



zigbee PRO 2017 Command Line (zbcli)
Reference Manual

Dana Sy

Copyright 2018, Exegin Technologies Limited

Contents

List of all zbcli commands	2
aps_add_ep	8
aps_bind	9
aps_data_req	10
aps_frag	11
aps_group	12
aps_link_key	13
aps_remove_device	14
aps_req_key	15
aps_reset	16
aps_switch_key	17
aps_tc_policy	18
aps_transport_key	19
aps_tx_opts	20
aps_update_device	21
debug	22
destroy	23
echo	24
example_xlink	25
exit	26
get_ib	27
gp_data_req	28
gp_dev_key	29

hash	30
help	31
ic	32
init_tbl_size	33
init	34
ip_data_req	35
log	36
log_level	37
monitor	38
nwk_addr_map	39
nwk_end_timeout	40
nwk_fc	41
nwk_fc_cooldown	42
nwk_ifc	43
nwk_joinlist	44
nwk_key	45
nwk_leave	46
nwk_nextaddr	47
nwk_pjoin	48
nwk_poll	49
nwk_raw_frame	50
nwk_rejoin	51
nwk_reset	52
nwk_route_delete	53
nwk_route_request	54

nwk_send_conflict	55
nwk_status	56
nwk_txpower	57
nwk_unknown_cmd	58
poll	59
prompt	60
quit	61
set_ib	62
shell	63
source	64
startup_channellist	65
startup_config	66
startup_fndbind	69
startup_form	70
startup_join	71
startup_persist	72
startup_rejoin	73
startup_tc_rejoin	74
status	75
status_code	76
testcase	77
version	78
wait	79
wpan	80
zb_leave	81

zb_reset	82
zcl_alarm_client	83
zcl_alarm_server	84
zcl_attr_disc	85
zcl_attr_read	86
zcl_attr_report	87
zcl_attr_reset	88
zcl_attr_write	89
zcl_ballast_config_client	90
zcl_ballast_config_server	91
zcl_basic	92
zcl_calendar_client	93
zcl_calendar_server	94
zcl_color_client	95
zcl_color_server	99
zcl_device_mgmt_client	100
zcl_device_mgmt_server	101
zcl_device_temp_client	102
zcl_device_temp_server	103
zcl_doorlock	104
zcl_drlc_client	105
zcl_drlc_server	106
zcl_endpoint_remove	108
zcl_events_client	109
zcl_events_server	110

zcl_fan	111
zcl_groups_client	112
zcl_groups_server	113
zcl_ias_ace_device	114
zcl_ias_cie_device	115
zcl_ias_warn_device	117
zcl_ias_zone_device	118
zcl_identify_client	119
zcl_identify_server	120
zcl_illum_meas_client	121
zcl_illum_meas_server	122
zcl_level_client	123
zcl_level_server	124
zcl_messaging_client	125
zcl_messaging_server	126
zcl_metering_client	127
zcl_metering_server	128
zcl_occupancy	129
zcl_onoff_client	130
zcl_onoff_server	131
zcl_ota_client	132
zcl_ota_server	133
zcl_power	134
zcl_prepay_client	135
zcl_prepay_server	136

zcl_price_client	137
zcl_price_server	138
zcl_sandbox_client	139
zcl_sandbox_server	140
zcl_scenes_client	141
zcl_scenes_server	143
zcl_seclevel	144
zcl_temperature_server	145
zcl_therm	146
zcl_time_client	147
zcl_time_server	148
zcl_touchlink	149
zcl_tp2	150
zcl_tunnel_client	151
zcl_tunnel_server	152
zcl_water_content_meas_client	153
zcl_water_content_meas_server	154
zcl_window_client	155
zcl_window_server	156
zdo_active_ep_req	157
zdo_bind_req	158
zdo_complex_desc_req	159
zdo_complex_desc_set	160
zdo_device_annce	161
zdo_discovery_cache_req	162

zdo_end_bind_req	163
zdo_ieee_addr_req	164
zdo_match_bind	165
zdo_match_desc_req	166
zdo_mgmt_bind_req	167
zdo_mgmt_leave_req	168
zdo_mgmt_lqi_req	169
zdo_mgmt_nwk_enh_update_req	170
zdo_mgmt_nwk_joining_list	171
zdo_mgmt_nwk_update_req	172
zdo_mgmt_permit_joining_req	173
zdo_mgmt_unsolicit_enh_update	174
zdo_node_desc_req	175
zdo_not_supported_req	176
zdo_nwk_addr_req	177
zdo_power_desc_req	178
zdo_simple_desc_req	179
zdo_system_server_disc_req	180
zdo_unbind_req	181
zdo_user_desc_req	182
zdo_user_desc_set	183
zgp_commands	184
zgp_pairing	185

Contents:

List of all zbcli commands

aps_add_ep - Add endpoint to device.

aps_bind - Display or insert bindings in the apsBindingTable

aps_data_req - Send APS packet

aps_frag - Selectively drop fragments from large APS packets for testing.

aps_group - Manage group membership of the local device's endpoints.

aps_link_key - Display or insert keys into the apsDeviceKeyPairSet

aps_remove_device - Request the removal of a remote device from the network using APS commands.

aps_req_key - Request a key from a trust center

aps_reset - Reset the APS layer attributes

aps_switch_key - Issue an APS command requesting the network to change the active encryption key.

aps_tc_policy - Display and manipulate the trust center policy flags

aps_transport_key - Issue a new encryption key to remote devices on the network

aps_tx_opts - Manipulate the default tx options used when sending APS messages from this tool.

aps_update_device - Override the encryption for the Update Device command.

debug - Display debug information

destroy - Resets zigbee stack, detaches and frees all radio devices, and frees zigbee stack.

echo - Enable or disable command echo

***example_xlink**

exit - Terminate program and exit

get_ib - Read attribute values from the stack's information base.

gp_data_req - Insert a greenpower frame into the gpTxQueue.

gp_dev_key - Manipulate the Green Power key table.

hash - Compute the AES-MMO hash of arbitrary data.

help - List available commands or display their help and syntax description.

ic - Parse, manipulate and load install codes into the stack.

init_tbl_size - Change the table sizes of various ZigBee tables

init - Initialize the ZigBee Stack, attach all radio devices and set default logging options.

ip_data_req - Send an interpan frame

log - Configure logging options

log_level - Set stack logging options

monitor - Monitor and display ZDO traffic.

nwk_addr_map - Display the address map table

nwk_end_timeout - Display or overwrite the end device timeout configuration in the neighbor table.

nwk_fc - Display and manipulate network security frame counters

nwk_fc_cooldown - Manipulate the network frame counter cooldown

nwk_ifc - Display and manipulate the network interface table

nwk_joinlist - Display and manipulate the IEEE device joining list

nwk_key - Display and manipulate network keys on the local device.

nwk_leave - Perform an NLME-LEAVE.request

nwk_nextaddr - Set the network address to be assigned to the next child that joins

nwk_pjoin - Enable or disable permit join

nwk_poll - Change automated polling behaviour when operating as a sleepy end device

nwk_raw_frame - Transmit a raw MAC frame

nwk_rejoin - Perform a secured rejoin with NLME-JOIN.request

nwk_reset - Reset the network layer to defaults, or issue a warm start.

nwk_route_delete - Remove entries from the network routing table

nwk_route_request - Send a network route request to the network

nwk_send_conflict - Forge a network address conflict command.

nwk_status - Send a Network Status command

nwk_txpower - Adjust the radio transmit Power

nwk_unknown_cmd - Transmit a network command with an unsupported ID or version.

poll - Execute a zbcli command when an events occur on file handle with a matching value

prompt - Change the command line prompt

quit - Terminate program and exit

set_ib - Set attribute values in the stack's information base.

shell - Execute commands in the system shell

source - read commands from FILE until EOF. On EOF input resumes from console

startup_channellist - Display and update stacks's channel set

startup_config - Display and update the startup configuration

startup_fndbind - Join a network using the startup configuration

startup_form - Form a network using the startup configuration

startup_join - Join a network using the startup configuration

startup_persist - Resume network operation from saved persistence file.

startup_rejoin - Join a network with secure rejoin commands using the startup configuration

startup_tc_rejoin - Join a network with TC rejoin commands using the startup configuration

status - Display status information

status_code - Describe status codes returned by the ZigBee stack

testcase - Enable special ZigBee stack behaviour for testing.

version - display versions of components

wait - Wait for the given amount of time. Stack and packet processing will still take place.

wpan - Create radio devices

zb_leave - Leave the network.

zb_reset - Leave the network.

zcl_alarm_client - ZCL alarm client cluster

zcl_alarm_server - ZCL alarm Server cluster

zcl_attr_disc - Send ZCL discover attribute command

zcl_attr_read - Send ZCL read attribute command

zcl_attr_report - Send ZCL configure or read reporting attribute command

zcl_attr_reset - resets all clusters attributes

zcl_attr_write - Send ZCL write attribute command

zcl_ballast_config_client - ZCL ballast configuration client cluster

zcl_ballast_config_server - ZCL ballast configuration server cluster

zcl_basic - ZCL basic cluster

zcl_calendar_client - ZCL calendar client cluster

zcl_calendar_server - ZCL calendar server cluster

zcl_color_client - Create or modify ZCL Color client cluster or make requests to remote server via binding

