrintString (buffer)

Definition at line **671** of file **debug-printing.h**.

#define emberAfDemandResponseLoadControlClusterPrint (...)

Definition at line **684** of file **debug-printing.h**.

#define emberAfDemandResponseLoadControlClusterPrintln (...)

Definition at line **685** of file **debug-printing.h**.

#define emberAfDemandResponseLoadControlClusterFlush ( )

Definition at line **686** of file **debug-printing.h**.

#define emberAfDemandResponseLoadControlClusterDebugExec (x)

Definition at line **687** of file **debug-printing.h**.

#define emberAfDemandResponseLoadControlClusterPrintBuffer (buffer, len, withSpace)

Definition at line **688** of file **debug-printing.h**.

#define emberAfDemandResponseLoadControlClusterPrintString (buffer)

Definition at line **689** of file **debug-printing.h**.

#define emberAfSimpleMeteringClusterPrint (...)

Definition at line **702** of file **debug-printing.h**.

#define emberAfSimpleMeteringClusterPrintln (...)

Definition at line **703** of file **debug-printing.h**.

#define emberAfSimpleMeteringClusterFlush ( )

Definition at line **704** of file **debug-printing.h**.

#define emberAfSimpleMeteringClusterDebugExec (x)

Definition at line **705** of file **debug-printing.h**.

#define emberAfSimpleMeteringClusterPrintBuffer (buffer, len, withSpace)

Definition at line **706** of file **debug-printing.h**.

#define emberAfSimpleMeteringClusterPrintString (buffer)

Definition at line **707** of file **debug-printing.h**.

#define emberAfPriceClusterPrint (...)

Definition at line **720** of file **debug-printing.h**.

#define emberAfPriceClusterPrintln (...)

Definition at line **721** of file **debug-printing.h**.

#define emberAfPriceClusterFlush ( )

Definition at line **722** of file **debug-printing.h**.

#define emberAfPriceClusterDebugExec (x)

Definition at line **723** of file **debug-printing.h**.

```
#define emberAfPriceClusterPrintBuffer ( buffer,
                                         len,
                                         withSpace )
```

Definition at line **724** of file **debug-printing.h**.

```
#define emberAfPriceClusterPrintString ( buffer )
```

Definition at line **725** of file **debug-printing.h**.

```
#define emberAfMessagingClusterPrint ( ... )
```

Definition at line **738** of file **debug-printing.h**.

```
#define emberAfMessagingClusterPrintln ( ... )
```

Definition at line **739** of file **debug-printing.h**.

```
#define emberAfMessagingClusterFlush ( )
```

Definition at line **740** of file **debug-printing.h**.

```
#define emberAfMessagingClusterDebugExec ( x )
```

Definition at line **741** of file **debug-printing.h**.

```
#define emberAfMessagingClusterPrintBuffer ( buffer,
                                             len,
                                             withSpace )
```

Definition at line **742** of file **debug-printing.h**.

```
#define emberAfMessagingClusterPrintString ( buffer )
```

Definition at line **743** of file **debug-printing.h**.

#define emberAfTunnelingClusterPrint (...)

Definition at line **756** of file **debug-printing.h**.

#define emberAfTunnelingClusterPrintln (...)

Definition at line **757** of file **debug-printing.h**.

#define emberAfTunnelingClusterFlush ( )

Definition at line **758** of file **debug-printing.h**.

#define emberAfTunnelingClusterDebugExec (x)

Definition at line **759** of file **debug-printing.h**.

#define emberAfTunnelingClusterPrintBuffer (buffer, len, withSpace)

Definition at line **760** of file **debug-printing.h**.

#define emberAfTunnelingClusterPrintString (buffer)

Definition at line **761** of file **debug-printing.h**.

#define emberAfPrepaymentClusterPrint (...)

Definition at line **774** of file **debug-printing.h**.

#define emberAfPrepaymentClusterPrintln (...)

Definition at line **775** of file **debug-printing.h**.

#define emberAfPrepaymentClusterFlush ( )

Definition at line **776** of file **debug-printing.h**.

#define emberAfPrepaymentClusterDebugExec (x)

Definition at line **777** of file **debug-printing.h**.

#define emberAfPrepaymentClusterPrintBuffer (buffer, len, withSpace)

Definition at line **778** of file **debug-printing.h**.

#define emberAfPrepaymentClusterPrintString (buffer)

Definition at line **779** of file **debug-printing.h**.

#define emberAfKeyEstablishmentClusterPrint (...)

Definition at line **792** of file **debug-printing.h**.

#define emberAfKeyEstablishmentClusterPrintln (...)

Definition at line **793** of file **debug-printing.h**.

#define emberAfKeyEstablishmentClusterFlush ( )

Definition at line **794** of file **debug-printing.h**.

#define emberAfKeyEstablishmentClusterDebugExec (x)

Definition at line **795** of file **debug-printing.h**.

#define emberAfKeyEstablishmentClusterPrintBuffer (buffer, len, withSpace)

Definition at line **796** of file **debug-printing.h**.

#define emberAfKeyEstablishmentClusterPrintString (buffer)

Definition at line **797** of file **debug-printing.h**.

#define emberAfSampleMfgSpecificClusterPrint (...)

Definition at line **810** of file **debug-printing.h**.

#define emberAfSampleMfgSpecificClusterPrintln (...)

Definition at line **811** of file **debug-printing.h**.

#define emberAfSampleMfgSpecificClusterFlush ( )

Definition at line **812** of file **debug-printing.h**.

#define emberAfSampleMfgSpecificClusterDebugExec (x)

Definition at line **813** of file **debug-printing.h**.

#define emberAfSampleMfgSpecificClusterPrintBuffer (buffer, len, withSpace)

Definition at line **814** of file **debug-printing.h**.

#define emberAfSampleMfgSpecificClusterPrintString (buffer)

Definition at line **815** of file **debug-printing.h**.

#define emberAfCorePrint (...)

Definition at line **829** of file **debug-printing.h**.

#define emberAfCorePrintln (...)

Definition at line **830** of file **debug-printing.h**.

#define emberAfCoreFlush ( )

Definition at line **831** of file **debug-printing.h**.

#define emberAfCoreDebugExec (x)

Definition at line **832** of file **debug-printing.h**.

#define emberAfCorePrintBuffer (buffer,
len,
withSpace)

Definition at line **833** of file **debug-printing.h**.

#define emberAfCorePrintString (buffer)

Definition at line **834** of file **debug-printing.h**.

#define emberAfDebugPrint (...)

Definition at line **848** of file **debug-printing.h**.

#define emberAfDebugPrintln (...)

Definition at line **849** of file **debug-printing.h**.

#define emberAfDebugFlush ( )

Definition at line **850** of file **debug-printing.h**.

#define emberAfDebugDebugExec (x)

Definition at line **851** of file **debug-printing.h**.

#define emberAfDebugPrintBuffer (buffer,

```
                                        len,
                                        withSpace    )
```

Definition at line **852** of file **debug-printing.h**.

```
#define emberAfDebugPrintString ( buffer    )
```

Definition at line **853** of file **debug-printing.h**.

```
#define emberAfAppPrint ( ...    )
```

Definition at line **867** of file **debug-printing.h**.

```
#define emberAfAppPrintln ( ...    )
```

Definition at line **868** of file **debug-printing.h**.

```
#define emberAfAppFlush (    )
```

Definition at line **869** of file **debug-printing.h**.

```
#define emberAfAppDebugExec ( x    )
```

Definition at line **870** of file **debug-printing.h**.

```
#define emberAfAppPrintBuffer ( buffer,
                                  len,
                                  withSpace    )
```

Definition at line **871** of file **debug-printing.h**.

```
#define emberAfAppPrintString ( buffer    )
```

Definition at line **872** of file **debug-printing.h**.

```
#define emberAfSecurityPrint ( ...    )
```

Definition at line **886** of file **debug-printing.h**.

#define emberAfSecurityPrintln (...)

Definition at line **887** of file **debug-printing.h**.

#define emberAfSecurityFlush ( )

Definition at line **888** of file **debug-printing.h**.

#define emberAfSecurityDebugExec (x)

Definition at line **889** of file **debug-printing.h**.

#define emberAfSecurityPrintBuffer (buffer, len, withSpace)

Definition at line **890** of file **debug-printing.h**.

#define emberAfSecurityPrintString (buffer)

Definition at line **891** of file **debug-printing.h**.

#define emberAfAttributesPrint (...)

Definition at line **905** of file **debug-printing.h**.

#define emberAfAttributesPrintln (...)

Definition at line **906** of file **debug-printing.h**.

#define emberAfAttributesFlush ( )

Definition at line **907** of file **debug-printing.h**.

#define emberAfAttributesDebugExec (x)

Definition at line **908** of file **debug-printing.h**.

#define emberAfAttributesPrintBuffer (buffer, len, withSpace)

Definition at line **909** of file **debug-printing.h**.

#define emberAfAttributesPrintString (buffer)

Definition at line **910** of file **debug-printing.h**.

#define emberAfReportingPrint (...)

Definition at line **924** of file **debug-printing.h**.

#define emberAfReportingPrintln (...)

Definition at line **925** of file **debug-printing.h**.

#define emberAfReportingFlush ( )

Definition at line **926** of file **debug-printing.h**.

#define emberAfReportingDebugExec (x)

Definition at line **927** of file **debug-printing.h**.

#define emberAfReportingPrintBuffer (buffer, len, withSpace)

Definition at line **928** of file **debug-printing.h**.

#define emberAfReportingPrintString (buffer)

Definition at line **929** of file **debug-printing.h**.

#define emberAfServiceDiscoveryPrint (...)

Definition at line **943** of file **debug-printing.h**.

#define emberAfServiceDiscoveryPrintln (...)

Definition at line **944** of file **debug-printing.h**.

#define emberAfServiceDiscoveryFlush ( )

Definition at line **945** of file **debug-printing.h**.

#define emberAfServiceDiscoveryDebugExec (x)

Definition at line **946** of file **debug-printing.h**.

#define emberAfServiceDiscoveryPrintBuffer (buffer, len, withSpace)

Definition at line **947** of file **debug-printing.h**.

#define emberAfServiceDiscoveryPrintString (buffer)

Definition at line **948** of file **debug-printing.h**.

#define emberAfRegistrationPrint (...)

Definition at line **962** of file **debug-printing.h**.

#define emberAfRegistrationPrintln (...)

Definition at line **963** of file **debug-printing.h**.

#define emberAfRegistrationFlush ( )

Definition at line **964** of file **debug-printing.h**.

#define emberAfRegistrationDebugExec (x)

Definition at line **965** of file **debug-printing.h**.

#define emberAfRegistrationPrintBuffer (buffer, len, withSpace)

Definition at line **966** of file **debug-printing.h**.

#define emberAfRegistrationPrintString (buffer)

Definition at line **967** of file **debug-printing.h**.

#define emberAfZdoPrint (...)

Definition at line **981** of file **debug-printing.h**.

#define emberAfZdoPrintln (...)

Definition at line **982** of file **debug-printing.h**.

#define emberAfZdoFlush ( )

Definition at line **983** of file **debug-printing.h**.

#define emberAfZdoDebugExec (x)

Definition at line **984** of file **debug-printing.h**.

#define emberAfZdoPrintBuffer (buffer, len, withSpace)

Definition at line **985** of file **debug-printing.h**.

#define emberAfZdoPrintString (buffer)

Definition at line **986** of file **debug-printing.h**.

#define emberAfCustom1Print (...)

Definition at line **1000** of file **debug-printing.h**.

#define emberAfCustom1Println (...)

Definition at line **1001** of file **debug-printing.h**.

#define emberAfCustom1Flush ( )

Definition at line **1002** of file **debug-printing.h**.

#define emberAfCustom1DebugExec (x)

Definition at line **1003** of file **debug-printing.h**.

#define emberAfCustom1PrintBuffer (buffer,
len,
withSpace)

Definition at line **1004** of file **debug-printing.h**.

#define emberAfCustom1PrintString (buffer)

Definition at line **1005** of file **debug-printing.h**.

#define emberAfCustom2Print (...)

Definition at line **1019** of file **debug-printing.h**.

#define emberAfCustom2Println (...)

Definition at line **1020** of file **debug-printing.h**.

#define emberAfCustom2Flush ( )

Definition at line **1021** of file **debug-printing.h**.

#define emberAfCustom2DebugExec (x)

Definition at line **1022** of file **debug-printing.h**.

#define emberAfCustom2PrintBuffer (buffer, len, withSpace)

Definition at line **1023** of file **debug-printing.h**.

#define emberAfCustom2PrintString (buffer)

Definition at line **1024** of file **debug-printing.h**.

#define emberAfCustom3Print (...)

Definition at line **1038** of file **debug-printing.h**.

#define emberAfCustom3Println (...)

Definition at line **1039** of file **debug-printing.h**.

#define emberAfCustom3Flush ( )

Definition at line **1040** of file **debug-printing.h**.

#define emberAfCustom3DebugExec (x)

Definition at line **1041** of file **debug-printing.h**.

```
#define emberAfCustom3PrintBuffer ( buffer,
                                    len,
                                    withSpace )
```

Definition at line **1042** of file **debug-printing.h**.

```
#define emberAfCustom3PrintString ( buffer )
```

Definition at line **1043** of file **debug-printing.h**.

---

debug.h File Reference

Go to the source code of this file.

Defines

```
#define ZCL_BASIC_DEBUG_ON
#define ZCL_BASIC_DEBUG_PRINT(...)
#define ZCL_BASIC_DEBUG_PRINT_LINE(...)
#define ZCL_BASIC_DEBUG_WAITSEND()
#define ZCL_BASIC_DEBUG_EXECUTE(x)
#define ZCL_POWER_CONFIG_DEBUG_ON
#define ZCL_POWER_CONFIG_DEBUG_PRINT(...)
#define ZCL_POWER_CONFIG_DEBUG_PRINT_LINE(...)
#define ZCL_POWER_CONFIG_DEBUG_WAITSEND()
#define ZCL_POWER_CONFIG_DEBUG_EXECUTE(x)
#define ZCL_DEVICE_TEMP_DEBUG_ON
#define ZCL_DEVICE_TEMP_DEBUG_PRINT(...)
#define ZCL_DEVICE_TEMP_DEBUG_PRINT_LINE(...)
#define ZCL_DEVICE_TEMP_DEBUG_WAITSEND()
#define ZCL_DEVICE_TEMP_DEBUG_EXECUTE(x)
#define ZCL_IDENTIFY_DEBUG_ON
#define ZCL_IDENTIFY_DEBUG_PRINT(...)
#define ZCL_IDENTIFY_DEBUG_PRINT_LINE(...)
#define ZCL_IDENTIFY_DEBUG_WAITSEND()
#define ZCL_IDENTIFY_DEBUG_EXECUTE(x)
#define ZCL_GROUPS_DEBUG_ON
#define ZCL_GROUPS_DEBUG_PRINT(...)
#define ZCL_GROUPS_DEBUG_PRINT_LINE(...)
#define ZCL_GROUPS_DEBUG_WAITSEND()
#define ZCL_GROUPS_DEBUG_EXECUTE(x)
#define ZCL_SCENES_DEBUG_ON
#define ZCL_SCENES_DEBUG_PRINT(...)
#define ZCL_SCENES_DEBUG_PRINT_LINE(...)
#define ZCL_SCENES_DEBUG_WAITSEND()
#define ZCL_SCENES_DEBUG_EXECUTE(x)
#define ZCL_ON_OFF_DEBUG_ON
#define ZCL_ON_OFF_DEBUG_PRINT(...)
#define ZCL_ON_OFF_DEBUG_PRINT_LINE(...)
#define ZCL_ON_OFF_DEBUG_WAITSEND()
```

```
#define ZCL_ON_OFF_DEBUG_EXECUTE(x)
#define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_ON
#define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_PRINT(...)
#define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_PRINT_LINE(...)
#define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_WAITSEND()
#define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_EXECUTE(x)
#define ZCL_LEVEL_CONTROL_DEBUG_ON
#define ZCL_LEVEL_CONTROL_DEBUG_PRINT(...)
#define ZCL_LEVEL_CONTROL_DEBUG_PRINT_LINE(...)
#define ZCL_LEVEL_CONTROL_DEBUG_WAITSEND()
#define ZCL_LEVEL_CONTROL_DEBUG_EXECUTE(x)
#define ZCL_TIME_DEBUG_ON
#define ZCL_TIME_DEBUG_PRINT(...)
#define ZCL_TIME_DEBUG_PRINT_LINE(...)
#define ZCL_TIME_DEBUG_WAITSEND()
#define ZCL_TIME_DEBUG_EXECUTE(x)
#define ZCL_BINARY_INPUT_BASIC_DEBUG_ON
#define ZCL_BINARY_INPUT_BASIC_DEBUG_PRINT(...)
#define ZCL_BINARY_INPUT_BASIC_DEBUG_PRINT_LINE(...)
#define ZCL_BINARY_INPUT_BASIC_DEBUG_WAITSEND()
#define ZCL_BINARY_INPUT_BASIC_DEBUG_EXECUTE(x)
#define ZCL_COMMISSIONING_DEBUG_ON
#define ZCL_COMMISSIONING_DEBUG_PRINT(...)
#define ZCL_COMMISSIONING_DEBUG_PRINT_LINE(...)
#define ZCL_COMMISSIONING_DEBUG_WAITSEND()
#define ZCL_COMMISSIONING_DEBUG_EXECUTE(x)
#define ZCL_SHADE_CONFIG_DEBUG_ON
#define ZCL_SHADE_CONFIG_DEBUG_PRINT(...)
#define ZCL_SHADE_CONFIG_DEBUG_PRINT_LINE(...)
#define ZCL_SHADE_CONFIG_DEBUG_WAITSEND()
#define ZCL_SHADE_CONFIG_DEBUG_EXECUTE(x)
#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_ON
#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_PRINT(...)
#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_PRINT_LINE(...)
#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_WAITSEND()
#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_EXECUTE(x)
#define ZCL_THERMOSTAT_DEBUG_ON
#define ZCL_THERMOSTAT_DEBUG_PRINT(...)
#define ZCL_THERMOSTAT_DEBUG_PRINT_LINE(...)
#define ZCL_THERMOSTAT_DEBUG_WAITSEND()
#define ZCL_THERMOSTAT_DEBUG_EXECUTE(x)
#define ZCL_FAN_CONTROL_DEBUG_ON
#define ZCL_FAN_CONTROL_DEBUG_PRINT(...)
#define ZCL_FAN_CONTROL_DEBUG_PRINT_LINE(...)
```

#define **ZCL_FAN_CONTROL_DEBUG_WAITSEND**()
#define **ZCL_FAN_CONTROL_DEBUG_EXECUTE**(x)
#define **ZCL_DEHUMID_CONTROL_DEBUG_ON**
#define **ZCL_DEHUMID_CONTROL_DEBUG_PRINT**(...)
#define **ZCL_DEHUMID_CONTROL_DEBUG_PRINT_LINE**(...)
#define **ZCL_DEHUMID_CONTROL_DEBUG_WAITSEND**()
#define **ZCL_DEHUMID_CONTROL_DEBUG_EXECUTE**(x)
#define **ZCL_THERMOSTAT_UI_CONFIG_DEBUG_ON**
#define **ZCL_THERMOSTAT_UI_CONFIG_DEBUG_PRINT**(...)
#define **ZCL_THERMOSTAT_UI_CONFIG_DEBUG_PRINT_LINE**(...)
#define **ZCL_THERMOSTAT_UI_CONFIG_DEBUG_WAITSEND**()
#define **ZCL_THERMOSTAT_UI_CONFIG_DEBUG_EXECUTE**(x)
#define **ZCL_COLOR_CONTROL_DEBUG_ON**
#define **ZCL_COLOR_CONTROL_DEBUG_PRINT**(...)
#define **ZCL_COLOR_CONTROL_DEBUG_PRINT_LINE**(...)
#define **ZCL_COLOR_CONTROL_DEBUG_WAITSEND**()
#define **ZCL_COLOR_CONTROL_DEBUG_EXECUTE**(x)
#define **ZCL_ILLUM_MEASUREMENT_DEBUG_ON**
#define **ZCL_ILLUM_MEASUREMENT_DEBUG_PRINT**(...)
#define **ZCL_ILLUM_MEASUREMENT_DEBUG_PRINT_LINE**(...)
#define **ZCL_ILLUM_MEASUREMENT_DEBUG_WAITSEND**()
#define **ZCL_ILLUM_MEASUREMENT_DEBUG_EXECUTE**(x)
#define **ZCL_ILLUM_LEVEL_SENSING_DEBUG_ON**
#define **ZCL_ILLUM_LEVEL_SENSING_DEBUG_PRINT**(...)
#define **ZCL_ILLUM_LEVEL_SENSING_DEBUG_PRINT_LINE**(...)
#define **ZCL_ILLUM_LEVEL_SENSING_DEBUG_WAITSEND**()
#define **ZCL_ILLUM_LEVEL_SENSING_DEBUG_EXECUTE**(x)
#define **ZCL_TEMP_MEASUREMENT_DEBUG_ON**
#define **ZCL_TEMP_MEASUREMENT_DEBUG_PRINT**(...)
#define **ZCL_TEMP_MEASUREMENT_DEBUG_PRINT_LINE**(...)
#define **ZCL_TEMP_MEASUREMENT_DEBUG_WAITSEND**()
#define **ZCL_TEMP_MEASUREMENT_DEBUG_EXECUTE**(x)
#define **ZCL_PRESSURE_MEASUREMENT_DEBUG_ON**
#define **ZCL_PRESSURE_MEASUREMENT_DEBUG_PRINT**(...)
#define **ZCL_PRESSURE_MEASUREMENT_DEBUG_PRINT_LINE**(...)
#define **ZCL_PRESSURE_MEASUREMENT_DEBUG_WAITSEND**()
#define **ZCL_PRESSURE_MEASUREMENT_DEBUG_EXECUTE**(x)
#define **ZCL_FLOW_MEASUREMENT_DEBUG_ON**
#define **ZCL_FLOW_MEASUREMENT_DEBUG_PRINT**(...)
#define **ZCL_FLOW_MEASUREMENT_DEBUG_PRINT_LINE**(...)
#define **ZCL_FLOW_MEASUREMENT_DEBUG_WAITSEND**()
#define **ZCL_FLOW_MEASUREMENT_DEBUG_EXECUTE**(x)
#define **ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_ON**
#define **ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_PRINT**(...)

```
#define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_PRINT_LINE(...)
#define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_WAITSEND()
#define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_EXECUTE(x)
#define ZCL_OCCUPANCY_SENSING_DEBUG_ON
#define ZCL_OCCUPANCY_SENSING_DEBUG_PRINT(...)
#define ZCL_OCCUPANCY_SENSING_DEBUG_PRINT_LINE(...)
#define ZCL_OCCUPANCY_SENSING_DEBUG_WAITSEND()
#define ZCL_OCCUPANCY_SENSING_DEBUG_EXECUTE(x)
#define ZCL_IAS_ZONE_DEBUG_ON
#define ZCL_IAS_ZONE_DEBUG_PRINT(...)
#define ZCL_IAS_ZONE_DEBUG_PRINT_LINE(...)
#define ZCL_IAS_ZONE_DEBUG_WAITSEND()
#define ZCL_IAS_ZONE_DEBUG_EXECUTE(x)
#define ZCL_IAS_ACE_DEBUG_ON
#define ZCL_IAS_ACE_DEBUG_PRINT(...)
#define ZCL_IAS_ACE_DEBUG_PRINT_LINE(...)
#define ZCL_IAS_ACE_DEBUG_WAITSEND()
#define ZCL_IAS_ACE_DEBUG_EXECUTE(x)
#define ZCL_IAS_WD_DEBUG_ON
#define ZCL_IAS_WD_DEBUG_PRINT(...)
#define ZCL_IAS_WD_DEBUG_PRINT_LINE(...)
#define ZCL_IAS_WD_DEBUG_WAITSEND()
#define ZCL_IAS_WD_DEBUG_EXECUTE(x)
#define ZCL_DOOR_LOCK_DEBUG_ON
#define ZCL_DOOR_LOCK_DEBUG_PRINT(...)
#define ZCL_DOOR_LOCK_DEBUG_PRINT_LINE(...)
#define ZCL_DOOR_LOCK_DEBUG_WAITSEND()
#define ZCL_DOOR_LOCK_DEBUG_EXECUTE(x)
#define ZCL_WINDOW_COVERING_DEBUG_ON
#define ZCL_WINDOW_COVERING_DEBUG_PRINT(...)
#define ZCL_WINDOW_COVERING_DEBUG_PRINT_LINE(...)
#define ZCL_WINDOW_COVERING_DEBUG_WAITSEND()
#define ZCL_WINDOW_COVERING_DEBUG_EXECUTE(x)
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_ON
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_PRINT(...)
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_PRINT_LINE(...)
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_WAITSEND()
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_EXECUTE(x)
#define ZCL_SIMPLE_METERING_DEBUG_ON
#define ZCL_SIMPLE_METERING_DEBUG_PRINT(...)
#define ZCL_SIMPLE_METERING_DEBUG_PRINT_LINE(...)
#define ZCL_SIMPLE_METERING_DEBUG_WAITSEND()
#define ZCL_SIMPLE_METERING_DEBUG_EXECUTE(x)
#define ZCL_PRICE_DEBUG_ON
```

#define **ZCL_PRICE_DEBUG_PRINT**(...)
#define **ZCL_PRICE_DEBUG_PRINT_LINE**(...)
#define **ZCL_PRICE_DEBUG_WAITSEND**()
#define **ZCL_PRICE_DEBUG_EXECUTE**(x)
#define **ZCL_MESSAGING_DEBUG_ON**
#define **ZCL_MESSAGING_DEBUG_PRINT**(...)
#define **ZCL_MESSAGING_DEBUG_PRINT_LINE**(...)
#define **ZCL_MESSAGING_DEBUG_WAITSEND**()
#define **ZCL_MESSAGING_DEBUG_EXECUTE**(x)
#define **ZCL_KEY_ESTABLISHMENT_DEBUG_ON**
#define **ZCL_KEY_ESTABLISHMENT_DEBUG_PRINT**(...)
#define **ZCL_KEY_ESTABLISHMENT_DEBUG_PRINT_LINE**(...)
#define **ZCL_KEY_ESTABLISHMENT_DEBUG_WAITSEND**()
#define **ZCL_KEY_ESTABLISHMENT_DEBUG_EXECUTE**(x)
#define **ZCL_GENERIC_TUNNEL_DEBUG_ON**
#define **ZCL_GENERIC_TUNNEL_DEBUG_PRINT**(...)
#define **ZCL_GENERIC_TUNNEL_DEBUG_PRINT_LINE**(...)
#define **ZCL_GENERIC_TUNNEL_DEBUG_WAITSEND**()
#define **ZCL_GENERIC_TUNNEL_DEBUG_EXECUTE**(x)
#define **ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_ON**
#define **ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_PRINT**(...)
#define **ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_PRINT_LINE**(...)
#define **ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_WAITSEND**()
#define **ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_EXECUTE**(x)
#define **DEBUG_PRINT_TEST**
#define **LOG_LEVELS**

Define Documentation

#define ZCL_BASIC_DEBUG_ON

Definition at line **60** of file **debug.h**.

#define ZCL_BASIC_DEBUG_PRINT (...)

Definition at line **61** of file **debug.h**.

#define ZCL_BASIC_DEBUG_PRINT_LINE (...)

Definition at line **62** of file **debug.h**.

#define ZCL_BASIC_DEBUG_WAITSEND ( )

Definition at line **63** of file **debug.h**.

#define ZCL_BASIC_DEBUG_EXECUTE (x)

Definition at line **64** of file **debug.h**.

#define ZCL_POWER_CONFIG_DEBUG_ON

Definition at line **78** of file **debug.h**.

#define ZCL_POWER_CONFIG_DEBUG_PRINT (...)

Definition at line **79** of file **debug.h**.

#define ZCL_POWER_CONFIG_DEBUG_PRINT_LINE (...)

Definition at line **80** of file **debug.h**.

#define ZCL_POWER_CONFIG_DEBUG_WAITSEND ( )

Definition at line **81** of file **debug.h**.

#define ZCL_POWER_CONFIG_DEBUG_EXECUTE (x)

Definition at line **82** of file **debug.h**.

#define ZCL_DEVICE_TEMP_DEBUG_ON

Definition at line **96** of file **debug.h**.

#define ZCL_DEVICE_TEMP_DEBUG_PRINT (...)

Definition at line **97** of file **debug.h**.

#define ZCL_DEVICE_TEMP_DEBUG_PRINT_LINE (...)

Definition at line **98** of file **debug.h**.

#define ZCL_DEVICE_TEMP_DEBUG_WAITSEND ( )

Definition at line **99** of file **debug.h**.

#define ZCL_DEVICE_TEMP_DEBUG_EXECUTE (x)

Definition at line **100** of file **debug.h**.

#define ZCL_IDENTIFY_DEBUG_ON

Definition at line **114** of file **debug.h**.

#define ZCL_IDENTIFY_DEBUG_PRINT (...)

Definition at line **115** of file **debug.h**.

#define ZCL_IDENTIFY_DEBUG_PRINT_LINE (...)

Definition at line **116** of file **debug.h**.

#define ZCL_IDENTIFY_DEBUG_WAITSEND ( )

Definition at line **117** of file **debug.h**.

#define ZCL_IDENTIFY_DEBUG_EXECUTE (x)

Definition at line **118** of file **debug.h**.

#define ZCL_GROUPS_DEBUG_ON

Definition at line **132** of file **debug.h**.

#define ZCL_GROUPS_DEBUG_PRINT (...)

Definition at line **133** of file **debug.h**.

#define ZCL_GROUPS_DEBUG_PRINT_LINE (...)

Definition at line **134** of file **debug.h**.

#define ZCL_GROUPS_DEBUG_WAITSEND ( )

Definition at line **135** of file **debug.h**.

#define ZCL_GROUPS_DEBUG_EXECUTE (x)

Definition at line **136** of file **debug.h**.

#define ZCL_SCENES_DEBUG_ON

Definition at line **150** of file **debug.h**.

#define ZCL_SCENES_DEBUG_PRINT (...)

Definition at line **151** of file **debug.h**.

#define ZCL_SCENES_DEBUG_PRINT_LINE (...)

Definition at line **152** of file **debug.h**.

#define ZCL_SCENES_DEBUG_WAITSEND ( )

Definition at line **153** of file **debug.h**.

#define ZCL_SCENES_DEBUG_EXECUTE (x)

Definition at line **154** of file **debug.h**.

#define ZCL_ON_OFF_DEBUG_ON

Definition at line **168** of file **debug.h**.

#define ZCL_ON_OFF_DEBUG_PRINT (...)

Definition at line **169** of file **debug.h**.

#define ZCL_ON_OFF_DEBUG_PRINT_LINE (...)

Definition at line **170** of file **debug.h**.

#define ZCL_ON_OFF_DEBUG_WAITSEND ( )

Definition at line **171** of file **debug.h**.

#define ZCL_ON_OFF_DEBUG_EXECUTE (x)

Definition at line **172** of file **debug.h**.

#define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_ON

Definition at line **186** of file **debug.h**.

#define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_PRINT (...)

Definition at line **187** of file **debug.h**.

#define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_PRINT_LINE (...)

Definition at line **188** of file **debug.h**.

#define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_WAITSEND ( )

Definition at line **189** of file **debug.h**.

#define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_EXECUTE (x)

Definition at line **190** of file **debug.h**.

#define ZCL_LEVEL_CONTROL_DEBUG_ON

Definition at line **204** of file **debug.h**.

#define ZCL_LEVEL_CONTROL_DEBUG_PRINT (...)

Definition at line **205** of file **debug.h**.

#define ZCL_LEVEL_CONTROL_DEBUG_PRINT_LINE (...)

Definition at line **206** of file **debug.h**.

#define ZCL_LEVEL_CONTROL_DEBUG_WAITSEND ( )

Definition at line **207** of file **debug.h**.

#define ZCL_LEVEL_CONTROL_DEBUG_EXECUTE (x)

Definition at line **208** of file **debug.h**.

#define ZCL_TIME_DEBUG_ON

Definition at line **222** of file **debug.h**.

#define ZCL_TIME_DEBUG_PRINT (...)

Definition at line **223** of file **debug.h**.

#define ZCL_TIME_DEBUG_PRINT_LINE (...)

Definition at line **224** of file **debug.h**.

#define ZCL_TIME_DEBUG_WAITSEND ( )

Definition at line **225** of file **debug.h**.

#define ZCL_TIME_DEBUG_EXECUTE (x)

Definition at line **226** of file **debug.h**.

#define ZCL_BINARY_INPUT_BASIC_DEBUG_ON

Definition at line **240** of file **debug.h**.

#define ZCL_BINARY_INPUT_BASIC_DEBUG_PRINT (...)

Definition at line **241** of file **debug.h**.

#define ZCL_BINARY_INPUT_BASIC_DEBUG_PRINT_LINE (...)

Definition at line **242** of file **debug.h**.

#define ZCL_BINARY_INPUT_BASIC_DEBUG_WAITSEND ( )

Definition at line **243** of file **debug.h**.

#define ZCL_BINARY_INPUT_BASIC_DEBUG_EXECUTE (x)

Definition at line **244** of file **debug.h**.

#define ZCL_COMMISSIONING_DEBUG_ON

Definition at line **258** of file **debug.h**.

#define ZCL_COMMISSIONING_DEBUG_PRINT (...)

Definition at line **259** of file **debug.h**.

#define ZCL_COMMISSIONING_DEBUG_PRINT_LINE (...)

Definition at line **260** of file **debug.h**.

#define ZCL_COMMISSIONING_DEBUG_WAITSEND ( )

Definition at line **261** of file **debug.h**.

#define ZCL_COMMISSIONING_DEBUG_EXECUTE (x)

Definition at line **262** of file **debug.h**.

#define ZCL_SHADE_CONFIG_DEBUG_ON

Definition at line **276** of file **debug.h**.

#define ZCL_SHADE_CONFIG_DEBUG_PRINT (...)

Definition at line **277** of file **debug.h**.

#define ZCL_SHADE_CONFIG_DEBUG_PRINT_LINE (...)

Definition at line **278** of file **debug.h**.

#define ZCL_SHADE_CONFIG_DEBUG_WAITSEND ( )

Definition at line **279** of file **debug.h**.

#define ZCL_SHADE_CONFIG_DEBUG_EXECUTE (x)

Definition at line **280** of file **debug.h**.

#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_ON

Definition at line **294** of file **debug.h**.

#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_PRINT (...)

Definition at line **295** of file **debug.h**.

#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_PRINT_LINE (...)

Definition at line **296** of file **debug.h**.

#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_WAITSEND ( )

Definition at line **297** of file **debug.h**.

#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_EXECUTE (x)

Definition at line **298** of file **debug.h**.

#define ZCL_THERMOSTAT_DEBUG_ON

Definition at line **312** of file **debug.h**.

#define ZCL_THERMOSTAT_DEBUG_PRINT (...)

Definition at line **313** of file **debug.h**.

#define ZCL_THERMOSTAT_DEBUG_PRINT_LINE (...)

Definition at line **314** of file **debug.h**.

#define ZCL_THERMOSTAT_DEBUG_WAITSEND ( )

Definition at line **315** of file **debug.h**.

#define ZCL_THERMOSTAT_DEBUG_EXECUTE (x)

Definition at line **316** of file **debug.h**.

#define ZCL_FAN_CONTROL_DEBUG_ON

Definition at line **330** of file **debug.h**.

#define ZCL_FAN_CONTROL_DEBUG_PRINT (...)

Definition at line **331** of file **debug.h**.

#define ZCL_FAN_CONTROL_DEBUG_PRINT_LINE (...)

Definition at line **332** of file **debug.h**.

#define ZCL_FAN_CONTROL_DEBUG_WAITSEND ( )

Definition at line **333** of file **debug.h**.

#define ZCL_FAN_CONTROL_DEBUG_EXECUTE (x)

Definition at line **334** of file **debug.h**.

#define ZCL_DEHUMID_CONTROL_DEBUG_ON

Definition at line **348** of file **debug.h**.

#define ZCL_DEHUMID_CONTROL_DEBUG_PRINT (...)

Definition at line **349** of file **debug.h**.

#define ZCL_DEHUMID_CONTROL_DEBUG_PRINT_LINE (...)

Definition at line **350** of file **debug.h**.

#define ZCL_DEHUMID_CONTROL_DEBUG_WAITSEND ( )

Definition at line **351** of file **debug.h**.

#define ZCL_DEHUMID_CONTROL_DEBUG_EXECUTE (x)

Definition at line **352** of file **debug.h**.

#define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_ON

Definition at line **366** of file **debug.h**.

#define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_PRINT (...)

Definition at line **367** of file **debug.h**.

#define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_PRINT_LINE (...)

Definition at line **368** of file **debug.h**.

#define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_WAITSEND ( )

Definition at line **369** of file **debug.h**.

#define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_EXECUTE (x)

Definition at line **370** of file **debug.h**.

#define ZCL_COLOR_CONTROL_DEBUG_ON

Definition at line **384** of file **debug.h**.

#define ZCL_COLOR_CONTROL_DEBUG_PRINT (...)

Definition at line **385** of file **debug.h**.

#define ZCL_COLOR_CONTROL_DEBUG_PRINT_LINE (...)

Definition at line **386** of file **debug.h**.

#define ZCL_COLOR_CONTROL_DEBUG_WAITSEND ( )

Definition at line **387** of file **debug.h**.

#define ZCL_COLOR_CONTROL_DEBUG_EXECUTE (x)

Definition at line **388** of file **debug.h**.

#define ZCL_ILLUM_MEASUREMENT_DEBUG_ON

Definition at line **402** of file **debug.h**.

#define ZCL_ILLUM_MEASUREMENT_DEBUG_PRINT (...)

Definition at line **403** of file **debug.h**.

#define ZCL_ILLUM_MEASUREMENT_DEBUG_PRINT_LINE (...)

Definition at line **404** of file **debug.h**.

#define ZCL_ILLUM_MEASUREMENT_DEBUG_WAITSEND ( )

Definition at line **405** of file **debug.h**.

#define ZCL_ILLUM_MEASUREMENT_DEBUG_EXECUTE (x)

Definition at line **406** of file **debug.h**.

#define ZCL_ILLUM_LEVEL_SENSING_DEBUG_ON

Definition at line **420** of file **debug.h**.

#define ZCL_ILLUM_LEVEL_SENSING_DEBUG_PRINT (...)

Definition at line **421** of file **debug.h**.

#define ZCL_ILLUM_LEVEL_SENSING_DEBUG_PRINT_LINE (...)

Definition at line **422** of file **debug.h**.

#define ZCL_ILLUM_LEVEL_SENSING_DEBUG_WAITSEND ( )

Definition at line **423** of file **debug.h**.

#define ZCL_ILLUM_LEVEL_SENSING_DEBUG_EXECUTE (x)

Definition at line **424** of file **debug.h**.

#define ZCL_TEMP_MEASUREMENT_DEBUG_ON

Definition at line **438** of file **debug.h**.

#define ZCL_TEMP_MEASUREMENT_DEBUG_PRINT (...)

Definition at line **439** of file **debug.h**.

#define ZCL_TEMP_MEASUREMENT_DEBUG_PRINT_LINE (...)

Definition at line **440** of file **debug.h**.

#define ZCL_TEMP_MEASUREMENT_DEBUG_WAITSEND ( )

Definition at line **441** of file **debug.h**.

#define ZCL_TEMP_MEASUREMENT_DEBUG_EXECUTE (x)

Definition at line **442** of file **debug.h**.

#define ZCL_PRESSURE_MEASUREMENT_DEBUG_ON

Definition at line **456** of file **debug.h**.

#define ZCL_PRESSURE_MEASUREMENT_DEBUG_PRINT (...)

Definition at line **457** of file **debug.h**.

#define ZCL_PRESSURE_MEASUREMENT_DEBUG_PRINT_LINE (...)

Definition at line **458** of file **debug.h**.

#define ZCL_PRESSURE_MEASUREMENT_DEBUG_WAITSEND ( )

Definition at line **459** of file **debug.h**.

#define ZCL_PRESSURE_MEASUREMENT_DEBUG_EXECUTE (x)

Definition at line **460** of file **debug.h**.

#define ZCL_FLOW_MEASUREMENT_DEBUG_ON

Definition at line **474** of file **debug.h**.

#define ZCL_FLOW_MEASUREMENT_DEBUG_PRINT (...)

Definition at line **475** of file **debug.h**.

#define ZCL_FLOW_MEASUREMENT_DEBUG_PRINT_LINE (...)

Definition at line **476** of file **debug.h**.

#define ZCL_FLOW_MEASUREMENT_DEBUG_WAITSEND ( )

Definition at line **477** of file **debug.h**.

#define ZCL_FLOW_MEASUREMENT_DEBUG_EXECUTE (x)

Definition at line **478** of file **debug.h**.

#define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_ON

Definition at line **492** of file **debug.h**.

#define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_PRINT (...)

Definition at line **493** of file **debug.h**.

#define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_PRINT_LINE (...)

Definition at line **494** of file **debug.h**.

#define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_WAITSEND ( )

Definition at line **495** of file **debug.h**.

#define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_EXECUTE (x)

Definition at line **496** of file **debug.h**.

#define ZCL_OCCUPANCY_SENSING_DEBUG_ON

Definition at line **510** of file **debug.h**.

#define ZCL_OCCUPANCY_SENSING_DEBUG_PRINT (...)

Definition at line **511** of file **debug.h**.

#define ZCL_OCCUPANCY_SENSING_DEBUG_PRINT_LINE (...)

Definition at line **512** of file **debug.h**.

#define ZCL_OCCUPANCY_SENSING_DEBUG_WAITSEND ( )

Definition at line **513** of file **debug.h**.

#define ZCL_OCCUPANCY_SENSING_DEBUG_EXECUTE (x)

Definition at line **514** of file **debug.h**.

#define ZCL_IAS_ZONE_DEBUG_ON

Definition at line **528** of file **debug.h**.

#define ZCL_IAS_ZONE_DEBUG_PRINT (...)

Definition at line **529** of file **debug.h**.

#define ZCL_IAS_ZONE_DEBUG_PRINT_LINE (...)

Definition at line **530** of file **debug.h**.

#define ZCL_IAS_ZONE_DEBUG_WAITSEND ( )

Definition at line **531** of file **debug.h**.

#define ZCL_IAS_ZONE_DEBUG_EXECUTE (x)

Definition at line **532** of file **debug.h**.

#define ZCL_IAS_ACE_DEBUG_ON

Definition at line **546** of file **debug.h**.

#define ZCL_IAS_ACE_DEBUG_PRINT (...)

Definition at line **547** of file **debug.h**.

#define ZCL_IAS_ACE_DEBUG_PRINT_LINE (...)

Definition at line **548** of file **debug.h**.

#define ZCL_IAS_ACE_DEBUG_WAITSEND ( )

Definition at line **549** of file **debug.h**.

#define ZCL_IAS_ACE_DEBUG_EXECUTE (x)

Definition at line **550** of file **debug.h**.

#define ZCL_IAS_WD_DEBUG_ON

Definition at line **564** of file **debug.h**.

#define ZCL_IAS_WD_DEBUG_PRINT (...)

Definition at line **565** of file **debug.h**.

#define ZCL_IAS_WD_DEBUG_PRINT_LINE (...)

Definition at line **566** of file **debug.h**.

#define ZCL_IAS_WD_DEBUG_WAITSEND ( )

Definition at line **567** of file **debug.h**.

#define ZCL_IAS_WD_DEBUG_EXECUTE (x)

Definition at line **568** of file **debug.h**.

#define ZCL_DOOR_LOCK_DEBUG_ON

Definition at line **582** of file **debug.h**.

#define ZCL_DOOR_LOCK_DEBUG_PRINT (...)

Definition at line **583** of file **debug.h**.

#define ZCL_DOOR_LOCK_DEBUG_PRINT_LINE (...)

Definition at line **584** of file **debug.h**.

#define ZCL_DOOR_LOCK_DEBUG_WAITSEND ( )

Definition at line **585** of file **debug.h**.

#define ZCL_DOOR_LOCK_DEBUG_EXECUTE (x)

Definition at line **586** of file **debug.h**.

#define ZCL_WINDOW_COVERING_DEBUG_ON

Definition at line **600** of file **debug.h**.

#define ZCL_WINDOW_COVERING_DEBUG_PRINT (...)

Definition at line **601** of file **debug.h**.

#define ZCL_WINDOW_COVERING_DEBUG_PRINT_LINE (...)

Definition at line **602** of file **debug.h**.

#define ZCL_WINDOW_COVERING_DEBUG_WAITSEND ( )

Definition at line **603** of file **debug.h**.

#define ZCL_WINDOW_COVERING_DEBUG_EXECUTE (x)

Definition at line **604** of file **debug.h**.

#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_ON

Definition at line **618** of file **debug.h**.

#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_PRINT (...)

Definition at line **619** of file **debug.h**.

#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_PRINT_LINE (...)

Definition at line **620** of file **debug.h**.

#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_WAITSEND ( )

Definition at line **621** of file **debug.h**.

#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_EXECUTE (x)

Definition at line **622** of file **debug.h**.

#define ZCL_SIMPLE_METERING_DEBUG_ON

Definition at line **636** of file **debug.h**.

#define ZCL_SIMPLE_METERING_DEBUG_PRINT (...)

Definition at line **637** of file **debug.h**.

#define ZCL_SIMPLE_METERING_DEBUG_PRINT_LINE (...)

Definition at line **638** of file **debug.h**.

#define ZCL_SIMPLE_METERING_DEBUG_WAITSEND ( )

Definition at line **639** of file **debug.h**.

#define ZCL_SIMPLE_METERING_DEBUG_EXECUTE (x)

Definition at line **640** of file **debug.h**.

#define ZCL_PRICE_DEBUG_ON

Definition at line **654** of file **debug.h**.

#define ZCL_PRICE_DEBUG_PRINT (...)

Definition at line **655** of file **debug.h**.

#define ZCL_PRICE_DEBUG_PRINT_LINE (...)

Definition at line **656** of file **debug.h**.

#define ZCL_PRICE_DEBUG_WAITSEND ( )

Definition at line **657** of file **debug.h**.

#define ZCL_PRICE_DEBUG_EXECUTE (x)

Definition at line **658** of file **debug.h**.

#define ZCL_MESSAGING_DEBUG_ON

Definition at line **672** of file **debug.h**.

#define ZCL_MESSAGING_DEBUG_PRINT (...)

Definition at line **673** of file **debug.h**.

#define ZCL_MESSAGING_DEBUG_PRINT_LINE (...)

Definition at line **674** of file **debug.h**.

#define ZCL_MESSAGING_DEBUG_WAITSEND ( )

Definition at line **675** of file **debug.h**.

#define ZCL_MESSAGING_DEBUG_EXECUTE (x)

Definition at line **676** of file **debug.h**.

#define ZCL_KEY_ESTABLISHMENT_DEBUG_ON

Definition at line **690** of file **debug.h**.

#define ZCL_KEY_ESTABLISHMENT_DEBUG_PRINT (...)

Definition at line **691** of file **debug.h**.

#define ZCL_KEY_ESTABLISHMENT_DEBUG_PRINT_LINE (...)

Definition at line **692** of file **debug.h**.

#define ZCL_KEY_ESTABLISHMENT_DEBUG_WAITSEND ( )

Definition at line **693** of file **debug.h**.

#define ZCL_KEY_ESTABLISHMENT_DEBUG_EXECUTE (x)

Definition at line **694** of file **debug.h**.

#define ZCL_GENERIC_TUNNEL_DEBUG_ON

Definition at line **708** of file **debug.h**.

#define ZCL_GENERIC_TUNNEL_DEBUG_PRINT (...)

Definition at line **709** of file **debug.h**.

#define ZCL_GENERIC_TUNNEL_DEBUG_PRINT_LINE (...)

Definition at line **710** of file **debug.h**.

#define ZCL_GENERIC_TUNNEL_DEBUG_WAITSEND ( )

Definition at line **711** of file **debug.h**.

#define ZCL_GENERIC_TUNNEL_DEBUG_EXECUTE (x)

Definition at line **712** of file **debug.h**.

#define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_ON

Definition at line **726** of file **debug.h**.

#define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_PRINT (...)

Definition at line **727** of file **debug.h**.

#define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_PRINT_LINE (...)

Definition at line **728** of file **debug.h**.

#define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_WAITSEND ( )

Definition at line **729** of file **debug.h**.

#define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_EXECUTE (x)

Definition at line **730** of file **debug.h**.

#define DEBUG_PRINT_TEST

Definition at line **734** of file **debug.h**.

#define LOG_LEVELS

Definition at line **776** of file **debug.h**.

---

demand-response-load-control.h File Reference

```
#include "load-control-event-table.h"
#include "stack/include/ember-types.h"
```

Go to the source code of this file.

Defines

```
#define EVENT_OPT_IN_DEFAULT
```

Functions

```
void afReportEventStatusHandler (int32u eventId, int8u eventCode, int32u startTime, int8u
    criticalityLevelApplied, int16s coolingTempSetPointApplied, int16s heatingTempSetPointApplied,
    int8s avgLoadAdjPercent, int8u dutyCycle, int8u eventControl, int8u messageLength, int8u
    *message, EmberSignatureData *signature)
```

Define Documentation

#define EVENT_OPT_IN_DEFAULT

Definition at line 13 of file demand-response-load-control.h.

Function Documentation

```
void afReportEventStatusHandler ( int32u                eventId,
                                  int8u                 eventCode,
                                  int32u                startTime,
                                  int8u                 criticalityLevelApplied,
                                  int16s                coolingTempSetPointApplied,
                                  int16s                heatingTempSetPointApplied,
                                  int8s                 avgLoadAdjPercent,
                                  int8u                 dutyCycle,
                                  int8u                 eventControl,
                                  int8u                 messageLength,
                                  int8u *               message,
                                  EmberSignatureData *  signature
                                )
```

drlc-callback.h File Reference

Go to the source code of this file.

Functions

boolean **emberAfPluginDrlcEventActionCallback** (**EmberAfLoadControlEvent** *loadControlEvent, **EmberAfAmiEventStatus** eventStatus, int8u sequenceNumber)

drlc-server.h File Reference

#include "../../include/af-types.h"

Go to the source code of this file.

Defines

#define EMBER_AF_PLUGIN_DRLC_SERVER_SCHEDULED_EVENT_TABLE_SIZE

Functions

void emAfClearScheduledLoadControlEvents (void)
EmberAfLoadControlEvent * emAfGetScheduledLoadControlEvent (int8u index)
void emAfPluginDrlcServerPrintInfo (void)

Define Documentation

#define EMBER_AF_PLUGIN_DRLC_SERVER_SCHEDULED_EVENT_TABLE_SIZE

Definition at line 34 of file drlc-server.h.

Function Documentation

void emAfClearScheduledLoadControlEvents (void)

EmberAfLoadControlEvent* emAfGetScheduledLoadControlEvent (int8u index)

void emAfPluginDrlcServerPrintInfo (void)

---

enums.h File Reference

Go to the source code of this file.

Enumerations

```
enum EmberAfCommissioningStartupControl {
        EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_NO_ACTION,
        EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_FORM_NETWORK,
        EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_REJOIN_NETWORK,
        EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_START_FROM_SCRATCH
      }
enum EmberAfMeteringBlockEnumerations {
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_NO_BLOCKS_IN_USE,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK1,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK2,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK3,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK4,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK5,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK6,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK7,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK8,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK9,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK10,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK11,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK12,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK13,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK14,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK15,
        EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK16
      }
```

```
enum EmberAfMeteringSupplyStatus {
    EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_OFF,
    EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_OFF_ARMED,
    EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_ON
  }
enum EmberAfMeteringTemperatureUnitOfMeasure {
    EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_KELVIN,
    EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_CELSIUS,
    EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_FAHRENHEIT,
    EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_KELVIN_BCD,
    EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_CELSIUS_BCD,
    EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_FAHRENHEIT_BCD
  }
enum EmberAfMeteringAlarmCode {
    EMBER_ZCL_METERING_ALARM_CODE_CHECK_METER,
    EMBER_ZCL_METERING_ALARM_CODE_LOW_BATTERY,
    EMBER_ZCL_METERING_ALARM_CODE_TAMPER_DETECT,
    EMBER_ZCL_METERING_ALARM_CODE_POWER_FAILURE_PIPE_EMPTY_TEMPERATURE_SENSOR,
    EMBER_ZCL_METERING_ALARM_CODE_POWER_QUALITY_LOW_PRESSURE_BURST_DETECT,
    EMBER_ZCL_METERING_ALARM_CODE_LEAK_DETECT,
    EMBER_ZCL_METERING_ALARM_CODE_SERVICE_DISCONNECT,
    EMBER_ZCL_METERING_ALARM_CODE_REVERSE_FLOW_FLOW_SENSOR,
    EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L1,
    EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L1,
    EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L2,
    EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L2,
    EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L3,
    EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L3,
    EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L1,
    EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L2,
    EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L3,
    EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L1,
    EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L1,
    EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L2,
    EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L2,
    EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L3,
    EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L3,
    EMBER_ZCL_METERING_ALARM_CODE_GROUND_FAULT,
    EMBER_ZCL_METERING_ALARM_CODE_ELECTRIC_TAMPER_DETECT,
    EMBER_ZCL_METERING_ALARM_CODE_BURST_DETECT,
```

```
    EMBER_ZCL_METERING_ALARM_CODE_PRESSURE_TOO_LOW,
    EMBER_ZCL_METERING_ALARM_CODE_PRESSURE_TOO_HIGH,
    EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_COMMUNICATION_ERROR,
    EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_MEASUREMENT_FAULT,
    EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_REVERSE_FLOW,
    EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_AIR_DETECT,
    EMBER_ZCL_METERING_ALARM_CODE_PIPE_EMPTY,
    EMBER_ZCL_METERING_ALARM_CODE_INLET_TEMPERATURE_SENSOR_FAULT,
    EMBER_ZCL_METERING_ALARM_CODE_OUTLET_TEMPERATURE_SENSOR_FAULT
  }
enum EmberAfMessagingControlTransmission {
    EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_NORMAL,
    EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_NORMAL_AND_ANONYMOUS,
    EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_ANONYMOUS,
    EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_RESERVED
  }
enum EmberAfMessagingControlImportance {
    EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_LOW,
    EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_MEDIUM,
    EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_HIGH,
    EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_CRITICAL
  }
enum EmberAfMessagingControlConfirmation {
    EMBER_ZCL_MESSAGING_CONTROL_CONFIRMATION_NOT_REQUIRED,
    EMBER_ZCL_MESSAGING_CONTROL_CONFIRMATION_REQUIRED
  }
enum EmberAfOperatingMode {
    EMBER_ZCL_OPERATING_MODE_NORMAL,
    EMBER_ZCL_OPERATING_MODE_CONFIGURE
  }
```

```
enum EmberAfStatus {
        EMBER_ZCL_STATUS_SUCCESS,
        EMBER_ZCL_STATUS_FAILURE,
        EMBER_ZCL_STATUS_REQUEST_DENIED,
        EMBER_ZCL_STATUS_MULTIPLE_REQUEST_NOT_ALLOWED,
        EMBER_ZCL_STATUS_INDICATION_REDIRECTION_TO_AP,
        EMBER_ZCL_STATUS_PREFERENCE_DENIED,
        EMBER_ZCL_STATUS_PREFERENCE_IGNORED,
        EMBER_ZCL_STATUS_NOT_AUTHORIZED,
        EMBER_ZCL_STATUS_RESERVED_FIELD_NOT_ZERO,
        EMBER_ZCL_STATUS_MALFORMED_COMMAND,
        EMBER_ZCL_STATUS_UNSUP_CLUSTER_COMMAND,
        EMBER_ZCL_STATUS_UNSUP_GENERAL_COMMAND,
        EMBER_ZCL_STATUS_UNSUP_MANUF_CLUSTER_COMMAND,
        EMBER_ZCL_STATUS_UNSUP_MANUF_GENERAL_COMMAND,
        EMBER_ZCL_STATUS_INVALID_FIELD,
        EMBER_ZCL_STATUS_UNSUPPORTED_ATTRIBUTE,
        EMBER_ZCL_STATUS_INVALID_VALUE,
        EMBER_ZCL_STATUS_READ_ONLY,
        EMBER_ZCL_STATUS_INSUFFICIENT_SPACE,
        EMBER_ZCL_STATUS_DUPLICATE_EXISTS,
        EMBER_ZCL_STATUS_NOT_FOUND,
        EMBER_ZCL_STATUS_UNREPORTABLE_ATTRIBUTE,
        EMBER_ZCL_STATUS_INVALID_DATA_TYPE,
        EMBER_ZCL_STATUS_INVALID_SELECTOR,
        EMBER_ZCL_STATUS_WRITE_ONLY,
        EMBER_ZCL_STATUS_INCONSISTENT_STARTUP_STATE,
        EMBER_ZCL_STATUS_DEFINED_OUT_OF_BAND,
        EMBER_ZCL_STATUS_ABORT,
        EMBER_ZCL_STATUS_INVALID_IMAGE,
        EMBER_ZCL_STATUS_WAIT_FOR_DATA,
        EMBER_ZCL_STATUS_NO_IMAGE_AVAILABLE,
        EMBER_ZCL_STATUS_REQUIRE_MORE_IMAGE,
        EMBER_ZCL_STATUS_HARDWARE_FAILURE,
        EMBER_ZCL_STATUS_SOFTWARE_FAILURE,
        EMBER_ZCL_STATUS_CALIBRATION_ERROR
      }
```

```
enum EmberAfPowerSource {
    EMBER_ZCL_POWER_SOURCE_UNKNOWN,
    EMBER_ZCL_POWER_SOURCE_SINGLE_PHASE_MAINS,
    EMBER_ZCL_POWER_SOURCE_THREE_PHASE_MAINS,
    EMBER_ZCL_POWER_SOURCE_BATTERY,
    EMBER_ZCL_POWER_SOURCE_DC_SOURCE,
    EMBER_ZCL_POWER_SOURCE_EMERGENCY_MAINS_CONSTANT_POWER,
    EMBER_ZCL_POWER_SOURCE_EMERGENCY_MAINS_TRANSFER_SWITCH,
    EMBER_ZCL_POWER_SOURCE_BATTERY_BACKUP
}
enum EmberAfPhysicalEnvironment {
    EMBER_ZCL_PHYSICAL_ENVIRONMENT_UNSPECIFIED,
    EMBER_ZCL_PHYSICAL_ENVIRONMENT_FIRST_PROFILE_SPECIFIED_VALUE,
    EMBER_ZCL_PHYSICAL_ENVIRONMENT_LAST_PROFILE_SPECIFIED_VALUE,
    EMBER_ZCL_PHYSICAL_ENVIRONMENT_UNKNOWN
}
enum EmberAfBatterySize {
    EMBER_ZCL_BATTERY_SIZE_NO_BATTERY,
    EMBER_ZCL_BATTERY_SIZE_BUILT_IN,
    EMBER_ZCL_BATTERY_SIZE_OTHER,
    EMBER_ZCL_BATTERY_SIZE_AA,
    EMBER_ZCL_BATTERY_SIZE_AAA,
    EMBER_ZCL_BATTERY_SIZE_C,
    EMBER_ZCL_BATTERY_SIZE_D,
    EMBER_ZCL_BATTERY_SIZE_UNKNOWN
}
enum EmberAfSwitchType {
    EMBER_ZCL_SWITCH_TYPE_TOGGLE,
    EMBER_ZCL_SWITCH_TYPE_MOMENTARY
}
enum EmberAfSwitchActions {
    EMBER_ZCL_SWITCH_ACTIONS_ON,
    EMBER_ZCL_SWITCH_ACTIONS_OFF,
    EMBER_ZCL_SWITCH_ACTIONS_TOGGLE
}
enum EmberAfMoveMode {
    EMBER_ZCL_MOVE_MODE_UP,
    EMBER_ZCL_MOVE_MODE_DOWN
}
```

```
enum EmberAfStepMode {
    EMBER_ZCL_STEP_MODE_UP,
    EMBER_ZCL_STEP_MODE_DOWN
}
```

```
enum EmberAfLocationMethod {
    EMBER_ZCL_LOCATION_METHOD_LATERATION,
    EMBER_ZCL_LOCATION_METHOD_SIGNPOSTING,
    EMBER_ZCL_LOCATION_METHOD_RF_FINGERPRINTING,
    EMBER_ZCL_LOCATION_METHOD_OUT_OF_BAND
}
```

```
enum EmberAfPumpOperationMode {
    EMBER_ZCL_PUMP_OPERATION_MODE_NORMAL,
    EMBER_ZCL_PUMP_OPERATION_MODE_MINIMUM,
    EMBER_ZCL_PUMP_OPERATION_MODE_MAXIMUM,
    EMBER_ZCL_PUMP_OPERATION_MODE_LOCAL
}
```

```
enum EmberAfPumpControlMode {
    EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_SPEED,
    EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_PRESSURE,
    EMBER_ZCL_PUMP_CONTROL_MODE_PROPORTIONAL_PRESSURE,
    EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_FLOW,
    EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_TEMPERATURE,
    EMBER_ZCL_PUMP_CONTROL_MODE_AUTOMATIC
}
```

```
enum EmberAfThermostatControlSequence {
    EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_ONLY,
    EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_WITH_REHEAT,
    EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_HEATING_ONLY,
    EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_HEATING_WITH_REHEAT,
    EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_AND_HEATING,
    EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_AND_HEATING_WITH_REHEAT
}
```

```
enum EmberAfThermostatSystemMode {
    EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_OFF,
    EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_AUTO,
    EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_COOL,
    EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_HEAT,
    EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_EMERGENCY_HEATING,
    EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_PRECOOLING,
    EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_FAN_ONLY
}
enum EmberAfSetpointAdjustMode {
    EMBER_ZCL_SETPOINT_ADJUST_MODE_HEAT_SETPOINT,
    EMBER_ZCL_SETPOINT_ADJUST_MODE_COOL_SETPOINT,
    EMBER_ZCL_SETPOINT_ADJUST_MODE_HEAT_AND_COOL_SETPOINTS
}
enum EmberAfFanMode {
    EMBER_ZCL_FAN_MODE_OFF,
    EMBER_ZCL_FAN_MODE_LOW,
    EMBER_ZCL_FAN_MODE_MEDIUM,
    EMBER_ZCL_FAN_MODE_HIGH,
    EMBER_ZCL_FAN_MODE_ON,
    EMBER_ZCL_FAN_MODE_AUTO,
    EMBER_ZCL_FAN_MODE_SMART
}
enum EmberAfFanModeSequence {
    EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_MED_HIGH,
    EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_HIGH,
    EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_MED_HIGH_AUTO,
    EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_HIGH_AUTO,
    EMBER_ZCL_FAN_MODE_SEQUENCE_ON_AUTO
}
enum EmberAfRelativeHumidityMode {
    EMBER_ZCL_RELATIVE_HUMIDITY_MODE_MEASURE_LOCALLY,
    EMBER_ZCL_RELATIVE_HUMIDITY_MODE_UPDATED_OVER_THE_NETWORK
}
enum EmberAfDehumidifcationLockout {
    EMBER_ZCL_DEHUMIDIFCATION_LOCKOUT_NOT_ALLOWED,
    EMBER_ZCL_DEHUMIDIFCATION_LOCKOUT_ALLOWED
}
```

```
enum EmberAfRelativeHumidityDisplay {
    EMBER_ZCL_RELATIVE_HUMIDITY_DISPLAY_NOT_DISPLAYED,
    EMBER_ZCL_RELATIVE_HUMIDITY_DISPLAY_DISPLAYED
}
enum EmberAfTemperatureDisplayMode {
    EMBER_ZCL_TEMPERATURE_DISPLAY_MODE_CELSIUS,
    EMBER_ZCL_TEMPERATURE_DISPLAY_MODE_FARENHEIT
}
enum EmberAfKeypadLockout {
    EMBER_ZCL_KEYPAD_LOCKOUT_NO_LOCKOUT,
    EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_ONE_LOCKOUT,
    EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_TWO_LOCKOUT,
    EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_THREE_LOCKOUT,
    EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_FOUR_LOCKOUT,
    EMBER_ZCL_KEYPAD_LOCKOUT_LEVELFIVE_LOCKOUT
}
enum EmberAfHueDirection {
    EMBER_ZCL_HUE_DIRECTION_SHORTEST_DISTANCE,
    EMBER_ZCL_HUE_DIRECTION_LONGEST_DISTANCE,
    EMBER_ZCL_HUE_DIRECTION_UP,
    EMBER_ZCL_HUE_DIRECTION_DOWN
}
enum EmberAfHueMoveMode {
    EMBER_ZCL_HUE_MOVE_MODE_STOP,
    EMBER_ZCL_HUE_MOVE_MODE_UP,
    EMBER_ZCL_HUE_MOVE_MODE_DOWN
}
enum EmberAfHueStepMode {
    EMBER_ZCL_HUE_STEP_MODE_UP,
    EMBER_ZCL_HUE_STEP_MODE_DOWN
}
enum EmberAfSaturationMoveMode {
    EMBER_ZCL_SATURATION_MOVE_MODE_STOP,
    EMBER_ZCL_SATURATION_MOVE_MODE_UP,
    EMBER_ZCL_SATURATION_MOVE_MODE_DOWN
}
enum EmberAfSaturationStepMode {
    EMBER_ZCL_SATURATION_STEP_MODE_UP,
    EMBER_ZCL_SATURATION_STEP_MODE_DOWN
}
```

```
enum EmberAfColorMode {
        EMBER_ZCL_COLOR_MODE_CURRENT_HUE_AND_CURRENT_SATURATION,
        EMBER_ZCL_COLOR_MODE_CURRENT_X_AND_CURRENT_Y,
        EMBER_ZCL_COLOR_MODE_COLOR_TEMPERATURE
      }
enum EmberAfMeasurementLightSensorType {
        EMBER_ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_PHOTODIODE,
        EMBER_ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_CMOS
      }
enum EmberAfLevelStatus {
        EMBER_ZCL_LEVEL_STATUS_ON_TARGET,
        EMBER_ZCL_LEVEL_STATUS_BELOW_TARGET,
        EMBER_ZCL_LEVEL_STATUS_ABOVE_TARGET
      }
enum EmberAfSensingLightSensorType {
        EMBER_ZCL_SENSING_LIGHT_SENSOR_TYPE_PHOTODIODE,
        EMBER_ZCL_SENSING_LIGHT_SENSOR_TYPE_CMOS
      }
enum EmberAfOccupancySensorType {
        EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_PIR,
        EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_ULTRASONIC,
        EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_PIR_AND_ULTRASONIC
      }
enum EmberAfIasZoneState {
        EMBER_ZCL_IAS_ZONE_STATE_NOT_ENROLLED,
        EMBER_ZCL_IAS_ZONE_STATE_ENROLLED
      }
enum EmberAfIasZoneType {
        EMBER_ZCL_IAS_ZONE_TYPE_STANDARD_CIE,
        EMBER_ZCL_IAS_ZONE_TYPE_MOTION_SENSOR,
        EMBER_ZCL_IAS_ZONE_TYPE_CONTACT_SWITCH,
        EMBER_ZCL_IAS_ZONE_TYPE_FIRE_SENSOR,
        EMBER_ZCL_IAS_ZONE_TYPE_WATER_SENSOR,
        EMBER_ZCL_IAS_ZONE_TYPE_GAS_SENSOR,
        EMBER_ZCL_IAS_ZONE_TYPE_PERSONAL_EMERGENCY_DEVICE,
        EMBER_ZCL_IAS_ZONE_TYPE_VIBRATION_MOVEMENT_SENSOR,
        EMBER_ZCL_IAS_ZONE_TYPE_REMOTE_CONTROL,
        EMBER_ZCL_IAS_ZONE_TYPE_KEY_FOB,
        EMBER_ZCL_IAS_ZONE_TYPE_KEYPAD,
        EMBER_ZCL_IAS_ZONE_TYPE_STANDARD_WARNING_DEVICE
```

```
}
enum EmberAfIasEnrollResponseCode {
  EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_SUCCESS,
  EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_NOT_SUPPORTED,
  EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_NO_ENROLL_PERMIT,
  EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_TOO_MANY_ZONES
}
enum EmberAfIasAceArmMode {
  EMBER_ZCL_IAS_ACE_ARM_MODE_DISARM,
  EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_DAY_HOME_ZONES_ONLY,
  EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_NIGHT_SLEEP_ZONES_ONLY,
  EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_ALL_ZONES
}
enum EmberAfIasAceArmNotification {
  EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ALL_ZONES_DISARMED,
  EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ONLY_DAY_HOME_ZONES_ARMED,
  EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ONLY_NIGHT_SLEEP_ZONES_ARMED,
  EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ALL_ZONES_ARMED
}
enum EmberAfWarningMode {
  EMBER_ZCL_WARNING_MODE_STOP,
  EMBER_ZCL_WARNING_MODE_BURGLAR,
  EMBER_ZCL_WARNING_MODE_FIRE,
  EMBER_ZCL_WARNING_MODE_EMERGENCY
}
enum EmberAfWarningStobe {
  EMBER_ZCL_WARNING_STOBE_NO_STROBE,
  EMBER_ZCL_WARNING_STOBE_USE_STROBE
}
enum EmberAfSquawkMode {
  EMBER_ZCL_SQUAWK_MODE_SYSTEM_IS_ARMED,
  EMBER_ZCL_SQUAWK_MODE_SYSTEM_IS_DISARMED
}
enum EmberAfSquawkStobe {
  EMBER_ZCL_SQUAWK_STOBE_NO_STROBE,
  EMBER_ZCL_SQUAWK_STOBE_USE_STROBE
}
```

```
enum EmberAfSquawkLevel {
        EMBER_ZCL_SQUAWK_LEVEL_LOW_LEVEL,
        EMBER_ZCL_SQUAWK_LEVEL_MEDIUM_LEVEL,
        EMBER_ZCL_SQUAWK_LEVEL_VERY_HIGH_LEVEL
      }
enum EmberAfAmiCriticalityLevel {
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_RESERVED,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_GREEN,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_1,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_2,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_3,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_4,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_5,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_EMERGENCY,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_PLANNED_OUTAGE,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_SERVICE_DISCONNECT,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED1,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED2,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED3,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED4,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED5,
        EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED6
      }
enum EmberAfAmiEventStatus {
        EMBER_ZCL_AMI_EVENT_STATUS_LOAD_CONTROL_EVENT_COMMAND_RX,
        EMBER_ZCL_AMI_EVENT_STATUS_EVENT_STARTED,
        EMBER_ZCL_AMI_EVENT_STATUS_EVENT_COMPLETED,
        EMBER_ZCL_AMI_EVENT_STATUS_USER_HAS_CHOOSE_TO_OPT_OUT,
        EMBER_ZCL_AMI_EVENT_STATUS_USER_HAS_CHOOSE_TO_OPT_IN,
        EMBER_ZCL_AMI_EVENT_STATUS_THE_EVENT_HAS_BEEN_CANCELED,
        EMBER_ZCL_AMI_EVENT_STATUS_THE_EVENT_HAS_BEEN_SUPERSEDED,
        EMBER_ZCL_AMI_EVENT_STATUS_EVENT_PARTIALLY_COMPLETED_WITH_USER_OPT_OUT,
        EMBER_ZCL_AMI_EVENT_STATUS_EVENT_PARTIALLY_COMPLETED_DUE_TO_USER_OPT_IN,
        EMBER_ZCL_AMI_EVENT_STATUS_EVENT_COMPLETED_NO_USER_PARTICIPATION_PREVIOUS_OPT_OUT,
        EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_COMMAND,
        EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_COMMAND_INVALID_EFFECTIVE_TIME,
        EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_EVENT_EXPIRED,
        EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_UNDEFINED_EVENT,
        EMBER_ZCL_AMI_EVENT_STATUS_LOAD_CONTROL_EVENT_COMMAND_REJECTED
      }
```

```
enum EmberAfAmiUnitOfMeasure {
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_KILO_WATT_HOURS,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_METER_PER_HOUR,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_FEET_PER_HOUR,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_CENTUM_CUBIC_FEET_PER_HOUR,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_US_GALLONS_PER_HOUR,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_IMPERIAL_GALLONS_PER_HOUR,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_BT_US_OR_BTU_PER_HOUR,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_LITERS_OR_LITERS_PER_HOUR,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_GAUGE,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_ABSOLUTE,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_MCF_OR_MCF_PER_SECOND,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_UNITLESS,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_MJ_OR_MJ_PER_SECOND,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_KILO_WATT_HOURS_BCD,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_METER_PER_HOUR_BCD,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_FEET_PER_HOUR_BCD,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_CENTUM_CUBIC_FEET_PER_HOUR_BCD,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_US_GALLONS_PER_HOUR_BCD,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_IMPERIAL_GALLONS_PER_HOUR_BCD,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_BT_US_OR_BTU_PER_HOUR_BCD,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_LITERS_OR_LITERS_PER_HOUR_BCD,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_GUAGE_BCD,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_ABSOLUTE_BCD,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_MCF_OR_MCF_PER_SECOND_BCD,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_UNITLESS_BCD,
    EMBER_ZCL_AMI_UNIT_OF_MEASURE_MJ_OR_MJ_PER_SECOND_BCD
}
```

```
enum EmberAfAmiIntervalPeriod {
    EMBER_ZCL_AMI_INTERVAL_PERIOD_DAILY,
    EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES60,
    EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES30,
    EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES15,
    EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES10,
    EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES7P5,
    EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES5,
    EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES2P5
}
```

```
enum EmberAfAmiIntervalChannel {
        EMBER_ZCL_AMI_INTERVAL_CHANNEL_CONSUMPTION_DELIVERED,
        EMBER_ZCL_AMI_INTERVAL_CHANNEL_CONSUMPTION_RECEIVED
      }
enum EmberAfAmiRegistrationState {
        EMBER_ZCL_AMI_REGISTRATION_STATE_UNREGISTERED,
        EMBER_ZCL_AMI_REGISTRATION_STATE_JOINING_NETWORK,
        EMBER_ZCL_AMI_REGISTRATION_STATE_JOINED_NETWORK,
        EMBER_ZCL_AMI_REGISTRATION_STATE_SUBMITTED_REGISTRATION_REQUEST,
        EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTRATION_REJECTED,
        EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTERED,
        EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTERATION_NOT_POSSIBLE
      }
enum EmberAfAnonymousDataState {
        EMBER_ZCL_ANONYMOUS_DATA_STATE_NO_SOURCE_FOUND,
        EMBER_ZCL_ANONYMOUS_DATA_STATE_SOURCE_FOUND
      }
enum EmberAfPriceTier {
        EMBER_ZCL_PRICE_TIER_NO_TIER_RELATED,
        EMBER_ZCL_PRICE_TIER_TIER1_PRICE_LABEL,
        EMBER_ZCL_PRICE_TIER_TIER2_PRICE_LABEL,
        EMBER_ZCL_PRICE_TIER_TIER3_PRICE_LABEL,
        EMBER_ZCL_PRICE_TIER_TIER4_PRICE_LABEL,
        EMBER_ZCL_PRICE_TIER_TIER5_PRICE_LABEL,
        EMBER_ZCL_PRICE_TIER_TIER6_PRICE_LABEL,
        EMBER_ZCL_PRICE_TIER_TIER7_PRICE_LABEL,
        EMBER_ZCL_PRICE_TIER_TIER8_PRICE_LABEL,
        EMBER_ZCL_PRICE_TIER_TIER9_PRICE_LABEL,
        EMBER_ZCL_PRICE_TIER_TIER10_PRICE_LABEL,
        EMBER_ZCL_PRICE_TIER_TIER11_PRICE_LABEL,
        EMBER_ZCL_PRICE_TIER_TIER12_PRICE_LABEL,
        EMBER_ZCL_PRICE_TIER_TIER13_PRICE_LABEL,
        EMBER_ZCL_PRICE_TIER_TIER14_PRICE_LABEL,
        EMBER_ZCL_PRICE_TIER_TIER15_PRICE_LABEL
      }
enum EmberAfPriceControlAcknowledgement {
        EMBER_ZCL_PRICE_CONTROL_ACKNOWLEDGEMENT_NOT_REQUIRED,
        EMBER_ZCL_PRICE_CONTROL_ACKNOWLEDGEMENT_REQUIRED
      }
```

```
enum EmberAfRegisterTier {
    EMBER_ZCL_REGISTER_TIER_NO_TIER_RELATED,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER1_SUMMATION_DELIVERED_ATTRIBUTE,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER2_SUMMATION_DELIVERED_ATTRIBUTE,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER3_SUMMATION_DELIVERED_ATTRIBUTE,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER4_SUMMATION_DELIVERED_ATTRIBUTE,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER5_SUMMATION_DELIVERED_ATTRIBUTE,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER6_SUMMATION_DELIVERED_ATTRIBUTE,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER7_SUMMATION_DELIVERED_ATTRIBUTE,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER8_SUMMATION_DELIVERED_ATTRIBUTE,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER9_SUMMATION_DELIVERED_ATTRIBUTE,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER10_SUMMATION_DELIVERED_ATTRIBUTE,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER11_SUMMATION_DELIVERED_ATTRIBUTE,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER12_SUMMATION_DELIVERED_ATTRIBUTE,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER13_SUMMATION_DELIVERED_ATTRIBUTE,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER14_SUMMATION_DELIVERED_ATTRIBUTE,
    EMBER_ZCL_REGISTER_TIER_CURRENT_TIER15_SUMMATION_DELIVERED_ATTRIBUTE
}
enum EmberAfAlternateCostUnit { EMBER_ZCL_ALTERNATE_COST_UNIT_KG_OF_CO2_PER_UNIT_OF_MEASURE }
enum EmberAfAmiKeyEstablishmentStatus {
    EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_SUCCESS,
    EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_UNKNOWN_ISSUER,
    EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_BAD_KEY_CONFIRM,
    EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_BAD_MESSAGE,
    EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_NO_RESOURCES,
    EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_UNSUPPORTED_SUITE
}
enum EmberAfAmiGetProfileStatus {
    EMBER_ZCL_AMI_GET_PROFILE_STATUS_SUCCESS,
    EMBER_ZCL_AMI_GET_PROFILE_STATUS_UNDEFINED_INTERVAL_CHANNEL_REQUESTED,
    EMBER_ZCL_AMI_GET_PROFILE_STATUS_INTERVAL_CHANNEL_NOT_SUPPORTED,
    EMBER_ZCL_AMI_GET_PROFILE_STATUS_INVALID_END_TIME,
    EMBER_ZCL_AMI_GET_PROFILE_STATUS_MORE_PERIODS_REQUESTED_THAN_CAN_BE_RETURNED,
    EMBER_ZCL_AMI_GET_PROFILE_STATUS_NO_INTERVALS_AVAILABLE_FOR_THE_REQUESTED_TIME
}
enum EmberAfAttributeWritePermission {
    EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_DENY_WRITE,
    EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_ALLOW_WRITE_NORMAL,
    EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_ALLOW_WRITE_OF_READ_ONLY
}
```

```
enum EmberAfMeterDeviceType {
    EMBER_ZCL_METER_DEVICE_TYPE_ELECTRIC_METER,
    EMBER_ZCL_METER_DEVICE_TYPE_GAS_METER,
    EMBER_ZCL_METER_DEVICE_TYPE_WATER_METER,
    EMBER_ZCL_METER_DEVICE_TYPE_THERMAL_METER,
    EMBER_ZCL_METER_DEVICE_TYPE_PRESSURE_METER,
    EMBER_ZCL_METER_DEVICE_TYPE_HEAT_METER,
    EMBER_ZCL_METER_DEVICE_TYPE_COOLING_METER,
    EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_GAS_METER,
    EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_WATER_METER,
    EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_THERMAL_METER,
    EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_PRESSURE_METER,
    EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_HEAT_METER,
    EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_COOLING_METER,
    EMBER_ZCL_METER_DEVICE_TYPE_UNDEFINED_MIRROR_METER
}
enum EmberAfTunnelingProtocolId {
    EMBER_ZCL_TUNNELING_PROTOCOL_ID_DLMS_COSEM,
    EMBER_ZCL_TUNNELING_PROTOCOL_ID_IEC_61107,
    EMBER_ZCL_TUNNELING_PROTOCOL_ID_ANSI_C12,
    EMBER_ZCL_TUNNELING_PROTOCOL_ID_M_BUS,
    EMBER_ZCL_TUNNELING_PROTOCOL_ID_SML,
    EMBER_ZCL_TUNNELING_PROTOCOL_ID_CLIMATE_TALK
}
enum EmberAfTunnelingTransferDataStatus {
    EMBER_ZCL_TUNNELING_TRANSFER_DATA_STATUS_NO_SUCH_TUNNEL,
    EMBER_ZCL_TUNNELING_TRANSFER_DATA_STATUS_WRONG_DEVICE
}
enum EmberAfTunnelingTunnelStatus {
    EMBER_ZCL_TUNNELING_TUNNEL_STATUS_SUCCESS,
    EMBER_ZCL_TUNNELING_TUNNEL_STATUS_BUSY,
    EMBER_ZCL_TUNNELING_TUNNEL_STATUS_NO_MORE_TUNNEL_IDS,
    EMBER_ZCL_TUNNELING_TUNNEL_STATUS_PROTOCOL_NOT_SUPPORTED,
    EMBER_ZCL_TUNNELING_TUNNEL_STATUS_FLOW_CONTROL_NOT_SUPPORTED
}
```

```
enum EmberAfPrepaymentRecoveryPeriod {
        EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_HOUR,
        EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_DAY,
        EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_WEEK,
        EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_MONTH,
        EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_QUARTER
      }
enum EmberAfPrepaymentOriginatingDevice {
        EMBER_ZCL_PREPAYMENT_ORIGINATING_DEVICE_ENERGY_SERVICE_INTERFACE,
        EMBER_ZCL_PREPAYMENT_ORIGINATING_DEVICE_IN_PREMISES_DISPLAY_DEVICE
      }
enum EmberAfPrepaymentSupplyStatus {
        EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_OFF,
        EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_OFF_ARMED,
        EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_ON
      }
enum EmberAfReportingDirection {
        EMBER_ZCL_REPORTING_DIRECTION_REPORTED,
        EMBER_ZCL_REPORTING_DIRECTION_RECEIVED
      }
enum EmberAfSignatureType {
        EMBER_ZCL_SIGNATURE_TYPE_RESERVED,
        EMBER_ZCL_SIGNATURE_TYPE_ECDSA
      }
```

Enumeration Type Documentation

enum EmberAfCommissioningStartupControl

Enumerator:

EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_NO_ACTION
EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_FORM_NETWORK
EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_REJOIN_NETWORK
EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_START_FROM_SCRATCH

Definition at line **17** of file **enums.h**.

enum EmberAfMeteringBlockEnumerations

Enumerator:

EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_NO_BLOCKS_IN_USE
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK1
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK2
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK3
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK4
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK5
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK6
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK7
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK8
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK9
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK10
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK11
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK12
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK13
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK14
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK15
EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK16

Definition at line **25** of file **enums.h**.

enum EmberAfMeteringSupplyStatus

Enumerator:

EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_OFF
EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_OFF_ARMED
EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_ON

Definition at line **46** of file **enums.h**.

enum EmberAfMeteringTemperatureUnitOfMeasure

Enumerator:

EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_KELVIN

EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_CELSIUS
EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_FAHRENHEIT
EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_KELVIN_BCD
EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_CELSIUS_BCD
EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_FAHRENHEIT_BCD

Definition at line **53** of file **enums.h**.

enum EmberAfMeteringAlarmCode

Enumerator:

EMBER_ZCL_METERING_ALARM_CODE_CHECK_METER
EMBER_ZCL_METERING_ALARM_CODE_LOW_BATTERY
EMBER_ZCL_METERING_ALARM_CODE_TAMPER_DETECT
EMBER_ZCL_METERING_ALARM_CODE_POWER_FAILURE_PIPE_EMPTY_TEMPERATURE_SENSOR
EMBER_ZCL_METERING_ALARM_CODE_POWER_QUALITY_LOW_PRESSURE_BURST_DETECT
EMBER_ZCL_METERING_ALARM_CODE_LEAK_DETECT
EMBER_ZCL_METERING_ALARM_CODE_SERVICE_DISCONNECT
EMBER_ZCL_METERING_ALARM_CODE_REVERSE_FLOW_FLOW_SENSOR
EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L1
EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L1
EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L2
EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L2
EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L3
EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L3
EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L1
EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L2
EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L3
EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L1
EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L1
EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L2
EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L2
EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L3
EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L3
EMBER_ZCL_METERING_ALARM_CODE_GROUND_FAULT
EMBER_ZCL_METERING_ALARM_CODE_ELECTRIC_TAMPER_DETECT
EMBER_ZCL_METERING_ALARM_CODE_BURST_DETECT
EMBER_ZCL_METERING_ALARM_CODE_PRESSURE_TOO_LOW

EMBER_ZCL_METERING_ALARM_CODE_PRESSURE_TOO_HIGH
EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_COMMUNICATION_ERROR
EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_MEASUREMENT_FAULT
EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_REVERSE_FLOW
EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_AIR_DETECT
EMBER_ZCL_METERING_ALARM_CODE_PIPE_EMPTY
EMBER_ZCL_METERING_ALARM_CODE_INLET_TEMPERATURE_SENSOR_FAULT
EMBER_ZCL_METERING_ALARM_CODE_OUTLET_TEMPERATURE_SENSOR_FAULT

Definition at line **63** of file **enums.h**.

enum EmberAfMessagingControlTransmission

Enumerator:
EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_NORMAL
EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_NORMAL_AND_ANONYMOUS
EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_ANONYMOUS
EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_RESERVED

Definition at line **102** of file **enums.h**.

enum EmberAfMessagingControlImportance

Enumerator:
EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_LOW
EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_MEDIUM
EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_HIGH
EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_CRITICAL

Definition at line **110** of file **enums.h**.

enum EmberAfMessagingControlConfirmation

Enumerator:
EMBER_ZCL_MESSAGING_CONTROL_CONFIRMATION_NOT_REQUIRED
EMBER_ZCL_MESSAGING_CONTROL_CONFIRMATION_REQUIRED

Definition at line **118** of file **enums.h**.

enum EmberAfOperatingMode

Enumerator:

EMBER_ZCL_OPERATING_MODE_NORMAL
EMBER_ZCL_OPERATING_MODE_CONFIGURE

Definition at line **124** of file **enums.h**.

enum EmberAfStatus

Enumerator:

EMBER_ZCL_STATUS_SUCCESS
EMBER_ZCL_STATUS_FAILURE
EMBER_ZCL_STATUS_REQUEST_DENIED
EMBER_ZCL_STATUS_MULTIPLE_REQUEST_NOT_ALLOWED
EMBER_ZCL_STATUS_INDICATION_REDIRECTION_TO_AP
EMBER_ZCL_STATUS_PREFERENCE_DENIED
EMBER_ZCL_STATUS_PREFERENCE_IGNORED
EMBER_ZCL_STATUS_NOT_AUTHORIZED
EMBER_ZCL_STATUS_RESERVED_FIELD_NOT_ZERO
EMBER_ZCL_STATUS_MALFORMED_COMMAND
EMBER_ZCL_STATUS_UNSUP_CLUSTER_COMMAND
EMBER_ZCL_STATUS_UNSUP_GENERAL_COMMAND
EMBER_ZCL_STATUS_UNSUP_MANUF_CLUSTER_COMMAND
EMBER_ZCL_STATUS_UNSUP_MANUF_GENERAL_COMMAND
EMBER_ZCL_STATUS_INVALID_FIELD
EMBER_ZCL_STATUS_UNSUPPORTED_ATTRIBUTE
EMBER_ZCL_STATUS_INVALID_VALUE
EMBER_ZCL_STATUS_READ_ONLY
EMBER_ZCL_STATUS_INSUFFICIENT_SPACE
EMBER_ZCL_STATUS_DUPLICATE_EXISTS
EMBER_ZCL_STATUS_NOT_FOUND
EMBER_ZCL_STATUS_UNREPORTABLE_ATTRIBUTE
EMBER_ZCL_STATUS_INVALID_DATA_TYPE

EMBER_ZCL_STATUS_INVALID_SELECTOR
EMBER_ZCL_STATUS_WRITE_ONLY
EMBER_ZCL_STATUS_INCONSISTENT_STARTUP_STATE
EMBER_ZCL_STATUS_DEFINED_OUT_OF_BAND
EMBER_ZCL_STATUS_ABORT
EMBER_ZCL_STATUS_INVALID_IMAGE
EMBER_ZCL_STATUS_WAIT_FOR_DATA
EMBER_ZCL_STATUS_NO_IMAGE_AVAILABLE
EMBER_ZCL_STATUS_REQUIRE_MORE_IMAGE
EMBER_ZCL_STATUS_HARDWARE_FAILURE
EMBER_ZCL_STATUS_SOFTWARE_FAILURE
EMBER_ZCL_STATUS_CALIBRATION_ERROR

Definition at line **130** of file **enums.h**.

enum EmberAfPowerSource

Enumerator:

EMBER_ZCL_POWER_SOURCE_UNKNOWN
EMBER_ZCL_POWER_SOURCE_SINGLE_PHASE_MAINS
EMBER_ZCL_POWER_SOURCE_THREE_PHASE_MAINS
EMBER_ZCL_POWER_SOURCE_BATTERY
EMBER_ZCL_POWER_SOURCE_DC_SOURCE
EMBER_ZCL_POWER_SOURCE_EMERGENCY_MAINS_CONSTANT_POWER
EMBER_ZCL_POWER_SOURCE_EMERGENCY_MAINS_TRANSFER_SWITCH
EMBER_ZCL_POWER_SOURCE_BATTERY_BACKUP

Definition at line **169** of file **enums.h**.

enum EmberAfPhysicalEnvironment

Enumerator:

EMBER_ZCL_PHYSICAL_ENVIRONMENT_UNSPECIFIED
EMBER_ZCL_PHYSICAL_ENVIRONMENT_FIRST_PROFILE_SPECIFIED_VALUE
EMBER_ZCL_PHYSICAL_ENVIRONMENT_LAST_PROFILE_SPECIFIED_VALUE
EMBER_ZCL_PHYSICAL_ENVIRONMENT_UNKNOWN

Definition at line **181** of file **enums.h**.

enum EmberAfBatterySize

Enumerator:

EMBER_ZCL_BATTERY_SIZE_NO_BATTERY
EMBER_ZCL_BATTERY_SIZE_BUILT_IN
EMBER_ZCL_BATTERY_SIZE_OTHER
EMBER_ZCL_BATTERY_SIZE_AA
EMBER_ZCL_BATTERY_SIZE_AAA
EMBER_ZCL_BATTERY_SIZE_C
EMBER_ZCL_BATTERY_SIZE_D
EMBER_ZCL_BATTERY_SIZE_UNKNOWN

Definition at line **189** of file **enums.h**.

enum EmberAfSwitchType

Enumerator:

EMBER_ZCL_SWITCH_TYPE_TOGGLE
EMBER_ZCL_SWITCH_TYPE_MOMENTARY

Definition at line **201** of file **enums.h**.

enum EmberAfSwitchActions

Enumerator:

EMBER_ZCL_SWITCH_ACTIONS_ON
EMBER_ZCL_SWITCH_ACTIONS_OFF
EMBER_ZCL_SWITCH_ACTIONS_TOGGLE

Definition at line **207** of file **enums.h**.

enum EmberAfMoveMode

Enumerator:

EMBER_ZCL_MOVE_MODE_UP
EMBER_ZCL_MOVE_MODE_DOWN

Definition at line **214** of file **enums.h**.

enum EmberAfStepMode

Enumerator:

EMBER_ZCL_STEP_MODE_UP
EMBER_ZCL_STEP_MODE_DOWN

Definition at line **220** of file **enums.h**.

enum EmberAfLocationMethod

Enumerator:

EMBER_ZCL_LOCATION_METHOD_LATERATION
EMBER_ZCL_LOCATION_METHOD_SIGNPOSTING
EMBER_ZCL_LOCATION_METHOD_RF_FINGERPRINTING
EMBER_ZCL_LOCATION_METHOD_OUT_OF_BAND

Definition at line **226** of file **enums.h**.

enum EmberAfPumpOperationMode

Enumerator:

EMBER_ZCL_PUMP_OPERATION_MODE_NORMAL
EMBER_ZCL_PUMP_OPERATION_MODE_MINIMUM
EMBER_ZCL_PUMP_OPERATION_MODE_MAXIMUM
EMBER_ZCL_PUMP_OPERATION_MODE_LOCAL

Definition at line **234** of file **enums.h**.

enum EmberAfPumpControlMode

Enumerator:

EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_SPEED
EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_PRESSURE
EMBER_ZCL_PUMP_CONTROL_MODE_PROPORTIONAL_PRESSURE
EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_FLOW
EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_TEMPERATURE
EMBER_ZCL_PUMP_CONTROL_MODE_AUTOMATIC

Definition at line **242** of file **enums.h**.

enum EmberAfThermostatControlSequence

Enumerator:

EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_ONLY
EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_WITH_REHEAT
EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_HEATING_ONLY
EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_HEATING_WITH_REHEAT
EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_AND_HEATING
EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_AND_HEATING_WITH_REHEAT

Definition at line **252** of file **enums.h**.

enum EmberAfThermostatSystemMode

Enumerator:

EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_OFF
EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_AUTO
EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_COOL
EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_HEAT
EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_EMERGENCY_HEATING
EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_PRECOOLING
EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_FAN_ONLY

Definition at line **262** of file **enums.h**.

enum EmberAfSetpointAdjustMode

Enumerator:

EMBER_ZCL_SETPOINT_ADJUST_MODE_HEAT_SETPOINT
EMBER_ZCL_SETPOINT_ADJUST_MODE_COOL_SETPOINT
EMBER_ZCL_SETPOINT_ADJUST_MODE_HEAT_AND_COOL_SETPOINTS

Definition at line **273** of file **enums.h**.

enum EmberAfFanMode

Enumerator:

EMBER_ZCL_FAN_MODE_OFF
EMBER_ZCL_FAN_MODE_LOW
EMBER_ZCL_FAN_MODE_MEDIUM
EMBER_ZCL_FAN_MODE_HIGH
EMBER_ZCL_FAN_MODE_ON
EMBER_ZCL_FAN_MODE_AUTO
EMBER_ZCL_FAN_MODE_SMART

Definition at line **280** of file **enums.h**.

enum EmberAfFanModeSequence

Enumerator:

EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_MED_HIGH
EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_HIGH
EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_MED_HIGH_AUTO
EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_HIGH_AUTO
EMBER_ZCL_FAN_MODE_SEQUENCE_ON_AUTO

Definition at line **291** of file **enums.h**.

enum EmberAfRelativeHumidityMode

Enumerator:

EMBER_ZCL_RELATIVE_HUMIDITY_MODE_MEASURE_LOCALLY
EMBER_ZCL_RELATIVE_HUMIDITY_MODE_UPDATED_OVER_THE_NETWORK

Definition at line **300** of file **enums.h**.

enum EmberAfDehumidifcationLockout

Enumerator:

EMBER_ZCL_DEHUMIDIFCATION_LOCKOUT_NOT_ALLOWED
EMBER_ZCL_DEHUMIDIFCATION_LOCKOUT_ALLOWED

Definition at line **306** of file **enums.h**.

enum EmberAfRelativeHumidityDisplay

Enumerator:

EMBER_ZCL_RELATIVE_HUMIDITY_DISPLAY_NOT_DISPLAYED
EMBER_ZCL_RELATIVE_HUMIDITY_DISPLAY_DISPLAYED

Definition at line **312** of file **enums.h**.

enum EmberAfTemperatureDisplayMode

Enumerator:

EMBER_ZCL_TEMPERATURE_DISPLAY_MODE_CELSIUS
EMBER_ZCL_TEMPERATURE_DISPLAY_MODE_FARENHEIT

Definition at line **318** of file **enums.h**.

enum EmberAfKeypadLockout

Enumerator:

EMBER_ZCL_KEYPAD_LOCKOUT_NO_LOCKOUT
EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_ONE_LOCKOUT

EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_TWO_LOCKOUT
EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_THREE_LOCKOUT
EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_FOUR_LOCKOUT
EMBER_ZCL_KEYPAD_LOCKOUT_LEVELFIVE_LOCKOUT

Definition at line **324** of file **enums.h**.

enum EmberAfHueDirection

Enumerator:
EMBER_ZCL_HUE_DIRECTION_SHORTEST_DISTANCE
EMBER_ZCL_HUE_DIRECTION_LONGEST_DISTANCE
EMBER_ZCL_HUE_DIRECTION_UP
EMBER_ZCL_HUE_DIRECTION_DOWN

Definition at line **334** of file **enums.h**.

enum EmberAfHueMoveMode

Enumerator:
EMBER_ZCL_HUE_MOVE_MODE_STOP
EMBER_ZCL_HUE_MOVE_MODE_UP
EMBER_ZCL_HUE_MOVE_MODE_DOWN

Definition at line **342** of file **enums.h**.

enum EmberAfHueStepMode

Enumerator:
EMBER_ZCL_HUE_STEP_MODE_UP
EMBER_ZCL_HUE_STEP_MODE_DOWN

Definition at line **349** of file **enums.h**.

enum EmberAfSaturationMoveMode

Enumerator:

EMBER_ZCL_SATURATION_MOVE_MODE_STOP
EMBER_ZCL_SATURATION_MOVE_MODE_UP
EMBER_ZCL_SATURATION_MOVE_MODE_DOWN

Definition at line **355** of file **enums.h**.

enum EmberAfSaturationStepMode

Enumerator:

EMBER_ZCL_SATURATION_STEP_MODE_UP
EMBER_ZCL_SATURATION_STEP_MODE_DOWN

Definition at line **362** of file **enums.h**.

enum EmberAfColorMode

Enumerator:

EMBER_ZCL_COLOR_MODE_CURRENT_HUE_AND_CURRENT_SATURATION
EMBER_ZCL_COLOR_MODE_CURRENT_X_AND_CURRENT_Y
EMBER_ZCL_COLOR_MODE_COLOR_TEMPERATURE

Definition at line **368** of file **enums.h**.

enum EmberAfMeasurementLightSensorType

Enumerator:

EMBER_ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_PHOTODIODE
EMBER_ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_CMOS

Definition at line **375** of file **enums.h**.

enum EmberAfLevelStatus

Enumerator:

EMBER_ZCL_LEVEL_STATUS_ON_TARGET
EMBER_ZCL_LEVEL_STATUS_BELOW_TARGET
EMBER_ZCL_LEVEL_STATUS_ABOVE_TARGET

Definition at line **381** of file **enums.h**.

enum EmberAfSensingLightSensorType

Enumerator:

EMBER_ZCL_SENSING_LIGHT_SENSOR_TYPE_PHOTODIODE
EMBER_ZCL_SENSING_LIGHT_SENSOR_TYPE_CMOS

Definition at line **388** of file **enums.h**.

enum EmberAfOccupancySensorType

Enumerator:

EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_PIR
EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_ULTRASONIC
EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_PIR_AND_ULTRASONIC

Definition at line **394** of file **enums.h**.

enum EmberAfIasZoneState

Enumerator:

EMBER_ZCL_IAS_ZONE_STATE_NOT_ENROLLED
EMBER_ZCL_IAS_ZONE_STATE_ENROLLED

Definition at line **401** of file **enums.h**.

enum EmberAfIasZoneType

Enumerator:

EMBER_ZCL_IAS_ZONE_TYPE_STANDARD_CIE
EMBER_ZCL_IAS_ZONE_TYPE_MOTION_SENSOR
EMBER_ZCL_IAS_ZONE_TYPE_CONTACT_SWITCH
EMBER_ZCL_IAS_ZONE_TYPE_FIRE_SENSOR
EMBER_ZCL_IAS_ZONE_TYPE_WATER_SENSOR
EMBER_ZCL_IAS_ZONE_TYPE_GAS_SENSOR
EMBER_ZCL_IAS_ZONE_TYPE_PERSONAL_EMERGENCY_DEVICE
EMBER_ZCL_IAS_ZONE_TYPE_VIBRATION_MOVEMENT_SENSOR
EMBER_ZCL_IAS_ZONE_TYPE_REMOTE_CONTROL
EMBER_ZCL_IAS_ZONE_TYPE_KEY_FOB
EMBER_ZCL_IAS_ZONE_TYPE_KEYPAD
EMBER_ZCL_IAS_ZONE_TYPE_STANDARD_WARNING_DEVICE

Definition at line **407** of file **enums.h**.

enum EmberAfIasEnrollResponseCode

Enumerator:

EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_SUCCESS
EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_NOT_SUPPORTED
EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_NO_ENROLL_PERMIT
EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_TOO_MANY_ZONES

Definition at line **423** of file **enums.h**.

enum EmberAfIasAceArmMode

Enumerator:

EMBER_ZCL_IAS_ACE_ARM_MODE_DISARM
EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_DAY_HOME_ZONES_ONLY
EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_NIGHT_SLEEP_ZONES_ONLY
EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_ALL_ZONES

Definition at line **431** of file **enums.h**.

enum EmberAfIasAceArmNotification

Enumerator:

EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ALL_ZONES_DISARMED
EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ONLY_DAY_HOME_ZONES_ARMED
EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ONLY_NIGHT_SLEEP_ZONES_ARMED
EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ALL_ZONES_ARMED

Definition at line **439** of file **enums.h**.

enum EmberAfWarningMode

Enumerator:

EMBER_ZCL_WARNING_MODE_STOP
EMBER_ZCL_WARNING_MODE_BURGLAR
EMBER_ZCL_WARNING_MODE_FIRE
EMBER_ZCL_WARNING_MODE_EMERGENCY

Definition at line **447** of file **enums.h**.

enum EmberAfWarningStobe

Enumerator:

EMBER_ZCL_WARNING_STOBE_NO_STROBE
EMBER_ZCL_WARNING_STOBE_USE_STROBE

Definition at line **455** of file **enums.h**.

enum EmberAfSquawkMode

Enumerator:

EMBER_ZCL_SQUAWK_MODE_SYSTEM_IS_ARMED
EMBER_ZCL_SQUAWK_MODE_SYSTEM_IS_DISARMED

Definition at line **461** of file **enums.h**.

enum EmberAfSquawkStobe

Enumerator:

EMBER_ZCL_SQUAWK_STOBE_NO_STROBE
EMBER_ZCL_SQUAWK_STOBE_USE_STROBE

Definition at line **467** of file **enums.h**.

enum EmberAfSquawkLevel

Enumerator:

EMBER_ZCL_SQUAWK_LEVEL_LOW_LEVEL
EMBER_ZCL_SQUAWK_LEVEL_MEDIUM_LEVEL
EMBER_ZCL_SQUAWK_LEVEL_VERY_HIGH_LEVEL

Definition at line **473** of file **enums.h**.

enum EmberAfAmiCriticalityLevel

Enumerator:

EMBER_ZCL_AMI_CRITICALITY_LEVEL_RESERVED
EMBER_ZCL_AMI_CRITICALITY_LEVEL_GREEN
EMBER_ZCL_AMI_CRITICALITY_LEVEL_1
EMBER_ZCL_AMI_CRITICALITY_LEVEL_2
EMBER_ZCL_AMI_CRITICALITY_LEVEL_3
EMBER_ZCL_AMI_CRITICALITY_LEVEL_4
EMBER_ZCL_AMI_CRITICALITY_LEVEL_5
EMBER_ZCL_AMI_CRITICALITY_LEVEL_EMERGENCY
EMBER_ZCL_AMI_CRITICALITY_LEVEL_PLANNED_OUTAGE
EMBER_ZCL_AMI_CRITICALITY_LEVEL_SERVICE_DISCONNECT
EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED1
EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED2
EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED3
EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED4
EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED5
EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED6

Definition at line **480** of file **enums.h**.

enum EmberAfAmiEventStatus

Enumerator:

EMBER_ZCL_AMI_EVENT_STATUS_LOAD_CONTROL_EVENT_COMMAND_RX
EMBER_ZCL_AMI_EVENT_STATUS_EVENT_STARTED
EMBER_ZCL_AMI_EVENT_STATUS_EVENT_COMPLETED
EMBER_ZCL_AMI_EVENT_STATUS_USER_HAS_CHOOSE_TO_OPT_OUT
EMBER_ZCL_AMI_EVENT_STATUS_USER_HAS_CHOOSE_TO_OPT_IN
EMBER_ZCL_AMI_EVENT_STATUS_THE_EVENT_HAS_BEEN_CANCELED
EMBER_ZCL_AMI_EVENT_STATUS_THE_EVENT_HAS_BEEN_SUPERSEDED
EMBER_ZCL_AMI_EVENT_STATUS_EVENT_PARTIALLY_COMPLETED_WITH_USER_OPT_OUT
EMBER_ZCL_AMI_EVENT_STATUS_EVENT_PARTIALLY_COMPLETED_DUE_TO_USER_OPT_IN
EMBER_ZCL_AMI_EVENT_STATUS_EVENT_COMPLETED_NO_USER_PARTICIPATION_PREVIOUS_OPT_OUT
EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_COMMAND
EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_COMMAND_INVALID_EFFECTIVE_TIME
EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_EVENT_EXPIRED
EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_UNDEFINED_EVENT
EMBER_ZCL_AMI_EVENT_STATUS_LOAD_CONTROL_EVENT_COMMAND_REJECTED

Definition at line **500** of file **enums.h**.

enum EmberAfAmiUnitOfMeasure

Enumerator:

EMBER_ZCL_AMI_UNIT_OF_MEASURE_KILO_WATT_HOURS
EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_METER_PER_HOUR
EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_FEET_PER_HOUR
EMBER_ZCL_AMI_UNIT_OF_MEASURE_CENTUM_CUBIC_FEET_PER_HOUR
EMBER_ZCL_AMI_UNIT_OF_MEASURE_US_GALLONS_PER_HOUR
EMBER_ZCL_AMI_UNIT_OF_MEASURE_IMPERIAL_GALLONS_PER_HOUR
EMBER_ZCL_AMI_UNIT_OF_MEASURE_BT_US_OR_BTU_PER_HOUR
EMBER_ZCL_AMI_UNIT_OF_MEASURE_LITERS_OR_LITERS_PER_HOUR
EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_GAUGE
EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_ABSOLUTE
EMBER_ZCL_AMI_UNIT_OF_MEASURE_MCF_OR_MCF_PER_SECOND

EMBER_ZCL_AMI_UNIT_OF_MEASURE_UNITLESS
EMBER_ZCL_AMI_UNIT_OF_MEASURE_MJ_OR_MJ_PER_SECOND
EMBER_ZCL_AMI_UNIT_OF_MEASURE_KILO_WATT_HOURS_BCD
EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_METER_PER_HOUR_BCD
EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_FEET_PER_HOUR_BCD
EMBER_ZCL_AMI_UNIT_OF_MEASURE_CENTUM_CUBIC_FEET_PER_HOUR_BCD
EMBER_ZCL_AMI_UNIT_OF_MEASURE_US_GALLONS_PER_HOUR_BCD
EMBER_ZCL_AMI_UNIT_OF_MEASURE_IMPERIAL_GALLONS_PER_HOUR_BCD
EMBER_ZCL_AMI_UNIT_OF_MEASURE_BT_US_OR_BTU_PER_HOUR_BCD
EMBER_ZCL_AMI_UNIT_OF_MEASURE_LITERS_OR_LITERS_PER_HOUR_BCD
EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_GUAGE_BCD
EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_ABSOLUTE_BCD
EMBER_ZCL_AMI_UNIT_OF_MEASURE_MCF_OR_MCF_PER_SECOND_BCD
EMBER_ZCL_AMI_UNIT_OF_MEASURE_UNITLESS_BCD
EMBER_ZCL_AMI_UNIT_OF_MEASURE_MJ_OR_MJ_PER_SECOND_BCD

Definition at line **519** of file **enums.h**.

enum EmberAfAmiIntervalPeriod

Enumerator:

EMBER_ZCL_AMI_INTERVAL_PERIOD_DAILY
EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES60
EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES30
EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES15
EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES10
EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES7P5
EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES5
EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES2P5

Definition at line **549** of file **enums.h**.

enum EmberAfAmiIntervalChannel

Enumerator:

EMBER_ZCL_AMI_INTERVAL_CHANNEL_CONSUMPTION_DELIVERED
EMBER_ZCL_AMI_INTERVAL_CHANNEL_CONSUMPTION_RECEIVED

Definition at line **561** of file **enums.h**.

enum EmberAfAmiRegistrationState

Enumerator:

EMBER_ZCL_AMI_REGISTRATION_STATE_UNREGISTERED
EMBER_ZCL_AMI_REGISTRATION_STATE_JOINING_NETWORK
EMBER_ZCL_AMI_REGISTRATION_STATE_JOINED_NETWORK
EMBER_ZCL_AMI_REGISTRATION_STATE_SUBMITTED_REGISTRATION_REQUEST
EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTRATION_REJECTED
EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTERED
EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTERATION_NOT_POSSIBLE

Definition at line **567** of file **enums.h**.

enum EmberAfAnonymousDataState

Enumerator:

EMBER_ZCL_ANONYMOUS_DATA_STATE_NO_SOURCE_FOUND
EMBER_ZCL_ANONYMOUS_DATA_STATE_SOURCE_FOUND

Definition at line **578** of file **enums.h**.

enum EmberAfPriceTier

Enumerator:

EMBER_ZCL_PRICE_TIER_NO_TIER_RELATED
EMBER_ZCL_PRICE_TIER_TIER1_PRICE_LABEL
EMBER_ZCL_PRICE_TIER_TIER2_PRICE_LABEL
EMBER_ZCL_PRICE_TIER_TIER3_PRICE_LABEL
EMBER_ZCL_PRICE_TIER_TIER4_PRICE_LABEL
EMBER_ZCL_PRICE_TIER_TIER5_PRICE_LABEL
EMBER_ZCL_PRICE_TIER_TIER6_PRICE_LABEL
EMBER_ZCL_PRICE_TIER_TIER7_PRICE_LABEL
EMBER_ZCL_PRICE_TIER_TIER8_PRICE_LABEL

EMBER_ZCL_PRICE_TIER_TIER9_PRICE_LABEL
EMBER_ZCL_PRICE_TIER_TIER10_PRICE_LABEL
EMBER_ZCL_PRICE_TIER_TIER11_PRICE_LABEL
EMBER_ZCL_PRICE_TIER_TIER12_PRICE_LABEL
EMBER_ZCL_PRICE_TIER_TIER13_PRICE_LABEL
EMBER_ZCL_PRICE_TIER_TIER14_PRICE_LABEL
EMBER_ZCL_PRICE_TIER_TIER15_PRICE_LABEL

Definition at line **584** of file **enums.h**.

enum EmberAfPriceControlAcknowledgement

Enumerator:

EMBER_ZCL_PRICE_CONTROL_ACKNOWLEDGEMENT_NOT_REQUIRED
EMBER_ZCL_PRICE_CONTROL_ACKNOWLEDGEMENT_REQUIRED

Definition at line **604** of file **enums.h**.

enum EmberAfRegisterTier

Enumerator:

EMBER_ZCL_REGISTER_TIER_NO_TIER_RELATED
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER1_SUMMATION_DELIVERED_ATTRIBUTE
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER2_SUMMATION_DELIVERED_ATTRIBUTE
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER3_SUMMATION_DELIVERED_ATTRIBUTE
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER4_SUMMATION_DELIVERED_ATTRIBUTE
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER5_SUMMATION_DELIVERED_ATTRIBUTE
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER6_SUMMATION_DELIVERED_ATTRIBUTE
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER7_SUMMATION_DELIVERED_ATTRIBUTE
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER8_SUMMATION_DELIVERED_ATTRIBUTE
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER9_SUMMATION_DELIVERED_ATTRIBUTE
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER10_SUMMATION_DELIVERED_ATTRIBUTE
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER11_SUMMATION_DELIVERED_ATTRIBUTE
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER12_SUMMATION_DELIVERED_ATTRIBUTE
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER13_SUMMATION_DELIVERED_ATTRIBUTE
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER14_SUMMATION_DELIVERED_ATTRIBUTE
EMBER_ZCL_REGISTER_TIER_CURRENT_TIER15_SUMMATION_DELIVERED_ATTRIBUTE

Definition at line **610** of file **enums.h**.

enum EmberAfAlternateCostUnit

Enumerator:

EMBER_ZCL_ALTERNATE_COST_UNIT_KG_OF_CO2_PER_UNIT_OF_MEASURE

Definition at line **630** of file **enums.h**.

enum EmberAfAmiKeyEstablishmentStatus

Enumerator:

EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_SUCCESS
EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_UNKNOWN_ISSUER
EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_BAD_KEY_CONFIRM
EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_BAD_MESSAGE
EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_NO_RESOURCES
EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_UNSUPPORTED_SUITE

Definition at line **635** of file **enums.h**.

enum EmberAfAmiGetProfileStatus

Enumerator:

EMBER_ZCL_AMI_GET_PROFILE_STATUS_SUCCESS
EMBER_ZCL_AMI_GET_PROFILE_STATUS_UNDEFINED_INTERVAL_CHANNEL_REQUESTED
EMBER_ZCL_AMI_GET_PROFILE_STATUS_INTERVAL_CHANNEL_NOT_SUPPORTED
EMBER_ZCL_AMI_GET_PROFILE_STATUS_INVALID_END_TIME
EMBER_ZCL_AMI_GET_PROFILE_STATUS_MORE_PERIODS_REQUESTED_THAN_CAN_BE_RETURNED
EMBER_ZCL_AMI_GET_PROFILE_STATUS_NO_INTERVALS_AVAILABLE_FOR_THE_REQUESTED_TIME

Definition at line **645** of file **enums.h**.

enum EmberAfAttributeWritePermission

Enumerator:

EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_DENY_WRITE
EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_ALLOW_WRITE_NORMAL
EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_ALLOW_WRITE_OF_READ_ONLY

Definition at line **655** of file **enums.h**.

enum EmberAfMeterDeviceType

Enumerator:

EMBER_ZCL_METER_DEVICE_TYPE_ELECTRIC_METER
EMBER_ZCL_METER_DEVICE_TYPE_GAS_METER
EMBER_ZCL_METER_DEVICE_TYPE_WATER_METER
EMBER_ZCL_METER_DEVICE_TYPE_THERMAL_METER
EMBER_ZCL_METER_DEVICE_TYPE_PRESSURE_METER
EMBER_ZCL_METER_DEVICE_TYPE_HEAT_METER
EMBER_ZCL_METER_DEVICE_TYPE_COOLING_METER
EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_GAS_METER
EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_WATER_METER
EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_THERMAL_METER
EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_PRESSURE_METER
EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_HEAT_METER
EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_COOLING_METER
EMBER_ZCL_METER_DEVICE_TYPE_UNDEFINED_MIRROR_METER

Definition at line **662** of file **enums.h**.

enum EmberAfTunnelingProtocolId

Enumerator:

EMBER_ZCL_TUNNELING_PROTOCOL_ID_DLMS_COSEM
EMBER_ZCL_TUNNELING_PROTOCOL_ID_IEC_61107
EMBER_ZCL_TUNNELING_PROTOCOL_ID_ANSI_C12
EMBER_ZCL_TUNNELING_PROTOCOL_ID_M_BUS
EMBER_ZCL_TUNNELING_PROTOCOL_ID_SML
EMBER_ZCL_TUNNELING_PROTOCOL_ID_CLIMATE_TALK

Definition at line **680** of file **enums.h**.

enum EmberAfTunnelingTransferDataStatus

Enumerator:

EMBER_ZCL_TUNNELING_TRANSFER_DATA_STATUS_NO_SUCH_TUNNEL
EMBER_ZCL_TUNNELING_TRANSFER_DATA_STATUS_WRONG_DEVICE

Definition at line **690** of file **enums.h**.

enum EmberAfTunnelingTunnelStatus

Enumerator:

EMBER_ZCL_TUNNELING_TUNNEL_STATUS_SUCCESS
EMBER_ZCL_TUNNELING_TUNNEL_STATUS_BUSY
EMBER_ZCL_TUNNELING_TUNNEL_STATUS_NO_MORE_TUNNEL_IDS
EMBER_ZCL_TUNNELING_TUNNEL_STATUS_PROTOCOL_NOT_SUPPORTED
EMBER_ZCL_TUNNELING_TUNNEL_STATUS_FLOW_CONTROL_NOT_SUPPORTED

Definition at line **696** of file **enums.h**.

enum EmberAfPrepaymentRecoveryPeriod

Enumerator:

EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_HOUR
EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_DAY
EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_WEEK
EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_MONTH
EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_QUARTER

Definition at line **705** of file **enums.h**.

enum EmberAfPrepaymentOriginatingDevice

Enumerator:

EMBER_ZCL_PREPAYMENT_ORIGINATING_DEVICE_ENERGY_SERVICE_INTERFACE
EMBER_ZCL_PREPAYMENT_ORIGINATING_DEVICE_IN_PREMISES_DISPLAY_DEVICE

Definition at line **714** of file **enums.h**.

enum EmberAfPrepaymentSupplyStatus

Enumerator:
EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_OFF
EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_OFF_ARMED
EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_ON

Definition at line **720** of file **enums.h**.

enum EmberAfReportingDirection

Enumerator:
EMBER_ZCL_REPORTING_DIRECTION_REPORTED
EMBER_ZCL_REPORTING_DIRECTION_RECEIVED

Definition at line **727** of file **enums.h**.

enum EmberAfSignatureType

Enumerator:
EMBER_ZCL_SIGNATURE_TYPE_RESERVED
EMBER_ZCL_SIGNATURE_TYPE_ECDSA

Definition at line **733** of file **enums.h**.

---

generic-tunnel.h File Reference

Go to the source code of this file.

Defines

#define **EMBER_AF_PLUGIN_GENERIC_TUNNEL_PROTOCOL_ADDRESS_SIZE**

Functions

void **emAfPluginGenericTunnelPrintInfo** (void)
int8u * **emAfPluginGenericTunnelGetProtocolAddress** (int8u endpoint)

Define Documentation

#define EMBER_AF_PLUGIN_GENERIC_TUNNEL_PROTOCOL_ADDRESS_SIZE

Definition at line **9** of file **generic-tunnel.h**.

Function Documentation

void emAfPluginGenericTunnelPrintInfo (void)

int8u* emAfPluginGenericTunnelGetProtocolAddress (int8u endpoint)

---

groups-server.h File Reference

Go to the source code of this file.

Functions

void **emAfPluginGroupsServerPrintInfo** (void)

Function Documentation

void emAfPluginGroupsServerPrintInfo (void)

ias-ace-server.h File Reference

Go to the source code of this file.

Functions

void **emAfIasAcePrintInfo** (void)

Function Documentation

void emAfIasAcePrintInfo (void)

ias-wd-server.h File Reference

Go to the source code of this file.

Defines

#define ZCL_WARNING_INFO_MODE_MASK
#define ZCL_WARNING_INFO_STOP
#define ZCL_WARNING_INFO_BURGLAR
#define ZCL_WARNING_INFO_FIRE
#define ZCL_WARNING_INFO_EMERGENCY
#define ZCL_WARNING_INFO_STROBE_MASK
#define ZCL_WARNING_INFO_NO_STROBE
#define ZCL_WARNING_INFO_STROBE
#define ZCL_SQUAWK_INFO_MODE_MASK
#define ZCL_SQUAWK_INFO_ARMED
#define ZCL_SQUAWK_INFO_DISARMED
#define ZCL_SQUAWK_INFO_STROBE_MASK
#define ZCL_SQUAWK_INFO_NO_STROBE
#define ZCL_SQUAWK_INFO_STROBE
#define ZCL_SQUAWK_INFO_LEVEL_MASK
#define ZCL_SQUAWK_INFO_LOW
#define ZCL_SQUAWK_INFO_MEDIUM
#define ZCL_SQUAWK_INFO_HIGH
#define ZCL_SQUAWK_INFO_VERY_HIGH

Define Documentation

#define ZCL_WARNING_INFO_MODE_MASK

Definition at line 11 of file ias-wd-server.h.

#define ZCL_WARNING_INFO_STOP

Definition at line 12 of file ias-wd-server.h.

#define ZCL_WARNING_INFO_BURGLAR

Definition at line **13** of file **ias-wd-server.h**.

#define ZCL_WARNING_INFO_FIRE

Definition at line **14** of file **ias-wd-server.h**.

#define ZCL_WARNING_INFO_EMERGENCY

Definition at line **15** of file **ias-wd-server.h**.

#define ZCL_WARNING_INFO_STROBE_MASK

Definition at line **17** of file **ias-wd-server.h**.

#define ZCL_WARNING_INFO_NO_STROBE

Definition at line **18** of file **ias-wd-server.h**.

#define ZCL_WARNING_INFO_STROBE

Definition at line **19** of file **ias-wd-server.h**.

#define ZCL_SQUAWK_INFO_MODE_MASK

Definition at line **21** of file **ias-wd-server.h**.

#define ZCL_SQUAWK_INFO_ARMED

Definition at line **22** of file **ias-wd-server.h**.

#define ZCL_SQUAWK_INFO_DISARMED

Definition at line **23** of file **ias-wd-server.h**.

#define ZCL_SQUAWK_INFO_STROBE_MASK

Definition at line **25** of file **ias-wd-server.h**.

#define ZCL_SQUAWK_INFO_NO_STROBE

Definition at line **26** of file **ias-wd-server.h**.

#define ZCL_SQUAWK_INFO_STROBE

Definition at line **27** of file **ias-wd-server.h**.

#define ZCL_SQUAWK_INFO_LEVEL_MASK

Definition at line **29** of file **ias-wd-server.h**.

#define ZCL_SQUAWK_INFO_LOW

Definition at line **30** of file **ias-wd-server.h**.

#define ZCL_SQUAWK_INFO_MEDIUM

Definition at line **31** of file **ias-wd-server.h**.

#define ZCL_SQUAWK_INFO_HIGH

Definition at line **32** of file **ias-wd-server.h**.

#define ZCL_SQUAWK_INFO_VERY_HIGH

Definition at line **33** of file **ias-wd-server.h**.

---

ias-zone-table.h File Reference

Go to the source code of this file.

Data Structures

struct **EmberAfIasZoneTableEntry**

Defines

#define **ZONE_TABLE_SIZE**
#define **ZONE_ID_MAP_SECTION_SIZE**
#define **ZONE_TABLE_MAP_SIZE**
#define **ZONE_TABLE_VALUE_DEFAULT_8**
#define **ZONE_TABLE_VALUE_DEFAULT_16**

Functions

void **enroll** (int8u zoneId, int16u zoneType, int8u *ieeeAdd)
void **unenroll** (int8u zoneId)
void **initZoneTable** (void)
void **getZoneIdMap** (int16u *map)
EmberAfIasZoneTableEntry * **getZoneInfo** (int8u zoneId)
EmberAfStatus **getNextZoneId** (int8u *zoneId)
void **printZoneIdMap** (int16u *map)
void **printZoneInfo** (int8u zoneId)
void **printZoneTable** (void)

Define Documentation

#define ZONE_TABLE_SIZE

Definition at line **20** of file **ias-zone-table.h**.

#define ZONE_ID_MAP_SECTION_SIZE

Definition at line **21** of file **ias-zone-table.h**.

#define ZONE_TABLE_MAP_SIZE

Definition at line **22** of file **ias-zone-table.h**.

#define ZONE_TABLE_VALUE_DEFAULT_8

Definition at line **23** of file **ias-zone-table.h**.

#define ZONE_TABLE_VALUE_DEFAULT_16

Definition at line **24** of file **ias-zone-table.h**.

Function Documentation

```
void enroll ( int8u    zoneId,
              int16u   zoneType,
              int8u *  ieeeAdd
            )
```

```
void unenroll ( int8u  zoneId )
```

```
void initZoneTable ( void )
```

```
void getZoneIdMap ( int16u *  map )
```

```
EmberAfIasZoneTableEntry* getZoneInfo ( int8u  zoneId )
```

```
EmberAfStatus getNextZoneId ( int8u *  zoneId )
```

```
void printZoneIdMap ( int16u *  map )
```

void printZoneInfo (int8u zoneId)

void printZoneTable (void)

---

identify.h File Reference

Go to the source code of this file.

Functions

boolean	**emberAfIsDeviceIdentifying** (int8u endpoint)
void	**emAfIdentifyPrintInfo** (void)

Function Documentation

boolean emberAfIsDeviceIdentifying (int8u endpoint)

void emAfIdentifyPrintInfo (void)

interpan.h File Reference

Plugin for receiving InterPAN messages. See message for documentation. More...

Go to the source code of this file.

Defines

#define **MAX_INTER_PAN_MAC_SIZE**
#define **STUB_NWK_SIZE**
#define **STUB_NWK_FRAME_CONTROL**
#define **INTERPAN_APS_UNICAST_SIZE**
#define **INTERPAN_APS_BROADCAST_SIZE**
#define **INTERPAN_APS_MULTICAST_SIZE**
#define **MAX_STUB_APS_SIZE**
#define **MIN_STUB_APS_SIZE**
#define **MAX_INTER_PAN_HEADER_SIZE**
#define **INTERPAN_APS_FRAME_TYPE**
#define **INTERPAN_APS_FRAME_TYPE_MASK**
#define **INTERPAN_APS_FRAME_CONTROL_NO_DELIVERY_MODE**
#define **INTERPAN_APS_FRAME_DELIVERY_MODE_MASK**
#define **INTERPAN_APS_FRAME_SECURITY**
#define **INTERPAN_APS_ENCRYPTION_OVERHEAD**

Functions

boolean	**emAfPluginInterpanProcessMessage** (int8u messageLength, int8u *messageContents)
EmberStatus	**emAfPluginInterpanSendRawMessage** (int8u length, int8u *message)
EmberStatus	**emAfInterpanApsCryptMessage** (boolean encrypt, int8u *apsFrame, int8u *messageLength, int8u apsHeaderLength, EmberEUI64 remoteEui64)

Detailed Description

Plugin for receiving InterPAN messages. See message for documentation.

Definition in file **interpan.h**.

Define Documentation

#define MAX_INTER_PAN_MAC_SIZE

Definition at line **23** of file **interpan.h**.

#define STUB_NWK_SIZE

Definition at line **26** of file **interpan.h**.

#define STUB_NWK_FRAME_CONTROL

Definition at line **27** of file **interpan.h**.

#define INTERPAN_APS_UNICAST_SIZE

Definition at line **33** of file **interpan.h**.

#define INTERPAN_APS_BROADCAST_SIZE

Definition at line **36** of file **interpan.h**.

#define INTERPAN_APS_MULTICAST_SIZE

Definition at line **43** of file **interpan.h**.

#define MAX_STUB_APS_SIZE

Definition at line **45** of file **interpan.h**.

#define MIN_STUB_APS_SIZE

Definition at line **46** of file **interpan.h**.

#define MAX_INTER_PAN_HEADER_SIZE

Definition at line **48** of file **interpan.h**.

#define INTERPAN_APS_FRAME_TYPE

Definition at line **51** of file **interpan.h**.

#define INTERPAN_APS_FRAME_TYPE_MASK

Definition at line **52** of file **interpan.h**.

#define INTERPAN_APS_FRAME_CONTROL_NO_DELIVERY_MODE

Definition at line **55** of file **interpan.h**.

#define INTERPAN_APS_FRAME_DELIVERY_MODE_MASK

Definition at line **57** of file **interpan.h**.

#define INTERPAN_APS_FRAME_SECURITY

Definition at line **58** of file **interpan.h**.

#define INTERPAN_APS_ENCRYPTION_OVERHEAD

Definition at line **61** of file **interpan.h**.

Function Documentation

```
boolean emAfPluginInterpanProcessMessage ( int8u    messageLength,
                                           int8u *  messageContents
                                         )
```

```
EmberStatus emAfPluginInterpanSendRawMessage ( int8u    length,
                                               int8u *  message
                                             )
```

```
EmberStatus emAfInterpanApsCryptMessage ( boolean      encrypt,
                                          int8u *      apsFrame,
                                          int8u *      messageLength,
                                          int8u        apsHeaderLength,
                                          EmberEUI64   remoteEui64
                                        )
```

key-establishment-storage.h File Reference

Go to the source code of this file.

Functions

boolean	**storePublicPartnerData** (boolean isCertificate, int8u *data)
boolean	**retrieveAndClearPublicPartnerData** (EmberCertificateData *partnerCertificate, EmberPublicKeyData *partnerEphemeralPublicKey)
boolean	**storeSmac** (EmberSmacData *smac)
boolean	**getSmacPointer** (EmberSmacData **smacPtr)
void	**clearAllTemporaryPublicData** (void)

Function Documentation

```
boolean storePublicPartnerData ( boolean  isCertificate,
                                 int8u *  data
                               )
```

```
boolean retrieveAndClearPublicPartnerData ( EmberCertificateData *  partnerCertificate,
                                            EmberPublicKeyData *  partnerEphemeralPublicKey
                                          )
```

```
boolean storeSmac ( EmberSmacData *  smac )
```

```
boolean getSmacPointer ( EmberSmacData **  smacPtr )
```

```
void clearAllTemporaryPublicData ( void )
```

key-establishment.h File Reference

Go to the source code of this file.

Defines

#define EM_AF_KE_INITIATE_SIZE
#define EM_AF_KE_EPHEMERAL_SIZE
#define EM_AF_KE_SMAC_SIZE
#define EM_AF_KE_TERMINATE_SIZE

Define Documentation

#define EM_AF_KE_INITIATE_SIZE

Definition at line 10 of file key-establishment.h.

#define EM_AF_KE_EPHEMERAL_SIZE

Definition at line 12 of file key-establishment.h.

#define EM_AF_KE_SMAC_SIZE

Definition at line 13 of file key-establishment.h.

#define EM_AF_KE_TERMINATE_SIZE

Definition at line 16 of file key-establishment.h.

---

level-control.h File Reference

Go to the source code of this file.

Defines

#define LC_TICKS_PER_SECOND
#define LC_TICK_TIME
#define LEVEL_CONTROL_MINIMUM_LEVEL
#define LEVEL_CONTROL_MAXIMUM_LEVEL

Define Documentation

#define LC_TICKS_PER_SECOND

Definition at line 11 of file level-control.h.

#define LC_TICK_TIME

Definition at line 12 of file level-control.h.

#define LEVEL_CONTROL_MINIMUM_LEVEL

Definition at line 14 of file level-control.h.

#define LEVEL_CONTROL_MAXIMUM_LEVEL

Definition at line 15 of file level-control.h.

---

load-control-event-table.h File Reference

```
#include "../../include/af.h"
```

Go to the source code of this file.

Defines

```
#define LOAD_CONTROL_EVENT_TABLE_ENTRY_BYTES
#define LOAD_CONTROL_EVENT_TABLE_MEM_FOOTPRINT
#define RANDOMIZE_START_TIME_FLAG
#define RANDOMIZE_END_TIME_FLAG
#define CANCEL_WITH_RANDOMIZATION
```

Enumerations

```
enum {
        ENTRY_VOID,
        ENTRY_SCHEDULED,
        ENTRY_STARTED,
        ENTRY_IS_SUPERSEDED_EVENT,
        ENTRY_IS_CANCELLED_EVENT
      }
enum {
        EVENT_OPT_FLAG_OPT_IN,
        EVENT_OPT_FLAG_PARTIAL
      }
```

Functions

```
    void afLoadControlEventTableInit (int8u endpoint)
    void emAfScheduleLoadControlEvent (EmberAfLoadControlEvent *event)
    void emAfLoadControlEventTableTick (int8u endpoint)
 boolean emAfCancelAllLoadControlEvents (int8u cancelControl)
    void emAfCancelLoadControlEvent (int32u eventId, int8u cancelControl, int32u
           effectiveTime)
    void afScheduleCancelEvent (EmberAfLoadControlEvent *e)
    void emAfLoadControlEventOptInOrOut (int32u eventId, boolean optIn)
```

void	**emberAfEventAction** (**EmberAfLoadControlEvent** *event, int8u eventStatus, int8u sequenceNumber)
void	**emAfNoteSignatureFailure** (void)
void	**emAfLoadControlEventTablePrint** (void)

Define Documentation

#define LOAD_CONTROL_EVENT_TABLE_ENTRY_BYTES

Definition at line **37** of file **load-control-event-table.h**.

#define LOAD_CONTROL_EVENT_TABLE_MEM_FOOTPRINT

Definition at line **38** of file **load-control-event-table.h**.

#define RANDOMIZE_START_TIME_FLAG

Definition at line **41** of file **load-control-event-table.h**.

#define RANDOMIZE_END_TIME_FLAG

Definition at line **42** of file **load-control-event-table.h**.

#define CANCEL_WITH_RANDOMIZATION

Definition at line **44** of file **load-control-event-table.h**.

Enumeration Type Documentation

anonymous enum

Enumerator:

ENTRY_VOID
ENTRY_SCHEDULED
ENTRY_STARTED

ENTRY_IS_SUPERSEDED_EVENT
ENTRY_IS_CANCELLED_EVENT

Definition at line **47** of file **load-control-event-table.h**.

anonymous enum

Enumerator:
EVENT_OPT_FLAG_OPT_IN
EVENT_OPT_FLAG_PARTIAL

Definition at line **55** of file **load-control-event-table.h**.

Function Documentation

void afLoadControlEventTableInit (int8u endpoint)

void emAfScheduleLoadControlEvent (EmberAfLoadControlEvent * event)

This function is used to schedule events in the load control event table. The interface expects that the user will populate a LoadControlEvent with all the necessary fields and will pass a pointer to this event. The passed event will be copied into the event table so it does not need to survive the call to scheduleEvent.

A call to this function always generates an event response OTA

void emAfLoadControlEventTableTick (int8u endpoint)

Tells the Event table when a tick has taken place. This function should be called by the cluster that uses the event table.

boolean emAfCancelAllLoadControlEvents (int8u cancelControl)

Used to cancel all events in the event table

Returns:
A boolean value indicating that a response was generated for this action.

```
void emAfCancelLoadControlEvent ( int32u  eventId,
                                  int8u   cancelControl,
                                  int32u  effectiveTime
                                )
```

Used to cancel an event in the event table

A call to this function always generates an event response OTA.

```
void afScheduleCancelEvent ( EmberAfLoadControlEvent *  e  )
```

Used to schedule a call to cancel an event. Takes a LoadControlEvent struct which is used to encapsulate the necessary information for the scheduled cancel.

The scheduleCancelEvent function only uses three fields from the LoadControlEvent struct, eventId, startTime, and eventControl, as follows:

eventId: The eventId of the event to cancel.

startTime: The starttime of the event should be the same as the effective time of the cancel and will be used to schedule the call to cancel.

eventControl: The eventControl of the event should be the same as the cancelControl passed when sending the initial cancel event call.

NOTE: A call to cancelAllEvents() will wipe out this scheduled cancel as well as all other events which should be ok, since the event scheduled to cancel would be affected as well by the cancel all call anyway.

```
void emAfLoadControlEventOptInOrOut ( int32u   eventId,
                                      boolean  optIn
                                    )
```

An interface for opting in and out of an event

```
void emberAfEventAction ( EmberAfLoadControlEvent *  event,
                          int8u                      eventStatus,
                          int8u                      sequenceNumber
                        )
```

Called by the event table when an events status has changed. This handle should be used to inform the ESP and or react to scheduled events. The event table will take care of clearing itself if the event is completed.

void emAfNoteSignatureFailure (void)

A routine that prints a message indicating an attempt to append a signature to the event status message has failed.

void emAfLoadControlEventTablePrint (void)

Prints the load control event table

---

messaging-client-callback.h File Reference

Go to the source code of this file.

Functions

void	**emberAfPluginMessagingClientDisplayMessageCallback** (EmberAfPluginMessagingClientMessage *message)
void	**emberAfPluginMessagingClientCancelMessageCallback** (EmberAfPluginMessagingClientMessage *message)

messaging-client.h File Reference

Go to the source code of this file.

Defines

#define ZCL_MESSAGING_CLUSTER_TRANSMISSION_MASK
#define ZCL_MESSAGING_CLUSTER_IMPORTANCE_MASK
#define ZCL_MESSAGING_CLUSTER_CONFIRMATION_MASK
#define ZCL_MESSAGING_CLUSTER_START_TIME_NOW
#define ZCL_MESSAGING_CLUSTER_END_TIME_NEVER
#define ZCL_MESSAGING_CLUSTER_DURATION_UNTIL_CHANGED

Functions

void emAfPluginMessagingClientPrintInfo (void)
EmberAfStatus emberAfPluginMessagingClientConfirmMessage (void)

Define Documentation

#define ZCL_MESSAGING_CLUSTER_TRANSMISSION_MASK

Definition at line 12 of file messaging-client.h.

#define ZCL_MESSAGING_CLUSTER_IMPORTANCE_MASK

Definition at line 13 of file messaging-client.h.

#define ZCL_MESSAGING_CLUSTER_CONFIRMATION_MASK

Definition at line 14 of file messaging-client.h.

#define ZCL_MESSAGING_CLUSTER_START_TIME_NOW

Definition at line **16** of file **messaging-client.h**.

#define ZCL_MESSAGING_CLUSTER_END_TIME_NEVER

Definition at line **17** of file **messaging-client.h**.

#define ZCL_MESSAGING_CLUSTER_DURATION_UNTIL_CHANGED

Definition at line **18** of file **messaging-client.h**.

Function Documentation

void emAfPluginMessagingClientPrintInfo (void)

EmberAfStatus emberAfPluginMessagingClientConfirmMessage (void)

Confirm a message.

This function can be used to confirm a message. The Messaging client plugin will send the confirmation command to the endpoint on the node that sent the message.

Returns:

EMBER_ZCL_STATUS_SUCCESS if the confirmation was sent,
EMBER_ZCL_STATUS_FAILURE if an error occurred, or
EMBER_ZCL_STATUS_NOT_FOUND if the message does not exist.

---

messaging-server.h File Reference

Go to the source code of this file.

Data Structures

struct **EmberAfMessagingServerMessage**

Defines

#define **ZCL_MESSAGING_CLUSTER_TRANSMISSION_MASK**
#define **ZCL_MESSAGING_CLUSTER_IMPORTANCE_MASK**
#define **ZCL_MESSAGING_CLUSTER_CONFIRMATION_MASK**
#define **ZCL_MESSAGING_CLUSTER_START_TIME_NOW**
#define **ZCL_MESSAGING_CLUSTER_END_TIME_NEVER**
#define **ZCL_MESSAGING_CLUSTER_DURATION_UNTIL_CHANGED**

Functions

void **emAfPluginMessagingServerPrintInfo** (void)
EmberAfMessagingServerMessage * **emAfPluginMessagingServerGetMessageStruct** (void)

Define Documentation

#define ZCL_MESSAGING_CLUSTER_TRANSMISSION_MASK

Definition at line **12** of file **messaging-server.h**.

#define ZCL_MESSAGING_CLUSTER_IMPORTANCE_MASK

Definition at line **13** of file **messaging-server.h**.

#define ZCL_MESSAGING_CLUSTER_CONFIRMATION_MASK

Definition at line **14** of file **messaging-server.h**.

#define ZCL_MESSAGING_CLUSTER_START_TIME_NOW

Definition at line **16** of file **messaging-server.h**.

#define ZCL_MESSAGING_CLUSTER_END_TIME_NEVER

Definition at line **17** of file **messaging-server.h**.

#define ZCL_MESSAGING_CLUSTER_DURATION_UNTIL_CHANGED

Definition at line **18** of file **messaging-server.h**.

Function Documentation

void emAfPluginMessagingServerPrintInfo (void)

EmberAfMessagingServerMessage* emAfPluginMessagingServerGetMessageStruct (void)

---

network-cli.h File Reference

Go to the source code of this file.

Functions

void	**emberAfPrintChannelListFromMask** (int32u channelMask)

Variables

EmberCommandEntry	**networkCommands** []

Function Documentation

void emberAfPrintChannelListFromMask (int32u channelMask)

Variable Documentation

EmberCommandEntry networkCommands[]

---

on-off.h File Reference

Go to the source code of this file.

Functions

EmberAfStatus	emberAfOnOffSetValue (int8u endpoint, int8u command, boolean initiatedByLevelChange)

Function Documentation

```
EmberAfStatus emberAfOnOffSetValue ( int8u    endpoint,
                                     int8u    command,
                                     boolean  initiatedByLevelChange
                                   )
```

ota-bootload-ncp.h File Reference

Go to the source code of this file.

Functions

boolean	**emAfStartNcpBootloaderCommunications** (void)
void	**emAfPostNcpBootload** (boolean success)
boolean	**emAfRebootNcpAfterBootload** (void)
boolean	**emAfBootloadSendData** (const int8u *data, int16u length)
boolean	**emAfBootloadSendByte** (int8u byte)
boolean	**emAfBootloadWaitChar** (int8u *data, boolean expect, int8u expected)

Function Documentation

```
boolean emAfStartNcpBootloaderCommunications ( void )
```

```
void emAfPostNcpBootload ( boolean success )
```

```
boolean emAfRebootNcpAfterBootload ( void )
```

```
boolean emAfBootloadSendData ( const int8u *  data,
                               int16u         length
                             )
```

```
boolean emAfBootloadSendByte ( int8u byte )
```

```
boolean emAfBootloadWaitChar ( int8u *   data,
                               boolean   expect,
                               int8u     expected
                             )
```

ota-bootload-xmodem.h File Reference

Go to the source code of this file.

Functions

void	**emAfInitXmodemState** (boolean startImmediately)
boolean	**emAfSendXmodemData** (const int8u *data, int16u length, boolean finished)

Function Documentation

```
void emAfInitXmodemState ( boolean startImmediately )
```

```
boolean emAfSendXmodemData ( const int8u *  data,
                             int16u         length,
                             boolean        finished
                           )
```

ota-cli.h File Reference

Go to the source code of this file.

Defines

#define **OTA_COMMANDS**
#define **OTA_CLIENT_COMMANDS**
#define **OTA_SERVER_COMMANDS**

Functions

void **emAfOtaPrintAllImages** (void)
EmberAfOtaImageId emAfOtaFindImageIdByIndex (int8u index)
void **otaFindServerCommand** (void)
void **otaQueryServerCommand** (void)
void **otaUsePageRequestCommand** (void)
void **otaQuerySpecificFileCommand** (void)
void **otaSendUpgradeCommand** (void)
void **emAfOtaImageDelete** (void)
void **otaImageNotifyCommand** (void)

Define Documentation

#define OTA_COMMANDS

Definition at line **20** of file **ota-cli.h**.

#define OTA_CLIENT_COMMANDS

Definition at line **33** of file **ota-cli.h**.

#define OTA_SERVER_COMMANDS

Definition at line **50** of file **ota-cli.h**.

Function Documentation

void emAfOtaPrintAllImages (void)

EmberAfOtaImageId emAfOtaFindImageIdByIndex (int8u index)

void otaFindServerCommand (void)

void otaQueryServerCommand (void)

void otaUsePageRequestCommand (void)

void otaQuerySpecificFileCommand (void)

void otaSendUpgradeCommand (void)

void emAfOtaImageDelete (void)

void otaImageNotifyCommand (void)

ota-client-page-request-test.h File Reference

Go to the source code of this file.

Functions

void **pageRequestTest** (void)

Function Documentation

void pageRequestTest (void)

ota-client-page-request.h File Reference

Go to the source code of this file.

Defines

#define	**EM_AF_PAGE_REQUEST_BLOCK_SIZE**

Typedefs

typedef int8u	**EmAfPageRequestClientStatus**

Enumerations

enum	{ **EM_AF_NO_PAGE_REQUEST**, **EM_AF_WAITING_PAGE_REQUEST_REPLIES**, **EM_AF_RETRY_MISSED_PACKETS**, **EM_AF_PAGE_REQUEST_COMPLETE**, **EM_AF_PAGE_REQUEST_ERROR** }

Functions

int32u	**emAfInitPageRequestClient** (int32u offsetForPageRequest, int32u totalImageSize)
void	**emAfPageRequestTimerExpired** (void)
boolean	**emAfHandlingPageRequestClient** (void)
EmAfPageRequestClientStatus	**emAfGetCurrentPageRequestStatus** (void)
EmAfPageRequestClientStatus	**emAfNoteReceivedBlockForPageRequestClient** (int32u offset)
EmAfPageRequestClientStatus	**emAfNextMissedBlockRequestOffset** (int32u *nextOffset)
int32u	**emAfGetPageRequestMissedPacketDelayMs** (void)
int32u	**emAfGetFinishedPageRequestOffset** (void)
void	**emAfAbortPageRequest** (void)

Define Documentation

#define EM_AF_PAGE_REQUEST_BLOCK_SIZE

Definition at line **20** of file **ota-client-page-request.h**.

Typedef Documentation

typedef int8u EmAfPageRequestClientStatus

Definition at line **18** of file **ota-client-page-request.h**.

Enumeration Type Documentation

anonymous enum

Enumerator:

EM_AF_NO_PAGE_REQUEST
EM_AF_WAITING_PAGE_REQUEST_REPLIES
EM_AF_RETRY_MISSED_PACKETS
EM_AF_PAGE_REQUEST_COMPLETE
EM_AF_PAGE_REQUEST_ERROR

Definition at line **11** of file **ota-client-page-request.h**.

Function Documentation

int32u emAfInitPageRequestClient (int32u offsetForPageRequest, int32u totalImageSize)

void emAfPageRequestTimerExpired (void)

boolean emAfHandlingPageRequestClient (void)

EmAfPageRequestClientStatus emAfGetCurrentPageRequestStatus (void)

EmAfPageRequestClientStatus emAfNoteReceivedBlockForPageRequestClient (int32u offset)

EmAfPageRequestClientStatus emAfNextMissedBlockRequestOffset (int32u * nextOffset)

int32u emAfGetPageRequestMissedPacketDelayMs (void)

int32u emAfGetFinishedPageRequestOffset (void)

void emAfAbortPageRequest (void)

ota-client-policy.h File Reference

Go to the source code of this file.

Defines

```
#define EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_IMAGE_TYPE_ID
#define EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_FIRMWARE_VERSION
#define SECONDS_IN_MS
#define MINUTES_IN_MS
#define HOURS_IN_MS
#define EMBER_AF_INVALID_HARDWARE_VERSION
#define EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_HARDWARE_VERSION
```

Typedefs

```
typedef int8u EmberAfOtaDownloadResult
```

Enumerations

```
enum {
    EMBER_AF_OTA_DOWNLOAD_AND_VERIFY_SUCCESS,
    EMBER_AF_OTA_DOWNLOAD_TIME_OUT,
    EMBER_AF_OTA_VERIFY_FAILED,
    EMBER_AF_OTA_SERVER_ABORTED,
    EMBER_AF_OTA_CLIENT_ABORTED
}
```

Define Documentation

#define EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_IMAGE_TYPE_ID

Definition at line 17 of file ota-client-policy.h.

#define EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_FIRMWARE_VERSION

Definition at line 21 of file ota-client-policy.h.

#define SECONDS_IN_MS

Definition at line **24** of file **ota-client-policy.h**.

#define MINUTES_IN_MS

Definition at line **25** of file **ota-client-policy.h**.

#define HOURS_IN_MS

Definition at line **26** of file **ota-client-policy.h**.

#define EMBER_AF_INVALID_HARDWARE_VERSION

Definition at line **33** of file **ota-client-policy.h**.

#define EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_HARDWARE_VERSION

Definition at line **35** of file **ota-client-policy.h**.

Typedef Documentation

typedef int8u EmberAfOtaDownloadResult

Definition at line **48** of file **ota-client-policy.h**.

Enumeration Type Documentation

anonymous enum

Enumerator:

EMBER_AF_OTA_DOWNLOAD_AND_VERIFY_SUCCESS
EMBER_AF_OTA_DOWNLOAD_TIME_OUT
EMBER_AF_OTA_VERIFY_FAILED
EMBER_AF_OTA_SERVER_ABORTED

EMBER_AF_OTA_CLIENT_ABORTED

Definition at line **41** of file **ota-client-policy.h**.

---

ota-client-signature-verify.h File Reference

Go to the source code of this file.

Defines

#define	**MAX_DIGEST_CALCULATIONS_PER_CALL**

Functions

EmberAfImageVerifyStatus	**emAfOtaImageSignatureVerify** (int16u maxHashCalculations, const **EmberAfOtaImageId** *id, boolean newVerification)
void	**emAfOtaVerifyStoredDataFinish** (**EmberAfImageVerifyStatus** status)
void	**emAfOtaClientSignatureVerifyPrintSigners** (void)

Define Documentation

#define MAX_DIGEST_CALCULATIONS_PER_CALL

Definition at line **27** of file **ota-client-signature-verify.h**.

Function Documentation

```
EmberAfImageVerifyStatus emAfOtaImageSignatureVerify ( int16u                    maxHashCalculations,
                                                       const EmberAfOtaImageId * id,
                                                       boolean                   newVerification
                                                     )
```

```
void emAfOtaVerifyStoredDataFinish ( EmberAfImageVerifyStatus status )
```

```
void emAfOtaClientSignatureVerifyPrintSigners ( void )
```

ota-client.h File Reference

Go to the source code of this file.

Defines

#define EMBER_AF_PLUGIN_OTA_CLIENT_DOWNLOAD_DELAY_MS
#define EMBER_AF_OTA_SERVER_DISCOVERY_DELAY_MS
#define EMBER_AF_OTA_QUERY_DELAY_MS
#define EMBER_AF_PLUGIN_OTA_CLIENT_DOWNLOAD_ERROR_THRESHOLD
#define EMBER_AF_RUN_UPGRADE_REQUEST_DELAY_MS
#define EMBER_AF_PLUGIN_OTA_CLIENT_UPGRADE_WAIT_THRESHOLD
#define EMBER_AF_PLUGIN_OTA_CLIENT_PAGE_REQUEST_SIZE
#define EMBER_AF_OTA_CLIENT_PAGE_REQUEST_SPACING_MS
#define EMBER_AF_PLUGIN_OTA_CLIENT_VERIFY_DELAY_MS
#define NULL_EUI64
#define EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI0
#define EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI1
#define EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI2

Functions

void emAfOtaClientStop (void)
void emAfOtaClientPrintState (void)
void emAfSendImageBlockRequestTest (void)
void emAfSetPageRequest (boolean pageRequest)
boolean emAfUsingPageRequest (void)

Define Documentation

#define EMBER_AF_PLUGIN_OTA_CLIENT_DOWNLOAD_DELAY_MS

Definition at line 14 of file ota-client.h.

#define EMBER_AF_OTA_SERVER_DISCOVERY_DELAY_MS

Definition at line **24** of file **ota-client.h**.

#define EMBER_AF_OTA_QUERY_DELAY_MS

Definition at line **33** of file **ota-client.h**.

#define EMBER_AF_PLUGIN_OTA_CLIENT_DOWNLOAD_ERROR_THRESHOLD

Definition at line **39** of file **ota-client.h**.

#define EMBER_AF_RUN_UPGRADE_REQUEST_DELAY_MS

Definition at line **48** of file **ota-client.h**.

#define EMBER_AF_PLUGIN_OTA_CLIENT_UPGRADE_WAIT_THRESHOLD

Definition at line **55** of file **ota-client.h**.

#define EMBER_AF_PLUGIN_OTA_CLIENT_PAGE_REQUEST_SIZE

Definition at line **59** of file **ota-client.h**.

#define EMBER_AF_OTA_CLIENT_PAGE_REQUEST_SPACING_MS

Definition at line **65** of file **ota-client.h**.

#define EMBER_AF_PLUGIN_OTA_CLIENT_VERIFY_DELAY_MS

Definition at line **73** of file **ota-client.h**.

#define NULL_EUI64

Definition at line **76** of file **ota-client.h**.

#define EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI0

Definition at line **80** of file **ota-client.h**.

#define EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI1

Definition at line **84** of file **ota-client.h**.

#define EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI2

Definition at line **88** of file **ota-client.h**.

Function Documentation

void emAfOtaClientStop (void)

void emAfOtaClientPrintState (void)

void emAfSendImageBlockRequestTest (void)

void emAfSetPageRequest (boolean pageRequest)

boolean emAfUsingPageRequest (void)

---

ota-server-policy.h File Reference

Go to the source code of this file.

Functions

void	**emAfOtaServerPolicyPrint** (void)
void	**emAfOtaServerSetQueryPolicy** (int8u value)
void	**emAfOtaServerSetBlockRequestPolicy** (int8u value)
void	**emAfOtaServerSetUpgradePolicy** (int8u value)
boolean	**emAfServerPageRequestTickCallback** (int16u relativeOffset, int8u blockSize)
void	**emAfSetPageRequestMissedBlockModulus** (int16u modulus)
void	**emAfOtaServerSetPageRequestPolicy** (int8u value)

Function Documentation

void emAfOtaServerPolicyPrint (void)

void emAfOtaServerSetQueryPolicy (int8u value)

void emAfOtaServerSetBlockRequestPolicy (int8u value)

void emAfOtaServerSetUpgradePolicy (int8u value)

boolean emAfServerPageRequestTickCallback (int16u relativeOffset, int8u blockSize)

void emAfSetPageRequestMissedBlockModulus (int16u modulus)

void emAfOtaServerSetPageRequestPolicy (int8u value)

ota-server.h File Reference

Go to the source code of this file.

Functions

int8u	**emAfOtaServerGetBlockSize** (void)
int8u	**emAfOtaImageBlockRequestHandler** (EmberNodeId source, const **EmberAfOtaImageId** *id, int32u offset, int8u maxDataSize)
boolean	**emAfOtaPageRequestErrorHandler** (void)
void	**emAfOtaPageRequestTick** (int8u endpoint)
int8u	**emAfOtaPageRequestHandler** (int8u endpoint, const **EmberAfOtaImageId** *id, int32u offset, int8u maxDataSize, int16u pageSize, int16u responseSpacing)
boolean	**emAfOtaServerHandlingPageRequest** (void)
void	**emberAfOtaServerSendUpgradeCommandCallback** (EmberNodeId dest, int8u endpoint, const **EmberAfOtaImageId** *id)

Function Documentation

```
int8u emAfOtaServerGetBlockSize ( void )
```

```
int8u emAfOtaImageBlockRequestHandler ( EmberNodeId                source,
                                        const EmberAfOtaImageId *  id,
                                        int32u                     offset,
                                        int8u                      maxDataSize
                                      )
```

```
boolean emAfOtaPageRequestErrorHandler ( void )
```

```
void emAfOtaPageRequestTick ( int8u endpoint )
```

```
int8u emAfOtaPageRequestHandler ( int8u                      endpoint,
                                  const EmberAfOtaImageId *  id,
                                  int32u                     offset,
                                  int8u                      maxDataSize,
                                  int16u                     pageSize,
                                  int16u                     responseSpacing
                                )
```

```
boolean emAfOtaServerHandlingPageRequest ( void )
```

```
void emberAfOtaServerSendUpgradeCommandCallback ( EmberNodeId                dest,
```

```
                                                int8u                          endpoint,
                                                const EmberAfOtaImageId *      id
                                              )
```

ota-static-file-data.h File Reference

Go to the source code of this file.

Defines

#define STATIC_IMAGE_DATA
#define STATIC_IMAGE_DATA_SIZE

Define Documentation

#define STATIC_IMAGE_DATA

Definition at line 47 of file ota-static-file-data.h.

#define STATIC_IMAGE_DATA_SIZE

Definition at line 73 of file ota-static-file-data.h.

---

ota-storage-linux.h File Reference

Go to the source code of this file.

Data Structures

struct **EmAfOtaStorageLinuxConfig**

Typedefs

typedef void(**EmAfOtaStorageFileAddedHandler**)(const **EmberAfOtaHeader** *)

Functions

void **emAfOtaStorageGetConfig** (**EmAfOtaStorageLinuxConfig** *currentConfig)
void **emAfOtaStorageSetConfig** (const **EmAfOtaStorageLinuxConfig** *newConfig)

Typedef Documentation

typedef void(EmAfOtaStorageFileAddedHandler)(const EmberAfOtaHeader *)

Definition at line **3** of file **ota-storage-linux.h**.

Function Documentation

void emAfOtaStorageGetConfig (EmAfOtaStorageLinuxConfig * currentConfig)

void emAfOtaStorageSetConfig (const EmAfOtaStorageLinuxConfig * newConfig)

---

ota-storage-simple-custom.h File Reference

Go to the source code of this file.

Functions

boolean	**emberAfCustomStorageInitCallback** (void)
boolean	**emberAfCustomStorageReadCallback** (int32u offset, int32u length, int8u *returnData)
boolean	**emberAfCustomStorageWriteCallback** (const int8u *dataToWrite, int32u offset, int32u length)

Function Documentation

```
boolean emberAfCustomStorageInitCallback ( void )
```

```
boolean emberAfCustomStorageReadCallback ( int32u offset,
                                           int32u length,
                                           int8u * returnData
                                         )
```

```
boolean emberAfCustomStorageWriteCallback ( const int8u * dataToWrite,
                                            int32u offset,
                                            int32u length
                                          )
```

ota-storage-simple-driver.h File Reference

Go to the source code of this file.

Functions

boolean	**emberAfCustomStorageInitCallback** (void)
boolean	**emberAfCustomStorageReadCallback** (int32u offset, int32u length, int8u *returnData)
boolean	**emberAfCustomStorageWriteCallback** (const int8u *dataToWrite, int32u offset, int32u length)

Function Documentation

```
boolean emberAfCustomStorageInitCallback ( void )
```

```
boolean emberAfCustomStorageReadCallback ( int32u offset,
                                           int32u length,
                                           int8u * returnData
                                         )
```

```
boolean emberAfCustomStorageWriteCallback ( const int8u * dataToWrite,
                                            int32u offset,
                                            int32u length
                                          )
```

ota-storage.h File Reference

Go to the source code of this file.

Defines

#define **OTA_MINIMUM_HEADER_LENGTH**
#define **OTA_MAXIMUM_HEADER_LENGTH**
#define **OTA_FILE_MAGIC_NUMBER**
#define **MAGIC_NUMBER_OFFSET**
#define **HEADER_VERSION_OFFSET**
#define **HEADER_LENGTH_OFFSET**
#define **FIELD_CONTROL_OFFSET**
#define **MANUFACTURER_ID_OFFSET**
#define **IMAGE_TYPE_ID_OFFSET**
#define **VERSION_OFFSET**
#define **STACK_VERSION_OFFSET**
#define **HEADER_STRING_OFFSET**
#define **IMAGE_SIZE_OFFSET**
#define **OPTIONAL_FIELDS_OFFSET**
#define **HEADER_LENGTH_FIELD_LENGTH**
#define **TAG_OVERHEAD**
#define **OTA_HEADER_VERSION**
#define **SECURITY_CREDENTIAL_VERSION_FIELD_PRESENT_MASK**
#define **DEVICE_SPECIFIC_FILE_PRESENT_MASK**
#define **HARDWARE_VERSIONS_PRESENT_MASK**
#define **headerHasSecurityCredentials**(header)
#define **headerHasUpgradeFileDest**(header)
#define **headerHasHardwareVersions**(header)
#define **SIGNATURE_TAG_DATA_SIZE**
#define **INVALID_MANUFACTURER_ID**
#define **INVALID_DEVICE_ID**
#define **INVALID_FIRMWARE_VERSION**
#define **INVALID_EUI64**
#define **INVALID_OTA_IMAGE_ID**

Functions

EmberAfOtaStorageStatus	emAfOtaStorageSetDevice (const void *device)
void	emAfOtaStorageClose (void)
const char *	emAfOtaStorageGetFilepath (const EmberAfOtaImageId *id)
EmberAfOtaStorageStatus	emAfOtaStorageAddImageFile (const char *filename)
EmberAfOtaStorageStatus	emAfOtaStorageCreateImage (EmberAfOtaHeader *header, const char *filename)
EmberAfOtaStorageStatus	emAfOtaStorageAppendImageData (const char *filename, int32u length, const int8u *data)
EmberAfOtaStorageStatus	emAfOtaStorageGetHeaderLengthAndImageSize (const EmberAfOtaImageId *id, int32u *returnHeaderLength, int32u *returnImageSize)
EmberAfOtaImageId	emAfOtaStorageGetImageIdFromHeader (const EmberAfOtaHeader *header)
EmberAfOtaStorageStatus	emAfOtaStorageGetTagOffsetAndSize (const EmberAfOtaImageId *id, int16u tag, int32u *returnTagOffset, int32u *returnTagSize)
EmberAfOtaStorageStatus	emAfOtaStorageGetTagDataFromImage (const EmberAfOtaImageId *id, int16u tag, int8u *returnData, int32u *returnDataLength, int32u maxReturnDataLength)
EmberAfOtaStorageStatus	emAfOtaStorageGetRawHeaderData (const EmberAfOtaImageId *id, int8u *returnData, int32u *returnDataLength, int32u maxReturnDataLength)
EmberAfOtaStorageStatus	emAfOtaStorageReadAllTagInfo (const EmberAfOtaImageId *id, EmberAfTagData *tagInfo, int16u maxTags, int16u *totalTags)
boolean	emberAfIsOtaImageIdValid (const EmberAfOtaImageId *idToCompare)
EmberAfOtaStorageStatus	emberAfOtaStorageDeleteImageCallback (const EmberAfOtaImageId *id)
int32u	emberAfOtaStorageDriverRetrieveLastStoredOffsetCallback (void)
void	emberAfOtaStorageDriverDownloadFinishCallback (int32u finalOffset)
int	remainingAllocations (void)

Define Documentation

#define OTA_MINIMUM_HEADER_LENGTH

Definition at line 12 of file ota-storage.h.

#define OTA_MAXIMUM_HEADER_LENGTH

Definition at line 14 of file ota-storage.h.

#define OTA_FILE_MAGIC_NUMBER

Definition at line **16** of file **ota-storage.h**.

#define MAGIC_NUMBER_OFFSET

Definition at line **18** of file **ota-storage.h**.

#define HEADER_VERSION_OFFSET

Definition at line **19** of file **ota-storage.h**.

#define HEADER_LENGTH_OFFSET

Definition at line **20** of file **ota-storage.h**.

#define FIELD_CONTROL_OFFSET

Definition at line **21** of file **ota-storage.h**.

#define MANUFACTURER_ID_OFFSET

Definition at line **22** of file **ota-storage.h**.

#define IMAGE_TYPE_ID_OFFSET

Definition at line **23** of file **ota-storage.h**.

#define VERSION_OFFSET

Definition at line **24** of file **ota-storage.h**.

#define STACK_VERSION_OFFSET

Definition at line **25** of file **ota-storage.h**.

#define HEADER_STRING_OFFSET

Definition at line **26** of file **ota-storage.h**.

#define IMAGE_SIZE_OFFSET

Definition at line **27** of file **ota-storage.h**.

#define OPTIONAL_FIELDS_OFFSET

Definition at line **28** of file **ota-storage.h**.

#define HEADER_LENGTH_FIELD_LENGTH

Definition at line **31** of file **ota-storage.h**.

#define TAG_OVERHEAD

Definition at line **33** of file **ota-storage.h**.

#define OTA_HEADER_VERSION

Definition at line **35** of file **ota-storage.h**.

#define SECURITY_CREDENTIAL_VERSION_FIELD_PRESENT_MASK

Definition at line **37** of file **ota-storage.h**.

#define DEVICE_SPECIFIC_FILE_PRESENT_MASK

Definition at line **38** of file **ota-storage.h**.

#define HARDWARE_VERSIONS_PRESENT_MASK

Definition at line **39** of file **ota-storage.h**.

#define headerHasSecurityCredentials (header)

Definition at line **41** of file **ota-storage.h**.

