## Brief

This document is the user manual for telink zigbee gateway controller software, and note that only the generic step is shown here, please refer to “AN\_17110600-E2\_Telink Zigbee 3.0 SDK Guide” for more information.

## System Architecture

Serial port

Zigbee Gateway

Computer B

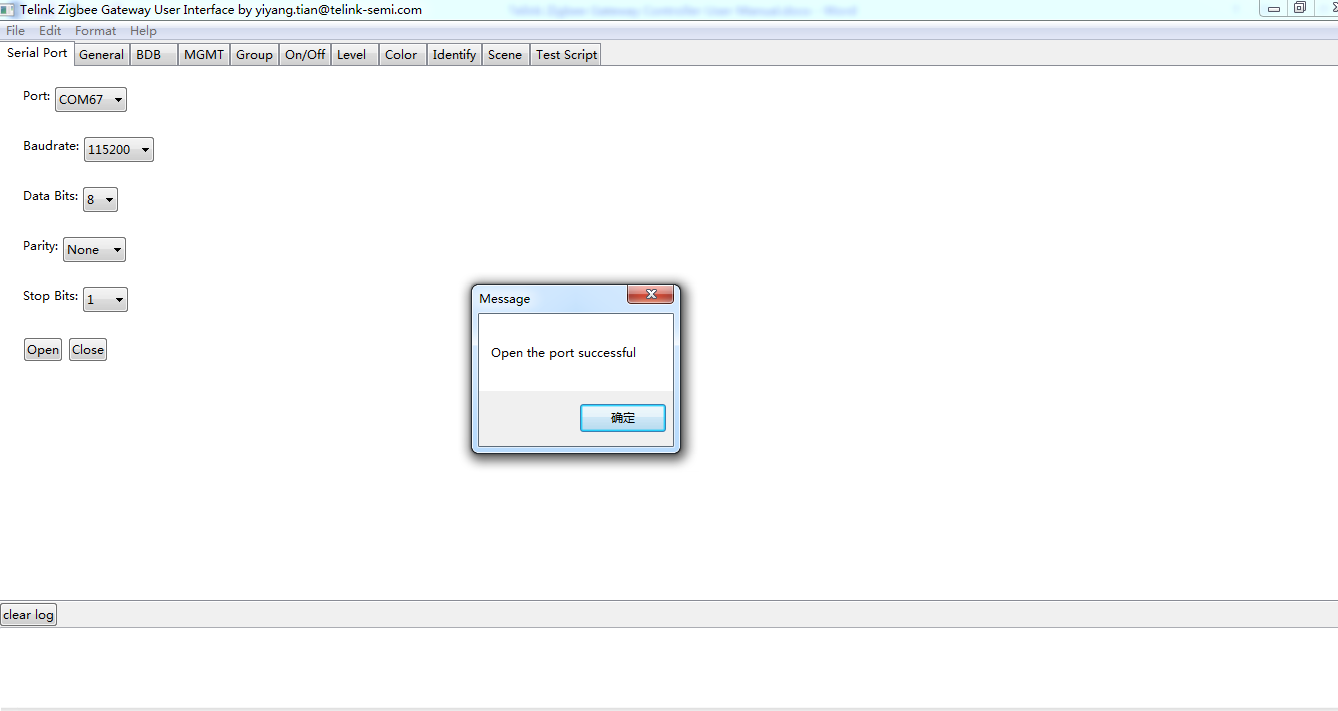
Computer

Telink Zigbee Gateway Controller Software

## Step-by-Step Guide

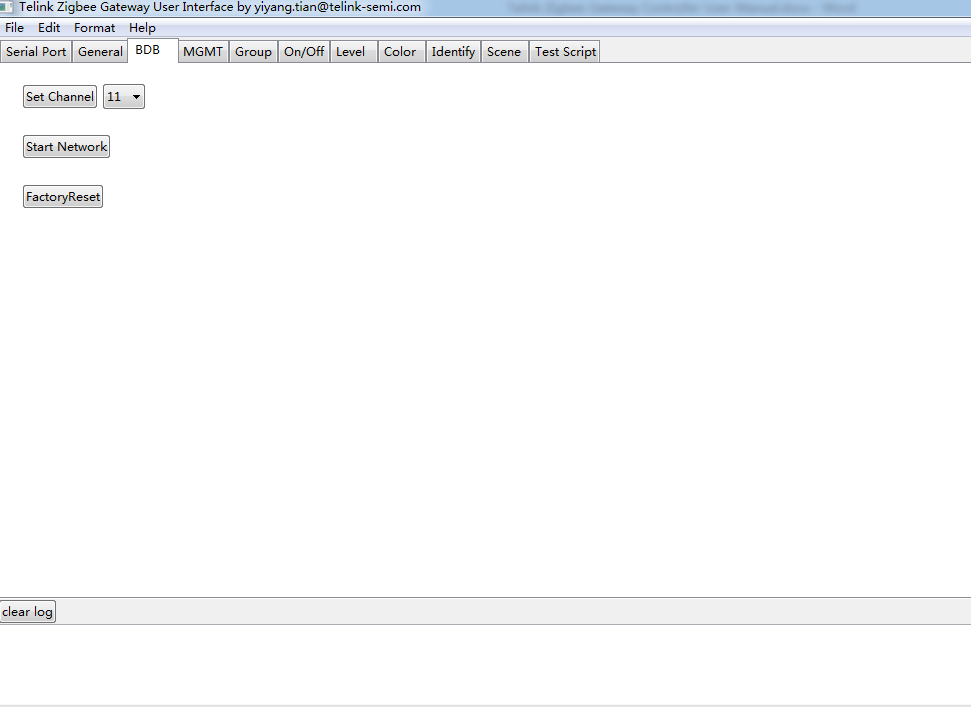
Step 1: Open the port

1. Select the port number, and other options can use the default value.
2. Click on the “Open” button and open the port.



Step 2: Set the channel

1. Switch to the BDB page
2. Select a channel you want to use and click the “Set Channel” button.

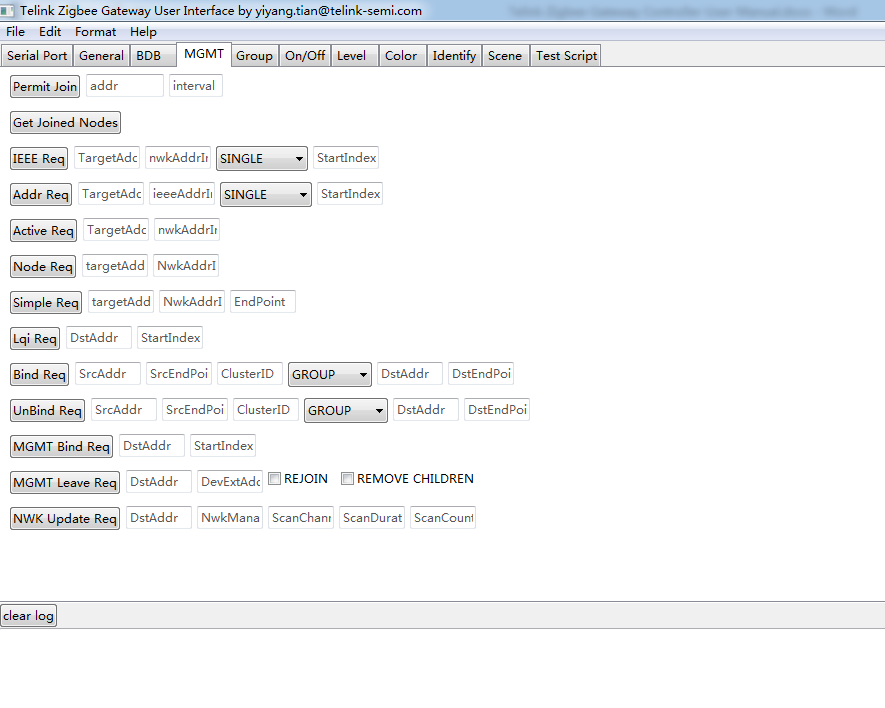


Step 3: Start the network

1. Click the “Start Network” button to start the network

Step 4: Set the permit join interval

1. Switch to the MGMT page
2. Enter the address and interval
3. Click the “Permit Join” button



Step 5: Power on the Zigbee device to join the network before the interval is expired,

and now you can use other commands to control the device.