zcl_color_server - Create or modify ZCL Color server cluster

zcl_device_mgmt_client - ZCL Device Management Client Cluster

zcl_device_mgmt_server - ZCL Device Management Server Cluster

zcl_device_temp_client - ZCL device temperature client cluster

zcl_device_temp_server - ZCL device temperature server cluster

zcl_doorlock - ZCL doorlock config cluster

zcl_drlc_client - ZCL DRLC Client Cluster
zcl_drlc_server - ZCL DRLC Server Cluster
zcl_endpoint_remove - Remove an Endpoint
zcl_events_client - ZCL Events Client Cluster
zcl_events_server - ZCL Events Server Cluster
zcl_fan - ZCL FAN config cluster
zcl_groups_client - ZCL Groups Client Cluster
zcl_groups_server - ZCL Groups Server Cluster
zcl_ias_ace_device - ZCL IAS ACE Device Cluster
zcl_ias_cie_device - ZCL IAS CIE Device Cluster
zcl_ias_warn_device - ZCL IAS Warning Device Cluster
zcl_ias_zone_device - ZCL IAS Zone Device Cluster
zcl_identify_client - ZCL Identify Client Cluster
zcl_identify_server - ZCL Identify Server Cluster
zcl_illum_meas_client - ZCL illuminance measurement client cluster
zcl_illum_meas_server - ZCL illuminance measurement server cluster
zcl_level_client - ZCL level client cluster init and commands
zcl_level_server - ZCL level server cluster init
zcl_messaging_client - ZCL Messaging Client Cluster
zcl_messaging_server - ZCL Messaging Server Cluster
zcl_metering_client - ZCL Metering Client Cluster
zcl_metering_server - ZCL Metering Server Cluster
zcl_occupancy - ZCL occupancy config cluster
zcl_onoff_client - ZCL OnOff Client Cluster
zcl_onoff_server - ZCL OnOff Server Cluster
zcl_ota_client - ZCL OTA cluster commands to exercise and test
zcl_ota_server - ZCL OTA cluster commands to exercise and test
zcl_power - ZCL power config cluster
zcl_prepay_client - ZCL Prepay Client Cluster
zcl_prepay_server - ZCL Prepay Server Cluster
zcl_price_client - ZCL Price Client Cluster
zcl_price_server - ZCL Price Server Cluster

zcl_sandbox_client - ZCL Sandbox Client Cluster

zcl_sandbox_server - ZCL Sandbox Server Cluster

zcl_scenes_client - ZCL Scenes Client Cluster

zcl_scenes_server - ZCL Scenes Server Cluster

zcl_seclevel - Configure the default ZCL SE minimum security level for incoming ZCL messages.

zcl_temperature_server - ZCL Temperature Measurement Server cluster

zcl_therm - ZCL THERM luster

zcl_time_client - ZCL Time Client Cluster

zcl_time_server - ZCL Time Server Cluster

zcl_touchlink - ZCL Touchlink Initiator and Target Cluster Commands

zcl_tp2 - ZCL Test Profile 2 Commands

zcl_tunnel_client - ZCL Tunnel Client Cluster

zcl_tunnel_server - ZCL Tunnel Server Cluster

zcl_water_content_meas_client - ZCL water content measurement client cluster

zcl_water_content_meas_server - ZCL water content measurement server cluster

zcl_window_client - ZCL window closure client cluster

zcl_window_server - ZCL window closure server cluster

zdo_active_ep_req - Send a ZDO Active_EP_req command to a remote device.

zdo_bind_req - ZDO Bind Request

zdo_complex_desc_req - Send a ZDO Complex_desc_req to a remote device.

zdo_complex_desc_set - Configure or clear the local device's complex descriptor

zdo_device_annce - Send a ZDO device announcement

zdo_discovery_cache_req - Send a ZDO Discovery Cache request (Deprecated command* - for GU use only)

zdo_end_bind_req - Send a ZDO End_Device_Bind_req to the coordinator

zdo_ieee_addr_req - Send a ZDO IEEE_Addr_req to a remote device

zdo_match_bind - Perform match descriptor request and perform binding on first discovered endpoint.

zdo_match_desc_req - Send a ZDO Match_desc_req to a remote device

zdo_mgmt_bind_req - Send a ZDO Mgmt_Bind_req to a remote device

zdo_mgmt_leave_req - Send a ZDO Mgmt_Leave_req to a remote device

zdo_mgmt_lqi_req - Send a ZDO Mgmt_Lqi_req to a remote device

zdo_mgmt_nwk_enh_update_req - Send a ZDO Mgmt_Nwk_Enh_update_req to a remote device

zdo_mgmt_nwk_joining_list - Send ZDO Mgmt_Nwk_Joininglist message to remote devices

zdo_mgmt_nwk_update_req - Frequency agility commands using the ZDO Mgmt_Nwk_Update_req message