#define headerHasUpgradeFileDest (header)

Definition at line **43** of file **ota-storage.h**.

#define headerHasHardwareVersions (header)

Definition at line **45** of file **ota-storage.h**.

#define SIGNATURE_TAG_DATA_SIZE

Definition at line **50** of file **ota-storage.h**.

#define INVALID_MANUFACTURER_ID

Definition at line **52** of file **ota-storage.h**.

#define INVALID_DEVICE_ID

Definition at line **53** of file **ota-storage.h**.

#define INVALID_FIRMWARE_VERSION

Definition at line **54** of file **ota-storage.h**.

#define INVALID_EUI64

Definition at line **55** of file **ota-storage.h**.

#define INVALID_OTA_IMAGE_ID

Definition at line **57** of file **ota-storage.h**.

Function Documentation

```
EmberAfOtaStorageStatus emAfOtaStorageSetDevice ( const void * device )
```

```
void emAfOtaStorageClose ( void )
```

```
const char* emAfOtaStorageGetFilepath ( const EmberAfOtaImageId * id )
```

```
EmberAfOtaStorageStatus emAfOtaStorageAddImageFile ( const char * filename )
```

```
EmberAfOtaStorageStatus emAfOtaStorageCreateImage ( EmberAfOtaHeader * header,
                                                    const char *       filename
                                                  )
```

```
EmberAfOtaStorageStatus emAfOtaStorageAppendImageData ( const char *   filename,
                                                        int32u         length,
                                                        const int8u *  data
                                                      )
```

```
EmberAfOtaStorageStatus emAfOtaStorageGetHeaderLengthAndImageSize ( const EmberAfOtaImageId * id,
                                                                    int32u *                  returnHeaderLength,
                                                                    int32u *                  returnImageSize
                                                                  )
```

```
EmberAfOtaImageId emAfOtaStorageGetImageIdFromHeader ( const EmberAfOtaHeader * header )
```

```
EmberAfOtaStorageStatus emAfOtaStorageGetTagOffsetAndSize ( const EmberAfOtaImageId * id,
                                                            int16u                    tag,
                                                            int32u *                  returnTagOffset,
                                                            int32u *                  returnTagSize
```

```
)
```

```
EmberAfOtaStorageStatus emAfOtaStorageGetTagDataFromImage ( const EmberAfOtaImageId * id,
                                                            int16u                    tag,
                                                            int8u *                   returnData,
                                                            int32u *                  returnDataLength,
                                                            int32u                    maxReturnDataLength
                                                          )
```

```
EmberAfOtaStorageStatus emAfOtaStorageGetRawHeaderData ( const EmberAfOtaImageId * id,
                                                         int8u *                   returnData,
                                                         int32u *                  returnDataLength,
                                                         int32u                    maxReturnDataLength
                                                       )
```

```
EmberAfOtaStorageStatus emAfOtaStorageReadAllTagInfo ( const EmberAfOtaImageId * id,
                                                       EmberAfTagData *          tagInfo,
                                                       int16u                    maxTags,
                                                       int16u *                  totalTags
                                                     )
```

```
boolean emberAfIsOtaImageIdValid ( const EmberAfOtaImageId * idToCompare )
```

```
EmberAfOtaStorageStatus emberAfOtaStorageDeleteImageCallback ( const EmberAfOtaImageId * id )
```

```
int32u emberAfOtaStorageDriverRetrieveLastStoredOffsetCallback ( void )
```

```
void emberAfOtaStorageDriverDownloadFinishCallback ( int32u finalOffset )
```

```
int remainingAllocations ( void )
```

---

ota.h File Reference

Go to the source code of this file.

Defines

```
#define OTA_UPGRADE_END_RESPONSE_RUN_NOW
#define MFG_ID_WILD_CARD
#define IMAGE_TYPE_WILD_CARD
#define IMAGE_TYPE_SECURITY
#define IMAGE_TYPE_CONFIG
#define IMAGE_TYPE_LOG
#define FILE_VERSION_WILD_CARD
#define ZIGBEE_2006_STACK_VERSION
#define ZIGBEE_2007_STACK_VERSION
#define ZIGBEE_PRO_STACK_VERSION
#define OTA_HW_VERSION_BIT_MASK
#define OTA_NODE_EUI64_BIT_MASK
#define OTA_TAG_UPGRADE_IMAGE
#define OTA_TAG_ECDSA_SIGNATURE
#define OTA_TAG_ECDSA_SIGNING_CERTIFICATE
#define OTA_TAG_RESERVED_START
#define OTA_TAG_RESERVED_END
#define OTA_TAG_MANUFACTURER_SPECIFIC_START
#define EM_AF_OTA_MAX_COMMAND_ID
#define otaPrintln(...)
#define otaPrint(...)
#define emAfPrintPercentageSetStartAndEnd(x, y)
#define emAfPrintPercentageUpdate(x, y, z)
```

Enumerations

```
enum {
    OTA_USE_OFFSET_TIME,
    OTA_USE_UTC_TIME
}
enum {
    OTA_UPGRADE_STATUS_NORMAL,
    OTA_UPGRADE_STATUS_DOWNLOAD_IN_PROGRESS,
    OTA_UPGRADE_STATUS_DOWNLOAD_COMPLETE,
    OTA_UPGRADE_STATUS_WAIT,
    OTA_UPGRADE_STATUS_COUNTDOWN
}
```

Functions

EmberAfOtaImageId	**emAfOtaCreateEmberAfOtaImageIdStruct** (int16u manufacturerId, int16u imageType, int32u fileVersion)
int8u	**emAfOtaParseImageIdFromMessage** (**EmberAfOtaImageId** *returnId, const int8u *buffer, int8u length)

Variables

EmberNodeId	**upgradeServerNodeId**
PGM int8u	**emAfOtaMinMessageLengths** []

Define Documentation

#define OTA_UPGRADE_END_RESPONSE_RUN_NOW

Definition at line **11** of file **ota.h**.

#define MFG_ID_WILD_CARD

Definition at line **13** of file **ota.h**.

#define IMAGE_TYPE_WILD_CARD

Definition at line **14** of file **ota.h**.

#define IMAGE_TYPE_SECURITY

Definition at line **15** of file **ota.h**.

#define IMAGE_TYPE_CONFIG

Definition at line **16** of file **ota.h**.

#define IMAGE_TYPE_LOG

Definition at line **17** of file **ota.h**.

#define FILE_VERSION_WILD_CARD

Definition at line **18** of file **ota.h**.

#define ZIGBEE_2006_STACK_VERSION

Definition at line **20** of file **ota.h**.

#define ZIGBEE_2007_STACK_VERSION

Definition at line **21** of file **ota.h**.

#define ZIGBEE_PRO_STACK_VERSION

Definition at line **22** of file **ota.h**.

#define OTA_HW_VERSION_BIT_MASK

Definition at line **25** of file **ota.h**.

#define OTA_NODE_EUI64_BIT_MASK

Definition at line **26** of file **ota.h**.

#define OTA_TAG_UPGRADE_IMAGE

Definition at line **37** of file **ota.h**.

#define OTA_TAG_ECDSA_SIGNATURE

Definition at line **38** of file **ota.h**.

#define OTA_TAG_ECDSA_SIGNING_CERTIFICATE

Definition at line **39** of file **ota.h**.

#define OTA_TAG_RESERVED_START

Definition at line **40** of file **ota.h**.

#define OTA_TAG_RESERVED_END

Definition at line **41** of file **ota.h**.

#define OTA_TAG_MANUFACTURER_SPECIFIC_START

Definition at line **42** of file **ota.h**.

#define EM_AF_OTA_MAX_COMMAND_ID

Definition at line **47** of file **ota.h**.

#define otaPrintln (...)

Definition at line **50** of file **ota.h**.

#define otaPrint (...)

Definition at line **51** of file **ota.h**.

#define emAfPrintPercentageSetStartAndEnd (x, y)

Definition at line **70** of file **ota.h**.

#define emAfPrintPercentageUpdate (x, y, z)

Definition at line **71** of file **ota.h**.

Enumeration Type Documentation

anonymous enum

Enumerator:

OTA_USE_OFFSET_TIME
OTA_USE_UTC_TIME

Definition at line **7** of file **ota.h**.

anonymous enum

Enumerator:

OTA_UPGRADE_STATUS_NORMAL
OTA_UPGRADE_STATUS_DOWNLOAD_IN_PROGRESS
OTA_UPGRADE_STATUS_DOWNLOAD_COMPLETE
OTA_UPGRADE_STATUS_WAIT
OTA_UPGRADE_STATUS_COUNTDOWN

Definition at line **29** of file **ota.h**.

Function Documentation

```
EmberAfOtaImageId emAfOtaCreateEmberAfOtaImageIdStruct ( int16u  manufacturerId,
                                                         int16u  imageType,
                                                         int32u  fileVersion
                                                       )
```

```
int8u emAfOtaParseImageIdFromMessage ( EmberAfOtaImageId *  returnId,
                                       const int8u *        buffer,
                                       int8u                length
                                     )
```

Variable Documentation

```
EmberNodeId upgradeServerNodeId
```

```
PGM int8u emAfOtaMinMessageLengths[]
```

---

partner-link-key-exchange.h File Reference

Go to the source code of this file.

Defines

```
#define EMBER_AF_PLUGIN_PARTNER_LINK_KEY_EXCHANGE_TIMEOUT_MILLISECONDS
```

Define Documentation

#define EMBER_AF_PLUGIN_PARTNER_LINK_KEY_EXCHANGE_TIMEOUT_MILLISECONDS

Definition at line 9 of file partner-link-key-exchange.h.

---

plugin-cli.h File Reference

```
#include <ATTRIBUTE_STORAGE_CONFIGURATION>
```

Go to the source code of this file.

Defines

```
#define EMBER_AF_PLUGIN_COMMANDS
```

Define Documentation

#define EMBER_AF_PLUGIN_COMMANDS

Definition at line 15 of file plugin-cli.h.

---

price-client-callback.h File Reference

Go to the source code of this file.

Functions

void	**emberAfPluginPriceClientPriceStartedCallback** (EmberAfPluginPriceClientPrice *price)
void	**emberAfPluginPriceClientPriceExpiredCallback** (EmberAfPluginPriceClientPrice *price)

price-client.h File Reference

Go to the source code of this file.

Defines

#define EMBER_AF_PLUGIN_PRICE_CLIENT_TABLE_SIZE
#define ZCL_PRICE_CLUSTER_PRICE_ACKNOWLEDGEMENT_MASK
#define ZCL_PRICE_CLUSTER_START_TIME_NOW
#define ZCL_PRICE_CLUSTER_END_TIME_NEVER
#define ZCL_PRICE_CLUSTER_DURATION_UNTIL_CHANGED
#define ZCL_PRICE_CLUSTER_PRICE_RATIO_NOT_USED
#define ZCL_PRICE_CLUSTER_GENERATION_PRICE_NOT_USED
#define ZCL_PRICE_CLUSTER_GENERATION_PRICE_RATIO_NOT_USED
#define ZCL_PRICE_CLUSTER_ALTERNATE_COST_DELIVERED_NOT_USED
#define ZCL_PRICE_CLUSTER_ALTERNATE_COST_UNIT_NOT_USED
#define ZCL_PRICE_CLUSTER_ALTERNATE_COST_TRAILING_DIGIT_NOT_USED
#define ZCL_PRICE_CLUSTER_NUMBER_OF_BLOCK_THRESHOLDS_NOT_USED
#define ZCL_PRICE_CLUSTER_PRICE_CONTROL_NOT_USED

Functions

void emAfPluginPriceClientPrintInfo (void)

Define Documentation

#define EMBER_AF_PLUGIN_PRICE_CLIENT_TABLE_SIZE

Definition at line 9 of file price-client.h.

#define ZCL_PRICE_CLUSTER_PRICE_ACKNOWLEDGEMENT_MASK

Definition at line 12 of file price-client.h.

#define ZCL_PRICE_CLUSTER_START_TIME_NOW

Definition at line 14 of file price-client.h.

#define ZCL_PRICE_CLUSTER_END_TIME_NEVER

Definition at line **15** of file **price-client.h**.

#define ZCL_PRICE_CLUSTER_DURATION_UNTIL_CHANGED

Definition at line **16** of file **price-client.h**.

#define ZCL_PRICE_CLUSTER_PRICE_RATIO_NOT_USED

Definition at line **17** of file **price-client.h**.

#define ZCL_PRICE_CLUSTER_GENERATION_PRICE_NOT_USED

Definition at line **18** of file **price-client.h**.

#define ZCL_PRICE_CLUSTER_GENERATION_PRICE_RATIO_NOT_USED

Definition at line **19** of file **price-client.h**.

#define ZCL_PRICE_CLUSTER_ALTERNATE_COST_DELIVERED_NOT_USED

Definition at line **20** of file **price-client.h**.

#define ZCL_PRICE_CLUSTER_ALTERNATE_COST_UNIT_NOT_USED

Definition at line **21** of file **price-client.h**.

#define ZCL_PRICE_CLUSTER_ALTERNATE_COST_TRAILING_DIGIT_NOT_USED

Definition at line **22** of file **price-client.h**.

#define ZCL_PRICE_CLUSTER_NUMBER_OF_BLOCK_THRESHOLDS_NOT_USED

Definition at line **23** of file **price-client.h**.

#define ZCL_PRICE_CLUSTER_PRICE_CONTROL_NOT_USED

Definition at line **24** of file **price-client.h**.

Function Documentation

void emAfPluginPriceClientPrintInfo (void)

---

price-server.h File Reference

Go to the source code of this file.

Data Structures

struct	EmberAfScheduledPrice

Defines

#define	EMBER_AF_PLUGIN_PRICE_SERVER_PRICE_TABLE_SIZE
#define	ZCL_PRICE_CLUSTER_PRICE_ACKNOWLEDGEMENT_MASK
#define	ZCL_PRICE_CLUSTER_START_TIME_NOW
#define	ZCL_PRICE_CLUSTER_END_TIME_NEVER
#define	ZCL_PRICE_CLUSTER_DURATION_UNTIL_CHANGED
#define	ZCL_PRICE_CLUSTER_NUMBER_OF_EVENTS_ALL
#define	ZCL_PRICE_INVALID_INDEX

Functions

void	emberAfPriceClearPriceTable (void)
EmberAfScheduledPrice *	emberAfPriceGetPriceTableEntry (int8u index)
void	emberAfPricePrintTable (void)
EmberAfScheduledPrice *	emberAfGetCurrentPrice (void)

Define Documentation

#define EMBER_AF_PLUGIN_PRICE_SERVER_PRICE_TABLE_SIZE

Definition at line 9 of file price-server.h.

#define ZCL_PRICE_CLUSTER_PRICE_ACKNOWLEDGEMENT_MASK

Definition at line 12 of file price-server.h.

#define ZCL_PRICE_CLUSTER_START_TIME_NOW

Definition at line 14 of file price-server.h.

#define ZCL_PRICE_CLUSTER_END_TIME_NEVER

Definition at line **15** of file **price-server.h**.

#define ZCL_PRICE_CLUSTER_DURATION_UNTIL_CHANGED

Definition at line **16** of file **price-server.h**.

#define ZCL_PRICE_CLUSTER_NUMBER_OF_EVENTS_ALL

Definition at line **17** of file **price-server.h**.

#define ZCL_PRICE_INVALID_INDEX

Definition at line **42** of file **price-server.h**.

Function Documentation

void emberAfPriceClearPriceTable (void)

EmberAfScheduledPrice* emberAfPriceGetPriceTableEntry (int8u index)

void emberAfPricePrintTable (void)

EmberAfScheduledPrice* emberAfGetCurrentPrice (void)

---

print-cluster.h File Reference

Go to the source code of this file.

Defines

```
#define __PRINTCLUSTER_BASIC_CLUSTER
#define __PRINTCLUSTER_POWER_CONFIG_CLUSTER
#define __PRINTCLUSTER_DEVICE_TEMP_CLUSTER
#define __PRINTCLUSTER_IDENTIFY_CLUSTER
#define __PRINTCLUSTER_GROUPS_CLUSTER
#define __PRINTCLUSTER_SCENES_CLUSTER
#define __PRINTCLUSTER_ON_OFF_CLUSTER
#define __PRINTCLUSTER_ON_OFF_SWITCH_CONFIG_CLUSTER
#define __PRINTCLUSTER_LEVEL_CONTROL_CLUSTER
#define __PRINTCLUSTER_ALARM_CLUSTER
#define __PRINTCLUSTER_TIME_CLUSTER
#define __PRINTCLUSTER_RSSI_LOCATION_CLUSTER
#define __PRINTCLUSTER_BINARY_INPUT_BASIC_CLUSTER
#define __PRINTCLUSTER_COMMISSIONING_CLUSTER
#define __PRINTCLUSTER_SHADE_CONFIG_CLUSTER
#define __PRINTCLUSTER_PUMP_CONFIG_CONTROL_CLUSTER
#define __PRINTCLUSTER_THERMOSTAT_CLUSTER
#define __PRINTCLUSTER_FAN_CONTROL_CLUSTER
#define __PRINTCLUSTER_DEHUMID_CONTROL_CLUSTER
#define __PRINTCLUSTER_THERMOSTAT_UI_CONFIG_CLUSTER
#define __PRINTCLUSTER_COLOR_CONTROL_CLUSTER
#define __PRINTCLUSTER_BALLAST_CONFIGURATION_CLUSTER
#define __PRINTCLUSTER_ILLUM_MEASUREMENT_CLUSTER
#define __PRINTCLUSTER_ILLUM_LEVEL_SENSING_CLUSTER
#define __PRINTCLUSTER_TEMP_MEASUREMENT_CLUSTER
#define __PRINTCLUSTER_PRESSURE_MEASUREMENT_CLUSTER
#define __PRINTCLUSTER_FLOW_MEASUREMENT_CLUSTER
#define __PRINTCLUSTER_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER
#define __PRINTCLUSTER_OCCUPANCY_SENSING_CLUSTER
#define __PRINTCLUSTER_IAS_ZONE_CLUSTER
#define __PRINTCLUSTER_IAS_ACE_CLUSTER
#define __PRINTCLUSTER_IAS_WD_CLUSTER
```

#define __PRINTCLUSTER_DOOR_LOCK_CLUSTER
#define __PRINTCLUSTER_WINDOW_COVERING_CLUSTER
#define __PRINTCLUSTER_GENERIC_TUNNEL_CLUSTER
#define __PRINTCLUSTER_BACNET_PROTOCOL_TUNNEL_CLUSTER
#define __PRINTCLUSTER_OTA_BOOTLOAD_CLUSTER
#define __PRINTCLUSTER_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER
#define __PRINTCLUSTER_SIMPLE_METERING_CLUSTER
#define __PRINTCLUSTER_PRICE_CLUSTER
#define __PRINTCLUSTER_MESSAGING_CLUSTER
#define __PRINTCLUSTER_TUNNELING_CLUSTER
#define __PRINTCLUSTER_PREPAYMENT_CLUSTER
#define __PRINTCLUSTER_KEY_ESTABLISHMENT_CLUSTER
#define __PRINTCLUSTER_SAMPLE_MFG_SPECIFIC_CLUSTER
#define CLUSTER_IDS_TO_NAMES
#define MAX_CLUSTER_NAME_LENGTH

Define Documentation

#define __PRINTCLUSTER_BASIC_CLUSTER

Definition at line 18 of file print-cluster.h.

#define __PRINTCLUSTER_POWER_CONFIG_CLUSTER

Definition at line 23 of file print-cluster.h.

#define __PRINTCLUSTER_DEVICE_TEMP_CLUSTER

Definition at line 28 of file print-cluster.h.

#define __PRINTCLUSTER_IDENTIFY_CLUSTER

Definition at line 33 of file print-cluster.h.

#define __PRINTCLUSTER_GROUPS_CLUSTER

Definition at line 38 of file print-cluster.h.

#define __PRINTCLUSTER_SCENES_CLUSTER

Definition at line **43** of file **print-cluster.h**.

#define __PRINTCLUSTER_ON_OFF_CLUSTER

Definition at line **48** of file **print-cluster.h**.

#define __PRINTCLUSTER_ON_OFF_SWITCH_CONFIG_CLUSTER

Definition at line **53** of file **print-cluster.h**.

#define __PRINTCLUSTER_LEVEL_CONTROL_CLUSTER

Definition at line **58** of file **print-cluster.h**.

#define __PRINTCLUSTER_ALARM_CLUSTER

Definition at line **63** of file **print-cluster.h**.

#define __PRINTCLUSTER_TIME_CLUSTER

Definition at line **68** of file **print-cluster.h**.

#define __PRINTCLUSTER_RSSI_LOCATION_CLUSTER

Definition at line **73** of file **print-cluster.h**.

#define __PRINTCLUSTER_BINARY_INPUT_BASIC_CLUSTER

Definition at line **78** of file **print-cluster.h**.

#define __PRINTCLUSTER_COMMISSIONING_CLUSTER

Definition at line **83** of file **print-cluster.h**.

#define __PRINTCLUSTER_SHADE_CONFIG_CLUSTER

Definition at line **88** of file **print-cluster.h**.

#define __PRINTCLUSTER_PUMP_CONFIG_CONTROL_CLUSTER

Definition at line **93** of file **print-cluster.h**.

#define __PRINTCLUSTER_THERMOSTAT_CLUSTER

Definition at line **98** of file **print-cluster.h**.

#define __PRINTCLUSTER_FAN_CONTROL_CLUSTER

Definition at line **103** of file **print-cluster.h**.

#define __PRINTCLUSTER_DEHUMID_CONTROL_CLUSTER

Definition at line **108** of file **print-cluster.h**.

#define __PRINTCLUSTER_THERMOSTAT_UI_CONFIG_CLUSTER

Definition at line **113** of file **print-cluster.h**.

#define __PRINTCLUSTER_COLOR_CONTROL_CLUSTER

Definition at line **118** of file **print-cluster.h**.

#define __PRINTCLUSTER_BALLAST_CONFIGURATION_CLUSTER

Definition at line **123** of file **print-cluster.h**.

#define __PRINTCLUSTER_ILLUM_MEASUREMENT_CLUSTER

Definition at line **128** of file **print-cluster.h**.

#define __PRINTCLUSTER_ILLUM_LEVEL_SENSING_CLUSTER

Definition at line **133** of file **print-cluster.h**.

#define __PRINTCLUSTER_TEMP_MEASUREMENT_CLUSTER

Definition at line **138** of file **print-cluster.h**.

#define __PRINTCLUSTER_PRESSURE_MEASUREMENT_CLUSTER

Definition at line **143** of file **print-cluster.h**.

#define __PRINTCLUSTER_FLOW_MEASUREMENT_CLUSTER

Definition at line **148** of file **print-cluster.h**.

#define __PRINTCLUSTER_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER

Definition at line **153** of file **print-cluster.h**.

#define __PRINTCLUSTER_OCCUPANCY_SENSING_CLUSTER

Definition at line **158** of file **print-cluster.h**.

#define __PRINTCLUSTER_IAS_ZONE_CLUSTER

Definition at line **163** of file **print-cluster.h**.

#define __PRINTCLUSTER_IAS_ACE_CLUSTER

Definition at line **168** of file **print-cluster.h**.

#define __PRINTCLUSTER_IAS_WD_CLUSTER

Definition at line **173** of file **print-cluster.h**.

#define __PRINTCLUSTER_DOOR_LOCK_CLUSTER

Definition at line **178** of file **print-cluster.h**.

#define __PRINTCLUSTER_WINDOW_COVERING_CLUSTER

Definition at line **183** of file **print-cluster.h**.

#define __PRINTCLUSTER_GENERIC_TUNNEL_CLUSTER

Definition at line **188** of file **print-cluster.h**.

#define __PRINTCLUSTER_BACNET_PROTOCOL_TUNNEL_CLUSTER

Definition at line **193** of file **print-cluster.h**.

#define __PRINTCLUSTER_OTA_BOOTLOAD_CLUSTER

Definition at line **198** of file **print-cluster.h**.

#define __PRINTCLUSTER_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER

Definition at line **203** of file **print-cluster.h**.

#define __PRINTCLUSTER_SIMPLE_METERING_CLUSTER

Definition at line **208** of file **print-cluster.h**.

#define __PRINTCLUSTER_PRICE_CLUSTER

Definition at line **213** of file **print-cluster.h**.

#define __PRINTCLUSTER_MESSAGING_CLUSTER

Definition at line **218** of file **print-cluster.h**.

#define __PRINTCLUSTER_TUNNELING_CLUSTER

Definition at line **223** of file **print-cluster.h**.

#define __PRINTCLUSTER_PREPAYMENT_CLUSTER

Definition at line **228** of file **print-cluster.h**.

#define __PRINTCLUSTER_KEY_ESTABLISHMENT_CLUSTER

Definition at line **233** of file **print-cluster.h**.

#define __PRINTCLUSTER_SAMPLE_MFG_SPECIFIC_CLUSTER

Definition at line **238** of file **print-cluster.h**.

#define CLUSTER_IDS_TO_NAMES

Definition at line **240** of file **print-cluster.h**.

#define MAX_CLUSTER_NAME_LENGTH

Definition at line **288** of file **print-cluster.h**.

---

psl-commissioning-callback.h File Reference

Go to the source code of this file.

Functions

void	**emberAfPluginPslCommissioningTouchLinkCompleteCallback** (const EmberPslNetwork *networkInfo, int8u deviceInformationRecordCount, const EmberPslDeviceInfoRecord *deviceInformationRecordList)
void	**emberAfPluginPslCommissioningTouchLinkFailedCallback** (void)
int8u	**emberAfPluginPslCommissioningGroupIdentifierCountCallback** (int8u endpoint)
boolean	**emberAfPluginPslCommissioningGroupIdentifierCallback** (int8u endpoint, int8u index, EmberAfPluginPslCommissioningGroupInformationRecord *record)
int8u	**emberAfPluginPslCommissioningEndpointInformationCountCallback** (int8u endpoint)
boolean	**emberAfPluginPslCommissioningEndpointInformationCallback** (int8u endpoint, int8u index, EmberAfPluginPslCommissioningEndpointInformationRecord *record)
void	**emberAfPluginPslCommissioningIdentifyCallback** (int16u duration)

psl-commissioning.h File Reference

Go to the source code of this file.

Defines

#define	**EMBER_AF_PLUGIN_PSL_COMMISSIONING_TOUCH_LINK_MILLISECONDS_DELAY**
#define	**EMBER_AF_PLUGIN_PSL_COMMISSIONING_DEVICE_IS_SLEEPY**

Functions

EmberStatus	**emberAfPslSetInitialSecurityState** (void)
EmberStatus	**emberAfPslInitiateTouchLink** (void)
void	**emberAfPslAbortTouchLink** (void)
boolean	**emberAfPslTouchLinkInProgress** (void)
void	**emberAfPslResetToFactoryNew** (void)
EmberStatus	**emberAfPslDeviceInformationRequest** (const EmberEUI64 target, int32u transaction, int8u startIndex)
EmberStatus	**emberAfPslIdentifyRequest** (const EmberEUI64 target, int32u transaction, int16u identifyDuration)
EmberStatus	**emberAfPslResetToFactoryNewRequest** (const EmberEUI64 target, int32u transaction)

Variables

int32u	**emPslChannelMask**

Define Documentation

#define EMBER_AF_PLUGIN_PSL_COMMISSIONING_TOUCH_LINK_MILLISECONDS_DELAY

Definition at line **109** of file **psl-commissioning.h**.

#define EMBER_AF_PLUGIN_PSL_COMMISSIONING_DEVICE_IS_SLEEPY

Definition at line **120** of file **psl-commissioning.h**.

Function Documentation

EmberStatus emberAfPslSetInitialSecurityState (void)

Generate a random network key and initialize the security state of the device.

This function is a convenience wrapper for emberPslSetInitialSecurityState, which must be called before

starting or joining a network. The plugin will initialize the security state for the initiator during touch linking. The target must initialize its own security state prior to forming a network either by using this function or by calling emberPslSetInitialSecurityState directly.

Returns:
An EmberStatus value that indicates the success or failure of the command.

EmberStatus emberAfPslInitiateTouchLink (void)

Initiate the touch link procedure.

This function will cause the stack to broadcast a series of ScanRequest commands via inter-PAN messaging. The plugin will select the end device target that sent a ScanResponse command with the strongest RSSI and attempt to link with it. If touch linking completes successfully, the plugin will call **emberAfPluginPslCommissioningTouchLinkCompleteCallback** with information about the network and the target. If touch linking fails, the plugin will call **emberAfPluginPslCommissioningTouchLinkFailedCallback**.

Returns:
An EmberStatus value that indicates the success or failure of the command.

void emberAfPslAbortTouchLink (void)

Abort the touch link procedure.

This function can be called to cancel the touch link procedure. This can be useful, for example, if the touch link target is incorrect.

boolean emberAfPslTouchLinkInProgress (void)

Indicates if a touch link procedure is currently in progress.

Returns:
TRUE if a touch link is in progress or FALSE otherwise.

void emberAfPslResetToFactoryNew (void)

Reset the local device to a factory new state.

This function will cause the device to leave the network and clear its network parameters, reset its attributes to their default values, and clear the group and scene tables.

EmberStatus emberAfPslDeviceInformationRequest (const EmberEUI64 target, int32u transaction, int8u startIndex)

Send a DeviceInformationRequest command to a target.

This function will unicast a DeviceInformationRequest command to the given target via inter-PAN messaging. The target device may reject the command if the transaction id is no longer valid in the network.

Parameters:

target	The IEEE address of the target device.
transaction	The identifier for the inter-PAN transaction.
startIndex	The starting index into the remote device table.

Returns:
An EmberStatus value that indicates the success or failure of the command.

```
EmberStatus emberAfPslIdentifyRequest ( const EmberEUI64  target,
                                        int32u            transaction,
                                        int16u            identifyDuration
                                      )
```

Send an IdentifyRequest command to a target.

This function will unicast a IdentifyRequest command to the given target via inter-PAN messaging. The target device may reject the command if the transaction id is no longer valid in the network.

Parameters:

target	The IEEE address of the target device.
transaction	The identifier for the inter-PAN transaction.
identifyDuration	The length of time the target is to remain in identify mode.

Returns:
An EmberStatus value that indicates the success or failure of the command.

```
EmberStatus emberAfPslResetToFactoryNewRequest ( const EmberEUI64  target,
                                                 int32u            transaction
                                               )
```

Send a ResetToFactoryNewRequest command to a target.

This function will unicast a ResetToFactoryNewRequest command to the given target via inter-PAN messaging. The target device may reject the command if the transaction id is no longer valid in the network.

Parameters:

target	The IEEE address of the target device.
transaction	The identifier for the inter-PAN transaction.

Returns:
An EmberStatus value that indicates the success or failure of the command.

Variable Documentation

int32u emPslChannelMask

---

scenes.h File Reference

Go to the source code of this file.

Functions

EmberAfStatus	emberAfScenesSetSceneCountAttribute (int8u endpoint, int8u newCount)
EmberAfStatus	emberAfScenesMakeValid (int8u endpoint, int8u sceneId, int16u groupId)
EmberAfStatus	emberAfScenesMakeInvalid (int8u endpoint)
void	emAfPluginScenesServerPrintInfo (void)

Function Documentation

```
EmberAfStatus emberAfScenesSetSceneCountAttribute ( int8u  endpoint,
                                                    int8u  newCount
                                                  )
```

```
EmberAfStatus emberAfScenesMakeValid ( int8u   endpoint,
                                       int8u   sceneId,
                                       int16u  groupId
                                     )
```

```
EmberAfStatus emberAfScenesMakeInvalid ( int8u  endpoint )
```

```
void emAfPluginScenesServerPrintInfo ( void )
```

security-cli.h File Reference

Go to the source code of this file.

Defines

#define **KEY_UPDATE_COMMANDS**

Functions

void **changeKeyCommand** (void)
void **keyUpdateCommand** (void)
void **cbkeStartCommand** (void)
void **cbkeInterPanCommand** (void)
void **cbkePartnerCommand** (void)
void **printKeyInfo** (void)

Variables

EmberCommandEntry	**cbkeCommands** []
EmberCommandEntry	**changeKeyCommands** []
EmberKeyData	**cliPreconfiguredLinkKey**
EmberKeyData	**cliNetworkKey**

Define Documentation

#define KEY_UPDATE_COMMANDS

Definition at line **24** of file **security-cli.h**.

Function Documentation

void changeKeyCommand (void)

void keyUpdateCommand (void)

void cbkeStartCommand (void)

void cbkeInterPanCommand (void)

void cbkePartnerCommand (void)

void printKeyInfo (void)

Variable Documentation

EmberCommandEntry cbkeCommands[]

EmberCommandEntry changeKeyCommands[]

EmberKeyData cliPreconfiguredLinkKey

EmberKeyData cliNetworkKey

---

simple-metering-client-callback.h File Reference

Go to the source code of this file.

Functions

int16u	**emberAfPluginSimpleMeteringClientRequestMirrorCallback** (EmberEUI64 requestingDeviceIeeeAddress)
int16u	**emberAfPluginSimpleMeteringClientRemoveMirrorCallback** (EmberEUI64 requestingDeviceIeeeAddress)

simple-metering-test.h File Reference

Go to the source code of this file.

Defines

#define **PROFILE_INTERVAL_PERIOD_TIMEFRAME**
#define **PROFILE_INTERVAL_PERIOD_IN_MINUTES**
#define **PROFILE_INTERVAL_PERIOD_IN_SECONDS**
#define **PROFILE_INTERVAL_PERIOD_IN_MILLISECONDS**
#define **MAX_PROFILE_INDEX**
#define **TOTAL_PROFILE_TIME_SPAN_IN_SECONDS**

Functions

void	**emAfTestMeterTick** (int8u endpoint)
void	**emAfTestMeterInit** (int8u endpoint)
void	**afTestMeterPrint** (void)
void	**afTestMeterSetConsumptionRate** (int16u rate)
void	**afTestMeterSetConsumptionVariance** (int16u variance)
void	**afTestMeterAdjust** (int8u endpoint)
void	**afTestMeterMode** (int8u endpoint, int8u electric)
void	**afTestMeterSetError** (int8u endpoint, int8u error)
void	**afTestMeterRandomError** (int8u changeIn256)
void	**afTestMeterEnableProfiles** (int8u enable)
boolean	**emAfTestMeterGetProfiles** (int8u intervalChannel, int32u endTime, int8u numberOfPeriods)

Define Documentation

#define PROFILE_INTERVAL_PERIOD_TIMEFRAME

Definition at line **13** of file **simple-metering-test.h**.

#define PROFILE_INTERVAL_PERIOD_IN_MINUTES

Definition at line **14** of file **simple-metering-test.h**.

#define PROFILE_INTERVAL_PERIOD_IN_SECONDS

Definition at line **15** of file **simple-metering-test.h**.

#define PROFILE_INTERVAL_PERIOD_IN_MILLISECONDS

Definition at line **16** of file **simple-metering-test.h**.

#define MAX_PROFILE_INDEX

Definition at line **17** of file **simple-metering-test.h**.

#define TOTAL_PROFILE_TIME_SPAN_IN_SECONDS

Definition at line **18** of file **simple-metering-test.h**.

Function Documentation

void emAfTestMeterTick (int8u endpoint)

void emAfTestMeterInit (int8u endpoint)

void afTestMeterPrint (void)

void afTestMeterSetConsumptionRate (int16u rate)

void afTestMeterSetConsumptionVariance (int16u variance)

void afTestMeterAdjust (int8u endpoint)

```
void afTestMeterMode ( int8u  endpoint,
                       int8u  electric
                     )
```

```
void afTestMeterSetError ( int8u  endpoint,
                           int8u  error
                         )
```

```
void afTestMeterRandomError ( int8u  changeIn256 )
```

```
void afTestMeterEnableProfiles ( int8u  enable )
```

```
boolean emAfTestMeterGetProfiles ( int8u   intervalChannel,
                                   int32u  endTime,
                                   int8u   numberOfPeriods
                                 )
```

---

smart-energy-registration.h File Reference

Go to the source code of this file.

Defines

#define	EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_ERROR_LIMIT
#define	EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_INITIAL
#define	EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_RETRY
#define	EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_BRIEF
#define	EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_RESUME
#define	EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_TRANSITION
#define	EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_NONE

Functions

int8u	emAfPluginSmartEnergyRegistrationTrustCenterKeyEstablishmentEndpoint (void)
void	emAfPluginSmartEnergyRegistrationTick (void)

Variables

EmberEventControl	emAfPluginSmartEnergyRegistrationEventControl

---

Define Documentation

#define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_ERROR_LIMIT

Definition at line 8 of file smart-energy-registration.h.

#define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_INITIAL

Definition at line 11 of file smart-energy-registration.h.

#define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_RETRY

Definition at line 12 of file smart-energy-registration.h.

#define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_BRIEF

Definition at line 13 of file smart-energy-registration.h.

#define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_RESUME

Definition at line **14** of file **smart-energy-registration.h**.

#define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_TRANSITION

Definition at line **15** of file **smart-energy-registration.h**.

#define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_NONE

Definition at line **16** of file **smart-energy-registration.h**.

Function Documentation

int8u emAfPluginSmartEnergyRegistrationTrustCenterKeyEstablishmentEndpoint (void)

void emAfPluginSmartEnergyRegistrationTick (void)

Variable Documentation

EmberEventControl emAfPluginSmartEnergyRegistrationEventControl

---

stack-handlers.h File Reference

Go to the source code of this file.

Defines

```
#define EMBER_APPLICATION_HAS_POLL_COMPLETE_HANDLER
```

Define Documentation

#define EMBER_APPLICATION_HAS_POLL_COMPLETE_HANDLER

Definition at line 31 of file stack-handlers.h.

temperature-measurement.h File Reference

Go to the source code of this file.

---

test-callback.h File Reference

Go to the source code of this file.

Functions

boolean **emberAfPluginTestExampleCustomerFunctionCallback** (int8u arg1, int8u *arg2, boolean arg3)

thermostat.h File Reference

Go to the source code of this file.

Functions

EmberAfStatus emAfSetLocalTemperature (int8u endpoint, int16s localTemperature)

Function Documentation

```
EmberAfStatus emAfSetLocalTemperature ( int8u   endpoint,
                                        int16s  localTemperature
                                      )
```

time-client.h File Reference

Go to the source code of this file.

Functions

int32u	**emAfTimeClusterClientGetCurrentTime** (void)
void	**emAfTimeClusterClientSetCurrentTime** (int32u utcTime)

Function Documentation

int32u emAfTimeClusterClientGetCurrentTime (void)

void emAfTimeClusterClientSetCurrentTime (int32u utcTime)

time-server.h File Reference

Go to the source code of this file.

Functions

int32u	**emAfTimeClusterServerGetCurrentTime** (void)
void	**emAfTimeClusterServerSetCurrentTime** (int32u utcTime)

Function Documentation

int32u emAfTimeClusterServerGetCurrentTime (void)

void emAfTimeClusterServerSetCurrentTime (int32u utcTime)

trust-center-keepalive.h File Reference

Go to the source code of this file.

Defines

```
#define EMBER_AF_PLUGIN_TRUST_CENTER_KEEPALIVE_DELAY_INTERVAL
#define EMBER_AF_PLUGIN_TRUST_CENTER_KEEPALIVE_FAILURE_LIMIT
```

Functions

```
void emAfPluginTrustCenterKeepaliveReadAttributesResponseCallback (int8u *buffer,
    int16u bufLen)
```

Define Documentation

#define EMBER_AF_PLUGIN_TRUST_CENTER_KEEPALIVE_DELAY_INTERVAL

Definition at line **11** of file **trust-center-keepalive.h**.

#define EMBER_AF_PLUGIN_TRUST_CENTER_KEEPALIVE_FAILURE_LIMIT

Definition at line **18** of file **trust-center-keepalive.h**.

Function Documentation

```
void emAfPluginTrustCenterKeepaliveReadAttributesResponseCallback ( int8u *  buffer,
                                                                    int16u   bufLen
                                                                  )
```

tunneling-client-callback.h File Reference

Go to the source code of this file.

Functions

void	**emberAfPluginTunnelingClientTunnelOpenedCallback** (int16u tunnelId)
void	**emberAfPluginTunnelingClientDataReceivedCallback** (int16u tunnelId, int8u *data, int16u dataLen)
void	**emberAfPluginTunnelingClientDataErrorCallback** (int16u tunnelId, **EmberAfTunnelingTransferDataStatus** transferDataStatus)

tunneling-client.h File Reference

Go to the source code of this file.

Defines

#define **EMBER_AF_PLUGIN_TUNNELING_CLIENT_TUNNEL_LIMIT**

Functions

EmberAfStatus emberAfPluginTunnelingClientTransferData (int16u tunnelId, int8u *data, int16u dataLen)
EmberAfStatus emberAfPluginTunnelingClientCloseTunnel (int16u tunnelId)

Define Documentation

#define EMBER_AF_PLUGIN_TUNNELING_CLIENT_TUNNEL_LIMIT

Definition at line **9** of file **tunneling-client.h**.

Function Documentation

EmberAfStatus emberAfPluginTunnelingClientTransferData (int16u tunnelId, int8u * data, int16u dataLen)

Transfer data to a server through a Tunneling cluster tunnel.

This function can be used to transfer data to a server through a tunnel. The Tunneling client plugin will send the data to the endpoint on the node that is managing the given tunnel.

Parameters:

tunnelId The identifier of the tunnel through which to send the data.

data Buffer containing the raw octets of the data.
dataLen The length in octets of the data.

Returns:
EMBER_ZCL_STATUS_SUCCESS if the data was sent,
EMBER_ZCL_STATUS_FAILURE if an error occurred, or
EMBER_ZCL_STATUS_NOT_FOUND if the tunnel does not exist.

EmberAfStatus emberAfPluginTunnelingClientCloseTunnel (int16u tunnelId)

Close a Tunneling cluster tunnel.

This function can be used to close a tunnel. The Tunneling client plugin will send the close command to the endpoint on the node that is managing the given tunnel.

Parameters:
tunnelId The identifier of the tunnel to close.

Returns:
EMBER_ZCL_STATUS_SUCCESS if the close request was sent,
EMBER_ZCL_STATUS_FAILURE if an error occurred, or
EMBER_ZCL_STATUS_NOT_FOUND if the tunnel does not exist.

---

tunneling-server-callback.h File Reference

Go to the source code of this file.

Functions

boolean	**emberAfPluginTunnelingServerIsProtocolSupportedCallback** (int8u protocolId, int16u manufacturerCode)
void	**emberAfPluginTunnelingServerTunnelOpenedCallback** (int16u tunnelId, int8u protocolId, int16u manufacturerCode)
void	**emberAfPluginTunnelingServerDataReceivedCallback** (int16u tunnelId, int8u *data, int16u dataLen)
void	**emberAfPluginTunnelingServerDataErrorCallback** (int16u tunnelId, **EmberAfTunnelingTransferDataStatus** transferDataStatus)
void	**emberAfPluginTunnelingServerTunnelClosedCallback** (int16u tunnelId, boolean clientInitiated)

tunneling-server.h File Reference

Go to the source code of this file.

Defines

#define ZCL_TUNNELING_CLUSTER_INVALID_TUNNEL_ID
#define CLOSE_INITIATED_BY_CLIENT
#define CLOSE_INITIATED_BY_SERVER
#define EMBER_AF_PLUGIN_TUNNELING_SERVER_TUNNEL_LIMIT

Functions

EmberAfStatus emberAfPluginTunnelingServerTransferData (int16u tunnelId, int8u *data, int16u dataLen)

Define Documentation

#define ZCL_TUNNELING_CLUSTER_INVALID_TUNNEL_ID

Definition at line 8 of file tunneling-server.h.

#define CLOSE_INITIATED_BY_CLIENT

Definition at line 9 of file tunneling-server.h.

#define CLOSE_INITIATED_BY_SERVER

Definition at line 10 of file tunneling-server.h.

#define EMBER_AF_PLUGIN_TUNNELING_SERVER_TUNNEL_LIMIT

Definition at line 13 of file tunneling-server.h.

Function Documentation

```
EmberAfStatus emberAfPluginTunnelingServerTransferData ( int16u  tunnelId,
                                                         int8u * data,
                                                         int16u  dataLen
                                                       )
```

Transfer data to a client through a Tunneling cluster tunnel.

This function can be used to transfer data to a client through a tunnel. The Tunneling server plugin will send the data to the endpoint on the node that opened the given tunnel.

Parameters:

tunnelId	The identifier of the tunnel through which to send the data.
data	Buffer containing the raw octets of the data.
dataLen	The length in octets of the data.

Returns:

EMBER_ZCL_STATUS_SUCCESS if the data was sent,
EMBER_ZCL_STATUS_FAILURE if an error occurred, or
EMBER_ZCL_STATUS_NOT_FOUND if the tunnel does not exist.

---

zcl-cli.h File Reference

Go to the source code of this file.

Functions

void	**emAfCliSendCommand** (void)
void	**emAfCliBsendCommand** (void)
void	**emAfCliReadCommand** (void)
void	**emAfCliWriteCommand** (void)
void	**emAfCliTimesyncCommand** (void)
void	**emAfCliRawCommand** (void)
void	**emAfCliClearEventsCommand** (void)
void	**emAfCliAddReportEntryCommand** (void)

Variables

EmberCommandEntry	**optionCommands** []
EmberCommandEntry	**keysCommands** []
EmberCommandEntry	**interpanCommands** []
EmberCommandEntry	**printCommands** []
EmberCommandEntry	**zclCommands** []
EmberCommandEntry	**certificationCommands** []

Function Documentation

void emAfCliSendCommand (void)

void emAfCliBsendCommand (void)

void emAfCliReadCommand (void)

void emAfCliWriteCommand (void)

void emAfCliTimesyncCommand (void)

void emAfCliRawCommand (void)

void emAfCliClearEventsCommand (void)

void emAfCliAddReportEntryCommand (void)

Variable Documentation

EmberCommandEntry optionCommands[]

EmberCommandEntry keysCommands[]

EmberCommandEntry interpanCommands[]

EmberCommandEntry printCommands[]

EmberCommandEntry zclCommands[]

EmberCommandEntry certificationCommands[]

---

zdo-cli.h File Reference

Go to the source code of this file.

Defines

#define **ZDO_COMMANDS**

Variables

EmberCommandEntry **zdoCommands** []

Define Documentation

#define ZDO_COMMANDS

Definition at line **16** of file **zdo-cli.h**.

Variable Documentation

EmberCommandEntry zdoCommands[]

Directories

This directory hierarchy is sorted roughly, but not completely, alphabetically:

- app
  - framework
    - cli
    - gen
    - include
    - plugin
      - bacnet-protocol-tunnel
      - basic
        - gen
      - color-control
      - drlc
        - gen
      - drlc-server
      - generic-tunnel
      - groups-server
      - ias-ace-zone
      - ias-wd-server
      - identify
      - interpan
      - key-establishment
      - level-control
      - messaging-client
        - gen
      - messaging-server
      - on-off
      - ota-bootload
      - ota-client
      - ota-client-policy
      - ota-common
      - ota-server
      - ota-server-policy
      - ota-storage-common
      - ota-storage-posix-filesystem
      - ota-storage-simple
      - ota-storage-simple-ram
      - partner-link-key-exchange
      - price-client
        - gen

- price-server
- psl-commissioning
  - gen
- scenes
- simple-metering-client
  - gen
- simple-metering-server
- smart-energy-registration
- temperature-measurement
- test
  - gen
- thermostat
- time-client
- time-server
- trust-center-keepalive
- tunneling-client
  - gen
- tunneling-server
  - gen

- util

---

app

app Directory Reference

Directories

directory **framework**

framework Directory Reference

Directories

directory	cli
directory	gen
directory	include
directory	plugin
directory	util

cli Directory Reference

Files

file	**core-cli.h** [code]
file	**custom-cli.h** [code]
file	**network-cli.h** [code]
file	**ota-cli.h** [code]
file	**plugin-cli.h** [code]
file	**security-cli.h** [code]
file	**zcl-cli.h** [code]
file	**zdo-cli.h** [code]

gen Directory Reference

Files

file	**af-structs.h** [code]
file	**attribute-id.h** [code]
file	**attribute-size.h** [code]
file	**attribute-storage.h** [code]
file	**attribute-type.h** [code]
file	**call-command-handler.h** [code]
file	**callback.h** [code]
file	**client-command-macro.h** [code]
file	**cluster-id.h** [code]
file	**command-id.h** [code]
file	**debug-printing-test.h** [code]
file	**debug-printing.h** [code]
file	**debug.h** [code]
file	**enums.h** [code]
file	**print-cluster.h** [code]
file	**stack-handlers.h** [code]

include Directory Reference

Files

file	**af-types.h** [code]
file	**af.h** [code]

plugin Directory Reference

Directories

directory	bacnet-protocol-tunnel
directory	basic
directory	color-control
directory	drlc
directory	drlc-server
directory	generic-tunnel
directory	groups-server
directory	ias-ace-zone
directory	ias-wd-server
directory	identify
directory	interpan
directory	key-establishment
directory	level-control
directory	messaging-client
directory	messaging-server
directory	on-off
directory	ota-bootload
directory	ota-client
directory	ota-client-policy
directory	ota-common
directory	ota-server
directory	ota-server-policy
directory	ota-storage-common
directory	ota-storage-posix-filesystem
directory	ota-storage-simple
directory	ota-storage-simple-ram
directory	partner-link-key-exchange
directory	price-client
directory	price-server
directory	psl-commissioning
directory	scenes
directory	simple-metering-client
directory	simple-metering-server
directory	smart-energy-registration

directory	**temperature-measurement**
directory	**test**
directory	**thermostat**
directory	**time-client**
directory	**time-server**
directory	**trust-center-keepalive**
directory	**tunneling-client**
directory	**tunneling-server**

bacnet-protocol-tunnel Directory Reference

Files

file	**bacnet-protocol-tunnel.h** [code]

basic Directory Reference

Directories

directory	gen

Files

file	basic.h [code]

gen Directory Reference

Files

file	**basic-callback.h** [code]

color-control Directory Reference

Files

file	color-control.h [code]

drlc Directory Reference

Directories

directory	gen

Files

file	demand-response-load-control.h [code]
file	load-control-event-table.h [code]

gen Directory Reference

Files

file	drlc-callback.h [code]

drlc-server Directory Reference

Files

file **drlc-server.h** [code]

generic-tunnel Directory Reference

Files

file	generic-tunnel.h [code]

groups-server Directory Reference

Files

file	groups-server.h [code]

ias-ace-zone Directory Reference

Files

file	ias-ace-server.h [code]
file	ias-zone-table.h [code]

ias-wd-server Directory Reference

Files

file **ias-wd-server.h** [code]

identify Directory Reference

Files

file	identify.h [code]

interpan Directory Reference

Files

file	interpan.h [code]

key-establishment Directory Reference

Files

file	key-establishment-storage.h [code]
file	key-establishment.h [code]

level-control Directory Reference

Files

file	**level-control.h** [code]

messaging-client Directory Reference

Directories

directory	gen

Files

file	messaging-client.h [code]

gen Directory Reference

Files

file	**messaging-client-callback.h** [code]

messaging-server Directory Reference

Files

file	messaging-server.h [code]

on-off Directory Reference

Files

file	on-off.h [code]

ota-bootload Directory Reference

Files

file	ota-bootload-ncp.h [code]
file	ota-bootload-xmodem.h [code]

ota-client Directory Reference

Files

file	**ota-client-page-request-test.h** [code]
file	**ota-client-page-request.h** [code]
file	**ota-client-signature-verify.h** [code]
file	**ota-client.h** [code]

ota-client-policy Directory Reference

Files

file	ota-client-policy.h [code]

ota-common Directory Reference

Files

file	**ota.h** [code]

ota-server Directory Reference

Files

file	ota-server.h [code]

ota-server-policy Directory Reference

Files

file	ota-server-policy.h [code]

ota-storage-common Directory Reference

Files

file	ota-storage-simple-custom.h [code]
file	ota-storage.h [code]

ota-storage-posix-filesystem Directory Reference

Files

file	ota-storage-linux.h [code]

ota-storage-simple Directory Reference

Files

file	ota-storage-simple-driver.h [code]

ota-storage-simple-ram Directory Reference

Files

file	ota-static-file-data.h [code]

partner-link-key-exchange Directory Reference

Files

file	partner-link-key-exchange.h [code]

price-client Directory Reference

Directories

directory	gen

Files

file	price-client.h [code]

gen Directory Reference

Files

file **price-client-callback.h** [code]

price-server Directory Reference

Files

file	**price-server.h** [code]

psl-commissioning Directory Reference

Directories

directory	gen

Files

file	psl-commissioning.h [code]

gen Directory Reference

Files

file	psl-commissioning-callback.h [code]

scenes Directory Reference

Files

file	**scenes.h** [code]

simple-metering-client Directory Reference

Directories

directory gen

gen Directory Reference

Files

file	simple-metering-client-callback.h [code]

simple-metering-server Directory Reference

Files

file **simple-metering-test.h** [code]

smart-energy-registration Directory Reference

Files

file **smart-energy-registration.h** [code]

temperature-measurement Directory Reference

Files

file **temperature-measurement.h** [code]

test Directory Reference

Directories

directory gen

gen Directory Reference

Files

file	test-callback.h [code]

thermostat Directory Reference

Files

file	**thermostat.h** [code]

time-client Directory Reference

Files

file	time-client.h [code]

time-server Directory Reference

Files

file	time-server.h [code]

trust-center-keepalive Directory Reference

Files

file	trust-center-keepalive.h [code]

tunneling-client Directory Reference

Directories

directory	gen

Files

file	tunneling-client.h [code]

gen Directory Reference

Files

file	tunneling-client-callback.h [code]

tunneling-server Directory Reference

Directories

directory	gen

Files

file	tunneling-server.h [code]

gen Directory Reference

Files

file	tunneling-server-callback.h [code]

util Directory Reference

Files

file	client-api.h [code]

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- _ -

- __PRINTCLUSTER_ALARM_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_BACNET_PROTOCOL_TUNNEL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_BALLAST_CONFIGURATION_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_BASIC_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_BINARY_INPUT_BASIC_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_COLOR_CONTROL_CLUSTER : **print-cluster.h**

- __PRINTCLUSTER_COMMISSIONING_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_DEHUMID_CONTROL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_DEVICE_TEMP_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_DOOR_LOCK_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_FAN_CONTROL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_FLOW_MEASUREMENT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_GENERIC_TUNNEL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_GROUPS_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_IAS_ACE_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_IAS_WD_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_IAS_ZONE_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_IDENTIFY_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_ILLUM_LEVEL_SENSING_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_ILLUM_MEASUREMENT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_KEY_ESTABLISHMENT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_LEVEL_CONTROL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_MESSAGING_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_OCCUPANCY_SENSING_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_ON_OFF_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_ON_OFF_SWITCH_CONFIG_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_OTA_BOOTLOAD_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_POWER_CONFIG_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_PREPAYMENT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_PRESSURE_MEASUREMENT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_PRICE_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_PUMP_CONFIG_CONTROL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_RSSI_LOCATION_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_SAMPLE_MFG_SPECIFIC_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_SCENES_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_SHADE_CONFIG_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_SIMPLE_METERING_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_TEMP_MEASUREMENT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_THERMOSTAT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_THERMOSTAT_UI_CONFIG_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_TIME_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_TUNNELING_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_WINDOW_COVERING_CLUSTER : **print-cluster.h**

---

app » framework » include

af.h

Go to the documentation of this file.

```
00001
00020 #ifndef __AF_API__
00021 #define __AF_API__
00022
00023 #ifndef CONFIGURATION_HEADER
00024   #define CONFIGURATION_HEADER "app/framework/util/config.h"
00025 #endif
00026 #include CONFIGURATION_HEADER
00027
00028 // Micro and compiler specific typedefs and macros
00029 #include PLATFORM_HEADER
00030
00031 #ifdef EZSP_HOST
00032   // Includes needed for ember related functions for the EZSP host
00033   #include "stack/include/error.h"
00034   #include "stack/include/ember-types.h"
00035   #include "app/util/ezsp/ezsp-protocol.h"
00036   #include "app/util/ezsp/ezsp.h"
00037   #include "app/util/ezsp/ezsp-utils.h"
00038   #include "app/util/ezsp/serial-interface.h"
00039 #else
00040   // Includes needed for ember related functions for the EM250
00041   #include "stack/include/ember.h"
00042 #endif // EZSP_HOST
00043
00044 // HAL - hardware abstraction layer
00045 #include "hal/hal.h"
00046 #include "app/util/serial/serial.h"  // Serial utility APIs
00047
00048 #include "stack/include/event.h"
00049 #include "stack/include/error.h"
00050 #include "af-types.h"
00051 #include "app/framework/util/print.h"
00052 #include "app/framework/gen/af-structs.h"
00053 #include "app/framework/gen/attribute-id.h"
00054 #include "app/framework/gen/attribute-storage.h"
00055 #include "app/framework/gen/attribute-type.h"
00056 #include "app/framework/gen/call-command-handler.h"
00057 #include "app/framework/gen/callback.h"
00058 #include "app/framework/gen/client-command-macro.h"
00059 #include "app/framework/gen/cluster-id.h"
00060 #include "app/framework/gen/command-id.h"
00061 #include "app/framework/gen/debug-printing.h"
00062 #include "app/framework/gen/enums.h"
00063 #include "app/framework/gen/print-cluster.h"
00064 #include "app/framework/gen/stack-handlers.h"
00065 #include "app/framework/util/client-api.h"
00066
00067
00069 // @{
```

```
EmberAfAttributeMetadata *emberAfLocateAttributeMetadata(int8u endpoint,
                                                         EmberAfClusterId cluster,
                                                         EmberAfAttributeId attribute,
                                                         int8u mask,
                                                         int16u manufacturerCode);

boolean emberAfContainsCluster(int8u endpoint, EmberAfClusterId clusterId);

boolean emberAfContainsServer(int8u endpoint, EmberAfClusterId clusterId);

boolean emberAfContainsClient(int8u endpoint, EmberAfClusterId clusterId);


boolean emberAfReadOrUpdateAttribute(EmberAfAttributeSearchRecord *attRecord,
                                     EmberAfAttributeMetadata **metadata,
                                     int8u *buffer,
                                     boolean write);



EmberAfStatus emberAfWriteAttribute(int8u endpoint,
                                    EmberAfClusterId cluster,
                                    EmberAfAttributeId attributeID,
                                    int8u mask,
                                    int8u* dataPtr,
                                    int8u dataType);

EmberAfStatus emberAfWriteServerAttribute(int8u endpoint,
                                          EmberAfClusterId cluster,
                                          EmberAfAttributeId attributeID,
                                          int8u* dataPtr,
                                          int8u dataType);

EmberAfStatus emberAfWriteClientAttribute(int8u endpoint,
                                          EmberAfClusterId cluster,
                                          EmberAfAttributeId attributeID,
                                          int8u* dataPtr,
                                          int8u dataType);

EmberAfStatus emberAfWriteManufacturerSpecificServerAttribute(int8u endpoint,
                                                              EmberAfClusterId cluster,
                                                              EmberAfAttributeId attributeID,
                                                              int16u manufacturerCode,
                                                              int8u* dataPtr,
                                                              int8u dataType);

EmberAfStatus emberAfWriteManufacturerSpecificClientAttribute(int8u endpoint,
                                                              EmberAfClusterId cluster,
                                                              EmberAfAttributeId attributeID,
                                                              int16u manufacturerCode,
                                                              int8u* dataPtr,
                                                              int8u dataType);


```

```
EmberAfStatus emberAfVerifyAttributeWrite(int8u endpoint,
                                          EmberAfClusterId cluster,
                                          EmberAfAttributeId attributeID,
                                          int8u mask,
                                          int16u manufacturerCode,
                                          int8u* buffer,
                                          int8u dataType);

EmberAfStatus emberAfReadAttribute(int8u endpoint,
                                   EmberAfClusterId cluster,
                                   EmberAfAttributeId attributeID,
                                   int8u mask,
                                   int8u* dataPtr,
                                   int8u maxLength,
                                   int8u* dataType);

EmberAfStatus emberAfReadServerAttribute(int8u endpoint,
                                         EmberAfClusterId cluster,
                                         EmberAfAttributeId attributeID,
                                         int8u* dataPtr,
                                         int8u maxLength);

EmberAfStatus emberAfReadClientAttribute(int8u endpoint,
                                         EmberAfClusterId cluster,
                                         EmberAfAttributeId attributeID,
                                         int8u* dataPtr,
                                         int8u maxLength);

EmberAfStatus emberAfReadManufacturerSpecificServerAttribute(int8u endpoint,
                                                             EmberAfClusterId cluster,
                                                             EmberAfAttributeId attributeID,
                                                             int16u manufacturerCode,
                                                             int8u* dataPtr,
                                                             int8u maxLength);

EmberAfStatus emberAfReadManufacturerSpecificClientAttribute(int8u endpoint,
                                                             EmberAfClusterId cluster,
                                                             EmberAfAttributeId attributeID,
                                                             int16u manufacturerCode,
                                                             int8u* dataPtr,
                                                             int8u maxLength);


int8u emberAfGetDataSize(int8u dataType);

#define emberAfClusterIsManufacturerSpecific(cluster) ((cluster)->clusterId >= 0xFC00)


#define emberAfAttributeIsReadOnly(metadata) (((metadata)->mask & ATTRIBUTE_MASK_WRITABLE) == 0)

#define emberAfAttributeIsClient(metadata) (((metadata)->mask & ATTRIBUTE_MASK_CLIENT) != 0)

#define emberAfAttributeIsTokenized(metadata) (((metadata)->mask & ATTRIBUTE_MASK_TOKENIZE) != 0)

#define emberAfAttributeIsSingleton(metadata) (((metadata)->mask & ATTRIBUTE_MASK_SINGLETON) != 0)
```

```
00408
00414 #define emberAfAttributeIsManufacturerSpecific(metadata) (((metadata)->mask
& ATTRIBUTE_MASK_MANUFACTURER_SPECIFIC) != 0)
00415
00416
00422 #define emberAfAttributeSize(metadata) ((metadata)->size)
00423
00424 // master array of all defined endpoints
00425 extern EmberAfDefinedEndpoint endpoints[];
00426
00430 #define emberAfEndpointFromIndex(index) (endpoints[(index)].endpoint)
00431
00435 int8u emberAfIndexFromEndpoint(int8u endpoint);
00436
00440 #define emberAfProfileIdFromIndex(index) (endpoints[(index)].profileId)
00441
00445 #define emberAfDeviceIdFromIndex(index) (endpoints[(index)].deviceId)
00446
00450 #define emberAfDeviceVersionFromIndex(index) (endpoints[(index)].deviceVersion)
00451
00458 #define emberAfPrimaryProfileId()       emberAfProfileIdFromIndex(0)
00459
00463 #define emberAfPrimaryEndpoint() (endpoints[0].endpoint)
00464
00465
00469 int8u emberAfEndpointCount(void);
00470
00475 enum {
00476   EMBER_AF_DATA_TYPE_ANALOG     = 0,
00477   EMBER_AF_DATA_TYPE_DISCRETE   = 1,
00478   EMBER_AF_DATA_TYPE_NONE       = 2
00479 };
00483 int8u emberAfGetAttributeAnalogOrDiscreteType(int8u dataType);
00484
00488 boolean emberAfIsTypeSigned(int8u dataType);
00489
00493 int32u emberAfGetInt32u(const int8u* message, int16u currentIndex, int16u msgLen);
00494
00498 int32u emberAfGetInt24u(const int8u* message, int16u currentIndex, int16u msgLen);
00499
00503 int16u emberAfGetInt16u(const int8u* message, int16u currentIndex, int16u msgLen);
00507 int8u* emberAfGetString(int8u* message, int16u currentIndex, int16u msgLen);
00508
00512 #define emberAfGetInt8u(message, currentIndex, msgLen) message[currentIndex]
00513
00517 #define emberAfCopyInt8u(data,index,x) (data[index] = (x))
00518
00521 void emberAfCopyInt16u(int8u *data, int16u index, int16u x);
00525 void emberAfCopyInt24u(int8u *data, int16u index, int32u x);
00529 void emberAfCopyInt32u(int8u *data, int16u index, int32u x);
00530 /*
00531  * @brief Function that copies a ZigBee string into a buffer.  The size
00532  * parameter should indicate the maximum number of characters to copy to the
00533  * destination buffer not including the length byte.
00534  */
```

```
00535 void emberAfCopyString(int8u *dest, int8u *src, int8u size);
00536 /*
00537  * @brief Function that determines the length of a ZigBee string.
00538  */
00539 int8u emberAfStringLength(const int8u *buffer);
00540
00544 // @{
00545
00554 boolean emberAfIsDeviceEnabled(int8u endpoint);
00555
00564 void emberAfSetDeviceEnabled(int8u endpoint, boolean enabled);
00565
00569 // @{
00570
00571
00575 #define EMBER_AF_NEW_IMAGE_VERIFICATION TRUE
00576
00581 #define EMBER_AF_CONTINUE_IMAGE_VERIFY  FALSE
00582
00583
00590 extern PGM EmberAfOtaImageId emberAfInvalidImageId;
00591
00592
00603 boolean emberAfIsThisDataTypeAStringType(int8u dataType);
00604
00612 int8u emberAfNextSequence(void);
00613
00621 int8s emberAfCompareValues(int8u* val1, int8u* val2, int8u len);
00622
00626 void emberAfGetEui64(EmberEUI64 returnEui64);
00627
00631 EmberNodeId emberAfGetNodeId(void);
00632
00636 // @{
00637
00638 // Guaranteed print
00643 #define emberAfGuaranteedPrint(...)   emberAfPrint(0xFFFF, __VA_ARGS__)
00644
00648 #define emberAfGuaranteedPrintln(...) emberAfPrintln(0xFFFF, __VA_ARGS__)
00649
00653 #define emberAfGuaranteedPrintBuffer(buffer,len,withSpace) emberAfPrintBuffer(0xFFFF,(buffer),(len),(withSpace))
00654
00658 #define emberAfGuaranteedPrintString(buffer) emberAfPrintString(0xFFFF,(buffer))
00659
00664 #define emberAfGuaranteedFlush()      emberAfFlush(0xFFFF)
00665
00669 boolean emberAfPrintEnabled(int16u functionality);
00670
00674 void emberAfPrintBuffer(int16u area, const int8u *buffer, int16u bufferLen, boolean withSpaces);
00675
00680 void emberAfPrintString(int16u area, const int8u *buffer);
00681
00687 void emberAfPrint(int16u functionality, PGM_P formatString, ...);
00688
00695 void emberAfPrintln(int16u functionality, PGM_P formatString, ...);
```

```
void emberAfFlush(int16u functionality);

void emberAfPrintOn(int16u functionality);

void emberAfPrintOff(int16u functionality);

void emberAfPrintAllOn(void);

void emberAfPrintAllOff(void);

void emberAfPrintStatus(void);

void emberAfPrintLittleEndianEui64(const EmberEUI64 eui64);


void emberAfPrintBigEndianEui64(const EmberEUI64 eui64);

void emberAfPrintMessageData(int8u* data, int16u length);


#define MILLISECOND_TICKS_PER_QUARTERSECOND (MILLISECOND_TICKS_PER_SECOND >> 2)
#define MILLISECOND_TICKS_PER_MINUTE (60L * MILLISECOND_TICKS_PER_SECOND)
#define MILLISECOND_TICKS_PER_HOUR   (60L * MILLISECOND_TICKS_PER_MINUTE)

extern int32u emberAfHibernateDuration;

#define EMBER_AF_SET_HIBERNATE_DURATION(newValue)       \
  emberAfHibernateDuration = newValue

#define EMBER_AF_SET_NAP_DURATION(newValue)     \
  emberAfNapDuration = newValue

enum {
  // we may be able to remove these top two since they are
  // handled by the stack on the SOC.
  EMBER_AF_WAITING_FOR_DATA_ACK                 = 0x01, //not needed?
  EMBER_AF_LAST_POLL_GOT_DATA                   = 0x02, //not needed?
  EMBER_AF_WAITING_FOR_SERVICE_DISCOVERY        = 0x04,
  EMBER_AF_WAITING_FOR_ZDO_RESPONSE             = 0x08,
  EMBER_AF_WAITING_FOR_ZCL_RESPONSE             = 0x10,
  EMBER_AF_WAITING_FOR_REGISTRATION             = 0x20,
  EMBER_AF_WAITING_FOR_PARTNER_LINK_KEY_EXCHANGE = 0x40,
};

#ifndef EMBER_AF_WAKE_TIMEOUT
#define EMBER_AF_WAKE_TIMEOUT 4 // 1 second
#endif

#ifndef EMBER_AF_WAKE_TIMEOUT_MASK
#define EMBER_AF_WAKE_TIMEOUT_MASK (EMBER_AF_WAITING_FOR_ZDO_RESPONSE \
                                    | EMBER_AF_WAITING_FOR_ZCL_RESPONSE)
#endif

extern int32u emAfCurrentAppTasks;
```

```
00837
00841 void emberAfAddToCurrentAppTasks(int32u mask);
00842
00846 void emberAfRemoveFromCurrentAppTasks(int32u mask);
00847
00858 boolean emberAfOkToHibernate(void);
00859
00867 #define emberAfOkToNap() (emberAfGetCurrentSleepControl() < EMBER_AF_STAY_AWAKE)
00868
00874 #ifdef ZA_DEVICE_IS_SLEEPY
00875 #define emberAfCurrentAppTasks() (emAfCurrentAppTasks)
00876 #else
00877 #define emberAfCurrentAppTasks()
00878 #endif //ZA_DEVICE_IS_SLEEPY
00879
00886 void emberAfRunEvents(void);
00887
00900 boolean emberAfStartMove(void);
00901
00906 #define EMBER_AF_CLIENT_CLUSTER_TICK TRUE
00907
00912 #define EMBER_AF_SERVER_CLUSTER_TICK FALSE
00913
00941 EmberStatus emberAfScheduleClusterTick(int8u endpoint,
00942                                        int16u clusterId,
00943                                        boolean isClient,
00944                                        int32u timeMs,
00945                                        EmberAfEventSleepControl sleepControl);
00946
00964 EmberStatus emberAfDeactivateClusterTick(int8u endpoint,
00965                                          int16u clusterId,
00966                                          boolean isClient);
00967
00990 EmberStatus emberAfEventControlSetDelay(EmberEventControl *eventControl, int32u timeMs);
00991
00992
01002 int32u emberAfMsToNextEvent(int32u maxMs);
01003
01014 #define emberAfQSToNextEvent(maxQS) (emberAfMsToNextEvent(maxQS * \
01015                                      (MILLISECOND_TICKS_PER_QUARTERSECOND)) \
01016                                      / (MILLISECOND_TICKS_PER_QUARTERSECOND))
01017
01032 EmberAfEventSleepControl emberAfGetCurrentSleepControl(void);
01033
01043 void  emberAfSetDefaultSleepControl(EmberAfEventSleepControl control);
01044
01053  extern EmberAfEventSleepControl emAfDefaultSleepControl;
01054  #define emberAfGetDefaultSleepControl() (emAfDefaultSleepControl)
01055
01060 // @{
01061
01074 int32u emberAfGetCurrentTime(void);
01075
01087 void emberAfSetTime(int32u utcTime);
01088
01089
```

```
void emberAfPrintTime(int32u utcTime);

// @{

extern boolean emberAfResponseWaitingFlag;

#define emberAfSetResponseWaiting() emberAfResponseWaitingFlag = TRUE

EmberStatus emberAfSendResponse(void);

EmberStatus emberAfSendRequest(void);

EmberStatus emberAfSendMulticast(EmberMulticastId multicastId,
                                 EmberApsFrame *apsFrame,
                                 int16u messageLength,
                                 int8u* message);

EmberStatus emberAfSendBroadcast(EmberNodeId destination,
                                 EmberApsFrame *apsFrame,
                                 int16u messageLength,
                                 int8u* message);

EmberStatus emberAfSendUnicast(EmberOutgoingMessageType type,
                               int16u indexOrDestination,
                               EmberApsFrame *apsFrame,
                               int16u messageLength,
                               int8u* message);

EmberStatus emberAfSendInterPan(EmberPanId panId,
                                const EmberEUI64 destinationLongId,
                                EmberNodeId destinationShortId,
                                EmberMulticastId multicastId,
                                EmberAfClusterId clusterId,
                                EmberAfProfileId profileId,
                                int16u messageLength,
                                int8u* messageBytes);

// DEPRECATED.
#define emberAfSendInterpanUnicast(destinationShortAddress,  \
                                   destinationPanId,         \
                                   destinationAppProfileId,  \
                                   destinationClusterId,     \
                                   multicastIdOrZero,        \
                                   longAddressOrNull,        \
                                   messageLength,            \
                                   messageBytes)             \
  emberAfSendInterPan(destinationPanId,                      \
                      longAddressOrNull,                     \
                      destinationShortAddress,               \
                      multicastIdOrZero,                     \
                      destinationClusterId,                  \
                      destinationAppProfileId,               \
                      messageLength,                         \
                      messageBytes)

EmberStatus emberAfSendEndDeviceBind(int8u endpoint);
```

```
01220
01228 EmberStatus emberAfSendCommandMulticast(EmberMulticastId multicastId);
01229
01237 EmberStatus emberAfSendCommandUnicast(EmberOutgoingMessageType type,
01238                                       int16u indexOrDestination);
01239
01246 EmberStatus emberAfSendCommandBroadcast(EmberNodeId destination);
01247
01258 EmberStatus emberAfSendCommandInterPan(EmberPanId panId,
01259                                        const EmberEUI64 destinationLongId,
01260                                        EmberNodeId destinationShortId,
01261                                        EmberMulticastId multicastId,
01262                                        EmberAfProfileId profileId);
01263
01264 // DEPRECATED.
01265 #define emberAfSendCommandInterpanUnicast(eui64,       \
01266                                           nodeId,      \
01267                                           panId,       \
01268                                           multicastId, \
01269                                           profileId)   \
01270   emberAfSendCommandInterPan(panId,                    \
01271                              eui64,                    \
01272                              nodeId,                   \
01273                              multicastId,              \
01274                              profileId)
01275
01287 EmberStatus emberAfSendDefaultResponse(const EmberAfClusterCommand *cmd,
01288                                        EmberAfStatus status);
01289
01306 int8u emberAfMaximumApsPayloadLength(EmberOutgoingMessageType type,
01307                                      int16u indexOrDestination,
01308                                      EmberApsFrame *apsFrame);
01309
01313 EmberApsFrame *emberAfGetCommandApsFrame(void);
01314
01320 #define EMBER_AF_CLIENT_CLUSTER_DISCOVERY FALSE
01321
01327 #define EMBER_AF_SERVER_CLUSTER_DISCOVERY TRUE
01328
01329 #ifdef DOXYGEN_SHOULD_SKIP_THIS
01330
01363 EmberStatus emberAfFindDevicesSupportingCluster(EmberNodeId target,
01364                                                 EmberAfClusterId clusterId,
01365                                                 boolean serverCluster,
01366                                                 EmberAfServiceDiscoveryCallback *callback);
01367 #else // doxygen
01368 #define emberAfFindDevicesSupportingCluster( target, clusterId, serverCluster, callback) \
01369  ( emberAfFindDevicesByProfileAndCluster( target, \
01370     emberAfPrimaryProfileId(),        \
01371     clusterId, \
01372     serverCluster, \
01373     callback) )
01374 #endif // doxygen
01375
01376
01402 EmberStatus emberAfFindDevicesByProfileAndCluster(EmberNodeId target,
```

```
                                    EmberAfProfileId profileId,
                                    EmberAfClusterId clusterId,
                                    boolean serverCluster,
                                    EmberAfServiceDiscoveryCallback *callback);


EmberStatus emberAfFindIeeeAddress(EmberNodeId shortAddress,
                                   EmberAfServiceDiscoveryCallback *callback);

EmberStatus emberAfFindNodeId(EmberEUI64 longAddress,
                              EmberAfServiceDiscoveryCallback *callback);

int8u emberAfAddAddressTableEntry(EmberEUI64 longId, EmberNodeId shortId);

EmberStatus emberAfSetAddressTableEntry(int8u index,
                                        EmberEUI64 longId,
                                        EmberNodeId shortId);

#define emberAfCurrentCommand() (emAfCurrentCommand)
extern EmberAfClusterCommand *emAfCurrentCommand;

#define emberAfCurrentEndpoint() (emberAfCurrentCommand()->apsFrame->destinationEndpoint)

#ifdef DOXYGEN_SHOULD_SKIP_THIS

EmberStatus emberAfInitiateKeyEstablishment(EmberNodeId nodeId, int8u endpoint);

EmberStatus emberAfInitiateInterPanKeyEstablishment(EmberPanId panId,
                                                    const EmberEUI64 eui64);

boolean emberAfPerformingKeyEstablishment(void);

EmberStatus emberAfInitiatePartnerLinkKeyExchange(EmberNodeId target,
                                                  int8u endpoint,
                                                  EmberAfPartnerLinkKeyExchangeCallback *callback);
#else
  #define emberAfInitiateKeyEstablishment(nodeId, endpoint) \
    emberAfInitiateKeyEstablishmentCallback(nodeId, endpoint)
  #define emberAfInitiateInterPanKeyEstablishment(panId, eui64) \
    emberAfInitiateInterPanKeyEstablishmentCallback(panId, eui64)
  #define emberAfPerformingKeyEstablishment() \
    emberAfPerformingKeyEstablishmentCallback()
  #define emberAfInitiatePartnerLinkKeyExchange(target, endpoint, callback) \
    emberAfInitiatePartnerLinkKeyExchangeCallback(target, endpoint, callback)
#endif

// @{
// Frame control fields (8 bits total)
// Bits 0 and 1 are Frame Type Sub-field
#define ZCL_FRAME_CONTROL_FRAME_TYPE_MASK     (BIT(0)|BIT(1))
#define ZCL_CLUSTER_SPECIFIC_COMMAND          BIT(0)
#define ZCL_PROFILE_WIDE_COMMAND              0
// Bit 2 is Manufacturer Specific Sub-field
#define ZCL_MANUFACTURER_SPECIFIC_MASK        BIT(2)
// Bit 3 is Direction Sub-field
```

```
#define ZCL_FRAME_CONTROL_DIRECTION_MASK      BIT(3)
#define ZCL_FRAME_CONTROL_SERVER_TO_CLIENT    BIT(3)
#define ZCL_FRAME_CONTROL_CLIENT_TO_SERVER    0
// Bit 4 is Disable Default Response Sub-field
#define ZCL_DISABLE_DEFAULT_RESPONSE_MASK     BIT(4)
// Bits 5 to 7 are reserved

// Packet must be at least 3 bytes for ZCL overhead.
//   Frame Control (1-byte)
//   Sequence Number (1-byte)
//   Command Id (1-byte)
#define EMBER_AF_ZCL_OVERHEAD                     3
#define EMBER_AF_ZCL_MANUFACTURER_SPECIFIC_OVERHEAD 5

// ${
void emberAfPollCompleteHandler(EmberStatus status);


void emberAfEzspPollCompleteHandler(EmberStatus status);
// ${

EmberStatus emberAfFormNetwork(EmberNetworkParameters *parameters);

EmberStatus emberAfJoinNetwork(EmberNetworkParameters *parameters);

#ifdef DOXYGEN_SHOULD_SKIP_THIS

EmberStatus emberAfFindUnusedPanIdAndForm(void);
EmberStatus emberAfStartSearchForJoinableNetwork(void);
#else
  #define emberAfFindUnusedPanIdAndForm()       emberAfFindUnusedPanIdAndFormCallback()
  #define emberAfStartSearchForJoinableNetwork() emberAfStartSearchForJoinableNetworkCallback()
#endif

#endif // __AF_API__
```

app » framework » include

af-types.h

Go to the documentation of this file.

```
#ifndef __AF_API_TYPES__
#define __AF_API_TYPES__

#include "app/framework/gen/enums.h"

typedef int16u EmberAfProfileId;

typedef int16u EmberAfAttributeId;

typedef int16u EmberAfClusterId;

typedef int8u EmberAfAttributeType;

typedef int8u EmberAfClusterMask;

typedef int8u EmberAfAttributeMask;

typedef void (*EmberAfGenericClusterFunction)(void);

#define EMBER_AF_NULL_MANUFACTURER_CODE 0x0000

typedef union {
  int8u *ptrToDefaultValue;
  int16u defaultValue;
} EmberAfDefaultAttributeValue;

typedef struct {
  EmberAfDefaultAttributeValue defaultValue;
  EmberAfDefaultAttributeValue minValue;
  EmberAfDefaultAttributeValue maxValue;
} EmberAfAttributeMinMaxValue;

typedef union {
  int8u *ptrToDefaultValue;
  int16u defaultValue;
  EmberAfAttributeMinMaxValue *ptrToMinMaxValue;
} EmberAfDefaultOrMinMaxAttributeValue;


typedef struct {
  EmberAfAttributeId attributeId;
  EmberAfAttributeType attributeType;
  int8u size;
  EmberAfAttributeMask mask;
  EmberAfDefaultOrMinMaxAttributeValue  defaultValue;
```

```
00158 } EmberAfAttributeMetadata;
00159
00163 typedef struct {
00167   EmberAfClusterId clusterId;
00171   EmberAfAttributeMetadata *attributes;
00175   int16u attributeCount;
00179   int16u clusterSize;
00184   EmberAfClusterMask mask;
00185
00191   const EmberAfGenericClusterFunction *functions;
00192
00193 } EmberAfCluster;
00194
00200 typedef struct {
00201
00205   int8u endpoint;
00206
00213   EmberAfClusterId clusterId;
00214
00220   EmberAfClusterMask clusterMask;
00221
00227   EmberAfAttributeId attributeId;
00228
00238   int16u manufacturerCode;
00239 } EmberAfAttributeSearchRecord;
00240
00245 typedef struct {
00246   int16u index;
00247   int16u manufacturerCode;
00248 } EmberAfManufacturerCodeEntry;
00249
00255 typedef struct {
00259   EmberIncomingMessageType type;
00263   EmberApsFrame* apsFrame;
00267   int8u* message;
00271   int16u msgLen;
00275   int16u source;
00280   int8u lastHopLqi;
00284   int8s lastHopRssi;
00289   int8u bindingTableIndex;
00295   int8u addressTableIndex;
00296 } EmberAfIncomingMessage;
00297
00298
00302 typedef int8u EmberAfInterpanMessageType;
00303 #define EMBER_AF_INTER_PAN_UNICAST   0x00
00304 #define EMBER_AF_INTER_PAN_BROADCAST 0x08
00305 #define EMBER_AF_INTER_PAN_MULTICAST 0x0C
00306
00307 // Legacy names
00308 #define INTER_PAN_UNICAST   EMBER_AF_INTER_PAN_UNICAST
00309 #define INTER_PAN_BROADCAST EMBER_AF_INTER_PAN_BROADCAST
```

```
00310 #define INTER_PAN_MULTICAST EMBER_AF_INTER_PAN_MULTICAST
00311
00312
00313 #define EMBER_AF_INTERPAN_OPTION_NONE              0x0000
00314 #define EMBER_AF_INTERPAN_OPTION_APS_ENCRYPT       0x0001
00315 #define EMBER_AF_INTERPAN_OPTION_MAC_HAS_LONG_ADDRESS 0x0002
00316
00320 typedef int16u EmberAfInterpanOptions;
00321
00326 typedef struct {
00327   EmberAfInterpanMessageType messageType;
00328
00334   EmberEUI64 longAddress;
00335   EmberNodeId shortAddress;
00336   EmberPanId panId;
00337
00341   EmberAfProfileId profileId;
00342   EmberAfClusterId clusterId;
00347   EmberMulticastId groupId;
00348   EmberAfInterpanOptions options;
00349 } EmberAfInterpanHeader;
00350
00351 // Legacy Name
00352 #define InterPanHeader EmberAfInterpanHeader
00353
00357 typedef int8u EmberAfAllowedInterpanOptions;
00358
00359 #define EMBER_AF_INTERPAN_DIRECTION_CLIENT_TO_SERVER 0x01
00360 #define EMBER_AF_INTERPAN_DIRECTION_SERVER_TO_CLIENT 0x02
00361 #define EMBER_AF_INTERPAN_DIRECTION_BOTH           0x03
00362 #define EMBER_AF_INTERPAN_GLOBAL_COMMAND           0x04
00363 #define EMBER_AF_INTERPAN_MANUFACTURER_SPECIFIC    0x08
00364
00369 typedef struct {
00370   EmberAfProfileId profileId;
00371   EmberAfClusterId clusterId;
00372   int8u commandId;
00373   EmberAfAllowedInterpanOptions options;
00374 } EmberAfAllowedInterPanMessage;
00375
00376
00385 typedef struct {
00389   EmberApsFrame           *apsFrame;
00390   EmberIncomingMessageType  type;
00391   EmberNodeId             source;
00392   int8u                   *buffer;
00393   int16u                  bufLen;
00394   boolean                 clusterSpecific;
00395   boolean                 mfgSpecific;
00396   int16u                  mfgCode;
00397   int8u                   seqNum;
```

```
00398   int8u                   commandId;
00399   int8u                   payloadStartIndex;
00400   int8u                   direction;
00401   EmberAfInterpanHeader    *interPanHeader;
00402 } EmberAfClusterCommand;
00403
00407 typedef struct {
00412   EmberAfCluster *cluster;
00416   int8u  clusterCount;
00420   int16u endpointSize;
00421 } EmberAfEndpointType;
00422
00426 typedef struct {
00430   int8u endpoint;
00434   EmberAfProfileId profileId;
00438   int16u deviceId;
00442   int8u deviceVersion;
00446   EmberAfEndpointType *endpointType;
00447 } EmberAfDefinedEndpoint;
00448
00449
00450 // Cluster specific types
00451
00458 typedef struct {
00459   int32u     eventId;
00460   EmberEUI64 source;
00461   int8u      sourceEndpoint;
00462   int8u      destinationEndpoint;
00463   int16u     deviceClass;
00464   int8u      utilityEnrollmentGroup;
00468   int32u     startTime;
00472   int16u     duration;
00473   int8u      criticalityLevel;
00474   int8u      coolingTempOffset;
00475   int8u      heatingTempOffset;
00476   int16s     coolingTempSetPoint;
00477   int16s     heatingTempSetPoint;
00478   int8s      avgLoadPercentage;
00479   int8u      dutyCycle;
00480   int8u      eventControl;
00481   int16u     startRand;
00482   int16u     endRand;
00483   int8u      optionControl;
00484 } EmberAfLoadControlEvent;
00485
00497 typedef enum {
00498   EMBER_AF_BROADCAST_SERVICE_DISCOVERY_COMPLETE            = 0x00,
00499   EMBER_AF_BROADCAST_SERVICE_DISCOVERY_RESPONSE_RECEIVED    = 0x01,
00500   EMBER_AF_UNICAST_SERVICE_DISCOVERY_TIMEOUT                = 0x02,
00501   EMBER_AF_UNICAST_SERVICE_DISCOVERY_COMPLETE_WITH_RESPONSE = 0x03,
00502 } EmberAfServiceDiscoveryStatus;
```

```
#define EM_AF_DISCOVERY_RESPONSE_MASK (0x01)

#define emberAfHaveDiscoveryResponseStatus(status)  ((status) & EM_AF_DISCOVERY_RESPONSE_MASK)

typedef struct {
  EmberAfServiceDiscoveryStatus status;

  int16u zdoRequestClusterId;

  EmberNodeId matchAddress;

  const void* responseData;
} EmberAfServiceDiscoveryResult;

typedef struct {
  int8u count;
  const int8u* list;
} EmberAfEndpointList;

typedef void (EmberAfServiceDiscoveryCallback)(const EmberAfServiceDiscoveryResult* result);

typedef void (EmberAfPartnerLinkKeyExchangeCallback)(boolean success);

typedef enum {
  EMBER_AF_OK_TO_HIBERNATE,
  EMBER_AF_OK_TO_NAP,
  EMBER_AF_STAY_AWAKE
} EmberAfEventSleepControl;

typedef struct {
  int8u endpoint;
  int16u clusterId;
  boolean isClient;
  EmberAfEventSleepControl sleepControl;
  EmberEventControl *eventControl;
} EmberAfEventContext;

typedef struct {
  boolean inUse;
  EmberAfReportingDirection direction;
  EmberNodeId nodeId;
  int8u sourceEndpoint;
  int8u destinationEndpoint;
  EmberAfClusterId clusterId;
  EmberAfAttributeId attributeId;
  int8u mask;
  int16u manufacturerCode;
  int16u minInterval;
  int16u maxInterval;
  int16u currentInterval;
```

```
00705   int32u reportableChange;
00710   int32u lastReportValue;
00715   int16u timeout;
00720   int8u failCount;
00721 } EmberAfReportEntry;
00722
00723 #ifdef EMBER_AF_PLUGIN_GROUPS_SERVER
00724
00727 #define EMBER_AF_GROUP_TABLE_NULL_INDEX 0xFF
00728
00732 #define EMBER_AF_GROUP_TABLE_UNUSED_ENDPOINT_ID 0x00
00733
00736 #define ZCL_GROUPS_CLUSTER_MAXIMUM_NAME_LENGTH 16
00737
00742 typedef struct {
00743   int8u  endpoint; // 0x00 when not in use
00744   int16u groupId;
00745   int8u  bindingIndex;
00746 #ifdef EMBER_AF_PLUGIN_GROUPS_SERVER_NAME_SUPPORT
00747   int8u  name[ZCL_GROUPS_CLUSTER_MAXIMUM_NAME_LENGTH + 1];
00748 #endif
00749 } EmberAfGroupTableEntry;
00750 #endif //EMBER_AF_PLUGIN_GROUPS_SERVER
00751
00755 #define EMBER_AF_SCENE_TABLE_NULL_INDEX 0xFF
00756
00760 #define EMBER_AF_SCENE_TABLE_UNUSED_ENDPOINT_ID 0x00
00761
00764 #define ZCL_SCENES_CLUSTER_MAXIMUM_NAME_LENGTH 16
00765
00768 #define ZCL_SCENES_GLOBAL_SCENE_GROUP_ID 0x0000
00769
00774 typedef struct {
00775   int8u   endpoint;              // 0x00 when this record is not in use
00776   int16u  groupId;               // 0x0000 if not associated with a group
00777   int8u   sceneId;
00778 #ifdef EMBER_AF_PLUGIN_SCENES_NAME_SUPPORT
00779   int8u   name[ZCL_SCENES_CLUSTER_MAXIMUM_NAME_LENGTH + 1];
00780 #endif
00781   int16u  transitionTime;        // in seconds
00782 #ifdef ZCL_USING_ON_OFF_CLUSTER_SERVER
00783   boolean hasOnOffValue;
00784   boolean onOffValue;
00785 #endif
00786 #ifdef ZCL_USING_LEVEL_CONTROL_CLUSTER_SERVER
00787   boolean hasCurrentLevelValue;
00788   int8u   currentLevelValue;
00789 #endif
00790 #ifdef ZCL_USING_THERMOSTAT_CLUSTER_SERVER
00791   boolean hasOccupiedCoolingSetpointValue;
00792   int16s  occupiedCoolingSetpointValue;
00793   boolean hasOccupiedHeatingSetpointValue;
00794   int16s  occupiedHeatingSetpointValue;
00795   boolean hasSystemModeValue;
```

```
  int8u   systemModeValue;
#endif
#ifdef ZCL_USING_COLOR_CONTROL_CLUSTER_SERVER
  boolean hasCurrentXValue;
  int16u  currentXValue;
  boolean hasCurrentYValue;
  int16u  currentYValue;
#endif
#ifdef ZCL_USING_DOOR_LOCK_CLUSTER_SERVER
  boolean hasLockStateValue;
  int8u   lockStateValue;
#endif
#ifdef ZCL_USING_WINDOW_COVERING_CLUSTER_SERVER
  boolean hasCurrentPositionLiftPercentageValue;
  int8u   currentPositionLiftPercentageValue;
  boolean hasCurrentPositionTiltPercentageValue;
  int8u   currentPositionTiltPercentageValue;
#endif
} EmberAfSceneTableEntry;

#ifdef EMBER_AF_PLUGIN_MESSAGING_CLIENT
typedef struct {
  boolean    active;
  EmberEUI64 server;
  int8u      serverEndpoint;
  int8u      clientEndpoint;
  int32u     messageId;
  int8u      messageControl;
  int32u     startTime;
  int32u     endTime;
  int16u     durationInMinutes;
  int8u      message[EMBER_AF_PLUGIN_MESSAGING_CLIENT_MESSAGE_SIZE + 1];
} EmberAfPluginMessagingClientMessage;
#endif //EMBER_AF_PLUGIN_MESSAGING_CLIENT

#define ZCL_PRICE_CLUSTER_MAXIMUM_RATE_LABEL_LENGTH 11
#ifdef EMBER_AF_PLUGIN_PRICE_CLIENT
typedef struct {
  boolean valid;
  boolean active;
  int32u  providerId;
  int8u   rateLabel[ZCL_PRICE_CLUSTER_MAXIMUM_RATE_LABEL_LENGTH + 1];
  int32u  issuerEventId;
  int32u  currentTime;
  int8u   unitOfMeasure;
  int16u  currency;
  int8u   priceTrailingDigitAndPriceTier;
  int8u   numberOfPriceTiersAndRegisterTier;
  int32u  startTime;
  int32u  endTime;
  int16u  durationInMinutes;
  int32u  price;
  int8u   priceRatio;
  int32u  generationPrice;
  int8u   generationPriceRatio;
  int32u  alternateCostDelivered;
```

```
  int8u   alternateCostUnit;
  int8u   alternateCostTrailingDigit;
  int8u   numberOfBlockThresholds;
  int8u   priceControl;
} EmberAfPluginPriceClientPrice;
#endif //EMBER_AF_PLUGIN_PRICE_CLIENT

#ifdef EMBER_AF_PLUGIN_PSL_COMMISSIONING
typedef struct {
  EmberMulticastId groupId;
  int8u            groupType;
} EmberAfPluginPslCommissioningGroupInformationRecord;

typedef struct {
  EmberNodeId networkAddress;
  int8u       endpointId;
  int16u      profileId;
  int16u      deviceId;
  int8u       version;
} EmberAfPluginPslCommissioningEndpointInformationRecord;
#endif

typedef struct {
  int16u manufacturerId;
  int16u imageTypeId;
  int32u firmwareVersion;

  int8u deviceSpecificFileEui64[EUI64_SIZE];
} EmberAfOtaImageId;

typedef enum {
  EMBER_AF_OTA_STORAGE_SUCCESS             = 0,
  EMBER_AF_OTA_STORAGE_ERROR               = 1,
  EMBER_AF_OTA_STORAGE_RETURN_DATA_TOO_LONG = 2,
  EMBER_AF_OTA_STORAGE_PARTIAL_FILE_FOUND   = 3,
} EmberAfOtaStorageStatus;

#define EMBER_AF_OTA_MAX_HEADER_STRING_LENGTH 32

typedef struct {
  // Magic Number omitted since it is always the same.
  int16u headerVersion;
  int16u headerLength;
  int16u fieldControl;
  int16u manufacturerId;
  int16u imageTypeId;          // a.k.a. Device ID
  int32u firmwareVersion;
  int16u zigbeeStackVersion;

  int8u headerString[EMBER_AF_OTA_MAX_HEADER_STRING_LENGTH + 1];

  int32u imageSize;

```

```
00944   int8u securityCredentials;
00945   int8u upgradeFileDestination[EUI64_SIZE];
00946   int16u minimumHardwareVersion;
00947   int16u maximumHardwareVersion;
00948 } EmberAfOtaHeader;
00949
00954 typedef struct {
00955   int16u id;
00956   int32u length;
00957 } EmberAfTagData;
00958
00959 typedef enum {
00960   NO_APP_MESSAGE              = 0,
00961   RECEIVED_PARTNER_CERTIFICATE = 1,
00962   GENERATING_EPHEMERAL_KEYS    = 2,
00963   GENERATING_SHARED_SECRET     = 3,
00964   KEY_GENERATION_DONE          = 4,
00965   GENERATE_SHARED_SECRET_DONE  = 5,
00970   LINK_KEY_ESTABLISHED         = 6,
00971
00972
00977   NO_LOCAL_RESOURCES           = 7,
00978   PARTNER_NO_RESOURCES         = 8,
00979   TIMEOUT_OCCURRED             = 9,
00980   INVALID_APP_COMMAND          = 10,
00981   MESSAGE_SEND_FAILURE         = 11,
00982   PARTNER_SENT_TERMINATE       = 12,
00983   INVALID_PARTNER_MESSAGE      = 13,
00984
00990   BAD_CERTIFICATE_ISSUER       = 14,
00991   KEY_CONFIRM_FAILURE          = 15,
00992   BAD_KEY_ESTABLISHMENT_SUITE = 16,
00993
00994   KEY_TABLE_FULL               = 17,
00995
01001   NO_ESTABLISHMENT_ALLOWED     = 18,
01002 } EmberAfKeyEstablishmentNotifyMessage;
01003
01004 #define APP_NOTIFY_ERROR_CODE_START NO_LOCAL_RESOURCES
01005 #define APP_NOTIFY_MESSAGE_TEXT { \
01006   "None",                      \
01007   "Received Cert",             \
01008   "Generate keys",             \
01009   "Generate secret",           \
01010   "Key generate done",         \
01011   "Generate secret done",      \
01012   "Link key verified",         \
01013                                \
01014   /* Transient Error codes */  \
01015   "No local resources",        \
01016   "Partner no resources",      \
01017   "Timeout",                   \
```

```
01018   "Invalid app. command",        \
01019   "Message send failure",        \
01020   "Partner sent terminate",      \
01021   "Bad message",                 \
01022                                  \
01023   /* Fatal errors */             \
01024   "Bad cert issuer",             \
01025   "Key confirm failure",         \
01026   "Bad key est. suite",          \
01027   "Key table full",              \
01028   "Not allowed",                 \
01029 }
01030
01036 typedef enum {
01037   EMBER_AF_IMAGE_GOOD                = 0,
01038   EMBER_AF_IMAGE_BAD                 = 1,
01039   EMBER_AF_IMAGE_VERIFY_IN_PROGRESS   = 2,
01040
01041 #ifndef DOXYGEN_SHOULD_SKIP_THIS
01042   // Internal use only.
01043   EMBER_AF_IMAGE_VERIFY_WAIT         = 3,
01044   EMBER_AF_IMAGE_VERIFY_ERROR        = 4,
01045   EMBER_AF_IMAGE_UNKNOWN             = 5,
01046   EMBER_AF_NO_IMAGE_VERIFY_SUPPORT    = 6,
01047 #endif
01048 } EmberAfImageVerifyStatus;
01049
01057 typedef void (*EmberAfTickFunction)(int8u endpoint);
01058
01059
01066 typedef void (*EmberAfInitFunction)(int8u endpoint);
01067
01073 typedef void (*EmberAfClusterAttributeChangedCallback)(int8u endpoint,
01074                                                        EmberAfAttributeId attributeId);
01075
01082 typedef void (*EmberAfManufacturerSpecificClusterAttributeChangedCallback)(int8u endpoint,
01083                                                        EmberAfAttributeId attributeId,
01084                                                        int16u manufacturerCode);
01085
01086
01093 typedef void (*EmberAfDefaultResponseFunction)(int8u endpoint,
01094                                                int8u commandId,
01095                                                EmberAfStatus status);
01096
01102 typedef void (*EmberAfMessageSentFunction)(EmberOutgoingMessageType type,
01103                                           int16u indexOrDestination,
01104                                           EmberApsFrame *apsFrame,
01105                                           int16u msgLen,
01106                                           int8u *message,
01107                                           EmberStatus status);
01108
01109 #endif // __AF_API_TYPES__
```

client-command-macro.h

Go to the documentation of this file.

```
00001 // Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
00002 //
00003 // This file is generated. Please do not edit manually.
00004 //
00005 // This file is generated from java code via the following methods:
00006 // com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generateClientAPI()
00007 //
00008
00009 // Enclosing macro to prevent multiple inclusion
00010 #ifndef CLUSTER_CLIENT_API
00011 #define CLUSTER_CLIENT_API
00012
00013
00014 // This is generated client API
00015
00016
00017
00025 // @{
00033 #define emberAfFillCommandGlobalServerToClientReadAttributes(clusterId,attributeIds,attributeIdsLen) \
00034   emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
00035                             clusterId,\
00036                             ZCL_READ_ATTRIBUTES_COMMAND_ID,\
00037                             "b",\
00038                             attributeIds,\
00039                             attributeIdsLen\
00040                             );
00041
00042
00050 #define emberAfFillCommandGlobalClientToServerReadAttributes(clusterId,attributeIds,attributeIdsLen) \
00051   emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
00052                             clusterId,\
00053                             ZCL_READ_ATTRIBUTES_COMMAND_ID,\
00054                             "b",\
00055                             attributeIds,\
00056                             attributeIdsLen\
00057                             );
00058
00059
00067 #define emberAfFillCommandGlobalServerToClientReadAttributesResponse(clusterId,
readAttributeStatusRecords,readAttributeStatusRecordsLen) \
00068   emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
00069                             clusterId,\
00070                             ZCL_READ_ATTRIBUTES_RESPONSE_COMMAND_ID,\
00071                             "b",\
00072                             readAttributeStatusRecords,\
00073                             readAttributeStatusRecordsLen\
00074                             );
00075
00076
00084 #define emberAfFillCommandGlobalClientToServerReadAttributesResponse(clusterId,
readAttributeStatusRecords,readAttributeStatusRecordsLen) \
00085   emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
00086                             clusterId,\
```

```
                ZCL_READ_ATTRIBUTES_RESPONSE_COMMAND_ID,\
                "b",\
                readAttributeStatusRecords,\
                readAttributeStatusRecordsLen\
                );


#define emberAfFillCommandGlobalServerToClientWriteAttributes(clusterId,
writeAttributeRecords,writeAttributeRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                clusterId,\
                ZCL_WRITE_ATTRIBUTES_COMMAND_ID,\
                "b",\
                writeAttributeRecords,\
                writeAttributeRecordsLen\
                );


#define emberAfFillCommandGlobalClientToServerWriteAttributes(clusterId,
writeAttributeRecords,writeAttributeRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                clusterId,\
                ZCL_WRITE_ATTRIBUTES_COMMAND_ID,\
                "b",\
                writeAttributeRecords,\
                writeAttributeRecordsLen\
                );


#define emberAfFillCommandGlobalServerToClientWriteAttributesUndivided(clusterId,
writeAttributeRecords,writeAttributeRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                clusterId,\
                ZCL_WRITE_ATTRIBUTES_UNDIVIDED_COMMAND_ID,\
                "b",\
                writeAttributeRecords,\
                writeAttributeRecordsLen\
                );


#define emberAfFillCommandGlobalClientToServerWriteAttributesUndivided(clusterId,
writeAttributeRecords,writeAttributeRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                clusterId,\
                ZCL_WRITE_ATTRIBUTES_UNDIVIDED_COMMAND_ID,\
                "b",\
                writeAttributeRecords,\
                writeAttributeRecordsLen\
                );


#define emberAfFillCommandGlobalServerToClientWriteAttributesResponse(clusterId,
writeAttributeStatusRecords,writeAttributeStatusRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                clusterId,\
                ZCL_WRITE_ATTRIBUTES_RESPONSE_COMMAND_ID,\
                "b",\
                writeAttributeStatusRecords,\
                writeAttributeStatusRecordsLen\
```

```
                    );


#define emberAfFillCommandGlobalClientToServerWriteAttributesResponse(clusterId,
writeAttributeStatusRecords,writeAttributeStatusRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    clusterId,\
                    ZCL_WRITE_ATTRIBUTES_RESPONSE_COMMAND_ID,\
                    "b",\
                    writeAttributeStatusRecords,\
                    writeAttributeStatusRecordsLen\
                    );


#define emberAfFillCommandGlobalServerToClientWriteAttributesNoResponse(clusterId,
writeAttributeRecords,writeAttributeRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    clusterId,\
                    ZCL_WRITE_ATTRIBUTES_NO_RESPONSE_COMMAND_ID,\
                    "b",\
                    writeAttributeRecords,\
                    writeAttributeRecordsLen\
                    );


#define emberAfFillCommandGlobalClientToServerWriteAttributesNoResponse(clusterId,
writeAttributeRecords,writeAttributeRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    clusterId,\
                    ZCL_WRITE_ATTRIBUTES_NO_RESPONSE_COMMAND_ID,\
                    "b",\
                    writeAttributeRecords,\
                    writeAttributeRecordsLen\
                    );


#define emberAfFillCommandGlobalServerToClientConfigureReporting(clusterId,
configureReportingRecords,configureReportingRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    clusterId,\
                    ZCL_CONFIGURE_REPORTING_COMMAND_ID,\
                    "b",\
                    configureReportingRecords,\
                    configureReportingRecordsLen\
                    );


#define emberAfFillCommandGlobalClientToServerConfigureReporting(clusterId,
configureReportingRecords,configureReportingRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    clusterId,\
                    ZCL_CONFIGURE_REPORTING_COMMAND_ID,\
                    "b",\
                    configureReportingRecords,\
                    configureReportingRecordsLen\
                    );


#define emberAfFillCommandGlobalServerToClientConfigureReportingResponse
```

```
(clusterId,configureReportingStatusRecords,configureReportingStatusRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                            clusterId,\
                            ZCL_CONFIGURE_REPORTING_RESPONSE_COMMAND_ID,\
                            "b",\
                            configureReportingStatusRecords,\
                            configureReportingStatusRecordsLen\
                            );


#define emberAfFillCommandGlobalClientToServerConfigureReportingResponse
(clusterId,configureReportingStatusRecords,configureReportingStatusRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                            clusterId,\
                            ZCL_CONFIGURE_REPORTING_RESPONSE_COMMAND_ID,\
                            "b",\
                            configureReportingStatusRecords,\
                            configureReportingStatusRecordsLen\
                            );


#define emberAfFillCommandGlobalServerToClientReadReportingConfiguration
(clusterId,readReportingConfigurationAttributeRecords,readReportingConfigurationAttributeRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                            clusterId,\
                            ZCL_READ_REPORTING_CONFIGURATION_COMMAND_ID,\
                            "b",\
                            readReportingConfigurationAttributeRecords,\
                            readReportingConfigurationAttributeRecordsLen\
                            );


#define emberAfFillCommandGlobalClientToServerReadReportingConfiguration
(clusterId,readReportingConfigurationAttributeRecords,readReportingConfigurationAttributeRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                            clusterId,\
                            ZCL_READ_REPORTING_CONFIGURATION_COMMAND_ID,\
                            "b",\
                            readReportingConfigurationAttributeRecords,\
                            readReportingConfigurationAttributeRecordsLen\
                            );


#define emberAfFillCommandGlobalServerToClientReadReportingConfigurationResponse
(clusterId,readReportingConfigurationRecords,readReportingConfigurationRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                            clusterId,\
                            ZCL_READ_REPORTING_CONFIGURATION_RESPONSE_COMMAND_ID,\
                            "b",\
                            readReportingConfigurationRecords,\
                            readReportingConfigurationRecordsLen\
                            );


#define emberAfFillCommandGlobalClientToServerReadReportingConfigurationResponse
(clusterId,readReportingConfigurationRecords,readReportingConfigurationRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                            clusterId,\
                            ZCL_READ_REPORTING_CONFIGURATION_RESPONSE_COMMAND_ID,\
```

```
                    "b",\
                    readReportingConfigurationRecords,\
                    readReportingConfigurationRecordsLen\
                    );


#define emberAfFillCommandGlobalServerToClientReportAttributes(clusterId,
reportAttributeRecords,reportAttributeRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    clusterId,\
                    ZCL_REPORT_ATTRIBUTES_COMMAND_ID,\
                    "b",\
                    reportAttributeRecords,\
                    reportAttributeRecordsLen\
                    );


#define emberAfFillCommandGlobalClientToServerReportAttributes(clusterId,
reportAttributeRecords,reportAttributeRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    clusterId,\
                    ZCL_REPORT_ATTRIBUTES_COMMAND_ID,\
                    "b",\
                    reportAttributeRecords,\
                    reportAttributeRecordsLen\
                    );


#define emberAfFillCommandGlobalServerToClientDefaultResponse(clusterId,commandId,status) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    clusterId,\
                    ZCL_DEFAULT_RESPONSE_COMMAND_ID,\
                    "uu",\
                    commandId,\
                    status\
                    );


#define emberAfFillCommandGlobalClientToServerDefaultResponse(clusterId,commandId,status) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    clusterId,\
                    ZCL_DEFAULT_RESPONSE_COMMAND_ID,\
                    "uu",\
                    commandId,\
                    status\
                    );


#define emberAfFillCommandGlobalServerToClientDiscoverAttributes(clusterId,startId,maxAttributeIds) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    clusterId,\
                    ZCL_DISCOVER_ATTRIBUTES_COMMAND_ID,\
                    "vu",\
                    startId,\
                    maxAttributeIds\
                    );


#define emberAfFillCommandGlobalClientToServerDiscoverAttributes(clusterId,startId,maxAttributeIds) \
```

```
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                            clusterId,\
                            ZCL_DISCOVER_ATTRIBUTES_COMMAND_ID,\
                            "vu",\
                            startId,\
                            maxAttributeIds\
                            );


#define emberAfFillCommandGlobalServerToClientDiscoverAttributesResponse(clusterId,
discoveryComplete,discoverAttributesInfoRecords,discoverAttributesInfoRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                            clusterId,\
                            ZCL_DISCOVER_ATTRIBUTES_RESPONSE_COMMAND_ID,\
                            "ub",\
                            discoveryComplete,\
                            discoverAttributesInfoRecords,\
                            discoverAttributesInfoRecordsLen\
                            );


#define emberAfFillCommandGlobalClientToServerDiscoverAttributesResponse(clusterId,
discoveryComplete,discoverAttributesInfoRecords,discoverAttributesInfoRecordsLen) \
  emberAfFillExternalBuffer(ZCL_PROFILE_WIDE_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                            clusterId,\
                            ZCL_DISCOVER_ATTRIBUTES_RESPONSE_COMMAND_ID,\
                            "ub",\
                            discoveryComplete,\
                            discoverAttributesInfoRecords,\
                            discoverAttributesInfoRecordsLen\
                            );


// @{
#define emberAfFillCommandBasicClusterResetToFactoryDefaults() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                            ZCL_BASIC_CLUSTER_ID,\
                            ZCL_RESET_TO_FACTORY_DEFAULTS_COMMAND_ID,\
                            ""\
                            );


// @{
// @{
// @{
#define emberAfFillCommandIdentifyClusterIdentify(identifyTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                            ZCL_IDENTIFY_CLUSTER_ID,\
                            ZCL_IDENTIFY_COMMAND_ID,\
                            "v",\
                            identifyTime\
                            );


#define emberAfFillCommandIdentifyClusterIdentifyQuery() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
```

```
                ZCL_IDENTIFY_CLUSTER_ID,\
                ZCL_IDENTIFY_QUERY_COMMAND_ID,\
                ""\
                );


#define emberAfFillCommandIdentifyClusterIdentifyQueryResponse(timeout) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                ZCL_IDENTIFY_CLUSTER_ID,\
                ZCL_IDENTIFY_QUERY_RESPONSE_COMMAND_ID,\
                "v",\
                timeout\
                );


// @{
#define emberAfFillCommandGroupsClusterAddGroup(groupId,groupName) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_GROUPS_CLUSTER_ID,\
                ZCL_ADD_GROUP_COMMAND_ID,\
                "vs",\
                groupId,\
                groupName\
                );


#define emberAfFillCommandGroupsClusterViewGroup(groupId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_GROUPS_CLUSTER_ID,\
                ZCL_VIEW_GROUP_COMMAND_ID,\
                "v",\
                groupId\
                );


#define emberAfFillCommandGroupsClusterGetGroupMembership(groupCount,groupList,groupListLen) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_GROUPS_CLUSTER_ID,\
                ZCL_GET_GROUP_MEMBERSHIP_COMMAND_ID,\
                "ub",\
                groupCount,\
                groupList,\
                groupListLen\
                );


#define emberAfFillCommandGroupsClusterRemoveGroup(groupId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_GROUPS_CLUSTER_ID,\
                ZCL_REMOVE_GROUP_COMMAND_ID,\
                "v",\
                groupId\
                );

```

```
#define emberAfFillCommandGroupsClusterRemoveAllGroups() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_GROUPS_CLUSTER_ID,\
                    ZCL_REMOVE_ALL_GROUPS_COMMAND_ID,\
                    ""\
                    );


#define emberAfFillCommandGroupsClusterAddGroupIfIdentifying(groupId,groupName) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_GROUPS_CLUSTER_ID,\
                    ZCL_ADD_GROUP_IF_IDENTIFYING_COMMAND_ID,\
                    "vs",\
                    groupId,\
                    groupName\
                    );


#define emberAfFillCommandGroupsClusterAddGroupResponse(status,groupId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_GROUPS_CLUSTER_ID,\
                    ZCL_ADD_GROUP_RESPONSE_COMMAND_ID,\
                    "uv",\
                    status,\
                    groupId\
                    );


#define emberAfFillCommandGroupsClusterViewGroupResponse(status,groupId,groupName) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_GROUPS_CLUSTER_ID,\
                    ZCL_VIEW_GROUP_RESPONSE_COMMAND_ID,\
                    "uvs",\
                    status,\
                    groupId,\
                    groupName\
                    );


#define emberAfFillCommandGroupsClusterGetGroupMembershipResponse(capacity,groupCount,
groupList,groupListLen) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_GROUPS_CLUSTER_ID,\
                    ZCL_GET_GROUP_MEMBERSHIP_RESPONSE_COMMAND_ID,\
                    "uub",\
                    capacity,\
                    groupCount,\
                    groupList,\
                    groupListLen\
                    );


#define emberAfFillCommandGroupsClusterRemoveGroupResponse(status,groupId) \
```

```
emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                ZCL_GROUPS_CLUSTER_ID,\
                ZCL_REMOVE_GROUP_RESPONSE_COMMAND_ID,\
                "uv",\
                status,\
                groupId\
                );


// @{
#define emberAfFillCommandScenesClusterAddScene(groupId,sceneId,transitionTime,sceneName,
extensionFieldSets,extensionFieldSetsLen) \
emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_SCENES_CLUSTER_ID,\
                ZCL_ADD_SCENE_COMMAND_ID,\
                "vuvsb",\
                groupId,\
                sceneId,\
                transitionTime,\
                sceneName,\
                extensionFieldSets,\
                extensionFieldSetsLen\
                );


#define emberAfFillCommandScenesClusterViewScene(groupId,sceneId) \
emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_SCENES_CLUSTER_ID,\
                ZCL_VIEW_SCENE_COMMAND_ID,\
                "vu",\
                groupId,\
                sceneId\
                );


#define emberAfFillCommandScenesClusterRemoveScene(groupId,sceneId) \
emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_SCENES_CLUSTER_ID,\
                ZCL_REMOVE_SCENE_COMMAND_ID,\
                "vu",\
                groupId,\
                sceneId\
                );


#define emberAfFillCommandScenesClusterRemoveAllScenes(groupId) \
emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_SCENES_CLUSTER_ID,\
                ZCL_REMOVE_ALL_SCENES_COMMAND_ID,\
                "v",\
                groupId\
                );


```

```
#define emberAfFillCommandScenesClusterStoreScene(groupId,sceneId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_SCENES_CLUSTER_ID,\
                    ZCL_STORE_SCENE_COMMAND_ID,\
                    "vu",\
                    groupId,\
                    sceneId\
                    );


#define emberAfFillCommandScenesClusterRecallScene(groupId,sceneId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_SCENES_CLUSTER_ID,\
                    ZCL_RECALL_SCENE_COMMAND_ID,\
                    "vu",\
                    groupId,\
                    sceneId\
                    );


#define emberAfFillCommandScenesClusterGetSceneMembership(groupId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_SCENES_CLUSTER_ID,\
                    ZCL_GET_SCENE_MEMBERSHIP_COMMAND_ID,\
                    "v",\
                    groupId\
                    );


#define emberAfFillCommandScenesClusterAddSceneResponse(status,groupId,sceneId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_SCENES_CLUSTER_ID,\
                    ZCL_ADD_SCENE_RESPONSE_COMMAND_ID,\
                    "uvu",\
                    status,\
                    groupId,\
                    sceneId\
                    );


#define emberAfFillCommandScenesClusterViewSceneResponse(status,groupId,sceneId,transitionTime,
sceneName,extensionFieldSets,extensionFieldSetsLen) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_SCENES_CLUSTER_ID,\
                    ZCL_VIEW_SCENE_RESPONSE_COMMAND_ID,\
                    "uvuvsb",\
                    status,\
                    groupId,\
                    sceneId,\
                    transitionTime,\
                    sceneName,\
                    extensionFieldSets,\
                    extensionFieldSetsLen\
                    );
```

```
#define emberAfFillCommandScenesClusterRemoveSceneResponse(status,groupId,sceneId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_SCENES_CLUSTER_ID,\
                    ZCL_REMOVE_SCENE_RESPONSE_COMMAND_ID,\
                    "uvu",\
                    status,\
                    groupId,\
                    sceneId\
                    );


#define emberAfFillCommandScenesClusterRemoveAllScenesResponse(status,groupId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_SCENES_CLUSTER_ID,\
                    ZCL_REMOVE_ALL_SCENES_RESPONSE_COMMAND_ID,\
                    "uv",\
                    status,\
                    groupId\
                    );


#define emberAfFillCommandScenesClusterStoreSceneResponse(status,groupId,sceneId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_SCENES_CLUSTER_ID,\
                    ZCL_STORE_SCENE_RESPONSE_COMMAND_ID,\
                    "uvu",\
                    status,\
                    groupId,\
                    sceneId\
                    );


#define emberAfFillCommandScenesClusterGetSceneMembershipResponse(status,capacity,groupId,sceneCount,
sceneList,sceneListLen) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_SCENES_CLUSTER_ID,\
                    ZCL_GET_SCENE_MEMBERSHIP_RESPONSE_COMMAND_ID,\
                    "uuvub",\
                    status,\
                    capacity,\
                    groupId,\
                    sceneCount,\
                    sceneList,\
                    sceneListLen\
                    );


// @{
#define emberAfFillCommandOnOffClusterOff() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_ON_OFF_CLUSTER_ID,\
                    ZCL_OFF_COMMAND_ID,\
```

```
                        ""\
                        );


#define emberAfFillCommandOnOffClusterOn() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_ON_OFF_CLUSTER_ID,\
                        ZCL_ON_COMMAND_ID,\
                        ""\
                        );


#define emberAfFillCommandOnOffClusterToggle() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_ON_OFF_CLUSTER_ID,\
                        ZCL_TOGGLE_COMMAND_ID,\
                        ""\
                        );


#define emberAfFillCommandOnOffClusterSampleMfgSpecificOffWithTransition() \
  emberAfFillExternalManufacturerSpecificBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_ON_OFF_CLUSTER_ID,\
                        0x1002,\
                        ZCL_SAMPLE_MFG_SPECIFIC_OFF_WITH_TRANSITION_COMMAND_ID,\
                        ""\
                        );


#define emberAfFillCommandOnOffClusterSampleMfgSpecificOnWithTransition() \
  emberAfFillExternalManufacturerSpecificBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_ON_OFF_CLUSTER_ID,\
                        0x1002,\
                        ZCL_SAMPLE_MFG_SPECIFIC_ON_WITH_TRANSITION_COMMAND_ID,\
                        ""\
                        );


#define emberAfFillCommandOnOffClusterSampleMfgSpecificToggleWithTransition() \
  emberAfFillExternalManufacturerSpecificBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_ON_OFF_CLUSTER_ID,\
                        0x1002,\
                        ZCL_SAMPLE_MFG_SPECIFIC_TOGGLE_WITH_TRANSITION_COMMAND_ID,\
                        ""\
                        );


// @{
// @{
#define emberAfFillCommandLevelControlClusterMoveToLevel(level,transitionTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_LEVEL_CONTROL_CLUSTER_ID,\
                        ZCL_MOVE_TO_LEVEL_COMMAND_ID,\
```

```
                    "uv",\
                    level,\
                    transitionTime\
                    );


#define emberAfFillCommandLevelControlClusterMove(moveMode,rate) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_LEVEL_CONTROL_CLUSTER_ID,\
                    ZCL_MOVE_COMMAND_ID,\
                    "uu",\
                    moveMode,\
                    rate\
                    );


#define emberAfFillCommandLevelControlClusterStep(stepMode,stepSize,transitionTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_LEVEL_CONTROL_CLUSTER_ID,\
                    ZCL_STEP_COMMAND_ID,\
                    "uuv",\
                    stepMode,\
                    stepSize,\
                    transitionTime\
                    );


#define emberAfFillCommandLevelControlClusterStop() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_LEVEL_CONTROL_CLUSTER_ID,\
                    ZCL_STOP_COMMAND_ID,\
                    ""\
                    );


#define emberAfFillCommandLevelControlClusterMoveToLevelWithOnOff(level,transitionTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_LEVEL_CONTROL_CLUSTER_ID,\
                    ZCL_MOVE_TO_LEVEL_WITH_ON_OFF_COMMAND_ID,\
                    "uv",\
                    level,\
                    transitionTime\
                    );


#define emberAfFillCommandLevelControlClusterMoveWithOnOff(moveMode,rate) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_LEVEL_CONTROL_CLUSTER_ID,\
                    ZCL_MOVE_WITH_ON_OFF_COMMAND_ID,\
                    "uu",\
                    moveMode,\
                    rate\
                    );

```

```
#define emberAfFillCommandLevelControlClusterStepWithOnOff(stepMode,stepSize,transitionTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_LEVEL_CONTROL_CLUSTER_ID,\
                        ZCL_STEP_WITH_ON_OFF_COMMAND_ID,\
                        "uuv",\
                        stepMode,\
                        stepSize,\
                        transitionTime\
                        );


#define emberAfFillCommandLevelControlClusterStopWithOnOff() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_LEVEL_CONTROL_CLUSTER_ID,\
                        ZCL_STOP_WITH_ON_OFF_COMMAND_ID,\
                        ""\
                        );


// @{
#define emberAfFillCommandAlarmClusterResetAlarm(alarmCode,clusterId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_ALARM_CLUSTER_ID,\
                        ZCL_RESET_ALARM_COMMAND_ID,\
                        "uv",\
                        alarmCode,\
                        clusterId\
                        );


#define emberAfFillCommandAlarmClusterResetAllAlarms() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_ALARM_CLUSTER_ID,\
                        ZCL_RESET_ALL_ALARMS_COMMAND_ID,\
                        ""\
                        );


#define emberAfFillCommandAlarmClusterGetAlarm() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_ALARM_CLUSTER_ID,\
                        ZCL_GET_ALARM_COMMAND_ID,\
                        ""\
                        );


#define emberAfFillCommandAlarmClusterResetAlarmLog() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_ALARM_CLUSTER_ID,\
                        ZCL_RESET_ALARM_LOG_COMMAND_ID,\
                        ""\
                        );
```

```
#define emberAfFillCommandAlarmClusterAlarm(alarmCode,clusterId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_ALARM_CLUSTER_ID,\
                    ZCL_ALARM_COMMAND_ID,\
                    "uv",\
                    alarmCode,\
                    clusterId\
                    );


#define emberAfFillCommandAlarmClusterGetAlarmResponse(status,alarmCode,clusterId,timeStamp) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_ALARM_CLUSTER_ID,\
                    ZCL_GET_ALARM_RESPONSE_COMMAND_ID,\
                    "uuvw",\
                    status,\
                    alarmCode,\
                    clusterId,\
                    timeStamp\
                    );


// @{
// @{
#define emberAfFillCommandRssiLocationClusterSetAbsoluteLocation(coordinate1,coordinate2,coordinate3,
power,pathLossExponent) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_RSSI_LOCATION_CLUSTER_ID,\
                    ZCL_SET_ABSOLUTE_LOCATION_COMMAND_ID,\
                    "vvvvv",\
                    coordinate1,\
                    coordinate2,\
                    coordinate3,\
                    power,\
                    pathLossExponent\
                    );


#define emberAfFillCommandRssiLocationClusterSetDeviceConfiguration(power,pathLossExponent,
calculationPeriod,numberRssiMeasurements,reportingPeriod) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_RSSI_LOCATION_CLUSTER_ID,\
                    ZCL_SET_DEVICE_CONFIGURATION_COMMAND_ID,\
                    "vvvuv",\
                    power,\
                    pathLossExponent,\
                    calculationPeriod,\
                    numberRssiMeasurements,\
                    reportingPeriod\
                    );


#define emberAfFillCommandRssiLocationClusterGetDeviceConfiguration(targetAddress) \
```

```
emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_RSSI_LOCATION_CLUSTER_ID,\
                        ZCL_GET_DEVICE_CONFIGURATION_COMMAND_ID,\
                        "8",\
                        targetAddress\
                        );


#define emberAfFillCommandRssiLocationClusterGetLocationData(flags,numberResponses,targetAddress) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_RSSI_LOCATION_CLUSTER_ID,\
                        ZCL_GET_LOCATION_DATA_COMMAND_ID,\
                        "uu8",\
                        flags,\
                        numberResponses,\
                        targetAddress\
                        );


#define emberAfFillCommandRssiLocationClusterRssiResponse(replyingDevice,coordinate1,coordinate2,coordinate3,
rssi,numberRssiMeasurements) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_RSSI_LOCATION_CLUSTER_ID,\
                        ZCL_RSSI_RESPONSE_COMMAND_ID,\
                        "8vvvuu",\
                        replyingDevice,\
                        coordinate1,\
                        coordinate2,\
                        coordinate3,\
                        rssi,\
                        numberRssiMeasurements\
                        );


#define emberAfFillCommandRssiLocationClusterSendPings(targetAddress,numberRssiMeasurements,calculationPeriod) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_RSSI_LOCATION_CLUSTER_ID,\
                        ZCL_SEND_PINGS_COMMAND_ID,\
                        "8uv",\
                        targetAddress,\
                        numberRssiMeasurements,\
                        calculationPeriod\
                        );


#define emberAfFillCommandRssiLocationClusterAnchorNodeAnnounce(anchorNodeIeeeAddress,
coordinate1,coordinate2,coordinate3) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_RSSI_LOCATION_CLUSTER_ID,\
                        ZCL_ANCHOR_NODE_ANNOUNCE_COMMAND_ID,\
                        "8vvv",\
                        anchorNodeIeeeAddress,\
                        coordinate1,\
                        coordinate2,\
                        coordinate3\
```

```
                    );


#define emberAfFillCommandRssiLocationClusterDeviceConfigurationResponse(status,power,pathLossExponent,calculationPeriod,numberRssiMeasurements,reportingPeriod) \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_RSSI_LOCATION_CLUSTER_ID,\
                    ZCL_DEVICE_CONFIGURATION_RESPONSE_COMMAND_ID,\
                    "uvvvuv",\
                    status,\
                    power,\
                    pathLossExponent,\
                    calculationPeriod,\
                    numberRssiMeasurements,\
                    reportingPeriod\
                    );


#define emberAfFillCommandRssiLocationClusterLocationDataResponse(status,locationType,coordinate1,coordinate2,coordinate3,power,pathLossExponent,locationMethod,qualityMeasure,locationAge) \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_RSSI_LOCATION_CLUSTER_ID,\
                    ZCL_LOCATION_DATA_RESPONSE_COMMAND_ID,\
                    "uuvvvvvuuv",\
                    status,\
                    locationType,\
                    coordinate1,\
                    coordinate2,\
                    coordinate3,\
                    power,\
                    pathLossExponent,\
                    locationMethod,\
                    qualityMeasure,\
                    locationAge\
                    );


#define emberAfFillCommandRssiLocationClusterLocationDataNotification(locationType,coordinate1,coordinate2,coordinate3,power,pathLossExponent,locationMethod,qualityMeasure,locationAge) \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_RSSI_LOCATION_CLUSTER_ID,\
                    ZCL_LOCATION_DATA_NOTIFICATION_COMMAND_ID,\
                    "uvvvvvuuv",\
                    locationType,\
                    coordinate1,\
                    coordinate2,\
                    coordinate3,\
                    power,\
                    pathLossExponent,\
                    locationMethod,\
                    qualityMeasure,\
                    locationAge\
                    );


#define emberAfFillCommandRssiLocationClusterCompactLocationDataNotification(locationType,
```

```
coordinate1,coordinate2,coordinate3,qualityMeasure,locationAge) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                ZCL_RSSI_LOCATION_CLUSTER_ID,\
                ZCL_COMPACT_LOCATION_DATA_NOTIFICATION_COMMAND_ID,\
                "uvvvuv",\
                locationType,\
                coordinate1,\
                coordinate2,\
                coordinate3,\
                qualityMeasure,\
                locationAge\
                );


#define emberAfFillCommandRssiLocationClusterRssiPing(locationType) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                ZCL_RSSI_LOCATION_CLUSTER_ID,\
                ZCL_RSSI_PING_COMMAND_ID,\
                "u",\
                locationType\
                );


#define emberAfFillCommandRssiLocationClusterRssiRequest() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                ZCL_RSSI_LOCATION_CLUSTER_ID,\
                ZCL_RSSI_REQUEST_COMMAND_ID,\
                ""\
                );


#define emberAfFillCommandRssiLocationClusterReportRssiMeasurements(measuringDevice,neighbors,
neighborsInfo,neighborsInfoLen) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                ZCL_RSSI_LOCATION_CLUSTER_ID,\
                ZCL_REPORT_RSSI_MEASUREMENTS_COMMAND_ID,\
                "8ub",\
                measuringDevice,\
                neighbors,\
                neighborsInfo,\
                neighborsInfoLen\
                );


#define emberAfFillCommandRssiLocationClusterRequestOwnLocation(blindNode) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                ZCL_RSSI_LOCATION_CLUSTER_ID,\
                ZCL_REQUEST_OWN_LOCATION_COMMAND_ID,\
                "8",\
                blindNode\
                );


// @{
// @{
```

```
#define emberAfFillCommandCommissioningClusterRestartDevice(options,delay,jitter) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_COMMISSIONING_CLUSTER_ID,\
                    ZCL_RESTART_DEVICE_COMMAND_ID,\
                    "uuu",\
                    options,\
                    delay,\
                    jitter\
                    );


#define emberAfFillCommandCommissioningClusterSaveStartupParameters(options,index) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_COMMISSIONING_CLUSTER_ID,\
                    ZCL_SAVE_STARTUP_PARAMETERS_COMMAND_ID,\
                    "uu",\
                    options,\
                    index\
                    );


#define emberAfFillCommandCommissioningClusterRestoreStartupParameters(options,index) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_COMMISSIONING_CLUSTER_ID,\
                    ZCL_RESTORE_STARTUP_PARAMETERS_COMMAND_ID,\
                    "uu",\
                    options,\
                    index\
                    );


#define emberAfFillCommandCommissioningClusterResetStartupParameters(options,index) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_COMMISSIONING_CLUSTER_ID,\
                    ZCL_RESET_STARTUP_PARAMETERS_COMMAND_ID,\
                    "uu",\
                    options,\
                    index\
                    );


#define emberAfFillCommandCommissioningClusterRestartDeviceResponse(status) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_COMMISSIONING_CLUSTER_ID,\
                    ZCL_RESTART_DEVICE_RESPONSE_COMMAND_ID,\
                    "u",\
                    status\
                    );


#define emberAfFillCommandCommissioningClusterSaveStartupParametersResponse(status) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_COMMISSIONING_CLUSTER_ID,\
```

```
                    ZCL_SAVE_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID,\
                    "u",\
                    status\
                    );


#define emberAfFillCommandCommissioningClusterRestoreStartupParametersResponse(status) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_COMMISSIONING_CLUSTER_ID,\
                    ZCL_RESTORE_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID,\
                    "u",\
                    status\
                    );


#define emberAfFillCommandCommissioningClusterResetStartupParametersResponse(status) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_COMMISSIONING_CLUSTER_ID,\
                    ZCL_RESET_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID,\
                    "u",\
                    status\
                    );


// @{
// @{
// @{
#define emberAfFillCommandThermostatClusterSetpointRaiseLower(mode,amount) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_THERMOSTAT_CLUSTER_ID,\
                    ZCL_SETPOINT_RAISE_LOWER_COMMAND_ID,\
                    "uu",\
                    mode,\
                    amount\
                    );


// @{
// @{
// @{
// @{
#define emberAfFillCommandColorControlClusterMoveToHue(hue,direction,transitionTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_COLOR_CONTROL_CLUSTER_ID,\
                    ZCL_MOVE_TO_HUE_COMMAND_ID,\
                    "uuv",\
                    hue,\
                    direction,\
                    transitionTime\
                    );


#define emberAfFillCommandColorControlClusterMoveHue(moveMode,rate) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
```

```
                    ZCL_COLOR_CONTROL_CLUSTER_ID,\
                    ZCL_MOVE_HUE_COMMAND_ID,\
                    "uu",\
                    moveMode,\
                    rate\
                    );


#define emberAfFillCommandColorControlClusterStepHue(stepMode,stepSize,transitionTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_COLOR_CONTROL_CLUSTER_ID,\
                    ZCL_STEP_HUE_COMMAND_ID,\
                    "uuu",\
                    stepMode,\
                    stepSize,\
                    transitionTime\
                    );


#define emberAfFillCommandColorControlClusterMoveToSaturation(saturation,transitionTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_COLOR_CONTROL_CLUSTER_ID,\
                    ZCL_MOVE_TO_SATURATION_COMMAND_ID,\
                    "uv",\
                    saturation,\
                    transitionTime\
                    );


#define emberAfFillCommandColorControlClusterMoveSaturation(moveMode,rate) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_COLOR_CONTROL_CLUSTER_ID,\
                    ZCL_MOVE_SATURATION_COMMAND_ID,\
                    "uu",\
                    moveMode,\
                    rate\
                    );


#define emberAfFillCommandColorControlClusterStepSaturation(stepMode,stepSize,transitionTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_COLOR_CONTROL_CLUSTER_ID,\
                    ZCL_STEP_SATURATION_COMMAND_ID,\
                    "uuu",\
                    stepMode,\
                    stepSize,\
                    transitionTime\
                    );


#define emberAfFillCommandColorControlClusterMoveToHueAndSaturation(hue,saturation,transitionTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_COLOR_CONTROL_CLUSTER_ID,\
                    ZCL_MOVE_TO_HUE_AND_SATURATION_COMMAND_ID,\
```

```
                    "uuv",\
                    hue,\
                    saturation,\
                    transitionTime\
                    );


#define emberAfFillCommandColorControlClusterMoveToColor(colorX,colorY,transitionTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_COLOR_CONTROL_CLUSTER_ID,\
                    ZCL_MOVE_TO_COLOR_COMMAND_ID,\
                    "vvv",\
                    colorX,\
                    colorY,\
                    transitionTime\
                    );


#define emberAfFillCommandColorControlClusterMoveColor(rateX,rateY) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_COLOR_CONTROL_CLUSTER_ID,\
                    ZCL_MOVE_COLOR_COMMAND_ID,\
                    "vv",\
                    rateX,\
                    rateY\
                    );


#define emberAfFillCommandColorControlClusterStepColor(stepX,stepY,transitionTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_COLOR_CONTROL_CLUSTER_ID,\
                    ZCL_STEP_COLOR_COMMAND_ID,\
                    "vvv",\
                    stepX,\
                    stepY,\
                    transitionTime\
                    );


#define emberAfFillCommandColorControlClusterMoveToColorTemperature(colorTemperature,transitionTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_COLOR_CONTROL_CLUSTER_ID,\
                    ZCL_MOVE_TO_COLOR_TEMPERATURE_COMMAND_ID,\
                    "vv",\
                    colorTemperature,\
                    transitionTime\
                    );


// @{
// @{
// @{
// @{
// @{
// @{
// @{
```

```
// @{
// @{
#define emberAfFillCommandIasZoneClusterZoneEnrollResponse(enrollResponseCode,zoneId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_IAS_ZONE_CLUSTER_ID,\
                        ZCL_ZONE_ENROLL_RESPONSE_COMMAND_ID,\
                        "uu",\
                        enrollResponseCode,\
                        zoneId\
                        );


#define emberAfFillCommandIasZoneClusterZoneStatusChangeNotification(zoneStatus,extendedStatus) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                        ZCL_IAS_ZONE_CLUSTER_ID,\
                        ZCL_ZONE_STATUS_CHANGE_NOTIFICATION_COMMAND_ID,\
                        "vu",\
                        zoneStatus,\
                        extendedStatus\
                        );


#define emberAfFillCommandIasZoneClusterZoneEnrollRequest(zoneType,manufacturerCode) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                        ZCL_IAS_ZONE_CLUSTER_ID,\
                        ZCL_ZONE_ENROLL_REQUEST_COMMAND_ID,\
                        "vv",\
                        zoneType,\
                        manufacturerCode\
                        );


// @{
#define emberAfFillCommandIasAceClusterArm(armMode) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_IAS_ACE_CLUSTER_ID,\
                        ZCL_ARM_COMMAND_ID,\
                        "u",\
                        armMode\
                        );


#define emberAfFillCommandIasAceClusterBypass(numberOfZones,zoneIds,zoneIdsLen) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_IAS_ACE_CLUSTER_ID,\
                        ZCL_BYPASS_COMMAND_ID,\
                        "ub",\
                        numberOfZones,\
                        zoneIds,\
                        zoneIdsLen\
                        );


#define emberAfFillCommandIasAceClusterEmergency() \
```

```
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_IAS_ACE_CLUSTER_ID,\
                        ZCL_EMERGENCY_COMMAND_ID,\
                        ""\
                        );


#define emberAfFillCommandIasAceClusterFire() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_IAS_ACE_CLUSTER_ID,\
                        ZCL_FIRE_COMMAND_ID,\
                        ""\
                        );


#define emberAfFillCommandIasAceClusterPanic() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_IAS_ACE_CLUSTER_ID,\
                        ZCL_PANIC_COMMAND_ID,\
                        ""\
                        );


#define emberAfFillCommandIasAceClusterGetZoneIdMap() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_IAS_ACE_CLUSTER_ID,\
                        ZCL_GET_ZONE_ID_MAP_COMMAND_ID,\
                        ""\
                        );


#define emberAfFillCommandIasAceClusterGetZoneInformation(zoneId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_IAS_ACE_CLUSTER_ID,\
                        ZCL_GET_ZONE_INFORMATION_COMMAND_ID,\
                        "u",\
                        zoneId\
                        );


#define emberAfFillCommandIasAceClusterArmResponse(armNotification) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                        ZCL_IAS_ACE_CLUSTER_ID,\
                        ZCL_ARM_RESPONSE_COMMAND_ID,\
                        "u",\
                        armNotification\
                        );


#define emberAfFillCommandIasAceClusterGetZoneIdMapResponse(section0,section1,section2,section3,
section4,section5,section6,section7,section8,section9,section10,section11,section12,section13,section14,section15) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
```

```
                    ZCL_IAS_ACE_CLUSTER_ID,\
                    ZCL_GET_ZONE_ID_MAP_RESPONSE_COMMAND_ID,\
                    "vvvvvvvvvvvvvvvv",\
                    section0,\
                    section1,\
                    section2,\
                    section3,\
                    section4,\
                    section5,\
                    section6,\
                    section7,\
                    section8,\
                    section9,\
                    section10,\
                    section11,\
                    section12,\
                    section13,\
                    section14,\
                    section15\
                    );


#define emberAfFillCommandIasAceClusterGetZoneInformationResponse(zoneId,zoneType,ieeeAddress) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_IAS_ACE_CLUSTER_ID,\
                    ZCL_GET_ZONE_INFORMATION_RESPONSE_COMMAND_ID,\
                    "uv8",\
                    zoneId,\
                    zoneType,\
                    ieeeAddress\
                    );


// @{
#define emberAfFillCommandIasWdClusterStartWarning(warningInfo,warningDuration) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_IAS_WD_CLUSTER_ID,\
                    ZCL_START_WARNING_COMMAND_ID,\
                    "uv",\
                    warningInfo,\
                    warningDuration\
                    );


#define emberAfFillCommandIasWdClusterSquawk(squawkInfo) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_IAS_WD_CLUSTER_ID,\
                    ZCL_SQUAWK_COMMAND_ID,\
                    "u",\
                    squawkInfo\
                    );


// @{
#define emberAfFillCommandDoorLockClusterLockDoor() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
```

```
                    ZCL_DOOR_LOCK_CLUSTER_ID,\
                    ZCL_LOCK_DOOR_COMMAND_ID,\
                    ""\
                    );


#define emberAfFillCommandDoorLockClusterUnlockDoor() \
  emberAfFillExternalBuffer (ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_DOOR_LOCK_CLUSTER_ID,\
                    ZCL_UNLOCK_DOOR_COMMAND_ID,\
                    ""\
                    );


#define emberAfFillCommandDoorLockClusterLockDoorResponse(status) \
  emberAfFillExternalBuffer (ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_DOOR_LOCK_CLUSTER_ID,\
                    ZCL_LOCK_DOOR_RESPONSE_COMMAND_ID,\
                    "u",\
                    status\
                    );


#define emberAfFillCommandDoorLockClusterUnlockDoorResponse(status) \
  emberAfFillExternalBuffer (ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_DOOR_LOCK_CLUSTER_ID,\
                    ZCL_UNLOCK_DOOR_RESPONSE_COMMAND_ID,\
                    "u",\
                    status\
                    );


// @{
#define emberAfFillCommandWindowCoveringClusterWindowCoveringUpOpen() \
  emberAfFillExternalBuffer (ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_WINDOW_COVERING_CLUSTER_ID,\
                    ZCL_WINDOW_COVERING_UP_OPEN_COMMAND_ID,\
                    ""\
                    );


#define emberAfFillCommandWindowCoveringClusterWindowCoveringDownClose() \
  emberAfFillExternalBuffer (ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_WINDOW_COVERING_CLUSTER_ID,\
                    ZCL_WINDOW_COVERING_DOWN_CLOSE_COMMAND_ID,\
                    ""\
                    );


#define emberAfFillCommandWindowCoveringClusterWindowCoveringStop() \
  emberAfFillExternalBuffer (ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_WINDOW_COVERING_CLUSTER_ID,\
                    ZCL_WINDOW_COVERING_STOP_COMMAND_ID,\
```

```
                    ""\
                    );


#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftSetpoint(indexOfLiftSetpoint) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_WINDOW_COVERING_CLUSTER_ID,\
                    ZCL_WINDOW_COVERING_GO_TO_LIFT_SETPOINT_COMMAND_ID,\
                    "u",\
                    indexOfLiftSetpoint\
                    );


#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftValue(liftValue) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_WINDOW_COVERING_CLUSTER_ID,\
                    ZCL_WINDOW_COVERING_GO_TO_LIFT_VALUE_COMMAND_ID,\
                    "v",\
                    liftValue\
                    );


#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftPercentage(percentageLiftValue) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_WINDOW_COVERING_CLUSTER_ID,\
                    ZCL_WINDOW_COVERING_GO_TO_LIFT_PERCENTAGE_COMMAND_ID,\
                    "u",\
                    percentageLiftValue\
                    );


#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltSetpoint(indexOfTiltSetpoint) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_WINDOW_COVERING_CLUSTER_ID,\
                    ZCL_WINDOW_COVERING_GO_TO_TILT_SETPOINT_COMMAND_ID,\
                    "u",\
                    indexOfTiltSetpoint\
                    );


#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltValue(tiltValue) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_WINDOW_COVERING_CLUSTER_ID,\
                    ZCL_WINDOW_COVERING_GO_TO_TILT_VALUE_COMMAND_ID,\
                    "v",\
                    tiltValue\
                    );


#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltPercentage(percentageTiltValue) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_WINDOW_COVERING_CLUSTER_ID,\
                    ZCL_WINDOW_COVERING_GO_TO_TILT_PERCENTAGE_COMMAND_ID,\
```

```
                    "u",\
                    percentageTiltValue\
                    );


#define emberAfFillCommandWindowCoveringClusterWindowCoveringGoToProgramSetpoint(setpointType,setpointIndex,setpointValue) \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_WINDOW_COVERING_CLUSTER_ID,\
                    ZCL_WINDOW_COVERING_GO_TO_PROGRAM_SETPOINT_COMMAND_ID,\
                    "uuv",\
                    setpointType,\
                    setpointIndex,\
                    setpointValue\
                    );


// @{
#define emberAfFillCommandGenericTunnelClusterMatchProtocolAddress(protocolAddress) \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_GENERIC_TUNNEL_CLUSTER_ID,\
                    ZCL_MATCH_PROTOCOL_ADDRESS_COMMAND_ID,\
                    "s",\
                    protocolAddress\
                    );


#define emberAfFillCommandGenericTunnelClusterMatchProtocolAddressResponse() \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_GENERIC_TUNNEL_CLUSTER_ID,\
                    ZCL_MATCH_PROTOCOL_ADDRESS_RESPONSE_COMMAND_ID,\
                    ""\
                    );


#define emberAfFillCommandGenericTunnelClusterAdvertiseProtocolAddress(protocolAddress) \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_GENERIC_TUNNEL_CLUSTER_ID,\
                    ZCL_ADVERTISE_PROTOCOL_ADDRESS_COMMAND_ID,\
                    "s",\
                    protocolAddress\
                    );


// @{
#define emberAfFillCommandBacnetProtocolTunnelClusterTransferNpdu(NPDU) \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_BACNET_PROTOCOL_TUNNEL_CLUSTER_ID,\
                    ZCL_TRANSFER_NPDU_COMMAND_ID,\
                    "s",\
                    NPDU\
                    );


```

```
// @{
#define emberAfFillCommandOtaBootloadClusterImageNotify(payloadType,queryJitter,manufacturerId,
imageType,newFileVersion) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_OTA_BOOTLOAD_CLUSTER_ID,\
                    ZCL_IMAGE_NOTIFY_COMMAND_ID,\
                    "uuvvw",\
                    payloadType,\
                    queryJitter,\
                    manufacturerId,\
                    imageType,\
                    newFileVersion\
                    );


#define emberAfFillCommandOtaBootloadClusterQueryNextImageRequest(fieldControl,manufacturerId,
imageType,currentFileVersion,hardwareVersion) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_OTA_BOOTLOAD_CLUSTER_ID,\
                    ZCL_QUERY_NEXT_IMAGE_REQUEST_COMMAND_ID,\
                    "uvvwv",\
                    fieldControl,\
                    manufacturerId,\
                    imageType,\
                    currentFileVersion,\
                    hardwareVersion\
                    );


#define emberAfFillCommandOtaBootloadClusterQueryNextImageResponse(status,manufacturerId,
imageType,fileVersion,imageSize) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_OTA_BOOTLOAD_CLUSTER_ID,\
                    ZCL_QUERY_NEXT_IMAGE_RESPONSE_COMMAND_ID,\
                    "uvvww",\
                    status,\
                    manufacturerId,\
                    imageType,\
                    fileVersion,\
                    imageSize\
                    );


#define emberAfFillCommandOtaBootloadClusterImageBlockRequest(fieldControl,manufacturerId,
imageType,fileVersion,fileOffset,maxDataSize,requestNodeAddress) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_OTA_BOOTLOAD_CLUSTER_ID,\
                    ZCL_IMAGE_BLOCK_REQUEST_COMMAND_ID,\
                    "uvvwwu8",\
                    fieldControl,\
                    manufacturerId,\
                    imageType,\
                    fileVersion,\
                    fileOffset,\
                    maxDataSize,\
                    requestNodeAddress\
```

```
02764                   );
02765
02766
02781 #define emberAfFillCommandOtaBootloadClusterImagePageRequest(fieldControl,manufacturerId,imageType,
fileVersion,fileOffset,maxDataSize,pageSize,responseSpacing,requestNodeAddress) \
02782   emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
02783                   ZCL_OTA_BOOTLOAD_CLUSTER_ID,\
02784                   ZCL_IMAGE_PAGE_REQUEST_COMMAND_ID,\
02785                   "uvvwwuvv8",\
02786                   fieldControl,\
02787                   manufacturerId,\
02788                   imageType,\
02789                   fileVersion,\
02790                   fileOffset,\
02791                   maxDataSize,\
02792                   pageSize,\
02793                   responseSpacing,\
02794                   requestNodeAddress\
02795                   );
02796
02797
02810 #define emberAfFillCommandOtaBootloadClusterImageBlockResponse(status,manufacturerId,imageType,
fileVersion,fileOffset,dataSize,imageData) \
02811   emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
02812                   ZCL_OTA_BOOTLOAD_CLUSTER_ID,\
02813                   ZCL_IMAGE_BLOCK_RESPONSE_COMMAND_ID,\
02814                   "uvvwwus",\
02815                   status,\
02816                   manufacturerId,\
02817                   imageType,\
02818                   fileVersion,\
02819                   fileOffset,\
02820                   dataSize,\
02821                   imageData\
02822                   );
02823
02824
02834 #define emberAfFillCommandOtaBootloadClusterUpgradeEndRequest(status,manufacturerId,imageType,fileVersion) \
02835   emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
02836                   ZCL_OTA_BOOTLOAD_CLUSTER_ID,\
02837                   ZCL_UPGRADE_END_REQUEST_COMMAND_ID,\
02838                   "uvvw",\
02839                   status,\
02840                   manufacturerId,\
02841                   imageType,\
02842                   fileVersion\
02843                   );
02844
02845
02856 #define emberAfFillCommandOtaBootloadClusterUpgradeEndResponse(manufacturerId,imageType,
fileVersion,currentTime,upgradeTime) \
02857   emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
02858                   ZCL_OTA_BOOTLOAD_CLUSTER_ID,\
02859                   ZCL_UPGRADE_END_RESPONSE_COMMAND_ID,\
02860                   "vvwww",\
```

```
                    manufacturerId,\
                    imageType,\
                    fileVersion,\
                    currentTime,\
                    upgradeTime\
                    );


#define emberAfFillCommandOtaBootloadClusterQuerySpecificFileRequest(requestNodeAddress,
manufacturerId,imageType,fileVersion,currentZigbeeStackVersion) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_OTA_BOOTLOAD_CLUSTER_ID,\
                    ZCL_QUERY_SPECIFIC_FILE_REQUEST_COMMAND_ID,\
                    "8vvwv",\
                    requestNodeAddress,\
                    manufacturerId,\
                    imageType,\
                    fileVersion,\
                    currentZigbeeStackVersion\
                    );


#define emberAfFillCommandOtaBootloadClusterQuerySpecificFileResponse(status,manufacturerId,
imageType,fileVersion,imageSize) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_OTA_BOOTLOAD_CLUSTER_ID,\
                    ZCL_QUERY_SPECIFIC_FILE_RESPONSE_COMMAND_ID,\
                    "uvvww",\
                    status,\
                    manufacturerId,\
                    imageType,\
                    fileVersion,\
                    imageSize\
                    );


// @{
#define emberAfFillCommandDemandResponseLoadControlClusterLoadControlEvent(issuerEventId,
deviceClass,utilityEnrollmentGroup,startTime,durationInMinutes,criticalityLevel,
coolingTemperatureOffset,heatingTemperatureOffset,coolingTemperatureSetPoint,
heatingTemperatureSetPoint,averageLoadAdjustmentPercentage,dutyCycle,eventControl) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID,\
                    ZCL_LOAD_CONTROL_EVENT_COMMAND_ID,\
                    "wvuwvuuuvvuuu",\
                    issuerEventId,\
                    deviceClass,\
                    utilityEnrollmentGroup,\
                    startTime,\
                    durationInMinutes,\
                    criticalityLevel,\
                    coolingTemperatureOffset,\
                    heatingTemperatureOffset,\
                    coolingTemperatureSetPoint,\
                    heatingTemperatureSetPoint,\
                    averageLoadAdjustmentPercentage,\
                    dutyCycle,\
```

```
                    eventControl\
                    );


#define emberAfFillCommandDemandResponseLoadControlClusterCancelLoadControlEvent(issuerEventId,
deviceClass,utilityEnrollmentGroup,cancelControl,effectiveTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID,\
                    ZCL_CANCEL_LOAD_CONTROL_EVENT_COMMAND_ID,\
                    "wvuuw",\
                    issuerEventId,\
                    deviceClass,\
                    utilityEnrollmentGroup,\
                    cancelControl,\
                    effectiveTime\
                    );


#define emberAfFillCommandDemandResponseLoadControlClusterCancelAllLoadControlEvents(cancelControl) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID,\
                    ZCL_CANCEL_ALL_LOAD_CONTROL_EVENTS_COMMAND_ID,\
                    "u",\
                    cancelControl\
                    );


#define emberAfFillCommandDemandResponseLoadControlClusterReportEventStatus(issuerEventId,
eventStatus,eventStatusTime,criticalityLevelApplied,coolingTemperatureSetPointApplied,
heatingTemperatureSetPointApplied,averageLoadAdjustmentPercentageApplied,dutyCycleApplied,eventControl,
signatureType,signature) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID,\
                    ZCL_REPORT_EVENT_STATUS_COMMAND_ID,\
                    "wuwuvvuuuub",\
                    issuerEventId,\
                    eventStatus,\
                    eventStatusTime,\
                    criticalityLevelApplied,\
                    coolingTemperatureSetPointApplied,\
                    heatingTemperatureSetPointApplied,\
                    averageLoadAdjustmentPercentageApplied,\
                    dutyCycleApplied,\
                    eventControl,\
                    signatureType,\
                    signature,\
                    42\
                    );


#define emberAfFillCommandDemandResponseLoadControlClusterGetScheduledEvents(startTime,numberOfEvents) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID,\
                    ZCL_GET_SCHEDULED_EVENTS_COMMAND_ID,\
                    "wu",\
```

```
                    startTime,\
                    numberOfEvents\
                    );


// @{
#define emberAfFillCommandSimpleMeteringClusterGetProfileResponse(endTime,status,profileIntervalPeriod,numberOfPeriodsDelivered,intervals,intervalsLen) \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_SIMPLE_METERING_CLUSTER_ID,\
                    ZCL_GET_PROFILE_RESPONSE_COMMAND_ID,\
                    "wuuub",\
                    endTime,\
                    status,\
                    profileIntervalPeriod,\
                    numberOfPeriodsDelivered,\
                    intervals,\
                    intervalsLen\
                    );


#define emberAfFillCommandSimpleMeteringClusterRequestMirror() \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_SIMPLE_METERING_CLUSTER_ID,\
                    ZCL_REQUEST_MIRROR_COMMAND_ID,\
                    ""\
                    );


#define emberAfFillCommandSimpleMeteringClusterRemoveMirror() \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_SIMPLE_METERING_CLUSTER_ID,\
                    ZCL_REMOVE_MIRROR_COMMAND_ID,\
                    ""\
                    );


#define emberAfFillCommandSimpleMeteringClusterRequestFastPollModeResponse(appliedUpdatePeriod,fastPollModeEndtime) \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_SIMPLE_METERING_CLUSTER_ID,\
                    ZCL_REQUEST_FAST_POLL_MODE_RESPONSE_COMMAND_ID,\
                    "uw",\
                    appliedUpdatePeriod,\
                    fastPollModeEndtime\
                    );


#define emberAfFillCommandSimpleMeteringClusterGetProfile(intervalChannel,endTime,numberOfPeriods) \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_SIMPLE_METERING_CLUSTER_ID,\
                    ZCL_GET_PROFILE_COMMAND_ID,\
                    "uwu",\
                    intervalChannel,\
```

```
                    endTime,\
                    numberOfPeriods\
                    );


#define emberAfFillCommandSimpleMeteringClusterRequestMirrorResponse(endpointId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_SIMPLE_METERING_CLUSTER_ID,\
                    ZCL_REQUEST_MIRROR_RESPONSE_COMMAND_ID,\
                    "v",\
                    endpointId\
                    );


#define emberAfFillCommandSimpleMeteringClusterMirrorRemoved(endpointId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_SIMPLE_METERING_CLUSTER_ID,\
                    ZCL_MIRROR_REMOVED_COMMAND_ID,\
                    "v",\
                    endpointId\
                    );


#define emberAfFillCommandSimpleMeteringClusterRequestFastPollMode(fastPollUpdatePeriod,duration) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_SIMPLE_METERING_CLUSTER_ID,\
                    ZCL_REQUEST_FAST_POLL_MODE_COMMAND_ID,\
                    "uu",\
                    fastPollUpdatePeriod,\
                    duration\
                    );


// @{
#define emberAfFillCommandPriceClusterPublishPrice(providerId,rateLabel,issuerEventId,currentTime,
unitOfMeasure,currency,priceTrailingDigitAndPriceTier,numberOfPriceTiersAndRegisterTier,startTime,durationInMinutes,
price,priceRatio,generationPrice,generationPriceRatio,alternateCostDelivered,alternateCostUnit,
alternateCostTrailingDigit,numberOfBlockThresholds,priceControl) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_PRICE_CLUSTER_ID,\
                    ZCL_PUBLISH_PRICE_COMMAND_ID,\
                    "wswwuvuuwvwuwuwuuuu",\
                    providerId,\
                    rateLabel,\
                    issuerEventId,\
                    currentTime,\
                    unitOfMeasure,\
                    currency,\
                    priceTrailingDigitAndPriceTier,\
                    numberOfPriceTiersAndRegisterTier,\
                    startTime,\
                    durationInMinutes,\
                    price,\
                    priceRatio,\
                    generationPrice,\
                    generationPriceRatio,\
```

```
                    alternateCostDelivered,\
                    alternateCostUnit,\
                    alternateCostTrailingDigit,\
                    numberOfBlockThresholds,\
                    priceControl\
                    );


#define emberAfFillCommandPriceClusterPublishBlockPeriod(providerId,issuerEventId,blockPeriodStartTime,blockPeriodDurationInMinutes,numberOfPriceTiersAndNumberOfBlockThresholds,blockPeriodControl) \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_PRICE_CLUSTER_ID,\
                    ZCL_PUBLISH_BLOCK_PERIOD_COMMAND_ID,\
                    "www3uu",\
                    providerId,\
                    issuerEventId,\
                    blockPeriodStartTime,\
                    blockPeriodDurationInMinutes,\
                    numberOfPriceTiersAndNumberOfBlockThresholds,\
                    blockPeriodControl\
                    );


#define emberAfFillCommandPriceClusterGetCurrentPrice(commandOptions) \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_PRICE_CLUSTER_ID,\
                    ZCL_GET_CURRENT_PRICE_COMMAND_ID,\
                    "u",\
                    commandOptions\
                    );


#define emberAfFillCommandPriceClusterGetScheduledPrices(startTime,numberOfEvents) \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_PRICE_CLUSTER_ID,\
                    ZCL_GET_SCHEDULED_PRICES_COMMAND_ID,\
                    "wu",\
                    startTime,\
                    numberOfEvents\
                    );


#define emberAfFillCommandPriceClusterPriceAcknowledgement(providerId,issuerEventId,priceAckTime,control) \
  emberAfFillExternalBuffer(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_PRICE_CLUSTER_ID,\
                    ZCL_PRICE_ACKNOWLEDGEMENT_COMMAND_ID,\
                    "wwwu",\
                    providerId,\
                    issuerEventId,\
                    priceAckTime,\
                    control\
                    );


#define emberAfFillCommandPriceClusterGetBlockPeriods(startTime,numberOfEvents) \
```

```
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_PRICE_CLUSTER_ID,\
                        ZCL_GET_BLOCK_PERIODS_COMMAND_ID,\
                        "wu",\
                        startTime,\
                        numberOfEvents\
                        );


// @{
#define emberAfFillCommandMessagingClusterDisplayMessage(messageId,messageControl,startTime,
durationInMinutes,message) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                        ZCL_MESSAGING_CLUSTER_ID,\
                        ZCL_DISPLAY_MESSAGE_COMMAND_ID,\
                        "wuwvs",\
                        messageId,\
                        messageControl,\
                        startTime,\
                        durationInMinutes,\
                        message\
                        );


#define emberAfFillCommandMessagingClusterCancelMessage(messageId,messageControl) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                        ZCL_MESSAGING_CLUSTER_ID,\
                        ZCL_CANCEL_MESSAGE_COMMAND_ID,\
                        "wu",\
                        messageId,\
                        messageControl\
                        );


#define emberAfFillCommandMessagingClusterGetLastMessage() \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_MESSAGING_CLUSTER_ID,\
                        ZCL_GET_LAST_MESSAGE_COMMAND_ID,\
                        ""\
                        );


#define emberAfFillCommandMessagingClusterMessageConfirmation(messageId,confirmationTime) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                        ZCL_MESSAGING_CLUSTER_ID,\
                        ZCL_MESSAGE_CONFIRMATION_COMMAND_ID,\
                        "ww",\
                        messageId,\
                        confirmationTime\
                        );


// @{
#define emberAfFillCommandTunnelingClusterRequestTunnel(protocolId,manufacturerCode,flowControlSupport) \
```

```
emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_TUNNELING_CLUSTER_ID,\
                ZCL_REQUEST_TUNNEL_COMMAND_ID,\
                "uvu",\
                protocolId,\
                manufacturerCode,\
                flowControlSupport\
                );


#define emberAfFillCommandTunnelingClusterCloseTunnel(tunnelId) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_TUNNELING_CLUSTER_ID,\
                ZCL_CLOSE_TUNNEL_COMMAND_ID,\
                "v",\
                tunnelId\
                );


#define emberAfFillCommandTunnelingClusterTransferDataClientToServer(tunnelId,data,dataLen) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_TUNNELING_CLUSTER_ID,\
                ZCL_TRANSFER_DATA_CLIENT_TO_SERVER_COMMAND_ID,\
                "vb",\
                tunnelId,\
                data,\
                dataLen\
                );


#define emberAfFillCommandTunnelingClusterTransferDataErrorClientToServer(tunnelId,transferDataStatus) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_TUNNELING_CLUSTER_ID,\
                ZCL_TRANSFER_DATA_ERROR_CLIENT_TO_SERVER_COMMAND_ID,\
                "vu",\
                tunnelId,\
                transferDataStatus\
                );


#define emberAfFillCommandTunnelingClusterAckTransferDataClientToServer(tunnelId,numberOfBytesLeft) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_TUNNELING_CLUSTER_ID,\
                ZCL_ACK_TRANSFER_DATA_CLIENT_TO_SERVER_COMMAND_ID,\
                "vv",\
                tunnelId,\
                numberOfBytesLeft\
                );


#define emberAfFillCommandTunnelingClusterReadyDataClientToServer(tunnelId,numberOfOctetsLeft) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_TUNNELING_CLUSTER_ID,\
```

```
                    ZCL_READY_DATA_CLIENT_TO_SERVER_COMMAND_ID,\
                    "vv",\
                    tunnelId,\
                    numberOfOctetsLeft\
                    );


#define emberAfFillCommandTunnelingClusterRequestTunnelResponse(tunnelId,tunnelStatus) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_TUNNELING_CLUSTER_ID,\
                    ZCL_REQUEST_TUNNEL_RESPONSE_COMMAND_ID,\
                    "vu",\
                    tunnelId,\
                    tunnelStatus\
                    );


#define emberAfFillCommandTunnelingClusterTransferDataServerToClient(tunnelId,data,dataLen) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_TUNNELING_CLUSTER_ID,\
                    ZCL_TRANSFER_DATA_SERVER_TO_CLIENT_COMMAND_ID,\
                    "vb",\
                    tunnelId,\
                    data,\
                    dataLen\
                    );


#define emberAfFillCommandTunnelingClusterTransferDataErrorServerToClient(tunnelId,transferDataStatus) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_TUNNELING_CLUSTER_ID,\
                    ZCL_TRANSFER_DATA_ERROR_SERVER_TO_CLIENT_COMMAND_ID,\
                    "vu",\
                    tunnelId,\
                    transferDataStatus\
                    );


#define emberAfFillCommandTunnelingClusterAckTransferDataServerToClient(tunnelId,numberOfBytesLeft) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_TUNNELING_CLUSTER_ID,\
                    ZCL_ACK_TRANSFER_DATA_SERVER_TO_CLIENT_COMMAND_ID,\
                    "vv",\
                    tunnelId,\
                    numberOfBytesLeft\
                    );


#define emberAfFillCommandTunnelingClusterReadyDataServerToClient(tunnelId,numberOfOctetsLeft) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_TUNNELING_CLUSTER_ID,\
                    ZCL_READY_DATA_SERVER_TO_CLIENT_COMMAND_ID,\
                    "vv",\
                    tunnelId,\
```

```
                    numberOfOctetsLeft\
                    );


// @{
#define emberAfFillCommandPrepaymentClusterSelectAvailableEmergencyCredit(commandDateTime,
originatingDevice,siteId,meterSerialNumber) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_PREPAYMENT_CLUSTER_ID,\
                    ZCL_SELECT_AVAILABLE_EMERGENCY_CREDIT_COMMAND_ID,\
                    "wuss",\
                    commandDateTime,\
                    originatingDevice,\
                    siteId,\
                    meterSerialNumber\
                    );


#define emberAfFillCommandPrepaymentClusterChangeSupply(providerId,requestDateTime,siteId,
meterSerialNumber,implementationDateTime,proposedSupplyStatus,originatorIdSupplyControlBits) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_PREPAYMENT_CLUSTER_ID,\
                    ZCL_CHANGE_SUPPLY_COMMAND_ID,\
                    "wwsswuu",\
                    providerId,\
                    requestDateTime,\
                    siteId,\
                    meterSerialNumber,\
                    implementationDateTime,\
                    proposedSupplyStatus,\
                    originatorIdSupplyControlBits\
                    );


#define emberAfFillCommandPrepaymentClusterSupplyStatusResponse(providerId,
implementationtDateTime,supplyStatus) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_PREPAYMENT_CLUSTER_ID,\
                    ZCL_SUPPLY_STATUS_RESPONSE_COMMAND_ID,\
                    "wwu",\
                    providerId,\
                    implementationtDateTime,\
                    supplyStatus\
                    );


// @{
#define emberAfFillCommandKeyEstablishmentClusterInitiateKeyEstablishmentRequest
(keyEstablishmentSuite,ephemeralDataGenerateTime,confirmKeyGenerateTime,identity) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_KEY_ESTABLISHMENT_CLUSTER_ID,\
                    ZCL_INITIATE_KEY_ESTABLISHMENT_REQUEST_COMMAND_ID,\
                    "vuub",\
                    keyEstablishmentSuite,\
                    ephemeralDataGenerateTime,\
                    confirmKeyGenerateTime,\
```

```
                identity,\
                48\
                );


#define emberAfFillCommandKeyEstablishmentClusterEphemeralDataRequest(ephemeralData) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_KEY_ESTABLISHMENT_CLUSTER_ID,\
                ZCL_EPHEMERAL_DATA_REQUEST_COMMAND_ID,\
                "b",\
                ephemeralData,\
                22\
                );


#define emberAfFillCommandKeyEstablishmentClusterConfirmKeyDataRequest(secureMessageAuthenticationCode) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_KEY_ESTABLISHMENT_CLUSTER_ID,\
                ZCL_CONFIRM_KEY_DATA_REQUEST_COMMAND_ID,\
                "G",\
                secureMessageAuthenticationCode\
                );


#define emberAfFillCommandKeyEstablishmentClusterServerToClientTerminateKeyEstablishment(statusCode,
waitTime,keyEstablishmentSuite) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                ZCL_KEY_ESTABLISHMENT_CLUSTER_ID,\
                ZCL_TERMINATE_KEY_ESTABLISHMENT_COMMAND_ID,\
                "uuv",\
                statusCode,\
                waitTime,\
                keyEstablishmentSuite\
                );


#define emberAfFillCommandKeyEstablishmentClusterClientToServerTerminateKeyEstablishment(statusCode,
waitTime,keyEstablishmentSuite) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                ZCL_KEY_ESTABLISHMENT_CLUSTER_ID,\
                ZCL_TERMINATE_KEY_ESTABLISHMENT_COMMAND_ID,\
                "uuv",\
                statusCode,\
                waitTime,\
                keyEstablishmentSuite\
                );


#define
emberAfFillCommandKeyEstablishmentClusterInitiateKeyEstablishmentResponse
(requestedKeyEstablishmentSuite,ephemeralDataGenerateTime,confirmKeyGenerateTime,identity) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                ZCL_KEY_ESTABLISHMENT_CLUSTER_ID,\
                ZCL_INITIATE_KEY_ESTABLISHMENT_RESPONSE_COMMAND_ID,\
```

```
                    "vuub",\
                    requestedKeyEstablishmentSuite,\
                    ephemeralDataGenerateTime,\
                    confirmKeyGenerateTime,\
                    identity,\
                    48\
                    );


#define emberAfFillCommandKeyEstablishmentClusterEphemeralDataResponse(ephemeralData) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_KEY_ESTABLISHMENT_CLUSTER_ID,\
                    ZCL_EPHEMERAL_DATA_RESPONSE_COMMAND_ID,\
                    "b",\
                    ephemeralData,\
                    22\
                    );


#define emberAfFillCommandKeyEstablishmentClusterConfirmKeyDataResponse(secureMessageAuthenticationCode) \
  emberAfFillExternalBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_SERVER_TO_CLIENT,\
                    ZCL_KEY_ESTABLISHMENT_CLUSTER_ID,\
                    ZCL_CONFIRM_KEY_DATA_RESPONSE_COMMAND_ID,\
                    "G",\
                    secureMessageAuthenticationCode\
                    );


// @{
#define emberAfFillCommandSampleMfgSpecificClusterCommandOne(argOne) \
  emberAfFillExternalManufacturerSpecificBuffer
(ZCL_CLUSTER_SPECIFIC_COMMAND|ZCL_FRAME_CONTROL_CLIENT_TO_SERVER,\
                    ZCL_SAMPLE_MFG_SPECIFIC_CLUSTER_ID,\
                    0x1002,\
                    ZCL_COMMAND_ONE_COMMAND_ID,\
                    "u",\
                    argOne\
                    );


#endif // CLUSTER_CLIENT_API
```

af-structs.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generateStructs()
//

// Enclosing macro to prevent multiple inclusion
#ifndef __EMBER_AF_STRUCTS__
#define __EMBER_AF_STRUCTS__


// Generated structs from the metadata
// Struct for ReadAttributeStatusRecord
typedef struct _ReadAttributeStatusRecord {
  int16u attributeId;
  int8u status;
  int8u attributeType;
  int8u attributeLocation;
} ReadAttributeStatusRecord;


// Struct for WriteAttributeRecord
typedef struct _WriteAttributeRecord {
  int16u attributeId;
  int8u attributeType;
  int8u attributeLocation;
} WriteAttributeRecord;


// Struct for WriteAttributeStatusRecord
typedef struct _WriteAttributeStatusRecord {
  int8u status;
  int16u attributeId;
} WriteAttributeStatusRecord;


// Struct for ConfigureReportingRecord
typedef struct _ConfigureReportingRecord {
```

```
00041   int8u direction;
00042   int16u attributeId;
00043   int8u attributeType;
00044   int16u minimumReportingInterval;
00045   int16u maximumReportingInterval;
00046   int8u reportableChangeLocation;
00047   int16u timeoutPeriod;
00048 } ConfigureReportingRecord;
00049
00050
00051 // Struct for ConfigureReportingStatusRecord
00052 typedef struct _ConfigureReportingStatusRecord {
00053   int8u status;
00054   int8u direction;
00055   int16u attributeId;
00056 } ConfigureReportingStatusRecord;
00057
00058
00059 // Struct for ReadReportingConfigurationRecord
00060 typedef struct _ReadReportingConfigurationRecord {
00061   int8u status;
00062   int8u direction;
00063   int16u attributeId;
00064   int8u attributeType;
00065   int16u minimumReportingInterval;
00066   int16u maximumReportingInterval;
00067   int8u reportableChangeLocation;
00068   int16u timeoutPeriod;
00069 } ReadReportingConfigurationRecord;
00070
00071
00072 // Struct for ReadReportingConfigurationAttributeRecord
00073 typedef struct _ReadReportingConfigurationAttributeRecord {
00074   int8u direction;
00075   int16u attributeId;
00076 } ReadReportingConfigurationAttributeRecord;
00077
00078
00079 // Struct for ReportAttributeRecord
00080 typedef struct _ReportAttributeRecord {
00081   int16u attributeId;
00082   int8u attributeType;
00083   int8u attributeLocation;
00084 } ReportAttributeRecord;
00085
```

```
00086
00087 // Struct for DiscoverAttributesInfoRecord
00088 typedef struct _DiscoverAttributesInfoRecord {
00089   int16u attributeId;
00090   int8u attributeType;
00091 } DiscoverAttributesInfoRecord;
00092
00093
00094 // Struct for SceneExtensionAttributeInfo
00095 typedef struct _SceneExtensionAttributeInfo {
00096   int8u attributeType;
00097   int8u attributeLocation;
00098 } SceneExtensionAttributeInfo;
00099
00100
00101 // Struct for SceneExtensionFieldSet
00102 typedef struct _SceneExtensionFieldSet {
00103   int16u clusterId;
00104   int8u length;
00105   int8u value;
00106 } SceneExtensionFieldSet;
00107
00108
00109 // Struct for BlockThreshold
00110 typedef struct _BlockThreshold {
00111   int8u blockThreshold;
00112   int8u priceControl;
00113   int32u blockPeriodStartTime;
00114   int32u blockPeriodDurationMinutes;
00115   int8u fuelType;
00116   int32u standingCharge;
00117 } BlockThreshold;
00118
00119
00120 // Struct for Notification
00121 typedef struct _Notification {
00122   int16u contentId;
00123   int8u statusFeedback;
00124 } Notification;
00125
00126
00127 // Struct for NeighborInfo
00128 typedef struct _NeighborInfo {
00129   int8u* neighbor;
00130   int16s x;
```

```
  int16s y;
  int16s z;
  int8s rssi;
  int8u numberRssiMeasurements;
} NeighborInfo;


// Struct for ChatParticipant
typedef struct _ChatParticipant {
  int16u uid;
  int8u* nickname;
} ChatParticipant;


// Struct for ChatRoom
typedef struct _ChatRoom {
  int16u cid;
  int8u* name;
} ChatRoom;


// Struct for NodeInformation
typedef struct _NodeInformation {
  int16u uid;
  int16u address;
  int8u endpoint;
  int8u* nickname;
} NodeInformation;


// Void typedef for Identity which is empty.
// this will result in all the references to the data being as int8u*
typedef int8u Identity;


// Void typedef for EphemeralData which is empty.
// this will result in all the references to the data being as int8u*
typedef int8u EphemeralData;


// Void typedef for Smac which is empty.
// this will result in all the references to the data being as int8u*
typedef int8u Smac;


// Void typedef for Signature which is empty.
```

```
// this will result in all the references to the data being as int8u*
typedef int8u Signature;


#endif // __EMBER_AF_STRUCTS__
```

ota-storage-linux.h

Go to the documentation of this file.