## Zigbee SDK HCI

### 4.1 HCI command Format

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Start | Command identifier | Length | Check Sum | Payload | End |
| 1Bytes | 2Bytes | 2 Bytes | 1 Byte | various | 1Byte |

*Figure 1: Message Forma*t

Field definitions are as follows:

|  |  |
| --- | --- |
| Field | Descipription |
| Start | Start flag ,shall be 0x55 |
| Message Type | Message type, big-endian |
| Length | Payload length, big-endian |
| Checksum | The checksum |
| payload | payload |
| End | End flag, SHALL be 0xAA |
|  |  |
|  |  |

Note: big-endian is for all the field of the HCI commands

### 4.2 Communication flow

Request

Acknowledge

Response

Acknowledge

Host

ZigbHostee unit

### 4.2.1 Acknowledge format

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_ACKNOWLEDGE* | 0x8000 |

|  |  |  |
| --- | --- | --- |
| cmdId | status | resv |
| *2Bytes* | *1Byte* | *1Byte* |

***cmdId: shall be the request/response command identifier***

***status: 0=success, 1=wrong parameter, 2=un-support command, 3=busy,4=no memory***

***packet format*: 55 80 00 00 04 00** cmdId status resv **AA**

# 2. Network management command

## 2.1 Command for BDB

### 2.1.1 command Type(Host)

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_BDB\_COMMISSION\_FORMATION*  *ZBHCI\_CMD\_BDB\_COMMISSION\_STEER*  *ZBHCI\_CMD\_BDB\_COMMISSION\_TOUCHLINK*  *ZBHCI\_CMD\_BDB\_COMMISSION\_FINDBIND*  *ZBHCI\_CMD\_BDB\_FACTORY\_RESET*  *ZBHCI\_CMD\_BDB\_PRE\_INSTALL\_CODE*  *ZBHCI\_CMD\_BDB\_CHANNEL\_SET* | 0x0001  0x0002  0x0003  0x0004  0x0005  0x0006  0x0007 |

### 2.1.2 BDB commissioning command frame format

***2.1.2.1 ZBHCI\_CMD\_BDB\_COMMISION\_FORMATION***

*payload*: empty

***packet format*: 55 00 01 00 00 00 AA**

**2.1.2.2  *ZBHCI\_CMD\_BDB\_COMMISION\_STEER***

*payload*: empty

***packet format*: 55 00 02 00 00 00 AA**

**2.1.2.3  *ZBHCI\_CMD\_BDB\_COMMISION\_TOUCHLINK***

|  |
| --- |
| role |
| *1Byte* |

*role*: 1 =touch link initiator, 2=touch link target

***packet format*: 55 00 03 00 01 00** role **AA**

**2.1.2.4 *ZBHCI\_CMD\_BDB\_COMMISION\_FINDBIND***

|  |
| --- |
| role |
| *1Byte* |

*role*: 1 = finding&binding initiator, 2 = finding&binding target

***packet format*: 55 00 04 00 01 00** role **AA**

**2.1.2.5 *ZBHCI\_CMD\_BDB\_FACTORY\_NEW***

payload: empty

***packet format*: 55 00 05 00 00 00 AA**

**2.1.2.6 *ZBHCI\_CMD\_BDB\_PRE\_INSTALL\_CODE***

|  |  |
| --- | --- |
| devAddr | UniqueLinkKey |
| *8Byte* | *16Byte* |

*devAddr*: the ieee address of the device

*Unique link key*: the link key for the device which ieee address is devAddr

***packet format*: 55 00 06 00 18 00** devAddr[0-7] UniqueLinkKey[0-15] **AA**

**2.1.2.7 *ZBHCI\_CMD\_BDB\_CHANNEL\_SET***

|  |
| --- |
| channelIdx |
| *1Byte* |

*channelIdx*: 11-26

***packet format*: 55 00 07 00 01 00** channelIdx **AA**

## 2.2 Command for other network manage

### 2.2.1 Command type(Host)

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_DISCOVERY\_NWK\_ADDR\_REQ*  *ZBHCI\_CMD\_DISCOVERY\_IEEE\_ADDR\_REQ*  *ZBHCI\_CMD\_DISCOVERY\_NODE\_DESC\_REQ*  *ZBHCI\_CMD\_DISCOVERY\_SIMPLE\_DESC\_REQ*  *ZBHCI\_CMD\_DISCOVERY\_MATCH\_DESC\_REQ*  *ZBHCI\_CMD\_DISCOVERY\_ACTIVE\_EP\_REQ*  *ZBHCI\_CMD\_DISCOVERY\_LEAVE\_REQ*  *ZBHCI\_CMD\_BIND\_REQ*  *ZBHCI\_CMD\_UNBIND\_REQ*  *ZBHCI\_CMD\_MGMT\_LQI\_REQ*  *ZBHCI\_CMD\_MGMT\_BIND\_REQ*  *ZBHCI\_CMD\_MGMT\_LEAVE\_REQ*  *ZBHCI\_CMD\_MGMT\_DIRECT\_JOIN\_REQ*  *ZBHCI\_CMD\_MGMT\_PERMIT\_JOIN\_REQ*  *ZBHCI\_CMD\_MGMT\_NWK\_UPDATE\_REQ*  *ZBHCI\_CMD\_NODES\_JOINED\_GET\_REQ*  **ZBHCI\_CMD\_NODES\_TOGLE\_TEST\_REQ** | 0x0010  0x0011  0x0012  0x0013  0x0014  0x0015  0x0016  0x0020  0x0021  0x0030  0x0031  0x0032  0x0033  0x0034  0x0035  0x0040  0x0041 |

2.2.2 Command frame format

***2.2.2.1 ZBHCI\_CMD\_DISCOVERY\_NWK\_ADDR\_REQ***

|  |  |  |  |
| --- | --- | --- | --- |
| DstAddr | IEEE Address | RequestType | StartIndex |
| *2Bytes* | *8Bytes* | *1Byte* | *1Bytes* |

*DstAddr:* the address of this command will be send to, shall be unicast or broadcast to all devices for which macRxOnWhenIdle = TRUE.

*IEEE Address* : The IEEE address to be matched by the Remote Device

*RequestType* : Request type for this command:

0x00 – Single device response

0x01 – Extended response

0x02-0xFF – reserved

*StartIndex:* If the Request type for this command is extended response, the StartIndex provides the starting index for the requested elements of the associated devices list.

***packet format*: 55 00 10 00 0c 00** DstAddr [1-0] IEEEAddress[7-0] requestType StartIndex **AA**

***2.2.2.2 ZBHCI\_CMD\_DISCOVERY\_IEEE\_ADDR\_REQ***

|  |  |  |  |
| --- | --- | --- | --- |
| DstAddr | NWKAddrOfInterest | RequesType | StartIndex |
| *2Bytes* | *2Bytes* | *1Byte* | *1Bytes* |

DstAddr*:* the address of this command will be send to.

*NWKAddrOfInterest* : NWK address that is used for IEEE address mapping.

*RequestType* : Request type for this command:

0x00 – Single device response

0x01 – Extended response

0x02-0xFF – reserved

*StartIndex:* If the Request type for this command is extended response, the StartIndex provides the starting index for the requested elements of the associated devices list

***packet format*: 55 00 11 00 06 00** DstAddr[1-0] NWKAddrOfInterest [1-0] requestType StartIndex **AA**

***2.2.2.3 ZBHCI\_CMD\_DISCOVERY\_NODE\_DESC\_REQ***

.to obtain the node descriptor of the device

|  |  |
| --- | --- |
| DstAddr | NWKAddrOfInterest |
| *2Bytes* | *2Bytes* |

DstAddr*:* the address of this command will be send to*.*

*NWKAddrOfInterest*: the 16-bits address of the target

***packet format*: 55 00 12 00 04 00** DstAddr[1-0] NWKAddrOfInterest[1-0] **AA**

***2.2.2.4 ZBHCI\_CMD\_DISCOVERY\_SIMPLE\_DESC\_REQ***

|  |  |  |
| --- | --- | --- |
| DstAddr | NWKAddrOfInterest | EndPoint |
| *2Bytes* | *2Bytes* | *1Bytes* |

DstAddr*:* the address of this command will be send to*.*

*NWKAddrOfInterest*: NWK address for the request

endPoint: The endpoint on the destination

***packet format*: 55 00 13 00 05 00** DstAddr[1-0] NWKAddrOfInterest [1-0] EndPoint **AA**

***2.2.2.5 ZBHCI\_CMD\_DISCOVERY\_MATCH\_DESC\_REQ***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| DstAddr | NWKAddrOfInterest | ProfileID | NumInClusters | NumOutClusters | ClusterList |
| *2Bytes* | *2Bytes* | *2Bytes* | *1Byte* | *1Bytes* | *nByte* |

DstAddr*:* the address of this command will be send to*.*

*NWKAddrOfInterest* : NWK address for the request.

