CEL California Eastern Laboratories

ZIC2410 Series

ZIC2410 User Guide
ZigBee™ Device Profile
0005-05-08-09-001

(Rev B)

Table of Contents

1 INTRODUCTION

This document explains the ZigBee Device Profile (ZDP) of the ***CEL ZigBee Stack***.

CEL ZigBee Stack, ZigBee Stack Library, provided by CEL supports IEEE 802.15.4 and ZigBee2006.

CEL ZigBee Stack includes API functions to support the functionality defined in ZigBee standard and communication functions defined in IEEE 802.15.4.

1.1 REFERENCED DOCUMENTS

Category	*filename [.pdf]*	Document Name
Stack API References	*zic07_aps_api*	***ZIC2410 User Guide Application Support Layer API Reference***

2 ZIGBEE2006 LAYER AND FRAME STRUCTURE

Figure 1 and Figure 2 show the ZigBee2006 layers and the frame structure for each layer.

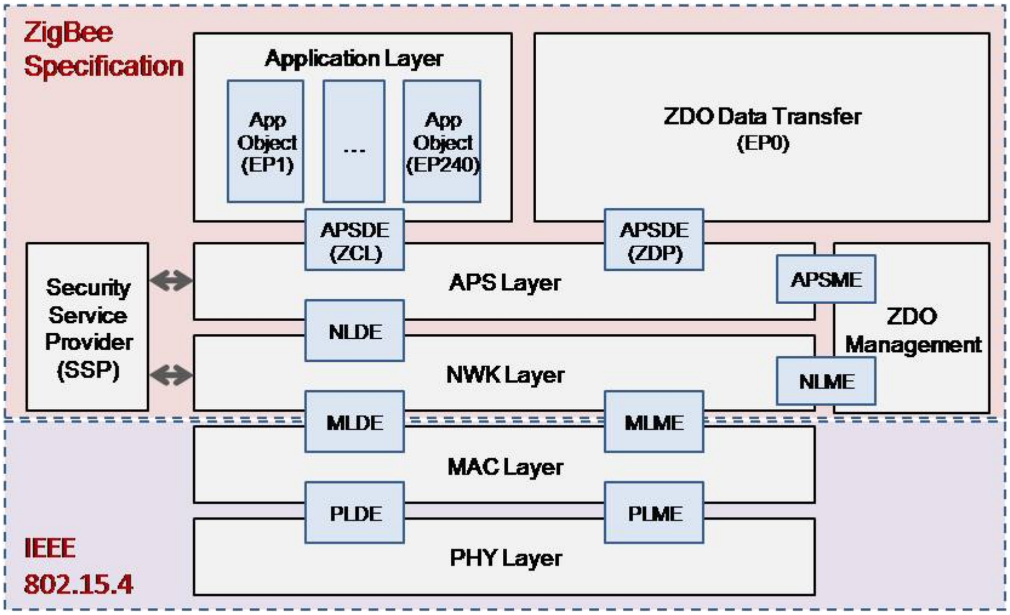

Figure 1 – ZigBee2006 Layers

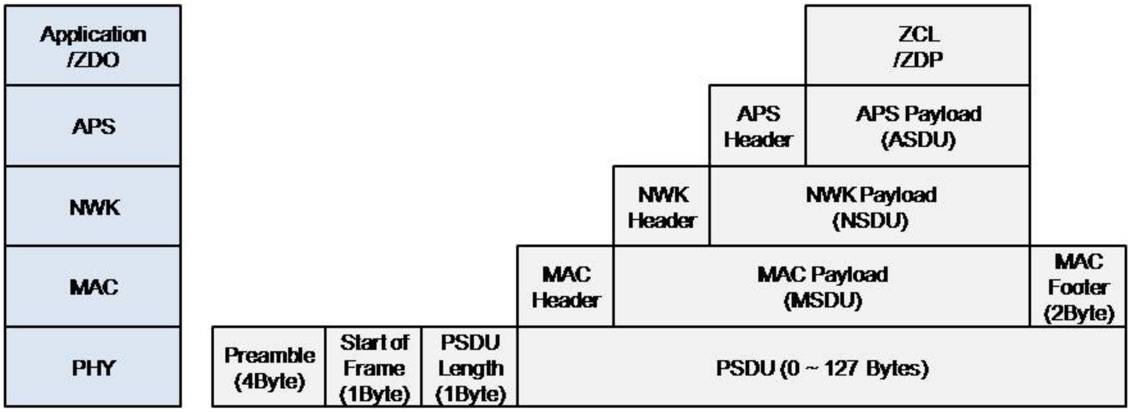

Figure 2 – ZigBee2006 Frame Structure

3 ZIGBEE DEVICE PROFILE (ZDP)

ZDP is a profile defined to support the basic function of ZigBee network and it consists of several commands. Therefore, all ZigBee devices should support ZDP. ZDP supports following services.

- Device and Service Discovery
- End Device Bind
- Bind and Unbind
- Binding Table Management

3.1 Frame Format

The following describes each ZDP frame.

Octet: 1	Variable
Transaction Sequence Number	Transaction Data

3.1.1 Transaction Sequence Number

The transaction sequence number field is eight bits in length and specifies an identification number for the ZDP transaction so that a response command frame can be related to the request frame. The application object itself shall maintain an eight-bit counter that is copied into this field and incremented by one for each command sent. When a value of 0xff is reached, the next command shall restart the counter with a value of 0x00

3.1.2 Transaction Data

The transaction data field has a variable length and contains the data for the individual ZDP transaction. The format and length of this field is dependent on the command being transmitted.