```
// Internal definitions for the OTA storage Linux.

typedef void (EmAfOtaStorageFileAddedHandler)(const EmberAfOtaHeader*);

typedef struct {
  boolean memoryDebug;
  boolean fileDebug;
  boolean fieldDebug;
  boolean ignoreFilesWithUnderscorePrefix;
  boolean printFileDiscoveryOrRemoval;
  EmAfOtaStorageFileAddedHandler* fileAddedHandler;
} EmAfOtaStorageLinuxConfig;

void emAfOtaStorageGetConfig(EmAfOtaStorageLinuxConfig* currentConfig);
void emAfOtaStorageSetConfig(const EmAfOtaStorageLinuxConfig* newConfig);

```

ias-zone-table.h

Go to the documentation of this file.

```
// ********************************************************************
// * ias-zone-table.h
// *
// * The IAS ZONE Table maintains information related to the enrollment
// * of security devices as defined by the ZCL Security and Safety
// * Specification. (Document 075123r00ZB)
// *
// * This table provides for the functions required by the specification
// * including, enroll, unenroll, retrieval of the Zone Id Map and
// * retrieval of specific zone enrollment information.
// *
// * This table is intended to be included with any IAS Zone or ACE
// * client responsible for maintaining information related to zone
// * enrollment.
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************

#define ZONE_TABLE_SIZE 16
#define ZONE_ID_MAP_SECTION_SIZE 16
#define ZONE_TABLE_MAP_SIZE ZONE_TABLE_SIZE / ZONE_ID_MAP_SECTION_SIZE
#define ZONE_TABLE_VALUE_DEFAULT_8 0xff
#define ZONE_TABLE_VALUE_DEFAULT_16 0xffff

typedef struct {
  int16u zoneType;
  int8u ieeeAdd[8];
} EmberAfIasZoneTableEntry;

void enroll(int8u zoneId, int16u zoneType, int8u* ieeeAdd);
void unenroll(int8u zoneId);
void initZoneTable(void);
void getZoneIdMap(int16u *map);
EmberAfIasZoneTableEntry *getZoneInfo(int8u zoneId);
EmberAfStatus getNextZoneId(int8u* zoneId);
void printZoneIdMap(int16u *map);
void printZoneInfo(int8u zoneId);
void printZoneTable(void);
```

messaging-server.h

Go to the documentation of this file.

```
// ********************************************************************
// * messaging-server.h
// *
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************

// ----------------------------------------------------------------------
// Message Control byte
// ----------------------------------------------------------------------

#define ZCL_MESSAGING_CLUSTER_TRANSMISSION_MASK (BIT(1) | BIT(0))
#define ZCL_MESSAGING_CLUSTER_IMPORTANCE_MASK   (BIT(3) | BIT(2))
#define ZCL_MESSAGING_CLUSTER_CONFIRMATION_MASK BIT(7)

#define ZCL_MESSAGING_CLUSTER_START_TIME_NOW        0x00000000UL
#define ZCL_MESSAGING_CLUSTER_END_TIME_NEVER        0xFFFFFFFFUL
#define ZCL_MESSAGING_CLUSTER_DURATION_UNTIL_CHANGED 0xFFFF

typedef struct {
  boolean valid;
  int32u  messageId;
  int8u   messageControl;
  int32u  startTime;
  int16u  durationInMinutes;
  int8u   message[EMBER_AF_PLUGIN_MESSAGING_SERVER_MESSAGE_SIZE + 1];
} EmberAfMessagingServerMessage;

void emAfPluginMessagingServerPrintInfo(void);
EmberAfMessagingServerMessage *emAfPluginMessagingServerGetMessageStruct(void);
```

price-server.h

Go to the documentation of this file.

```
// ******************************************************************
// * price.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.            *80*
// ******************************************************************

#ifndef EMBER_AF_PLUGIN_PRICE_SERVER_PRICE_TABLE_SIZE
  #define EMBER_AF_PLUGIN_PRICE_SERVER_PRICE_TABLE_SIZE 5
#endif

#define ZCL_PRICE_CLUSTER_PRICE_ACKNOWLEDGEMENT_MASK BIT(0)

#define ZCL_PRICE_CLUSTER_START_TIME_NOW        0x00000000UL
#define ZCL_PRICE_CLUSTER_END_TIME_NEVER        0xFFFFFFFFUL
#define ZCL_PRICE_CLUSTER_DURATION_UNTIL_CHANGED 0xFFFF
#define ZCL_PRICE_CLUSTER_NUMBER_OF_EVENTS_ALL   0x00

// This struct is used to store the scheduled price.
typedef struct {
  int32u  providerId;
  int8u   rateLabel[ZCL_PRICE_CLUSTER_MAXIMUM_RATE_LABEL_LENGTH + 1];
  int32u  issuerEventID;
  int8u   unitOfMeasure;
  int16u  currency;
  int8u   priceTrailingDigitAndTier;
  int8u   numberOfPriceTiersAndTier; // added later in errata
  int32u  startTime;
  int16u  duration; // in minutes
  int32u  price;
  int8u   priceRatio;
  int32u  generationPrice;
  int8u   generationPriceRatio;
  int32u  alternateCostDelivered;
  int8u   alternateCostUnit;
  int8u   alternateCostTrailingDigit;
  int8u   numberOfBlockThresholds;
```

```
  int8u   priceControl;
  boolean valid; // for use in the price table
} EmberAfScheduledPrice;

#define ZCL_PRICE_INVALID_INDEX 0xFF

// these are for inputting prices into a price table
void emberAfPriceClearPriceTable(void);
EmberAfScheduledPrice* emberAfPriceGetPriceTableEntry(int8u index);
void emberAfPricePrintTable(void);
EmberAfScheduledPrice* emberAfGetCurrentPrice(void);
```

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- a -

- address : **_NodeInformation**
- addressTableIndex : **EmberAfIncomingMessage**
- alternateCostDelivered : **EmberAfScheduledPrice**
- alternateCostTrailingDigit : **EmberAfScheduledPrice**
- alternateCostUnit : **EmberAfScheduledPrice**
- apsFrame : **EmberAfIncomingMessage** , **EmberAfClusterCommand**
- attributeCount : **EmberAfCluster**
- attributeId : **_ConfigureReportingStatusRecord** , **_ReadReportingConfigurationRecord** , **EmberAfAttributeMetadata** , **_ReadReportingConfigurationAttributeRecord** , **_ReportAttributeRecord** , **EmberAfAttributeSearchRecord** , **_DiscoverAttributesInfoRecord** , **EmberAfReportEntry** , **_ReadAttributeStatusRecord** , **_WriteAttributeRecord** , **_WriteAttributeStatusRecord** , **_ConfigureReportingRecord**

- attributeLocation : **_ReadAttributeStatusRecord** , **_ReportAttributeRecord** , **_WriteAttributeRecord** , **_SceneExtensionAttributeInfo**
- attributes : **EmberAfCluster**
- attributeType : **_ReadAttributeStatusRecord** , **_SceneExtensionAttributeInfo** , **_ConfigureReportingRecord** , **_ReportAttributeRecord** , **_DiscoverAttributesInfoRecord** , **_ReadReportingConfigurationRecord** , **_WriteAttributeRecord** , **EmberAfAttributeMetadata**
- avgLoadPercentage : **EmberAfLoadControlEvent**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- b -

- bindingTableIndex : **EmberAfIncomingMessage**
- blockPeriodDurationMinutes : **_BlockThreshold**
- blockPeriodStartTime : **_BlockThreshold**
- blockThreshold : **_BlockThreshold**
- buffer : **EmberAfClusterCommand**
- bufLen : **EmberAfClusterCommand**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- c -

- cid : **_ChatRoom**
- cluster : **EmberAfEndpointType**
- clusterCount : **EmberAfEndpointType**
- clusterId : **EmberAfCluster** , **EmberAfEventContext** , **EmberAfReportEntry** , **EmberAfAttributeSearchRecord** , **_SceneExtensionFieldSet** , **EmberAfInterpanHeader** , **EmberAfAllowedInterPanMessage**
- clusterMask : **EmberAfAttributeSearchRecord**
- clusterSize : **EmberAfCluster**
- clusterSpecific : **EmberAfClusterCommand**
- commandId : **EmberAfAllowedInterPanMessage** , **EmberAfClusterCommand**
- contentId : **_Notification**
- coolingTempOffset : **EmberAfLoadControlEvent**
- coolingTempSetPoint : **EmberAfLoadControlEvent**

- count : **EmberAfEndpointList**
- criticalityLevel : **EmberAfLoadControlEvent**
- currency : **EmberAfScheduledPrice**
- currentInterval : **EmberAfReportEntry**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- d -

- defaultValue : **EmberAfDefaultAttributeValue** , **EmberAfAttributeMinMaxValue** , **EmberAfAttributeMetadata** , **EmberAfDefaultOrMinMaxAttributeValue**
- destinationEndpoint : **EmberAfLoadControlEvent** , **EmberAfReportEntry**
- deviceClass : **EmberAfLoadControlEvent**
- deviceId : **EmberAfDefinedEndpoint**
- deviceSpecificFileEui64 : **EmberAfOtaImageId**
- deviceVersion : **EmberAfDefinedEndpoint**
- direction : **EmberAfClusterCommand** , **_ReadReportingConfigurationAttributeRecord** , **_ConfigureReportingRecord** , **_ReadReportingConfigurationRecord** , **_ConfigureReportingStatusRecord** , **EmberAfReportEntry**
- duration : **EmberAfLoadControlEvent** , **EmberAfScheduledPrice**
- durationInMinutes : **EmberAfMessagingServerMessage**

- dutyCycle : **EmberAfLoadControlEvent**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- e -

- endpoint : **EmberAfAttributeSearchRecord** , **EmberAfDefinedEndpoint** , **EmberAfSceneTableEntry** , **_NodeInformation** , **EmberAfEventContext**
- endpointSize : **EmberAfEndpointType**
- endpointType : **EmberAfDefinedEndpoint**
- endRand : **EmberAfLoadControlEvent**
- eventControl : **EmberAfLoadControlEvent** , **EmberAfEventContext**
- eventId : **EmberAfLoadControlEvent**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- f -

- failCount : **EmberAfReportEntry**
- fieldControl : **EmberAfOtaHeader**
- fieldDebug : **EmAfOtaStorageLinuxConfig**
- fileAddedHandler : **EmAfOtaStorageLinuxConfig**
- fileDebug : **EmAfOtaStorageLinuxConfig**
- firmwareVersion : **EmberAfOtaHeader** , **EmberAfOtaImageId**
- fuelType : **_BlockThreshold**
- functions : **EmberAfCluster**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- g -

- generationPrice : **EmberAfScheduledPrice**
- generationPriceRatio : **EmberAfScheduledPrice**
- groupId : **EmberAfInterpanHeader** , **EmberAfSceneTableEntry**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- h -

- headerLength : **EmberAfOtaHeader**
- headerString : **EmberAfOtaHeader**
- headerVersion : **EmberAfOtaHeader**
- heatingTempOffset : **EmberAfLoadControlEvent**
- heatingTempSetPoint : **EmberAfLoadControlEvent**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- i -

- id : **EmberAfTagData**
- ieeeAdd : **EmberAfIasZoneTableEntry**
- ignoreFilesWithUnderscorePrefix : **EmAfOtaStorageLinuxConfig**
- imageSize : **EmberAfOtaHeader**
- imageTypeId : **EmberAfOtaImageId** , **EmberAfOtaHeader**
- index : **EmberAfManufacturerCodeEntry**
- interPanHeader : **EmberAfClusterCommand**
- inUse : **EmberAfReportEntry**
- isClient : **EmberAfEventContext**
- issuerEventID : **EmberAfScheduledPrice**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- l -

- lastHopLqi : **EmberAfIncomingMessage**
- lastHopRssi : **EmberAfIncomingMessage**
- lastReportValue : **EmberAfReportEntry**
- length : **_SceneExtensionFieldSet** , **EmberAfTagData**
- list : **EmberAfEndpointList**
- longAddress : **EmberAfInterpanHeader**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- m -

- manufacturerCode : **EmberAfAttributeSearchRecord** , **EmberAfManufacturerCodeEntry** , **EmberAfReportEntry**
- manufacturerId : **EmberAfOtaImageId** , **EmberAfOtaHeader**
- mask : **EmberAfReportEntry** , **EmberAfAttributeMetadata** , **EmberAfCluster**
- matchAddress : **EmberAfServiceDiscoveryResult**
- maximumHardwareVersion : **EmberAfOtaHeader**
- maximumReportingInterval : **_ConfigureReportingRecord** , **_ReadReportingConfigurationRecord**
- maxInterval : **EmberAfReportEntry**
- maxValue : **EmberAfAttributeMinMaxValue**
- memoryDebug : **EmAfOtaStorageLinuxConfig**
- message : **EmberAfIncomingMessage** , **EmberAfMessagingServerMessage**
- messageControl : **EmberAfMessagingServerMessage**

- messageId : **EmberAfMessagingServerMessage**
- messageType : **EmberAfInterpanHeader**
- mfgCode : **EmberAfClusterCommand**
- mfgSpecific : **EmberAfClusterCommand**
- minimumHardwareVersion : **EmberAfOtaHeader**
- minimumReportingInterval : **_ReadReportingConfigurationRecord** , **_ConfigureReportingRecord**
- minInterval : **EmberAfReportEntry**
- minValue : **EmberAfAttributeMinMaxValue**
- msgLen : **EmberAfIncomingMessage**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- n -

- name : **_ChatRoom**
- neighbor : **_NeighborInfo**
- nickname : **_NodeInformation** , **_ChatParticipant**
- nodeId : **EmberAfReportEntry**
- numberOfBlockThresholds : **EmberAfScheduledPrice**
- numberOfPriceTiersAndTier : **EmberAfScheduledPrice**
- numberRssiMeasurements : **_NeighborInfo**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- o -

- optionControl : **EmberAfLoadControlEvent**
- options : **EmberAfAllowedInterPanMessage** , **EmberAfInterpanHeader**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- p -

- panId : **EmberAfInterpanHeader**
- payloadStartIndex : **EmberAfClusterCommand**
- price : **EmberAfScheduledPrice**
- priceControl : **_BlockThreshold** , **EmberAfScheduledPrice**
- priceRatio : **EmberAfScheduledPrice**
- priceTrailingDigitAndTier : **EmberAfScheduledPrice**
- printFileDiscoveryOrRemoval : **EmAfOtaStorageLinuxConfig**
- profileId : **EmberAfInterpanHeader** , **EmberAfAllowedInterPanMessage** , **EmberAfDefinedEndpoint**
- providerId : **EmberAfScheduledPrice**
- ptrToDefaultValue : **EmberAfDefaultOrMinMaxAttributeValue** , **EmberAfDefaultAttributeValue**
- ptrToMinMaxValue : **EmberAfDefaultOrMinMaxAttributeValue**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- r -

- rateLabel : **EmberAfScheduledPrice**
- reportableChange : **EmberAfReportEntry**
- reportableChangeLocation : **_ReadReportingConfigurationRecord** , **_ConfigureReportingRecord**
- responseData : **EmberAfServiceDiscoveryResult**
- rssi : **_NeighborInfo**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- s -

- sceneId : **EmberAfSceneTableEntry**
- securityCredentials : **EmberAfOtaHeader**
- seqNum : **EmberAfClusterCommand**
- shortAddress : **EmberAfInterpanHeader**
- size : **EmberAfAttributeMetadata**
- sleepControl : **EmberAfEventContext**
- source : **EmberAfClusterCommand** , **EmberAfLoadControlEvent** , **EmberAfIncomingMessage**
- sourceEndpoint : **EmberAfLoadControlEvent** , **EmberAfReportEntry**
- standingCharge : **_BlockThreshold**
- startRand : **EmberAfLoadControlEvent**
- startTime : **EmberAfMessagingServerMessage** , **EmberAfScheduledPrice** , **EmberAfLoadControlEvent**

- status : **_WriteAttributeStatusRecord** , **_ReadReportingConfigurationRecord** , **_ReadAttributeStatusRecord** , **_ConfigureReportingStatusRecord** , **EmberAfServiceDiscoveryResult**
- statusFeedback : **_Notification**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- t -

- timeout : **EmberAfReportEntry**
- timeoutPeriod : **_ConfigureReportingRecord** , **_ReadReportingConfigurationRecord**
- transitionTime : **EmberAfSceneTableEntry**
- type : **EmberAfIncomingMessage** , **EmberAfClusterCommand**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- u -

- uid : **_ChatParticipant** , **_NodeInformation**
- unitOfMeasure : **EmberAfScheduledPrice**
- upgradeFileDestination : **EmberAfOtaHeader**
- utilityEnrollmentGroup : **EmberAfLoadControlEvent**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- v -

- valid : **EmberAfMessagingServerMessage** , **EmberAfScheduledPrice**
- value : **_SceneExtensionFieldSet**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- x -

- x : **_NeighborInfo**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- y -

- y : **_NeighborInfo**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

Here is a list of all struct and union fields with links to the structures/unions they belong to:

- z -

- z : **_NeighborInfo**
- zdoRequestClusterId : **EmberAfServiceDiscoveryResult**
- zigbeeStackVersion : **EmberAfOtaHeader**
- zoneType : **EmberAfIasZoneTableEntry**

_AF_API.top

Go to the documentation of this file.

```
00001
```

attribute-id.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generateAttributeIDs()
//

// Enclosing macro to prevent multiple inclusion
#ifndef __EMBER_AF_ATTRIBUTE_ID__
#define __EMBER_AF_ATTRIBUTE_ID__


// Attribute types for cluster: Basic
// Cluster specification level: 075123r03

// Server attributes
#define ZCL_VERSION_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_APPLICATION_VERSION_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_STACK_VERSION_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_HW_VERSION_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_MANUFACTURER_NAME_ATTRIBUTE_ID 0x0004    // Ver.: always
#define ZCL_MODEL_IDENTIFIER_ATTRIBUTE_ID 0x0005    // Ver.: always
#define ZCL_DATE_CODE_ATTRIBUTE_ID 0x0006    // Ver.: always
#define ZCL_POWER_SOURCE_ATTRIBUTE_ID 0x0007    // Ver.: always
#define ZCL_LOCATION_DESCRIPTION_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_PHYSICAL_ENVIRONMENT_ATTRIBUTE_ID 0x0011    // Ver.: always
#define ZCL_DEVICE_ENABLED_ATTRIBUTE_ID 0x0012    // Ver.: always
#define ZCL_ALARM_MASK_ATTRIBUTE_ID 0x0013    // Ver.: always
#define ZCL_DISABLE_LOCAL_CONFIG_ATTRIBUTE_ID 0x0014    // Ver.: always


// Attribute types for cluster: Power Configuration
// Cluster specification level: 075123r03

// Server attributes
#define ZCL_MAINS_VOLTAGE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_MAINS_FREQUENCY_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_MAINS_ALARM_MASK_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_MAINS_VOLTAGE_MIN_THRESHOLD_ATTRIBUTE_ID 0x0011    // Ver.: always
#define ZCL_MAINS_VOLTAGE_MAX_THRESHOLD_ATTRIBUTE_ID 0x0012    // Ver.: always
#define ZCL_MAINS_VOLTAGE_DWELL_TRIP_POINT_ATTRIBUTE_ID 0x0013    // Ver.: always
#define ZCL_BATTERY_VOLTAGE_ATTRIBUTE_ID 0x0020    // Ver.: always
#define ZCL_BATTERY_MANUFACTURER_ATTRIBUTE_ID 0x0030    // Ver.: always
#define ZCL_BATTERY_SIZE_ATTRIBUTE_ID 0x0031    // Ver.: always
#define ZCL_BATTERY_AHR_RATING_ATTRIBUTE_ID 0x0032    // Ver.: always
#define ZCL_BATTERY_QUANTITY_ATTRIBUTE_ID 0x0033    // Ver.: always
#define ZCL_BATTERY_RATED_VOLTAGE_ATTRIBUTE_ID 0x0034    // Ver.: always
#define ZCL_BATTERY_ALARM_MASK_ATTRIBUTE_ID 0x0035    // Ver.: always
#define ZCL_BATTERY_VOLTAGE_MIN_THRESHOLD_ATTRIBUTE_ID 0x0036    // Ver.: always
```

```
// Attribute types for cluster: Device Temperature Configuration
// Cluster specification level: 075123r03

// Server attributes
#define ZCL_CURRENT_TEMPERATURE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_MIN_TEMP_EXPERIENCED_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_MAX_TEMP_EXPERIENCED_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_OVER_TEMP_TOTAL_DWELL_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_DEVICE_TEMP_ALARM_MASK_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_LOW_TEMP_THRESHOLD_ATTRIBUTE_ID 0x0011    // Ver.: always
#define ZCL_HIGH_TEMP_THRESHOLD_ATTRIBUTE_ID 0x0012    // Ver.: always
#define ZCL_LOW_TEMP_DWELL_TRIP_POINT_ATTRIBUTE_ID 0x0013    // Ver.: always
#define ZCL_HIGH_TEMP_DWELL_TRIP_POINT_ATTRIBUTE_ID 0x0014    // Ver.: always


// Attribute types for cluster: Identify
// Cluster specification level: 075123r03

// Server attributes
#define ZCL_IDENTIFY_TIME_ATTRIBUTE_ID 0x0000    // Ver.: always


// Attribute types for cluster: Groups
// Cluster specification level: 075123r03

// Server attributes
#define ZCL_GROUP_NAME_SUPPORT_ATTRIBUTE_ID 0x0000    // Ver.: always


// Attribute types for cluster: Scenes
// Cluster specification level: 075123r03

// Server attributes
#define ZCL_SCENE_COUNT_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_CURRENT_SCENE_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_CURRENT_GROUP_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_SCENE_VALID_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_SCENE_NAME_SUPPORT_ATTRIBUTE_ID 0x0004    // Ver.: always
#define ZCL_LAST_CONFIGURED_BY_ATTRIBUTE_ID 0x0005    // Ver.: always


// Attribute types for cluster: On/off
// Cluster specification level: 075123r03

// Server attributes
#define ZCL_ON_OFF_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_SAMPLE_MFG_SPECIFIC_TRANSITION_TIME_ATTRIBUTE_ID 0x0000    // Ver.: always


// Attribute types for cluster: On/off Switch Configuration
// Cluster specification level: 075123r03

// Server attributes
#define ZCL_SWITCH_TYPE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_SWITCH_ACTIONS_ATTRIBUTE_ID 0x0010    // Ver.: always

```

```
// Attribute types for cluster: Level Control
// Cluster specification level: 075123r03

// Server attributes
#define ZCL_CURRENT_LEVEL_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_LEVEL_CONTROL_REMAINING_TIME_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_ON_OFF_TRANSITION_TIME_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_ON_LEVEL_ATTRIBUTE_ID 0x0011    // Ver.: always


// Attribute types for cluster: Alarms
// Cluster specification level: 075123r03

// Server attributes
#define ZCL_ALARM_COUNT_ATTRIBUTE_ID 0x0000    // Ver.: always


// Attribute types for cluster: Time
// Cluster specification level: 075123r03

// Server attributes
#define ZCL_TIME_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_TIME_STATUS_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_TIME_ZONE_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_DST_START_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_DST_END_ATTRIBUTE_ID 0x0004    // Ver.: always
#define ZCL_DST_SHIFT_ATTRIBUTE_ID 0x0005    // Ver.: always
#define ZCL_STANDARD_TIME_ATTRIBUTE_ID 0x0006    // Ver.: always
#define ZCL_LOCAL_TIME_ATTRIBUTE_ID 0x0007    // Ver.: always
#define ZCL_LAST_SET_TIME_ATTRIBUTE_ID 0x0008    // Ver.: always
#define ZCL_VALID_UNTIL_TIME_ATTRIBUTE_ID 0x0009    // Ver.: always


// Attribute types for cluster: RSSI Location
// Cluster specification level: 075123r03

// Server attributes
#define ZCL_LOCATION_TYPE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_LOCATION_METHOD_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_LOCATION_AGE_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_QUALITY_MEASURE_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_NUMBER_OF_DEVICES_ATTRIBUTE_ID 0x0004    // Ver.: always
#define ZCL_COORDINATE1_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_COORDINATE2_ATTRIBUTE_ID 0x0011    // Ver.: always
#define ZCL_COORDINATE3_ATTRIBUTE_ID 0x0012    // Ver.: always
#define ZCL_POWER_ATTRIBUTE_ID 0x0013    // Ver.: always
#define ZCL_PATH_LOSS_EXPONENT_ATTRIBUTE_ID 0x0014    // Ver.: always
#define ZCL_REPORTING_PERIOD_ATTRIBUTE_ID 0x0015    // Ver.: always
#define ZCL_CALCULATION_PERIOD_ATTRIBUTE_ID 0x0016    // Ver.: always
#define ZCL_NUMBER_RSSI_MEASUREMENTS_ATTRIBUTE_ID 0x0017    // Ver.: always


// Attribute types for cluster: Binary Input (Basic)
// Cluster specification level: 075123r03

// Server attributes
```

```
#define ZCL_ACTIVE_TEXT_ATTRIBUTE_ID 0x0004    // Ver.: always
#define ZCL_DESCRIPTION_ATTRIBUTE_ID 0x001C    // Ver.: always
#define ZCL_INACTIVE_TEXT_ATTRIBUTE_ID 0x002E    // Ver.: always
#define ZCL_OUT_OF_SERVICE_ATTRIBUTE_ID 0x0051    // Ver.: always
#define ZCL_POLARITY_ATTRIBUTE_ID 0x0054    // Ver.: always
#define ZCL_PRESENT_VALUE_ATTRIBUTE_ID 0x0055    // Ver.: always
#define ZCL_RELIABILITY_ATTRIBUTE_ID 0x0067    // Ver.: always
#define ZCL_STATUS_FLAGS_ATTRIBUTE_ID 0x006F    // Ver.: always
#define ZCL_APPLICATION_TYPE_ATTRIBUTE_ID 0x0100    // Ver.: always


// Attribute types for cluster: Commissioning
// Cluster specification level: 075123r03

// Server attributes
#define ZCL_SHORT_ADDRESS_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_EXTENDED_PAN_ID_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_PAN_ID_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_CHANNEL_MASK_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_PROTOCOL_VERSION_ATTRIBUTE_ID 0x0004    // Ver.: always
#define ZCL_STACK_PROFILE_ATTRIBUTE_ID 0x0005    // Ver.: always
#define ZCL_STARTUP_CONTROL_ATTRIBUTE_ID 0x0006    // Ver.: always
#define ZCL_TRUST_CENTER_ADDRESS_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_TRUST_CENTER_MASTER_KEY_ATTRIBUTE_ID 0x0011    // Ver.: always
#define ZCL_NETWORK_KEY_ATTRIBUTE_ID 0x0012    // Ver.: always
#define ZCL_USE_INSECURE_JOIN_ATTRIBUTE_ID 0x0013    // Ver.: always
#define ZCL_PRECONFIGURED_LINK_KEY_ATTRIBUTE_ID 0x0014    // Ver.: always
#define ZCL_NETWORK_KEY_SEQUENCE_NUMBER_ATTRIBUTE_ID 0x0015    // Ver.: always
#define ZCL_NETWORK_KEY_TYPE_ATTRIBUTE_ID 0x0016    // Ver.: always
#define ZCL_NETWORK_MANAGER_ADDRESS_ATTRIBUTE_ID 0x0017    // Ver.: always
#define ZCL_SCAN_ATTEMPTS_ATTRIBUTE_ID 0x0020    // Ver.: always
#define ZCL_TIME_BETWEEN_SCANS_ATTRIBUTE_ID 0x0021    // Ver.: always
#define ZCL_REJOIN_INTERVAL_ATTRIBUTE_ID 0x0022    // Ver.: always
#define ZCL_MAX_REJOIN_INTERVAL_ATTRIBUTE_ID 0x0023    // Ver.: always
#define ZCL_INDERECT_POLL_RATE_ATTRIBUTE_ID 0x0030    // Ver.: always
#define ZCL_PARENT_RETRY_THRESHOLD_ATTRIBUTE_ID 0x0031    // Ver.: always
#define ZCL_CONCENTRATOR_FLAG_ATTRIBUTE_ID 0x0040    // Ver.: always
#define ZCL_CONCENTRATOR_RADIUS_ATTRIBUTE_ID 0x0041    // Ver.: always
#define ZCL_CONCENTRATOR_DISCOVERY_TIME_ATTRIBUTE_ID 0x0042    // Ver.: always


// Attribute types for cluster: Shade Configuration
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_SHADE_CONFIG_PHYSICAL_CLOSED_LIMIT_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_SHADE_CONFIG_MOTOR_STEP_SIZE_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_SHADE_CONFIG_STATUS_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_SHADE_CONFIG_CLOSED_LIMIT_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_SHADE_CONFIG_MODE_ATTRIBUTE_ID 0x0011    // Ver.: always


// Attribute types for cluster: Pump Configuration and Control
// Cluster specification level: 1.1, 053520r27

```

```
// Server attributes
#define ZCL_MAX_PRESSURE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_MAX_SPEED_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_MAX_FLOW_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_MIN_CONST_PRESSURE_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_MAX_CONST_PRESSURE_ATTRIBUTE_ID 0x0004    // Ver.: always
#define ZCL_MIN_COMP_PRESSURE_ATTRIBUTE_ID 0x0005    // Ver.: always
#define ZCL_MAX_COMP_PRESSURE_ATTRIBUTE_ID 0x0006    // Ver.: always
#define ZCL_MIN_CONST_SPEED_ATTRIBUTE_ID 0x0007    // Ver.: always
#define ZCL_MAX_CONST_SPEED_ATTRIBUTE_ID 0x0008    // Ver.: always
#define ZCL_MIN_CONST_FLOW_ATTRIBUTE_ID 0x0009    // Ver.: always
#define ZCL_MAX_CONST_FLOW_ATTRIBUTE_ID 0x000A    // Ver.: always
#define ZCL_MIN_CONST_TEMP_ATTRIBUTE_ID 0x000B    // Ver.: always
#define ZCL_MAX_CONST_TEMP_ATTRIBUTE_ID 0x000C    // Ver.: always
#define ZCL_PUMP_STATUS_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_EFFECTIVE_OPERATION_MODE_ATTRIBUTE_ID 0x0011    // Ver.: always
#define ZCL_EFFECTIVE_CONTROL_MODE_ATTRIBUTE_ID 0x0012    // Ver.: always
#define ZCL_CAPACITY_ATTRIBUTE_ID 0x0013    // Ver.: always
#define ZCL_SPEED_ATTRIBUTE_ID 0x0014    // Ver.: always
#define ZCL_LIFETIME_RUNNING_HOURS_ATTRIBUTE_ID 0x0015    // Ver.: always
#define ZCL_PUMP_POWER_ATTRIBUTE_ID 0x0016    // Ver.: always
#define ZCL_LIFETIME_ENERGY_CONSUMED_ATTRIBUTE_ID 0x0017    // Ver.: always
#define ZCL_OPERATION_MODE_ATTRIBUTE_ID 0x0020    // Ver.: always
#define ZCL_CONTROL_MODE_ATTRIBUTE_ID 0x0021    // Ver.: always
#define ZCL_PUMP_ALARM_MASK_ATTRIBUTE_ID 0x0022    // Ver.: always


// Attribute types for cluster: Thermostat
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_LOCAL_TEMPERATURE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_OUTDOOR_TEMPERATURE_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_THERMOSTAT_OCCUPANCY_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_ABS_MIN_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_ABS_MAX_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID 0x0004    // Ver.: always
#define ZCL_ABS_MIN_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID 0x0005    // Ver.: always
#define ZCL_ABS_MAX_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID 0x0006    // Ver.: always
#define ZCL_PI_COOLING_DEMAND_ATTRIBUTE_ID 0x0007    // Ver.: always
#define ZCL_PI_HEATING_DEMAND_ATTRIBUTE_ID 0x0008    // Ver.: always
#define ZCL_LOCAL_TEMPERATURE_CALIBRATION_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_OCCUPIED_COOLING_SETPOINT_ATTRIBUTE_ID 0x0011    // Ver.: always
#define ZCL_OCCUPIED_HEATING_SETPOINT_ATTRIBUTE_ID 0x0012    // Ver.: always
#define ZCL_UNOCCUPIED_COOLING_SETPOINT_ATTRIBUTE_ID 0x0013    // Ver.: always
#define ZCL_UNOCCUPIED_HEATING_SETPOINT_ATTRIBUTE_ID 0x0014    // Ver.: always
#define ZCL_MIN_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID 0x0015    // Ver.: always
#define ZCL_MAX_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID 0x0016    // Ver.: always
#define ZCL_MIN_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID 0x0017    // Ver.: always
#define ZCL_MAX_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID 0x0018    // Ver.: always
#define ZCL_MIN_SETPOINT_DEAD_BAND_ATTRIBUTE_ID 0x0019    // Ver.: always
#define ZCL_REMOTE_SENSING_ATTRIBUTE_ID 0x001A    // Ver.: always
#define ZCL_CONTROL_SEQUENCE_OF_OPERATION_ATTRIBUTE_ID 0x001B    // Ver.: always
#define ZCL_SYSTEM_MODE_ATTRIBUTE_ID 0x001C    // Ver.: always
#define ZCL_THERMOSTAT_ALARM_MASK_ATTRIBUTE_ID 0x001D    // Ver.: always
```

```
// Attribute types for cluster: Fan Control
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_FAN_CONTROL_FAN_MODE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_FAN_CONTROL_FAN_MODE_SEQUENCE_ATTRIBUTE_ID 0x0001    // Ver.: always


// Attribute types for cluster: Dehumidification Control
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_RELATIVE_HUMIDITY_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_DEHUMIDIFICATION_COOLING_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_RH_DEHUMIDIFICATION_SETPOINT_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_RELATIVE_HUMIDITY_MODE_ATTRIBUTE_ID 0x0011    // Ver.: always
#define ZCL_DEHUMIDIFICATION_LOCKOUT_ATTRIBUTE_ID 0x0012    // Ver.: always
#define ZCL_DEHUMIDIFICATION_HYSTERESIS_ATTRIBUTE_ID 0x0013    // Ver.: always
#define ZCL_DEHUMIDIFICATION_MAX_COOL_ATTRIBUTE_ID 0x0014    // Ver.: always
#define ZCL_RELATIVE_HUMIDITY_DISPLAY_ATTRIBUTE_ID 0x0015    // Ver.: always


// Attribute types for cluster: Thermostat User Interface Configuration
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_TEMPERATURE_DISPLAY_MODE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_KEYPAD_LOCKOUT_ATTRIBUTE_ID 0x0001    // Ver.: always


// Attribute types for cluster: Color Control
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_COLOR_CONTROL_CURRENT_HUE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_COLOR_CONTROL_CURRENT_SATURATION_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_COLOR_CONTROL_REMAINING_TIME_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_COLOR_CONTROL_CURRENT_X_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_COLOR_CONTROL_CURRENT_Y_ATTRIBUTE_ID 0x0004    // Ver.: always
#define ZCL_COLOR_CONTROL_DRIFT_COMPENSATION_ATTRIBUTE_ID 0x0005    // Ver.: always
#define ZCL_COLOR_CONTROL_COMPENSATION_TEXT_ATTRIBUTE_ID 0x0006    // Ver.: always
#define ZCL_COLOR_CONTROL_COLOR_TEMPERATURE_ATTRIBUTE_ID 0x0007    // Ver.: always
#define ZCL_COLOR_CONTROL_COLOR_MODE_ATTRIBUTE_ID 0x0008    // Ver.: always
#define ZCL_COLOR_CONTROL_NUMBER_OF_PRIMARIES_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_1_X_ATTRIBUTE_ID 0x0011    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_1_Y_ATTRIBUTE_ID 0x0012    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_1_INTENSITY_ATTRIBUTE_ID 0x0013    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_2_X_ATTRIBUTE_ID 0x0015    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_2_Y_ATTRIBUTE_ID 0x0016    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_2_INTENSITY_ATTRIBUTE_ID 0x0017    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_3_X_ATTRIBUTE_ID 0x0019    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_3_Y_ATTRIBUTE_ID 0x001A    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_3_INTENSITY_ATTRIBUTE_ID 0x001B    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_4_X_ATTRIBUTE_ID 0x0020    // Ver.: always
```

```
#define ZCL_COLOR_CONTROL_PRIMARY_4_Y_ATTRIBUTE_ID 0x0021    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_4_INTENSITY_ATTRIBUTE_ID 0x0022    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_5_X_ATTRIBUTE_ID 0x0024    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_5_Y_ATTRIBUTE_ID 0x0025    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_5_INTENSITY_ATTRIBUTE_ID 0x0026    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_6_X_ATTRIBUTE_ID 0x0028    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_6_Y_ATTRIBUTE_ID 0x0029    // Ver.: always
#define ZCL_COLOR_CONTROL_PRIMARY_6_INTENSITY_ATTRIBUTE_ID 0x002A    // Ver.: always
#define ZCL_COLOR_CONTROL_WHITE_POINT_X_ATTRIBUTE_ID 0x0030    // Ver.: always
#define ZCL_COLOR_CONTROL_WHITE_POINT_Y_ATTRIBUTE_ID 0x0031    // Ver.: always
#define ZCL_COLOR_CONTROL_COLOR_POINT_R_X_ATTRIBUTE_ID 0x0032    // Ver.: always
#define ZCL_COLOR_CONTROL_COLOR_POINT_R_Y_ATTRIBUTE_ID 0x0033    // Ver.: always
#define ZCL_COLOR_CONTROL_COLOR_POINT_R_INTENSITY_ATTRIBUTE_ID 0x0034    // Ver.: always
#define ZCL_COLOR_CONTROL_COLOR_POINT_G_X_ATTRIBUTE_ID 0x0036    // Ver.: always
#define ZCL_COLOR_CONTROL_COLOR_POINT_G_Y_ATTRIBUTE_ID 0x0037    // Ver.: always
#define ZCL_COLOR_CONTROL_COLOR_POINT_G_INTENSITY_ATTRIBUTE_ID 0x0038    // Ver.: always
#define ZCL_COLOR_CONTROL_COLOR_POINT_B_X_ATTRIBUTE_ID 0x003A    // Ver.: always
#define ZCL_COLOR_CONTROL_COLOR_POINT_B_Y_ATTRIBUTE_ID 0x003B    // Ver.: always
#define ZCL_COLOR_CONTROL_COLOR_POINT_B_INTENSITY_ATTRIBUTE_ID 0x003C    // Ver.: always


// Attribute types for cluster: Ballast Configuration
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_PHYSICAL_MIN_LEVEL_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_PHYSICAL_MAX_LEVEL_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_BALLAST_STATUS_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_MIN_LEVEL_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_MAX_LEVEL_ATTRIBUTE_ID 0x0011    // Ver.: always
#define ZCL_POWER_ON_LEVEL_ATTRIBUTE_ID 0x0012    // Ver.: always
#define ZCL_POWER_ON_FADE_TIME_ATTRIBUTE_ID 0x0013    // Ver.: always
#define ZCL_INTRINSIC_BALLAST_FACTOR_ATTRIBUTE_ID 0x0014    // Ver.: always
#define ZCL_BALLAST_FACTOR_ADJUSTMENT_ATTRIBUTE_ID 0x0015    // Ver.: always
#define ZCL_LAMP_QUALITY_ATTRIBUTE_ID 0x0020    // Ver.: always
#define ZCL_LAMP_TYPE_ATTRIBUTE_ID 0x0030    // Ver.: always
#define ZCL_LAMP_MANUFACTURER_ATTRIBUTE_ID 0x0031    // Ver.: always
#define ZCL_LAMP_RATED_HOURS_ATTRIBUTE_ID 0x0032    // Ver.: always
#define ZCL_LAMP_BURN_HOURS_ATTRIBUTE_ID 0x0033    // Ver.: always
#define ZCL_LAMP_ALARM_MODE_ATTRIBUTE_ID 0x0034    // Ver.: always
#define ZCL_LAMP_BURN_HOURS_TRIP_POINT_ATTRIBUTE_ID 0x0035    // Ver.: always


// Attribute types for cluster: Illuminance Measurement
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_ILLUM_MEASURED_VALUE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_ILLUM_MIN_MEASURED_VALUE_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_ILLUM_MAX_MEASURED_VALUE_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_ILLUM_TOLERANCE_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_ATTRIBUTE_ID 0x0004    // Ver.: always


// Attribute types for cluster: Illuminance Level Sensing
```

```
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_LEVEL_STATUS_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_SENSING_LIGHT_SENSOR_TYPE_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_ILLUMINANCE_TARGET_LEVEL_ATTRIBUTE_ID 0x0010    // Ver.: always


// Attribute types for cluster: Temperature Measurement
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_TEMP_MEASURED_VALUE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_TEMP_MIN_MEASURED_VALUE_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_TEMP_MAX_MEASURED_VALUE_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_TEMP_TOLERANCE_ATTRIBUTE_ID 0x0003    // Ver.: always


// Attribute types for cluster: Pressure Measurement
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_PRESSURE_MEASURED_VALUE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_PRESSURE_MIN_MEASURED_VALUE_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_PRESSURE_MAX_MEASURED_VALUE_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_PRESSURE_TOLERANCE_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_PRESSURE_SCALED_VALUE_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_PRESSURE_MIN_SCALED_VALUE_ATTRIBUTE_ID 0x0011    // Ver.: always
#define ZCL_PRESSURE_MAX_SCALED_VALUE_ATTRIBUTE_ID 0x0012    // Ver.: always
#define ZCL_PRESSURE_SCALED_TOLERANCE_ATTRIBUTE_ID 0x0013    // Ver.: always
#define ZCL_PRESSURE_SCALE_ATTRIBUTE_ID 0x0014    // Ver.: always


// Attribute types for cluster: Flow Measurement
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_FLOW_MEASURED_VALUE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_FLOW_MIN_MEASURED_VALUE_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_FLOW_MAX_MEASURED_VALUE_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_FLOW_TOLERANCE_ATTRIBUTE_ID 0x0003    // Ver.: always


// Attribute types for cluster: Relative Humidity Measurement
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_RELATIVE_HUMIDITY_MEASURED_VALUE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_RELATIVE_HUMIDITY_MIN_MEASURED_VALUE_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_RELATIVE_HUMIDITY_MAX_MEASURED_VALUE_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_RELATIVE_HUMIDITY_TOLERANCE_ATTRIBUTE_ID 0x0003    // Ver.: always


// Attribute types for cluster: Occupancy Sensing
// Cluster specification level: 1.1, 053520r27

// Server attributes
```

```
#define ZCL_OCCUPANCY_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_OCCUPANCY_SENSOR_TYPE_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_PIR_OCCUPIED_TO_UNOCCUPIED_DELAY_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_PIR_UNOCCUPIED_TO_OCCUPIED_DELAY_ATTRIBUTE_ID 0x0011    // Ver.: always
#define ZCL_PIR_UNOCCUPIED_TO_OCCUPIED_THRESHOLD_ATTRIBUTE_ID 0x0012    // Ver.: always
#define ZCL_ULTRASONIC_OCCUPIED_TO_UNOCCUPIED_DELAY_ATTRIBUTE_ID 0x0020    // Ver.: always
#define ZCL_ULTRASONIC_UNOCCUPIED_TO_OCCUPIED_DELAY_ATTRIBUTE_ID 0x0021    // Ver.: always
#define ZCL_ULTRASONIC_UNOCCUPIED_TO_OCCUPIED_THRESHOLD_ATTRIBUTE_ID 0x0022    //
Ver.: always


// Attribute types for cluster: IAS Zone
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_ZONE_STATE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_ZONE_TYPE_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_ZONE_STATUS_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_IAS_CIE_ADDRESS_ATTRIBUTE_ID 0x0010    // Ver.: always


// Attribute types for cluster: IAS WD
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_MAX_DURATION_ATTRIBUTE_ID 0x0000    // Ver.: always


// Attribute types for cluster: Door Lock
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_LOCK_STATE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_LOCK_TYPE_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_ACTUATOR_ENABLED_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_DOOR_STATE_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_DOOR_OPEN_EVENTS_ATTRIBUTE_ID 0x0004    // Ver.: always
#define ZCL_DOOR_CLOSED_EVENTS_ATTRIBUTE_ID 0x0005    // Ver.: always
#define ZCL_OPEN_PERIOD_ATTRIBUTE_ID 0x0006    // Ver.: always


// Attribute types for cluster: Window Covering
// Cluster specification level: 1.1, 053520r27

// Server attributes
#define ZCL_COVERING_TYPE_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_LIMIT_LIFT_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_LIMIT_TILT_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_CURRENT_LIFT_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_CURRENT_TILT_ATTRIBUTE_ID 0x0004    // Ver.: always
#define ZCL_NUMBER_LIFT_ATTRIBUTE_ID 0x0005    // Ver.: always
#define ZCL_NUMBER_TILT_ATTRIBUTE_ID 0x0006    // Ver.: always
#define ZCL_CONFIG_STATUS_ATTRIBUTE_ID 0x0007    // Ver.: always
#define ZCL_CURRENT_LIFT_PERCENTAGE_ATTRIBUTE_ID 0x0008    // Ver.: always
#define ZCL_CURRENT_TILT_PERCENTAGE_ATTRIBUTE_ID 0x0009    // Ver.: always
#define ZCL_OPEN_LIMIT_LIFT_ATTRIBUTE_ID 0x0010    // Ver.: always
```

```
#define ZCL_CLOSED_LIMIT_LIFT_ATTRIBUTE_ID 0x0011    // Ver.: always
#define ZCL_OPEN_LIMIT_TILT_ATTRIBUTE_ID 0x0012    // Ver.: always
#define ZCL_CLOSED_LIMIT_TILT_ATTRIBUTE_ID 0x0013    // Ver.: always
#define ZCL_VELOCITY_LIFT_ATTRIBUTE_ID 0x0014    // Ver.: always
#define ZCL_ACCELERATION_LIFT_ATTRIBUTE_ID 0x0015    // Ver.: always
#define ZCL_DECELERATION_LIFT_ATTRIBUTE_ID 0x0016    // Ver.: always
#define ZCL_MODE_ATTRIBUTE_ID 0x0017    // Ver.: always
#define ZCL_SETPOINTS_LIFT_ATTRIBUTE_ID 0x0018    // Ver.: always
#define ZCL_SETPOINTS_TILT_ATTRIBUTE_ID 0x0019    // Ver.: always


// Attribute types for cluster: Generic Tunnel
// Cluster specification level: unofficial

// Server attributes
#define ZCL_MAXIMUM_INCOMING_TRANSFER_SIZE_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_MAXIMUM_OUTGOING_TRANSFER_SIZE_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_PROTOCOL_ADDRESS_ATTRIBUTE_ID 0x0003    // Ver.: always


// Attribute types for cluster: Over the Air Bootloading Cluster
// Cluster specification level: 075123r03

// Client attributes
#define ZCL_UPGRADE_SERVER_ID_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_FILE_OFFSET_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_CURRENT_FILE_VERSION_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_CURRENT_ZIGBEE_STACK_VERSION_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_DOWNLOADED_FILE_VERSION_ATTRIBUTE_ID 0x0004    // Ver.: always
#define ZCL_DOWNLOADED_ZIGBEE_STACK_VERSION_ATTRIBUTE_ID 0x0005    // Ver.: always
#define ZCL_IMAGE_UPGRADE_STATUS_ATTRIBUTE_ID 0x0006    // Ver.: always
#define ZCL_MANUFACTURER_ID_ATTRIBUTE_ID 0x0007    // Ver.: always
#define ZCL_IMAGE_TYPE_ID_ATTRIBUTE_ID 0x0008    // Ver.: always


// Attribute types for cluster: Demand Response and Load Control
// Cluster specification level: 1.1, 105638r09

// Client attributes
#define ZCL_UTILITY_ENROLLMENT_GROUP_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_START_RANDOMIZE_MINUTES_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_STOP_RANDOMIZE_MINUTES_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_DEVICE_CLASS_VALUE_ATTRIBUTE_ID 0x0003    // Ver.: always


// Attribute types for cluster: Simple Metering
// Cluster specification level: 1.1, 105638r09

// Server attributes
#define ZCL_CURRENT_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_CURRENT_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_CURRENT_MAX_DEMAND_DELIVERED_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_CURRENT_MAX_DEMAND_RECEIVED_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_DFT_SUMMATION_ATTRIBUTE_ID 0x0004    // Ver.: always
#define ZCL_DAILY_FREEZE_TIME_ATTRIBUTE_ID 0x0005    // Ver.: always
#define ZCL_POWER_FACTOR_ATTRIBUTE_ID 0x0006    // Ver.: always
```

```
#define ZCL_READING_SNAP_SHOT_TIME_ATTRIBUTE_ID 0x0007    // Ver.: always
#define ZCL_CURRENT_MAX_DEMAND_DELIVERED_TIME_ATTRIBUTE_ID 0x0008    // Ver.: always
#define ZCL_CURRENT_MAX_DEMAND_RECEIVED_TIME_ATTRIBUTE_ID 0x0009    // Ver.: always
#define ZCL_DEFAULT_UPDATE_PERIOD_ATTRIBUTE_ID 0x000A    // Ver.: always
#define ZCL_FAST_POLL_UPDATE_PERIOD_ATTRIBUTE_ID 0x000B    // Ver.: always
#define ZCL_CURRENT_BLOCK_PERIOD_CONSUMPTION_DELIVERED_ATTRIBUTE_ID 0x000C    //
Ver.: always
#define ZCL_DAILY_CONSUMPTION_TARGET_ATTRIBUTE_ID 0x000D    // Ver.: always
#define ZCL_CURRENT_BLOCK_ATTRIBUTE_ID 0x000E    // Ver.: always
#define ZCL_PROFILE_INTERVAL_PERIOD_ATTRIBUTE_ID 0x000F    // Ver.: always
#define ZCL_INTERVAL_READ_REPORTING_PERIOD_ATTRIBUTE_ID 0x0010    // Ver.: always
#define ZCL_PRESET_READING_TIME_ATTRIBUTE_ID 0x0011    // Ver.: always
#define ZCL_VOLUME_PER_REPORT_ATTRIBUTE_ID 0x0012    // Ver.: always
#define ZCL_FLOW_RESTRICTION_ATTRIBUTE_ID 0x0013    // Ver.: always
#define ZCL_SUPPLY_STATUS_ATTRIBUTE_ID 0x0014    // Ver.: always
#define ZCL_CURRENT_INLET_ENERGY_CARRIER_SUMMATION_ATTRIBUTE_ID 0x0015    // Ver.: always
#define ZCL_CURRENT_OUTLET_ENERGY_CARRIER_SUMMATION_ATTRIBUTE_ID 0x0016    // Ver.: always
#define ZCL_INLET_TEMPERATURE_ATTRIBUTE_ID 0x0017    // Ver.: always
#define ZCL_OUTLET_TEMPERATURE_ATTRIBUTE_ID 0x0018    // Ver.: always
#define ZCL_CONTROL_TEMPERATURE_ATTRIBUTE_ID 0x0019    // Ver.: always
#define ZCL_CURRENT_INLET_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID 0x001A    // Ver.: always
#define ZCL_CURRENT_OUTLET_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID 0x001B    // Ver.: always
#define ZCL_CURRENT_TIER1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0100    // Ver.: always
#define ZCL_CURRENT_TIER1_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x0101    // Ver.: always
#define ZCL_CURRENT_TIER2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0102    // Ver.: always
#define ZCL_CURRENT_TIER2_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x0103    // Ver.: always
#define ZCL_CURRENT_TIER3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0104    // Ver.: always
#define ZCL_CURRENT_TIER3_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x0105    // Ver.: always
#define ZCL_CURRENT_TIER4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0106    // Ver.: always
#define ZCL_CURRENT_TIER4_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x0107    // Ver.: always
#define ZCL_CURRENT_TIER5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0108    // Ver.: always
#define ZCL_CURRENT_TIER5_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x0109    // Ver.: always
#define ZCL_CURRENT_TIER6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x010A    // Ver.: always
#define ZCL_CURRENT_TIER6_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x010B    // Ver.: always
#define ZCL_CURRENT_TIER7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x010C    // Ver.: always
#define ZCL_CURRENT_TIER7_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x010D    // Ver.: always
#define ZCL_CURRENT_TIER8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x010E    // Ver.: always
#define ZCL_CURRENT_TIER8_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x010F    // Ver.: always
#define ZCL_CURRENT_TIER9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0110    // Ver.: always
#define ZCL_CURRENT_TIER9_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x0111    // Ver.: always
#define ZCL_CURRENT_TIER10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0112    // Ver.: always
#define ZCL_CURRENT_TIER10_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x0113    // Ver.: always
#define ZCL_CURRENT_TIER11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0114    // Ver.: always
#define ZCL_CURRENT_TIER11_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x0115    // Ver.: always
#define ZCL_CURRENT_TIER12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0116    // Ver.: always
#define ZCL_CURRENT_TIER12_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x0117    // Ver.: always
#define ZCL_CURRENT_TIER13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0118    // Ver.: always
#define ZCL_CURRENT_TIER13_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x0119    // Ver.: always
#define ZCL_CURRENT_TIER14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x011A    // Ver.: always
#define ZCL_CURRENT_TIER14_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x011B    // Ver.: always
#define ZCL_CURRENT_TIER15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x011C    // Ver.: always
#define ZCL_CURRENT_TIER15_SUMMATION_RECEIVED_ATTRIBUTE_ID 0x011D    // Ver.: always
#define ZCL_STATUS_ATTRIBUTE_ID 0x0200    // Ver.: always
#define ZCL_REMAINING_BATTERY_LIFE_ATTRIBUTE_ID 0x0201    // Ver.: always
```

```
#define ZCL_HOURS_IN_OPERATION_ATTRIBUTE_ID 0x0202    // Ver.: always
#define ZCL_HOURS_IN_FAULT_ATTRIBUTE_ID 0x0203    // Ver.: always
#define ZCL_UNIT_OF_MEASURE_ATTRIBUTE_ID 0x0300    // Ver.: always
#define ZCL_MULTIPLIER_ATTRIBUTE_ID 0x0301    // Ver.: always
#define ZCL_DIVISOR_ATTRIBUTE_ID 0x0302    // Ver.: always
#define ZCL_SUMMATION_FORMATTING_ATTRIBUTE_ID 0x0303    // Ver.: always
#define ZCL_DEMAND_FORMATTING_ATTRIBUTE_ID 0x0304    // Ver.: always
#define ZCL_HISTORICAL_CONSUMPTION_FORMATTING_ATTRIBUTE_ID 0x0305    // Ver.: always
#define ZCL_METERING_DEVICE_TYPE_ATTRIBUTE_ID 0x0306    // Ver.: always
#define ZCL_SITE_ID_ATTRIBUTE_ID 0x0307    // Ver.: always
#define ZCL_METER_SERIAL_NUMBER_ATTRIBUTE_ID 0x0308    // Ver.: always
#define ZCL_ENERGY_CARRIER_UNIT_OF_MEASURE_ATTRIBUTE_ID 0x0309    // Ver.: always
#define ZCL_ENERGY_CARRIER_SUMMATION_FORMATTING_ATTRIBUTE_ID 0x030A    // Ver.: always
#define ZCL_ENERGY_CARRIER_DEMAND_FORMATTING_ATTRIBUTE_ID 0x030B    // Ver.: always
#define ZCL_TEMPERATURE_UNIT_OF_MEASURE_ATTRIBUTE_ID 0x030C    // Ver.: always
#define ZCL_TEMPERATURE_FORMATTING_ATTRIBUTE_ID 0x030D    // Ver.: always
#define ZCL_INSTANTANEOUS_DEMAND_ATTRIBUTE_ID 0x0400    // Ver.: always
#define ZCL_CURRENT_DAY_CONSUMPTION_DELIVERED_ATTRIBUTE_ID 0x0401    // Ver.: always
#define ZCL_CURRENT_DAY_CONSUMPTION_RECEIVED_ATTRIBUTE_ID 0x0402    // Ver.: always
#define ZCL_PREVIOUS_DAY_CONSUMPTION_DELIVERED_ATTRIBUTE_ID 0x0403    // Ver.: always
#define ZCL_PREVIOUS_DAY_CONSUMPTION_RECEIVED_ATTRIBUTE_ID 0x0404    // Ver.: always
#define
ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_START_TIME_DELIVERED_ATTRIBUTE_ID 0x0405    // Ver.: always
#define ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_START_TIME_RECEIVED_ATTRIBUTE_ID
0x0406    // Ver.: always
#define ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_VALUE_DELIVERED_ATTRIBUTE_ID
0x0407    // Ver.: always
#define ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_VALUE_RECEIVED_ATTRIBUTE_ID 0x0408    //
Ver.: always
#define ZCL_CURRENT_DAY_MAX_PRESSURE_ATTRIBUTE_ID 0x0409    // Ver.: always
#define ZCL_CURRENT_DAY_MIN_PRESSURE_ATTRIBUTE_ID 0x040A    // Ver.: always
#define ZCL_PREVIOUS_DAY_MAX_PRESSURE_ATTRIBUTE_ID 0x040B    // Ver.: always
#define ZCL_PREVIOUS_DAY_MIN_PRESSURE_ATTRIBUTE_ID 0x040C    // Ver.: always
#define ZCL_CURRENT_DAY_MAX_DEMAND_ATTRIBUTE_ID 0x040D    // Ver.: always
#define ZCL_PREVIOUS_DAY_MAX_DEMAND_ATTRIBUTE_ID 0x040E    // Ver.: always
#define ZCL_CURRENT_MONTH_MAX_DEMAND_ATTRIBUTE_ID 0x040F    // Ver.: always
#define ZCL_CURRENT_YEAR_MAX_DEMAND_ATTRIBUTE_ID 0x0410    // Ver.: always
#define ZCL_CURRENT_DAY_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID 0x0411    // Ver.: always
#define ZCL_PREVIOUS_DAY_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID 0x0412    // Ver.: always
#define ZCL_CURRENT_MONTH_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID 0x0413    // Ver.: always
#define ZCL_CURRENT_MONTH_MIN_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID 0x0414    // Ver.: always
#define ZCL_CURRENT_YEAR_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID 0x0415    // Ver.: always
#define ZCL_CURRENT_YEAR_MIN_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID 0x0416    // Ver.: always
#define ZCL_MAX_NUMBER_OF_PERIODS_DELIVERED_ATTRIBUTE_ID 0x0500    // Ver.: always
#define ZCL_CURRENT_DEMAND_DELIVERED_ATTRIBUTE_ID 0x0600    // Ver.: always
#define ZCL_DEMAND_LIMIT_ATTRIBUTE_ID 0x0601    // Ver.: always
#define ZCL_DEMAND_INTEGRATION_PERIOD_ATTRIBUTE_ID 0x0602    // Ver.: always
#define ZCL_NUMBER_OF_DEMAND_SUBINTERVALS_ATTRIBUTE_ID 0x0603    // Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0700    //
Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0701    //
Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0702    //
Ver.: always
```

```
#define ZCL_CURRENT_NO_TIER_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0703    // Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0704    // Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0705    // Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0706    // Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0707    // Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0708    // Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0709    // Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x070A    // Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x070B    // Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x070C    // Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x070D    // Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x070E    // Ver.: always
#define ZCL_CURRENT_NO_TIER_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x070F    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0710    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0711    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0712    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0713    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0714    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0715    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0716    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0717    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0718    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0719    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x071A    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x071B    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x071C    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x071D    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x071E    // Ver.: always
#define ZCL_CURRENT_TIER1_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x071F    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0720    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0721    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0722    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0723    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0724    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0725    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0726    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0727    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0728    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0729    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x072A    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x072B    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x072C    // Ver.: always
```

```
#define ZCL_CURRENT_TIER2_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x072D    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x072E    // Ver.: always
#define ZCL_CURRENT_TIER2_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x072F    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0730    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0731    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0732    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0733    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0734    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0735    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0736    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0737    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0738    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0739    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x073A    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x073B    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x073C    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x073D    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x073E    // Ver.: always
#define ZCL_CURRENT_TIER3_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x073F    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0740    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0741    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0742    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0743    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0744    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0745    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0746    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0747    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0748    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0749    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x074A    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x074B    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x074C    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x074D    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x074E    // Ver.: always
#define ZCL_CURRENT_TIER4_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x074F    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0750    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0751    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0752    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0753    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0754    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0755    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0756    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0757    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0758    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0759    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x075A    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x075B    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x075C    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x075D    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x075E    // Ver.: always
#define ZCL_CURRENT_TIER5_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x075F    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0760    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0761    // Ver.: always
```

```
#define ZCL_CURRENT_TIER6_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0762    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0763    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0764    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0765    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0766    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0767    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0768    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0769    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x076A    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x076B    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x076C    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x076D    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x076E    // Ver.: always
#define ZCL_CURRENT_TIER6_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x076F    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0770    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0771    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0772    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0773    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0774    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0775    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0776    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0777    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0778    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0779    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x077A    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x077B    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x077C    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x077D    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x077E    // Ver.: always
#define ZCL_CURRENT_TIER7_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x077F    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0780    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0781    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0782    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0783    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0784    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0785    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0786    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0787    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0788    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0789    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x078A    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x078B    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x078C    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x078D    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x078E    // Ver.: always
#define ZCL_CURRENT_TIER8_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x078F    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0790    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0791    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0792    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0793    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0794    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0795    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0796    // Ver.: always
```

```
#define ZCL_CURRENT_TIER9_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0797    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0798    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x0799    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x079A    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x079B    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x079C    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x079D    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x079E    // Ver.: always
#define ZCL_CURRENT_TIER9_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x079F    // Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07A0    // Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07A1    // Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07A2    // Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07A3    // Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07A4    // Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07A5    // Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07A6    // Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07A7    // Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07A8    // Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07A9    // Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07AA    //
Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07AB    // Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07AC    // Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07AD    //
Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07AE    // Ver.: always
#define ZCL_CURRENT_TIER10_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07AF    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07B0    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07B1    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07B2    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07B3    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07B4    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07B5    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07B6    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07B7    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07B8    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07B9    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07BA    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07BB    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07BC    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07BD    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07BE    // Ver.: always
#define ZCL_CURRENT_TIER11_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07BF    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07C0    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07C1    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07C2    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07C3    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07C4    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07C5    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07C6    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07C7    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07C8    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07C9    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07CA    // Ver.: always
```

```
#define ZCL_CURRENT_TIER12_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07CB    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07CC    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07CD    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07CE    // Ver.: always
#define ZCL_CURRENT_TIER12_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07CF    // Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07D0    // Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07D1    // Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07D2    // Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07D3    // Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07D4    // Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07D5    // Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07D6    // Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07D7    // Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07D8    // Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07D9    // Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07DA    //
Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07DB    // Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07DC    // Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07DD    //
Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07DE    // Ver.: always
#define ZCL_CURRENT_TIER13_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07DF    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07E0    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07E1    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07E2    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07E3    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07E4    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07E5    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07E6    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07E7    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07E8    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07E9    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07EA    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07EB    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07EC    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07ED    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07EE    // Ver.: always
#define ZCL_CURRENT_TIER14_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07EF    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07F0    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07F1    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07F2    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07F3    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07F4    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07F5    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07F6    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07F7    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07F8    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07F9    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07FA    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07FB    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07FC    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07FD    // Ver.: always
#define ZCL_CURRENT_TIER15_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07FE    // Ver.: always
```

```
#define ZCL_CURRENT_TIER15_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID 0x07FF    // Ver.: always
#define ZCL_GENERIC_ALARM_MASK_ATTRIBUTE_ID 0x0800    // Ver.: always
#define ZCL_ELECTRICITY_ALARM_MASK_ATTRIBUTE_ID 0x0801    // Ver.: always
#define ZCL_GENERIC_FLOW_PRESSURE_ALARM_MASK_ATTRIBUTE_ID 0x0802    // Ver.: always
#define ZCL_WATER_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID 0x0803    // Ver.: always
#define ZCL_HEAT_AND_COOLING_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID 0x0804    // Ver.: always
#define ZCL_GAS_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID 0x0805    // Ver.: always
#define ZCL_MIRROR_IEEE_ADDRESS_ATTRIBUTE_ID 0xF000    // Ver.: always
#define ZCL_MIRROR_SYNC_PERIOD_ATTRIBUTE_ID 0xF001    // Ver.: always
#define ZCL_MIRROR_LAST_SYNC_TIME_ATTRIBUTE_ID 0xF002    // Ver.: always


// Attribute types for cluster: Price
// Cluster specification level: 1.1, 105638r09

// Client attributes
#define ZCL_PRICE_INCREASE_RANDOMIZE_MINUTES_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_PRICE_DECREASE_RANDOMIZE_MINUTES_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_COMMODITY_TYPE_CLIENT_ATTRIBUTE_ID 0x0002    // Ver.: always

// Server attributes
#define ZCL_TIER1_PRICE_LABEL_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_TIER2_PRICE_LABEL_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_TIER3_PRICE_LABEL_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_TIER4_PRICE_LABEL_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_TIER5_PRICE_LABEL_ATTRIBUTE_ID 0x0004    // Ver.: always
#define ZCL_TIER6_PRICE_LABEL_ATTRIBUTE_ID 0x0005    // Ver.: always
#define ZCL_TIER7_PRICE_LABEL_ATTRIBUTE_ID 0x0006    // Ver.: always
#define ZCL_TIER8_PRICE_LABEL_ATTRIBUTE_ID 0x0007    // Ver.: always
#define ZCL_TIER9_PRICE_LABEL_ATTRIBUTE_ID 0x0008    // Ver.: always
#define ZCL_TIER10_PRICE_LABEL_ATTRIBUTE_ID 0x0009    // Ver.: always
#define ZCL_TIER11_PRICE_LABEL_ATTRIBUTE_ID 0x000A    // Ver.: always
#define ZCL_TIER12_PRICE_LABEL_ATTRIBUTE_ID 0x000B    // Ver.: always
#define ZCL_TIER13_PRICE_LABEL_ATTRIBUTE_ID 0x000C    // Ver.: always
#define ZCL_TIER14_PRICE_LABEL_ATTRIBUTE_ID 0x000D    // Ver.: always
#define ZCL_TIER15_PRICE_LABEL_ATTRIBUTE_ID 0x000E    // Ver.: always
#define ZCL_BLOCK1_THRESHOLD_ATTRIBUTE_ID 0x0100    // Ver.: always
#define ZCL_BLOCK2_THRESHOLD_ATTRIBUTE_ID 0x0101    // Ver.: always
#define ZCL_BLOCK3_THRESHOLD_ATTRIBUTE_ID 0x0102    // Ver.: always
#define ZCL_BLOCK4_THRESHOLD_ATTRIBUTE_ID 0x0103    // Ver.: always
#define ZCL_BLOCK5_THRESHOLD_ATTRIBUTE_ID 0x0104    // Ver.: always
#define ZCL_BLOCK6_THRESHOLD_ATTRIBUTE_ID 0x0105    // Ver.: always
#define ZCL_BLOCK7_THRESHOLD_ATTRIBUTE_ID 0x0106    // Ver.: always
#define ZCL_BLOCK8_THRESHOLD_ATTRIBUTE_ID 0x0107    // Ver.: always
#define ZCL_BLOCK9_THRESHOLD_ATTRIBUTE_ID 0x0108    // Ver.: always
#define ZCL_BLOCK10_THRESHOLD_ATTRIBUTE_ID 0x0109    // Ver.: always
#define ZCL_BLOCK11_THRESHOLD_ATTRIBUTE_ID 0x010A    // Ver.: always
#define ZCL_BLOCK12_THRESHOLD_ATTRIBUTE_ID 0x010B    // Ver.: always
#define ZCL_BLOCK13_THRESHOLD_ATTRIBUTE_ID 0x010C    // Ver.: always
#define ZCL_BLOCK14_THRESHOLD_ATTRIBUTE_ID 0x010D    // Ver.: always
#define ZCL_BLOCK15_THRESHOLD_ATTRIBUTE_ID 0x010E    // Ver.: always
#define ZCL_START_OF_BLOCK_PERIOD_ATTRIBUTE_ID 0x0200    // Ver.: always
#define ZCL_BLOCK_PERIOD_DURATION_MINUTES_ATTRIBUTE_ID 0x0201    // Ver.: always
#define ZCL_THRESHOLD_MULTIPLIER_ATTRIBUTE_ID 0x0202    // Ver.: always
```

```
#define ZCL_THRESHOLD_DIVISOR_ATTRIBUTE_ID 0x0203    // Ver.: always
#define ZCL_COMMODITY_TYPE_SERVER_ATTRIBUTE_ID 0x0300    // Ver.: always
#define ZCL_STANDING_CHARGE_ATTRIBUTE_ID 0x0301    // Ver.: always
#define ZCL_NO_TIER_BLOCK1_PRICE_ATTRIBUTE_ID 0x0400    // Ver.: always
#define ZCL_NO_TIER_BLOCK2_PRICE_ATTRIBUTE_ID 0x0401    // Ver.: always
#define ZCL_NO_TIER_BLOCK3_PRICE_ATTRIBUTE_ID 0x0402    // Ver.: always
#define ZCL_NO_TIER_BLOCK4_PRICE_ATTRIBUTE_ID 0x0403    // Ver.: always
#define ZCL_NO_TIER_BLOCK5_PRICE_ATTRIBUTE_ID 0x0404    // Ver.: always
#define ZCL_NO_TIER_BLOCK6_PRICE_ATTRIBUTE_ID 0x0405    // Ver.: always
#define ZCL_NO_TIER_BLOCK7_PRICE_ATTRIBUTE_ID 0x0406    // Ver.: always
#define ZCL_NO_TIER_BLOCK8_PRICE_ATTRIBUTE_ID 0x0407    // Ver.: always
#define ZCL_NO_TIER_BLOCK9_PRICE_ATTRIBUTE_ID 0x0408    // Ver.: always
#define ZCL_NO_TIER_BLOCK10_PRICE_ATTRIBUTE_ID 0x0409    // Ver.: always
#define ZCL_NO_TIER_BLOCK11_PRICE_ATTRIBUTE_ID 0x040A    // Ver.: always
#define ZCL_NO_TIER_BLOCK12_PRICE_ATTRIBUTE_ID 0x040B    // Ver.: always
#define ZCL_NO_TIER_BLOCK13_PRICE_ATTRIBUTE_ID 0x040C    // Ver.: always
#define ZCL_NO_TIER_BLOCK14_PRICE_ATTRIBUTE_ID 0x040D    // Ver.: always
#define ZCL_NO_TIER_BLOCK15_PRICE_ATTRIBUTE_ID 0x040E    // Ver.: always
#define ZCL_NO_TIER_BLOCK16_PRICE_ATTRIBUTE_ID 0x040F    // Ver.: always
#define ZCL_TIER1_BLOCK1_PRICE_ATTRIBUTE_ID 0x0410    // Ver.: always
#define ZCL_TIER1_BLOCK2_PRICE_ATTRIBUTE_ID 0x0411    // Ver.: always
#define ZCL_TIER1_BLOCK3_PRICE_ATTRIBUTE_ID 0x0412    // Ver.: always
#define ZCL_TIER1_BLOCK4_PRICE_ATTRIBUTE_ID 0x0413    // Ver.: always
#define ZCL_TIER1_BLOCK5_PRICE_ATTRIBUTE_ID 0x0414    // Ver.: always
#define ZCL_TIER1_BLOCK6_PRICE_ATTRIBUTE_ID 0x0415    // Ver.: always
#define ZCL_TIER1_BLOCK7_PRICE_ATTRIBUTE_ID 0x0416    // Ver.: always
#define ZCL_TIER1_BLOCK8_PRICE_ATTRIBUTE_ID 0x0417    // Ver.: always
#define ZCL_TIER1_BLOCK9_PRICE_ATTRIBUTE_ID 0x0418    // Ver.: always
#define ZCL_TIER1_BLOCK10_PRICE_ATTRIBUTE_ID 0x0419    // Ver.: always
#define ZCL_TIER1_BLOCK11_PRICE_ATTRIBUTE_ID 0x041A    // Ver.: always
#define ZCL_TIER1_BLOCK12_PRICE_ATTRIBUTE_ID 0x041B    // Ver.: always
#define ZCL_TIER1_BLOCK13_PRICE_ATTRIBUTE_ID 0x041C    // Ver.: always
#define ZCL_TIER1_BLOCK14_PRICE_ATTRIBUTE_ID 0x041D    // Ver.: always
#define ZCL_TIER1_BLOCK15_PRICE_ATTRIBUTE_ID 0x041E    // Ver.: always
#define ZCL_TIER1_BLOCK16_PRICE_ATTRIBUTE_ID 0x041F    // Ver.: always
#define ZCL_TIER2_BLOCK1_PRICE_ATTRIBUTE_ID 0x0420    // Ver.: always
#define ZCL_TIER2_BLOCK2_PRICE_ATTRIBUTE_ID 0x0421    // Ver.: always
#define ZCL_TIER2_BLOCK3_PRICE_ATTRIBUTE_ID 0x0422    // Ver.: always
#define ZCL_TIER2_BLOCK4_PRICE_ATTRIBUTE_ID 0x0423    // Ver.: always
#define ZCL_TIER2_BLOCK5_PRICE_ATTRIBUTE_ID 0x0424    // Ver.: always
#define ZCL_TIER2_BLOCK6_PRICE_ATTRIBUTE_ID 0x0425    // Ver.: always
#define ZCL_TIER2_BLOCK7_PRICE_ATTRIBUTE_ID 0x0426    // Ver.: always
#define ZCL_TIER2_BLOCK8_PRICE_ATTRIBUTE_ID 0x0427    // Ver.: always
#define ZCL_TIER2_BLOCK9_PRICE_ATTRIBUTE_ID 0x0428    // Ver.: always
#define ZCL_TIER2_BLOCK10_PRICE_ATTRIBUTE_ID 0x0429    // Ver.: always
#define ZCL_TIER2_BLOCK11_PRICE_ATTRIBUTE_ID 0x042A    // Ver.: always
#define ZCL_TIER2_BLOCK12_PRICE_ATTRIBUTE_ID 0x042B    // Ver.: always
#define ZCL_TIER2_BLOCK13_PRICE_ATTRIBUTE_ID 0x042C    // Ver.: always
#define ZCL_TIER2_BLOCK14_PRICE_ATTRIBUTE_ID 0x042D    // Ver.: always
#define ZCL_TIER2_BLOCK15_PRICE_ATTRIBUTE_ID 0x042E    // Ver.: always
#define ZCL_TIER2_BLOCK16_PRICE_ATTRIBUTE_ID 0x042F    // Ver.: always
#define ZCL_TIER3_BLOCK1_PRICE_ATTRIBUTE_ID 0x0430    // Ver.: always
#define ZCL_TIER3_BLOCK2_PRICE_ATTRIBUTE_ID 0x0431    // Ver.: always
```

```
#define ZCL_TIER3_BLOCK3_PRICE_ATTRIBUTE_ID 0x0432    // Ver.: always
#define ZCL_TIER3_BLOCK4_PRICE_ATTRIBUTE_ID 0x0433    // Ver.: always
#define ZCL_TIER3_BLOCK5_PRICE_ATTRIBUTE_ID 0x0434    // Ver.: always
#define ZCL_TIER3_BLOCK6_PRICE_ATTRIBUTE_ID 0x0435    // Ver.: always
#define ZCL_TIER3_BLOCK7_PRICE_ATTRIBUTE_ID 0x0436    // Ver.: always
#define ZCL_TIER3_BLOCK8_PRICE_ATTRIBUTE_ID 0x0437    // Ver.: always
#define ZCL_TIER3_BLOCK9_PRICE_ATTRIBUTE_ID 0x0438    // Ver.: always
#define ZCL_TIER3_BLOCK10_PRICE_ATTRIBUTE_ID 0x0439    // Ver.: always
#define ZCL_TIER3_BLOCK11_PRICE_ATTRIBUTE_ID 0x043A    // Ver.: always
#define ZCL_TIER3_BLOCK12_PRICE_ATTRIBUTE_ID 0x043B    // Ver.: always
#define ZCL_TIER3_BLOCK13_PRICE_ATTRIBUTE_ID 0x043C    // Ver.: always
#define ZCL_TIER3_BLOCK14_PRICE_ATTRIBUTE_ID 0x043D    // Ver.: always
#define ZCL_TIER3_BLOCK15_PRICE_ATTRIBUTE_ID 0x043E    // Ver.: always
#define ZCL_TIER3_BLOCK16_PRICE_ATTRIBUTE_ID 0x043F    // Ver.: always
#define ZCL_TIER4_BLOCK1_PRICE_ATTRIBUTE_ID 0x0440    // Ver.: always
#define ZCL_TIER4_BLOCK2_PRICE_ATTRIBUTE_ID 0x0441    // Ver.: always
#define ZCL_TIER4_BLOCK3_PRICE_ATTRIBUTE_ID 0x0442    // Ver.: always
#define ZCL_TIER4_BLOCK4_PRICE_ATTRIBUTE_ID 0x0443    // Ver.: always
#define ZCL_TIER4_BLOCK5_PRICE_ATTRIBUTE_ID 0x0444    // Ver.: always
#define ZCL_TIER4_BLOCK6_PRICE_ATTRIBUTE_ID 0x0445    // Ver.: always
#define ZCL_TIER4_BLOCK7_PRICE_ATTRIBUTE_ID 0x0446    // Ver.: always
#define ZCL_TIER4_BLOCK8_PRICE_ATTRIBUTE_ID 0x0447    // Ver.: always
#define ZCL_TIER4_BLOCK9_PRICE_ATTRIBUTE_ID 0x0448    // Ver.: always
#define ZCL_TIER4_BLOCK10_PRICE_ATTRIBUTE_ID 0x0449    // Ver.: always
#define ZCL_TIER4_BLOCK11_PRICE_ATTRIBUTE_ID 0x044A    // Ver.: always
#define ZCL_TIER4_BLOCK12_PRICE_ATTRIBUTE_ID 0x044B    // Ver.: always
#define ZCL_TIER4_BLOCK13_PRICE_ATTRIBUTE_ID 0x044C    // Ver.: always
#define ZCL_TIER4_BLOCK14_PRICE_ATTRIBUTE_ID 0x044D    // Ver.: always
#define ZCL_TIER4_BLOCK15_PRICE_ATTRIBUTE_ID 0x044E    // Ver.: always
#define ZCL_TIER4_BLOCK16_PRICE_ATTRIBUTE_ID 0x044F    // Ver.: always
#define ZCL_TIER5_BLOCK1_PRICE_ATTRIBUTE_ID 0x0450    // Ver.: always
#define ZCL_TIER5_BLOCK2_PRICE_ATTRIBUTE_ID 0x0451    // Ver.: always
#define ZCL_TIER5_BLOCK3_PRICE_ATTRIBUTE_ID 0x0452    // Ver.: always
#define ZCL_TIER5_BLOCK4_PRICE_ATTRIBUTE_ID 0x0453    // Ver.: always
#define ZCL_TIER5_BLOCK5_PRICE_ATTRIBUTE_ID 0x0454    // Ver.: always
#define ZCL_TIER5_BLOCK6_PRICE_ATTRIBUTE_ID 0x0455    // Ver.: always
#define ZCL_TIER5_BLOCK7_PRICE_ATTRIBUTE_ID 0x0456    // Ver.: always
#define ZCL_TIER5_BLOCK8_PRICE_ATTRIBUTE_ID 0x0457    // Ver.: always
#define ZCL_TIER5_BLOCK9_PRICE_ATTRIBUTE_ID 0x0458    // Ver.: always
#define ZCL_TIER5_BLOCK10_PRICE_ATTRIBUTE_ID 0x0459    // Ver.: always
#define ZCL_TIER5_BLOCK11_PRICE_ATTRIBUTE_ID 0x045A    // Ver.: always
#define ZCL_TIER5_BLOCK12_PRICE_ATTRIBUTE_ID 0x045B    // Ver.: always
#define ZCL_TIER5_BLOCK13_PRICE_ATTRIBUTE_ID 0x045C    // Ver.: always
#define ZCL_TIER5_BLOCK14_PRICE_ATTRIBUTE_ID 0x045D    // Ver.: always
#define ZCL_TIER5_BLOCK15_PRICE_ATTRIBUTE_ID 0x045E    // Ver.: always
#define ZCL_TIER5_BLOCK16_PRICE_ATTRIBUTE_ID 0x045F    // Ver.: always
#define ZCL_TIER6_BLOCK1_PRICE_ATTRIBUTE_ID 0x0460    // Ver.: always
#define ZCL_TIER6_BLOCK2_PRICE_ATTRIBUTE_ID 0x0461    // Ver.: always
#define ZCL_TIER6_BLOCK3_PRICE_ATTRIBUTE_ID 0x0462    // Ver.: always
#define ZCL_TIER6_BLOCK4_PRICE_ATTRIBUTE_ID 0x0463    // Ver.: always
#define ZCL_TIER6_BLOCK5_PRICE_ATTRIBUTE_ID 0x0464    // Ver.: always
#define ZCL_TIER6_BLOCK6_PRICE_ATTRIBUTE_ID 0x0465    // Ver.: always
#define ZCL_TIER6_BLOCK7_PRICE_ATTRIBUTE_ID 0x0466    // Ver.: always
```

```
#define ZCL_TIER6_BLOCK8_PRICE_ATTRIBUTE_ID 0x0467    // Ver.: always
#define ZCL_TIER6_BLOCK9_PRICE_ATTRIBUTE_ID 0x0468    // Ver.: always
#define ZCL_TIER6_BLOCK10_PRICE_ATTRIBUTE_ID 0x0469    // Ver.: always
#define ZCL_TIER6_BLOCK11_PRICE_ATTRIBUTE_ID 0x046A    // Ver.: always
#define ZCL_TIER6_BLOCK12_PRICE_ATTRIBUTE_ID 0x046B    // Ver.: always
#define ZCL_TIER6_BLOCK13_PRICE_ATTRIBUTE_ID 0x046C    // Ver.: always
#define ZCL_TIER6_BLOCK14_PRICE_ATTRIBUTE_ID 0x046D    // Ver.: always
#define ZCL_TIER6_BLOCK15_PRICE_ATTRIBUTE_ID 0x046E    // Ver.: always
#define ZCL_TIER6_BLOCK16_PRICE_ATTRIBUTE_ID 0x046F    // Ver.: always
#define ZCL_TIER7_BLOCK1_PRICE_ATTRIBUTE_ID 0x0470    // Ver.: always
#define ZCL_TIER7_BLOCK2_PRICE_ATTRIBUTE_ID 0x0471    // Ver.: always
#define ZCL_TIER7_BLOCK3_PRICE_ATTRIBUTE_ID 0x0472    // Ver.: always
#define ZCL_TIER7_BLOCK4_PRICE_ATTRIBUTE_ID 0x0473    // Ver.: always
#define ZCL_TIER7_BLOCK5_PRICE_ATTRIBUTE_ID 0x0474    // Ver.: always
#define ZCL_TIER7_BLOCK6_PRICE_ATTRIBUTE_ID 0x0475    // Ver.: always
#define ZCL_TIER7_BLOCK7_PRICE_ATTRIBUTE_ID 0x0476    // Ver.: always
#define ZCL_TIER7_BLOCK8_PRICE_ATTRIBUTE_ID 0x0477    // Ver.: always
#define ZCL_TIER7_BLOCK9_PRICE_ATTRIBUTE_ID 0x0478    // Ver.: always
#define ZCL_TIER7_BLOCK10_PRICE_ATTRIBUTE_ID 0x0479    // Ver.: always
#define ZCL_TIER7_BLOCK11_PRICE_ATTRIBUTE_ID 0x047A    // Ver.: always
#define ZCL_TIER7_BLOCK12_PRICE_ATTRIBUTE_ID 0x047B    // Ver.: always
#define ZCL_TIER7_BLOCK13_PRICE_ATTRIBUTE_ID 0x047C    // Ver.: always
#define ZCL_TIER7_BLOCK14_PRICE_ATTRIBUTE_ID 0x047D    // Ver.: always
#define ZCL_TIER7_BLOCK15_PRICE_ATTRIBUTE_ID 0x047E    // Ver.: always
#define ZCL_TIER7_BLOCK16_PRICE_ATTRIBUTE_ID 0x047F    // Ver.: always
#define ZCL_TIER8_BLOCK1_PRICE_ATTRIBUTE_ID 0x0480    // Ver.: always
#define ZCL_TIER8_BLOCK2_PRICE_ATTRIBUTE_ID 0x0481    // Ver.: always
#define ZCL_TIER8_BLOCK3_PRICE_ATTRIBUTE_ID 0x0482    // Ver.: always
#define ZCL_TIER8_BLOCK4_PRICE_ATTRIBUTE_ID 0x0483    // Ver.: always
#define ZCL_TIER8_BLOCK5_PRICE_ATTRIBUTE_ID 0x0484    // Ver.: always
#define ZCL_TIER8_BLOCK6_PRICE_ATTRIBUTE_ID 0x0485    // Ver.: always
#define ZCL_TIER8_BLOCK7_PRICE_ATTRIBUTE_ID 0x0486    // Ver.: always
#define ZCL_TIER8_BLOCK8_PRICE_ATTRIBUTE_ID 0x0487    // Ver.: always
#define ZCL_TIER8_BLOCK9_PRICE_ATTRIBUTE_ID 0x0488    // Ver.: always
#define ZCL_TIER8_BLOCK10_PRICE_ATTRIBUTE_ID 0x0489    // Ver.: always
#define ZCL_TIER8_BLOCK11_PRICE_ATTRIBUTE_ID 0x048A    // Ver.: always
#define ZCL_TIER8_BLOCK12_PRICE_ATTRIBUTE_ID 0x048B    // Ver.: always
#define ZCL_TIER8_BLOCK13_PRICE_ATTRIBUTE_ID 0x048C    // Ver.: always
#define ZCL_TIER8_BLOCK14_PRICE_ATTRIBUTE_ID 0x048D    // Ver.: always
#define ZCL_TIER8_BLOCK15_PRICE_ATTRIBUTE_ID 0x048E    // Ver.: always
#define ZCL_TIER8_BLOCK16_PRICE_ATTRIBUTE_ID 0x048F    // Ver.: always
#define ZCL_TIER9_BLOCK1_PRICE_ATTRIBUTE_ID 0x0490    // Ver.: always
#define ZCL_TIER9_BLOCK2_PRICE_ATTRIBUTE_ID 0x0491    // Ver.: always
#define ZCL_TIER9_BLOCK3_PRICE_ATTRIBUTE_ID 0x0492    // Ver.: always
#define ZCL_TIER9_BLOCK4_PRICE_ATTRIBUTE_ID 0x0493    // Ver.: always
#define ZCL_TIER9_BLOCK5_PRICE_ATTRIBUTE_ID 0x0494    // Ver.: always
#define ZCL_TIER9_BLOCK6_PRICE_ATTRIBUTE_ID 0x0495    // Ver.: always
#define ZCL_TIER9_BLOCK7_PRICE_ATTRIBUTE_ID 0x0496    // Ver.: always
#define ZCL_TIER9_BLOCK8_PRICE_ATTRIBUTE_ID 0x0497    // Ver.: always
#define ZCL_TIER9_BLOCK9_PRICE_ATTRIBUTE_ID 0x0498    // Ver.: always
#define ZCL_TIER9_BLOCK10_PRICE_ATTRIBUTE_ID 0x0499    // Ver.: always
#define ZCL_TIER9_BLOCK11_PRICE_ATTRIBUTE_ID 0x049A    // Ver.: always
#define ZCL_TIER9_BLOCK12_PRICE_ATTRIBUTE_ID 0x049B    // Ver.: always
```

```
#define ZCL_TIER9_BLOCK13_PRICE_ATTRIBUTE_ID 0x049C    // Ver.: always
#define ZCL_TIER9_BLOCK14_PRICE_ATTRIBUTE_ID 0x049D    // Ver.: always
#define ZCL_TIER9_BLOCK15_PRICE_ATTRIBUTE_ID 0x049E    // Ver.: always
#define ZCL_TIER9_BLOCK16_PRICE_ATTRIBUTE_ID 0x049F    // Ver.: always
#define ZCL_TIER10_BLOCK1_PRICE_ATTRIBUTE_ID 0x04A0    // Ver.: always
#define ZCL_TIER10_BLOCK2_PRICE_ATTRIBUTE_ID 0x04A1    // Ver.: always
#define ZCL_TIER10_BLOCK3_PRICE_ATTRIBUTE_ID 0x04A2    // Ver.: always
#define ZCL_TIER10_BLOCK4_PRICE_ATTRIBUTE_ID 0x04A3    // Ver.: always
#define ZCL_TIER10_BLOCK5_PRICE_ATTRIBUTE_ID 0x04A4    // Ver.: always
#define ZCL_TIER10_BLOCK6_PRICE_ATTRIBUTE_ID 0x04A5    // Ver.: always
#define ZCL_TIER10_BLOCK7_PRICE_ATTRIBUTE_ID 0x04A6    // Ver.: always
#define ZCL_TIER10_BLOCK8_PRICE_ATTRIBUTE_ID 0x04A7    // Ver.: always
#define ZCL_TIER10_BLOCK9_PRICE_ATTRIBUTE_ID 0x04A8    // Ver.: always
#define ZCL_TIER10_BLOCK10_PRICE_ATTRIBUTE_ID 0x04A9    // Ver.: always
#define ZCL_TIER10_BLOCK11_PRICE_ATTRIBUTE_ID 0x04AA    // Ver.: always
#define ZCL_TIER10_BLOCK12_PRICE_ATTRIBUTE_ID 0x04AB    // Ver.: always
#define ZCL_TIER10_BLOCK13_PRICE_ATTRIBUTE_ID 0x04AC    // Ver.: always
#define ZCL_TIER10_BLOCK14_PRICE_ATTRIBUTE_ID 0x04AD    // Ver.: always
#define ZCL_TIER10_BLOCK15_PRICE_ATTRIBUTE_ID 0x04AE    // Ver.: always
#define ZCL_TIER10_BLOCK16_PRICE_ATTRIBUTE_ID 0x04AF    // Ver.: always
#define ZCL_TIER11_BLOCK1_PRICE_ATTRIBUTE_ID 0x04B0    // Ver.: always
#define ZCL_TIER11_BLOCK2_PRICE_ATTRIBUTE_ID 0x04B1    // Ver.: always
#define ZCL_TIER11_BLOCK3_PRICE_ATTRIBUTE_ID 0x04B2    // Ver.: always
#define ZCL_TIER11_BLOCK4_PRICE_ATTRIBUTE_ID 0x04B3    // Ver.: always
#define ZCL_TIER11_BLOCK5_PRICE_ATTRIBUTE_ID 0x04B4    // Ver.: always
#define ZCL_TIER11_BLOCK6_PRICE_ATTRIBUTE_ID 0x04B5    // Ver.: always
#define ZCL_TIER11_BLOCK7_PRICE_ATTRIBUTE_ID 0x04B6    // Ver.: always
#define ZCL_TIER11_BLOCK8_PRICE_ATTRIBUTE_ID 0x04B7    // Ver.: always
#define ZCL_TIER11_BLOCK9_PRICE_ATTRIBUTE_ID 0x04B8    // Ver.: always
#define ZCL_TIER11_BLOCK10_PRICE_ATTRIBUTE_ID 0x04B9    // Ver.: always
#define ZCL_TIER11_BLOCK11_PRICE_ATTRIBUTE_ID 0x04BA    // Ver.: always
#define ZCL_TIER11_BLOCK12_PRICE_ATTRIBUTE_ID 0x04BB    // Ver.: always
#define ZCL_TIER11_BLOCK13_PRICE_ATTRIBUTE_ID 0x04BC    // Ver.: always
#define ZCL_TIER11_BLOCK14_PRICE_ATTRIBUTE_ID 0x04BD    // Ver.: always
#define ZCL_TIER11_BLOCK15_PRICE_ATTRIBUTE_ID 0x04BE    // Ver.: always
#define ZCL_TIER11_BLOCK16_PRICE_ATTRIBUTE_ID 0x04BF    // Ver.: always
#define ZCL_TIER12_BLOCK1_PRICE_ATTRIBUTE_ID 0x04C0    // Ver.: always
#define ZCL_TIER12_BLOCK2_PRICE_ATTRIBUTE_ID 0x04C1    // Ver.: always
#define ZCL_TIER12_BLOCK3_PRICE_ATTRIBUTE_ID 0x04C2    // Ver.: always
#define ZCL_TIER12_BLOCK4_PRICE_ATTRIBUTE_ID 0x04C3    // Ver.: always
#define ZCL_TIER12_BLOCK5_PRICE_ATTRIBUTE_ID 0x04C4    // Ver.: always
#define ZCL_TIER12_BLOCK6_PRICE_ATTRIBUTE_ID 0x04C5    // Ver.: always
#define ZCL_TIER12_BLOCK7_PRICE_ATTRIBUTE_ID 0x04C6    // Ver.: always
#define ZCL_TIER12_BLOCK8_PRICE_ATTRIBUTE_ID 0x04C7    // Ver.: always
#define ZCL_TIER12_BLOCK9_PRICE_ATTRIBUTE_ID 0x04C8    // Ver.: always
#define ZCL_TIER12_BLOCK10_PRICE_ATTRIBUTE_ID 0x04C9    // Ver.: always
#define ZCL_TIER12_BLOCK11_PRICE_ATTRIBUTE_ID 0x04CA    // Ver.: always
#define ZCL_TIER12_BLOCK12_PRICE_ATTRIBUTE_ID 0x04CB    // Ver.: always
#define ZCL_TIER12_BLOCK13_PRICE_ATTRIBUTE_ID 0x04CC    // Ver.: always
#define ZCL_TIER12_BLOCK14_PRICE_ATTRIBUTE_ID 0x04CD    // Ver.: always
#define ZCL_TIER12_BLOCK15_PRICE_ATTRIBUTE_ID 0x04CE    // Ver.: always
#define ZCL_TIER12_BLOCK16_PRICE_ATTRIBUTE_ID 0x04CF    // Ver.: always
#define ZCL_TIER13_BLOCK1_PRICE_ATTRIBUTE_ID 0x04D0    // Ver.: always
```

```
#define ZCL_TIER13_BLOCK2_PRICE_ATTRIBUTE_ID 0x04D1    // Ver.: always
#define ZCL_TIER13_BLOCK3_PRICE_ATTRIBUTE_ID 0x04D2    // Ver.: always
#define ZCL_TIER13_BLOCK4_PRICE_ATTRIBUTE_ID 0x04D3    // Ver.: always
#define ZCL_TIER13_BLOCK5_PRICE_ATTRIBUTE_ID 0x04D4    // Ver.: always
#define ZCL_TIER13_BLOCK6_PRICE_ATTRIBUTE_ID 0x04D5    // Ver.: always
#define ZCL_TIER13_BLOCK7_PRICE_ATTRIBUTE_ID 0x04D6    // Ver.: always
#define ZCL_TIER13_BLOCK8_PRICE_ATTRIBUTE_ID 0x04D7    // Ver.: always
#define ZCL_TIER13_BLOCK9_PRICE_ATTRIBUTE_ID 0x04D8    // Ver.: always
#define ZCL_TIER13_BLOCK10_PRICE_ATTRIBUTE_ID 0x04D9    // Ver.: always
#define ZCL_TIER13_BLOCK11_PRICE_ATTRIBUTE_ID 0x04DA    // Ver.: always
#define ZCL_TIER13_BLOCK12_PRICE_ATTRIBUTE_ID 0x04DB    // Ver.: always
#define ZCL_TIER13_BLOCK13_PRICE_ATTRIBUTE_ID 0x04DC    // Ver.: always
#define ZCL_TIER13_BLOCK14_PRICE_ATTRIBUTE_ID 0x04DD    // Ver.: always
#define ZCL_TIER13_BLOCK15_PRICE_ATTRIBUTE_ID 0x04DE    // Ver.: always
#define ZCL_TIER13_BLOCK16_PRICE_ATTRIBUTE_ID 0x04DF    // Ver.: always
#define ZCL_TIER14_BLOCK1_PRICE_ATTRIBUTE_ID 0x04E0    // Ver.: always
#define ZCL_TIER14_BLOCK2_PRICE_ATTRIBUTE_ID 0x04E1    // Ver.: always
#define ZCL_TIER14_BLOCK3_PRICE_ATTRIBUTE_ID 0x04E2    // Ver.: always
#define ZCL_TIER14_BLOCK4_PRICE_ATTRIBUTE_ID 0x04E3    // Ver.: always
#define ZCL_TIER14_BLOCK5_PRICE_ATTRIBUTE_ID 0x04E4    // Ver.: always
#define ZCL_TIER14_BLOCK6_PRICE_ATTRIBUTE_ID 0x04E5    // Ver.: always
#define ZCL_TIER14_BLOCK7_PRICE_ATTRIBUTE_ID 0x04E6    // Ver.: always
#define ZCL_TIER14_BLOCK8_PRICE_ATTRIBUTE_ID 0x04E7    // Ver.: always
#define ZCL_TIER14_BLOCK9_PRICE_ATTRIBUTE_ID 0x04E8    // Ver.: always
#define ZCL_TIER14_BLOCK10_PRICE_ATTRIBUTE_ID 0x04E9    // Ver.: always
#define ZCL_TIER14_BLOCK11_PRICE_ATTRIBUTE_ID 0x04EA    // Ver.: always
#define ZCL_TIER14_BLOCK12_PRICE_ATTRIBUTE_ID 0x04EB    // Ver.: always
#define ZCL_TIER14_BLOCK13_PRICE_ATTRIBUTE_ID 0x04EC    // Ver.: always
#define ZCL_TIER14_BLOCK14_PRICE_ATTRIBUTE_ID 0x04ED    // Ver.: always
#define ZCL_TIER14_BLOCK15_PRICE_ATTRIBUTE_ID 0x04EE    // Ver.: always
#define ZCL_TIER14_BLOCK16_PRICE_ATTRIBUTE_ID 0x04EF    // Ver.: always
#define ZCL_TIER15_BLOCK1_PRICE_ATTRIBUTE_ID 0x04F0    // Ver.: always
#define ZCL_TIER15_BLOCK2_PRICE_ATTRIBUTE_ID 0x04F1    // Ver.: always
#define ZCL_TIER15_BLOCK3_PRICE_ATTRIBUTE_ID 0x04F2    // Ver.: always
#define ZCL_TIER15_BLOCK4_PRICE_ATTRIBUTE_ID 0x04F3    // Ver.: always
#define ZCL_TIER15_BLOCK5_PRICE_ATTRIBUTE_ID 0x04F4    // Ver.: always
#define ZCL_TIER15_BLOCK6_PRICE_ATTRIBUTE_ID 0x04F5    // Ver.: always
#define ZCL_TIER15_BLOCK7_PRICE_ATTRIBUTE_ID 0x04F6    // Ver.: always
#define ZCL_TIER15_BLOCK8_PRICE_ATTRIBUTE_ID 0x04F7    // Ver.: always
#define ZCL_TIER15_BLOCK9_PRICE_ATTRIBUTE_ID 0x04F8    // Ver.: always
#define ZCL_TIER15_BLOCK10_PRICE_ATTRIBUTE_ID 0x04F9    // Ver.: always
#define ZCL_TIER15_BLOCK11_PRICE_ATTRIBUTE_ID 0x04FA    // Ver.: always
#define ZCL_TIER15_BLOCK12_PRICE_ATTRIBUTE_ID 0x04FB    // Ver.: always
#define ZCL_TIER15_BLOCK13_PRICE_ATTRIBUTE_ID 0x04FC    // Ver.: always
#define ZCL_TIER15_BLOCK14_PRICE_ATTRIBUTE_ID 0x04FD    // Ver.: always
#define ZCL_TIER15_BLOCK15_PRICE_ATTRIBUTE_ID 0x04FE    // Ver.: always
#define ZCL_TIER15_BLOCK16_PRICE_ATTRIBUTE_ID 0x04FF    // Ver.: always


// Attribute types for cluster: Tunneling
// Cluster specification level: 1.1, 105638r09

// Server attributes
#define ZCL_CLOSE_TUNNEL_TIMEOUT_ATTRIBUTE_ID 0x0000    // Ver.: always
```

```
// Attribute types for cluster: Prepayment
// Cluster specification level: 1.1, 105638r09

// Server attributes
#define ZCL_PAYMENT_CONTROL_ATTRIBUTE_ID 0x0000    // Ver.: always
#define ZCL_CREDIT_REMAINING_ATTRIBUTE_ID 0x0001    // Ver.: always
#define ZCL_EMERGENCY_CREDIT_REMAINING_ATTRIBUTE_ID 0x0002    // Ver.: always
#define ZCL_CREDIT_STATUS_ATTRIBUTE_ID 0x0003    // Ver.: always
#define ZCL_TOP_UP_DATE_TIME_1_ATTRIBUTE_ID 0x0100    // Ver.: always
#define ZCL_TOP_UP_AMOUNT_1_ATTRIBUTE_ID 0x0101    // Ver.: always
#define ZCL_ORIGINATING_DEVICE_1_ATTRIBUTE_ID 0x0102    // Ver.: always
#define ZCL_TOP_UP_DATE_TIME_2_ATTRIBUTE_ID 0x0110    // Ver.: always
#define ZCL_TOP_UP_AMOUNT_2_ATTRIBUTE_ID 0x0111    // Ver.: always
#define ZCL_ORIGINATING_DEVICE_2_ATTRIBUTE_ID 0x0112    // Ver.: always
#define ZCL_TOP_UP_DATE_TIME_3_ATTRIBUTE_ID 0x0120    // Ver.: always
#define ZCL_TOP_UP_AMOUNT_3_ATTRIBUTE_ID 0x0121    // Ver.: always
#define ZCL_ORIGINATING_DEVICE_3_ATTRIBUTE_ID 0x0122    // Ver.: always
#define ZCL_TOP_UP_DATE_TIME_4_ATTRIBUTE_ID 0x0130    // Ver.: always
#define ZCL_TOP_UP_AMOUNT_4_ATTRIBUTE_ID 0x0131    // Ver.: always
#define ZCL_ORIGINATING_DEVICE_4_ATTRIBUTE_ID 0x0132    // Ver.: always
#define ZCL_TOP_UP_DATE_TIME_5_ATTRIBUTE_ID 0x0140    // Ver.: always
#define ZCL_TOP_UP_AMOUNT_5_ATTRIBUTE_ID 0x0141    // Ver.: always
#define ZCL_ORIGINATING_DEVICE_5_ATTRIBUTE_ID 0x0142    // Ver.: always
#define ZCL_FUEL_DEBT_REMAINING_ATTRIBUTE_ID 0x0200    // Ver.: always
#define ZCL_FUEL_DEBT_RECOVERY_RATE_ATTRIBUTE_ID 0x0201    // Ver.: always
#define ZCL_FUEL_DEBT_RECOVERY_PERIOD_ATTRIBUTE_ID 0x0202    // Ver.: always
#define ZCL_NON_FUEL_DEBT_REMAINING_ATTRIBUTE_ID 0x0203    // Ver.: always
#define ZCL_NON_FUEL_DEBT_RECOVERY_RATE_ATTRIBUTE_ID 0x0204    // Ver.: always
#define ZCL_NON_FUEL_DEBT_RECOVERY_PERIOD_ATTRIBUTE_ID 0x0205    // Ver.: always
#define ZCL_PROPOSED_CHANGE_PROVIDER_ID_ATTRIBUTE_ID 0x0300    // Ver.: always
#define ZCL_PROPOSED_CHANGE_IMPLEMENTATION_TIME_ATTRIBUTE_ID 0x0301    // Ver.: always
#define ZCL_PROPOSED_CHANGE_SUPPLY_STATUS_ATTRIBUTE_ID 0x0302    // Ver.: always
#define ZCL_DELAYED_SUPPLY_INTERRUPT_VALUE_REMAINING_ATTRIBUTE_ID 0x0303    // Ver.: always
#define ZCL_DELAYED_SUPPLY_INTERRUPT_VALUE_TYPE_ATTRIBUTE_ID 0x0304    // Ver.: always


// Attribute types for cluster: Key establishment
// Cluster specification level: 1.1, 105638r09

// Client attributes
#define ZCL_KEY_ESTABLISHMENT_SUITE_CLIENT_ATTRIBUTE_ID 0x0000    // Ver.: always

// Server attributes
#define ZCL_KEY_ESTABLISHMENT_SUITE_SERVER_ATTRIBUTE_ID 0x0000    // Ver.: always


// Attribute types for cluster: Sample Mfg Specific Cluster
// Cluster specification level: UNKNOWN

// Server attributes
#define ZCL_ATTRIBUTE_ONE_ATTRIBUTE_ID 0x0000    // Ver.: always


#endif // __EMBER_AF_ATTRIBUTE_ID__
```

attribute-size.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generateAttributeSizes()
//

// Used ZCL attribute type sizes
  ZCL_INT8U_ATTRIBUTE_TYPE, 1,
  ZCL_CHAR_STRING_ATTRIBUTE_TYPE, 16,
  ZCL_ENUM8_ATTRIBUTE_TYPE, 1,
  ZCL_BOOLEAN_ATTRIBUTE_TYPE, 1,
  ZCL_BITMAP8_ATTRIBUTE_TYPE, 1,
  ZCL_INT16U_ATTRIBUTE_TYPE, 2,
  ZCL_INT16S_ATTRIBUTE_TYPE, 2,
  ZCL_INT24U_ATTRIBUTE_TYPE, 3,
  ZCL_IEEE_ADDRESS_ATTRIBUTE_TYPE, 8,
  ZCL_UTC_TIME_ATTRIBUTE_TYPE, 4,
  ZCL_INT32S_ATTRIBUTE_TYPE, 4,
  ZCL_INT32U_ATTRIBUTE_TYPE, 4,
  ZCL_DATA8_ATTRIBUTE_TYPE, 1,
  ZCL_BITMAP32_ATTRIBUTE_TYPE, 4,
  ZCL_SECURITY_KEY_ATTRIBUTE_TYPE, 16,
  ZCL_BITMAP16_ATTRIBUTE_TYPE, 2,
  ZCL_INT8S_ATTRIBUTE_TYPE, 1,
  ZCL_ENUM16_ATTRIBUTE_TYPE, 2,
  ZCL_OCTET_STRING_ATTRIBUTE_TYPE, 64,
  ZCL_INT48U_ATTRIBUTE_TYPE, 6,
  ZCL_INT24S_ATTRIBUTE_TYPE, 3,
```

attribute-storage.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generateAttributeStorageHeader()
//

// Enclosing macro to prevent multiple inclusion
#ifndef __ATTRIBUTE_STORAGE_GEN__
#define __ATTRIBUTE_STORAGE_GEN__


// Attribute masks modify how attributes are used by the framework
// Attribute that has this mask is NOT read-only
#define ATTRIBUTE_MASK_WRITABLE (0x01)
// Attribute that has this mask is saved to a token
#define ATTRIBUTE_MASK_TOKENIZE (0x02)
// Attribute that has this mask has a min/max values
#define ATTRIBUTE_MASK_MIN_MAX (0x04)
// Manufacturer specific attribute
#define ATTRIBUTE_MASK_MANUFACTURER_SPECIFIC (0x08)
// Attribute deferred to external storage
#define ATTRIBUTE_MASK_EXTERNAL_STORAGE (0x10)
// Attribute is singleton
#define ATTRIBUTE_MASK_SINGLETON (0x20)
// Attribute is a client attribute
#define ATTRIBUTE_MASK_CLIENT (0x40)


// Cluster masks modify how clusters are used by the framework
// Does this cluster have init function?
#define CLUSTER_MASK_INIT_FUNCTION (0x01)
// Does this cluster have attribute changed function?
#define CLUSTER_MASK_ATTRIBUTE_CHANGED_FUNCTION (0x02)
// Does this cluster have default response function?
#define CLUSTER_MASK_DEFAULT_RESPONSE_FUNCTION (0x04)
// Does this cluster have message sent function?
#define CLUSTER_MASK_MESSAGE_SENT_FUNCTION (0x08)
// Does this cluster have manufacturer specific attribute changed funciton?
#define CLUSTER_MASK_MANUFACTURER_SPECIFIC_ATTRIBUTE_CHANGED_FUNCTION (0x10)
// Cluster is a server
#define CLUSTER_MASK_SERVER (0x40)
// Cluster is a client
#define CLUSTER_MASK_CLIENT (0x80)

```

```
#endif // __ATTRIBUTE_STORAGE_GEN__
```

attribute-type.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generateAttributeTypes()
//

// Enclosing macro to prevent multiple inclusion
#ifndef __EMBER_AF_ATTRIBUTE_TYPES__
#define __EMBER_AF_ATTRIBUTE_TYPES__


// ZCL attribute types
enum {
  ZCL_NO_DATA_ATTRIBUTE_TYPE                = 0x00, // No data
  ZCL_DATA8_ATTRIBUTE_TYPE                  = 0x08, // 8-bit data
  ZCL_DATA16_ATTRIBUTE_TYPE                 = 0x09, // 16-bit data
  ZCL_DATA24_ATTRIBUTE_TYPE                 = 0x0A, // 24-bit data
  ZCL_DATA32_ATTRIBUTE_TYPE                 = 0x0B, // 32-bit data
  ZCL_DATA40_ATTRIBUTE_TYPE                 = 0x0C, // 40-bit data
  ZCL_DATA48_ATTRIBUTE_TYPE                 = 0x0D, // 48-bit data
  ZCL_DATA56_ATTRIBUTE_TYPE                 = 0x0E, // 56-bit data
  ZCL_DATA64_ATTRIBUTE_TYPE                 = 0x0F, // 64-bit data
  ZCL_BOOLEAN_ATTRIBUTE_TYPE                = 0x10, // Boolean
  ZCL_BITMAP8_ATTRIBUTE_TYPE                = 0x18, // 8-bit bitmap
  ZCL_BITMAP16_ATTRIBUTE_TYPE               = 0x19, // 16-bit bitmap
  ZCL_BITMAP24_ATTRIBUTE_TYPE               = 0x1A, // 24-bit bitmap
  ZCL_BITMAP32_ATTRIBUTE_TYPE               = 0x1B, // 32-bit bitmap
  ZCL_BITMAP40_ATTRIBUTE_TYPE               = 0x1C, // 40-bit bitmap
  ZCL_BITMAP48_ATTRIBUTE_TYPE               = 0x1D, // 48-bit bitmap
  ZCL_BITMAP56_ATTRIBUTE_TYPE               = 0x1E, // 56-bit bitmap
  ZCL_BITMAP64_ATTRIBUTE_TYPE               = 0x1F, // 64-bit bitmap
  ZCL_INT8U_ATTRIBUTE_TYPE                  = 0x20, // Unsigned 8-bit integer
  ZCL_INT16U_ATTRIBUTE_TYPE                 = 0x21, // Unsigned 16-bit integer
  ZCL_INT24U_ATTRIBUTE_TYPE                 = 0x22, // Unsigned 24-bit integer
  ZCL_INT32U_ATTRIBUTE_TYPE                 = 0x23, // Unsigned 32-bit integer
  ZCL_INT40U_ATTRIBUTE_TYPE                 = 0x24, // Unsigned 40-bit integer
  ZCL_INT48U_ATTRIBUTE_TYPE                 = 0x25, // Unsigned 48-bit integer
  ZCL_INT56U_ATTRIBUTE_TYPE                 = 0x26, // Unsigned 56-bit integer
  ZCL_INT64U_ATTRIBUTE_TYPE                 = 0x27, // Unsigned 64-bit integer
  ZCL_INT8S_ATTRIBUTE_TYPE                  = 0x28, // Signed 8-bit integer
```

```
ZCL_INT16S_ATTRIBUTE_TYPE                    = 0x29, // Signed 16-bit integer
ZCL_INT24S_ATTRIBUTE_TYPE                    = 0x2A, // Signed 24-bit integer
ZCL_INT32S_ATTRIBUTE_TYPE                    = 0x2B, // Signed 32-bit integer
ZCL_INT40S_ATTRIBUTE_TYPE                    = 0x2C, // Signed 40-bit integer
ZCL_INT48S_ATTRIBUTE_TYPE                    = 0x2D, // Signed 48-bit integer
ZCL_INT56S_ATTRIBUTE_TYPE                    = 0x2E, // Signed 56-bit integer
ZCL_INT64S_ATTRIBUTE_TYPE                    = 0x2F, // Signed 64-bit integer
ZCL_ENUM8_ATTRIBUTE_TYPE                     = 0x30, // 8-bit enumeration
ZCL_ENUM16_ATTRIBUTE_TYPE                    = 0x31, // 16-bit enumeration
ZCL_FLOAT_SEMI_ATTRIBUTE_TYPE                = 0x38, // Semi-precision
ZCL_FLOAT_SINGLE_ATTRIBUTE_TYPE              = 0x39, // Single precision
ZCL_FLOAT_DOUBLE_ATTRIBUTE_TYPE              = 0x3A, // Double precision
ZCL_OCTET_STRING_ATTRIBUTE_TYPE              = 0x41, // Octet string
ZCL_CHAR_STRING_ATTRIBUTE_TYPE               = 0x42, // Character string
ZCL_LONG_OCTET_STRING_ATTRIBUTE_TYPE         = 0x43, // Long octet string
ZCL_LONG_CHAR_STRING_ATTRIBUTE_TYPE          = 0x44, // Long character string
ZCL_ARRAY_ATTRIBUTE_TYPE                     = 0x48, // Array
ZCL_STRUCT_ATTRIBUTE_TYPE                    = 0x4C, // Structure
ZCL_SET_ATTRIBUTE_TYPE                       = 0x50, // Set
ZCL_BAG_ATTRIBUTE_TYPE                       = 0x51, // Bag
ZCL_TIME_OF_DAY_ATTRIBUTE_TYPE               = 0xE0, // Time of day
ZCL_DATE_ATTRIBUTE_TYPE                      = 0xE1, // Date
ZCL_UTC_TIME_ATTRIBUTE_TYPE                  = 0xE2, // UTC Time
ZCL_CLUSTER_ID_ATTRIBUTE_TYPE                = 0xE8, // Cluster ID
ZCL_ATTRIBUTE_ID_ATTRIBUTE_TYPE              = 0xE9, // Attribute ID
ZCL_BACNET_OID_ATTRIBUTE_TYPE                = 0xEA, // BACnet OID
ZCL_IEEE_ADDRESS_ATTRIBUTE_TYPE              = 0xF0, // IEEE address
ZCL_SECURITY_KEY_ATTRIBUTE_TYPE              = 0xF1, // 128-bit security key
ZCL_UNKNOWN_ATTRIBUTE_TYPE                   = 0xFF // Unknown

};
#endif // __EMBER_AF_ATTRIBUTE_TYPES__
```

bacnet-protocol-tunnel.h

Go to the documentation of this file.

```
// ********************************************************************
// * bacnet-protocol-tunnel.h
// *
// *
// * Copyright 2009 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************

// can be called to generate a "Transfer NPDU" command
void bacnetProtocolTxTransferNpdu(int16u npduLength,
                                  int8u* npdu,
                                  int16u destShortAddress,
                                  int8u destEndpoint,
                                  int8u sourceEndpoint);

#define BACNET_PROTOCOL_CLUSTER_MAX_NPDU_SIZE 504

```

basic-callback.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generatePluginCallbackPrototypes()
//

void emberAfPluginBasicResetToFactoryDefaultsCallback(void);
```

basic.h

Go to the documentation of this file.

```
// ******************************************************************
// * basic.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.          *80*
// ******************************************************************
```

call-command-handler.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generateParseCommandHeader()
//

// Enclosing macro to prevent multiple inclusion
#ifndef __EMBER_AF_COMMAND_PARSE_HEADER__
#define __EMBER_AF_COMMAND_PARSE_HEADER__


// This is a set of generated prototype for functions that parse the
// the incomming message, and call appropriate command handler.


// Cluster: Basic, client
//    No commands avialable.


// Cluster: Basic, server
EmberAfStatus emberAfBasicClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Power Configuration, client
//    No commands avialable.


// Cluster: Power Configuration, server
//    No commands avialable.


// Cluster: Device Temperature Configuration, client
//    No commands avialable.


// Cluster: Device Temperature Configuration, server
//    No commands avialable.


// Cluster: Identify, client
EmberAfStatus emberAfIdentifyClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Identify, server
EmberAfStatus emberAfIdentifyClusterServerCommandParse( EmberAfClusterCommand *cmd);

```

```
// Cluster: Groups, client
EmberAfStatus emberAfGroupsClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Groups, server
EmberAfStatus emberAfGroupsClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Scenes, client
EmberAfStatus emberAfScenesClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Scenes, server
EmberAfStatus emberAfScenesClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: On/off, client
//    No commands avialable.


// Cluster: On/off, server
EmberAfStatus emberAfOnOffClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: On/off Switch Configuration, client
//    No commands avialable.


// Cluster: On/off Switch Configuration, server
//    No commands avialable.


// Cluster: Level Control, client
//    No commands avialable.


// Cluster: Level Control, server
EmberAfStatus emberAfLevelControlClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Alarms, client
EmberAfStatus emberAfAlarmClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Alarms, server
EmberAfStatus emberAfAlarmClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Time, client
//    No commands avialable.

```

```

// Cluster: Time, server
//    No commands avialable.



// Cluster: RSSI Location, client
EmberAfStatus emberAfRssiLocationClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: RSSI Location, server
EmberAfStatus emberAfRssiLocationClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Binary Input (Basic), client
//    No commands avialable.


// Cluster: Binary Input (Basic), server
//    No commands avialable.


// Cluster: Commissioning, client
EmberAfStatus emberAfCommissioningClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Commissioning, server
EmberAfStatus emberAfCommissioningClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Shade Configuration, client
//    No commands avialable.


// Cluster: Shade Configuration, server
//    No commands avialable.


// Cluster: Pump Configuration and Control, client
//    No commands avialable.


// Cluster: Pump Configuration and Control, server
//    No commands avialable.


// Cluster: Thermostat, client
//    No commands avialable.


// Cluster: Thermostat, server
EmberAfStatus emberAfThermostatClusterServerCommandParse( EmberAfClusterCommand *cmd);

```

```
00169
00170
00171 // Cluster: Fan Control, client
00172 //    No commands avialable.
00173
00174
00175 // Cluster: Fan Control, server
00176 //    No commands avialable.
00177
00178
00179
00180 // Cluster: Dehumidification Control, client
00181 //    No commands avialable.
00182
00183
00184 // Cluster: Dehumidification Control, server
00185 //    No commands avialable.
00186
00187
00188
00189 // Cluster: Thermostat User Interface Configuration, client
00190 //    No commands avialable.
00191
00192
00193 // Cluster: Thermostat User Interface Configuration, server
00194 //    No commands avialable.
00195
00196
00197
00198 // Cluster: Color Control, client
00199 //    No commands avialable.
00200
00201
00202 // Cluster: Color Control, server
00203 EmberAfStatus emberAfColorControlClusterServerCommandParse( EmberAfClusterCommand *cmd);
00204
00205
00206
00207 // Cluster: Ballast Configuration, client
00208 //    No commands avialable.
00209
00210
00211 // Cluster: Ballast Configuration, server
00212 //    No commands avialable.
00213
00214
00215
00216 // Cluster: Illuminance Measurement, client
00217 //    No commands avialable.
00218
00219
00220 // Cluster: Illuminance Measurement, server
00221 //    No commands avialable.
00222
00223
00224
00225 // Cluster: Illuminance Level Sensing, client
00226 //    No commands avialable.
00227
```

```
// Cluster: Illuminance Level Sensing, server
//    No commands avialable.


// Cluster: Temperature Measurement, client
//    No commands avialable.


// Cluster: Temperature Measurement, server
//    No commands avialable.


// Cluster: Pressure Measurement, client
//    No commands avialable.


// Cluster: Pressure Measurement, server
//    No commands avialable.


// Cluster: Flow Measurement, client
//    No commands avialable.


// Cluster: Flow Measurement, server
//    No commands avialable.


// Cluster: Relative Humidity Measurement, client
//    No commands avialable.


// Cluster: Relative Humidity Measurement, server
//    No commands avialable.


// Cluster: Occupancy Sensing, client
//    No commands avialable.


// Cluster: Occupancy Sensing, server
//    No commands avialable.


// Cluster: IAS Zone, client
EmberAfStatus emberAfIasZoneClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: IAS Zone, server
EmberAfStatus emberAfIasZoneClusterServerCommandParse( EmberAfClusterCommand *cmd);

```

```
// Cluster: IAS ACE, client
EmberAfStatus emberAfIasAceClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: IAS ACE, server
EmberAfStatus emberAfIasAceClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: IAS WD, client
//    No commands avialable.


// Cluster: IAS WD, server
EmberAfStatus emberAfIasWdClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Door Lock, client
EmberAfStatus emberAfDoorLockClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Door Lock, server
EmberAfStatus emberAfDoorLockClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Window Covering, client
//    No commands avialable.


// Cluster: Window Covering, server
EmberAfStatus emberAfWindowCoveringClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Generic Tunnel, client
EmberAfStatus emberAfGenericTunnelClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Generic Tunnel, server
EmberAfStatus emberAfGenericTunnelClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: BACnet Protocol Tunnel, client
//    No commands avialable.


// Cluster: BACnet Protocol Tunnel, server
EmberAfStatus emberAfBacnetProtocolTunnelClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Demand Response and Load Control, client
EmberAfStatus emberAfDemandResponseLoadControlClusterClientCommandParse( EmberAfClusterCommand *cmd);

```

```
// Cluster: Demand Response and Load Control, server
EmberAfStatus emberAfDemandResponseLoadControlClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Simple Metering, client
EmberAfStatus emberAfSimpleMeteringClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Simple Metering, server
EmberAfStatus emberAfSimpleMeteringClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Price, client
EmberAfStatus emberAfPriceClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Price, server
EmberAfStatus emberAfPriceClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Messaging, client
EmberAfStatus emberAfMessagingClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Messaging, server
EmberAfStatus emberAfMessagingClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Tunneling, client
EmberAfStatus emberAfTunnelingClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Tunneling, server
EmberAfStatus emberAfTunnelingClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Prepayment, client
EmberAfStatus emberAfPrepaymentClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Prepayment, server
EmberAfStatus emberAfPrepaymentClusterServerCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Key establishment, client
EmberAfStatus emberAfKeyEstablishmentClusterClientCommandParse( EmberAfClusterCommand *cmd);


// Cluster: Key establishment, server
EmberAfStatus emberAfKeyEstablishmentClusterServerCommandParse( EmberAfClusterCommand *cmd);
```