*ProfileID:* Profile ID to be matched at the destination.

*NumInClusters:* The number of Input Clusters provided for matching within the InClusterList.

*NumOutClusters:* The number of Output Clusters provided for matching within OutClusterList.

*ClusterList: InClusterList* + *OutClusterList*

List of Input ClusterIDs to be used for matching; the InClusterList is the desired list

to be matched by the Remote Device (the elements of the In-ClusterList are the

supported output clusters of the Local Device).

List of Output ClusterIDs to be used for match-ing; the OutClusterList is the desired

list to be matched by the Remote Device (the elements of the OutClusterList are the

supported input clus-ters of the Local Device).

***packet format*: 55 00 14 lenH lenL 00** DstAddr[1-0] NWKAddrOfInterest [0-1] *ProfileID NumInClusters NumOutClusters InClusterList OutClusterList* ***AA***

***2.2.2.6 ZBHCI\_CMD\_DISCOVERY\_ ACTIVE\_ENDPOINT \_REQ***

|  |  |
| --- | --- |
| DstAddr | NWKAddrOfInterest |
| *2Bytes* | *2Bytes* |

DstAddr*:* the address of this command will be send to*.*

*NWKAddrOfInterest* : NWK address for the request.

***packet format*: 55 00 15 00 04 00** DstAddr[1-0] NWKAddrOfInterest[1-0]*AA*

***2.2.2.7 ZBHCI\_CMD\_DISCOVERY\_ LEAVE \_REQ***

|  |  |  |
| --- | --- | --- |
| DeviceAddress | Rejoin | RemoveChildren |
| *8Bytes* | *1Bytes* | *1Bytes* |

*SrcAddress:* The IEEE address of the entry to be removed or NULL if the device removes itself.

*Rejoin:* TRUE if the device being asked to leave from the current parent is requested to rejoin the network. Otherwise, the parameter has a value of FALSE.

*RemoveChildren:* TRUE if the device being asked to leave the network is also being asked to remove its child device, if any. Otherwise, it has a value of FALSE.

**packet format for Leave: 55 00 20 lenH lenL 00** DeviceAddress[7-0] Rejoin RemoveChildren **AA**

***2.2.2.8 ZBHCI\_CMD\_BIND \_REQ/ ZBHCI\_CMD\_UNBIND \_REQ***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SrcAddress | SrcEndp | ClusterID | DstAddressMode | DstAddress | DstEndp |
| *8Bytes* | *1Byte* | *2Bytes* | *1Byte* | *2/8 Bytes* | *0/1Bytes* |

*SrcAddress:* The IEEE address for the source.

*SrcEndp:*  The source endpoint for the binding entry

*ClusterID:* The identifier of the cluster on the source device that is bound to the destination.

*DstAddressMode:* The addressing mode for the destination address used in this

Command.

0x00 = reserved

0x01 = 16-bit group address for DstAddress and DstEndp not present

0x02 = reserved

0x03 = 64-bit extended address for DstAddress and DstEndp present

0x04 – 0xff = reserved

*DstAddress :*  The destination address for the binding entry.

*DstEndp :*  shall be present only if the DstAddrMode field has a value of 0x03 and, if present,

shall be the destination endpoint for the binding entry

**packet format for Bind: 55 00 20 lenH lenL 00** SrcAddress[7-0] SrcEndp ClusterID

DstAddressMode DstAddress[7/1-0] DstEndp **AA**

**packet format for unBind: 55 00 21 lenH lenL 00** SrcAddress[0-7] SrcEndp ClusterID

DstAddressMode DstAddress DstEndp **AA**

**2.2.2.9 ZBHCI\_CMD\_MGMT\_LQI\_REQ**

|  |  |
| --- | --- |
| dstAddr | StartIndex |
| 2Bytes | 1Byte |

*dstAddr:* the address this command will be send to, shall be unicast.

*StartIndex*: Starting Index for the requested elements of the Neighbor Table.

**packet format: 55 00 30 00 03 00** dstAddr[1-0] StartIndex **AA**

**2.2.2.10 ZBHCI\_CMD\_MGMT\_BIND\_REQ**

|  |  |
| --- | --- |
| dstAddr | StartIndex |
| 2Bytes | 1Byte |

*dstAddr:* the address this command will be send to.

*StartIndex:* Starting Index for the requested elements of the Neighbor Table.

**packet format: 55 00 31 00 03 00** dstAddr[1-0] StartIndex **AA**

**2.2.2.11 ZBHCI\_CMD\_MGMT\_LEAVE\_REQ**

|  |  |  |  |
| --- | --- | --- | --- |
| dstAddr | DeviceAddress | Rejoin | RemoveChildren |
| 2Bytes | 8Bytes | 1Byte | 1Byte |

*dstAddr:* the address this command will be send to.

*Device Address*: IEEE address.

*Rejoin*: Rejoin=1 if the device being asked to leave from the current parent andrequested to rejoin

the network. Otherwise, it has a value of 0

*RemoveChildren:* RemoveChildren =1 if the device being asked to leave the network is also

being asked to remove, its child devices, if any. Otherwise, it has a value of 0.

**packet format: 55 00 32 00 0c 00** dstAddr[1-0] DeviceAddress[7-0]RejoinRemoveChildren AA.

**2.2.2.12 ZBHCI\_CMD\_MGMT\_DIRECT\_JOIN\_REQ**

|  |  |
| --- | --- |
| DeviceAddress | CapabilityInformation |
| 8Bytes | 1Byte |

*DeviceAddress:*  ieee address.

*Capability Information*: The operating capabilities of the device being directly joined.

**packet format: 55 00 33 00 09 00** DeviceAddress[0-7] CapabilityInformation **AA.**

**2.2.2.13 ZBHCI\_CMD\_MGMT\_PERMIT\_JOIN\_REQ**

|  |  |  |
| --- | --- | --- |
| dstAddr | PermitDuration | TC\_Significance |
| 2Bytes | 1Byte | 1Byte |

*dstAddr:* the address this command will be send to.

*PermitDuration:* PermitDuration param-eter within NLME-PERMIT-JOINING.request.

*TC\_Significance:* This field shall always have a value of 1, indicating a request to change the

Trust Center policy. If a frame is received with a value of 0, it shall be treated as having a value of 1.

**packet format: 55 00 34 00 04 00** dstAddr [1-0] PermitDuration TC\_Significance **AA**

**2.2.2.14 ZBHCI\_CMD\_MGMT\_NWK\_UPDATE\_REQ**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| dstAddr | nwkManagerAddr | ScanChannels | ScanDuration | ScanCount/ nwkUpdateId |
| 2Bytes | 2Bytes | 4Bytes | 1Bytes | 1Byte |

*dstAddr:* the address this command will be send to.

*ScanChannels:* channel bit mask, the 32-bit field structure.

*ScanDuration:*  A value used to calculate the length of time to spend scanning each channel.

0x00-0x05 or 0xfe or 0xff.

*ScanCount:*  This field represents the number of energy scans to be conducted and reported.

It shall be present only if the ScanDuration is within the range of 0x00 to 0x05.

*nwkUpdateId:* The value of the nwkUpdated contained in this request. This value is set by the

Network Channel Manager prior to sending the message.