3.1 Client and Server Service

A ZDP command is divided into Client service and Server service according to the command's purpose. Client service is used for requesting the information of a target device (Server) or announcing its own information to a target device. Server service is used for send the requested information as a response for a Client service.

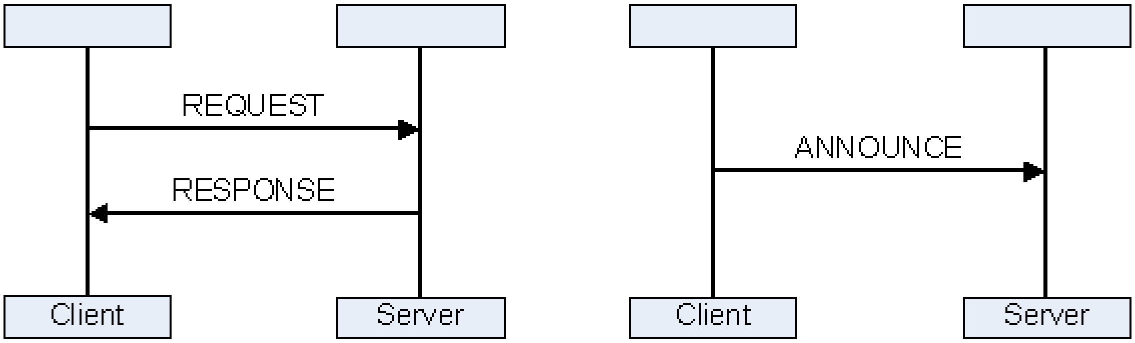

Figure 3 – ZDP Command Flow: Client and Server

3.2 ZDP Command List

Commands are divided into Mandatory (M) and Optional (O) command. Generally, for each request, there is a response.

Each command has its own ClusterID. The ClusterID is a 2-byte field which included the APS Header.

3.2.1 Device and Service Discovery Client Services

The commands used when discovering the functions and services supported by a device. The major commands are described in the sections of this document indicated in 'Sec' Column of Table 1. For descriptions of the other commands, please refer to paragraphs 2.4.3.1 and 2.4.4.1 of the ZigBee Specification, Doc # 053474r17.

Table 1 – Device and Service Discovery Commands

C: Client, S: Server *M: Mandatory, O: Optional*

Request					Response			
Sec	**Command**	**ClusID**	**C**	**S**	**Sec**	**Command**	**ClusID**	**S**
0	NWK_Addr_req	0x0000	O	M	4.2	NWK_Addr_rsp	0x8000	M
4.3	IEEE_Addr_req	0x0001	O	M	4.4	IEEE_Addr_rsp	0x8001	M
4.5	Node_Desc_req	0x0002	O	M	4.6	Node_Desc_rsp	0x8002	M
4.7	Power_Desc_req	0x0003	O	M	4.8	Power_Desc_rsp	0x8003	M
4.9	Simple_Desc_req	0x0004	O	M	4.10	Simple_Desc_rsp	0x8004	M
4.11	Active_EP_req	0x0005	O	M	4.12	Active_EP_rsp	0x8005	M
4.13	Match_Desc_req	0x0006	O	M	4.14	Match_Desc_rsp	0x8006	M
	Complex_Desc_req	0x0010	O	O		Complex_Desc_rsp	0x8010	O
	User_Desc_req	0x0011	O	O		User_Desc_rsp	0x8011	O
	Discovery_Cache_req	0x0012	O	M		Discovery_Cache_rsp	0x8012	M
4.15	End_Device_Annce	0x0013	O	O		--	--	--
	User_Desc_Set	0x0014	O	O		User_Desc_Conf	0x8014	O
4.16	System_Server_Discover_req	0x0015	O	M	4.17	System_Server_Discover_rsp	0x8015	M
	Discovery_Store_req	0x0016	O	O		Discovery_Store_rsp	0x8016	O
	Node_Desc_Store_req	0x0017	O	O		Node_Desc_Store_rsp	0x8017	O
	Power_Desc_Store_req	0x0018	O	O		Power_Desc_Store_rsp	0x8018	O
	Active_EP_Store_req	0x0019	O	O		Active_EP_Store_rsp	0x8019	O
	Simple_Desc_Store_req	0x001a	O	O		Simple_Desc_Store_rsp	0x801a	O
	Remove_Node_Cache_req	0x001b	O	O		Remove_Node_Cache_rsp	0x801b	O
	Find_Node_Cache_req	0x001c	O	M		Find_Node_Cache_rsp	0x801c	O

3.2.2 End Device Bind, Unbind and Bind Management

The commands used for End Device Binding and Unbinding. . The major commands are described in the sections of this document indicated in 'Sec' Column of Table 2. For descriptions of the other commands, please refer to paragraphs 2.4.3.2 and 2.4.4.2 of the ZigBee Specification, Doc # 053474r17.