```
00402
00403
00404
00405 // Cluster: Sample Mfg Specific Cluster, client
00406 //    No commands avialable.
00407
00408
00409 // Cluster: Sample Mfg Specific Cluster, server
00410 EmberAfStatus emberAfSampleMfgSpecificClusterServerCommandParse( EmberAfClusterCommand *cmd);
00411
00412
00413
00414 #endif // __EMBER_AF_COMMAND_PARSE_HEADER__
```

callback.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generateCallbackPrototypes()
//

// Enclosing macro to prevent multiple inclusion
#ifndef __EMBER_AF_CALLBACK_PROTOTYPES__
#define __EMBER_AF_CALLBACK_PROTOTYPES__


#include "app/framework/include/af-types.h"


// @{
boolean emberAfReadAttributesResponseCallback(EmberAfClusterId clusterId,
                                              int8u * buffer,
                                              int16u bufLen);
boolean emberAfWriteAttributesResponseCallback(EmberAfClusterId clusterId,
                                               int8u * buffer,
                                               int16u bufLen);
boolean emberAfConfigureReportingResponseCallback(EmberAfClusterId clusterId,
                                                  int8u * buffer,
                                                  int16u bufLen);
boolean emberAfReadReportingConfigurationResponseCallback(EmberAfClusterId clusterId,
                                                          int8u * buffer,
                                                          int16u bufLen);
boolean emberAfReportAttributesCallback(EmberAfClusterId clusterId,
                                        int8u * buffer,
                                        int16u bufLen);
boolean emberAfDefaultResponseCallback(EmberAfClusterId clusterId,
                                       int8u commandId,
                                       EmberAfStatus status);
boolean emberAfDiscoverAttributesResponseCallback(EmberAfClusterId clusterId,
                                                  boolean discoveryComplete,
                                                  int8u * buffer,
                                                  int16u bufLen);
EmberAfAttributeWritePermission emberAfAllowNetworkWriteAttributeCallback(int8u endpoint,
                                                                          EmberAfClusterId clusterId,
                                                                          EmberAfAttributeId attributeId,
                                                                          int8u mask,
                                                                          int16u manufacturerCode,
                                                                          int8u* value,
                                                                          int8u type);
void emberAfPreAttributeChangeCallback(int8u endpoint,
                                       EmberAfClusterId clusterId,
                                       EmberAfAttributeId attributeId,
                                       int8u mask,
                                       int16u manufacturerCode,
                                       int8u type,
```

```
                                int8u size,
                                int8u* value);
void emberAfPostAttributeChangeCallback(int8u endpoint,
                                EmberAfClusterId clusterId,
                                EmberAfAttributeId attributeId,
                                int8u mask,
                                int16u manufacturerCode,
                                int8u type,
                                int8u size,
                                int8u* value);
boolean emberAfPreMessageReceivedCallback(EmberAfIncomingMessage* incomingMessage);
boolean emberAfPreCommandReceivedCallback(EmberAfClusterCommand* cmd);
boolean emberAfPreZDOMessageReceivedCallback(EmberNodeId emberNodeId,
                                    EmberApsFrame* apsFrame,
                                    int8u* message,
                                    int16u length);
boolean emberAfMessageSentCallback(EmberOutgoingMessageType type,
                            int16u indexOrDestination,
                            EmberApsFrame* apsFrame,
                            int16u msgLen,
                            int8u* message,
                            EmberStatus status);
boolean emberAfStackStatusCallback(EmberStatus status);
void emberAfNcpInitCallback(void);
void emberAfMainInitCallback(void);
void emberAfClusterInitCallback(int8u endpoint,
                          EmberAfClusterId clusterId);
boolean emberAfPreGoToSleepCallback(int32u sleepDurationAttempt);
void emberAfPostWakeUpCallback(int32u sleepDuration);
void emberAfMainTickCallback(void);
boolean emberAfExternalAttributeWriteCallback(int8u endpoint,
                                    EmberAfClusterId clusterId,
                                    EmberAfAttributeMetadata * attributeMetadata,
                                    int16u manufacturerCode,
                                    int8u * buffer);
boolean emberAfExternalAttributeReadCallback(int8u endpoint,
                                    EmberAfClusterId clusterId,
                                    EmberAfAttributeMetadata * attributeMetadata,
                                    int16u manufacturerCode,
                                    int8u * buffer);
EmberAfStatus emberAfReportEntryConfiguredCallback(EmberAfReportEntry * entry);
void emberAfTrustCenterJoinCallback(EmberNodeId newNodeId,
                            EmberEUI64 newNodeEui64,
                            EmberNodeId parentOfNewNode,
                            EmberDeviceUpdate status,
                            EmberJoinDecision decision);
int32u emberAfGetCurrentTimeCallback(void);
void emberAfSetTimeCallback(int32u utcTime);
EmberStatus emberAfInitiateKeyEstablishmentCallback(EmberNodeId nodeId,
                                        int8u endpoint);
EmberStatus emberAfInitiateInterPanKeyEstablishmentCallback(EmberPanId panId,
                                            const EmberEUI64 eui64);
boolean emberAfKeyEstablishmentCallback(EmberAfKeyEstablishmentNotifyMessage status,
                                boolean amInitiator,
                                EmberNodeId partnerShortId,
                                int8u delayInSeconds);
```

```
boolean emberAfInterPanKeyEstablishmentCallback(EmberAfKeyEstablishmentNotifyMessage status,
                                                boolean amInitiator,
                                                EmberPanId panId,
                                                const EmberEUI64 eui64,
                                                int8u delayInSeconds);
boolean emberAfPerformingKeyEstablishmentCallback(void);
EmberStatus emberAfInitiatePartnerLinkKeyExchangeCallback(EmberNodeId target,
                                                          int8u endpoint,
                                                          EmberAfPartnerLinkKeyExchangeCallback * callback);
EmberStatus emberAfPartnerLinkKeyExchangeRequestCallback(EmberEUI64 partner);
void emberAfPartnerLinkKeyExchangeResponseCallback(EmberNodeId sender,
                                                   EmberZdoStatus status);
EmberStatus emberAfRegistrationStartCallback(void);
void emberAfRegistrationAbortCallback(void);
void emberAfRegistrationCallback(boolean success);
void emberAfTrustCenterKeepaliveUpdateCallback(boolean registrationComplete);
void emberAfTrustCenterKeepaliveAbortCallback(void);
void emberAfCustomCommandLineCallback(void);
boolean emberAfOtaServerIncomingMessageRawCallback(EmberAfClusterCommand * message);
boolean emberAfOtaServerSendImageNotifyCallback(EmberNodeId dest,
                                                int8u endpoint,
                                                int8u payloadType,
                                                int8u queryJitter,
                                                const EmberAfOtaImageId* id);
boolean emberAfOtaStorageDriverInitCallback(void);
boolean emberAfOtaStorageDriverReadCallback(int32u offset,
                                            int32u length,
                                            int8u* returnData);
boolean emberAfOtaStorageDriverWriteCallback(const int8u* dataToWrite,
                                             int32u offset,
                                             int32u length);
int8u emberAfOtaBootloadCallback(const EmberAfOtaImageId* id,
                                 int16u ncpUpgradeTagId);
EmberAfOtaStorageStatus emberAfOtaStorageInitCallback(void);
void emberAfOtaStorageCloseCallback(void);
int8u emberAfOtaStorageGetCountCallback(void);
EmberAfOtaImageId emberAfOtaStorageSearchCallback(int16u manufacturerId,
                                                  int16u imageTypeId,
                                                  const int16u* hardwareVersion);
int32u emberAfOtaStorageGetTotalImageSizeCallback(const EmberAfOtaImageId* id);
EmberAfOtaImageId emberAfOtaStorageIteratorFirstCallback(void);
EmberAfOtaImageId emberAfOtaStorageIteratorNextCallback(void);
EmberAfOtaStorageStatus emberAfOtaStorageGetFullHeaderCallback(const EmberAfOtaImageId* id,
                                                               EmberAfOtaHeader* returnData);
EmberAfOtaStorageStatus emberAfOtaStorageReadImageDataCallback(const EmberAfOtaImageId* id,
                                                               int32u offset,
                                                               int32u length,
                                                               int8u* returnData,
                                                               int32u* returnedLength);
EmberAfOtaStorageStatus emberAfOtaStorageClearTempDataCallback(void);
EmberAfOtaStorageStatus emberAfOtaStorageWriteTempDataCallback(int32u offset,
                                                               int32u length,
                                                               const int8u* data);
EmberAfOtaStorageStatus emberAfOtaStorageCheckTempDataCallback(int32u* currentOffset,
                                                               int32u* totalImageSize,
                                                               EmberAfOtaImageId* newFileInfo);
```

```
EmberAfOtaStorageStatus emberAfOtaStorageFinishDownloadCallback(int32u offset);
boolean emberAfOtaClientIncomingMessageRawCallback(EmberAfClusterCommand * message);
void emberAfOtaClientStartCallback(void);
void emberAfOtaClientVersionInfoCallback(EmberAfOtaImageId* currentImageInfo,
                                         int16u* hardwareVersion);
EmberAfImageVerifyStatus emberAfOtaClientCustomVerifyCallback(boolean newVerification,
                                                              const EmberAfOtaImageId* id);
boolean emberAfOtaClientDownloadCompleteCallback(boolean success,
                                                 const EmberAfOtaImageId* id);
void emberAfOtaClientBootloadCallback(const EmberAfOtaImageId* id);
int8u emberAfOtaServerQueryCallback(const EmberAfOtaImageId* currentImageId,
                                    int16u* hardwareVersion,
                                    EmberAfOtaImageId* nextUpgradeImageId);
int8u emberAfOtaServerBlockSizeCallback(EmberNodeId clientNodeId);
int8u emberAfOtaServerImageBlockRequestCallback(const EmberAfOtaImageId* id,
                                                int32u offset,
                                                int8u maxDataSize,
                                                int32u* waitTimeMinutes);
boolean emberAfOtaServerUpgradeEndRequestCallback(EmberNodeId source,
                                                  int8u status,
                                                  int32u* returnValue,
                                                  const EmberAfOtaImageId* imageId);
int8u emberAfOtaPageRequestServerPolicyCallback(void);
EmberStatus emberAfInterpanSendMessageCallback(EmberAfInterpanHeader* header,
                                               int16u messageLength,
                                               int8u* message);
void emberAfInterpanInitCallback(void);
void emberAfUnusedPanIdFoundCallback(EmberPanId panId,
                                     int8u channel);
void emberAfJoinableNetworkFoundCallback(EmberZigbeeNetwork * networkFound,
                                         int8u lqi,
                                         int8s rssi);
void emberAfScanErrorCallback(EmberStatus status);
EmberStatus emberAfFindUnusedPanIdAndFormCallback(void);
EmberStatus emberAfStartSearchForJoinableNetworkCallback(void);
void emberAfContinueJoinableNetworkSearchCallback(void);
void emberAfNetworkFindOperationCompleteCallback(void);
// @{

boolean emberAfBasicClusterResetToFactoryDefaultsCallback(void);
void emberAfBasicClusterServerInitCallback(int8u endpoint);
void emberAfBasicClusterClientInitCallback(int8u endpoint);
void emberAfBasicClusterServerTickCallback(int8u endpoint);
void emberAfBasicClusterClientTickCallback(int8u endpoint);
void emberAfBasicClusterServerAttributeChangedCallback(int8u endpoint,
                                                       EmberAfAttributeId attributeId);
void emberAfBasicClusterClientAttributeChangedCallback(int8u endpoint,
                                                       EmberAfAttributeId attributeId);
void emberAfBasicClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                                           EmberAfAttributeId attributeId,
                                                                           int16u manufacturerCode);
void emberAfBasicClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                                           EmberAfAttributeId attributeId,
                                                                           int16u manufacturerCode);
void emberAfBasicClusterServerDefaultResponseCallback(int8u endpoint,
```

```
01168                                   int8u commandId,
01169                                   EmberAfStatus status);
01178 void emberAfBasicClusterClientDefaultResponseCallback(int8u endpoint,
01179                                   int8u commandId,
01180                                   EmberAfStatus status);
01193 void emberAfBasicClusterServerMessageSentCallback(EmberOutgoingMessageType type,
01194                                 int16u indexOrDestination,
01195                                 EmberApsFrame * apsFrame,
01196                                 int16u msgLen,
01197                                 int8u * message,
01198                                 EmberStatus status);
01211 void emberAfBasicClusterClientMessageSentCallback(EmberOutgoingMessageType type,
01212                                 int16u indexOrDestination,
01213                                 EmberApsFrame * apsFrame,
01214                                 int16u msgLen,
01215                                 int8u * message,
01216                                 EmberStatus status);
01217
01221 // @{
01222
01229 void emberAfPowerConfigClusterServerInitCallback(int8u endpoint);
01236 void emberAfPowerConfigClusterClientInitCallback(int8u endpoint);
01243 void emberAfPowerConfigClusterServerTickCallback(int8u endpoint);
01250 void emberAfPowerConfigClusterClientTickCallback(int8u endpoint);
01258 void emberAfPowerConfigClusterServerAttributeChangedCallback(int8u endpoint,
01259                                   EmberAfAttributeId attributeId);
01267 void emberAfPowerConfigClusterClientAttributeChangedCallback(int8u endpoint,
01268                                   EmberAfAttributeId attributeId);
01278 void emberAfPowerConfigClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
01279                                                   EmberAfAttributeId attributeId,
01280                                                   int16u manufacturerCode);
01290 void emberAfPowerConfigClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
01291                                                   EmberAfAttributeId attributeId,
01292                                                   int16u manufacturerCode);
01301 void emberAfPowerConfigClusterServerDefaultResponseCallback(int8u endpoint,
01302                                   int8u commandId,
01303                                   EmberAfStatus status);
01312 void emberAfPowerConfigClusterClientDefaultResponseCallback(int8u endpoint,
01313                                   int8u commandId,
01314                                   EmberAfStatus status);
01327 void emberAfPowerConfigClusterServerMessageSentCallback(EmberOutgoingMessageType type,
01328                                 int16u indexOrDestination,
01329                                 EmberApsFrame * apsFrame,
01330                                 int16u msgLen,
01331                                 int8u * message,
01332                                 EmberStatus status);
01345 void emberAfPowerConfigClusterClientMessageSentCallback(EmberOutgoingMessageType type,
01346                                 int16u indexOrDestination,
01347                                 EmberApsFrame * apsFrame,
01348                                 int16u msgLen,
01349                                 int8u * message,
01350                                 EmberStatus status);
01351
01355 // @{
01356
01363 void emberAfDeviceTempClusterServerInitCallback(int8u endpoint);
01370 void emberAfDeviceTempClusterClientInitCallback(int8u endpoint);
```

```
01377 void emberAfDeviceTempClusterServerTickCallback(int8u endpoint);
01384 void emberAfDeviceTempClusterClientTickCallback(int8u endpoint);
01392 void emberAfDeviceTempClusterServerAttributeChangedCallback(int8u endpoint,
01393                                                             EmberAfAttributeId attributeId);
01401 void emberAfDeviceTempClusterClientAttributeChangedCallback(int8u endpoint,
01402                                                             EmberAfAttributeId attributeId);
01413 void emberAfDeviceTempClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
01414                                                                                 EmberAfAttributeId attributeId,
01415                                                                                 int16u manufacturerCode);
01426 void emberAfDeviceTempClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
01427                                                                                 EmberAfAttributeId attributeId,
01428                                                                                 int16u manufacturerCode);
01437 void emberAfDeviceTempClusterServerDefaultResponseCallback(int8u endpoint,
01438                                                           int8u commandId,
01439                                                           EmberAfStatus status);
01448 void emberAfDeviceTempClusterClientDefaultResponseCallback(int8u endpoint,
01449                                                           int8u commandId,
01450                                                           EmberAfStatus status);
01463 void emberAfDeviceTempClusterServerMessageSentCallback(EmberOutgoingMessageType type,
01464                                                        int16u indexOrDestination,
01465                                                        EmberApsFrame * apsFrame,
01466                                                        int16u msgLen,
01467                                                        int8u * message,
01468                                                        EmberStatus status);
01481 void emberAfDeviceTempClusterClientMessageSentCallback(EmberOutgoingMessageType type,
01482                                                        int16u indexOrDestination,
01483                                                        EmberApsFrame * apsFrame,
01484                                                        int16u msgLen,
01485                                                        int8u * message,
01486                                                        EmberStatus status);
01487
01491 // @{
01492
01498 boolean emberAfIdentifyClusterIdentifyCallback(int16u identifyTime);
01503 boolean emberAfIdentifyClusterIdentifyQueryCallback(void);
01509 boolean emberAfIdentifyClusterIdentifyQueryResponseCallback(int16u timeout);
01516 void emberAfIdentifyClusterServerInitCallback(int8u endpoint);
01523 void emberAfIdentifyClusterClientInitCallback(int8u endpoint);
01530 void emberAfIdentifyClusterServerTickCallback(int8u endpoint);
01537 void emberAfIdentifyClusterClientTickCallback(int8u endpoint);
01545 void emberAfIdentifyClusterServerAttributeChangedCallback(int8u endpoint,
01546                                                           EmberAfAttributeId attributeId);
01554 void emberAfIdentifyClusterClientAttributeChangedCallback(int8u endpoint,
01555                                                           EmberAfAttributeId attributeId);
01565 void emberAfIdentifyClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
01566                                                                               EmberAfAttributeId attributeId,
01567                                                                               int16u manufacturerCode);
01577 void emberAfIdentifyClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
01578                                                                               EmberAfAttributeId attributeId,
01579                                                                               int16u manufacturerCode);
01588 void emberAfIdentifyClusterServerDefaultResponseCallback(int8u endpoint,
01589                                                         int8u commandId,
01590                                                         EmberAfStatus status);
01599 void emberAfIdentifyClusterClientDefaultResponseCallback(int8u endpoint,
01600                                                         int8u commandId,
01601                                                         EmberAfStatus status);
```

```
01614 void emberAfIdentifyClusterServerMessageSentCallback(EmberOutgoingMessageType type,
01615                                                      int16u indexOrDestination,
01616                                                      EmberApsFrame * apsFrame,
01617                                                      int16u msgLen,
01618                                                      int8u * message,
01619                                                      EmberStatus status);
01632 void emberAfIdentifyClusterClientMessageSentCallback(EmberOutgoingMessageType type,
01633                                                      int16u indexOrDestination,
01634                                                      EmberApsFrame * apsFrame,
01635                                                      int16u msgLen,
01636                                                      int8u * message,
01637                                                      EmberStatus status);
01638
01642 // @{
01643
01653 boolean emberAfGroupsClusterEndpointInGroupCallback(int8u endpoint,
01654                                                     int16u groupId);
01661 void emberAfGroupsClusterClearGroupTableCallback(void);
01668 boolean emberAfGroupsClusterAddGroupCallback(int16u groupId,
01669                                              int8u* groupName);
01675 boolean emberAfGroupsClusterViewGroupCallback(int16u groupId);
01682 boolean emberAfGroupsClusterGetGroupMembershipCallback(int8u groupCount,
01683                                                        int8u* groupList);
01689 boolean emberAfGroupsClusterRemoveGroupCallback(int16u groupId);
01694 boolean emberAfGroupsClusterRemoveAllGroupsCallback(void);
01701 boolean emberAfGroupsClusterAddGroupIfIdentifyingCallback(int16u groupId,
01702                                                           int8u* groupName);
01709 boolean emberAfGroupsClusterAddGroupResponseCallback(int8u status,
01710                                                      int16u groupId);
01718 boolean emberAfGroupsClusterViewGroupResponseCallback(int8u status,
01719                                                       int16u groupId,
01720                                                       int8u* groupName);
01728 boolean emberAfGroupsClusterGetGroupMembershipResponseCallback(int8u capacity,
01729                                                                int8u groupCount,
01730                                                                int8u* groupList);
01737 boolean emberAfGroupsClusterRemoveGroupResponseCallback(int8u status,
01738                                                         int16u groupId);
01745 void emberAfGroupsClusterServerInitCallback(int8u endpoint);
01752 void emberAfGroupsClusterClientInitCallback(int8u endpoint);
01759 void emberAfGroupsClusterServerTickCallback(int8u endpoint);
01766 void emberAfGroupsClusterClientTickCallback(int8u endpoint);
01774 void emberAfGroupsClusterServerAttributeChangedCallback(int8u endpoint,
01775                                                       EmberAfAttributeId attributeId);
01783 void emberAfGroupsClusterClientAttributeChangedCallback(int8u endpoint,
01784                                                       EmberAfAttributeId attributeId);
01794 void emberAfGroupsClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
01795                                                                             EmberAfAttributeId attributeId,
01796                                                                             int16u manufacturerCode);
01806 void emberAfGroupsClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
01807                                                                             EmberAfAttributeId attributeId,
01808                                                                             int16u manufacturerCode);
01817 void emberAfGroupsClusterServerDefaultResponseCallback(int8u endpoint,
01818                                                       int8u commandId,
01819                                                       EmberAfStatus status);
01828 void emberAfGroupsClusterClientDefaultResponseCallback(int8u endpoint,
01829                                                       int8u commandId,
```

```
01830                                       EmberAfStatus status);
01843 void emberAfGroupsClusterServerMessageSentCallback(EmberOutgoingMessageType type,
01844                                       int16u indexOrDestination,
01845                                       EmberApsFrame * apsFrame,
01846                                       int16u msgLen,
01847                                       int8u * message,
01848                                       EmberStatus status);
01861 void emberAfGroupsClusterClientMessageSentCallback(EmberOutgoingMessageType type,
01862                                       int16u indexOrDestination,
01863                                       EmberApsFrame * apsFrame,
01864                                       int16u msgLen,
01865                                       int8u * message,
01866                                       EmberStatus status);
01867
01871 // @{
01872
01885 EmberAfStatus emberAfScenesClusterStoreCurrentSceneCallback(int8u endpoint,
01886                                         int16u groupId,
01887                                         int8u sceneId);
01897 EmberAfStatus emberAfScenesClusterRecallSavedSceneCallback(int8u endpoint,
01898                                        int16u groupId,
01899                                        int8u sceneId);
01906 void emberAfScenesClusterClearSceneTableCallback(void);
01916 boolean emberAfScenesClusterAddSceneCallback(int16u groupId,
01917                                   int8u sceneId,
01918                                   int16u transitionTime,
01919                                   int8u* sceneName,
01920                                   int8u* extensionFieldSets);
01927 boolean emberAfScenesClusterViewSceneCallback(int16u groupId,
01928                                    int8u sceneId);
01935 boolean emberAfScenesClusterRemoveSceneCallback(int16u groupId,
01936                                     int8u sceneId);
01942 boolean emberAfScenesClusterRemoveAllScenesCallback(int16u groupId);
01949 boolean emberAfScenesClusterStoreSceneCallback(int16u groupId,
01950                                   int8u sceneId);
01957 boolean emberAfScenesClusterRecallSceneCallback(int16u groupId,
01958                                     int8u sceneId);
01964 boolean emberAfScenesClusterGetSceneMembershipCallback(int16u groupId);
01972 boolean emberAfScenesClusterAddSceneResponseCallback(int8u status,
01973                                      int16u groupId,
01974                                      int8u sceneId);
01985 boolean emberAfScenesClusterViewSceneResponseCallback(int8u status,
01986                                       int16u groupId,
01987                                       int8u sceneId,
01988                                       int16u transitionTime,
01989                                       int8u* sceneName,
01990                                       int8u* extensionFieldSets);
01998 boolean emberAfScenesClusterRemoveSceneResponseCallback(int8u status,
01999                                        int16u groupId,
02000                                        int8u sceneId);
02007 boolean emberAfScenesClusterRemoveAllScenesResponseCallback(int8u status,
02008                                         int16u groupId);
02016 boolean emberAfScenesClusterStoreSceneResponseCallback(int8u status,
02017                                       int16u groupId,
02018                                       int8u sceneId);
02028 boolean emberAfScenesClusterGetSceneMembershipResponseCallback(int8u status,
02029                                           int8u capacity,
02030                                           int16u groupId,
```

```
                                        int8u sceneCount,
                                        int8u* sceneList);
void emberAfScenesClusterServerInitCallback(int8u endpoint);
void emberAfScenesClusterClientInitCallback(int8u endpoint);
void emberAfScenesClusterServerTickCallback(int8u endpoint);
void emberAfScenesClusterClientTickCallback(int8u endpoint);
void emberAfScenesClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfScenesClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfScenesClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfScenesClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfScenesClusterServerDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfScenesClusterClientDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfScenesClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                        int16u indexOrDestination,
                                        EmberApsFrame * apsFrame,
                                        int16u msgLen,
                                        int8u * message,
                                        EmberStatus status);
void emberAfScenesClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                        int16u indexOrDestination,
                                        EmberApsFrame * apsFrame,
                                        int16u msgLen,
                                        int8u * message,
                                        EmberStatus status);

// @{

boolean emberAfOnOffClusterOffCallback(void);
boolean emberAfOnOffClusterOnCallback(void);
boolean emberAfOnOffClusterToggleCallback(void);
void emberAfOnOffClusterServerInitCallback(int8u endpoint);
void emberAfOnOffClusterClientInitCallback(int8u endpoint);
void emberAfOnOffClusterServerTickCallback(int8u endpoint);
void emberAfOnOffClusterClientTickCallback(int8u endpoint);
void emberAfOnOffClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfOnOffClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfOnOffClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfOnOffClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfOnOffClusterServerDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
```

```
                                    EmberAfStatus status);
void emberAfOnOffClusterClientDefaultResponseCallback(int8u endpoint,
                                    int8u commandId,
                                    EmberAfStatus status);
void emberAfOnOffClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                  int16u indexOrDestination,
                                  EmberApsFrame * apsFrame,
                                  int16u msgLen,
                                  int8u * message,
                                  EmberStatus status);
void emberAfOnOffClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                  int16u indexOrDestination,
                                  EmberApsFrame * apsFrame,
                                  int16u msgLen,
                                  int8u * message,
                                  EmberStatus status);
boolean emberAfOnOffClusterSampleMfgSpecificOffWithTransitionCallback(void);
boolean emberAfOnOffClusterSampleMfgSpecificOnWithTransitionCallback(void);
boolean emberAfOnOffClusterSampleMfgSpecificToggleWithTransitionCallback(void);

// @{

void emberAfOnOffSwitchConfigClusterServerInitCallback(int8u endpoint);
void emberAfOnOffSwitchConfigClusterClientInitCallback(int8u endpoint);
void emberAfOnOffSwitchConfigClusterServerTickCallback(int8u endpoint);
void emberAfOnOffSwitchConfigClusterClientTickCallback(int8u endpoint);
void emberAfOnOffSwitchConfigClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfOnOffSwitchConfigClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfOnOffSwitchConfigClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfOnOffSwitchConfigClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfOnOffSwitchConfigClusterServerDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfOnOffSwitchConfigClusterClientDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfOnOffSwitchConfigClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                      int16u indexOrDestination,
                                      EmberApsFrame * apsFrame,
                                      int16u msgLen,
                                      int8u * message,
                                      EmberStatus status);
void emberAfOnOffSwitchConfigClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                      int16u indexOrDestination,
                                      EmberApsFrame * apsFrame,
                                      int16u msgLen,
                                      int8u * message,
                                      EmberStatus status);

// @{
```

```
boolean emberAfLevelControlClusterMoveToLevelCallback(int8u level,
                                                      int16u transitionTime);
boolean emberAfLevelControlClusterMoveCallback(int8u moveMode,
                                               int8u rate);
boolean emberAfLevelControlClusterStepCallback(int8u stepMode,
                                               int8u stepSize,
                                               int16u transitionTime);
boolean emberAfLevelControlClusterStopCallback(void);
boolean emberAfLevelControlClusterMoveToLevelWithOnOffCallback(int8u level,
                                                               int16u transitionTime);
boolean emberAfLevelControlClusterMoveWithOnOffCallback(int8u moveMode,
                                                        int8u rate);
boolean emberAfLevelControlClusterStepWithOnOffCallback(int8u stepMode,
                                                        int8u stepSize,
                                                        int16u transitionTime);
boolean emberAfLevelControlClusterStopWithOnOffCallback(void);
void emberAfLevelControlClusterServerInitCallback(int8u endpoint);
void emberAfLevelControlClusterClientInitCallback(int8u endpoint);
void emberAfLevelControlClusterServerTickCallback(int8u endpoint);
void emberAfLevelControlClusterClientTickCallback(int8u endpoint);
void emberAfLevelControlClusterServerAttributeChangedCallback(int8u endpoint,
                                                              EmberAfAttributeId attributeId);
void emberAfLevelControlClusterClientAttributeChangedCallback(int8u endpoint,
                                                              EmberAfAttributeId attributeId);
void emberAfLevelControlClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                                                  EmberAfAttributeId attributeId,
                                                                                  int16u manufacturerCode);
void emberAfLevelControlClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                                                  EmberAfAttributeId attributeId,
                                                                                  int16u manufacturerCode);
void emberAfLevelControlClusterServerDefaultResponseCallback(int8u endpoint,
                                                             int8u commandId,
                                                             EmberAfStatus status);
void emberAfLevelControlClusterClientDefaultResponseCallback(int8u endpoint,
                                                             int8u commandId,
                                                             EmberAfStatus status);
void emberAfLevelControlClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                                         int16u indexOrDestination,
                                                         EmberApsFrame * apsFrame,
                                                         int16u msgLen,
                                                         int8u * message,
                                                         EmberStatus status);
void emberAfLevelControlClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                                         int16u indexOrDestination,
                                                         EmberApsFrame * apsFrame,
                                                         int16u msgLen,
                                                         int8u * message,
                                                         EmberStatus status);

// @{

boolean emberAfAlarmClusterResetAlarmCallback(int8u alarmCode,
                                              int16u clusterId);
boolean emberAfAlarmClusterResetAllAlarmsCallback(void);
boolean emberAfAlarmClusterGetAlarmCallback(void);
```

```
boolean emberAfAlarmClusterResetAlarmLogCallback(void);
boolean emberAfAlarmClusterAlarmCallback(int8u alarmCode,
                                         int16u clusterId);
boolean emberAfAlarmClusterGetAlarmResponseCallback(int8u status,
                                                    int8u alarmCode,
                                                    int16u clusterId,
                                                    int32u timeStamp);
void emberAfAlarmClusterServerInitCallback(int8u endpoint);
void emberAfAlarmClusterClientInitCallback(int8u endpoint);
void emberAfAlarmClusterServerTickCallback(int8u endpoint);
void emberAfAlarmClusterClientTickCallback(int8u endpoint);
void emberAfAlarmClusterServerAttributeChangedCallback(int8u endpoint,
                                                       EmberAfAttributeId attributeId);
void emberAfAlarmClusterClientAttributeChangedCallback(int8u endpoint,
                                                       EmberAfAttributeId attributeId);
void emberAfAlarmClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                                           EmberAfAttributeId attributeId,
                                                                           int16u manufacturerCode);
void emberAfAlarmClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                                           EmberAfAttributeId attributeId,
                                                                           int16u manufacturerCode);
void emberAfAlarmClusterServerDefaultResponseCallback(int8u endpoint,
                                                      int8u commandId,
                                                      EmberAfStatus status);
void emberAfAlarmClusterClientDefaultResponseCallback(int8u endpoint,
                                                      int8u commandId,
                                                      EmberAfStatus status);
void emberAfAlarmClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                                  int16u indexOrDestination,
                                                  EmberApsFrame * apsFrame,
                                                  int16u msgLen,
                                                  int8u * message,
                                                  EmberStatus status);
void emberAfAlarmClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                                  int16u indexOrDestination,
                                                  EmberApsFrame * apsFrame,
                                                  int16u msgLen,
                                                  int8u * message,
                                                  EmberStatus status);

// @{

void emberAfTimeClusterServerInitCallback(int8u endpoint);
void emberAfTimeClusterClientInitCallback(int8u endpoint);
void emberAfTimeClusterServerTickCallback(int8u endpoint);
void emberAfTimeClusterClientTickCallback(int8u endpoint);
void emberAfTimeClusterServerAttributeChangedCallback(int8u endpoint,
                                                      EmberAfAttributeId attributeId);
void emberAfTimeClusterClientAttributeChangedCallback(int8u endpoint,
                                                      EmberAfAttributeId attributeId);
void emberAfTimeClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                                          EmberAfAttributeId attributeId,
                                                                          int16u manufacturerCode);
void emberAfTimeClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                                          EmberAfAttributeId attributeId,
                                                                          int16u manufacturerCode);
```

```
02918 void emberAfTimeClusterServerDefaultResponseCallback(int8u endpoint,
02919                                                      int8u commandId,
02920                                                      EmberAfStatus status);
02929 void emberAfTimeClusterClientDefaultResponseCallback(int8u endpoint,
02930                                                      int8u commandId,
02931                                                      EmberAfStatus status);
02944 void emberAfTimeClusterServerMessageSentCallback(EmberOutgoingMessageType type,
02945                                                  int16u indexOrDestination,
02946                                                  EmberApsFrame * apsFrame,
02947                                                  int16u msgLen,
02948                                                  int8u * message,
02949                                                  EmberStatus status);
02962 void emberAfTimeClusterClientMessageSentCallback(EmberOutgoingMessageType type,
02963                                                  int16u indexOrDestination,
02964                                                  EmberApsFrame * apsFrame,
02965                                                  int16u msgLen,
02966                                                  int8u * message,
02967                                                  EmberStatus status);
02968
02972 // @{
02973
02983 boolean emberAfRssiLocationClusterSetAbsoluteLocationCallback(int16s coordinate1,
02984                                                               int16s coordinate2,
02985                                                               int16s coordinate3,
02986                                                               int16s power,
02987                                                               int16u pathLossExponent);
02997 boolean emberAfRssiLocationClusterSetDeviceConfigurationCallback(int16s power,
02998                                                                  int16u pathLossExponent,
02999                                                                  int16u calculationPeriod,
03000                                                                  int8u numberRssiMeasurements,
03001                                                                  int16u reportingPeriod);
03007 boolean emberAfRssiLocationClusterGetDeviceConfigurationCallback(int8u* targetAddress);
03015 boolean emberAfRssiLocationClusterGetLocationDataCallback(int8u flags,
03016                                                           int8u numberResponses,
03017                                                           int8u* targetAddress);
03028 boolean emberAfRssiLocationClusterRssiResponseCallback(int8u* replyingDevice,
03029                                                        int16s coordinate1,
03030                                                        int16s coordinate2,
03031                                                        int16s coordinate3,
03032                                                        int8s rssi,
03033                                                        int8u numberRssiMeasurements);
03041 boolean emberAfRssiLocationClusterSendPingsCallback(int8u* targetAddress,
03042                                                     int8u numberRssiMeasurements,
03043                                                     int16u calculationPeriod);
03052 boolean emberAfRssiLocationClusterAnchorNodeAnnounceCallback(int8u* anchorNodeIeeeAddress,
03053                                                              int16s coordinate1,
03054                                                              int16s coordinate2,
03055                                                              int16s coordinate3);
03066 boolean emberAfRssiLocationClusterDeviceConfigurationResponseCallback(int8u status,
03067                                                                       int16s power,
03068                                                                       int16u pathLossExponent,
03069                                                                       int16u calculationPeriod,
03070                                                                       int8u numberRssiMeasurements,
03071                                                                       int16u reportingPeriod);
03086 boolean emberAfRssiLocationClusterLocationDataResponseCallback(int8u status,
03087                                                                int8u locationType,
03088                                                                int16s coordinate1,
03089                                                                int16s coordinate2,
```

```
                                        int16s coordinate3,
                                        int16s power,
                                        int16u pathLossExponent,
                                        int8u locationMethod,
                                        int8u qualityMeasure,
                                        int16u locationAge);
boolean emberAfRssiLocationClusterLocationDataNotificationCallback(int8u locationType,
                                          int16s coordinate1,
                                          int16s coordinate2,
                                          int16s coordinate3,
                                          int16s power,
                                          int16u pathLossExponent,
                                          int8u locationMethod,
                                          int8u qualityMeasure,
                                          int16u locationAge);
boolean emberAfRssiLocationClusterCompactLocationDataNotificationCallback(int8u locationType,
                                              int16s coordinate1,
                                              int16s coordinate2,
                                              int16s coordinate3,
                                              int8u qualityMeasure,
                                              int16u locationAge);
boolean emberAfRssiLocationClusterRssiPingCallback(int8u locationType);
boolean emberAfRssiLocationClusterRssiRequestCallback(void);
boolean emberAfRssiLocationClusterReportRssiMeasurementsCallback(int8u* measuringDevice,
                                         int8u neighbors,
                                         int8u* neighborsInfo);
boolean emberAfRssiLocationClusterRequestOwnLocationCallback(int8u* blindNode);
void emberAfRssiLocationClusterServerInitCallback(int8u endpoint);
void emberAfRssiLocationClusterClientInitCallback(int8u endpoint);
void emberAfRssiLocationClusterServerTickCallback(int8u endpoint);
void emberAfRssiLocationClusterClientTickCallback(int8u endpoint);
void emberAfRssiLocationClusterServerAttributeChangedCallback(int8u endpoint,
                                       EmberAfAttributeId attributeId);
void emberAfRssiLocationClusterClientAttributeChangedCallback(int8u endpoint,
                                       EmberAfAttributeId attributeId);
void emberAfRssiLocationClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                           EmberAfAttributeId attributeId,
                                           int16u manufacturerCode);
void emberAfRssiLocationClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                           EmberAfAttributeId attributeId,
                                           int16u manufacturerCode);
void emberAfRssiLocationClusterServerDefaultResponseCallback(int8u endpoint,
                                      int8u commandId,
                                      EmberAfStatus status);
void emberAfRssiLocationClusterClientDefaultResponseCallback(int8u endpoint,
                                      int8u commandId,
                                      EmberAfStatus status);
void emberAfRssiLocationClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                     int16u indexOrDestination,
                                     EmberApsFrame * apsFrame,
                                     int16u msgLen,
                                     int8u * message,
                                     EmberStatus status);
void emberAfRssiLocationClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                     int16u indexOrDestination,
                                     EmberApsFrame * apsFrame,
                                     int16u msgLen,
```

```
                                        int8u * message,
                                        EmberStatus status);

// @{

void emberAfBinaryInputBasicClusterServerInitCallback(int8u endpoint);
void emberAfBinaryInputBasicClusterClientInitCallback(int8u endpoint);
void emberAfBinaryInputBasicClusterServerTickCallback(int8u endpoint);
void emberAfBinaryInputBasicClusterClientTickCallback(int8u endpoint);
void emberAfBinaryInputBasicClusterServerAttributeChangedCallback(int8u endpoint,
                                                  EmberAfAttributeId attributeId);
void emberAfBinaryInputBasicClusterClientAttributeChangedCallback(int8u endpoint,
                                                  EmberAfAttributeId attributeId);
void emberAfBinaryInputBasicClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                          EmberAfAttributeId attributeId,
                                                          int16u manufacturerCode);
void emberAfBinaryInputBasicClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                          EmberAfAttributeId attributeId,
                                                          int16u manufacturerCode);
void emberAfBinaryInputBasicClusterServerDefaultResponseCallback(int8u endpoint,
                                                 int8u commandId,
                                                 EmberAfStatus status);
void emberAfBinaryInputBasicClusterClientDefaultResponseCallback(int8u endpoint,
                                                 int8u commandId,
                                                 EmberAfStatus status);
void emberAfBinaryInputBasicClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                             int16u indexOrDestination,
                                             EmberApsFrame * apsFrame,
                                             int16u msgLen,
                                             int8u * message,
                                             EmberStatus status);
void emberAfBinaryInputBasicClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                             int16u indexOrDestination,
                                             EmberApsFrame * apsFrame,
                                             int16u msgLen,
                                             int8u * message,
                                             EmberStatus status);

// @{

boolean emberAfCommissioningClusterRestartDeviceCallback(int8u options,
                                        int8u delay,
                                        int8u jitter);
boolean emberAfCommissioningClusterSaveStartupParametersCallback(int8u options,
                                                 int8u index);
boolean emberAfCommissioningClusterRestoreStartupParametersCallback(int8u options,
                                                  int8u index);
boolean emberAfCommissioningClusterResetStartupParametersCallback(int8u options,
                                                 int8u index);
boolean emberAfCommissioningClusterRestartDeviceResponseCallback(int8u status);
boolean emberAfCommissioningClusterSaveStartupParametersResponseCallback(int8u status);
boolean emberAfCommissioningClusterRestoreStartupParametersResponseCallback(int8u status);
boolean emberAfCommissioningClusterResetStartupParametersResponseCallback(int8u status);
void emberAfCommissioningClusterServerInitCallback(int8u endpoint);
void emberAfCommissioningClusterClientInitCallback(int8u endpoint);
void emberAfCommissioningClusterServerTickCallback(int8u endpoint);
```

```
void emberAfCommissioningClusterClientTickCallback(int8u endpoint);
void emberAfCommissioningClusterServerAttributeChangedCallback(int8u endpoint,
                                                               EmberAfAttributeId attributeId);
void emberAfCommissioningClusterClientAttributeChangedCallback(int8u endpoint,
                                                               EmberAfAttributeId attributeId);
void emberAfCommissioningClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                                                   EmberAfAttributeId attributeId,
                                                                                   int16u manufacturerCode);
void emberAfCommissioningClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                                                   EmberAfAttributeId attributeId,
                                                                                   int16u manufacturerCode);
void emberAfCommissioningClusterServerDefaultResponseCallback(int8u endpoint,
                                                              int8u commandId,
                                                              EmberAfStatus status);
void emberAfCommissioningClusterClientDefaultResponseCallback(int8u endpoint,
                                                              int8u commandId,
                                                              EmberAfStatus status);
void emberAfCommissioningClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                                          int16u indexOrDestination,
                                                          EmberApsFrame * apsFrame,
                                                          int16u msgLen,
                                                          int8u * message,
                                                          EmberStatus status);
void emberAfCommissioningClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                                          int16u indexOrDestination,
                                                          EmberApsFrame * apsFrame,
                                                          int16u msgLen,
                                                          int8u * message,
                                                          EmberStatus status);

// @{

void emberAfShadeConfigClusterServerInitCallback(int8u endpoint);
void emberAfShadeConfigClusterClientInitCallback(int8u endpoint);
void emberAfShadeConfigClusterServerTickCallback(int8u endpoint);
void emberAfShadeConfigClusterClientTickCallback(int8u endpoint);
void emberAfShadeConfigClusterServerAttributeChangedCallback(int8u endpoint,
                                                             EmberAfAttributeId attributeId);
void emberAfShadeConfigClusterClientAttributeChangedCallback(int8u endpoint,
                                                             EmberAfAttributeId attributeId);
void emberAfShadeConfigClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                                                 EmberAfAttributeId attributeId,
                                                                                 int16u manufacturerCode);
void emberAfShadeConfigClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                                                 EmberAfAttributeId attributeId,
                                                                                 int16u manufacturerCode);
void emberAfShadeConfigClusterServerDefaultResponseCallback(int8u endpoint,
                                                            int8u commandId,
                                                            EmberAfStatus status);
void emberAfShadeConfigClusterClientDefaultResponseCallback(int8u endpoint,
                                                            int8u commandId,
                                                            EmberAfStatus status);
void emberAfShadeConfigClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                                        int16u indexOrDestination,
                                                        EmberApsFrame * apsFrame,
                                                        int16u msgLen,
```

```
                                        int8u * message,
                                        EmberStatus status);
void emberAfShadeConfigClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                        int16u indexOrDestination,
                                        EmberApsFrame * apsFrame,
                                        int16u msgLen,
                                        int8u * message,
                                        EmberStatus status);

// @{

void emberAfPumpConfigControlClusterServerInitCallback(int8u endpoint);
void emberAfPumpConfigControlClusterClientInitCallback(int8u endpoint);
void emberAfPumpConfigControlClusterServerTickCallback(int8u endpoint);
void emberAfPumpConfigControlClusterClientTickCallback(int8u endpoint);
void emberAfPumpConfigControlClusterServerAttributeChangedCallback(int8u endpoint,
                                            EmberAfAttributeId attributeId);
void emberAfPumpConfigControlClusterClientAttributeChangedCallback(int8u endpoint,
                                            EmberAfAttributeId attributeId);
void emberAfPumpConfigControlClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                        EmberAfAttributeId attributeId,
                                                        int16u manufacturerCode);
void emberAfPumpConfigControlClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                        EmberAfAttributeId attributeId,
                                                        int16u manufacturerCode);
void emberAfPumpConfigControlClusterServerDefaultResponseCallback(int8u endpoint,
                                            int8u commandId,
                                            EmberAfStatus status);
void emberAfPumpConfigControlClusterClientDefaultResponseCallback(int8u endpoint,
                                            int8u commandId,
                                            EmberAfStatus status);
void emberAfPumpConfigControlClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                        int16u indexOrDestination,
                                        EmberApsFrame * apsFrame,
                                        int16u msgLen,
                                        int8u * message,
                                        EmberStatus status);
void emberAfPumpConfigControlClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                        int16u indexOrDestination,
                                        EmberApsFrame * apsFrame,
                                        int16u msgLen,
                                        int8u * message,
                                        EmberStatus status);

// @{

boolean emberAfThermostatClusterSetpointRaiseLowerCallback(int8u mode,
                                        int8s amount);
void emberAfThermostatClusterServerInitCallback(int8u endpoint);
void emberAfThermostatClusterClientInitCallback(int8u endpoint);
void emberAfThermostatClusterServerTickCallback(int8u endpoint);
void emberAfThermostatClusterClientTickCallback(int8u endpoint);
void emberAfThermostatClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfThermostatClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
```

```
03956 void emberAfThermostatClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
03957                                                                                 EmberAfAttributeId attributeId,
03958                                                                                 int16u manufacturerCode);
03968 void emberAfThermostatClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
03969                                                                                 EmberAfAttributeId attributeId,
03970                                                                                 int16u manufacturerCode);
03979 void emberAfThermostatClusterServerDefaultResponseCallback(int8u endpoint,
03980                                                           int8u commandId,
03981                                                           EmberAfStatus status);
03990 void emberAfThermostatClusterClientDefaultResponseCallback(int8u endpoint,
03991                                                           int8u commandId,
03992                                                           EmberAfStatus status);
04005 void emberAfThermostatClusterServerMessageSentCallback(EmberOutgoingMessageType type,
04006                                                       int16u indexOrDestination,
04007                                                       EmberApsFrame * apsFrame,
04008                                                       int16u msgLen,
04009                                                       int8u * message,
04010                                                       EmberStatus status);
04023 void emberAfThermostatClusterClientMessageSentCallback(EmberOutgoingMessageType type,
04024                                                       int16u indexOrDestination,
04025                                                       EmberApsFrame * apsFrame,
04026                                                       int16u msgLen,
04027                                                       int8u * message,
04028                                                       EmberStatus status);
04029
04033 // @{
04034
04041 void emberAfFanControlClusterServerInitCallback(int8u endpoint);
04048 void emberAfFanControlClusterClientInitCallback(int8u endpoint);
04055 void emberAfFanControlClusterServerTickCallback(int8u endpoint);
04062 void emberAfFanControlClusterClientTickCallback(int8u endpoint);
04070 void emberAfFanControlClusterServerAttributeChangedCallback(int8u endpoint,
04071                                                            EmberAfAttributeId attributeId);
04079 void emberAfFanControlClusterClientAttributeChangedCallback(int8u endpoint,
04080                                                            EmberAfAttributeId attributeId);
04090 void emberAfFanControlClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
04091                                                                                 EmberAfAttributeId attributeId,
04092                                                                                 int16u manufacturerCode);
04102 void emberAfFanControlClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
04103                                                                                 EmberAfAttributeId attributeId,
04104                                                                                 int16u manufacturerCode);
04113 void emberAfFanControlClusterServerDefaultResponseCallback(int8u endpoint,
04114                                                           int8u commandId,
04115                                                           EmberAfStatus status);
04124 void emberAfFanControlClusterClientDefaultResponseCallback(int8u endpoint,
04125                                                           int8u commandId,
04126                                                           EmberAfStatus status);
04139 void emberAfFanControlClusterServerMessageSentCallback(EmberOutgoingMessageType type,
04140                                                       int16u indexOrDestination,
04141                                                       EmberApsFrame * apsFrame,
04142                                                       int16u msgLen,
04143                                                       int8u * message,
04144                                                       EmberStatus status);
04157 void emberAfFanControlClusterClientMessageSentCallback(EmberOutgoingMessageType type,
04158                                                       int16u indexOrDestination,
04159                                                       EmberApsFrame * apsFrame,
```

```
                                        int16u msgLen,
                                        int8u * message,
                                        EmberStatus status);

// @{

void emberAfDehumidControlClusterServerInitCallback(int8u endpoint);
void emberAfDehumidControlClusterClientInitCallback(int8u endpoint);
void emberAfDehumidControlClusterServerTickCallback(int8u endpoint);
void emberAfDehumidControlClusterClientTickCallback(int8u endpoint);
void emberAfDehumidControlClusterServerAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId);
void emberAfDehumidControlClusterClientAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId);
void emberAfDehumidControlClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                        EmberAfAttributeId attributeId,
                                                        int16u manufacturerCode);
void emberAfDehumidControlClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                        EmberAfAttributeId attributeId,
                                                        int16u manufacturerCode);
void emberAfDehumidControlClusterServerDefaultResponseCallback(int8u endpoint,
                                                int8u commandId,
                                                EmberAfStatus status);
void emberAfDehumidControlClusterClientDefaultResponseCallback(int8u endpoint,
                                                int8u commandId,
                                                EmberAfStatus status);
void emberAfDehumidControlClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                            int16u indexOrDestination,
                                            EmberApsFrame * apsFrame,
                                            int16u msgLen,
                                            int8u * message,
                                            EmberStatus status);
void emberAfDehumidControlClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                            int16u indexOrDestination,
                                            EmberApsFrame * apsFrame,
                                            int16u msgLen,
                                            int8u * message,
                                            EmberStatus status);

// @{

void emberAfThermostatUiConfigClusterServerInitCallback(int8u endpoint);
void emberAfThermostatUiConfigClusterClientInitCallback(int8u endpoint);
void emberAfThermostatUiConfigClusterServerTickCallback(int8u endpoint);
void emberAfThermostatUiConfigClusterClientTickCallback(int8u endpoint);
void emberAfThermostatUiConfigClusterServerAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId);
void emberAfThermostatUiConfigClusterClientAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId);
void emberAfThermostatUiConfigClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                        EmberAfAttributeId attributeId,
                                                        int16u manufacturerCode);
void emberAfThermostatUiConfigClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                        EmberAfAttributeId attributeId,
                                                        int16u manufacturerCode);
void emberAfThermostatUiConfigClusterServerDefaultResponseCallback(int8u endpoint,
```

```
                                                int8u commandId,
                                                EmberAfStatus status);
void emberAfThermostatUiConfigClusterClientDefaultResponseCallback(int8u endpoint,
                                                int8u commandId,
                                                EmberAfStatus status);
void emberAfThermostatUiConfigClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                                int16u indexOrDestination,
                                                EmberApsFrame * apsFrame,
                                                int16u msgLen,
                                                int8u * message,
                                                EmberStatus status);
void emberAfThermostatUiConfigClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                                int16u indexOrDestination,
                                                EmberApsFrame * apsFrame,
                                                int16u msgLen,
                                                int8u * message,
                                                EmberStatus status);

// @{

boolean emberAfColorControlClusterMoveToHueCallback(int8u hue,
                                                int8u direction,
                                                int16u transitionTime);
boolean emberAfColorControlClusterMoveHueCallback(int8u moveMode,
                                                int8u rate);
boolean emberAfColorControlClusterStepHueCallback(int8u stepMode,
                                                int8u stepSize,
                                                int8u transitionTime);
boolean emberAfColorControlClusterMoveToSaturationCallback(int8u saturation,
                                                int16u transitionTime);
boolean emberAfColorControlClusterMoveSaturationCallback(int8u moveMode,
                                                int8u rate);
boolean emberAfColorControlClusterStepSaturationCallback(int8u stepMode,
                                                int8u stepSize,
                                                int8u transitionTime);
boolean emberAfColorControlClusterMoveToHueAndSaturationCallback(int8u hue,
                                                int8u saturation,
                                                int16u transitionTime);
boolean emberAfColorControlClusterMoveToColorCallback(int16u colorX,
                                                int16u colorY,
                                                int16u transitionTime);
boolean emberAfColorControlClusterMoveColorCallback(int16s rateX,
                                                int16s rateY);
boolean emberAfColorControlClusterStepColorCallback(int16s stepX,
                                                int16s stepY,
                                                int16u transitionTime);
boolean emberAfColorControlClusterMoveToColorTemperatureCallback(int16u colorTemperature,
                                                int16u transitionTime);
void emberAfColorControlClusterServerInitCallback(int8u endpoint);
void emberAfColorControlClusterClientInitCallback(int8u endpoint);
void emberAfColorControlClusterServerTickCallback(int8u endpoint);
void emberAfColorControlClusterClientTickCallback(int8u endpoint);
void emberAfColorControlClusterServerAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId);
void emberAfColorControlClusterClientAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId);
void emberAfColorControlClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
```

```
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfColorControlClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfColorControlClusterServerDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfColorControlClusterClientDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfColorControlClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                    int16u indexOrDestination,
                                    EmberApsFrame * apsFrame,
                                    int16u msgLen,
                                    int8u * message,
                                    EmberStatus status);
void emberAfColorControlClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                    int16u indexOrDestination,
                                    EmberApsFrame * apsFrame,
                                    int16u msgLen,
                                    int8u * message,
                                    EmberStatus status);

// @{

void emberAfBallastConfigurationClusterServerInitCallback(int8u endpoint);
void emberAfBallastConfigurationClusterClientInitCallback(int8u endpoint);
void emberAfBallastConfigurationClusterServerTickCallback(int8u endpoint);
void emberAfBallastConfigurationClusterClientTickCallback(int8u endpoint);
void emberAfBallastConfigurationClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfBallastConfigurationClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfBallastConfigurationClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfBallastConfigurationClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfBallastConfigurationClusterServerDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfBallastConfigurationClusterClientDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfBallastConfigurationClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                    int16u indexOrDestination,
                                    EmberApsFrame * apsFrame,
                                    int16u msgLen,
                                    int8u * message,
                                    EmberStatus status);
void emberAfBallastConfigurationClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                    int16u indexOrDestination,
                                    EmberApsFrame * apsFrame,
                                    int16u msgLen,
                                    int8u * message,
```

```
                                        EmberStatus status);

// @{

void emberAfIllumMeasurementClusterServerInitCallback(int8u endpoint);
void emberAfIllumMeasurementClusterClientInitCallback(int8u endpoint);
void emberAfIllumMeasurementClusterServerTickCallback(int8u endpoint);
void emberAfIllumMeasurementClusterClientTickCallback(int8u endpoint);
void emberAfIllumMeasurementClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfIllumMeasurementClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfIllumMeasurementClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfIllumMeasurementClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfIllumMeasurementClusterServerDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfIllumMeasurementClusterClientDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfIllumMeasurementClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                       int16u indexOrDestination,
                                       EmberApsFrame * apsFrame,
                                       int16u msgLen,
                                       int8u * message,
                                       EmberStatus status);
void emberAfIllumMeasurementClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                       int16u indexOrDestination,
                                       EmberApsFrame * apsFrame,
                                       int16u msgLen,
                                       int8u * message,
                                       EmberStatus status);

// @{

void emberAfIllumLevelSensingClusterServerInitCallback(int8u endpoint);
void emberAfIllumLevelSensingClusterClientInitCallback(int8u endpoint);
void emberAfIllumLevelSensingClusterServerTickCallback(int8u endpoint);
void emberAfIllumLevelSensingClusterClientTickCallback(int8u endpoint);
void emberAfIllumLevelSensingClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfIllumLevelSensingClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfIllumLevelSensingClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfIllumLevelSensingClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfIllumLevelSensingClusterServerDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
```

```
05038 void emberAfIllumLevelSensingClusterClientDefaultResponseCallback(int8u endpoint,
05039                                                                   int8u commandId,
05040                                                                   EmberAfStatus status);
05053 void emberAfIllumLevelSensingClusterServerMessageSentCallback(EmberOutgoingMessageType type,
05054                                                               int16u indexOrDestination,
05055                                                               EmberApsFrame * apsFrame,
05056                                                               int16u msgLen,
05057                                                               int8u * message,
05058                                                               EmberStatus status);
05071 void emberAfIllumLevelSensingClusterClientMessageSentCallback(EmberOutgoingMessageType type,
05072                                                               int16u indexOrDestination,
05073                                                               EmberApsFrame * apsFrame,
05074                                                               int16u msgLen,
05075                                                               int8u * message,
05076                                                               EmberStatus status);
05077
05081 // @{
05082
05089 int32s emberAfTempMeasurementClusterConvertVoltsToCelsiusCallback(int16u voltage);
05096 void emberAfTempMeasurementClusterServerInitCallback(int8u endpoint);
05103 void emberAfTempMeasurementClusterClientInitCallback(int8u endpoint);
05110 void emberAfTempMeasurementClusterServerTickCallback(int8u endpoint);
05117 void emberAfTempMeasurementClusterClientTickCallback(int8u endpoint);
05125 void emberAfTempMeasurementClusterServerAttributeChangedCallback(int8u endpoint,
05126                                                                  EmberAfAttributeId attributeId);
05134 void emberAfTempMeasurementClusterClientAttributeChangedCallback(int8u endpoint,
05135                                                                  EmberAfAttributeId attributeId);
05146 void emberAfTempMeasurementClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
05147                                                                                      EmberAfAttributeId attributeId,
05148                                                                                      int16u manufacturerCode);
05159 void emberAfTempMeasurementClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
05160                                                                                      EmberAfAttributeId attributeId,
05161                                                                                      int16u manufacturerCode);
05170 void emberAfTempMeasurementClusterServerDefaultResponseCallback(int8u endpoint,
05171                                                                 int8u commandId,
05172                                                                 EmberAfStatus status);
05181 void emberAfTempMeasurementClusterClientDefaultResponseCallback(int8u endpoint,
05182                                                                 int8u commandId,
05183                                                                 EmberAfStatus status);
05196 void emberAfTempMeasurementClusterServerMessageSentCallback(EmberOutgoingMessageType type,
05197                                                             int16u indexOrDestination,
05198                                                             EmberApsFrame * apsFrame,
05199                                                             int16u msgLen,
05200                                                             int8u * message,
05201                                                             EmberStatus status);
05214 void emberAfTempMeasurementClusterClientMessageSentCallback(EmberOutgoingMessageType type,
05215                                                             int16u indexOrDestination,
05216                                                             EmberApsFrame * apsFrame,
05217                                                             int16u msgLen,
05218                                                             int8u * message,
05219                                                             EmberStatus status);
05220
05224 // @{
05225
05232 void emberAfPressureMeasurementClusterServerInitCallback(int8u endpoint);
05239 void emberAfPressureMeasurementClusterClientInitCallback(int8u endpoint);
05246 void emberAfPressureMeasurementClusterServerTickCallback(int8u endpoint);
```

```
05253 void emberAfPressureMeasurementClusterClientTickCallback(int8u endpoint);
05261 void emberAfPressureMeasurementClusterServerAttributeChangedCallback(int8u endpoint,
05262                                                                      EmberAfAttributeId attributeId);
05270 void emberAfPressureMeasurementClusterClientAttributeChangedCallback(int8u endpoint,
05271                                                                      EmberAfAttributeId attributeId);
05282 void emberAfPressureMeasurementClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
05283                                                                                          EmberAfAttributeId attributeId,
05284                                                                                          int16u manufacturerCode);
05295 void emberAfPressureMeasurementClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
05296                                                                                          EmberAfAttributeId attributeId,
05297                                                                                          int16u manufacturerCode);
05306 void emberAfPressureMeasurementClusterServerDefaultResponseCallback(int8u endpoint,
05307                                                                     int8u commandId,
05308                                                                     EmberAfStatus status);
05317 void emberAfPressureMeasurementClusterClientDefaultResponseCallback(int8u endpoint,
05318                                                                     int8u commandId,
05319                                                                     EmberAfStatus status);
05332 void emberAfPressureMeasurementClusterServerMessageSentCallback(EmberOutgoingMessageType type,
05333                                                                int16u indexOrDestination,
05334                                                                EmberApsFrame * apsFrame,
05335                                                                int16u msgLen,
05336                                                                int8u * message,
05337                                                                EmberStatus status);
05350 void emberAfPressureMeasurementClusterClientMessageSentCallback(EmberOutgoingMessageType type,
05351                                                                int16u indexOrDestination,
05352                                                                EmberApsFrame * apsFrame,
05353                                                                int16u msgLen,
05354                                                                int8u * message,
05355                                                                EmberStatus status);
05356
05360 // @{
05361
05368 void emberAfFlowMeasurementClusterServerInitCallback(int8u endpoint);
05375 void emberAfFlowMeasurementClusterClientInitCallback(int8u endpoint);
05382 void emberAfFlowMeasurementClusterServerTickCallback(int8u endpoint);
05389 void emberAfFlowMeasurementClusterClientTickCallback(int8u endpoint);
05397 void emberAfFlowMeasurementClusterServerAttributeChangedCallback(int8u endpoint,
05398                                                                  EmberAfAttributeId attributeId);
05406 void emberAfFlowMeasurementClusterClientAttributeChangedCallback(int8u endpoint,
05407                                                                  EmberAfAttributeId attributeId);
05417 void emberAfFlowMeasurementClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
05418                                                                                      EmberAfAttributeId attributeId,
05419                                                                                      int16u manufacturerCode);
05429 void emberAfFlowMeasurementClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
05430                                                                                      EmberAfAttributeId attributeId,
05431                                                                                      int16u manufacturerCode);
05440 void emberAfFlowMeasurementClusterServerDefaultResponseCallback(int8u endpoint,
05441                                                                int8u commandId,
05442                                                                EmberAfStatus status);
05451 void emberAfFlowMeasurementClusterClientDefaultResponseCallback(int8u endpoint,
05452                                                                int8u commandId,
05453                                                                EmberAfStatus status);
05466 void emberAfFlowMeasurementClusterServerMessageSentCallback(EmberOutgoingMessageType type,
05467                                                            int16u indexOrDestination,
05468                                                            EmberApsFrame * apsFrame,
05469                                                            int16u msgLen,
```

```
                                        int8u * message,
                                        EmberStatus status);
void emberAfFlowMeasurementClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                        int16u indexOrDestination,
                                        EmberApsFrame * apsFrame,
                                        int16u msgLen,
                                        int8u * message,
                                        EmberStatus status);

// @{

void emberAfRelativeHumidityMeasurementClusterServerInitCallback(int8u endpoint);
void emberAfRelativeHumidityMeasurementClusterClientInitCallback(int8u endpoint);
void emberAfRelativeHumidityMeasurementClusterServerTickCallback(int8u endpoint);
void emberAfRelativeHumidityMeasurementClusterClientTickCallback(int8u endpoint);
void emberAfRelativeHumidityMeasurementClusterServerAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId);
void emberAfRelativeHumidityMeasurementClusterClientAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId);
void emberAfRelativeHumidityMeasurementClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                        EmberAfAttributeId attributeId,
                                                        int16u manufacturerCode);
void emberAfRelativeHumidityMeasurementClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                        EmberAfAttributeId attributeId,
                                                        int16u manufacturerCode);
void emberAfRelativeHumidityMeasurementClusterServerDefaultResponseCallback(int8u endpoint,
                                                int8u commandId,
                                                EmberAfStatus status);
void emberAfRelativeHumidityMeasurementClusterClientDefaultResponseCallback(int8u endpoint,
                                                int8u commandId,
                                                EmberAfStatus status);
void emberAfRelativeHumidityMeasurementClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                                int16u indexOrDestination,
                                                EmberApsFrame * apsFrame,
                                                int16u msgLen,
                                                int8u * message,
                                                EmberStatus status);
void emberAfRelativeHumidityMeasurementClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                                int16u indexOrDestination,
                                                EmberApsFrame * apsFrame,
                                                int16u msgLen,
                                                int8u * message,
                                                EmberStatus status);

// @{

void emberAfOccupancySensingClusterServerInitCallback(int8u endpoint);
void emberAfOccupancySensingClusterClientInitCallback(int8u endpoint);
void emberAfOccupancySensingClusterServerTickCallback(int8u endpoint);
void emberAfOccupancySensingClusterClientTickCallback(int8u endpoint);
void emberAfOccupancySensingClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfOccupancySensingClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfOccupancySensingClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
```

```
                                                    int16u manufacturerCode);
void emberAfOccupancySensingClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                    EmberAfAttributeId attributeId,
                                                    int16u manufacturerCode);
void emberAfOccupancySensingClusterServerDefaultResponseCallback(int8u endpoint,
                                                int8u commandId,
                                                EmberAfStatus status);
void emberAfOccupancySensingClusterClientDefaultResponseCallback(int8u endpoint,
                                                int8u commandId,
                                                EmberAfStatus status);
void emberAfOccupancySensingClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                              int16u indexOrDestination,
                                              EmberApsFrame * apsFrame,
                                              int16u msgLen,
                                              int8u * message,
                                              EmberStatus status);
void emberAfOccupancySensingClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                              int16u indexOrDestination,
                                              EmberApsFrame * apsFrame,
                                              int16u msgLen,
                                              int8u * message,
                                              EmberStatus status);

// @{

boolean emberAfIasZoneClusterZoneEnrollResponseCallback(int8u enrollResponseCode,
                                            int8u zoneId);
boolean emberAfIasZoneClusterZoneStatusChangeNotificationCallback(int16u zoneStatus,
                                                int8u extendedStatus);
boolean emberAfIasZoneClusterZoneEnrollRequestCallback(int16u zoneType,
                                           int16u manufacturerCode);
void emberAfIasZoneClusterServerInitCallback(int8u endpoint);
void emberAfIasZoneClusterClientInitCallback(int8u endpoint);
void emberAfIasZoneClusterServerTickCallback(int8u endpoint);
void emberAfIasZoneClusterClientTickCallback(int8u endpoint);
void emberAfIasZoneClusterServerAttributeChangedCallback(int8u endpoint,
                                             EmberAfAttributeId attributeId);
void emberAfIasZoneClusterClientAttributeChangedCallback(int8u endpoint,
                                             EmberAfAttributeId attributeId);
void emberAfIasZoneClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                     EmberAfAttributeId attributeId,
                                                     int16u manufacturerCode);
void emberAfIasZoneClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                     EmberAfAttributeId attributeId,
                                                     int16u manufacturerCode);
void emberAfIasZoneClusterServerDefaultResponseCallback(int8u endpoint,
                                            int8u commandId,
                                            EmberAfStatus status);
void emberAfIasZoneClusterClientDefaultResponseCallback(int8u endpoint,
                                            int8u commandId,
                                            EmberAfStatus status);
void emberAfIasZoneClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                           int16u indexOrDestination,
                                           EmberApsFrame * apsFrame,
                                           int16u msgLen,
                                           int8u * message,
```

```
                                        EmberStatus status);
void emberAfIasZoneClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                        int16u indexOrDestination,
                                        EmberApsFrame * apsFrame,
                                        int16u msgLen,
                                        int8u * message,
                                        EmberStatus status);

// @{

boolean emberAfIasAceClusterArmCallback(int8u armMode);
boolean emberAfIasAceClusterBypassCallback(int8u numberOfZones,
                                    int8u* zoneIds);
boolean emberAfIasAceClusterEmergencyCallback(void);
boolean emberAfIasAceClusterFireCallback(void);
boolean emberAfIasAceClusterPanicCallback(void);
boolean emberAfIasAceClusterGetZoneIdMapCallback(void);
boolean emberAfIasAceClusterGetZoneInformationCallback(int8u zoneId);
boolean emberAfIasAceClusterArmResponseCallback(int8u armNotification);
boolean emberAfIasAceClusterGetZoneIdMapResponseCallback(int16u section0,
                                        int16u section1,
                                        int16u section2,
                                        int16u section3,
                                        int16u section4,
                                        int16u section5,
                                        int16u section6,
                                        int16u section7,
                                        int16u section8,
                                        int16u section9,
                                        int16u section10,
                                        int16u section11,
                                        int16u section12,
                                        int16u section13,
                                        int16u section14,
                                        int16u section15);
boolean emberAfIasAceClusterGetZoneInformationResponseCallback(int8u zoneId,
                                            int16u zoneType,
                                            int8u* ieeeAddress);
void emberAfIasAceClusterServerInitCallback(int8u endpoint);
void emberAfIasAceClusterClientInitCallback(int8u endpoint);
void emberAfIasAceClusterServerTickCallback(int8u endpoint);
void emberAfIasAceClusterClientTickCallback(int8u endpoint);
void emberAfIasAceClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfIasAceClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfIasAceClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfIasAceClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfIasAceClusterServerDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfIasAceClusterClientDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
```

```
06107                                   EmberAfStatus status);
06120 void emberAfIasAceClusterServerMessageSentCallback(EmberOutgoingMessageType type,
06121                                   int16u indexOrDestination,
06122                                   EmberApsFrame * apsFrame,
06123                                   int16u msgLen,
06124                                   int8u * message,
06125                                   EmberStatus status);
06138 void emberAfIasAceClusterClientMessageSentCallback(EmberOutgoingMessageType type,
06139                                   int16u indexOrDestination,
06140                                   EmberApsFrame * apsFrame,
06141                                   int16u msgLen,
06142                                   int8u * message,
06143                                   EmberStatus status);
06144
06148 // @{
06149
06156 boolean emberAfIasWdClusterStartWarningCallback(int8u warningInfo,
06157                                 int16u warningDuration);
06163 boolean emberAfIasWdClusterSquawkCallback(int8u squawkInfo);
06170 void emberAfIasWdClusterServerInitCallback(int8u endpoint);
06177 void emberAfIasWdClusterClientInitCallback(int8u endpoint);
06184 void emberAfIasWdClusterServerTickCallback(int8u endpoint);
06191 void emberAfIasWdClusterClientTickCallback(int8u endpoint);
06199 void emberAfIasWdClusterServerAttributeChangedCallback(int8u endpoint,
06200                                   EmberAfAttributeId attributeId);
06208 void emberAfIasWdClusterClientAttributeChangedCallback(int8u endpoint,
06209                                   EmberAfAttributeId attributeId);
06219 void emberAfIasWdClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
06220                                           EmberAfAttributeId attributeId,
06221                                           int16u manufacturerCode);
06231 void emberAfIasWdClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
06232                                           EmberAfAttributeId attributeId,
06233                                           int16u manufacturerCode);
06242 void emberAfIasWdClusterServerDefaultResponseCallback(int8u endpoint,
06243                                   int8u commandId,
06244                                   EmberAfStatus status);
06253 void emberAfIasWdClusterClientDefaultResponseCallback(int8u endpoint,
06254                                   int8u commandId,
06255                                   EmberAfStatus status);
06268 void emberAfIasWdClusterServerMessageSentCallback(EmberOutgoingMessageType type,
06269                                   int16u indexOrDestination,
06270                                   EmberApsFrame * apsFrame,
06271                                   int16u msgLen,
06272                                   int8u * message,
06273                                   EmberStatus status);
06286 void emberAfIasWdClusterClientMessageSentCallback(EmberOutgoingMessageType type,
06287                                   int16u indexOrDestination,
06288                                   EmberApsFrame * apsFrame,
06289                                   int16u msgLen,
06290                                   int8u * message,
06291                                   EmberStatus status);
06292
06296 // @{
06297
06302 boolean emberAfDoorLockClusterLockDoorCallback(void);
06307 boolean emberAfDoorLockClusterUnlockDoorCallback(void);
06313 boolean emberAfDoorLockClusterLockDoorResponseCallback(int8u status);
```

```
06319 boolean emberAfDoorLockClusterUnlockDoorResponseCallback(int8u status);
06326 void emberAfDoorLockClusterServerInitCallback(int8u endpoint);
06333 void emberAfDoorLockClusterClientInitCallback(int8u endpoint);
06340 void emberAfDoorLockClusterServerTickCallback(int8u endpoint);
06347 void emberAfDoorLockClusterClientTickCallback(int8u endpoint);
06355 void emberAfDoorLockClusterServerAttributeChangedCallback(int8u endpoint,
06356                                                          EmberAfAttributeId attributeId);
06364 void emberAfDoorLockClusterClientAttributeChangedCallback(int8u endpoint,
06365                                                          EmberAfAttributeId attributeId);
06375 void emberAfDoorLockClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
06376                                                                               EmberAfAttributeId attributeId,
06377                                                                               int16u manufacturerCode);
06387 void emberAfDoorLockClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
06388                                                                               EmberAfAttributeId attributeId,
06389                                                                               int16u manufacturerCode);
06398 void emberAfDoorLockClusterServerDefaultResponseCallback(int8u endpoint,
06399                                                         int8u commandId,
06400                                                         EmberAfStatus status);
06409 void emberAfDoorLockClusterClientDefaultResponseCallback(int8u endpoint,
06410                                                         int8u commandId,
06411                                                         EmberAfStatus status);
06424 void emberAfDoorLockClusterServerMessageSentCallback(EmberOutgoingMessageType type,
06425                                                     int16u indexOrDestination,
06426                                                     EmberApsFrame * apsFrame,
06427                                                     int16u msgLen,
06428                                                     int8u * message,
06429                                                     EmberStatus status);
06442 void emberAfDoorLockClusterClientMessageSentCallback(EmberOutgoingMessageType type,
06443                                                     int16u indexOrDestination,
06444                                                     EmberApsFrame * apsFrame,
06445                                                     int16u msgLen,
06446                                                     int8u * message,
06447                                                     EmberStatus status);
06448
06452 // @{
06453
06458 boolean emberAfWindowCoveringClusterWindowCoveringUpOpenCallback(void);
06463 boolean emberAfWindowCoveringClusterWindowCoveringDownCloseCallback(void);
06468 boolean emberAfWindowCoveringClusterWindowCoveringStopCallback(void);
06474 boolean emberAfWindowCoveringClusterWindowCoveringGoToLiftSetpointCallback(int8u indexOfLiftSetpoint);
06480 boolean emberAfWindowCoveringClusterWindowCoveringGoToLiftValueCallback(int16u liftValue);
06486 boolean emberAfWindowCoveringClusterWindowCoveringGoToLiftPercentageCallback(int8u percentageLiftValue);
06492 boolean emberAfWindowCoveringClusterWindowCoveringGoToTiltSetpointCallback(int8u indexOfTiltSetpoint);
06498 boolean emberAfWindowCoveringClusterWindowCoveringGoToTiltValueCallback(int16u tiltValue);
06504 boolean emberAfWindowCoveringClusterWindowCoveringGoToTiltPercentageCallback(int8u percentageTiltValue);
06512 boolean emberAfWindowCoveringClusterWindowCoveringGoToProgramSetpointCallback(int8u setpointType,
06513                                                                               int8u setpointIndex,
06514                                                                               int16u setpointValue);
06521 void emberAfWindowCoveringClusterServerInitCallback(int8u endpoint);
06528 void emberAfWindowCoveringClusterClientInitCallback(int8u endpoint);
06535 void emberAfWindowCoveringClusterServerTickCallback(int8u endpoint);
06542 void emberAfWindowCoveringClusterClientTickCallback(int8u endpoint);
06550 void emberAfWindowCoveringClusterServerAttributeChangedCallback(int8u endpoint,
06551                                                               EmberAfAttributeId attributeId);
06559 void emberAfWindowCoveringClusterClientAttributeChangedCallback(int8u endpoint,
```

```
                                        EmberAfAttributeId attributeId);
void emberAfWindowCoveringClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfWindowCoveringClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfWindowCoveringClusterServerDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfWindowCoveringClusterClientDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfWindowCoveringClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                      int16u indexOrDestination,
                                      EmberApsFrame * apsFrame,
                                      int16u msgLen,
                                      int8u * message,
                                      EmberStatus status);
void emberAfWindowCoveringClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                      int16u indexOrDestination,
                                      EmberApsFrame * apsFrame,
                                      int16u msgLen,
                                      int8u * message,
                                      EmberStatus status);

// @{

boolean emberAfGenericTunnelClusterMatchProtocolAddressCallback(int8u* protocolAddress);
boolean emberAfGenericTunnelClusterMatchProtocolAddressResponseCallback(void);
boolean emberAfGenericTunnelClusterAdvertiseProtocolAddressCallback(int8u* protocolAddress);
void emberAfGenericTunnelClusterServerInitCallback(int8u endpoint);
void emberAfGenericTunnelClusterClientInitCallback(int8u endpoint);
void emberAfGenericTunnelClusterServerTickCallback(int8u endpoint);
void emberAfGenericTunnelClusterClientTickCallback(int8u endpoint);
void emberAfGenericTunnelClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfGenericTunnelClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfGenericTunnelClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfGenericTunnelClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfGenericTunnelClusterServerDefaultResponseCallback(int8u endpoint,
                                       int8u commandId,
                                       EmberAfStatus status);
void emberAfGenericTunnelClusterClientDefaultResponseCallback(int8u endpoint,
                                       int8u commandId,
                                       EmberAfStatus status);
void emberAfGenericTunnelClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                     int16u indexOrDestination,
                                     EmberApsFrame * apsFrame,
                                     int16u msgLen,
                                     int8u * message,
```

```
                                        EmberStatus status);
void emberAfGenericTunnelClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                        int16u indexOrDestination,
                                        EmberApsFrame * apsFrame,
                                        int16u msgLen,
                                        int8u * message,
                                        EmberStatus status);

// @{

boolean emberAfBacnetProtocolTunnelClusterTransferNpduCallback(int8u* NPDU);
void emberAfBacnetProtocolTunnelClusterServerInitCallback(int8u endpoint);
void emberAfBacnetProtocolTunnelClusterClientInitCallback(int8u endpoint);
void emberAfBacnetProtocolTunnelClusterServerTickCallback(int8u endpoint);
void emberAfBacnetProtocolTunnelClusterClientTickCallback(int8u endpoint);
void emberAfBacnetProtocolTunnelClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfBacnetProtocolTunnelClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfBacnetProtocolTunnelClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfBacnetProtocolTunnelClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfBacnetProtocolTunnelClusterServerDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfBacnetProtocolTunnelClusterClientDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfBacnetProtocolTunnelClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                        int16u indexOrDestination,
                                        EmberApsFrame * apsFrame,
                                        int16u msgLen,
                                        int8u * message,
                                        EmberStatus status);
void emberAfBacnetProtocolTunnelClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                        int16u indexOrDestination,
                                        EmberApsFrame * apsFrame,
                                        int16u msgLen,
                                        int8u * message,
                                        EmberStatus status);

// @{

boolean emberAfOtaBootloadClusterImageNotifyCallback(int8u payloadType,
                                        int8u queryJitter,
                                        int16u manufacturerId,
                                        int16u imageType,
                                        int32u newFileVersion);
boolean emberAfOtaBootloadClusterQueryNextImageRequestCallback(int8u fieldControl,
                                        int16u manufacturerId,
                                        int16u imageType,
                                        int32u currentFileVersion,
                                        int16u hardwareVersion);
boolean emberAfOtaBootloadClusterQueryNextImageResponseCallback(int8u status,
```

```
                                        int16u manufacturerId,
                                        int16u imageType,
                                        int32u fileVersion,
                                        int32u imageSize);
boolean emberAfOtaBootloadClusterImageBlockRequestCallback(int8u fieldControl,
                                       int16u manufacturerId,
                                       int16u imageType,
                                       int32u fileVersion,
                                       int32u fileOffset,
                                       int8u maxDataSize,
                                       int8u* requestNodeAddress);
boolean emberAfOtaBootloadClusterImagePageRequestCallback(int8u fieldControl,
                                      int16u manufacturerId,
                                      int16u imageType,
                                      int32u fileVersion,
                                      int32u fileOffset,
                                      int8u maxDataSize,
                                      int16u pageSize,
                                      int16u responseSpacing,
                                      int8u* requestNodeAddress);
boolean emberAfOtaBootloadClusterImageBlockResponseCallback(int8u status,
                                        int16u manufacturerId,
                                        int16u imageType,
                                        int32u fileVersion,
                                        int32u fileOffset,
                                        int8u dataSize,
                                        int8u* imageData);
boolean emberAfOtaBootloadClusterUpgradeEndRequestCallback(int8u status,
                                       int16u manufacturerId,
                                       int16u imageType,
                                       int32u fileVersion);
boolean emberAfOtaBootloadClusterUpgradeEndResponseCallback(int16u manufacturerId,
                                        int16u imageType,
                                        int32u fileVersion,
                                        int32u currentTime,
                                        int32u upgradeTime);
boolean emberAfOtaBootloadClusterQuerySpecificFileRequestCallback(int8u* requestNodeAddress,
                                          int16u manufacturerId,
                                          int16u imageType,
                                          int32u fileVersion,
                                          int16u currentZigbeeStackVersion);
boolean emberAfOtaBootloadClusterQuerySpecificFileResponseCallback(int8u status,
                                           int16u manufacturerId,
                                           int16u imageType,
                                           int32u fileVersion,
                                           int32u imageSize);
void emberAfOtaBootloadClusterServerInitCallback(int8u endpoint);
void emberAfOtaBootloadClusterClientInitCallback(int8u endpoint);
void emberAfOtaBootloadClusterServerTickCallback(int8u endpoint);
void emberAfOtaBootloadClusterClientTickCallback(int8u endpoint);
void emberAfOtaBootloadClusterServerAttributeChangedCallback(int8u endpoint,
                                         EmberAfAttributeId attributeId);
void emberAfOtaBootloadClusterClientAttributeChangedCallback(int8u endpoint,
                                         EmberAfAttributeId attributeId);
void emberAfOtaBootloadClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                 EmberAfAttributeId attributeId,
                                                 int16u manufacturerCode);
void emberAfOtaBootloadClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
```

```
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfOtaBootloadClusterServerDefaultResponseCallback(int8u endpoint,
                                            int8u commandId,
                                            EmberAfStatus status);
void emberAfOtaBootloadClusterClientDefaultResponseCallback(int8u endpoint,
                                            int8u commandId,
                                            EmberAfStatus status);
void emberAfOtaBootloadClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                        int16u indexOrDestination,
                                        EmberApsFrame * apsFrame,
                                        int16u msgLen,
                                        int8u * message,
                                        EmberStatus status);
void emberAfOtaBootloadClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                        int16u indexOrDestination,
                                        EmberApsFrame * apsFrame,
                                        int16u msgLen,
                                        int8u * message,
                                        EmberStatus status);

// @{

boolean emberAfDemandResponseLoadControlClusterLoadControlEventCallback(int32u issuerEventId,
                                                int16u deviceClass,
                                                int8u utilityEnrollmentGroup,
                                                int32u startTime,
                                                int16u durationInMinutes,
                                                int8u criticalityLevel,
                                                int8u coolingTemperatureOffset,
                                                int8u heatingTemperatureOffset,
                                                int16s coolingTemperatureSetPoint,
                                                int16s heatingTemperatureSetPoint,
                                                int8s averageLoadAdjustmentPercentage,
                                                int8u dutyCycle,
                                                int8u eventControl);
boolean emberAfDemandResponseLoadControlClusterCancelLoadControlEventCallback(int32u issuerEventId,
                                                    int16u deviceClass,
                                                    int8u utilityEnrollmentGroup,
                                                    int8u cancelControl,
                                                    int32u effectiveTime);
boolean emberAfDemandResponseLoadControlClusterCancelAllLoadControlEventsCallback(int8u cancelControl);
boolean emberAfDemandResponseLoadControlClusterReportEventStatusCallback(int32u issuerEventId,
                                                int8u eventStatus,
                                                int32u eventStatusTime,
                                                int8u criticalityLevelApplied,
                                                int16s coolingTemperatureSetPointApplied,
                                                int16s heatingTemperatureSetPointApplied,
                                                int8s averageLoadAdjustmentPercentageApplied,
                                                int8u dutyCycleApplied,
                                                int8u eventControl,
                                                int8u signatureType,
                                                int8u* signature);
boolean emberAfDemandResponseLoadControlClusterGetScheduledEventsCallback(int32u startTime,
                                                int8u numberOfEvents);
void emberAfDemandResponseLoadControlClusterServerInitCallback(int8u endpoint);
void emberAfDemandResponseLoadControlClusterClientInitCallback(int8u endpoint);
void emberAfDemandResponseLoadControlClusterServerTickCallback(int8u endpoint);
```

```
void emberAfDemandResponseLoadControlClusterClientTickCallback(int8u endpoint);
void emberAfDemandResponseLoadControlClusterServerAttributeChangedCallback(int8u endpoint,
                                                                            EmberAfAttributeId attributeId);
void emberAfDemandResponseLoadControlClusterClientAttributeChangedCallback(int8u endpoint,
                                                                            EmberAfAttributeId attributeId);
void emberAfDemandResponseLoadControlClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                                                EmberAfAttributeId attributeId,
                                                                                int16u manufacturerCode);
void emberAfDemandResponseLoadControlClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                                                EmberAfAttributeId attributeId,
                                                                                int16u manufacturerCode);
void emberAfDemandResponseLoadControlClusterServerDefaultResponseCallback(int8u endpoint,
                                                                          int8u commandId,
                                                                          EmberAfStatus status);
void emberAfDemandResponseLoadControlClusterClientDefaultResponseCallback(int8u endpoint,
                                                                          int8u commandId,
                                                                          EmberAfStatus status);
void emberAfDemandResponseLoadControlClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                                                      int16u indexOrDestination,
                                                                      EmberApsFrame * apsFrame,
                                                                      int16u msgLen,
                                                                      int8u * message,
                                                                      EmberStatus status);
void emberAfDemandResponseLoadControlClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                                                      int16u indexOrDestination,
                                                                      EmberApsFrame * apsFrame,
                                                                      int16u msgLen,
                                                                      int8u * message,
                                                                      EmberStatus status);

// @{

boolean emberAfSimpleMeteringClusterGetProfileResponseCallback(int32u endTime,
                                                               int8u status,
                                                               int8u profileIntervalPeriod,
                                                               int8u numberOfPeriodsDelivered,
                                                               int8u* intervals);
boolean emberAfSimpleMeteringClusterRequestMirrorCallback(void);
boolean emberAfSimpleMeteringClusterRemoveMirrorCallback(void);
boolean emberAfSimpleMeteringClusterRequestFastPollModeResponseCallback(int8u appliedUpdatePeriod,
                                                                        int32u fastPollModeEndtime);
boolean emberAfSimpleMeteringClusterGetProfileCallback(int8u intervalChannel,
                                                       int32u endTime,
                                                       int8u numberOfPeriods);
boolean emberAfSimpleMeteringClusterRequestMirrorResponseCallback(int16u endpointId);
boolean emberAfSimpleMeteringClusterMirrorRemovedCallback(int16u endpointId);
boolean emberAfSimpleMeteringClusterRequestFastPollModeCallback(int8u fastPollUpdatePeriod,
                                                                int8u duration);
void emberAfSimpleMeteringClusterServerInitCallback(int8u endpoint);
void emberAfSimpleMeteringClusterClientInitCallback(int8u endpoint);
void emberAfSimpleMeteringClusterServerTickCallback(int8u endpoint);
void emberAfSimpleMeteringClusterClientTickCallback(int8u endpoint);
void emberAfSimpleMeteringClusterServerAttributeChangedCallback(int8u endpoint,
                                                                EmberAfAttributeId attributeId);
void emberAfSimpleMeteringClusterClientAttributeChangedCallback(int8u endpoint,
                                                                EmberAfAttributeId attributeId);
```

```
07569 void emberAfSimpleMeteringClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
07570                                                                                     EmberAfAttributeId attributeId,
07571                                                                                     int16u manufacturerCode);
07581 void emberAfSimpleMeteringClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
07582                                                                                     EmberAfAttributeId attributeId,
07583                                                                                     int16u manufacturerCode);
07592 void emberAfSimpleMeteringClusterServerDefaultResponseCallback(int8u endpoint,
07593                                                                int8u commandId,
07594                                                                EmberAfStatus status);
07603 void emberAfSimpleMeteringClusterClientDefaultResponseCallback(int8u endpoint,
07604                                                                int8u commandId,
07605                                                                EmberAfStatus status);
07618 void emberAfSimpleMeteringClusterServerMessageSentCallback(EmberOutgoingMessageType type,
07619                                                            int16u indexOrDestination,
07620                                                            EmberApsFrame * apsFrame,
07621                                                            int16u msgLen,
07622                                                            int8u * message,
07623                                                            EmberStatus status);
07636 void emberAfSimpleMeteringClusterClientMessageSentCallback(EmberOutgoingMessageType type,
07637                                                            int16u indexOrDestination,
07638                                                            EmberApsFrame * apsFrame,
07639                                                            int16u msgLen,
07640                                                            int8u * message,
07641                                                            EmberStatus status);
07642
07646 // @{
07647
07671 boolean emberAfPriceClusterPublishPriceCallback(int32u providerId,
07672                                                 int8u* rateLabel,
07673                                                 int32u issuerEventId,
07674                                                 int32u currentTime,
07675                                                 int8u unitOfMeasure,
07676                                                 int16u currency,
07677                                                 int8u priceTrailingDigitAndPriceTier,
07678                                                 int8u numberOfPriceTiersAndRegisterTier,
07679                                                 int32u startTime,
07680                                                 int16u durationInMinutes,
07681                                                 int32u price,
07682                                                 int8u priceRatio,
07683                                                 int32u generationPrice,
07684                                                 int8u generationPriceRatio,
07685                                                 int32u alternateCostDelivered,
07686                                                 int8u alternateCostUnit,
07687                                                 int8u alternateCostTrailingDigit,
07688                                                 int8u numberOfBlockThresholds,
07689                                                 int8u priceControl);
07700 boolean emberAfPriceClusterPublishBlockPeriodCallback(int32u providerId,
07701                                                       int32u issuerEventId,
07702                                                       int32u blockPeriodStartTime,
07703                                                       int32u blockPeriodDurationInMinutes,
07704                                                       int8u numberOfPriceTiersAndNumberOfBlockThresholds,
07705                                                       int8u blockPeriodControl);
07711 boolean emberAfPriceClusterGetCurrentPriceCallback(int8u commandOptions);
07718 boolean emberAfPriceClusterGetScheduledPricesCallback(int32u startTime,
07719                                                       int8u numberOfEvents);
07728 boolean emberAfPriceClusterPriceAcknowledgementCallback(int32u providerId,
07729                                                         int32u issuerEventId,
07730                                                         int32u priceAckTime,
```

```
                                        int8u control);
boolean emberAfPriceClusterGetBlockPeriodsCallback(int32u startTime,
                                       int8u numberOfEvents);
void emberAfPriceClusterServerInitCallback(int8u endpoint);
void emberAfPriceClusterClientInitCallback(int8u endpoint);
void emberAfPriceClusterServerTickCallback(int8u endpoint);
void emberAfPriceClusterClientTickCallback(int8u endpoint);
void emberAfPriceClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfPriceClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfPriceClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfPriceClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfPriceClusterServerDefaultResponseCallback(int8u endpoint,
                                       int8u commandId,
                                       EmberAfStatus status);
void emberAfPriceClusterClientDefaultResponseCallback(int8u endpoint,
                                       int8u commandId,
                                       EmberAfStatus status);
void emberAfPriceClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                      int16u indexOrDestination,
                                      EmberApsFrame * apsFrame,
                                      int16u msgLen,
                                      int8u * message,
                                      EmberStatus status);
void emberAfPriceClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                      int16u indexOrDestination,
                                      EmberApsFrame * apsFrame,
                                      int16u msgLen,
                                      int8u * message,
                                      EmberStatus status);

// @{

boolean emberAfMessagingClusterDisplayMessageCallback(int32u messageId,
                                       int8u messageControl,
                                       int32u startTime,
                                       int16u durationInMinutes,
                                       int8u* message);
boolean emberAfMessagingClusterCancelMessageCallback(int32u messageId,
                                      int8u messageControl);
boolean emberAfMessagingClusterGetLastMessageCallback(void);
boolean emberAfMessagingClusterMessageConfirmationCallback(int32u messageId,
                                        int32u confirmationTime);
void emberAfMessagingClusterServerInitCallback(int8u endpoint);
void emberAfMessagingClusterClientInitCallback(int8u endpoint);
void emberAfMessagingClusterServerTickCallback(int8u endpoint);
void emberAfMessagingClusterClientTickCallback(int8u endpoint);
void emberAfMessagingClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfMessagingClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
```

```
07964 void emberAfMessagingClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
07965                                                                                EmberAfAttributeId attributeId,
07966                                                                                int16u manufacturerCode);
07976 void emberAfMessagingClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
07977                                                                                EmberAfAttributeId attributeId,
07978                                                                                int16u manufacturerCode);
07987 void emberAfMessagingClusterServerDefaultResponseCallback(int8u endpoint,
07988                                                           int8u commandId,
07989                                                           EmberAfStatus status);
07998 void emberAfMessagingClusterClientDefaultResponseCallback(int8u endpoint,
07999                                                           int8u commandId,
08000                                                           EmberAfStatus status);
08013 void emberAfMessagingClusterServerMessageSentCallback(EmberOutgoingMessageType type,
08014                                                       int16u indexOrDestination,
08015                                                       EmberApsFrame * apsFrame,
08016                                                       int16u msgLen,
08017                                                       int8u * message,
08018                                                       EmberStatus status);
08031 void emberAfMessagingClusterClientMessageSentCallback(EmberOutgoingMessageType type,
08032                                                       int16u indexOrDestination,
08033                                                       EmberApsFrame * apsFrame,
08034                                                       int16u msgLen,
08035                                                       int8u * message,
08036                                                       EmberStatus status);
08037
08041 // @{
08042
08050 boolean emberAfTunnelingClusterRequestTunnelCallback(int8u protocolId,
08051                                                     int16u manufacturerCode,
08052                                                     int8u flowControlSupport);
08058 boolean emberAfTunnelingClusterCloseTunnelCallback(int16u tunnelId);
08065 boolean emberAfTunnelingClusterTransferDataClientToServerCallback(int16u tunnelId,
08066                                                                   int8u* data);
08073 boolean emberAfTunnelingClusterTransferDataErrorClientToServerCallback(int16u tunnelId,
08074                                                                        int8u transferDataStatus);
08081 boolean emberAfTunnelingClusterAckTransferDataClientToServerCallback(int16u tunnelId,
08082                                                                      int16u numberOfBytesLeft);
08089 boolean emberAfTunnelingClusterReadyDataClientToServerCallback(int16u tunnelId,
08090                                                                int16u numberOfOctetsLeft);
08097 boolean emberAfTunnelingClusterRequestTunnelResponseCallback(int16u tunnelId,
08098                                                              int8u tunnelStatus);
08105 boolean emberAfTunnelingClusterTransferDataServerToClientCallback(int16u tunnelId,
08106                                                                   int8u* data);
08113 boolean emberAfTunnelingClusterTransferDataErrorServerToClientCallback(int16u tunnelId,
08114                                                                        int8u transferDataStatus);
08121 boolean emberAfTunnelingClusterAckTransferDataServerToClientCallback(int16u tunnelId,
08122                                                                      int16u numberOfBytesLeft);
08129 boolean emberAfTunnelingClusterReadyDataServerToClientCallback(int16u tunnelId,
08130                                                                int16u numberOfOctetsLeft);
08137 void emberAfTunnelingClusterServerInitCallback(int8u endpoint);
08144 void emberAfTunnelingClusterClientInitCallback(int8u endpoint);
08151 void emberAfTunnelingClusterServerTickCallback(int8u endpoint);
08158 void emberAfTunnelingClusterClientTickCallback(int8u endpoint);
08166 void emberAfTunnelingClusterServerAttributeChangedCallback(int8u endpoint,
08167                                                           EmberAfAttributeId attributeId);
08175 void emberAfTunnelingClusterClientAttributeChangedCallback(int8u endpoint,
```

```
                                        EmberAfAttributeId attributeId);
void emberAfTunnelingClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfTunnelingClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfTunnelingClusterServerDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfTunnelingClusterClientDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfTunnelingClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                      int16u indexOrDestination,
                                      EmberApsFrame * apsFrame,
                                      int16u msgLen,
                                      int8u * message,
                                      EmberStatus status);
void emberAfTunnelingClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                      int16u indexOrDestination,
                                      EmberApsFrame * apsFrame,
                                      int16u msgLen,
                                      int8u * message,
                                      EmberStatus status);

// @{

boolean emberAfPrepaymentClusterSelectAvailableEmergencyCreditCallback(int32u commandDateTime,
                                            int8u originatingDevice,
                                            int8u* siteId,
                                            int8u* meterSerialNumber);
boolean emberAfPrepaymentClusterChangeSupplyCallback(int32u providerId,
                                     int32u requestDateTime,
                                     int8u* siteId,
                                     int8u* meterSerialNumber,
                                     int32u implementationDateTime,
                                     int8u proposedSupplyStatus,
                                     int8u originatorIdSupplyControlBits);
boolean emberAfPrepaymentClusterSupplyStatusResponseCallback(int32u providerId,
                                         int32u implementationtDateTime,
                                         int8u supplyStatus);
void emberAfPrepaymentClusterServerInitCallback(int8u endpoint);
void emberAfPrepaymentClusterClientInitCallback(int8u endpoint);
void emberAfPrepaymentClusterServerTickCallback(int8u endpoint);
void emberAfPrepaymentClusterClientTickCallback(int8u endpoint);
void emberAfPrepaymentClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfPrepaymentClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfPrepaymentClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfPrepaymentClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfPrepaymentClusterServerDefaultResponseCallback(int8u endpoint,
```

```
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfPrepaymentClusterClientDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfPrepaymentClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                      int16u indexOrDestination,
                                      EmberApsFrame * apsFrame,
                                      int16u msgLen,
                                      int8u * message,
                                      EmberStatus status);
void emberAfPrepaymentClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                      int16u indexOrDestination,
                                      EmberApsFrame * apsFrame,
                                      int16u msgLen,
                                      int8u * message,
                                      EmberStatus status);

// @{

boolean emberAfKeyEstablishmentClusterClientCommandReceivedCallback(EmberAfClusterCommand * cmd);
boolean emberAfKeyEstablishmentClusterServerCommandReceivedCallback(EmberAfClusterCommand * cmd);
boolean emberAfKeyEstablishmentClusterInitiateKeyEstablishmentRequestCallback(int16u keyEstablishmentSuite,
                                                int8u ephemeralDataGenerateTime,
                                                int8u confirmKeyGenerateTime,
                                                int8u* identity);
boolean emberAfKeyEstablishmentClusterEphemeralDataRequestCallback(int8u* ephemeralData);
boolean emberAfKeyEstablishmentClusterConfirmKeyDataRequestCallback(int8u* secureMessageAuthenticationCode);
boolean emberAfKeyEstablishmentClusterTerminateKeyEstablishmentCallback(int8u statusCode,
                                              int8u waitTime,
                                              int16u keyEstablishmentSuite);
boolean emberAfKeyEstablishmentClusterInitiateKeyEstablishmentResponseCallback
(int16u requestedKeyEstablishmentSuite,
                                                int8u ephemeralDataGenerateTime,
                                                int8u confirmKeyGenerateTime,
                                                int8u* identity);
boolean emberAfKeyEstablishmentClusterEphemeralDataResponseCallback(int8u* ephemeralData);
boolean emberAfKeyEstablishmentClusterConfirmKeyDataResponseCallback(int8u* secureMessageAuthenticationCode);
void emberAfKeyEstablishmentClusterServerInitCallback(int8u endpoint);
void emberAfKeyEstablishmentClusterClientInitCallback(int8u endpoint);
void emberAfKeyEstablishmentClusterServerTickCallback(int8u endpoint);
void emberAfKeyEstablishmentClusterClientTickCallback(int8u endpoint);
void emberAfKeyEstablishmentClusterServerAttributeChangedCallback(int8u endpoint,
                                            EmberAfAttributeId attributeId);
void emberAfKeyEstablishmentClusterClientAttributeChangedCallback(int8u endpoint,
                                            EmberAfAttributeId attributeId);
void emberAfKeyEstablishmentClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfKeyEstablishmentClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfKeyEstablishmentClusterServerDefaultResponseCallback(int8u endpoint,
                                           int8u commandId,
                                           EmberAfStatus status);
void emberAfKeyEstablishmentClusterClientDefaultResponseCallback(int8u endpoint,
```

```
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfKeyEstablishmentClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                      int16u indexOrDestination,
                                      EmberApsFrame * apsFrame,
                                      int16u msgLen,
                                      int8u * message,
                                      EmberStatus status);
void emberAfKeyEstablishmentClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                      int16u indexOrDestination,
                                      EmberApsFrame * apsFrame,
                                      int16u msgLen,
                                      int8u * message,
                                      EmberStatus status);

// @{

boolean emberAfSampleMfgSpecificClusterCommandOneCallback(int8u argOne);
void emberAfSampleMfgSpecificClusterServerInitCallback(int8u endpoint);
void emberAfSampleMfgSpecificClusterClientInitCallback(int8u endpoint);
void emberAfSampleMfgSpecificClusterServerTickCallback(int8u endpoint);
void emberAfSampleMfgSpecificClusterClientTickCallback(int8u endpoint);
void emberAfSampleMfgSpecificClusterServerAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfSampleMfgSpecificClusterClientAttributeChangedCallback(int8u endpoint,
                                        EmberAfAttributeId attributeId);
void emberAfSampleMfgSpecificClusterServerManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfSampleMfgSpecificClusterClientManufacturerSpecificAttributeChangedCallback(int8u endpoint,
                                                EmberAfAttributeId attributeId,
                                                int16u manufacturerCode);
void emberAfSampleMfgSpecificClusterServerDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfSampleMfgSpecificClusterClientDefaultResponseCallback(int8u endpoint,
                                        int8u commandId,
                                        EmberAfStatus status);
void emberAfSampleMfgSpecificClusterServerMessageSentCallback(EmberOutgoingMessageType type,
                                      int16u indexOrDestination,
                                      EmberApsFrame * apsFrame,
                                      int16u msgLen,
                                      int8u * message,
                                      EmberStatus status);
void emberAfSampleMfgSpecificClusterClientMessageSentCallback(EmberOutgoingMessageType type,
                                      int16u indexOrDestination,
                                      EmberApsFrame * apsFrame,
                                      int16u msgLen,
                                      int8u * message,
                                      EmberStatus status);

#endif // __EMBER_AF_CALLBACK_PROTOTYPES__
```

client-api.h

Go to the documentation of this file.

```
00001
00006 #ifndef __CLIENT_API__
00007 #define __CLIENT_API__
00008
00010 // @{
00011
00035 int16u emberAfFillExternalBuffer(int8u frameControl,
00036                                 EmberAfClusterId clusterId,
00037                                 int8u commandId,
00038                                 PGM_P format,
00039                                 ...);
00040
00065 int16u emberAfFillExternalManufacturerSpecificBuffer(int8u frameControl,
00066                                                      EmberAfClusterId clusterId,
00067                                                      int16u manufacturerCode,
00068                                                      int8u commandId,
00069                                                      PGM_P format,
00070                                                      ...);
00071
00081 void emberAfSetExternalBuffer(int8u *buffer,
00082                               int16u bufferLen,
00083                               int16u *responseLengthPtr,
00084                               EmberApsFrame *apsFramePtr);
00085
00093 int16u emberAfFillBuffer(int8u *buffer,
00094                          int16u bufferLen,
00095                          int8u frameControl,
00096                          int8u commandId,
00097                          PGM_P format,
00098                          ...);
00099
00100 // Generated macros
00101 #include "../gen/client-command-macro.h"
00102
00105 #endif // __CLIENT_API__
```

cluster-id.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generateClusterIDs()
//

// Enclosing macro to prevent multiple inclusion
#ifndef __EMBER_AF_CLUSTER_ID__
#define __EMBER_AF_CLUSTER_ID__


// Cluster domain specification levels:
//   * General: 075123r03
//   * Closures: 1.1, 053520r27
//   * HA: 1.1, 053520r27
//   * HVAC: 1.1, 053520r27
//   * Lighting: 1.1, 053520r27
//   * Measurement & Sensing: 1.1, 053520r27
//   * Security & Safety: 1.1, 053520r27
//   * CBA: unofficial
//   * SE: 1.1, 105638r09
//   * Ember: UNKNOWN
#define ZCL_BASIC_CLUSTER_ID                    0x0000
#define ZCL_POWER_CONFIG_CLUSTER_ID             0x0001
#define ZCL_DEVICE_TEMP_CLUSTER_ID              0x0002
#define ZCL_IDENTIFY_CLUSTER_ID                 0x0003
#define ZCL_GROUPS_CLUSTER_ID                   0x0004
#define ZCL_SCENES_CLUSTER_ID                   0x0005
#define ZCL_ON_OFF_CLUSTER_ID                   0x0006
#define ZCL_ON_OFF_SWITCH_CONFIG_CLUSTER_ID     0x0007
#define ZCL_LEVEL_CONTROL_CLUSTER_ID            0x0008
#define ZCL_ALARM_CLUSTER_ID                    0x0009
#define ZCL_TIME_CLUSTER_ID                     0x000A
#define ZCL_RSSI_LOCATION_CLUSTER_ID            0x000B
#define ZCL_BINARY_INPUT_BASIC_CLUSTER_ID       0x000F
#define ZCL_COMMISSIONING_CLUSTER_ID            0x0015
#define ZCL_SHADE_CONFIG_CLUSTER_ID             0x0100
#define ZCL_PUMP_CONFIG_CONTROL_CLUSTER_ID      0x0200
#define ZCL_THERMOSTAT_CLUSTER_ID               0x0201
#define ZCL_FAN_CONTROL_CLUSTER_ID              0x0202
```

```
#define ZCL_DEHUMID_CONTROL_CLUSTER_ID              0x0203
#define ZCL_THERMOSTAT_UI_CONFIG_CLUSTER_ID         0x0204
#define ZCL_COLOR_CONTROL_CLUSTER_ID                0x0300
#define ZCL_BALLAST_CONFIGURATION_CLUSTER_ID        0x0301
#define ZCL_ILLUM_MEASUREMENT_CLUSTER_ID            0x0400
#define ZCL_ILLUM_LEVEL_SENSING_CLUSTER_ID          0x0401
#define ZCL_TEMP_MEASUREMENT_CLUSTER_ID             0x0402
#define ZCL_PRESSURE_MEASUREMENT_CLUSTER_ID         0x0403
#define ZCL_FLOW_MEASUREMENT_CLUSTER_ID             0x0404
#define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER_ID 0x0405
#define ZCL_OCCUPANCY_SENSING_CLUSTER_ID            0x0406
#define ZCL_IAS_ZONE_CLUSTER_ID                     0x0500
#define ZCL_IAS_ACE_CLUSTER_ID                      0x0501
#define ZCL_IAS_WD_CLUSTER_ID                       0x0502
#define ZCL_DOOR_LOCK_CLUSTER_ID                    0x0101
#define ZCL_WINDOW_COVERING_CLUSTER_ID              0x0102
#define ZCL_GENERIC_TUNNEL_CLUSTER_ID               0x0600
#define ZCL_BACNET_PROTOCOL_TUNNEL_CLUSTER_ID       0x0601
#define ZCL_OTA_BOOTLOAD_CLUSTER_ID                 0x0019
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID  0x0701
#define ZCL_SIMPLE_METERING_CLUSTER_ID              0x0702
#define ZCL_PRICE_CLUSTER_ID                        0x0700
#define ZCL_MESSAGING_CLUSTER_ID                    0x0703
#define ZCL_TUNNELING_CLUSTER_ID                    0x0704
#define ZCL_PREPAYMENT_CLUSTER_ID                   0x0705
#define ZCL_KEY_ESTABLISHMENT_CLUSTER_ID            0x0800
#define ZCL_SAMPLE_MFG_SPECIFIC_CLUSTER_ID          0xFC00
#endif // __EMBER_AF_CLUSTER_ID__
```

color-control.h

Go to the documentation of this file.

```
// ********************************************************************
// * color-control.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************

// These two define the value of a "step". Color control
// spec mentions stepping commands as units, which may be different
// than changing the value of the actual hue and/or saturation by 1.
// Therefore step or move functions use these. You may set this to 1
// so that the "step" will be the same as changing value by 1.
// Note: the moveTo command will not use these value as they
// move continuously by 1 (or whatever needed by the tick).
#define ZCL_COLOR_CONTROL_HUE_STEP_UNIT 1
#define ZCL_COLOR_CONTROL_SATURATION_STEP_UNIT 1
#define ZCL_INVALID_COLOR_CONTROL_COMMAND_ID 0xFF
```

app » framework » gen

command-id.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generateCommandIDs()
//

// Enclosing macro to prevent multiple inclusion
#ifndef __EMBER_AF_COMMAND_ID__
#define __EMBER_AF_COMMAND_ID__


// Global commands

// Either direction
#define ZCL_READ_ATTRIBUTES_COMMAND_ID                    0x00    // Ver.: always
#define ZCL_READ_ATTRIBUTES_RESPONSE_COMMAND_ID           0x01    // Ver.: always
#define ZCL_WRITE_ATTRIBUTES_COMMAND_ID                   0x02    // Ver.: always
#define ZCL_WRITE_ATTRIBUTES_UNDIVIDED_COMMAND_ID         0x03    // Ver.: always
#define ZCL_WRITE_ATTRIBUTES_RESPONSE_COMMAND_ID          0x04    // Ver.: always
#define ZCL_WRITE_ATTRIBUTES_NO_RESPONSE_COMMAND_ID       0x05    // Ver.: always
#define ZCL_CONFIGURE_REPORTING_COMMAND_ID                0x06    // Ver.: always
#define ZCL_CONFIGURE_REPORTING_RESPONSE_COMMAND_ID       0x07    // Ver.: always
#define ZCL_READ_REPORTING_CONFIGURATION_COMMAND_ID       0x08    // Ver.: always
#define ZCL_READ_REPORTING_CONFIGURATION_RESPONSE_COMMAND_ID 0x09    // Ver.: always
#define ZCL_REPORT_ATTRIBUTES_COMMAND_ID                  0x0A    // Ver.: always
#define ZCL_DEFAULT_RESPONSE_COMMAND_ID                   0x0B    // Ver.: always
#define ZCL_DISCOVER_ATTRIBUTES_COMMAND_ID                0x0C    // Ver.: always
#define ZCL_DISCOVER_ATTRIBUTES_RESPONSE_COMMAND_ID       0x0D    // Ver.: always
// Command types for cluster: Basic
// Cluster specification level: 075123r03

// Client to server
#define ZCL_RESET_TO_FACTORY_DEFAULTS_COMMAND_ID 0x00    // Ver.: always


// Command types for cluster: Identify
// Cluster specification level: 075123r03

// Server to client
#define ZCL_IDENTIFY_QUERY_RESPONSE_COMMAND_ID 0x00    // Ver.: always

// Client to server
#define ZCL_IDENTIFY_COMMAND_ID 0x00    // Ver.: always
#define ZCL_IDENTIFY_QUERY_COMMAND_ID 0x01    // Ver.: always


// Command types for cluster: Groups
// Cluster specification level: 075123r03

```

```
// Server to client
#define ZCL_ADD_GROUP_RESPONSE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_VIEW_GROUP_RESPONSE_COMMAND_ID 0x01    // Ver.: always
#define ZCL_GET_GROUP_MEMBERSHIP_RESPONSE_COMMAND_ID 0x02    // Ver.: always
#define ZCL_REMOVE_GROUP_RESPONSE_COMMAND_ID 0x03    // Ver.: always

// Client to server
#define ZCL_ADD_GROUP_COMMAND_ID 0x00    // Ver.: always
#define ZCL_VIEW_GROUP_COMMAND_ID 0x01    // Ver.: always
#define ZCL_GET_GROUP_MEMBERSHIP_COMMAND_ID 0x02    // Ver.: always
#define ZCL_REMOVE_GROUP_COMMAND_ID 0x03    // Ver.: always
#define ZCL_REMOVE_ALL_GROUPS_COMMAND_ID 0x04    // Ver.: always
#define ZCL_ADD_GROUP_IF_IDENTIFYING_COMMAND_ID 0x05    // Ver.: always


// Command types for cluster: Scenes
// Cluster specification level: 075123r03

// Server to client
#define ZCL_ADD_SCENE_RESPONSE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_VIEW_SCENE_RESPONSE_COMMAND_ID 0x01    // Ver.: always
#define ZCL_REMOVE_SCENE_RESPONSE_COMMAND_ID 0x02    // Ver.: always
#define ZCL_REMOVE_ALL_SCENES_RESPONSE_COMMAND_ID 0x03    // Ver.: always
#define ZCL_STORE_SCENE_RESPONSE_COMMAND_ID 0x04    // Ver.: always
#define ZCL_GET_SCENE_MEMBERSHIP_RESPONSE_COMMAND_ID 0x06    // Ver.: always

// Client to server
#define ZCL_ADD_SCENE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_VIEW_SCENE_COMMAND_ID 0x01    // Ver.: always
#define ZCL_REMOVE_SCENE_COMMAND_ID 0x02    // Ver.: always
#define ZCL_REMOVE_ALL_SCENES_COMMAND_ID 0x03    // Ver.: always
#define ZCL_STORE_SCENE_COMMAND_ID 0x04    // Ver.: always
#define ZCL_RECALL_SCENE_COMMAND_ID 0x05    // Ver.: always
#define ZCL_GET_SCENE_MEMBERSHIP_COMMAND_ID 0x06    // Ver.: always


// Command types for cluster: On/off
// Cluster specification level: 075123r03

// Client to server
#define ZCL_OFF_COMMAND_ID 0x00    // Ver.: always
#define ZCL_ON_COMMAND_ID 0x01    // Ver.: always
#define ZCL_TOGGLE_COMMAND_ID 0x02    // Ver.: always
#define ZCL_SAMPLE_MFG_SPECIFIC_OFF_WITH_TRANSITION_COMMAND_ID 0x00    // Ver.: always mfgCode: 0x1002
#define ZCL_SAMPLE_MFG_SPECIFIC_ON_WITH_TRANSITION_COMMAND_ID 0x01    // Ver.: always mfgCode: 0x1002
#define ZCL_SAMPLE_MFG_SPECIFIC_TOGGLE_WITH_TRANSITION_COMMAND_ID 0x02    // Ver.: always mfgCode: 0x1002


// Command types for cluster: Level Control
// Cluster specification level: 075123r03

// Client to server
#define ZCL_MOVE_TO_LEVEL_COMMAND_ID 0x00    // Ver.: always
```

```
#define ZCL_MOVE_COMMAND_ID 0x01    // Ver.: always
#define ZCL_STEP_COMMAND_ID 0x02    // Ver.: always
#define ZCL_STOP_COMMAND_ID 0x03    // Ver.: always
#define ZCL_MOVE_TO_LEVEL_WITH_ON_OFF_COMMAND_ID 0x04    // Ver.: always
#define ZCL_MOVE_WITH_ON_OFF_COMMAND_ID 0x05    // Ver.: always
#define ZCL_STEP_WITH_ON_OFF_COMMAND_ID 0x06    // Ver.: always
#define ZCL_STOP_WITH_ON_OFF_COMMAND_ID 0x07    // Ver.: always


// Command types for cluster: Alarms
// Cluster specification level: 075123r03

// Server to client
#define ZCL_ALARM_COMMAND_ID 0x00    // Ver.: always
#define ZCL_GET_ALARM_RESPONSE_COMMAND_ID 0x01    // Ver.: always

// Client to server
#define ZCL_RESET_ALARM_COMMAND_ID 0x00    // Ver.: always
#define ZCL_RESET_ALL_ALARMS_COMMAND_ID 0x01    // Ver.: always
#define ZCL_GET_ALARM_COMMAND_ID 0x02    // Ver.: always
#define ZCL_RESET_ALARM_LOG_COMMAND_ID 0x03    // Ver.: always


// Command types for cluster: RSSI Location
// Cluster specification level: 075123r03

// Server to client
#define ZCL_DEVICE_CONFIGURATION_RESPONSE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_LOCATION_DATA_RESPONSE_COMMAND_ID 0x01    // Ver.: always
#define ZCL_LOCATION_DATA_NOTIFICATION_COMMAND_ID 0x02    // Ver.: always
#define ZCL_COMPACT_LOCATION_DATA_NOTIFICATION_COMMAND_ID 0x03    // Ver.: always
#define ZCL_RSSI_PING_COMMAND_ID 0x04    // Ver.: always
#define ZCL_RSSI_REQUEST_COMMAND_ID 0x05    // Ver.: always
#define ZCL_REPORT_RSSI_MEASUREMENTS_COMMAND_ID 0x06    // Ver.: always
#define ZCL_REQUEST_OWN_LOCATION_COMMAND_ID 0x07    // Ver.: always

// Client to server
#define ZCL_SET_ABSOLUTE_LOCATION_COMMAND_ID 0x00    // Ver.: always
#define ZCL_SET_DEVICE_CONFIGURATION_COMMAND_ID 0x01    // Ver.: always
#define ZCL_GET_DEVICE_CONFIGURATION_COMMAND_ID 0x02    // Ver.: always
#define ZCL_GET_LOCATION_DATA_COMMAND_ID 0x03    // Ver.: always
#define ZCL_RSSI_RESPONSE_COMMAND_ID 0x04    // Ver.: always
#define ZCL_SEND_PINGS_COMMAND_ID 0x05    // Ver.: always
#define ZCL_ANCHOR_NODE_ANNOUNCE_COMMAND_ID 0x06    // Ver.: always


// Command types for cluster: Commissioning
// Cluster specification level: 075123r03

// Server to client
#define ZCL_RESTART_DEVICE_RESPONSE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_SAVE_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID 0x01    // Ver.: always
#define ZCL_RESTORE_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID 0x02    // Ver.: always
#define ZCL_RESET_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID 0x03    // Ver.: always

// Client to server
```

```
#define ZCL_RESTART_DEVICE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_SAVE_STARTUP_PARAMETERS_COMMAND_ID 0x01    // Ver.: always
#define ZCL_RESTORE_STARTUP_PARAMETERS_COMMAND_ID 0x02    // Ver.: always
#define ZCL_RESET_STARTUP_PARAMETERS_COMMAND_ID 0x03    // Ver.: always


// Command types for cluster: Thermostat
// Cluster specification level: 1.1, 053520r27

// Client to server
#define ZCL_SETPOINT_RAISE_LOWER_COMMAND_ID 0x00    // Ver.: always


// Command types for cluster: Color Control
// Cluster specification level: 1.1, 053520r27

// Client to server
#define ZCL_MOVE_TO_HUE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_MOVE_HUE_COMMAND_ID 0x01    // Ver.: always
#define ZCL_STEP_HUE_COMMAND_ID 0x02    // Ver.: always
#define ZCL_MOVE_TO_SATURATION_COMMAND_ID 0x03    // Ver.: always
#define ZCL_MOVE_SATURATION_COMMAND_ID 0x04    // Ver.: always
#define ZCL_STEP_SATURATION_COMMAND_ID 0x05    // Ver.: always
#define ZCL_MOVE_TO_HUE_AND_SATURATION_COMMAND_ID 0x06    // Ver.: always
#define ZCL_MOVE_TO_COLOR_COMMAND_ID 0x07    // Ver.: always
#define ZCL_MOVE_COLOR_COMMAND_ID 0x08    // Ver.: always
#define ZCL_STEP_COLOR_COMMAND_ID 0x09    // Ver.: always
#define ZCL_MOVE_TO_COLOR_TEMPERATURE_COMMAND_ID 0x0A    // Ver.: always


// Command types for cluster: IAS Zone
// Cluster specification level: 1.1, 053520r27

// Server to client
#define ZCL_ZONE_STATUS_CHANGE_NOTIFICATION_COMMAND_ID 0x00    // Ver.: always
#define ZCL_ZONE_ENROLL_REQUEST_COMMAND_ID 0x01    // Ver.: always

// Client to server
#define ZCL_ZONE_ENROLL_RESPONSE_COMMAND_ID 0x00    // Ver.: always


// Command types for cluster: IAS ACE
// Cluster specification level: 1.1, 053520r27

// Server to client
#define ZCL_ARM_RESPONSE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_GET_ZONE_ID_MAP_RESPONSE_COMMAND_ID 0x01    // Ver.: always
#define ZCL_GET_ZONE_INFORMATION_RESPONSE_COMMAND_ID 0x02    // Ver.: always

// Client to server
#define ZCL_ARM_COMMAND_ID 0x00    // Ver.: always
#define ZCL_BYPASS_COMMAND_ID 0x01    // Ver.: always
#define ZCL_EMERGENCY_COMMAND_ID 0x02    // Ver.: always
#define ZCL_FIRE_COMMAND_ID 0x03    // Ver.: always
#define ZCL_PANIC_COMMAND_ID 0x04    // Ver.: always
#define ZCL_GET_ZONE_ID_MAP_COMMAND_ID 0x05    // Ver.: always
```

```
#define ZCL_GET_ZONE_INFORMATION_COMMAND_ID 0x06    // Ver.: always


// Command types for cluster: IAS WD
// Cluster specification level: 1.1, 053520r27

// Client to server
#define ZCL_START_WARNING_COMMAND_ID 0x00    // Ver.: always
#define ZCL_SQUAWK_COMMAND_ID 0x01    // Ver.: always


// Command types for cluster: Door Lock
// Cluster specification level: 1.1, 053520r27

// Server to client
#define ZCL_LOCK_DOOR_RESPONSE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_UNLOCK_DOOR_RESPONSE_COMMAND_ID 0x01    // Ver.: always

// Client to server
#define ZCL_LOCK_DOOR_COMMAND_ID 0x00    // Ver.: always
#define ZCL_UNLOCK_DOOR_COMMAND_ID 0x01    // Ver.: always


// Command types for cluster: Window Covering
// Cluster specification level: 1.1, 053520r27

// Client to server
#define ZCL_WINDOW_COVERING_UP_OPEN_COMMAND_ID 0x00    // Ver.: always
#define ZCL_WINDOW_COVERING_DOWN_CLOSE_COMMAND_ID 0x01    // Ver.: always
#define ZCL_WINDOW_COVERING_STOP_COMMAND_ID 0x02    // Ver.: always
#define ZCL_WINDOW_COVERING_GO_TO_LIFT_SETPOINT_COMMAND_ID 0x03    // Ver.: always
#define ZCL_WINDOW_COVERING_GO_TO_LIFT_VALUE_COMMAND_ID 0x04    // Ver.: always
#define ZCL_WINDOW_COVERING_GO_TO_LIFT_PERCENTAGE_COMMAND_ID 0x05    // Ver.: always
#define ZCL_WINDOW_COVERING_GO_TO_TILT_SETPOINT_COMMAND_ID 0x06    // Ver.: always
#define ZCL_WINDOW_COVERING_GO_TO_TILT_VALUE_COMMAND_ID 0x07    // Ver.: always
#define ZCL_WINDOW_COVERING_GO_TO_TILT_PERCENTAGE_COMMAND_ID 0x08    // Ver.: always
#define ZCL_WINDOW_COVERING_GO_TO_PROGRAM_SETPOINT_COMMAND_ID 0x09    // Ver.: always


// Command types for cluster: Generic Tunnel
// Cluster specification level: unofficial

// Server to client
#define ZCL_MATCH_PROTOCOL_ADDRESS_RESPONSE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_ADVERTISE_PROTOCOL_ADDRESS_COMMAND_ID 0x01    // Ver.: always

// Client to server
#define ZCL_MATCH_PROTOCOL_ADDRESS_COMMAND_ID 0x00    // Ver.: always


// Command types for cluster: BACnet Protocol Tunnel
// Cluster specification level: unofficial

// Client to server
#define ZCL_TRANSFER_NPDU_COMMAND_ID 0x00    // Ver.: always


```

```
// Command types for cluster: Over the Air Bootloading Cluster
// Cluster specification level: 075123r03

// Server to client
#define ZCL_IMAGE_NOTIFY_COMMAND_ID 0x00    // Ver.: always
#define ZCL_QUERY_NEXT_IMAGE_RESPONSE_COMMAND_ID 0x02    // Ver.: always
#define ZCL_IMAGE_BLOCK_RESPONSE_COMMAND_ID 0x05    // Ver.: always
#define ZCL_UPGRADE_END_RESPONSE_COMMAND_ID 0x07    // Ver.: always
#define ZCL_QUERY_SPECIFIC_FILE_RESPONSE_COMMAND_ID 0x09    // Ver.: always

// Client to server
#define ZCL_QUERY_NEXT_IMAGE_REQUEST_COMMAND_ID 0x01    // Ver.: always
#define ZCL_IMAGE_BLOCK_REQUEST_COMMAND_ID 0x03    // Ver.: always
#define ZCL_IMAGE_PAGE_REQUEST_COMMAND_ID 0x04    // Ver.: always
#define ZCL_UPGRADE_END_REQUEST_COMMAND_ID 0x06    // Ver.: always
#define ZCL_QUERY_SPECIFIC_FILE_REQUEST_COMMAND_ID 0x08    // Ver.: always


// Command types for cluster: Demand Response and Load Control
// Cluster specification level: 1.1, 105638r09

// Server to client
#define ZCL_LOAD_CONTROL_EVENT_COMMAND_ID 0x00    // Ver.: always
#define ZCL_CANCEL_LOAD_CONTROL_EVENT_COMMAND_ID 0x01    // Ver.: always
#define ZCL_CANCEL_ALL_LOAD_CONTROL_EVENTS_COMMAND_ID 0x02    // Ver.: always

// Client to server
#define ZCL_REPORT_EVENT_STATUS_COMMAND_ID 0x00    // Ver.: always
#define ZCL_GET_SCHEDULED_EVENTS_COMMAND_ID 0x01    // Ver.: always


// Command types for cluster: Simple Metering
// Cluster specification level: 1.1, 105638r09

// Server to client
#define ZCL_GET_PROFILE_RESPONSE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_REQUEST_MIRROR_COMMAND_ID 0x01    // Ver.: always
#define ZCL_REMOVE_MIRROR_COMMAND_ID 0x02    // Ver.: always
#define ZCL_REQUEST_FAST_POLL_MODE_RESPONSE_COMMAND_ID 0x03    // Ver.: always

// Client to server
#define ZCL_GET_PROFILE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_REQUEST_MIRROR_RESPONSE_COMMAND_ID 0x01    // Ver.: always
#define ZCL_MIRROR_REMOVED_COMMAND_ID 0x02    // Ver.: always
#define ZCL_REQUEST_FAST_POLL_MODE_COMMAND_ID 0x03    // Ver.: always


// Command types for cluster: Price
// Cluster specification level: 1.1, 105638r09

// Server to client
#define ZCL_PUBLISH_PRICE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_PUBLISH_BLOCK_PERIOD_COMMAND_ID 0x01    // Ver.: always

// Client to server
#define ZCL_GET_CURRENT_PRICE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_GET_SCHEDULED_PRICES_COMMAND_ID 0x01    // Ver.: always
```

```
#define ZCL_PRICE_ACKNOWLEDGEMENT_COMMAND_ID 0x02    // Ver.: always
#define ZCL_GET_BLOCK_PERIODS_COMMAND_ID 0x03    // Ver.: always


// Command types for cluster: Messaging
// Cluster specification level: 1.1, 105638r09

// Server to client
#define ZCL_DISPLAY_MESSAGE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_CANCEL_MESSAGE_COMMAND_ID 0x01    // Ver.: always

// Client to server
#define ZCL_GET_LAST_MESSAGE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_MESSAGE_CONFIRMATION_COMMAND_ID 0x01    // Ver.: always


// Command types for cluster: Tunneling
// Cluster specification level: 1.1, 105638r09

// Server to client
#define ZCL_REQUEST_TUNNEL_RESPONSE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_TRANSFER_DATA_SERVER_TO_CLIENT_COMMAND_ID 0x01    // Ver.: always
#define ZCL_TRANSFER_DATA_ERROR_SERVER_TO_CLIENT_COMMAND_ID 0x02    // Ver.: always
#define ZCL_ACK_TRANSFER_DATA_SERVER_TO_CLIENT_COMMAND_ID 0x03    // Ver.: always
#define ZCL_READY_DATA_SERVER_TO_CLIENT_COMMAND_ID 0x04    // Ver.: always

// Client to server
#define ZCL_REQUEST_TUNNEL_COMMAND_ID 0x00    // Ver.: always
#define ZCL_CLOSE_TUNNEL_COMMAND_ID 0x01    // Ver.: always
#define ZCL_TRANSFER_DATA_CLIENT_TO_SERVER_COMMAND_ID 0x02    // Ver.: always
#define ZCL_TRANSFER_DATA_ERROR_CLIENT_TO_SERVER_COMMAND_ID 0x03    // Ver.: always
#define ZCL_ACK_TRANSFER_DATA_CLIENT_TO_SERVER_COMMAND_ID 0x04    // Ver.: always
#define ZCL_READY_DATA_CLIENT_TO_SERVER_COMMAND_ID 0x05    // Ver.: always


// Command types for cluster: Prepayment
// Cluster specification level: 1.1, 105638r09

// Server to client
#define ZCL_SUPPLY_STATUS_RESPONSE_COMMAND_ID 0x00    // Ver.: always

// Client to server
#define ZCL_SELECT_AVAILABLE_EMERGENCY_CREDIT_COMMAND_ID 0x00    // Ver.: always
#define ZCL_CHANGE_SUPPLY_COMMAND_ID 0x01    // Ver.: always


// Command types for cluster: Key establishment
// Cluster specification level: 1.1, 105638r09

// Server to client
#define ZCL_INITIATE_KEY_ESTABLISHMENT_RESPONSE_COMMAND_ID 0x00    // Ver.: always
#define ZCL_EPHEMERAL_DATA_RESPONSE_COMMAND_ID 0x01    // Ver.: always
#define ZCL_CONFIRM_KEY_DATA_RESPONSE_COMMAND_ID 0x02    // Ver.: always

// Client to server
#define ZCL_INITIATE_KEY_ESTABLISHMENT_REQUEST_COMMAND_ID 0x00    // Ver.: always
#define ZCL_EPHEMERAL_DATA_REQUEST_COMMAND_ID 0x01    // Ver.: always
```

```
#define ZCL_CONFIRM_KEY_DATA_REQUEST_COMMAND_ID 0x02    // Ver.: always

// Either direction
#define ZCL_TERMINATE_KEY_ESTABLISHMENT_COMMAND_ID 0x03    // Ver.: always


// Command types for cluster: Sample Mfg Specific Cluster
// Cluster specification level: UNKNOWN

// Client to server
#define ZCL_COMMAND_ONE_COMMAND_ID 0x00    // Ver.: always mfgCode: 0x1002


#endif // __EMBER_AF_COMMAND_ID__
```

core-cli.h

Go to the documentation of this file.

```
// ****************************************************************************
// * core-cli.h
// *
// * Core CLI commands used by all applications regardless of profile.
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.             *80*
// ****************************************************************************

extern EmberCommandEntry debugPrintCommands[];
void emAfCliCountersCommand(void);
void emAfCliHelpCommand(void);
void emAfCliInfoCommand(void);
void emAfCliResetCommand(void);
void emAfCliEchoCommand(void);

#define EMBER_AF_CORE_COMMANDS                          \
  {"counters", emAfCliCountersCommand, ""},             \
  {"help", emAfCliHelpCommand, ""},                     \
  {"reset", emAfCliResetCommand, ""},                   \
  {"echo",  emAfCliEchoCommand, "u" },

#define EMBER_AF_INFO_COMMAND {"info", emAfCliInfoCommand, ""},
```

custom-cli.h

Go to the documentation of this file.

```
#ifndef CUSTOM_CLI_H
#define CUSTOM_CLI_H
#ifdef EMBER_AF_ENABLE_CUSTOM_COMMANDS
extern EmberCommandEntry customCommands[];
#define CUSTOM_COMMANDS   {"custom", NULL, (PGM_P)customCommands},
#else
#define CUSTOM_COMMANDS
#endif
#endif
```

debug-printing-test.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generateDebugPrintingTest()
//

// This is the test header, that enables all printing
// Enclosing macro to prevent multiple inclusion
#ifndef __EMBER_AF_PRINTING_TEST__
#define __EMBER_AF_PRINTING_TEST__


#define EMBER_AF_PRINT_ENABLE
#define EMBER_AF_PRINT_BASIC_CLUSTER 0x0001
#define EMBER_AF_PRINT_POWER_CONFIG_CLUSTER 0x0002
#define EMBER_AF_PRINT_DEVICE_TEMP_CLUSTER 0x0004
#define EMBER_AF_PRINT_IDENTIFY_CLUSTER 0x0008
#define EMBER_AF_PRINT_GROUPS_CLUSTER 0x0010
#define EMBER_AF_PRINT_SCENES_CLUSTER 0x0020
#define EMBER_AF_PRINT_ON_OFF_CLUSTER 0x0040
#define EMBER_AF_PRINT_ON_OFF_SWITCH_CONFIG_CLUSTER 0x0080
#define EMBER_AF_PRINT_LEVEL_CONTROL_CLUSTER 0x0101
#define EMBER_AF_PRINT_ALARM_CLUSTER 0x0102
#define EMBER_AF_PRINT_TIME_CLUSTER 0x0104
#define EMBER_AF_PRINT_RSSI_LOCATION_CLUSTER 0x0108
#define EMBER_AF_PRINT_BINARY_INPUT_BASIC_CLUSTER 0x0110
#define EMBER_AF_PRINT_COMMISSIONING_CLUSTER 0x0120
#define EMBER_AF_PRINT_SHADE_CONFIG_CLUSTER 0x0140
#define EMBER_AF_PRINT_PUMP_CONFIG_CONTROL_CLUSTER 0x0180
#define EMBER_AF_PRINT_THERMOSTAT_CLUSTER 0x0201
#define EMBER_AF_PRINT_FAN_CONTROL_CLUSTER 0x0202
#define EMBER_AF_PRINT_DEHUMID_CONTROL_CLUSTER 0x0204
#define EMBER_AF_PRINT_THERMOSTAT_UI_CONFIG_CLUSTER 0x0208
#define EMBER_AF_PRINT_COLOR_CONTROL_CLUSTER 0x0210
#define EMBER_AF_PRINT_BALLAST_CONFIGURATION_CLUSTER 0x0220
#define EMBER_AF_PRINT_ILLUM_MEASUREMENT_CLUSTER 0x0240
#define EMBER_AF_PRINT_ILLUM_LEVEL_SENSING_CLUSTER 0x0280
#define EMBER_AF_PRINT_TEMP_MEASUREMENT_CLUSTER 0x0301
#define EMBER_AF_PRINT_PRESSURE_MEASUREMENT_CLUSTER 0x0302
#define EMBER_AF_PRINT_FLOW_MEASUREMENT_CLUSTER 0x0304
#define EMBER_AF_PRINT_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER 0x0308
#define EMBER_AF_PRINT_OCCUPANCY_SENSING_CLUSTER 0x0310
```

```
#define EMBER_AF_PRINT_IAS_ZONE_CLUSTER 0x0320
#define EMBER_AF_PRINT_IAS_ACE_CLUSTER 0x0340
#define EMBER_AF_PRINT_IAS_WD_CLUSTER 0x0380
#define EMBER_AF_PRINT_DOOR_LOCK_CLUSTER 0x0401
#define EMBER_AF_PRINT_WINDOW_COVERING_CLUSTER 0x0402
#define EMBER_AF_PRINT_GENERIC_TUNNEL_CLUSTER 0x0404
#define EMBER_AF_PRINT_BACNET_PROTOCOL_TUNNEL_CLUSTER 0x0408
#define EMBER_AF_PRINT_OTA_BOOTLOAD_CLUSTER 0x0410
#define EMBER_AF_PRINT_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER 0x0420
#define EMBER_AF_PRINT_SIMPLE_METERING_CLUSTER 0x0440
#define EMBER_AF_PRINT_PRICE_CLUSTER 0x0480
#define EMBER_AF_PRINT_MESSAGING_CLUSTER 0x0501
#define EMBER_AF_PRINT_TUNNELING_CLUSTER 0x0502
#define EMBER_AF_PRINT_PREPAYMENT_CLUSTER 0x0504
#define EMBER_AF_PRINT_KEY_ESTABLISHMENT_CLUSTER 0x0508
#define EMBER_AF_PRINT_SAMPLE_MFG_SPECIFIC_CLUSTER 0x0510
#define EMBER_AF_PRINT_CORE 0x0520
#define EMBER_AF_PRINT_DEBUG 0x0540
#define EMBER_AF_PRINT_APP 0x0580
#define EMBER_AF_PRINT_SECURITY 0x0601
#define EMBER_AF_PRINT_ATTRIBUTES 0x0602
#define EMBER_AF_PRINT_REPORTING 0x0604
#define EMBER_AF_PRINT_SERVICE_DISCOVERY 0x0608
#define EMBER_AF_PRINT_REGISTRATION 0x0610
#define EMBER_AF_PRINT_ZDO 0x0620
#define EMBER_AF_PRINT_CUSTOM1 0x0640
#define EMBER_AF_PRINT_CUSTOM2 0x0680
#define EMBER_AF_PRINT_CUSTOM3 0x0701


#define EMBER_AF_PRINT_OUTPUT 1


#define EMBER_AF_PRINT_NAMES { \
  "Basic",\
  "Power Configuration",\
  "Device Temperature Configuration",\
  "Identify",\
  "Groups",\
  "Scenes",\
  "On/off",\
  "On/off Switch Configuration",\
  "Level Control",\
  "Alarms",\
  "Time",\
  "RSSI Location",\
  "Binary Input (Basic)",\
  "Commissioning",\
  "Shade Configuration",\
  "Pump Configuration and Control",\
```

```
"Thermostat",\
"Fan Control",\
"Dehumidification Control",\
"Thermostat User Interface Configuration",\
"Color Control",\
"Ballast Configuration",\
"Illuminance Measurement",\
"Illuminance Level Sensing",\
"Temperature Measurement",\
"Pressure Measurement",\
"Flow Measurement",\
"Relative Humidity Measurement",\
"Occupancy Sensing",\
"IAS Zone",\
"IAS ACE",\
"IAS WD",\
"Door Lock",\
"Window Covering",\
"Generic Tunnel",\
"BACnet Protocol Tunnel",\
"Over the Air Bootloading Cluster",\
"Demand Response and Load Control",\
"Simple Metering",\
"Price",\
"Messaging",\
"Tunneling",\
"Prepayment",\
"Key establishment",\
"Sample Mfg Specific Cluster",\
"Core",\
"Debug",\
"Application",\
"Security",\
"Attributes",\
"Reporting",\
"Service discovery",\
"Registration",\
"ZDO (ZigBee Device Object)",\
"Custom messages (1)",\
"Custom messages (2)",\
"Custom messages (3)"\
}


#define EMBER_AF_PRINT_NAME_NUMBER 57
#define EMBER_AF_PRINT_BITS { 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF }


#endif // __EMBER_AF_PRINTING_TEST__
```

debug-printing.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generateDebugPrinting()
//

// Printing macros for cluster: Basic
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_BASIC_CLUSTER)
#define emberAfBasicClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_BASIC_CLUSTER, __VA_ARGS__)
#define emberAfBasicClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_BASIC_CLUSTER,  __VA_ARGS__)
#define emberAfBasicClusterFlush() emberAfFlush(EMBER_AF_PRINT_BASIC_CLUSTER)
#define emberAfBasicClusterDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_BASIC_CLUSTER) ) { x; }
#define emberAfBasicClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_BASIC_CLUSTER, (buffer), (len), (withSpace))
#define emberAfBasicClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_BASIC_CLUSTER, (buffer))
#else
#define emberAfBasicClusterPrint(...)
#define emberAfBasicClusterPrintln(...)
#define emberAfBasicClusterFlush()
#define emberAfBasicClusterDebugExec(x)
#define emberAfBasicClusterPrintBuffer(buffer, len, withSpace)
#define emberAfBasicClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_BASIC_CLUSTER)


// Printing macros for cluster: Power Configuration
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_POWER_CONFIG_CLUSTER)
#define emberAfPowerConfigClusterPrint(...) emberAfPrint
(EMBER_AF_PRINT_POWER_CONFIG_CLUSTER, __VA_ARGS__)
#define emberAfPowerConfigClusterPrintln(...) emberAfPrintln
(EMBER_AF_PRINT_POWER_CONFIG_CLUSTER,  __VA_ARGS__)
#define emberAfPowerConfigClusterFlush() emberAfFlush(EMBER_AF_PRINT_POWER_CONFIG_CLUSTER)
#define emberAfPowerConfigClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_POWER_CONFIG_CLUSTER) ) { x; }
#define emberAfPowerConfigClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_POWER_CONFIG_CLUSTER, (buffer), (len), (withSpace))
#define emberAfPowerConfigClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_POWER_CONFIG_CLUSTER, (buffer))
#else
#define emberAfPowerConfigClusterPrint(...)
#define emberAfPowerConfigClusterPrintln(...)
#define emberAfPowerConfigClusterFlush()
#define emberAfPowerConfigClusterDebugExec(x)
#define emberAfPowerConfigClusterPrintBuffer(buffer, len, withSpace)
#define emberAfPowerConfigClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_POWER_CONFIG_CLUSTER)


// Printing macros for cluster: Device Temperature Configuration
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_DEVICE_TEMP_CLUSTER)
```

```
#define emberAfDeviceTempClusterPrint(...) emberAfPrint
(EMBER_AF_PRINT_DEVICE_TEMP_CLUSTER, __VA_ARGS__)
#define emberAfDeviceTempClusterPrintln(...) emberAfPrintln
(EMBER_AF_PRINT_DEVICE_TEMP_CLUSTER,  __VA_ARGS__)
#define emberAfDeviceTempClusterFlush() emberAfFlush(EMBER_AF_PRINT_DEVICE_TEMP_CLUSTER)
#define emberAfDeviceTempClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_DEVICE_TEMP_CLUSTER) ) { x; }
#define emberAfDeviceTempClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_DEVICE_TEMP_CLUSTER, (buffer), (len), (withSpace))
#define emberAfDeviceTempClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_DEVICE_TEMP_CLUSTER, (buffer))
#else
#define emberAfDeviceTempClusterPrint(...)
#define emberAfDeviceTempClusterPrintln(...)
#define emberAfDeviceTempClusterFlush()
#define emberAfDeviceTempClusterDebugExec(x)
#define emberAfDeviceTempClusterPrintBuffer(buffer, len, withSpace)
#define emberAfDeviceTempClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_DEVICE_TEMP_CLUSTER)


// Printing macros for cluster: Identify
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_IDENTIFY_CLUSTER)
#define emberAfIdentifyClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_IDENTIFY_CLUSTER, __VA_ARGS__)
#define emberAfIdentifyClusterPrintln(...) emberAfPrintln
(EMBER_AF_PRINT_IDENTIFY_CLUSTER,  __VA_ARGS__)
#define emberAfIdentifyClusterFlush() emberAfFlush(EMBER_AF_PRINT_IDENTIFY_CLUSTER)
#define emberAfIdentifyClusterDebugExec(x) if ( emberAfPrintEnabled
(EMBER_AF_PRINT_IDENTIFY_CLUSTER) ) { x; }
#define emberAfIdentifyClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_IDENTIFY_CLUSTER, (buffer), (len), (withSpace))
#define emberAfIdentifyClusterPrintString(buffer) emberAfPrintString
(EMBER_AF_PRINT_IDENTIFY_CLUSTER, (buffer))
#else
#define emberAfIdentifyClusterPrint(...)
#define emberAfIdentifyClusterPrintln(...)
#define emberAfIdentifyClusterFlush()
#define emberAfIdentifyClusterDebugExec(x)
#define emberAfIdentifyClusterPrintBuffer(buffer, len, withSpace)
#define emberAfIdentifyClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_IDENTIFY_CLUSTER)


// Printing macros for cluster: Groups
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_GROUPS_CLUSTER)
#define emberAfGroupsClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_GROUPS_CLUSTER, __VA_ARGS__)
#define emberAfGroupsClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_GROUPS_CLUSTER,  __VA_ARGS__)
#define emberAfGroupsClusterFlush() emberAfFlush(EMBER_AF_PRINT_GROUPS_CLUSTER)
#define emberAfGroupsClusterDebugExec(x) if ( emberAfPrintEnabled
(EMBER_AF_PRINT_GROUPS_CLUSTER) ) { x; }
#define emberAfGroupsClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_GROUPS_CLUSTER, (buffer), (len), (withSpace))
#define emberAfGroupsClusterPrintString(buffer) emberAfPrintString
(EMBER_AF_PRINT_GROUPS_CLUSTER, (buffer))
#else
#define emberAfGroupsClusterPrint(...)
#define emberAfGroupsClusterPrintln(...)
#define emberAfGroupsClusterFlush()
```

```
#define emberAfGroupsClusterDebugExec(x)
#define emberAfGroupsClusterPrintBuffer(buffer, len, withSpace)
#define emberAfGroupsClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_GROUPS_CLUSTER)


// Printing macros for cluster: Scenes
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_SCENES_CLUSTER)
#define emberAfScenesClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_SCENES_CLUSTER, __VA_ARGS__)
#define emberAfScenesClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_SCENES_CLUSTER,  __VA_ARGS__)
#define emberAfScenesClusterFlush() emberAfFlush(EMBER_AF_PRINT_SCENES_CLUSTER)
#define emberAfScenesClusterDebugExec(x) if ( emberAfPrintEnabled
(EMBER_AF_PRINT_SCENES_CLUSTER) ) { x; }
#define emberAfScenesClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_SCENES_CLUSTER, (buffer), (len), (withSpace))
#define emberAfScenesClusterPrintString(buffer) emberAfPrintString
(EMBER_AF_PRINT_SCENES_CLUSTER, (buffer))
#else
#define emberAfScenesClusterPrint(...)
#define emberAfScenesClusterPrintln(...)
#define emberAfScenesClusterFlush()
#define emberAfScenesClusterDebugExec(x)
#define emberAfScenesClusterPrintBuffer(buffer, len, withSpace)
#define emberAfScenesClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_SCENES_CLUSTER)


// Printing macros for cluster: On/off
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_ON_OFF_CLUSTER)
#define emberAfOnOffClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_ON_OFF_CLUSTER, __VA_ARGS__)
#define emberAfOnOffClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_ON_OFF_CLUSTER,  __VA_ARGS__)
#define emberAfOnOffClusterFlush() emberAfFlush(EMBER_AF_PRINT_ON_OFF_CLUSTER)
#define emberAfOnOffClusterDebugExec(x) if ( emberAfPrintEnabled
(EMBER_AF_PRINT_ON_OFF_CLUSTER) ) { x; }
#define emberAfOnOffClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_ON_OFF_CLUSTER, (buffer), (len), (withSpace))
#define emberAfOnOffClusterPrintString(buffer) emberAfPrintString
(EMBER_AF_PRINT_ON_OFF_CLUSTER, (buffer))
#else
#define emberAfOnOffClusterPrint(...)
#define emberAfOnOffClusterPrintln(...)
#define emberAfOnOffClusterFlush()
#define emberAfOnOffClusterDebugExec(x)
#define emberAfOnOffClusterPrintBuffer(buffer, len, withSpace)
#define emberAfOnOffClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_ON_OFF_CLUSTER)


// Printing macros for cluster: On/off Switch Configuration
#if defined(EMBER_AF_PRINT_ENABLE) &&
defined(EMBER_AF_PRINT_ON_OFF_SWITCH_CONFIG_CLUSTER)
#define emberAfOnOffSwitchConfigClusterPrint(...)
emberAfPrint(EMBER_AF_PRINT_ON_OFF_SWITCH_CONFIG_CLUSTER, __VA_ARGS__)
#define emberAfOnOffSwitchConfigClusterPrintln(...)
emberAfPrintln(EMBER_AF_PRINT_ON_OFF_SWITCH_CONFIG_CLUSTER,  __VA_ARGS__)
#define emberAfOnOffSwitchConfigClusterFlush()
emberAfFlush(EMBER_AF_PRINT_ON_OFF_SWITCH_CONFIG_CLUSTER)
#define emberAfOnOffSwitchConfigClusterDebugExec(x) if
```

```
( emberAfPrintEnabled(EMBER_AF_PRINT_ON_OFF_SWITCH_CONFIG_CLUSTER) ) { x; }
#define emberAfOnOffSwitchConfigClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_ON_OFF_SWITCH_CONFIG_CLUSTER, (buffer), (len), (withSpace))
#define emberAfOnOffSwitchConfigClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_ON_OFF_SWITCH_CONFIG_CLUSTER, (buffer))
#else
#define emberAfOnOffSwitchConfigClusterPrint(...)
#define emberAfOnOffSwitchConfigClusterPrintln(...)
#define emberAfOnOffSwitchConfigClusterFlush()
#define emberAfOnOffSwitchConfigClusterDebugExec(x)
#define emberAfOnOffSwitchConfigClusterPrintBuffer(buffer, len, withSpace)
#define emberAfOnOffSwitchConfigClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_ON_OFF_SWITCH_CONFIG_CLUSTER)


// Printing macros for cluster: Level Control
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_LEVEL_CONTROL_CLUSTER)
#define emberAfLevelControlClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_LEVEL_CONTROL_CLUSTER, __VA_ARGS__)
#define emberAfLevelControlClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_LEVEL_CONTROL_CLUSTER, __VA_ARGS__)
#define emberAfLevelControlClusterFlush() emberAfFlush(EMBER_AF_PRINT_LEVEL_CONTROL_CLUSTER)
#define emberAfLevelControlClusterDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_LEVEL_CONTROL_CLUSTER) ) { x; }
#define emberAfLevelControlClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_LEVEL_CONTROL_CLUSTER, (buffer), (len), (withSpace))
#define emberAfLevelControlClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_LEVEL_CONTROL_CLUSTER, (buffer))
#else
#define emberAfLevelControlClusterPrint(...)
#define emberAfLevelControlClusterPrintln(...)
#define emberAfLevelControlClusterFlush()
#define emberAfLevelControlClusterDebugExec(x)
#define emberAfLevelControlClusterPrintBuffer(buffer, len, withSpace)
#define emberAfLevelControlClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_LEVEL_CONTROL_CLUSTER)


// Printing macros for cluster: Alarms
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_ALARM_CLUSTER)
#define emberAfAlarmClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_ALARM_CLUSTER, __VA_ARGS__)
#define emberAfAlarmClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_ALARM_CLUSTER, __VA_ARGS__)
#define emberAfAlarmClusterFlush() emberAfFlush(EMBER_AF_PRINT_ALARM_CLUSTER)
#define emberAfAlarmClusterDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_ALARM_CLUSTER) ) { x; }
#define emberAfAlarmClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_ALARM_CLUSTER, (buffer), (len), (withSpace))
#define emberAfAlarmClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_ALARM_CLUSTER, (buffer))
#else
#define emberAfAlarmClusterPrint(...)
#define emberAfAlarmClusterPrintln(...)
#define emberAfAlarmClusterFlush()
#define emberAfAlarmClusterDebugExec(x)
#define emberAfAlarmClusterPrintBuffer(buffer, len, withSpace)
#define emberAfAlarmClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_ALARM_CLUSTER)
```

```
// Printing macros for cluster: Time
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_TIME_CLUSTER)
#define emberAfTimeClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_TIME_CLUSTER, __VA_ARGS__)
#define emberAfTimeClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_TIME_CLUSTER,  __VA_ARGS__)
#define emberAfTimeClusterFlush() emberAfFlush(EMBER_AF_PRINT_TIME_CLUSTER)
#define emberAfTimeClusterDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_TIME_CLUSTER) ) { x; }
#define emberAfTimeClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_TIME_CLUSTER, (buffer), (len), (withSpace))
#define emberAfTimeClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_TIME_CLUSTER, (buffer))
#else
#define emberAfTimeClusterPrint(...)
#define emberAfTimeClusterPrintln(...)
#define emberAfTimeClusterFlush()
#define emberAfTimeClusterDebugExec(x)
#define emberAfTimeClusterPrintBuffer(buffer, len, withSpace)
#define emberAfTimeClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_TIME_CLUSTER)


// Printing macros for cluster: RSSI Location
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_RSSI_LOCATION_CLUSTER)
#define emberAfRssiLocationClusterPrint(...) emberAfPrint
(EMBER_AF_PRINT_RSSI_LOCATION_CLUSTER, __VA_ARGS__)
#define emberAfRssiLocationClusterPrintln(...) emberAfPrintln
(EMBER_AF_PRINT_RSSI_LOCATION_CLUSTER,  __VA_ARGS__)
#define emberAfRssiLocationClusterFlush() emberAfFlush(EMBER_AF_PRINT_RSSI_LOCATION_CLUSTER)
#define emberAfRssiLocationClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_RSSI_LOCATION_CLUSTER) ) { x; }
#define emberAfRssiLocationClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_RSSI_LOCATION_CLUSTER, (buffer), (len), (withSpace))
#define emberAfRssiLocationClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_RSSI_LOCATION_CLUSTER, (buffer))
#else
#define emberAfRssiLocationClusterPrint(...)
#define emberAfRssiLocationClusterPrintln(...)
#define emberAfRssiLocationClusterFlush()
#define emberAfRssiLocationClusterDebugExec(x)
#define emberAfRssiLocationClusterPrintBuffer(buffer, len, withSpace)
#define emberAfRssiLocationClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_RSSI_LOCATION_CLUSTER)


// Printing macros for cluster: Binary Input (Basic)
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_BINARY_INPUT_BASIC_CLUSTER)
#define emberAfBinaryInputBasicClusterPrint(...)
emberAfPrint(EMBER_AF_PRINT_BINARY_INPUT_BASIC_CLUSTER, __VA_ARGS__)
#define emberAfBinaryInputBasicClusterPrintln(...)
emberAfPrintln(EMBER_AF_PRINT_BINARY_INPUT_BASIC_CLUSTER,  __VA_ARGS__)
#define emberAfBinaryInputBasicClusterFlush()
emberAfFlush(EMBER_AF_PRINT_BINARY_INPUT_BASIC_CLUSTER)
#define emberAfBinaryInputBasicClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_BINARY_INPUT_BASIC_CLUSTER) ) { x; }
#define emberAfBinaryInputBasicClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_BINARY_INPUT_BASIC_CLUSTER, (buffer), (len), (withSpace))
#define emberAfBinaryInputBasicClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_BINARY_INPUT_BASIC_CLUSTER, (buffer))
#else
```

```
#define emberAfBinaryInputBasicClusterPrint(...)
#define emberAfBinaryInputBasicClusterPrintln(...)
#define emberAfBinaryInputBasicClusterFlush()
#define emberAfBinaryInputBasicClusterDebugExec(x)
#define emberAfBinaryInputBasicClusterPrintBuffer(buffer, len, withSpace)
#define emberAfBinaryInputBasicClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_BINARY_INPUT_BASIC_CLUSTER)


// Printing macros for cluster: Commissioning
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_COMMISSIONING_CLUSTER)
#define emberAfCommissioningClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_COMMISSIONING_CLUSTER, __VA_ARGS__)
#define emberAfCommissioningClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_COMMISSIONING_CLUSTER, __VA_ARGS__)
#define emberAfCommissioningClusterFlush() emberAfFlush(EMBER_AF_PRINT_COMMISSIONING_CLUSTER)
#define emberAfCommissioningClusterDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_COMMISSIONING_CLUSTER) ) { x; }
#define emberAfCommissioningClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_COMMISSIONING_CLUSTER, (buffer), (len), (withSpace))
#define emberAfCommissioningClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_COMMISSIONING_CLUSTER, (buffer))
#else
#define emberAfCommissioningClusterPrint(...)
#define emberAfCommissioningClusterPrintln(...)
#define emberAfCommissioningClusterFlush()
#define emberAfCommissioningClusterDebugExec(x)
#define emberAfCommissioningClusterPrintBuffer(buffer, len, withSpace)
#define emberAfCommissioningClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_COMMISSIONING_CLUSTER)


// Printing macros for cluster: Shade Configuration
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_SHADE_CONFIG_CLUSTER)
#define emberAfShadeConfigClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_SHADE_CONFIG_CLUSTER, __VA_ARGS__)
#define emberAfShadeConfigClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_SHADE_CONFIG_CLUSTER, __VA_ARGS__)
#define emberAfShadeConfigClusterFlush() emberAfFlush(EMBER_AF_PRINT_SHADE_CONFIG_CLUSTER)
#define emberAfShadeConfigClusterDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_SHADE_CONFIG_CLUSTER) ) { x; }
#define emberAfShadeConfigClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_SHADE_CONFIG_CLUSTER, (buffer), (len), (withSpace))
#define emberAfShadeConfigClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_SHADE_CONFIG_CLUSTER, (buffer))
#else
#define emberAfShadeConfigClusterPrint(...)
#define emberAfShadeConfigClusterPrintln(...)
#define emberAfShadeConfigClusterFlush()
#define emberAfShadeConfigClusterDebugExec(x)
#define emberAfShadeConfigClusterPrintBuffer(buffer, len, withSpace)
#define emberAfShadeConfigClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_SHADE_CONFIG_CLUSTER)


// Printing macros for cluster: Pump Configuration and Control
#if defined(EMBER_AF_PRINT_ENABLE) &&
```

```
defined(EMBER_AF_PRINT_PUMP_CONFIG_CONTROL_CLUSTER)
00281 #define emberAfPumpConfigControlClusterPrint(...)
emberAfPrint(EMBER_AF_PRINT_PUMP_CONFIG_CONTROL_CLUSTER, __VA_ARGS__)
00282 #define emberAfPumpConfigControlClusterPrintln(...)
emberAfPrintln(EMBER_AF_PRINT_PUMP_CONFIG_CONTROL_CLUSTER,  __VA_ARGS__)
00283 #define emberAfPumpConfigControlClusterFlush()
emberAfFlush(EMBER_AF_PRINT_PUMP_CONFIG_CONTROL_CLUSTER)
00284 #define emberAfPumpConfigControlClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_PUMP_CONFIG_CONTROL_CLUSTER) ) { x; }
00285 #define emberAfPumpConfigControlClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_PUMP_CONFIG_CONTROL_CLUSTER, (buffer), (len), (withSpace))
00286 #define emberAfPumpConfigControlClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_PUMP_CONFIG_CONTROL_CLUSTER, (buffer))
00287 #else
00288 #define emberAfPumpConfigControlClusterPrint(...)
00289 #define emberAfPumpConfigControlClusterPrintln(...)
00290 #define emberAfPumpConfigControlClusterFlush()
00291 #define emberAfPumpConfigControlClusterDebugExec(x)
00292 #define emberAfPumpConfigControlClusterPrintBuffer(buffer, len, withSpace)
00293 #define emberAfPumpConfigControlClusterPrintString(buffer)
00294 #endif //defined(EMBER_AF_PRINT_ENABLE) &&
defined(EMBER_AF_PRINT_PUMP_CONFIG_CONTROL_CLUSTER)
00295
00296
00297 // Printing macros for cluster: Thermostat
00298 #if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_THERMOSTAT_CLUSTER)
00299 #define emberAfThermostatClusterPrint(...) emberAfPrint
(EMBER_AF_PRINT_THERMOSTAT_CLUSTER, __VA_ARGS__)
00300 #define emberAfThermostatClusterPrintln(...) emberAfPrintln
(EMBER_AF_PRINT_THERMOSTAT_CLUSTER,  __VA_ARGS__)
00301 #define emberAfThermostatClusterFlush() emberAfFlush(EMBER_AF_PRINT_THERMOSTAT_CLUSTER)
00302 #define emberAfThermostatClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_THERMOSTAT_CLUSTER) ) { x; }
00303 #define emberAfThermostatClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_THERMOSTAT_CLUSTER, (buffer), (len), (withSpace))
00304 #define emberAfThermostatClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_THERMOSTAT_CLUSTER, (buffer))
00305 #else
00306 #define emberAfThermostatClusterPrint(...)
00307 #define emberAfThermostatClusterPrintln(...)
00308 #define emberAfThermostatClusterFlush()
00309 #define emberAfThermostatClusterDebugExec(x)
00310 #define emberAfThermostatClusterPrintBuffer(buffer, len, withSpace)
00311 #define emberAfThermostatClusterPrintString(buffer)
00312 #endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_THERMOSTAT_CLUSTER)
00313
00314
00315 // Printing macros for cluster: Fan Control
00316 #if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_FAN_CONTROL_CLUSTER)
00317 #define emberAfFanControlClusterPrint(...) emberAfPrint
(EMBER_AF_PRINT_FAN_CONTROL_CLUSTER, __VA_ARGS__)
00318 #define emberAfFanControlClusterPrintln(...) emberAfPrintln
(EMBER_AF_PRINT_FAN_CONTROL_CLUSTER,  __VA_ARGS__)
00319 #define emberAfFanControlClusterFlush() emberAfFlush(EMBER_AF_PRINT_FAN_CONTROL_CLUSTER)
00320 #define emberAfFanControlClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_FAN_CONTROL_CLUSTER) ) { x; }
00321 #define emberAfFanControlClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_FAN_CONTROL_CLUSTER, (buffer), (len), (withSpace))
00322 #define emberAfFanControlClusterPrintString(buffer)
```

```
emberAfPrintString(EMBER_AF_PRINT_FAN_CONTROL_CLUSTER, (buffer))
#else
#define emberAfFanControlClusterPrint(...)
#define emberAfFanControlClusterPrintln(...)
#define emberAfFanControlClusterFlush()
#define emberAfFanControlClusterDebugExec(x)
#define emberAfFanControlClusterPrintBuffer(buffer, len, withSpace)
#define emberAfFanControlClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_FAN_CONTROL_CLUSTER)


// Printing macros for cluster: Dehumidification Control
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_DEHUMID_CONTROL_CLUSTER)
#define emberAfDehumidControlClusterPrint(...)
emberAfPrint(EMBER_AF_PRINT_DEHUMID_CONTROL_CLUSTER, __VA_ARGS__)
#define emberAfDehumidControlClusterPrintln(...)
emberAfPrintln(EMBER_AF_PRINT_DEHUMID_CONTROL_CLUSTER,  __VA_ARGS__)
#define emberAfDehumidControlClusterFlush()
emberAfFlush(EMBER_AF_PRINT_DEHUMID_CONTROL_CLUSTER)
#define emberAfDehumidControlClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_DEHUMID_CONTROL_CLUSTER) ) { x; }
#define emberAfDehumidControlClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_DEHUMID_CONTROL_CLUSTER, (buffer), (len), (withSpace))
#define emberAfDehumidControlClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_DEHUMID_CONTROL_CLUSTER, (buffer))
#else
#define emberAfDehumidControlClusterPrint(...)
#define emberAfDehumidControlClusterPrintln(...)
#define emberAfDehumidControlClusterFlush()
#define emberAfDehumidControlClusterDebugExec(x)
#define emberAfDehumidControlClusterPrintBuffer(buffer, len, withSpace)
#define emberAfDehumidControlClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_DEHUMID_CONTROL_CLUSTER)


// Printing macros for cluster: Thermostat User Interface Configuration
#if defined(EMBER_AF_PRINT_ENABLE) &&
defined(EMBER_AF_PRINT_THERMOSTAT_UI_CONFIG_CLUSTER)
#define emberAfThermostatUiConfigClusterPrint(...)
emberAfPrint(EMBER_AF_PRINT_THERMOSTAT_UI_CONFIG_CLUSTER, __VA_ARGS__)
#define emberAfThermostatUiConfigClusterPrintln(...)
emberAfPrintln(EMBER_AF_PRINT_THERMOSTAT_UI_CONFIG_CLUSTER,  __VA_ARGS__)
#define emberAfThermostatUiConfigClusterFlush()
emberAfFlush(EMBER_AF_PRINT_THERMOSTAT_UI_CONFIG_CLUSTER)
#define emberAfThermostatUiConfigClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_THERMOSTAT_UI_CONFIG_CLUSTER) ) { x; }
#define emberAfThermostatUiConfigClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_THERMOSTAT_UI_CONFIG_CLUSTER, (buffer), (len), (withSpace))
#define emberAfThermostatUiConfigClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_THERMOSTAT_UI_CONFIG_CLUSTER, (buffer))
#else
#define emberAfThermostatUiConfigClusterPrint(...)
#define emberAfThermostatUiConfigClusterPrintln(...)
#define emberAfThermostatUiConfigClusterFlush()
#define emberAfThermostatUiConfigClusterDebugExec(x)
#define emberAfThermostatUiConfigClusterPrintBuffer(buffer, len, withSpace)
#define emberAfThermostatUiConfigClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) &&
```

```
defined(EMBER_AF_PRINT_THERMOSTAT_UI_CONFIG_CLUSTER)
00367
00368
00369 // Printing macros for cluster: Color Control
00370 #if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_COLOR_CONTROL_CLUSTER)
00371 #define emberAfColorControlClusterPrint(...) emberAfPrint
(EMBER_AF_PRINT_COLOR_CONTROL_CLUSTER, __VA_ARGS__)
00372 #define emberAfColorControlClusterPrintln(...) emberAfPrintln
(EMBER_AF_PRINT_COLOR_CONTROL_CLUSTER, __VA_ARGS__)
00373 #define emberAfColorControlClusterFlush() emberAfFlush(EMBER_AF_PRINT_COLOR_CONTROL_CLUSTER)
00374 #define emberAfColorControlClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_COLOR_CONTROL_CLUSTER) ) { x; }
00375 #define emberAfColorControlClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_COLOR_CONTROL_CLUSTER, (buffer), (len), (withSpace))
00376 #define emberAfColorControlClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_COLOR_CONTROL_CLUSTER, (buffer))
00377 #else
00378 #define emberAfColorControlClusterPrint(...)
00379 #define emberAfColorControlClusterPrintln(...)
00380 #define emberAfColorControlClusterFlush()
00381 #define emberAfColorControlClusterDebugExec(x)
00382 #define emberAfColorControlClusterPrintBuffer(buffer, len, withSpace)
00383 #define emberAfColorControlClusterPrintString(buffer)
00384 #endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_COLOR_CONTROL_CLUSTER)
00385
00386
00387 // Printing macros for cluster: Ballast Configuration
00388 #if defined(EMBER_AF_PRINT_ENABLE) &&
defined(EMBER_AF_PRINT_BALLAST_CONFIGURATION_CLUSTER)
00389 #define emberAfBallastConfigurationClusterPrint(...)
emberAfPrint(EMBER_AF_PRINT_BALLAST_CONFIGURATION_CLUSTER, __VA_ARGS__)
00390 #define emberAfBallastConfigurationClusterPrintln(...)
emberAfPrintln(EMBER_AF_PRINT_BALLAST_CONFIGURATION_CLUSTER, __VA_ARGS__)
00391 #define emberAfBallastConfigurationClusterFlush()
emberAfFlush(EMBER_AF_PRINT_BALLAST_CONFIGURATION_CLUSTER)
00392 #define emberAfBallastConfigurationClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_BALLAST_CONFIGURATION_CLUSTER) ) { x; }
00393 #define emberAfBallastConfigurationClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_BALLAST_CONFIGURATION_CLUSTER, (buffer), (len), (withSpace))
00394 #define emberAfBallastConfigurationClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_BALLAST_CONFIGURATION_CLUSTER, (buffer))
00395 #else
00396 #define emberAfBallastConfigurationClusterPrint(...)
00397 #define emberAfBallastConfigurationClusterPrintln(...)
00398 #define emberAfBallastConfigurationClusterFlush()
00399 #define emberAfBallastConfigurationClusterDebugExec(x)
00400 #define emberAfBallastConfigurationClusterPrintBuffer(buffer, len, withSpace)
00401 #define emberAfBallastConfigurationClusterPrintString(buffer)
00402 #endif //defined(EMBER_AF_PRINT_ENABLE) &&
defined(EMBER_AF_PRINT_BALLAST_CONFIGURATION_CLUSTER)
00403
00404
00405 // Printing macros for cluster: Illuminance Measurement
00406 #if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_ILLUM_MEASUREMENT_CLUSTER)
00407 #define emberAfIllumMeasurementClusterPrint(...)
emberAfPrint(EMBER_AF_PRINT_ILLUM_MEASUREMENT_CLUSTER, __VA_ARGS__)
00408 #define emberAfIllumMeasurementClusterPrintln(...)
emberAfPrintln(EMBER_AF_PRINT_ILLUM_MEASUREMENT_CLUSTER, __VA_ARGS__)
00409 #define emberAfIllumMeasurementClusterFlush()
```

```
emberAfFlush(EMBER_AF_PRINT_ILLUM_MEASUREMENT_CLUSTER)
#define emberAfIllumMeasurementClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_ILLUM_MEASUREMENT_CLUSTER) ) { x; }
#define emberAfIllumMeasurementClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_ILLUM_MEASUREMENT_CLUSTER, (buffer), (len), (withSpace))
#define emberAfIllumMeasurementClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_ILLUM_MEASUREMENT_CLUSTER, (buffer))
#else
#define emberAfIllumMeasurementClusterPrint(...)
#define emberAfIllumMeasurementClusterPrintln(...)
#define emberAfIllumMeasurementClusterFlush()
#define emberAfIllumMeasurementClusterDebugExec(x)
#define emberAfIllumMeasurementClusterPrintBuffer(buffer, len, withSpace)
#define emberAfIllumMeasurementClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) &&
defined(EMBER_AF_PRINT_ILLUM_MEASUREMENT_CLUSTER)


// Printing macros for cluster: Illuminance Level Sensing
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_ILLUM_LEVEL_SENSING_CLUSTER)
#define emberAfIllumLevelSensingClusterPrint(...)
emberAfPrint(EMBER_AF_PRINT_ILLUM_LEVEL_SENSING_CLUSTER, __VA_ARGS__)
#define emberAfIllumLevelSensingClusterPrintln(...)
emberAfPrintln(EMBER_AF_PRINT_ILLUM_LEVEL_SENSING_CLUSTER,  __VA_ARGS__)
#define emberAfIllumLevelSensingClusterFlush()
emberAfFlush(EMBER_AF_PRINT_ILLUM_LEVEL_SENSING_CLUSTER)
#define emberAfIllumLevelSensingClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_ILLUM_LEVEL_SENSING_CLUSTER) ) { x; }
#define emberAfIllumLevelSensingClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_ILLUM_LEVEL_SENSING_CLUSTER, (buffer), (len), (withSpace))
#define emberAfIllumLevelSensingClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_ILLUM_LEVEL_SENSING_CLUSTER, (buffer))
#else
#define emberAfIllumLevelSensingClusterPrint(...)
#define emberAfIllumLevelSensingClusterPrintln(...)
#define emberAfIllumLevelSensingClusterFlush()
#define emberAfIllumLevelSensingClusterDebugExec(x)
#define emberAfIllumLevelSensingClusterPrintBuffer(buffer, len, withSpace)
#define emberAfIllumLevelSensingClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) &&
defined(EMBER_AF_PRINT_ILLUM_LEVEL_SENSING_CLUSTER)


// Printing macros for cluster: Temperature Measurement
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_TEMP_MEASUREMENT_CLUSTER)
#define emberAfTempMeasurementClusterPrint(...)
emberAfPrint(EMBER_AF_PRINT_TEMP_MEASUREMENT_CLUSTER, __VA_ARGS__)
#define emberAfTempMeasurementClusterPrintln(...)
emberAfPrintln(EMBER_AF_PRINT_TEMP_MEASUREMENT_CLUSTER,  __VA_ARGS__)
#define emberAfTempMeasurementClusterFlush()
emberAfFlush(EMBER_AF_PRINT_TEMP_MEASUREMENT_CLUSTER)
#define emberAfTempMeasurementClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_TEMP_MEASUREMENT_CLUSTER) ) { x; }
#define emberAfTempMeasurementClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_TEMP_MEASUREMENT_CLUSTER, (buffer), (len), (withSpace))
#define emberAfTempMeasurementClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_TEMP_MEASUREMENT_CLUSTER, (buffer))
#else
#define emberAfTempMeasurementClusterPrint(...)
```

```
#define emberAfTempMeasurementClusterPrintln(...)
#define emberAfTempMeasurementClusterFlush()
#define emberAfTempMeasurementClusterDebugExec(x)
#define emberAfTempMeasurementClusterPrintBuffer(buffer, len, withSpace)
#define emberAfTempMeasurementClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_TEMP_MEASUREMENT_CLUSTER)


// Printing macros for cluster: Pressure Measurement
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_PRESSURE_MEASUREMENT_CLUSTER)
#define emberAfPressureMeasurementClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_PRESSURE_MEASUREMENT_CLUSTER, __VA_ARGS__)
#define emberAfPressureMeasurementClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_PRESSURE_MEASUREMENT_CLUSTER, __VA_ARGS__)
#define emberAfPressureMeasurementClusterFlush() emberAfFlush(EMBER_AF_PRINT_PRESSURE_MEASUREMENT_CLUSTER)
#define emberAfPressureMeasurementClusterDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_PRESSURE_MEASUREMENT_CLUSTER) ) { x; }
#define emberAfPressureMeasurementClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_PRESSURE_MEASUREMENT_CLUSTER, (buffer), (len), (withSpace))
#define emberAfPressureMeasurementClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_PRESSURE_MEASUREMENT_CLUSTER, (buffer))
#else
#define emberAfPressureMeasurementClusterPrint(...)
#define emberAfPressureMeasurementClusterPrintln(...)
#define emberAfPressureMeasurementClusterFlush()
#define emberAfPressureMeasurementClusterDebugExec(x)
#define emberAfPressureMeasurementClusterPrintBuffer(buffer, len, withSpace)
#define emberAfPressureMeasurementClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_PRESSURE_MEASUREMENT_CLUSTER)


// Printing macros for cluster: Flow Measurement
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_FLOW_MEASUREMENT_CLUSTER)
#define emberAfFlowMeasurementClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_FLOW_MEASUREMENT_CLUSTER, __VA_ARGS__)
#define emberAfFlowMeasurementClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_FLOW_MEASUREMENT_CLUSTER, __VA_ARGS__)
#define emberAfFlowMeasurementClusterFlush() emberAfFlush(EMBER_AF_PRINT_FLOW_MEASUREMENT_CLUSTER)
#define emberAfFlowMeasurementClusterDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_FLOW_MEASUREMENT_CLUSTER) ) { x; }
#define emberAfFlowMeasurementClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_FLOW_MEASUREMENT_CLUSTER, (buffer), (len), (withSpace))
#define emberAfFlowMeasurementClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_FLOW_MEASUREMENT_CLUSTER, (buffer))
#else
#define emberAfFlowMeasurementClusterPrint(...)
#define emberAfFlowMeasurementClusterPrintln(...)
#define emberAfFlowMeasurementClusterFlush()
#define emberAfFlowMeasurementClusterDebugExec(x)
#define emberAfFlowMeasurementClusterPrintBuffer(buffer, len, withSpace)
#define emberAfFlowMeasurementClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_FLOW_MEASUREMENT_CLUSTER)
```

```
// Printing macros for cluster: Relative Humidity Measurement
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER)
#define emberAfRelativeHumidityMeasurementClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER, __VA_ARGS__)
#define emberAfRelativeHumidityMeasurementClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER, __VA_ARGS__)
#define emberAfRelativeHumidityMeasurementClusterFlush() emberAfFlush(EMBER_AF_PRINT_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER)
#define emberAfRelativeHumidityMeasurementClusterDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER) ) { x; }
#define emberAfRelativeHumidityMeasurementClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER, (buffer), (len), (withSpace))
#define emberAfRelativeHumidityMeasurementClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER, (buffer))
#else
#define emberAfRelativeHumidityMeasurementClusterPrint(...)
#define emberAfRelativeHumidityMeasurementClusterPrintln(...)
#define emberAfRelativeHumidityMeasurementClusterFlush()
#define emberAfRelativeHumidityMeasurementClusterDebugExec(x)
#define emberAfRelativeHumidityMeasurementClusterPrintBuffer(buffer, len, withSpace)
#define emberAfRelativeHumidityMeasurementClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER)


// Printing macros for cluster: Occupancy Sensing
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_OCCUPANCY_SENSING_CLUSTER)
#define emberAfOccupancySensingClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_OCCUPANCY_SENSING_CLUSTER, __VA_ARGS__)
#define emberAfOccupancySensingClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_OCCUPANCY_SENSING_CLUSTER, __VA_ARGS__)
#define emberAfOccupancySensingClusterFlush() emberAfFlush(EMBER_AF_PRINT_OCCUPANCY_SENSING_CLUSTER)
#define emberAfOccupancySensingClusterDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_OCCUPANCY_SENSING_CLUSTER) ) { x; }
#define emberAfOccupancySensingClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_OCCUPANCY_SENSING_CLUSTER, (buffer), (len), (withSpace))
#define emberAfOccupancySensingClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_OCCUPANCY_SENSING_CLUSTER, (buffer))
#else
#define emberAfOccupancySensingClusterPrint(...)
#define emberAfOccupancySensingClusterPrintln(...)
#define emberAfOccupancySensingClusterFlush()
#define emberAfOccupancySensingClusterDebugExec(x)
#define emberAfOccupancySensingClusterPrintBuffer(buffer, len, withSpace)
#define emberAfOccupancySensingClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_OCCUPANCY_SENSING_CLUSTER)


// Printing macros for cluster: IAS Zone
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_IAS_ZONE_CLUSTER)
#define emberAfIasZoneClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_IAS_ZONE_CLUSTER, __VA_ARGS__)
#define emberAfIasZoneClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_IAS_ZONE_CLUSTER, __VA_ARGS__)
```

```
#define emberAfIasZoneClusterFlush() emberAfFlush(EMBER_AF_PRINT_IAS_ZONE_CLUSTER)
#define emberAfIasZoneClusterDebugExec(x) if ( emberAfPrintEnabled
(EMBER_AF_PRINT_IAS_ZONE_CLUSTER) ) { x; }
#define emberAfIasZoneClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_IAS_ZONE_CLUSTER, (buffer), (len), (withSpace))
#define emberAfIasZoneClusterPrintString(buffer) emberAfPrintString
(EMBER_AF_PRINT_IAS_ZONE_CLUSTER, (buffer))
#else
#define emberAfIasZoneClusterPrint(...)
#define emberAfIasZoneClusterPrintln(...)
#define emberAfIasZoneClusterFlush()
#define emberAfIasZoneClusterDebugExec(x)
#define emberAfIasZoneClusterPrintBuffer(buffer, len, withSpace)
#define emberAfIasZoneClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_IAS_ZONE_CLUSTER)


// Printing macros for cluster: IAS ACE
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_IAS_ACE_CLUSTER)
#define emberAfIasAceClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_IAS_ACE_CLUSTER, __VA_ARGS__)
#define emberAfIasAceClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_IAS_ACE_CLUSTER, __VA_ARGS__)
#define emberAfIasAceClusterFlush() emberAfFlush(EMBER_AF_PRINT_IAS_ACE_CLUSTER)
#define emberAfIasAceClusterDebugExec(x) if ( emberAfPrintEnabled
(EMBER_AF_PRINT_IAS_ACE_CLUSTER) ) { x; }
#define emberAfIasAceClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_IAS_ACE_CLUSTER, (buffer), (len), (withSpace))
#define emberAfIasAceClusterPrintString(buffer) emberAfPrintString
(EMBER_AF_PRINT_IAS_ACE_CLUSTER, (buffer))
#else
#define emberAfIasAceClusterPrint(...)
#define emberAfIasAceClusterPrintln(...)
#define emberAfIasAceClusterFlush()
#define emberAfIasAceClusterDebugExec(x)
#define emberAfIasAceClusterPrintBuffer(buffer, len, withSpace)
#define emberAfIasAceClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_IAS_ACE_CLUSTER)


// Printing macros for cluster: IAS WD
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_IAS_WD_CLUSTER)
#define emberAfIasWdClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_IAS_WD_CLUSTER, __VA_ARGS__)
#define emberAfIasWdClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_IAS_WD_CLUSTER, __VA_ARGS__)
#define emberAfIasWdClusterFlush() emberAfFlush(EMBER_AF_PRINT_IAS_WD_CLUSTER)
#define emberAfIasWdClusterDebugExec(x) if ( emberAfPrintEnabled
(EMBER_AF_PRINT_IAS_WD_CLUSTER) ) { x; }
#define emberAfIasWdClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_IAS_WD_CLUSTER, (buffer), (len), (withSpace))
#define emberAfIasWdClusterPrintString(buffer) emberAfPrintString
(EMBER_AF_PRINT_IAS_WD_CLUSTER, (buffer))
#else
#define emberAfIasWdClusterPrint(...)
#define emberAfIasWdClusterPrintln(...)
#define emberAfIasWdClusterFlush()
#define emberAfIasWdClusterDebugExec(x)
#define emberAfIasWdClusterPrintBuffer(buffer, len, withSpace)
#define emberAfIasWdClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_IAS_WD_CLUSTER)

```

```
// Printing macros for cluster: Door Lock
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_DOOR_LOCK_CLUSTER)
#define emberAfDoorLockClusterPrint(...) emberAfPrint
(EMBER_AF_PRINT_DOOR_LOCK_CLUSTER, __VA_ARGS__)
#define emberAfDoorLockClusterPrintln(...) emberAfPrintln
(EMBER_AF_PRINT_DOOR_LOCK_CLUSTER,  __VA_ARGS__)
#define emberAfDoorLockClusterFlush() emberAfFlush(EMBER_AF_PRINT_DOOR_LOCK_CLUSTER)
#define emberAfDoorLockClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_DOOR_LOCK_CLUSTER) ) { x; }
#define emberAfDoorLockClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_DOOR_LOCK_CLUSTER, (buffer), (len), (withSpace))
#define emberAfDoorLockClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_DOOR_LOCK_CLUSTER, (buffer))
#else
#define emberAfDoorLockClusterPrint(...)
#define emberAfDoorLockClusterPrintln(...)
#define emberAfDoorLockClusterFlush()
#define emberAfDoorLockClusterDebugExec(x)
#define emberAfDoorLockClusterPrintBuffer(buffer, len, withSpace)
#define emberAfDoorLockClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_DOOR_LOCK_CLUSTER)


// Printing macros for cluster: Window Covering
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_WINDOW_COVERING_CLUSTER)
#define emberAfWindowCoveringClusterPrint(...)
emberAfPrint(EMBER_AF_PRINT_WINDOW_COVERING_CLUSTER, __VA_ARGS__)
#define emberAfWindowCoveringClusterPrintln(...)
emberAfPrintln(EMBER_AF_PRINT_WINDOW_COVERING_CLUSTER,  __VA_ARGS__)
#define emberAfWindowCoveringClusterFlush()
emberAfFlush(EMBER_AF_PRINT_WINDOW_COVERING_CLUSTER)
#define emberAfWindowCoveringClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_WINDOW_COVERING_CLUSTER) ) { x; }
#define emberAfWindowCoveringClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_WINDOW_COVERING_CLUSTER, (buffer), (len), (withSpace))
#define emberAfWindowCoveringClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_WINDOW_COVERING_CLUSTER, (buffer))
#else
#define emberAfWindowCoveringClusterPrint(...)
#define emberAfWindowCoveringClusterPrintln(...)
#define emberAfWindowCoveringClusterFlush()
#define emberAfWindowCoveringClusterDebugExec(x)
#define emberAfWindowCoveringClusterPrintBuffer(buffer, len, withSpace)
#define emberAfWindowCoveringClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) &&
defined(EMBER_AF_PRINT_WINDOW_COVERING_CLUSTER)


// Printing macros for cluster: Generic Tunnel
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_GENERIC_TUNNEL_CLUSTER)
#define emberAfGenericTunnelClusterPrint(...) emberAfPrint
(EMBER_AF_PRINT_GENERIC_TUNNEL_CLUSTER, __VA_ARGS__)
#define emberAfGenericTunnelClusterPrintln(...)
emberAfPrintln(EMBER_AF_PRINT_GENERIC_TUNNEL_CLUSTER,  __VA_ARGS__)
#define emberAfGenericTunnelClusterFlush() emberAfFlush(EMBER_AF_PRINT_GENERIC_TUNNEL_CLUSTER)
#define emberAfGenericTunnelClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_GENERIC_TUNNEL_CLUSTER) ) { x; }
#define emberAfGenericTunnelClusterPrintBuffer(buffer, len, withSpace)
```

```
emberAfPrintBuffer(EMBER_AF_PRINT_GENERIC_TUNNEL_CLUSTER, (buffer), (len), (withSpace))
00628 #define emberAfGenericTunnelClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_GENERIC_TUNNEL_CLUSTER, (buffer))
00629 #else
00630 #define emberAfGenericTunnelClusterPrint(...)
00631 #define emberAfGenericTunnelClusterPrintln(...)
00632 #define emberAfGenericTunnelClusterFlush()
00633 #define emberAfGenericTunnelClusterDebugExec(x)
00634 #define emberAfGenericTunnelClusterPrintBuffer(buffer, len, withSpace)
00635 #define emberAfGenericTunnelClusterPrintString(buffer)
00636 #endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_GENERIC_TUNNEL_CLUSTER)
00637
00638
00639 // Printing macros for cluster: BACnet Protocol Tunnel
00640 #if defined(EMBER_AF_PRINT_ENABLE) &&
defined(EMBER_AF_PRINT_BACNET_PROTOCOL_TUNNEL_CLUSTER)
00641 #define emberAfBacnetProtocolTunnelClusterPrint(...)
emberAfPrint(EMBER_AF_PRINT_BACNET_PROTOCOL_TUNNEL_CLUSTER, __VA_ARGS__)
00642 #define emberAfBacnetProtocolTunnelClusterPrintln(...)
emberAfPrintln(EMBER_AF_PRINT_BACNET_PROTOCOL_TUNNEL_CLUSTER, __VA_ARGS__)
00643 #define emberAfBacnetProtocolTunnelClusterFlush()
emberAfFlush(EMBER_AF_PRINT_BACNET_PROTOCOL_TUNNEL_CLUSTER)
00644 #define emberAfBacnetProtocolTunnelClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_BACNET_PROTOCOL_TUNNEL_CLUSTER) ) { x; }
00645 #define emberAfBacnetProtocolTunnelClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_BACNET_PROTOCOL_TUNNEL_CLUSTER, (buffer), (len), (withSpace))
00646 #define emberAfBacnetProtocolTunnelClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_BACNET_PROTOCOL_TUNNEL_CLUSTER, (buffer))
00647 #else
00648 #define emberAfBacnetProtocolTunnelClusterPrint(...)
00649 #define emberAfBacnetProtocolTunnelClusterPrintln(...)
00650 #define emberAfBacnetProtocolTunnelClusterFlush()
00651 #define emberAfBacnetProtocolTunnelClusterDebugExec(x)
00652 #define emberAfBacnetProtocolTunnelClusterPrintBuffer(buffer, len, withSpace)
00653 #define emberAfBacnetProtocolTunnelClusterPrintString(buffer)
00654 #endif //defined(EMBER_AF_PRINT_ENABLE) &&
defined(EMBER_AF_PRINT_BACNET_PROTOCOL_TUNNEL_CLUSTER)
00655
00656
00657 // Printing macros for cluster: Over the Air Bootloading Cluster
00658 #if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_OTA_BOOTLOAD_CLUSTER)
00659 #define emberAfOtaBootloadClusterPrint(...) emberAfPrint
(EMBER_AF_PRINT_OTA_BOOTLOAD_CLUSTER, __VA_ARGS__)
00660 #define emberAfOtaBootloadClusterPrintln(...) emberAfPrintln
(EMBER_AF_PRINT_OTA_BOOTLOAD_CLUSTER, __VA_ARGS__)
00661 #define emberAfOtaBootloadClusterFlush() emberAfFlush(EMBER_AF_PRINT_OTA_BOOTLOAD_CLUSTER)
00662 #define emberAfOtaBootloadClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_OTA_BOOTLOAD_CLUSTER) ) { x; }
00663 #define emberAfOtaBootloadClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_OTA_BOOTLOAD_CLUSTER, (buffer), (len), (withSpace))
00664 #define emberAfOtaBootloadClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_OTA_BOOTLOAD_CLUSTER, (buffer))
00665 #else
00666 #define emberAfOtaBootloadClusterPrint(...)
00667 #define emberAfOtaBootloadClusterPrintln(...)
00668 #define emberAfOtaBootloadClusterFlush()
00669 #define emberAfOtaBootloadClusterDebugExec(x)
00670 #define emberAfOtaBootloadClusterPrintBuffer(buffer, len, withSpace)
```

```
#define emberAfOtaBootloadClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_OTA_BOOTLOAD_CLUSTER)


// Printing macros for cluster: Demand Response and Load Control
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER)
#define emberAfDemandResponseLoadControlClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER, __VA_ARGS__)
#define emberAfDemandResponseLoadControlClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER, __VA_ARGS__)
#define emberAfDemandResponseLoadControlClusterFlush() emberAfFlush(EMBER_AF_PRINT_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER)
#define emberAfDemandResponseLoadControlClusterDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER) ) { x; }
#define emberAfDemandResponseLoadControlClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER, (buffer), (len), (withSpace))
#define emberAfDemandResponseLoadControlClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER, (buffer))
#else
#define emberAfDemandResponseLoadControlClusterPrint(...)
#define emberAfDemandResponseLoadControlClusterPrintln(...)
#define emberAfDemandResponseLoadControlClusterFlush()
#define emberAfDemandResponseLoadControlClusterDebugExec(x)
#define emberAfDemandResponseLoadControlClusterPrintBuffer(buffer, len, withSpace)
#define emberAfDemandResponseLoadControlClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER)


// Printing macros for cluster: Simple Metering
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_SIMPLE_METERING_CLUSTER)
#define emberAfSimpleMeteringClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_SIMPLE_METERING_CLUSTER, __VA_ARGS__)
#define emberAfSimpleMeteringClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_SIMPLE_METERING_CLUSTER, __VA_ARGS__)
#define emberAfSimpleMeteringClusterFlush() emberAfFlush(EMBER_AF_PRINT_SIMPLE_METERING_CLUSTER)
#define emberAfSimpleMeteringClusterDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_SIMPLE_METERING_CLUSTER) ) { x; }
#define emberAfSimpleMeteringClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_SIMPLE_METERING_CLUSTER, (buffer), (len), (withSpace))
#define emberAfSimpleMeteringClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_SIMPLE_METERING_CLUSTER, (buffer))
#else
#define emberAfSimpleMeteringClusterPrint(...)
#define emberAfSimpleMeteringClusterPrintln(...)
#define emberAfSimpleMeteringClusterFlush()
#define emberAfSimpleMeteringClusterDebugExec(x)
#define emberAfSimpleMeteringClusterPrintBuffer(buffer, len, withSpace)
#define emberAfSimpleMeteringClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_SIMPLE_METERING_CLUSTER)


// Printing macros for cluster: Price
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_PRICE_CLUSTER)
#define emberAfPriceClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_PRICE_CLUSTER, __VA_ARGS__)
#define emberAfPriceClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_PRICE_CLUSTER, __VA_ARGS__)
#define emberAfPriceClusterFlush() emberAfFlush(EMBER_AF_PRINT_PRICE_CLUSTER)
#define emberAfPriceClusterDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_PRICE_CLUSTER) ) { x; }
```

```
#define emberAfPriceClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_PRICE_CLUSTER, (buffer), (len), (withSpace))
#define emberAfPriceClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_PRICE_CLUSTER, (buffer))
#else
#define emberAfPriceClusterPrint(...)
#define emberAfPriceClusterPrintln(...)
#define emberAfPriceClusterFlush()
#define emberAfPriceClusterDebugExec(x)
#define emberAfPriceClusterPrintBuffer(buffer, len, withSpace)
#define emberAfPriceClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_PRICE_CLUSTER)


// Printing macros for cluster: Messaging
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_MESSAGING_CLUSTER)
#define emberAfMessagingClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_MESSAGING_CLUSTER, __VA_ARGS__)
#define emberAfMessagingClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_MESSAGING_CLUSTER, __VA_ARGS__)
#define emberAfMessagingClusterFlush() emberAfFlush(EMBER_AF_PRINT_MESSAGING_CLUSTER)
#define emberAfMessagingClusterDebugExec(x) if( emberAfPrintEnabled(EMBER_AF_PRINT_MESSAGING_CLUSTER) ) { x; }
#define emberAfMessagingClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_MESSAGING_CLUSTER, (buffer), (len), (withSpace))
#define emberAfMessagingClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_MESSAGING_CLUSTER, (buffer))
#else
#define emberAfMessagingClusterPrint(...)
#define emberAfMessagingClusterPrintln(...)
#define emberAfMessagingClusterFlush()
#define emberAfMessagingClusterDebugExec(x)
#define emberAfMessagingClusterPrintBuffer(buffer, len, withSpace)
#define emberAfMessagingClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_MESSAGING_CLUSTER)


// Printing macros for cluster: Tunneling
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_TUNNELING_CLUSTER)
#define emberAfTunnelingClusterPrint(...) emberAfPrint(EMBER_AF_PRINT_TUNNELING_CLUSTER, __VA_ARGS__)
#define emberAfTunnelingClusterPrintln(...) emberAfPrintln(EMBER_AF_PRINT_TUNNELING_CLUSTER, __VA_ARGS__)
#define emberAfTunnelingClusterFlush() emberAfFlush(EMBER_AF_PRINT_TUNNELING_CLUSTER)
#define emberAfTunnelingClusterDebugExec(x) if( emberAfPrintEnabled(EMBER_AF_PRINT_TUNNELING_CLUSTER) ) { x; }
#define emberAfTunnelingClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_TUNNELING_CLUSTER, (buffer), (len), (withSpace))
#define emberAfTunnelingClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_TUNNELING_CLUSTER, (buffer))
#else
#define emberAfTunnelingClusterPrint(...)
#define emberAfTunnelingClusterPrintln(...)
#define emberAfTunnelingClusterFlush()
#define emberAfTunnelingClusterDebugExec(x)
#define emberAfTunnelingClusterPrintBuffer(buffer, len, withSpace)
#define emberAfTunnelingClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_TUNNELING_CLUSTER)