This field shall only be present of the ScanDuration is 0xfe or 0xff. If the ScanDuration

is 0xff, then the value in the nwkUpdateID shall be ignored.

*nwkManagerAddr:* This field shall be present only if the ScanDuration is set to 0xff, and, where

present, indicates the NWK address for the device with the Network Manager bit

set in its Node De-scriptor.

**packet format: 55 00 35 00 0a 00** stAddr[1-0] nwkManagerAddr[1-0] ScanChannels

ScanDuration ScanCount/nwkUpdateId **AA**

**2.2.2.15 ZBHCI\_CMD\_NODES\_JOINED\_GET\_REQ**

|  |
| --- |
| startIndex |
| 1Bytes |

startIndex*:* Starting Index for the requested elements of the joined nodes list.

**packet format: 55 00 40 00 02 00** startIndex **AA**

**2.2.2.16 ZBHCI\_CMD\_NODES\_TOGLE\_TEST\_REQ**

**To test the performance of the nodes joined the network**

|  |  |
| --- | --- |
| On/Off | Data send interval(0.1s) |
| 1Bytes | 1Byte |

*On/Off: send On/Off command to a node*

Data send interval*: the interval of the data transmission*

2.2.3 Command Type(Slave)

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_DISCOVERY\_NWK\_ADDR\_RSP*  *ZBHCI\_CMD\_DISCOVERY\_IEEE\_ADDR\_RSP*  *ZBHCI\_CMD\_DISCOVERY\_NODE\_DESC\_RSP*  *ZBHCI\_CMD\_DISCOVERY\_SIMPLE\_DESC\_RSP*  *ZBHCI\_CMD\_DISCOVERY\_MATCH\_DESC\_RSP*  *ZBHCI\_CMD\_DISCOVERY\_ACTIVE\_EP\_RSP*  *ZBHCI\_CMD\_BIND\_RSP*  *ZBHCI\_CMD\_UNBIND\_RSP*  *ZBHCI\_CMD\_MGMT\_LQI\_RSP*  *ZBHCI\_CMD\_MGMT\_BIND\_RSP*  *ZBHCI\_CMD\_MGMT\_LEAVE\_RSP*  *ZBHCI\_CMD\_MGMT\_DIRECT\_JOIN\_RSP*  *ZBHCI\_CMD\_MGMT\_PERMIT\_JOIN\_RSP*  *ZBHCI\_CMD\_MGMT\_NWK\_UPDATE\_RSP*  *ZBHCI\_CMD\_NODES\_JOINED\_GET\_RSP* | 0x8010  0x8011  0x8012  0x8013  0x8014  0x8015  0x8020  0x8021  0x8030  0x8031  0x8032  0x8033  0x8034  0x8035  0x8040 |

***2.2.3.1 ZBHCI\_CMD\_DISCOVERY\_NWK\_ADDR\_RSP***

***2.2.3.2 ZBHCI\_CMD\_DISCOVERY\_IEEE\_ADDR\_RSP***

*SeqNo:*  ZDP transaction sequence number.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| SeqNo | status | IEEEAddr  RemoteDev | NWKAddr  RemoteDev | Num  AssocDev | StartIndex | NWKAddr  AssocDevList |
| *1Byte* | *1Byte* | *8Bytes* | *2Bytes* | *0/1Byte* | *0/1byte* | *0/nBytes* |

*status:* The status of the NWK\_addr\_req command.

*IEEEAddrRemoteDev:* 64-bit address for the Remote Device.

*NWKAddrRemoteDev :* 16-bit address for the Remote Device.

*NumAssocDev:* Count of the number of 16-bit short addresses to follow.

*StartIndex:* starting index into the list of associated devices for this report.

*NWKAddrAssocDevList:* the list of associated devices.

***packet format of ZBHCI\_CMD\_DISCOVERY\_NWK\_ADDR\_RSP:***

**55 80 10 lenH lenL 00** seqNostatus ieeAddr[7-0] nwkAddr[1-0] {num StartIndex list} **AA**

***packet format of ZBHCI\_CMD\_DISCOVERY\_IEEE\_ADDR\_RSP:***

**55 80 11 lenH lenL 00** seqNostatus ieeAddr[7-0] nwkAddr[1-0] {num StartIndex list} **AA**

***2.2.3.3 ZBHCI\_CMD\_DISCOVERY\_NODE\_DESC\_RSP***

|  |  |  |  |
| --- | --- | --- | --- |
| SeqNo | status | NWKAddrOfInterest | NodeDescription |
| *1Byte* | *1Byte* | *2Bytes* | *0/13Bytes* |

*SeqNo:*  ZDP transaction sequence number.

*status:* The status of the Node\_Desc\_req command.

*NWKAddrOfInterest:* NWK address for the request.

*NodeDescription:* This field shall only be included in the frame if the status field is equal to SUCCESS.

***packet format* : 55 80 12 lenH lenL 00** seqNostatus *NWKAddrOfInterest* [1-0] {NodeDescription} AA

***2.2.3.4 ZBHCI\_CMD\_DISCOVERY\_SIMPLE\_DESC\_RSP***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SeqNo | status | NWKAddrOfInterest | lenght | resv | simpleDescription |
| *1Byte* | *1Bytes* | *2Byte* | *1Byte* | *1Byte* | *nBytes* |

*SeqNo:*  ZDP transaction sequence number.

*status:* The status of the Simple\_Desc\_req command.

*NWKAddrOfInterest:* NWK address for the request.

*Length*: the length of the simple descriptor.

*Resv:*

*simpleDescription:* This field shall only be included in the frame if the status field is equal to SUCCESS.

***packet format* : 55 80 13 lenH lenL 00** seqNostatus *NWKAddrOfInterest* [1-0]

{ simpleDescription } **AA**

***2.2.3.5 ZBHCI\_CMD\_DISCOVERY\_MATCH\_DESC\_RSP***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SeqNo | status | NWKAddrOfInterest | MatchLen | MatchList |
| *1Byte* | *1Bytes* | *2Byte* | *1Byte* | *1Byte* |

*SeqNo:*  ZDP transaction sequence number.

*status:* The status of the Node\_Desc\_req command.

*NWKAddrOfInterest:* NWK address for the request.

*MatchLen:*  The count of endpoints on the Remote Device that match the request criteria.

*MatchList:* List of bytes each of which represents an 8-bit endpoint.

***packet format* : 55 80 14 lenH lenL 00** seqNostatus *NWKAddrOfInterest* [1-0] MatchLen

MatchList **AA**

***2.2.3.6 ZBHCI\_CMD\_DISCOVERY\_ACTIVE\_EP\_RSP***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SeqNo | status | NWKAddrOfInterest | activeEpCount | epList |
| *1Byte* | *1Bytes* | *2Byte* | *1Byte* | *nByte* |

*SeqNo:*  ZDP transaction sequence number.

*status:* The status of the Active\_Ep\_req command.

*NWKAddrOfInterest:* NWK address for the request.

*activeEpCount*: the number of the active Endpoint.

*epList:* the list of the active endpoint.

***packet format* : 55 80 15 lenH lenL 00** seqNostatus *NWKAddrOfInterest* [1-0] activeEpCount

epList **AA**

***2.2.3.7 ZBHCI\_CMD\_BIND\_RSP***

***2.2.3.8 ZBHCI\_CMD\_UNBIND\_RSP***

|  |
| --- |
| Status |
| *1Byte* |

*status:* The status of the Bind/Unbind\_req command.0=success, 0x88=no entry.

***packet format of ZBHCI\_CMD\_BIND\_RSP:* 55 80 20 00 01 00** status **AA.**

***packet format of ZBHCI\_CMD\_UNBIND\_RSP:* 55 80 21 00 01 00** status **AA.**

***2.2.3.9 ZBHCI\_CMD\_DISCOVERY\_MGMT\_LQI\_RSP***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Status | NeighborTable  Entries | StartIndex | NeighborTable  ListCount | NeighborTableList |
| *1Byte* | *1Bytes* | *1Byte* | *1Byte* | *nByte* |

*status:* The status of the Mgmt\_Lqi\_req command.

*NeighborTableEntries:* Total number of Neighbor Table entries within the Remote Device.

*StartIndex:*  Starting index within the Neighbor Table to begin reporting for the NeighborTableList.