zdo_mgmt_permit_joining_req - Send a ZDO Mgmt_Permit_Joining_req to a remote device

zdo_mgmt_unsolicit_enh_update - Send a ZDO MGMT_NWK_Unsolicited_Enhanced_Update_Notify

zdo_node_desc_req - Send a ZDO Node_desc_req to a remote device

zdo_not_supported_req - Sends an unsupported ZDP command (0x00ff)

zdo_nwk_addr_req - Send a ZDO Nwk_Addr_req to a remote device

zdo_power_desc_req - Send a ZDO Power_desc_req to a remote device

zdo_simple_desc_req - Send a ZDO Simple_desc_req to a remote device

zdo_system_server_disc_req - Broadcast a ZDO System_Server_discovery_req to the network.

zdo_unbind_req - ZDO Unbind Request

zdo_user_desc_req - Send a ZDO User_desc_req to a remote device

zdo_user_desc_set - Send a ZDO User_desc_set to a remote device

zgp_commands - Send a ZGP GP Command

zgp_pairing - Send a ZGP GP Pairing Command

aps_add_ep

usage: `aps_add_ep [-e ENDPT] [-p PROFILE] [-d DEVICE] [-v VERSION]`

Add endpoint to device.

optional arguments:

- e, -ep ENDPT endpoint
- p, -profile PROFILE profile ID (ha = 0x0104, se = 0x0109, zll = 0xc05e)
- d, -device DEVICE device ID
- v, -version VERSION version
- help Display this message and exit

aps_bind

Usage: aps_bind <ARGS>

list

add SRC_ENDPT CLUSTER DST_ENDPT group|ext DST_ADDR

Display or insert bindings in the apsBindingTable

optional arguments:

-help Display this message and exit

aps_data_req

usage: aps_data_req [-e ADDR] [-g GROUP] [-n ADDR] [-d DSTEP] [-s SRCEP] [-c CLUSTER] [-p PROFILE] [-r RADIUS] [-l RNDSZ] [-ack] [-frag] [PAYLOAD]

Send APS packet

positional arguments:

PAYLOAD Packet payload data

optional arguments:

-e, --ext ADDR Send to an extended destination address

-g, --group GROUP Send to a group address

-n, --nwk ADDR Send to a short address

-d, --dstep DSTEP Destination endpoint

-s, --srcep SRCEP Source endpoint

-c, --cluster CLUSTER Cluster ID

-p, --profile PROFILE Source endpoint

-r, --radius RADIUS Maximum number of hops the packet may be routed

-l, --length RNDSZ Generate a random payload of RNDSZ bytes in length

-ack Request acknowledgement

-frag Enable APS fragmentation

-help Display this message and exit

aps_frag

Usage: aps_frag <COMMAND>
 tx_drop_add <BLOCK_NUM>
 tx_drop_clear

Selectively drop fragments from large APS packets for testing.

optional arguments:

 -help Display this message and exit

aps_group

Usage: `aps_group <ARGS>`
 `add <GROUP_ADDR> <ENDPT>`
 `list`
 `remove <GROUP_ADDR> <ENDPT>`
 `remove-all <ENDPT>`

Manage group membership of the local device's endpoints.

optional arguments:

`-help` Display this message and exit

aps_link_key

Usage: aps_link_key add|list|remove
APS Unique TC Link Key operations.

add EUI64 KEY : Add a TC Link Key

list EUI64 : List the TC Link Key for the partner address, if present

remove EUI64 : Remove a TC Link Key

Display or insert keys into the apsDeviceKeyPairSet

optional arguments:

-help Display this message and exit

aps_remove_device

Usage: `aps_remove_device DST-ADDR TARGET [OPTIONS]`

DST-ADDR : EUI64 address to send this command. Typically this is either the parent of TARGET, or TARGET itself.

TARGET : EUI64 address of device to remove

OPTIONS

`-no_aps_security` : Don't APS encrypt this command.

Request the removal of a remote device from the network using APS commands.

optional arguments:

`-help` Display this message and exit

aps_req_key

usage: aps_req_key DST KEYTYPE [-partner PARTNER]

Request a key from a trust center

positional arguments:

DST Extended address of the destination (e.g. trust center)

KEYTYPE Key type to request (nwk = 0x01, app = 0x03, tc = 0x04)

optional arguments:

-partner PARTNER Extended address of the partner (only used with application keys)

-help Display this message and exit

aps_reset

usage: `aps_reset`

Reset the APS layer attributes

optional arguments:

- help Display this message and exit

aps_switch_key

Usage: aps_switch_key MODE ADDR SEQNO

MODE : short | ext

ADDR : Short or extended address

SEQNO : Key sequence number to switch to

Issue an APS command requesting the network to change the active encryption key.

optional arguments:

-help Display this message and exit

aps_tc_policy

Usage: `aps_tc_policy [enable|disable <POLICY>]`

POLICIES:

- allow_joins
- allow_rejoins
- ic_supported
- ic_required
- tclk_update
- dummy_key

Display and manipulate the trust center policy flags

optional arguments:

- help Display this message and exit

aps_transport_key

Usage: `aps_transport_key TYPE ARGS`

Send a NWK Transport Key command.