```

```

// Printing macros for cluster: Prepayment
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_PREPAYMENT_CLUSTER)
#define emberAfPrepaymentClusterPrint(...) emberAfPrint
(EMBER_AF_PRINT_PREPAYMENT_CLUSTER, __VA_ARGS__)
#define emberAfPrepaymentClusterPrintln(...) emberAfPrintln
(EMBER_AF_PRINT_PREPAYMENT_CLUSTER,  __VA_ARGS__)
#define emberAfPrepaymentClusterFlush() emberAfFlush(EMBER_AF_PRINT_PREPAYMENT_CLUSTER)
#define emberAfPrepaymentClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_PREPAYMENT_CLUSTER) ) { x; }
#define emberAfPrepaymentClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_PREPAYMENT_CLUSTER, (buffer), (len), (withSpace))
#define emberAfPrepaymentClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_PREPAYMENT_CLUSTER, (buffer))
#else
#define emberAfPrepaymentClusterPrint(...)
#define emberAfPrepaymentClusterPrintln(...)
#define emberAfPrepaymentClusterFlush()
#define emberAfPrepaymentClusterDebugExec(x)
#define emberAfPrepaymentClusterPrintBuffer(buffer, len, withSpace)
#define emberAfPrepaymentClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_PREPAYMENT_CLUSTER)


// Printing macros for cluster: Key establishment
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_KEY_ESTABLISHMENT_CLUSTER)
#define emberAfKeyEstablishmentClusterPrint(...)
emberAfPrint(EMBER_AF_PRINT_KEY_ESTABLISHMENT_CLUSTER, __VA_ARGS__)
#define emberAfKeyEstablishmentClusterPrintln(...)
emberAfPrintln(EMBER_AF_PRINT_KEY_ESTABLISHMENT_CLUSTER,  __VA_ARGS__)
#define emberAfKeyEstablishmentClusterFlush()
emberAfFlush(EMBER_AF_PRINT_KEY_ESTABLISHMENT_CLUSTER)
#define emberAfKeyEstablishmentClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_KEY_ESTABLISHMENT_CLUSTER) ) { x; }
#define emberAfKeyEstablishmentClusterPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_KEY_ESTABLISHMENT_CLUSTER, (buffer), (len), (withSpace))
#define emberAfKeyEstablishmentClusterPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_KEY_ESTABLISHMENT_CLUSTER, (buffer))
#else
#define emberAfKeyEstablishmentClusterPrint(...)
#define emberAfKeyEstablishmentClusterPrintln(...)
#define emberAfKeyEstablishmentClusterFlush()
#define emberAfKeyEstablishmentClusterDebugExec(x)
#define emberAfKeyEstablishmentClusterPrintBuffer(buffer, len, withSpace)
#define emberAfKeyEstablishmentClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) &&
defined(EMBER_AF_PRINT_KEY_ESTABLISHMENT_CLUSTER)


// Printing macros for cluster: Sample Mfg Specific Cluster
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_SAMPLE_MFG_SPECIFIC_CLUSTER)
#define emberAfSampleMfgSpecificClusterPrint(...)
emberAfPrint(EMBER_AF_PRINT_SAMPLE_MFG_SPECIFIC_CLUSTER, __VA_ARGS__)
#define emberAfSampleMfgSpecificClusterPrintln(...)
emberAfPrintln(EMBER_AF_PRINT_SAMPLE_MFG_SPECIFIC_CLUSTER,  __VA_ARGS__)
#define emberAfSampleMfgSpecificClusterFlush()
emberAfFlush(EMBER_AF_PRINT_SAMPLE_MFG_SPECIFIC_CLUSTER)
#define emberAfSampleMfgSpecificClusterDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_SAMPLE_MFG_SPECIFIC_CLUSTER) ) { x; }
```

```
#define emberAfSampleMfgSpecificClusterPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_SAMPLE_MFG_SPECIFIC_CLUSTER, (buffer), (len), (withSpace))
#define emberAfSampleMfgSpecificClusterPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_SAMPLE_MFG_SPECIFIC_CLUSTER, (buffer))
#else
#define emberAfSampleMfgSpecificClusterPrint(...)
#define emberAfSampleMfgSpecificClusterPrintln(...)
#define emberAfSampleMfgSpecificClusterFlush()
#define emberAfSampleMfgSpecificClusterDebugExec(x)
#define emberAfSampleMfgSpecificClusterPrintBuffer(buffer, len, withSpace)
#define emberAfSampleMfgSpecificClusterPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_SAMPLE_MFG_SPECIFIC_CLUSTER)


// Printing macros for Core
// Prints messages for global flow of the receive/send
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_CORE)
#define emberAfCorePrint(...) emberAfPrint(EMBER_AF_PRINT_CORE, __VA_ARGS__)
#define emberAfCorePrintln(...) emberAfPrintln(EMBER_AF_PRINT_CORE,  __VA_ARGS__)
#define emberAfCoreFlush() emberAfFlush(EMBER_AF_PRINT_CORE)
#define emberAfCoreDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_CORE) ) { x; }
#define emberAfCorePrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_CORE, (buffer), (len), (withSpace))
#define emberAfCorePrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_CORE, (buffer))
#else
#define emberAfCorePrint(...)
#define emberAfCorePrintln(...)
#define emberAfCoreFlush()
#define emberAfCoreDebugExec(x)
#define emberAfCorePrintBuffer(buffer, len, withSpace)
#define emberAfCorePrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_CORE)


// Printing macros for Debug
// Prints messages for random debugging
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_DEBUG)
#define emberAfDebugPrint(...) emberAfPrint(EMBER_AF_PRINT_DEBUG, __VA_ARGS__)
#define emberAfDebugPrintln(...) emberAfPrintln(EMBER_AF_PRINT_DEBUG,  __VA_ARGS__)
#define emberAfDebugFlush() emberAfFlush(EMBER_AF_PRINT_DEBUG)
#define emberAfDebugDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_DEBUG) ) { x; }
#define emberAfDebugPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_DEBUG, (buffer), (len), (withSpace))
#define emberAfDebugPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_DEBUG, (buffer))
#else
#define emberAfDebugPrint(...)
#define emberAfDebugPrintln(...)
#define emberAfDebugFlush()
#define emberAfDebugDebugExec(x)
#define emberAfDebugPrintBuffer(buffer, len, withSpace)
#define emberAfDebugPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_DEBUG)


// Printing macros for Application
// Prints messages for application part
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_APP)
```

```
#define emberAfAppPrint(...) emberAfPrint(EMBER_AF_PRINT_APP, __VA_ARGS__)
#define emberAfAppPrintln(...) emberAfPrintln(EMBER_AF_PRINT_APP,  __VA_ARGS__)
#define emberAfAppFlush() emberAfFlush(EMBER_AF_PRINT_APP)
#define emberAfAppDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_APP) ) { x; }
#define emberAfAppPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_APP, (buffer), (len), (withSpace))
#define emberAfAppPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_APP, (buffer))
#else
#define emberAfAppPrint(...)
#define emberAfAppPrintln(...)
#define emberAfAppFlush()
#define emberAfAppDebugExec(x)
#define emberAfAppPrintBuffer(buffer, len, withSpace)
#define emberAfAppPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_APP)


// Printing macros for Security
// Prints messages related to security
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_SECURITY)
#define emberAfSecurityPrint(...) emberAfPrint(EMBER_AF_PRINT_SECURITY, __VA_ARGS__)
#define emberAfSecurityPrintln(...) emberAfPrintln(EMBER_AF_PRINT_SECURITY,  __VA_ARGS__)
#define emberAfSecurityFlush() emberAfFlush(EMBER_AF_PRINT_SECURITY)
#define emberAfSecurityDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_SECURITY) ) { x; }
#define emberAfSecurityPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_SECURITY, (buffer), (len), (withSpace))
#define emberAfSecurityPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_SECURITY, (buffer))
#else
#define emberAfSecurityPrint(...)
#define emberAfSecurityPrintln(...)
#define emberAfSecurityFlush()
#define emberAfSecurityDebugExec(x)
#define emberAfSecurityPrintBuffer(buffer, len, withSpace)
#define emberAfSecurityPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_SECURITY)


// Printing macros for Attributes
// Prints messages related to attributes
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_ATTRIBUTES)
#define emberAfAttributesPrint(...) emberAfPrint(EMBER_AF_PRINT_ATTRIBUTES, __VA_ARGS__)
#define emberAfAttributesPrintln(...) emberAfPrintln(EMBER_AF_PRINT_ATTRIBUTES,  __VA_ARGS__)
#define emberAfAttributesFlush() emberAfFlush(EMBER_AF_PRINT_ATTRIBUTES)
#define emberAfAttributesDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_ATTRIBUTES) ) { x; }
#define emberAfAttributesPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_ATTRIBUTES, (buffer), (len), (withSpace))
#define emberAfAttributesPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_ATTRIBUTES, (buffer))
#else
#define emberAfAttributesPrint(...)
#define emberAfAttributesPrintln(...)
#define emberAfAttributesFlush()
#define emberAfAttributesDebugExec(x)
#define emberAfAttributesPrintBuffer(buffer, len, withSpace)
#define emberAfAttributesPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_ATTRIBUTES)


// Printing macros for Reporting
```

```
// Prints messages related to reporting
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_REPORTING)
#define emberAfReportingPrint(...) emberAfPrint(EMBER_AF_PRINT_REPORTING, __VA_ARGS__)
#define emberAfReportingPrintln(...) emberAfPrintln(EMBER_AF_PRINT_REPORTING,  __VA_ARGS__)
#define emberAfReportingFlush() emberAfFlush(EMBER_AF_PRINT_REPORTING)
#define emberAfReportingDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_REPORTING) ) { x; }
#define emberAfReportingPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer
(EMBER_AF_PRINT_REPORTING, (buffer), (len), (withSpace))
#define emberAfReportingPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_REPORTING, (buffer))
#else
#define emberAfReportingPrint(...)
#define emberAfReportingPrintln(...)
#define emberAfReportingFlush()
#define emberAfReportingDebugExec(x)
#define emberAfReportingPrintBuffer(buffer, len, withSpace)
#define emberAfReportingPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_REPORTING)


// Printing macros for Service discovery
// Prints messages related to service discovery
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_SERVICE_DISCOVERY)
#define emberAfServiceDiscoveryPrint(...) emberAfPrint
(EMBER_AF_PRINT_SERVICE_DISCOVERY, __VA_ARGS__)
#define emberAfServiceDiscoveryPrintln(...) emberAfPrintln
(EMBER_AF_PRINT_SERVICE_DISCOVERY,  __VA_ARGS__)
#define emberAfServiceDiscoveryFlush() emberAfFlush(EMBER_AF_PRINT_SERVICE_DISCOVERY)
#define emberAfServiceDiscoveryDebugExec(x) if
( emberAfPrintEnabled(EMBER_AF_PRINT_SERVICE_DISCOVERY) ) { x; }
#define emberAfServiceDiscoveryPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_SERVICE_DISCOVERY, (buffer), (len), (withSpace))
#define emberAfServiceDiscoveryPrintString(buffer)
emberAfPrintString(EMBER_AF_PRINT_SERVICE_DISCOVERY, (buffer))
#else
#define emberAfServiceDiscoveryPrint(...)
#define emberAfServiceDiscoveryPrintln(...)
#define emberAfServiceDiscoveryFlush()
#define emberAfServiceDiscoveryDebugExec(x)
#define emberAfServiceDiscoveryPrintBuffer(buffer, len, withSpace)
#define emberAfServiceDiscoveryPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_SERVICE_DISCOVERY)


// Printing macros for Registration
// Prints messages related to registration
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_REGISTRATION)
#define emberAfRegistrationPrint(...) emberAfPrint(EMBER_AF_PRINT_REGISTRATION, __VA_ARGS__)
#define emberAfRegistrationPrintln(...) emberAfPrintln(EMBER_AF_PRINT_REGISTRATION,  __VA_ARGS__)
#define emberAfRegistrationFlush() emberAfFlush(EMBER_AF_PRINT_REGISTRATION)
#define emberAfRegistrationDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_REGISTRATION) ) { x; }
#define emberAfRegistrationPrintBuffer(buffer, len, withSpace)
emberAfPrintBuffer(EMBER_AF_PRINT_REGISTRATION, (buffer), (len), (withSpace))
#define emberAfRegistrationPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_REGISTRATION, (buffer))
#else
#define emberAfRegistrationPrint(...)
#define emberAfRegistrationPrintln(...)
#define emberAfRegistrationFlush()
#define emberAfRegistrationDebugExec(x)
```

```
#define emberAfRegistrationPrintBuffer(buffer, len, withSpace)
#define emberAfRegistrationPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_REGISTRATION)


// Printing macros for ZDO (ZigBee Device Object)
// Prints messages related to ZDO functionality
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_ZDO)
#define emberAfZdoPrint(...) emberAfPrint(EMBER_AF_PRINT_ZDO, __VA_ARGS__)
#define emberAfZdoPrintln(...) emberAfPrintln(EMBER_AF_PRINT_ZDO,  __VA_ARGS__)
#define emberAfZdoFlush() emberAfFlush(EMBER_AF_PRINT_ZDO)
#define emberAfZdoDebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_ZDO) ) { x; }
#define emberAfZdoPrintBuffer(buffer, len, withSpace) emberAfPrintBuffer(EMBER_AF_PRINT_ZDO, (buffer),
(len), (withSpace))
#define emberAfZdoPrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_ZDO, (buffer))
#else
#define emberAfZdoPrint(...)
#define emberAfZdoPrintln(...)
#define emberAfZdoFlush()
#define emberAfZdoDebugExec(x)
#define emberAfZdoPrintBuffer(buffer, len, withSpace)
#define emberAfZdoPrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_ZDO)


// Printing macros for Custom messages (1)
// Messages that can be used by the end developer
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_CUSTOM1)
#define emberAfCustom1Print(...) emberAfPrint(EMBER_AF_PRINT_CUSTOM1, __VA_ARGS__)
#define emberAfCustom1Println(...) emberAfPrintln(EMBER_AF_PRINT_CUSTOM1,  __VA_ARGS__)
#define emberAfCustom1Flush() emberAfFlush(EMBER_AF_PRINT_CUSTOM1)
#define emberAfCustom1DebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_CUSTOM1) ) { x; }
#define emberAfCustom1PrintBuffer(buffer, len, withSpace) emberAfPrintBuffer
(EMBER_AF_PRINT_CUSTOM1, (buffer), (len), (withSpace))
#define emberAfCustom1PrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_CUSTOM1, (buffer))
#else
#define emberAfCustom1Print(...)
#define emberAfCustom1Println(...)
#define emberAfCustom1Flush()
#define emberAfCustom1DebugExec(x)
#define emberAfCustom1PrintBuffer(buffer, len, withSpace)
#define emberAfCustom1PrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_CUSTOM1)


// Printing macros for Custom messages (2)
// Messages that can be used by the end developer
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_CUSTOM2)
#define emberAfCustom2Print(...) emberAfPrint(EMBER_AF_PRINT_CUSTOM2, __VA_ARGS__)
#define emberAfCustom2Println(...) emberAfPrintln(EMBER_AF_PRINT_CUSTOM2,  __VA_ARGS__)
#define emberAfCustom2Flush() emberAfFlush(EMBER_AF_PRINT_CUSTOM2)
#define emberAfCustom2DebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_CUSTOM2) ) { x; }
#define emberAfCustom2PrintBuffer(buffer, len, withSpace) emberAfPrintBuffer
(EMBER_AF_PRINT_CUSTOM2, (buffer), (len), (withSpace))
#define emberAfCustom2PrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_CUSTOM2, (buffer))
#else
#define emberAfCustom2Print(...)
#define emberAfCustom2Println(...)
```

```
#define emberAfCustom2Flush()
#define emberAfCustom2DebugExec(x)
#define emberAfCustom2PrintBuffer(buffer, len, withSpace)
#define emberAfCustom2PrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_CUSTOM2)


// Printing macros for Custom messages (3)
// Messages that can be used by the end developer
#if defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_CUSTOM3)
#define emberAfCustom3Print(...) emberAfPrint(EMBER_AF_PRINT_CUSTOM3, __VA_ARGS__)
#define emberAfCustom3Println(...) emberAfPrintln(EMBER_AF_PRINT_CUSTOM3,  __VA_ARGS__)
#define emberAfCustom3Flush() emberAfFlush(EMBER_AF_PRINT_CUSTOM3)
#define emberAfCustom3DebugExec(x) if ( emberAfPrintEnabled(EMBER_AF_PRINT_CUSTOM3) ) { x; }
#define emberAfCustom3PrintBuffer(buffer, len, withSpace) emberAfPrintBuffer
(EMBER_AF_PRINT_CUSTOM3, (buffer), (len), (withSpace))
#define emberAfCustom3PrintString(buffer) emberAfPrintString(EMBER_AF_PRINT_CUSTOM3, (buffer))
#else
#define emberAfCustom3Print(...)
#define emberAfCustom3Println(...)
#define emberAfCustom3Flush()
#define emberAfCustom3DebugExec(x)
#define emberAfCustom3PrintBuffer(buffer, len, withSpace)
#define emberAfCustom3PrintString(buffer)
#endif //defined(EMBER_AF_PRINT_ENABLE) && defined(EMBER_AF_PRINT_CUSTOM3)


```

debug.h

Go to the documentation of this file.

```
// Copyright 2009 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
//

//If global debug is turned on, enable debug for all clusters.
#if defined ZCL_DEBUG_ENABLE_ALL_CLUSTERS
  #define ZCL_BASIC_DEBUG
  #define ZCL_POWER_CONFIG_DEBUG
  #define ZCL_DEVICE_TEMP_DEBUG
  #define ZCL_IDENTIFY_DEBUG
  #define ZCL_GROUPS_DEBUG
  #define ZCL_SCENES_DEBUG
  #define ZCL_ON_OFF_DEBUG
  #define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG
  #define ZCL_LEVEL_CONTROL_DEBUG
  #define ZCL_TIME_DEBUG
  #define ZCL_BINARY_INPUT_BASIC_DEBUG
  #define ZCL_COMMISSIONING_DEBUG
  #define ZCL_SHADE_CONFIG_DEBUG
  #define ZCL_PUMP_CONFIG_CONTROL_DEBUG
  #define ZCL_THERMOSTAT_DEBUG
  #define ZCL_FAN_CONTROL_DEBUG
  #define ZCL_DEHUMID_CONTROL_DEBUG
  #define ZCL_THERMOSTAT_UI_CONFIG_DEBUG
  #define ZCL_COLOR_CONTROL_DEBUG
  #define ZCL_ILLUM_MEASUREMENT_DEBUG
  #define ZCL_ILLUM_LEVEL_SENSING_DEBUG
  #define ZCL_TEMP_MEASUREMENT_DEBUG
  #define ZCL_PRESSURE_MEASUREMENT_DEBUG
  #define ZCL_FLOW_MEASUREMENT_DEBUG
  #define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG
  #define ZCL_OCCUPANCY_SENSING_DEBUG
  #define ZCL_IAS_ZONE_DEBUG
  #define ZCL_IAS_ACE_DEBUG
  #define ZCL_IAS_WD_DEBUG
  #define ZCL_DOOR_LOCK_DEBUG
  #define ZCL_WINDOW_COVERING_DEBUG
  #define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG
  #define ZCL_SIMPLE_METERING_DEBUG
  #define ZCL_PRICE_DEBUG
  #define ZCL_MESSAGING_DEBUG
  #define ZCL_KEY_ESTABLISHMENT_DEBUG
  #define ZCL_GENERIC_TUNNEL_DEBUG
  #define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG
#endif // ZCL_DEBUG_ENABLE_ALL_CLUSTERS

#if defined ZCL_BASIC_DEBUG
  #define ZCL_BASIC_DEBUG_ON 1
  #define ZCL_BASIC_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_BASIC_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_BASIC_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_BASIC_CLUSTER_ID, __VA_ARGS__)
```

```
#define ZCL_BASIC_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
#define ZCL_BASIC_DEBUG_EXECUTE(x) \
  ZCL_DEBUG(ZCL_BASIC_CLUSTER_ID, x)
#else
#define ZCL_BASIC_DEBUG_ON 0
#define ZCL_BASIC_DEBUG_PRINT(...)
#define ZCL_BASIC_DEBUG_PRINT_LINE(...)
#define ZCL_BASIC_DEBUG_WAITSEND()
#define ZCL_BASIC_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_POWER_CONFIG_DEBUG
#define ZCL_POWER_CONFIG_DEBUG_ON 1
#define ZCL_POWER_CONFIG_DEBUG_PRINT(...) \
  ZCL_LOG(ZCL_POWER_CONFIG_CLUSTER_ID, __VA_ARGS__)
#define ZCL_POWER_CONFIG_DEBUG_PRINT_LINE(...) \
  ZCL_LOG_LINE(ZCL_POWER_CONFIG_CLUSTER_ID, __VA_ARGS__)
#define ZCL_POWER_CONFIG_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
#define ZCL_POWER_CONFIG_DEBUG_EXECUTE(x) \
  ZCL_DEBUG(ZCL_POWER_CONFIG_CLUSTER_ID, x)
#else
#define ZCL_POWER_CONFIG_DEBUG_ON 0
#define ZCL_POWER_CONFIG_DEBUG_PRINT(...)
#define ZCL_POWER_CONFIG_DEBUG_PRINT_LINE(...)
#define ZCL_POWER_CONFIG_DEBUG_WAITSEND()
#define ZCL_POWER_CONFIG_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_DEVICE_TEMP_DEBUG
#define ZCL_DEVICE_TEMP_DEBUG_ON 1
#define ZCL_DEVICE_TEMP_DEBUG_PRINT(...) \
  ZCL_LOG(ZCL_DEVICE_TEMP_CLUSTER_ID, __VA_ARGS__)
#define ZCL_DEVICE_TEMP_DEBUG_PRINT_LINE(...) \
  ZCL_LOG_LINE(ZCL_DEVICE_TEMP_CLUSTER_ID, __VA_ARGS__)
#define ZCL_DEVICE_TEMP_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
#define ZCL_DEVICE_TEMP_DEBUG_EXECUTE(x) \
  ZCL_DEBUG(ZCL_DEVICE_TEMP_CLUSTER_ID, x)
#else
#define ZCL_DEVICE_TEMP_DEBUG_ON 0
#define ZCL_DEVICE_TEMP_DEBUG_PRINT(...)
#define ZCL_DEVICE_TEMP_DEBUG_PRINT_LINE(...)
#define ZCL_DEVICE_TEMP_DEBUG_WAITSEND()
#define ZCL_DEVICE_TEMP_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_IDENTIFY_DEBUG
#define ZCL_IDENTIFY_DEBUG_ON 1
#define ZCL_IDENTIFY_DEBUG_PRINT(...) \
  ZCL_LOG(ZCL_IDENTIFY_CLUSTER_ID, __VA_ARGS__)
#define ZCL_IDENTIFY_DEBUG_PRINT_LINE(...) \
  ZCL_LOG_LINE(ZCL_IDENTIFY_CLUSTER_ID, __VA_ARGS__)
#define ZCL_IDENTIFY_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
#define ZCL_IDENTIFY_DEBUG_EXECUTE(x) \
  ZCL_DEBUG(ZCL_IDENTIFY_CLUSTER_ID, x)
```

```
#else
  #define ZCL_IDENTIFY_DEBUG_ON 0
  #define ZCL_IDENTIFY_DEBUG_PRINT(...)
  #define ZCL_IDENTIFY_DEBUG_PRINT_LINE(...)
  #define ZCL_IDENTIFY_DEBUG_WAITSEND()
  #define ZCL_IDENTIFY_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_GROUPS_DEBUG
  #define ZCL_GROUPS_DEBUG_ON 1
  #define ZCL_GROUPS_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_GROUPS_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_GROUPS_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_GROUPS_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_GROUPS_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_GROUPS_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_GROUPS_CLUSTER_ID, x)
#else
  #define ZCL_GROUPS_DEBUG_ON 0
  #define ZCL_GROUPS_DEBUG_PRINT(...)
  #define ZCL_GROUPS_DEBUG_PRINT_LINE(...)
  #define ZCL_GROUPS_DEBUG_WAITSEND()
  #define ZCL_GROUPS_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_SCENES_DEBUG
  #define ZCL_SCENES_DEBUG_ON 1
  #define ZCL_SCENES_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_SCENES_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_SCENES_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_SCENES_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_SCENES_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_SCENES_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_SCENES_CLUSTER_ID, x)
#else
  #define ZCL_SCENES_DEBUG_ON 0
  #define ZCL_SCENES_DEBUG_PRINT(...)
  #define ZCL_SCENES_DEBUG_PRINT_LINE(...)
  #define ZCL_SCENES_DEBUG_WAITSEND()
  #define ZCL_SCENES_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_ON_OFF_DEBUG
  #define ZCL_ON_OFF_DEBUG_ON 1
  #define ZCL_ON_OFF_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_ON_OFF_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_ON_OFF_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_ON_OFF_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_ON_OFF_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_ON_OFF_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_ON_OFF_CLUSTER_ID, x)
#else
  #define ZCL_ON_OFF_DEBUG_ON 0
  #define ZCL_ON_OFF_DEBUG_PRINT(...)
  #define ZCL_ON_OFF_DEBUG_PRINT_LINE(...)
```

```
  #define ZCL_ON_OFF_DEBUG_WAITSEND()
  #define ZCL_ON_OFF_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_ON_OFF_SWITCH_CONFIG_DEBUG
  #define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_ON 1
  #define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_ON_OFF_SWITCH_CONFIG_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_ON_OFF_SWITCH_CONFIG_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_ON_OFF_SWITCH_CONFIG_CLUSTER_ID, x)
#else
  #define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_ON 0
  #define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_PRINT(...)
  #define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_PRINT_LINE(...)
  #define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_WAITSEND()
  #define ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_LEVEL_CONTROL_DEBUG
  #define ZCL_LEVEL_CONTROL_DEBUG_ON 1
  #define ZCL_LEVEL_CONTROL_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_LEVEL_CONTROL_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_LEVEL_CONTROL_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_LEVEL_CONTROL_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_LEVEL_CONTROL_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_LEVEL_CONTROL_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_LEVEL_CONTROL_CLUSTER_ID, x)
#else
  #define ZCL_LEVEL_CONTROL_DEBUG_ON 0
  #define ZCL_LEVEL_CONTROL_DEBUG_PRINT(...)
  #define ZCL_LEVEL_CONTROL_DEBUG_PRINT_LINE(...)
  #define ZCL_LEVEL_CONTROL_DEBUG_WAITSEND()
  #define ZCL_LEVEL_CONTROL_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_TIME_DEBUG
  #define ZCL_TIME_DEBUG_ON 1
  #define ZCL_TIME_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_TIME_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_TIME_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_TIME_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_TIME_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_TIME_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_TIME_CLUSTER_ID, x)
#else
  #define ZCL_TIME_DEBUG_ON 0
  #define ZCL_TIME_DEBUG_PRINT(...)
  #define ZCL_TIME_DEBUG_PRINT_LINE(...)
  #define ZCL_TIME_DEBUG_WAITSEND()
  #define ZCL_TIME_DEBUG_EXECUTE(x)
#endif

```

```
#if defined ZCL_BINARY_INPUT_BASIC_DEBUG
  #define ZCL_BINARY_INPUT_BASIC_DEBUG_ON 1
  #define ZCL_BINARY_INPUT_BASIC_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_BINARY_INPUT_BASIC_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_BINARY_INPUT_BASIC_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_BINARY_INPUT_BASIC_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_BINARY_INPUT_BASIC_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_BINARY_INPUT_BASIC_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_BINARY_INPUT_BASIC_CLUSTER_ID, x)
#else
  #define ZCL_BINARY_INPUT_BASIC_DEBUG_ON 0
  #define ZCL_BINARY_INPUT_BASIC_DEBUG_PRINT(...)
  #define ZCL_BINARY_INPUT_BASIC_DEBUG_PRINT_LINE(...)
  #define ZCL_BINARY_INPUT_BASIC_DEBUG_WAITSEND()
  #define ZCL_BINARY_INPUT_BASIC_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_COMMISSIONING_DEBUG
  #define ZCL_COMMISSIONING_DEBUG_ON 1
  #define ZCL_COMMISSIONING_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_COMMISSIONING_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_COMMISSIONING_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_COMMISSIONING_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_COMMISSIONING_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_COMMISSIONING_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_COMMISSIONING_CLUSTER_ID, x)
#else
  #define ZCL_COMMISSIONING_DEBUG_ON 0
  #define ZCL_COMMISSIONING_DEBUG_PRINT(...)
  #define ZCL_COMMISSIONING_DEBUG_PRINT_LINE(...)
  #define ZCL_COMMISSIONING_DEBUG_WAITSEND()
  #define ZCL_COMMISSIONING_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_SHADE_CONFIG_DEBUG
  #define ZCL_SHADE_CONFIG_DEBUG_ON 1
  #define ZCL_SHADE_CONFIG_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_SHADE_CONFIG_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_SHADE_CONFIG_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_SHADE_CONFIG_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_SHADE_CONFIG_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_SHADE_CONFIG_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_SHADE_CONFIG_CLUSTER_ID, x)
#else
  #define ZCL_SHADE_CONFIG_DEBUG_ON 0
  #define ZCL_SHADE_CONFIG_DEBUG_PRINT(...)
  #define ZCL_SHADE_CONFIG_DEBUG_PRINT_LINE(...)
  #define ZCL_SHADE_CONFIG_DEBUG_WAITSEND()
  #define ZCL_SHADE_CONFIG_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_PUMP_CONFIG_CONTROL_DEBUG
  #define ZCL_PUMP_CONFIG_CONTROL_DEBUG_ON 1
  #define ZCL_PUMP_CONFIG_CONTROL_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_PUMP_CONFIG_CONTROL_CLUSTER_ID, __VA_ARGS__)
```

```
#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_PRINT_LINE(...) \
  ZCL_LOG_LINE(ZCL_PUMP_CONFIG_CONTROL_CLUSTER_ID, __VA_ARGS__)
#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_EXECUTE(x) \
  ZCL_DEBUG(ZCL_PUMP_CONFIG_CONTROL_CLUSTER_ID, x)
#else
#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_ON 0
#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_PRINT(...)
#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_PRINT_LINE(...)
#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_WAITSEND()
#define ZCL_PUMP_CONFIG_CONTROL_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_THERMOSTAT_DEBUG
#define ZCL_THERMOSTAT_DEBUG_ON 1
#define ZCL_THERMOSTAT_DEBUG_PRINT(...) \
  ZCL_LOG(ZCL_THERMOSTAT_CLUSTER_ID, __VA_ARGS__)
#define ZCL_THERMOSTAT_DEBUG_PRINT_LINE(...) \
  ZCL_LOG_LINE(ZCL_THERMOSTAT_CLUSTER_ID, __VA_ARGS__)
#define ZCL_THERMOSTAT_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
#define ZCL_THERMOSTAT_DEBUG_EXECUTE(x) \
  ZCL_DEBUG(ZCL_THERMOSTAT_CLUSTER_ID, x)
#else
#define ZCL_THERMOSTAT_DEBUG_ON 0
#define ZCL_THERMOSTAT_DEBUG_PRINT(...)
#define ZCL_THERMOSTAT_DEBUG_PRINT_LINE(...)
#define ZCL_THERMOSTAT_DEBUG_WAITSEND()
#define ZCL_THERMOSTAT_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_FAN_CONTROL_DEBUG
#define ZCL_FAN_CONTROL_DEBUG_ON 1
#define ZCL_FAN_CONTROL_DEBUG_PRINT(...) \
  ZCL_LOG(ZCL_FAN_CONTROL_CLUSTER_ID, __VA_ARGS__)
#define ZCL_FAN_CONTROL_DEBUG_PRINT_LINE(...) \
  ZCL_LOG_LINE(ZCL_FAN_CONTROL_CLUSTER_ID, __VA_ARGS__)
#define ZCL_FAN_CONTROL_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
#define ZCL_FAN_CONTROL_DEBUG_EXECUTE(x) \
  ZCL_DEBUG(ZCL_FAN_CONTROL_CLUSTER_ID, x)
#else
#define ZCL_FAN_CONTROL_DEBUG_ON 0
#define ZCL_FAN_CONTROL_DEBUG_PRINT(...)
#define ZCL_FAN_CONTROL_DEBUG_PRINT_LINE(...)
#define ZCL_FAN_CONTROL_DEBUG_WAITSEND()
#define ZCL_FAN_CONTROL_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_DEHUMID_CONTROL_DEBUG
#define ZCL_DEHUMID_CONTROL_DEBUG_ON 1
#define ZCL_DEHUMID_CONTROL_DEBUG_PRINT(...) \
  ZCL_LOG(ZCL_DEHUMID_CONTROL_CLUSTER_ID, __VA_ARGS__)
#define ZCL_DEHUMID_CONTROL_DEBUG_PRINT_LINE(...) \
  ZCL_LOG_LINE(ZCL_DEHUMID_CONTROL_CLUSTER_ID, __VA_ARGS__)
#define ZCL_DEHUMID_CONTROL_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
```

```
  #define ZCL_DEHUMID_CONTROL_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_DEHUMID_CONTROL_CLUSTER_ID, x)
#else
  #define ZCL_DEHUMID_CONTROL_DEBUG_ON 0
  #define ZCL_DEHUMID_CONTROL_DEBUG_PRINT(...)
  #define ZCL_DEHUMID_CONTROL_DEBUG_PRINT_LINE(...)
  #define ZCL_DEHUMID_CONTROL_DEBUG_WAITSEND()
  #define ZCL_DEHUMID_CONTROL_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_THERMOSTAT_UI_CONFIG_DEBUG
  #define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_ON 1
  #define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_THERMOSTAT_UI_CONFIG_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_THERMOSTAT_UI_CONFIG_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_THERMOSTAT_UI_CONFIG_CLUSTER_ID, x)
#else
  #define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_ON 0
  #define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_PRINT(...)
  #define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_PRINT_LINE(...)
  #define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_WAITSEND()
  #define ZCL_THERMOSTAT_UI_CONFIG_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_COLOR_CONTROL_DEBUG
  #define ZCL_COLOR_CONTROL_DEBUG_ON 1
  #define ZCL_COLOR_CONTROL_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_COLOR_CONTROL_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_COLOR_CONTROL_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_COLOR_CONTROL_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_COLOR_CONTROL_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_COLOR_CONTROL_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_COLOR_CONTROL_CLUSTER_ID, x)
#else
  #define ZCL_COLOR_CONTROL_DEBUG_ON 0
  #define ZCL_COLOR_CONTROL_DEBUG_PRINT(...)
  #define ZCL_COLOR_CONTROL_DEBUG_PRINT_LINE(...)
  #define ZCL_COLOR_CONTROL_DEBUG_WAITSEND()
  #define ZCL_COLOR_CONTROL_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_ILLUM_MEASUREMENT_DEBUG
  #define ZCL_ILLUM_MEASUREMENT_DEBUG_ON 1
  #define ZCL_ILLUM_MEASUREMENT_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_ILLUM_MEASUREMENT_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_ILLUM_MEASUREMENT_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_ILLUM_MEASUREMENT_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_ILLUM_MEASUREMENT_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_ILLUM_MEASUREMENT_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_ILLUM_MEASUREMENT_CLUSTER_ID, x)
#else
  #define ZCL_ILLUM_MEASUREMENT_DEBUG_ON 0
```

```
  #define ZCL_ILLUM_MEASUREMENT_DEBUG_PRINT(...)
  #define ZCL_ILLUM_MEASUREMENT_DEBUG_PRINT_LINE(...)
  #define ZCL_ILLUM_MEASUREMENT_DEBUG_WAITSEND()
  #define ZCL_ILLUM_MEASUREMENT_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_ILLUM_LEVEL_SENSING_DEBUG
  #define ZCL_ILLUM_LEVEL_SENSING_DEBUG_ON 1
  #define ZCL_ILLUM_LEVEL_SENSING_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_ILLUM_LEVEL_SENSING_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_ILLUM_LEVEL_SENSING_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_ILLUM_LEVEL_SENSING_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_ILLUM_LEVEL_SENSING_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_ILLUM_LEVEL_SENSING_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_ILLUM_LEVEL_SENSING_CLUSTER_ID, x)
#else
  #define ZCL_ILLUM_LEVEL_SENSING_DEBUG_ON 0
  #define ZCL_ILLUM_LEVEL_SENSING_DEBUG_PRINT(...)
  #define ZCL_ILLUM_LEVEL_SENSING_DEBUG_PRINT_LINE(...)
  #define ZCL_ILLUM_LEVEL_SENSING_DEBUG_WAITSEND()
  #define ZCL_ILLUM_LEVEL_SENSING_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_TEMP_MEASUREMENT_DEBUG
  #define ZCL_TEMP_MEASUREMENT_DEBUG_ON 1
  #define ZCL_TEMP_MEASUREMENT_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_TEMP_MEASUREMENT_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_TEMP_MEASUREMENT_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_TEMP_MEASUREMENT_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_TEMP_MEASUREMENT_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_TEMP_MEASUREMENT_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_TEMP_MEASUREMENT_CLUSTER_ID, x)
#else
  #define ZCL_TEMP_MEASUREMENT_DEBUG_ON 0
  #define ZCL_TEMP_MEASUREMENT_DEBUG_PRINT(...)
  #define ZCL_TEMP_MEASUREMENT_DEBUG_PRINT_LINE(...)
  #define ZCL_TEMP_MEASUREMENT_DEBUG_WAITSEND()
  #define ZCL_TEMP_MEASUREMENT_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_PRESSURE_MEASUREMENT_DEBUG
  #define ZCL_PRESSURE_MEASUREMENT_DEBUG_ON 1
  #define ZCL_PRESSURE_MEASUREMENT_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_PRESSURE_MEASUREMENT_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_PRESSURE_MEASUREMENT_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_PRESSURE_MEASUREMENT_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_PRESSURE_MEASUREMENT_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_PRESSURE_MEASUREMENT_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_PRESSURE_MEASUREMENT_CLUSTER_ID, x)
#else
  #define ZCL_PRESSURE_MEASUREMENT_DEBUG_ON 0
  #define ZCL_PRESSURE_MEASUREMENT_DEBUG_PRINT(...)
  #define ZCL_PRESSURE_MEASUREMENT_DEBUG_PRINT_LINE(...)
  #define ZCL_PRESSURE_MEASUREMENT_DEBUG_WAITSEND()
  #define ZCL_PRESSURE_MEASUREMENT_DEBUG_EXECUTE(x)
```

```
#endif

#if defined ZCL_FLOW_MEASUREMENT_DEBUG
  #define ZCL_FLOW_MEASUREMENT_DEBUG_ON 1
  #define ZCL_FLOW_MEASUREMENT_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_FLOW_MEASUREMENT_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_FLOW_MEASUREMENT_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_FLOW_MEASUREMENT_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_FLOW_MEASUREMENT_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_FLOW_MEASUREMENT_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_FLOW_MEASUREMENT_CLUSTER_ID, x)
#else
  #define ZCL_FLOW_MEASUREMENT_DEBUG_ON 0
  #define ZCL_FLOW_MEASUREMENT_DEBUG_PRINT(...)
  #define ZCL_FLOW_MEASUREMENT_DEBUG_PRINT_LINE(...)
  #define ZCL_FLOW_MEASUREMENT_DEBUG_WAITSEND()
  #define ZCL_FLOW_MEASUREMENT_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG
  #define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_ON 1
  #define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER_ID, x)
#else
  #define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_ON 0
  #define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_PRINT(...)
  #define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_PRINT_LINE(...)
  #define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_WAITSEND()
  #define ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_OCCUPANCY_SENSING_DEBUG
  #define ZCL_OCCUPANCY_SENSING_DEBUG_ON 1
  #define ZCL_OCCUPANCY_SENSING_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_OCCUPANCY_SENSING_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_OCCUPANCY_SENSING_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_OCCUPANCY_SENSING_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_OCCUPANCY_SENSING_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_OCCUPANCY_SENSING_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_OCCUPANCY_SENSING_CLUSTER_ID, x)
#else
  #define ZCL_OCCUPANCY_SENSING_DEBUG_ON 0
  #define ZCL_OCCUPANCY_SENSING_DEBUG_PRINT(...)
  #define ZCL_OCCUPANCY_SENSING_DEBUG_PRINT_LINE(...)
  #define ZCL_OCCUPANCY_SENSING_DEBUG_WAITSEND()
  #define ZCL_OCCUPANCY_SENSING_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_IAS_ZONE_DEBUG
  #define ZCL_IAS_ZONE_DEBUG_ON 1
```

```
#define ZCL_IAS_ZONE_DEBUG_PRINT(...) \
  ZCL_LOG(ZCL_IAS_ZONE_CLUSTER_ID, __VA_ARGS__)
#define ZCL_IAS_ZONE_DEBUG_PRINT_LINE(...) \
  ZCL_LOG_LINE(ZCL_IAS_ZONE_CLUSTER_ID, __VA_ARGS__)
#define ZCL_IAS_ZONE_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
#define ZCL_IAS_ZONE_DEBUG_EXECUTE(x) \
  ZCL_DEBUG(ZCL_IAS_ZONE_CLUSTER_ID, x)
#else
#define ZCL_IAS_ZONE_DEBUG_ON 0
#define ZCL_IAS_ZONE_DEBUG_PRINT(...)
#define ZCL_IAS_ZONE_DEBUG_PRINT_LINE(...)
#define ZCL_IAS_ZONE_DEBUG_WAITSEND()
#define ZCL_IAS_ZONE_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_IAS_ACE_DEBUG
#define ZCL_IAS_ACE_DEBUG_ON 1
#define ZCL_IAS_ACE_DEBUG_PRINT(...) \
  ZCL_LOG(ZCL_IAS_ACE_CLUSTER_ID, __VA_ARGS__)
#define ZCL_IAS_ACE_DEBUG_PRINT_LINE(...) \
  ZCL_LOG_LINE(ZCL_IAS_ACE_CLUSTER_ID, __VA_ARGS__)
#define ZCL_IAS_ACE_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
#define ZCL_IAS_ACE_DEBUG_EXECUTE(x) \
  ZCL_DEBUG(ZCL_IAS_ACE_CLUSTER_ID, x)
#else
#define ZCL_IAS_ACE_DEBUG_ON 0
#define ZCL_IAS_ACE_DEBUG_PRINT(...)
#define ZCL_IAS_ACE_DEBUG_PRINT_LINE(...)
#define ZCL_IAS_ACE_DEBUG_WAITSEND()
#define ZCL_IAS_ACE_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_IAS_WD_DEBUG
#define ZCL_IAS_WD_DEBUG_ON 1
#define ZCL_IAS_WD_DEBUG_PRINT(...) \
  ZCL_LOG(ZCL_IAS_WD_CLUSTER_ID, __VA_ARGS__)
#define ZCL_IAS_WD_DEBUG_PRINT_LINE(...) \
  ZCL_LOG_LINE(ZCL_IAS_WD_CLUSTER_ID, __VA_ARGS__)
#define ZCL_IAS_WD_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
#define ZCL_IAS_WD_DEBUG_EXECUTE(x) \
  ZCL_DEBUG(ZCL_IAS_WD_CLUSTER_ID, x)
#else
#define ZCL_IAS_WD_DEBUG_ON 0
#define ZCL_IAS_WD_DEBUG_PRINT(...)
#define ZCL_IAS_WD_DEBUG_PRINT_LINE(...)
#define ZCL_IAS_WD_DEBUG_WAITSEND()
#define ZCL_IAS_WD_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_DOOR_LOCK_DEBUG
#define ZCL_DOOR_LOCK_DEBUG_ON 1
#define ZCL_DOOR_LOCK_DEBUG_PRINT(...) \
  ZCL_LOG(ZCL_DOOR_LOCK_CLUSTER_ID, __VA_ARGS__)
#define ZCL_DOOR_LOCK_DEBUG_PRINT_LINE(...) \
  ZCL_LOG_LINE(ZCL_DOOR_LOCK_CLUSTER_ID, __VA_ARGS__)
```

```
#define ZCL_DOOR_LOCK_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
#define ZCL_DOOR_LOCK_DEBUG_EXECUTE(x) \
  ZCL_DEBUG(ZCL_DOOR_LOCK_CLUSTER_ID, x)
#else
#define ZCL_DOOR_LOCK_DEBUG_ON 0
#define ZCL_DOOR_LOCK_DEBUG_PRINT(...)
#define ZCL_DOOR_LOCK_DEBUG_PRINT_LINE(...)
#define ZCL_DOOR_LOCK_DEBUG_WAITSEND()
#define ZCL_DOOR_LOCK_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_WINDOW_COVERING_DEBUG
#define ZCL_WINDOW_COVERING_DEBUG_ON 1
#define ZCL_WINDOW_COVERING_DEBUG_PRINT(...) \
  ZCL_LOG(ZCL_WINDOW_COVERING_CLUSTER_ID, __VA_ARGS__)
#define ZCL_WINDOW_COVERING_DEBUG_PRINT_LINE(...) \
  ZCL_LOG_LINE(ZCL_WINDOW_COVERING_CLUSTER_ID, __VA_ARGS__)
#define ZCL_WINDOW_COVERING_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
#define ZCL_WINDOW_COVERING_DEBUG_EXECUTE(x) \
  ZCL_DEBUG(ZCL_WINDOW_COVERING_CLUSTER_ID, x)
#else
#define ZCL_WINDOW_COVERING_DEBUG_ON 0
#define ZCL_WINDOW_COVERING_DEBUG_PRINT(...)
#define ZCL_WINDOW_COVERING_DEBUG_PRINT_LINE(...)
#define ZCL_WINDOW_COVERING_DEBUG_WAITSEND()
#define ZCL_WINDOW_COVERING_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_ON 1
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_PRINT(...) \
  ZCL_LOG(ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID, __VA_ARGS__)
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_PRINT_LINE(...) \
  ZCL_LOG_LINE(ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID, __VA_ARGS__)
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_EXECUTE(x) \
  ZCL_DEBUG(ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID, x)
#else
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_ON 0
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_PRINT(...)
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_PRINT_LINE(...)
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_WAITSEND()
#define ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_SIMPLE_METERING_DEBUG
#define ZCL_SIMPLE_METERING_DEBUG_ON 1
#define ZCL_SIMPLE_METERING_DEBUG_PRINT(...) \
  ZCL_LOG(ZCL_SIMPLE_METERING_CLUSTER_ID, __VA_ARGS__)
#define ZCL_SIMPLE_METERING_DEBUG_PRINT_LINE(...) \
  ZCL_LOG_LINE(ZCL_SIMPLE_METERING_CLUSTER_ID, __VA_ARGS__)
#define ZCL_SIMPLE_METERING_DEBUG_WAITSEND() \
  ZCL_LOG_WAITSEND()
#define ZCL_SIMPLE_METERING_DEBUG_EXECUTE(x) \
  ZCL_DEBUG(ZCL_SIMPLE_METERING_CLUSTER_ID, x)
```

```
#else
  #define ZCL_SIMPLE_METERING_DEBUG_ON 0
  #define ZCL_SIMPLE_METERING_DEBUG_PRINT(...)
  #define ZCL_SIMPLE_METERING_DEBUG_PRINT_LINE(...)
  #define ZCL_SIMPLE_METERING_DEBUG_WAITSEND()
  #define ZCL_SIMPLE_METERING_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_PRICE_DEBUG
  #define ZCL_PRICE_DEBUG_ON 1
  #define ZCL_PRICE_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_PRICE_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_PRICE_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_PRICE_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_PRICE_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_PRICE_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_PRICE_CLUSTER_ID, x)
#else
  #define ZCL_PRICE_DEBUG_ON 0
  #define ZCL_PRICE_DEBUG_PRINT(...)
  #define ZCL_PRICE_DEBUG_PRINT_LINE(...)
  #define ZCL_PRICE_DEBUG_WAITSEND()
  #define ZCL_PRICE_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_MESSAGING_DEBUG
  #define ZCL_MESSAGING_DEBUG_ON 1
  #define ZCL_MESSAGING_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_MESSAGING_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_MESSAGING_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_MESSAGING_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_MESSAGING_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_MESSAGING_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_MESSAGING_CLUSTER_ID, x)
#else
  #define ZCL_MESSAGING_DEBUG_ON 0
  #define ZCL_MESSAGING_DEBUG_PRINT(...)
  #define ZCL_MESSAGING_DEBUG_PRINT_LINE(...)
  #define ZCL_MESSAGING_DEBUG_WAITSEND()
  #define ZCL_MESSAGING_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_KEY_ESTABLISHMENT_DEBUG
  #define ZCL_KEY_ESTABLISHMENT_DEBUG_ON 1
  #define ZCL_KEY_ESTABLISHMENT_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_KEY_ESTABLISHMENT_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_KEY_ESTABLISHMENT_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_KEY_ESTABLISHMENT_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_KEY_ESTABLISHMENT_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_KEY_ESTABLISHMENT_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_KEY_ESTABLISHMENT_CLUSTER_ID, x)
#else
  #define ZCL_KEY_ESTABLISHMENT_DEBUG_ON 0
  #define ZCL_KEY_ESTABLISHMENT_DEBUG_PRINT(...)
  #define ZCL_KEY_ESTABLISHMENT_DEBUG_PRINT_LINE(...)
```

```
  #define ZCL_KEY_ESTABLISHMENT_DEBUG_WAITSEND()
  #define ZCL_KEY_ESTABLISHMENT_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_GENERIC_TUNNEL_DEBUG
  #define ZCL_GENERIC_TUNNEL_DEBUG_ON 1
  #define ZCL_GENERIC_TUNNEL_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_GENERIC_TUNNEL_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_GENERIC_TUNNEL_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_GENERIC_TUNNEL_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_GENERIC_TUNNEL_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_GENERIC_TUNNEL_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_GENERIC_TUNNEL_CLUSTER_ID, x)
#else
  #define ZCL_GENERIC_TUNNEL_DEBUG_ON 0
  #define ZCL_GENERIC_TUNNEL_DEBUG_PRINT(...)
  #define ZCL_GENERIC_TUNNEL_DEBUG_PRINT_LINE(...)
  #define ZCL_GENERIC_TUNNEL_DEBUG_WAITSEND()
  #define ZCL_GENERIC_TUNNEL_DEBUG_EXECUTE(x)
#endif

#if defined ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG
  #define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_ON 1
  #define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_PRINT(...) \
    ZCL_LOG(ZCL_BACNET_PROTOCOL_TUNNEL_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_PRINT_LINE(...) \
    ZCL_LOG_LINE(ZCL_BACNET_PROTOCOL_TUNNEL_CLUSTER_ID, __VA_ARGS__)
  #define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_WAITSEND() \
    ZCL_LOG_WAITSEND()
  #define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_EXECUTE(x) \
    ZCL_DEBUG(ZCL_BACNET_PROTOCOL_TUNNEL_CLUSTER_ID, x)
#else
  #define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_ON 0
  #define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_PRINT(...)
  #define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_PRINT_LINE(...)
  #define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_WAITSEND()
  #define ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_EXECUTE(x)
#endif

// Test code that exercises every cluster print routine.
#define DEBUG_PRINT_TEST \
  ZCL_BASIC_DEBUG_PRINT_LINE("BASIC_CLUSTER debug on."); \
  ZCL_POWER_CONFIG_DEBUG_PRINT_LINE("POWER_CONFIG_CLUSTER debug on."); \
  ZCL_DEVICE_TEMP_DEBUG_PRINT_LINE("DEVICE_TEMP_CLUSTER debug on."); \
  ZCL_IDENTIFY_DEBUG_PRINT_LINE("IDENTIFY_CLUSTER debug on."); \
  ZCL_GROUPS_DEBUG_PRINT_LINE("GROUPS_CLUSTER debug on."); \
  ZCL_SCENES_DEBUG_PRINT_LINE("SCENES_CLUSTER debug on."); \
  ZCL_ON_OFF_DEBUG_PRINT_LINE("ON_OFF_CLUSTER debug on."); \
  ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_PRINT_LINE("ON_OFF_SWITCH_CONFIG_CLUSTER debug on."); \
  ZCL_LEVEL_CONTROL_DEBUG_PRINT_LINE("LEVEL_CONTROL_CLUSTER debug on."); \
  ZCL_TIME_DEBUG_PRINT_LINE("TIME_CLUSTER debug on."); \
  ZCL_BINARY_INPUT_BASIC_DEBUG_PRINT_LINE("BINARY_INPUT_BASIC_CLUSTER debug on."); \
  ZCL_COMMISSIONING_DEBUG_PRINT_LINE("COMMISSIONING_CLUSTER debug on."); \
  ZCL_SHADE_CONFIG_DEBUG_PRINT_LINE("SHADE_CONFIG_CLUSTER debug on."); \
  ZCL_PUMP_CONFIG_CONTROL_DEBUG_PRINT_LINE("PUMP_CONFIG_CONTROL_CLUSTER debug on."); \
  ZCL_THERMOSTAT_DEBUG_PRINT_LINE("THERMOSTAT_CLUSTER debug on."); \
  ZCL_FAN_CONTROL_DEBUG_PRINT_LINE("FAN_CONTROL_CLUSTER debug on."); \
  ZCL_DEHUMID_CONTROL_DEBUG_PRINT_LINE("DEHUMID_CONTROL_CLUSTER debug on."); \
```

```
ZCL_THERMOSTAT_UI_CONFIG_DEBUG_PRINT_LINE("THERMOSTAT_UI_CONFIG_CLUSTER debug on."); \
ZCL_COLOR_CONTROL_DEBUG_PRINT_LINE("COLOR_CONTROL_CLUSTER debug on."); \
ZCL_ILLUM_MEASUREMENT_DEBUG_PRINT_LINE("ILLUM_MEASUREMENT_CLUSTER debug on."); \
ZCL_ILLUM_LEVEL_SENSING_DEBUG_PRINT_LINE("ILLUM_LEVEL_SENSING_CLUSTER debug on."); \
ZCL_TEMP_MEASUREMENT_DEBUG_PRINT_LINE("TEMP_MEASUREMENT_CLUSTER debug on."); \
ZCL_PRESSURE_MEASUREMENT_DEBUG_PRINT_LINE("PRESSURE_MEASUREMENT_CLUSTER debug on."); \
ZCL_FLOW_MEASUREMENT_DEBUG_PRINT_LINE("FLOW_MEASUREMENT_CLUSTER debug on."); \
ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_PRINT_LINE("RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER debug on."); \
ZCL_OCCUPANCY_SENSING_DEBUG_PRINT_LINE("OCCUPANCY_SENSING_CLUSTER debug on."); \
ZCL_IAS_ZONE_DEBUG_PRINT_LINE("IAS_ZONE_CLUSTER debug on."); \
ZCL_IAS_ACE_DEBUG_PRINT_LINE("IAS_ACE_CLUSTER debug on."); \
ZCL_IAS_WD_DEBUG_PRINT_LINE("IAS_WD_CLUSTER debug on."); \
ZCL_DOOR_LOCK_DEBUG_PRINT_LINE("DOOR_LOCK_CLUSTER debug on."); \
ZCL_WINDOW_COVERING_DEBUG_PRINT_LINE("WINDOW_COVERING_CLUSTER debug on."); \
ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_PRINT_LINE("DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER debug on."); \
ZCL_SIMPLE_METERING_DEBUG_PRINT_LINE("SIMPLE_METERING_CLUSTER debug on."); \
ZCL_PRICE_DEBUG_PRINT_LINE("PRICE_CLUSTER debug on."); \
ZCL_MESSAGING_DEBUG_PRINT_LINE("MESSAGING_CLUSTER debug on."); \
ZCL_KEY_ESTABLISHMENT_DEBUG_PRINT_LINE("KEY_ESTABLISHMENT_CLUSTER debug on."); \
ZCL_GENERIC_TUNNEL_DEBUG_PRINT_LINE("GENERIC_TUNNEL_CLUSTER debug on."); \
ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_PRINT_LINE("BACNET_PROTOCOL_TUNNEL_CLUSTER debug on."); \


// A mapping of cluster IDs to booleans indicating whether debug is turned or off
#define LOG_LEVELS \
  { ZCL_BASIC_CLUSTER_ID,                    ZCL_BASIC_DEBUG_ON                    }, \
  { ZCL_POWER_CONFIG_CLUSTER_ID,             ZCL_POWER_CONFIG_DEBUG_ON             }, \
  { ZCL_DEVICE_TEMP_CLUSTER_ID,              ZCL_DEVICE_TEMP_DEBUG_ON              }, \
  { ZCL_IDENTIFY_CLUSTER_ID,                 ZCL_IDENTIFY_DEBUG_ON                 }, \
  { ZCL_GROUPS_CLUSTER_ID,                   ZCL_GROUPS_DEBUG_ON                   }, \
  { ZCL_SCENES_CLUSTER_ID,                   ZCL_SCENES_DEBUG_ON                   }, \
  { ZCL_ON_OFF_CLUSTER_ID,                   ZCL_ON_OFF_DEBUG_ON                   }, \
  { ZCL_ON_OFF_SWITCH_CONFIG_CLUSTER_ID,     ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_ON     }, \
  { ZCL_LEVEL_CONTROL_CLUSTER_ID,            ZCL_LEVEL_CONTROL_DEBUG_ON            }, \
  { ZCL_TIME_CLUSTER_ID,                     ZCL_TIME_DEBUG_ON                     }, \
  { ZCL_BINARY_INPUT_BASIC_CLUSTER_ID,       ZCL_BINARY_INPUT_BASIC_DEBUG_ON       }, \
  { ZCL_COMMISSIONING_CLUSTER_ID,            ZCL_COMMISSIONING_DEBUG_ON            }, \
  { ZCL_SHADE_CONFIG_CLUSTER_ID,             ZCL_SHADE_CONFIG_DEBUG_ON             }, \
  { ZCL_PUMP_CONFIG_CONTROL_CLUSTER_ID, ZCL_PUMP_CONFIG_CONTROL_DEBUG_ON          }, \
  { ZCL_THERMOSTAT_CLUSTER_ID,               ZCL_THERMOSTAT_DEBUG_ON               }, \
  { ZCL_FAN_CONTROL_CLUSTER_ID,              ZCL_FAN_CONTROL_DEBUG_ON              }, \
  { ZCL_DEHUMID_CONTROL_CLUSTER_ID,          ZCL_DEHUMID_CONTROL_DEBUG_ON          }, \
{ ZCL_THERMOSTAT_UI_CONFIG_CLUSTER_ID,       ZCL_THERMOSTAT_UI_CONFIG_DEBUG_ON     }, \
  { ZCL_COLOR_CONTROL_CLUSTER_ID,            ZCL_COLOR_CONTROL_DEBUG_ON            }, \
  { ZCL_ILLUM_MEASUREMENT_CLUSTER_ID,        ZCL_ILLUM_MEASUREMENT_DEBUG_ON        }, \
  { ZCL_ILLUM_LEVEL_SENSING_CLUSTER_ID,      ZCL_ILLUM_LEVEL_SENSING_DEBUG_ON      }, \
  { ZCL_TEMP_MEASUREMENT_CLUSTER_ID,         ZCL_TEMP_MEASUREMENT_DEBUG_ON         }, \
{ ZCL_PRESSURE_MEASUREMENT_CLUSTER_ID,       ZCL_PRESSURE_MEASUREMENT_DEBUG_ON     }, \
  { ZCL_FLOW_MEASUREMENT_CLUSTER_ID,         ZCL_FLOW_MEASUREMENT_DEBUG_ON         }, \
{ ZCL_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER_ID,
```

```
ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_ON }, \
00802   { ZCL_OCCUPANCY_SENSING_CLUSTER_ID,          ZCL_OCCUPANCY_SENSING_DEBUG_ON          }, \
00803   { ZCL_IAS_ZONE_CLUSTER_ID,                   ZCL_IAS_ZONE_DEBUG_ON                   }, \
00804   { ZCL_IAS_ACE_CLUSTER_ID,                    ZCL_IAS_ACE_DEBUG_ON                    }, \
00805   { ZCL_IAS_WD_CLUSTER_ID,                     ZCL_IAS_WD_DEBUG_ON                     }, \
00806   { ZCL_DOOR_LOCK_CLUSTER_ID,                  ZCL_DOOR_LOCK_DEBUG_ON                  }, \
00807   { ZCL_WINDOW_COVERING_CLUSTER_ID,            ZCL_WINDOW_COVERING_DEBUG_ON            }, \
00808
{ ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID,
ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_ON  }, \
00809   { ZCL_SIMPLE_METERING_CLUSTER_ID,            ZCL_SIMPLE_METERING_DEBUG_ON            }, \
00810   { ZCL_PRICE_CLUSTER_ID,                      ZCL_PRICE_DEBUG_ON                      }, \
00811   { ZCL_MESSAGING_CLUSTER_ID,                  ZCL_MESSAGING_DEBUG_ON                  }, \
00812   { ZCL_KEY_ESTABLISHMENT_CLUSTER_ID,          ZCL_KEY_ESTABLISHMENT_DEBUG_ON          }, \
00813   { ZCL_GENERIC_TUNNEL_CLUSTER_ID,             ZCL_GENERIC_TUNNEL_DEBUG_ON             }, \
00814
{ ZCL_BACNET_PROTOCOL_TUNNEL_CLUSTER_ID,       ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_ON       }, \
00815
```

demand-response-load-control.h

Go to the documentation of this file.

```
// ********************************************************************
// * ami-demand-response-load-control.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************

#include "load-control-event-table.h"

// needed for EmberSignatureData
#include "stack/include/ember-types.h"

#define EVENT_OPT_IN_DEFAULT  0x01

void afReportEventStatusHandler(int32u eventId,
                                int8u eventCode,
                                int32u startTime,
                                int8u criticalityLevelApplied,
                                int16s coolingTempSetPointApplied,
                                int16s heatingTempSetPointApplied,
                                int8s avgLoadAdjPercent,
                                int8u dutyCycle,
                                int8u eventControl,
                                int8u messageLength,
                                int8u* message,
                                EmberSignatureData* signature);
```

drlc-callback.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generatePluginCallbackPrototypes()
//

boolean emberAfPluginDrlcEventActionCallback(EmberAfLoadControlEvent * loadControlEvent,
                                             EmberAfAmiEventStatus eventStatus,
                                             int8u sequenceNumber);


```

drlc-server.h

Go to the documentation of this file.

```
// ******************************************************************
// * drlc-server.h
// *
// * This is an implementation of a drlc server which is intended to
// * be used for testing only. Generally it is expected that the drlc
// * server will act as a passthrough for drlc events and will not maintain
// * a list of scheduled events of its own. However for testing purposes
// * it is highly useful to have the server store scheduled events so that
// * - minus a backend - the ESP can respond to get scheduled events
// * commands.
// *
// * There are a couple things about
// * this code that make it incomplete as an actual drlc server table
// *
// * 1. Scheduled events are not stored in persistent memory, which they
// *    would have to be if this were intended for an ESP which for some
// *    reason were storing events.
// *
// * 2. It is possible to schedule an event with time 0x0000, which when
// *    retrieved means start this event now. If this is active for all
// *    time then the event will always start now, whatever now might be.
// *
// * Note: The EmberAfLoadControlEvent source field is used to activate
// * / deactivate an event. If the field is set to 0xffff, then
// * the event is considered inactive, If it is set to 0x0000 then the
// * event is considered active.
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.             *80*
// ******************************************************************

#include "../../include/af-types.h"

#ifndef EMBER_AF_PLUGIN_DRLC_SERVER_SCHEDULED_EVENT_TABLE_SIZE
  #define  EMBER_AF_PLUGIN_DRLC_SERVER_SCHEDULED_EVENT_TABLE_SIZE 2
#endif //EMBER_AF_PLUGIN_DRLC_SERVER_SCHEDULED_EVENT_TABLE_SIZE

// used by the cli to clear all scheduled events in the drlc server
void emAfClearScheduledLoadControlEvents(void);

// returns a pointer to the next free scheduled load control event table entry
// or NULL if none exists.
```

```
EmberAfLoadControlEvent *emAfGetScheduledLoadControlEvent(int8u index);

void emAfPluginDrlcServerPrintInfo(void);
```

enums.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generateEnums()
//

// Enclosing macro to prevent multiple inclusion
#ifndef __EMBER_AF_ENUMS__
#define __EMBER_AF_ENUMS__


// Enums

// CommissioningStartupControl
typedef enum {
  EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_NO_ACTION          = 0x00,
  EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_FORM_NETWORK       = 0x01,
  EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_REJOIN_NETWORK     = 0x02,
  EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_START_FROM_SCRATCH  = 0x03
} EmberAfCommissioningStartupControl;

// MeteringBlockEnumerations
typedef enum {
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_NO_BLOCKS_IN_USE  = 0x00,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK1           = 0x01,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK2           = 0x02,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK3           = 0x03,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK4           = 0x04,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK5           = 0x05,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK6           = 0x06,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK7           = 0x07,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK8           = 0x08,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK9           = 0x09,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK10          = 0x0A,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK11          = 0x0B,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK12          = 0x0C,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK13          = 0x0D,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK14          = 0x0E,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK15          = 0x0F,
  EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK16          = 0x10
} EmberAfMeteringBlockEnumerations;

// MeteringSupplyStatus
typedef enum {
  EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_OFF        = 0x00,
  EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_OFF_ARMED  = 0x01,
  EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_ON         = 0x02
} EmberAfMeteringSupplyStatus;
```

```
// MeteringTemperatureUnitOfMeasure
typedef enum {
  EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_KELVIN          = 0x00,
  EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_CELSIUS         = 0x01,
  EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_FAHRENHEIT      = 0x02,
  EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_KELVIN_BCD      = 0x80,
  EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_CELSIUS_BCD     = 0x81,
  EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_FAHRENHEIT_BCD  = 0x82
} EmberAfMeteringTemperatureUnitOfMeasure;

// MeteringAlarmCode
typedef enum {
  EMBER_ZCL_METERING_ALARM_CODE_CHECK_METER                          = 0x00,
  EMBER_ZCL_METERING_ALARM_CODE_LOW_BATTERY                          = 0x01,
  EMBER_ZCL_METERING_ALARM_CODE_TAMPER_DETECT                        = 0x02,
  EMBER_ZCL_METERING_ALARM_CODE_POWER_FAILURE_PIPE_EMPTY_TEMPERATURE_SENSOR
= 0x03,
  EMBER_ZCL_METERING_ALARM_CODE_POWER_QUALITY_LOW_PRESSURE_BURST_DETECT      = 0x04,
  EMBER_ZCL_METERING_ALARM_CODE_LEAK_DETECT                          = 0x05,
  EMBER_ZCL_METERING_ALARM_CODE_SERVICE_DISCONNECT                   = 0x06,
  EMBER_ZCL_METERING_ALARM_CODE_REVERSE_FLOW_FLOW_SENSOR             = 0x07,
  EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L1                       = 0x10,
  EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L1                      = 0x11,
  EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L2                       = 0x12,
  EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L2                      = 0x13,
  EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L3                       = 0x14,
  EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L3                      = 0x15,
  EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L1                      = 0x16,
  EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L2                      = 0x17,
  EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L3                      = 0x18,
  EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L1                 = 0x19,
  EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L1                = 0x1A,
  EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L2                 = 0x1B,
  EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L2                = 0x1C,
  EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L3                 = 0x1D,
  EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L3                = 0x1E,
  EMBER_ZCL_METERING_ALARM_CODE_GROUND_FAULT                         = 0x1F,
  EMBER_ZCL_METERING_ALARM_CODE_ELECTRIC_TAMPER_DETECT               = 0x20,
  EMBER_ZCL_METERING_ALARM_CODE_BURST_DETECT                         = 0x30,
  EMBER_ZCL_METERING_ALARM_CODE_PRESSURE_TOO_LOW                     = 0x31,
  EMBER_ZCL_METERING_ALARM_CODE_PRESSURE_TOO_HIGH                    = 0x32,
  EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_COMMUNICATION_ERROR      = 0x33,
  EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_MEASUREMENT_FAULT        = 0x34,
  EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_REVERSE_FLOW             = 0x35,
  EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_AIR_DETECT               = 0x36,
  EMBER_ZCL_METERING_ALARM_CODE_PIPE_EMPTY                           = 0x37,
  EMBER_ZCL_METERING_ALARM_CODE_INLET_TEMPERATURE_SENSOR_FAULT       = 0x50,
  EMBER_ZCL_METERING_ALARM_CODE_OUTLET_TEMPERATURE_SENSOR_FAULT      = 0x51
} EmberAfMeteringAlarmCode;

// MessagingControlTransmission
typedef enum {
  EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_NORMAL                    = 0x00,
```

```
  EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_NORMAL_AND_ANONYMOUS  = 0x01,
  EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_ANONYMOUS             = 0x02,
  EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_RESERVED              = 0x03
} EmberAfMessagingControlTransmission;

// MessagingControlImportance
typedef enum {
  EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_LOW       = 0x00,
  EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_MEDIUM    = 0x04,
  EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_HIGH      = 0x08,
  EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_CRITICAL  = 0x0C
} EmberAfMessagingControlImportance;

// MessagingControlConfirmation
typedef enum {
  EMBER_ZCL_MESSAGING_CONTROL_CONFIRMATION_NOT_REQUIRED  = 0x00,
  EMBER_ZCL_MESSAGING_CONTROL_CONFIRMATION_REQUIRED      = 0x80
} EmberAfMessagingControlConfirmation;

// OperatingMode
typedef enum {
  EMBER_ZCL_OPERATING_MODE_NORMAL     = 0x00,
  EMBER_ZCL_OPERATING_MODE_CONFIGURE  = 0x01
} EmberAfOperatingMode;

// Status
typedef enum {
  EMBER_ZCL_STATUS_SUCCESS                    = 0x00,
  EMBER_ZCL_STATUS_FAILURE                    = 0x01,
  EMBER_ZCL_STATUS_REQUEST_DENIED             = 0x70,
  EMBER_ZCL_STATUS_MULTIPLE_REQUEST_NOT_ALLOWED  = 0x71,
  EMBER_ZCL_STATUS_INDICATION_REDIRECTION_TO_AP  = 0x72,
  EMBER_ZCL_STATUS_PREFERENCE_DENIED          = 0x73,
  EMBER_ZCL_STATUS_PREFERENCE_IGNORED         = 0x74,
  EMBER_ZCL_STATUS_NOT_AUTHORIZED             = 0x7E,
  EMBER_ZCL_STATUS_RESERVED_FIELD_NOT_ZERO    = 0x7F,
  EMBER_ZCL_STATUS_MALFORMED_COMMAND          = 0x80,
  EMBER_ZCL_STATUS_UNSUP_CLUSTER_COMMAND      = 0x81,
  EMBER_ZCL_STATUS_UNSUP_GENERAL_COMMAND      = 0x82,
  EMBER_ZCL_STATUS_UNSUP_MANUF_CLUSTER_COMMAND   = 0x83,
  EMBER_ZCL_STATUS_UNSUP_MANUF_GENERAL_COMMAND   = 0x84,
  EMBER_ZCL_STATUS_INVALID_FIELD              = 0x85,
  EMBER_ZCL_STATUS_UNSUPPORTED_ATTRIBUTE      = 0x86,
  EMBER_ZCL_STATUS_INVALID_VALUE              = 0x87,
  EMBER_ZCL_STATUS_READ_ONLY                  = 0x88,
  EMBER_ZCL_STATUS_INSUFFICIENT_SPACE         = 0x89,
  EMBER_ZCL_STATUS_DUPLICATE_EXISTS           = 0x8A,
  EMBER_ZCL_STATUS_NOT_FOUND                  = 0x8B,
  EMBER_ZCL_STATUS_UNREPORTABLE_ATTRIBUTE     = 0x8C,
  EMBER_ZCL_STATUS_INVALID_DATA_TYPE          = 0x8D,
  EMBER_ZCL_STATUS_INVALID_SELECTOR           = 0x8E,
  EMBER_ZCL_STATUS_WRITE_ONLY                 = 0x8F,
  EMBER_ZCL_STATUS_INCONSISTENT_STARTUP_STATE    = 0x90,
  EMBER_ZCL_STATUS_DEFINED_OUT_OF_BAND        = 0x91,
```

```
  EMBER_ZCL_STATUS_ABORT                    = 0x95,
  EMBER_ZCL_STATUS_INVALID_IMAGE            = 0x96,
  EMBER_ZCL_STATUS_WAIT_FOR_DATA            = 0x97,
  EMBER_ZCL_STATUS_NO_IMAGE_AVAILABLE       = 0x98,
  EMBER_ZCL_STATUS_REQUIRE_MORE_IMAGE       = 0x99,
  EMBER_ZCL_STATUS_HARDWARE_FAILURE         = 0xC0,
  EMBER_ZCL_STATUS_SOFTWARE_FAILURE         = 0xC1,
  EMBER_ZCL_STATUS_CALIBRATION_ERROR        = 0xC2
} EmberAfStatus;

// PowerSource
typedef enum {
  EMBER_ZCL_POWER_SOURCE_UNKNOWN                    = 0x00,
  EMBER_ZCL_POWER_SOURCE_SINGLE_PHASE_MAINS         = 0x01,
  EMBER_ZCL_POWER_SOURCE_THREE_PHASE_MAINS          = 0x02,
  EMBER_ZCL_POWER_SOURCE_BATTERY                    = 0x03,
  EMBER_ZCL_POWER_SOURCE_DC_SOURCE                  = 0x04,
  EMBER_ZCL_POWER_SOURCE_EMERGENCY_MAINS_CONSTANT_POWER   = 0x05,
  EMBER_ZCL_POWER_SOURCE_EMERGENCY_MAINS_TRANSFER_SWITCH  = 0x06,
  EMBER_ZCL_POWER_SOURCE_BATTERY_BACKUP             = 0x80
} EmberAfPowerSource;

// PhysicalEnvironment
typedef enum {
  EMBER_ZCL_PHYSICAL_ENVIRONMENT_UNSPECIFIED                = 0x00,
  EMBER_ZCL_PHYSICAL_ENVIRONMENT_FIRST_PROFILE_SPECIFIED_VALUE  = 0x01,
  EMBER_ZCL_PHYSICAL_ENVIRONMENT_LAST_PROFILE_SPECIFIED_VALUE   = 0x7F,
  EMBER_ZCL_PHYSICAL_ENVIRONMENT_UNKNOWN                    = 0xFF
} EmberAfPhysicalEnvironment;

// BatterySize
typedef enum {
  EMBER_ZCL_BATTERY_SIZE_NO_BATTERY  = 0x00,
  EMBER_ZCL_BATTERY_SIZE_BUILT_IN    = 0x01,
  EMBER_ZCL_BATTERY_SIZE_OTHER       = 0x02,
  EMBER_ZCL_BATTERY_SIZE_AA          = 0x03,
  EMBER_ZCL_BATTERY_SIZE_AAA         = 0x04,
  EMBER_ZCL_BATTERY_SIZE_C           = 0x05,
  EMBER_ZCL_BATTERY_SIZE_D           = 0x06,
  EMBER_ZCL_BATTERY_SIZE_UNKNOWN     = 0xFF
} EmberAfBatterySize;

// SwitchType
typedef enum {
  EMBER_ZCL_SWITCH_TYPE_TOGGLE     = 0x00,
  EMBER_ZCL_SWITCH_TYPE_MOMENTARY  = 0x01
} EmberAfSwitchType;

// SwitchActions
typedef enum {
  EMBER_ZCL_SWITCH_ACTIONS_ON      = 0x00,
  EMBER_ZCL_SWITCH_ACTIONS_OFF     = 0x01,
  EMBER_ZCL_SWITCH_ACTIONS_TOGGLE  = 0x02
} EmberAfSwitchActions;

```

```
// MoveMode
typedef enum {
  EMBER_ZCL_MOVE_MODE_UP    = 0x00,
  EMBER_ZCL_MOVE_MODE_DOWN  = 0x01
} EmberAfMoveMode;

// StepMode
typedef enum {
  EMBER_ZCL_STEP_MODE_UP    = 0x00,
  EMBER_ZCL_STEP_MODE_DOWN  = 0x01
} EmberAfStepMode;

// LocationMethod
typedef enum {
  EMBER_ZCL_LOCATION_METHOD_LATERATION        = 0x00,
  EMBER_ZCL_LOCATION_METHOD_SIGNPOSTING       = 0x01,
  EMBER_ZCL_LOCATION_METHOD_RF_FINGERPRINTING  = 0x02,
  EMBER_ZCL_LOCATION_METHOD_OUT_OF_BAND       = 0x03
} EmberAfLocationMethod;

// PumpOperationMode
typedef enum {
  EMBER_ZCL_PUMP_OPERATION_MODE_NORMAL   = 0x00,
  EMBER_ZCL_PUMP_OPERATION_MODE_MINIMUM  = 0x01,
  EMBER_ZCL_PUMP_OPERATION_MODE_MAXIMUM  = 0x02,
  EMBER_ZCL_PUMP_OPERATION_MODE_LOCAL    = 0x03
} EmberAfPumpOperationMode;

// PumpControlMode
typedef enum {
  EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_SPEED        = 0x00,
  EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_PRESSURE     = 0x01,
  EMBER_ZCL_PUMP_CONTROL_MODE_PROPORTIONAL_PRESSURE  = 0x02,
  EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_FLOW         = 0x03,
  EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_TEMPERATURE   = 0x05,
  EMBER_ZCL_PUMP_CONTROL_MODE_AUTOMATIC             = 0x07
} EmberAfPumpControlMode;

// ThermostatControlSequence
typedef enum {
  EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_ONLY                = 0x00,
  EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_WITH_REHEAT         = 0x01,
  EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_HEATING_ONLY                = 0x02,
  EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_HEATING_WITH_REHEAT         = 0x03,
  EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_AND_HEATING         = 0x04,
  EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_AND_HEATING_WITH_REHEAT  = 0x05
} EmberAfThermostatControlSequence;

// ThermostatSystemMode
typedef enum {
  EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_OFF              = 0x00,
  EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_AUTO             = 0x01,
  EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_COOL             = 0x03,
  EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_HEAT             = 0x04,
  EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_EMERGENCY_HEATING  = 0x05,
```

```
  EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_PRECOOLING        = 0x06,
  EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_FAN_ONLY          = 0x07
} EmberAfThermostatSystemMode;

// SetpointAdjustMode
typedef enum {
  EMBER_ZCL_SETPOINT_ADJUST_MODE_HEAT_SETPOINT           = 0x00,
  EMBER_ZCL_SETPOINT_ADJUST_MODE_COOL_SETPOINT           = 0x01,
  EMBER_ZCL_SETPOINT_ADJUST_MODE_HEAT_AND_COOL_SETPOINTS  = 0x02
} EmberAfSetpointAdjustMode;

// FanMode
typedef enum {
  EMBER_ZCL_FAN_MODE_OFF     = 0x00,
  EMBER_ZCL_FAN_MODE_LOW     = 0x01,
  EMBER_ZCL_FAN_MODE_MEDIUM  = 0x02,
  EMBER_ZCL_FAN_MODE_HIGH    = 0x03,
  EMBER_ZCL_FAN_MODE_ON      = 0x04,
  EMBER_ZCL_FAN_MODE_AUTO    = 0x05,
  EMBER_ZCL_FAN_MODE_SMART   = 0x06
} EmberAfFanMode;

// FanModeSequence
typedef enum {
  EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_MED_HIGH       = 0x00,
  EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_HIGH           = 0x01,
  EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_MED_HIGH_AUTO  = 0x02,
  EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_HIGH_AUTO      = 0x03,
  EMBER_ZCL_FAN_MODE_SEQUENCE_ON_AUTO            = 0x04
} EmberAfFanModeSequence;

// RelativeHumidityMode
typedef enum {
  EMBER_ZCL_RELATIVE_HUMIDITY_MODE_MEASURE_LOCALLY           = 0x00,
  EMBER_ZCL_RELATIVE_HUMIDITY_MODE_UPDATED_OVER_THE_NETWORK  = 0x01
} EmberAfRelativeHumidityMode;

// DehumidifcationLockout
typedef enum {
  EMBER_ZCL_DEHUMIDIFCATION_LOCKOUT_NOT_ALLOWED  = 0x00,
  EMBER_ZCL_DEHUMIDIFCATION_LOCKOUT_ALLOWED      = 0x01
} EmberAfDehumidifcationLockout;

// RelativeHumidityDisplay
typedef enum {
  EMBER_ZCL_RELATIVE_HUMIDITY_DISPLAY_NOT_DISPLAYED  = 0x00,
  EMBER_ZCL_RELATIVE_HUMIDITY_DISPLAY_DISPLAYED      = 0x01
} EmberAfRelativeHumidityDisplay;

// TemperatureDisplayMode
typedef enum {
  EMBER_ZCL_TEMPERATURE_DISPLAY_MODE_CELSIUS    = 0x00,
  EMBER_ZCL_TEMPERATURE_DISPLAY_MODE_FARENHEIT  = 0x01
} EmberAfTemperatureDisplayMode;
```

```
// KeypadLockout
typedef enum {
  EMBER_ZCL_KEYPAD_LOCKOUT_NO_LOCKOUT          = 0x00,
  EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_ONE_LOCKOUT    = 0x01,
  EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_TWO_LOCKOUT    = 0x02,
  EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_THREE_LOCKOUT  = 0x03,
  EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_FOUR_LOCKOUT   = 0x04,
  EMBER_ZCL_KEYPAD_LOCKOUT_LEVELFIVE_LOCKOUT    = 0x05
} EmberAfKeypadLockout;

// HueDirection
typedef enum {
  EMBER_ZCL_HUE_DIRECTION_SHORTEST_DISTANCE  = 0x00,
  EMBER_ZCL_HUE_DIRECTION_LONGEST_DISTANCE   = 0x01,
  EMBER_ZCL_HUE_DIRECTION_UP                 = 0x02,
  EMBER_ZCL_HUE_DIRECTION_DOWN               = 0x03
} EmberAfHueDirection;

// HueMoveMode
typedef enum {
  EMBER_ZCL_HUE_MOVE_MODE_STOP  = 0x00,
  EMBER_ZCL_HUE_MOVE_MODE_UP    = 0x01,
  EMBER_ZCL_HUE_MOVE_MODE_DOWN  = 0x03
} EmberAfHueMoveMode;

// HueStepMode
typedef enum {
  EMBER_ZCL_HUE_STEP_MODE_UP    = 0x01,
  EMBER_ZCL_HUE_STEP_MODE_DOWN  = 0x03
} EmberAfHueStepMode;

// SaturationMoveMode
typedef enum {
  EMBER_ZCL_SATURATION_MOVE_MODE_STOP  = 0x00,
  EMBER_ZCL_SATURATION_MOVE_MODE_UP    = 0x01,
  EMBER_ZCL_SATURATION_MOVE_MODE_DOWN  = 0x03
} EmberAfSaturationMoveMode;

// SaturationStepMode
typedef enum {
  EMBER_ZCL_SATURATION_STEP_MODE_UP    = 0x01,
  EMBER_ZCL_SATURATION_STEP_MODE_DOWN  = 0x03
} EmberAfSaturationStepMode;

// ColorMode
typedef enum {
  EMBER_ZCL_COLOR_MODE_CURRENT_HUE_AND_CURRENT_SATURATION  = 0x00,
  EMBER_ZCL_COLOR_MODE_CURRENT_X_AND_CURRENT_Y             = 0x01,
  EMBER_ZCL_COLOR_MODE_COLOR_TEMPERATURE                   = 0x02
} EmberAfColorMode;

// MeasurementLightSensorType
typedef enum {
  EMBER_ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_PHOTODIODE  = 0x00,
  EMBER_ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_CMOS        = 0x01
```

```
} EmberAfMeasurementLightSensorType;

// LevelStatus
typedef enum {
  EMBER_ZCL_LEVEL_STATUS_ON_TARGET     = 0x00,
  EMBER_ZCL_LEVEL_STATUS_BELOW_TARGET  = 0x01,
  EMBER_ZCL_LEVEL_STATUS_ABOVE_TARGET  = 0x02
} EmberAfLevelStatus;

// SensingLightSensorType
typedef enum {
  EMBER_ZCL_SENSING_LIGHT_SENSOR_TYPE_PHOTODIODE  = 0x00,
  EMBER_ZCL_SENSING_LIGHT_SENSOR_TYPE_CMOS        = 0x01
} EmberAfSensingLightSensorType;

// OccupancySensorType
typedef enum {
  EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_PIR                = 0x00,
  EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_ULTRASONIC         = 0x01,
  EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_PIR_AND_ULTRASONIC  = 0x02
} EmberAfOccupancySensorType;

// IasZoneState
typedef enum {
  EMBER_ZCL_IAS_ZONE_STATE_NOT_ENROLLED  = 0x00,
  EMBER_ZCL_IAS_ZONE_STATE_ENROLLED      = 0x01
} EmberAfIasZoneState;

// IasZoneType
typedef enum {
  EMBER_ZCL_IAS_ZONE_TYPE_STANDARD_CIE              = 0x0000,
  EMBER_ZCL_IAS_ZONE_TYPE_MOTION_SENSOR             = 0x000D,
  EMBER_ZCL_IAS_ZONE_TYPE_CONTACT_SWITCH            = 0x0015,
  EMBER_ZCL_IAS_ZONE_TYPE_FIRE_SENSOR               = 0x0028,
  EMBER_ZCL_IAS_ZONE_TYPE_WATER_SENSOR              = 0x002A,
  EMBER_ZCL_IAS_ZONE_TYPE_GAS_SENSOR                = 0x002B,
  EMBER_ZCL_IAS_ZONE_TYPE_PERSONAL_EMERGENCY_DEVICE  = 0x002C,
  EMBER_ZCL_IAS_ZONE_TYPE_VIBRATION_MOVEMENT_SENSOR  = 0x002D,
  EMBER_ZCL_IAS_ZONE_TYPE_REMOTE_CONTROL            = 0x010F,
  EMBER_ZCL_IAS_ZONE_TYPE_KEY_FOB                   = 0x0115,
  EMBER_ZCL_IAS_ZONE_TYPE_KEYPAD                    = 0x021D,
  EMBER_ZCL_IAS_ZONE_TYPE_STANDARD_WARNING_DEVICE    = 0x0225
} EmberAfIasZoneType;

// IasEnrollResponseCode
typedef enum {
  EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_SUCCESS          = 0x00,
  EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_NOT_SUPPORTED     = 0x01,
  EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_NO_ENROLL_PERMIT  = 0x02,
  EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_TOO_MANY_ZONES    = 0x03
} EmberAfIasEnrollResponseCode;

// IasAceArmMode
typedef enum {
  EMBER_ZCL_IAS_ACE_ARM_MODE_DISARM                    = 0x00,
```

```
  EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_DAY_HOME_ZONES_ONLY     = 0x01,
  EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_NIGHT_SLEEP_ZONES_ONLY  = 0x02,
  EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_ALL_ZONES               = 0x03
} EmberAfIasAceArmMode;

// IasAceArmNotification
typedef enum {
  EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ALL_ZONES_DISARMED           = 0x00,
  EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ONLY_DAY_HOME_ZONES_ARMED     = 0x01,
  EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ONLY_NIGHT_SLEEP_ZONES_ARMED  = 0x02,
  EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ALL_ZONES_ARMED              = 0x03
} EmberAfIasAceArmNotification;

// WarningMode
typedef enum {
  EMBER_ZCL_WARNING_MODE_STOP      = 0x00,
  EMBER_ZCL_WARNING_MODE_BURGLAR    = 0x01,
  EMBER_ZCL_WARNING_MODE_FIRE      = 0x02,
  EMBER_ZCL_WARNING_MODE_EMERGENCY  = 0x03
} EmberAfWarningMode;

// WarningStobe
typedef enum {
  EMBER_ZCL_WARNING_STOBE_NO_STROBE   = 0x00,
  EMBER_ZCL_WARNING_STOBE_USE_STROBE  = 0x01
} EmberAfWarningStobe;

// SquawkMode
typedef enum {
  EMBER_ZCL_SQUAWK_MODE_SYSTEM_IS_ARMED     = 0x00,
  EMBER_ZCL_SQUAWK_MODE_SYSTEM_IS_DISARMED  = 0x01
} EmberAfSquawkMode;

// SquawkStobe
typedef enum {
  EMBER_ZCL_SQUAWK_STOBE_NO_STROBE   = 0x00,
  EMBER_ZCL_SQUAWK_STOBE_USE_STROBE  = 0x01
} EmberAfSquawkStobe;

// SquawkLevel
typedef enum {
  EMBER_ZCL_SQUAWK_LEVEL_LOW_LEVEL       = 0x00,
  EMBER_ZCL_SQUAWK_LEVEL_MEDIUM_LEVEL     = 0x01,
  EMBER_ZCL_SQUAWK_LEVEL_VERY_HIGH_LEVEL  = 0x02
} EmberAfSquawkLevel;

// AmiCriticalityLevel
typedef enum {
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_RESERVED          = 0x00,
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_GREEN             = 0x01,
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_1                 = 0x02,
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_2                 = 0x03,
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_3                 = 0x04,
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_4                 = 0x05,
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_5                 = 0x06,
```

```
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_EMERGENCY          = 0x07,
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_PLANNED_OUTAGE     = 0x08,
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_SERVICE_DISCONNECT  = 0x09,
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED1    = 0x0A,
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED2    = 0x0B,
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED3    = 0x0C,
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED4    = 0x0D,
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED5    = 0x0E,
  EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED6    = 0x0F
} EmberAfAmiCriticalityLevel;

// AmiEventStatus
typedef enum {
  EMBER_ZCL_AMI_EVENT_STATUS_LOAD_CONTROL_EVENT_COMMAND_RX                    = 0x01,
  EMBER_ZCL_AMI_EVENT_STATUS_EVENT_STARTED                                    = 0x02,
  EMBER_ZCL_AMI_EVENT_STATUS_EVENT_COMPLETED                                  = 0x03,
  EMBER_ZCL_AMI_EVENT_STATUS_USER_HAS_CHOOSE_TO_OPT_OUT                       = 0x04,
  EMBER_ZCL_AMI_EVENT_STATUS_USER_HAS_CHOOSE_TO_OPT_IN                        = 0x05,
  EMBER_ZCL_AMI_EVENT_STATUS_THE_EVENT_HAS_BEEN_CANCELED                      = 0x06,
  EMBER_ZCL_AMI_EVENT_STATUS_THE_EVENT_HAS_BEEN_SUPERSEDED                    = 0x07,
  EMBER_ZCL_AMI_EVENT_STATUS_EVENT_PARTIALLY_COMPLETED_WITH_USER_OPT_OUT
= 0x08,
  EMBER_ZCL_AMI_EVENT_STATUS_EVENT_PARTIALLY_COMPLETED_DUE_TO_USER_OPT_IN
= 0x09,

EMBER_ZCL_AMI_EVENT_STATUS_EVENT_COMPLETED_NO_USER_PARTICIPATION_PREVIOUS_OPT_OUT
= 0x0A,
  EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_COMMAND                  = 0xF8,

EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_COMMAND_INVALID_EFFECTIVE_TIME
= 0xF9,
  EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_EVENT_EXPIRED                           = 0xFB,
  EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_UNDEFINED_EVENT          = 0xFD,
  EMBER_ZCL_AMI_EVENT_STATUS_LOAD_CONTROL_EVENT_COMMAND_REJECTED              = 0xFE
} EmberAfAmiEventStatus;

// AmiUnitOfMeasure
typedef enum {
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_KILO_WATT_HOURS              = 0x00,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_METER_PER_HOUR         = 0x01,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_FEET_PER_HOUR          = 0x02,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_CENTUM_CUBIC_FEET_PER_HOUR   = 0x03,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_US_GALLONS_PER_HOUR          = 0x04,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_IMPERIAL_GALLONS_PER_HOUR    = 0x05,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_BT_US_OR_BTU_PER_HOUR        = 0x06,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_LITERS_OR_LITERS_PER_HOUR    = 0x07,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_GAUGE                    = 0x08,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_ABSOLUTE                 = 0x09,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_MCF_OR_MCF_PER_SECOND        = 0x0A,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_UNITLESS                     = 0x0B,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_MJ_OR_MJ_PER_SECOND          = 0x0C,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_KILO_WATT_HOURS_BCD          = 0x80,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_METER_PER_HOUR_BCD     = 0x81,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_FEET_PER_HOUR_BCD      = 0x82,
```

```
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_CENTUM_CUBIC_FEET_PER_HOUR_BCD  = 0x83,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_US_GALLONS_PER_HOUR_BCD         = 0x84,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_IMPERIAL_GALLONS_PER_HOUR_BCD   = 0x85,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_BT_US_OR_BTU_PER_HOUR_BCD       = 0x86,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_LITERS_OR_LITERS_PER_HOUR_BCD   = 0x87,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_GUAGE_BCD                   = 0x88,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_ABSOLUTE_BCD                = 0x89,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_MCF_OR_MCF_PER_SECOND_BCD       = 0x8A,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_UNITLESS_BCD                    = 0x8B,
  EMBER_ZCL_AMI_UNIT_OF_MEASURE_MJ_OR_MJ_PER_SECOND_BCD         = 0x8C
} EmberAfAmiUnitOfMeasure;

// AmiIntervalPeriod
typedef enum {
  EMBER_ZCL_AMI_INTERVAL_PERIOD_DAILY       = 0x00,
  EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES60   = 0x01,
  EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES30   = 0x02,
  EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES15   = 0x03,
  EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES10   = 0x04,
  EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES7P5  = 0x05,
  EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES5    = 0x06,
  EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES2P5  = 0x07
} EmberAfAmiIntervalPeriod;

// AmiIntervalChannel
typedef enum {
  EMBER_ZCL_AMI_INTERVAL_CHANNEL_CONSUMPTION_DELIVERED  = 0x00,
  EMBER_ZCL_AMI_INTERVAL_CHANNEL_CONSUMPTION_RECEIVED   = 0x01
} EmberAfAmiIntervalChannel;

// AmiRegistrationState
typedef enum {
  EMBER_ZCL_AMI_REGISTRATION_STATE_UNREGISTERED                  = 0x00,
  EMBER_ZCL_AMI_REGISTRATION_STATE_JOINING_NETWORK               = 0x01,
  EMBER_ZCL_AMI_REGISTRATION_STATE_JOINED_NETWORK                = 0x02,
  EMBER_ZCL_AMI_REGISTRATION_STATE_SUBMITTED_REGISTRATION_REQUEST  = 0x03,
  EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTRATION_REJECTED         = 0x04,
  EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTERED                    = 0x05,
  EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTERATION_NOT_POSSIBLE    = 0x06
} EmberAfAmiRegistrationState;

// AnonymousDataState
typedef enum {
  EMBER_ZCL_ANONYMOUS_DATA_STATE_NO_SOURCE_FOUND  = 0x00,
  EMBER_ZCL_ANONYMOUS_DATA_STATE_SOURCE_FOUND     = 0x01
} EmberAfAnonymousDataState;

// PriceTier
typedef enum {
  EMBER_ZCL_PRICE_TIER_NO_TIER_RELATED     = 0x00,
  EMBER_ZCL_PRICE_TIER_TIER1_PRICE_LABEL   = 0x01,
  EMBER_ZCL_PRICE_TIER_TIER2_PRICE_LABEL   = 0x02,
  EMBER_ZCL_PRICE_TIER_TIER3_PRICE_LABEL   = 0x03,
  EMBER_ZCL_PRICE_TIER_TIER4_PRICE_LABEL   = 0x04,
  EMBER_ZCL_PRICE_TIER_TIER5_PRICE_LABEL   = 0x05,
```

```
  EMBER_ZCL_PRICE_TIER_TIER6_PRICE_LABEL   = 0x06,
  EMBER_ZCL_PRICE_TIER_TIER7_PRICE_LABEL   = 0x07,
  EMBER_ZCL_PRICE_TIER_TIER8_PRICE_LABEL   = 0x08,
  EMBER_ZCL_PRICE_TIER_TIER9_PRICE_LABEL   = 0x09,
  EMBER_ZCL_PRICE_TIER_TIER10_PRICE_LABEL  = 0x0A,
  EMBER_ZCL_PRICE_TIER_TIER11_PRICE_LABEL  = 0x0B,
  EMBER_ZCL_PRICE_TIER_TIER12_PRICE_LABEL  = 0x0C,
  EMBER_ZCL_PRICE_TIER_TIER13_PRICE_LABEL  = 0x0D,
  EMBER_ZCL_PRICE_TIER_TIER14_PRICE_LABEL  = 0x0E,
  EMBER_ZCL_PRICE_TIER_TIER15_PRICE_LABEL  = 0x0F
} EmberAfPriceTier;

// PriceControlAcknowledgement
typedef enum {
  EMBER_ZCL_PRICE_CONTROL_ACKNOWLEDGEMENT_NOT_REQUIRED  = 0x00,
  EMBER_ZCL_PRICE_CONTROL_ACKNOWLEDGEMENT_REQUIRED      = 0x01
} EmberAfPriceControlAcknowledgement;

// RegisterTier
typedef enum {
  EMBER_ZCL_REGISTER_TIER_NO_TIER_RELATED                          = 0x00,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER1_SUMMATION_DELIVERED_ATTRIBUTE   = 0x01,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER2_SUMMATION_DELIVERED_ATTRIBUTE   = 0x02,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER3_SUMMATION_DELIVERED_ATTRIBUTE   = 0x03,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER4_SUMMATION_DELIVERED_ATTRIBUTE   = 0x04,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER5_SUMMATION_DELIVERED_ATTRIBUTE   = 0x05,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER6_SUMMATION_DELIVERED_ATTRIBUTE   = 0x06,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER7_SUMMATION_DELIVERED_ATTRIBUTE   = 0x07,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER8_SUMMATION_DELIVERED_ATTRIBUTE   = 0x08,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER9_SUMMATION_DELIVERED_ATTRIBUTE   = 0x09,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER10_SUMMATION_DELIVERED_ATTRIBUTE  = 0x0A,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER11_SUMMATION_DELIVERED_ATTRIBUTE  = 0x0B,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER12_SUMMATION_DELIVERED_ATTRIBUTE  = 0x0C,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER13_SUMMATION_DELIVERED_ATTRIBUTE  = 0x0D,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER14_SUMMATION_DELIVERED_ATTRIBUTE  = 0x0E,
  EMBER_ZCL_REGISTER_TIER_CURRENT_TIER15_SUMMATION_DELIVERED_ATTRIBUTE  = 0x0F
} EmberAfRegisterTier;

// AlternateCostUnit
typedef enum {
  EMBER_ZCL_ALTERNATE_COST_UNIT_KG_OF_CO2_PER_UNIT_OF_MEASURE  = 0x02
} EmberAfAlternateCostUnit;

// AmiKeyEstablishmentStatus
typedef enum {
  EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_SUCCESS           = 0x00,
  EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_UNKNOWN_ISSUER    = 0x01,
  EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_BAD_KEY_CONFIRM   = 0x02,
  EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_BAD_MESSAGE       = 0x03,
  EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_NO_RESOURCES      = 0x04,
  EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_UNSUPPORTED_SUITE  = 0x05
} EmberAfAmiKeyEstablishmentStatus;

// AmiGetProfileStatus
```

```
typedef enum {
  EMBER_ZCL_AMI_GET_PROFILE_STATUS_SUCCESS                                = 0x00,
  EMBER_ZCL_AMI_GET_PROFILE_STATUS_UNDEFINED_INTERVAL_CHANNEL_REQUESTED          = 0x01,
  EMBER_ZCL_AMI_GET_PROFILE_STATUS_INTERVAL_CHANNEL_NOT_SUPPORTED               = 0x02,
  EMBER_ZCL_AMI_GET_PROFILE_STATUS_INVALID_END_TIME                       = 0x03,
  EMBER_ZCL_AMI_GET_PROFILE_STATUS_MORE_PERIODS_REQUESTED_THAN_CAN_BE_RETURNED
= 0x04,
  EMBER_ZCL_AMI_GET_PROFILE_STATUS_NO_INTERVALS_AVAILABLE_FOR_THE_REQUESTED_TIME
= 0x05
} EmberAfAmiGetProfileStatus;

// AttributeWritePermission
typedef enum {
  EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_DENY_WRITE              = 0x00,
  EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_ALLOW_WRITE_NORMAL       = 0x01,
  EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_ALLOW_WRITE_OF_READ_ONLY  = 0x02
} EmberAfAttributeWritePermission;

// MeterDeviceType
typedef enum {
  EMBER_ZCL_METER_DEVICE_TYPE_ELECTRIC_METER          = 0x00,
  EMBER_ZCL_METER_DEVICE_TYPE_GAS_METER               = 0x01,
  EMBER_ZCL_METER_DEVICE_TYPE_WATER_METER             = 0x02,
  EMBER_ZCL_METER_DEVICE_TYPE_THERMAL_METER           = 0x03,
  EMBER_ZCL_METER_DEVICE_TYPE_PRESSURE_METER          = 0x04,
  EMBER_ZCL_METER_DEVICE_TYPE_HEAT_METER              = 0x05,
  EMBER_ZCL_METER_DEVICE_TYPE_COOLING_METER           = 0x06,
  EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_GAS_METER      = 0x80,
  EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_WATER_METER    = 0x81,
  EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_THERMAL_METER  = 0x82,
  EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_PRESSURE_METER = 0x83,
  EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_HEAT_METER     = 0x84,
  EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_COOLING_METER  = 0x85,
  EMBER_ZCL_METER_DEVICE_TYPE_UNDEFINED_MIRROR_METER  = 0xFE
} EmberAfMeterDeviceType;

// TunnelingProtocolId
typedef enum {
  EMBER_ZCL_TUNNELING_PROTOCOL_ID_DLMS_COSEM    = 0x00,
  EMBER_ZCL_TUNNELING_PROTOCOL_ID_IEC_61107     = 0x01,
  EMBER_ZCL_TUNNELING_PROTOCOL_ID_ANSI_C12      = 0x02,
  EMBER_ZCL_TUNNELING_PROTOCOL_ID_M_BUS         = 0x03,
  EMBER_ZCL_TUNNELING_PROTOCOL_ID_SML           = 0x04,
  EMBER_ZCL_TUNNELING_PROTOCOL_ID_CLIMATE_TALK  = 0x05
} EmberAfTunnelingProtocolId;

// TunnelingTransferDataStatus
typedef enum {
  EMBER_ZCL_TUNNELING_TRANSFER_DATA_STATUS_NO_SUCH_TUNNEL  = 0x00,
  EMBER_ZCL_TUNNELING_TRANSFER_DATA_STATUS_WRONG_DEVICE    = 0x01
} EmberAfTunnelingTransferDataStatus;

// TunnelingTunnelStatus
typedef enum {
  EMBER_ZCL_TUNNELING_TUNNEL_STATUS_SUCCESS                = 0x00,
```

```
  EMBER_ZCL_TUNNELING_TUNNEL_STATUS_BUSY                    = 0x01,
  EMBER_ZCL_TUNNELING_TUNNEL_STATUS_NO_MORE_TUNNEL_IDS        = 0x02,
  EMBER_ZCL_TUNNELING_TUNNEL_STATUS_PROTOCOL_NOT_SUPPORTED     = 0x03,
  EMBER_ZCL_TUNNELING_TUNNEL_STATUS_FLOW_CONTROL_NOT_SUPPORTED  = 0x04
} EmberAfTunnelingTunnelStatus;

// PrepaymentRecoveryPeriod
typedef enum {
  EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_HOUR     = 0x00,
  EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_DAY      = 0x01,
  EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_WEEK     = 0x02,
  EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_MONTH    = 0x03,
  EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_QUARTER  = 0x04
} EmberAfPrepaymentRecoveryPeriod;

// PrepaymentOriginatingDevice
typedef enum {
  EMBER_ZCL_PREPAYMENT_ORIGINATING_DEVICE_ENERGY_SERVICE_INTERFACE    = 0x00,
  EMBER_ZCL_PREPAYMENT_ORIGINATING_DEVICE_IN_PREMISES_DISPLAY_DEVICE  = 0x02
} EmberAfPrepaymentOriginatingDevice;

// PrepaymentSupplyStatus
typedef enum {
  EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_OFF        = 0x00,
  EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_OFF_ARMED  = 0x01,
  EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_ON         = 0x02
} EmberAfPrepaymentSupplyStatus;

// ReportingDirection
typedef enum {
  EMBER_ZCL_REPORTING_DIRECTION_REPORTED  = 0x00,
  EMBER_ZCL_REPORTING_DIRECTION_RECEIVED  = 0x01
} EmberAfReportingDirection;

// SignatureType
typedef enum {
  EMBER_ZCL_SIGNATURE_TYPE_RESERVED  = 0x00,
  EMBER_ZCL_SIGNATURE_TYPE_ECDSA     = 0x01
} EmberAfSignatureType;

#endif // __EMBER_AF_ENUMS__
```

generic-tunnel.h

Go to the documentation of this file.

```
// ********************************************************************
// * generic-tunnel.h
// *
// *
// * Copyright 2009 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************

#ifndef EMBER_AF_PLUGIN_GENERIC_TUNNEL_PROTOCOL_ADDRESS_SIZE
#define EMBER_AF_PLUGIN_GENERIC_TUNNEL_PROTOCOL_ADDRESS_SIZE 8
#endif

void emAfPluginGenericTunnelPrintInfo(void);
int8u *emAfPluginGenericTunnelGetProtocolAddress(int8u endpoint);
```

groups-server.h

Go to the documentation of this file.

```
// ********************************************************************
// * groups-server.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************

void emAfPluginGroupsServerPrintInfo(void);
```

ias-ace-server.h

Go to the documentation of this file.

```
// ******************************************************************
// * ias-ace-server.h
// *
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.          *80*
// ******************************************************************

void emAfIasAcePrintInfo(void);
```

ias-wd-server.h

Go to the documentation of this file.

```
// ***********************************************************************
// * ias-wd.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.             *80*
// ***********************************************************************

// ----------------------------------------------------------------------------
// Bits
// ----------------------------------------------------------------------------
#define ZCL_WARNING_INFO_MODE_MASK 0xF0
#define ZCL_WARNING_INFO_STOP      0x00
#define ZCL_WARNING_INFO_BURGLAR   0x10
#define ZCL_WARNING_INFO_FIRE      0x20
#define ZCL_WARNING_INFO_EMERGENCY 0x30

#define ZCL_WARNING_INFO_STROBE_MASK 0x0C
#define ZCL_WARNING_INFO_NO_STROBE   0x00
#define ZCL_WARNING_INFO_STROBE      0x04

#define ZCL_SQUAWK_INFO_MODE_MASK 0xF0
#define ZCL_SQUAWK_INFO_ARMED     0x00
#define ZCL_SQUAWK_INFO_DISARMED  0x10

#define ZCL_SQUAWK_INFO_STROBE_MASK 0x08
#define ZCL_SQUAWK_INFO_NO_STROBE   0x00
#define ZCL_SQUAWK_INFO_STROBE      0x08

#define ZCL_SQUAWK_INFO_LEVEL_MASK 0x03
#define ZCL_SQUAWK_INFO_LOW        0x00
#define ZCL_SQUAWK_INFO_MEDIUM     0x01
#define ZCL_SQUAWK_INFO_HIGH       0x02
#define ZCL_SQUAWK_INFO_VERY_HIGH  0x03
```

identify.h

Go to the documentation of this file.

```
// ******************************************************************
// * identify.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.          *80*
// ******************************************************************

boolean emberAfIsDeviceIdentifying(int8u endpoint);
void emAfIdentifyPrintInfo(void);
```

interpan.h

Go to the documentation of this file.

```
// The three types of inter-PAN messages.  The values are actually the
// corresponding APS frame controls.
//
// 0x03 is the special interPAN message type.  Unicast mode is 0x00,
// broadcast mode is 0x08, and multicast mode is 0x0C.
//

// MAC Frame Max size
// - Frame control   (2-bytes)
// - Sequence        (1-byte)
// - Dest PAN ID     (2-bytes)
// - Dest long       (8-bytes)
// - Source PAN ID   (2-bytes)
// - Source long     (8-bytes)
#define MAX_INTER_PAN_MAC_SIZE 23

// NWK stub frame has two control bytes.
#define STUB_NWK_SIZE 2
#define STUB_NWK_FRAME_CONTROL 0x000B

// Interpan APS Unicast
//  - Frame Control   (1-byte)
//  - Cluster ID      (2-bytes)
//  - Profile ID      (2-bytes)
#define INTERPAN_APS_UNICAST_SIZE 5

// Interpan APS Broadcast, same as unicast
#define INTERPAN_APS_BROADCAST_SIZE 5

// Interpan APS Multicast
//  - Frame Control   (1-byte)
//  - Group ID        (2-bytes)
//  - Cluster ID      (2-bytes)
//  - Profile ID      (2-bytes)
#define INTERPAN_APS_MULTICAST_SIZE  7

#define MAX_STUB_APS_SIZE (INTERPAN_APS_MULTICAST_SIZE)
#define MIN_STUB_APS_SIZE (INTERPAN_APS_UNICAST_SIZE)

#define MAX_INTER_PAN_HEADER_SIZE \
  (MAX_INTER_PAN_MAC_SIZE + STUB_NWK_SIZE + MAX_STUB_APS_SIZE)

#define INTERPAN_APS_FRAME_TYPE      0x03
#define INTERPAN_APS_FRAME_TYPE_MASK 0x03

// The only allowed APS FC value (without the delivery mode subfield)
#define INTERPAN_APS_FRAME_CONTROL_NO_DELIVERY_MODE (INTERPAN_APS_FRAME_TYPE)
```

```
#define INTERPAN_APS_FRAME_DELIVERY_MODE_MASK 0x0C
#define INTERPAN_APS_FRAME_SECURITY          0x20

// 5 byte AUX header + 4 byte MIC
#define INTERPAN_APS_ENCRYPTION_OVERHEAD      (5 + 4)

boolean emAfPluginInterpanProcessMessage(int8u messageLength,
                                         int8u *messageContents);

EmberStatus emAfPluginInterpanSendRawMessage(int8u length, int8u* message);

EmberStatus emAfInterpanApsCryptMessage(boolean encrypt,
                                        int8u* apsFrame,
                                        int8u* messageLength,
                                        int8u apsHeaderLength,
                                        EmberEUI64 remoteEui64);
```

key-establishment-storage.h

Go to the documentation of this file.

```
// ********************************************************************
// * key-establishment-storage.h
// *
// * API for the storage of public temporary partner data.
// * - Partner Certificate
// * - Partner Ephemeral Public Key
// * - A single SMAC
// *
// * Copyright 2008 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************

// If isCertificate is FALSE, data is a public key.
boolean storePublicPartnerData(boolean isCertificate,
                               int8u* data);
boolean retrieveAndClearPublicPartnerData(EmberCertificateData* partnerCertificate,
                                          EmberPublicKeyData* partnerEphemeralPublicKey);

boolean storeSmac(EmberSmacData* smac);
boolean getSmacPointer(EmberSmacData** smacPtr);

void clearAllTemporaryPublicData(void);
```

key-establishment.h

Go to the documentation of this file.

```
// ********************************************************************
// * key-establishment.h
// *
// *
// * Copyright 2008 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************


// Init - bytes: suite (2), key gen time (1), derive secret time (1), cert (48)
#define EM_AF_KE_INITIATE_SIZE (2 + 1 + 1 + EMBER_CERTIFICATE_SIZE)

#define EM_AF_KE_EPHEMERAL_SIZE EMBER_PUBLIC_KEY_SIZE
#define EM_AF_KE_SMAC_SIZE      EMBER_SMAC_SIZE

  // Terminate - bytes: status (1), wait time (1), suite (2)
#define EM_AF_KE_TERMINATE_SIZE (1 + 1 + 2)

```

level-control.h

Go to the documentation of this file.

```
// ********************************************************************
// * level-control.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************

// Rate of level control tick execution.
// To increase tick frequency (for more granular updates of device state based
// on level), increase LC_TICKS_PER_SECOND.
#define LC_TICKS_PER_SECOND 32
#define LC_TICK_TIME (MILLISECOND_TICKS_PER_SECOND / LC_TICKS_PER_SECOND)

#define LEVEL_CONTROL_MINIMUM_LEVEL 0x00
#define LEVEL_CONTROL_MAXIMUM_LEVEL 0xFF
```

load-control-event-table.h

Go to the documentation of this file.

```
// ********************************************************************
// * load-control-event-table.h
// *
// * The Demand Response Load Control Event Table is responsible
// * for keeping track of all load control events scheduled
// * by the Energy Service Provider. This module provides
// * interfaces used to schedule and inform load shedding
// * devices of scheduled events.
// *
// * Any code that uses this event table is responsible for
// * providing four things:
// *   1. frequent calls to eventTableTick(), one per millisecond
// *      will do. These calls are used to drive the
// *      table's timing mechanism.
// *   2. A way to get the real time by implementing
// *      getCurrentTime(int32u *currentTime);
// *   3. An implementation of eventAction which
// *      will be called whenever event status changes
// *
// * The load control event table expects that currentTime, startTime
// * and randomization are provided in seconds. And that duration is
// * provided in minutes.
// *
// * The implementing code is responsible for over the
// * air communication based on event status changes
// * reported through the eventAction interface
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************

#ifndef __LOAD_CONTROL_EVENT_TABLE_H__
#define __LOAD_CONTROL_EVENT_TABLE_H__

// include global header for public LoadControlEvent struct
#include "../../include/af.h"

#define LOAD_CONTROL_EVENT_TABLE_ENTRY_BYTES 29
#define LOAD_CONTROL_EVENT_TABLE_MEM_FOOTPRINT
(EMBER_AF_PLUGIN_DRLC_EVENT_TABLE_SIZE * \
  LOAD_CONTROL_EVENT_EVENT_TABLE_ENTRY_BYTES)

#define RANDOMIZE_START_TIME_FLAG     1
```

```
00042 #define RANDOMIZE_END_TIME_FLAG       2
00043
00044 #define CANCEL_WITH_RANDOMIZATION       1
00045
00046 // Table entry status
00047 enum {
00048   ENTRY_VOID,
00049   ENTRY_SCHEDULED,
00050   ENTRY_STARTED,
00051   ENTRY_IS_SUPERSEDED_EVENT,
00052   ENTRY_IS_CANCELLED_EVENT
00053 };
00054
00055 enum {
00056   EVENT_OPT_FLAG_OPT_IN                        = 0x01,
00057   EVENT_OPT_FLAG_PARTIAL                        = 0x02
00058 };
00059
00060
00061 // EVENT TABLE API
00062
00063 void afLoadControlEventTableInit(int8u endpoint);
00064
00065
00077 void emAfScheduleLoadControlEvent(EmberAfLoadControlEvent *event);
00078
00084 void emAfLoadControlEventTableTick(int8u endpoint);
00085
00092 boolean emAfCancelAllLoadControlEvents(int8u cancelControl);
00093
00099 void emAfCancelLoadControlEvent(int32u eventId, int8u cancelControl, int32u effectiveTime);
00100
00126 void afScheduleCancelEvent(EmberAfLoadControlEvent *e);
00127
00131 void emAfLoadControlEventOptInOrOut(int32u eventId, boolean optIn);
00132
00133 // The module using this table is responsible for providing the following
00134 // functions.
00142 void emberAfEventAction(EmberAfLoadControlEvent *event,
00143                        int8u eventStatus,
00144                        int8u sequenceNumber);
00145
00150 void emAfNoteSignatureFailure(void);
00151
00154 void emAfLoadControlEventTablePrint(void);
00155
```

```
#endif
```

messaging-client-callback.h

Go to the documentation of this file.

```
00001 // Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
00002 //
00003 // This file is generated. Please do not edit manually.
00004 //
00005 // This file is generated from java code via the following methods:
00006 // com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generatePluginCallbackPrototypes()
00007 //
00008
00024 void emberAfPluginMessagingClientDisplayMessageCallback(EmberAfPluginMessagingClientMessage * message);
00025
00034 void emberAfPluginMessagingClientCancelMessageCallback(EmberAfPluginMessagingClientMessage * message);
00035
00036
00037
```

messaging-client.h

Go to the documentation of this file.

```
// ***********************************************************************
// * messaging-client.h
// *
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.             *80*
// ***********************************************************************

// ----------------------------------------------------------------------------
// Message Control byte
// ----------------------------------------------------------------------------

#define ZCL_MESSAGING_CLUSTER_TRANSMISSION_MASK (BIT(1) | BIT(0))
#define ZCL_MESSAGING_CLUSTER_IMPORTANCE_MASK   (BIT(3) | BIT(2))
#define ZCL_MESSAGING_CLUSTER_CONFIRMATION_MASK BIT(7)

#define ZCL_MESSAGING_CLUSTER_START_TIME_NOW        0x00000000UL
#define ZCL_MESSAGING_CLUSTER_END_TIME_NEVER        0xFFFFFFFFUL
#define ZCL_MESSAGING_CLUSTER_DURATION_UNTIL_CHANGED 0xFFFF

void emAfPluginMessagingClientPrintInfo(void);

EmberAfStatus emberAfPluginMessagingClientConfirmMessage(void);
```

network-cli.h

Go to the documentation of this file.

```
// File: network-cli.h
//
// Copyright 2007 by Ember Corporation. All rights reserved.                *80*

extern EmberCommandEntry networkCommands[];

void emberAfPrintChannelListFromMask(int32u channelMask);
```

on-off.h

Go to the documentation of this file.

```
// ******************************************************************
// * on-off.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.          *80*
// ******************************************************************

EmberAfStatus emberAfOnOffSetValue(int8u endpoint,
                                   int8u command,
                                   boolean initiatedByLevelChange);
```

ota-bootload-ncp.h

Go to the documentation of this file.

```
// ***************************************************************************
// * ota-bootload-ncp.h
// *
// * SPI/UART Interface to bootloading the NCP from a connected host.
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.            *80*
// ****************************************************************************


boolean emAfStartNcpBootloaderCommunications(void);
void emAfPostNcpBootload(boolean success);
boolean emAfRebootNcpAfterBootload(void);

// These primitives are called by the Xmodem code to pass data via
// the specific mechanism (UART or SPI).
boolean emAfBootloadSendData(const int8u *data, int16u length);
boolean emAfBootloadSendByte(int8u byte);
boolean emAfBootloadWaitChar(int8u *data, boolean expect, int8u expected);
```

ota-bootload-xmodem.h

Go to the documentation of this file.

```
// ***************************************************************************
// * ota-bootload-xmodem.c
// *
// * Routines for sending data via xmodem
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.             *80*
// ****************************************************************************

// Initialize xmodem state machine for a new transfer
// If startImmediately is set, will not wait for an initial 'C' character
//   to be received before sending the first block of data
void emAfInitXmodemState(boolean startImmediately);

// Send a chunk of data via xmodem.  Arbitrary lengths of data may be passed,
//   it will be broken up into appropriate sized chunks for transmission. Xmodem
//   itself transfers data in 128 byte chunks
// Note: This function will block for the duration of time required to send
//   the data that is passed in.
// The "finished" flag should be set when called with the final chunk to be
//   transferred to terminate the transfer properly
boolean emAfSendXmodemData(const int8u *data, int16u length, boolean finished);
```

ota-cli.h

Go to the documentation of this file.

```
// ********************************************************************
// * ota-cli.h
// *
// * Zigbee Over-the-air bootload cluster for upgrading firmware and
// * downloading specific file.  This is the CLI to interact with the
// * main cluster code.
// *
// * Copyright 2009 by Ember Corporation. All rights reserved.              *80*
// ********************************************************************


// Common CLI interface

#if defined (EMBER_AF_PLUGIN_OTA_CLIENT) || \
    defined (EMBER_AF_PLUGIN_OTA_SERVER)
  extern EmberCommandEntry otaCommands[];
  #define OTA_COMMANDS \
{"ota", NULL, (PGM_P)otaCommands},
#else
  #define OTA_COMMANDS
#endif

void emAfOtaPrintAllImages(void);
EmberAfOtaImageId emAfOtaFindImageIdByIndex(int8u index);

// Client CLI interface

#if defined (EMBER_AF_PLUGIN_OTA_CLIENT)
  extern EmberCommandEntry otaClientCommands[];
  #define OTA_CLIENT_COMMANDS \
{ "client", NULL, (PGM_P)otaClientCommands},
#else
  #define OTA_CLIENT_COMMANDS
#endif

void otaFindServerCommand(void);
void otaQueryServerCommand(void);
void otaUsePageRequestCommand(void);
```

```
00039 void otaQuerySpecificFileCommand(void);
00040 void otaSendUpgradeCommand(void);
00041 void emAfOtaImageDelete(void);
00042
00043 // Server CLI interface
00044
00045 #if defined (EMBER_AF_PLUGIN_OTA_SERVER)
00046   extern EmberCommandEntry otaServerCommands[];
00047   #define OTA_SERVER_COMMANDS \
00048 { "server", NULL, (PGM_P)otaServerCommands},
00049 #else
00050   #define OTA_SERVER_COMMANDS
00051 #endif
00052
00053 void otaImageNotifyCommand(void);
```

ota-client-page-request-test.h

Go to the documentation of this file.

```
00001
00002
00003 void pageRequestTest(void);
```

ota-client-page-request.h

Go to the documentation of this file.

```
// ******************************************************************************
// * ota-client-page-request.h
// *
// * Zigbee Over-the-air bootload cluster for upgrading firmware and
// * downloading device specific file.
// * This file handles the page request feature for the client.
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.            *80*
// ******************************************************************************

enum {
  EM_AF_NO_PAGE_REQUEST            = 0,
  EM_AF_WAITING_PAGE_REQUEST_REPLIES = 1,
  EM_AF_RETRY_MISSED_PACKETS        = 2,
  EM_AF_PAGE_REQUEST_COMPLETE       = 3,
  EM_AF_PAGE_REQUEST_ERROR          = 0xFF
};
typedef int8u EmAfPageRequestClientStatus;

#define EM_AF_PAGE_REQUEST_BLOCK_SIZE 32

// This routine returns a timer indicating how long we should wait for
// the page request responses to come in.  0 if there was an error.
int32u emAfInitPageRequestClient(int32u offsetForPageRequest,
                                 int32u totalImageSize);
void emAfPageRequestTimerExpired(void);
boolean emAfHandlingPageRequestClient(void);
EmAfPageRequestClientStatus emAfGetCurrentPageRequestStatus(void);
EmAfPageRequestClientStatus emAfNoteReceivedBlockForPageRequestClient(int32u offset);
EmAfPageRequestClientStatus emAfNextMissedBlockRequestOffset(int32u* nextOffset);

int32u emAfGetPageRequestMissedPacketDelayMs(void);
int32u emAfGetFinishedPageRequestOffset(void);
void emAfAbortPageRequest(void);
```

ota-client-policy.h

Go to the documentation of this file.

```
// ******************************************************************************
// * ota-client-policy.h
// *
// * Config for Zigbee Over-the-air bootload cluster for upgrading firmware and
// * downloading device specific file.
// *
// * This file defines the interface for the customer's application to
// * control the behavior of the OTA client.
// *
// * Copyright 2009 by Ember Corporation. All rights reserved.              *80*
// ******************************************************************************


// Note: EMBER_AF_MANUFACTURER_CODE defined in client's config

#if !defined(EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_IMAGE_TYPE_ID)
  #define EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_IMAGE_TYPE_ID    0x5678
#endif

#if !defined(EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_FIRMWARE_VERSION)
  #define EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_FIRMWARE_VERSION 0x00000005L
#endif

#define SECONDS_IN_MS (1000L)
#define MINUTES_IN_MS (60 * SECONDS_IN_MS)
#define HOURS_IN_MS (60 * MINUTES_IN_MS)

// By default if hardware version is not defined, it is not used.
// Most products do not limit upgrade images based on the version.
// Instead they have different images for different hardware.  However
// this can provide support for an image that only supports certain hardware
// revision numbers.
#define EMBER_AF_INVALID_HARDWARE_VERSION 0xFFFF
#if !defined(EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_HARDWARE_VERSION)
  #define EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_HARDWARE_VERSION EMBER_AF_INVALID_HARDWARE_VERSION
#endif

//------------------------------------------------------------------------------
// Custom Application Callbacks

enum {
  EMBER_AF_OTA_DOWNLOAD_AND_VERIFY_SUCCESS = 0,
```

```
EMBER_AF_OTA_DOWNLOAD_TIME_OUT          = 1,
EMBER_AF_OTA_VERIFY_FAILED              = 2,
EMBER_AF_OTA_SERVER_ABORTED             = 3,
EMBER_AF_OTA_CLIENT_ABORTED             = 4,
};
typedef int8u EmberAfOtaDownloadResult;
```

ota-client-signature-verify.h

Go to the documentation of this file.

```
00001
00002
00003 // Hashing a file can be expensive for small processors. The maxHashCalculations
00004 // determines how many iterations are made before returning back to the caller.
00005 // A value of 0 indicates to completely calculate the digest before returning.
00006 // A value greater than 0 means that a number of hashes will be performed and
00007 // then the routine will return EMBER_AF_OTA_VERIFICATION_IN_PROGRESS.  In that
00008 // case it is expected that this function must be called repeatedly until it
00009 // returns another status code.
00010 // When EMBER_AF_OTA_VERIFICATION_WAIT is returned then no further calls
00011 // are necessary.  The verify code will fire the callback
00012 // emAfOtaVerifyStoredDataFinish() when it is has a result.
00013 EmberAfImageVerifyStatus emAfOtaImageSignatureVerify(int16u maxHashCalculations,
00014                                                 const EmberAfOtaImageId* id,
00015                                                 boolean newVerification);
00016
00017 // This is the maximum number of digest calculations we perform per call to
00018 // emAfOtaImageSignatureVerify().  Arbitrary chosen value to limit
00019 // time spent in this routine.  A value of 0 means we will NOT return
00020 // until we are completely done with our digest calculations.
00021 // Empirically waiting until digest calculations are complete can
00022 // take quite a while for EZSP hosts (~40 seconds for a UART connected host).
00023 // So we want to make sure that other parts of the framework can run during
00024 // this time.  On SOC systems a similar problem occurs.  If we set this to 0
00025 // then emberTick() will not fire and therefore the watchdog timer will not be
00026 // serviced.
00027 #define MAX_DIGEST_CALCULATIONS_PER_CALL 5
00028
00029 void emAfOtaVerifyStoredDataFinish(EmberAfImageVerifyStatus status);
00030
00031 void emAfOtaClientSignatureVerifyPrintSigners(void);
```

ota-client.h

Go to the documentation of this file.

```
00001
00002
00003 void emAfOtaClientStop(void);
00004 void emAfOtaClientPrintState(void);
00005 void emAfSendImageBlockRequestTest(void);
00006
00007 void emAfSetPageRequest(boolean pageRequest);
00008 boolean emAfUsingPageRequest(void);
00009
00010 #if !defined(EMBER_AF_PLUGIN_OTA_CLIENT_DOWNLOAD_DELAY_MS)
00011   // How often the client will ask for a piece of an upgrade image being
00012   // actively downloaded.  A rate of 0 means the client will pull down the data
00013   // as fast as it can.
00014   #define EMBER_AF_PLUGIN_OTA_CLIENT_DOWNLOAD_DELAY_MS  0L
00015 #endif
00016
00017 // How often the OTA client looks for an OTA server when there is NOT
00018 // one present in the network.  Once it has found one, it queries the
00019 // same one forever (or until it reboots).
00020 #if defined(EMBER_AF_PLUGIN_OTA_CLIENT_SERVER_DISCOVERY_DELAY_MINUTES)
00021   #define EMBER_AF_OTA_SERVER_DISCOVERY_DELAY_MS \
00022     (EMBER_AF_PLUGIN_OTA_CLIENT_SERVER_DISCOVERY_DELAY_MINUTES * MINUTES_IN_MS)
00023 #else
00024   #define EMBER_AF_OTA_SERVER_DISCOVERY_DELAY_MS     (2 * MINUTES_IN_MS)
00025 #endif
00026
00027 // How often the OTA client asks the OTA server if there is a new image
00028 // available
00029 #if defined(EMBER_AF_PLUGIN_OTA_CLIENT_QUERY_DELAY_MINUTES)
00030   #define EMBER_AF_OTA_QUERY_DELAY_MS  \
00031     (EMBER_AF_PLUGIN_OTA_CLIENT_QUERY_DELAY_MINUTES * MINUTES_IN_MS)
00032 #else
00033   #define EMBER_AF_OTA_QUERY_DELAY_MS (5 * MINUTES_IN_MS)
00034 #endif
00035
00036 // The maximum number of sequential errors when downloading that will trigger
00037 // the OTA client to abort the download.
00038 #if !defined(EMBER_AF_PLUGIN_OTA_CLIENT_DOWNLOAD_ERROR_THRESHOLD)
00039   #define EMBER_AF_PLUGIN_OTA_CLIENT_DOWNLOAD_ERROR_THRESHOLD  10
00040 #endif
00041
00042 // The delay between attempts to request to initiate the bootload
00043 // of a successfully downloaded file.
00044 #if defined(EMBER_AF_PLUGIN_OTA_CLIENT_RUN_UPGRADE_REQUEST_DELAY_MINUTES)
00045   #define EMBER_AF_RUN_UPGRADE_REQUEST_DELAY_MS \
00046     (EMBER_AF_PLUGIN_OTA_CLIENT_RUN_UPGRADE_REQUEST_DELAY_MINUTES * MINUTES_IN_MS)
00047 #else
00048   #define EMBER_AF_RUN_UPGRADE_REQUEST_DELAY_MS (10 * MINUTES_IN_MS)
00049 #endif
00050
00051 // The maximum number of sequential errors when asking the OTA Server when to
```

```
// upgrade that will cause the OTA client to apply the upgrade without the
// server telling it to do so.
#if !defined(EMBER_AF_PLUGIN_OTA_CLIENT_UPGRADE_WAIT_THRESHOLD)
  #define EMBER_AF_PLUGIN_OTA_CLIENT_UPGRADE_WAIT_THRESHOLD  10
#endif

#if !defined(EMBER_AF_PLUGIN_OTA_CLIENT_PAGE_REQUEST_SIZE)
  #define EMBER_AF_PLUGIN_OTA_CLIENT_PAGE_REQUEST_SIZE 1024
#endif

// The spacing requested by the client between the image blocks sent by the
// server to the client during a page request.
#if !defined(EMBER_AF_OTA_CLIENT_PAGE_REQUEST_SPACING_MS)
  #define EMBER_AF_OTA_CLIENT_PAGE_REQUEST_SPACING_MS 100L
#endif

// This is the time delay between calls to emAfOtaImageDownloadVerify().
// Verification can take a while (especially in the case of signature
// checking for Smart Energy) so this provides the ability for other
// parts of the system to run.
#if !defined(EMBER_AF_PLUGIN_OTA_CLIENT_VERIFY_DELAY_MS)
  #define EMBER_AF_PLUGIN_OTA_CLIENT_VERIFY_DELAY_MS    10L
#endif

#define NULL_EUI64 { 0, 0, 0, 0, 0, 0, 0, 0 }

// A NULL eui64 is an invalid signer.  It will never match.
#if !defined(EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI0)
  #define EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI0 NULL_EUI64
#endif

#if !defined(EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI1)
  #define EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI1 NULL_EUI64
#endif

#if !defined(EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI2)
  #define EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI2 NULL_EUI64
#endif
```

ota-server-policy.h

Go to the documentation of this file.

```
// ********************************************************************
// * ota-server-policy.h
// *
// * A sample policy file that implements the callbacks for the
// * Zigbee Over-the-air bootload cluster server.
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************

void emAfOtaServerPolicyPrint(void);

void emAfOtaServerSetQueryPolicy(int8u value);
void emAfOtaServerSetBlockRequestPolicy(int8u value);
void emAfOtaServerSetUpgradePolicy(int8u value);

boolean emAfServerPageRequestTickCallback(int16u relativeOffset, int8u blockSize);
void emAfSetPageRequestMissedBlockModulus(int16u modulus);
void emAfOtaServerSetPageRequestPolicy(int8u value);
```

ota-server.h

Go to the documentation of this file.

```
// ********************************************************************
// * ota-server.h
// *
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.          *80*
// ********************************************************************

int8u emAfOtaServerGetBlockSize(void);
int8u emAfOtaImageBlockRequestHandler(EmberNodeId source,
                                      const EmberAfOtaImageId* id,
                                      int32u offset,
                                      int8u maxDataSize);

boolean emAfOtaPageRequestErrorHandler(void);

void emAfOtaPageRequestTick(int8u endpoint);

// Returns the status code to the request.
int8u emAfOtaPageRequestHandler(int8u endpoint,
                                const EmberAfOtaImageId* id,
                                int32u offset,
                                int8u maxDataSize,
                                int16u pageSize,
                                int16u responseSpacing);

boolean emAfOtaServerHandlingPageRequest(void);

// This will eventually be moved into a Plugin specific callbacks file.
void emberAfOtaServerSendUpgradeCommandCallback(EmberNodeId dest,
                                                int8u endpoint,
                                                const EmberAfOtaImageId* id);


#if defined(EMBER_TEST) && !defined(EM_AF_TEST_HARNESS_CODE)
  #define EM_AF_TEST_HARNESS_CODE
#endif

```

ota-static-file-data.h

Go to the documentation of this file.

```
// Auto-generated C header created by ota-static-file.pl
// Input file: app/framework/plugin/ota-storage-simple-ram/ota-static-sample.ota

//  image-builder (C) 2010 by Ember Corporation.
//  Version: 1.0
//  ECC signature support present.
//
//
//  File: app/framework/plugin/ota-storage-simple-ram/ota-static-sample.ota
//  Magic Number:        0x0BEEF11E
//  Header Version:      0x0100
//  Header Length:       56 bytes
//  Field Control:       0x0000
//  Manufacturer ID:     0x1002
//  Image Type:          0x5678
//  Firmware Version:    0x00000005
//  Stack Version:       0x0002
//  Header String:       The latest and greatest upgrade.
//  Total Image Size:    182 bytes
//  Total Tags:          3
//    ID:               0xF000   (Manufacturer Specific)
//    Length:           10 bytes
//
//    ID:               0x0002   (ECDSA Signing Certificate)
//    Length:           48 bytes
//      Subject: (>)000D6F0000198B36
//      Issuer:  (>)5445535453454341 (Certicom TEST CA)
//
//    ID:               0x0001   (ECDSA Signature)
//    Length:           50 bytes
//      Signer:  (>)000D6F0000198B36
//       Data:     01C2C31CB8C40064  EABB3189AD8969EA
//                 C25893183A023BD2  8D5FB2134D3E07B9
//                 2E06A258E478D20C  7ACC
//
//
//  Using Certicom TEST CA issued certificate.
//
//  Message Digest: E26A886E8030458DA084586F2DCB08A2
```

```
//
//  Signature is valid
//
//
//  Found 1 files.


#define STATIC_IMAGE_DATA { \
  0x1e, 0xf1, 0xee, 0x0b, 0x00, 0x01, 0x38, 0x00, \
  0x00, 0x00, 0x02, 0x10, 0x78, 0x56, 0x05, 0x00, \
  0x00, 0x00, 0x02, 0x00, 0x54, 0x68, 0x65, 0x20, \
  0x6c, 0x61, 0x74, 0x65, 0x73, 0x74, 0x20, 0x61, \
  0x6e, 0x64, 0x20, 0x67, 0x72, 0x65, 0x61, 0x74, \
  0x65, 0x73, 0x74, 0x20, 0x75, 0x70, 0x67, 0x72, \
  0x61, 0x64, 0x65, 0x2e, 0xb6, 0x00, 0x00, 0x00, \
  0x00, 0xf0, 0x0a, 0x00, 0x00, 0x00, 0x00, 0x01, \
  0x02, 0x03, 0x04, 0x05, 0x06, 0x07, 0x08, 0x09, \
  0x02, 0x00, 0x30, 0x00, 0x00, 0x00, 0x03, 0x07, \
  0x79, 0x29, 0x47, 0xb3, 0x85, 0x0a, 0x95, 0x85, \
  0xbf, 0x8e, 0x25, 0xc1, 0x9d, 0x8e, 0x86, 0x78, \
  0x43, 0x4f, 0x58, 0x36, 0x00, 0x0d, 0x6f, 0x00, \
  0x00, 0x19, 0x8b, 0x36, 0x54, 0x45, 0x53, 0x54, \
  0x53, 0x45, 0x43, 0x41, 0x01, 0x09, 0x00, 0x00, \
  0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01, 0x00, \
  0x32, 0x00, 0x00, 0x00, 0x36, 0x8b, 0x19, 0x00, \
  0x00, 0x6f, 0x0d, 0x00, 0x01, 0xc2, 0xc3, 0x1c, \
  0xb8, 0xc4, 0x00, 0x64, 0xea, 0xbb, 0x31, 0x89, \
  0xad, 0x89, 0x69, 0xea, 0xc2, 0x58, 0x93, 0x18, \
  0x3a, 0x02, 0x3b, 0xd2, 0x8d, 0x5f, 0xb2, 0x13, \
  0x4d, 0x3e, 0x07, 0xb9, 0x2e, 0x06, 0xa2, 0x58, \
  0xe4, 0x78, 0xd2, 0x0c, 0x7a, 0xcc, \
}

#define STATIC_IMAGE_DATA_SIZE 182L
```

ota-storage-simple-custom.h

Go to the documentation of this file.

```
// ******************************************************************************
// * ota-storage-custom.h
// *
// * This defines the custom storage interface used by ota-storage-simple.c
// * to interact with the actual hardware where the images are stored.
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.            *80*
// ******************************************************************************

boolean emberAfCustomStorageInitCallback(void);
boolean emberAfCustomStorageReadCallback(int32u offset,
                                         int32u length,
                                         int8u* returnData);
boolean emberAfCustomStorageWriteCallback(const int8u* dataToWrite,
                                          int32u offset,
                                          int32u length);
```

ota-storage-simple-driver.h

Go to the documentation of this file.

```
// ******************************************************************************
// * ota-storage-simple-driver.h
// *
// * The Simple Storage Module driver interface.  In other words, primitives
// * for reading / writing and storing data about the OTA file that is stored,
// * or is in the process of being downloaded and stored.
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.            *80*
// ******************************************************************************

boolean emberAfCustomStorageInitCallback(void);

boolean emberAfCustomStorageReadCallback(int32u offset,
                                         int32u length,
                                         int8u* returnData);

boolean emberAfCustomStorageWriteCallback(const int8u* dataToWrite,
                                          int32u offset,
                                          int32u length);
```

ota-storage.h

Go to the documentation of this file.

```
// ******************************************************************************
// * ota-server-storage.h
// *
// * This file defines the interface to a Over-the-air (OTA) storage device.  It
// * can be used by either a server or client.  The server can store 0 or more
// * files that are indexed uniquely by an identifier made up of their Version
// * Number, Manufacturer ID, and Device ID.
// *
// * Copyright 2009 by Ember Corporation. All rights reserved.              *80*
// ******************************************************************************

#define OTA_MINIMUM_HEADER_LENGTH (20 + 32 + 4)
// Optional fields are: security credentials, upgrade dest, and HW versions
#define OTA_MAXIMUM_HEADER_LENGTH (OTA_MINIMUM_HEADER_LENGTH + 1 + 8 + 4)

#define OTA_FILE_MAGIC_NUMBER       0x0BEEF11EL

#define MAGIC_NUMBER_OFFSET    0
#define HEADER_VERSION_OFFSET  4
#define HEADER_LENGTH_OFFSET   6
#define FIELD_CONTROL_OFFSET   8
#define MANUFACTURER_ID_OFFSET 10
#define IMAGE_TYPE_ID_OFFSET   12
#define VERSION_OFFSET         14
#define STACK_VERSION_OFFSET   18
#define HEADER_STRING_OFFSET   20
#define IMAGE_SIZE_OFFSET      52
#define OPTIONAL_FIELDS_OFFSET 56
// The rest are optional fields.

#define HEADER_LENGTH_FIELD_LENGTH 2

#define TAG_OVERHEAD (2 + 4)   // 2 bytes for the tag ID, 4 bytes for the length

#define OTA_HEADER_VERSION 0x0100

#define SECURITY_CREDENTIAL_VERSION_FIELD_PRESENT_MASK 0x0001
#define DEVICE_SPECIFIC_FILE_PRESENT_MASK              0x0002
#define HARDWARE_VERSIONS_PRESENT_MASK                 0x0004

#define headerHasSecurityCredentials(header) \
  ((header)->fieldControl & SECURITY_CREDENTIAL_VERSION_FIELD_PRESENT_MASK)
#define headerHasUpgradeFileDest(header)  \
  ((header)->fieldControl & DEVICE_SPECIFIC_FILE_PRESENT_MASK)
#define headerHasHardwareVersions(header) \
```

```
00046   ((header)->fieldControl & HARDWARE_VERSIONS_PRESENT_MASK)
00047
00048
00049 // This size does NOT include the tag overhead.
00050 #define SIGNATURE_TAG_DATA_SIZE (EUI64_SIZE + EMBER_SIGNATURE_SIZE)
00051
00052 #define INVALID_MANUFACTURER_ID  0xFFFF
00053 #define INVALID_DEVICE_ID        0xFFFF
00054 #define INVALID_FIRMWARE_VERSION 0xFFFFFFFFL
00055 #define INVALID_EUI64 { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 }
00056
00057 #define INVALID_OTA_IMAGE_ID    \
00058   { INVALID_MANUFACTURER_ID,    \
00059     INVALID_DEVICE_ID,          \
00060     INVALID_FIRMWARE_VERSION,   \
00061     INVALID_EUI64,              \
00062   }
00063
00064 //------------------------------------------------------------------------------
00065
00066 // Initialization
00067 // (For the POSIX implementation the device will be a file or directory)
00068 EmberAfOtaStorageStatus emAfOtaStorageSetDevice(const void* device);
00069 void emAfOtaStorageClose(void);
00070
00071 const char* emAfOtaStorageGetFilepath(const EmberAfOtaImageId* id);
00072
00073 EmberAfOtaStorageStatus emAfOtaStorageAddImageFile(const char* filename);
00074
00075
00076 // Creating (two options)
00077 //  - Create a file based on a passed "EmberAfOtaHeader" structure, stored
00078 //      at the filename passed to the function.  This is usually done by a
00079 //      PC tool.
00080 //  - Create a file based on raw data (presumably received over the air)
00081 //      This will be stored in a single static temp file.
00082 EmberAfOtaStorageStatus emAfOtaStorageCreateImage(EmberAfOtaHeader* header,
00083                                                  const char* filename);
00084 EmberAfOtaStorageStatus emAfOtaStorageAppendImageData(const char* filename,
00085                                                      int32u length,
00086                                                      const int8u* data);
00087
00088 //------------------------------------------------------------------------------
00089 // Generic routines that are independent of the actual storage mechanism.
00090
00091 EmberAfOtaStorageStatus emAfOtaStorageGetHeaderLengthAndImageSize(const EmberAfOtaImageId* id,
00092                                                                  int32u *returnHeaderLength,
00093                                                                  int32u *returnImageSize);
00094
00095 EmberAfOtaImageId emAfOtaStorageGetImageIdFromHeader(const EmberAfOtaHeader* header);
00096
00097 // Returns the offset and size of the actual data (does not include
```

```
// tag meta-data) in the specified tag.
EmberAfOtaStorageStatus emAfOtaStorageGetTagOffsetAndSize(const EmberAfOtaImageId* id,
                                                          int16u tag,
                                                          int32u* returnTagOffset,
                                                          int32u* returnTagSize);

EmberAfOtaStorageStatus emAfOtaStorageGetTagDataFromImage(const EmberAfOtaImageId* id,
                                                          int16u tag,
                                                          int8u* returnData,
                                                          int32u* returnDataLength,
                                                          int32u maxReturnDataLength);

// This gets the OTA header as it is formatted in the file, including
// the magic number.
EmberAfOtaStorageStatus emAfOtaStorageGetRawHeaderData(const EmberAfOtaImageId* id,
                                                       int8u* returnData,
                                                       int32u* returnDataLength,
                                                       int32u maxReturnDataLength);

// This retrieves a list of all tags in the file and their lengths.
// It will read at most 'maxTags' and return that array data in tagInfo.
EmberAfOtaStorageStatus emAfOtaStorageReadAllTagInfo(const EmberAfOtaImageId* id,
                                                     EmberAfTagData* tagInfo,
                                                     int16u maxTags,
                                                     int16u* totalTags);

boolean emberAfIsOtaImageIdValid(const EmberAfOtaImageId* idToCompare);

// This should be moved into the plugin callbacks file.
EmberAfOtaStorageStatus emberAfOtaStorageDeleteImageCallback(const EmberAfOtaImageId* id);
int32u emberAfOtaStorageDriverRetrieveLastStoredOffsetCallback(void);
void emberAfOtaStorageDriverDownloadFinishCallback(int32u finalOffset);

//-------------------------------------------------------------------------------
// Internal (for debugging malloc() and free())
int remainingAllocations(void);

```

ota.h

Go to the documentation of this file.

```
// ********************************************************************
// * ota.h
// *
// *
// * Copyright 2009 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************
enum {
  OTA_USE_OFFSET_TIME,
  OTA_USE_UTC_TIME
};
#define OTA_UPGRADE_END_RESPONSE_RUN_NOW  0x00

#define MFG_ID_WILD_CARD       0xFFFF
#define IMAGE_TYPE_WILD_CARD    0xFFFF
#define IMAGE_TYPE_SECURITY     0xFFC0
#define IMAGE_TYPE_CONFIG       0xFFC1
#define IMAGE_TYPE_LOG          0xFFC2
#define FILE_VERSION_WILD_CARD  0xFFFFFFFF

#define ZIGBEE_2006_STACK_VERSION 0x0000
#define ZIGBEE_2007_STACK_VERSION 0x0001
#define ZIGBEE_PRO_STACK_VERSION  0x0002

// These apply to the field control of the Over-the-air messages
#define OTA_HW_VERSION_BIT_MASK   0x01
#define OTA_NODE_EUI64_BIT_MASK   0x01

// OTA Upgrade status as defined by OTA cluster spec.
enum {
  OTA_UPGRADE_STATUS_NORMAL,
  OTA_UPGRADE_STATUS_DOWNLOAD_IN_PROGRESS,
  OTA_UPGRADE_STATUS_DOWNLOAD_COMPLETE,
  OTA_UPGRADE_STATUS_WAIT,
  OTA_UPGRADE_STATUS_COUNTDOWN
};

#define OTA_TAG_UPGRADE_IMAGE             0x0000
#define OTA_TAG_ECDSA_SIGNATURE           0x0001
#define OTA_TAG_ECDSA_SIGNING_CERTIFICATE   0x0002
#define OTA_TAG_RESERVED_START            0x0003
#define OTA_TAG_RESERVED_END              0xEFFF
#define OTA_TAG_MANUFACTURER_SPECIFIC_START 0xFFFF

extern EmberNodeId upgradeServerNodeId;
extern PGM int8u emAfOtaMinMessageLengths[];

```

```
#define EM_AF_OTA_MAX_COMMAND_ID ZCL_QUERY_SPECIFIC_FILE_RESPONSE_COMMAND_ID

// Create a shorter name for printing to make the code more readable.
#define otaPrintln(...) emberAfOtaBootloadClusterPrintln(__VA_ARGS__)
#define otaPrint(...) emberAfOtaBootloadClusterPrint(__VA_ARGS__)

// -------------------------------------------------------
// Common to both client and server
// -------------------------------------------------------
EmberAfOtaImageId emAfOtaCreateEmberAfOtaImageIdStruct(int16u manufacturerId,
                                                       int16u imageType,
                                                       int32u fileVersion);

int8u emAfOtaParseImageIdFromMessage(EmberAfOtaImageId* returnId,
                                     const int8u* buffer,
                                     int8u length);

#if defined(EMBER_AF_PRINT_CORE)
void emAfPrintPercentageSetStartAndEnd(int32u startingOffset, int32u endOffset);
int8u emAfPrintPercentageUpdate(PGM_P prefixString,
                                int8u updateFrequency,
                                int32u currentOffset);
#else
  #define emAfPrintPercentageSetStartAndEnd(x,y)
  #define emAfPrintPercentageUpdate(x,y,z)
#endif
```

partner-link-key-exchange.h

Go to the documentation of this file.

```
// ********************************************************************
// * partner-link-key-exchange.h
// *
// * Support of requesting link keys with another device.
// *
// * Copyright 2008 by Ember Corporation. All rights reserved.             *80*
// ********************************************************************

#define EMBER_AF_PLUGIN_PARTNER_LINK_KEY_EXCHANGE_TIMEOUT_MILLISECONDS \
  ((int32u)EMBER_AF_PLUGIN_PARTNER_LINK_KEY_EXCHANGE_TIMEOUT_SECONDS   \
   * (int32u)MILLISECOND_TICKS_PER_SECOND)
```

plugin-cli.h

Go to the documentation of this file.

```
// *******************************************************************
// * plugin-cli.h
// *
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.             *80*
// *******************************************************************

#include ATTRIBUTE_STORAGE_CONFIGURATION

#ifdef EMBER_AF_GENERATED_PLUGIN_COMMAND_DECLARATIONS
  #define EMBER_AF_PLUGIN_COMMANDS {"plugin", NULL, (PGM_P)emberAfPluginCommands},
  extern EmberCommandEntry emberAfPluginCommands[];
  EMBER_AF_GENERATED_PLUGIN_COMMAND_DECLARATIONS
#else
  #define EMBER_AF_PLUGIN_COMMANDS
#endif
```

price-client-callback.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generatePluginCallbackPrototypes()
//

void emberAfPluginPriceClientPriceStartedCallback(EmberAfPluginPriceClientPrice * price);

void emberAfPluginPriceClientPriceExpiredCallback(EmberAfPluginPriceClientPrice * price);
```

price-client.h

Go to the documentation of this file.

```
// ********************************************************************
// * price-client.h
// *
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.          *80*
// ********************************************************************

#ifndef EMBER_AF_PLUGIN_PRICE_CLIENT_TABLE_SIZE
#define EMBER_AF_PLUGIN_PRICE_CLIENT_TABLE_SIZE 2
#endif //EMBER_AF_PLUGIN_PRICE_CLIENT_TABLE_SIZE

#define ZCL_PRICE_CLUSTER_PRICE_ACKNOWLEDGEMENT_MASK BIT(0)

#define ZCL_PRICE_CLUSTER_START_TIME_NOW                    0x00000000UL
#define ZCL_PRICE_CLUSTER_END_TIME_NEVER                    0xFFFFFFFFUL
#define ZCL_PRICE_CLUSTER_DURATION_UNTIL_CHANGED            0xFFFF
#define ZCL_PRICE_CLUSTER_PRICE_RATIO_NOT_USED              0xFF
#define ZCL_PRICE_CLUSTER_GENERATION_PRICE_NOT_USED         0xFFFFFFFFUL
#define ZCL_PRICE_CLUSTER_GENERATION_PRICE_RATIO_NOT_USED   0xFF
#define ZCL_PRICE_CLUSTER_ALTERNATE_COST_DELIVERED_NOT_USED 0xFFFFFFFFUL
#define ZCL_PRICE_CLUSTER_ALTERNATE_COST_UNIT_NOT_USED      0xFF
#define ZCL_PRICE_CLUSTER_ALTERNATE_COST_TRAILING_DIGIT_NOT_USED 0xFF
#define ZCL_PRICE_CLUSTER_NUMBER_OF_BLOCK_THRESHOLDS_NOT_USED    0xFF
#define ZCL_PRICE_CLUSTER_PRICE_CONTROL_NOT_USED            0x00

void emAfPluginPriceClientPrintInfo(void);
```

print-cluster.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generatePrintCluster()
//


// This is the mapping of IDs to cluster names assuming a format according
// to the "EmberAfClusterName" defined in the ZCL header.
// The names of clusters that are not present, are removed.


#if defined(ZCL_USING_BASIC_CLUSTER_SERVER) || defined(ZCL_USING_BASIC_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_BASIC_CLUSTER {ZCL_BASIC_CLUSTER_ID, "Basic" },
#else
  #define __PRINTCLUSTER_BASIC_CLUSTER
#endif
#if defined(ZCL_USING_POWER_CONFIG_CLUSTER_SERVER) || defined(ZCL_USING_POWER_CONFIG_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_POWER_CONFIG_CLUSTER {ZCL_POWER_CONFIG_CLUSTER_ID, "Power Configuration" },
#else
  #define __PRINTCLUSTER_POWER_CONFIG_CLUSTER
#endif
#if defined(ZCL_USING_DEVICE_TEMP_CLUSTER_SERVER) || defined(ZCL_USING_DEVICE_TEMP_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_DEVICE_TEMP_CLUSTER {ZCL_DEVICE_TEMP_CLUSTER_ID, "Device Temperature Configuration" },
#else
  #define __PRINTCLUSTER_DEVICE_TEMP_CLUSTER
#endif
#if defined(ZCL_USING_IDENTIFY_CLUSTER_SERVER) || defined(ZCL_USING_IDENTIFY_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_IDENTIFY_CLUSTER {ZCL_IDENTIFY_CLUSTER_ID, "Identify" },
#else
  #define __PRINTCLUSTER_IDENTIFY_CLUSTER
#endif
#if defined(ZCL_USING_GROUPS_CLUSTER_SERVER) || defined(ZCL_USING_GROUPS_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_GROUPS_CLUSTER {ZCL_GROUPS_CLUSTER_ID, "Groups" },
#else
  #define __PRINTCLUSTER_GROUPS_CLUSTER
#endif
#if defined(ZCL_USING_SCENES_CLUSTER_SERVER) || defined(ZCL_USING_SCENES_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_SCENES_CLUSTER {ZCL_SCENES_CLUSTER_ID, "Scenes" },
#else
  #define __PRINTCLUSTER_SCENES_CLUSTER
#endif
#if defined(ZCL_USING_ON_OFF_CLUSTER_SERVER) || defined(ZCL_USING_ON_OFF_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_ON_OFF_CLUSTER {ZCL_ON_OFF_CLUSTER_ID, "On/off" },
#else
  #define __PRINTCLUSTER_ON_OFF_CLUSTER
#endif
```

```
#if defined(ZCL_USING_ON_OFF_SWITCH_CONFIG_CLUSTER_SERVER) || defined(ZCL_USING_ON_OFF_SWITCH_CONFIG_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_ON_OFF_SWITCH_CONFIG_CLUSTER {ZCL_ON_OFF_SWITCH_CONFIG_CLUSTER_ID, "On/off Switch Configuration" },
#else
  #define __PRINTCLUSTER_ON_OFF_SWITCH_CONFIG_CLUSTER
#endif
#if defined(ZCL_USING_LEVEL_CONTROL_CLUSTER_SERVER) || defined(ZCL_USING_LEVEL_CONTROL_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_LEVEL_CONTROL_CLUSTER {ZCL_LEVEL_CONTROL_CLUSTER_ID, "Level Control" },
#else
  #define __PRINTCLUSTER_LEVEL_CONTROL_CLUSTER
#endif
#if defined(ZCL_USING_ALARM_CLUSTER_SERVER) || defined(ZCL_USING_ALARM_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_ALARM_CLUSTER {ZCL_ALARM_CLUSTER_ID, "Alarms" },
#else
  #define __PRINTCLUSTER_ALARM_CLUSTER
#endif
#if defined(ZCL_USING_TIME_CLUSTER_SERVER) || defined(ZCL_USING_TIME_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_TIME_CLUSTER {ZCL_TIME_CLUSTER_ID, "Time" },
#else
  #define __PRINTCLUSTER_TIME_CLUSTER
#endif
#if defined(ZCL_USING_RSSI_LOCATION_CLUSTER_SERVER) || defined(ZCL_USING_RSSI_LOCATION_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_RSSI_LOCATION_CLUSTER {ZCL_RSSI_LOCATION_CLUSTER_ID, "RSSI Location" },
#else
  #define __PRINTCLUSTER_RSSI_LOCATION_CLUSTER
#endif
#if defined(ZCL_USING_BINARY_INPUT_BASIC_CLUSTER_SERVER) || defined(ZCL_USING_BINARY_INPUT_BASIC_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_BINARY_INPUT_BASIC_CLUSTER {ZCL_BINARY_INPUT_BASIC_CLUSTER_ID, "Binary Input (Basic)" },
#else
  #define __PRINTCLUSTER_BINARY_INPUT_BASIC_CLUSTER
#endif
#if defined(ZCL_USING_COMMISSIONING_CLUSTER_SERVER) || defined(ZCL_USING_COMMISSIONING_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_COMMISSIONING_CLUSTER {ZCL_COMMISSIONING_CLUSTER_ID, "Commissioning" },
#else
  #define __PRINTCLUSTER_COMMISSIONING_CLUSTER
#endif
#if defined(ZCL_USING_SHADE_CONFIG_CLUSTER_SERVER) || defined(ZCL_USING_SHADE_CONFIG_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_SHADE_CONFIG_CLUSTER {ZCL_SHADE_CONFIG_CLUSTER_ID, "Shade Configuration" },
#else
  #define __PRINTCLUSTER_SHADE_CONFIG_CLUSTER
#endif
#if defined(ZCL_USING_PUMP_CONFIG_CONTROL_CLUSTER_SERVER) || defined(ZCL_USING_PUMP_CONFIG_CONTROL_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_PUMP_CONFIG_CONTROL_CLUSTER {ZCL_PUMP_CONFIG_CONTROL_CLUSTER_ID, "Pump Configuration and Control" },
#else
```

```
#define __PRINTCLUSTER_PUMP_CONFIG_CONTROL_CLUSTER
#endif
#if defined(ZCL_USING_THERMOSTAT_CLUSTER_SERVER) || defined(ZCL_USING_THERMOSTAT_CLUSTER_CLIENT)
#define __PRINTCLUSTER_THERMOSTAT_CLUSTER {ZCL_THERMOSTAT_CLUSTER_ID, "Thermostat" },
#else
#define __PRINTCLUSTER_THERMOSTAT_CLUSTER
#endif
#if defined(ZCL_USING_FAN_CONTROL_CLUSTER_SERVER) || defined(ZCL_USING_FAN_CONTROL_CLUSTER_CLIENT)
#define __PRINTCLUSTER_FAN_CONTROL_CLUSTER {ZCL_FAN_CONTROL_CLUSTER_ID, "Fan Control" },
#else
#define __PRINTCLUSTER_FAN_CONTROL_CLUSTER
#endif
#if defined(ZCL_USING_DEHUMID_CONTROL_CLUSTER_SERVER) || defined(ZCL_USING_DEHUMID_CONTROL_CLUSTER_CLIENT)
#define __PRINTCLUSTER_DEHUMID_CONTROL_CLUSTER {ZCL_DEHUMID_CONTROL_CLUSTER_ID, "Dehumidification Control" },
#else
#define __PRINTCLUSTER_DEHUMID_CONTROL_CLUSTER
#endif
#if defined(ZCL_USING_THERMOSTAT_UI_CONFIG_CLUSTER_SERVER) || defined(ZCL_USING_THERMOSTAT_UI_CONFIG_CLUSTER_CLIENT)
#define __PRINTCLUSTER_THERMOSTAT_UI_CONFIG_CLUSTER {ZCL_THERMOSTAT_UI_CONFIG_CLUSTER_ID, "Thermostat User Interface Configuration" },
#else
#define __PRINTCLUSTER_THERMOSTAT_UI_CONFIG_CLUSTER
#endif
#if defined(ZCL_USING_COLOR_CONTROL_CLUSTER_SERVER) || defined(ZCL_USING_COLOR_CONTROL_CLUSTER_CLIENT)
#define __PRINTCLUSTER_COLOR_CONTROL_CLUSTER {ZCL_COLOR_CONTROL_CLUSTER_ID, "Color Control" },
#else
#define __PRINTCLUSTER_COLOR_CONTROL_CLUSTER
#endif
#if defined(ZCL_USING_BALLAST_CONFIGURATION_CLUSTER_SERVER) || defined(ZCL_USING_BALLAST_CONFIGURATION_CLUSTER_CLIENT)
#define __PRINTCLUSTER_BALLAST_CONFIGURATION_CLUSTER {ZCL_BALLAST_CONFIGURATION_CLUSTER_ID, "Ballast Configuration" },
#else
#define __PRINTCLUSTER_BALLAST_CONFIGURATION_CLUSTER
#endif
#if defined(ZCL_USING_ILLUM_MEASUREMENT_CLUSTER_SERVER) || defined(ZCL_USING_ILLUM_MEASUREMENT_CLUSTER_CLIENT)
#define __PRINTCLUSTER_ILLUM_MEASUREMENT_CLUSTER {ZCL_ILLUM_MEASUREMENT_CLUSTER_ID, "Illuminance Measurement" },
#else
#define __PRINTCLUSTER_ILLUM_MEASUREMENT_CLUSTER
#endif
#if defined(ZCL_USING_ILLUM_LEVEL_SENSING_CLUSTER_SERVER) || defined(ZCL_USING_ILLUM_LEVEL_SENSING_CLUSTER_CLIENT)
#define __PRINTCLUSTER_ILLUM_LEVEL_SENSING_CLUSTER {ZCL_ILLUM_LEVEL_SENSING_CLUSTER_ID, "Illuminance Level Sensing" },
#else
#define __PRINTCLUSTER_ILLUM_LEVEL_SENSING_CLUSTER
```

```
#endif
#if defined(ZCL_USING_TEMP_MEASUREMENT_CLUSTER_SERVER) || defined(ZCL_USING_TEMP_MEASUREMENT_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_TEMP_MEASUREMENT_CLUSTER {ZCL_TEMP_MEASUREMENT_CLUSTER_ID, "Temperature Measurement" },
#else
  #define __PRINTCLUSTER_TEMP_MEASUREMENT_CLUSTER
#endif
#if defined(ZCL_USING_PRESSURE_MEASUREMENT_CLUSTER_SERVER) || defined(ZCL_USING_PRESSURE_MEASUREMENT_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_PRESSURE_MEASUREMENT_CLUSTER {ZCL_PRESSURE_MEASUREMENT_CLUSTER_ID, "Pressure Measurement" },
#else
  #define __PRINTCLUSTER_PRESSURE_MEASUREMENT_CLUSTER
#endif
#if defined(ZCL_USING_FLOW_MEASUREMENT_CLUSTER_SERVER) || defined(ZCL_USING_FLOW_MEASUREMENT_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_FLOW_MEASUREMENT_CLUSTER {ZCL_FLOW_MEASUREMENT_CLUSTER_ID, "Flow Measurement" },
#else
  #define __PRINTCLUSTER_FLOW_MEASUREMENT_CLUSTER
#endif
#if defined(ZCL_USING_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER_SERVER) || defined(ZCL_USING_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER {ZCL_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER_ID, "Relative Humidity Measurement" },
#else
  #define __PRINTCLUSTER_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER
#endif
#if defined(ZCL_USING_OCCUPANCY_SENSING_CLUSTER_SERVER) || defined(ZCL_USING_OCCUPANCY_SENSING_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_OCCUPANCY_SENSING_CLUSTER {ZCL_OCCUPANCY_SENSING_CLUSTER_ID, "Occupancy Sensing" },
#else
  #define __PRINTCLUSTER_OCCUPANCY_SENSING_CLUSTER
#endif
#if defined(ZCL_USING_IAS_ZONE_CLUSTER_SERVER) || defined(ZCL_USING_IAS_ZONE_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_IAS_ZONE_CLUSTER {ZCL_IAS_ZONE_CLUSTER_ID, "IAS Zone" },
#else
  #define __PRINTCLUSTER_IAS_ZONE_CLUSTER
#endif
#if defined(ZCL_USING_IAS_ACE_CLUSTER_SERVER) || defined(ZCL_USING_IAS_ACE_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_IAS_ACE_CLUSTER {ZCL_IAS_ACE_CLUSTER_ID, "IAS ACE" },
#else
  #define __PRINTCLUSTER_IAS_ACE_CLUSTER
#endif
#if defined(ZCL_USING_IAS_WD_CLUSTER_SERVER) || defined(ZCL_USING_IAS_WD_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_IAS_WD_CLUSTER {ZCL_IAS_WD_CLUSTER_ID, "IAS WD" },
#else
  #define __PRINTCLUSTER_IAS_WD_CLUSTER
#endif
#if defined(ZCL_USING_DOOR_LOCK_CLUSTER_SERVER) || defined(ZCL_USING_DOOR_LOCK_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_DOOR_LOCK_CLUSTER {ZCL_DOOR_LOCK_CLUSTER_ID, "Door Lock" },
#else
  #define __PRINTCLUSTER_DOOR_LOCK_CLUSTER
#endif
```

```
#if defined(ZCL_USING_WINDOW_COVERING_CLUSTER_SERVER) || defined(ZCL_USING_WINDOW_COVERING_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_WINDOW_COVERING_CLUSTER {ZCL_WINDOW_COVERING_CLUSTER_ID, "Window Covering" },
#else
  #define __PRINTCLUSTER_WINDOW_COVERING_CLUSTER
#endif
#if defined(ZCL_USING_GENERIC_TUNNEL_CLUSTER_SERVER) || defined(ZCL_USING_GENERIC_TUNNEL_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_GENERIC_TUNNEL_CLUSTER {ZCL_GENERIC_TUNNEL_CLUSTER_ID, "Generic Tunnel" },
#else
  #define __PRINTCLUSTER_GENERIC_TUNNEL_CLUSTER
#endif
#if defined(ZCL_USING_BACNET_PROTOCOL_TUNNEL_CLUSTER_SERVER) || defined(ZCL_USING_BACNET_PROTOCOL_TUNNEL_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_BACNET_PROTOCOL_TUNNEL_CLUSTER {ZCL_BACNET_PROTOCOL_TUNNEL_CLUSTER_ID, "BACnet Protocol Tunnel" },
#else
  #define __PRINTCLUSTER_BACNET_PROTOCOL_TUNNEL_CLUSTER
#endif
#if defined(ZCL_USING_OTA_BOOTLOAD_CLUSTER_SERVER) || defined(ZCL_USING_OTA_BOOTLOAD_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_OTA_BOOTLOAD_CLUSTER {ZCL_OTA_BOOTLOAD_CLUSTER_ID, "Over the Air Bootloading Cluster" },
#else
  #define __PRINTCLUSTER_OTA_BOOTLOAD_CLUSTER
#endif
#if defined(ZCL_USING_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_SERVER) || defined(ZCL_USING_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER {ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID, "Demand Response and Load Control" },
#else
  #define __PRINTCLUSTER_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER
#endif
#if defined(ZCL_USING_SIMPLE_METERING_CLUSTER_SERVER) || defined(ZCL_USING_SIMPLE_METERING_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_SIMPLE_METERING_CLUSTER {ZCL_SIMPLE_METERING_CLUSTER_ID, "Simple Metering" },
#else
  #define __PRINTCLUSTER_SIMPLE_METERING_CLUSTER
#endif
#if defined(ZCL_USING_PRICE_CLUSTER_SERVER) || defined(ZCL_USING_PRICE_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_PRICE_CLUSTER {ZCL_PRICE_CLUSTER_ID, "Price" },
#else
  #define __PRINTCLUSTER_PRICE_CLUSTER
#endif
#if defined(ZCL_USING_MESSAGING_CLUSTER_SERVER) || defined(ZCL_USING_MESSAGING_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_MESSAGING_CLUSTER {ZCL_MESSAGING_CLUSTER_ID, "Messaging" },
#else
  #define __PRINTCLUSTER_MESSAGING_CLUSTER
#endif
#if defined(ZCL_USING_TUNNELING_CLUSTER_SERVER) || defined(ZCL_USING_TUNNELING_CLUSTER_CLIENT)
  #define __PRINTCLUSTER_TUNNELING_CLUSTER {ZCL_TUNNELING_CLUSTER_ID, "Tunneling" },
#else
```

```
00223   #define __PRINTCLUSTER_TUNNELING_CLUSTER
00224 #endif
00225 #if defined(ZCL_USING_PREPAYMENT_CLUSTER_SERVER) ||
defined(ZCL_USING_PREPAYMENT_CLUSTER_CLIENT)
00226   #define __PRINTCLUSTER_PREPAYMENT_CLUSTER {ZCL_PREPAYMENT_CLUSTER_ID, "Prepayment" },
00227 #else
00228   #define __PRINTCLUSTER_PREPAYMENT_CLUSTER
00229 #endif
00230 #if defined(ZCL_USING_KEY_ESTABLISHMENT_CLUSTER_SERVER) ||
defined(ZCL_USING_KEY_ESTABLISHMENT_CLUSTER_CLIENT)
00231   #define __PRINTCLUSTER_KEY_ESTABLISHMENT_CLUSTER
{ZCL_KEY_ESTABLISHMENT_CLUSTER_ID, "Key establishment" },
00232 #else
00233   #define __PRINTCLUSTER_KEY_ESTABLISHMENT_CLUSTER
00234 #endif
00235 #if defined(ZCL_USING_SAMPLE_MFG_SPECIFIC_CLUSTER_SERVER) ||
defined(ZCL_USING_SAMPLE_MFG_SPECIFIC_CLUSTER_CLIENT)
00236   #define
__PRINTCLUSTER_SAMPLE_MFG_SPECIFIC_CLUSTER {ZCL_SAMPLE_MFG_SPECIFIC_CLUSTER_ID, "Sample
Mfg Specific Cluster" },
00237 #else
00238   #define __PRINTCLUSTER_SAMPLE_MFG_SPECIFIC_CLUSTER
00239 #endif
00240 #define CLUSTER_IDS_TO_NAMES \
00241   __PRINTCLUSTER_BASIC_CLUSTER \
00242   __PRINTCLUSTER_POWER_CONFIG_CLUSTER \
00243   __PRINTCLUSTER_DEVICE_TEMP_CLUSTER \
00244   __PRINTCLUSTER_IDENTIFY_CLUSTER \
00245   __PRINTCLUSTER_GROUPS_CLUSTER \
00246   __PRINTCLUSTER_SCENES_CLUSTER \
00247   __PRINTCLUSTER_ON_OFF_CLUSTER \
00248   __PRINTCLUSTER_ON_OFF_SWITCH_CONFIG_CLUSTER \
00249   __PRINTCLUSTER_LEVEL_CONTROL_CLUSTER \
00250   __PRINTCLUSTER_ALARM_CLUSTER \
00251   __PRINTCLUSTER_TIME_CLUSTER \
00252   __PRINTCLUSTER_RSSI_LOCATION_CLUSTER \
00253   __PRINTCLUSTER_BINARY_INPUT_BASIC_CLUSTER \
00254   __PRINTCLUSTER_COMMISSIONING_CLUSTER \
00255   __PRINTCLUSTER_SHADE_CONFIG_CLUSTER \
00256   __PRINTCLUSTER_PUMP_CONFIG_CONTROL_CLUSTER \
00257   __PRINTCLUSTER_THERMOSTAT_CLUSTER \
00258   __PRINTCLUSTER_FAN_CONTROL_CLUSTER \
00259   __PRINTCLUSTER_DEHUMID_CONTROL_CLUSTER \
00260   __PRINTCLUSTER_THERMOSTAT_UI_CONFIG_CLUSTER \
00261   __PRINTCLUSTER_COLOR_CONTROL_CLUSTER \
00262   __PRINTCLUSTER_BALLAST_CONFIGURATION_CLUSTER \
00263   __PRINTCLUSTER_ILLUM_MEASUREMENT_CLUSTER \
00264   __PRINTCLUSTER_ILLUM_LEVEL_SENSING_CLUSTER \
00265   __PRINTCLUSTER_TEMP_MEASUREMENT_CLUSTER \
00266   __PRINTCLUSTER_PRESSURE_MEASUREMENT_CLUSTER \
00267   __PRINTCLUSTER_FLOW_MEASUREMENT_CLUSTER \
00268   __PRINTCLUSTER_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER \
00269   __PRINTCLUSTER_OCCUPANCY_SENSING_CLUSTER \
00270   __PRINTCLUSTER_IAS_ZONE_CLUSTER \
00271   __PRINTCLUSTER_IAS_ACE_CLUSTER \
00272   __PRINTCLUSTER_IAS_WD_CLUSTER \
00273   __PRINTCLUSTER_DOOR_LOCK_CLUSTER \
00274   __PRINTCLUSTER_WINDOW_COVERING_CLUSTER \
00275   __PRINTCLUSTER_GENERIC_TUNNEL_CLUSTER \
```

```
__PRINTCLUSTER_BACNET_PROTOCOL_TUNNEL_CLUSTER \
__PRINTCLUSTER_OTA_BOOTLOAD_CLUSTER \
__PRINTCLUSTER_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER \
__PRINTCLUSTER_SIMPLE_METERING_CLUSTER \
__PRINTCLUSTER_PRICE_CLUSTER \
__PRINTCLUSTER_MESSAGING_CLUSTER \
__PRINTCLUSTER_TUNNELING_CLUSTER \
__PRINTCLUSTER_PREPAYMENT_CLUSTER \
__PRINTCLUSTER_KEY_ESTABLISHMENT_CLUSTER \
__PRINTCLUSTER_SAMPLE_MFG_SPECIFIC_CLUSTER \


#define MAX_CLUSTER_NAME_LENGTH  39
```

psl-commissioning-callback.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generatePluginCallbackPrototypes()
//

void emberAfPluginPslCommissioningTouchLinkCompleteCallback(const EmberPslNetwork * networkInfo,
                                                            int8u deviceInformationRecordCount,
                                                            const EmberPslDeviceInfoRecord * deviceInformationRecordList);

void emberAfPluginPslCommissioningTouchLinkFailedCallback(void);

int8u emberAfPluginPslCommissioningGroupIdentifierCountCallback(int8u endpoint);

boolean emberAfPluginPslCommissioningGroupIdentifierCallback(int8u endpoint,
                                                             int8u index,
                                                             EmberAfPluginPslCommissioningGroupInformationRecord * record);

int8u emberAfPluginPslCommissioningEndpointInformationCountCallback(int8u endpoint);

boolean emberAfPluginPslCommissioningEndpointInformationCallback(int8u endpoint,
                                                                 int8u index,
                                                                 EmberAfPluginPslCommissioningEndpointInformationRecord * record);

void emberAfPluginPslCommissioningIdentifyCallback(int16u duration);
```

psl-commissioning.h

Go to the documentation of this file.