Table 2 – End Device Bind, Unbind and Bind Management Commands

C: Client, S: Server *M: Mandatory, O: Optional*

Request					Response			
Sec	**Command**	**ClusID**	**C**	**S**	**Sec**	**Command**	**ClusID**	**S**
4.18	End_Dev_Bind_req	0x0020	O	O	4.19	End_Dev_Bind_rsp	0x8020	O
4.20	Bind_req	0x0021	O	O	4.21	Bind_rsp	0x8021	O
4.22	Unbind_req	0x0022	O	O	4.23	Unbind_rsp	0x8022	O
	Bind_Register_req	0x0023	O	O		Bind_Register_rsp	0x8023	O
	Replace_Device_req	0x0024	O	O		Replace_Device_rsp	0x8024	O
	Store_Bkup_Bind_Entry_req	0x0025	O	O		Store_Bkup_Bind_Entry_rsp	0x8025	O
	Rm_Bkup_Bind_Entry_req	0x0026	O	O		Rm_Bkup_Bind_Entry_rsp	0x8026	O
	Backup_Bind_Table_req	0x0027	O	O		Backup_Bind_Table_rsp	0x8027	O
	Recover_Bind_Table_req	0x0028	O	O		Recover_Bind_Table_rsp	0x8028	O
	Backup_Source_Bind_req	0x0029	O	O		Backup_Source_Bind_rsp	0x8029	O
	Recover_Source_Bind_req	0x002a	O	O		Recover_Source_Bind_rsp	0x802a	O

3.2.3 Network Management

Commands for network management. For descriptions of these commands, please refer to paragraphs 2.4.3.3 and 2.4.4.3 of the ZigBee Specification, Doc # 053474r17.

Table 3 – Network Management Commands

C: Client, S: Server *M: Mandatory, O: Optional*

Request				Response		
Command	**ClusID**	**C**	**S**	**Command**	**ClusID**	**S**
Mgmt_NWK_Disc_req	0x0030	O	O	Mgmt_NWK_Disc_rsp	0x8030	O
Mgmt_LQI_req	0x0031	O	O	Mgmt_LQI_rsp	0x8031	O
Mgmt_Rtg_req	0x0032	O	O	Mgmt_Rtg_rsp	0x8032	O
Mgmt_Bind_req	0x0033	O	O	Mgmt_Bind_rsp	0x8033	O
Mgmt_Leave_req	0x0034	O	O	Mgmt_Leave_rsp	0x8034	O
Mgmt_Direct_Join_req	0x0035	O	O	Mgmt_Direct_Join_rsp	0x8035	O
Mgmt_Permit_Join_req	0x0036	O	M	Mgmt_Permit_Join_rsp	0x8036	M
Mgmt_Cache_req	0x0037	O	O	Mgmt_Cache_rsp	0x8037	O

3.2.4 ZDP Status

The following shows the status field values of the responses for ZDP requests.

Response Status Fields	Value	Description
SUCCESS	0x00	Requested operation or transmission is completed successfully
	0x01-0x7F	Reserved
INV_REQUESTTYPE	0x80	The supplied request type is not valid
DEVICE_NOT_FOUND	0x81	The requested device does not exist on a remote device following a child descriptor request to a parent.
INVALID_EP	0x82	The requested end point is equal to 0x00 or between 0xF1and 0xFF.
NOT_ACTIVE	0x83	The requested endpoint is not described by a simple descriptor
NOT_SUPPORTED	0x84	The requested optional feature is not supported by the target device.
TIMEOUT	0x85	A timeout has occurred with the requested operation.
NO_MATCH	0x86	The end device bind request is unsuccessful due to a failure to match any suitable clusters.
	0x87	Reserved
NO_ENTRY	0x88	The unbind request was unsuccessful due to the coordinator or source device not having an entry in the Binding table to unbind.
NO_DESCRIPTOR	0x89	A child descriptor was not available following a discovery request to a parent
INSUFFICIENT_SPACE	0x8A	The device does not have storage space to support the requested operation.
NOT_PERMITTED	0x8B	The device cannot support the requested operation.
TABLE_FULL	0x8C	The device does not have table space to support the requested operation.
	0x8E-0xFF	Reserved

4 THE MAJOR ZDP COMMANDS

ZDO sends ZDP command to obtain the information from other devices in the network. All ZDP data is sent from the Least Significant Byte (LSB) first (using Little Endian format). For example, when sending 16-bit NWK Address, first the low byte (NWK Address [7:0]) of the NWK Address is sent and then the high byte (NWK Address[15:8]).

ZDPS_NWK_ADDR_req (Section 0)	NWK_ADDR_REQ
ZDPS_NWK_ADDR_rsp (Section 4.2)	NWK_ADDR_RSP
ZDPS_IEEE_ADDR_req (Section 4.3)	IEEE_ADDR_REQ
ZDPS_IEEE_ADDR_rsp (Section 4.4)	IEEE_ADDR_RSP
ZDPS_NODE_DESC_req (Section 4.5)	NODE_DESC_REQ
ZDPS_NODE_DESC_rsp (Section 4.6)	NODE_DESC_RSP
ZDPS_POWER_DESC_req (Section 4.7)	POWER_DESC_REQ
ZDPS_POWER_DESC_rsp (Section 4.8)	POWER_DESC_RSP
ZDPS_SIMP_DESC_req (Section 4.9)	SIMP_DESC_REQ
ZDPS_SIMP_DESC_rsp (Section 4.10)	SIMP_DESC_RSP
ZDPS_ACTIVE_EP_req (Section 4.11)	ACTIVE_EP_REQ
ZDPS_ACTIVE_EP_rsp (Section 4.12)	ACTIVE_EP_RSP
ZDPS_MATCH_DESC_req (Section 4.13)	MATCH_DESC_REQ
ZDPS_MATCH_DESC_rsp (Section 4.14)	MATCH_DESC_RSP
ZDPS_END_DEVICE_annce (Section 4.15)	END_DEV_ANNCE
ZDPS_SYS_SERV_DISC_req (Section 4.16)	SYS_SERV_DISC_REQ
ZDPS_SYS_SERV_DISC_rsp (Section 4.17)	SYS_SERV_DISC_RSP
ZDPS_END_DEVICE_BIND_req (Section 4.18)	END_DEV_BIND_REQ
ZDPS_END_DEVICE_BIND_rsp (Section 4.19)	END_DEV_BIND_RSP
ZDPS_BIND_req (Section 4.20)	BIND_REQ
ZDPS_BIND_rsp (Section 4.21)	BIND_RSP
ZDPS_UNBIND_req (Section 4.22)	
ZDPS_UNBIND_rsp (Section 4.23)	