`app -partner PARTNER -key KEY [-initiator]`

`nwk [-key KEY] [-seqno SEQNO]`

KEY : New key, otherwise it's randomly generated.

SEQNO : Key sequence number to use, otherwise current network key seqno is used.

OPTIONS

`-dstnwk ADDR` : Specify the destination network address

`-dstext ADDR` : Specify the destination extended address

If neither are specified, the Transport Key is sent to 0xFFFF.

Issue a new encryption key to remote devices on the network

positional arguments:

TYPE `app` or `nwk`

optional arguments:

`-dstnwk DSTNWK` Destination Short Address

`-dstext DSTNWK` Destination Extended Address

`-seqno SEQNO` Network Key Sequence Number

`-partner PARTNER` Extended Partner Address (for APP link key only)

`-initiator INITIATOR` for APP link key only

`-key KEY` Key

`-help` Display this message and exit

aps_tx_opts

Usage: `aps_tx_opts [<TX_OPTS>]`

TX_OPTS ; can be a `uint8_t` hex value, or a combination of:
[ack] [frag] [nwk] [security]

Manipulate the default tx options used when sending APS messages from this tool.

optional arguments:

-help Display this message and exit

aps_update_device

Usage: `aps_update_device <ARGS>`

keyid ID Override the encryption for the Update Device command.

Default behaviour is to use APS Link Key security if a unique link key exists to the TC, or if no security or using a preconfigured link key, NWK security is used.

ID `nwk | aps | both | default`

Override the encryption for the Update Device command.

optional arguments:

`-help` Display this message and exit

debug

Usage: debug ARG

ARGUMENTS

constants

heap

wpan

zcl

Display debug information

optional arguments:

-help Display this message and exit

destroy

Usage: destroy

Resets zigbee stack, detaches and frees all radio devices, and frees zigbee stack.

optional arguments:

- help Display this message and exit

echo

Usage: echo on|off

Enable or disable command echo

Enable or disable command echo

optional arguments:

-help Display this message and exit

example_xlink

Usage: example_xlink SRC_EP COMMAND

COMMANDS:

init

send NWKADDR DST_EP [NUM]

optional arguments:

-help Display this message and exit

exit

usage: exit [-leave]

Terminate program and exit

optional arguments:

- leave Leave network before terminating.
- help Display this message and exit

get_ib

Usage: get_ib [ID] [OP EXPECT]

If no parameters provided, a list of supported IB ID values and names are printed.

ID IB ID by hex value (e.g. 80) or text name (e.g. panid)

OP Operator to use with EXPECT. Can be one of:

== != < <= > >=

EXPECT Value to expect (for test verification)

Read attribute values from the stack's information base.

optional arguments:

-help Display this message and exit

gp_data_req

usage: gp_data_req [-e EP] [-l SEC] GPDID CMD [PAYLOAD]

Insert a greenpower frame into the gpTxQueue.

positional arguments:

GPDID GPD Source ID or EUI-64 address

CMD Greenpower command ID

PAYLOAD Greenpower command payload

optional arguments:

-e, --endpoint EP Use an applicationId of 0b10 with destination endpoint EP

-l, --lifetime SEC gpTxQueue lifetime in seconds

-help Display this message and exit

gp_dev_key

usage: gp_dev_key [-e EP] [-t TYPE] [-l LEVEL] [-m MIC] GPDID [KEY]

Manipulate the Green Power key table.

positional arguments:

GPDID GPD Source ID or EUI-64 address

KEY Greenpower security key

optional arguments:

-e, --endpoint EP Use an applicationId of 0b10 with destination endpoint EP

-t, --type TYPE Greenpower key type (default: none)

-l, --level LEVEL Require a security level of LEVEL (default: none)

-m, --mic MIC Encrypted key MIC

-help Display this message and exit

hash

Usage: hash <aesmmo> <HEX_DATA>

HEX_DATA is an array of hexadecimal bytes of any length

Compute the AES-MMO hash of arbitrary data.

optional arguments:

-help Display this message and exit

help

usage: help [-l] [-list] [MATCH]

List available commands or display their help and syntax description.

positional arguments:

MATCH Display help for matching commands

optional arguments:

-l, -list List commands with brief descriptions

-help Display this message and exit

ic

Usage: ic <COMMAND> check <CODE> [install]

create <LENGTH> Generates an Install Code of LENGTH bytes, including the 2-byte CRC.