```
// ********************************************************************
// * psl-commissioning.h
// *
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************

EmberStatus emberAfPslSetInitialSecurityState(void);

EmberStatus emberAfPslInitiateTouchLink(void);

void emberAfPslAbortTouchLink(void);

boolean emberAfPslTouchLinkInProgress(void);

void emberAfPslResetToFactoryNew(void);

EmberStatus emberAfPslDeviceInformationRequest(const EmberEUI64 target,
                                               int32u transaction,
                                               int8u startIndex);

EmberStatus emberAfPslIdentifyRequest(const EmberEUI64 target,
                                      int32u transaction,
                                      int16u identifyDuration);

EmberStatus emberAfPslResetToFactoryNewRequest(const EmberEUI64 target,
                                               int32u transaction);

#define EMBER_AF_PLUGIN_PSL_COMMISSIONING_TOUCH_LINK_MILLISECONDS_DELAY \
  ((int32u)EMBER_AF_PLUGIN_PSL_COMMISSIONING_TOUCH_LINK_SECONDS_DELAY   \
   * (int32u)MILLISECOND_TICKS_PER_SECOND)

#if (EMBER_AF_PLUGIN_PSL_COMMISSIONING_SUB_DEVICE_TABLE_SIZE != 0)
  #define EMBER_AF_PLUGIN_PSL_COMMISSIONING_HAS_SUB_DEVICE_TABLE
#endif

#ifdef ZA_DEVICE_IS_SLEEPY
  #define EMBER_AF_PLUGIN_PSL_COMMISSIONING_DEVICE_IS_SLEEPY TRUE
#else
  #define EMBER_AF_PLUGIN_PSL_COMMISSIONING_DEVICE_IS_SLEEPY FALSE
#endif

extern int32u emPslChannelMask;
```

scenes.h

Go to the documentation of this file.

```
// ********************************************************************
// * scenes.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.          *80*
// ********************************************************************

EmberAfStatus emberAfScenesSetSceneCountAttribute(int8u endpoint,
                                                  int8u newCount);
EmberAfStatus emberAfScenesMakeValid(int8u endpoint,
                                     int8u sceneId,
                                     int16u groupId);
EmberAfStatus emberAfScenesMakeInvalid(int8u endpoint);
void emAfPluginScenesServerPrintInfo(void);
```

security-cli.h

Go to the documentation of this file.

```
// File: security-cli.h
//
// Routines to print info about the security keys on the device.
//
// Author(s): Rob Alexander <ralexander@ember.com>
//
// Copyright 2008 by Ember Corporation.  All rights reserved.              *80*

extern EmberCommandEntry cbkeCommands[];
extern EmberCommandEntry changeKeyCommands[];

extern EmberKeyData cliPreconfiguredLinkKey;
extern EmberKeyData cliNetworkKey;

void changeKeyCommand(void);
void keyUpdateCommand(void);

void cbkeStartCommand(void);
void cbkeInterPanCommand(void);
void cbkePartnerCommand(void);

void printKeyInfo(void);

#define KEY_UPDATE_COMMANDS \
  { "key-update", keyUpdateCommand, "" },
```

simple-metering-client-callback.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generatePluginCallbackPrototypes()
//

int16u emberAfPluginSimpleMeteringClientRequestMirrorCallback(EmberEUI64 requestingDeviceIeeeAddress);

int16u emberAfPluginSimpleMeteringClientRemoveMirrorCallback(EmberEUI64 requestingDeviceIeeeAddress);


```

simple-metering-test.h

Go to the documentation of this file.

```
// ******************************************************************
// * simple-metering-test.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.            *80*
// ******************************************************************

void emAfTestMeterTick(int8u endpoint);
void emAfTestMeterInit(int8u endpoint);

// The Test meter's profile interval period timeframe enum value
// is 3 which according to the SE spec is 15 minutes
#define PROFILE_INTERVAL_PERIOD_TIMEFRAME 3
#define PROFILE_INTERVAL_PERIOD_IN_MINUTES 15
#define PROFILE_INTERVAL_PERIOD_IN_SECONDS (PROFILE_INTERVAL_PERIOD_IN_MINUTES * 60)
#define PROFILE_INTERVAL_PERIOD_IN_MILLISECONDS ((PROFILE_INTERVAL_PERIOD_IN_MINUTES
* 60) * 1000)
#define
MAX_PROFILE_INDEX (EMBER_AF_PLUGIN_SIMPLE_METERING_SERVER_TEST_METER_PROFILES - 1)
#define
TOTAL_PROFILE_TIME_SPAN_IN_SECONDS
(EMBER_AF_PLUGIN_SIMPLE_METERING_SERVER_TEST_METER_PROFILES
* (PROFILE_INTERVAL_PERIOD_IN_MINUTES * 60))

void afTestMeterPrint(void);
void afTestMeterSetConsumptionRate(int16u rate);
void afTestMeterSetConsumptionVariance(int16u variance);
void afTestMeterAdjust(int8u endpoint);

// 0 off, 1 if electric, 2 if gas
void afTestMeterMode(int8u endpoint, int8u electric);

void afTestMeterSetError(int8u endpoint, int8u error);
// Sets the random error occurence:
//   data = 0: disable
//   otherwise:
void afTestMeterRandomError(int8u changeIn256);

void afTestMeterEnableProfiles(int8u enable);

boolean emAfTestMeterGetProfiles(int8u intervalChannel,
                                 int32u endTime,
                                 int8u numberOfPeriods);

```

smart-energy-registration.h

Go to the documentation of this file.

```
// ******************************************************************************
// * smart-energy-registration.c
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.            *80*
// ******************************************************************************

#ifndef EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_ERROR_LIMIT
#define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_ERROR_LIMIT 3
#endif //EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_ERROR_LIMIT

#define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_INITIAL (MILLISECOND_TICKS_PER_SECOND << 2)
#define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_RETRY (MILLISECOND_TICKS_PER_SECOND << 5)
#define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_BRIEF MILLISECOND_TICKS_PER_SECOND
#define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_RESUME (MILLISECOND_TICKS_PER_SECOND >> 1)
#define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_TRANSITION (MILLISECOND_TICKS_PER_SECOND >> 2)
#define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_NONE      0

// ESI discovery and binding is only required if the device implements certain
// Smart Energy clusters.  If it doesn't implement these clusters (e.g., it is
// an ESI), the ESI discovery and binding process can be skipped altogether.  If
// discovery is required, it shall be repeated on a period of no more than once
// every three hours and no less than once every 24 hours, according to section
// 5.4.9.2 of 105638r09.
#if defined(EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION)              \
    && (defined(ZCL_USING_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_CLIENT) \
        || defined(ZCL_USING_MESSAGING_CLUSTER_CLIENT)              \
        || defined(ZCL_USING_PRICE_CLUSTER_CLIENT))
  #define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_ESI_DISCOVERY_REQUIRED

  // The plugin option for the discovery period is specified in hours and is
  // converted here to milliseconds.
  #define EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_PERIOD          \
    ((int32u)EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_ESI_DISCOVERY_PERIOD \
     * (int32u)MILLISECOND_TICKS_PER_HOUR)

  void emAfPluginSmartEnergyRegistrationReadAttributesResponseCallback(int8u *buffer,
                                                                       int16u bufLen);
#endif

int8u emAfPluginSmartEnergyRegistrationTrustCenterKeyEstablishmentEndpoint(void);

```

```
extern EmberEventControl emAfPluginSmartEnergyRegistrationEventControl;
void emAfPluginSmartEnergyRegistrationTick(void);

```

stack-handlers.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generateStackHandlerMacros()
//

// Enclosing macro to prevent multiple inclusion
#ifndef __EMBER_AF_STACK_HANDLERS__
#define __EMBER_AF_STACK_HANDLERS__


// Macros required for implemented stack handlers.


#ifdef EZSP_HOST
// This is called by the EZSP when a data poll to the parent is complete. See EZSP documentation for more information.
//
// Application framework itself defines this callback and calls function emberAfEzspPollCompleteHandler(). If you wish to
// add functionality to this handler, while still maintaining app framework functionality, then implement it yourself and make
// sure you call emberAfEzspPollCompleteHandler() function within your implementation of ezspPollCompleteHandler().
  #define EZSP_APPLICATION_HAS_POLL_COMPLETE_HANDLER

#else // ! EZSP_HOST
// This is called by the stack when a data poll to the parent is complete. See stack documentation for more information.
//
// Application framework itself defines this callback and calls function emberAfPollCompleteHandler(). If you wish to
// add functionality to this handler, while still maintaining app framework functionality, then implement it yourself and make
// sure you call emberAfPollCompleteHandler() function within your implementation of emberPollCompleteHandler().
  #define EMBER_APPLICATION_HAS_POLL_COMPLETE_HANDLER

#endif // EZSP_HOST
#endif // __EMBER_AF_STACK_HANDLERS__
```

temperature-measurement.h

Go to the documentation of this file.

```
// ***********************************************************************
// * temperature-measurement.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.            *80*
// ***********************************************************************
```

test-callback.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generatePluginCallbackPrototypes()
//

boolean emberAfPluginTestExampleCustomerFunctionCallback(int8u arg1,
                                                         int8u * arg2,
                                                         boolean arg3);


```

thermostat.h

Go to the documentation of this file.

```
// ******************************************************************
// * thermostat.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.            *80*
// ******************************************************************

EmberAfStatus emAfSetLocalTemperature(int8u endpoint, int16s localTemperature);
```

time-client.h

Go to the documentation of this file.

```
// ********************************************************************
// * time-client.h
// *
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.          *80*
// ********************************************************************

int32u emAfTimeClusterClientGetCurrentTime(void);
void emAfTimeClusterClientSetCurrentTime(int32u utcTime);
```

time-server.h

Go to the documentation of this file.

```
// ********************************************************************
// * time-server.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.          *80*
// ********************************************************************

int32u emAfTimeClusterServerGetCurrentTime(void);
void emAfTimeClusterServerSetCurrentTime(int32u utcTime);
```

trust-center-keepalive.h

Go to the documentation of this file.

```
// ******************************************************************************
// * trust-center-keepalive.c
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.            *80*
// ******************************************************************************

// The duration in milliseconds to wait between two successive keepalives.  The
// period shall be between five and 20 minutes, according to section 5.4.2.2.3.4
// of 105638r09.  The plugin option, specified in minutes, is converted here to
// milliseconds.
#define EMBER_AF_PLUGIN_TRUST_CENTER_KEEPALIVE_DELAY_INTERVAL \
  ((int32u)EMBER_AF_PLUGIN_TRUST_CENTER_KEEPALIVE_INTERVAL    \
   * (int32u)MILLISECOND_TICKS_PER_MINUTE)

// The number of unacknowledged keepalives permitted before declaring that the
// trust center is inaccessible and initiating a search for it.  Section
// 5.4.2.2.3.4 of 105638r09 specifies that this value shall be three.
#define EMBER_AF_PLUGIN_TRUST_CENTER_KEEPALIVE_FAILURE_LIMIT 3

void emAfPluginTrustCenterKeepaliveReadAttributesResponseCallback(int8u *buffer,
                                                                  int16u bufLen);
```

tunneling-client-callback.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generatePluginCallbackPrototypes()
//

void emberAfPluginTunnelingClientTunnelOpenedCallback(int16u tunnelId);

void emberAfPluginTunnelingClientDataReceivedCallback(int16u tunnelId,
                                                      int8u * data,
                                                      int16u dataLen);

void emberAfPluginTunnelingClientDataErrorCallback(int16u tunnelId,
                                                   EmberAfTunnelingTransferDataStatus transferDataStatus);


```

tunneling-client.h

Go to the documentation of this file.

```
// ********************************************************************
// * tunneling-client.h
// *
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.          *80*
// ********************************************************************

#ifndef EMBER_AF_PLUGIN_TUNNELING_CLIENT_TUNNEL_LIMIT
#define EMBER_AF_PLUGIN_TUNNELING_CLIENT_TUNNEL_LIMIT 8
#endif

EmberAfStatus emberAfPluginTunnelingClientTransferData(int16u tunnelId,
                                                       int8u *data,
                                                       int16u dataLen);

EmberAfStatus emberAfPluginTunnelingClientCloseTunnel(int16u tunnelId);
```

tunneling-server-callback.h

Go to the documentation of this file.

```
// Copyright 2007 - 2011 by Ember Corporation. All rights reserved.
//
// This file is generated. Please do not edit manually.
//
// This file is generated from java code via the following methods:
// com.ember.workbench.app_configurator.generator.EmbeddedCodeGenerator.generatePluginCallbackPrototypes()
//

boolean emberAfPluginTunnelingServerIsProtocolSupportedCallback(int8u protocolId,
                                                                int16u manufacturerCode);

void emberAfPluginTunnelingServerTunnelOpenedCallback(int16u tunnelId,
                                                      int8u protocolId,
                                                      int16u manufacturerCode);

void emberAfPluginTunnelingServerDataReceivedCallback(int16u tunnelId,
                                                      int8u * data,
                                                      int16u dataLen);

void emberAfPluginTunnelingServerDataErrorCallback(int16u tunnelId,
                                                   EmberAfTunnelingTransferDataStatus transferDataStatus);

void emberAfPluginTunnelingServerTunnelClosedCallback(int16u tunnelId,
                                                      boolean clientInitiated);
```

tunneling-server.h

Go to the documentation of this file.

```
// ********************************************************************
// * tunneling-server.h
// *
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.          *80*
// ********************************************************************

#define ZCL_TUNNELING_CLUSTER_INVALID_TUNNEL_ID 0xFFFF
#define CLOSE_INITIATED_BY_CLIENT TRUE
#define CLOSE_INITIATED_BY_SERVER FALSE

#ifndef EMBER_AF_PLUGIN_TUNNELING_SERVER_TUNNEL_LIMIT
#define EMBER_AF_PLUGIN_TUNNELING_SERVER_TUNNEL_LIMIT 8
#endif

EmberAfStatus emberAfPluginTunnelingServerTransferData(int16u tunnelId,
                                                       int8u *data,
                                                       int16u dataLen);
```

zcl-cli.h

Go to the documentation of this file.

```
// ********************************************************************
// * zcl-cli.h
// *
// *
// * Copyright 2007 by Ember Corporation. All rights reserved.            *80*
// ********************************************************************

void emAfCliSendCommand(void);
void emAfCliBsendCommand(void);
void emAfCliReadCommand(void);
void emAfCliWriteCommand(void);
void emAfCliTimesyncCommand(void);
void emAfCliRawCommand(void);
void emAfCliClearEventsCommand(void);
void emAfCliAddReportEntryCommand(void);

extern EmberCommandEntry optionCommands[];
extern EmberCommandEntry keysCommands[];
extern EmberCommandEntry interpanCommands[];
extern EmberCommandEntry printCommands[];
extern EmberCommandEntry zclCommands[];
extern EmberCommandEntry certificationCommands[];
```

zdo-cli.h

Go to the documentation of this file.

```
// ****************************************************************************
// * zdo-cli.h
// *
// * CLI commands for sending ZDO messages.
// *
// * Copyright 2010 by Ember Corporation. All rights reserved.            *80*
// ****************************************************************************

extern EmberCommandEntry zdoCommands[];