4.1 NWK_Addr_req (ClusterID = 0x0000)

The command used when a Local device inquires as to the 16-bit NWK Address of a remote device based on its 64-bit IEEE address. Destination address of this command is Unicast or Broadcast (0xFFFD) to all device for which RxOnWhenIdle is TRUE.

4.1.1 Transaction Data Format

Transaction Data		
Octet : 8	1	1
IEEEAddr	RequestType	StartIndex

4.1.2 Field Description

Name	Range	Description
IEEEAddr	UINT8[8]	The IEEE address to be matched by the Remote Device.
RequestType	0x00 or 0x01	0x00=Single Device Response. There is no StartIndex field. 0x01=Extended Response. There is a StartIndex
StartIndex	UINT8	The Starting Index for the requested elements of associated device list for Remote Device if RequestType is 1

4.2 NWK_Addr_rsp (ClusterID = 0x8000)

The Remote device, which receives NWK_Addr_req, compares IEEEAddr field and its own IEEE Address or IEEE Address of associated device list which it has. If there is a corresponding address, NWK_Addr_rsp is sent. RequestType is 1 (Extended Response), also associated device list from the position, which StartIndex indicates, is sent

Destination address of NWK_addr_rsp is the device which sends NWK_addr_req and it sends by unicast method. Remote Device checks whether IEEE address of Remote device and IEEEAddr of NWK_addr_req is matched or not.

4.2.1 Transaction Data Format

Transaction Data					
Octet : 1	8	2	1	1	Variable
Status	IEEEAddr RemoteDev	NwkAddr Remote Dev	NumAssocDev	StartIndex	NwkAddr AssocDevList

4.2.2 Field Description

Name	Range	Description
Status	UINT8	The status of the NWK_Addr_req command: SUCCESS INV_REQUESTTYPE DEVICE_NOT_FOUND
IEEEAddrRemoteDev	UINT8 Array	64-bit IEEEAddress for the Remote device
NwkAddrRemoteDev	UINT16	16-bit NWK Address for the Remote device
NumAssocDev	UINT8	Existing field when RequestType of NWK_Addr_req is 1. It means the number of device list to be included NwkAddrAssocDevList field.
StartIndex	UINT8	Starting index in the list of associated devices if RequestType is 1 and associated devices exist.
NwkAddrAssocDevList	UINT8 Array	List of 16-bit NWK Addresses for devices associated to the Remote device if RequestType of NWK_Addr_req is 1 and NumAssocDev value not 0.

4.3 IEEE_Addr_req (ClusterID = 0x0001)

This is the command used when Local device inquires as to 64-bit IEEE address based on the 16-bit NWK Address of remote device. Destination address of this command is NWK Address of remote device and it is sent by unicast method.

4.3.1 Transaction Data Format

Transaction Data		
Octet : 2	1	1
NwkAddrOfInterest	RequestType	StartIndex

4.3.2 Field Description

Name	**Range**	**Description**
NwkAddrOfInterest	UINT16	16-bit NWK Address of remote device
RequestType	0 or 1	0x00=Single Device Response. 0x01=Extended Response
StartIndex	UINT8	The start index of the requested elements of the associated device list if RequestType is 1.

4.4 IEEE_Addr_rsp (ClusterID = 0x8001)

When a remote device receives an IEEE_Addr_req, it sends an IEEE_Addr_rsp which includes its own IEEE Address.

Destination address of IEEE_addr_rsp is the device which sent the IEEE_addr_req when sent by unicast method.

4.4.1 Transaction Data Format

Transaction Data					
Octet : 1	8	2	1	1	Variable
Status	IEEEAddr RemoteDev	NwkAddr Remote Dev	NumAssocDev	StartIndex	NwkAddr AssocDevList

4.4.2 Field Description

Name	**Range**	**Description**
Status	UINT8	The status of the IEEE_Addr_req command: SUCCESS INV_REQUESTTYPE DEVICE_NOT_FOUND
IEEEAddrRemoteDev	UINT8 Array	64-bit IEEE Address of remote device.
NwkAddrRemoteDev	UINT16	16-bit NWK Address of remote device.
NumAssocDev	UINT8	The number of the device included to NwkAddrAssocDevList field if RequestType of IEEE_Addr_req is 1.
StartIndex	UINT8	The StartIndex of IEEE_Addr_req if RequestType of IEEE_Addr_req is 1.
NwkAddrAssocDevList	UINT8 Array	List of 16-bit NWK Addresses for devices associated to the Remote device if RequestType of IEEE_Adr_req is 1 and NumAssocDev value is not 0.

4.5 Node_Desc_req (ClusterID = 0x0002)

This is the command used when the local device inquires as to the node descriptor of a remote device. Destination address of this command is the remote device or the cache device which includes the information of discovery for the remote device and it is sent by unicast method.