crc <CODE> Computes and prints the CRC for the input Install Code.

check <CODE> [-preconf] Checks the code and prints the key if successful.
-preconf will load the key into the startup
preconfigured link key.

add <EUI64> <CODE> Adds a link-key device pair.

file <FILENAME> Adds link-key device pairs from a file.
Line format: ic <EUI64> <CODE>

Parse, manipulate and load install codes into the stack.

optional arguments:

-help Display this message and exit

init_tbl_size

usage: init_tbl_size [-c] [-n SIZE] [-r SIZE] [-m SIZE] [-l SIZE] [-coord] [-a SIZE]

Change the table sizes of various ZigBee tables

optional arguments:

- n, -nwk-nnt SIZE Sets size of NWK Neighbour table
- r, -nwk-route SIZE Sets size of NWK Routing table
- m, -nwk-addr SIZE Sets size of NWK Address Map table
- l, -aps-link SIZE Sets size of APS Link Key table
- c, -coord Set all table sizes to 200 entries
- a, -all SIZE Sets all table size to user defined entries
- help Display this message and exit

init

usage: init [EXTADDR]

Initialize the ZigBee Stack, attach all radio devices and set default logging options.

positional arguments:

EXTADDR Local device extended address

optional arguments:

-help Display this message and exit

ip_data_req

usage: ip_data_req [-d ADDR] [-g GROUP] PANID PROFILE CLUSTER [PAYLOAD]

Send an interpan frame

positional arguments:

PANID Destination PAN identifier

PROFILE Profile identifier

CLUSTER Cluster identifier

PAYLOAD Packet payload data

optional arguments:

-d, --dest ADDR Send to an extended destination address

-g, --group GROUP Send to a group address

-help Display this message and exit

log

usage: log [-c] [-f FILE] [-close] [-t FMT]

Configure logging options

optional arguments:

- f, --filename FILE start logging to this file, after closing the existing file if there is one
- c, --close close the current log file if there is one
- t, --timestamp FMT provide a strftime format string for the timestamp to be appended to filename, otherwise a default will be used
- help Display this message and exit

log_level

Usage: log_level <OPTION>

where OPTION is one of

- fatal
- error
- warning
- info
- debug
- zcl_cluster
- zcl_report
- persist
- routing
- lqi
- nwksec
- apssend
- apsfrag
- greenpower
- rsi

Set stack logging options

optional arguments:

- help Display this message and exit

monitor

Usage: monitor [ZDO]

Monitor for and display ZDO traffic, where ZDO can be:
deviceannce Device Announce Messages

Monitor and display ZDO traffic.

optional arguments:

-help Display this message and exit

nwk_addr_map

nwk_addr_map ext|nwk ADDR

Display the address map table

optional arguments:

-help Display this message and exit

nwk_end_timeout

usage: nwk_end_timeout EXTADDR [TIMEOUT]

Display or overwrite the end device timeout configuration in the neighbor table.

positional arguments:

EXTADDR Extended address to search in the neighbor table

TIMEOUT New timeout enumeration.

optional arguments:

-help Display this message and exit

nwk_fc

nwk_fc in EXTADDR

Display the current incoming network frame counter for device with address EXTADDR

nwk_fc COUNTER [EXTADDR]

Reset the incoming frame counter for address EXTADDR to the 32-bit COUNTER value. If extended address EXTADDR omitted, 0, or our own, then the outgoing frame counter is reset. The ActiveKeySeqNo NWK NIB is used to determine the currently active network key and matching frame counter

Display and manipulate network security frame counters

optional arguments:

-help Display this message and exit

nwk_fc_cooldown

Usage: nwk_fc_cooldown [<SECONDS>]

Set nwkFrameCounterCooldown to SEC seconds or display current if SEC omitted

A value of 0 means never reset, 1 is virtually disable checking

nwk_ifc

Usage: nwk_ifc <ARGS>

info

routers <IDX> enable|disable

Display and manipulate the network interface table

optional arguments:

-help Display this message and exit

nwk_joinlist

Usage: nwk_joinlist <COMMAND>

COMMANDS

clear : Clear all IeeeJoiningList entries.

 Increments the IeeeJoiningListUpdateID if the list changes.

add <EXTADDR...> : Add an extended address (or list of addresses) to IeeeJoiningList.

 Increments the IeeeJoiningListUpdateID.

policy <none|all|ieee> : Set the JoiningPolicy.

 Increments the IeeeJoiningListUpdateID.

refresh : If JoiningPolicy == IEEEELIST_JOIN, this command restarts the mibIeeeExpiryIntervalCountdown timer. The mibIeeeExpiryInterval is configurable through ZB_NWK_NIB_ID_JoiningListExpiryInterval.