//#define EMBER_AF_ENABLE_TX_ZDO
#ifdef EMBER_AF_ENABLE_TX_ZDO
  #define ZDO_COMMANDS \
{"zdo", NULL, (PGM_P)zdoCommands},
#else
  #define ZDO_COMMANDS
#endif

```

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- a
- b
- c
- e
- g
- i
- k
- o
- p
- r
- s
- u

- a -

- afLoadControlEventTableInit() : **load-control-event-table.h**
- afReportEventStatusHandler() : **demand-response-load-control.h**
- afScheduleCancelEvent() : **load-control-event-table.h**
- afTestMeterAdjust() : **simple-metering-test.h**
- afTestMeterEnableProfiles() : **simple-metering-test.h**
- afTestMeterMode() : **simple-metering-test.h**
- afTestMeterPrint() : **simple-metering-test.h**
- afTestMeterRandomError() : **simple-metering-test.h**
- afTestMeterSetConsumptionRate() : **simple-metering-test.h**
- afTestMeterSetConsumptionVariance() : **simple-metering-test.h**
- afTestMeterSetError() : **simple-metering-test.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- c
- d
- e
- i
- k
- n
- o
- p
- u
- z

- c -

- cbkeCommands : **security-cli.h**
- certificationCommands : **zcl-cli.h**
- changeKeyCommands : **security-cli.h**
- cliNetworkKey : **security-cli.h**
- cliPreconfiguredLinkKey : **security-cli.h**

- d -

- debugPrintCommands : **core-cli.h**

- e -

- emAfCurrentAppTasks : **af.h**
- emAfCurrentCommand : **af.h**
- emAfDefaultSleepControl : **af.h**
- emAfOtaMinMessageLengths : **ota.h**
- emAfPluginSmartEnergyRegistrationEventControl : **smart-energy-registration.h**
- emberAfHibernateDuration : **af.h**
- emberAfInvalidImageId : **af.h**

- emberAfResponseWaitingFlag : **af.h**
- emPslChannelMask : **psl-commissioning.h**
- endpoints : **af.h**

- i -

- interpanCommands : **zcl-cli.h**

- k -

- keysCommands : **zcl-cli.h**

- n -

- networkCommands : **network-cli.h**

- o -

- optionCommands : **zcl-cli.h**

- p -

- printCommands : **zcl-cli.h**

- u -

- upgradeServerNodeId : **ota.h**

- z -

- ZCL_BITMAP16_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_BITMAP32_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_BITMAP8_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_BOOLEAN_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_CHAR_STRING_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_DATA8_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_ENUM16_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_ENUM8_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_IEEE_ADDRESS_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_INT16S_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_INT16U_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_INT24S_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_INT24U_ATTRIBUTE_TYPE : **attribute-size.h**

- ZCL_INT32S_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_INT32U_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_INT48U_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_INT8S_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_INT8U_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_OCTET_STRING_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_SECURITY_KEY_ATTRIBUTE_TYPE : **attribute-size.h**
- ZCL_UTC_TIME_ATTRIBUTE_TYPE : **attribute-size.h**
- zclCommands : **zcl-cli.h**
- zdoCommands : **zdo-cli.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- b
- c
- d
- e
- i
- n
- r
- s
- w

- b -

- BlockThreshold : **af-structs.h**

- c -

- ChatParticipant : **af-structs.h**
- ChatRoom : **af-structs.h**
- ConfigureReportingRecord : **af-structs.h**
- ConfigureReportingStatusRecord : **af-structs.h**

- d -

- DiscoverAttributesInfoRecord : **af-structs.h**

- e -

- EmAfOtaStorageFileAddedHandler : **ota-storage-linux.h**
- EmAfPageRequestClientStatus : **ota-client-page-request.h**
- EmberAfAllowedInterpanOptions : **af-types.h**
- EmberAfAttributeId : **af-types.h**
- EmberAfAttributeMask : **af-types.h**

- EmberAfAttributeType : **af-types.h**
- EmberAfClusterAttributeChangedCallback : **af-types.h**
- EmberAfClusterId : **af-types.h**
- EmberAfClusterMask : **af-types.h**
- EmberAfDefaultResponseFunction : **af-types.h**
- EmberAfGenericClusterFunction : **af-types.h**
- EmberAfInitFunction : **af-types.h**
- EmberAfInterpanMessageType : **af-types.h**
- EmberAfInterpanOptions : **af-types.h**
- EmberAfManufacturerSpecificClusterAttributeChangedCallback : **af-types.h**
- EmberAfMessageSentFunction : **af-types.h**
- EmberAfOtaDownloadResult : **ota-client-policy.h**
- EmberAfPartnerLinkKeyExchangeCallback : **af-types.h**
- EmberAfProfileId : **af-types.h**
- EmberAfServiceDiscoveryCallback : **af-types.h**
- EmberAfTickFunction : **af-types.h**
- EphemeralData : **af-structs.h**

- i -

- Identity : **af-structs.h**

- n -

- NeighborInfo : **af-structs.h**
- NodeInformation : **af-structs.h**
- Notification : **af-structs.h**

- r -

- ReadAttributeStatusRecord : **af-structs.h**
- ReadReportingConfigurationAttributeRecord : **af-structs.h**
- ReadReportingConfigurationRecord : **af-structs.h**
- ReportAttributeRecord : **af-structs.h**

- s -

- SceneExtensionAttributeInfo : **af-structs.h**
- SceneExtensionFieldSet : **af-structs.h**
- Signature : **af-structs.h**
- Smac : **af-structs.h**

- w -

- WriteAttributeRecord : **af-structs.h**
- WriteAttributeStatusRecord : **af-structs.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- e

- e -

- EmberAfAlternateCostUnit : **enums.h**
- EmberAfAmiCriticalityLevel : **enums.h**
- EmberAfAmiEventStatus : **enums.h**
- EmberAfAmiGetProfileStatus : **enums.h**
- EmberAfAmiIntervalChannel : **enums.h**
- EmberAfAmiIntervalPeriod : **enums.h**
- EmberAfAmiKeyEstablishmentStatus : **enums.h**
- EmberAfAmiRegistrationState : **enums.h**
- EmberAfAmiUnitOfMeasure : **enums.h**
- EmberAfAnonymousDataState : **enums.h**
- EmberAfAttributeWritePermission : **enums.h**
- EmberAfBatterySize : **enums.h**
- EmberAfColorMode : **enums.h**
- EmberAfCommissioningStartupControl : **enums.h**
- EmberAfDehumidifcationLockout : **enums.h**
- EmberAfEventSleepControl : **af-types.h**
- EmberAfFanMode : **enums.h**
- EmberAfFanModeSequence : **enums.h**
- EmberAfHueDirection : **enums.h**
- EmberAfHueMoveMode : **enums.h**
- EmberAfHueStepMode : **enums.h**
- EmberAfIasAceArmMode : **enums.h**
- EmberAfIasAceArmNotification : **enums.h**
- EmberAfIasEnrollResponseCode : **enums.h**
- EmberAfIasZoneState : **enums.h**
- EmberAfIasZoneType : **enums.h**
- EmberAfImageVerifyStatus : **af-types.h**
- EmberAfKeyEstablishmentNotifyMessage : **af-types.h**
- EmberAfKeypadLockout : **enums.h**

- EmberAfLevelStatus : **enums.h**
- EmberAfLocationMethod : **enums.h**
- EmberAfMeasurementLightSensorType : **enums.h**
- EmberAfMessagingControlConfirmation : **enums.h**
- EmberAfMessagingControlImportance : **enums.h**
- EmberAfMessagingControlTransmission : **enums.h**
- EmberAfMeterDeviceType : **enums.h**
- EmberAfMeteringAlarmCode : **enums.h**
- EmberAfMeteringBlockEnumerations : **enums.h**
- EmberAfMeteringSupplyStatus : **enums.h**
- EmberAfMeteringTemperatureUnitOfMeasure : **enums.h**
- EmberAfMoveMode : **enums.h**
- EmberAfOccupancySensorType : **enums.h**
- EmberAfOperatingMode : **enums.h**
- EmberAfOtaStorageStatus : **af-types.h**
- EmberAfPhysicalEnvironment : **enums.h**
- EmberAfPowerSource : **enums.h**
- EmberAfPrepaymentOriginatingDevice : **enums.h**
- EmberAfPrepaymentRecoveryPeriod : **enums.h**
- EmberAfPrepaymentSupplyStatus : **enums.h**
- EmberAfPriceControlAcknowledgement : **enums.h**
- EmberAfPriceTier : **enums.h**
- EmberAfPumpControlMode : **enums.h**
- EmberAfPumpOperationMode : **enums.h**
- EmberAfRegisterTier : **enums.h**
- EmberAfRelativeHumidityDisplay : **enums.h**
- EmberAfRelativeHumidityMode : **enums.h**
- EmberAfReportingDirection : **enums.h**
- EmberAfSaturationMoveMode : **enums.h**
- EmberAfSaturationStepMode : **enums.h**
- EmberAfSensingLightSensorType : **enums.h**
- EmberAfServiceDiscoveryStatus : **af-types.h**
- EmberAfSetpointAdjustMode : **enums.h**
- EmberAfSignatureType : **enums.h**
- EmberAfSquawkLevel : **enums.h**
- EmberAfSquawkMode : **enums.h**
- EmberAfSquawkStobe : **enums.h**
- EmberAfStatus : **enums.h**
- EmberAfStepMode : **enums.h**
- EmberAfSwitchActions : **enums.h**
- EmberAfSwitchType : **enums.h**
- EmberAfTemperatureDisplayMode : **enums.h**
- EmberAfThermostatControlSequence : **enums.h**
- EmberAfThermostatSystemMode : **enums.h**

- EmberAfTunnelingProtocolId : **enums.h**
- EmberAfTunnelingTransferDataStatus : **enums.h**
- EmberAfTunnelingTunnelStatus : **enums.h**
- EmberAfWarningMode : **enums.h**
- EmberAfWarningStobe : **enums.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- b
- e
- g
- i
- k
- l
- m
- n
- o
- p
- r
- t
- z

- b -

- BAD_CERTIFICATE_ISSUER : **af-types.h**
- BAD_KEY_ESTABLISHMENT_SUITE : **af-types.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- _ -

- __PRINTCLUSTER_ALARM_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_BACNET_PROTOCOL_TUNNEL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_BALLAST_CONFIGURATION_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_BASIC_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_BINARY_INPUT_BASIC_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_COLOR_CONTROL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_COMMISSIONING_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_DEHUMID_CONTROL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_DEVICE_TEMP_CLUSTER : **print-cluster.h**

- __PRINTCLUSTER_DOOR_LOCK_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_FAN_CONTROL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_FLOW_MEASUREMENT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_GENERIC_TUNNEL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_GROUPS_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_IAS_ACE_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_IAS_WD_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_IAS_ZONE_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_IDENTIFY_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_ILLUM_LEVEL_SENSING_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_ILLUM_MEASUREMENT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_KEY_ESTABLISHMENT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_LEVEL_CONTROL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_MESSAGING_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_OCCUPANCY_SENSING_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_ON_OFF_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_ON_OFF_SWITCH_CONFIG_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_OTA_BOOTLOAD_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_POWER_CONFIG_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_PREPAYMENT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_PRESSURE_MEASUREMENT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_PRICE_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_PUMP_CONFIG_CONTROL_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_RSSI_LOCATION_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_SAMPLE_MFG_SPECIFIC_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_SCENES_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_SHADE_CONFIG_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_SIMPLE_METERING_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_TEMP_MEASUREMENT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_THERMOSTAT_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_THERMOSTAT_UI_CONFIG_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_TIME_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_TUNNELING_CLUSTER : **print-cluster.h**
- __PRINTCLUSTER_WINDOW_COVERING_CLUSTER : **print-cluster.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- a -

- afLoadControlEventTableInit() : **load-control-event-table.h**
- afReportEventStatusHandler() : **demand-response-load-control.h**
- afScheduleCancelEvent() : **load-control-event-table.h**
- afTestMeterAdjust() : **simple-metering-test.h**
- afTestMeterEnableProfiles() : **simple-metering-test.h**
- afTestMeterMode() : **simple-metering-test.h**

- afTestMeterPrint() : **simple-metering-test.h**
- afTestMeterRandomError() : **simple-metering-test.h**
- afTestMeterSetConsumptionRate() : **simple-metering-test.h**
- afTestMeterSetConsumptionVariance() : **simple-metering-test.h**
- afTestMeterSetError() : **simple-metering-test.h**
- APP_NOTIFY_ERROR_CODE_START : **af-types.h**
- APP_NOTIFY_MESSAGE_TEXT : **af-types.h**
- ATTRIBUTE_MASK_CLIENT : **attribute-storage.h**
- ATTRIBUTE_MASK_EXTERNAL_STORAGE : **attribute-storage.h**
- ATTRIBUTE_MASK_MANUFACTURER_SPECIFIC : **attribute-storage.h**
- ATTRIBUTE_MASK_MIN_MAX : **attribute-storage.h**
- ATTRIBUTE_MASK_SINGLETON : **attribute-storage.h**
- ATTRIBUTE_MASK_TOKENIZE : **attribute-storage.h**
- ATTRIBUTE_MASK_WRITABLE : **attribute-storage.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- b -

- BACNET_PROTOCOL_CLUSTER_MAX_NPDU_SIZE : **bacnet-protocol-tunnel.h**
- bacnetProtocolTxTransferNpdu() : **bacnet-protocol-tunnel.h**
- BAD_CERTIFICATE_ISSUER : **af-types.h**
- BAD_KEY_ESTABLISHMENT_SUITE : **af-types.h**
- BlockThreshold : **af-structs.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- c -

- CANCEL_WITH_RANDOMIZATION : **load-control-event-table.h**
- cbkeCommands : **security-cli.h**
- cbkeInterPanCommand() : **security-cli.h**
- cbkePartnerCommand() : **security-cli.h**
- cbkeStartCommand() : **security-cli.h**
- certificationCommands : **zcl-cli.h**
- changeKeyCommand() : **security-cli.h**

- changeKeyCommands : **security-cli.h**
- ChatParticipant : **af-structs.h**
- ChatRoom : **af-structs.h**
- clearAllTemporaryPublicData() : **key-establishment-storage.h**
- cliNetworkKey : **security-cli.h**
- cliPreconfiguredLinkKey : **security-cli.h**
- CLOSE_INITIATED_BY_CLIENT : **tunneling-server.h**
- CLOSE_INITIATED_BY_SERVER : **tunneling-server.h**
- CLUSTER_IDS_TO_NAMES : **print-cluster.h**
- CLUSTER_MASK_ATTRIBUTE_CHANGED_FUNCTION : **attribute-storage.h**
- CLUSTER_MASK_CLIENT : **attribute-storage.h**
- CLUSTER_MASK_DEFAULT_RESPONSE_FUNCTION : **attribute-storage.h**
- CLUSTER_MASK_INIT_FUNCTION : **attribute-storage.h**
- CLUSTER_MASK_MANUFACTURER_SPECIFIC_ATTRIBUTE_CHANGED_FUNCTION : **attribute-storage.h**
- CLUSTER_MASK_MESSAGE_SENT_FUNCTION : **attribute-storage.h**
- CLUSTER_MASK_SERVER : **attribute-storage.h**
- CONFIGURATION_HEADER : **af.h**
- ConfigureReportingRecord : **af-structs.h**
- ConfigureReportingStatusRecord : **af-structs.h**
- CUSTOM_COMMANDS : **custom-cli.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- d -

- DEBUG_PRINT_TEST : **debug.h**
- debugPrintCommands : **core-cli.h**
- DEVICE_SPECIFIC_FILE_PRESENT_MASK : **ota-storage.h**
- DiscoverAttributesInfoRecord : **af-structs.h**

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- e -

- EM_AF_DISCOVERY_RESPONSE_MASK : **af-types.h**
- EM_AF_KE_EPHEMERAL_SIZE : **key-establishment.h**
- EM_AF_KE_INITIATE_SIZE : **key-establishment.h**
- EM_AF_KE_SMAC_SIZE : **key-establishment.h**
- EM_AF_KE_TERMINATE_SIZE : **key-establishment.h**
- EM_AF_NO_PAGE_REQUEST : **ota-client-page-request.h**
- EM_AF_OTA_MAX_COMMAND_ID : **ota.h**
- EM_AF_PAGE_REQUEST_BLOCK_SIZE : **ota-client-page-request.h**
- EM_AF_PAGE_REQUEST_COMPLETE : **ota-client-page-request.h**
- EM_AF_PAGE_REQUEST_ERROR : **ota-client-page-request.h**
- EM_AF_RETRY_MISSED_PACKETS : **ota-client-page-request.h**
- EM_AF_WAITING_PAGE_REQUEST_REPLIES : **ota-client-page-request.h**
- emAfAbortPageRequest() : **ota-client-page-request.h**
- emAfBootloadSendByte() : **ota-bootload-ncp.h**
- emAfBootloadSendData() : **ota-bootload-ncp.h**
- emAfBootloadWaitChar() : **ota-bootload-ncp.h**
- emAfCancelAllLoadControlEvents() : **load-control-event-table.h**
- emAfCancelLoadControlEvent() : **load-control-event-table.h**
- emAfClearScheduledLoadControlEvents() : **drlc-server.h**
- emAfCliAddReportEntryCommand() : **zcl-cli.h**

- emAfCliBsendCommand() : **zcl-cli.h**
- emAfCliClearEventsCommand() : **zcl-cli.h**
- emAfCliCountersCommand() : **core-cli.h**
- emAfCliEchoCommand() : **core-cli.h**
- emAfCliHelpCommand() : **core-cli.h**
- emAfCliInfoCommand() : **core-cli.h**
- emAfCliRawCommand() : **zcl-cli.h**
- emAfCliReadCommand() : **zcl-cli.h**
- emAfCliResetCommand() : **core-cli.h**
- emAfCliSendCommand() : **zcl-cli.h**
- emAfCliTimesyncCommand() : **zcl-cli.h**
- emAfCliWriteCommand() : **zcl-cli.h**
- emAfCurrentAppTasks : **af.h**
- emAfCurrentCommand : **af.h**
- emAfDefaultSleepControl : **af.h**
- emAfGetCurrentPageRequestStatus() : **ota-client-page-request.h**
- emAfGetFinishedPageRequestOffset() : **ota-client-page-request.h**
- emAfGetPageRequestMissedPacketDelayMs() : **ota-client-page-request.h**
- emAfGetScheduledLoadControlEvent() : **drlc-server.h**
- emAfHandlingPageRequestClient() : **ota-client-page-request.h**
- emAfIasAcePrintInfo() : **ias-ace-server.h**
- emAfIdentifyPrintInfo() : **identify.h**
- emAfInitPageRequestClient() : **ota-client-page-request.h**
- emAfInitXmodemState() : **ota-bootload-xmodem.h**
- emAfInterpanApsCryptMessage() : **interpan.h**
- emAfLoadControlEventOptInOrOut() : **load-control-event-table.h**
- emAfLoadControlEventTablePrint() : **load-control-event-table.h**
- emAfLoadControlEventTableTick() : **load-control-event-table.h**
- emAfNextMissedBlockRequestOffset() : **ota-client-page-request.h**
- emAfNoteReceivedBlockForPageRequestClient() : **ota-client-page-request.h**
- emAfNoteSignatureFailure() : **load-control-event-table.h**
- emAfOtaClientPrintState() : **ota-client.h**
- emAfOtaClientSignatureVerifyPrintSigners() : **ota-client-signature-verify.h**
- emAfOtaClientStop() : **ota-client.h**
- emAfOtaCreateEmberAfOtaImageIdStruct() : **ota.h**
- emAfOtaFindImageIdByIndex() : **ota-cli.h**
- emAfOtaImageBlockRequestHandler() : **ota-server.h**
- emAfOtaImageDelete() : **ota-cli.h**
- emAfOtaImageSignatureVerify() : **ota-client-signature-verify.h**
- emAfOtaMinMessageLengths : **ota.h**
- emAfOtaPageRequestErrorHandler() : **ota-server.h**
- emAfOtaPageRequestHandler() : **ota-server.h**
- emAfOtaPageRequestTick() : **ota-server.h**
- emAfOtaParseImageIdFromMessage() : **ota.h**
- emAfOtaPrintAllImages() : **ota-cli.h**
- emAfOtaServerGetBlockSize() : **ota-server.h**
- emAfOtaServerHandlingPageRequest() : **ota-server.h**
- emAfOtaServerPolicyPrint() : **ota-server-policy.h**
- emAfOtaServerSetBlockRequestPolicy() : **ota-server-policy.h**
- emAfOtaServerSetPageRequestPolicy() : **ota-server-policy.h**
- emAfOtaServerSetQueryPolicy() : **ota-server-policy.h**
- emAfOtaServerSetUpgradePolicy() : **ota-server-policy.h**
- emAfOtaStorageAddImageFile() : **ota-storage.h**
- emAfOtaStorageAppendImageData() : **ota-storage.h**
- emAfOtaStorageClose() : **ota-storage.h**
- emAfOtaStorageCreateImage() : **ota-storage.h**
- EmAfOtaStorageFileAddedHandler : **ota-storage-linux.h**

- emAfOtaStorageGetConfig() : **ota-storage-linux.h**
- emAfOtaStorageGetFilepath() : **ota-storage.h**
- emAfOtaStorageGetHeaderLengthAndImageSize() : **ota-storage.h**
- emAfOtaStorageGetImageIdFromHeader() : **ota-storage.h**
- emAfOtaStorageGetRawHeaderData() : **ota-storage.h**
- emAfOtaStorageGetTagDataFromImage() : **ota-storage.h**
- emAfOtaStorageGetTagOffsetAndSize() : **ota-storage.h**
- emAfOtaStorageReadAllTagInfo() : **ota-storage.h**
- emAfOtaStorageSetConfig() : **ota-storage-linux.h**
- emAfOtaStorageSetDevice() : **ota-storage.h**
- emAfOtaVerifyStoredDataFinish() : **ota-client-signature-verify.h**
- EmAfPageRequestClientStatus : **ota-client-page-request.h**
- emAfPageRequestTimerExpired() : **ota-client-page-request.h**
- emAfPluginDrlcServerPrintInfo() : **drlc-server.h**
- emAfPluginGenericTunnelGetProtocolAddress() : **generic-tunnel.h**
- emAfPluginGenericTunnelPrintInfo() : **generic-tunnel.h**
- emAfPluginGroupsServerPrintInfo() : **groups-server.h**
- emAfPluginInterpanProcessMessage() : **interpan.h**
- emAfPluginInterpanSendRawMessage() : **interpan.h**
- emAfPluginMessagingClientPrintInfo() : **messaging-client.h**
- emAfPluginMessagingServerGetMessageStruct() : **messaging-server.h**
- emAfPluginMessagingServerPrintInfo() : **messaging-server.h**
- emAfPluginPriceClientPrintInfo() : **price-client.h**
- emAfPluginScenesServerPrintInfo() : **scenes.h**
- emAfPluginSmartEnergyRegistrationEventControl : **smart-energy-registration.h**
- emAfPluginSmartEnergyRegistrationTick() : **smart-energy-registration.h**
- emAfPluginSmartEnergyRegistrationTrustCenterKeyEstablishmentEndpoint() : **smart-energy-registration.h**
- emAfPluginTrustCenterKeepaliveReadAttributesResponseCallback() : **trust-center-keepalive.h**
- emAfPostNcpBootload() : **ota-bootload-ncp.h**
- emAfPrintPercentageSetStartAndEnd : **ota.h**
- emAfPrintPercentageUpdate : **ota.h**
- emAfRebootNcpAfterBootload() : **ota-bootload-ncp.h**
- emAfScheduleLoadControlEvent() : **load-control-event-table.h**
- emAfSendImageBlockRequestTest() : **ota-client.h**
- emAfSendXmodemData() : **ota-bootload-xmodem.h**
- emAfServerPageRequestTickCallback() : **ota-server-policy.h**
- emAfSetLocalTemperature() : **thermostat.h**
- emAfSetPageRequest() : **ota-client.h**
- emAfSetPageRequestMissedBlockModulus() : **ota-server-policy.h**
- emAfStartNcpBootloaderCommunications() : **ota-bootload-ncp.h**
- emAfTestMeterGetProfiles() : **simple-metering-test.h**
- emAfTestMeterInit() : **simple-metering-test.h**
- emAfTestMeterTick() : **simple-metering-test.h**
- emAfTimeClusterClientGetCurrentTime() : **time-client.h**
- emAfTimeClusterClientSetCurrentTime() : **time-client.h**
- emAfTimeClusterServerGetCurrentTime() : **time-server.h**
- emAfTimeClusterServerSetCurrentTime() : **time-server.h**
- emAfUsingPageRequest() : **ota-client.h**
- EMBER_AF_BROADCAST_SERVICE_DISCOVERY_COMPLETE : **af-types.h**
- EMBER_AF_BROADCAST_SERVICE_DISCOVERY_RESPONSE_RECEIVED : **af-types.h**
- EMBER_AF_CLIENT_CLUSTER_DISCOVERY : **af.h**
- EMBER_AF_CLIENT_CLUSTER_TICK : **af.h**
- EMBER_AF_CONTINUE_IMAGE_VERIFY : **af.h**
- EMBER_AF_CORE_COMMANDS : **core-cli.h**
- EMBER_AF_DATA_TYPE_ANALOG : **af.h**
- EMBER_AF_DATA_TYPE_DISCRETE : **af.h**
- EMBER_AF_DATA_TYPE_NONE : **af.h**

- EMBER_AF_IMAGE_BAD : **af-types.h**
- EMBER_AF_IMAGE_GOOD : **af-types.h**
- EMBER_AF_IMAGE_VERIFY_IN_PROGRESS : **af-types.h**
- EMBER_AF_INFO_COMMAND : **core-cli.h**
- EMBER_AF_INTER_PAN_BROADCAST : **af-types.h**
- EMBER_AF_INTER_PAN_MULTICAST : **af-types.h**
- EMBER_AF_INTER_PAN_UNICAST : **af-types.h**
- EMBER_AF_INTERPAN_DIRECTION_BOTH : **af-types.h**
- EMBER_AF_INTERPAN_DIRECTION_CLIENT_TO_SERVER : **af-types.h**
- EMBER_AF_INTERPAN_DIRECTION_SERVER_TO_CLIENT : **af-types.h**
- EMBER_AF_INTERPAN_GLOBAL_COMMAND : **af-types.h**
- EMBER_AF_INTERPAN_MANUFACTURER_SPECIFIC : **af-types.h**
- EMBER_AF_INTERPAN_OPTION_APS_ENCRYPT : **af-types.h**
- EMBER_AF_INTERPAN_OPTION_MAC_HAS_LONG_ADDRESS : **af-types.h**
- EMBER_AF_INTERPAN_OPTION_NONE : **af-types.h**
- EMBER_AF_INVALID_HARDWARE_VERSION : **ota-client-policy.h**
- EMBER_AF_LAST_POLL_GOT_DATA : **af.h**
- EMBER_AF_NEW_IMAGE_VERIFICATION : **af.h**
- EMBER_AF_NULL_MANUFACTURER_CODE : **af-types.h**
- EMBER_AF_OK_TO_HIBERNATE : **af-types.h**
- EMBER_AF_OK_TO_NAP : **af-types.h**
- EMBER_AF_OTA_CLIENT_ABORTED : **ota-client-policy.h**
- EMBER_AF_OTA_CLIENT_PAGE_REQUEST_SPACING_MS : **ota-client.h**
- EMBER_AF_OTA_DOWNLOAD_AND_VERIFY_SUCCESS : **ota-client-policy.h**
- EMBER_AF_OTA_DOWNLOAD_TIME_OUT : **ota-client-policy.h**
- EMBER_AF_OTA_MAX_HEADER_STRING_LENGTH : **af-types.h**
- EMBER_AF_OTA_QUERY_DELAY_MS : **ota-client.h**
- EMBER_AF_OTA_SERVER_ABORTED : **ota-client-policy.h**
- EMBER_AF_OTA_SERVER_DISCOVERY_DELAY_MS : **ota-client.h**
- EMBER_AF_OTA_STORAGE_ERROR : **af-types.h**
- EMBER_AF_OTA_STORAGE_PARTIAL_FILE_FOUND : **af-types.h**
- EMBER_AF_OTA_STORAGE_RETURN_DATA_TOO_LONG : **af-types.h**
- EMBER_AF_OTA_STORAGE_SUCCESS : **af-types.h**
- EMBER_AF_OTA_VERIFY_FAILED : **ota-client-policy.h**
- EMBER_AF_PLUGIN_COMMANDS : **plugin-cli.h**
- EMBER_AF_PLUGIN_DRLC_SERVER_SCHEDULED_EVENT_TABLE_SIZE : **drlc-server.h**
- EMBER_AF_PLUGIN_GENERIC_TUNNEL_PROTOCOL_ADDRESS_SIZE : **generic-tunnel.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_DOWNLOAD_DELAY_MS : **ota-client.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_DOWNLOAD_ERROR_THRESHOLD : **ota-client.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_PAGE_REQUEST_SIZE : **ota-client.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_FIRMWARE_VERSION : **ota-client-policy.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_HARDWARE_VERSION : **ota-client-policy.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_IMAGE_TYPE_ID : **ota-client-policy.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI0 : **ota-client.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI1 : **ota-client.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI2 : **ota-client.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_UPGRADE_WAIT_THRESHOLD : **ota-client.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_VERIFY_DELAY_MS : **ota-client.h**
- EMBER_AF_PLUGIN_PARTNER_LINK_KEY_EXCHANGE_TIMEOUT_MILLISECONDS : **partner-link-key-exchange.h**
- EMBER_AF_PLUGIN_PRICE_CLIENT_TABLE_SIZE : **price-client.h**
- EMBER_AF_PLUGIN_PRICE_SERVER_PRICE_TABLE_SIZE : **price-server.h**
- EMBER_AF_PLUGIN_PSL_COMMISSIONING_DEVICE_IS_SLEEPY : **psl-commissioning.h**
- EMBER_AF_PLUGIN_PSL_COMMISSIONING_TOUCH_LINK_MILLISECONDS_DELAY : **psl-commissioning.h**
- EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_BRIEF : **smart-energy-registration.h**
- EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_INITIAL : **smart-energy-registration.h**
- EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_NONE : **smart-energy-registration.h**

- EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_RESUME : **smart-energy-registration.h**
- EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_RETRY : **smart-energy-registration.h**
- EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_TRANSITION : **smart-energy-registration.h**
- EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_ERROR_LIMIT : **smart-energy-registration.h**
- EMBER_AF_PLUGIN_TRUST_CENTER_KEEPALIVE_DELAY_INTERVAL : **trust-center-keepalive.h**
- EMBER_AF_PLUGIN_TRUST_CENTER_KEEPALIVE_FAILURE_LIMIT : **trust-center-keepalive.h**
- EMBER_AF_PLUGIN_TUNNELING_CLIENT_TUNNEL_LIMIT : **tunneling-client.h**
- EMBER_AF_PLUGIN_TUNNELING_SERVER_TUNNEL_LIMIT : **tunneling-server.h**
- EMBER_AF_PRINT_ALARM_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_APP : **debug-printing-test.h**
- EMBER_AF_PRINT_ATTRIBUTES : **debug-printing-test.h**
- EMBER_AF_PRINT_BACNET_PROTOCOL_TUNNEL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_BALLAST_CONFIGURATION_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_BASIC_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_BINARY_INPUT_BASIC_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_BITS : **debug-printing-test.h**
- EMBER_AF_PRINT_COLOR_CONTROL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_COMMISSIONING_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_CORE : **debug-printing-test.h**
- EMBER_AF_PRINT_CUSTOM1 : **debug-printing-test.h**
- EMBER_AF_PRINT_CUSTOM2 : **debug-printing-test.h**
- EMBER_AF_PRINT_CUSTOM3 : **debug-printing-test.h**
- EMBER_AF_PRINT_DEBUG : **debug-printing-test.h**
- EMBER_AF_PRINT_DEHUMID_CONTROL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_DEVICE_TEMP_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_DOOR_LOCK_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_ENABLE : **debug-printing-test.h**
- EMBER_AF_PRINT_FAN_CONTROL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_FLOW_MEASUREMENT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_GENERIC_TUNNEL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_GROUPS_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_IAS_ACE_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_IAS_WD_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_IAS_ZONE_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_IDENTIFY_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_ILLUM_LEVEL_SENSING_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_ILLUM_MEASUREMENT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_KEY_ESTABLISHMENT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_LEVEL_CONTROL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_MESSAGING_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_NAME_NUMBER : **debug-printing-test.h**
- EMBER_AF_PRINT_NAMES : **debug-printing-test.h**
- EMBER_AF_PRINT_OCCUPANCY_SENSING_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_ON_OFF_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_ON_OFF_SWITCH_CONFIG_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_OTA_BOOTLOAD_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_OUTPUT : **debug-printing-test.h**
- EMBER_AF_PRINT_POWER_CONFIG_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_PREPAYMENT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_PRESSURE_MEASUREMENT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_PRICE_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_PUMP_CONFIG_CONTROL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_REGISTRATION : **debug-printing-test.h**
- EMBER_AF_PRINT_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_REPORTING : **debug-printing-test.h**
- EMBER_AF_PRINT_RSSI_LOCATION_CLUSTER : **debug-printing-test.h**

- EMBER_AF_PRINT_SAMPLE_MFG_SPECIFIC_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_SCENES_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_SECURITY : **debug-printing-test.h**
- EMBER_AF_PRINT_SERVICE_DISCOVERY : **debug-printing-test.h**
- EMBER_AF_PRINT_SHADE_CONFIG_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_SIMPLE_METERING_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_TEMP_MEASUREMENT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_THERMOSTAT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_THERMOSTAT_UI_CONFIG_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_TIME_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_TUNNELING_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_WINDOW_COVERING_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_ZDO : **debug-printing-test.h**
- EMBER_AF_RUN_UPGRADE_REQUEST_DELAY_MS : **ota-client.h**
- EMBER_AF_SCENE_TABLE_NULL_INDEX : **af-types.h**
- EMBER_AF_SCENE_TABLE_UNUSED_ENDPOINT_ID : **af-types.h**
- EMBER_AF_SERVER_CLUSTER_DISCOVERY : **af.h**
- EMBER_AF_SERVER_CLUSTER_TICK : **af.h**
- EMBER_AF_SET_HIBERNATE_DURATION : **af.h**
- EMBER_AF_SET_NAP_DURATION : **af.h**
- EMBER_AF_STAY_AWAKE : **af-types.h**
- EMBER_AF_UNICAST_SERVICE_DISCOVERY_COMPLETE_WITH_RESPONSE : **af-types.h**
- EMBER_AF_UNICAST_SERVICE_DISCOVERY_TIMEOUT : **af-types.h**
- EMBER_AF_WAITING_FOR_DATA_ACK : **af.h**
- EMBER_AF_WAITING_FOR_PARTNER_LINK_KEY_EXCHANGE : **af.h**
- EMBER_AF_WAITING_FOR_REGISTRATION : **af.h**
- EMBER_AF_WAITING_FOR_SERVICE_DISCOVERY : **af.h**
- EMBER_AF_WAITING_FOR_ZCL_RESPONSE : **af.h**
- EMBER_AF_WAITING_FOR_ZDO_RESPONSE : **af.h**
- EMBER_AF_WAKE_TIMEOUT : **af.h**
- EMBER_AF_WAKE_TIMEOUT_MASK : **af.h**
- EMBER_AF_ZCL_MANUFACTURER_SPECIFIC_OVERHEAD : **af.h**
- EMBER_AF_ZCL_OVERHEAD : **af.h**
- EMBER_APPLICATION_HAS_POLL_COMPLETE_HANDLER : **stack-handlers.h**
- EMBER_ZCL_ALTERNATE_COST_UNIT_KG_OF_CO2_PER_UNIT_OF_MEASURE : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_1 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_2 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_3 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_4 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_5 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_EMERGENCY : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_GREEN : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_PLANNED_OUTAGE : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_RESERVED : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_SERVICE_DISCONNECT : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED1 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED2 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED3 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED4 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED5 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED6 : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_EVENT_COMPLETED : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_EVENT_COMPLETED_NO_USER_PARTICIPATION_PREVIOUS_OPT_OUT : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_EVENT_PARTIALLY_COMPLETED_DUE_TO_USER_OPT_IN : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_EVENT_PARTIALLY_COMPLETED_WITH_USER_OPT_OUT : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_EVENT_STARTED : **enums.h**

- EMBER_ZCL_AMI_EVENT_STATUS_LOAD_CONTROL_EVENT_COMMAND_REJECTED : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_LOAD_CONTROL_EVENT_COMMAND_RX : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_EVENT_EXPIRED : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_COMMAND : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_COMMAND_INVALID_EFFECTIVE_TIME : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_UNDEFINED_EVENT : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_THE_EVENT_HAS_BEEN_CANCELED : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_THE_EVENT_HAS_BEEN_SUPERSEDED : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_USER_HAS_CHOOSE_TO_OPT_IN : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_USER_HAS_CHOOSE_TO_OPT_OUT : **enums.h**
- EMBER_ZCL_AMI_GET_PROFILE_STATUS_INTERVAL_CHANNEL_NOT_SUPPORTED : **enums.h**
- EMBER_ZCL_AMI_GET_PROFILE_STATUS_INVALID_END_TIME : **enums.h**
- EMBER_ZCL_AMI_GET_PROFILE_STATUS_MORE_PERIODS_REQUESTED_THAN_CAN_BE_RETURNED : **enums.h**
- EMBER_ZCL_AMI_GET_PROFILE_STATUS_NO_INTERVALS_AVAILABLE_FOR_THE_REQUESTED_TIME : **enums.h**
- EMBER_ZCL_AMI_GET_PROFILE_STATUS_SUCCESS : **enums.h**
- EMBER_ZCL_AMI_GET_PROFILE_STATUS_UNDEFINED_INTERVAL_CHANNEL_REQUESTED : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_CHANNEL_CONSUMPTION_DELIVERED : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_CHANNEL_CONSUMPTION_RECEIVED : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_DAILY : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES10 : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES15 : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES2P5 : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES30 : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES5 : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES60 : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES7P5 : **enums.h**
- EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_BAD_KEY_CONFIRM : **enums.h**
- EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_BAD_MESSAGE : **enums.h**
- EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_NO_RESOURCES : **enums.h**
- EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_SUCCESS : **enums.h**
- EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_UNKNOWN_ISSUER : **enums.h**
- EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_UNSUPPORTED_SUITE : **enums.h**
- EMBER_ZCL_AMI_REGISTRATION_STATE_JOINED_NETWORK : **enums.h**
- EMBER_ZCL_AMI_REGISTRATION_STATE_JOINING_NETWORK : **enums.h**
- EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTERATION_NOT_POSSIBLE : **enums.h**
- EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTERED : **enums.h**
- EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTRATION_REJECTED : **enums.h**
- EMBER_ZCL_AMI_REGISTRATION_STATE_SUBMITTED_REGISTRATION_REQUEST : **enums.h**
- EMBER_ZCL_AMI_REGISTRATION_STATE_UNREGISTERED : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_BT_US_OR_BTU_PER_HOUR : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_BT_US_OR_BTU_PER_HOUR_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_CENTUM_CUBIC_FEET_PER_HOUR : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_CENTUM_CUBIC_FEET_PER_HOUR_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_FEET_PER_HOUR : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_FEET_PER_HOUR_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_METER_PER_HOUR : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_METER_PER_HOUR_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_IMPERIAL_GALLONS_PER_HOUR : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_IMPERIAL_GALLONS_PER_HOUR_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_KILO_WATT_HOURS : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_KILO_WATT_HOURS_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_ABSOLUTE : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_ABSOLUTE_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_GAUGE : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_GUAGE_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_LITERS_OR_LITERS_PER_HOUR : **enums.h**

- EMBER_ZCL_AMI_UNIT_OF_MEASURE_LITERS_OR_LITERS_PER_HOUR_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_MCF_OR_MCF_PER_SECOND : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_MCF_OR_MCF_PER_SECOND_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_MJ_OR_MJ_PER_SECOND : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_MJ_OR_MJ_PER_SECOND_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_UNITLESS : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_UNITLESS_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_US_GALLONS_PER_HOUR : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_US_GALLONS_PER_HOUR_BCD : **enums.h**
- EMBER_ZCL_ANONYMOUS_DATA_STATE_NO_SOURCE_FOUND : **enums.h**
- EMBER_ZCL_ANONYMOUS_DATA_STATE_SOURCE_FOUND : **enums.h**
- EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_ALLOW_WRITE_NORMAL : **enums.h**
- EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_ALLOW_WRITE_OF_READ_ONLY : **enums.h**
- EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_DENY_WRITE : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_AA : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_AAA : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_BUILT_IN : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_C : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_D : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_NO_BATTERY : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_OTHER : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_UNKNOWN : **enums.h**
- EMBER_ZCL_COLOR_MODE_COLOR_TEMPERATURE : **enums.h**
- EMBER_ZCL_COLOR_MODE_CURRENT_HUE_AND_CURRENT_SATURATION : **enums.h**
- EMBER_ZCL_COLOR_MODE_CURRENT_X_AND_CURRENT_Y : **enums.h**
- EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_FORM_NETWORK : **enums.h**
- EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_NO_ACTION : **enums.h**
- EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_REJOIN_NETWORK : **enums.h**
- EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_START_FROM_SCRATCH : **enums.h**
- EMBER_ZCL_DEHUMIDIFCATION_LOCKOUT_ALLOWED : **enums.h**
- EMBER_ZCL_DEHUMIDIFCATION_LOCKOUT_NOT_ALLOWED : **enums.h**
- EMBER_ZCL_FAN_MODE_AUTO : **enums.h**
- EMBER_ZCL_FAN_MODE_HIGH : **enums.h**
- EMBER_ZCL_FAN_MODE_LOW : **enums.h**
- EMBER_ZCL_FAN_MODE_MEDIUM : **enums.h**
- EMBER_ZCL_FAN_MODE_OFF : **enums.h**
- EMBER_ZCL_FAN_MODE_ON : **enums.h**
- EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_HIGH : **enums.h**
- EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_HIGH_AUTO : **enums.h**
- EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_MED_HIGH : **enums.h**
- EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_MED_HIGH_AUTO : **enums.h**
- EMBER_ZCL_FAN_MODE_SEQUENCE_ON_AUTO : **enums.h**
- EMBER_ZCL_FAN_MODE_SMART : **enums.h**
- EMBER_ZCL_HUE_DIRECTION_DOWN : **enums.h**
- EMBER_ZCL_HUE_DIRECTION_LONGEST_DISTANCE : **enums.h**
- EMBER_ZCL_HUE_DIRECTION_SHORTEST_DISTANCE : **enums.h**
- EMBER_ZCL_HUE_DIRECTION_UP : **enums.h**
- EMBER_ZCL_HUE_MOVE_MODE_DOWN : **enums.h**
- EMBER_ZCL_HUE_MOVE_MODE_STOP : **enums.h**
- EMBER_ZCL_HUE_MOVE_MODE_UP : **enums.h**
- EMBER_ZCL_HUE_STEP_MODE_DOWN : **enums.h**
- EMBER_ZCL_HUE_STEP_MODE_UP : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_ALL_ZONES : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_DAY_HOME_ZONES_ONLY : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_NIGHT_SLEEP_ZONES_ONLY : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_MODE_DISARM : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ALL_ZONES_ARMED : **enums.h**

- EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ALL_ZONES_DISARMED : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ONLY_DAY_HOME_ZONES_ARMED : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ONLY_NIGHT_SLEEP_ZONES_ARMED : **enums.h**
- EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_NO_ENROLL_PERMIT : **enums.h**
- EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_NOT_SUPPORTED : **enums.h**
- EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_SUCCESS : **enums.h**
- EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_TOO_MANY_ZONES : **enums.h**
- EMBER_ZCL_IAS_ZONE_STATE_ENROLLED : **enums.h**
- EMBER_ZCL_IAS_ZONE_STATE_NOT_ENROLLED : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_CONTACT_SWITCH : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_FIRE_SENSOR : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_GAS_SENSOR : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_KEY_FOB : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_KEYPAD : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_MOTION_SENSOR : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_PERSONAL_EMERGENCY_DEVICE : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_REMOTE_CONTROL : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_STANDARD_CIE : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_STANDARD_WARNING_DEVICE : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_VIBRATION_MOVEMENT_SENSOR : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_WATER_SENSOR : **enums.h**
- EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_FOUR_LOCKOUT : **enums.h**
- EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_ONE_LOCKOUT : **enums.h**
- EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_THREE_LOCKOUT : **enums.h**
- EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_TWO_LOCKOUT : **enums.h**
- EMBER_ZCL_KEYPAD_LOCKOUT_LEVELFIVE_LOCKOUT : **enums.h**
- EMBER_ZCL_KEYPAD_LOCKOUT_NO_LOCKOUT : **enums.h**
- EMBER_ZCL_LEVEL_STATUS_ABOVE_TARGET : **enums.h**
- EMBER_ZCL_LEVEL_STATUS_BELOW_TARGET : **enums.h**
- EMBER_ZCL_LEVEL_STATUS_ON_TARGET : **enums.h**
- EMBER_ZCL_LOCATION_METHOD_LATERATION : **enums.h**
- EMBER_ZCL_LOCATION_METHOD_OUT_OF_BAND : **enums.h**
- EMBER_ZCL_LOCATION_METHOD_RF_FINGERPRINTING : **enums.h**
- EMBER_ZCL_LOCATION_METHOD_SIGNPOSTING : **enums.h**
- EMBER_ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_CMOS : **enums.h**
- EMBER_ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_PHOTODIODE : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_CONFIRMATION_NOT_REQUIRED : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_CONFIRMATION_REQUIRED : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_CRITICAL : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_HIGH : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_LOW : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_MEDIUM : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_ANONYMOUS : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_NORMAL : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_NORMAL_AND_ANONYMOUS : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_RESERVED : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_COOLING_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_ELECTRIC_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_GAS_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_HEAT_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_COOLING_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_GAS_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_HEAT_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_PRESSURE_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_THERMAL_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_WATER_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_PRESSURE_METER : **enums.h**

- EMBER_ZCL_METER_DEVICE_TYPE_THERMAL_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_UNDEFINED_MIRROR_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_WATER_METER : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_BURST_DETECT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_CHECK_METER : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_ELECTRIC_TAMPER_DETECT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_AIR_DETECT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_COMMUNICATION_ERROR : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_MEASUREMENT_FAULT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_REVERSE_FLOW : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L1 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L2 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L3 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L1 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L2 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L3 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_GROUND_FAULT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L1 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L2 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L3 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_INLET_TEMPERATURE_SENSOR_FAULT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_LEAK_DETECT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_LOW_BATTERY : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L1 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L2 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L3 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_OUTLET_TEMPERATURE_SENSOR_FAULT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L1 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L2 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L3 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_PIPE_EMPTY : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_POWER_FAILURE_PIPE_EMPTY_TEMPERATURE_SENSOR : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_POWER_QUALITY_LOW_PRESSURE_BURST_DETECT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_PRESSURE_TOO_HIGH : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_PRESSURE_TOO_LOW : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_REVERSE_FLOW_FLOW_SENSOR : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_SERVICE_DISCONNECT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_TAMPER_DETECT : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK1 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK10 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK11 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK12 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK13 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK14 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK15 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK16 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK2 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK3 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK4 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK5 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK6 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK7 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK8 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK9 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_NO_BLOCKS_IN_USE : **enums.h**
- EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_OFF : **enums.h**
- EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_OFF_ARMED : **enums.h**

- EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_ON : **enums.h**
- EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_CELSIUS : **enums.h**
- EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_CELSIUS_BCD : **enums.h**
- EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_FAHRENHEIT : **enums.h**
- EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_FAHRENHEIT_BCD : **enums.h**
- EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_KELVIN : **enums.h**
- EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_KELVIN_BCD : **enums.h**
- EMBER_ZCL_MOVE_MODE_DOWN : **enums.h**
- EMBER_ZCL_MOVE_MODE_UP : **enums.h**
- EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_PIR : **enums.h**
- EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_PIR_AND_ULTRASONIC : **enums.h**
- EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_ULTRASONIC : **enums.h**
- EMBER_ZCL_OPERATING_MODE_CONFIGURE : **enums.h**
- EMBER_ZCL_OPERATING_MODE_NORMAL : **enums.h**
- EMBER_ZCL_PHYSICAL_ENVIRONMENT_FIRST_PROFILE_SPECIFIED_VALUE : **enums.h**
- EMBER_ZCL_PHYSICAL_ENVIRONMENT_LAST_PROFILE_SPECIFIED_VALUE : **enums.h**
- EMBER_ZCL_PHYSICAL_ENVIRONMENT_UNKNOWN : **enums.h**
- EMBER_ZCL_PHYSICAL_ENVIRONMENT_UNSPECIFIED : **enums.h**
- EMBER_ZCL_POWER_SOURCE_BATTERY : **enums.h**
- EMBER_ZCL_POWER_SOURCE_BATTERY_BACKUP : **enums.h**
- EMBER_ZCL_POWER_SOURCE_DC_SOURCE : **enums.h**
- EMBER_ZCL_POWER_SOURCE_EMERGENCY_MAINS_CONSTANT_POWER : **enums.h**
- EMBER_ZCL_POWER_SOURCE_EMERGENCY_MAINS_TRANSFER_SWITCH : **enums.h**
- EMBER_ZCL_POWER_SOURCE_SINGLE_PHASE_MAINS : **enums.h**
- EMBER_ZCL_POWER_SOURCE_THREE_PHASE_MAINS : **enums.h**
- EMBER_ZCL_POWER_SOURCE_UNKNOWN : **enums.h**
- EMBER_ZCL_PREPAYMENT_ORIGINATING_DEVICE_ENERGY_SERVICE_INTERFACE : **enums.h**
- EMBER_ZCL_PREPAYMENT_ORIGINATING_DEVICE_IN_PREMISES_DISPLAY_DEVICE : **enums.h**
- EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_DAY : **enums.h**
- EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_HOUR : **enums.h**
- EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_MONTH : **enums.h**
- EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_QUARTER : **enums.h**
- EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_WEEK : **enums.h**
- EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_OFF : **enums.h**
- EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_OFF_ARMED : **enums.h**
- EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_ON : **enums.h**
- EMBER_ZCL_PRICE_CONTROL_ACKNOWLEDGEMENT_NOT_REQUIRED : **enums.h**
- EMBER_ZCL_PRICE_CONTROL_ACKNOWLEDGEMENT_REQUIRED : **enums.h**
- EMBER_ZCL_PRICE_TIER_NO_TIER_RELATED : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER10_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER11_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER12_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER13_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER14_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER15_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER1_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER2_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER3_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER4_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER5_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER6_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER7_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER8_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER9_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PUMP_CONTROL_MODE_AUTOMATIC : **enums.h**
- EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_FLOW : **enums.h**
- EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_PRESSURE : **enums.h**

- EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_SPEED : **enums.h**
- EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_TEMPERATURE : **enums.h**
- EMBER_ZCL_PUMP_CONTROL_MODE_PROPORTIONAL_PRESSURE : **enums.h**
- EMBER_ZCL_PUMP_OPERATION_MODE_LOCAL : **enums.h**
- EMBER_ZCL_PUMP_OPERATION_MODE_MAXIMUM : **enums.h**
- EMBER_ZCL_PUMP_OPERATION_MODE_MINIMUM : **enums.h**
- EMBER_ZCL_PUMP_OPERATION_MODE_NORMAL : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER10_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER11_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER12_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER13_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER14_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER15_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER1_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER2_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER3_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER4_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER5_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER6_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER7_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER8_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER9_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_NO_TIER_RELATED : **enums.h**
- EMBER_ZCL_RELATIVE_HUMIDITY_DISPLAY_DISPLAYED : **enums.h**
- EMBER_ZCL_RELATIVE_HUMIDITY_DISPLAY_NOT_DISPLAYED : **enums.h**
- EMBER_ZCL_RELATIVE_HUMIDITY_MODE_MEASURE_LOCALLY : **enums.h**
- EMBER_ZCL_RELATIVE_HUMIDITY_MODE_UPDATED_OVER_THE_NETWORK : **enums.h**
- EMBER_ZCL_REPORTING_DIRECTION_RECEIVED : **enums.h**
- EMBER_ZCL_REPORTING_DIRECTION_REPORTED : **enums.h**
- EMBER_ZCL_SATURATION_MOVE_MODE_DOWN : **enums.h**
- EMBER_ZCL_SATURATION_MOVE_MODE_STOP : **enums.h**
- EMBER_ZCL_SATURATION_MOVE_MODE_UP : **enums.h**
- EMBER_ZCL_SATURATION_STEP_MODE_DOWN : **enums.h**
- EMBER_ZCL_SATURATION_STEP_MODE_UP : **enums.h**
- EMBER_ZCL_SENSING_LIGHT_SENSOR_TYPE_CMOS : **enums.h**
- EMBER_ZCL_SENSING_LIGHT_SENSOR_TYPE_PHOTODIODE : **enums.h**
- EMBER_ZCL_SETPOINT_ADJUST_MODE_COOL_SETPOINT : **enums.h**
- EMBER_ZCL_SETPOINT_ADJUST_MODE_HEAT_AND_COOL_SETPOINTS : **enums.h**
- EMBER_ZCL_SETPOINT_ADJUST_MODE_HEAT_SETPOINT : **enums.h**
- EMBER_ZCL_SIGNATURE_TYPE_ECDSA : **enums.h**
- EMBER_ZCL_SIGNATURE_TYPE_RESERVED : **enums.h**
- EMBER_ZCL_SQUAWK_LEVEL_LOW_LEVEL : **enums.h**
- EMBER_ZCL_SQUAWK_LEVEL_MEDIUM_LEVEL : **enums.h**
- EMBER_ZCL_SQUAWK_LEVEL_VERY_HIGH_LEVEL : **enums.h**
- EMBER_ZCL_SQUAWK_MODE_SYSTEM_IS_ARMED : **enums.h**
- EMBER_ZCL_SQUAWK_MODE_SYSTEM_IS_DISARMED : **enums.h**
- EMBER_ZCL_SQUAWK_STOBE_NO_STROBE : **enums.h**
- EMBER_ZCL_SQUAWK_STOBE_USE_STROBE : **enums.h**
- EMBER_ZCL_STATUS_ABORT : **enums.h**
- EMBER_ZCL_STATUS_CALIBRATION_ERROR : **enums.h**
- EMBER_ZCL_STATUS_DEFINED_OUT_OF_BAND : **enums.h**
- EMBER_ZCL_STATUS_DUPLICATE_EXISTS : **enums.h**
- EMBER_ZCL_STATUS_FAILURE : **enums.h**
- EMBER_ZCL_STATUS_HARDWARE_FAILURE : **enums.h**
- EMBER_ZCL_STATUS_INCONSISTENT_STARTUP_STATE : **enums.h**
- EMBER_ZCL_STATUS_INDICATION_REDIRECTION_TO_AP : **enums.h**
- EMBER_ZCL_STATUS_INSUFFICIENT_SPACE : **enums.h**

- EMBER_ZCL_STATUS_INVALID_DATA_TYPE : **enums.h**
- EMBER_ZCL_STATUS_INVALID_FIELD : **enums.h**
- EMBER_ZCL_STATUS_INVALID_IMAGE : **enums.h**
- EMBER_ZCL_STATUS_INVALID_SELECTOR : **enums.h**
- EMBER_ZCL_STATUS_INVALID_VALUE : **enums.h**
- EMBER_ZCL_STATUS_MALFORMED_COMMAND : **enums.h**
- EMBER_ZCL_STATUS_MULTIPLE_REQUEST_NOT_ALLOWED : **enums.h**
- EMBER_ZCL_STATUS_NO_IMAGE_AVAILABLE : **enums.h**
- EMBER_ZCL_STATUS_NOT_AUTHORIZED : **enums.h**
- EMBER_ZCL_STATUS_NOT_FOUND : **enums.h**
- EMBER_ZCL_STATUS_PREFERENCE_DENIED : **enums.h**
- EMBER_ZCL_STATUS_PREFERENCE_IGNORED : **enums.h**
- EMBER_ZCL_STATUS_READ_ONLY : **enums.h**
- EMBER_ZCL_STATUS_REQUEST_DENIED : **enums.h**
- EMBER_ZCL_STATUS_REQUIRE_MORE_IMAGE : **enums.h**
- EMBER_ZCL_STATUS_RESERVED_FIELD_NOT_ZERO : **enums.h**
- EMBER_ZCL_STATUS_SOFTWARE_FAILURE : **enums.h**
- EMBER_ZCL_STATUS_SUCCESS : **enums.h**
- EMBER_ZCL_STATUS_UNREPORTABLE_ATTRIBUTE : **enums.h**
- EMBER_ZCL_STATUS_UNSUP_CLUSTER_COMMAND : **enums.h**
- EMBER_ZCL_STATUS_UNSUP_GENERAL_COMMAND : **enums.h**
- EMBER_ZCL_STATUS_UNSUP_MANUF_CLUSTER_COMMAND : **enums.h**
- EMBER_ZCL_STATUS_UNSUP_MANUF_GENERAL_COMMAND : **enums.h**
- EMBER_ZCL_STATUS_UNSUPPORTED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_STATUS_WAIT_FOR_DATA : **enums.h**
- EMBER_ZCL_STATUS_WRITE_ONLY : **enums.h**
- EMBER_ZCL_STEP_MODE_DOWN : **enums.h**
- EMBER_ZCL_STEP_MODE_UP : **enums.h**
- EMBER_ZCL_SWITCH_ACTIONS_OFF : **enums.h**
- EMBER_ZCL_SWITCH_ACTIONS_ON : **enums.h**
- EMBER_ZCL_SWITCH_ACTIONS_TOGGLE : **enums.h**
- EMBER_ZCL_SWITCH_TYPE_MOMENTARY : **enums.h**
- EMBER_ZCL_SWITCH_TYPE_TOGGLE : **enums.h**
- EMBER_ZCL_TEMPERATURE_DISPLAY_MODE_CELSIUS : **enums.h**
- EMBER_ZCL_TEMPERATURE_DISPLAY_MODE_FARENHEIT : **enums.h**
- EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_AND_HEATING : **enums.h**
- EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_AND_HEATING_WITH_REHEAT : **enums.h**
- EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_ONLY : **enums.h**
- EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_WITH_REHEAT : **enums.h**
- EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_HEATING_ONLY : **enums.h**
- EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_HEATING_WITH_REHEAT : **enums.h**
- EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_AUTO : **enums.h**
- EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_COOL : **enums.h**
- EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_EMERGENCY_HEATING : **enums.h**
- EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_FAN_ONLY : **enums.h**
- EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_HEAT : **enums.h**
- EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_OFF : **enums.h**
- EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_PRECOOLING : **enums.h**
- EMBER_ZCL_TUNNELING_PROTOCOL_ID_ANSI_C12 : **enums.h**
- EMBER_ZCL_TUNNELING_PROTOCOL_ID_CLIMATE_TALK : **enums.h**
- EMBER_ZCL_TUNNELING_PROTOCOL_ID_DLMS_COSEM : **enums.h**
- EMBER_ZCL_TUNNELING_PROTOCOL_ID_IEC_61107 : **enums.h**
- EMBER_ZCL_TUNNELING_PROTOCOL_ID_M_BUS : **enums.h**
- EMBER_ZCL_TUNNELING_PROTOCOL_ID_SML : **enums.h**
- EMBER_ZCL_TUNNELING_TRANSFER_DATA_STATUS_NO_SUCH_TUNNEL : **enums.h**
- EMBER_ZCL_TUNNELING_TRANSFER_DATA_STATUS_WRONG_DEVICE : **enums.h**
- EMBER_ZCL_TUNNELING_TUNNEL_STATUS_BUSY : **enums.h**

- EMBER_ZCL_TUNNELING_TUNNEL_STATUS_FLOW_CONTROL_NOT_SUPPORTED : **enums.h**
- EMBER_ZCL_TUNNELING_TUNNEL_STATUS_NO_MORE_TUNNEL_IDS : **enums.h**
- EMBER_ZCL_TUNNELING_TUNNEL_STATUS_PROTOCOL_NOT_SUPPORTED : **enums.h**
- EMBER_ZCL_TUNNELING_TUNNEL_STATUS_SUCCESS : **enums.h**
- EMBER_ZCL_WARNING_MODE_BURGLAR : **enums.h**
- EMBER_ZCL_WARNING_MODE_EMERGENCY : **enums.h**
- EMBER_ZCL_WARNING_MODE_FIRE : **enums.h**
- EMBER_ZCL_WARNING_MODE_STOP : **enums.h**
- EMBER_ZCL_WARNING_STOBE_NO_STROBE : **enums.h**
- EMBER_ZCL_WARNING_STOBE_USE_STROBE : **enums.h**
- emberAfAddAddressTableEntry() : **af.h**
- emberAfAddToCurrentAppTasks() : **af.h**
- emberAfAlarmClusterAlarmCallback() : **callback.h**
- emberAfAlarmClusterClientAttributeChangedCallback() : **callback.h**
- emberAfAlarmClusterClientCommandParse() : **call-command-handler.h**
- emberAfAlarmClusterClientDefaultResponseCallback() : **callback.h**
- emberAfAlarmClusterClientInitCallback() : **callback.h**
- emberAfAlarmClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfAlarmClusterClientMessageSentCallback() : **callback.h**
- emberAfAlarmClusterClientTickCallback() : **callback.h**
- emberAfAlarmClusterDebugExec : **debug-printing.h**
- emberAfAlarmClusterFlush : **debug-printing.h**
- emberAfAlarmClusterGetAlarmCallback() : **callback.h**
- emberAfAlarmClusterGetAlarmResponseCallback() : **callback.h**
- emberAfAlarmClusterPrint : **debug-printing.h**
- emberAfAlarmClusterPrintBuffer : **debug-printing.h**
- emberAfAlarmClusterPrintIn : **debug-printing.h**
- emberAfAlarmClusterPrintString : **debug-printing.h**
- emberAfAlarmClusterResetAlarmCallback() : **callback.h**
- emberAfAlarmClusterResetAlarmLogCallback() : **callback.h**
- emberAfAlarmClusterResetAllAlarmsCallback() : **callback.h**
- emberAfAlarmClusterServerAttributeChangedCallback() : **callback.h**
- emberAfAlarmClusterServerCommandParse() : **call-command-handler.h**
- emberAfAlarmClusterServerDefaultResponseCallback() : **callback.h**
- emberAfAlarmClusterServerInitCallback() : **callback.h**
- emberAfAlarmClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfAlarmClusterServerMessageSentCallback() : **callback.h**
- emberAfAlarmClusterServerTickCallback() : **callback.h**
- EmberAfAllowedInterpanOptions : **af-types.h**
- emberAfAllowNetworkWriteAttributeCallback() : **callback.h**
- EmberAfAlternateCostUnit : **enums.h**
- EmberAfAmiCriticalityLevel : **enums.h**
- EmberAfAmiEventStatus : **enums.h**
- EmberAfAmiGetProfileStatus : **enums.h**
- EmberAfAmiIntervalChannel : **enums.h**
- EmberAfAmiIntervalPeriod : **enums.h**
- EmberAfAmiKeyEstablishmentStatus : **enums.h**
- EmberAfAmiRegistrationState : **enums.h**
- EmberAfAmiUnitOfMeasure : **enums.h**
- EmberAfAnonymousDataState : **enums.h**
- emberAfAppDebugExec : **debug-printing.h**
- emberAfAppFlush : **debug-printing.h**
- emberAfAppPrint : **debug-printing.h**
- emberAfAppPrintBuffer : **debug-printing.h**
- emberAfAppPrintIn : **debug-printing.h**
- emberAfAppPrintString : **debug-printing.h**
- EmberAfAttributeId : **af-types.h**

- emberAfAttributeIsClient : **af.h**
- emberAfAttributeIsManufacturerSpecific : **af.h**
- emberAfAttributeIsReadOnly : **af.h**
- emberAfAttributeIsSingleton : **af.h**
- emberAfAttributeIsTokenized : **af.h**
- EmberAfAttributeMask : **af-types.h**
- emberAfAttributesDebugExec : **debug-printing.h**
- emberAfAttributesFlush : **debug-printing.h**
- emberAfAttributeSize : **af.h**
- emberAfAttributesPrint : **debug-printing.h**
- emberAfAttributesPrintBuffer : **debug-printing.h**
- emberAfAttributesPrintln : **debug-printing.h**
- emberAfAttributesPrintString : **debug-printing.h**
- EmberAfAttributeType : **af-types.h**
- EmberAfAttributeWritePermission : **enums.h**
- emberAfBacnetProtocolTunnelClusterClientAttributeChangedCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterClientDefaultResponseCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterClientInitCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterClientMessageSentCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterClientTickCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterDebugExec : **debug-printing.h**
- emberAfBacnetProtocolTunnelClusterFlush : **debug-printing.h**
- emberAfBacnetProtocolTunnelClusterPrint : **debug-printing.h**
- emberAfBacnetProtocolTunnelClusterPrintBuffer : **debug-printing.h**
- emberAfBacnetProtocolTunnelClusterPrintln : **debug-printing.h**
- emberAfBacnetProtocolTunnelClusterPrintString : **debug-printing.h**
- emberAfBacnetProtocolTunnelClusterServerAttributeChangedCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterServerCommandParse() : **call-command-handler.h**
- emberAfBacnetProtocolTunnelClusterServerDefaultResponseCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterServerInitCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterServerMessageSentCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterServerTickCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterTransferNpduCallback() : **callback.h**
- emberAfBallastConfigurationClusterClientAttributeChangedCallback() : **callback.h**
- emberAfBallastConfigurationClusterClientDefaultResponseCallback() : **callback.h**
- emberAfBallastConfigurationClusterClientInitCallback() : **callback.h**
- emberAfBallastConfigurationClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBallastConfigurationClusterClientMessageSentCallback() : **callback.h**
- emberAfBallastConfigurationClusterClientTickCallback() : **callback.h**
- emberAfBallastConfigurationClusterDebugExec : **debug-printing.h**
- emberAfBallastConfigurationClusterFlush : **debug-printing.h**
- emberAfBallastConfigurationClusterPrint : **debug-printing.h**
- emberAfBallastConfigurationClusterPrintBuffer : **debug-printing.h**
- emberAfBallastConfigurationClusterPrintln : **debug-printing.h**
- emberAfBallastConfigurationClusterPrintString : **debug-printing.h**
- emberAfBallastConfigurationClusterServerAttributeChangedCallback() : **callback.h**
- emberAfBallastConfigurationClusterServerDefaultResponseCallback() : **callback.h**
- emberAfBallastConfigurationClusterServerInitCallback() : **callback.h**
- emberAfBallastConfigurationClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBallastConfigurationClusterServerMessageSentCallback() : **callback.h**
- emberAfBallastConfigurationClusterServerTickCallback() : **callback.h**
- emberAfBasicClusterClientAttributeChangedCallback() : **callback.h**
- emberAfBasicClusterClientDefaultResponseCallback() : **callback.h**
- emberAfBasicClusterClientInitCallback() : **callback.h**
- emberAfBasicClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**

- emberAfBasicClusterClientMessageSentCallback() : **callback.h**
- emberAfBasicClusterClientTickCallback() : **callback.h**
- emberAfBasicClusterDebugExec : **debug-printing.h**
- emberAfBasicClusterFlush : **debug-printing.h**
- emberAfBasicClusterPrint : **debug-printing.h**
- emberAfBasicClusterPrintBuffer : **debug-printing.h**
- emberAfBasicClusterPrintln : **debug-printing.h**
- emberAfBasicClusterPrintString : **debug-printing.h**
- emberAfBasicClusterResetToFactoryDefaultsCallback() : **callback.h**
- emberAfBasicClusterServerAttributeChangedCallback() : **callback.h**
- emberAfBasicClusterServerCommandParse() : **call-command-handler.h**
- emberAfBasicClusterServerDefaultResponseCallback() : **callback.h**
- emberAfBasicClusterServerInitCallback() : **callback.h**
- emberAfBasicClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBasicClusterServerMessageSentCallback() : **callback.h**
- emberAfBasicClusterServerTickCallback() : **callback.h**
- EmberAfBatterySize : **enums.h**
- emberAfBinaryInputBasicClusterClientAttributeChangedCallback() : **callback.h**
- emberAfBinaryInputBasicClusterClientDefaultResponseCallback() : **callback.h**
- emberAfBinaryInputBasicClusterClientInitCallback() : **callback.h**
- emberAfBinaryInputBasicClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBinaryInputBasicClusterClientMessageSentCallback() : **callback.h**
- emberAfBinaryInputBasicClusterClientTickCallback() : **callback.h**
- emberAfBinaryInputBasicClusterDebugExec : **debug-printing.h**
- emberAfBinaryInputBasicClusterFlush : **debug-printing.h**
- emberAfBinaryInputBasicClusterPrint : **debug-printing.h**
- emberAfBinaryInputBasicClusterPrintBuffer : **debug-printing.h**
- emberAfBinaryInputBasicClusterPrintln : **debug-printing.h**
- emberAfBinaryInputBasicClusterPrintString : **debug-printing.h**
- emberAfBinaryInputBasicClusterServerAttributeChangedCallback() : **callback.h**
- emberAfBinaryInputBasicClusterServerDefaultResponseCallback() : **callback.h**
- emberAfBinaryInputBasicClusterServerInitCallback() : **callback.h**
- emberAfBinaryInputBasicClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBinaryInputBasicClusterServerMessageSentCallback() : **callback.h**
- emberAfBinaryInputBasicClusterServerTickCallback() : **callback.h**
- EmberAfClusterAttributeChangedCallback : **af-types.h**
- EmberAfClusterId : **af-types.h**
- emberAfClusterInitCallback() : **callback.h**
- emberAfClusterIsManufacturerSpecific : **af.h**
- EmberAfClusterMask : **af-types.h**
- emberAfColorControlClusterClientAttributeChangedCallback() : **callback.h**
- emberAfColorControlClusterClientDefaultResponseCallback() : **callback.h**
- emberAfColorControlClusterClientInitCallback() : **callback.h**
- emberAfColorControlClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfColorControlClusterClientMessageSentCallback() : **callback.h**
- emberAfColorControlClusterClientTickCallback() : **callback.h**
- emberAfColorControlClusterDebugExec : **debug-printing.h**
- emberAfColorControlClusterFlush : **debug-printing.h**
- emberAfColorControlClusterMoveColorCallback() : **callback.h**
- emberAfColorControlClusterMoveHueCallback() : **callback.h**
- emberAfColorControlClusterMoveSaturationCallback() : **callback.h**
- emberAfColorControlClusterMoveToColorCallback() : **callback.h**
- emberAfColorControlClusterMoveToColorTemperatureCallback() : **callback.h**
- emberAfColorControlClusterMoveToHueAndSaturationCallback() : **callback.h**
- emberAfColorControlClusterMoveToHueCallback() : **callback.h**
- emberAfColorControlClusterMoveToSaturationCallback() : **callback.h**
- emberAfColorControlClusterPrint : **debug-printing.h**

- emberAfColorControlClusterPrintBuffer : **debug-printing.h**
- emberAfColorControlClusterPrintIn : **debug-printing.h**
- emberAfColorControlClusterPrintString : **debug-printing.h**
- emberAfColorControlClusterServerAttributeChangedCallback() : **callback.h**
- emberAfColorControlClusterServerCommandParse() : **call-command-handler.h**
- emberAfColorControlClusterServerDefaultResponseCallback() : **callback.h**
- emberAfColorControlClusterServerInitCallback() : **callback.h**
- emberAfColorControlClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfColorControlClusterServerMessageSentCallback() : **callback.h**
- emberAfColorControlClusterServerTickCallback() : **callback.h**
- emberAfColorControlClusterStepColorCallback() : **callback.h**
- emberAfColorControlClusterStepHueCallback() : **callback.h**
- emberAfColorControlClusterStepSaturationCallback() : **callback.h**
- EmberAfColorMode : **enums.h**
- emberAfCommissioningClusterClientAttributeChangedCallback() : **callback.h**
- emberAfCommissioningClusterClientCommandParse() : **call-command-handler.h**
- emberAfCommissioningClusterClientDefaultResponseCallback() : **callback.h**
- emberAfCommissioningClusterClientInitCallback() : **callback.h**
- emberAfCommissioningClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfCommissioningClusterClientMessageSentCallback() : **callback.h**
- emberAfCommissioningClusterClientTickCallback() : **callback.h**
- emberAfCommissioningClusterDebugExec : **debug-printing.h**
- emberAfCommissioningClusterFlush : **debug-printing.h**
- emberAfCommissioningClusterPrint : **debug-printing.h**
- emberAfCommissioningClusterPrintBuffer : **debug-printing.h**
- emberAfCommissioningClusterPrintIn : **debug-printing.h**
- emberAfCommissioningClusterPrintString : **debug-printing.h**
- emberAfCommissioningClusterResetStartupParametersCallback() : **callback.h**
- emberAfCommissioningClusterResetStartupParametersResponseCallback() : **callback.h**
- emberAfCommissioningClusterRestartDeviceCallback() : **callback.h**
- emberAfCommissioningClusterRestartDeviceResponseCallback() : **callback.h**
- emberAfCommissioningClusterRestoreStartupParametersCallback() : **callback.h**
- emberAfCommissioningClusterRestoreStartupParametersResponseCallback() : **callback.h**
- emberAfCommissioningClusterSaveStartupParametersCallback() : **callback.h**
- emberAfCommissioningClusterSaveStartupParametersResponseCallback() : **callback.h**
- emberAfCommissioningClusterServerAttributeChangedCallback() : **callback.h**
- emberAfCommissioningClusterServerCommandParse() : **call-command-handler.h**
- emberAfCommissioningClusterServerDefaultResponseCallback() : **callback.h**
- emberAfCommissioningClusterServerInitCallback() : **callback.h**
- emberAfCommissioningClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfCommissioningClusterServerMessageSentCallback() : **callback.h**
- emberAfCommissioningClusterServerTickCallback() : **callback.h**
- EmberAfCommissioningStartupControl : **enums.h**
- emberAfCompareValues() : **af.h**
- emberAfConfigureReportingResponseCallback() : **callback.h**
- emberAfContainsClient() : **af.h**
- emberAfContainsCluster() : **af.h**
- emberAfContainsServer() : **af.h**
- emberAfContinueJoinableNetworkSearchCallback() : **callback.h**
- emberAfCopyInt16u() : **af.h**
- emberAfCopyInt24u() : **af.h**
- emberAfCopyInt32u() : **af.h**
- emberAfCopyInt8u : **af.h**
- emberAfCopyString() : **af.h**
- emberAfCoreDebugExec : **debug-printing.h**
- emberAfCoreFlush : **debug-printing.h**
- emberAfCorePrint : **debug-printing.h**

- emberAfCorePrintBuffer : **debug-printing.h**
- emberAfCorePrintln : **debug-printing.h**
- emberAfCorePrintString : **debug-printing.h**
- emberAfCurrentAppTasks : **af.h**
- emberAfCurrentCommand : **af.h**
- emberAfCurrentEndpoint : **af.h**
- emberAfCustom1DebugExec : **debug-printing.h**
- emberAfCustom1Flush : **debug-printing.h**
- emberAfCustom1Print : **debug-printing.h**
- emberAfCustom1PrintBuffer : **debug-printing.h**
- emberAfCustom1Println : **debug-printing.h**
- emberAfCustom1PrintString : **debug-printing.h**
- emberAfCustom2DebugExec : **debug-printing.h**
- emberAfCustom2Flush : **debug-printing.h**
- emberAfCustom2Print : **debug-printing.h**
- emberAfCustom2PrintBuffer : **debug-printing.h**
- emberAfCustom2Println : **debug-printing.h**
- emberAfCustom2PrintString : **debug-printing.h**
- emberAfCustom3DebugExec : **debug-printing.h**
- emberAfCustom3Flush : **debug-printing.h**
- emberAfCustom3Print : **debug-printing.h**
- emberAfCustom3PrintBuffer : **debug-printing.h**
- emberAfCustom3Println : **debug-printing.h**
- emberAfCustom3PrintString : **debug-printing.h**
- emberAfCustomCommandLineCallback() : **callback.h**
- emberAfCustomStorageInitCallback() : **ota-storage-simple-custom.h** , **ota-storage-simple-driver.h**
- emberAfCustomStorageReadCallback() : **ota-storage-simple-custom.h** , **ota-storage-simple-driver.h**
- emberAfCustomStorageWriteCallback() : **ota-storage-simple-custom.h** , **ota-storage-simple-driver.h**
- emberAfDeactivateClusterTick() : **af.h**
- emberAfDebugDebugExec : **debug-printing.h**
- emberAfDebugFlush : **debug-printing.h**
- emberAfDebugPrint : **debug-printing.h**
- emberAfDebugPrintBuffer : **debug-printing.h**
- emberAfDebugPrintln : **debug-printing.h**
- emberAfDebugPrintString : **debug-printing.h**
- emberAfDefaultResponseCallback() : **callback.h**
- EmberAfDefaultResponseFunction : **af-types.h**
- emberAfDehumidControlClusterClientAttributeChangedCallback() : **callback.h**
- emberAfDehumidControlClusterClientDefaultResponseCallback() : **callback.h**
- emberAfDehumidControlClusterClientInitCallback() : **callback.h**
- emberAfDehumidControlClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDehumidControlClusterClientMessageSentCallback() : **callback.h**
- emberAfDehumidControlClusterClientTickCallback() : **callback.h**
- emberAfDehumidControlClusterDebugExec : **debug-printing.h**
- emberAfDehumidControlClusterFlush : **debug-printing.h**
- emberAfDehumidControlClusterPrint : **debug-printing.h**
- emberAfDehumidControlClusterPrintBuffer : **debug-printing.h**
- emberAfDehumidControlClusterPrintln : **debug-printing.h**
- emberAfDehumidControlClusterPrintString : **debug-printing.h**
- emberAfDehumidControlClusterServerAttributeChangedCallback() : **callback.h**
- emberAfDehumidControlClusterServerDefaultResponseCallback() : **callback.h**
- emberAfDehumidControlClusterServerInitCallback() : **callback.h**
- emberAfDehumidControlClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDehumidControlClusterServerMessageSentCallback() : **callback.h**
- emberAfDehumidControlClusterServerTickCallback() : **callback.h**
- EmberAfDehumidifcationLockout : **enums.h**
- emberAfDemandResponseLoadControlClusterCancelAllLoadControlEventsCallback() : **callback.h**

- emberAfDemandResponseLoadControlClusterCancelLoadControlEventCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterClientAttributeChangedCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterClientCommandParse() : **call-command-handler.h**
- emberAfDemandResponseLoadControlClusterClientDefaultResponseCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterClientInitCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterClientMessageSentCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterClientTickCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterDebugExec : **debug-printing.h**
- emberAfDemandResponseLoadControlClusterFlush : **debug-printing.h**
- emberAfDemandResponseLoadControlClusterGetScheduledEventsCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterLoadControlEventCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterPrint : **debug-printing.h**
- emberAfDemandResponseLoadControlClusterPrintBuffer : **debug-printing.h**
- emberAfDemandResponseLoadControlClusterPrintln : **debug-printing.h**
- emberAfDemandResponseLoadControlClusterPrintString : **debug-printing.h**
- emberAfDemandResponseLoadControlClusterReportEventStatusCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterServerAttributeChangedCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterServerCommandParse() : **call-command-handler.h**
- emberAfDemandResponseLoadControlClusterServerDefaultResponseCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterServerInitCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterServerMessageSentCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterServerTickCallback() : **callback.h**
- emberAfDeviceIdFromIndex : **af.h**
- emberAfDeviceTempClusterClientAttributeChangedCallback() : **callback.h**
- emberAfDeviceTempClusterClientDefaultResponseCallback() : **callback.h**
- emberAfDeviceTempClusterClientInitCallback() : **callback.h**
- emberAfDeviceTempClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDeviceTempClusterClientMessageSentCallback() : **callback.h**
- emberAfDeviceTempClusterClientTickCallback() : **callback.h**
- emberAfDeviceTempClusterDebugExec : **debug-printing.h**
- emberAfDeviceTempClusterFlush : **debug-printing.h**
- emberAfDeviceTempClusterPrint : **debug-printing.h**
- emberAfDeviceTempClusterPrintBuffer : **debug-printing.h**
- emberAfDeviceTempClusterPrintln : **debug-printing.h**
- emberAfDeviceTempClusterPrintString : **debug-printing.h**
- emberAfDeviceTempClusterServerAttributeChangedCallback() : **callback.h**
- emberAfDeviceTempClusterServerDefaultResponseCallback() : **callback.h**
- emberAfDeviceTempClusterServerInitCallback() : **callback.h**
- emberAfDeviceTempClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDeviceTempClusterServerMessageSentCallback() : **callback.h**
- emberAfDeviceTempClusterServerTickCallback() : **callback.h**
- emberAfDeviceVersionFromIndex : **af.h**
- emberAfDiscoverAttributesResponseCallback() : **callback.h**
- emberAfDoorLockClusterClientAttributeChangedCallback() : **callback.h**
- emberAfDoorLockClusterClientCommandParse() : **call-command-handler.h**
- emberAfDoorLockClusterClientDefaultResponseCallback() : **callback.h**
- emberAfDoorLockClusterClientInitCallback() : **callback.h**
- emberAfDoorLockClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDoorLockClusterClientMessageSentCallback() : **callback.h**
- emberAfDoorLockClusterClientTickCallback() : **callback.h**
- emberAfDoorLockClusterDebugExec : **debug-printing.h**
- emberAfDoorLockClusterFlush : **debug-printing.h**
- emberAfDoorLockClusterLockDoorCallback() : **callback.h**
- emberAfDoorLockClusterLockDoorResponseCallback() : **callback.h**
- emberAfDoorLockClusterPrint : **debug-printing.h**

- emberAfDoorLockClusterPrintBuffer : **debug-printing.h**
- emberAfDoorLockClusterPrintln : **debug-printing.h**
- emberAfDoorLockClusterPrintString : **debug-printing.h**
- emberAfDoorLockClusterServerAttributeChangedCallback() : **callback.h**
- emberAfDoorLockClusterServerCommandParse() : **call-command-handler.h**
- emberAfDoorLockClusterServerDefaultResponseCallback() : **callback.h**
- emberAfDoorLockClusterServerInitCallback() : **callback.h**
- emberAfDoorLockClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDoorLockClusterServerMessageSentCallback() : **callback.h**
- emberAfDoorLockClusterServerTickCallback() : **callback.h**
- emberAfDoorLockClusterUnlockDoorCallback() : **callback.h**
- emberAfDoorLockClusterUnlockDoorResponseCallback() : **callback.h**
- emberAfEndpointCount() : **af.h**
- emberAfEndpointFromIndex : **af.h**
- emberAfEventAction() : **load-control-event-table.h**
- emberAfEventControlSetDelay() : **af.h**
- EmberAfEventSleepControl : **af-types.h**
- emberAfExternalAttributeReadCallback() : **callback.h**
- emberAfExternalAttributeWriteCallback() : **callback.h**
- emberAfEzspPollCompleteHandler() : **af.h**
- emberAfFanControlClusterClientAttributeChangedCallback() : **callback.h**
- emberAfFanControlClusterClientDefaultResponseCallback() : **callback.h**
- emberAfFanControlClusterClientInitCallback() : **callback.h**
- emberAfFanControlClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfFanControlClusterClientMessageSentCallback() : **callback.h**
- emberAfFanControlClusterClientTickCallback() : **callback.h**
- emberAfFanControlClusterDebugExec : **debug-printing.h**
- emberAfFanControlClusterFlush : **debug-printing.h**
- emberAfFanControlClusterPrint : **debug-printing.h**
- emberAfFanControlClusterPrintBuffer : **debug-printing.h**
- emberAfFanControlClusterPrintln : **debug-printing.h**
- emberAfFanControlClusterPrintString : **debug-printing.h**
- emberAfFanControlClusterServerAttributeChangedCallback() : **callback.h**
- emberAfFanControlClusterServerDefaultResponseCallback() : **callback.h**
- emberAfFanControlClusterServerInitCallback() : **callback.h**
- emberAfFanControlClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfFanControlClusterServerMessageSentCallback() : **callback.h**
- emberAfFanControlClusterServerTickCallback() : **callback.h**
- EmberAfFanMode : **enums.h**
- EmberAfFanModeSequence : **enums.h**
- emberAfFillBuffer() : **client-api.h**
- emberAfFillCommandAlarmClusterAlarm : **client-command-macro.h**
- emberAfFillCommandAlarmClusterGetAlarm : **client-command-macro.h**
- emberAfFillCommandAlarmClusterGetAlarmResponse : **client-command-macro.h**
- emberAfFillCommandAlarmClusterResetAlarm : **client-command-macro.h**
- emberAfFillCommandAlarmClusterResetAlarmLog : **client-command-macro.h**
- emberAfFillCommandAlarmClusterResetAllAlarms : **client-command-macro.h**
- emberAfFillCommandBacnetProtocolTunnelClusterTransferNpdu : **client-command-macro.h**
- emberAfFillCommandBasicClusterResetToFactoryDefaults : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveColor : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveHue : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveSaturation : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveToColor : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveToColorTemperature : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveToHue : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveToHueAndSaturation : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveToSaturation : **client-command-macro.h**

- emberAfFillCommandColorControlClusterStepColor : **client-command-macro.h**
- emberAfFillCommandColorControlClusterStepHue : **client-command-macro.h**
- emberAfFillCommandColorControlClusterStepSaturation : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterResetStartupParameters : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterResetStartupParametersResponse : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterRestartDevice : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterRestartDeviceResponse : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterRestoreStartupParameters : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterRestoreStartupParametersResponse : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterSaveStartupParameters : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterSaveStartupParametersResponse : **client-command-macro.h**
- emberAfFillCommandDemandResponseLoadControlClusterCancelAllLoadControlEvents : **client-command-macro.h**
- emberAfFillCommandDemandResponseLoadControlClusterCancelLoadControlEvent : **client-command-macro.h**
- emberAfFillCommandDemandResponseLoadControlClusterGetScheduledEvents : **client-command-macro.h**
- emberAfFillCommandDemandResponseLoadControlClusterLoadControlEvent : **client-command-macro.h**
- emberAfFillCommandDemandResponseLoadControlClusterReportEventStatus : **client-command-macro.h**
- emberAfFillCommandDoorLockClusterLockDoor : **client-command-macro.h**
- emberAfFillCommandDoorLockClusterLockDoorResponse : **client-command-macro.h**
- emberAfFillCommandDoorLockClusterUnlockDoor : **client-command-macro.h**
- emberAfFillCommandDoorLockClusterUnlockDoorResponse : **client-command-macro.h**
- emberAfFillCommandGenericTunnelClusterAdvertiseProtocolAddress : **client-command-macro.h**
- emberAfFillCommandGenericTunnelClusterMatchProtocolAddress : **client-command-macro.h**
- emberAfFillCommandGenericTunnelClusterMatchProtocolAddressResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerConfigureReporting : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerConfigureReportingResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerDefaultResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerDiscoverAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerDiscoverAttributesResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerReadAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerReadAttributesResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerReadReportingConfiguration : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerReadReportingConfigurationResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerReportAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerWriteAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerWriteAttributesNoResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerWriteAttributesResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerWriteAttributesUndivided : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientConfigureReporting : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientConfigureReportingResponse : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientDefaultResponse : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientDiscoverAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientDiscoverAttributesResponse : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientReadAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientReadAttributesResponse : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientReadReportingConfiguration : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientReadReportingConfigurationResponse : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientReportAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientWriteAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientWriteAttributesNoResponse : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientWriteAttributesResponse : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientWriteAttributesUndivided : **client-command-macro.h**
- emberAfFillCommandGroupsClusterAddGroup : **client-command-macro.h**
- emberAfFillCommandGroupsClusterAddGroupIfIdentifying : **client-command-macro.h**
- emberAfFillCommandGroupsClusterAddGroupResponse : **client-command-macro.h**
- emberAfFillCommandGroupsClusterGetGroupMembership : **client-command-macro.h**
- emberAfFillCommandGroupsClusterGetGroupMembershipResponse : **client-command-macro.h**

- emberAfFillCommandGroupsClusterRemoveAllGroups : **client-command-macro.h**
- emberAfFillCommandGroupsClusterRemoveGroup : **client-command-macro.h**
- emberAfFillCommandGroupsClusterRemoveGroupResponse : **client-command-macro.h**
- emberAfFillCommandGroupsClusterViewGroup : **client-command-macro.h**
- emberAfFillCommandGroupsClusterViewGroupResponse : **client-command-macro.h**
- emberAfFillCommandIasAceClusterArm : **client-command-macro.h**
- emberAfFillCommandIasAceClusterArmResponse : **client-command-macro.h**
- emberAfFillCommandIasAceClusterBypass : **client-command-macro.h**
- emberAfFillCommandIasAceClusterEmergency : **client-command-macro.h**
- emberAfFillCommandIasAceClusterFire : **client-command-macro.h**
- emberAfFillCommandIasAceClusterGetZoneIdMap : **client-command-macro.h**
- emberAfFillCommandIasAceClusterGetZoneIdMapResponse : **client-command-macro.h**
- emberAfFillCommandIasAceClusterGetZoneInformation : **client-command-macro.h**
- emberAfFillCommandIasAceClusterGetZoneInformationResponse : **client-command-macro.h**
- emberAfFillCommandIasAceClusterPanic : **client-command-macro.h**
- emberAfFillCommandIasWdClusterSquawk : **client-command-macro.h**
- emberAfFillCommandIasWdClusterStartWarning : **client-command-macro.h**
- emberAfFillCommandIasZoneClusterZoneEnrollRequest : **client-command-macro.h**
- emberAfFillCommandIasZoneClusterZoneEnrollResponse : **client-command-macro.h**
- emberAfFillCommandIasZoneClusterZoneStatusChangeNotification : **client-command-macro.h**
- emberAfFillCommandIdentifyClusterIdentify : **client-command-macro.h**
- emberAfFillCommandIdentifyClusterIdentifyQuery : **client-command-macro.h**
- emberAfFillCommandIdentifyClusterIdentifyQueryResponse : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterClientToServerTerminateKeyEstablishment : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterConfirmKeyDataRequest : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterConfirmKeyDataResponse : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterEphemeralDataRequest : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterEphemeralDataResponse : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterInitiateKeyEstablishmentRequest : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterInitiateKeyEstablishmentResponse : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterServerToClientTerminateKeyEstablishment : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterMove : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterMoveToLevel : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterMoveToLevelWithOnOff : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterMoveWithOnOff : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterStep : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterStepWithOnOff : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterStop : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterStopWithOnOff : **client-command-macro.h**
- emberAfFillCommandMessagingClusterCancelMessage : **client-command-macro.h**
- emberAfFillCommandMessagingClusterDisplayMessage : **client-command-macro.h**
- emberAfFillCommandMessagingClusterGetLastMessage : **client-command-macro.h**
- emberAfFillCommandMessagingClusterMessageConfirmation : **client-command-macro.h**
- emberAfFillCommandOnOffClusterOff : **client-command-macro.h**
- emberAfFillCommandOnOffClusterOn : **client-command-macro.h**
- emberAfFillCommandOnOffClusterSampleMfgSpecificOffWithTransition : **client-command-macro.h**
- emberAfFillCommandOnOffClusterSampleMfgSpecificOnWithTransition : **client-command-macro.h**
- emberAfFillCommandOnOffClusterSampleMfgSpecificToggleWithTransition : **client-command-macro.h**
- emberAfFillCommandOnOffClusterToggle : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterImageBlockRequest : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterImageBlockResponse : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterImageNotify : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterImagePageRequest : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterQueryNextImageRequest : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterQueryNextImageResponse : **client-command-macro.h**

- emberAfFillCommandOtaBootloadClusterQuerySpecificFileRequest : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterQuerySpecificFileResponse : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterUpgradeEndRequest : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterUpgradeEndResponse : **client-command-macro.h**
- emberAfFillCommandPrepaymentClusterChangeSupply : **client-command-macro.h**
- emberAfFillCommandPrepaymentClusterSelectAvailableEmergencyCredit : **client-command-macro.h**
- emberAfFillCommandPrepaymentClusterSupplyStatusResponse : **client-command-macro.h**
- emberAfFillCommandPriceClusterGetBlockPeriods : **client-command-macro.h**
- emberAfFillCommandPriceClusterGetCurrentPrice : **client-command-macro.h**
- emberAfFillCommandPriceClusterGetScheduledPrices : **client-command-macro.h**
- emberAfFillCommandPriceClusterPriceAcknowledgement : **client-command-macro.h**
- emberAfFillCommandPriceClusterPublishBlockPeriod : **client-command-macro.h**
- emberAfFillCommandPriceClusterPublishPrice : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterAnchorNodeAnnounce : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterCompactLocationDataNotification : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterDeviceConfigurationResponse : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterGetDeviceConfiguration : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterGetLocationData : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterLocationDataNotification : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterLocationDataResponse : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterReportRssiMeasurements : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterRequestOwnLocation : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterRssiPing : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterRssiRequest : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterRssiResponse : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterSendPings : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterSetAbsoluteLocation : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterSetDeviceConfiguration : **client-command-macro.h**
- emberAfFillCommandSampleMfgSpecificClusterCommandOne : **client-command-macro.h**
- emberAfFillCommandScenesClusterAddScene : **client-command-macro.h**
- emberAfFillCommandScenesClusterAddSceneResponse : **client-command-macro.h**
- emberAfFillCommandScenesClusterGetSceneMembership : **client-command-macro.h**
- emberAfFillCommandScenesClusterGetSceneMembershipResponse : **client-command-macro.h**
- emberAfFillCommandScenesClusterRecallScene : **client-command-macro.h**
- emberAfFillCommandScenesClusterRemoveAllScenes : **client-command-macro.h**
- emberAfFillCommandScenesClusterRemoveAllScenesResponse : **client-command-macro.h**
- emberAfFillCommandScenesClusterRemoveScene : **client-command-macro.h**
- emberAfFillCommandScenesClusterRemoveSceneResponse : **client-command-macro.h**
- emberAfFillCommandScenesClusterStoreScene : **client-command-macro.h**
- emberAfFillCommandScenesClusterStoreSceneResponse : **client-command-macro.h**
- emberAfFillCommandScenesClusterViewScene : **client-command-macro.h**
- emberAfFillCommandScenesClusterViewSceneResponse : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterGetProfile : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterGetProfileResponse : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterMirrorRemoved : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterRemoveMirror : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterRequestFastPollMode : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterRequestFastPollModeResponse : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterRequestMirror : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterRequestMirrorResponse : **client-command-macro.h**
- emberAfFillCommandThermostatClusterSetpointRaiseLower : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterAckTransferDataClientToServer : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterAckTransferDataServerToClient : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterCloseTunnel : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterReadyDataClientToServer : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterReadyDataServerToClient : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterRequestTunnel : **client-command-macro.h**

- emberAfFillCommandTunnelingClusterRequestTunnelResponse : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterTransferDataClientToServer : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterTransferDataErrorClientToServer : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterTransferDataErrorServerToClient : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterTransferDataServerToClient : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringDownClose : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftPercentage : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftSetpoint : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftValue : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringGoToProgramSetpoint : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltPercentage : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltSetpoint : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltValue : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringStop : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringUpOpen : **client-command-macro.h**
- emberAfFillExternalBuffer() : **client-api.h**
- emberAfFillExternalManufacturerSpecificBuffer() : **client-api.h**
- emberAfFindDevicesByProfileAndCluster() : **af.h**
- emberAfFindDevicesSupportingCluster() : **af.h**
- emberAfFindIeeeAddress() : **af.h**
- emberAfFindNodeId() : **af.h**
- emberAfFindUnusedPanIdAndForm() : **af.h**
- emberAfFindUnusedPanIdAndFormCallback() : **callback.h**
- emberAfFlowMeasurementClusterClientAttributeChangedCallback() : **callback.h**
- emberAfFlowMeasurementClusterClientDefaultResponseCallback() : **callback.h**
- emberAfFlowMeasurementClusterClientInitCallback() : **callback.h**
- emberAfFlowMeasurementClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfFlowMeasurementClusterClientMessageSentCallback() : **callback.h**
- emberAfFlowMeasurementClusterClientTickCallback() : **callback.h**
- emberAfFlowMeasurementClusterDebugExec : **debug-printing.h**
- emberAfFlowMeasurementClusterFlush : **debug-printing.h**
- emberAfFlowMeasurementClusterPrint : **debug-printing.h**
- emberAfFlowMeasurementClusterPrintBuffer : **debug-printing.h**
- emberAfFlowMeasurementClusterPrintln : **debug-printing.h**
- emberAfFlowMeasurementClusterPrintString : **debug-printing.h**
- emberAfFlowMeasurementClusterServerAttributeChangedCallback() : **callback.h**
- emberAfFlowMeasurementClusterServerDefaultResponseCallback() : **callback.h**
- emberAfFlowMeasurementClusterServerInitCallback() : **callback.h**
- emberAfFlowMeasurementClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfFlowMeasurementClusterServerMessageSentCallback() : **callback.h**
- emberAfFlowMeasurementClusterServerTickCallback() : **callback.h**
- emberAfFlush() : **af.h**
- emberAfFormNetwork() : **af.h**
- EmberAfGenericClusterFunction : **af-types.h**
- emberAfGenericTunnelClusterAdvertiseProtocolAddressCallback() : **callback.h**
- emberAfGenericTunnelClusterClientAttributeChangedCallback() : **callback.h**
- emberAfGenericTunnelClusterClientCommandParse() : **call-command-handler.h**
- emberAfGenericTunnelClusterClientDefaultResponseCallback() : **callback.h**
- emberAfGenericTunnelClusterClientInitCallback() : **callback.h**
- emberAfGenericTunnelClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfGenericTunnelClusterClientMessageSentCallback() : **callback.h**
- emberAfGenericTunnelClusterClientTickCallback() : **callback.h**
- emberAfGenericTunnelClusterDebugExec : **debug-printing.h**
- emberAfGenericTunnelClusterFlush : **debug-printing.h**
- emberAfGenericTunnelClusterMatchProtocolAddressCallback() : **callback.h**
- emberAfGenericTunnelClusterMatchProtocolAddressResponseCallback() : **callback.h**
- emberAfGenericTunnelClusterPrint : **debug-printing.h**

- emberAfGenericTunnelClusterPrintBuffer : **debug-printing.h**
- emberAfGenericTunnelClusterPrintIn : **debug-printing.h**
- emberAfGenericTunnelClusterPrintString : **debug-printing.h**
- emberAfGenericTunnelClusterServerAttributeChangedCallback() : **callback.h**
- emberAfGenericTunnelClusterServerCommandParse() : **call-command-handler.h**
- emberAfGenericTunnelClusterServerDefaultResponseCallback() : **callback.h**
- emberAfGenericTunnelClusterServerInitCallback() : **callback.h**
- emberAfGenericTunnelClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfGenericTunnelClusterServerMessageSentCallback() : **callback.h**
- emberAfGenericTunnelClusterServerTickCallback() : **callback.h**
- emberAfGetAttributeAnalogOrDiscreteType() : **af.h**
- emberAfGetCommandApsFrame() : **af.h**
- emberAfGetCurrentPrice() : **price-server.h**
- emberAfGetCurrentSleepControl() : **af.h**
- emberAfGetCurrentTime() : **af.h**
- emberAfGetCurrentTimeCallback() : **callback.h**
- emberAfGetDataSize() : **af.h**
- emberAfGetDefaultSleepControl : **af.h**
- emberAfGetEui64() : **af.h**
- emberAfGetInt16u() : **af.h**
- emberAfGetInt24u() : **af.h**
- emberAfGetInt32u() : **af.h**
- emberAfGetInt8u : **af.h**
- emberAfGetNodeId() : **af.h**
- emberAfGetString() : **af.h**
- emberAfGroupsClusterAddGroupCallback() : **callback.h**
- emberAfGroupsClusterAddGroupIfIdentifyingCallback() : **callback.h**
- emberAfGroupsClusterAddGroupResponseCallback() : **callback.h**
- emberAfGroupsClusterClearGroupTableCallback() : **callback.h**
- emberAfGroupsClusterClientAttributeChangedCallback() : **callback.h**
- emberAfGroupsClusterClientCommandParse() : **call-command-handler.h**
- emberAfGroupsClusterClientDefaultResponseCallback() : **callback.h**
- emberAfGroupsClusterClientInitCallback() : **callback.h**
- emberAfGroupsClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfGroupsClusterClientMessageSentCallback() : **callback.h**
- emberAfGroupsClusterClientTickCallback() : **callback.h**
- emberAfGroupsClusterDebugExec : **debug-printing.h**
- emberAfGroupsClusterEndpointInGroupCallback() : **callback.h**
- emberAfGroupsClusterFlush : **debug-printing.h**
- emberAfGroupsClusterGetGroupMembershipCallback() : **callback.h**
- emberAfGroupsClusterGetGroupMembershipResponseCallback() : **callback.h**
- emberAfGroupsClusterPrint : **debug-printing.h**
- emberAfGroupsClusterPrintBuffer : **debug-printing.h**
- emberAfGroupsClusterPrintIn : **debug-printing.h**
- emberAfGroupsClusterPrintString : **debug-printing.h**
- emberAfGroupsClusterRemoveAllGroupsCallback() : **callback.h**
- emberAfGroupsClusterRemoveGroupCallback() : **callback.h**
- emberAfGroupsClusterRemoveGroupResponseCallback() : **callback.h**
- emberAfGroupsClusterServerAttributeChangedCallback() : **callback.h**
- emberAfGroupsClusterServerCommandParse() : **call-command-handler.h**
- emberAfGroupsClusterServerDefaultResponseCallback() : **callback.h**
- emberAfGroupsClusterServerInitCallback() : **callback.h**
- emberAfGroupsClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfGroupsClusterServerMessageSentCallback() : **callback.h**
- emberAfGroupsClusterServerTickCallback() : **callback.h**
- emberAfGroupsClusterViewGroupCallback() : **callback.h**
- emberAfGroupsClusterViewGroupResponseCallback() : **callback.h**

- emberAfGuaranteedFlush : **af.h**
- emberAfGuaranteedPrint : **af.h**
- emberAfGuaranteedPrintBuffer : **af.h**
- emberAfGuaranteedPrintln : **af.h**
- emberAfGuaranteedPrintString : **af.h**
- emberAfHaveDiscoveryResponseStatus : **af-types.h**
- emberAfHibernateDuration : **af.h**
- EmberAfHueDirection : **enums.h**
- EmberAfHueMoveMode : **enums.h**
- EmberAfHueStepMode : **enums.h**
- EmberAfIasAceArmMode : **enums.h**
- EmberAfIasAceArmNotification : **enums.h**
- emberAfIasAceClusterArmCallback() : **callback.h**
- emberAfIasAceClusterArmResponseCallback() : **callback.h**
- emberAfIasAceClusterBypassCallback() : **callback.h**
- emberAfIasAceClusterClientAttributeChangedCallback() : **callback.h**
- emberAfIasAceClusterClientCommandParse() : **call-command-handler.h**
- emberAfIasAceClusterClientDefaultResponseCallback() : **callback.h**
- emberAfIasAceClusterClientInitCallback() : **callback.h**
- emberAfIasAceClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIasAceClusterClientMessageSentCallback() : **callback.h**
- emberAfIasAceClusterClientTickCallback() : **callback.h**
- emberAfIasAceClusterDebugExec : **debug-printing.h**
- emberAfIasAceClusterEmergencyCallback() : **callback.h**
- emberAfIasAceClusterFireCallback() : **callback.h**
- emberAfIasAceClusterFlush : **debug-printing.h**
- emberAfIasAceClusterGetZoneIdMapCallback() : **callback.h**
- emberAfIasAceClusterGetZoneIdMapResponseCallback() : **callback.h**
- emberAfIasAceClusterGetZoneInformationCallback() : **callback.h**
- emberAfIasAceClusterGetZoneInformationResponseCallback() : **callback.h**
- emberAfIasAceClusterPanicCallback() : **callback.h**
- emberAfIasAceClusterPrint : **debug-printing.h**
- emberAfIasAceClusterPrintBuffer : **debug-printing.h**
- emberAfIasAceClusterPrintln : **debug-printing.h**
- emberAfIasAceClusterPrintString : **debug-printing.h**
- emberAfIasAceClusterServerAttributeChangedCallback() : **callback.h**
- emberAfIasAceClusterServerCommandParse() : **call-command-handler.h**
- emberAfIasAceClusterServerDefaultResponseCallback() : **callback.h**
- emberAfIasAceClusterServerInitCallback() : **callback.h**
- emberAfIasAceClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIasAceClusterServerMessageSentCallback() : **callback.h**
- emberAfIasAceClusterServerTickCallback() : **callback.h**
- EmberAfIasEnrollResponseCode : **enums.h**
- emberAfIasWdClusterClientAttributeChangedCallback() : **callback.h**
- emberAfIasWdClusterClientDefaultResponseCallback() : **callback.h**
- emberAfIasWdClusterClientInitCallback() : **callback.h**
- emberAfIasWdClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIasWdClusterClientMessageSentCallback() : **callback.h**
- emberAfIasWdClusterClientTickCallback() : **callback.h**
- emberAfIasWdClusterDebugExec : **debug-printing.h**
- emberAfIasWdClusterFlush : **debug-printing.h**
- emberAfIasWdClusterPrint : **debug-printing.h**
- emberAfIasWdClusterPrintBuffer : **debug-printing.h**
- emberAfIasWdClusterPrintln : **debug-printing.h**
- emberAfIasWdClusterPrintString : **debug-printing.h**
- emberAfIasWdClusterServerAttributeChangedCallback() : **callback.h**
- emberAfIasWdClusterServerCommandParse() : **call-command-handler.h**

- emberAfIasWdClusterServerDefaultResponseCallback() : **callback.h**
- emberAfIasWdClusterServerInitCallback() : **callback.h**
- emberAfIasWdClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIasWdClusterServerMessageSentCallback() : **callback.h**
- emberAfIasWdClusterServerTickCallback() : **callback.h**
- emberAfIasWdClusterSquawkCallback() : **callback.h**
- emberAfIasWdClusterStartWarningCallback() : **callback.h**
- emberAfIasZoneClusterClientAttributeChangedCallback() : **callback.h**
- emberAfIasZoneClusterClientCommandParse() : **call-command-handler.h**
- emberAfIasZoneClusterClientDefaultResponseCallback() : **callback.h**
- emberAfIasZoneClusterClientInitCallback() : **callback.h**
- emberAfIasZoneClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIasZoneClusterClientMessageSentCallback() : **callback.h**
- emberAfIasZoneClusterClientTickCallback() : **callback.h**
- emberAfIasZoneClusterDebugExec : **debug-printing.h**
- emberAfIasZoneClusterFlush : **debug-printing.h**
- emberAfIasZoneClusterPrint : **debug-printing.h**
- emberAfIasZoneClusterPrintBuffer : **debug-printing.h**
- emberAfIasZoneClusterPrintln : **debug-printing.h**
- emberAfIasZoneClusterPrintString : **debug-printing.h**
- emberAfIasZoneClusterServerAttributeChangedCallback() : **callback.h**
- emberAfIasZoneClusterServerCommandParse() : **call-command-handler.h**
- emberAfIasZoneClusterServerDefaultResponseCallback() : **callback.h**
- emberAfIasZoneClusterServerInitCallback() : **callback.h**
- emberAfIasZoneClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIasZoneClusterServerMessageSentCallback() : **callback.h**
- emberAfIasZoneClusterServerTickCallback() : **callback.h**
- emberAfIasZoneClusterZoneEnrollRequestCallback() : **callback.h**
- emberAfIasZoneClusterZoneEnrollResponseCallback() : **callback.h**
- emberAfIasZoneClusterZoneStatusChangeNotificationCallback() : **callback.h**
- EmberAfIasZoneState : **enums.h**
- EmberAfIasZoneType : **enums.h**
- emberAfIdentifyClusterClientAttributeChangedCallback() : **callback.h**
- emberAfIdentifyClusterClientCommandParse() : **call-command-handler.h**
- emberAfIdentifyClusterClientDefaultResponseCallback() : **callback.h**
- emberAfIdentifyClusterClientInitCallback() : **callback.h**
- emberAfIdentifyClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIdentifyClusterClientMessageSentCallback() : **callback.h**
- emberAfIdentifyClusterClientTickCallback() : **callback.h**
- emberAfIdentifyClusterDebugExec : **debug-printing.h**
- emberAfIdentifyClusterFlush : **debug-printing.h**
- emberAfIdentifyClusterIdentifyCallback() : **callback.h**
- emberAfIdentifyClusterIdentifyQueryCallback() : **callback.h**
- emberAfIdentifyClusterIdentifyQueryResponseCallback() : **callback.h**
- emberAfIdentifyClusterPrint : **debug-printing.h**
- emberAfIdentifyClusterPrintBuffer : **debug-printing.h**
- emberAfIdentifyClusterPrintln : **debug-printing.h**
- emberAfIdentifyClusterPrintString : **debug-printing.h**
- emberAfIdentifyClusterServerAttributeChangedCallback() : **callback.h**
- emberAfIdentifyClusterServerCommandParse() : **call-command-handler.h**
- emberAfIdentifyClusterServerDefaultResponseCallback() : **callback.h**
- emberAfIdentifyClusterServerInitCallback() : **callback.h**
- emberAfIdentifyClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIdentifyClusterServerMessageSentCallback() : **callback.h**
- emberAfIdentifyClusterServerTickCallback() : **callback.h**
- emberAfIllumLevelSensingClusterClientAttributeChangedCallback() : **callback.h**
- emberAfIllumLevelSensingClusterClientDefaultResponseCallback() : **callback.h**

- emberAfIllumLevelSensingClusterClientInitCallback() : **callback.h**
- emberAfIllumLevelSensingClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIllumLevelSensingClusterClientMessageSentCallback() : **callback.h**
- emberAfIllumLevelSensingClusterClientTickCallback() : **callback.h**
- emberAfIllumLevelSensingClusterDebugExec : **debug-printing.h**
- emberAfIllumLevelSensingClusterFlush : **debug-printing.h**
- emberAfIllumLevelSensingClusterPrint : **debug-printing.h**
- emberAfIllumLevelSensingClusterPrintBuffer : **debug-printing.h**
- emberAfIllumLevelSensingClusterPrintln : **debug-printing.h**
- emberAfIllumLevelSensingClusterPrintString : **debug-printing.h**
- emberAfIllumLevelSensingClusterServerAttributeChangedCallback() : **callback.h**
- emberAfIllumLevelSensingClusterServerDefaultResponseCallback() : **callback.h**
- emberAfIllumLevelSensingClusterServerInitCallback() : **callback.h**
- emberAfIllumLevelSensingClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIllumLevelSensingClusterServerMessageSentCallback() : **callback.h**
- emberAfIllumLevelSensingClusterServerTickCallback() : **callback.h**
- emberAfIllumMeasurementClusterClientAttributeChangedCallback() : **callback.h**
- emberAfIllumMeasurementClusterClientDefaultResponseCallback() : **callback.h**
- emberAfIllumMeasurementClusterClientInitCallback() : **callback.h**
- emberAfIllumMeasurementClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIllumMeasurementClusterClientMessageSentCallback() : **callback.h**
- emberAfIllumMeasurementClusterClientTickCallback() : **callback.h**
- emberAfIllumMeasurementClusterDebugExec : **debug-printing.h**
- emberAfIllumMeasurementClusterFlush : **debug-printing.h**
- emberAfIllumMeasurementClusterPrint : **debug-printing.h**
- emberAfIllumMeasurementClusterPrintBuffer : **debug-printing.h**
- emberAfIllumMeasurementClusterPrintln : **debug-printing.h**
- emberAfIllumMeasurementClusterPrintString : **debug-printing.h**
- emberAfIllumMeasurementClusterServerAttributeChangedCallback() : **callback.h**
- emberAfIllumMeasurementClusterServerDefaultResponseCallback() : **callback.h**
- emberAfIllumMeasurementClusterServerInitCallback() : **callback.h**
- emberAfIllumMeasurementClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIllumMeasurementClusterServerMessageSentCallback() : **callback.h**
- emberAfIllumMeasurementClusterServerTickCallback() : **callback.h**
- EmberAfImageVerifyStatus : **af-types.h**
- emberAfIndexFromEndpoint() : **af.h**
- EmberAfInitFunction : **af-types.h**
- emberAfInitiateInterPanKeyEstablishment() : **af.h**
- emberAfInitiateInterPanKeyEstablishmentCallback() : **callback.h**
- emberAfInitiateKeyEstablishment() : **af.h**
- emberAfInitiateKeyEstablishmentCallback() : **callback.h**
- emberAfInitiatePartnerLinkKeyExchange() : **af.h**
- emberAfInitiatePartnerLinkKeyExchangeCallback() : **callback.h**
- emberAfInterpanInitCallback() : **callback.h**
- emberAfInterPanKeyEstablishmentCallback() : **callback.h**
- EmberAfInterpanMessageType : **af-types.h**
- EmberAfInterpanOptions : **af-types.h**
- emberAfInterpanSendMessageCallback() : **callback.h**
- emberAfInvalidImageId : **af.h**
- emberAfIsDeviceEnabled() : **af.h**
- emberAfIsDeviceIdentifying() : **identify.h**
- emberAfIsOtaImageIdValid() : **ota-storage.h**
- emberAfIsThisDataTypeAStringType() : **af.h**
- emberAfIsTypeSigned() : **af.h**
- emberAfJoinableNetworkFoundCallback() : **callback.h**
- emberAfJoinNetwork() : **af.h**
- emberAfKeyEstablishmentCallback() : **callback.h**

- emberAfKeyEstablishmentClusterClientAttributeChangedCallback() : **callback.h**
- emberAfKeyEstablishmentClusterClientCommandParse() : **call-command-handler.h**
- emberAfKeyEstablishmentClusterClientCommandReceivedCallback() : **callback.h**
- emberAfKeyEstablishmentClusterClientDefaultResponseCallback() : **callback.h**
- emberAfKeyEstablishmentClusterClientInitCallback() : **callback.h**
- emberAfKeyEstablishmentClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfKeyEstablishmentClusterClientMessageSentCallback() : **callback.h**
- emberAfKeyEstablishmentClusterClientTickCallback() : **callback.h**
- emberAfKeyEstablishmentClusterConfirmKeyDataRequestCallback() : **callback.h**
- emberAfKeyEstablishmentClusterConfirmKeyDataResponseCallback() : **callback.h**
- emberAfKeyEstablishmentClusterDebugExec : **debug-printing.h**
- emberAfKeyEstablishmentClusterEphemeralDataRequestCallback() : **callback.h**
- emberAfKeyEstablishmentClusterEphemeralDataResponseCallback() : **callback.h**
- emberAfKeyEstablishmentClusterFlush : **debug-printing.h**
- emberAfKeyEstablishmentClusterInitiateKeyEstablishmentRequestCallback() : **callback.h**
- emberAfKeyEstablishmentClusterInitiateKeyEstablishmentResponseCallback() : **callback.h**
- emberAfKeyEstablishmentClusterPrint : **debug-printing.h**
- emberAfKeyEstablishmentClusterPrintBuffer : **debug-printing.h**
- emberAfKeyEstablishmentClusterPrintln : **debug-printing.h**
- emberAfKeyEstablishmentClusterPrintString : **debug-printing.h**
- emberAfKeyEstablishmentClusterServerAttributeChangedCallback() : **callback.h**
- emberAfKeyEstablishmentClusterServerCommandParse() : **call-command-handler.h**
- emberAfKeyEstablishmentClusterServerCommandReceivedCallback() : **callback.h**
- emberAfKeyEstablishmentClusterServerDefaultResponseCallback() : **callback.h**
- emberAfKeyEstablishmentClusterServerInitCallback() : **callback.h**
- emberAfKeyEstablishmentClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfKeyEstablishmentClusterServerMessageSentCallback() : **callback.h**
- emberAfKeyEstablishmentClusterServerTickCallback() : **callback.h**
- emberAfKeyEstablishmentClusterTerminateKeyEstablishmentCallback() : **callback.h**
- EmberAfKeyEstablishmentNotifyMessage : **af-types.h**
- EmberAfKeypadLockout : **enums.h**
- emberAfLevelControlClusterClientAttributeChangedCallback() : **callback.h**
- emberAfLevelControlClusterClientDefaultResponseCallback() : **callback.h**
- emberAfLevelControlClusterClientInitCallback() : **callback.h**
- emberAfLevelControlClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfLevelControlClusterClientMessageSentCallback() : **callback.h**
- emberAfLevelControlClusterClientTickCallback() : **callback.h**
- emberAfLevelControlClusterDebugExec : **debug-printing.h**
- emberAfLevelControlClusterFlush : **debug-printing.h**
- emberAfLevelControlClusterMoveCallback() : **callback.h**
- emberAfLevelControlClusterMoveToLevelCallback() : **callback.h**
- emberAfLevelControlClusterMoveToLevelWithOnOffCallback() : **callback.h**
- emberAfLevelControlClusterMoveWithOnOffCallback() : **callback.h**
- emberAfLevelControlClusterPrint : **debug-printing.h**
- emberAfLevelControlClusterPrintBuffer : **debug-printing.h**
- emberAfLevelControlClusterPrintln : **debug-printing.h**
- emberAfLevelControlClusterPrintString : **debug-printing.h**
- emberAfLevelControlClusterServerAttributeChangedCallback() : **callback.h**
- emberAfLevelControlClusterServerCommandParse() : **call-command-handler.h**
- emberAfLevelControlClusterServerDefaultResponseCallback() : **callback.h**
- emberAfLevelControlClusterServerInitCallback() : **callback.h**
- emberAfLevelControlClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfLevelControlClusterServerMessageSentCallback() : **callback.h**
- emberAfLevelControlClusterServerTickCallback() : **callback.h**
- emberAfLevelControlClusterStepCallback() : **callback.h**
- emberAfLevelControlClusterStepWithOnOffCallback() : **callback.h**
- emberAfLevelControlClusterStopCallback() : **callback.h**

- emberAfLevelControlClusterStopWithOnOffCallback() : **callback.h**
- EmberAfLevelStatus : **enums.h**
- emberAfLocateAttributeMetadata() : **af.h**
- EmberAfLocationMethod : **enums.h**
- emberAfMainInitCallback() : **callback.h**
- emberAfMainTickCallback() : **callback.h**
- EmberAfManufacturerSpecificClusterAttributeChangedCallback : **af-types.h**
- emberAfMaximumApsPayloadLength() : **af.h**
- EmberAfMeasurementLightSensorType : **enums.h**
- emberAfMessageSentCallback() : **callback.h**
- EmberAfMessageSentFunction : **af-types.h**
- emberAfMessagingClusterCancelMessageCallback() : **callback.h**
- emberAfMessagingClusterClientAttributeChangedCallback() : **callback.h**
- emberAfMessagingClusterClientCommandParse() : **call-command-handler.h**
- emberAfMessagingClusterClientDefaultResponseCallback() : **callback.h**
- emberAfMessagingClusterClientInitCallback() : **callback.h**
- emberAfMessagingClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfMessagingClusterClientMessageSentCallback() : **callback.h**
- emberAfMessagingClusterClientTickCallback() : **callback.h**
- emberAfMessagingClusterDebugExec : **debug-printing.h**
- emberAfMessagingClusterDisplayMessageCallback() : **callback.h**
- emberAfMessagingClusterFlush : **debug-printing.h**
- emberAfMessagingClusterGetLastMessageCallback() : **callback.h**
- emberAfMessagingClusterMessageConfirmationCallback() : **callback.h**
- emberAfMessagingClusterPrint : **debug-printing.h**
- emberAfMessagingClusterPrintBuffer : **debug-printing.h**
- emberAfMessagingClusterPrintIn : **debug-printing.h**
- emberAfMessagingClusterPrintString : **debug-printing.h**
- emberAfMessagingClusterServerAttributeChangedCallback() : **callback.h**
- emberAfMessagingClusterServerCommandParse() : **call-command-handler.h**
- emberAfMessagingClusterServerDefaultResponseCallback() : **callback.h**
- emberAfMessagingClusterServerInitCallback() : **callback.h**
- emberAfMessagingClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfMessagingClusterServerMessageSentCallback() : **callback.h**
- emberAfMessagingClusterServerTickCallback() : **callback.h**
- EmberAfMessagingControlConfirmation : **enums.h**
- EmberAfMessagingControlImportance : **enums.h**
- EmberAfMessagingControlTransmission : **enums.h**
- EmberAfMeterDeviceType : **enums.h**
- EmberAfMeteringAlarmCode : **enums.h**
- EmberAfMeteringBlockEnumerations : **enums.h**
- EmberAfMeteringSupplyStatus : **enums.h**
- EmberAfMeteringTemperatureUnitOfMeasure : **enums.h**
- EmberAfMoveMode : **enums.h**
- emberAfMsToNextEvent() : **af.h**
- emberAfNcpInitCallback() : **callback.h**
- emberAfNetworkFindOperationCompleteCallback() : **callback.h**
- emberAfNextSequence() : **af.h**
- emberAfOccupancySensingClusterClientAttributeChangedCallback() : **callback.h**
- emberAfOccupancySensingClusterClientDefaultResponseCallback() : **callback.h**
- emberAfOccupancySensingClusterClientInitCallback() : **callback.h**
- emberAfOccupancySensingClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOccupancySensingClusterClientMessageSentCallback() : **callback.h**
- emberAfOccupancySensingClusterClientTickCallback() : **callback.h**
- emberAfOccupancySensingClusterDebugExec : **debug-printing.h**
- emberAfOccupancySensingClusterFlush : **debug-printing.h**
- emberAfOccupancySensingClusterPrint : **debug-printing.h**

- emberAfOccupancySensingClusterPrintBuffer : **debug-printing.h**
- emberAfOccupancySensingClusterPrintln : **debug-printing.h**
- emberAfOccupancySensingClusterPrintString : **debug-printing.h**
- emberAfOccupancySensingClusterServerAttributeChangedCallback() : **callback.h**
- emberAfOccupancySensingClusterServerDefaultResponseCallback() : **callback.h**
- emberAfOccupancySensingClusterServerInitCallback() : **callback.h**
- emberAfOccupancySensingClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOccupancySensingClusterServerMessageSentCallback() : **callback.h**
- emberAfOccupancySensingClusterServerTickCallback() : **callback.h**
- EmberAfOccupancySensorType : **enums.h**
- emberAfOkToHibernate() : **af.h**
- emberAfOkToNap : **af.h**
- emberAfOnOffClusterClientAttributeChangedCallback() : **callback.h**
- emberAfOnOffClusterClientDefaultResponseCallback() : **callback.h**
- emberAfOnOffClusterClientInitCallback() : **callback.h**
- emberAfOnOffClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOnOffClusterClientMessageSentCallback() : **callback.h**
- emberAfOnOffClusterClientTickCallback() : **callback.h**
- emberAfOnOffClusterDebugExec : **debug-printing.h**
- emberAfOnOffClusterFlush : **debug-printing.h**
- emberAfOnOffClusterOffCallback() : **callback.h**
- emberAfOnOffClusterOnCallback() : **callback.h**
- emberAfOnOffClusterPrint : **debug-printing.h**
- emberAfOnOffClusterPrintBuffer : **debug-printing.h**
- emberAfOnOffClusterPrintln : **debug-printing.h**
- emberAfOnOffClusterPrintString : **debug-printing.h**
- emberAfOnOffClusterSampleMfgSpecificOffWithTransitionCallback() : **callback.h**
- emberAfOnOffClusterSampleMfgSpecificOnWithTransitionCallback() : **callback.h**
- emberAfOnOffClusterSampleMfgSpecificToggleWithTransitionCallback() : **callback.h**
- emberAfOnOffClusterServerAttributeChangedCallback() : **callback.h**
- emberAfOnOffClusterServerCommandParse() : **call-command-handler.h**
- emberAfOnOffClusterServerDefaultResponseCallback() : **callback.h**
- emberAfOnOffClusterServerInitCallback() : **callback.h**
- emberAfOnOffClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOnOffClusterServerMessageSentCallback() : **callback.h**
- emberAfOnOffClusterServerTickCallback() : **callback.h**
- emberAfOnOffClusterToggleCallback() : **callback.h**
- emberAfOnOffSetValue() : **on-off.h**
- emberAfOnOffSwitchConfigClusterClientAttributeChangedCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterClientDefaultResponseCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterClientInitCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterClientMessageSentCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterClientTickCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterDebugExec : **debug-printing.h**
- emberAfOnOffSwitchConfigClusterFlush : **debug-printing.h**
- emberAfOnOffSwitchConfigClusterPrint : **debug-printing.h**
- emberAfOnOffSwitchConfigClusterPrintBuffer : **debug-printing.h**
- emberAfOnOffSwitchConfigClusterPrintln : **debug-printing.h**
- emberAfOnOffSwitchConfigClusterPrintString : **debug-printing.h**
- emberAfOnOffSwitchConfigClusterServerAttributeChangedCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterServerDefaultResponseCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterServerInitCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterServerMessageSentCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterServerTickCallback() : **callback.h**
- EmberAfOperatingMode : **enums.h**

- emberAfOtaBootloadCallback() : **callback.h**
- emberAfOtaBootloadClusterClientAttributeChangedCallback() : **callback.h**
- emberAfOtaBootloadClusterClientDefaultResponseCallback() : **callback.h**
- emberAfOtaBootloadClusterClientInitCallback() : **callback.h**
- emberAfOtaBootloadClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOtaBootloadClusterClientMessageSentCallback() : **callback.h**
- emberAfOtaBootloadClusterClientTickCallback() : **callback.h**
- emberAfOtaBootloadClusterDebugExec : **debug-printing.h**
- emberAfOtaBootloadClusterFlush : **debug-printing.h**
- emberAfOtaBootloadClusterImageBlockRequestCallback() : **callback.h**
- emberAfOtaBootloadClusterImageBlockResponseCallback() : **callback.h**
- emberAfOtaBootloadClusterImageNotifyCallback() : **callback.h**
- emberAfOtaBootloadClusterImagePageRequestCallback() : **callback.h**
- emberAfOtaBootloadClusterPrint : **debug-printing.h**
- emberAfOtaBootloadClusterPrintBuffer : **debug-printing.h**
- emberAfOtaBootloadClusterPrintln : **debug-printing.h**
- emberAfOtaBootloadClusterPrintString : **debug-printing.h**
- emberAfOtaBootloadClusterQueryNextImageRequestCallback() : **callback.h**
- emberAfOtaBootloadClusterQueryNextImageResponseCallback() : **callback.h**
- emberAfOtaBootloadClusterQuerySpecificFileRequestCallback() : **callback.h**
- emberAfOtaBootloadClusterQuerySpecificFileResponseCallback() : **callback.h**
- emberAfOtaBootloadClusterServerAttributeChangedCallback() : **callback.h**
- emberAfOtaBootloadClusterServerDefaultResponseCallback() : **callback.h**
- emberAfOtaBootloadClusterServerInitCallback() : **callback.h**
- emberAfOtaBootloadClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOtaBootloadClusterServerMessageSentCallback() : **callback.h**
- emberAfOtaBootloadClusterServerTickCallback() : **callback.h**
- emberAfOtaBootloadClusterUpgradeEndRequestCallback() : **callback.h**
- emberAfOtaBootloadClusterUpgradeEndResponseCallback() : **callback.h**
- emberAfOtaClientBootloadCallback() : **callback.h**
- emberAfOtaClientCustomVerifyCallback() : **callback.h**
- emberAfOtaClientDownloadCompleteCallback() : **callback.h**
- emberAfOtaClientIncomingMessageRawCallback() : **callback.h**
- emberAfOtaClientStartCallback() : **callback.h**
- emberAfOtaClientVersionInfoCallback() : **callback.h**
- EmberAfOtaDownloadResult : **ota-client-policy.h**
- emberAfOtaPageRequestServerPolicyCallback() : **callback.h**
- emberAfOtaServerBlockSizeCallback() : **callback.h**
- emberAfOtaServerImageBlockRequestCallback() : **callback.h**
- emberAfOtaServerIncomingMessageRawCallback() : **callback.h**
- emberAfOtaServerQueryCallback() : **callback.h**
- emberAfOtaServerSendImageNotifyCallback() : **callback.h**
- emberAfOtaServerSendUpgradeCommandCallback() : **ota-server.h**
- emberAfOtaServerUpgradeEndRequestCallback() : **callback.h**
- emberAfOtaStorageCheckTempDataCallback() : **callback.h**
- emberAfOtaStorageClearTempDataCallback() : **callback.h**
- emberAfOtaStorageCloseCallback() : **callback.h**
- emberAfOtaStorageDeleteImageCallback() : **ota-storage.h**
- emberAfOtaStorageDriverDownloadFinishCallback() : **ota-storage.h**
- emberAfOtaStorageDriverInitCallback() : **callback.h**
- emberAfOtaStorageDriverReadCallback() : **callback.h**
- emberAfOtaStorageDriverRetrieveLastStoredOffsetCallback() : **ota-storage.h**
- emberAfOtaStorageDriverWriteCallback() : **callback.h**
- emberAfOtaStorageFinishDownloadCallback() : **callback.h**
- emberAfOtaStorageGetCountCallback() : **callback.h**
- emberAfOtaStorageGetFullHeaderCallback() : **callback.h**
- emberAfOtaStorageGetTotalImageSizeCallback() : **callback.h**

- emberAfOtaStorageInitCallback() : **callback.h**
- emberAfOtaStorageIteratorFirstCallback() : **callback.h**
- emberAfOtaStorageIteratorNextCallback() : **callback.h**
- emberAfOtaStorageReadImageDataCallback() : **callback.h**
- emberAfOtaStorageSearchCallback() : **callback.h**
- EmberAfOtaStorageStatus : **af-types.h**
- emberAfOtaStorageWriteTempDataCallback() : **callback.h**
- EmberAfPartnerLinkKeyExchangeCallback : **af-types.h**
- emberAfPartnerLinkKeyExchangeRequestCallback() : **callback.h**
- emberAfPartnerLinkKeyExchangeResponseCallback() : **callback.h**
- emberAfPerformingKeyEstablishment() : **af.h**
- emberAfPerformingKeyEstablishmentCallback() : **callback.h**
- EmberAfPhysicalEnvironment : **enums.h**
- emberAfPluginBasicResetToFactoryDefaultsCallback() : **basic-callback.h**
- emberAfPluginDrlcEventActionCallback() : **drlc-callback.h**
- emberAfPluginMessagingClientCancelMessageCallback() : **messaging-client-callback.h**
- emberAfPluginMessagingClientConfirmMessage() : **messaging-client.h**
- emberAfPluginMessagingClientDisplayMessageCallback() : **messaging-client-callback.h**
- emberAfPluginPriceClientPriceExpiredCallback() : **price-client-callback.h**
- emberAfPluginPriceClientPriceStartedCallback() : **price-client-callback.h**
- emberAfPluginPslCommissioningEndpointInformationCallback() : **psl-commissioning-callback.h**
- emberAfPluginPslCommissioningEndpointInformationCountCallback() : **psl-commissioning-callback.h**
- emberAfPluginPslCommissioningGroupIdentifierCallback() : **psl-commissioning-callback.h**
- emberAfPluginPslCommissioningGroupIdentifierCountCallback() : **psl-commissioning-callback.h**
- emberAfPluginPslCommissioningIdentifyCallback() : **psl-commissioning-callback.h**
- emberAfPluginPslCommissioningTouchLinkCompleteCallback() : **psl-commissioning-callback.h**
- emberAfPluginPslCommissioningTouchLinkFailedCallback() : **psl-commissioning-callback.h**
- emberAfPluginSimpleMeteringClientRemoveMirrorCallback() : **simple-metering-client-callback.h**
- emberAfPluginSimpleMeteringClientRequestMirrorCallback() : **simple-metering-client-callback.h**
- emberAfPluginTestExampleCustomerFunctionCallback() : **test-callback.h**
- emberAfPluginTunnelingClientCloseTunnel() : **tunneling-client.h**
- emberAfPluginTunnelingClientDataErrorCallback() : **tunneling-client-callback.h**
- emberAfPluginTunnelingClientDataReceivedCallback() : **tunneling-client-callback.h**
- emberAfPluginTunnelingClientTransferData() : **tunneling-client.h**
- emberAfPluginTunnelingClientTunnelOpenedCallback() : **tunneling-client-callback.h**
- emberAfPluginTunnelingServerDataErrorCallback() : **tunneling-server-callback.h**
- emberAfPluginTunnelingServerDataReceivedCallback() : **tunneling-server-callback.h**
- emberAfPluginTunnelingServerIsProtocolSupportedCallback() : **tunneling-server-callback.h**
- emberAfPluginTunnelingServerTransferData() : **tunneling-server.h**
- emberAfPluginTunnelingServerTunnelClosedCallback() : **tunneling-server-callback.h**
- emberAfPluginTunnelingServerTunnelOpenedCallback() : **tunneling-server-callback.h**
- emberAfPollCompleteHandler() : **af.h**
- emberAfPostAttributeChangeCallback() : **callback.h**
- emberAfPostWakeUpCallback() : **callback.h**
- emberAfPowerConfigClusterClientAttributeChangedCallback() : **callback.h**
- emberAfPowerConfigClusterClientDefaultResponseCallback() : **callback.h**
- emberAfPowerConfigClusterClientInitCallback() : **callback.h**
- emberAfPowerConfigClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPowerConfigClusterClientMessageSentCallback() : **callback.h**
- emberAfPowerConfigClusterClientTickCallback() : **callback.h**
- emberAfPowerConfigClusterDebugExec : **debug-printing.h**
- emberAfPowerConfigClusterFlush : **debug-printing.h**
- emberAfPowerConfigClusterPrint : **debug-printing.h**
- emberAfPowerConfigClusterPrintBuffer : **debug-printing.h**
- emberAfPowerConfigClusterPrintln : **debug-printing.h**
- emberAfPowerConfigClusterPrintString : **debug-printing.h**
- emberAfPowerConfigClusterServerAttributeChangedCallback() : **callback.h**

- emberAfPowerConfigClusterServerDefaultResponseCallback() : **callback.h**
- emberAfPowerConfigClusterServerInitCallback() : **callback.h**
- emberAfPowerConfigClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPowerConfigClusterServerMessageSentCallback() : **callback.h**
- emberAfPowerConfigClusterServerTickCallback() : **callback.h**
- EmberAfPowerSource : **enums.h**
- emberAfPreAttributeChangeCallback() : **callback.h**
- emberAfPreCommandReceivedCallback() : **callback.h**
- emberAfPreGoToSleepCallback() : **callback.h**
- emberAfPreMessageReceivedCallback() : **callback.h**
- emberAfPrepaymentClusterChangeSupplyCallback() : **callback.h**
- emberAfPrepaymentClusterClientAttributeChangedCallback() : **callback.h**
- emberAfPrepaymentClusterClientCommandParse() : **call-command-handler.h**
- emberAfPrepaymentClusterClientDefaultResponseCallback() : **callback.h**
- emberAfPrepaymentClusterClientInitCallback() : **callback.h**
- emberAfPrepaymentClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPrepaymentClusterClientMessageSentCallback() : **callback.h**
- emberAfPrepaymentClusterClientTickCallback() : **callback.h**
- emberAfPrepaymentClusterDebugExec : **debug-printing.h**
- emberAfPrepaymentClusterFlush : **debug-printing.h**
- emberAfPrepaymentClusterPrint : **debug-printing.h**
- emberAfPrepaymentClusterPrintBuffer : **debug-printing.h**
- emberAfPrepaymentClusterPrintln : **debug-printing.h**
- emberAfPrepaymentClusterPrintString : **debug-printing.h**
- emberAfPrepaymentClusterSelectAvailableEmergencyCreditCallback() : **callback.h**
- emberAfPrepaymentClusterServerAttributeChangedCallback() : **callback.h**
- emberAfPrepaymentClusterServerCommandParse() : **call-command-handler.h**
- emberAfPrepaymentClusterServerDefaultResponseCallback() : **callback.h**
- emberAfPrepaymentClusterServerInitCallback() : **callback.h**
- emberAfPrepaymentClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPrepaymentClusterServerMessageSentCallback() : **callback.h**
- emberAfPrepaymentClusterServerTickCallback() : **callback.h**
- emberAfPrepaymentClusterSupplyStatusResponseCallback() : **callback.h**
- EmberAfPrepaymentOriginatingDevice : **enums.h**
- EmberAfPrepaymentRecoveryPeriod : **enums.h**
- EmberAfPrepaymentSupplyStatus : **enums.h**
- emberAfPressureMeasurementClusterClientAttributeChangedCallback() : **callback.h**
- emberAfPressureMeasurementClusterClientDefaultResponseCallback() : **callback.h**
- emberAfPressureMeasurementClusterClientInitCallback() : **callback.h**
- emberAfPressureMeasurementClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPressureMeasurementClusterClientMessageSentCallback() : **callback.h**
- emberAfPressureMeasurementClusterClientTickCallback() : **callback.h**
- emberAfPressureMeasurementClusterDebugExec : **debug-printing.h**
- emberAfPressureMeasurementClusterFlush : **debug-printing.h**
- emberAfPressureMeasurementClusterPrint : **debug-printing.h**
- emberAfPressureMeasurementClusterPrintBuffer : **debug-printing.h**
- emberAfPressureMeasurementClusterPrintln : **debug-printing.h**
- emberAfPressureMeasurementClusterPrintString : **debug-printing.h**
- emberAfPressureMeasurementClusterServerAttributeChangedCallback() : **callback.h**
- emberAfPressureMeasurementClusterServerDefaultResponseCallback() : **callback.h**
- emberAfPressureMeasurementClusterServerInitCallback() : **callback.h**
- emberAfPressureMeasurementClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPressureMeasurementClusterServerMessageSentCallback() : **callback.h**
- emberAfPressureMeasurementClusterServerTickCallback() : **callback.h**
- emberAfPreZDOMessageReceivedCallback() : **callback.h**
- emberAfPriceClearPriceTable() : **price-server.h**
- emberAfPriceClusterClientAttributeChangedCallback() : **callback.h**

- emberAfPriceClusterClientCommandParse() : **call-command-handler.h**
- emberAfPriceClusterClientDefaultResponseCallback() : **callback.h**
- emberAfPriceClusterClientInitCallback() : **callback.h**
- emberAfPriceClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPriceClusterClientMessageSentCallback() : **callback.h**
- emberAfPriceClusterClientTickCallback() : **callback.h**
- emberAfPriceClusterDebugExec : **debug-printing.h**
- emberAfPriceClusterFlush : **debug-printing.h**
- emberAfPriceClusterGetBlockPeriodsCallback() : **callback.h**
- emberAfPriceClusterGetCurrentPriceCallback() : **callback.h**
- emberAfPriceClusterGetScheduledPricesCallback() : **callback.h**
- emberAfPriceClusterPriceAcknowledgementCallback() : **callback.h**
- emberAfPriceClusterPrint : **debug-printing.h**
- emberAfPriceClusterPrintBuffer : **debug-printing.h**
- emberAfPriceClusterPrintIn : **debug-printing.h**
- emberAfPriceClusterPrintString : **debug-printing.h**
- emberAfPriceClusterPublishBlockPeriodCallback() : **callback.h**
- emberAfPriceClusterPublishPriceCallback() : **callback.h**
- emberAfPriceClusterServerAttributeChangedCallback() : **callback.h**
- emberAfPriceClusterServerCommandParse() : **call-command-handler.h**
- emberAfPriceClusterServerDefaultResponseCallback() : **callback.h**
- emberAfPriceClusterServerInitCallback() : **callback.h**
- emberAfPriceClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPriceClusterServerMessageSentCallback() : **callback.h**
- emberAfPriceClusterServerTickCallback() : **callback.h**
- EmberAfPriceControlAcknowledgement : **enums.h**
- emberAfPriceGetPriceTableEntry() : **price-server.h**
- emberAfPricePrintTable() : **price-server.h**
- EmberAfPriceTier : **enums.h**
- emberAfPrimaryEndpoint : **af.h**
- emberAfPrimaryProfileId : **af.h**
- emberAfPrint() : **af.h**
- emberAfPrintAllOff() : **af.h**
- emberAfPrintAllOn() : **af.h**
- emberAfPrintBigEndianEui64() : **af.h**
- emberAfPrintBuffer() : **af.h**
- emberAfPrintChannelListFromMask() : **network-cli.h**
- emberAfPrintEnabled() : **af.h**
- emberAfPrintLittleEndianEui64() : **af.h**
- emberAfPrintIn() : **af.h**
- emberAfPrintMessageData() : **af.h**
- emberAfPrintOff() : **af.h**
- emberAfPrintOn() : **af.h**
- emberAfPrintStatus() : **af.h**
- emberAfPrintString() : **af.h**
- emberAfPrintTime() : **af.h**
- EmberAfProfileId : **af-types.h**
- emberAfProfileIdFromIndex : **af.h**
- emberAfPslAbortTouchLink() : **psl-commissioning.h**
- emberAfPslDeviceInformationRequest() : **psl-commissioning.h**
- emberAfPslIdentifyRequest() : **psl-commissioning.h**
- emberAfPslInitiateTouchLink() : **psl-commissioning.h**
- emberAfPslResetToFactoryNew() : **psl-commissioning.h**
- emberAfPslResetToFactoryNewRequest() : **psl-commissioning.h**
- emberAfPslSetInitialSecurityState() : **psl-commissioning.h**
- emberAfPslTouchLinkInProgress() : **psl-commissioning.h**
- emberAfPumpConfigControlClusterClientAttributeChangedCallback() : **callback.h**

- emberAfPumpConfigControlClusterClientDefaultResponseCallback() : **callback.h**
- emberAfPumpConfigControlClusterClientInitCallback() : **callback.h**
- emberAfPumpConfigControlClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPumpConfigControlClusterClientMessageSentCallback() : **callback.h**
- emberAfPumpConfigControlClusterClientTickCallback() : **callback.h**
- emberAfPumpConfigControlClusterDebugExec : **debug-printing.h**
- emberAfPumpConfigControlClusterFlush : **debug-printing.h**
- emberAfPumpConfigControlClusterPrint : **debug-printing.h**
- emberAfPumpConfigControlClusterPrintBuffer : **debug-printing.h**
- emberAfPumpConfigControlClusterPrintln : **debug-printing.h**
- emberAfPumpConfigControlClusterPrintString : **debug-printing.h**
- emberAfPumpConfigControlClusterServerAttributeChangedCallback() : **callback.h**
- emberAfPumpConfigControlClusterServerDefaultResponseCallback() : **callback.h**
- emberAfPumpConfigControlClusterServerInitCallback() : **callback.h**
- emberAfPumpConfigControlClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPumpConfigControlClusterServerMessageSentCallback() : **callback.h**
- emberAfPumpConfigControlClusterServerTickCallback() : **callback.h**
- EmberAfPumpControlMode : **enums.h**
- EmberAfPumpOperationMode : **enums.h**
- emberAfQSToNextEvent : **af.h**
- emberAfReadAttribute() : **af.h**
- emberAfReadAttributesResponseCallback() : **callback.h**
- emberAfReadClientAttribute() : **af.h**
- emberAfReadManufacturerSpecificClientAttribute() : **af.h**
- emberAfReadManufacturerSpecificServerAttribute() : **af.h**
- emberAfReadOrUpdateAttribute() : **af.h**
- emberAfReadReportingConfigurationResponseCallback() : **callback.h**
- emberAfReadServerAttribute() : **af.h**
- EmberAfRegisterTier : **enums.h**
- emberAfRegistrationAbortCallback() : **callback.h**
- emberAfRegistrationCallback() : **callback.h**
- emberAfRegistrationDebugExec : **debug-printing.h**
- emberAfRegistrationFlush : **debug-printing.h**
- emberAfRegistrationPrint : **debug-printing.h**
- emberAfRegistrationPrintBuffer : **debug-printing.h**
- emberAfRegistrationPrintln : **debug-printing.h**
- emberAfRegistrationPrintString : **debug-printing.h**
- emberAfRegistrationStartCallback() : **callback.h**
- EmberAfRelativeHumidityDisplay : **enums.h**
- emberAfRelativeHumidityMeasurementClusterClientAttributeChangedCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterClientDefaultResponseCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterClientInitCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterClientMessageSentCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterClientTickCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterDebugExec : **debug-printing.h**
- emberAfRelativeHumidityMeasurementClusterFlush : **debug-printing.h**
- emberAfRelativeHumidityMeasurementClusterPrint : **debug-printing.h**
- emberAfRelativeHumidityMeasurementClusterPrintBuffer : **debug-printing.h**
- emberAfRelativeHumidityMeasurementClusterPrintln : **debug-printing.h**
- emberAfRelativeHumidityMeasurementClusterPrintString : **debug-printing.h**
- emberAfRelativeHumidityMeasurementClusterServerAttributeChangedCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterServerDefaultResponseCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterServerInitCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterServerMessageSentCallback() : **callback.h**

- emberAfRelativeHumidityMeasurementClusterServerTickCallback() : **callback.h**
- EmberAfRelativeHumidityMode : **enums.h**
- emberAfRemoveFromCurrentAppTasks() : **af.h**
- emberAfReportAttributesCallback() : **callback.h**
- emberAfReportEntryConfiguredCallback() : **callback.h**
- emberAfReportingDebugExec : **debug-printing.h**
- EmberAfReportingDirection : **enums.h**
- emberAfReportingFlush : **debug-printing.h**
- emberAfReportingPrint : **debug-printing.h**
- emberAfReportingPrintBuffer : **debug-printing.h**
- emberAfReportingPrintln : **debug-printing.h**
- emberAfReportingPrintString : **debug-printing.h**
- emberAfResponseWaitingFlag : **af.h**
- emberAfRssiLocationClusterAnchorNodeAnnounceCallback() : **callback.h**
- emberAfRssiLocationClusterClientAttributeChangedCallback() : **callback.h**
- emberAfRssiLocationClusterClientCommandParse() : **call-command-handler.h**
- emberAfRssiLocationClusterClientDefaultResponseCallback() : **callback.h**
- emberAfRssiLocationClusterClientInitCallback() : **callback.h**
- emberAfRssiLocationClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfRssiLocationClusterClientMessageSentCallback() : **callback.h**
- emberAfRssiLocationClusterClientTickCallback() : **callback.h**
- emberAfRssiLocationClusterCompactLocationDataNotificationCallback() : **callback.h**
- emberAfRssiLocationClusterDebugExec : **debug-printing.h**
- emberAfRssiLocationClusterDeviceConfigurationResponseCallback() : **callback.h**
- emberAfRssiLocationClusterFlush : **debug-printing.h**
- emberAfRssiLocationClusterGetDeviceConfigurationCallback() : **callback.h**
- emberAfRssiLocationClusterGetLocationDataCallback() : **callback.h**
- emberAfRssiLocationClusterLocationDataNotificationCallback() : **callback.h**
- emberAfRssiLocationClusterLocationDataResponseCallback() : **callback.h**
- emberAfRssiLocationClusterPrint : **debug-printing.h**
- emberAfRssiLocationClusterPrintBuffer : **debug-printing.h**
- emberAfRssiLocationClusterPrintln : **debug-printing.h**
- emberAfRssiLocationClusterPrintString : **debug-printing.h**
- emberAfRssiLocationClusterReportRssiMeasurementsCallback() : **callback.h**
- emberAfRssiLocationClusterRequestOwnLocationCallback() : **callback.h**
- emberAfRssiLocationClusterRssiPingCallback() : **callback.h**
- emberAfRssiLocationClusterRssiRequestCallback() : **callback.h**
- emberAfRssiLocationClusterRssiResponseCallback() : **callback.h**
- emberAfRssiLocationClusterSendPingsCallback() : **callback.h**
- emberAfRssiLocationClusterServerAttributeChangedCallback() : **callback.h**
- emberAfRssiLocationClusterServerCommandParse() : **call-command-handler.h**
- emberAfRssiLocationClusterServerDefaultResponseCallback() : **callback.h**
- emberAfRssiLocationClusterServerInitCallback() : **callback.h**
- emberAfRssiLocationClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfRssiLocationClusterServerMessageSentCallback() : **callback.h**
- emberAfRssiLocationClusterServerTickCallback() : **callback.h**
- emberAfRssiLocationClusterSetAbsoluteLocationCallback() : **callback.h**
- emberAfRssiLocationClusterSetDeviceConfigurationCallback() : **callback.h**
- emberAfRunEvents() : **af.h**
- emberAfSampleMfgSpecificClusterClientAttributeChangedCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterClientDefaultResponseCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterClientInitCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterClientMessageSentCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterClientTickCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterCommandOneCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterDebugExec : **debug-printing.h**

- emberAfSampleMfgSpecificClusterFlush : **debug-printing.h**
- emberAfSampleMfgSpecificClusterPrint : **debug-printing.h**
- emberAfSampleMfgSpecificClusterPrintBuffer : **debug-printing.h**
- emberAfSampleMfgSpecificClusterPrintln : **debug-printing.h**
- emberAfSampleMfgSpecificClusterPrintString : **debug-printing.h**
- emberAfSampleMfgSpecificClusterServerAttributeChangedCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterServerCommandParse() : **call-command-handler.h**
- emberAfSampleMfgSpecificClusterServerDefaultResponseCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterServerInitCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterServerMessageSentCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterServerTickCallback() : **callback.h**
- EmberAfSaturationMoveMode : **enums.h**
- EmberAfSaturationStepMode : **enums.h**
- emberAfScanErrorCallback() : **callback.h**
- emberAfScenesClusterAddSceneCallback() : **callback.h**
- emberAfScenesClusterAddSceneResponseCallback() : **callback.h**
- emberAfScenesClusterClearSceneTableCallback() : **callback.h**
- emberAfScenesClusterClientAttributeChangedCallback() : **callback.h**
- emberAfScenesClusterClientCommandParse() : **call-command-handler.h**
- emberAfScenesClusterClientDefaultResponseCallback() : **callback.h**
- emberAfScenesClusterClientInitCallback() : **callback.h**
- emberAfScenesClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfScenesClusterClientMessageSentCallback() : **callback.h**
- emberAfScenesClusterClientTickCallback() : **callback.h**
- emberAfScenesClusterDebugExec : **debug-printing.h**
- emberAfScenesClusterFlush : **debug-printing.h**
- emberAfScenesClusterGetSceneMembershipCallback() : **callback.h**
- emberAfScenesClusterGetSceneMembershipResponseCallback() : **callback.h**
- emberAfScenesClusterPrint : **debug-printing.h**
- emberAfScenesClusterPrintBuffer : **debug-printing.h**
- emberAfScenesClusterPrintIn : **debug-printing.h**
- emberAfScenesClusterPrintString : **debug-printing.h**
- emberAfScenesClusterRecallSavedSceneCallback() : **callback.h**
- emberAfScenesClusterRecallSceneCallback() : **callback.h**
- emberAfScenesClusterRemoveAllScenesCallback() : **callback.h**
- emberAfScenesClusterRemoveAllScenesResponseCallback() : **callback.h**
- emberAfScenesClusterRemoveSceneCallback() : **callback.h**
- emberAfScenesClusterRemoveSceneResponseCallback() : **callback.h**
- emberAfScenesClusterServerAttributeChangedCallback() : **callback.h**
- emberAfScenesClusterServerCommandParse() : **call-command-handler.h**
- emberAfScenesClusterServerDefaultResponseCallback() : **callback.h**
- emberAfScenesClusterServerInitCallback() : **callback.h**
- emberAfScenesClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfScenesClusterServerMessageSentCallback() : **callback.h**
- emberAfScenesClusterServerTickCallback() : **callback.h**
- emberAfScenesClusterStoreCurrentSceneCallback() : **callback.h**
- emberAfScenesClusterStoreSceneCallback() : **callback.h**
- emberAfScenesClusterStoreSceneResponseCallback() : **callback.h**
- emberAfScenesClusterViewSceneCallback() : **callback.h**
- emberAfScenesClusterViewSceneResponseCallback() : **callback.h**
- emberAfScenesMakeInvalid() : **scenes.h**
- emberAfScenesMakeValid() : **scenes.h**
- emberAfScenesSetSceneCountAttribute() : **scenes.h**
- emberAfScheduleClusterTick() : **af.h**
- emberAfSecurityDebugExec : **debug-printing.h**
- emberAfSecurityFlush : **debug-printing.h**

- emberAfSecurityPrint : **debug-printing.h**
- emberAfSecurityPrintBuffer : **debug-printing.h**
- emberAfSecurityPrintln : **debug-printing.h**
- emberAfSecurityPrintString : **debug-printing.h**
- emberAfSendBroadcast() : **af.h**
- emberAfSendCommandBroadcast() : **af.h**
- emberAfSendCommandInterPan() : **af.h**
- emberAfSendCommandInterpanUnicast : **af.h**
- emberAfSendCommandMulticast() : **af.h**
- emberAfSendCommandUnicast() : **af.h**
- emberAfSendDefaultResponse() : **af.h**
- emberAfSendEndDeviceBind() : **af.h**
- emberAfSendInterPan() : **af.h**
- emberAfSendInterpanUnicast : **af.h**
- emberAfSendMulticast() : **af.h**
- emberAfSendRequest() : **af.h**
- emberAfSendResponse() : **af.h**
- emberAfSendUnicast() : **af.h**
- EmberAfSensingLightSensorType : **enums.h**
- EmberAfServiceDiscoveryCallback : **af-types.h**
- emberAfServiceDiscoveryDebugExec : **debug-printing.h**
- emberAfServiceDiscoveryFlush : **debug-printing.h**
- emberAfServiceDiscoveryPrint : **debug-printing.h**
- emberAfServiceDiscoveryPrintBuffer : **debug-printing.h**
- emberAfServiceDiscoveryPrintln : **debug-printing.h**
- emberAfServiceDiscoveryPrintString : **debug-printing.h**
- EmberAfServiceDiscoveryStatus : **af-types.h**
- emberAfSetAddressTableEntry() : **af.h**
- emberAfSetDefaultSleepControl() : **af.h**
- emberAfSetDeviceEnabled() : **af.h**
- emberAfSetExternalBuffer() : **client-api.h**
- EmberAfSetpointAdjustMode : **enums.h**
- emberAfSetResponseWaiting : **af.h**
- emberAfSetTime() : **af.h**
- emberAfSetTimeCallback() : **callback.h**
- emberAfShadeConfigClusterClientAttributeChangedCallback() : **callback.h**
- emberAfShadeConfigClusterClientDefaultResponseCallback() : **callback.h**
- emberAfShadeConfigClusterClientInitCallback() : **callback.h**
- emberAfShadeConfigClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfShadeConfigClusterClientMessageSentCallback() : **callback.h**
- emberAfShadeConfigClusterClientTickCallback() : **callback.h**
- emberAfShadeConfigClusterDebugExec : **debug-printing.h**
- emberAfShadeConfigClusterFlush : **debug-printing.h**
- emberAfShadeConfigClusterPrint : **debug-printing.h**
- emberAfShadeConfigClusterPrintBuffer : **debug-printing.h**
- emberAfShadeConfigClusterPrintln : **debug-printing.h**
- emberAfShadeConfigClusterPrintString : **debug-printing.h**
- emberAfShadeConfigClusterServerAttributeChangedCallback() : **callback.h**
- emberAfShadeConfigClusterServerDefaultResponseCallback() : **callback.h**
- emberAfShadeConfigClusterServerInitCallback() : **callback.h**
- emberAfShadeConfigClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfShadeConfigClusterServerMessageSentCallback() : **callback.h**
- emberAfShadeConfigClusterServerTickCallback() : **callback.h**
- EmberAfSignatureType : **enums.h**
- emberAfSimpleMeteringClusterClientAttributeChangedCallback() : **callback.h**
- emberAfSimpleMeteringClusterClientCommandParse() : **call-command-handler.h**
- emberAfSimpleMeteringClusterClientDefaultResponseCallback() : **callback.h**

- emberAfSimpleMeteringClusterClientInitCallback() : **callback.h**
- emberAfSimpleMeteringClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfSimpleMeteringClusterClientMessageSentCallback() : **callback.h**
- emberAfSimpleMeteringClusterClientTickCallback() : **callback.h**
- emberAfSimpleMeteringClusterDebugExec : **debug-printing.h**
- emberAfSimpleMeteringClusterFlush : **debug-printing.h**
- emberAfSimpleMeteringClusterGetProfileCallback() : **callback.h**
- emberAfSimpleMeteringClusterGetProfileResponseCallback() : **callback.h**
- emberAfSimpleMeteringClusterMirrorRemovedCallback() : **callback.h**
- emberAfSimpleMeteringClusterPrint : **debug-printing.h**
- emberAfSimpleMeteringClusterPrintBuffer : **debug-printing.h**
- emberAfSimpleMeteringClusterPrintln : **debug-printing.h**
- emberAfSimpleMeteringClusterPrintString : **debug-printing.h**
- emberAfSimpleMeteringClusterRemoveMirrorCallback() : **callback.h**
- emberAfSimpleMeteringClusterRequestFastPollModeCallback() : **callback.h**
- emberAfSimpleMeteringClusterRequestFastPollModeResponseCallback() : **callback.h**
- emberAfSimpleMeteringClusterRequestMirrorCallback() : **callback.h**
- emberAfSimpleMeteringClusterRequestMirrorResponseCallback() : **callback.h**
- emberAfSimpleMeteringClusterServerAttributeChangedCallback() : **callback.h**
- emberAfSimpleMeteringClusterServerCommandParse() : **call-command-handler.h**
- emberAfSimpleMeteringClusterServerDefaultResponseCallback() : **callback.h**
- emberAfSimpleMeteringClusterServerInitCallback() : **callback.h**
- emberAfSimpleMeteringClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfSimpleMeteringClusterServerMessageSentCallback() : **callback.h**
- emberAfSimpleMeteringClusterServerTickCallback() : **callback.h**
- EmberAfSquawkLevel : **enums.h**
- EmberAfSquawkMode : **enums.h**
- EmberAfSquawkStobe : **enums.h**
- emberAfStackStatusCallback() : **callback.h**
- emberAfStartMove() : **af.h**
- emberAfStartSearchForJoinableNetwork() : **af.h**
- emberAfStartSearchForJoinableNetworkCallback() : **callback.h**
- EmberAfStatus : **enums.h**
- EmberAfStepMode : **enums.h**
- emberAfStringLength() : **af.h**
- EmberAfSwitchActions : **enums.h**
- EmberAfSwitchType : **enums.h**
- EmberAfTemperatureDisplayMode : **enums.h**
- emberAfTempMeasurementClusterClientAttributeChangedCallback() : **callback.h**
- emberAfTempMeasurementClusterClientDefaultResponseCallback() : **callback.h**
- emberAfTempMeasurementClusterClientInitCallback() : **callback.h**
- emberAfTempMeasurementClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfTempMeasurementClusterClientMessageSentCallback() : **callback.h**
- emberAfTempMeasurementClusterClientTickCallback() : **callback.h**
- emberAfTempMeasurementClusterConvertVoltsToCelsiusCallback() : **callback.h**
- emberAfTempMeasurementClusterDebugExec : **debug-printing.h**
- emberAfTempMeasurementClusterFlush : **debug-printing.h**
- emberAfTempMeasurementClusterPrint : **debug-printing.h**
- emberAfTempMeasurementClusterPrintBuffer : **debug-printing.h**
- emberAfTempMeasurementClusterPrintln : **debug-printing.h**
- emberAfTempMeasurementClusterPrintString : **debug-printing.h**
- emberAfTempMeasurementClusterServerAttributeChangedCallback() : **callback.h**
- emberAfTempMeasurementClusterServerDefaultResponseCallback() : **callback.h**
- emberAfTempMeasurementClusterServerInitCallback() : **callback.h**
- emberAfTempMeasurementClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfTempMeasurementClusterServerMessageSentCallback() : **callback.h**
- emberAfTempMeasurementClusterServerTickCallback() : **callback.h**

- emberAfThermostatClusterClientAttributeChangedCallback() : **callback.h**
- emberAfThermostatClusterClientDefaultResponseCallback() : **callback.h**
- emberAfThermostatClusterClientInitCallback() : **callback.h**
- emberAfThermostatClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfThermostatClusterClientMessageSentCallback() : **callback.h**
- emberAfThermostatClusterClientTickCallback() : **callback.h**
- emberAfThermostatClusterDebugExec : **debug-printing.h**
- emberAfThermostatClusterFlush : **debug-printing.h**
- emberAfThermostatClusterPrint : **debug-printing.h**
- emberAfThermostatClusterPrintBuffer : **debug-printing.h**
- emberAfThermostatClusterPrintln : **debug-printing.h**
- emberAfThermostatClusterPrintString : **debug-printing.h**
- emberAfThermostatClusterServerAttributeChangedCallback() : **callback.h**
- emberAfThermostatClusterServerCommandParse() : **call-command-handler.h**
- emberAfThermostatClusterServerDefaultResponseCallback() : **callback.h**
- emberAfThermostatClusterServerInitCallback() : **callback.h**
- emberAfThermostatClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfThermostatClusterServerMessageSentCallback() : **callback.h**
- emberAfThermostatClusterServerTickCallback() : **callback.h**
- emberAfThermostatClusterSetpointRaiseLowerCallback() : **callback.h**
- EmberAfThermostatControlSequence : **enums.h**
- EmberAfThermostatSystemMode : **enums.h**
- emberAfThermostatUiConfigClusterClientAttributeChangedCallback() : **callback.h**
- emberAfThermostatUiConfigClusterClientDefaultResponseCallback() : **callback.h**
- emberAfThermostatUiConfigClusterClientInitCallback() : **callback.h**
- emberAfThermostatUiConfigClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfThermostatUiConfigClusterClientMessageSentCallback() : **callback.h**
- emberAfThermostatUiConfigClusterClientTickCallback() : **callback.h**
- emberAfThermostatUiConfigClusterDebugExec : **debug-printing.h**
- emberAfThermostatUiConfigClusterFlush : **debug-printing.h**
- emberAfThermostatUiConfigClusterPrint : **debug-printing.h**
- emberAfThermostatUiConfigClusterPrintBuffer : **debug-printing.h**
- emberAfThermostatUiConfigClusterPrintln : **debug-printing.h**
- emberAfThermostatUiConfigClusterPrintString : **debug-printing.h**
- emberAfThermostatUiConfigClusterServerAttributeChangedCallback() : **callback.h**
- emberAfThermostatUiConfigClusterServerDefaultResponseCallback() : **callback.h**
- emberAfThermostatUiConfigClusterServerInitCallback() : **callback.h**
- emberAfThermostatUiConfigClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfThermostatUiConfigClusterServerMessageSentCallback() : **callback.h**
- emberAfThermostatUiConfigClusterServerTickCallback() : **callback.h**
- EmberAfTickFunction : **af-types.h**
- emberAfTimeClusterClientAttributeChangedCallback() : **callback.h**
- emberAfTimeClusterClientDefaultResponseCallback() : **callback.h**
- emberAfTimeClusterClientInitCallback() : **callback.h**
- emberAfTimeClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfTimeClusterClientMessageSentCallback() : **callback.h**
- emberAfTimeClusterClientTickCallback() : **callback.h**
- emberAfTimeClusterDebugExec : **debug-printing.h**
- emberAfTimeClusterFlush : **debug-printing.h**
- emberAfTimeClusterPrint : **debug-printing.h**
- emberAfTimeClusterPrintBuffer : **debug-printing.h**
- emberAfTimeClusterPrintln : **debug-printing.h**
- emberAfTimeClusterPrintString : **debug-printing.h**
- emberAfTimeClusterServerAttributeChangedCallback() : **callback.h**
- emberAfTimeClusterServerDefaultResponseCallback() : **callback.h**
- emberAfTimeClusterServerInitCallback() : **callback.h**
- emberAfTimeClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**

- emberAfTimeClusterServerMessageSentCallback() : **callback.h**
- emberAfTimeClusterServerTickCallback() : **callback.h**
- emberAfTrustCenterJoinCallback() : **callback.h**
- emberAfTrustCenterKeepaliveAbortCallback() : **callback.h**
- emberAfTrustCenterKeepaliveUpdateCallback() : **callback.h**
- emberAfTunnelingClusterAckTransferDataClientToServerCallback() : **callback.h**
- emberAfTunnelingClusterAckTransferDataServerToClientCallback() : **callback.h**
- emberAfTunnelingClusterClientAttributeChangedCallback() : **callback.h**
- emberAfTunnelingClusterClientCommandParse() : **call-command-handler.h**
- emberAfTunnelingClusterClientDefaultResponseCallback() : **callback.h**
- emberAfTunnelingClusterClientInitCallback() : **callback.h**
- emberAfTunnelingClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfTunnelingClusterClientMessageSentCallback() : **callback.h**
- emberAfTunnelingClusterClientTickCallback() : **callback.h**
- emberAfTunnelingClusterCloseTunnelCallback() : **callback.h**
- emberAfTunnelingClusterDebugExec : **debug-printing.h**
- emberAfTunnelingClusterFlush : **debug-printing.h**
- emberAfTunnelingClusterPrint : **debug-printing.h**
- emberAfTunnelingClusterPrintBuffer : **debug-printing.h**
- emberAfTunnelingClusterPrintln : **debug-printing.h**
- emberAfTunnelingClusterPrintString : **debug-printing.h**
- emberAfTunnelingClusterReadyDataClientToServerCallback() : **callback.h**
- emberAfTunnelingClusterReadyDataServerToClientCallback() : **callback.h**
- emberAfTunnelingClusterRequestTunnelCallback() : **callback.h**
- emberAfTunnelingClusterRequestTunnelResponseCallback() : **callback.h**
- emberAfTunnelingClusterServerAttributeChangedCallback() : **callback.h**
- emberAfTunnelingClusterServerCommandParse() : **call-command-handler.h**
- emberAfTunnelingClusterServerDefaultResponseCallback() : **callback.h**
- emberAfTunnelingClusterServerInitCallback() : **callback.h**
- emberAfTunnelingClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfTunnelingClusterServerMessageSentCallback() : **callback.h**
- emberAfTunnelingClusterServerTickCallback() : **callback.h**
- emberAfTunnelingClusterTransferDataClientToServerCallback() : **callback.h**
- emberAfTunnelingClusterTransferDataErrorClientToServerCallback() : **callback.h**
- emberAfTunnelingClusterTransferDataErrorServerToClientCallback() : **callback.h**
- emberAfTunnelingClusterTransferDataServerToClientCallback() : **callback.h**
- EmberAfTunnelingProtocolId : **enums.h**
- EmberAfTunnelingTransferDataStatus : **enums.h**
- EmberAfTunnelingTunnelStatus : **enums.h**
- emberAfUnusedPanIdFoundCallback() : **callback.h**
- emberAfVerifyAttributeWrite() : **af.h**
- EmberAfWarningMode : **enums.h**
- EmberAfWarningStobe : **enums.h**
- emberAfWindowCoveringClusterClientAttributeChangedCallback() : **callback.h**
- emberAfWindowCoveringClusterClientDefaultResponseCallback() : **callback.h**
- emberAfWindowCoveringClusterClientInitCallback() : **callback.h**
- emberAfWindowCoveringClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfWindowCoveringClusterClientMessageSentCallback() : **callback.h**
- emberAfWindowCoveringClusterClientTickCallback() : **callback.h**
- emberAfWindowCoveringClusterDebugExec : **debug-printing.h**
- emberAfWindowCoveringClusterFlush : **debug-printing.h**
- emberAfWindowCoveringClusterPrint : **debug-printing.h**
- emberAfWindowCoveringClusterPrintBuffer : **debug-printing.h**
- emberAfWindowCoveringClusterPrintln : **debug-printing.h**
- emberAfWindowCoveringClusterPrintString : **debug-printing.h**
- emberAfWindowCoveringClusterServerAttributeChangedCallback() : **callback.h**
- emberAfWindowCoveringClusterServerCommandParse() : **call-command-handler.h**

- emberAfWindowCoveringClusterServerDefaultResponseCallback() : **callback.h**
- emberAfWindowCoveringClusterServerInitCallback() : **callback.h**
- emberAfWindowCoveringClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfWindowCoveringClusterServerMessageSentCallback() : **callback.h**
- emberAfWindowCoveringClusterServerTickCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringDownCloseCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringGoToLiftPercentageCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringGoToLiftSetpointCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringGoToLiftValueCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringGoToProgramSetpointCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringGoToTiltPercentageCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringGoToTiltSetpointCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringGoToTiltValueCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringStopCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringUpOpenCallback() : **callback.h**
- emberAfWriteAttribute() : **af.h**
- emberAfWriteAttributesResponseCallback() : **callback.h**
- emberAfWriteClientAttribute() : **af.h**
- emberAfWriteManufacturerSpecificClientAttribute() : **af.h**
- emberAfWriteManufacturerSpecificServerAttribute() : **af.h**
- emberAfWriteServerAttribute() : **af.h**
- emberAfZdoDebugExec : **debug-printing.h**
- emberAfZdoFlush : **debug-printing.h**
- emberAfZdoPrint : **debug-printing.h**
- emberAfZdoPrintBuffer : **debug-printing.h**
- emberAfZdoPrintIn : **debug-printing.h**
- emberAfZdoPrintString : **debug-printing.h**
- emPslChannelMask : **psl-commissioning.h**
- endpoints : **af.h**
- enroll() : **ias-zone-table.h**
- ENTRY_IS_CANCELLED_EVENT : **load-control-event-table.h**
- ENTRY_IS_SUPERSEDED_EVENT : **load-control-event-table.h**
- ENTRY_SCHEDULED : **load-control-event-table.h**
- ENTRY_STARTED : **load-control-event-table.h**
- ENTRY_VOID : **load-control-event-table.h**
- EphemeralData : **af-structs.h**
- EVENT_OPT_FLAG_OPT_IN : **load-control-event-table.h**
- EVENT_OPT_FLAG_PARTIAL : **load-control-event-table.h**
- EVENT_OPT_IN_DEFAULT : **demand-response-load-control.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- f -

- FIELD_CONTROL_OFFSET : **ota-storage.h**
- FILE_VERSION_WILD_CARD : **ota.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- g -

- GENERATE_SHARED_SECRET_DONE : **af-types.h**
- GENERATING_EPHEMERAL_KEYS : **af-types.h**
- GENERATING_SHARED_SECRET : **af-types.h**
- getNextZoneId() : **ias-zone-table.h**
- getSmacPointer() : **key-establishment-storage.h**
- getZoneIdMap() : **ias-zone-table.h**

- getZoneInfo() : **ias-zone-table.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- h -

- HARDWARE_VERSIONS_PRESENT_MASK : **ota-storage.h**
- HEADER_LENGTH_FIELD_LENGTH : **ota-storage.h**
- HEADER_LENGTH_OFFSET : **ota-storage.h**
- HEADER_STRING_OFFSET : **ota-storage.h**
- HEADER_VERSION_OFFSET : **ota-storage.h**
- headerHasHardwareVersions : **ota-storage.h**

- headerHasSecurityCredentials : **ota-storage.h**
- headerHasUpgradeFileDest : **ota-storage.h**
- HOURS_IN_MS : **ota-client-policy.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- i -

- Identity : **af-structs.h**
- IMAGE_SIZE_OFFSET : **ota-storage.h**
- IMAGE_TYPE_CONFIG : **ota.h**
- IMAGE_TYPE_ID_OFFSET : **ota-storage.h**
- IMAGE_TYPE_LOG : **ota.h**
- IMAGE_TYPE_SECURITY : **ota.h**

- IMAGE_TYPE_WILD_CARD : **ota.h**
- initZoneTable() : **ias-zone-table.h**
- INTER_PAN_BROADCAST : **af-types.h**
- INTER_PAN_MULTICAST : **af-types.h**
- INTER_PAN_UNICAST : **af-types.h**
- INTERPAN_APS_BROADCAST_SIZE : **interpan.h**
- INTERPAN_APS_ENCRYPTION_OVERHEAD : **interpan.h**
- INTERPAN_APS_FRAME_CONTROL_NO_DELIVERY_MODE : **interpan.h**
- INTERPAN_APS_FRAME_DELIVERY_MODE_MASK : **interpan.h**
- INTERPAN_APS_FRAME_SECURITY : **interpan.h**
- INTERPAN_APS_FRAME_TYPE : **interpan.h**
- INTERPAN_APS_FRAME_TYPE_MASK : **interpan.h**
- INTERPAN_APS_MULTICAST_SIZE : **interpan.h**
- INTERPAN_APS_UNICAST_SIZE : **interpan.h**
- interpanCommands : **zcl-cli.h**
- InterPanHeader : **af-types.h**
- INVALID_APP_COMMAND : **af-types.h**
- INVALID_DEVICE_ID : **ota-storage.h**
- INVALID_EUI64 : **ota-storage.h**
- INVALID_FIRMWARE_VERSION : **ota-storage.h**
- INVALID_MANUFACTURER_ID : **ota-storage.h**
- INVALID_OTA_IMAGE_ID : **ota-storage.h**
- INVALID_PARTNER_MESSAGE : **af-types.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- k -

- KEY_CONFIRM_FAILURE : **af-types.h**
- KEY_GENERATION_DONE : **af-types.h**
- KEY_TABLE_FULL : **af-types.h**
- KEY_UPDATE_COMMANDS : **security-cli.h**
- keysCommands : **zcl-cli.h**
- keyUpdateCommand() : **security-cli.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- l -

- LC_TICK_TIME : **level-control.h**
- LC_TICKS_PER_SECOND : **level-control.h**
- LEVEL_CONTROL_MAXIMUM_LEVEL : **level-control.h**
- LEVEL_CONTROL_MINIMUM_LEVEL : **level-control.h**
- LINK_KEY_ESTABLISHED : **af-types.h**
- LOAD_CONTROL_EVENT_TABLE_ENTRY_BYTES : **load-control-event-table.h**

- LOAD_CONTROL_EVENT_TABLE_MEM_FOOTPRINT : **load-control-event-table.h**
- LOG_LEVELS : **debug.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- m -

- MAGIC_NUMBER_OFFSET : **ota-storage.h**
- MANUFACTURER_ID_OFFSET : **ota-storage.h**
- MAX_CLUSTER_NAME_LENGTH : **print-cluster.h**
- MAX_DIGEST_CALCULATIONS_PER_CALL : **ota-client-signature-verify.h**
- MAX_INTER_PAN_HEADER_SIZE : **interpan.h**
- MAX_INTER_PAN_MAC_SIZE : **interpan.h**

- MAX_PROFILE_INDEX : **simple-metering-test.h**
- MAX_STUB_APS_SIZE : **interpan.h**
- MESSAGE_SEND_FAILURE : **af-types.h**
- MFG_ID_WILD_CARD : **ota.h**
- MILLISECOND_TICKS_PER_HOUR : **af.h**
- MILLISECOND_TICKS_PER_MINUTE : **af.h**
- MILLISECOND_TICKS_PER_QUARTERSECOND : **af.h**
- MIN_STUB_APS_SIZE : **interpan.h**
- MINUTES_IN_MS : **ota-client-policy.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- n -

- NeighborInfo : **af-structs.h**
- networkCommands : **network-cli.h**
- NO_APP_MESSAGE : **af-types.h**
- NO_ESTABLISHMENT_ALLOWED : **af-types.h**
- NO_LOCAL_RESOURCES : **af-types.h**
- NodeInformation : **af-structs.h**

- Notification : **af-structs.h**
- NULL_EUI64 : **ota-client.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- o -

- OPTIONAL_FIELDS_OFFSET : **ota-storage.h**
- optionCommands : **zcl-cli.h**
- OTA_CLIENT_COMMANDS : **ota-cli.h**
- OTA_COMMANDS : **ota-cli.h**
- OTA_FILE_MAGIC_NUMBER : **ota-storage.h**
- OTA_HEADER_VERSION : **ota-storage.h**

- OTA_HW_VERSION_BIT_MASK : **ota.h**
- OTA_MAXIMUM_HEADER_LENGTH : **ota-storage.h**
- OTA_MINIMUM_HEADER_LENGTH : **ota-storage.h**
- OTA_NODE_EUI64_BIT_MASK : **ota.h**
- OTA_SERVER_COMMANDS : **ota-cli.h**
- OTA_TAG_ECDSA_SIGNATURE : **ota.h**
- OTA_TAG_ECDSA_SIGNING_CERTIFICATE : **ota.h**
- OTA_TAG_MANUFACTURER_SPECIFIC_START : **ota.h**
- OTA_TAG_RESERVED_END : **ota.h**
- OTA_TAG_RESERVED_START : **ota.h**
- OTA_TAG_UPGRADE_IMAGE : **ota.h**
- OTA_UPGRADE_END_RESPONSE_RUN_NOW : **ota.h**
- OTA_UPGRADE_STATUS_COUNTDOWN : **ota.h**
- OTA_UPGRADE_STATUS_DOWNLOAD_COMPLETE : **ota.h**
- OTA_UPGRADE_STATUS_DOWNLOAD_IN_PROGRESS : **ota.h**
- OTA_UPGRADE_STATUS_NORMAL : **ota.h**
- OTA_UPGRADE_STATUS_WAIT : **ota.h**
- OTA_USE_OFFSET_TIME : **ota.h**
- OTA_USE_UTC_TIME : **ota.h**
- otaFindServerCommand() : **ota-cli.h**
- otaImageNotifyCommand() : **ota-cli.h**
- otaPrint : **ota.h**
- otaPrintln : **ota.h**
- otaQueryServerCommand() : **ota-cli.h**
- otaQuerySpecificFileCommand() : **ota-cli.h**
- otaSendUpgradeCommand() : **ota-cli.h**
- otaUsePageRequestCommand() : **ota-cli.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- p -

- pageRequestTest() : **ota-client-page-request-test.h**
- PARTNER_NO_RESOURCES : **af-types.h**
- PARTNER_SENT_TERMINATE : **af-types.h**
- printCommands : **zcl-cli.h**
- printKeyInfo() : **security-cli.h**
- printZoneIdMap() : **ias-zone-table.h**

- printZoneInfo() : **ias-zone-table.h**
- printZoneTable() : **ias-zone-table.h**
- PROFILE_INTERVAL_PERIOD_IN_MILLISECONDS : **simple-metering-test.h**
- PROFILE_INTERVAL_PERIOD_IN_MINUTES : **simple-metering-test.h**
- PROFILE_INTERVAL_PERIOD_IN_SECONDS : **simple-metering-test.h**
- PROFILE_INTERVAL_PERIOD_TIMEFRAME : **simple-metering-test.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- r -

- RANDOMIZE_END_TIME_FLAG : **load-control-event-table.h**
- RANDOMIZE_START_TIME_FLAG : **load-control-event-table.h**
- ReadAttributeStatusRecord : **af-structs.h**
- ReadReportingConfigurationAttributeRecord : **af-structs.h**
- ReadReportingConfigurationRecord : **af-structs.h**
- RECEIVED_PARTNER_CERTIFICATE : **af-types.h**

- remainingAllocations() : **ota-storage.h**
- ReportAttributeRecord : **af-structs.h**
- retrieveAndClearPublicPartnerData() : **key-establishment-storage.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- s -

- SceneExtensionAttributeInfo : **af-structs.h**
- SceneExtensionFieldSet : **af-structs.h**
- SECONDS_IN_MS : **ota-client-policy.h**
- SECURITY_CREDENTIAL_VERSION_FIELD_PRESENT_MASK : **ota-storage.h**
- Signature : **af-structs.h**
- SIGNATURE_TAG_DATA_SIZE : **ota-storage.h**

- Smac : **af-structs.h**
- STACK_VERSION_OFFSET : **ota-storage.h**
- STATIC_IMAGE_DATA : **ota-static-file-data.h**
- STATIC_IMAGE_DATA_SIZE : **ota-static-file-data.h**
- storePublicPartnerData() : **key-establishment-storage.h**
- storeSmac() : **key-establishment-storage.h**
- STUB_NWK_FRAME_CONTROL : **interpan.h**
- STUB_NWK_SIZE : **interpan.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- t -

- TAG_OVERHEAD : **ota-storage.h**
- TIMEOUT_OCCURRED : **af-types.h**
- TOTAL_PROFILE_TIME_SPAN_IN_SECONDS : **simple-metering-test.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- u -

- unenroll() : **ias-zone-table.h**
- upgradeServerNodeId : **ota.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- v -

- VERSION_OFFSET : **ota-storage.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- w -

- WriteAttributeRecord : **af-structs.h**
- WriteAttributeStatusRecord : **af-structs.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- g
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- w
- z

Here is a list of all functions, variables, defines, enums, and typedefs with links to the files they belong to:

- z -

- ZCL_ABS_MAX_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ABS_MAX_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ABS_MIN_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ABS_MIN_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ACCELERATION_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ACK_TRANSFER_DATA_CLIENT_TO_SERVER_COMMAND_ID : **command-id.h**
- ZCL_ACK_TRANSFER_DATA_SERVER_TO_CLIENT_COMMAND_ID : **command-id.h**
- ZCL_ACTIVE_TEXT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ACTUATOR_ENABLED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ADD_GROUP_COMMAND_ID : **command-id.h**
- ZCL_ADD_GROUP_IF_IDENTIFYING_COMMAND_ID : **command-id.h**

- ZCL_ADD_GROUP_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_ADD_SCENE_COMMAND_ID : **command-id.h**
- ZCL_ADD_SCENE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_ADVERTISE_PROTOCOL_ADDRESS_COMMAND_ID : **command-id.h**
- ZCL_ALARM_CLUSTER_ID : **cluster-id.h**
- ZCL_ALARM_COMMAND_ID : **command-id.h**
- ZCL_ALARM_COUNT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ANCHOR_NODE_ANNOUNCE_COMMAND_ID : **command-id.h**
- ZCL_APPLICATION_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_APPLICATION_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ARM_COMMAND_ID : **command-id.h**
- ZCL_ARM_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_ARRAY_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_ATTRIBUTE_ID_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_ATTRIBUTE_ONE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BACNET_OID_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BACNET_PROTOCOL_TUNNEL_CLUSTER_ID : **cluster-id.h**
- ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_EXECUTE : **debug.h**
- ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_ON : **debug.h**
- ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_PRINT : **debug.h**
- ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_WAITSEND : **debug.h**
- ZCL_BAG_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BALLAST_CONFIGURATION_CLUSTER_ID : **cluster-id.h**
- ZCL_BALLAST_FACTOR_ADJUSTMENT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BALLAST_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BASIC_CLUSTER_ID : **cluster-id.h**
- ZCL_BASIC_DEBUG_EXECUTE : **debug.h**
- ZCL_BASIC_DEBUG_ON : **debug.h**
- ZCL_BASIC_DEBUG_PRINT : **debug.h**
- ZCL_BASIC_DEBUG_PRINT_LINE : **debug.h**
- ZCL_BASIC_DEBUG_WAITSEND : **debug.h**
- ZCL_BATTERY_AHR_RATING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BATTERY_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BATTERY_MANUFACTURER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BATTERY_QUANTITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BATTERY_RATED_VOLTAGE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BATTERY_SIZE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BATTERY_VOLTAGE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BATTERY_VOLTAGE_MIN_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BINARY_INPUT_BASIC_CLUSTER_ID : **cluster-id.h**
- ZCL_BINARY_INPUT_BASIC_DEBUG_EXECUTE : **debug.h**
- ZCL_BINARY_INPUT_BASIC_DEBUG_ON : **debug.h**
- ZCL_BINARY_INPUT_BASIC_DEBUG_PRINT : **debug.h**
- ZCL_BINARY_INPUT_BASIC_DEBUG_PRINT_LINE : **debug.h**
- ZCL_BINARY_INPUT_BASIC_DEBUG_WAITSEND : **debug.h**
- ZCL_BITMAP16_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_BITMAP24_ATTRIBUTE_TYPE : **attribute-type.h**

- ZCL_BITMAP32_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_BITMAP40_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BITMAP48_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BITMAP56_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BITMAP64_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BITMAP8_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_BLOCK10_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK11_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK12_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK13_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK14_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK15_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK1_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK2_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK3_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK4_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK5_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK6_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK7_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK8_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK9_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK_PERIOD_DURATION_MINUTES_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BOOLEAN_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_BYPASS_COMMAND_ID : **command-id.h**
- ZCL_CALCULATION_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CANCEL_ALL_LOAD_CONTROL_EVENTS_COMMAND_ID : **command-id.h**
- ZCL_CANCEL_LOAD_CONTROL_EVENT_COMMAND_ID : **command-id.h**
- ZCL_CANCEL_MESSAGE_COMMAND_ID : **command-id.h**
- ZCL_CAPACITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CHANGE_SUPPLY_COMMAND_ID : **command-id.h**
- ZCL_CHANNEL_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CHAR_STRING_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_CLOSE_TUNNEL_COMMAND_ID : **command-id.h**
- ZCL_CLOSE_TUNNEL_TIMEOUT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CLOSED_LIMIT_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CLOSED_LIMIT_TILT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CLUSTER_ID_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_CLUSTER_SPECIFIC_COMMAND : **af.h**
- ZCL_COLOR_CONTROL_CLUSTER_ID : **cluster-id.h**
- ZCL_COLOR_CONTROL_COLOR_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_B_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_B_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_B_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_G_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_G_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_G_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_R_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_R_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_R_Y_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_COLOR_CONTROL_COLOR_TEMPERATURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COMPENSATION_TEXT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_CURRENT_HUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_CURRENT_SATURATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_CURRENT_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_CURRENT_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_DEBUG_EXECUTE : **debug.h**
- ZCL_COLOR_CONTROL_DEBUG_ON : **debug.h**
- ZCL_COLOR_CONTROL_DEBUG_PRINT : **debug.h**
- ZCL_COLOR_CONTROL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_COLOR_CONTROL_DEBUG_WAITSEND : **debug.h**
- ZCL_COLOR_CONTROL_DRIFT_COMPENSATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_HUE_STEP_UNIT : **color-control.h**
- ZCL_COLOR_CONTROL_NUMBER_OF_PRIMARIES_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_1_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_1_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_1_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_2_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_2_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_2_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_3_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_3_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_3_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_4_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_4_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_4_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_5_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_5_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_5_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_6_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_6_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_6_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_REMAINING_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_SATURATION_STEP_UNIT : **color-control.h**
- ZCL_COLOR_CONTROL_WHITE_POINT_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_WHITE_POINT_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COMMAND_ONE_COMMAND_ID : **command-id.h**
- ZCL_COMMISSIONING_CLUSTER_ID : **cluster-id.h**
- ZCL_COMMISSIONING_DEBUG_EXECUTE : **debug.h**
- ZCL_COMMISSIONING_DEBUG_ON : **debug.h**
- ZCL_COMMISSIONING_DEBUG_PRINT : **debug.h**
- ZCL_COMMISSIONING_DEBUG_PRINT_LINE : **debug.h**
- ZCL_COMMISSIONING_DEBUG_WAITSEND : **debug.h**
- ZCL_COMMODITY_TYPE_CLIENT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COMMODITY_TYPE_SERVER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COMPACT_LOCATION_DATA_NOTIFICATION_COMMAND_ID : **command-id.h**
- ZCL_CONCENTRATOR_DISCOVERY_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CONCENTRATOR_FLAG_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CONCENTRATOR_RADIUS_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_CONFIG_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CONFIGURE_REPORTING_COMMAND_ID : **command-id.h**
- ZCL_CONFIGURE_REPORTING_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_CONFIRM_KEY_DATA_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_CONFIRM_KEY_DATA_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_CONTROL_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CONTROL_SEQUENCE_OF_OPERATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CONTROL_TEMPERATURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COORDINATE1_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COORDINATE2_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COORDINATE3_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COVERING_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CREDIT_REMAINING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CREDIT_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_BLOCK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_BLOCK_PERIOD_CONSUMPTION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_DAY_CONSUMPTION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_DAY_CONSUMPTION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_DAY_MAX_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_DAY_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_DAY_MAX_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_DAY_MIN_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_DEMAND_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_FILE_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_GROUP_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_INLET_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_INLET_ENERGY_CARRIER_SUMMATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_LIFT_PERCENTAGE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_MAX_DEMAND_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_MAX_DEMAND_DELIVERED_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_MAX_DEMAND_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_MAX_DEMAND_RECEIVED_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_MONTH_MAX_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_MONTH_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_MONTH_MIN_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_CURRENT_NO_TIER_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_OUTLET_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_OUTLET_ENERGY_CARRIER_SUMMATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_START_TIME_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_START_TIME_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_VALUE_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_VALUE_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_SCENE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TEMPERATURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_CURRENT_TIER11_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_CURRENT_TIER14_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_CURRENT_TIER2_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_CURRENT_TIER4_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_CURRENT_TIER7_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TILT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TILT_PERCENTAGE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_YEAR_MAX_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_YEAR_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_YEAR_MIN_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_ZIGBEE_STACK_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DAILY_CONSUMPTION_TARGET_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_DAILY_FREEZE_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DATA16_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATA24_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATA32_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATA40_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATA48_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATA56_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATA64_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATA8_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_DATE_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATE_CODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DECELERATION_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEFAULT_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_DEFAULT_UPDATE_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEHUMID_CONTROL_CLUSTER_ID : **cluster-id.h**
- ZCL_DEHUMID_CONTROL_DEBUG_EXECUTE : **debug.h**
- ZCL_DEHUMID_CONTROL_DEBUG_ON : **debug.h**
- ZCL_DEHUMID_CONTROL_DEBUG_PRINT : **debug.h**
- ZCL_DEHUMID_CONTROL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_DEHUMID_CONTROL_DEBUG_WAITSEND : **debug.h**
- ZCL_DEHUMIDIFICATION_COOLING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEHUMIDIFICATION_HYSTERESIS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEHUMIDIFICATION_LOCKOUT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEHUMIDIFICATION_MAX_COOL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DELAYED_SUPPLY_INTERRUPT_VALUE_REMAINING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DELAYED_SUPPLY_INTERRUPT_VALUE_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEMAND_FORMATTING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEMAND_INTEGRATION_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEMAND_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID : **cluster-id.h**
- ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_EXECUTE : **debug.h**
- ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_ON : **debug.h**
- ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_PRINT : **debug.h**
- ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_WAITSEND : **debug.h**
- ZCL_DESCRIPTION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEVICE_CLASS_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEVICE_CONFIGURATION_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_DEVICE_ENABLED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEVICE_TEMP_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEVICE_TEMP_CLUSTER_ID : **cluster-id.h**
- ZCL_DEVICE_TEMP_DEBUG_EXECUTE : **debug.h**
- ZCL_DEVICE_TEMP_DEBUG_ON : **debug.h**
- ZCL_DEVICE_TEMP_DEBUG_PRINT : **debug.h**
- ZCL_DEVICE_TEMP_DEBUG_PRINT_LINE : **debug.h**
- ZCL_DEVICE_TEMP_DEBUG_WAITSEND : **debug.h**
- ZCL_DFT_SUMMATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DISABLE_DEFAULT_RESPONSE_MASK : **af.h**
- ZCL_DISABLE_LOCAL_CONFIG_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_DISCOVER_ATTRIBUTES_COMMAND_ID : **command-id.h**
- ZCL_DISCOVER_ATTRIBUTES_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_DISPLAY_MESSAGE_COMMAND_ID : **command-id.h**
- ZCL_DIVISOR_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DOOR_CLOSED_EVENTS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DOOR_LOCK_CLUSTER_ID : **cluster-id.h**
- ZCL_DOOR_LOCK_DEBUG_EXECUTE : **debug.h**
- ZCL_DOOR_LOCK_DEBUG_ON : **debug.h**
- ZCL_DOOR_LOCK_DEBUG_PRINT : **debug.h**
- ZCL_DOOR_LOCK_DEBUG_PRINT_LINE : **debug.h**
- ZCL_DOOR_LOCK_DEBUG_WAITSEND : **debug.h**
- ZCL_DOOR_OPEN_EVENTS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DOOR_STATE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DOWNLOADED_FILE_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DOWNLOADED_ZIGBEE_STACK_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DST_END_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DST_SHIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DST_START_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_EFFECTIVE_CONTROL_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_EFFECTIVE_OPERATION_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ELECTRICITY_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_EMERGENCY_COMMAND_ID : **command-id.h**
- ZCL_EMERGENCY_CREDIT_REMAINING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ENERGY_CARRIER_DEMAND_FORMATTING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ENERGY_CARRIER_SUMMATION_FORMATTING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ENERGY_CARRIER_UNIT_OF_MEASURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ENUM16_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_ENUM8_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_EPHEMERAL_DATA_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_EPHEMERAL_DATA_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_EXTENDED_PAN_ID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FAN_CONTROL_CLUSTER_ID : **cluster-id.h**
- ZCL_FAN_CONTROL_DEBUG_EXECUTE : **debug.h**
- ZCL_FAN_CONTROL_DEBUG_ON : **debug.h**
- ZCL_FAN_CONTROL_DEBUG_PRINT : **debug.h**
- ZCL_FAN_CONTROL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_FAN_CONTROL_DEBUG_WAITSEND : **debug.h**
- ZCL_FAN_CONTROL_FAN_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FAN_CONTROL_FAN_MODE_SEQUENCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FAST_POLL_UPDATE_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FILE_OFFSET_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FIRE_COMMAND_ID : **command-id.h**
- ZCL_FLOAT_DOUBLE_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_FLOAT_SEMI_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_FLOAT_SINGLE_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_FLOW_MAX_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FLOW_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FLOW_MEASUREMENT_CLUSTER_ID : **cluster-id.h**
- ZCL_FLOW_MEASUREMENT_DEBUG_EXECUTE : **debug.h**

- ZCL_FLOW_MEASUREMENT_DEBUG_ON : **debug.h**
- ZCL_FLOW_MEASUREMENT_DEBUG_PRINT : **debug.h**
- ZCL_FLOW_MEASUREMENT_DEBUG_PRINT_LINE : **debug.h**
- ZCL_FLOW_MEASUREMENT_DEBUG_WAITSEND : **debug.h**
- ZCL_FLOW_MIN_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FLOW_RESTRICTION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FLOW_TOLERANCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FRAME_CONTROL_CLIENT_TO_SERVER : **af.h**
- ZCL_FRAME_CONTROL_DIRECTION_MASK : **af.h**
- ZCL_FRAME_CONTROL_FRAME_TYPE_MASK : **af.h**
- ZCL_FRAME_CONTROL_SERVER_TO_CLIENT : **af.h**
- ZCL_FUEL_DEBT_RECOVERY_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FUEL_DEBT_RECOVERY_RATE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FUEL_DEBT_REMAINING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_GAS_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_GENERIC_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_GENERIC_FLOW_PRESSURE_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_GENERIC_TUNNEL_CLUSTER_ID : **cluster-id.h**
- ZCL_GENERIC_TUNNEL_DEBUG_EXECUTE : **debug.h**
- ZCL_GENERIC_TUNNEL_DEBUG_ON : **debug.h**
- ZCL_GENERIC_TUNNEL_DEBUG_PRINT : **debug.h**
- ZCL_GENERIC_TUNNEL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_GENERIC_TUNNEL_DEBUG_WAITSEND : **debug.h**
- ZCL_GET_ALARM_COMMAND_ID : **command-id.h**
- ZCL_GET_ALARM_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_GET_BLOCK_PERIODS_COMMAND_ID : **command-id.h**
- ZCL_GET_CURRENT_PRICE_COMMAND_ID : **command-id.h**
- ZCL_GET_DEVICE_CONFIGURATION_COMMAND_ID : **command-id.h**
- ZCL_GET_GROUP_MEMBERSHIP_COMMAND_ID : **command-id.h**
- ZCL_GET_GROUP_MEMBERSHIP_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_GET_LAST_MESSAGE_COMMAND_ID : **command-id.h**
- ZCL_GET_LOCATION_DATA_COMMAND_ID : **command-id.h**
- ZCL_GET_PROFILE_COMMAND_ID : **command-id.h**
- ZCL_GET_PROFILE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_GET_SCENE_MEMBERSHIP_COMMAND_ID : **command-id.h**
- ZCL_GET_SCENE_MEMBERSHIP_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_GET_SCHEDULED_EVENTS_COMMAND_ID : **command-id.h**
- ZCL_GET_SCHEDULED_PRICES_COMMAND_ID : **command-id.h**
- ZCL_GET_ZONE_ID_MAP_COMMAND_ID : **command-id.h**
- ZCL_GET_ZONE_ID_MAP_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_GET_ZONE_INFORMATION_COMMAND_ID : **command-id.h**
- ZCL_GET_ZONE_INFORMATION_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_GROUP_NAME_SUPPORT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_GROUPS_CLUSTER_ID : **cluster-id.h**
- ZCL_GROUPS_DEBUG_EXECUTE : **debug.h**
- ZCL_GROUPS_DEBUG_ON : **debug.h**
- ZCL_GROUPS_DEBUG_PRINT : **debug.h**
- ZCL_GROUPS_DEBUG_PRINT_LINE : **debug.h**
- ZCL_GROUPS_DEBUG_WAITSEND : **debug.h**

- ZCL_HEAT_AND_COOLING_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_HIGH_TEMP_DWELL_TRIP_POINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_HIGH_TEMP_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_HISTORICAL_CONSUMPTION_FORMATTING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_HOURS_IN_FAULT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_HOURS_IN_OPERATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_HW_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_IAS_ACE_CLUSTER_ID : **cluster-id.h**
- ZCL_IAS_ACE_DEBUG_EXECUTE : **debug.h**
- ZCL_IAS_ACE_DEBUG_ON : **debug.h**
- ZCL_IAS_ACE_DEBUG_PRINT : **debug.h**
- ZCL_IAS_ACE_DEBUG_PRINT_LINE : **debug.h**
- ZCL_IAS_ACE_DEBUG_WAITSEND : **debug.h**
- ZCL_IAS_CIE_ADDRESS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_IAS_WD_CLUSTER_ID : **cluster-id.h**
- ZCL_IAS_WD_DEBUG_EXECUTE : **debug.h**
- ZCL_IAS_WD_DEBUG_ON : **debug.h**
- ZCL_IAS_WD_DEBUG_PRINT : **debug.h**
- ZCL_IAS_WD_DEBUG_PRINT_LINE : **debug.h**
- ZCL_IAS_WD_DEBUG_WAITSEND : **debug.h**
- ZCL_IAS_ZONE_CLUSTER_ID : **cluster-id.h**
- ZCL_IAS_ZONE_DEBUG_EXECUTE : **debug.h**
- ZCL_IAS_ZONE_DEBUG_ON : **debug.h**
- ZCL_IAS_ZONE_DEBUG_PRINT : **debug.h**
- ZCL_IAS_ZONE_DEBUG_PRINT_LINE : **debug.h**
- ZCL_IAS_ZONE_DEBUG_WAITSEND : **debug.h**
- ZCL_IDENTIFY_CLUSTER_ID : **cluster-id.h**
- ZCL_IDENTIFY_COMMAND_ID : **command-id.h**
- ZCL_IDENTIFY_DEBUG_EXECUTE : **debug.h**
- ZCL_IDENTIFY_DEBUG_ON : **debug.h**
- ZCL_IDENTIFY_DEBUG_PRINT : **debug.h**
- ZCL_IDENTIFY_DEBUG_PRINT_LINE : **debug.h**
- ZCL_IDENTIFY_DEBUG_WAITSEND : **debug.h**
- ZCL_IDENTIFY_QUERY_COMMAND_ID : **command-id.h**
- ZCL_IDENTIFY_QUERY_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_IDENTIFY_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_IEEE_ADDRESS_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_ILLUM_LEVEL_SENSING_CLUSTER_ID : **cluster-id.h**
- ZCL_ILLUM_LEVEL_SENSING_DEBUG_EXECUTE : **debug.h**
- ZCL_ILLUM_LEVEL_SENSING_DEBUG_ON : **debug.h**
- ZCL_ILLUM_LEVEL_SENSING_DEBUG_PRINT : **debug.h**
- ZCL_ILLUM_LEVEL_SENSING_DEBUG_PRINT_LINE : **debug.h**
- ZCL_ILLUM_LEVEL_SENSING_DEBUG_WAITSEND : **debug.h**
- ZCL_ILLUM_MAX_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ILLUM_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ILLUM_MEASUREMENT_CLUSTER_ID : **cluster-id.h**
- ZCL_ILLUM_MEASUREMENT_DEBUG_EXECUTE : **debug.h**
- ZCL_ILLUM_MEASUREMENT_DEBUG_ON : **debug.h**
- ZCL_ILLUM_MEASUREMENT_DEBUG_PRINT : **debug.h**

- ZCL_ILLUM_MEASUREMENT_DEBUG_PRINT_LINE : **debug.h**
- ZCL_ILLUM_MEASUREMENT_DEBUG_WAITSEND : **debug.h**
- ZCL_ILLUM_MIN_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ILLUM_TOLERANCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ILLUMINANCE_TARGET_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_IMAGE_BLOCK_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_IMAGE_BLOCK_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_IMAGE_NOTIFY_COMMAND_ID : **command-id.h**
- ZCL_IMAGE_PAGE_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_IMAGE_TYPE_ID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_IMAGE_UPGRADE_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_INACTIVE_TEXT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_INDERECT_POLL_RATE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_INITIATE_KEY_ESTABLISHMENT_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_INITIATE_KEY_ESTABLISHMENT_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_INLET_TEMPERATURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_INSTANTANEOUS_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_INT16S_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_INT16U_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_INT24S_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_INT24U_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_INT32S_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_INT32U_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_INT40S_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT40U_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT48S_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT48U_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_INT56S_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT56U_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT64S_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT64U_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT8S_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_INT8U_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_INTERVAL_READ_REPORTING_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_INTRINSIC_BALLAST_FACTOR_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_INVALID_COLOR_CONTROL_COMMAND_ID : **color-control.h**
- ZCL_KEY_ESTABLISHMENT_CLUSTER_ID : **cluster-id.h**
- ZCL_KEY_ESTABLISHMENT_DEBUG_EXECUTE : **debug.h**
- ZCL_KEY_ESTABLISHMENT_DEBUG_ON : **debug.h**
- ZCL_KEY_ESTABLISHMENT_DEBUG_PRINT : **debug.h**
- ZCL_KEY_ESTABLISHMENT_DEBUG_PRINT_LINE : **debug.h**
- ZCL_KEY_ESTABLISHMENT_DEBUG_WAITSEND : **debug.h**
- ZCL_KEY_ESTABLISHMENT_SUITE_CLIENT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_KEY_ESTABLISHMENT_SUITE_SERVER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_KEYPAD_LOCKOUT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAMP_ALARM_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAMP_BURN_HOURS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAMP_BURN_HOURS_TRIP_POINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAMP_MANUFACTURER_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_LAMP_QUALITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAMP_RATED_HOURS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAMP_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAST_CONFIGURED_BY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAST_SET_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LEVEL_CONTROL_CLUSTER_ID : **cluster-id.h**
- ZCL_LEVEL_CONTROL_DEBUG_EXECUTE : **debug.h**
- ZCL_LEVEL_CONTROL_DEBUG_ON : **debug.h**
- ZCL_LEVEL_CONTROL_DEBUG_PRINT : **debug.h**
- ZCL_LEVEL_CONTROL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_LEVEL_CONTROL_DEBUG_WAITSEND : **debug.h**
- ZCL_LEVEL_CONTROL_REMAINING_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LEVEL_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LIFETIME_ENERGY_CONSUMED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LIFETIME_RUNNING_HOURS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LIMIT_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LIMIT_TILT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOAD_CONTROL_EVENT_COMMAND_ID : **command-id.h**
- ZCL_LOCAL_TEMPERATURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCAL_TEMPERATURE_CALIBRATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCAL_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCATION_AGE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCATION_DATA_NOTIFICATION_COMMAND_ID : **command-id.h**
- ZCL_LOCATION_DATA_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_LOCATION_DESCRIPTION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCATION_METHOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCATION_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCK_DOOR_COMMAND_ID : **command-id.h**
- ZCL_LOCK_DOOR_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_LOCK_STATE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCK_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LONG_CHAR_STRING_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_LONG_OCTET_STRING_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_LOW_TEMP_DWELL_TRIP_POINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOW_TEMP_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAINS_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAINS_FREQUENCY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAINS_VOLTAGE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAINS_VOLTAGE_DWELL_TRIP_POINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAINS_VOLTAGE_MAX_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAINS_VOLTAGE_MIN_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MANUFACTURER_ID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MANUFACTURER_NAME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MANUFACTURER_SPECIFIC_MASK : **af.h**
- ZCL_MATCH_PROTOCOL_ADDRESS_COMMAND_ID : **command-id.h**
- ZCL_MATCH_PROTOCOL_ADDRESS_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_MAX_COMP_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_CONST_FLOW_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_CONST_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_MAX_CONST_SPEED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_CONST_TEMP_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_DURATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_FLOW_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_NUMBER_OF_PERIODS_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_REJOIN_INTERVAL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_SPEED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_TEMP_EXPERIENCED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAXIMUM_INCOMING_TRANSFER_SIZE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAXIMUM_OUTGOING_TRANSFER_SIZE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MESSAGE_CONFIRMATION_COMMAND_ID : **command-id.h**
- ZCL_MESSAGING_CLUSTER_CONFIRMATION_MASK : **messaging-client.h** , **messaging-server.h**
- ZCL_MESSAGING_CLUSTER_DURATION_UNTIL_CHANGED : **messaging-client.h** , **messaging-server.h**
- ZCL_MESSAGING_CLUSTER_END_TIME_NEVER : **messaging-client.h** , **messaging-server.h**
- ZCL_MESSAGING_CLUSTER_ID : **cluster-id.h**
- ZCL_MESSAGING_CLUSTER_IMPORTANCE_MASK : **messaging-client.h** , **messaging-server.h**
- ZCL_MESSAGING_CLUSTER_START_TIME_NOW : **messaging-client.h** , **messaging-server.h**
- ZCL_MESSAGING_CLUSTER_TRANSMISSION_MASK : **messaging-server.h** , **messaging-client.h**
- ZCL_MESSAGING_DEBUG_EXECUTE : **debug.h**
- ZCL_MESSAGING_DEBUG_ON : **debug.h**
- ZCL_MESSAGING_DEBUG_PRINT : **debug.h**
- ZCL_MESSAGING_DEBUG_PRINT_LINE : **debug.h**
- ZCL_MESSAGING_DEBUG_WAITSEND : **debug.h**
- ZCL_METER_SERIAL_NUMBER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_METERING_DEVICE_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_COMP_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_CONST_FLOW_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_CONST_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_CONST_SPEED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_CONST_TEMP_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_SETPOINT_DEAD_BAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_TEMP_EXPERIENCED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIRROR_IEEE_ADDRESS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIRROR_LAST_SYNC_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIRROR_REMOVED_COMMAND_ID : **command-id.h**
- ZCL_MIRROR_SYNC_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MODEL_IDENTIFIER_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_MOVE_COLOR_COMMAND_ID : **command-id.h**
- ZCL_MOVE_COMMAND_ID : **command-id.h**
- ZCL_MOVE_HUE_COMMAND_ID : **command-id.h**
- ZCL_MOVE_SATURATION_COMMAND_ID : **command-id.h**
- ZCL_MOVE_TO_COLOR_COMMAND_ID : **command-id.h**
- ZCL_MOVE_TO_COLOR_TEMPERATURE_COMMAND_ID : **command-id.h**
- ZCL_MOVE_TO_HUE_AND_SATURATION_COMMAND_ID : **command-id.h**
- ZCL_MOVE_TO_HUE_COMMAND_ID : **command-id.h**
- ZCL_MOVE_TO_LEVEL_COMMAND_ID : **command-id.h**
- ZCL_MOVE_TO_LEVEL_WITH_ON_OFF_COMMAND_ID : **command-id.h**
- ZCL_MOVE_TO_SATURATION_COMMAND_ID : **command-id.h**
- ZCL_MOVE_WITH_ON_OFF_COMMAND_ID : **command-id.h**
- ZCL_MULTIPLIER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NETWORK_KEY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NETWORK_KEY_SEQUENCE_NUMBER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NETWORK_KEY_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NETWORK_MANAGER_ADDRESS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_DATA_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_NO_TIER_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NON_FUEL_DEBT_RECOVERY_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NON_FUEL_DEBT_RECOVERY_RATE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NON_FUEL_DEBT_REMAINING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NUMBER_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NUMBER_OF_DEMAND_SUBINTERVALS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NUMBER_OF_DEVICES_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NUMBER_RSSI_MEASUREMENTS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NUMBER_TILT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OCCUPANCY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OCCUPANCY_SENSING_CLUSTER_ID : **cluster-id.h**
- ZCL_OCCUPANCY_SENSING_DEBUG_EXECUTE : **debug.h**
- ZCL_OCCUPANCY_SENSING_DEBUG_ON : **debug.h**
- ZCL_OCCUPANCY_SENSING_DEBUG_PRINT : **debug.h**
- ZCL_OCCUPANCY_SENSING_DEBUG_PRINT_LINE : **debug.h**
- ZCL_OCCUPANCY_SENSING_DEBUG_WAITSEND : **debug.h**

- ZCL_OCCUPANCY_SENSOR_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OCCUPIED_COOLING_SETPOINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OCCUPIED_HEATING_SETPOINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OCTET_STRING_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_OFF_COMMAND_ID : **command-id.h**
- ZCL_ON_COMMAND_ID : **command-id.h**
- ZCL_ON_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ON_OFF_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ON_OFF_CLUSTER_ID : **cluster-id.h**
- ZCL_ON_OFF_DEBUG_EXECUTE : **debug.h**
- ZCL_ON_OFF_DEBUG_ON : **debug.h**
- ZCL_ON_OFF_DEBUG_PRINT : **debug.h**
- ZCL_ON_OFF_DEBUG_PRINT_LINE : **debug.h**
- ZCL_ON_OFF_DEBUG_WAITSEND : **debug.h**
- ZCL_ON_OFF_SWITCH_CONFIG_CLUSTER_ID : **cluster-id.h**
- ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_EXECUTE : **debug.h**
- ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_ON : **debug.h**
- ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_PRINT : **debug.h**
- ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_PRINT_LINE : **debug.h**
- ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_WAITSEND : **debug.h**
- ZCL_ON_OFF_TRANSITION_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OPEN_LIMIT_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OPEN_LIMIT_TILT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OPEN_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OPERATION_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ORIGINATING_DEVICE_1_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ORIGINATING_DEVICE_2_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ORIGINATING_DEVICE_3_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ORIGINATING_DEVICE_4_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ORIGINATING_DEVICE_5_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OTA_BOOTLOAD_CLUSTER_ID : **cluster-id.h**
- ZCL_OUT_OF_SERVICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OUTDOOR_TEMPERATURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OUTLET_TEMPERATURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OVER_TEMP_TOTAL_DWELL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PAN_ID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PANIC_COMMAND_ID : **command-id.h**
- ZCL_PARENT_RETRY_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PATH_LOSS_EXPONENT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PAYMENT_CONTROL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PHYSICAL_ENVIRONMENT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PHYSICAL_MAX_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PHYSICAL_MIN_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PI_COOLING_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PI_HEATING_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PIR_OCCUPIED_TO_UNOCCUPIED_DELAY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PIR_UNOCCUPIED_TO_OCCUPIED_DELAY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PIR_UNOCCUPIED_TO_OCCUPIED_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_POLARITY_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_POWER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_POWER_CONFIG_CLUSTER_ID : **cluster-id.h**
- ZCL_POWER_CONFIG_DEBUG_EXECUTE : **debug.h**
- ZCL_POWER_CONFIG_DEBUG_ON : **debug.h**
- ZCL_POWER_CONFIG_DEBUG_PRINT : **debug.h**
- ZCL_POWER_CONFIG_DEBUG_PRINT_LINE : **debug.h**
- ZCL_POWER_CONFIG_DEBUG_WAITSEND : **debug.h**
- ZCL_POWER_FACTOR_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_POWER_ON_FADE_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_POWER_ON_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_POWER_SOURCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRECONFIGURED_LINK_KEY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PREPAYMENT_CLUSTER_ID : **cluster-id.h**
- ZCL_PRESENT_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESET_READING_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_MAX_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_MAX_SCALED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_MEASUREMENT_CLUSTER_ID : **cluster-id.h**
- ZCL_PRESSURE_MEASUREMENT_DEBUG_EXECUTE : **debug.h**
- ZCL_PRESSURE_MEASUREMENT_DEBUG_ON : **debug.h**
- ZCL_PRESSURE_MEASUREMENT_DEBUG_PRINT : **debug.h**
- ZCL_PRESSURE_MEASUREMENT_DEBUG_PRINT_LINE : **debug.h**
- ZCL_PRESSURE_MEASUREMENT_DEBUG_WAITSEND : **debug.h**
- ZCL_PRESSURE_MIN_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_MIN_SCALED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_SCALE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_SCALED_TOLERANCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_SCALED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_TOLERANCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PREVIOUS_DAY_CONSUMPTION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PREVIOUS_DAY_CONSUMPTION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PREVIOUS_DAY_MAX_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PREVIOUS_DAY_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PREVIOUS_DAY_MAX_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PREVIOUS_DAY_MIN_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRICE_ACKNOWLEDGEMENT_COMMAND_ID : **command-id.h**
- ZCL_PRICE_CLUSTER_ALTERNATE_COST_DELIVERED_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_ALTERNATE_COST_TRAILING_DIGIT_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_ALTERNATE_COST_UNIT_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_DURATION_UNTIL_CHANGED : **price-server.h** , **price-client.h**
- ZCL_PRICE_CLUSTER_END_TIME_NEVER : **price-client.h** , **price-server.h**
- ZCL_PRICE_CLUSTER_GENERATION_PRICE_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_GENERATION_PRICE_RATIO_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_ID : **cluster-id.h**
- ZCL_PRICE_CLUSTER_MAXIMUM_RATE_LABEL_LENGTH : **af-types.h**
- ZCL_PRICE_CLUSTER_NUMBER_OF_BLOCK_THRESHOLDS_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_NUMBER_OF_EVENTS_ALL : **price-server.h**
- ZCL_PRICE_CLUSTER_PRICE_ACKNOWLEDGEMENT_MASK : **price-client.h** , **price-server.h**

- ZCL_PRICE_CLUSTER_PRICE_CONTROL_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_PRICE_RATIO_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_START_TIME_NOW : **price-client.h** , **price-server.h**
- ZCL_PRICE_DEBUG_EXECUTE : **debug.h**
- ZCL_PRICE_DEBUG_ON : **debug.h**
- ZCL_PRICE_DEBUG_PRINT : **debug.h**
- ZCL_PRICE_DEBUG_PRINT_LINE : **debug.h**
- ZCL_PRICE_DEBUG_WAITSEND : **debug.h**
- ZCL_PRICE_DECREASE_RANDOMIZE_MINUTES_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRICE_INCREASE_RANDOMIZE_MINUTES_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRICE_INVALID_INDEX : **price-server.h**
- ZCL_PROFILE_INTERVAL_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PROFILE_WIDE_COMMAND : **af.h**
- ZCL_PROPOSED_CHANGE_IMPLEMENTATION_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PROPOSED_CHANGE_PROVIDER_ID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PROPOSED_CHANGE_SUPPLY_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PROTOCOL_ADDRESS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PROTOCOL_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PUBLISH_BLOCK_PERIOD_COMMAND_ID : **command-id.h**
- ZCL_PUBLISH_PRICE_COMMAND_ID : **command-id.h**
- ZCL_PUMP_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PUMP_CONFIG_CONTROL_CLUSTER_ID : **cluster-id.h**
- ZCL_PUMP_CONFIG_CONTROL_DEBUG_EXECUTE : **debug.h**
- ZCL_PUMP_CONFIG_CONTROL_DEBUG_ON : **debug.h**
- ZCL_PUMP_CONFIG_CONTROL_DEBUG_PRINT : **debug.h**
- ZCL_PUMP_CONFIG_CONTROL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_PUMP_CONFIG_CONTROL_DEBUG_WAITSEND : **debug.h**
- ZCL_PUMP_POWER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PUMP_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_QUALITY_MEASURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_QUERY_NEXT_IMAGE_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_QUERY_NEXT_IMAGE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_QUERY_SPECIFIC_FILE_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_QUERY_SPECIFIC_FILE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_READ_ATTRIBUTES_COMMAND_ID : **command-id.h**
- ZCL_READ_ATTRIBUTES_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_READ_REPORTING_CONFIGURATION_COMMAND_ID : **command-id.h**
- ZCL_READ_REPORTING_CONFIGURATION_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_READING_SNAP_SHOT_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_READY_DATA_CLIENT_TO_SERVER_COMMAND_ID : **command-id.h**
- ZCL_READY_DATA_SERVER_TO_CLIENT_COMMAND_ID : **command-id.h**
- ZCL_RECALL_SCENE_COMMAND_ID : **command-id.h**
- ZCL_REJOIN_INTERVAL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELATIVE_HUMIDITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELATIVE_HUMIDITY_DISPLAY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELATIVE_HUMIDITY_MAX_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELATIVE_HUMIDITY_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER_ID : **cluster-id.h**
- ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_EXECUTE : **debug.h**

- ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_ON : **debug.h**
- ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_PRINT : **debug.h**
- ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_PRINT_LINE : **debug.h**
- ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_WAITSEND : **debug.h**
- ZCL_RELATIVE_HUMIDITY_MIN_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELATIVE_HUMIDITY_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELATIVE_HUMIDITY_TOLERANCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELIABILITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_REMAINING_BATTERY_LIFE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_REMOTE_SENSING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_REMOVE_ALL_GROUPS_COMMAND_ID : **command-id.h**
- ZCL_REMOVE_ALL_SCENES_COMMAND_ID : **command-id.h**
- ZCL_REMOVE_ALL_SCENES_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_REMOVE_GROUP_COMMAND_ID : **command-id.h**
- ZCL_REMOVE_GROUP_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_REMOVE_MIRROR_COMMAND_ID : **command-id.h**
- ZCL_REMOVE_SCENE_COMMAND_ID : **command-id.h**
- ZCL_REMOVE_SCENE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_REPORT_ATTRIBUTES_COMMAND_ID : **command-id.h**
- ZCL_REPORT_EVENT_STATUS_COMMAND_ID : **command-id.h**
- ZCL_REPORT_RSSI_MEASUREMENTS_COMMAND_ID : **command-id.h**
- ZCL_REPORTING_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_REQUEST_FAST_POLL_MODE_COMMAND_ID : **command-id.h**
- ZCL_REQUEST_FAST_POLL_MODE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_REQUEST_MIRROR_COMMAND_ID : **command-id.h**
- ZCL_REQUEST_MIRROR_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_REQUEST_OWN_LOCATION_COMMAND_ID : **command-id.h**
- ZCL_REQUEST_TUNNEL_COMMAND_ID : **command-id.h**
- ZCL_REQUEST_TUNNEL_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_RESET_ALARM_COMMAND_ID : **command-id.h**
- ZCL_RESET_ALARM_LOG_COMMAND_ID : **command-id.h**
- ZCL_RESET_ALL_ALARMS_COMMAND_ID : **command-id.h**
- ZCL_RESET_STARTUP_PARAMETERS_COMMAND_ID : **command-id.h**
- ZCL_RESET_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_RESET_TO_FACTORY_DEFAULTS_COMMAND_ID : **command-id.h**
- ZCL_RESTART_DEVICE_COMMAND_ID : **command-id.h**
- ZCL_RESTART_DEVICE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_RESTORE_STARTUP_PARAMETERS_COMMAND_ID : **command-id.h**
- ZCL_RESTORE_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_RH_DEHUMIDIFICATION_SETPOINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RSSI_LOCATION_CLUSTER_ID : **cluster-id.h**
- ZCL_RSSI_PING_COMMAND_ID : **command-id.h**
- ZCL_RSSI_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_RSSI_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_SAMPLE_MFG_SPECIFIC_CLUSTER_ID : **cluster-id.h**
- ZCL_SAMPLE_MFG_SPECIFIC_OFF_WITH_TRANSITION_COMMAND_ID : **command-id.h**
- ZCL_SAMPLE_MFG_SPECIFIC_ON_WITH_TRANSITION_COMMAND_ID : **command-id.h**
- ZCL_SAMPLE_MFG_SPECIFIC_TOGGLE_WITH_TRANSITION_COMMAND_ID : **command-id.h**
- ZCL_SAMPLE_MFG_SPECIFIC_TRANSITION_TIME_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_SAVE_STARTUP_PARAMETERS_COMMAND_ID : **command-id.h**
- ZCL_SAVE_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_SCAN_ATTEMPTS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SCENE_COUNT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SCENE_NAME_SUPPORT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SCENE_VALID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SCENES_CLUSTER_ID : **cluster-id.h**
- ZCL_SCENES_CLUSTER_MAXIMUM_NAME_LENGTH : **af-types.h**
- ZCL_SCENES_DEBUG_EXECUTE : **debug.h**
- ZCL_SCENES_DEBUG_ON : **debug.h**
- ZCL_SCENES_DEBUG_PRINT : **debug.h**
- ZCL_SCENES_DEBUG_PRINT_LINE : **debug.h**
- ZCL_SCENES_DEBUG_WAITSEND : **debug.h**
- ZCL_SCENES_GLOBAL_SCENE_GROUP_ID : **af-types.h**
- ZCL_SECURITY_KEY_ATTRIBUTE_TYPE : **attribute-type.h** , **attribute-size.h**
- ZCL_SELECT_AVAILABLE_EMERGENCY_CREDIT_COMMAND_ID : **command-id.h**
- ZCL_SEND_PINGS_COMMAND_ID : **command-id.h**
- ZCL_SENSING_LIGHT_SENSOR_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SET_ABSOLUTE_LOCATION_COMMAND_ID : **command-id.h**
- ZCL_SET_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_SET_DEVICE_CONFIGURATION_COMMAND_ID : **command-id.h**
- ZCL_SETPOINT_RAISE_LOWER_COMMAND_ID : **command-id.h**
- ZCL_SETPOINTS_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SETPOINTS_TILT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SHADE_CONFIG_CLOSED_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SHADE_CONFIG_CLUSTER_ID : **cluster-id.h**
- ZCL_SHADE_CONFIG_DEBUG_EXECUTE : **debug.h**
- ZCL_SHADE_CONFIG_DEBUG_ON : **debug.h**
- ZCL_SHADE_CONFIG_DEBUG_PRINT : **debug.h**
- ZCL_SHADE_CONFIG_DEBUG_PRINT_LINE : **debug.h**
- ZCL_SHADE_CONFIG_DEBUG_WAITSEND : **debug.h**
- ZCL_SHADE_CONFIG_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SHADE_CONFIG_MOTOR_STEP_SIZE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SHADE_CONFIG_PHYSICAL_CLOSED_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SHADE_CONFIG_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SHORT_ADDRESS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SIMPLE_METERING_CLUSTER_ID : **cluster-id.h**
- ZCL_SIMPLE_METERING_DEBUG_EXECUTE : **debug.h**
- ZCL_SIMPLE_METERING_DEBUG_ON : **debug.h**
- ZCL_SIMPLE_METERING_DEBUG_PRINT : **debug.h**
- ZCL_SIMPLE_METERING_DEBUG_PRINT_LINE : **debug.h**
- ZCL_SIMPLE_METERING_DEBUG_WAITSEND : **debug.h**
- ZCL_SITE_ID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SPEED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SQUAWK_COMMAND_ID : **command-id.h**
- ZCL_SQUAWK_INFO_ARMED : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_DISARMED : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_HIGH : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_LEVEL_MASK : **ias-wd-server.h**

- ZCL_SQUAWK_INFO_LOW : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_MEDIUM : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_MODE_MASK : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_NO_STROBE : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_STROBE : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_STROBE_MASK : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_VERY_HIGH : **ias-wd-server.h**
- ZCL_STACK_PROFILE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_STACK_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_STANDARD_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_STANDING_CHARGE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_START_OF_BLOCK_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_START_RANDOMIZE_MINUTES_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_START_WARNING_COMMAND_ID : **command-id.h**
- ZCL_STARTUP_CONTROL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_STATUS_FLAGS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_STEP_COLOR_COMMAND_ID : **command-id.h**
- ZCL_STEP_COMMAND_ID : **command-id.h**
- ZCL_STEP_HUE_COMMAND_ID : **command-id.h**
- ZCL_STEP_SATURATION_COMMAND_ID : **command-id.h**
- ZCL_STEP_WITH_ON_OFF_COMMAND_ID : **command-id.h**
- ZCL_STOP_COMMAND_ID : **command-id.h**
- ZCL_STOP_RANDOMIZE_MINUTES_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_STOP_WITH_ON_OFF_COMMAND_ID : **command-id.h**
- ZCL_STORE_SCENE_COMMAND_ID : **command-id.h**
- ZCL_STORE_SCENE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_STRUCT_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_SUMMATION_FORMATTING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SUPPLY_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SUPPLY_STATUS_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_SWITCH_ACTIONS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SWITCH_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SYSTEM_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TEMP_MAX_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TEMP_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TEMP_MEASUREMENT_CLUSTER_ID : **cluster-id.h**
- ZCL_TEMP_MEASUREMENT_DEBUG_EXECUTE : **debug.h**
- ZCL_TEMP_MEASUREMENT_DEBUG_ON : **debug.h**
- ZCL_TEMP_MEASUREMENT_DEBUG_PRINT : **debug.h**
- ZCL_TEMP_MEASUREMENT_DEBUG_PRINT_LINE : **debug.h**
- ZCL_TEMP_MEASUREMENT_DEBUG_WAITSEND : **debug.h**
- ZCL_TEMP_MIN_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TEMP_TOLERANCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TEMPERATURE_DISPLAY_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TEMPERATURE_FORMATTING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TEMPERATURE_UNIT_OF_MEASURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TERMINATE_KEY_ESTABLISHMENT_COMMAND_ID : **command-id.h**
- ZCL_THERMOSTAT_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_THERMOSTAT_CLUSTER_ID : **cluster-id.h**
- ZCL_THERMOSTAT_DEBUG_EXECUTE : **debug.h**
- ZCL_THERMOSTAT_DEBUG_ON : **debug.h**
- ZCL_THERMOSTAT_DEBUG_PRINT : **debug.h**
- ZCL_THERMOSTAT_DEBUG_PRINT_LINE : **debug.h**
- ZCL_THERMOSTAT_DEBUG_WAITSEND : **debug.h**
- ZCL_THERMOSTAT_OCCUPANCY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_THERMOSTAT_UI_CONFIG_CLUSTER_ID : **cluster-id.h**
- ZCL_THERMOSTAT_UI_CONFIG_DEBUG_EXECUTE : **debug.h**
- ZCL_THERMOSTAT_UI_CONFIG_DEBUG_ON : **debug.h**
- ZCL_THERMOSTAT_UI_CONFIG_DEBUG_PRINT : **debug.h**
- ZCL_THERMOSTAT_UI_CONFIG_DEBUG_PRINT_LINE : **debug.h**
- ZCL_THERMOSTAT_UI_CONFIG_DEBUG_WAITSEND : **debug.h**
- ZCL_THRESHOLD_DIVISOR_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_THRESHOLD_MULTIPLIER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_TIER12_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_TIER14_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_TIER2_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_TIER5_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_TIER8_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIME_BETWEEN_SCANS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIME_CLUSTER_ID : **cluster-id.h**
- ZCL_TIME_DEBUG_EXECUTE : **debug.h**
- ZCL_TIME_DEBUG_ON : **debug.h**
- ZCL_TIME_DEBUG_PRINT : **debug.h**
- ZCL_TIME_DEBUG_PRINT_LINE : **debug.h**
- ZCL_TIME_DEBUG_WAITSEND : **debug.h**
- ZCL_TIME_OF_DAY_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_TIME_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIME_ZONE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOGGLE_COMMAND_ID : **command-id.h**
- ZCL_TOP_UP_AMOUNT_1_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_AMOUNT_2_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_AMOUNT_3_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_AMOUNT_4_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_AMOUNT_5_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_DATE_TIME_1_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_DATE_TIME_2_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_DATE_TIME_3_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_DATE_TIME_4_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_DATE_TIME_5_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TRANSFER_DATA_CLIENT_TO_SERVER_COMMAND_ID : **command-id.h**
- ZCL_TRANSFER_DATA_ERROR_CLIENT_TO_SERVER_COMMAND_ID : **command-id.h**

- ZCL_TRANSFER_DATA_ERROR_SERVER_TO_CLIENT_COMMAND_ID : **command-id.h**
- ZCL_TRANSFER_DATA_SERVER_TO_CLIENT_COMMAND_ID : **command-id.h**
- ZCL_TRANSFER_NPDU_COMMAND_ID : **command-id.h**
- ZCL_TRUST_CENTER_ADDRESS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TRUST_CENTER_MASTER_KEY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TUNNELING_CLUSTER_ID : **cluster-id.h**
- ZCL_TUNNELING_CLUSTER_INVALID_TUNNEL_ID : **tunneling-server.h**
- ZCL_ULTRASONIC_OCCUPIED_TO_UNOCCUPIED_DELAY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ULTRASONIC_UNOCCUPIED_TO_OCCUPIED_DELAY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ULTRASONIC_UNOCCUPIED_TO_OCCUPIED_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_UNIT_OF_MEASURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_UNKNOWN_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_UNLOCK_DOOR_COMMAND_ID : **command-id.h**
- ZCL_UNLOCK_DOOR_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_UNOCCUPIED_COOLING_SETPOINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_UNOCCUPIED_HEATING_SETPOINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_UPGRADE_END_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_UPGRADE_END_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_UPGRADE_SERVER_ID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_USE_INSECURE_JOIN_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_UTC_TIME_ATTRIBUTE_TYPE : **attribute-size.h** , **attribute-type.h**
- ZCL_UTILITY_ENROLLMENT_GROUP_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_VALID_UNTIL_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_VELOCITY_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_VIEW_GROUP_COMMAND_ID : **command-id.h**
- ZCL_VIEW_GROUP_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_VIEW_SCENE_COMMAND_ID : **command-id.h**
- ZCL_VIEW_SCENE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_VOLUME_PER_REPORT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_WARNING_INFO_BURGLAR : **ias-wd-server.h**
- ZCL_WARNING_INFO_EMERGENCY : **ias-wd-server.h**
- ZCL_WARNING_INFO_FIRE : **ias-wd-server.h**
- ZCL_WARNING_INFO_MODE_MASK : **ias-wd-server.h**
- ZCL_WARNING_INFO_NO_STROBE : **ias-wd-server.h**
- ZCL_WARNING_INFO_STOP : **ias-wd-server.h**
- ZCL_WARNING_INFO_STROBE : **ias-wd-server.h**
- ZCL_WARNING_INFO_STROBE_MASK : **ias-wd-server.h**
- ZCL_WATER_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_WINDOW_COVERING_CLUSTER_ID : **cluster-id.h**
- ZCL_WINDOW_COVERING_DEBUG_EXECUTE : **debug.h**
- ZCL_WINDOW_COVERING_DEBUG_ON : **debug.h**
- ZCL_WINDOW_COVERING_DEBUG_PRINT : **debug.h**
- ZCL_WINDOW_COVERING_DEBUG_PRINT_LINE : **debug.h**
- ZCL_WINDOW_COVERING_DEBUG_WAITSEND : **debug.h**
- ZCL_WINDOW_COVERING_DOWN_CLOSE_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_GO_TO_LIFT_PERCENTAGE_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_GO_TO_LIFT_SETPOINT_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_GO_TO_LIFT_VALUE_COMMAND_ID : **command-id.h**