4.5.1 Transaction Data Format

Transaction Data
Octet : 2
NwkAddrOfInterest

4.5.2 Field Description

Name	Range	Description
NwkAddrOfInterest	UINT16	16-bit NWK Address of remote device

4.6 Node_Desc_rsp (ClusterID = 0x8002)

Remote device, which receives Node_Desc_req, or cache device sends Node_Desc_rsp based on node descriptor information.

4.6.1 Transaction Data Format

Transaction Data		
Octet : 1	2	Variable
Status	NwkAddrOfInterest	Node Descriptor

4.6.2 Field Description

Name	Range	Description
Status	UINT8	The status of the Node_Desc_req command: SUCCESS DEVICE_NOT_FOUND INV_REQUESTTYPE NO_DESCRIPTOR
NwkAddrOfInterest	UINT16	NWKAddrOfInterest field of Node_Desc_req.
NodeDescriptor	Descriptor Size	The Node descriptor of a remote device when status is 'SUCCESS'. See the details in Table 4.

Table 4 – Node Descriptors

Name	Range	Description
Avail_Type	UINT8	bit [7:5]: Reserved bit [4]: 1=There is user descriptor. bit [3]: 1= There is complex descriptor. bit [2:0]: Type. 0=ZC, 1=ZR, 2=ZED
Freq_Aps	UINT8	bit [7]: Reserved bit [6]: 1=Support 2.4GHz bandwidth. bit [5]: 1= Support 900MHz bandwidth. bit [4]: 1= Support 800MHz bandwidth. bit [3]: Reserved bit [2:0]: APS Flag. Always 0 in ZigBee2006.
MacCapability	UINT8	bit [7]: 1=Can assign the address to the child device. bit [6]: 1=Use security bit [5:4]: Reserved bit [3]: RxOnWhenIdle. 1=In idle status, receiver is turned on. bit [2]: PowerSource. 1=Mains-Power bit [1]: DeviceType. 1=FFD, 0=RFD. bit [0]: 1=Support Alternative PAN Coordinator.
Manufacture	UINT16	Manufacturer code.

Name	Range	Description
MaxBufSize	0x00 ~ 0x7F	Maximum length of ASDU to send by the packet.
MaxTransSize	0x0000 ~ 0xFFFF	Maximum length of data to send by one transaction.
ServerMask	UINT16	bit [15:6]: Reserved bit [5]: 1=Can operate as Backup Discovery Cache. bit [4]: 1= Can operate as Primary Discovery Cache. bit [3]: 1= Can operate as Backup Binding Table Cache. bit [2]: 1= Can operate as Primary Binding Table Cache. bit [1]: 1= Can operate as Backup Trust Center. bit [0]: 1= Can operate as Primary Trust Center.

4.7 Power_Desc_req (ClusterID = 0x0003)

The command used when local device inquires about the power descriptor of a remote device. Destination address of this unicast command is either the remote device or a cache device which contains the discovery information for the remote device.

Table 5

Type Class	Data Type ID	Type	Length(Octet)	Invalid
NULL	0x00	No Data	0	
General Data	0x08	8-bit Data	1	
	0x09	16-bit Data	2	
	0x0A	24-bit Data	3	
	0x0B	32-bit Data	4	
Logical	0x10	Boolean	1	0xFF
Bitmap	0x18	8-bit bitmap	1	
	0x19	16-bit bitmap	2	
	0x1A	24-bit bitmap	3	
	0x1B	32-bit bitmap	4	
Unsigned Integer	0x20	Unsigned 8-bit integer	1	0xFF
	0x21	Unsigned 16-bit integer	2	0xFFFF
	0x22	Unsigned 24-bit integer	3	0xFFFFFF
	0x23	Unsigned 32-bit integer	4	0xFFFFFFFF
Signed Integer	0x28	Signed 8-bit integer	1	0x80
	0x29	Signed 16-bit integer	2	0x8000
	0x2A	Signed 24-bit integer	3	0x800000
	0x2B	Signed 32-bit integer	4	0x80000000
Enumeration	0x30	8-bit enumeration	1	0xFF
	0x31	16-bit enumeration	2	0xFFFF
Floating point	0x38	Semi-precision	2	Not a Number
	0x39	Single precision	4	Not a Number
	0x3A	Double precision	8	Not a Number
String	0x41	Octet String	Defined in first octet	0xFF in first octet
	0x42	Character String	Defined in first octet	0xFF in first octet
Time	0xE0	Time of day	4	0xFFFFFFFF
	0xE1	Date	4	0xFFFFFFFF
Identifier	0xE8	Cluster ID	2	0xFFFF
	0xE9	Attribute ID	2	0xFFFF
	0xEA	BACnet OID	4	0xFFFFFFFF
Miscellaneous	0xF0	IEEE Address	8	0xFFFFFFFF
Unknown	0xFF	Unknown	0	

4.7.1 Transaction Data Format

Transaction Data
Octet : 2
NwkAddrOfInterest

4.7.2 Field Description

Name	Range	Description
NwkAddrOfInterest	UINT16	16-bit NWK Address of remote device

4.8 Power_Desc_rsp (ClusterID = 0x8003)

Remote device, which receives Power_Desc_req, or cache device sends Power_Desc_rsp based on power descriptor information.