Display and manipulate the IEEE device joining list

optional arguments:

 -help Display this message and exit

nwk_key

Usage: nwk_key <COMMAND>

list print current key

set <NWKKEY> <SEQNUM> set the network key to NWKKEY

Display and manipulate network keys on the local device.

optional arguments:

-help Display this message and exit

nwk_leave

usage: nwk_leave [-rejoin] [-remove] DEVICE_ADDR

Perform an NLME-LEAVE.request

positional arguments:

DEVICE_ADDR IEEE64 address of the child device to remove, otherwise set to the local device if omitted.

optional arguments:

- rejoin Sets the 'rejoin' flag
- remove Sets the 'remove children' flag
- help Display this message and exit

nwk_nextaddr

usage: nwk_nextaddr NWKADDR

Set the network address to be assigned to the next child that joins

positional arguments:

NWKADDR Network address to be assigned.

optional arguments:

-help Display this message and exit

nwk_pjoin

Usage: nwk_pjoin <INTERVAL>

Permit join for INTERVAL seconds (0-255), if

INTERVAL is 0 or “off”, permit join is disabled immediately

INTERVAL is 255 or “on” permit join is enabled indefinitely

otherwise joining is permitted for the next INTERVAL seconds

Enable or disable permit join

optional arguments:

-help Display this message and exit

nwk_poll

Usage: nwk_poll [<ARGS>]

Warning: this command should not normally be required.

If you want to setup a sleepy device to periodically poll its parent, use “set_ib nwkSlowPollPeriod TIME_MS” instead.

OPTIONS

fast [<PERIOD>] Change to fast poll mode.

Optionally specify the polling period (mS)

release Cancel one fast request, enters slow poll mode
(if enabled) when all are cancelled

Change automated polling behaviour when operating as a sleepy end device

optional arguments:

-help Display this message and exit

nwk_raw_frame

Usage: nwk_raw_frame <SRC_ADDRMODE> <DST_ADDRMODE> <DST_ADDR>
<PANID>
 <TXOPTS> <MSDU>

SRC_ADDRMODE : long (0x3) | short (0x2)

DST_ADDRMODE : long (0x3) | short (0x2)

DST_ADDR : The destination address

DST_PANID : The destination PAN ID

TXOPTS : uint8_t value of txOption

MSDU : MSDU octets, Octets are separated by spaces

Transmit a raw MAC frame

optional arguments:

 -help Display this message and exit

nwk_rejoin

usage: nwk_rejoin

Perform a secured rejoin with NLME-JOIN.request

optional arguments:

-help Display this message and exit

nwk_reset

Usage: nwk_reset [warm_start]

Reset the network layer to defaults, or issue a warm start.

optional arguments:

- help Display this message and exit

nwk_route_delete

Usage: nwk_route_delete all|<NWKADDR>

Remove entries from the network routing table

optional arguments:

-help Display this message and exit

nwk_route_request

Usage: nwk_route_request nwk|ext ADDR [-radius NUM] [-nocache]

Arguments:

nwk - network short address

ext - network extended address

Optional arguments:

radius - radius of the route request

nocache - set no cache option

Send a network route request to the network

optional arguments:

-help Display this message and exit

nwk_send_conflict

usage: nwk_send_conflict ADDR STATUS

Forge a network address conflict command.

positional arguments:

ADDR Network address to report the status of.

STATUS Network status code to report.

optional arguments:

-help Display this message and exit

nwk_status

Usage: nwk_status DSTADDR NWKADDR COMMAND

DSTADDR : Destination of Network Status command

NWKADDR : Target address in Network Status payload.

COMMANDS

conflict : 0x0d - Address Conflict

Send a Network Status command

optional arguments:

-help Display this message and exit

nwk_txpower

usage: nwk_txpower POWER

Adjust the radio transmit Power

positional arguments:

POWER Desired TX Power in dBm, must be within 5dBm of a valid power setting for the radio.

optional arguments:

-help Display this message and exit

nwk_unknown_cmd

usage: nwk_unknown_cmd NWKADDR [VERSION]

Transmit a network command with an unsupported ID or version.

positional arguments:

NWKADDR Destination network address for the command.