- ZCL_WINDOW_COVERING_GO_TO_PROGRAM_SETPOINT_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_GO_TO_TILT_PERCENTAGE_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_GO_TO_TILT_SETPOINT_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_GO_TO_TILT_VALUE_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_STOP_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_UP_OPEN_COMMAND_ID : **command-id.h**
- ZCL_WRITE_ATTRIBUTES_COMMAND_ID : **command-id.h**
- ZCL_WRITE_ATTRIBUTES_NO_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_WRITE_ATTRIBUTES_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_WRITE_ATTRIBUTES_UNDIVIDED_COMMAND_ID : **command-id.h**
- ZCL_ZONE_ENROLL_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_ZONE_ENROLL_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_ZONE_STATE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ZONE_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ZONE_STATUS_CHANGE_NOTIFICATION_COMMAND_ID : **command-id.h**
- ZCL_ZONE_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- zclCommands : **zcl-cli.h**
- ZDO_COMMANDS : **zdo-cli.h**
- zdoCommands : **zdo-cli.h**
- ZIGBEE_2006_STACK_VERSION : **ota.h**
- ZIGBEE_2007_STACK_VERSION : **ota.h**
- ZIGBEE_PRO_STACK_VERSION : **ota.h**
- ZONE_ID_MAP_SECTION_SIZE : **ias-zone-table.h**
- ZONE_TABLE_MAP_SIZE : **ias-zone-table.h**
- ZONE_TABLE_SIZE : **ias-zone-table.h**
- ZONE_TABLE_VALUE_DEFAULT_16 : **ias-zone-table.h**
- ZONE_TABLE_VALUE_DEFAULT_8 : **ias-zone-table.h**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- b -

- bindingTableIndex : **EmberAfIncomingMessage**
- blockPeriodDurationMinutes : **_BlockThreshold**
- blockPeriodStartTime : **_BlockThreshold**
- blockThreshold : **_BlockThreshold**
- buffer : **EmberAfClusterCommand**
- bufLen : **EmberAfClusterCommand**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- c -

- cid : **_ChatRoom**
- cluster : **EmberAfEndpointType**
- clusterCount : **EmberAfEndpointType**
- clusterId : **EmberAfCluster** , **EmberAfEventContext** , **EmberAfReportEntry** , **EmberAfAttributeSearchRecord** , **_SceneExtensionFieldSet** , **EmberAfInterpanHeader** , **EmberAfAllowedInterPanMessage**
- clusterMask : **EmberAfAttributeSearchRecord**
- clusterSize : **EmberAfCluster**
- clusterSpecific : **EmberAfClusterCommand**
- commandId : **EmberAfAllowedInterPanMessage** , **EmberAfClusterCommand**
- contentId : **_Notification**
- coolingTempOffset : **EmberAfLoadControlEvent**
- coolingTempSetPoint : **EmberAfLoadControlEvent**

- count : **EmberAfEndpointList**
- criticalityLevel : **EmberAfLoadControlEvent**
- currency : **EmberAfScheduledPrice**
- currentInterval : **EmberAfReportEntry**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- d -

- defaultValue : **EmberAfDefaultAttributeValue** , **EmberAfAttributeMinMaxValue** , **EmberAfAttributeMetadata** , **EmberAfDefaultOrMinMaxAttributeValue**
- destinationEndpoint : **EmberAfLoadControlEvent** , **EmberAfReportEntry**
- deviceClass : **EmberAfLoadControlEvent**
- deviceId : **EmberAfDefinedEndpoint**
- deviceSpecificFileEui64 : **EmberAfOtaImageId**
- deviceVersion : **EmberAfDefinedEndpoint**
- direction : **EmberAfClusterCommand** , **_ReadReportingConfigurationAttributeRecord** , **_ConfigureReportingRecord** , **_ReadReportingConfigurationRecord** , **_ConfigureReportingStatusRecord** , **EmberAfReportEntry**
- duration : **EmberAfLoadControlEvent** , **EmberAfScheduledPrice**
- durationInMinutes : **EmberAfMessagingServerMessage**

- dutyCycle : **EmberAfLoadControlEvent**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- e -

- endpoint : **EmberAfAttributeSearchRecord** , **EmberAfDefinedEndpoint** , **EmberAfSceneTableEntry** , **_NodeInformation** , **EmberAfEventContext**
- endpointSize : **EmberAfEndpointType**
- endpointType : **EmberAfDefinedEndpoint**
- endRand : **EmberAfLoadControlEvent**
- eventControl : **EmberAfLoadControlEvent** , **EmberAfEventContext**
- eventId : **EmberAfLoadControlEvent**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- f -

- failCount : **EmberAfReportEntry**
- fieldControl : **EmberAfOtaHeader**
- fieldDebug : **EmAfOtaStorageLinuxConfig**
- fileAddedHandler : **EmAfOtaStorageLinuxConfig**
- fileDebug : **EmAfOtaStorageLinuxConfig**
- firmwareVersion : **EmberAfOtaHeader** , **EmberAfOtaImageId**
- fuelType : **_BlockThreshold**
- functions : **EmberAfCluster**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- g -

- generationPrice : **EmberAfScheduledPrice**
- generationPriceRatio : **EmberAfScheduledPrice**
- groupId : **EmberAfInterpanHeader** , **EmberAfSceneTableEntry**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- h -

- headerLength : **EmberAfOtaHeader**
- headerString : **EmberAfOtaHeader**
- headerVersion : **EmberAfOtaHeader**
- heatingTempOffset : **EmberAfLoadControlEvent**
- heatingTempSetPoint : **EmberAfLoadControlEvent**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- i -

- id : **EmberAfTagData**
- ieeeAdd : **EmberAfIasZoneTableEntry**
- ignoreFilesWithUnderscorePrefix : **EmAfOtaStorageLinuxConfig**
- imageSize : **EmberAfOtaHeader**
- imageTypeId : **EmberAfOtaImageId** , **EmberAfOtaHeader**
- index : **EmberAfManufacturerCodeEntry**
- interPanHeader : **EmberAfClusterCommand**
- inUse : **EmberAfReportEntry**
- isClient : **EmberAfEventContext**
- issuerEventID : **EmberAfScheduledPrice**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- l -

- lastHopLqi : **EmberAfIncomingMessage**
- lastHopRssi : **EmberAfIncomingMessage**
- lastReportValue : **EmberAfReportEntry**
- length : **_SceneExtensionFieldSet** , **EmberAfTagData**
- list : **EmberAfEndpointList**
- longAddress : **EmberAfInterpanHeader**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- m -

- manufacturerCode : **EmberAfAttributeSearchRecord** , **EmberAfManufacturerCodeEntry** , **EmberAfReportEntry**
- manufacturerId : **EmberAfOtaImageId** , **EmberAfOtaHeader**
- mask : **EmberAfReportEntry** , **EmberAfAttributeMetadata** , **EmberAfCluster**
- matchAddress : **EmberAfServiceDiscoveryResult**
- maximumHardwareVersion : **EmberAfOtaHeader**
- maximumReportingInterval : **_ConfigureReportingRecord** , **_ReadReportingConfigurationRecord**
- maxInterval : **EmberAfReportEntry**
- maxValue : **EmberAfAttributeMinMaxValue**
- memoryDebug : **EmAfOtaStorageLinuxConfig**
- message : **EmberAfIncomingMessage** , **EmberAfMessagingServerMessage**
- messageControl : **EmberAfMessagingServerMessage**

- messageId : **EmberAfMessagingServerMessage**
- messageType : **EmberAfInterpanHeader**
- mfgCode : **EmberAfClusterCommand**
- mfgSpecific : **EmberAfClusterCommand**
- minimumHardwareVersion : **EmberAfOtaHeader**
- minimumReportingInterval : **_ReadReportingConfigurationRecord** , **_ConfigureReportingRecord**
- minInterval : **EmberAfReportEntry**
- minValue : **EmberAfAttributeMinMaxValue**
- msgLen : **EmberAfIncomingMessage**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- n -

- name : **_ChatRoom**
- neighbor : **_NeighborInfo**
- nickname : **_NodeInformation** , **_ChatParticipant**
- nodeId : **EmberAfReportEntry**
- numberOfBlockThresholds : **EmberAfScheduledPrice**
- numberOfPriceTiersAndTier : **EmberAfScheduledPrice**
- numberRssiMeasurements : **_NeighborInfo**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- o -

- optionControl : **EmberAfLoadControlEvent**
- options : **EmberAfAllowedInterPanMessage** , **EmberAfInterpanHeader**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- p -

- panId : **EmberAfInterpanHeader**
- payloadStartIndex : **EmberAfClusterCommand**
- price : **EmberAfScheduledPrice**
- priceControl : **_BlockThreshold** , **EmberAfScheduledPrice**
- priceRatio : **EmberAfScheduledPrice**
- priceTrailingDigitAndTier : **EmberAfScheduledPrice**
- printFileDiscoveryOrRemoval : **EmAfOtaStorageLinuxConfig**
- profileId : **EmberAfInterpanHeader** , **EmberAfAllowedInterPanMessage** , **EmberAfDefinedEndpoint**
- providerId : **EmberAfScheduledPrice**
- ptrToDefaultValue : **EmberAfDefaultOrMinMaxAttributeValue** , **EmberAfDefaultAttributeValue**
- ptrToMinMaxValue : **EmberAfDefaultOrMinMaxAttributeValue**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- r -

- rateLabel : **EmberAfScheduledPrice**
- reportableChange : **EmberAfReportEntry**
- reportableChangeLocation : **_ReadReportingConfigurationRecord** , **_ConfigureReportingRecord**
- responseData : **EmberAfServiceDiscoveryResult**
- rssi : **_NeighborInfo**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- s -

- sceneId : **EmberAfSceneTableEntry**
- securityCredentials : **EmberAfOtaHeader**
- seqNum : **EmberAfClusterCommand**
- shortAddress : **EmberAfInterpanHeader**
- size : **EmberAfAttributeMetadata**
- sleepControl : **EmberAfEventContext**
- source : **EmberAfClusterCommand** , **EmberAfLoadControlEvent** , **EmberAfIncomingMessage**
- sourceEndpoint : **EmberAfLoadControlEvent** , **EmberAfReportEntry**
- standingCharge : **_BlockThreshold**
- startRand : **EmberAfLoadControlEvent**
- startTime : **EmberAfMessagingServerMessage** , **EmberAfScheduledPrice** , **EmberAfLoadControlEvent**

- status : **_WriteAttributeStatusRecord** , **_ReadReportingConfigurationRecord** , **_ReadAttributeStatusRecord** , **_ConfigureReportingStatusRecord** , **EmberAfServiceDiscoveryResult**
- statusFeedback : **_Notification**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- t -

- timeout : **EmberAfReportEntry**
- timeoutPeriod : **_ConfigureReportingRecord** , **_ReadReportingConfigurationRecord**
- transitionTime : **EmberAfSceneTableEntry**
- type : **EmberAfIncomingMessage** , **EmberAfClusterCommand**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- u -

- uid : **_ChatParticipant** , **_NodeInformation**
- unitOfMeasure : **EmberAfScheduledPrice**
- upgradeFileDestination : **EmberAfOtaHeader**
- utilityEnrollmentGroup : **EmberAfLoadControlEvent**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- v -

- valid : **EmberAfMessagingServerMessage** , **EmberAfScheduledPrice**
- value : **_SceneExtensionFieldSet**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- x -

- x : **_NeighborInfo**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- y -

- y : **_NeighborInfo**

- All
- Variables

- a
- b
- c
- d
- e
- f
- g
- h
- i
- l
- m
- n
- o
- p
- r
- s
- t
- u
- v
- x
- y
- z

- z -

- z : **_NeighborInfo**
- zdoRequestClusterId : **EmberAfServiceDiscoveryResult**
- zigbeeStackVersion : **EmberAfOtaHeader**
- zoneType : **EmberAfIasZoneTableEntry**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- a
- b
- c
- e
- g
- i
- k
- o
- p
- r
- s
- u

- b -

- bacnetProtocolTxTransferNpdu() : **bacnet-protocol-tunnel.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- a
- b
- c
- e
- g
- i
- k
- o
- p
- r
- s
- u

- c -

- cbkeInterPanCommand() : **security-cli.h**
- cbkePartnerCommand() : **security-cli.h**
- cbkeStartCommand() : **security-cli.h**
- changeKeyCommand() : **security-cli.h**
- clearAllTemporaryPublicData() : **key-establishment-storage.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- a
- b
- c
- e
- g
- i
- k
- o
- p
- r
- s
- u

- e -

- emAfAbortPageRequest() : **ota-client-page-request.h**
- emAfBootloadSendByte() : **ota-bootload-ncp.h**
- emAfBootloadSendData() : **ota-bootload-ncp.h**
- emAfBootloadWaitChar() : **ota-bootload-ncp.h**
- emAfCancelAllLoadControlEvents() : **load-control-event-table.h**
- emAfCancelLoadControlEvent() : **load-control-event-table.h**
- emAfClearScheduledLoadControlEvents() : **drlc-server.h**
- emAfCliAddReportEntryCommand() : **zcl-cli.h**
- emAfCliBsendCommand() : **zcl-cli.h**
- emAfCliClearEventsCommand() : **zcl-cli.h**
- emAfCliCountersCommand() : **core-cli.h**
- emAfCliEchoCommand() : **core-cli.h**
- emAfCliHelpCommand() : **core-cli.h**
- emAfCliInfoCommand() : **core-cli.h**
- emAfCliRawCommand() : **zcl-cli.h**
- emAfCliReadCommand() : **zcl-cli.h**
- emAfCliResetCommand() : **core-cli.h**
- emAfCliSendCommand() : **zcl-cli.h**
- emAfCliTimesyncCommand() : **zcl-cli.h**
- emAfCliWriteCommand() : **zcl-cli.h**
- emAfGetCurrentPageRequestStatus() : **ota-client-page-request.h**
- emAfGetFinishedPageRequestOffset() : **ota-client-page-request.h**
- emAfGetPageRequestMissedPacketDelayMs() : **ota-client-page-request.h**
- emAfGetScheduledLoadControlEvent() : **drlc-server.h**
- emAfHandlingPageRequestClient() : **ota-client-page-request.h**
- emAfIasAcePrintInfo() : **ias-ace-server.h**
- emAfIdentifyPrintInfo() : **identify.h**

- emAfInitPageRequestClient() : **ota-client-page-request.h**
- emAfInitXmodemState() : **ota-bootload-xmodem.h**
- emAfInterpanApsCryptMessage() : **interpan.h**
- emAfLoadControlEventOptInOrOut() : **load-control-event-table.h**
- emAfLoadControlEventTablePrint() : **load-control-event-table.h**
- emAfLoadControlEventTableTick() : **load-control-event-table.h**
- emAfNextMissedBlockRequestOffset() : **ota-client-page-request.h**
- emAfNoteReceivedBlockForPageRequestClient() : **ota-client-page-request.h**
- emAfNoteSignatureFailure() : **load-control-event-table.h**
- emAfOtaClientPrintState() : **ota-client.h**
- emAfOtaClientSignatureVerifyPrintSigners() : **ota-client-signature-verify.h**
- emAfOtaClientStop() : **ota-client.h**
- emAfOtaCreateEmberAfOtaImageIdStruct() : **ota.h**
- emAfOtaFindImageIdByIndex() : **ota-cli.h**
- emAfOtaImageBlockRequestHandler() : **ota-server.h**
- emAfOtaImageDelete() : **ota-cli.h**
- emAfOtaImageSignatureVerify() : **ota-client-signature-verify.h**
- emAfOtaPageRequestErrorHandler() : **ota-server.h**
- emAfOtaPageRequestHandler() : **ota-server.h**
- emAfOtaPageRequestTick() : **ota-server.h**
- emAfOtaParseImageIdFromMessage() : **ota.h**
- emAfOtaPrintAllImages() : **ota-cli.h**
- emAfOtaServerGetBlockSize() : **ota-server.h**
- emAfOtaServerHandlingPageRequest() : **ota-server.h**
- emAfOtaServerPolicyPrint() : **ota-server-policy.h**
- emAfOtaServerSetBlockRequestPolicy() : **ota-server-policy.h**
- emAfOtaServerSetPageRequestPolicy() : **ota-server-policy.h**
- emAfOtaServerSetQueryPolicy() : **ota-server-policy.h**
- emAfOtaServerSetUpgradePolicy() : **ota-server-policy.h**
- emAfOtaStorageAddImageFile() : **ota-storage.h**
- emAfOtaStorageAppendImageData() : **ota-storage.h**
- emAfOtaStorageClose() : **ota-storage.h**
- emAfOtaStorageCreateImage() : **ota-storage.h**
- emAfOtaStorageGetConfig() : **ota-storage-linux.h**
- emAfOtaStorageGetFilepath() : **ota-storage.h**
- emAfOtaStorageGetHeaderLengthAndImageSize() : **ota-storage.h**
- emAfOtaStorageGetImageIdFromHeader() : **ota-storage.h**
- emAfOtaStorageGetRawHeaderData() : **ota-storage.h**
- emAfOtaStorageGetTagDataFromImage() : **ota-storage.h**
- emAfOtaStorageGetTagOffsetAndSize() : **ota-storage.h**
- emAfOtaStorageReadAllTagInfo() : **ota-storage.h**
- emAfOtaStorageSetConfig() : **ota-storage-linux.h**
- emAfOtaStorageSetDevice() : **ota-storage.h**
- emAfOtaVerifyStoredDataFinish() : **ota-client-signature-verify.h**
- emAfPageRequestTimerExpired() : **ota-client-page-request.h**
- emAfPluginDrlcServerPrintInfo() : **drlc-server.h**
- emAfPluginGenericTunnelGetProtocolAddress() : **generic-tunnel.h**
- emAfPluginGenericTunnelPrintInfo() : **generic-tunnel.h**
- emAfPluginGroupsServerPrintInfo() : **groups-server.h**
- emAfPluginInterpanProcessMessage() : **interpan.h**
- emAfPluginInterpanSendRawMessage() : **interpan.h**
- emAfPluginMessagingClientPrintInfo() : **messaging-client.h**
- emAfPluginMessagingServerGetMessageStruct() : **messaging-server.h**

- emAfPluginMessagingServerPrintInfo() : **messaging-server.h**
- emAfPluginPriceClientPrintInfo() : **price-client.h**
- emAfPluginScenesServerPrintInfo() : **scenes.h**
- emAfPluginSmartEnergyRegistrationTick() : **smart-energy-registration.h**
- emAfPluginSmartEnergyRegistrationTrustCenterKeyEstablishmentEndpoint() : **smart-energy-registration.h**
- emAfPluginTrustCenterKeepaliveReadAttributesResponseCallback() : **trust-center-keepalive.h**
- emAfPostNcpBootload() : **ota-bootload-ncp.h**
- emAfRebootNcpAfterBootload() : **ota-bootload-ncp.h**
- emAfScheduleLoadControlEvent() : **load-control-event-table.h**
- emAfSendImageBlockRequestTest() : **ota-client.h**
- emAfSendXmodemData() : **ota-bootload-xmodem.h**
- emAfServerPageRequestTickCallback() : **ota-server-policy.h**
- emAfSetLocalTemperature() : **thermostat.h**
- emAfSetPageRequest() : **ota-client.h**
- emAfSetPageRequestMissedBlockModulus() : **ota-server-policy.h**
- emAfStartNcpBootloaderCommunications() : **ota-bootload-ncp.h**
- emAfTestMeterGetProfiles() : **simple-metering-test.h**
- emAfTestMeterInit() : **simple-metering-test.h**
- emAfTestMeterTick() : **simple-metering-test.h**
- emAfTimeClusterClientGetCurrentTime() : **time-client.h**
- emAfTimeClusterClientSetCurrentTime() : **time-client.h**
- emAfTimeClusterServerGetCurrentTime() : **time-server.h**
- emAfTimeClusterServerSetCurrentTime() : **time-server.h**
- emAfUsingPageRequest() : **ota-client.h**
- emberAfAddAddressTableEntry() : **af.h**
- emberAfAddToCurrentAppTasks() : **af.h**
- emberAfAlarmClusterAlarmCallback() : **callback.h**
- emberAfAlarmClusterClientAttributeChangedCallback() : **callback.h**
- emberAfAlarmClusterClientCommandParse() : **call-command-handler.h**
- emberAfAlarmClusterClientDefaultResponseCallback() : **callback.h**
- emberAfAlarmClusterClientInitCallback() : **callback.h**
- emberAfAlarmClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfAlarmClusterClientMessageSentCallback() : **callback.h**
- emberAfAlarmClusterClientTickCallback() : **callback.h**
- emberAfAlarmClusterGetAlarmCallback() : **callback.h**
- emberAfAlarmClusterGetAlarmResponseCallback() : **callback.h**
- emberAfAlarmClusterResetAlarmCallback() : **callback.h**
- emberAfAlarmClusterResetAlarmLogCallback() : **callback.h**
- emberAfAlarmClusterResetAllAlarmsCallback() : **callback.h**
- emberAfAlarmClusterServerAttributeChangedCallback() : **callback.h**
- emberAfAlarmClusterServerCommandParse() : **call-command-handler.h**
- emberAfAlarmClusterServerDefaultResponseCallback() : **callback.h**
- emberAfAlarmClusterServerInitCallback() : **callback.h**
- emberAfAlarmClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfAlarmClusterServerMessageSentCallback() : **callback.h**
- emberAfAlarmClusterServerTickCallback() : **callback.h**
- emberAfAllowNetworkWriteAttributeCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterClientAttributeChangedCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterClientDefaultResponseCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterClientInitCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterClientMessageSentCallback() : **callback.h**

- emberAfBacnetProtocolTunnelClusterClientTickCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterServerAttributeChangedCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterServerCommandParse() : **call-command-handler.h**
- emberAfBacnetProtocolTunnelClusterServerDefaultResponseCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterServerInitCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterServerMessageSentCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterServerTickCallback() : **callback.h**
- emberAfBacnetProtocolTunnelClusterTransferNpduCallback() : **callback.h**
- emberAfBallastConfigurationClusterClientAttributeChangedCallback() : **callback.h**
- emberAfBallastConfigurationClusterClientDefaultResponseCallback() : **callback.h**
- emberAfBallastConfigurationClusterClientInitCallback() : **callback.h**
- emberAfBallastConfigurationClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBallastConfigurationClusterClientMessageSentCallback() : **callback.h**
- emberAfBallastConfigurationClusterClientTickCallback() : **callback.h**
- emberAfBallastConfigurationClusterServerAttributeChangedCallback() : **callback.h**
- emberAfBallastConfigurationClusterServerDefaultResponseCallback() : **callback.h**
- emberAfBallastConfigurationClusterServerInitCallback() : **callback.h**
- emberAfBallastConfigurationClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBallastConfigurationClusterServerMessageSentCallback() : **callback.h**
- emberAfBallastConfigurationClusterServerTickCallback() : **callback.h**
- emberAfBasicClusterClientAttributeChangedCallback() : **callback.h**
- emberAfBasicClusterClientDefaultResponseCallback() : **callback.h**
- emberAfBasicClusterClientInitCallback() : **callback.h**
- emberAfBasicClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBasicClusterClientMessageSentCallback() : **callback.h**
- emberAfBasicClusterClientTickCallback() : **callback.h**
- emberAfBasicClusterResetToFactoryDefaultsCallback() : **callback.h**
- emberAfBasicClusterServerAttributeChangedCallback() : **callback.h**
- emberAfBasicClusterServerCommandParse() : **call-command-handler.h**
- emberAfBasicClusterServerDefaultResponseCallback() : **callback.h**
- emberAfBasicClusterServerInitCallback() : **callback.h**
- emberAfBasicClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBasicClusterServerMessageSentCallback() : **callback.h**
- emberAfBasicClusterServerTickCallback() : **callback.h**
- emberAfBinaryInputBasicClusterClientAttributeChangedCallback() : **callback.h**
- emberAfBinaryInputBasicClusterClientDefaultResponseCallback() : **callback.h**
- emberAfBinaryInputBasicClusterClientInitCallback() : **callback.h**
- emberAfBinaryInputBasicClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBinaryInputBasicClusterClientMessageSentCallback() : **callback.h**
- emberAfBinaryInputBasicClusterClientTickCallback() : **callback.h**
- emberAfBinaryInputBasicClusterServerAttributeChangedCallback() : **callback.h**
- emberAfBinaryInputBasicClusterServerDefaultResponseCallback() : **callback.h**
- emberAfBinaryInputBasicClusterServerInitCallback() : **callback.h**
- emberAfBinaryInputBasicClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfBinaryInputBasicClusterServerMessageSentCallback() : **callback.h**
- emberAfBinaryInputBasicClusterServerTickCallback() : **callback.h**
- emberAfClusterInitCallback() : **callback.h**
- emberAfColorControlClusterClientAttributeChangedCallback() : **callback.h**
- emberAfColorControlClusterClientDefaultResponseCallback() : **callback.h**
- emberAfColorControlClusterClientInitCallback() : **callback.h**
- emberAfColorControlClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfColorControlClusterClientMessageSentCallback() : **callback.h**

- emberAfColorControlClusterClientTickCallback() : **callback.h**
- emberAfColorControlClusterMoveColorCallback() : **callback.h**
- emberAfColorControlClusterMoveHueCallback() : **callback.h**
- emberAfColorControlClusterMoveSaturationCallback() : **callback.h**
- emberAfColorControlClusterMoveToColorCallback() : **callback.h**
- emberAfColorControlClusterMoveToColorTemperatureCallback() : **callback.h**
- emberAfColorControlClusterMoveToHueAndSaturationCallback() : **callback.h**
- emberAfColorControlClusterMoveToHueCallback() : **callback.h**
- emberAfColorControlClusterMoveToSaturationCallback() : **callback.h**
- emberAfColorControlClusterServerAttributeChangedCallback() : **callback.h**
- emberAfColorControlClusterServerCommandParse() : **call-command-handler.h**
- emberAfColorControlClusterServerDefaultResponseCallback() : **callback.h**
- emberAfColorControlClusterServerInitCallback() : **callback.h**
- emberAfColorControlClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfColorControlClusterServerMessageSentCallback() : **callback.h**
- emberAfColorControlClusterServerTickCallback() : **callback.h**
- emberAfColorControlClusterStepColorCallback() : **callback.h**
- emberAfColorControlClusterStepHueCallback() : **callback.h**
- emberAfColorControlClusterStepSaturationCallback() : **callback.h**
- emberAfCommissioningClusterClientAttributeChangedCallback() : **callback.h**
- emberAfCommissioningClusterClientCommandParse() : **call-command-handler.h**
- emberAfCommissioningClusterClientDefaultResponseCallback() : **callback.h**
- emberAfCommissioningClusterClientInitCallback() : **callback.h**
- emberAfCommissioningClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfCommissioningClusterClientMessageSentCallback() : **callback.h**
- emberAfCommissioningClusterClientTickCallback() : **callback.h**
- emberAfCommissioningClusterResetStartupParametersCallback() : **callback.h**
- emberAfCommissioningClusterResetStartupParametersResponseCallback() : **callback.h**
- emberAfCommissioningClusterRestartDeviceCallback() : **callback.h**
- emberAfCommissioningClusterRestartDeviceResponseCallback() : **callback.h**
- emberAfCommissioningClusterRestoreStartupParametersCallback() : **callback.h**
- emberAfCommissioningClusterRestoreStartupParametersResponseCallback() : **callback.h**
- emberAfCommissioningClusterSaveStartupParametersCallback() : **callback.h**
- emberAfCommissioningClusterSaveStartupParametersResponseCallback() : **callback.h**
- emberAfCommissioningClusterServerAttributeChangedCallback() : **callback.h**
- emberAfCommissioningClusterServerCommandParse() : **call-command-handler.h**
- emberAfCommissioningClusterServerDefaultResponseCallback() : **callback.h**
- emberAfCommissioningClusterServerInitCallback() : **callback.h**
- emberAfCommissioningClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfCommissioningClusterServerMessageSentCallback() : **callback.h**
- emberAfCommissioningClusterServerTickCallback() : **callback.h**
- emberAfCompareValues() : **af.h**
- emberAfConfigureReportingResponseCallback() : **callback.h**
- emberAfContainsClient() : **af.h**
- emberAfContainsCluster() : **af.h**
- emberAfContainsServer() : **af.h**
- emberAfContinueJoinableNetworkSearchCallback() : **callback.h**
- emberAfCopyInt16u() : **af.h**
- emberAfCopyInt24u() : **af.h**
- emberAfCopyInt32u() : **af.h**
- emberAfCopyString() : **af.h**
- emberAfCustomCommandLineCallback() : **callback.h**
- emberAfCustomStorageInitCallback() : **ota-storage-simple-custom.h** , **ota-storage-simple-driver.h**

- emberAfCustomStorageReadCallback() : **ota-storage-simple-custom.h** , **ota-storage-simple-driver.h**
- emberAfCustomStorageWriteCallback() : **ota-storage-simple-custom.h** , **ota-storage-simple-driver.h**
- emberAfDeactivateClusterTick() : **af.h**
- emberAfDefaultResponseCallback() : **callback.h**
- emberAfDehumidControlClusterClientAttributeChangedCallback() : **callback.h**
- emberAfDehumidControlClusterClientDefaultResponseCallback() : **callback.h**
- emberAfDehumidControlClusterClientInitCallback() : **callback.h**
- emberAfDehumidControlClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDehumidControlClusterClientMessageSentCallback() : **callback.h**
- emberAfDehumidControlClusterClientTickCallback() : **callback.h**
- emberAfDehumidControlClusterServerAttributeChangedCallback() : **callback.h**
- emberAfDehumidControlClusterServerDefaultResponseCallback() : **callback.h**
- emberAfDehumidControlClusterServerInitCallback() : **callback.h**
- emberAfDehumidControlClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDehumidControlClusterServerMessageSentCallback() : **callback.h**
- emberAfDehumidControlClusterServerTickCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterCancelAllLoadControlEventsCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterCancelLoadControlEventCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterClientAttributeChangedCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterClientCommandParse() : **call-command-handler.h**
- emberAfDemandResponseLoadControlClusterClientDefaultResponseCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterClientInitCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterClientMessageSentCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterClientTickCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterGetScheduledEventsCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterLoadControlEventCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterReportEventStatusCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterServerAttributeChangedCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterServerCommandParse() : **call-command-handler.h**
- emberAfDemandResponseLoadControlClusterServerDefaultResponseCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterServerInitCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterServerMessageSentCallback() : **callback.h**
- emberAfDemandResponseLoadControlClusterServerTickCallback() : **callback.h**
- emberAfDeviceTempClusterClientAttributeChangedCallback() : **callback.h**
- emberAfDeviceTempClusterClientDefaultResponseCallback() : **callback.h**
- emberAfDeviceTempClusterClientInitCallback() : **callback.h**
- emberAfDeviceTempClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDeviceTempClusterClientMessageSentCallback() : **callback.h**
- emberAfDeviceTempClusterClientTickCallback() : **callback.h**
- emberAfDeviceTempClusterServerAttributeChangedCallback() : **callback.h**
- emberAfDeviceTempClusterServerDefaultResponseCallback() : **callback.h**
- emberAfDeviceTempClusterServerInitCallback() : **callback.h**
- emberAfDeviceTempClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDeviceTempClusterServerMessageSentCallback() : **callback.h**
- emberAfDeviceTempClusterServerTickCallback() : **callback.h**
- emberAfDiscoverAttributesResponseCallback() : **callback.h**
- emberAfDoorLockClusterClientAttributeChangedCallback() : **callback.h**
- emberAfDoorLockClusterClientCommandParse() : **call-command-handler.h**
- emberAfDoorLockClusterClientDefaultResponseCallback() : **callback.h**
- emberAfDoorLockClusterClientInitCallback() : **callback.h**

- emberAfDoorLockClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDoorLockClusterClientMessageSentCallback() : **callback.h**
- emberAfDoorLockClusterClientTickCallback() : **callback.h**
- emberAfDoorLockClusterLockDoorCallback() : **callback.h**
- emberAfDoorLockClusterLockDoorResponseCallback() : **callback.h**
- emberAfDoorLockClusterServerAttributeChangedCallback() : **callback.h**
- emberAfDoorLockClusterServerCommandParse() : **call-command-handler.h**
- emberAfDoorLockClusterServerDefaultResponseCallback() : **callback.h**
- emberAfDoorLockClusterServerInitCallback() : **callback.h**
- emberAfDoorLockClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfDoorLockClusterServerMessageSentCallback() : **callback.h**
- emberAfDoorLockClusterServerTickCallback() : **callback.h**
- emberAfDoorLockClusterUnlockDoorCallback() : **callback.h**
- emberAfDoorLockClusterUnlockDoorResponseCallback() : **callback.h**
- emberAfEndpointCount() : **af.h**
- emberAfEventAction() : **load-control-event-table.h**
- emberAfEventControlSetDelay() : **af.h**
- emberAfExternalAttributeReadCallback() : **callback.h**
- emberAfExternalAttributeWriteCallback() : **callback.h**
- emberAfEzspPollCompleteHandler() : **af.h**
- emberAfFanControlClusterClientAttributeChangedCallback() : **callback.h**
- emberAfFanControlClusterClientDefaultResponseCallback() : **callback.h**
- emberAfFanControlClusterClientInitCallback() : **callback.h**
- emberAfFanControlClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfFanControlClusterClientMessageSentCallback() : **callback.h**
- emberAfFanControlClusterClientTickCallback() : **callback.h**
- emberAfFanControlClusterServerAttributeChangedCallback() : **callback.h**
- emberAfFanControlClusterServerDefaultResponseCallback() : **callback.h**
- emberAfFanControlClusterServerInitCallback() : **callback.h**
- emberAfFanControlClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfFanControlClusterServerMessageSentCallback() : **callback.h**
- emberAfFanControlClusterServerTickCallback() : **callback.h**
- emberAfFillBuffer() : **client-api.h**
- emberAfFillExternalBuffer() : **client-api.h**
- emberAfFillExternalManufacturerSpecificBuffer() : **client-api.h**
- emberAfFindDevicesByProfileAndCluster() : **af.h**
- emberAfFindDevicesSupportingCluster() : **af.h**
- emberAfFindIeeeAddress() : **af.h**
- emberAfFindNodeId() : **af.h**
- emberAfFindUnusedPanIdAndForm() : **af.h**
- emberAfFindUnusedPanIdAndFormCallback() : **callback.h**
- emberAfFlowMeasurementClusterClientAttributeChangedCallback() : **callback.h**
- emberAfFlowMeasurementClusterClientDefaultResponseCallback() : **callback.h**
- emberAfFlowMeasurementClusterClientInitCallback() : **callback.h**
- emberAfFlowMeasurementClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfFlowMeasurementClusterClientMessageSentCallback() : **callback.h**
- emberAfFlowMeasurementClusterClientTickCallback() : **callback.h**
- emberAfFlowMeasurementClusterServerAttributeChangedCallback() : **callback.h**
- emberAfFlowMeasurementClusterServerDefaultResponseCallback() : **callback.h**
- emberAfFlowMeasurementClusterServerInitCallback() : **callback.h**
- emberAfFlowMeasurementClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfFlowMeasurementClusterServerMessageSentCallback() : **callback.h**
- emberAfFlowMeasurementClusterServerTickCallback() : **callback.h**

- emberAfFlush() : **af.h**
- emberAfFormNetwork() : **af.h**
- emberAfGenericTunnelClusterAdvertiseProtocolAddressCallback() : **callback.h**
- emberAfGenericTunnelClusterClientAttributeChangedCallback() : **callback.h**
- emberAfGenericTunnelClusterClientCommandParse() : **call-command-handler.h**
- emberAfGenericTunnelClusterClientDefaultResponseCallback() : **callback.h**
- emberAfGenericTunnelClusterClientInitCallback() : **callback.h**
- emberAfGenericTunnelClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfGenericTunnelClusterClientMessageSentCallback() : **callback.h**
- emberAfGenericTunnelClusterClientTickCallback() : **callback.h**
- emberAfGenericTunnelClusterMatchProtocolAddressCallback() : **callback.h**
- emberAfGenericTunnelClusterMatchProtocolAddressResponseCallback() : **callback.h**
- emberAfGenericTunnelClusterServerAttributeChangedCallback() : **callback.h**
- emberAfGenericTunnelClusterServerCommandParse() : **call-command-handler.h**
- emberAfGenericTunnelClusterServerDefaultResponseCallback() : **callback.h**
- emberAfGenericTunnelClusterServerInitCallback() : **callback.h**
- emberAfGenericTunnelClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfGenericTunnelClusterServerMessageSentCallback() : **callback.h**
- emberAfGenericTunnelClusterServerTickCallback() : **callback.h**
- emberAfGetAttributeAnalogOrDiscreteType() : **af.h**
- emberAfGetCommandApsFrame() : **af.h**
- emberAfGetCurrentPrice() : **price-server.h**
- emberAfGetCurrentSleepControl() : **af.h**
- emberAfGetCurrentTime() : **af.h**
- emberAfGetCurrentTimeCallback() : **callback.h**
- emberAfGetDataSize() : **af.h**
- emberAfGetEui64() : **af.h**
- emberAfGetInt16u() : **af.h**
- emberAfGetInt24u() : **af.h**
- emberAfGetInt32u() : **af.h**
- emberAfGetNodeId() : **af.h**
- emberAfGetString() : **af.h**
- emberAfGroupsClusterAddGroupCallback() : **callback.h**
- emberAfGroupsClusterAddGroupIfIdentifyingCallback() : **callback.h**
- emberAfGroupsClusterAddGroupResponseCallback() : **callback.h**
- emberAfGroupsClusterClearGroupTableCallback() : **callback.h**
- emberAfGroupsClusterClientAttributeChangedCallback() : **callback.h**
- emberAfGroupsClusterClientCommandParse() : **call-command-handler.h**
- emberAfGroupsClusterClientDefaultResponseCallback() : **callback.h**
- emberAfGroupsClusterClientInitCallback() : **callback.h**
- emberAfGroupsClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfGroupsClusterClientMessageSentCallback() : **callback.h**
- emberAfGroupsClusterClientTickCallback() : **callback.h**
- emberAfGroupsClusterEndpointInGroupCallback() : **callback.h**
- emberAfGroupsClusterGetGroupMembershipCallback() : **callback.h**
- emberAfGroupsClusterGetGroupMembershipResponseCallback() : **callback.h**
- emberAfGroupsClusterRemoveAllGroupsCallback() : **callback.h**
- emberAfGroupsClusterRemoveGroupCallback() : **callback.h**
- emberAfGroupsClusterRemoveGroupResponseCallback() : **callback.h**
- emberAfGroupsClusterServerAttributeChangedCallback() : **callback.h**
- emberAfGroupsClusterServerCommandParse() : **call-command-handler.h**
- emberAfGroupsClusterServerDefaultResponseCallback() : **callback.h**
- emberAfGroupsClusterServerInitCallback() : **callback.h**

- emberAfGroupsClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfGroupsClusterServerMessageSentCallback() : **callback.h**
- emberAfGroupsClusterServerTickCallback() : **callback.h**
- emberAfGroupsClusterViewGroupCallback() : **callback.h**
- emberAfGroupsClusterViewGroupResponseCallback() : **callback.h**
- emberAfIasAceClusterArmCallback() : **callback.h**
- emberAfIasAceClusterArmResponseCallback() : **callback.h**
- emberAfIasAceClusterBypassCallback() : **callback.h**
- emberAfIasAceClusterClientAttributeChangedCallback() : **callback.h**
- emberAfIasAceClusterClientCommandParse() : **call-command-handler.h**
- emberAfIasAceClusterClientDefaultResponseCallback() : **callback.h**
- emberAfIasAceClusterClientInitCallback() : **callback.h**
- emberAfIasAceClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIasAceClusterClientMessageSentCallback() : **callback.h**
- emberAfIasAceClusterClientTickCallback() : **callback.h**
- emberAfIasAceClusterEmergencyCallback() : **callback.h**
- emberAfIasAceClusterFireCallback() : **callback.h**
- emberAfIasAceClusterGetZoneIdMapCallback() : **callback.h**
- emberAfIasAceClusterGetZoneIdMapResponseCallback() : **callback.h**
- emberAfIasAceClusterGetZoneInformationCallback() : **callback.h**
- emberAfIasAceClusterGetZoneInformationResponseCallback() : **callback.h**
- emberAfIasAceClusterPanicCallback() : **callback.h**
- emberAfIasAceClusterServerAttributeChangedCallback() : **callback.h**
- emberAfIasAceClusterServerCommandParse() : **call-command-handler.h**
- emberAfIasAceClusterServerDefaultResponseCallback() : **callback.h**
- emberAfIasAceClusterServerInitCallback() : **callback.h**
- emberAfIasAceClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIasAceClusterServerMessageSentCallback() : **callback.h**
- emberAfIasAceClusterServerTickCallback() : **callback.h**
- emberAfIasWdClusterClientAttributeChangedCallback() : **callback.h**
- emberAfIasWdClusterClientDefaultResponseCallback() : **callback.h**
- emberAfIasWdClusterClientInitCallback() : **callback.h**
- emberAfIasWdClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIasWdClusterClientMessageSentCallback() : **callback.h**
- emberAfIasWdClusterClientTickCallback() : **callback.h**
- emberAfIasWdClusterServerAttributeChangedCallback() : **callback.h**
- emberAfIasWdClusterServerCommandParse() : **call-command-handler.h**
- emberAfIasWdClusterServerDefaultResponseCallback() : **callback.h**
- emberAfIasWdClusterServerInitCallback() : **callback.h**
- emberAfIasWdClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIasWdClusterServerMessageSentCallback() : **callback.h**
- emberAfIasWdClusterServerTickCallback() : **callback.h**
- emberAfIasWdClusterSquawkCallback() : **callback.h**
- emberAfIasWdClusterStartWarningCallback() : **callback.h**
- emberAfIasZoneClusterClientAttributeChangedCallback() : **callback.h**
- emberAfIasZoneClusterClientCommandParse() : **call-command-handler.h**
- emberAfIasZoneClusterClientDefaultResponseCallback() : **callback.h**
- emberAfIasZoneClusterClientInitCallback() : **callback.h**
- emberAfIasZoneClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIasZoneClusterClientMessageSentCallback() : **callback.h**
- emberAfIasZoneClusterClientTickCallback() : **callback.h**
- emberAfIasZoneClusterServerAttributeChangedCallback() : **callback.h**
- emberAfIasZoneClusterServerCommandParse() : **call-command-handler.h**

- emberAfIasZoneClusterServerDefaultResponseCallback() : **callback.h**
- emberAfIasZoneClusterServerInitCallback() : **callback.h**
- emberAfIasZoneClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIasZoneClusterServerMessageSentCallback() : **callback.h**
- emberAfIasZoneClusterServerTickCallback() : **callback.h**
- emberAfIasZoneClusterZoneEnrollRequestCallback() : **callback.h**
- emberAfIasZoneClusterZoneEnrollResponseCallback() : **callback.h**
- emberAfIasZoneClusterZoneStatusChangeNotificationCallback() : **callback.h**
- emberAfIdentifyClusterClientAttributeChangedCallback() : **callback.h**
- emberAfIdentifyClusterClientCommandParse() : **call-command-handler.h**
- emberAfIdentifyClusterClientDefaultResponseCallback() : **callback.h**
- emberAfIdentifyClusterClientInitCallback() : **callback.h**
- emberAfIdentifyClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIdentifyClusterClientMessageSentCallback() : **callback.h**
- emberAfIdentifyClusterClientTickCallback() : **callback.h**
- emberAfIdentifyClusterIdentifyCallback() : **callback.h**
- emberAfIdentifyClusterIdentifyQueryCallback() : **callback.h**
- emberAfIdentifyClusterIdentifyQueryResponseCallback() : **callback.h**
- emberAfIdentifyClusterServerAttributeChangedCallback() : **callback.h**
- emberAfIdentifyClusterServerCommandParse() : **call-command-handler.h**
- emberAfIdentifyClusterServerDefaultResponseCallback() : **callback.h**
- emberAfIdentifyClusterServerInitCallback() : **callback.h**
- emberAfIdentifyClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIdentifyClusterServerMessageSentCallback() : **callback.h**
- emberAfIdentifyClusterServerTickCallback() : **callback.h**
- emberAfIllumLevelSensingClusterClientAttributeChangedCallback() : **callback.h**
- emberAfIllumLevelSensingClusterClientDefaultResponseCallback() : **callback.h**
- emberAfIllumLevelSensingClusterClientInitCallback() : **callback.h**
- emberAfIllumLevelSensingClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIllumLevelSensingClusterClientMessageSentCallback() : **callback.h**
- emberAfIllumLevelSensingClusterClientTickCallback() : **callback.h**
- emberAfIllumLevelSensingClusterServerAttributeChangedCallback() : **callback.h**
- emberAfIllumLevelSensingClusterServerDefaultResponseCallback() : **callback.h**
- emberAfIllumLevelSensingClusterServerInitCallback() : **callback.h**
- emberAfIllumLevelSensingClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIllumLevelSensingClusterServerMessageSentCallback() : **callback.h**
- emberAfIllumLevelSensingClusterServerTickCallback() : **callback.h**
- emberAfIllumMeasurementClusterClientAttributeChangedCallback() : **callback.h**
- emberAfIllumMeasurementClusterClientDefaultResponseCallback() : **callback.h**
- emberAfIllumMeasurementClusterClientInitCallback() : **callback.h**
- emberAfIllumMeasurementClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIllumMeasurementClusterClientMessageSentCallback() : **callback.h**
- emberAfIllumMeasurementClusterClientTickCallback() : **callback.h**
- emberAfIllumMeasurementClusterServerAttributeChangedCallback() : **callback.h**
- emberAfIllumMeasurementClusterServerDefaultResponseCallback() : **callback.h**
- emberAfIllumMeasurementClusterServerInitCallback() : **callback.h**
- emberAfIllumMeasurementClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfIllumMeasurementClusterServerMessageSentCallback() : **callback.h**
- emberAfIllumMeasurementClusterServerTickCallback() : **callback.h**
- emberAfIndexFromEndpoint() : **af.h**
- emberAfInitiateInterPanKeyEstablishment() : **af.h**
- emberAfInitiateInterPanKeyEstablishmentCallback() : **callback.h**
- emberAfInitiateKeyEstablishment() : **af.h**

- emberAfInitiateKeyEstablishmentCallback() : **callback.h**
- emberAfInitiatePartnerLinkKeyExchange() : **af.h**
- emberAfInitiatePartnerLinkKeyExchangeCallback() : **callback.h**
- emberAfInterpanInitCallback() : **callback.h**
- emberAfInterPanKeyEstablishmentCallback() : **callback.h**
- emberAfInterpanSendMessageCallback() : **callback.h**
- emberAfIsDeviceEnabled() : **af.h**
- emberAfIsDeviceIdentifying() : **identify.h**
- emberAfIsOtaImageIdValid() : **ota-storage.h**
- emberAfIsThisDataTypeAStringType() : **af.h**
- emberAfIsTypeSigned() : **af.h**
- emberAfJoinableNetworkFoundCallback() : **callback.h**
- emberAfJoinNetwork() : **af.h**
- emberAfKeyEstablishmentCallback() : **callback.h**
- emberAfKeyEstablishmentClusterClientAttributeChangedCallback() : **callback.h**
- emberAfKeyEstablishmentClusterClientCommandParse() : **call-command-handler.h**
- emberAfKeyEstablishmentClusterClientCommandReceivedCallback() : **callback.h**
- emberAfKeyEstablishmentClusterClientDefaultResponseCallback() : **callback.h**
- emberAfKeyEstablishmentClusterClientInitCallback() : **callback.h**
- emberAfKeyEstablishmentClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfKeyEstablishmentClusterClientMessageSentCallback() : **callback.h**
- emberAfKeyEstablishmentClusterClientTickCallback() : **callback.h**
- emberAfKeyEstablishmentClusterConfirmKeyDataRequestCallback() : **callback.h**
- emberAfKeyEstablishmentClusterConfirmKeyDataResponseCallback() : **callback.h**
- emberAfKeyEstablishmentClusterEphemeralDataRequestCallback() : **callback.h**
- emberAfKeyEstablishmentClusterEphemeralDataResponseCallback() : **callback.h**
- emberAfKeyEstablishmentClusterInitiateKeyEstablishmentRequestCallback() : **callback.h**
- emberAfKeyEstablishmentClusterInitiateKeyEstablishmentResponseCallback() : **callback.h**
- emberAfKeyEstablishmentClusterServerAttributeChangedCallback() : **callback.h**
- emberAfKeyEstablishmentClusterServerCommandParse() : **call-command-handler.h**
- emberAfKeyEstablishmentClusterServerCommandReceivedCallback() : **callback.h**
- emberAfKeyEstablishmentClusterServerDefaultResponseCallback() : **callback.h**
- emberAfKeyEstablishmentClusterServerInitCallback() : **callback.h**
- emberAfKeyEstablishmentClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfKeyEstablishmentClusterServerMessageSentCallback() : **callback.h**
- emberAfKeyEstablishmentClusterServerTickCallback() : **callback.h**
- emberAfKeyEstablishmentClusterTerminateKeyEstablishmentCallback() : **callback.h**
- emberAfLevelControlClusterClientAttributeChangedCallback() : **callback.h**
- emberAfLevelControlClusterClientDefaultResponseCallback() : **callback.h**
- emberAfLevelControlClusterClientInitCallback() : **callback.h**
- emberAfLevelControlClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfLevelControlClusterClientMessageSentCallback() : **callback.h**
- emberAfLevelControlClusterClientTickCallback() : **callback.h**
- emberAfLevelControlClusterMoveCallback() : **callback.h**
- emberAfLevelControlClusterMoveToLevelCallback() : **callback.h**
- emberAfLevelControlClusterMoveToLevelWithOnOffCallback() : **callback.h**
- emberAfLevelControlClusterMoveWithOnOffCallback() : **callback.h**
- emberAfLevelControlClusterServerAttributeChangedCallback() : **callback.h**
- emberAfLevelControlClusterServerCommandParse() : **call-command-handler.h**
- emberAfLevelControlClusterServerDefaultResponseCallback() : **callback.h**
- emberAfLevelControlClusterServerInitCallback() : **callback.h**
- emberAfLevelControlClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfLevelControlClusterServerMessageSentCallback() : **callback.h**

- emberAfLevelControlClusterServerTickCallback() : **callback.h**
- emberAfLevelControlClusterStepCallback() : **callback.h**
- emberAfLevelControlClusterStepWithOnOffCallback() : **callback.h**
- emberAfLevelControlClusterStopCallback() : **callback.h**
- emberAfLevelControlClusterStopWithOnOffCallback() : **callback.h**
- emberAfLocateAttributeMetadata() : **af.h**
- emberAfMainInitCallback() : **callback.h**
- emberAfMainTickCallback() : **callback.h**
- emberAfMaximumApsPayloadLength() : **af.h**
- emberAfMessageSentCallback() : **callback.h**
- emberAfMessagingClusterCancelMessageCallback() : **callback.h**
- emberAfMessagingClusterClientAttributeChangedCallback() : **callback.h**
- emberAfMessagingClusterClientCommandParse() : **call-command-handler.h**
- emberAfMessagingClusterClientDefaultResponseCallback() : **callback.h**
- emberAfMessagingClusterClientInitCallback() : **callback.h**
- emberAfMessagingClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfMessagingClusterClientMessageSentCallback() : **callback.h**
- emberAfMessagingClusterClientTickCallback() : **callback.h**
- emberAfMessagingClusterDisplayMessageCallback() : **callback.h**
- emberAfMessagingClusterGetLastMessageCallback() : **callback.h**
- emberAfMessagingClusterMessageConfirmationCallback() : **callback.h**
- emberAfMessagingClusterServerAttributeChangedCallback() : **callback.h**
- emberAfMessagingClusterServerCommandParse() : **call-command-handler.h**
- emberAfMessagingClusterServerDefaultResponseCallback() : **callback.h**
- emberAfMessagingClusterServerInitCallback() : **callback.h**
- emberAfMessagingClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfMessagingClusterServerMessageSentCallback() : **callback.h**
- emberAfMessagingClusterServerTickCallback() : **callback.h**
- emberAfMsToNextEvent() : **af.h**
- emberAfNcpInitCallback() : **callback.h**
- emberAfNetworkFindOperationCompleteCallback() : **callback.h**
- emberAfNextSequence() : **af.h**
- emberAfOccupancySensingClusterClientAttributeChangedCallback() : **callback.h**
- emberAfOccupancySensingClusterClientDefaultResponseCallback() : **callback.h**
- emberAfOccupancySensingClusterClientInitCallback() : **callback.h**
- emberAfOccupancySensingClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOccupancySensingClusterClientMessageSentCallback() : **callback.h**
- emberAfOccupancySensingClusterClientTickCallback() : **callback.h**
- emberAfOccupancySensingClusterServerAttributeChangedCallback() : **callback.h**
- emberAfOccupancySensingClusterServerDefaultResponseCallback() : **callback.h**
- emberAfOccupancySensingClusterServerInitCallback() : **callback.h**
- emberAfOccupancySensingClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOccupancySensingClusterServerMessageSentCallback() : **callback.h**
- emberAfOccupancySensingClusterServerTickCallback() : **callback.h**
- emberAfOkToHibernate() : **af.h**
- emberAfOnOffClusterClientAttributeChangedCallback() : **callback.h**
- emberAfOnOffClusterClientDefaultResponseCallback() : **callback.h**
- emberAfOnOffClusterClientInitCallback() : **callback.h**
- emberAfOnOffClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOnOffClusterClientMessageSentCallback() : **callback.h**
- emberAfOnOffClusterClientTickCallback() : **callback.h**
- emberAfOnOffClusterOffCallback() : **callback.h**
- emberAfOnOffClusterOnCallback() : **callback.h**

- emberAfOnOffClusterSampleMfgSpecificOffWithTransitionCallback() : **callback.h**
- emberAfOnOffClusterSampleMfgSpecificOnWithTransitionCallback() : **callback.h**
- emberAfOnOffClusterSampleMfgSpecificToggleWithTransitionCallback() : **callback.h**
- emberAfOnOffClusterServerAttributeChangedCallback() : **callback.h**
- emberAfOnOffClusterServerCommandParse() : **call-command-handler.h**
- emberAfOnOffClusterServerDefaultResponseCallback() : **callback.h**
- emberAfOnOffClusterServerInitCallback() : **callback.h**
- emberAfOnOffClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOnOffClusterServerMessageSentCallback() : **callback.h**
- emberAfOnOffClusterServerTickCallback() : **callback.h**
- emberAfOnOffClusterToggleCallback() : **callback.h**
- emberAfOnOffSetValue() : **on-off.h**
- emberAfOnOffSwitchConfigClusterClientAttributeChangedCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterClientDefaultResponseCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterClientInitCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterClientMessageSentCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterClientTickCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterServerAttributeChangedCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterServerDefaultResponseCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterServerInitCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterServerMessageSentCallback() : **callback.h**
- emberAfOnOffSwitchConfigClusterServerTickCallback() : **callback.h**
- emberAfOtaBootloadCallback() : **callback.h**
- emberAfOtaBootloadClusterClientAttributeChangedCallback() : **callback.h**
- emberAfOtaBootloadClusterClientDefaultResponseCallback() : **callback.h**
- emberAfOtaBootloadClusterClientInitCallback() : **callback.h**
- emberAfOtaBootloadClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOtaBootloadClusterClientMessageSentCallback() : **callback.h**
- emberAfOtaBootloadClusterClientTickCallback() : **callback.h**
- emberAfOtaBootloadClusterImageBlockRequestCallback() : **callback.h**
- emberAfOtaBootloadClusterImageBlockResponseCallback() : **callback.h**
- emberAfOtaBootloadClusterImageNotifyCallback() : **callback.h**
- emberAfOtaBootloadClusterImagePageRequestCallback() : **callback.h**
- emberAfOtaBootloadClusterQueryNextImageRequestCallback() : **callback.h**
- emberAfOtaBootloadClusterQueryNextImageResponseCallback() : **callback.h**
- emberAfOtaBootloadClusterQuerySpecificFileRequestCallback() : **callback.h**
- emberAfOtaBootloadClusterQuerySpecificFileResponseCallback() : **callback.h**
- emberAfOtaBootloadClusterServerAttributeChangedCallback() : **callback.h**
- emberAfOtaBootloadClusterServerDefaultResponseCallback() : **callback.h**
- emberAfOtaBootloadClusterServerInitCallback() : **callback.h**
- emberAfOtaBootloadClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfOtaBootloadClusterServerMessageSentCallback() : **callback.h**
- emberAfOtaBootloadClusterServerTickCallback() : **callback.h**
- emberAfOtaBootloadClusterUpgradeEndRequestCallback() : **callback.h**
- emberAfOtaBootloadClusterUpgradeEndResponseCallback() : **callback.h**
- emberAfOtaClientBootloadCallback() : **callback.h**
- emberAfOtaClientCustomVerifyCallback() : **callback.h**
- emberAfOtaClientDownloadCompleteCallback() : **callback.h**
- emberAfOtaClientIncomingMessageRawCallback() : **callback.h**
- emberAfOtaClientStartCallback() : **callback.h**
- emberAfOtaClientVersionInfoCallback() : **callback.h**

- emberAfOtaPageRequestServerPolicyCallback() : **callback.h**
- emberAfOtaServerBlockSizeCallback() : **callback.h**
- emberAfOtaServerImageBlockRequestCallback() : **callback.h**
- emberAfOtaServerIncomingMessageRawCallback() : **callback.h**
- emberAfOtaServerQueryCallback() : **callback.h**
- emberAfOtaServerSendImageNotifyCallback() : **callback.h**
- emberAfOtaServerSendUpgradeCommandCallback() : **ota-server.h**
- emberAfOtaServerUpgradeEndRequestCallback() : **callback.h**
- emberAfOtaStorageCheckTempDataCallback() : **callback.h**
- emberAfOtaStorageClearTempDataCallback() : **callback.h**
- emberAfOtaStorageCloseCallback() : **callback.h**
- emberAfOtaStorageDeleteImageCallback() : **ota-storage.h**
- emberAfOtaStorageDriverDownloadFinishCallback() : **ota-storage.h**
- emberAfOtaStorageDriverInitCallback() : **callback.h**
- emberAfOtaStorageDriverReadCallback() : **callback.h**
- emberAfOtaStorageDriverRetrieveLastStoredOffsetCallback() : **ota-storage.h**
- emberAfOtaStorageDriverWriteCallback() : **callback.h**
- emberAfOtaStorageFinishDownloadCallback() : **callback.h**
- emberAfOtaStorageGetCountCallback() : **callback.h**
- emberAfOtaStorageGetFullHeaderCallback() : **callback.h**
- emberAfOtaStorageGetTotalImageSizeCallback() : **callback.h**
- emberAfOtaStorageInitCallback() : **callback.h**
- emberAfOtaStorageIteratorFirstCallback() : **callback.h**
- emberAfOtaStorageIteratorNextCallback() : **callback.h**
- emberAfOtaStorageReadImageDataCallback() : **callback.h**
- emberAfOtaStorageSearchCallback() : **callback.h**
- emberAfOtaStorageWriteTempDataCallback() : **callback.h**
- emberAfPartnerLinkKeyExchangeRequestCallback() : **callback.h**
- emberAfPartnerLinkKeyExchangeResponseCallback() : **callback.h**
- emberAfPerformingKeyEstablishment() : **af.h**
- emberAfPerformingKeyEstablishmentCallback() : **callback.h**
- emberAfPluginBasicResetToFactoryDefaultsCallback() : **basic-callback.h**
- emberAfPluginDrlcEventActionCallback() : **drlc-callback.h**
- emberAfPluginMessagingClientCancelMessageCallback() : **messaging-client-callback.h**
- emberAfPluginMessagingClientConfirmMessage() : **messaging-client.h**
- emberAfPluginMessagingClientDisplayMessageCallback() : **messaging-client-callback.h**
- emberAfPluginPriceClientPriceExpiredCallback() : **price-client-callback.h**
- emberAfPluginPriceClientPriceStartedCallback() : **price-client-callback.h**
- emberAfPluginPslCommissioningEndpointInformationCallback() : **psl-commissioning-callback.h**
- emberAfPluginPslCommissioningEndpointInformationCountCallback() : **psl-commissioning-callback.h**
- emberAfPluginPslCommissioningGroupIdentifierCallback() : **psl-commissioning-callback.h**
- emberAfPluginPslCommissioningGroupIdentifierCountCallback() : **psl-commissioning-callback.h**
- emberAfPluginPslCommissioningIdentifyCallback() : **psl-commissioning-callback.h**
- emberAfPluginPslCommissioningTouchLinkCompleteCallback() : **psl-commissioning-callback.h**
- emberAfPluginPslCommissioningTouchLinkFailedCallback() : **psl-commissioning-callback.h**
- emberAfPluginSimpleMeteringClientRemoveMirrorCallback() : **simple-metering-client-callback.h**
- emberAfPluginSimpleMeteringClientRequestMirrorCallback() : **simple-metering-client-callback.h**
- emberAfPluginTestExampleCustomerFunctionCallback() : **test-callback.h**
- emberAfPluginTunnelingClientCloseTunnel() : **tunneling-client.h**
- emberAfPluginTunnelingClientDataErrorCallback() : **tunneling-client-callback.h**
- emberAfPluginTunnelingClientDataReceivedCallback() : **tunneling-client-callback.h**
- emberAfPluginTunnelingClientTransferData() : **tunneling-client.h**
- emberAfPluginTunnelingClientTunnelOpenedCallback() : **tunneling-client-callback.h**

- emberAfPluginTunnelingServerDataErrorCallback() : **tunneling-server-callback.h**
- emberAfPluginTunnelingServerDataReceivedCallback() : **tunneling-server-callback.h**
- emberAfPluginTunnelingServerIsProtocolSupportedCallback() : **tunneling-server-callback.h**
- emberAfPluginTunnelingServerTransferData() : **tunneling-server.h**
- emberAfPluginTunnelingServerTunnelClosedCallback() : **tunneling-server-callback.h**
- emberAfPluginTunnelingServerTunnelOpenedCallback() : **tunneling-server-callback.h**
- emberAfPollCompleteHandler() : **af.h**
- emberAfPostAttributeChangeCallback() : **callback.h**
- emberAfPostWakeUpCallback() : **callback.h**
- emberAfPowerConfigClusterClientAttributeChangedCallback() : **callback.h**
- emberAfPowerConfigClusterClientDefaultResponseCallback() : **callback.h**
- emberAfPowerConfigClusterClientInitCallback() : **callback.h**
- emberAfPowerConfigClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPowerConfigClusterClientMessageSentCallback() : **callback.h**
- emberAfPowerConfigClusterClientTickCallback() : **callback.h**
- emberAfPowerConfigClusterServerAttributeChangedCallback() : **callback.h**
- emberAfPowerConfigClusterServerDefaultResponseCallback() : **callback.h**
- emberAfPowerConfigClusterServerInitCallback() : **callback.h**
- emberAfPowerConfigClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPowerConfigClusterServerMessageSentCallback() : **callback.h**
- emberAfPowerConfigClusterServerTickCallback() : **callback.h**
- emberAfPreAttributeChangeCallback() : **callback.h**
- emberAfPreCommandReceivedCallback() : **callback.h**
- emberAfPreGoToSleepCallback() : **callback.h**
- emberAfPreMessageReceivedCallback() : **callback.h**
- emberAfPrepaymentClusterChangeSupplyCallback() : **callback.h**
- emberAfPrepaymentClusterClientAttributeChangedCallback() : **callback.h**
- emberAfPrepaymentClusterClientCommandParse() : **call-command-handler.h**
- emberAfPrepaymentClusterClientDefaultResponseCallback() : **callback.h**
- emberAfPrepaymentClusterClientInitCallback() : **callback.h**
- emberAfPrepaymentClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPrepaymentClusterClientMessageSentCallback() : **callback.h**
- emberAfPrepaymentClusterClientTickCallback() : **callback.h**
- emberAfPrepaymentClusterSelectAvailableEmergencyCreditCallback() : **callback.h**
- emberAfPrepaymentClusterServerAttributeChangedCallback() : **callback.h**
- emberAfPrepaymentClusterServerCommandParse() : **call-command-handler.h**
- emberAfPrepaymentClusterServerDefaultResponseCallback() : **callback.h**
- emberAfPrepaymentClusterServerInitCallback() : **callback.h**
- emberAfPrepaymentClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPrepaymentClusterServerMessageSentCallback() : **callback.h**
- emberAfPrepaymentClusterServerTickCallback() : **callback.h**
- emberAfPrepaymentClusterSupplyStatusResponseCallback() : **callback.h**
- emberAfPressureMeasurementClusterClientAttributeChangedCallback() : **callback.h**
- emberAfPressureMeasurementClusterClientDefaultResponseCallback() : **callback.h**
- emberAfPressureMeasurementClusterClientInitCallback() : **callback.h**
- emberAfPressureMeasurementClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPressureMeasurementClusterClientMessageSentCallback() : **callback.h**
- emberAfPressureMeasurementClusterClientTickCallback() : **callback.h**
- emberAfPressureMeasurementClusterServerAttributeChangedCallback() : **callback.h**
- emberAfPressureMeasurementClusterServerDefaultResponseCallback() : **callback.h**
- emberAfPressureMeasurementClusterServerInitCallback() : **callback.h**
- emberAfPressureMeasurementClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPressureMeasurementClusterServerMessageSentCallback() : **callback.h**

- emberAfPressureMeasurementClusterServerTickCallback() : **callback.h**
- emberAfPreZDOMessageReceivedCallback() : **callback.h**
- emberAfPriceClearPriceTable() : **price-server.h**
- emberAfPriceClusterClientAttributeChangedCallback() : **callback.h**
- emberAfPriceClusterClientCommandParse() : **call-command-handler.h**
- emberAfPriceClusterClientDefaultResponseCallback() : **callback.h**
- emberAfPriceClusterClientInitCallback() : **callback.h**
- emberAfPriceClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPriceClusterClientMessageSentCallback() : **callback.h**
- emberAfPriceClusterClientTickCallback() : **callback.h**
- emberAfPriceClusterGetBlockPeriodsCallback() : **callback.h**
- emberAfPriceClusterGetCurrentPriceCallback() : **callback.h**
- emberAfPriceClusterGetScheduledPricesCallback() : **callback.h**
- emberAfPriceClusterPriceAcknowledgementCallback() : **callback.h**
- emberAfPriceClusterPublishBlockPeriodCallback() : **callback.h**
- emberAfPriceClusterPublishPriceCallback() : **callback.h**
- emberAfPriceClusterServerAttributeChangedCallback() : **callback.h**
- emberAfPriceClusterServerCommandParse() : **call-command-handler.h**
- emberAfPriceClusterServerDefaultResponseCallback() : **callback.h**
- emberAfPriceClusterServerInitCallback() : **callback.h**
- emberAfPriceClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPriceClusterServerMessageSentCallback() : **callback.h**
- emberAfPriceClusterServerTickCallback() : **callback.h**
- emberAfPriceGetPriceTableEntry() : **price-server.h**
- emberAfPricePrintTable() : **price-server.h**
- emberAfPrint() : **af.h**
- emberAfPrintAllOff() : **af.h**
- emberAfPrintAllOn() : **af.h**
- emberAfPrintBigEndianEui64() : **af.h**
- emberAfPrintBuffer() : **af.h**
- emberAfPrintChannelListFromMask() : **network-cli.h**
- emberAfPrintEnabled() : **af.h**
- emberAfPrintLittleEndianEui64() : **af.h**
- emberAfPrintln() : **af.h**
- emberAfPrintMessageData() : **af.h**
- emberAfPrintOff() : **af.h**
- emberAfPrintOn() : **af.h**
- emberAfPrintStatus() : **af.h**
- emberAfPrintString() : **af.h**
- emberAfPrintTime() : **af.h**
- emberAfPslAbortTouchLink() : **psl-commissioning.h**
- emberAfPslDeviceInformationRequest() : **psl-commissioning.h**
- emberAfPslIdentifyRequest() : **psl-commissioning.h**
- emberAfPslInitiateTouchLink() : **psl-commissioning.h**
- emberAfPslResetToFactoryNew() : **psl-commissioning.h**
- emberAfPslResetToFactoryNewRequest() : **psl-commissioning.h**
- emberAfPslSetInitialSecurityState() : **psl-commissioning.h**
- emberAfPslTouchLinkInProgress() : **psl-commissioning.h**
- emberAfPumpConfigControlClusterClientAttributeChangedCallback() : **callback.h**
- emberAfPumpConfigControlClusterClientDefaultResponseCallback() : **callback.h**
- emberAfPumpConfigControlClusterClientInitCallback() : **callback.h**
- emberAfPumpConfigControlClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPumpConfigControlClusterClientMessageSentCallback() : **callback.h**

- emberAfPumpConfigControlClusterClientTickCallback() : **callback.h**
- emberAfPumpConfigControlClusterServerAttributeChangedCallback() : **callback.h**
- emberAfPumpConfigControlClusterServerDefaultResponseCallback() : **callback.h**
- emberAfPumpConfigControlClusterServerInitCallback() : **callback.h**
- emberAfPumpConfigControlClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfPumpConfigControlClusterServerMessageSentCallback() : **callback.h**
- emberAfPumpConfigControlClusterServerTickCallback() : **callback.h**
- emberAfReadAttribute() : **af.h**
- emberAfReadAttributesResponseCallback() : **callback.h**
- emberAfReadClientAttribute() : **af.h**
- emberAfReadManufacturerSpecificClientAttribute() : **af.h**
- emberAfReadManufacturerSpecificServerAttribute() : **af.h**
- emberAfReadOrUpdateAttribute() : **af.h**
- emberAfReadReportingConfigurationResponseCallback() : **callback.h**
- emberAfReadServerAttribute() : **af.h**
- emberAfRegistrationAbortCallback() : **callback.h**
- emberAfRegistrationCallback() : **callback.h**
- emberAfRegistrationStartCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterClientAttributeChangedCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterClientDefaultResponseCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterClientInitCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterClientMessageSentCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterClientTickCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterServerAttributeChangedCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterServerDefaultResponseCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterServerInitCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterServerMessageSentCallback() : **callback.h**
- emberAfRelativeHumidityMeasurementClusterServerTickCallback() : **callback.h**
- emberAfRemoveFromCurrentAppTasks() : **af.h**
- emberAfReportAttributesCallback() : **callback.h**
- emberAfReportEntryConfiguredCallback() : **callback.h**
- emberAfRssiLocationClusterAnchorNodeAnnounceCallback() : **callback.h**
- emberAfRssiLocationClusterClientAttributeChangedCallback() : **callback.h**
- emberAfRssiLocationClusterClientCommandParse() : **call-command-handler.h**
- emberAfRssiLocationClusterClientDefaultResponseCallback() : **callback.h**
- emberAfRssiLocationClusterClientInitCallback() : **callback.h**
- emberAfRssiLocationClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfRssiLocationClusterClientMessageSentCallback() : **callback.h**
- emberAfRssiLocationClusterClientTickCallback() : **callback.h**
- emberAfRssiLocationClusterCompactLocationDataNotificationCallback() : **callback.h**
- emberAfRssiLocationClusterDeviceConfigurationResponseCallback() : **callback.h**
- emberAfRssiLocationClusterGetDeviceConfigurationCallback() : **callback.h**
- emberAfRssiLocationClusterGetLocationDataCallback() : **callback.h**
- emberAfRssiLocationClusterLocationDataNotificationCallback() : **callback.h**
- emberAfRssiLocationClusterLocationDataResponseCallback() : **callback.h**
- emberAfRssiLocationClusterReportRssiMeasurementsCallback() : **callback.h**
- emberAfRssiLocationClusterRequestOwnLocationCallback() : **callback.h**
- emberAfRssiLocationClusterRssiPingCallback() : **callback.h**
- emberAfRssiLocationClusterRssiRequestCallback() : **callback.h**
- emberAfRssiLocationClusterRssiResponseCallback() : **callback.h**

- emberAfRssiLocationClusterSendPingsCallback() : **callback.h**
- emberAfRssiLocationClusterServerAttributeChangedCallback() : **callback.h**
- emberAfRssiLocationClusterServerCommandParse() : **call-command-handler.h**
- emberAfRssiLocationClusterServerDefaultResponseCallback() : **callback.h**
- emberAfRssiLocationClusterServerInitCallback() : **callback.h**
- emberAfRssiLocationClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfRssiLocationClusterServerMessageSentCallback() : **callback.h**
- emberAfRssiLocationClusterServerTickCallback() : **callback.h**
- emberAfRssiLocationClusterSetAbsoluteLocationCallback() : **callback.h**
- emberAfRssiLocationClusterSetDeviceConfigurationCallback() : **callback.h**
- emberAfRunEvents() : **af.h**
- emberAfSampleMfgSpecificClusterClientAttributeChangedCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterClientDefaultResponseCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterClientInitCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterClientMessageSentCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterClientTickCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterCommandOneCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterServerAttributeChangedCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterServerCommandParse() : **call-command-handler.h**
- emberAfSampleMfgSpecificClusterServerDefaultResponseCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterServerInitCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterServerMessageSentCallback() : **callback.h**
- emberAfSampleMfgSpecificClusterServerTickCallback() : **callback.h**
- emberAfScanErrorCallback() : **callback.h**
- emberAfScenesClusterAddSceneCallback() : **callback.h**
- emberAfScenesClusterAddSceneResponseCallback() : **callback.h**
- emberAfScenesClusterClearSceneTableCallback() : **callback.h**
- emberAfScenesClusterClientAttributeChangedCallback() : **callback.h**
- emberAfScenesClusterClientCommandParse() : **call-command-handler.h**
- emberAfScenesClusterClientDefaultResponseCallback() : **callback.h**
- emberAfScenesClusterClientInitCallback() : **callback.h**
- emberAfScenesClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfScenesClusterClientMessageSentCallback() : **callback.h**
- emberAfScenesClusterClientTickCallback() : **callback.h**
- emberAfScenesClusterGetSceneMembershipCallback() : **callback.h**
- emberAfScenesClusterGetSceneMembershipResponseCallback() : **callback.h**
- emberAfScenesClusterRecallSavedSceneCallback() : **callback.h**
- emberAfScenesClusterRecallSceneCallback() : **callback.h**
- emberAfScenesClusterRemoveAllScenesCallback() : **callback.h**
- emberAfScenesClusterRemoveAllScenesResponseCallback() : **callback.h**
- emberAfScenesClusterRemoveSceneCallback() : **callback.h**
- emberAfScenesClusterRemoveSceneResponseCallback() : **callback.h**
- emberAfScenesClusterServerAttributeChangedCallback() : **callback.h**
- emberAfScenesClusterServerCommandParse() : **call-command-handler.h**
- emberAfScenesClusterServerDefaultResponseCallback() : **callback.h**
- emberAfScenesClusterServerInitCallback() : **callback.h**
- emberAfScenesClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfScenesClusterServerMessageSentCallback() : **callback.h**
- emberAfScenesClusterServerTickCallback() : **callback.h**
- emberAfScenesClusterStoreCurrentSceneCallback() : **callback.h**
- emberAfScenesClusterStoreSceneCallback() : **callback.h**

- emberAfScenesClusterStoreSceneResponseCallback() : **callback.h**
- emberAfScenesClusterViewSceneCallback() : **callback.h**
- emberAfScenesClusterViewSceneResponseCallback() : **callback.h**
- emberAfScenesMakeInvalid() : **scenes.h**
- emberAfScenesMakeValid() : **scenes.h**
- emberAfScenesSetSceneCountAttribute() : **scenes.h**
- emberAfScheduleClusterTick() : **af.h**
- emberAfSendBroadcast() : **af.h**
- emberAfSendCommandBroadcast() : **af.h**
- emberAfSendCommandInterPan() : **af.h**
- emberAfSendCommandMulticast() : **af.h**
- emberAfSendCommandUnicast() : **af.h**
- emberAfSendDefaultResponse() : **af.h**
- emberAfSendEndDeviceBind() : **af.h**
- emberAfSendInterPan() : **af.h**
- emberAfSendMulticast() : **af.h**
- emberAfSendRequest() : **af.h**
- emberAfSendResponse() : **af.h**
- emberAfSendUnicast() : **af.h**
- emberAfSetAddressTableEntry() : **af.h**
- emberAfSetDefaultSleepControl() : **af.h**
- emberAfSetDeviceEnabled() : **af.h**
- emberAfSetExternalBuffer() : **client-api.h**
- emberAfSetTime() : **af.h**
- emberAfSetTimeCallback() : **callback.h**
- emberAfShadeConfigClusterClientAttributeChangedCallback() : **callback.h**
- emberAfShadeConfigClusterClientDefaultResponseCallback() : **callback.h**
- emberAfShadeConfigClusterClientInitCallback() : **callback.h**
- emberAfShadeConfigClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfShadeConfigClusterClientMessageSentCallback() : **callback.h**
- emberAfShadeConfigClusterClientTickCallback() : **callback.h**
- emberAfShadeConfigClusterServerAttributeChangedCallback() : **callback.h**
- emberAfShadeConfigClusterServerDefaultResponseCallback() : **callback.h**
- emberAfShadeConfigClusterServerInitCallback() : **callback.h**
- emberAfShadeConfigClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfShadeConfigClusterServerMessageSentCallback() : **callback.h**
- emberAfShadeConfigClusterServerTickCallback() : **callback.h**
- emberAfSimpleMeteringClusterClientAttributeChangedCallback() : **callback.h**
- emberAfSimpleMeteringClusterClientCommandParse() : **call-command-handler.h**
- emberAfSimpleMeteringClusterClientDefaultResponseCallback() : **callback.h**
- emberAfSimpleMeteringClusterClientInitCallback() : **callback.h**
- emberAfSimpleMeteringClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfSimpleMeteringClusterClientMessageSentCallback() : **callback.h**
- emberAfSimpleMeteringClusterClientTickCallback() : **callback.h**
- emberAfSimpleMeteringClusterGetProfileCallback() : **callback.h**
- emberAfSimpleMeteringClusterGetProfileResponseCallback() : **callback.h**
- emberAfSimpleMeteringClusterMirrorRemovedCallback() : **callback.h**
- emberAfSimpleMeteringClusterRemoveMirrorCallback() : **callback.h**
- emberAfSimpleMeteringClusterRequestFastPollModeCallback() : **callback.h**
- emberAfSimpleMeteringClusterRequestFastPollModeResponseCallback() : **callback.h**
- emberAfSimpleMeteringClusterRequestMirrorCallback() : **callback.h**
- emberAfSimpleMeteringClusterRequestMirrorResponseCallback() : **callback.h**
- emberAfSimpleMeteringClusterServerAttributeChangedCallback() : **callback.h**

- emberAfSimpleMeteringClusterServerCommandParse() : **call-command-handler.h**
- emberAfSimpleMeteringClusterServerDefaultResponseCallback() : **callback.h**
- emberAfSimpleMeteringClusterServerInitCallback() : **callback.h**
- emberAfSimpleMeteringClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfSimpleMeteringClusterServerMessageSentCallback() : **callback.h**
- emberAfSimpleMeteringClusterServerTickCallback() : **callback.h**
- emberAfStackStatusCallback() : **callback.h**
- emberAfStartMove() : **af.h**
- emberAfStartSearchForJoinableNetwork() : **af.h**
- emberAfStartSearchForJoinableNetworkCallback() : **callback.h**
- emberAfStringLength() : **af.h**
- emberAfTempMeasurementClusterClientAttributeChangedCallback() : **callback.h**
- emberAfTempMeasurementClusterClientDefaultResponseCallback() : **callback.h**
- emberAfTempMeasurementClusterClientInitCallback() : **callback.h**
- emberAfTempMeasurementClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfTempMeasurementClusterClientMessageSentCallback() : **callback.h**
- emberAfTempMeasurementClusterClientTickCallback() : **callback.h**
- emberAfTempMeasurementClusterConvertVoltsToCelsiusCallback() : **callback.h**
- emberAfTempMeasurementClusterServerAttributeChangedCallback() : **callback.h**
- emberAfTempMeasurementClusterServerDefaultResponseCallback() : **callback.h**
- emberAfTempMeasurementClusterServerInitCallback() : **callback.h**
- emberAfTempMeasurementClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfTempMeasurementClusterServerMessageSentCallback() : **callback.h**
- emberAfTempMeasurementClusterServerTickCallback() : **callback.h**
- emberAfThermostatClusterClientAttributeChangedCallback() : **callback.h**
- emberAfThermostatClusterClientDefaultResponseCallback() : **callback.h**
- emberAfThermostatClusterClientInitCallback() : **callback.h**
- emberAfThermostatClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfThermostatClusterClientMessageSentCallback() : **callback.h**
- emberAfThermostatClusterClientTickCallback() : **callback.h**
- emberAfThermostatClusterServerAttributeChangedCallback() : **callback.h**
- emberAfThermostatClusterServerCommandParse() : **call-command-handler.h**
- emberAfThermostatClusterServerDefaultResponseCallback() : **callback.h**
- emberAfThermostatClusterServerInitCallback() : **callback.h**
- emberAfThermostatClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfThermostatClusterServerMessageSentCallback() : **callback.h**
- emberAfThermostatClusterServerTickCallback() : **callback.h**
- emberAfThermostatClusterSetpointRaiseLowerCallback() : **callback.h**
- emberAfThermostatUiConfigClusterClientAttributeChangedCallback() : **callback.h**
- emberAfThermostatUiConfigClusterClientDefaultResponseCallback() : **callback.h**
- emberAfThermostatUiConfigClusterClientInitCallback() : **callback.h**
- emberAfThermostatUiConfigClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfThermostatUiConfigClusterClientMessageSentCallback() : **callback.h**
- emberAfThermostatUiConfigClusterClientTickCallback() : **callback.h**
- emberAfThermostatUiConfigClusterServerAttributeChangedCallback() : **callback.h**
- emberAfThermostatUiConfigClusterServerDefaultResponseCallback() : **callback.h**
- emberAfThermostatUiConfigClusterServerInitCallback() : **callback.h**
- emberAfThermostatUiConfigClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfThermostatUiConfigClusterServerMessageSentCallback() : **callback.h**
- emberAfThermostatUiConfigClusterServerTickCallback() : **callback.h**
- emberAfTimeClusterClientAttributeChangedCallback() : **callback.h**
- emberAfTimeClusterClientDefaultResponseCallback() : **callback.h**
- emberAfTimeClusterClientInitCallback() : **callback.h**

- emberAfTimeClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfTimeClusterClientMessageSentCallback() : **callback.h**
- emberAfTimeClusterClientTickCallback() : **callback.h**
- emberAfTimeClusterServerAttributeChangedCallback() : **callback.h**
- emberAfTimeClusterServerDefaultResponseCallback() : **callback.h**
- emberAfTimeClusterServerInitCallback() : **callback.h**
- emberAfTimeClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfTimeClusterServerMessageSentCallback() : **callback.h**
- emberAfTimeClusterServerTickCallback() : **callback.h**
- emberAfTrustCenterJoinCallback() : **callback.h**
- emberAfTrustCenterKeepaliveAbortCallback() : **callback.h**
- emberAfTrustCenterKeepaliveUpdateCallback() : **callback.h**
- emberAfTunnelingClusterAckTransferDataClientToServerCallback() : **callback.h**
- emberAfTunnelingClusterAckTransferDataServerToClientCallback() : **callback.h**
- emberAfTunnelingClusterClientAttributeChangedCallback() : **callback.h**
- emberAfTunnelingClusterClientCommandParse() : **call-command-handler.h**
- emberAfTunnelingClusterClientDefaultResponseCallback() : **callback.h**
- emberAfTunnelingClusterClientInitCallback() : **callback.h**
- emberAfTunnelingClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfTunnelingClusterClientMessageSentCallback() : **callback.h**
- emberAfTunnelingClusterClientTickCallback() : **callback.h**
- emberAfTunnelingClusterCloseTunnelCallback() : **callback.h**
- emberAfTunnelingClusterReadyDataClientToServerCallback() : **callback.h**
- emberAfTunnelingClusterReadyDataServerToClientCallback() : **callback.h**
- emberAfTunnelingClusterRequestTunnelCallback() : **callback.h**
- emberAfTunnelingClusterRequestTunnelResponseCallback() : **callback.h**
- emberAfTunnelingClusterServerAttributeChangedCallback() : **callback.h**
- emberAfTunnelingClusterServerCommandParse() : **call-command-handler.h**
- emberAfTunnelingClusterServerDefaultResponseCallback() : **callback.h**
- emberAfTunnelingClusterServerInitCallback() : **callback.h**
- emberAfTunnelingClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfTunnelingClusterServerMessageSentCallback() : **callback.h**
- emberAfTunnelingClusterServerTickCallback() : **callback.h**
- emberAfTunnelingClusterTransferDataClientToServerCallback() : **callback.h**
- emberAfTunnelingClusterTransferDataErrorClientToServerCallback() : **callback.h**
- emberAfTunnelingClusterTransferDataErrorServerToClientCallback() : **callback.h**
- emberAfTunnelingClusterTransferDataServerToClientCallback() : **callback.h**
- emberAfUnusedPanIdFoundCallback() : **callback.h**
- emberAfVerifyAttributeWrite() : **af.h**
- emberAfWindowCoveringClusterClientAttributeChangedCallback() : **callback.h**
- emberAfWindowCoveringClusterClientDefaultResponseCallback() : **callback.h**
- emberAfWindowCoveringClusterClientInitCallback() : **callback.h**
- emberAfWindowCoveringClusterClientManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfWindowCoveringClusterClientMessageSentCallback() : **callback.h**
- emberAfWindowCoveringClusterClientTickCallback() : **callback.h**
- emberAfWindowCoveringClusterServerAttributeChangedCallback() : **callback.h**
- emberAfWindowCoveringClusterServerCommandParse() : **call-command-handler.h**
- emberAfWindowCoveringClusterServerDefaultResponseCallback() : **callback.h**
- emberAfWindowCoveringClusterServerInitCallback() : **callback.h**
- emberAfWindowCoveringClusterServerManufacturerSpecificAttributeChangedCallback() : **callback.h**
- emberAfWindowCoveringClusterServerMessageSentCallback() : **callback.h**
- emberAfWindowCoveringClusterServerTickCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringDownCloseCallback() : **callback.h**

- emberAfWindowCoveringClusterWindowCoveringGoToLiftPercentageCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringGoToLiftSetpointCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringGoToLiftValueCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringGoToProgramSetpointCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringGoToTiltPercentageCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringGoToTiltSetpointCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringGoToTiltValueCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringStopCallback() : **callback.h**
- emberAfWindowCoveringClusterWindowCoveringUpOpenCallback() : **callback.h**
- emberAfWriteAttribute() : **af.h**
- emberAfWriteAttributesResponseCallback() : **callback.h**
- emberAfWriteClientAttribute() : **af.h**
- emberAfWriteManufacturerSpecificClientAttribute() : **af.h**
- emberAfWriteManufacturerSpecificServerAttribute() : **af.h**
- emberAfWriteServerAttribute() : **af.h**
- enroll() : **ias-zone-table.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- a
- b
- c
- e
- g
- i
- k
- o
- p
- r
- s
- u

- g -

- getNextZoneId() : **ias-zone-table.h**
- getSmacPointer() : **key-establishment-storage.h**
- getZoneIdMap() : **ias-zone-table.h**
- getZoneInfo() : **ias-zone-table.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- a
- b
- c
- e
- g
- i
- k
- o
- p
- r
- s
- u

- i -

- initZoneTable() : **ias-zone-table.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- a
- b
- c
- e
- g
- i
- k
- o
- p
- r
- s
- u

- k -

- keyUpdateCommand() : **security-cli.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- a
- b
- c
- e
- g
- i
- k
- o
- p
- r
- s
- u

- o -

- otaFindServerCommand() : **ota-cli.h**
- otaImageNotifyCommand() : **ota-cli.h**
- otaQueryServerCommand() : **ota-cli.h**
- otaQuerySpecificFileCommand() : **ota-cli.h**
- otaSendUpgradeCommand() : **ota-cli.h**
- otaUsePageRequestCommand() : **ota-cli.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- a
- b
- c
- e
- g
- i
- k
- o
- p
- r
- s
- u

- p -

- pageRequestTest() : **ota-client-page-request-test.h**
- printKeyInfo() : **security-cli.h**
- printZoneIdMap() : **ias-zone-table.h**
- printZoneInfo() : **ias-zone-table.h**
- printZoneTable() : **ias-zone-table.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- a
- b
- c
- e
- g
- i
- k
- o
- p
- r
- s
- u

- r -

- remainingAllocations() : **ota-storage.h**
- retrieveAndClearPublicPartnerData() : **key-establishment-storage.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- a
- b
- c
- e
- g
- i
- k
- o
- p
- r
- s
- u

- s -

- storePublicPartnerData() : **key-establishment-storage.h**
- storeSmac() : **key-establishment-storage.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- a
- b
- c
- e
- g
- i
- k
- o
- p
- r
- s
- u

- u -

- unenroll() : **ias-zone-table.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- b
- e
- g
- i
- k
- l
- m
- n
- o
- p
- r
- t
- z

- e -

- EM_AF_NO_PAGE_REQUEST : **ota-client-page-request.h**
- EM_AF_PAGE_REQUEST_COMPLETE : **ota-client-page-request.h**
- EM_AF_PAGE_REQUEST_ERROR : **ota-client-page-request.h**
- EM_AF_RETRY_MISSED_PACKETS : **ota-client-page-request.h**
- EM_AF_WAITING_PAGE_REQUEST_REPLIES : **ota-client-page-request.h**
- EMBER_AF_BROADCAST_SERVICE_DISCOVERY_COMPLETE : **af-types.h**
- EMBER_AF_BROADCAST_SERVICE_DISCOVERY_RESPONSE_RECEIVED : **af-types.h**
- EMBER_AF_DATA_TYPE_ANALOG : **af.h**
- EMBER_AF_DATA_TYPE_DISCRETE : **af.h**
- EMBER_AF_DATA_TYPE_NONE : **af.h**
- EMBER_AF_IMAGE_BAD : **af-types.h**
- EMBER_AF_IMAGE_GOOD : **af-types.h**
- EMBER_AF_IMAGE_VERIFY_IN_PROGRESS : **af-types.h**
- EMBER_AF_LAST_POLL_GOT_DATA : **af.h**
- EMBER_AF_OK_TO_HIBERNATE : **af-types.h**
- EMBER_AF_OK_TO_NAP : **af-types.h**
- EMBER_AF_OTA_CLIENT_ABORTED : **ota-client-policy.h**
- EMBER_AF_OTA_DOWNLOAD_AND_VERIFY_SUCCESS : **ota-client-policy.h**
- EMBER_AF_OTA_DOWNLOAD_TIME_OUT : **ota-client-policy.h**
- EMBER_AF_OTA_SERVER_ABORTED : **ota-client-policy.h**
- EMBER_AF_OTA_STORAGE_ERROR : **af-types.h**
- EMBER_AF_OTA_STORAGE_PARTIAL_FILE_FOUND : **af-types.h**
- EMBER_AF_OTA_STORAGE_RETURN_DATA_TOO_LONG : **af-types.h**
- EMBER_AF_OTA_STORAGE_SUCCESS : **af-types.h**
- EMBER_AF_OTA_VERIFY_FAILED : **ota-client-policy.h**
- EMBER_AF_STAY_AWAKE : **af-types.h**
- EMBER_AF_UNICAST_SERVICE_DISCOVERY_COMPLETE_WITH_RESPONSE : **af-types.h**
- EMBER_AF_UNICAST_SERVICE_DISCOVERY_TIMEOUT : **af-types.h**
- EMBER_AF_WAITING_FOR_DATA_ACK : **af.h**
- EMBER_AF_WAITING_FOR_PARTNER_LINK_KEY_EXCHANGE : **af.h**

- EMBER_AF_WAITING_FOR_REGISTRATION : **af.h**
- EMBER_AF_WAITING_FOR_SERVICE_DISCOVERY : **af.h**
- EMBER_AF_WAITING_FOR_ZCL_RESPONSE : **af.h**
- EMBER_AF_WAITING_FOR_ZDO_RESPONSE : **af.h**
- EMBER_ZCL_ALTERNATE_COST_UNIT_KG_OF_CO2_PER_UNIT_OF_MEASURE : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_1 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_2 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_3 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_4 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_5 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_EMERGENCY : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_GREEN : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_PLANNED_OUTAGE : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_RESERVED : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_SERVICE_DISCONNECT : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED1 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED2 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED3 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED4 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED5 : **enums.h**
- EMBER_ZCL_AMI_CRITICALITY_LEVEL_UTILITY_DEFINED6 : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_EVENT_COMPLETED : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_EVENT_COMPLETED_NO_USER_PARTICIPATION_PREVIOUS_OPT_OUT : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_EVENT_PARTIALLY_COMPLETED_DUE_TO_USER_OPT_IN : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_EVENT_PARTIALLY_COMPLETED_WITH_USER_OPT_OUT : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_EVENT_STARTED : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_LOAD_CONTROL_EVENT_COMMAND_REJECTED : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_LOAD_CONTROL_EVENT_COMMAND_RX : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_EVENT_EXPIRED : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_COMMAND : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_COMMAND_INVALID_EFFECTIVE_TIME : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_REJECTED_INVALID_CANCEL_UNDEFINED_EVENT : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_THE_EVENT_HAS_BEEN_CANCELED : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_THE_EVENT_HAS_BEEN_SUPERSEDED : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_USER_HAS_CHOOSE_TO_OPT_IN : **enums.h**
- EMBER_ZCL_AMI_EVENT_STATUS_USER_HAS_CHOOSE_TO_OPT_OUT : **enums.h**
- EMBER_ZCL_AMI_GET_PROFILE_STATUS_INTERVAL_CHANNEL_NOT_SUPPORTED : **enums.h**
- EMBER_ZCL_AMI_GET_PROFILE_STATUS_INVALID_END_TIME : **enums.h**
- EMBER_ZCL_AMI_GET_PROFILE_STATUS_MORE_PERIODS_REQUESTED_THAN_CAN_BE_RETURNED : **enums.h**
- EMBER_ZCL_AMI_GET_PROFILE_STATUS_NO_INTERVALS_AVAILABLE_FOR_THE_REQUESTED_TIME : **enums.h**
- EMBER_ZCL_AMI_GET_PROFILE_STATUS_SUCCESS : **enums.h**
- EMBER_ZCL_AMI_GET_PROFILE_STATUS_UNDEFINED_INTERVAL_CHANNEL_REQUESTED : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_CHANNEL_CONSUMPTION_DELIVERED : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_CHANNEL_CONSUMPTION_RECEIVED : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_DAILY : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES10 : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES15 : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES2P5 : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES30 : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES5 : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES60 : **enums.h**
- EMBER_ZCL_AMI_INTERVAL_PERIOD_MINUTES7P5 : **enums.h**
- EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_BAD_KEY_CONFIRM : **enums.h**
- EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_BAD_MESSAGE : **enums.h**
- EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_NO_RESOURCES : **enums.h**

- EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_SUCCESS : **enums.h**
- EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_UNKNOWN_ISSUER : **enums.h**
- EMBER_ZCL_AMI_KEY_ESTABLISHMENT_STATUS_UNSUPPORTED_SUITE : **enums.h**
- EMBER_ZCL_AMI_REGISTRATION_STATE_JOINED_NETWORK : **enums.h**
- EMBER_ZCL_AMI_REGISTRATION_STATE_JOINING_NETWORK : **enums.h**
- EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTERATION_NOT_POSSIBLE : **enums.h**
- EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTERED : **enums.h**
- EMBER_ZCL_AMI_REGISTRATION_STATE_REGISTRATION_REJECTED : **enums.h**
- EMBER_ZCL_AMI_REGISTRATION_STATE_SUBMITTED_REGISTRATION_REQUEST : **enums.h**
- EMBER_ZCL_AMI_REGISTRATION_STATE_UNREGISTERED : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_BT_US_OR_BTU_PER_HOUR : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_BT_US_OR_BTU_PER_HOUR_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_CENTUM_CUBIC_FEET_PER_HOUR : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_CENTUM_CUBIC_FEET_PER_HOUR_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_FEET_PER_HOUR : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_FEET_PER_HOUR_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_METER_PER_HOUR : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_CUBIC_METER_PER_HOUR_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_IMPERIAL_GALLONS_PER_HOUR : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_IMPERIAL_GALLONS_PER_HOUR_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_KILO_WATT_HOURS : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_KILO_WATT_HOURS_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_ABSOLUTE : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_ABSOLUTE_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_GAUGE : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_KPA_GUAGE_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_LITERS_OR_LITERS_PER_HOUR : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_LITERS_OR_LITERS_PER_HOUR_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_MCF_OR_MCF_PER_SECOND : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_MCF_OR_MCF_PER_SECOND_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_MJ_OR_MJ_PER_SECOND : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_MJ_OR_MJ_PER_SECOND_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_UNITLESS : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_UNITLESS_BCD : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_US_GALLONS_PER_HOUR : **enums.h**
- EMBER_ZCL_AMI_UNIT_OF_MEASURE_US_GALLONS_PER_HOUR_BCD : **enums.h**
- EMBER_ZCL_ANONYMOUS_DATA_STATE_NO_SOURCE_FOUND : **enums.h**
- EMBER_ZCL_ANONYMOUS_DATA_STATE_SOURCE_FOUND : **enums.h**
- EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_ALLOW_WRITE_NORMAL : **enums.h**
- EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_ALLOW_WRITE_OF_READ_ONLY : **enums.h**
- EMBER_ZCL_ATTRIBUTE_WRITE_PERMISSION_DENY_WRITE : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_AA : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_AAA : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_BUILT_IN : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_C : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_D : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_NO_BATTERY : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_OTHER : **enums.h**
- EMBER_ZCL_BATTERY_SIZE_UNKNOWN : **enums.h**
- EMBER_ZCL_COLOR_MODE_COLOR_TEMPERATURE : **enums.h**
- EMBER_ZCL_COLOR_MODE_CURRENT_HUE_AND_CURRENT_SATURATION : **enums.h**
- EMBER_ZCL_COLOR_MODE_CURRENT_X_AND_CURRENT_Y : **enums.h**
- EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_FORM_NETWORK : **enums.h**
- EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_NO_ACTION : **enums.h**
- EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_REJOIN_NETWORK : **enums.h**
- EMBER_ZCL_COMMISSIONING_STARTUP_CONTROL_START_FROM_SCRATCH : **enums.h**
- EMBER_ZCL_DEHUMIDIFCATION_LOCKOUT_ALLOWED : **enums.h**

- EMBER_ZCL_DEHUMIDIFCATION_LOCKOUT_NOT_ALLOWED : **enums.h**
- EMBER_ZCL_FAN_MODE_AUTO : **enums.h**
- EMBER_ZCL_FAN_MODE_HIGH : **enums.h**
- EMBER_ZCL_FAN_MODE_LOW : **enums.h**
- EMBER_ZCL_FAN_MODE_MEDIUM : **enums.h**
- EMBER_ZCL_FAN_MODE_OFF : **enums.h**
- EMBER_ZCL_FAN_MODE_ON : **enums.h**
- EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_HIGH : **enums.h**
- EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_HIGH_AUTO : **enums.h**
- EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_MED_HIGH : **enums.h**
- EMBER_ZCL_FAN_MODE_SEQUENCE_LOW_MED_HIGH_AUTO : **enums.h**
- EMBER_ZCL_FAN_MODE_SEQUENCE_ON_AUTO : **enums.h**
- EMBER_ZCL_FAN_MODE_SMART : **enums.h**
- EMBER_ZCL_HUE_DIRECTION_DOWN : **enums.h**
- EMBER_ZCL_HUE_DIRECTION_LONGEST_DISTANCE : **enums.h**
- EMBER_ZCL_HUE_DIRECTION_SHORTEST_DISTANCE : **enums.h**
- EMBER_ZCL_HUE_DIRECTION_UP : **enums.h**
- EMBER_ZCL_HUE_MOVE_MODE_DOWN : **enums.h**
- EMBER_ZCL_HUE_MOVE_MODE_STOP : **enums.h**
- EMBER_ZCL_HUE_MOVE_MODE_UP : **enums.h**
- EMBER_ZCL_HUE_STEP_MODE_DOWN : **enums.h**
- EMBER_ZCL_HUE_STEP_MODE_UP : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_ALL_ZONES : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_DAY_HOME_ZONES_ONLY : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_MODE_ARM_NIGHT_SLEEP_ZONES_ONLY : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_MODE_DISARM : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ALL_ZONES_ARMED : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ALL_ZONES_DISARMED : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ONLY_DAY_HOME_ZONES_ARMED : **enums.h**
- EMBER_ZCL_IAS_ACE_ARM_NOTIFICATION_ONLY_NIGHT_SLEEP_ZONES_ARMED : **enums.h**
- EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_NO_ENROLL_PERMIT : **enums.h**
- EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_NOT_SUPPORTED : **enums.h**
- EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_SUCCESS : **enums.h**
- EMBER_ZCL_IAS_ENROLL_RESPONSE_CODE_TOO_MANY_ZONES : **enums.h**
- EMBER_ZCL_IAS_ZONE_STATE_ENROLLED : **enums.h**
- EMBER_ZCL_IAS_ZONE_STATE_NOT_ENROLLED : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_CONTACT_SWITCH : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_FIRE_SENSOR : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_GAS_SENSOR : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_KEY_FOB : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_KEYPAD : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_MOTION_SENSOR : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_PERSONAL_EMERGENCY_DEVICE : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_REMOTE_CONTROL : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_STANDARD_CIE : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_STANDARD_WARNING_DEVICE : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_VIBRATION_MOVEMENT_SENSOR : **enums.h**
- EMBER_ZCL_IAS_ZONE_TYPE_WATER_SENSOR : **enums.h**
- EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_FOUR_LOCKOUT : **enums.h**
- EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_ONE_LOCKOUT : **enums.h**
- EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_THREE_LOCKOUT : **enums.h**
- EMBER_ZCL_KEYPAD_LOCKOUT_LEVEL_TWO_LOCKOUT : **enums.h**
- EMBER_ZCL_KEYPAD_LOCKOUT_LEVELFIVE_LOCKOUT : **enums.h**
- EMBER_ZCL_KEYPAD_LOCKOUT_NO_LOCKOUT : **enums.h**
- EMBER_ZCL_LEVEL_STATUS_ABOVE_TARGET : **enums.h**
- EMBER_ZCL_LEVEL_STATUS_BELOW_TARGET : **enums.h**
- EMBER_ZCL_LEVEL_STATUS_ON_TARGET : **enums.h**

- EMBER_ZCL_LOCATION_METHOD_LATERATION : **enums.h**
- EMBER_ZCL_LOCATION_METHOD_OUT_OF_BAND : **enums.h**
- EMBER_ZCL_LOCATION_METHOD_RF_FINGERPRINTING : **enums.h**
- EMBER_ZCL_LOCATION_METHOD_SIGNPOSTING : **enums.h**
- EMBER_ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_CMOS : **enums.h**
- EMBER_ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_PHOTODIODE : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_CONFIRMATION_NOT_REQUIRED : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_CONFIRMATION_REQUIRED : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_CRITICAL : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_HIGH : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_LOW : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_IMPORTANCE_MEDIUM : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_ANONYMOUS : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_NORMAL : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_NORMAL_AND_ANONYMOUS : **enums.h**
- EMBER_ZCL_MESSAGING_CONTROL_TRANSMISSION_RESERVED : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_COOLING_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_ELECTRIC_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_GAS_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_HEAT_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_COOLING_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_GAS_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_HEAT_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_PRESSURE_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_THERMAL_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_MIRRORED_WATER_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_PRESSURE_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_THERMAL_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_UNDEFINED_MIRROR_METER : **enums.h**
- EMBER_ZCL_METER_DEVICE_TYPE_WATER_METER : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_BURST_DETECT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_CHECK_METER : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_ELECTRIC_TAMPER_DETECT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_AIR_DETECT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_COMMUNICATION_ERROR : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_MEASUREMENT_FAULT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FLOW_SENSOR_REVERSE_FLOW : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L1 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L2 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_HIGH_L3 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L1 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L2 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_FREQUENCY_TOO_LOW_L3 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_GROUND_FAULT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L1 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L2 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_HIGH_VOLTAGE_L3 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_INLET_TEMPERATURE_SENSOR_FAULT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_LEAK_DETECT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_LOW_BATTERY : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L1 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L2 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_LOW_VOLTAGE_L3 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_OUTLET_TEMPERATURE_SENSOR_FAULT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L1 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L2 : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_OVER_CURRENT_L3 : **enums.h**

- EMBER_ZCL_METERING_ALARM_CODE_PIPE_EMPTY : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_POWER_FAILURE_PIPE_EMPTY_TEMPERATURE_SENSOR : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_POWER_QUALITY_LOW_PRESSURE_BURST_DETECT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_PRESSURE_TOO_HIGH : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_PRESSURE_TOO_LOW : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_REVERSE_FLOW_FLOW_SENSOR : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_SERVICE_DISCONNECT : **enums.h**
- EMBER_ZCL_METERING_ALARM_CODE_TAMPER_DETECT : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK1 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK10 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK11 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK12 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK13 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK14 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK15 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK16 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK2 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK3 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK4 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK5 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK6 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK7 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK8 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_BLOCK9 : **enums.h**
- EMBER_ZCL_METERING_BLOCK_ENUMERATIONS_NO_BLOCKS_IN_USE : **enums.h**
- EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_OFF : **enums.h**
- EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_OFF_ARMED : **enums.h**
- EMBER_ZCL_METERING_SUPPLY_STATUS_SUPPLY_ON : **enums.h**
- EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_CELSIUS : **enums.h**
- EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_CELSIUS_BCD : **enums.h**
- EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_FAHRENHEIT : **enums.h**
- EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_FAHRENHEIT_BCD : **enums.h**
- EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_KELVIN : **enums.h**
- EMBER_ZCL_METERING_TEMPERATURE_UNIT_OF_MEASURE_KELVIN_BCD : **enums.h**
- EMBER_ZCL_MOVE_MODE_DOWN : **enums.h**
- EMBER_ZCL_MOVE_MODE_UP : **enums.h**
- EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_PIR : **enums.h**
- EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_PIR_AND_ULTRASONIC : **enums.h**
- EMBER_ZCL_OCCUPANCY_SENSOR_TYPE_ULTRASONIC : **enums.h**
- EMBER_ZCL_OPERATING_MODE_CONFIGURE : **enums.h**
- EMBER_ZCL_OPERATING_MODE_NORMAL : **enums.h**
- EMBER_ZCL_PHYSICAL_ENVIRONMENT_FIRST_PROFILE_SPECIFIED_VALUE : **enums.h**
- EMBER_ZCL_PHYSICAL_ENVIRONMENT_LAST_PROFILE_SPECIFIED_VALUE : **enums.h**
- EMBER_ZCL_PHYSICAL_ENVIRONMENT_UNKNOWN : **enums.h**
- EMBER_ZCL_PHYSICAL_ENVIRONMENT_UNSPECIFIED : **enums.h**
- EMBER_ZCL_POWER_SOURCE_BATTERY : **enums.h**
- EMBER_ZCL_POWER_SOURCE_BATTERY_BACKUP : **enums.h**
- EMBER_ZCL_POWER_SOURCE_DC_SOURCE : **enums.h**
- EMBER_ZCL_POWER_SOURCE_EMERGENCY_MAINS_CONSTANT_POWER : **enums.h**
- EMBER_ZCL_POWER_SOURCE_EMERGENCY_MAINS_TRANSFER_SWITCH : **enums.h**
- EMBER_ZCL_POWER_SOURCE_SINGLE_PHASE_MAINS : **enums.h**
- EMBER_ZCL_POWER_SOURCE_THREE_PHASE_MAINS : **enums.h**
- EMBER_ZCL_POWER_SOURCE_UNKNOWN : **enums.h**
- EMBER_ZCL_PREPAYMENT_ORIGINATING_DEVICE_ENERGY_SERVICE_INTERFACE : **enums.h**
- EMBER_ZCL_PREPAYMENT_ORIGINATING_DEVICE_IN_PREMISES_DISPLAY_DEVICE : **enums.h**
- EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_DAY : **enums.h**
- EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_HOUR : **enums.h**

- EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_MONTH : **enums.h**
- EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_QUARTER : **enums.h**
- EMBER_ZCL_PREPAYMENT_RECOVERY_PERIOD_PER_WEEK : **enums.h**
- EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_OFF : **enums.h**
- EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_OFF_ARMED : **enums.h**
- EMBER_ZCL_PREPAYMENT_SUPPLY_STATUS_SUPPLY_ON : **enums.h**
- EMBER_ZCL_PRICE_CONTROL_ACKNOWLEDGEMENT_NOT_REQUIRED : **enums.h**
- EMBER_ZCL_PRICE_CONTROL_ACKNOWLEDGEMENT_REQUIRED : **enums.h**
- EMBER_ZCL_PRICE_TIER_NO_TIER_RELATED : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER10_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER11_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER12_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER13_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER14_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER15_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER1_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER2_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER3_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER4_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER5_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER6_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER7_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER8_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PRICE_TIER_TIER9_PRICE_LABEL : **enums.h**
- EMBER_ZCL_PUMP_CONTROL_MODE_AUTOMATIC : **enums.h**
- EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_FLOW : **enums.h**
- EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_PRESSURE : **enums.h**
- EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_SPEED : **enums.h**
- EMBER_ZCL_PUMP_CONTROL_MODE_CONSTANT_TEMPERATURE : **enums.h**
- EMBER_ZCL_PUMP_CONTROL_MODE_PROPORTIONAL_PRESSURE : **enums.h**
- EMBER_ZCL_PUMP_OPERATION_MODE_LOCAL : **enums.h**
- EMBER_ZCL_PUMP_OPERATION_MODE_MAXIMUM : **enums.h**
- EMBER_ZCL_PUMP_OPERATION_MODE_MINIMUM : **enums.h**
- EMBER_ZCL_PUMP_OPERATION_MODE_NORMAL : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER10_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER11_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER12_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER13_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER14_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER15_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER1_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER2_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER3_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER4_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER5_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER6_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER7_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER8_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_CURRENT_TIER9_SUMMATION_DELIVERED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_REGISTER_TIER_NO_TIER_RELATED : **enums.h**
- EMBER_ZCL_RELATIVE_HUMIDITY_DISPLAY_DISPLAYED : **enums.h**
- EMBER_ZCL_RELATIVE_HUMIDITY_DISPLAY_NOT_DISPLAYED : **enums.h**
- EMBER_ZCL_RELATIVE_HUMIDITY_MODE_MEASURE_LOCALLY : **enums.h**
- EMBER_ZCL_RELATIVE_HUMIDITY_MODE_UPDATED_OVER_THE_NETWORK : **enums.h**
- EMBER_ZCL_REPORTING_DIRECTION_RECEIVED : **enums.h**
- EMBER_ZCL_REPORTING_DIRECTION_REPORTED : **enums.h**
- EMBER_ZCL_SATURATION_MOVE_MODE_DOWN : **enums.h**

- EMBER_ZCL_SATURATION_MOVE_MODE_STOP : **enums.h**
- EMBER_ZCL_SATURATION_MOVE_MODE_UP : **enums.h**
- EMBER_ZCL_SATURATION_STEP_MODE_DOWN : **enums.h**
- EMBER_ZCL_SATURATION_STEP_MODE_UP : **enums.h**
- EMBER_ZCL_SENSING_LIGHT_SENSOR_TYPE_CMOS : **enums.h**
- EMBER_ZCL_SENSING_LIGHT_SENSOR_TYPE_PHOTODIODE : **enums.h**
- EMBER_ZCL_SETPOINT_ADJUST_MODE_COOL_SETPOINT : **enums.h**
- EMBER_ZCL_SETPOINT_ADJUST_MODE_HEAT_AND_COOL_SETPOINTS : **enums.h**
- EMBER_ZCL_SETPOINT_ADJUST_MODE_HEAT_SETPOINT : **enums.h**
- EMBER_ZCL_SIGNATURE_TYPE_ECDSA : **enums.h**
- EMBER_ZCL_SIGNATURE_TYPE_RESERVED : **enums.h**
- EMBER_ZCL_SQUAWK_LEVEL_LOW_LEVEL : **enums.h**
- EMBER_ZCL_SQUAWK_LEVEL_MEDIUM_LEVEL : **enums.h**
- EMBER_ZCL_SQUAWK_LEVEL_VERY_HIGH_LEVEL : **enums.h**
- EMBER_ZCL_SQUAWK_MODE_SYSTEM_IS_ARMED : **enums.h**
- EMBER_ZCL_SQUAWK_MODE_SYSTEM_IS_DISARMED : **enums.h**
- EMBER_ZCL_SQUAWK_STOBE_NO_STROBE : **enums.h**
- EMBER_ZCL_SQUAWK_STOBE_USE_STROBE : **enums.h**
- EMBER_ZCL_STATUS_ABORT : **enums.h**
- EMBER_ZCL_STATUS_CALIBRATION_ERROR : **enums.h**
- EMBER_ZCL_STATUS_DEFINED_OUT_OF_BAND : **enums.h**
- EMBER_ZCL_STATUS_DUPLICATE_EXISTS : **enums.h**
- EMBER_ZCL_STATUS_FAILURE : **enums.h**
- EMBER_ZCL_STATUS_HARDWARE_FAILURE : **enums.h**
- EMBER_ZCL_STATUS_INCONSISTENT_STARTUP_STATE : **enums.h**
- EMBER_ZCL_STATUS_INDICATION_REDIRECTION_TO_AP : **enums.h**
- EMBER_ZCL_STATUS_INSUFFICIENT_SPACE : **enums.h**
- EMBER_ZCL_STATUS_INVALID_DATA_TYPE : **enums.h**
- EMBER_ZCL_STATUS_INVALID_FIELD : **enums.h**
- EMBER_ZCL_STATUS_INVALID_IMAGE : **enums.h**
- EMBER_ZCL_STATUS_INVALID_SELECTOR : **enums.h**
- EMBER_ZCL_STATUS_INVALID_VALUE : **enums.h**
- EMBER_ZCL_STATUS_MALFORMED_COMMAND : **enums.h**
- EMBER_ZCL_STATUS_MULTIPLE_REQUEST_NOT_ALLOWED : **enums.h**
- EMBER_ZCL_STATUS_NO_IMAGE_AVAILABLE : **enums.h**
- EMBER_ZCL_STATUS_NOT_AUTHORIZED : **enums.h**
- EMBER_ZCL_STATUS_NOT_FOUND : **enums.h**
- EMBER_ZCL_STATUS_PREFERENCE_DENIED : **enums.h**
- EMBER_ZCL_STATUS_PREFERENCE_IGNORED : **enums.h**
- EMBER_ZCL_STATUS_READ_ONLY : **enums.h**
- EMBER_ZCL_STATUS_REQUEST_DENIED : **enums.h**
- EMBER_ZCL_STATUS_REQUIRE_MORE_IMAGE : **enums.h**
- EMBER_ZCL_STATUS_RESERVED_FIELD_NOT_ZERO : **enums.h**
- EMBER_ZCL_STATUS_SOFTWARE_FAILURE : **enums.h**
- EMBER_ZCL_STATUS_SUCCESS : **enums.h**
- EMBER_ZCL_STATUS_UNREPORTABLE_ATTRIBUTE : **enums.h**
- EMBER_ZCL_STATUS_UNSUP_CLUSTER_COMMAND : **enums.h**
- EMBER_ZCL_STATUS_UNSUP_GENERAL_COMMAND : **enums.h**
- EMBER_ZCL_STATUS_UNSUP_MANUF_CLUSTER_COMMAND : **enums.h**
- EMBER_ZCL_STATUS_UNSUP_MANUF_GENERAL_COMMAND : **enums.h**
- EMBER_ZCL_STATUS_UNSUPPORTED_ATTRIBUTE : **enums.h**
- EMBER_ZCL_STATUS_WAIT_FOR_DATA : **enums.h**
- EMBER_ZCL_STATUS_WRITE_ONLY : **enums.h**
- EMBER_ZCL_STEP_MODE_DOWN : **enums.h**
- EMBER_ZCL_STEP_MODE_UP : **enums.h**
- EMBER_ZCL_SWITCH_ACTIONS_OFF : **enums.h**
- EMBER_ZCL_SWITCH_ACTIONS_ON : **enums.h**

- EMBER_ZCL_SWITCH_ACTIONS_TOGGLE : **enums.h**
- EMBER_ZCL_SWITCH_TYPE_MOMENTARY : **enums.h**
- EMBER_ZCL_SWITCH_TYPE_TOGGLE : **enums.h**
- EMBER_ZCL_TEMPERATURE_DISPLAY_MODE_CELSIUS : **enums.h**
- EMBER_ZCL_TEMPERATURE_DISPLAY_MODE_FARENHEIT : **enums.h**
- EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_AND_HEATING : **enums.h**
- EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_AND_HEATING_WITH_REHEAT : **enums.h**
- EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_ONLY : **enums.h**
- EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_COOLING_WITH_REHEAT : **enums.h**
- EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_HEATING_ONLY : **enums.h**
- EMBER_ZCL_THERMOSTAT_CONTROL_SEQUENCE_HEATING_WITH_REHEAT : **enums.h**
- EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_AUTO : **enums.h**
- EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_COOL : **enums.h**
- EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_EMERGENCY_HEATING : **enums.h**
- EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_FAN_ONLY : **enums.h**
- EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_HEAT : **enums.h**
- EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_OFF : **enums.h**
- EMBER_ZCL_THERMOSTAT_SYSTEM_MODE_PRECOOLING : **enums.h**
- EMBER_ZCL_TUNNELING_PROTOCOL_ID_ANSI_C12 : **enums.h**
- EMBER_ZCL_TUNNELING_PROTOCOL_ID_CLIMATE_TALK : **enums.h**
- EMBER_ZCL_TUNNELING_PROTOCOL_ID_DLMS_COSEM : **enums.h**
- EMBER_ZCL_TUNNELING_PROTOCOL_ID_IEC_61107 : **enums.h**
- EMBER_ZCL_TUNNELING_PROTOCOL_ID_M_BUS : **enums.h**
- EMBER_ZCL_TUNNELING_PROTOCOL_ID_SML : **enums.h**
- EMBER_ZCL_TUNNELING_TRANSFER_DATA_STATUS_NO_SUCH_TUNNEL : **enums.h**
- EMBER_ZCL_TUNNELING_TRANSFER_DATA_STATUS_WRONG_DEVICE : **enums.h**
- EMBER_ZCL_TUNNELING_TUNNEL_STATUS_BUSY : **enums.h**
- EMBER_ZCL_TUNNELING_TUNNEL_STATUS_FLOW_CONTROL_NOT_SUPPORTED : **enums.h**
- EMBER_ZCL_TUNNELING_TUNNEL_STATUS_NO_MORE_TUNNEL_IDS : **enums.h**
- EMBER_ZCL_TUNNELING_TUNNEL_STATUS_PROTOCOL_NOT_SUPPORTED : **enums.h**
- EMBER_ZCL_TUNNELING_TUNNEL_STATUS_SUCCESS : **enums.h**
- EMBER_ZCL_WARNING_MODE_BURGLAR : **enums.h**
- EMBER_ZCL_WARNING_MODE_EMERGENCY : **enums.h**
- EMBER_ZCL_WARNING_MODE_FIRE : **enums.h**
- EMBER_ZCL_WARNING_MODE_STOP : **enums.h**
- EMBER_ZCL_WARNING_STOBE_NO_STROBE : **enums.h**
- EMBER_ZCL_WARNING_STOBE_USE_STROBE : **enums.h**
- ENTRY_IS_CANCELLED_EVENT : **load-control-event-table.h**
- ENTRY_IS_SUPERSEDED_EVENT : **load-control-event-table.h**
- ENTRY_SCHEDULED : **load-control-event-table.h**
- ENTRY_STARTED : **load-control-event-table.h**
- ENTRY_VOID : **load-control-event-table.h**
- EVENT_OPT_FLAG_OPT_IN : **load-control-event-table.h**
- EVENT_OPT_FLAG_PARTIAL : **load-control-event-table.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- b
- e
- g
- i
- k
- l
- m
- n
- o
- p
- r
- t
- z

- g -

- GENERATE_SHARED_SECRET_DONE : **af-types.h**
- GENERATING_EPHEMERAL_KEYS : **af-types.h**
- GENERATING_SHARED_SECRET : **af-types.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- b
- e
- g
- i
- k
- l
- m
- n
- o
- p
- r
- t
- z

- i -

- INVALID_APP_COMMAND : **af-types.h**
- INVALID_PARTNER_MESSAGE : **af-types.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- b
- e
- g
- i
- k
- l
- m
- n
- o
- p
- r
- t
- z

- k -

- KEY_CONFIRM_FAILURE : **af-types.h**
- KEY_GENERATION_DONE : **af-types.h**
- KEY_TABLE_FULL : **af-types.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- b
- e
- g
- i
- k
- l
- m
- n
- o
- p
- r
- t
- z

- l -

- LINK_KEY_ESTABLISHED : **af-types.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- b
- e
- g
- i
- k
- l
- m
- n
- o
- p
- r
- t
- z

- m -

- MESSAGE_SEND_FAILURE : **af-types.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- b
- e
- g
- i
- k
- l
- m
- n
- o
- p
- r
- t
- z

- n -

- NO_APP_MESSAGE : **af-types.h**
- NO_ESTABLISHMENT_ALLOWED : **af-types.h**
- NO_LOCAL_RESOURCES : **af-types.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- b
- e
- g
- i
- k
- l
- m
- n
- o
- p
- r
- t
- z

- o -

- OTA_UPGRADE_STATUS_COUNTDOWN : **ota.h**
- OTA_UPGRADE_STATUS_DOWNLOAD_COMPLETE : **ota.h**
- OTA_UPGRADE_STATUS_DOWNLOAD_IN_PROGRESS : **ota.h**
- OTA_UPGRADE_STATUS_NORMAL : **ota.h**
- OTA_UPGRADE_STATUS_WAIT : **ota.h**
- OTA_USE_OFFSET_TIME : **ota.h**
- OTA_USE_UTC_TIME : **ota.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- b
- e
- g
- i
- k
- l
- m
- n
- o
- p
- r
- t
- z

- p -

- PARTNER_NO_RESOURCES : **af-types.h**
- PARTNER_SENT_TERMINATE : **af-types.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- b
- e
- g
- i
- k
- l
- m
- n
- o
- p
- r
- t
- z

- r -

- RECEIVED_PARTNER_CERTIFICATE : **af-types.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- b
- e
- g
- i
- k
- l
- m
- n
- o
- p
- r
- t
- z

- t -

- TIMEOUT_OCCURRED : **af-types.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- b
- e
- g
- i
- k
- l
- m
- n
- o
- p
- r
- t
- z

- z -

- ZCL_ARRAY_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_ATTRIBUTE_ID_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BACNET_OID_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BAG_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BITMAP16_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BITMAP24_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BITMAP32_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BITMAP40_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BITMAP48_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BITMAP56_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BITMAP64_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BITMAP8_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_BOOLEAN_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_CHAR_STRING_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_CLUSTER_ID_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATA16_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATA24_ATTRIBUTE_TYPE : **attribute-type.h**

- ZCL_DATA32_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATA40_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATA48_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATA56_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATA64_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATA8_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_DATE_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_ENUM16_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_ENUM8_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_FLOAT_DOUBLE_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_FLOAT_SEMI_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_FLOAT_SINGLE_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_IEEE_ADDRESS_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT16S_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT16U_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT24S_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT24U_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT32S_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT32U_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT40S_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT40U_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT48S_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT48U_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT56S_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT56U_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT64S_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT64U_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT8S_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_INT8U_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_LONG_CHAR_STRING_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_LONG_OCTET_STRING_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_NO_DATA_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_OCTET_STRING_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_SECURITY_KEY_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_SET_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_STRUCT_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_TIME_OF_DAY_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_UNKNOWN_ATTRIBUTE_TYPE : **attribute-type.h**
- ZCL_UTC_TIME_ATTRIBUTE_TYPE : **attribute-type.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- a -

- APP_NOTIFY_ERROR_CODE_START : **af-types.h**
- APP_NOTIFY_MESSAGE_TEXT : **af-types.h**
- ATTRIBUTE_MASK_CLIENT : **attribute-storage.h**
- ATTRIBUTE_MASK_EXTERNAL_STORAGE : **attribute-storage.h**
- ATTRIBUTE_MASK_MANUFACTURER_SPECIFIC : **attribute-storage.h**
- ATTRIBUTE_MASK_MIN_MAX : **attribute-storage.h**
- ATTRIBUTE_MASK_SINGLETON : **attribute-storage.h**
- ATTRIBUTE_MASK_TOKENIZE : **attribute-storage.h**
- ATTRIBUTE_MASK_WRITABLE : **attribute-storage.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- b -

- BACNET_PROTOCOL_CLUSTER_MAX_NPDU_SIZE : **bacnet-protocol-tunnel.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- c -

- CANCEL_WITH_RANDOMIZATION : **load-control-event-table.h**
- CLOSE_INITIATED_BY_CLIENT : **tunneling-server.h**
- CLOSE_INITIATED_BY_SERVER : **tunneling-server.h**
- CLUSTER_IDS_TO_NAMES : **print-cluster.h**
- CLUSTER_MASK_ATTRIBUTE_CHANGED_FUNCTION : **attribute-storage.h**
- CLUSTER_MASK_CLIENT : **attribute-storage.h**
- CLUSTER_MASK_DEFAULT_RESPONSE_FUNCTION : **attribute-storage.h**
- CLUSTER_MASK_INIT_FUNCTION : **attribute-storage.h**
- CLUSTER_MASK_MANUFACTURER_SPECIFIC_ATTRIBUTE_CHANGED_FUNCTION : **attribute-storage.h**
- CLUSTER_MASK_MESSAGE_SENT_FUNCTION : **attribute-storage.h**

- CLUSTER_MASK_SERVER : **attribute-storage.h**
- CONFIGURATION_HEADER : **af.h**
- CUSTOM_COMMANDS : **custom-cli.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- d -

- DEBUG_PRINT_TEST : **debug.h**
- DEVICE_SPECIFIC_FILE_PRESENT_MASK : **ota-storage.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- e -

- EM_AF_DISCOVERY_RESPONSE_MASK : **af-types.h**
- EM_AF_KE_EPHEMERAL_SIZE : **key-establishment.h**
- EM_AF_KE_INITIATE_SIZE : **key-establishment.h**
- EM_AF_KE_SMAC_SIZE : **key-establishment.h**
- EM_AF_KE_TERMINATE_SIZE : **key-establishment.h**
- EM_AF_OTA_MAX_COMMAND_ID : **ota.h**
- EM_AF_PAGE_REQUEST_BLOCK_SIZE : **ota-client-page-request.h**
- emAfPrintPercentageSetStartAndEnd : **ota.h**
- emAfPrintPercentageUpdate : **ota.h**
- EMBER_AF_CLIENT_CLUSTER_DISCOVERY : **af.h**
- EMBER_AF_CLIENT_CLUSTER_TICK : **af.h**
- EMBER_AF_CONTINUE_IMAGE_VERIFY : **af.h**
- EMBER_AF_CORE_COMMANDS : **core-cli.h**
- EMBER_AF_INFO_COMMAND : **core-cli.h**
- EMBER_AF_INTER_PAN_BROADCAST : **af-types.h**

- EMBER_AF_INTER_PAN_MULTICAST : **af-types.h**
- EMBER_AF_INTER_PAN_UNICAST : **af-types.h**
- EMBER_AF_INTERPAN_DIRECTION_BOTH : **af-types.h**
- EMBER_AF_INTERPAN_DIRECTION_CLIENT_TO_SERVER : **af-types.h**
- EMBER_AF_INTERPAN_DIRECTION_SERVER_TO_CLIENT : **af-types.h**
- EMBER_AF_INTERPAN_GLOBAL_COMMAND : **af-types.h**
- EMBER_AF_INTERPAN_MANUFACTURER_SPECIFIC : **af-types.h**
- EMBER_AF_INTERPAN_OPTION_APS_ENCRYPT : **af-types.h**
- EMBER_AF_INTERPAN_OPTION_MAC_HAS_LONG_ADDRESS : **af-types.h**
- EMBER_AF_INTERPAN_OPTION_NONE : **af-types.h**
- EMBER_AF_INVALID_HARDWARE_VERSION : **ota-client-policy.h**
- EMBER_AF_NEW_IMAGE_VERIFICATION : **af.h**
- EMBER_AF_NULL_MANUFACTURER_CODE : **af-types.h**
- EMBER_AF_OTA_CLIENT_PAGE_REQUEST_SPACING_MS : **ota-client.h**
- EMBER_AF_OTA_MAX_HEADER_STRING_LENGTH : **af-types.h**
- EMBER_AF_OTA_QUERY_DELAY_MS : **ota-client.h**
- EMBER_AF_OTA_SERVER_DISCOVERY_DELAY_MS : **ota-client.h**
- EMBER_AF_PLUGIN_COMMANDS : **plugin-cli.h**
- EMBER_AF_PLUGIN_DRLC_SERVER_SCHEDULED_EVENT_TABLE_SIZE : **drlc-server.h**
- EMBER_AF_PLUGIN_GENERIC_TUNNEL_PROTOCOL_ADDRESS_SIZE : **generic-tunnel.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_DOWNLOAD_DELAY_MS : **ota-client.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_DOWNLOAD_ERROR_THRESHOLD : **ota-client.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_PAGE_REQUEST_SIZE : **ota-client.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_FIRMWARE_VERSION : **ota-client-policy.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_HARDWARE_VERSION : **ota-client-policy.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_POLICY_IMAGE_TYPE_ID : **ota-client-policy.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI0 : **ota-client.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI1 : **ota-client.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_SIGNER_EUI2 : **ota-client.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_UPGRADE_WAIT_THRESHOLD : **ota-client.h**
- EMBER_AF_PLUGIN_OTA_CLIENT_VERIFY_DELAY_MS : **ota-client.h**
- EMBER_AF_PLUGIN_PARTNER_LINK_KEY_EXCHANGE_TIMEOUT_MILLISECONDS : **partner-link-key-exchange.h**
- EMBER_AF_PLUGIN_PRICE_CLIENT_TABLE_SIZE : **price-client.h**
- EMBER_AF_PLUGIN_PRICE_SERVER_PRICE_TABLE_SIZE : **price-server.h**
- EMBER_AF_PLUGIN_PSL_COMMISSIONING_DEVICE_IS_SLEEPY : **psl-commissioning.h**
- EMBER_AF_PLUGIN_PSL_COMMISSIONING_TOUCH_LINK_MILLISECONDS_DELAY : **psl-commissioning.h**
- EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_BRIEF : **smart-energy-registration.h**
- EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_INITIAL : **smart-energy-registration.h**
- EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_NONE : **smart-energy-registration.h**
- EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_RESUME : **smart-energy-registration.h**
- EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_RETRY : **smart-energy-registration.h**
- EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_DELAY_TRANSITION : **smart-energy-**

registration.h

- EMBER_AF_PLUGIN_SMART_ENERGY_REGISTRATION_ERROR_LIMIT : **smart-energy-registration.h**
- EMBER_AF_PLUGIN_TRUST_CENTER_KEEPALIVE_DELAY_INTERVAL : **trust-center-keepalive.h**
- EMBER_AF_PLUGIN_TRUST_CENTER_KEEPALIVE_FAILURE_LIMIT : **trust-center-keepalive.h**
- EMBER_AF_PLUGIN_TUNNELING_CLIENT_TUNNEL_LIMIT : **tunneling-client.h**
- EMBER_AF_PLUGIN_TUNNELING_SERVER_TUNNEL_LIMIT : **tunneling-server.h**
- EMBER_AF_PRINT_ALARM_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_APP : **debug-printing-test.h**
- EMBER_AF_PRINT_ATTRIBUTES : **debug-printing-test.h**
- EMBER_AF_PRINT_BACNET_PROTOCOL_TUNNEL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_BALLAST_CONFIGURATION_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_BASIC_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_BINARY_INPUT_BASIC_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_BITS : **debug-printing-test.h**
- EMBER_AF_PRINT_COLOR_CONTROL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_COMMISSIONING_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_CORE : **debug-printing-test.h**
- EMBER_AF_PRINT_CUSTOM1 : **debug-printing-test.h**
- EMBER_AF_PRINT_CUSTOM2 : **debug-printing-test.h**
- EMBER_AF_PRINT_CUSTOM3 : **debug-printing-test.h**
- EMBER_AF_PRINT_DEBUG : **debug-printing-test.h**
- EMBER_AF_PRINT_DEHUMID_CONTROL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_DEVICE_TEMP_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_DOOR_LOCK_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_ENABLE : **debug-printing-test.h**
- EMBER_AF_PRINT_FAN_CONTROL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_FLOW_MEASUREMENT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_GENERIC_TUNNEL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_GROUPS_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_IAS_ACE_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_IAS_WD_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_IAS_ZONE_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_IDENTIFY_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_ILLUM_LEVEL_SENSING_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_ILLUM_MEASUREMENT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_KEY_ESTABLISHMENT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_LEVEL_CONTROL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_MESSAGING_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_NAME_NUMBER : **debug-printing-test.h**
- EMBER_AF_PRINT_NAMES : **debug-printing-test.h**
- EMBER_AF_PRINT_OCCUPANCY_SENSING_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_ON_OFF_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_ON_OFF_SWITCH_CONFIG_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_OTA_BOOTLOAD_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_OUTPUT : **debug-printing-test.h**
- EMBER_AF_PRINT_POWER_CONFIG_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_PREPAYMENT_CLUSTER : **debug-printing-test.h**

- EMBER_AF_PRINT_PRESSURE_MEASUREMENT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_PRICE_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_PUMP_CONFIG_CONTROL_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_REGISTRATION : **debug-printing-test.h**
- EMBER_AF_PRINT_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_REPORTING : **debug-printing-test.h**
- EMBER_AF_PRINT_RSSI_LOCATION_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_SAMPLE_MFG_SPECIFIC_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_SCENES_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_SECURITY : **debug-printing-test.h**
- EMBER_AF_PRINT_SERVICE_DISCOVERY : **debug-printing-test.h**
- EMBER_AF_PRINT_SHADE_CONFIG_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_SIMPLE_METERING_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_TEMP_MEASUREMENT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_THERMOSTAT_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_THERMOSTAT_UI_CONFIG_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_TIME_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_TUNNELING_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_WINDOW_COVERING_CLUSTER : **debug-printing-test.h**
- EMBER_AF_PRINT_ZDO : **debug-printing-test.h**
- EMBER_AF_RUN_UPGRADE_REQUEST_DELAY_MS : **ota-client.h**
- EMBER_AF_SCENE_TABLE_NULL_INDEX : **af-types.h**
- EMBER_AF_SCENE_TABLE_UNUSED_ENDPOINT_ID : **af-types.h**
- EMBER_AF_SERVER_CLUSTER_DISCOVERY : **af.h**
- EMBER_AF_SERVER_CLUSTER_TICK : **af.h**
- EMBER_AF_SET_HIBERNATE_DURATION : **af.h**
- EMBER_AF_SET_NAP_DURATION : **af.h**
- EMBER_AF_WAKE_TIMEOUT : **af.h**
- EMBER_AF_WAKE_TIMEOUT_MASK : **af.h**
- EMBER_AF_ZCL_MANUFACTURER_SPECIFIC_OVERHEAD : **af.h**
- EMBER_AF_ZCL_OVERHEAD : **af.h**
- EMBER_APPLICATION_HAS_POLL_COMPLETE_HANDLER : **stack-handlers.h**
- emberAfAlarmClusterDebugExec : **debug-printing.h**
- emberAfAlarmClusterFlush : **debug-printing.h**
- emberAfAlarmClusterPrint : **debug-printing.h**
- emberAfAlarmClusterPrintBuffer : **debug-printing.h**
- emberAfAlarmClusterPrintln : **debug-printing.h**
- emberAfAlarmClusterPrintString : **debug-printing.h**
- emberAfAppDebugExec : **debug-printing.h**
- emberAfAppFlush : **debug-printing.h**
- emberAfAppPrint : **debug-printing.h**
- emberAfAppPrintBuffer : **debug-printing.h**
- emberAfAppPrintln : **debug-printing.h**
- emberAfAppPrintString : **debug-printing.h**
- emberAfAttributeIsClient : **af.h**
- emberAfAttributeIsManufacturerSpecific : **af.h**
- emberAfAttributeIsReadOnly : **af.h**
- emberAfAttributeIsSingleton : **af.h**
- emberAfAttributeIsTokenized : **af.h**

- emberAfAttributesDebugExec : **debug-printing.h**
- emberAfAttributesFlush : **debug-printing.h**
- emberAfAttributeSize : **af.h**
- emberAfAttributesPrint : **debug-printing.h**
- emberAfAttributesPrintBuffer : **debug-printing.h**
- emberAfAttributesPrintln : **debug-printing.h**
- emberAfAttributesPrintString : **debug-printing.h**
- emberAfBacnetProtocolTunnelClusterDebugExec : **debug-printing.h**
- emberAfBacnetProtocolTunnelClusterFlush : **debug-printing.h**
- emberAfBacnetProtocolTunnelClusterPrint : **debug-printing.h**
- emberAfBacnetProtocolTunnelClusterPrintBuffer : **debug-printing.h**
- emberAfBacnetProtocolTunnelClusterPrintln : **debug-printing.h**
- emberAfBacnetProtocolTunnelClusterPrintString : **debug-printing.h**
- emberAfBallastConfigurationClusterDebugExec : **debug-printing.h**
- emberAfBallastConfigurationClusterFlush : **debug-printing.h**
- emberAfBallastConfigurationClusterPrint : **debug-printing.h**
- emberAfBallastConfigurationClusterPrintBuffer : **debug-printing.h**
- emberAfBallastConfigurationClusterPrintln : **debug-printing.h**
- emberAfBallastConfigurationClusterPrintString : **debug-printing.h**
- emberAfBasicClusterDebugExec : **debug-printing.h**
- emberAfBasicClusterFlush : **debug-printing.h**
- emberAfBasicClusterPrint : **debug-printing.h**
- emberAfBasicClusterPrintBuffer : **debug-printing.h**
- emberAfBasicClusterPrintln : **debug-printing.h**
- emberAfBasicClusterPrintString : **debug-printing.h**
- emberAfBinaryInputBasicClusterDebugExec : **debug-printing.h**
- emberAfBinaryInputBasicClusterFlush : **debug-printing.h**
- emberAfBinaryInputBasicClusterPrint : **debug-printing.h**
- emberAfBinaryInputBasicClusterPrintBuffer : **debug-printing.h**
- emberAfBinaryInputBasicClusterPrintln : **debug-printing.h**
- emberAfBinaryInputBasicClusterPrintString : **debug-printing.h**
- emberAfClusterIsManufacturerSpecific : **af.h**
- emberAfColorControlClusterDebugExec : **debug-printing.h**
- emberAfColorControlClusterFlush : **debug-printing.h**
- emberAfColorControlClusterPrint : **debug-printing.h**
- emberAfColorControlClusterPrintBuffer : **debug-printing.h**
- emberAfColorControlClusterPrintln : **debug-printing.h**
- emberAfColorControlClusterPrintString : **debug-printing.h**
- emberAfCommissioningClusterDebugExec : **debug-printing.h**
- emberAfCommissioningClusterFlush : **debug-printing.h**
- emberAfCommissioningClusterPrint : **debug-printing.h**
- emberAfCommissioningClusterPrintBuffer : **debug-printing.h**
- emberAfCommissioningClusterPrintln : **debug-printing.h**
- emberAfCommissioningClusterPrintString : **debug-printing.h**
- emberAfCopyInt8u : **af.h**
- emberAfCoreDebugExec : **debug-printing.h**
- emberAfCoreFlush : **debug-printing.h**
- emberAfCorePrint : **debug-printing.h**
- emberAfCorePrintBuffer : **debug-printing.h**

- emberAfCorePrintln : **debug-printing.h**
- emberAfCorePrintString : **debug-printing.h**
- emberAfCurrentAppTasks : **af.h**
- emberAfCurrentCommand : **af.h**
- emberAfCurrentEndpoint : **af.h**
- emberAfCustom1DebugExec : **debug-printing.h**
- emberAfCustom1Flush : **debug-printing.h**
- emberAfCustom1Print : **debug-printing.h**
- emberAfCustom1PrintBuffer : **debug-printing.h**
- emberAfCustom1Println : **debug-printing.h**
- emberAfCustom1PrintString : **debug-printing.h**
- emberAfCustom2DebugExec : **debug-printing.h**
- emberAfCustom2Flush : **debug-printing.h**
- emberAfCustom2Print : **debug-printing.h**
- emberAfCustom2PrintBuffer : **debug-printing.h**
- emberAfCustom2Println : **debug-printing.h**
- emberAfCustom2PrintString : **debug-printing.h**
- emberAfCustom3DebugExec : **debug-printing.h**
- emberAfCustom3Flush : **debug-printing.h**
- emberAfCustom3Print : **debug-printing.h**
- emberAfCustom3PrintBuffer : **debug-printing.h**
- emberAfCustom3Println : **debug-printing.h**
- emberAfCustom3PrintString : **debug-printing.h**
- emberAfDebugDebugExec : **debug-printing.h**
- emberAfDebugFlush : **debug-printing.h**
- emberAfDebugPrint : **debug-printing.h**
- emberAfDebugPrintBuffer : **debug-printing.h**
- emberAfDebugPrintln : **debug-printing.h**
- emberAfDebugPrintString : **debug-printing.h**
- emberAfDehumidControlClusterDebugExec : **debug-printing.h**
- emberAfDehumidControlClusterFlush : **debug-printing.h**
- emberAfDehumidControlClusterPrint : **debug-printing.h**
- emberAfDehumidControlClusterPrintBuffer : **debug-printing.h**
- emberAfDehumidControlClusterPrintln : **debug-printing.h**
- emberAfDehumidControlClusterPrintString : **debug-printing.h**
- emberAfDemandResponseLoadControlClusterDebugExec : **debug-printing.h**
- emberAfDemandResponseLoadControlClusterFlush : **debug-printing.h**
- emberAfDemandResponseLoadControlClusterPrint : **debug-printing.h**
- emberAfDemandResponseLoadControlClusterPrintBuffer : **debug-printing.h**
- emberAfDemandResponseLoadControlClusterPrintln : **debug-printing.h**
- emberAfDemandResponseLoadControlClusterPrintString : **debug-printing.h**
- emberAfDeviceIdFromIndex : **af.h**
- emberAfDeviceTempClusterDebugExec : **debug-printing.h**
- emberAfDeviceTempClusterFlush : **debug-printing.h**
- emberAfDeviceTempClusterPrint : **debug-printing.h**
- emberAfDeviceTempClusterPrintBuffer : **debug-printing.h**
- emberAfDeviceTempClusterPrintln : **debug-printing.h**
- emberAfDeviceTempClusterPrintString : **debug-printing.h**
- emberAfDeviceVersionFromIndex : **af.h**

- emberAfDoorLockClusterDebugExec : **debug-printing.h**
- emberAfDoorLockClusterFlush : **debug-printing.h**
- emberAfDoorLockClusterPrint : **debug-printing.h**
- emberAfDoorLockClusterPrintBuffer : **debug-printing.h**
- emberAfDoorLockClusterPrintln : **debug-printing.h**
- emberAfDoorLockClusterPrintString : **debug-printing.h**
- emberAfEndpointFromIndex : **af.h**
- emberAfFanControlClusterDebugExec : **debug-printing.h**
- emberAfFanControlClusterFlush : **debug-printing.h**
- emberAfFanControlClusterPrint : **debug-printing.h**
- emberAfFanControlClusterPrintBuffer : **debug-printing.h**
- emberAfFanControlClusterPrintln : **debug-printing.h**
- emberAfFanControlClusterPrintString : **debug-printing.h**
- emberAfFillCommandAlarmClusterAlarm : **client-command-macro.h**
- emberAfFillCommandAlarmClusterGetAlarm : **client-command-macro.h**
- emberAfFillCommandAlarmClusterGetAlarmResponse : **client-command-macro.h**
- emberAfFillCommandAlarmClusterResetAlarm : **client-command-macro.h**
- emberAfFillCommandAlarmClusterResetAlarmLog : **client-command-macro.h**
- emberAfFillCommandAlarmClusterResetAllAlarms : **client-command-macro.h**
- emberAfFillCommandBacnetProtocolTunnelClusterTransferNpdu : **client-command-macro.h**
- emberAfFillCommandBasicClusterResetToFactoryDefaults : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveColor : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveHue : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveSaturation : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveToColor : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveToColorTemperature : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveToHue : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveToHueAndSaturation : **client-command-macro.h**
- emberAfFillCommandColorControlClusterMoveToSaturation : **client-command-macro.h**
- emberAfFillCommandColorControlClusterStepColor : **client-command-macro.h**
- emberAfFillCommandColorControlClusterStepHue : **client-command-macro.h**
- emberAfFillCommandColorControlClusterStepSaturation : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterResetStartupParameters : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterResetStartupParametersResponse : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterRestartDevice : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterRestartDeviceResponse : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterRestoreStartupParameters : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterRestoreStartupParametersResponse : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterSaveStartupParameters : **client-command-macro.h**
- emberAfFillCommandCommissioningClusterSaveStartupParametersResponse : **client-command-macro.h**
- emberAfFillCommandDemandResponseLoadControlClusterCancelAllLoadControlEvents : **client-command-macro.h**
- emberAfFillCommandDemandResponseLoadControlClusterCancelLoadControlEvent : **client-command-macro.h**
- emberAfFillCommandDemandResponseLoadControlClusterGetScheduledEvents : **client-**

command-macro.h

- emberAfFillCommandDemandResponseLoadControlClusterLoadControlEvent : **client-command-macro.h**
- emberAfFillCommandDemandResponseLoadControlClusterReportEventStatus : **client-command-macro.h**
- emberAfFillCommandDoorLockClusterLockDoor : **client-command-macro.h**
- emberAfFillCommandDoorLockClusterLockDoorResponse : **client-command-macro.h**
- emberAfFillCommandDoorLockClusterUnlockDoor : **client-command-macro.h**
- emberAfFillCommandDoorLockClusterUnlockDoorResponse : **client-command-macro.h**
- emberAfFillCommandGenericTunnelClusterAdvertiseProtocolAddress : **client-command-macro.h**
- emberAfFillCommandGenericTunnelClusterMatchProtocolAddress : **client-command-macro.h**
- emberAfFillCommandGenericTunnelClusterMatchProtocolAddressResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerConfigureReporting : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerConfigureReportingResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerDefaultResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerDiscoverAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerDiscoverAttributesResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerReadAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerReadAttributesResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerReadReportingConfiguration : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerReadReportingConfigurationResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerReportAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerWriteAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerWriteAttributesNoResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerWriteAttributesResponse : **client-command-macro.h**
- emberAfFillCommandGlobalClientToServerWriteAttributesUndivided : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientConfigureReporting : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientConfigureReportingResponse : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientDefaultResponse : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientDiscoverAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientDiscoverAttributesResponse : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientReadAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientReadAttributesResponse : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientReadReportingConfiguration : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientReadReportingConfigurationResponse : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientReportAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientWriteAttributes : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientWriteAttributesNoResponse : **client-command-macro.h**

- emberAfFillCommandGlobalServerToClientWriteAttributesResponse : **client-command-macro.h**
- emberAfFillCommandGlobalServerToClientWriteAttributesUndivided : **client-command-macro.h**
- emberAfFillCommandGroupsClusterAddGroup : **client-command-macro.h**
- emberAfFillCommandGroupsClusterAddGroupIfIdentifying : **client-command-macro.h**
- emberAfFillCommandGroupsClusterAddGroupResponse : **client-command-macro.h**
- emberAfFillCommandGroupsClusterGetGroupMembership : **client-command-macro.h**
- emberAfFillCommandGroupsClusterGetGroupMembershipResponse : **client-command-macro.h**
- emberAfFillCommandGroupsClusterRemoveAllGroups : **client-command-macro.h**
- emberAfFillCommandGroupsClusterRemoveGroup : **client-command-macro.h**
- emberAfFillCommandGroupsClusterRemoveGroupResponse : **client-command-macro.h**
- emberAfFillCommandGroupsClusterViewGroup : **client-command-macro.h**
- emberAfFillCommandGroupsClusterViewGroupResponse : **client-command-macro.h**
- emberAfFillCommandIasAceClusterArm : **client-command-macro.h**
- emberAfFillCommandIasAceClusterArmResponse : **client-command-macro.h**
- emberAfFillCommandIasAceClusterBypass : **client-command-macro.h**
- emberAfFillCommandIasAceClusterEmergency : **client-command-macro.h**
- emberAfFillCommandIasAceClusterFire : **client-command-macro.h**
- emberAfFillCommandIasAceClusterGetZoneIdMap : **client-command-macro.h**
- emberAfFillCommandIasAceClusterGetZoneIdMapResponse : **client-command-macro.h**
- emberAfFillCommandIasAceClusterGetZoneInformation : **client-command-macro.h**
- emberAfFillCommandIasAceClusterGetZoneInformationResponse : **client-command-macro.h**
- emberAfFillCommandIasAceClusterPanic : **client-command-macro.h**
- emberAfFillCommandIasWdClusterSquawk : **client-command-macro.h**
- emberAfFillCommandIasWdClusterStartWarning : **client-command-macro.h**
- emberAfFillCommandIasZoneClusterZoneEnrollRequest : **client-command-macro.h**
- emberAfFillCommandIasZoneClusterZoneEnrollResponse : **client-command-macro.h**
- emberAfFillCommandIasZoneClusterZoneStatusChangeNotification : **client-command-macro.h**
- emberAfFillCommandIdentifyClusterIdentify : **client-command-macro.h**
- emberAfFillCommandIdentifyClusterIdentifyQuery : **client-command-macro.h**
- emberAfFillCommandIdentifyClusterIdentifyQueryResponse : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterClientToServerTerminateKeyEstablishment : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterConfirmKeyDataRequest : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterConfirmKeyDataResponse : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterEphemeralDataRequest : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterEphemeralDataResponse : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterInitiateKeyEstablishmentRequest : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterInitiateKeyEstablishmentResponse : **client-command-macro.h**
- emberAfFillCommandKeyEstablishmentClusterServerToClientTerminateKeyEstablishment : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterMove : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterMoveToLevel : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterMoveToLevelWithOnOff : **client-command-macro.h**

- emberAfFillCommandLevelControlClusterMoveWithOnOff : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterStep : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterStepWithOnOff : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterStop : **client-command-macro.h**
- emberAfFillCommandLevelControlClusterStopWithOnOff : **client-command-macro.h**
- emberAfFillCommandMessagingClusterCancelMessage : **client-command-macro.h**
- emberAfFillCommandMessagingClusterDisplayMessage : **client-command-macro.h**
- emberAfFillCommandMessagingClusterGetLastMessage : **client-command-macro.h**
- emberAfFillCommandMessagingClusterMessageConfirmation : **client-command-macro.h**
- emberAfFillCommandOnOffClusterOff : **client-command-macro.h**
- emberAfFillCommandOnOffClusterOn : **client-command-macro.h**
- emberAfFillCommandOnOffClusterSampleMfgSpecificOffWithTransition : **client-command-macro.h**
- emberAfFillCommandOnOffClusterSampleMfgSpecificOnWithTransition : **client-command-macro.h**
- emberAfFillCommandOnOffClusterSampleMfgSpecificToggleWithTransition : **client-command-macro.h**
- emberAfFillCommandOnOffClusterToggle : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterImageBlockRequest : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterImageBlockResponse : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterImageNotify : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterImagePageRequest : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterQueryNextImageRequest : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterQueryNextImageResponse : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterQuerySpecificFileRequest : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterQuerySpecificFileResponse : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterUpgradeEndRequest : **client-command-macro.h**
- emberAfFillCommandOtaBootloadClusterUpgradeEndResponse : **client-command-macro.h**
- emberAfFillCommandPrepaymentClusterChangeSupply : **client-command-macro.h**
- emberAfFillCommandPrepaymentClusterSelectAvailableEmergencyCredit : **client-command-macro.h**
- emberAfFillCommandPrepaymentClusterSupplyStatusResponse : **client-command-macro.h**
- emberAfFillCommandPriceClusterGetBlockPeriods : **client-command-macro.h**
- emberAfFillCommandPriceClusterGetCurrentPrice : **client-command-macro.h**
- emberAfFillCommandPriceClusterGetScheduledPrices : **client-command-macro.h**
- emberAfFillCommandPriceClusterPriceAcknowledgement : **client-command-macro.h**
- emberAfFillCommandPriceClusterPublishBlockPeriod : **client-command-macro.h**
- emberAfFillCommandPriceClusterPublishPrice : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterAnchorNodeAnnounce : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterCompactLocationDataNotification : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterDeviceConfigurationResponse : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterGetDeviceConfiguration : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterGetLocationData : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterLocationDataNotification : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterLocationDataResponse : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterReportRssiMeasurements : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterRequestOwnLocation : **client-command-macro.h**

- emberAfFillCommandRssiLocationClusterRssiPing : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterRssiRequest : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterRssiResponse : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterSendPings : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterSetAbsoluteLocation : **client-command-macro.h**
- emberAfFillCommandRssiLocationClusterSetDeviceConfiguration : **client-command-macro.h**
- emberAfFillCommandSampleMfgSpecificClusterCommandOne : **client-command-macro.h**
- emberAfFillCommandScenesClusterAddScene : **client-command-macro.h**
- emberAfFillCommandScenesClusterAddSceneResponse : **client-command-macro.h**
- emberAfFillCommandScenesClusterGetSceneMembership : **client-command-macro.h**
- emberAfFillCommandScenesClusterGetSceneMembershipResponse : **client-command-macro.h**
- emberAfFillCommandScenesClusterRecallScene : **client-command-macro.h**
- emberAfFillCommandScenesClusterRemoveAllScenes : **client-command-macro.h**
- emberAfFillCommandScenesClusterRemoveAllScenesResponse : **client-command-macro.h**
- emberAfFillCommandScenesClusterRemoveScene : **client-command-macro.h**
- emberAfFillCommandScenesClusterRemoveSceneResponse : **client-command-macro.h**
- emberAfFillCommandScenesClusterStoreScene : **client-command-macro.h**
- emberAfFillCommandScenesClusterStoreSceneResponse : **client-command-macro.h**
- emberAfFillCommandScenesClusterViewScene : **client-command-macro.h**
- emberAfFillCommandScenesClusterViewSceneResponse : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterGetProfile : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterGetProfileResponse : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterMirrorRemoved : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterRemoveMirror : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterRequestFastPollMode : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterRequestFastPollModeResponse : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterRequestMirror : **client-command-macro.h**
- emberAfFillCommandSimpleMeteringClusterRequestMirrorResponse : **client-command-macro.h**
- emberAfFillCommandThermostatClusterSetpointRaiseLower : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterAckTransferDataClientToServer : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterAckTransferDataServerToClient : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterCloseTunnel : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterReadyDataClientToServer : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterReadyDataServerToClient : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterRequestTunnel : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterRequestTunnelResponse : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterTransferDataClientToServer : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterTransferDataErrorClientToServer : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterTransferDataErrorServerToClient : **client-command-macro.h**
- emberAfFillCommandTunnelingClusterTransferDataServerToClient : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringDownClose : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftPercentage : **client-command-macro.h**

- emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftSetpoint : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringGoToLiftValue : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringGoToProgramSetpoint : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltPercentage : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltSetpoint : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringGoToTiltValue : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringStop : **client-command-macro.h**
- emberAfFillCommandWindowCoveringClusterWindowCoveringUpOpen : **client-command-macro.h**
- emberAfFlowMeasurementClusterDebugExec : **debug-printing.h**
- emberAfFlowMeasurementClusterFlush : **debug-printing.h**
- emberAfFlowMeasurementClusterPrint : **debug-printing.h**
- emberAfFlowMeasurementClusterPrintBuffer : **debug-printing.h**
- emberAfFlowMeasurementClusterPrintln : **debug-printing.h**
- emberAfFlowMeasurementClusterPrintString : **debug-printing.h**
- emberAfGenericTunnelClusterDebugExec : **debug-printing.h**
- emberAfGenericTunnelClusterFlush : **debug-printing.h**
- emberAfGenericTunnelClusterPrint : **debug-printing.h**
- emberAfGenericTunnelClusterPrintBuffer : **debug-printing.h**
- emberAfGenericTunnelClusterPrintln : **debug-printing.h**
- emberAfGenericTunnelClusterPrintString : **debug-printing.h**
- emberAfGetDefaultSleepControl : **af.h**
- emberAfGetInt8u : **af.h**
- emberAfGroupsClusterDebugExec : **debug-printing.h**
- emberAfGroupsClusterFlush : **debug-printing.h**
- emberAfGroupsClusterPrint : **debug-printing.h**
- emberAfGroupsClusterPrintBuffer : **debug-printing.h**
- emberAfGroupsClusterPrintln : **debug-printing.h**
- emberAfGroupsClusterPrintString : **debug-printing.h**
- emberAfGuaranteedFlush : **af.h**
- emberAfGuaranteedPrint : **af.h**
- emberAfGuaranteedPrintBuffer : **af.h**
- emberAfGuaranteedPrintln : **af.h**
- emberAfGuaranteedPrintString : **af.h**
- emberAfHaveDiscoveryResponseStatus : **af-types.h**
- emberAfIasAceClusterDebugExec : **debug-printing.h**
- emberAfIasAceClusterFlush : **debug-printing.h**
- emberAfIasAceClusterPrint : **debug-printing.h**
- emberAfIasAceClusterPrintBuffer : **debug-printing.h**
- emberAfIasAceClusterPrintln : **debug-printing.h**
- emberAfIasAceClusterPrintString : **debug-printing.h**
- emberAfIasWdClusterDebugExec : **debug-printing.h**
- emberAfIasWdClusterFlush : **debug-printing.h**

- emberAfIasWdClusterPrint : **debug-printing.h**
- emberAfIasWdClusterPrintBuffer : **debug-printing.h**
- emberAfIasWdClusterPrintln : **debug-printing.h**
- emberAfIasWdClusterPrintString : **debug-printing.h**
- emberAfIasZoneClusterDebugExec : **debug-printing.h**
- emberAfIasZoneClusterFlush : **debug-printing.h**
- emberAfIasZoneClusterPrint : **debug-printing.h**
- emberAfIasZoneClusterPrintBuffer : **debug-printing.h**
- emberAfIasZoneClusterPrintln : **debug-printing.h**
- emberAfIasZoneClusterPrintString : **debug-printing.h**
- emberAfIdentifyClusterDebugExec : **debug-printing.h**
- emberAfIdentifyClusterFlush : **debug-printing.h**
- emberAfIdentifyClusterPrint : **debug-printing.h**
- emberAfIdentifyClusterPrintBuffer : **debug-printing.h**
- emberAfIdentifyClusterPrintln : **debug-printing.h**
- emberAfIdentifyClusterPrintString : **debug-printing.h**
- emberAfIllumLevelSensingClusterDebugExec : **debug-printing.h**
- emberAfIllumLevelSensingClusterFlush : **debug-printing.h**
- emberAfIllumLevelSensingClusterPrint : **debug-printing.h**
- emberAfIllumLevelSensingClusterPrintBuffer : **debug-printing.h**
- emberAfIllumLevelSensingClusterPrintln : **debug-printing.h**
- emberAfIllumLevelSensingClusterPrintString : **debug-printing.h**
- emberAfIllumMeasurementClusterDebugExec : **debug-printing.h**
- emberAfIllumMeasurementClusterFlush : **debug-printing.h**
- emberAfIllumMeasurementClusterPrint : **debug-printing.h**
- emberAfIllumMeasurementClusterPrintBuffer : **debug-printing.h**
- emberAfIllumMeasurementClusterPrintln : **debug-printing.h**
- emberAfIllumMeasurementClusterPrintString : **debug-printing.h**
- emberAfKeyEstablishmentClusterDebugExec : **debug-printing.h**
- emberAfKeyEstablishmentClusterFlush : **debug-printing.h**
- emberAfKeyEstablishmentClusterPrint : **debug-printing.h**
- emberAfKeyEstablishmentClusterPrintBuffer : **debug-printing.h**
- emberAfKeyEstablishmentClusterPrintln : **debug-printing.h**
- emberAfKeyEstablishmentClusterPrintString : **debug-printing.h**
- emberAfLevelControlClusterDebugExec : **debug-printing.h**
- emberAfLevelControlClusterFlush : **debug-printing.h**
- emberAfLevelControlClusterPrint : **debug-printing.h**
- emberAfLevelControlClusterPrintBuffer : **debug-printing.h**
- emberAfLevelControlClusterPrintln : **debug-printing.h**
- emberAfLevelControlClusterPrintString : **debug-printing.h**
- emberAfMessagingClusterDebugExec : **debug-printing.h**
- emberAfMessagingClusterFlush : **debug-printing.h**
- emberAfMessagingClusterPrint : **debug-printing.h**
- emberAfMessagingClusterPrintBuffer : **debug-printing.h**
- emberAfMessagingClusterPrintln : **debug-printing.h**
- emberAfMessagingClusterPrintString : **debug-printing.h**
- emberAfOccupancySensingClusterDebugExec : **debug-printing.h**
- emberAfOccupancySensingClusterFlush : **debug-printing.h**
- emberAfOccupancySensingClusterPrint : **debug-printing.h**

- emberAfOccupancySensingClusterPrintBuffer : **debug-printing.h**
- emberAfOccupancySensingClusterPrintln : **debug-printing.h**
- emberAfOccupancySensingClusterPrintString : **debug-printing.h**
- emberAfOkToNap : **af.h**
- emberAfOnOffClusterDebugExec : **debug-printing.h**
- emberAfOnOffClusterFlush : **debug-printing.h**
- emberAfOnOffClusterPrint : **debug-printing.h**
- emberAfOnOffClusterPrintBuffer : **debug-printing.h**
- emberAfOnOffClusterPrintln : **debug-printing.h**
- emberAfOnOffClusterPrintString : **debug-printing.h**
- emberAfOnOffSwitchConfigClusterDebugExec : **debug-printing.h**
- emberAfOnOffSwitchConfigClusterFlush : **debug-printing.h**
- emberAfOnOffSwitchConfigClusterPrint : **debug-printing.h**
- emberAfOnOffSwitchConfigClusterPrintBuffer : **debug-printing.h**
- emberAfOnOffSwitchConfigClusterPrintln : **debug-printing.h**
- emberAfOnOffSwitchConfigClusterPrintString : **debug-printing.h**
- emberAfOtaBootloadClusterDebugExec : **debug-printing.h**
- emberAfOtaBootloadClusterFlush : **debug-printing.h**
- emberAfOtaBootloadClusterPrint : **debug-printing.h**
- emberAfOtaBootloadClusterPrintBuffer : **debug-printing.h**
- emberAfOtaBootloadClusterPrintln : **debug-printing.h**
- emberAfOtaBootloadClusterPrintString : **debug-printing.h**
- emberAfPowerConfigClusterDebugExec : **debug-printing.h**
- emberAfPowerConfigClusterFlush : **debug-printing.h**
- emberAfPowerConfigClusterPrint : **debug-printing.h**
- emberAfPowerConfigClusterPrintBuffer : **debug-printing.h**
- emberAfPowerConfigClusterPrintln : **debug-printing.h**
- emberAfPowerConfigClusterPrintString : **debug-printing.h**
- emberAfPrepaymentClusterDebugExec : **debug-printing.h**
- emberAfPrepaymentClusterFlush : **debug-printing.h**
- emberAfPrepaymentClusterPrint : **debug-printing.h**
- emberAfPrepaymentClusterPrintBuffer : **debug-printing.h**
- emberAfPrepaymentClusterPrintln : **debug-printing.h**
- emberAfPrepaymentClusterPrintString : **debug-printing.h**
- emberAfPressureMeasurementClusterDebugExec : **debug-printing.h**
- emberAfPressureMeasurementClusterFlush : **debug-printing.h**
- emberAfPressureMeasurementClusterPrint : **debug-printing.h**
- emberAfPressureMeasurementClusterPrintBuffer : **debug-printing.h**
- emberAfPressureMeasurementClusterPrintln : **debug-printing.h**
- emberAfPressureMeasurementClusterPrintString : **debug-printing.h**
- emberAfPriceClusterDebugExec : **debug-printing.h**
- emberAfPriceClusterFlush : **debug-printing.h**
- emberAfPriceClusterPrint : **debug-printing.h**
- emberAfPriceClusterPrintBuffer : **debug-printing.h**
- emberAfPriceClusterPrintln : **debug-printing.h**
- emberAfPriceClusterPrintString : **debug-printing.h**
- emberAfPrimaryEndpoint : **af.h**
- emberAfPrimaryProfileId : **af.h**
- emberAfProfileIdFromIndex : **af.h**

- emberAfPumpConfigControlClusterDebugExec : **debug-printing.h**
- emberAfPumpConfigControlClusterFlush : **debug-printing.h**
- emberAfPumpConfigControlClusterPrint : **debug-printing.h**
- emberAfPumpConfigControlClusterPrintBuffer : **debug-printing.h**
- emberAfPumpConfigControlClusterPrintln : **debug-printing.h**
- emberAfPumpConfigControlClusterPrintString : **debug-printing.h**
- emberAfQSToNextEvent : **af.h**
- emberAfRegistrationDebugExec : **debug-printing.h**
- emberAfRegistrationFlush : **debug-printing.h**
- emberAfRegistrationPrint : **debug-printing.h**
- emberAfRegistrationPrintBuffer : **debug-printing.h**
- emberAfRegistrationPrintln : **debug-printing.h**
- emberAfRegistrationPrintString : **debug-printing.h**
- emberAfRelativeHumidityMeasurementClusterDebugExec : **debug-printing.h**
- emberAfRelativeHumidityMeasurementClusterFlush : **debug-printing.h**
- emberAfRelativeHumidityMeasurementClusterPrint : **debug-printing.h**
- emberAfRelativeHumidityMeasurementClusterPrintBuffer : **debug-printing.h**
- emberAfRelativeHumidityMeasurementClusterPrintln : **debug-printing.h**
- emberAfRelativeHumidityMeasurementClusterPrintString : **debug-printing.h**
- emberAfReportingDebugExec : **debug-printing.h**
- emberAfReportingFlush : **debug-printing.h**
- emberAfReportingPrint : **debug-printing.h**
- emberAfReportingPrintBuffer : **debug-printing.h**
- emberAfReportingPrintln : **debug-printing.h**
- emberAfReportingPrintString : **debug-printing.h**
- emberAfRssiLocationClusterDebugExec : **debug-printing.h**
- emberAfRssiLocationClusterFlush : **debug-printing.h**
- emberAfRssiLocationClusterPrint : **debug-printing.h**
- emberAfRssiLocationClusterPrintBuffer : **debug-printing.h**
- emberAfRssiLocationClusterPrintln : **debug-printing.h**
- emberAfRssiLocationClusterPrintString : **debug-printing.h**
- emberAfSampleMfgSpecificClusterDebugExec : **debug-printing.h**
- emberAfSampleMfgSpecificClusterFlush : **debug-printing.h**
- emberAfSampleMfgSpecificClusterPrint : **debug-printing.h**
- emberAfSampleMfgSpecificClusterPrintBuffer : **debug-printing.h**
- emberAfSampleMfgSpecificClusterPrintln : **debug-printing.h**
- emberAfSampleMfgSpecificClusterPrintString : **debug-printing.h**
- emberAfScenesClusterDebugExec : **debug-printing.h**
- emberAfScenesClusterFlush : **debug-printing.h**
- emberAfScenesClusterPrint : **debug-printing.h**
- emberAfScenesClusterPrintBuffer : **debug-printing.h**
- emberAfScenesClusterPrintln : **debug-printing.h**
- emberAfScenesClusterPrintString : **debug-printing.h**
- emberAfSecurityDebugExec : **debug-printing.h**
- emberAfSecurityFlush : **debug-printing.h**
- emberAfSecurityPrint : **debug-printing.h**
- emberAfSecurityPrintBuffer : **debug-printing.h**
- emberAfSecurityPrintln : **debug-printing.h**
- emberAfSecurityPrintString : **debug-printing.h**

- emberAfSendCommandInterpanUnicast : **af.h**
- emberAfSendInterpanUnicast : **af.h**
- emberAfServiceDiscoveryDebugExec : **debug-printing.h**
- emberAfServiceDiscoveryFlush : **debug-printing.h**
- emberAfServiceDiscoveryPrint : **debug-printing.h**
- emberAfServiceDiscoveryPrintBuffer : **debug-printing.h**
- emberAfServiceDiscoveryPrintln : **debug-printing.h**
- emberAfServiceDiscoveryPrintString : **debug-printing.h**
- emberAfSetResponseWaiting : **af.h**
- emberAfShadeConfigClusterDebugExec : **debug-printing.h**
- emberAfShadeConfigClusterFlush : **debug-printing.h**
- emberAfShadeConfigClusterPrint : **debug-printing.h**
- emberAfShadeConfigClusterPrintBuffer : **debug-printing.h**
- emberAfShadeConfigClusterPrintln : **debug-printing.h**
- emberAfShadeConfigClusterPrintString : **debug-printing.h**
- emberAfSimpleMeteringClusterDebugExec : **debug-printing.h**
- emberAfSimpleMeteringClusterFlush : **debug-printing.h**
- emberAfSimpleMeteringClusterPrint : **debug-printing.h**
- emberAfSimpleMeteringClusterPrintBuffer : **debug-printing.h**
- emberAfSimpleMeteringClusterPrintln : **debug-printing.h**
- emberAfSimpleMeteringClusterPrintString : **debug-printing.h**
- emberAfTempMeasurementClusterDebugExec : **debug-printing.h**
- emberAfTempMeasurementClusterFlush : **debug-printing.h**
- emberAfTempMeasurementClusterPrint : **debug-printing.h**
- emberAfTempMeasurementClusterPrintBuffer : **debug-printing.h**
- emberAfTempMeasurementClusterPrintln : **debug-printing.h**
- emberAfTempMeasurementClusterPrintString : **debug-printing.h**
- emberAfThermostatClusterDebugExec : **debug-printing.h**
- emberAfThermostatClusterFlush : **debug-printing.h**
- emberAfThermostatClusterPrint : **debug-printing.h**
- emberAfThermostatClusterPrintBuffer : **debug-printing.h**
- emberAfThermostatClusterPrintln : **debug-printing.h**
- emberAfThermostatClusterPrintString : **debug-printing.h**
- emberAfThermostatUiConfigClusterDebugExec : **debug-printing.h**
- emberAfThermostatUiConfigClusterFlush : **debug-printing.h**
- emberAfThermostatUiConfigClusterPrint : **debug-printing.h**
- emberAfThermostatUiConfigClusterPrintBuffer : **debug-printing.h**
- emberAfThermostatUiConfigClusterPrintln : **debug-printing.h**
- emberAfThermostatUiConfigClusterPrintString : **debug-printing.h**
- emberAfTimeClusterDebugExec : **debug-printing.h**
- emberAfTimeClusterFlush : **debug-printing.h**
- emberAfTimeClusterPrint : **debug-printing.h**
- emberAfTimeClusterPrintBuffer : **debug-printing.h**
- emberAfTimeClusterPrintln : **debug-printing.h**
- emberAfTimeClusterPrintString : **debug-printing.h**
- emberAfTunnelingClusterDebugExec : **debug-printing.h**
- emberAfTunnelingClusterFlush : **debug-printing.h**
- emberAfTunnelingClusterPrint : **debug-printing.h**
- emberAfTunnelingClusterPrintBuffer : **debug-printing.h**

- emberAfTunnelingClusterPrintln : **debug-printing.h**
- emberAfTunnelingClusterPrintString : **debug-printing.h**
- emberAfWindowCoveringClusterDebugExec : **debug-printing.h**
- emberAfWindowCoveringClusterFlush : **debug-printing.h**
- emberAfWindowCoveringClusterPrint : **debug-printing.h**
- emberAfWindowCoveringClusterPrintBuffer : **debug-printing.h**
- emberAfWindowCoveringClusterPrintln : **debug-printing.h**
- emberAfWindowCoveringClusterPrintString : **debug-printing.h**
- emberAfZdoDebugExec : **debug-printing.h**
- emberAfZdoFlush : **debug-printing.h**
- emberAfZdoPrint : **debug-printing.h**
- emberAfZdoPrintBuffer : **debug-printing.h**
- emberAfZdoPrintln : **debug-printing.h**
- emberAfZdoPrintString : **debug-printing.h**
- EVENT_OPT_IN_DEFAULT : **demand-response-load-control.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- f -

- FIELD_CONTROL_OFFSET : **ota-storage.h**
- FILE_VERSION_WILD_CARD : **ota.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- h -

- HARDWARE_VERSIONS_PRESENT_MASK : **ota-storage.h**
- HEADER_LENGTH_FIELD_LENGTH : **ota-storage.h**
- HEADER_LENGTH_OFFSET : **ota-storage.h**
- HEADER_STRING_OFFSET : **ota-storage.h**
- HEADER_VERSION_OFFSET : **ota-storage.h**
- headerHasHardwareVersions : **ota-storage.h**
- headerHasSecurityCredentials : **ota-storage.h**
- headerHasUpgradeFileDest : **ota-storage.h**
- HOURS_IN_MS : **ota-client-policy.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- i -

- IMAGE_SIZE_OFFSET : **ota-storage.h**
- IMAGE_TYPE_CONFIG : **ota.h**
- IMAGE_TYPE_ID_OFFSET : **ota-storage.h**
- IMAGE_TYPE_LOG : **ota.h**
- IMAGE_TYPE_SECURITY : **ota.h**
- IMAGE_TYPE_WILD_CARD : **ota.h**
- INTER_PAN_BROADCAST : **af-types.h**
- INTER_PAN_MULTICAST : **af-types.h**
- INTER_PAN_UNICAST : **af-types.h**
- INTERPAN_APS_BROADCAST_SIZE : **interpan.h**

- INTERPAN_APS_ENCRYPTION_OVERHEAD : **interpan.h**
- INTERPAN_APS_FRAME_CONTROL_NO_DELIVERY_MODE : **interpan.h**
- INTERPAN_APS_FRAME_DELIVERY_MODE_MASK : **interpan.h**
- INTERPAN_APS_FRAME_SECURITY : **interpan.h**
- INTERPAN_APS_FRAME_TYPE : **interpan.h**
- INTERPAN_APS_FRAME_TYPE_MASK : **interpan.h**
- INTERPAN_APS_MULTICAST_SIZE : **interpan.h**
- INTERPAN_APS_UNICAST_SIZE : **interpan.h**
- InterPanHeader : **af-types.h**
- INVALID_DEVICE_ID : **ota-storage.h**
- INVALID_EUI64 : **ota-storage.h**
- INVALID_FIRMWARE_VERSION : **ota-storage.h**
- INVALID_MANUFACTURER_ID : **ota-storage.h**
- INVALID_OTA_IMAGE_ID : **ota-storage.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- k -

- KEY_UPDATE_COMMANDS : **security-cli.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- l -

- LC_TICK_TIME : **level-control.h**
- LC_TICKS_PER_SECOND : **level-control.h**
- LEVEL_CONTROL_MAXIMUM_LEVEL : **level-control.h**
- LEVEL_CONTROL_MINIMUM_LEVEL : **level-control.h**
- LOAD_CONTROL_EVENT_TABLE_ENTRY_BYTES : **load-control-event-table.h**
- LOAD_CONTROL_EVENT_TABLE_MEM_FOOTPRINT : **load-control-event-table.h**
- LOG_LEVELS : **debug.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- m -

- MAGIC_NUMBER_OFFSET : **ota-storage.h**
- MANUFACTURER_ID_OFFSET : **ota-storage.h**
- MAX_CLUSTER_NAME_LENGTH : **print-cluster.h**
- MAX_DIGEST_CALCULATIONS_PER_CALL : **ota-client-signature-verify.h**
- MAX_INTER_PAN_HEADER_SIZE : **interpan.h**
- MAX_INTER_PAN_MAC_SIZE : **interpan.h**
- MAX_PROFILE_INDEX : **simple-metering-test.h**
- MAX_STUB_APS_SIZE : **interpan.h**
- MFG_ID_WILD_CARD : **ota.h**
- MILLISECOND_TICKS_PER_HOUR : **af.h**

- MILLISECOND_TICKS_PER_MINUTE : **af.h**
- MILLISECOND_TICKS_PER_QUARTERSECOND : **af.h**
- MIN_STUB_APS_SIZE : **interpan.h**
- MINUTES_IN_MS : **ota-client-policy.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- n -

- NULL_EUI64 : **ota-client.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- o -

- OPTIONAL_FIELDS_OFFSET : **ota-storage.h**
- OTA_CLIENT_COMMANDS : **ota-cli.h**
- OTA_COMMANDS : **ota-cli.h**
- OTA_FILE_MAGIC_NUMBER : **ota-storage.h**
- OTA_HEADER_VERSION : **ota-storage.h**
- OTA_HW_VERSION_BIT_MASK : **ota.h**
- OTA_MAXIMUM_HEADER_LENGTH : **ota-storage.h**
- OTA_MINIMUM_HEADER_LENGTH : **ota-storage.h**
- OTA_NODE_EUI64_BIT_MASK : **ota.h**
- OTA_SERVER_COMMANDS : **ota-cli.h**

- OTA_TAG_ECDSA_SIGNATURE : **ota.h**
- OTA_TAG_ECDSA_SIGNING_CERTIFICATE : **ota.h**
- OTA_TAG_MANUFACTURER_SPECIFIC_START : **ota.h**
- OTA_TAG_RESERVED_END : **ota.h**
- OTA_TAG_RESERVED_START : **ota.h**
- OTA_TAG_UPGRADE_IMAGE : **ota.h**
- OTA_UPGRADE_END_RESPONSE_RUN_NOW : **ota.h**
- otaPrint : **ota.h**
- otaPrintln : **ota.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- p -

- PROFILE_INTERVAL_PERIOD_IN_MILLISECONDS : **simple-metering-test.h**
- PROFILE_INTERVAL_PERIOD_IN_MINUTES : **simple-metering-test.h**
- PROFILE_INTERVAL_PERIOD_IN_SECONDS : **simple-metering-test.h**
- PROFILE_INTERVAL_PERIOD_TIMEFRAME : **simple-metering-test.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- r -

- RANDOMIZE_END_TIME_FLAG : **load-control-event-table.h**
- RANDOMIZE_START_TIME_FLAG : **load-control-event-table.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- s -

- SECONDS_IN_MS : **ota-client-policy.h**
- SECURITY_CREDENTIAL_VERSION_FIELD_PRESENT_MASK : **ota-storage.h**
- SIGNATURE_TAG_DATA_SIZE : **ota-storage.h**
- STACK_VERSION_OFFSET : **ota-storage.h**
- STATIC_IMAGE_DATA : **ota-static-file-data.h**
- STATIC_IMAGE_DATA_SIZE : **ota-static-file-data.h**
- STUB_NWK_FRAME_CONTROL : **interpan.h**
- STUB_NWK_SIZE : **interpan.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- t -

- TAG_OVERHEAD : **ota-storage.h**
- TOTAL_PROFILE_TIME_SPAN_IN_SECONDS : **simple-metering-test.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- v -

- VERSION_OFFSET : **ota-storage.h**

- All
- Functions
- Variables
- Typedefs
- Enumerations
- Enumerator
- Defines

- _
- a
- b
- c
- d
- e
- f
- h
- i
- k
- l
- m
- n
- o
- p
- r
- s
- t
- v
- z

- z -

- ZCL_ABS_MAX_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ABS_MAX_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ABS_MIN_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ABS_MIN_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ACCELERATION_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ACK_TRANSFER_DATA_CLIENT_TO_SERVER_COMMAND_ID : **command-id.h**
- ZCL_ACK_TRANSFER_DATA_SERVER_TO_CLIENT_COMMAND_ID : **command-id.h**
- ZCL_ACTIVE_TEXT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ACTUATOR_ENABLED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ADD_GROUP_COMMAND_ID : **command-id.h**
- ZCL_ADD_GROUP_IF_IDENTIFYING_COMMAND_ID : **command-id.h**
- ZCL_ADD_GROUP_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_ADD_SCENE_COMMAND_ID : **command-id.h**
- ZCL_ADD_SCENE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_ADVERTISE_PROTOCOL_ADDRESS_COMMAND_ID : **command-id.h**

- ZCL_ALARM_CLUSTER_ID : **cluster-id.h**
- ZCL_ALARM_COMMAND_ID : **command-id.h**
- ZCL_ALARM_COUNT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ANCHOR_NODE_ANNOUNCE_COMMAND_ID : **command-id.h**
- ZCL_APPLICATION_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_APPLICATION_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ARM_COMMAND_ID : **command-id.h**
- ZCL_ARM_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_ATTRIBUTE_ONE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BACNET_PROTOCOL_TUNNEL_CLUSTER_ID : **cluster-id.h**
- ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_EXECUTE : **debug.h**
- ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_ON : **debug.h**
- ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_PRINT : **debug.h**
- ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_BACNET_PROTOCOL_TUNNEL_DEBUG_WAITSEND : **debug.h**
- ZCL_BALLAST_CONFIGURATION_CLUSTER_ID : **cluster-id.h**
- ZCL_BALLAST_FACTOR_ADJUSTMENT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BALLAST_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BASIC_CLUSTER_ID : **cluster-id.h**
- ZCL_BASIC_DEBUG_EXECUTE : **debug.h**
- ZCL_BASIC_DEBUG_ON : **debug.h**
- ZCL_BASIC_DEBUG_PRINT : **debug.h**
- ZCL_BASIC_DEBUG_PRINT_LINE : **debug.h**
- ZCL_BASIC_DEBUG_WAITSEND : **debug.h**
- ZCL_BATTERY_AHR_RATING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BATTERY_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BATTERY_MANUFACTURER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BATTERY_QUANTITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BATTERY_RATED_VOLTAGE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BATTERY_SIZE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BATTERY_VOLTAGE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BATTERY_VOLTAGE_MIN_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BINARY_INPUT_BASIC_CLUSTER_ID : **cluster-id.h**
- ZCL_BINARY_INPUT_BASIC_DEBUG_EXECUTE : **debug.h**
- ZCL_BINARY_INPUT_BASIC_DEBUG_ON : **debug.h**
- ZCL_BINARY_INPUT_BASIC_DEBUG_PRINT : **debug.h**
- ZCL_BINARY_INPUT_BASIC_DEBUG_PRINT_LINE : **debug.h**
- ZCL_BINARY_INPUT_BASIC_DEBUG_WAITSEND : **debug.h**
- ZCL_BLOCK10_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK11_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK12_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK13_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK14_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK15_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK1_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK2_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK3_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK4_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_BLOCK5_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK6_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK7_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK8_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK9_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BLOCK_PERIOD_DURATION_MINUTES_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_BYPASS_COMMAND_ID : **command-id.h**
- ZCL_CALCULATION_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CANCEL_ALL_LOAD_CONTROL_EVENTS_COMMAND_ID : **command-id.h**
- ZCL_CANCEL_LOAD_CONTROL_EVENT_COMMAND_ID : **command-id.h**
- ZCL_CANCEL_MESSAGE_COMMAND_ID : **command-id.h**
- ZCL_CAPACITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CHANGE_SUPPLY_COMMAND_ID : **command-id.h**
- ZCL_CHANNEL_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CLOSE_TUNNEL_COMMAND_ID : **command-id.h**
- ZCL_CLOSE_TUNNEL_TIMEOUT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CLOSED_LIMIT_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CLOSED_LIMIT_TILT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CLUSTER_SPECIFIC_COMMAND : **af.h**
- ZCL_COLOR_CONTROL_CLUSTER_ID : **cluster-id.h**
- ZCL_COLOR_CONTROL_COLOR_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_B_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_B_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_B_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_G_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_G_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_G_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_R_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_R_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_POINT_R_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COLOR_TEMPERATURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_COMPENSATION_TEXT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_CURRENT_HUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_CURRENT_SATURATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_CURRENT_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_CURRENT_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_DEBUG_EXECUTE : **debug.h**
- ZCL_COLOR_CONTROL_DEBUG_ON : **debug.h**
- ZCL_COLOR_CONTROL_DEBUG_PRINT : **debug.h**
- ZCL_COLOR_CONTROL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_COLOR_CONTROL_DEBUG_WAITSEND : **debug.h**
- ZCL_COLOR_CONTROL_DRIFT_COMPENSATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_HUE_STEP_UNIT : **color-control.h**
- ZCL_COLOR_CONTROL_NUMBER_OF_PRIMARIES_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_1_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_1_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_1_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_2_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_2_X_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_COLOR_CONTROL_PRIMARY_2_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_3_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_3_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_3_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_4_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_4_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_4_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_5_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_5_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_5_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_6_INTENSITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_6_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_PRIMARY_6_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_REMAINING_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_SATURATION_STEP_UNIT : **color-control.h**
- ZCL_COLOR_CONTROL_WHITE_POINT_X_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COLOR_CONTROL_WHITE_POINT_Y_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COMMAND_ONE_COMMAND_ID : **command-id.h**
- ZCL_COMMISSIONING_CLUSTER_ID : **cluster-id.h**
- ZCL_COMMISSIONING_DEBUG_EXECUTE : **debug.h**
- ZCL_COMMISSIONING_DEBUG_ON : **debug.h**
- ZCL_COMMISSIONING_DEBUG_PRINT : **debug.h**
- ZCL_COMMISSIONING_DEBUG_PRINT_LINE : **debug.h**
- ZCL_COMMISSIONING_DEBUG_WAITSEND : **debug.h**
- ZCL_COMMODITY_TYPE_CLIENT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COMMODITY_TYPE_SERVER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COMPACT_LOCATION_DATA_NOTIFICATION_COMMAND_ID : **command-id.h**
- ZCL_CONCENTRATOR_DISCOVERY_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CONCENTRATOR_FLAG_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CONCENTRATOR_RADIUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CONFIG_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CONFIGURE_REPORTING_COMMAND_ID : **command-id.h**
- ZCL_CONFIGURE_REPORTING_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_CONFIRM_KEY_DATA_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_CONFIRM_KEY_DATA_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_CONTROL_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CONTROL_SEQUENCE_OF_OPERATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CONTROL_TEMPERATURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COORDINATE1_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COORDINATE2_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COORDINATE3_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_COVERING_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CREDIT_REMAINING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CREDIT_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_BLOCK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_BLOCK_PERIOD_CONSUMPTION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_DAY_CONSUMPTION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_DAY_CONSUMPTION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_DAY_MAX_DEMAND_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_CURRENT_DAY_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_DAY_MAX_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_DAY_MIN_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_DEMAND_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_FILE_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_GROUP_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_INLET_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_INLET_ENERGY_CARRIER_SUMMATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_LIFT_PERCENTAGE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_MAX_DEMAND_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_MAX_DEMAND_DELIVERED_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_MAX_DEMAND_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_MAX_DEMAND_RECEIVED_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_MONTH_MAX_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_MONTH_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_MONTH_MIN_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_NO_TIER_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_OUTLET_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_OUTLET_ENERGY_CARRIER_SUMMATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_START_TIME_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_START_TIME_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_VALUE_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_PARTIAL_PROFILE_INTERVAL_VALUE_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_SCENE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TEMPERATURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_CURRENT_TIER10_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER10_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER11_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_CURRENT_TIER12_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER12_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER13_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER14_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_CURRENT_TIER15_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER15_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER1_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER2_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_CURRENT_TIER3_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER3_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER4_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER5_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_CURRENT_TIER6_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER6_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER7_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_CURRENT_TIER8_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER8_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK10_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK11_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK12_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK13_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK14_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK15_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK16_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK1_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK2_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK3_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK4_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK5_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK6_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK7_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK8_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_BLOCK9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_SUMMATION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TIER9_SUMMATION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TILT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_TILT_PERCENTAGE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_YEAR_MAX_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_YEAR_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_YEAR_MIN_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_CURRENT_ZIGBEE_STACK_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DAILY_CONSUMPTION_TARGET_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DAILY_FREEZE_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DATE_CODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DECELERATION_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEFAULT_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_DEFAULT_UPDATE_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEHUMID_CONTROL_CLUSTER_ID : **cluster-id.h**
- ZCL_DEHUMID_CONTROL_DEBUG_EXECUTE : **debug.h**
- ZCL_DEHUMID_CONTROL_DEBUG_ON : **debug.h**
- ZCL_DEHUMID_CONTROL_DEBUG_PRINT : **debug.h**
- ZCL_DEHUMID_CONTROL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_DEHUMID_CONTROL_DEBUG_WAITSEND : **debug.h**
- ZCL_DEHUMIDIFICATION_COOLING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEHUMIDIFICATION_HYSTERESIS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEHUMIDIFICATION_LOCKOUT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEHUMIDIFICATION_MAX_COOL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DELAYED_SUPPLY_INTERRUPT_VALUE_REMAINING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DELAYED_SUPPLY_INTERRUPT_VALUE_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEMAND_FORMATTING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEMAND_INTEGRATION_PERIOD_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_DEMAND_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEMAND_RESPONSE_LOAD_CONTROL_CLUSTER_ID : **cluster-id.h**
- ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_EXECUTE : **debug.h**
- ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_ON : **debug.h**
- ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_PRINT : **debug.h**
- ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_DEMAND_RESPONSE_LOAD_CONTROL_DEBUG_WAITSEND : **debug.h**
- ZCL_DESCRIPTION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEVICE_CLASS_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEVICE_CONFIGURATION_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_DEVICE_ENABLED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEVICE_TEMP_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DEVICE_TEMP_CLUSTER_ID : **cluster-id.h**
- ZCL_DEVICE_TEMP_DEBUG_EXECUTE : **debug.h**
- ZCL_DEVICE_TEMP_DEBUG_ON : **debug.h**
- ZCL_DEVICE_TEMP_DEBUG_PRINT : **debug.h**
- ZCL_DEVICE_TEMP_DEBUG_PRINT_LINE : **debug.h**
- ZCL_DEVICE_TEMP_DEBUG_WAITSEND : **debug.h**
- ZCL_DFT_SUMMATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DISABLE_DEFAULT_RESPONSE_MASK : **af.h**
- ZCL_DISABLE_LOCAL_CONFIG_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DISCOVER_ATTRIBUTES_COMMAND_ID : **command-id.h**
- ZCL_DISCOVER_ATTRIBUTES_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_DISPLAY_MESSAGE_COMMAND_ID : **command-id.h**
- ZCL_DIVISOR_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DOOR_CLOSED_EVENTS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DOOR_LOCK_CLUSTER_ID : **cluster-id.h**
- ZCL_DOOR_LOCK_DEBUG_EXECUTE : **debug.h**
- ZCL_DOOR_LOCK_DEBUG_ON : **debug.h**
- ZCL_DOOR_LOCK_DEBUG_PRINT : **debug.h**
- ZCL_DOOR_LOCK_DEBUG_PRINT_LINE : **debug.h**
- ZCL_DOOR_LOCK_DEBUG_WAITSEND : **debug.h**
- ZCL_DOOR_OPEN_EVENTS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DOOR_STATE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DOWNLOADED_FILE_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DOWNLOADED_ZIGBEE_STACK_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DST_END_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DST_SHIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_DST_START_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_EFFECTIVE_CONTROL_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_EFFECTIVE_OPERATION_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ELECTRICITY_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_EMERGENCY_COMMAND_ID : **command-id.h**
- ZCL_EMERGENCY_CREDIT_REMAINING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ENERGY_CARRIER_DEMAND_FORMATTING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ENERGY_CARRIER_SUMMATION_FORMATTING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ENERGY_CARRIER_UNIT_OF_MEASURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_EPHEMERAL_DATA_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_EPHEMERAL_DATA_RESPONSE_COMMAND_ID : **command-id.h**

- ZCL_EXTENDED_PAN_ID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FAN_CONTROL_CLUSTER_ID : **cluster-id.h**
- ZCL_FAN_CONTROL_DEBUG_EXECUTE : **debug.h**
- ZCL_FAN_CONTROL_DEBUG_ON : **debug.h**
- ZCL_FAN_CONTROL_DEBUG_PRINT : **debug.h**
- ZCL_FAN_CONTROL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_FAN_CONTROL_DEBUG_WAITSEND : **debug.h**
- ZCL_FAN_CONTROL_FAN_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FAN_CONTROL_FAN_MODE_SEQUENCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FAST_POLL_UPDATE_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FILE_OFFSET_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FIRE_COMMAND_ID : **command-id.h**
- ZCL_FLOW_MAX_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FLOW_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FLOW_MEASUREMENT_CLUSTER_ID : **cluster-id.h**
- ZCL_FLOW_MEASUREMENT_DEBUG_EXECUTE : **debug.h**
- ZCL_FLOW_MEASUREMENT_DEBUG_ON : **debug.h**
- ZCL_FLOW_MEASUREMENT_DEBUG_PRINT : **debug.h**
- ZCL_FLOW_MEASUREMENT_DEBUG_PRINT_LINE : **debug.h**
- ZCL_FLOW_MEASUREMENT_DEBUG_WAITSEND : **debug.h**
- ZCL_FLOW_MIN_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FLOW_RESTRICTION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FLOW_TOLERANCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FRAME_CONTROL_CLIENT_TO_SERVER : **af.h**
- ZCL_FRAME_CONTROL_DIRECTION_MASK : **af.h**
- ZCL_FRAME_CONTROL_FRAME_TYPE_MASK : **af.h**
- ZCL_FRAME_CONTROL_SERVER_TO_CLIENT : **af.h**
- ZCL_FUEL_DEBT_RECOVERY_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FUEL_DEBT_RECOVERY_RATE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_FUEL_DEBT_REMAINING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_GAS_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_GENERIC_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_GENERIC_FLOW_PRESSURE_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_GENERIC_TUNNEL_CLUSTER_ID : **cluster-id.h**
- ZCL_GENERIC_TUNNEL_DEBUG_EXECUTE : **debug.h**
- ZCL_GENERIC_TUNNEL_DEBUG_ON : **debug.h**
- ZCL_GENERIC_TUNNEL_DEBUG_PRINT : **debug.h**
- ZCL_GENERIC_TUNNEL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_GENERIC_TUNNEL_DEBUG_WAITSEND : **debug.h**
- ZCL_GET_ALARM_COMMAND_ID : **command-id.h**
- ZCL_GET_ALARM_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_GET_BLOCK_PERIODS_COMMAND_ID : **command-id.h**
- ZCL_GET_CURRENT_PRICE_COMMAND_ID : **command-id.h**
- ZCL_GET_DEVICE_CONFIGURATION_COMMAND_ID : **command-id.h**
- ZCL_GET_GROUP_MEMBERSHIP_COMMAND_ID : **command-id.h**
- ZCL_GET_GROUP_MEMBERSHIP_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_GET_LAST_MESSAGE_COMMAND_ID : **command-id.h**
- ZCL_GET_LOCATION_DATA_COMMAND_ID : **command-id.h**
- ZCL_GET_PROFILE_COMMAND_ID : **command-id.h**

- ZCL_GET_PROFILE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_GET_SCENE_MEMBERSHIP_COMMAND_ID : **command-id.h**
- ZCL_GET_SCENE_MEMBERSHIP_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_GET_SCHEDULED_EVENTS_COMMAND_ID : **command-id.h**
- ZCL_GET_SCHEDULED_PRICES_COMMAND_ID : **command-id.h**
- ZCL_GET_ZONE_ID_MAP_COMMAND_ID : **command-id.h**
- ZCL_GET_ZONE_ID_MAP_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_GET_ZONE_INFORMATION_COMMAND_ID : **command-id.h**
- ZCL_GET_ZONE_INFORMATION_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_GROUP_NAME_SUPPORT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_GROUPS_CLUSTER_ID : **cluster-id.h**
- ZCL_GROUPS_DEBUG_EXECUTE : **debug.h**
- ZCL_GROUPS_DEBUG_ON : **debug.h**
- ZCL_GROUPS_DEBUG_PRINT : **debug.h**
- ZCL_GROUPS_DEBUG_PRINT_LINE : **debug.h**
- ZCL_GROUPS_DEBUG_WAITSEND : **debug.h**
- ZCL_HEAT_AND_COOLING_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_HIGH_TEMP_DWELL_TRIP_POINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_HIGH_TEMP_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_HISTORICAL_CONSUMPTION_FORMATTING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_HOURS_IN_FAULT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_HOURS_IN_OPERATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_HW_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_IAS_ACE_CLUSTER_ID : **cluster-id.h**
- ZCL_IAS_ACE_DEBUG_EXECUTE : **debug.h**
- ZCL_IAS_ACE_DEBUG_ON : **debug.h**
- ZCL_IAS_ACE_DEBUG_PRINT : **debug.h**
- ZCL_IAS_ACE_DEBUG_PRINT_LINE : **debug.h**
- ZCL_IAS_ACE_DEBUG_WAITSEND : **debug.h**
- ZCL_IAS_CIE_ADDRESS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_IAS_WD_CLUSTER_ID : **cluster-id.h**
- ZCL_IAS_WD_DEBUG_EXECUTE : **debug.h**
- ZCL_IAS_WD_DEBUG_ON : **debug.h**
- ZCL_IAS_WD_DEBUG_PRINT : **debug.h**
- ZCL_IAS_WD_DEBUG_PRINT_LINE : **debug.h**
- ZCL_IAS_WD_DEBUG_WAITSEND : **debug.h**
- ZCL_IAS_ZONE_CLUSTER_ID : **cluster-id.h**
- ZCL_IAS_ZONE_DEBUG_EXECUTE : **debug.h**
- ZCL_IAS_ZONE_DEBUG_ON : **debug.h**
- ZCL_IAS_ZONE_DEBUG_PRINT : **debug.h**
- ZCL_IAS_ZONE_DEBUG_PRINT_LINE : **debug.h**
- ZCL_IAS_ZONE_DEBUG_WAITSEND : **debug.h**
- ZCL_IDENTIFY_CLUSTER_ID : **cluster-id.h**
- ZCL_IDENTIFY_COMMAND_ID : **command-id.h**
- ZCL_IDENTIFY_DEBUG_EXECUTE : **debug.h**
- ZCL_IDENTIFY_DEBUG_ON : **debug.h**
- ZCL_IDENTIFY_DEBUG_PRINT : **debug.h**
- ZCL_IDENTIFY_DEBUG_PRINT_LINE : **debug.h**
- ZCL_IDENTIFY_DEBUG_WAITSEND : **debug.h**

- ZCL_IDENTIFY_QUERY_COMMAND_ID : **command-id.h**
- ZCL_IDENTIFY_QUERY_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_IDENTIFY_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ILLUM_LEVEL_SENSING_CLUSTER_ID : **cluster-id.h**
- ZCL_ILLUM_LEVEL_SENSING_DEBUG_EXECUTE : **debug.h**
- ZCL_ILLUM_LEVEL_SENSING_DEBUG_ON : **debug.h**
- ZCL_ILLUM_LEVEL_SENSING_DEBUG_PRINT : **debug.h**
- ZCL_ILLUM_LEVEL_SENSING_DEBUG_PRINT_LINE : **debug.h**
- ZCL_ILLUM_LEVEL_SENSING_DEBUG_WAITSEND : **debug.h**
- ZCL_ILLUM_MAX_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ILLUM_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ILLUM_MEASUREMENT_CLUSTER_ID : **cluster-id.h**
- ZCL_ILLUM_MEASUREMENT_DEBUG_EXECUTE : **debug.h**
- ZCL_ILLUM_MEASUREMENT_DEBUG_ON : **debug.h**
- ZCL_ILLUM_MEASUREMENT_DEBUG_PRINT : **debug.h**
- ZCL_ILLUM_MEASUREMENT_DEBUG_PRINT_LINE : **debug.h**
- ZCL_ILLUM_MEASUREMENT_DEBUG_WAITSEND : **debug.h**
- ZCL_ILLUM_MIN_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ILLUM_TOLERANCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ILLUMINANCE_TARGET_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_IMAGE_BLOCK_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_IMAGE_BLOCK_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_IMAGE_NOTIFY_COMMAND_ID : **command-id.h**
- ZCL_IMAGE_PAGE_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_IMAGE_TYPE_ID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_IMAGE_UPGRADE_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_INACTIVE_TEXT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_INDERECT_POLL_RATE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_INITIATE_KEY_ESTABLISHMENT_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_INITIATE_KEY_ESTABLISHMENT_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_INLET_TEMPERATURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_INSTANTANEOUS_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_INTERVAL_READ_REPORTING_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_INTRINSIC_BALLAST_FACTOR_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_INVALID_COLOR_CONTROL_COMMAND_ID : **color-control.h**
- ZCL_KEY_ESTABLISHMENT_CLUSTER_ID : **cluster-id.h**
- ZCL_KEY_ESTABLISHMENT_DEBUG_EXECUTE : **debug.h**
- ZCL_KEY_ESTABLISHMENT_DEBUG_ON : **debug.h**
- ZCL_KEY_ESTABLISHMENT_DEBUG_PRINT : **debug.h**
- ZCL_KEY_ESTABLISHMENT_DEBUG_PRINT_LINE : **debug.h**
- ZCL_KEY_ESTABLISHMENT_DEBUG_WAITSEND : **debug.h**
- ZCL_KEY_ESTABLISHMENT_SUITE_CLIENT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_KEY_ESTABLISHMENT_SUITE_SERVER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_KEYPAD_LOCKOUT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAMP_ALARM_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAMP_BURN_HOURS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAMP_BURN_HOURS_TRIP_POINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAMP_MANUFACTURER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAMP_QUALITY_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_LAMP_RATED_HOURS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAMP_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAST_CONFIGURED_BY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LAST_SET_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LEVEL_CONTROL_CLUSTER_ID : **cluster-id.h**
- ZCL_LEVEL_CONTROL_DEBUG_EXECUTE : **debug.h**
- ZCL_LEVEL_CONTROL_DEBUG_ON : **debug.h**
- ZCL_LEVEL_CONTROL_DEBUG_PRINT : **debug.h**
- ZCL_LEVEL_CONTROL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_LEVEL_CONTROL_DEBUG_WAITSEND : **debug.h**
- ZCL_LEVEL_CONTROL_REMAINING_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LEVEL_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LIFETIME_ENERGY_CONSUMED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LIFETIME_RUNNING_HOURS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LIMIT_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LIMIT_TILT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOAD_CONTROL_EVENT_COMMAND_ID : **command-id.h**
- ZCL_LOCAL_TEMPERATURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCAL_TEMPERATURE_CALIBRATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCAL_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCATION_AGE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCATION_DATA_NOTIFICATION_COMMAND_ID : **command-id.h**
- ZCL_LOCATION_DATA_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_LOCATION_DESCRIPTION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCATION_METHOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCATION_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCK_DOOR_COMMAND_ID : **command-id.h**
- ZCL_LOCK_DOOR_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_LOCK_STATE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOCK_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOW_TEMP_DWELL_TRIP_POINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_LOW_TEMP_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAINS_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAINS_FREQUENCY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAINS_VOLTAGE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAINS_VOLTAGE_DWELL_TRIP_POINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAINS_VOLTAGE_MAX_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAINS_VOLTAGE_MIN_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MANUFACTURER_ID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MANUFACTURER_NAME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MANUFACTURER_SPECIFIC_MASK : **af.h**
- ZCL_MATCH_PROTOCOL_ADDRESS_COMMAND_ID : **command-id.h**
- ZCL_MATCH_PROTOCOL_ADDRESS_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_MAX_COMP_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_CONST_FLOW_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_CONST_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_CONST_SPEED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_CONST_TEMP_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_MAX_DURATION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_FLOW_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_NUMBER_OF_PERIODS_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_REJOIN_INTERVAL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_SPEED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAX_TEMP_EXPERIENCED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAXIMUM_INCOMING_TRANSFER_SIZE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MAXIMUM_OUTGOING_TRANSFER_SIZE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MEASUREMENT_LIGHT_SENSOR_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MESSAGE_CONFIRMATION_COMMAND_ID : **command-id.h**
- ZCL_MESSAGING_CLUSTER_CONFIRMATION_MASK : **messaging-client.h** , **messaging-server.h**
- ZCL_MESSAGING_CLUSTER_DURATION_UNTIL_CHANGED : **messaging-client.h** , **messaging-server.h**
- ZCL_MESSAGING_CLUSTER_END_TIME_NEVER : **messaging-client.h** , **messaging-server.h**
- ZCL_MESSAGING_CLUSTER_ID : **cluster-id.h**
- ZCL_MESSAGING_CLUSTER_IMPORTANCE_MASK : **messaging-client.h** , **messaging-server.h**
- ZCL_MESSAGING_CLUSTER_START_TIME_NOW : **messaging-client.h** , **messaging-server.h**
- ZCL_MESSAGING_CLUSTER_TRANSMISSION_MASK : **messaging-client.h** , **messaging-server.h**
- ZCL_MESSAGING_DEBUG_EXECUTE : **debug.h**
- ZCL_MESSAGING_DEBUG_ON : **debug.h**
- ZCL_MESSAGING_DEBUG_PRINT : **debug.h**
- ZCL_MESSAGING_DEBUG_PRINT_LINE : **debug.h**
- ZCL_MESSAGING_DEBUG_WAITSEND : **debug.h**
- ZCL_METER_SERIAL_NUMBER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_METERING_DEVICE_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_COMP_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_CONST_FLOW_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_CONST_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_CONST_SPEED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_CONST_TEMP_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_COOL_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_HEAT_SETPOINT_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_SETPOINT_DEAD_BAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIN_TEMP_EXPERIENCED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIRROR_IEEE_ADDRESS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIRROR_LAST_SYNC_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MIRROR_REMOVED_COMMAND_ID : **command-id.h**
- ZCL_MIRROR_SYNC_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MODEL_IDENTIFIER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_MOVE_COLOR_COMMAND_ID : **command-id.h**
- ZCL_MOVE_COMMAND_ID : **command-id.h**
- ZCL_MOVE_HUE_COMMAND_ID : **command-id.h**

- ZCL_MOVE_SATURATION_COMMAND_ID : **command-id.h**
- ZCL_MOVE_TO_COLOR_COMMAND_ID : **command-id.h**
- ZCL_MOVE_TO_COLOR_TEMPERATURE_COMMAND_ID : **command-id.h**
- ZCL_MOVE_TO_HUE_AND_SATURATION_COMMAND_ID : **command-id.h**
- ZCL_MOVE_TO_HUE_COMMAND_ID : **command-id.h**
- ZCL_MOVE_TO_LEVEL_COMMAND_ID : **command-id.h**
- ZCL_MOVE_TO_LEVEL_WITH_ON_OFF_COMMAND_ID : **command-id.h**
- ZCL_MOVE_TO_SATURATION_COMMAND_ID : **command-id.h**
- ZCL_MOVE_WITH_ON_OFF_COMMAND_ID : **command-id.h**
- ZCL_MULTIPLIER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NETWORK_KEY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NETWORK_KEY_SEQUENCE_NUMBER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NETWORK_KEY_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NETWORK_MANAGER_ADDRESS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NO_TIER_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NON_FUEL_DEBT_RECOVERY_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NON_FUEL_DEBT_RECOVERY_RATE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NON_FUEL_DEBT_REMAINING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NUMBER_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NUMBER_OF_DEMAND_SUBINTERVALS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NUMBER_OF_DEVICES_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NUMBER_RSSI_MEASUREMENTS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_NUMBER_TILT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OCCUPANCY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OCCUPANCY_SENSING_CLUSTER_ID : **cluster-id.h**
- ZCL_OCCUPANCY_SENSING_DEBUG_EXECUTE : **debug.h**
- ZCL_OCCUPANCY_SENSING_DEBUG_ON : **debug.h**
- ZCL_OCCUPANCY_SENSING_DEBUG_PRINT : **debug.h**
- ZCL_OCCUPANCY_SENSING_DEBUG_PRINT_LINE : **debug.h**
- ZCL_OCCUPANCY_SENSING_DEBUG_WAITSEND : **debug.h**
- ZCL_OCCUPANCY_SENSOR_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OCCUPIED_COOLING_SETPOINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OCCUPIED_HEATING_SETPOINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OFF_COMMAND_ID : **command-id.h**

- ZCL_ON_COMMAND_ID : **command-id.h**
- ZCL_ON_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ON_OFF_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ON_OFF_CLUSTER_ID : **cluster-id.h**
- ZCL_ON_OFF_DEBUG_EXECUTE : **debug.h**
- ZCL_ON_OFF_DEBUG_ON : **debug.h**
- ZCL_ON_OFF_DEBUG_PRINT : **debug.h**
- ZCL_ON_OFF_DEBUG_PRINT_LINE : **debug.h**
- ZCL_ON_OFF_DEBUG_WAITSEND : **debug.h**
- ZCL_ON_OFF_SWITCH_CONFIG_CLUSTER_ID : **cluster-id.h**
- ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_EXECUTE : **debug.h**
- ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_ON : **debug.h**
- ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_PRINT : **debug.h**
- ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_PRINT_LINE : **debug.h**
- ZCL_ON_OFF_SWITCH_CONFIG_DEBUG_WAITSEND : **debug.h**
- ZCL_ON_OFF_TRANSITION_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OPEN_LIMIT_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OPEN_LIMIT_TILT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OPEN_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OPERATION_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ORIGINATING_DEVICE_1_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ORIGINATING_DEVICE_2_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ORIGINATING_DEVICE_3_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ORIGINATING_DEVICE_4_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ORIGINATING_DEVICE_5_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OTA_BOOTLOAD_CLUSTER_ID : **cluster-id.h**
- ZCL_OUT_OF_SERVICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OUTDOOR_TEMPERATURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OUTLET_TEMPERATURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_OVER_TEMP_TOTAL_DWELL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PAN_ID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PANIC_COMMAND_ID : **command-id.h**
- ZCL_PARENT_RETRY_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PATH_LOSS_EXPONENT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PAYMENT_CONTROL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PHYSICAL_ENVIRONMENT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PHYSICAL_MAX_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PHYSICAL_MIN_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PI_COOLING_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PI_HEATING_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PIR_OCCUPIED_TO_UNOCCUPIED_DELAY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PIR_UNOCCUPIED_TO_OCCUPIED_DELAY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PIR_UNOCCUPIED_TO_OCCUPIED_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_POLARITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_POWER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_POWER_CONFIG_CLUSTER_ID : **cluster-id.h**
- ZCL_POWER_CONFIG_DEBUG_EXECUTE : **debug.h**
- ZCL_POWER_CONFIG_DEBUG_ON : **debug.h**
- ZCL_POWER_CONFIG_DEBUG_PRINT : **debug.h**

- ZCL_POWER_CONFIG_DEBUG_PRINT_LINE : **debug.h**
- ZCL_POWER_CONFIG_DEBUG_WAITSEND : **debug.h**
- ZCL_POWER_FACTOR_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_POWER_ON_FADE_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_POWER_ON_LEVEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_POWER_SOURCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRECONFIGURED_LINK_KEY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PREPAYMENT_CLUSTER_ID : **cluster-id.h**
- ZCL_PRESENT_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESET_READING_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_MAX_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_MAX_SCALED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_MEASUREMENT_CLUSTER_ID : **cluster-id.h**
- ZCL_PRESSURE_MEASUREMENT_DEBUG_EXECUTE : **debug.h**
- ZCL_PRESSURE_MEASUREMENT_DEBUG_ON : **debug.h**
- ZCL_PRESSURE_MEASUREMENT_DEBUG_PRINT : **debug.h**
- ZCL_PRESSURE_MEASUREMENT_DEBUG_PRINT_LINE : **debug.h**
- ZCL_PRESSURE_MEASUREMENT_DEBUG_WAITSEND : **debug.h**
- ZCL_PRESSURE_MIN_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_MIN_SCALED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_SCALE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_SCALED_TOLERANCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_SCALED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRESSURE_TOLERANCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PREVIOUS_DAY_CONSUMPTION_DELIVERED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PREVIOUS_DAY_CONSUMPTION_RECEIVED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PREVIOUS_DAY_MAX_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PREVIOUS_DAY_MAX_ENERGY_CARRIER_DEMAND_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PREVIOUS_DAY_MAX_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PREVIOUS_DAY_MIN_PRESSURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRICE_ACKNOWLEDGEMENT_COMMAND_ID : **command-id.h**
- ZCL_PRICE_CLUSTER_ALTERNATE_COST_DELIVERED_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_ALTERNATE_COST_TRAILING_DIGIT_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_ALTERNATE_COST_UNIT_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_DURATION_UNTIL_CHANGED : **price-client.h** , **price-server.h**
- ZCL_PRICE_CLUSTER_END_TIME_NEVER : **price-client.h** , **price-server.h**
- ZCL_PRICE_CLUSTER_GENERATION_PRICE_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_GENERATION_PRICE_RATIO_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_ID : **cluster-id.h**
- ZCL_PRICE_CLUSTER_MAXIMUM_RATE_LABEL_LENGTH : **af-types.h**
- ZCL_PRICE_CLUSTER_NUMBER_OF_BLOCK_THRESHOLDS_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_NUMBER_OF_EVENTS_ALL : **price-server.h**
- ZCL_PRICE_CLUSTER_PRICE_ACKNOWLEDGEMENT_MASK : **price-client.h** , **price-server.h**
- ZCL_PRICE_CLUSTER_PRICE_CONTROL_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_PRICE_RATIO_NOT_USED : **price-client.h**
- ZCL_PRICE_CLUSTER_START_TIME_NOW : **price-client.h** , **price-server.h**
- ZCL_PRICE_DEBUG_EXECUTE : **debug.h**
- ZCL_PRICE_DEBUG_ON : **debug.h**

- ZCL_PRICE_DEBUG_PRINT : **debug.h**
- ZCL_PRICE_DEBUG_PRINT_LINE : **debug.h**
- ZCL_PRICE_DEBUG_WAITSEND : **debug.h**
- ZCL_PRICE_DECREASE_RANDOMIZE_MINUTES_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRICE_INCREASE_RANDOMIZE_MINUTES_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PRICE_INVALID_INDEX : **price-server.h**
- ZCL_PROFILE_INTERVAL_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PROFILE_WIDE_COMMAND : **af.h**
- ZCL_PROPOSED_CHANGE_IMPLEMENTATION_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PROPOSED_CHANGE_PROVIDER_ID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PROPOSED_CHANGE_SUPPLY_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PROTOCOL_ADDRESS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PROTOCOL_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PUBLISH_BLOCK_PERIOD_COMMAND_ID : **command-id.h**
- ZCL_PUBLISH_PRICE_COMMAND_ID : **command-id.h**
- ZCL_PUMP_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PUMP_CONFIG_CONTROL_CLUSTER_ID : **cluster-id.h**
- ZCL_PUMP_CONFIG_CONTROL_DEBUG_EXECUTE : **debug.h**
- ZCL_PUMP_CONFIG_CONTROL_DEBUG_ON : **debug.h**
- ZCL_PUMP_CONFIG_CONTROL_DEBUG_PRINT : **debug.h**
- ZCL_PUMP_CONFIG_CONTROL_DEBUG_PRINT_LINE : **debug.h**
- ZCL_PUMP_CONFIG_CONTROL_DEBUG_WAITSEND : **debug.h**
- ZCL_PUMP_POWER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_PUMP_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_QUALITY_MEASURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_QUERY_NEXT_IMAGE_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_QUERY_NEXT_IMAGE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_QUERY_SPECIFIC_FILE_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_QUERY_SPECIFIC_FILE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_READ_ATTRIBUTES_COMMAND_ID : **command-id.h**
- ZCL_READ_ATTRIBUTES_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_READ_REPORTING_CONFIGURATION_COMMAND_ID : **command-id.h**
- ZCL_READ_REPORTING_CONFIGURATION_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_READING_SNAP_SHOT_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_READY_DATA_CLIENT_TO_SERVER_COMMAND_ID : **command-id.h**
- ZCL_READY_DATA_SERVER_TO_CLIENT_COMMAND_ID : **command-id.h**
- ZCL_RECALL_SCENE_COMMAND_ID : **command-id.h**
- ZCL_REJOIN_INTERVAL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELATIVE_HUMIDITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELATIVE_HUMIDITY_DISPLAY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELATIVE_HUMIDITY_MAX_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELATIVE_HUMIDITY_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELATIVE_HUMIDITY_MEASUREMENT_CLUSTER_ID : **cluster-id.h**
- ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_EXECUTE : **debug.h**
- ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_ON : **debug.h**
- ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_PRINT : **debug.h**
- ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_PRINT_LINE : **debug.h**
- ZCL_RELATIVE_HUMIDITY_MEASUREMENT_DEBUG_WAITSEND : **debug.h**
- ZCL_RELATIVE_HUMIDITY_MIN_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_RELATIVE_HUMIDITY_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELATIVE_HUMIDITY_TOLERANCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RELIABILITY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_REMAINING_BATTERY_LIFE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_REMOTE_SENSING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_REMOVE_ALL_GROUPS_COMMAND_ID : **command-id.h**
- ZCL_REMOVE_ALL_SCENES_COMMAND_ID : **command-id.h**
- ZCL_REMOVE_ALL_SCENES_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_REMOVE_GROUP_COMMAND_ID : **command-id.h**
- ZCL_REMOVE_GROUP_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_REMOVE_MIRROR_COMMAND_ID : **command-id.h**
- ZCL_REMOVE_SCENE_COMMAND_ID : **command-id.h**
- ZCL_REMOVE_SCENE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_REPORT_ATTRIBUTES_COMMAND_ID : **command-id.h**
- ZCL_REPORT_EVENT_STATUS_COMMAND_ID : **command-id.h**
- ZCL_REPORT_RSSI_MEASUREMENTS_COMMAND_ID : **command-id.h**
- ZCL_REPORTING_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_REQUEST_FAST_POLL_MODE_COMMAND_ID : **command-id.h**
- ZCL_REQUEST_FAST_POLL_MODE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_REQUEST_MIRROR_COMMAND_ID : **command-id.h**
- ZCL_REQUEST_MIRROR_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_REQUEST_OWN_LOCATION_COMMAND_ID : **command-id.h**
- ZCL_REQUEST_TUNNEL_COMMAND_ID : **command-id.h**
- ZCL_REQUEST_TUNNEL_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_RESET_ALARM_COMMAND_ID : **command-id.h**
- ZCL_RESET_ALARM_LOG_COMMAND_ID : **command-id.h**
- ZCL_RESET_ALL_ALARMS_COMMAND_ID : **command-id.h**
- ZCL_RESET_STARTUP_PARAMETERS_COMMAND_ID : **command-id.h**
- ZCL_RESET_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_RESET_TO_FACTORY_DEFAULTS_COMMAND_ID : **command-id.h**
- ZCL_RESTART_DEVICE_COMMAND_ID : **command-id.h**
- ZCL_RESTART_DEVICE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_RESTORE_STARTUP_PARAMETERS_COMMAND_ID : **command-id.h**
- ZCL_RESTORE_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_RH_DEHUMIDIFICATION_SETPOINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_RSSI_LOCATION_CLUSTER_ID : **cluster-id.h**
- ZCL_RSSI_PING_COMMAND_ID : **command-id.h**
- ZCL_RSSI_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_RSSI_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_SAMPLE_MFG_SPECIFIC_CLUSTER_ID : **cluster-id.h**
- ZCL_SAMPLE_MFG_SPECIFIC_OFF_WITH_TRANSITION_COMMAND_ID : **command-id.h**
- ZCL_SAMPLE_MFG_SPECIFIC_ON_WITH_TRANSITION_COMMAND_ID : **command-id.h**
- ZCL_SAMPLE_MFG_SPECIFIC_TOGGLE_WITH_TRANSITION_COMMAND_ID : **command-id.h**
- ZCL_SAMPLE_MFG_SPECIFIC_TRANSITION_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SAVE_STARTUP_PARAMETERS_COMMAND_ID : **command-id.h**
- ZCL_SAVE_STARTUP_PARAMETERS_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_SCAN_ATTEMPTS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SCENE_COUNT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SCENE_NAME_SUPPORT_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_SCENE_VALID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SCENES_CLUSTER_ID : **cluster-id.h**
- ZCL_SCENES_CLUSTER_MAXIMUM_NAME_LENGTH : **af-types.h**
- ZCL_SCENES_DEBUG_EXECUTE : **debug.h**
- ZCL_SCENES_DEBUG_ON : **debug.h**
- ZCL_SCENES_DEBUG_PRINT : **debug.h**
- ZCL_SCENES_DEBUG_PRINT_LINE : **debug.h**
- ZCL_SCENES_DEBUG_WAITSEND : **debug.h**
- ZCL_SCENES_GLOBAL_SCENE_GROUP_ID : **af-types.h**
- ZCL_SELECT_AVAILABLE_EMERGENCY_CREDIT_COMMAND_ID : **command-id.h**
- ZCL_SEND_PINGS_COMMAND_ID : **command-id.h**
- ZCL_SENSING_LIGHT_SENSOR_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SET_ABSOLUTE_LOCATION_COMMAND_ID : **command-id.h**
- ZCL_SET_DEVICE_CONFIGURATION_COMMAND_ID : **command-id.h**
- ZCL_SETPOINT_RAISE_LOWER_COMMAND_ID : **command-id.h**
- ZCL_SETPOINTS_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SETPOINTS_TILT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SHADE_CONFIG_CLOSED_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SHADE_CONFIG_CLUSTER_ID : **cluster-id.h**
- ZCL_SHADE_CONFIG_DEBUG_EXECUTE : **debug.h**
- ZCL_SHADE_CONFIG_DEBUG_ON : **debug.h**
- ZCL_SHADE_CONFIG_DEBUG_PRINT : **debug.h**
- ZCL_SHADE_CONFIG_DEBUG_PRINT_LINE : **debug.h**
- ZCL_SHADE_CONFIG_DEBUG_WAITSEND : **debug.h**
- ZCL_SHADE_CONFIG_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SHADE_CONFIG_MOTOR_STEP_SIZE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SHADE_CONFIG_PHYSICAL_CLOSED_LIMIT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SHADE_CONFIG_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SHORT_ADDRESS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SIMPLE_METERING_CLUSTER_ID : **cluster-id.h**
- ZCL_SIMPLE_METERING_DEBUG_EXECUTE : **debug.h**
- ZCL_SIMPLE_METERING_DEBUG_ON : **debug.h**
- ZCL_SIMPLE_METERING_DEBUG_PRINT : **debug.h**
- ZCL_SIMPLE_METERING_DEBUG_PRINT_LINE : **debug.h**
- ZCL_SIMPLE_METERING_DEBUG_WAITSEND : **debug.h**
- ZCL_SITE_ID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SPEED_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SQUAWK_COMMAND_ID : **command-id.h**
- ZCL_SQUAWK_INFO_ARMED : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_DISARMED : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_HIGH : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_LEVEL_MASK : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_LOW : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_MEDIUM : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_MODE_MASK : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_NO_STROBE : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_STROBE : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_STROBE_MASK : **ias-wd-server.h**
- ZCL_SQUAWK_INFO_VERY_HIGH : **ias-wd-server.h**

- ZCL_STACK_PROFILE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_STACK_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_STANDARD_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_STANDING_CHARGE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_START_OF_BLOCK_PERIOD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_START_RANDOMIZE_MINUTES_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_START_WARNING_COMMAND_ID : **command-id.h**
- ZCL_STARTUP_CONTROL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_STATUS_FLAGS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_STEP_COLOR_COMMAND_ID : **command-id.h**
- ZCL_STEP_COMMAND_ID : **command-id.h**
- ZCL_STEP_HUE_COMMAND_ID : **command-id.h**
- ZCL_STEP_SATURATION_COMMAND_ID : **command-id.h**
- ZCL_STEP_WITH_ON_OFF_COMMAND_ID : **command-id.h**
- ZCL_STOP_COMMAND_ID : **command-id.h**
- ZCL_STOP_RANDOMIZE_MINUTES_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_STOP_WITH_ON_OFF_COMMAND_ID : **command-id.h**
- ZCL_STORE_SCENE_COMMAND_ID : **command-id.h**
- ZCL_STORE_SCENE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_SUMMATION_FORMATTING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SUPPLY_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SUPPLY_STATUS_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_SWITCH_ACTIONS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SWITCH_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_SYSTEM_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TEMP_MAX_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TEMP_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TEMP_MEASUREMENT_CLUSTER_ID : **cluster-id.h**
- ZCL_TEMP_MEASUREMENT_DEBUG_EXECUTE : **debug.h**
- ZCL_TEMP_MEASUREMENT_DEBUG_ON : **debug.h**
- ZCL_TEMP_MEASUREMENT_DEBUG_PRINT : **debug.h**
- ZCL_TEMP_MEASUREMENT_DEBUG_PRINT_LINE : **debug.h**
- ZCL_TEMP_MEASUREMENT_DEBUG_WAITSEND : **debug.h**
- ZCL_TEMP_MIN_MEASURED_VALUE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TEMP_TOLERANCE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TEMPERATURE_DISPLAY_MODE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TEMPERATURE_FORMATTING_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TEMPERATURE_UNIT_OF_MEASURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TERMINATE_KEY_ESTABLISHMENT_COMMAND_ID : **command-id.h**
- ZCL_THERMOSTAT_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_THERMOSTAT_CLUSTER_ID : **cluster-id.h**
- ZCL_THERMOSTAT_DEBUG_EXECUTE : **debug.h**
- ZCL_THERMOSTAT_DEBUG_ON : **debug.h**
- ZCL_THERMOSTAT_DEBUG_PRINT : **debug.h**
- ZCL_THERMOSTAT_DEBUG_PRINT_LINE : **debug.h**
- ZCL_THERMOSTAT_DEBUG_WAITSEND : **debug.h**
- ZCL_THERMOSTAT_OCCUPANCY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_THERMOSTAT_UI_CONFIG_CLUSTER_ID : **cluster-id.h**

- ZCL_THERMOSTAT_UI_CONFIG_DEBUG_EXECUTE : **debug.h**
- ZCL_THERMOSTAT_UI_CONFIG_DEBUG_ON : **debug.h**
- ZCL_THERMOSTAT_UI_CONFIG_DEBUG_PRINT : **debug.h**
- ZCL_THERMOSTAT_UI_CONFIG_DEBUG_PRINT_LINE : **debug.h**
- ZCL_THERMOSTAT_UI_CONFIG_DEBUG_WAITSEND : **debug.h**
- ZCL_THRESHOLD_DIVISOR_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_THRESHOLD_MULTIPLIER_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER10_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER11_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_TIER12_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER12_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER13_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER14_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_TIER15_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER15_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER1_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER2_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_TIER3_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER3_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER4_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER5_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_TIER6_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER6_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER7_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER8_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_TIER9_BLOCK10_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK11_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK12_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK13_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK14_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK15_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK16_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK1_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK2_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK3_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK4_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK5_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK6_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK7_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK8_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_BLOCK9_PRICE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIER9_PRICE_LABEL_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIME_BETWEEN_SCANS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIME_CLUSTER_ID : **cluster-id.h**
- ZCL_TIME_DEBUG_EXECUTE : **debug.h**
- ZCL_TIME_DEBUG_ON : **debug.h**
- ZCL_TIME_DEBUG_PRINT : **debug.h**
- ZCL_TIME_DEBUG_PRINT_LINE : **debug.h**
- ZCL_TIME_DEBUG_WAITSEND : **debug.h**
- ZCL_TIME_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TIME_ZONE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOGGLE_COMMAND_ID : **command-id.h**
- ZCL_TOP_UP_AMOUNT_1_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_AMOUNT_2_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_AMOUNT_3_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_AMOUNT_4_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_AMOUNT_5_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_DATE_TIME_1_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_DATE_TIME_2_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_DATE_TIME_3_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_DATE_TIME_4_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TOP_UP_DATE_TIME_5_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TRANSFER_DATA_CLIENT_TO_SERVER_COMMAND_ID : **command-id.h**
- ZCL_TRANSFER_DATA_ERROR_CLIENT_TO_SERVER_COMMAND_ID : **command-id.h**
- ZCL_TRANSFER_DATA_ERROR_SERVER_TO_CLIENT_COMMAND_ID : **command-id.h**
- ZCL_TRANSFER_DATA_SERVER_TO_CLIENT_COMMAND_ID : **command-id.h**
- ZCL_TRANSFER_NPDU_COMMAND_ID : **command-id.h**
- ZCL_TRUST_CENTER_ADDRESS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TRUST_CENTER_MASTER_KEY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_TUNNELING_CLUSTER_ID : **cluster-id.h**
- ZCL_TUNNELING_CLUSTER_INVALID_TUNNEL_ID : **tunneling-server.h**
- ZCL_ULTRASONIC_OCCUPIED_TO_UNOCCUPIED_DELAY_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ULTRASONIC_UNOCCUPIED_TO_OCCUPIED_DELAY_ATTRIBUTE_ID : **attribute-id.h**

- ZCL_ULTRASONIC_UNOCCUPIED_TO_OCCUPIED_THRESHOLD_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_UNIT_OF_MEASURE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_UNLOCK_DOOR_COMMAND_ID : **command-id.h**
- ZCL_UNLOCK_DOOR_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_UNOCCUPIED_COOLING_SETPOINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_UNOCCUPIED_HEATING_SETPOINT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_UPGRADE_END_REQUEST_COMMAND_ID : **command-id.h**
- ZCL_UPGRADE_END_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_UPGRADE_SERVER_ID_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_USE_INSECURE_JOIN_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_UTILITY_ENROLLMENT_GROUP_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_VALID_UNTIL_TIME_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_VELOCITY_LIFT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_VERSION_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_VIEW_GROUP_COMMAND_ID : **command-id.h**
- ZCL_VIEW_GROUP_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_VIEW_SCENE_COMMAND_ID : **command-id.h**
- ZCL_VIEW_SCENE_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_VOLUME_PER_REPORT_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_WARNING_INFO_BURGLAR : **ias-wd-server.h**
- ZCL_WARNING_INFO_EMERGENCY : **ias-wd-server.h**
- ZCL_WARNING_INFO_FIRE : **ias-wd-server.h**
- ZCL_WARNING_INFO_MODE_MASK : **ias-wd-server.h**
- ZCL_WARNING_INFO_NO_STROBE : **ias-wd-server.h**
- ZCL_WARNING_INFO_STOP : **ias-wd-server.h**
- ZCL_WARNING_INFO_STROBE : **ias-wd-server.h**
- ZCL_WARNING_INFO_STROBE_MASK : **ias-wd-server.h**
- ZCL_WATER_SPECIFIC_ALARM_MASK_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_WINDOW_COVERING_CLUSTER_ID : **cluster-id.h**
- ZCL_WINDOW_COVERING_DEBUG_EXECUTE : **debug.h**
- ZCL_WINDOW_COVERING_DEBUG_ON : **debug.h**
- ZCL_WINDOW_COVERING_DEBUG_PRINT : **debug.h**
- ZCL_WINDOW_COVERING_DEBUG_PRINT_LINE : **debug.h**
- ZCL_WINDOW_COVERING_DEBUG_WAITSEND : **debug.h**
- ZCL_WINDOW_COVERING_DOWN_CLOSE_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_GO_TO_LIFT_PERCENTAGE_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_GO_TO_LIFT_SETPOINT_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_GO_TO_LIFT_VALUE_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_GO_TO_PROGRAM_SETPOINT_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_GO_TO_TILT_PERCENTAGE_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_GO_TO_TILT_SETPOINT_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_GO_TO_TILT_VALUE_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_STOP_COMMAND_ID : **command-id.h**
- ZCL_WINDOW_COVERING_UP_OPEN_COMMAND_ID : **command-id.h**
- ZCL_WRITE_ATTRIBUTES_COMMAND_ID : **command-id.h**
- ZCL_WRITE_ATTRIBUTES_NO_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_WRITE_ATTRIBUTES_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_WRITE_ATTRIBUTES_UNDIVIDED_COMMAND_ID : **command-id.h**
- ZCL_ZONE_ENROLL_REQUEST_COMMAND_ID : **command-id.h**

- ZCL_ZONE_ENROLL_RESPONSE_COMMAND_ID : **command-id.h**
- ZCL_ZONE_STATE_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ZONE_STATUS_ATTRIBUTE_ID : **attribute-id.h**
- ZCL_ZONE_STATUS_CHANGE_NOTIFICATION_COMMAND_ID : **command-id.h**
- ZCL_ZONE_TYPE_ATTRIBUTE_ID : **attribute-id.h**
- ZDO_COMMANDS : **zdo-cli.h**
- ZIGBEE_2006_STACK_VERSION : **ota.h**
- ZIGBEE_2007_STACK_VERSION : **ota.h**
- ZIGBEE_PRO_STACK_VERSION : **ota.h**
- ZONE_ID_MAP_SECTION_SIZE : **ias-zone-table.h**
- ZONE_TABLE_MAP_SIZE : **ias-zone-table.h**
- ZONE_TABLE_SIZE : **ias-zone-table.h**
- ZONE_TABLE_VALUE_DEFAULT_16 : **ias-zone-table.h**
- ZONE_TABLE_VALUE_DEFAULT_8 : **ias-zone-table.h**