*NeighborTableListCount:* Number of Neighbor Table entries included within NeighborTableList.

*NeighborTableList:* A list of descriptors, beginning with the StartIndex element and continuing for

NeighborTableListCount.

***packet format* : 55 80 30 lenH lenL 00** status *NeighborTableEntries* *StartIndex*

*NeighborTableListCount NeighborTableList* **AA**

***2.2.3.10 ZBHCI\_CMD\_DISCOVERY\_MGMT\_BIND\_RSP***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Status | BindingTable  Entries | StartIndex | BindingTable  ListCount | BindingTableList |
| *1Byte* | *1Bytes* | *1Byte* | *1Byte* | *nByte* |

*status:* The status of the Mgmt\_Bind\_req command.

Binding*TableEntries:* Total number of Binding Table entries within the Remote Device.

*StartIndex:*  Starting index within the Binding Table to begin reporting for the BindingTableList.

Binding*TableListCount:* Number of Binding Table entries included within BindingTableList.

Binding*TableList:* A list of descriptors, beginning with the StartIndex element and continuing for

BindingTableList Count.

***packet format* : 55 80 31 lenH lenL 00** status Binding*TableEntries StartIndex*

Binding*TableListCount* Binding*TableList* **AA**

***2.2.3.11*** *ZBHCI\_CMD\_NODES\_JOINED\_GET\_RSP*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Status | totalCnt | startIndex | ListCount | macAddrList |
| *1Byte* | *1Byte* | *1Byte* | *1Byte* | *nBytes* |

status: the status of this response packet

totalCnt: the total count of the joined nodes

startIndex: the start index

listCnt: the count of the MAC address list in the current packet

macAddrList: the MAC address list in the current packet

***packet format* : 55 80 40 lenH lenL 00** totalCnt startIndexListCount macAddrList *t* **AA**

Cluster command

## 3.1 Cluster command frame format

|  |  |
| --- | --- |
| ZCLCmdHdr | payload |
| n-Bytes | n-Bytes |

ZCLCmdHdr format as follow:

|  |  |  |  |
| --- | --- | --- | --- |
| dstAddrMode | dstAddr | srcEp | dstEp |
| 1Byte | 0/2/8Bytes | 1Byte | 1Byte |

*dstAddrMode:* destination address mode

0: without destination address or destination endpoint, for binding

1: with group address

2: with destination network address and destination endpoint

3: with destination ieee address and destination endpoint

*dstAddr:* none, dstAddrMode = 0

group address(2Bytes) if dstAddrMode = 1

destination network address(2Bytes) if dstAddrMode = 2

destination ieee address(8Bytes) if dstAddrMode = 3

*srcEp:*  source endpoint

*dstEp:*  destination endpoint if dstAddrMod = 2 or3.

## General Cluster Command Type

3.2.1 Command Type(Host)







|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_ZCL\_ATTR\_READ*  *ZBHCI\_CMD\_ZCL\_ATTR\_WRITE*  *ZBHCI\_CMD\_ZCL\_CONFIG\_REPORT*  *ZBHCI\_CMD\_ZCL\_READ\_REPORT\_CFG* | 0x0100  0x0101  0x0102  0x0103 |

***3.2.1.1 ZBHCI\_CMD\_ZCL\_ATTR\_READ***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ZCLCmdHdr | direction | clusterID | attrNums | attrList |
| *n-Bytes* | *1Byte* | *2Bytes* | *1Byte* | *N Bytes* |

*direction*: 0 – client to server; 1 – server to client

*clusterID: cluster identifier*

*attrNums:* the number of the attributes to be read

*attrList: the list of the* attributes to be read, as follow:

|  |  |  |
| --- | --- | --- |
| attrID[0] | attrID[1] | attrID[n] |
| 2Bytes | 2Bytes | 2Bytes |

***packet format*: 55 01 00 lenH lenL 0** ZCLCmdHdr direction clusterId attrNumsattrList ***AA***

***3.2.1.2 ZBHCI\_CMD\_ZCL\_ATTR\_WRITE***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ZCLCmdHdr | direction | clusterID | attrNums | attrList |
| *n-Bytes* | *1Byte* | *2 Bytes* | *1Byte* | *N Bytes* |

*direction*: 0 – client to server; 1 – server to client

*clusterID: cluster identifier*

*attrNums:* the number of the attributes to be written

*attrList: the list of the* attributes to be written, as follow:

|  |  |  |
| --- | --- | --- |
| attrList[0] | attrList [1] | attrList [n] |
| *n-Bytes* | *n-Bytes* | *n-Bytes* |

attrList [n]: as follow:

|  |  |  |
| --- | --- | --- |
| attrID | dataType | attrData |
| 2Bytes | 1Byte | *n-Bytes* |

***packet format*: 55 01 01 lenH lenL 0** ZCLCmdHdr direction clusterID attrNumsattrList ***AA***

***3.2.1.3 ZBHCI\_CMD\_ZCL\_CONFIG\_REPORT***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ZCLCmdHdr | direction | clusterID | attrNums | attrList |
| *n-Bytes* | *1Byte* | *2 Bytes* | *1Byte* | *N Bytes* |

*direction*: 0 – client to server; 1 – server to client

*clusterID: cluster identifier*

*attrNums:* the number of the attributes to be report configured

*attrList: the list of the* attributes to be configure, the format as follow:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Direction | AttrID | DataType | MinRep Interval | MaxRep Interval | Reportable Change | Timeout Period |
| *1Byte* | *2Bytes* | *1Byte* | *2Bytes* | *2Bytes* | *N Byte(s)* | *2Bytes* |

If the direction field is set to 0, then the attribute data type field, the minimum reporting interval field, the maximum reporting interval field and the reportable change field are included in the payload, and the timeout period field is omitted. If the value is set to 1, the timeout period field is included in the payload, and the attribute data type field, the minimum reporting interval field, the maximum reporting interval field and the reportable change field are omitted.

For attributes with 'analog' data type, the field has the same data type as the attribute.

***packet format*: 55 01 02 lenH lenL 0** ZCLCmdHdr direction clusterID attrNumsattrList ***AA***

***3.2.1.4 ZBHCI\_CMD\_ZCL\_READ\_REPORT\_CFG***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ZCLCmdHdr | direction | clusterID | attrNums | attrList |
| *n-Bytes* | *1Byte* | *2 Bytes* | *1Byte* | *N Bytes* |

*direction*: 0 – client to server; 1 – server to client

*clusterID: cluster identifier*

*attrNums:* the number of the attributes to be report configured

*attrList: the list of the* attributes to be read, the format as follow:

|  |  |
| --- | --- |
| Direction | AttrID |
| *1 Bytes* | *2Bytes* |

***packet format*: 55 01 03 lenH lenL 0** ZCLCmdHdr direction clusterID attrNumsattrList ***AA***

3.2.2 Command Type(Slave)

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_ZCL\_ATTR\_READ\_RSP*  *ZBHCI\_CMD\_ZCL\_ATTR\_WRITE\_RSP*  *ZBHCI\_CMD\_ZCL\_CONFIG\_REPORT\_RSP*  *ZBHCI\_CMD\_ZCL\_READ\_REPORT\_CFG\_RSP*  *ZBHCI\_CMD\_ZCL\_REPORT\_MSG\_RCV* | 0x8100  0x8101  0x8102  0x8103  0x8104 |

***3.2.2.1 ZBHCI\_CMD\_ZCL\_ATTR\_READ\_RSP***

|  |  |
| --- | --- |
| attrNums | attrList |
| *1Byte* | *N Bytes* |

*attrNums:* the number of the attributes to be read

*attrList: the list of the* attributes to be read

attrID(2Bytes)+status(1Byte)+dataType(1Byte)+data(nByte)

***packet format*: 55 81 00 lenH lenL 0** attrNumsattrList ***AA***

***3.2.2.2 ZBHCI\_CMD\_ZCL\_ATTR\_WRITE\_RSP***

|  |  |
| --- | --- |
| attrNums | attrList |
| *1Byte* | *N Bytes* |

*attrNums:* the number of the attributes to be written

*attrList: the list of the* attributes to be written

.status(1Byte)+attrID(2Bytes)  
In the case of successful writing of all attributes, only a single write attribute status record SHALL be included in the command, with the status field set to SUCCESS and the attribute identifier field mitted.