4.8.1 Transaction Data Format

Transaction Data		
Octet : 1	2	variable
Status	NwkAddrOfInterest	Power Descriptor

4.8.2 Field Description

Name	Range	Description
Status	UINT8	SUCCESS DEVICE_NOT_FOUND INV_REQUESTTYPE NO_DESCRIPTOR
NwkAddrOfInterest	UINT16	NWKAddrOfInterest field of Power_Desc_req
Power Descriptor	Descriptor Size	It exists when status is 'SUCCESS'. The Power descriptor of a remote device. See the details in Table 6.

Table 6 – Power Descriptor

Name	Range	Description
AvailaPwrSrc_CurrPwrMode	UINT8	bit [7:4]: Available Power Source bit [7]: Reserved. bit [6]: Support Disposable Battery . bit [5]: Support Rechargeable Battery . bit [4]: Support Mains-Power. bit [3:0]: Current Power Mode. 0=Receiver is turned on by RxOnWhenIdle of Node Descriptor. 1=Receiver is turned on periodically. 2=Receiver is turned on when there is the external input. 3 ~ 15 : Reserved
CurrPwrLevel_CurrPwrSrc	UINT8	bit [7:4]: CurrPwrLevel. It shows current power level. 0=Critical, 1=33%, 2=66%, 3=100% bit [3:0]: CurrPwrSrc. It shows current power source. bit [3]: Reserved bit [2]: Operate as Disposable Battery. bit [1]: Operate as Rechargeable Battery. bit [0]: Operate as Mains-Power.

4.9 Simple_Desc_req (ClusterID = 0x0004)

This is the command used when the local device inquires as to a simple descriptor of a defined endpoint. Destination address of this command is the remote device or the cache device which includes the information of discovery for the remote device and it is sent by unicast method.

4.9.1 Transaction Data Format

Transaction Data	
Octet : 2	1
NwkAddrOfInterest	Endpoint

4.9.2 Field Description

Name	Range	Description
NwkAddrOfInterest	UINT16	16-bit NWK Address of remote device.
Endpoint	UINT8	Endpoint of remote device.

4.10 Simple_Desc_rsp (ClusterID = 0x8004)

Remote device, which receives Simple_Desc_req, or cache device sends Simple_Desc_rsp based on simple descriptor information.

4.10.1 Transaction Data Format

Transaction Data		
Octet : 1	2	variable
Status	NwkAddrOfInterest	Simple Descriptor

4.10.2 Field Description

Name	Range	Description
Status	UINT8	SUCCESS INVALID_EP NOT_ACTIVE DEVICE_NOT_FOUND INV_REQUESTTYPE NO_DESCRIPTOR
NwkAddrOfInterest	UINT16	NWKAddrOfInterest of field Simple_Desc_req
Length	UINT8	The number of byte of SimpleDescriptor
Simple Descriptor	Descriptor Size	It exists when the status is SUCCESS. The Simple descriptor of a remote device. See the details in Table 7.

Table 7 – Simple Descriptor

Name	Range	Description
EndPoint	UINT8	The number of endpoint.
AppProID	UINT16	Protocol ID of an application.
AppDevID	UINT16	Device ID of an application.
DevVersion	UINT8	bit [7:4]: Reserved bit [3:0]: Device version.
NumInClus	UINT8	The number of incoming clusters in an endpoint.
InClusList	UINT16 Array	Incoming Cluster List. The list of incoming clusters limited to the number in NumInClus.
NumOutClus	UINT8	The number of outgoing cluster in an endpoint.
OutClusList	UINT16 Array	Outgoing Cluster List. The list of outgoing clusters limited to the number in NumOutClus.

4.11 Active_EP_req (ClusterID = 0x0005)

This is the command used when the local device inquires as to the list of enabled endpoint on a remote device which has a simple descriptor. Destination address of this command is the remote device or the cache device which includes the information of discovery for the remote device and it is sent by unicast method.

4.11.1 Transaction Data Format

Transaction Data
Octet : 2
NwkAddrOfInterest

4.11.2 Field Description

Name	Range	Description
NwkAddrOfInterest	UINT16	16-bit NWK Address of remote device

4.12 Active_EP_rsp (ClusterID = 0x8005)

Remote device, which receives Active_EP_req, or cache device sends Active_EP_rsp based on the information of enabled endpoint of its own endpoints.

4.12.1 Transaction Data Format

Transaction Data			
Octet : 1	2	1	variable
Status	NwkAddrOfInterest	ActiveEPCount	ActiveEPList

4.12.2 Field Description

Name	Range	Description
Status	UINT8	SUCCESS DEVICE_NOT_FOUND INV_REQUESTTYPE NO_DESCRIPTOR
NwkAddrOfInterest	UINT16	NWKAddrOfInterest field of Active_EP_req.
ActiveEPCount	UINT8	The number of enabled endpoint.
ActiveEPList	EPList Size	It exists when ActiveEPCount field is not 0. It includes the information of enabled endpoint.

4.13 Match_Desc_req (ClusterID = 0x0006)

This is the command used when the local device inquires as to which remote devices support a defined simple descriptor. When this command is to be broadcast, the destination address is 0xFFFD (to broadcast to all devices with RxOnWhenIdle = 1.).

When this command is unicast to a remote device, the destination address is the cache device which either includes the remote device or the information of discovery for the remote device.