***packet format*: 55 81 01 lenH lenL 0** attrNumsattrList ***AA***

***3.2.2.3*** ZBHCI\_CMD\_ZCL\_CONFIG\_REPORT\_RSP

|  |  |
| --- | --- |
| attrNums | attrList |
| *1Byte* | *N Bytes* |

*attrNums:* the number of the attributes reports to be configure

*attrList: the list of the* attributes to be configure, the format as follow:

|  |  |  |
| --- | --- | --- |
| Status | Direction | AttrID |
| *1Byte* | *1 Byte* | *2Bytes* |

***packet format*: 55 81 02 lenH lenL 0** attrNumsattrList ***AA***

***3.2.2.4*** ZBHCI\_CMD\_ZCL\_READ\_REPORT\_CFG\_RSP

|  |  |
| --- | --- |
| attrNums | attrList |
| *1Byte* | *N Bytes* |

*attrNums:* the number of the attributes reports configure to be read

*attrList: the list of the* attributes configure to be read, the format as follow:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Status | direction | attrID | DataType | MinRep Interval | MaxRep Interval | Reportable Change | Timeout Period |
| *1Byte* | *1Byte* | *2Bytes* | *1Byte* | *2Bytes* | *2Bytes* | *N Byte(s)* | *2Bytes* |

If direction field value is set to 0x00, then the attribute data type field, the minimum reporting interval field, the maximum reporting interval field and the reportable change field are included in the payload, and the timeout period field is omitted. If this value is set to 0x01, then the timeout period field is included in the payload, and the attribute data type field, the minimum reporting interval field, the maximum reporting interval field and the reportable change field are omitted.

For attributes with 'analog' data type, the field has the same data type as the attribute.

***packet format*: 55 81 03 lenH lenL 0** attrNumsattrList ***AA***

**3.2.2.5** ZBHCI\_CMD\_ZCL\_REPORT\_MSG\_RCV

|  |  |
| --- | --- |
| attrNums | attrList |
| *1Byte* | *N Bytes* |

*attrNums:* the number of the attributes reports be reported

*attrList: the list of the* attributes be reported, the format as follow:

|  |  |  |
| --- | --- | --- |
| attrID | DataType | Data |
| *2Bytes* | *1 Byte* | *N Bytes* |

***packet format*: 55 81 04 lenH lenL 0** attrNumsattrList ***AA***

## Specific Cluster Command Type

3.3.1 Basic Cluster

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_ZCL\_BASIC\_RESET* | 0x0100 |

***3.3.1.1 ZBHCI\_CMD\_ZCL\_BASIC\_RESET***

|  |
| --- |
| ZCLCmdHdr |
| *n-Bytes* |

***packet format*: 55 01 10 lenH lenL 00** ZCLCmdHdr ***AA***

3.3.2 Group Cluster

***3.3.2.1*** *Command Type****(Host)***

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_ZCL\_GROUP\_ADD*  *ZBHCI\_CMD\_ZCL\_GROUP\_VIEW*  *ZBHCI\_CMD\_ZCL\_GROUP\_GET\_MEMBERSHIP*  *ZBHCI\_CMD\_ZCL\_GROUP\_REMOVE*  *ZBHCI\_CMD\_ZCL\_GROUP\_REMOVE\_ALL*  *ZBHCI\_CMD\_ZCL\_GROUP\_ADD\_IF\_IDENTIFYING* | 0x0120  0x0121  0x0122  0x0123  0x0124  0x0125 |

***3.3.2.1.1 ZBHCI\_CMD\_ZCL\_GROUP\_ADD***

|  |  |  |
| --- | --- | --- |
| ZCLCmdHdr | groupId | groupName |
| *n-Bytes* | 2Bytes | nBytes |

*groupId*: group identifier, data type: Unsigned 16-bit integer

*groupName:* group name, data type: Character string

***packet format*: 55 01 20 lenH lenL 00** ZCLCmdHdr groupId groupName ***AA***

***3.3.2.1.2 ZBHCI\_CMD\_ZCL\_GROUP\_VIEW***

|  |  |
| --- | --- |
| ZCLCmdHdr | groupId |
| *n-Bytes* | 2Bytes |

*groupId*: group identifier, data type: Unsigned 16-bit integer

***packet format*: 55 01 21 lenH lenL 00** ZCLCmdHdr groupId ***AA***

***3.3.2.1.3 ZBHCI\_CMD\_ZCL\_GROUP\_GET\_MEMBERSHIP***

|  |  |  |
| --- | --- | --- |
| ZCLCmdHdr | groupCount | groupList |
| *n-Bytes* | 1Byte | nBytes |

*groupCount*: group number

*groupList*: group list

***packet format*: 55 01 22 lenH lenL 00** ZCLCmdHdr groupCount groupList ***AA***

***3.3.2.1.4 ZBHCI\_CMD\_ZCL\_GROUP\_REMOVE***

|  |  |
| --- | --- |
| ZCLCmdHdr | groupId |
| *n-Bytes* | 2Bytes |

groupId: group identifier, data type: Unsigned 16-bit integer

***packet format*: 55 01 23 lenH lenL 00** ZCLCmdHdr groupId ***AA***

***3.3.2.1.5 ZBHCI\_CMD\_ZCL\_GROUP\_REMOVE\_ALL***

***packet format*: 55 01 24 lenH lenL 00** ZCLCmdHdr ***AA***

***3.3.2.1.6 ZBHCI\_CMD\_ZCL\_GROUP\_ADD\_IF\_IDENTIFYING***

|  |  |  |
| --- | --- | --- |
| ZCLCmdHdr | groupId | groupName |
| *n-Bytes* | 2Bytes | nBytes |

*groupId:* group identifier, data type: Unsigned 16-bit integer

*groupName:* group name, data type: Character string

***packet format*: 55 01 25 lenH lenL 00** ZCLCmdHdr groupId groupName ***AA***

***3.3.2.2 Command type(Slave)***

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_ZCL\_GROUP\_ADD\_RSP*  *ZBHCI\_CMD\_ZCL\_GROUP\_VIEW\_RSP*  *ZBHCI\_CMD\_ZCL\_GROUP\_GET\_MEMBERSHIP\_RSP*  *ZBHCI\_CMD\_ZCL\_GROUP\_REMOVE\_RSP* | 0x8120  0x8121  0x8122  0x8123 |

***3.3.2.2.1 ZBHCI\_CMD\_ZCL\_GROUP\_ADD\_RSP***

***3.3.2.2.2 ZBHCI\_CMD\_ZCL\_GROUP\_REMOVE\_RSP***

*status:* The Status field is set to SUCCESS, DUPLICATE\_EXISTS, or INSUFFICIENT\_SPACE as appropriate.

*groupID: received from the host*

***packet format*: 55 81 20 00 03 00** status groupID ***AA***

***packet format*: 55 81 23 00 03 00** status groupID ***AA***

***3.3.2.2.3 ZBHCI\_CMD\_ZCL\_GROUP\_VIEW\_RSP***

*status:* The Status field is set to SUCCESS, or NOT\_FOUND as appropriate.