4.13.1 Transaction Data Format

Transaction Data					
Octet : 2	2	1	variable	1	variable
NwkAddrOfInterest	ProfileID	NumInCluster	InClusterList	NumOutCluster	OutCluserList

4.13.2 Field Description

Name	**Range**	**Description**
NwkAddrOfInterest	UINT16	16-bit NWK Address of remote device
ProfileID	UINT16	ProfileID which should be match on remote device.
NumInCluster	UINT8	The number of cluster provided in InClusterList field to compare input cluster.
InClusterList	UINT16 Array	The list of input cluster ID to be used for comparison. It does not exist when NumInCluster is 0.
NumOutCluster	UINT8	The number of cluster provided in OutClusterList field to compare output cluster.
OutClusterList	UINT16 Array	The list of output cluster ID to be used for matching. It does not exist when NumOutClusterr is 0.

4.14 Match_Desc_rsp (ClusterID = 0x8006)

Remote device or cache device, which receives Match_Desc_req, compares simple descriptor and cluster list of the Match_Desc_req command. After that, it sends the Match_Desc_rsp based on the result.

When NwkAddrOfInterest of Match_Desc_req is 0xFFFD (broadcast to all devices with RxOnWhenIdle = 1), each remote device compares its own simple descriptor to clusters of the Match_Desc_req. In addition, if a remote device is a coordinator or router, it compares with the child simple descriptor of its own children. NwkAddrOfInterest of Match_Desc_rsp is set as the 16-bit NWK Address of any device having a corresponding simple descriptor to that of the original request command.

If NWKAddrOfInterest field value of Match_Desc_req is not 0xFFFD, the remote device compares its simple descriptor and the cluster of Match_Desc_req. NwkAddrOfInterest of Match_Desc_rsp is set as same as NwkAddrOfInterest of Match_Desc_req.

For comparing clusters, when one of the input clusters or one cluster of the output clusters is matched, it is considered successful, and Match_Desc_rsp is sent.

4.14.1 Transaction Data Format

Transaction Data			
Octet : 1	2	1	variable
Status	NwkAddrOfInterest	MatchLength	MatchList

4.14.2 Field Description

Name	Range	Description
Status	UINT8	SUCCESS DEVICE_NOT_FOUND INV_REQUESTTYPE NO_DESCRIPTOR
NwkAddrOfInterest	UINT16	When destination of Match_Desc_req is broadcast, it is 16-bit NWK Address of the device which is matched the result of comparison for simple descriptor When destination of Match_Desc_req is not broadcast, it Is NwkAddrOfInterest of Match_Desc_req.
MatchLength	UINT8	The number of endpoint for remote device which is matched to the Match_Desc_req request.
MatchList	Endpoint List	The list of endpoint for remote device which is matched to the Match_Desc_req request.

4.15 End_Device_Annce (ClusterID = 0x0013)

This is the command used to inform that end device is joined or rejoined to other ZigBee devices. In other words, when the end device is joined or re-joined to a network, the 64-bit IEEE Address of the end device, the new network address, and the capability of the remote device are sent to the other ZigBee devices in the network.

The destination address of this command is sent to all devices by broadcast.

4.15.1 Transaction Data Format

Transaction Data		
Octet : 2	8	1
NwkAddr	IEEEAddr	Capability

4.15.2 Field Description

Name	Range	Description
NwkAddr	UINT16	16-bit NWK Address of its own
IEEEAddr	UINT8[8]	64-bit IEEE Address of its own
Capability	UINT8	Capability of its own

4.16 System_Server_Discover_req (ClusterID = 0x0015)

This is the command used to find the same server as the local device defined as ServerMask parameter or to search for the position of a defined system server.

When sent by broadcast, the destination address of this command is all devices (0xFFFD) with RxWhenIdle = 1.

4.16.1 Transaction Data Format

Transaction Data
Octet : 2
ServerMask

4.16.2 Field Description

Name	Range	Description
ServerMask	16-bit bitmap	bit [15:6]: Reserved. bit [5]: 1 = Operated as Backup Discovery Cache. bit [4]: 1 = Operated as Primary Discovery Cache. bit [3]: 1 = Operated as Backup Binding Table Cache. bit [2]: 1 = Operated as Primary Binding Table Cache. bit [1]: 1 = Operated as Backup Trust Center. bit [0]: 1 = Operated as Primary Trust Center.

4.17 System_Server_Discover_rsp (ClusterID = 0x8015)

The remote device, which receives Match_Desc_req, sends System_Server_Discover_rsp when there are bits, which are all 1, of ServerMask field of node descriptor and ServerMask field of System_Server_Discover_req command. At this moment, TxOptions.AckRequest field of APS-DATA.req should always be 1.

4.17.1 Transaction Data Format

Transaction Data	
Octet : 1	2
Status	ServerMask

4.17.2 Field Description

Name	Range	Description
Status	UINT8	SUCCESS
ServerMask	UINT16	It is the result to operate ServerMask of System_Server _Discovery_req and ServerMask field of node descriptor for remote device by 'AND' per bit bit [15:6]: Reserved bit [5]: 1 = Operate as Backup Discovery Cache bit [4]: 1 = Operate as Primary Discovery Cache bit [3]: 1 = Operate as Backup Binding Table Cache bit [2]: 1 = Operate as Primary Binding Table Cache bit [1]: 1 = Operate as Backup Trust Center bit [0]: 1 = Operate as Primary Trust Center.

4.18 End_Dev_Bind_req (ClusterID = 0x0020)

Local device sends End_Device_Bind_req to generate end device binding with other remote device.