*groupID: Received from the host*

*groupName: Variable*

***packet format*: 55 81 21 lenH lenL 00** status groupID*groupName* ***AA***

***3.3.2.2.4 ZBHCI\_CMD\_ZCL\_GROUP\_*GET\_MEMBERSHIP*\_RSP***

|  |  |  |
| --- | --- | --- |
| Capabity | groupcnt | grouplist |
| *1 Byte* | 2Bytes | nBytes |

Capabity*:* The remaining capacity of the group table of the device.

groupcnt*:* The number of groups contained in the group list field

grouplist*: The list of groupID in the group list field*

***packet format*: 55 81 22 lenH lenL 00** Capabity groupcntgrouplist ***AA***

3.3.3 Identify Cluster

*3.3.3.1 Command Type****(Host)***

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_ZCL\_IDENTIFY*  *ZBHCI\_CMD\_ZCL\_IDENTIFY\_QUERY* | 0x0130  0x0131 |

***3.3.3.1.1 ZBHCI\_CMD\_ZCL\_IDENTIFY***

|  |  |
| --- | --- |
| ZCLCmdHdr | identifyTime |
| *n-Bytes* | 2Bytes |

*identifyTime:* data type: Unsigned 16-bit integer

***packet format*: 55 01 30 lenH lenL 00** ZCLCmdHdr identifyTime ***AA***

***3.3.3.1.2 ZBHCI\_CMD\_ZCL\_IDENTIFY\_QUERY***

|  |
| --- |
| ZCLCmdHdr |
| *n-Bytes* |

***packet format*: 55 01 31 lenH lenL 00** ZCLCmdHdr ***AA***

*3.3.3.2 Command Type****(Slave)***

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_ZCL\_IDENTIFY\_QUERY\_RSP* | 0x8131 |

On receipt of this command, if the device is currently identifying itself then it SHALL generate an appropriate Identify Query Response command. If the device is not currently identifying itself it SHALL take no further action.

|  |  |  |
| --- | --- | --- |
| ShortAddr | SrcEP | TimeOut |
| *2Bytes* | 1Byte | 2Bytes |

***packet format*: 55 01 30 00 05 00** ShortAddr SrcEPTimeOut ***AA***

3.3.4 On/Off Cluster

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_ZCL\_ONOFF\_ON*  *ZBHCI\_CMD\_ZCL\_ONOFF\_OFF*  *ZBHCI\_CMD\_ZCL\_ONOFF\_TOGGLE* | 0x0140  0x0141  0x0142 |

***3.3.4.1 ZBHCI\_CMD\_ZCL\_ONOFF\_ON***

|  |
| --- |
| ZCLCmdHdr |
| *n-Bytes* |

***packet format*: 55 01 40 lenH lenL 00** ZCLCmdHdr ***AA***

***3.3.4.2 ZBHCI\_CMD\_ZCL\_ONOFF\_OFF***

|  |
| --- |
| ZCLCmdHdr |
| *n-Bytes* |

***packet format*: 55 01 41 lenH lenL 00** ZCLCmdHdr ***AA***

***3.3.4.3 ZBHCI\_CMD\_ZCL\_ONOFF\_TOGGLE***

|  |
| --- |
| ZCLCmdHdr |
| *n-Bytes* |

***packet format*: 55 01 42 lenH lenL 00** ZCLCmdHdr ***AA***

3.3.5 Level Cluster

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_ZCL\_LEVEL\_MOVE2LEVEL*  *ZBHCI\_CMD\_ZCL\_LEVEL\_MOVE*  *ZBHCI\_CMD\_ZCL\_LEVEL\_STEP*  *ZBHCI\_CMD\_ZCL\_LEVEL\_STOP*  *ZBHCI\_CMD\_ZCL\_LEVEL\_MOVE2LEVEL\_WITHONOFF*  *ZBHCI\_CMD\_ZCL\_LEVEL\_MOVE\_WITHONOFF*  *ZBHCI\_CMD\_ZCL\_LEVEL\_STEP\_WITHONOFF*  *ZBHCI\_CMD\_ZCL\_LEVEL\_STOP\_WITHONOFF* | 0x0150  0x0151  0x0152  0x0153  0x0154  0x0155  0x0156  0x0157 |

***3.3.5.1 ZBHCI\_CMD\_ZCL\_LEVEL\_MOVE2LEVEL***

|  |  |  |
| --- | --- | --- |
| ZCLCmdHdr | Level | TransitionTime |
| *n-Bytes* | 1Byte | 2Bytes |

*level:* level

*TransitionTime:* Transition time, tenths of a second

***packet format*: 55 01 50 lenH lenL 00** ZCLCmdHdr level TransitionTime ***AA***

***3.3.5.2 ZBHCI\_CMD\_ZCL\_LEVEL\_MOVE***

|  |  |  |
| --- | --- | --- |
| ZCLCmdHdr | MoveMode | Rate |
| *n-Bytes* | 1Byte | 2Bytes |

On receipt of this command, a device SHALL move from its current level in an up or down direction in a continuous fashion.

*MoveMode:* 0= up, 1= down

*Rate:* the rate of movement in units per second.

***packet format*: 55 01 51 lenH lenL 00** ZCLCmdHdr MoveMode Rate ***AA***

***3.3.5.3 ZBHCI\_CMD\_ZCL\_LEVEL\_STEP***

|  |  |  |  |
| --- | --- | --- | --- |
| ZCLCmdHdr | StepMode | StepSize | TransitionTime |
| *n-Bytes* | 1Byte | 1Byte | 2Bytes |

*StepMode:* 0= up, 1= down

*StepSize:* level units

*TransitionTime:* Transition time, tenths of a second

***packet format*: 55 01 52 lenH lenL 00** ZCLCmdHdr StepMode StepSize TransitionTime ***AA***

***3.3.5.4 ZBHCI\_CMD\_ZCL\_LEVEL\_STOP***

|  |
| --- |
| ZCLCmdHdr |
| *n-Bytes* |

***packet format*: 55 01 53 lenH lenL 00** ZCLCmdHdr ***AA***

***3.3.5.5 ZBHCI\_CMD\_ZCL\_LEVEL\_MOVE2LEVEL\_WITHONOFF***

|  |  |  |
| --- | --- | --- |
| ZCLCmdHdr | Level | Transition time |
| *n-Bytes* | 1Byte | 2Bytes |

*level:* level

*TransitionTime:* Transition time, tenths of a second

***packet format*: 55 01 54 lenH lenL 00** ZCLCmdHdr level TransitionTime ***AA***

***3.3.5.6 ZBHCI\_CMD\_ZCL\_LEVEL\_MOVE\_WITHONOFF***

|  |  |  |
| --- | --- | --- |
| ZCLCmdHdr | MoveMode | Rate |
| *n-Bytes* | 1Byte | 2Bytes |

*MoveMode:* 0= up, 1= down

*Rate:* the rate of movement in units per second.

***packet format*: 55 01 55 lenH lenL 00** ZCLCmdHdr MoveMode Rate ***AA***

***3.3.5.7 ZBHCI\_CMD\_ZCL\_LEVEL\_STEP\_WITHONOFF***

|  |  |  |  |
| --- | --- | --- | --- |
| ZCLCmdHdr | MoveMode | MoveSize | TransitionTime |
| *n-Bytes* | 1Byte | 1Byte | 2Bytes |

*StepMode:* 0= up, 1= down

*StepSize:* level units

*TransitionTime:* Transition time, tenths of a second

***packet format*: 55 01 56 lenH lenL 00** ZCLCmdHdr MoveMode MoveSzie TransitionTime ***AA***

***3.3.5.8 ZBHCI\_CMD\_ZCL\_LEVEL\_STOP\_WITHONOFF***

|  |
| --- |
| ZCLCmdHdr |
| *n-Bytes* |

***packet format*: 55 01 57 lenH lenL 00** ZCLCmdHdr ***AA***

3.3.6 Scene Cluster

***3.3.6.1*** *Command Type****(Host)***

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_ZCL\_SCENE\_ADD*  *ZBHCI\_CMD\_ZCL\_SCENE\_VIEW,*  *ZBHCI\_CMD\_ZCL\_SCENE\_REMOVE,*  *ZBHCI\_CMD\_ZCL\_SCENE\_REMOVE\_ALL,*  *ZBHCI\_CMD\_ZCL\_SCENE\_STORE,*  *ZBHCI\_CMD\_ZCL\_SCENE\_RECALL,*  *ZBHCI\_CMD\_ZCL\_SCENE\_GET\_MEMBERSHIP,* | 0x0160  0x0161  0x0162  0x0163  0x0164  0x0165  0x0166 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ZCLCmdHdr | GroupID | SceneID | TransitionTime | SceneNameLen | SceneName | ExtFieldLen | ExtensionFieldSets |
| *n-Bytes* | 2Byte | 1Byte | 2Bytes | 1Byte | nBytes | 1Byte | nBytes |

***3.3.6.1.1 ZBHCI\_CMD\_ZCL\_SCENE\_ADD***



GroupID:The group ID for which this scene applies, or 0x0000 if the scene is not associated with a group.

SceneID: The identifier, unique within this group, which is used to identify this scene.

TransitionTime: The amount of time, in seconds, it will take for the device to change from its current state to the requested scene.

***packet format*: 55 01 60 lenH lenL 00** ZCLCmdHdr GroupIDSceneID TransitionTime SceneNameLen SceneName ExtFieldLen ExtensionFieldSets ***AA***

***3.3.6.1.2 ZBHCI\_CMD\_ZCL\_SCENE\_VIEW***

|  |  |  |
| --- | --- | --- |
| ZCLCmdHdr | GroupID | SceneID |
| *n-Bytes* | 2Byte | 1Byte |

***packet format*: 55 01 61 lenH lenL 00** ZCLCmdHdr GroupIDSceneID ***AA***

***3.3.6.1.3 ZBHCI\_CMD\_ZCL\_SCENE\_REMOVE***

|  |  |  |
| --- | --- | --- |
| ZCLCmdHdr | GroupID | SceneID |
| *n-Bytes* | 1Byte | 1Byte |

On receipt of this command, the device SHALL (if possible) remove from its Scene Table the entry with this Scene ID and group ID.

***packet format*: 55 01 62 lenH lenL 00** ZCLCmdHdr GroupIDSceneID ***AA***

***3.3.6.1.4 ZBHCI\_CMD\_ZCL\_SCENE\_REMOVE\_ALL***

|  |  |
| --- | --- |
| ZCLCmdHdr | GroupID |
| *n-Bytes* | 1Byte |

On receipt of this command, the device SHALL, if possible, remove from its Scene Table all entries with this Group ID.

***packet format*: 55 01 63 lenH lenL 00** ZCLCmdHdr GroupID ***AA***

***3.3.6.1.5 ZBHCI\_CMD\_ZCL\_SCENE\_STORE***

|  |  |  |
| --- | --- | --- |
| ZCLCmdHdr | GroupID | SceneID |
| *n-Bytes* | 1Byte | 1Byte |

On receipt of this command, the device SHALL (if possible) add an entry to the Scene Table with the Scene ID and Group ID given in the command, and all extension field sets corresponding to the current state of other clusters on the device.

***packet format*: 55 01 64 lenH lenL 00** ZCLCmdHdr GroupIDSceneID ***AA***

***3.3.6.1.6 ZBHCI\_CMD\_ZCL\_SCENE\_RECALL***

|  |  |  |
| --- | --- | --- |
| ZCLCmdHdr | GroupID | SceneID |
| *n-Bytes* | 1Byte | 1Byte |

***packet format*: 55 01 65 lenH lenL 00** ZCLCmdHdr GroupIDSceneID ***AA***

***3.3.6.1.7 ZBHCI\_CMD\_ZCL\_SCENE\_GET\_MEMBERSHIP***

|  |  |
| --- | --- |
| ZCLCmdHdr | GroupID |
| *n-Bytes* | 1Byte |

***packet format*: 55 01 66lenH lenL 00** ZCLCmdHdr GroupID ***AA***

***3.3.6.2*** *Command Type****(Slave)***

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_ZCL\_SCENE\_ADD\_RSP*  *ZBHCI\_CMD\_ZCL\_SCENE\_VIEW\_RSP,*  *ZBHCI\_CMD\_ZCL\_SCENE\_REMOVE\_RSP,*  *ZBHCI\_CMD\_ZCL\_SCENE\_REMOVE\_ALL\_RSP,*  *ZBHCI\_CMD\_ZCL\_SCENE\_STORE\_RSP,*  *ZBHCI\_CMD\_ZCL\_SCENE\_GET\_MEMBERSHIP\_RSP,* | 0x8160  0x8161  0x8162  0x8163  0x8164  0x8166 |

***3.3.6.2.1 ZBHCI\_CMD\_ZCL\_SCENE\_ADD\_RSP***

|  |  |  |
| --- | --- | --- |
| Status | GroupID | SceneID |
| 1Byte | 2Bytes | 1Byte |

Status: The Status field is set to SUCCESS, INSUFFICIENT\_SPACE or INVALID\_FIELD (the

group is not present in the Group Table) as appropriate.

***packet format*: 55 81 60 00 04 00** Status GroupID SceneID ***AA***

***3.3.6.2.2 ZBHCI\_CMD\_ZCL\_SCENE\_VIEW\_RSP***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Status | GroupID | SceneID | TransitionTime | SceneName | ExtensionFieldSets |
| *1Byte* | 2Byte | 1Byte | 2Bytes | nBytes | nBytes |

Status: The Status field is set to SUCCESS, NOT\_FOUND (the scene is not present in the Scene

Table) or INVALID\_FIELD (the group is not present in the Group Table) as appropriate. If the

status is SUCCESS, the Transition time, Scene Name and Extension field fields are copied from

the corresponding fields in the table entry, otherwise they are omitted.

***packet format*: 55 81 61 lenH lenL 00** Status GroupID SceneID TransitionTimeSceneName ExtensionFieldSets ***AA***

***3.3.6.2.1 ZBHCI\_CMD\_ZCL\_SCENE\_REMOVE\_RSP***

|  |  |  |
| --- | --- | --- |
| Status | GroupID | SceneID |
| 1Byte | 2Bytes | 1Byte |

Status: The Status field is set to SUCCESS, NOT\_FOUND (the scene is not present in the Scene

Table) or INVALID\_FIELD (the group is not present in the Group Table) as appropriate.

***packet format*: 55 81 62 00 04 00** Status GroupID SceneID ***AA***

***3.3.6.2.1 ZBHCI\_CMD\_ZCL\_SCENE\_REMOVE\_ALL\_RSP***

|  |  |
| --- | --- |
| Status | GroupID |
| 1Byte | 2Bytes |

Status: The Status field is set to SUCCESS or INVALID\_FIELD (the group is not present in the

Group Table) as appropriate.

***packet format*: 55 81 63 00 03 00** Status GroupID ***AA***

***3.3.6.2.1 ZBHCI\_CMD\_ZCL\_SCENE\_STORE\_RSP***

|  |  |  |
| --- | --- | --- |
| Status | GroupID | SceneID |
| 1Byte | 2Bytes | 1Byte |

Status: The Status field is set to SUCCESS, INSUFFICIENT\_SPACE or INVALID\_FIELD (the

group is not present in the Group Table) as appropriate.

***packet format*: 55 81 64 00 04 00** Status GroupID SceneID ***AA***

***3.3.6.2.1 ZBHCI\_CMD\_ZCL\_SCENE\_* GET\_MEMBERSHIP*\_RSP***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Status | Capacity | GroupID | SceneCnt | SceneList |
| 1Byte | 1Byte | 2Bytes | 1Byte | N Bytes |

Status: The Status field SHALL contain SUCCESS or INVALID\_FIELD (the group is not present

in the Group Table) as appropriate.

Capacity: contain the remaining capacity of the scene table of the device.

SceneCnt: contain the number of scenes contained in the Scene list field.

SceneList: contain the identifiers of all the scenes in the scene table with the corresponding Group ID.

If the status is not SUCCESS, then the Scene count and Scene list field are omitted.

***packet format*: 55 81 66 lenH lenL 00** Status Capacity GroupID SceneCnt SceneList ***AA***

3.3.7 OTA Cluster

|  |  |
| --- | --- |
| Type | Value |
| *ZBHCI\_CMD\_ZCL\_OTA\_IMAGE\_NOTIFY* | 0x0190 |

***3.3.7.1*** *ZBHCI\_CMD\_ZCL\_OTA\_IMAGE\_NOTIFY*

|  |  |  |
| --- | --- | --- |
| ZCLCmdHdr | PayloadType | QueryJitter |
| *n-Bytes* | *1Byte* | 1Byte |

The purpose of sending Image Notify command is so the server has a way to notify client devices of when the OTA upgrade images are available for them. If all the parameters are correct, then the OTA process will start automatically.

The parameters (manufacture code, image type, and new file version) are read from the header of the new image file, so don’t need input from the hci command.

***packet format*: 55 01 90 lenH lenL 00** ZCLCmdHdr PayloadType QueryJitter ***AA***