4.18.1 Transaction Data Format

Transaction Data							
Octet : 2	8	1	2	1	Variable	1	Variable
Binding Target	SrcIEEE Address	Src EP	Profile ID	Num InClusters	InCluster List	Num OutClusters	OutCluster List

4.18.2 Field Description

Name	Range	Description
Binding Target	UINT16	The address of target which is binding. It can be a local device or primary cache device.
SrcIEEE Address	UINT8[8]	IEEE address of the device, which requests, itself.
Src Endpoint	UINT8	Endpoint of the device, which requests, itself.
Profile ID	UINT16	It is profile ID to match between two End_Device_Bind_req, which ZigBee coordinator receives, within timeout value defined on ZigBee coordinator
Num InClusters	UINT8	The number of input cluster provided for end device binding in InClusterList.
InCluster List	UINT8 Array	The list of Input cluster ID used in matching. InClusterList is matched with output cluster of remote device on ZigBee coordinator.
Num OutClusters	UINT8	The number of output cluster provided for end device binding in OutClusterList.
OutCluster List	UINT8 Array	The list of output clustered used in matching. OutCluster List is matched with input cluster of the remote device on ZigBee coordinator.

4.19 End_Dev_Bind_rsp (ClusterID = 0x8020)

ZigBee coordinator sends End_Dev_Bind_rsp including status after processing 'End Device Bind' by receiving End_Dev_Bind_req from two devices to each device.

ZigBee coordinator, which supports end device binding, receives End_Dev_Bind_req from the first device and then should receive End_Dev_Bind_req from the second device within the defined time. After reception, when they are matched after comparing two parameters of request, 'End Device Binding' is formed.

4.19.1 Transaction Data Format

Transaction Data
Octet : 1
Status

4.19.2 Field Description

Name	Range	Description
Status	UINT8	SUCCESS NOT_SUPPORTED INVALID_EP TIMEOUT NO_MATCH

4.20 Bind_req (ClusterID = 0x0021)

This is the command used when a local device generates a binding table entry. The destination of this unicast command is the primary binding table cache or the device with the SrcAddress of the Binding Table Entry.

4.20.1 Transaction Data Format

Transaction Data					
Octets: 8	1	2	1	2/8	0/1
SrcAddress	SrcEndp	ClusterID	DstAddrMode	DstAddress	DstEndp

4.20.2 Field Description

Name	Range	Description
SrcAddress	UINT8[8]	IEEE Address of source device on binding entry.
SrcEndpoint	UINT8	Endpoint of source device on binding entry.
ClusterID	UINT16	The value of cluster ID of source device to be bound with destination.
DstAddrMode	UINT8	Address mode for the destination address. 0x00 = Reserved 0x01 = DstAddress means 16-bit group address and there is no DstEndpoint field. 0x02 = Reserved 0x03=DstAddress means 64-bit extended address and DstEndpoint field is included. 0x04 – 0xff = Reserved
DstAddress	UINT8[8]	Address of destination device on binding entry.
DstEndpoint	UINT8	When DstAddrMode is 0x03, it is endpoint of destination device on binding entry.

4.21 Bind_rsp (ClusterID = 0x8021)

Remote device, which receives Bind_req, adds binding table entry by APSME-BIND.req and sends the result to Bind_rsp.

Note: For APSME-BIND.req, please refer to the ***"ZIC2410 User Guide Application Support Layer API Reference"*** (CEL Doc #0005-05-08-07-001) *[filename: zic07_aps_api.pdf]*.

4.21.1 Transaction Data Format

Transaction Data
Octet : 1
Status

4.21.2 Field Description

Name	Range	Description
Status	UINT8	SUCCESS NOT_SUPPORTED INVALID_EP TABLE_FULL

4.22 Unbind_req(ClusterID = 0x0022)

This is the command used when a local device deletes a binding table entry. The destination of this unicast command is the primary binding table cache or the device which the SrcAddress of Binding Table Entry indicates.

4.22.1 Transaction Data Format

Transaction Data					
Octets: 8	1	2	1	2/8	0/1
SrcAddress	SrcEndp	ClusterID	DstAddrMode	DstAddress	DstEndp

4.22.2 Field Description

Name	Range	Description
SrcAddress	UINT8[8]	IEEE Address of source device on binding entry.
SrcEndpoint	UINT8	Endpoint of source device on binding entry.
ClusterID	UINT16	The value of cluster ID of source device to be bound with destination.
DstAddrMode	UINT8	Address mode for the destination address. 0x00 = Reserved 0x01 = DstAddress means 16-bit group address and there is no DstEndpoint field. 0x02 = Reserved 0x03=DstAddress means 64-bit extended address and DstEndpoint field is included. 0x04 – 0xff = Reserved
DstAddress	UINT8[8]	Address of destination device on binding entry.
DstEndpoint	UINT8	When DstAddrMode is 0x03, it is endpoint of destination device on binding entry.

4.23 Unbind_rsp(ClusterID = 0x8022)

Remote device, which receives Unbind_req, deletes binding table entry by APSME-UNBIND.req and sends the result to Unbind_rsp.

Note: For APSME-UNBIND.req, please refer to the ***"ZIC2410 User Guide Application Support Layer API Reference"*** (CEL Doc #0005-05-08-07-001) *[filename: zic07_aps_api.pdf]*.

4.23.1 Transaction Data Format

Transaction Data
Octet : 1
Status

4.23.2 Field Description

Name	Range	Description
Status	UINT8	SUCCESS/ NOT_SUPPORTED / INVALID_EP / NO_ENTRY

5 REVISION HISTORY

Revision	**Date**	**Description**
A	3Sep08	Released
B	13Jan09	Updated Figure 1;updated Table in Section 1.1; added filename to ‘Note:’ in Sections 4.21and 4.23; reworded throughout.