VERSION Network protocol version for the command.

optional arguments:

-help Display this message and exit

poll

Usage: poll

syntax:

```
[-init MAXPATH]
[-add PATH -value VALUE -cmd "CLI_COMMAND"]
```

Execute a zbcli command when an events occur on file handle with a matching value

optional arguments:

- i, -init Start monitor and allocate at most MAXPATH unique paths
- a, -add Add a new handler for the value
- v, -value Value that triggers the command
- c, -cmd zbcli command to execute if event on PATH matches VALUE. Note: use quotes if CLI_COMMAND contains spaces
- help Display this message and exit

prompt

usage: prompt PROMPT

Change the command line prompt

positional arguments:

PROMPT Set the command line prompt to PROMPT or use 'disable' to suppress the prompt

optional arguments:

-help Display this message and exit

quit

usage: quit [-leave]

Terminate program and exit

optional arguments:

- leave Leave network before terminating.
- help Display this message and exit

set_ib

Usage: set_ib [<NUMBER or NAME> <VALUE>]

Set IB by hex NUMBER (e.g. 80) or text NAME (e.g. panid) to VALUE
with neither shows a list of available values

Set attribute values in the stack's information base.

optional arguments:

-help Display this message and exit

shell

Usage: shell [EXECUTABLE] ([ARGUMENTS] ...)

Execute commands in the system shell

optional arguments:

-help Display this message and exit

source

usage: source FILE

read commands from FILE until EOF. On EOF input resumes from console

positional arguments:

FILE File to read commands from.

optional arguments:

-help Display this message and exit

startup_channellist

Usage: startup_channellist [clear|<PAGE CHANNEL[...]>|<MACRO>]

ARGUMENTS

clear Clears channel list.

PAGE CHANNEL... Adds a page and channel to the channel list

MACRO Adds a page and channel mask

‘868’ = page 0 channel 0

‘915’ = page 0 channels 1-10

‘2.4’ = page 0 channels 11-26

‘SE’ = page 0 channels 11, 14, 15, 19, 20, 24 or 25

‘gb28’ = page 28, 27 channels

‘gb29’ = page 29, 8 channels

‘gb30’ = page 30, 14 channels

‘gb31’ = page 31, 13 channels

Display and update stacks’s channel set

optional arguments:

-help Display this message and exit

startup_config

Usage: startup_config <ARGS>

ARGUMENTS

display

Display the current settings.

prodefaults

Reset all settings to the PRO defaults, this is done initially so this command is used to undo settings before join/form.

shortaddress <NWKADDR>

Set the startup shortaddress unless 0xFFFF is used.

extendedpanid <EPID>

Set the startup extendedpanid, see startup control below

panid <PANID>

Set the startup panid

stackprofile <PROFILE>

Set the stack profile in hex or one of the named profiles:
custom (0x00), home (0x01) or pro (0x02, default)

commissioning MODE

findbind : enable Finding & Binding

touchlink : enable Touchlink

startupcontrol <TYPE>

TYPE may be hex or preconfigured, form, rejoin, join

If startupcontrol is “preconfigured” (0x00) then “extendedpanid” must also be set

If startupControl is “form (0x01) and “extendedpanid” then the stack will use the WPAN device’s EUJ as the Extended PAN ID.

If startupControl is “rejoin” (0x02), extendedPanId must be provided.

If startupControl is “join” (0x03) and “extendedpanid” is 0 then the stack will try join any available network.

networkmanageraddress <NWKADDR>

Set the network manager address

security <ARGS>

level <LEVEL>

trustCenterAddress <IEEE64>

preconfiguredLinkKey <KEY>

distributedLinkKey <KEY>

cbke <ARGS>

none disable all suites and reset security material

v1 ARGS enable suite 1 and load CBKE Suite 1 security material

priv HEXSTR

capub HEXSTR

cert HEXSTR

v2 ARGS enable suite 2 and load CBKE Suite 2 security material

priv HEXSTR

capub HEXSTR

cert HEXSTR

ka_client Enable Keep Alive Client when joining

ka_server [BASE JITTER] Enable Keep Alive Server when forming.

Optionally configure Keep Alive Server attributes. If BASE is zero, use stack defaults.

join <ARGS>

attempts <NUM> set attempts, default is 3

capability <CAPABILITY> enable|disable

CAPABILITY:

altcoord

fullfunction

mainspower

rxonwhenidle

reserved1

reserved2

security

allocateaddr

end_timeout <VALUE>

Display and update the startup configuration

optional arguments:

-help Display this message and exit

startup_findbind

startup_findbind

Join a network using the startup configuration

optional arguments:

- help Display this message and exit

startup_form

usage: startup_form [JOINTIME]

Form a network using the startup configuration

positional arguments:

JOINTIME Enable permit join for JOINTIME seconds after formation

optional arguments:

-help Display this message and exit

startup_join

startup_join [ATTEMPTS [DELAY]]

ATTEMPTS : number of attempts to join (default = 1)

DELAY : delay in seconds between failed join attempts (default = 3)

Join a network using the startup configuration

optional arguments:

-help Display this message and exit

startup_persist

Usage: startup_persist <COMMAND> <ARGS>

COMMANDS

bdb [FILE] coord|router|end
 coord Form a network on persistence failure.
 router Join as a router on persistence failure.
 end Join as a end device on persistence failure.

startup [FILE] [OPTIONS]
 start the stack from the persistence FILE.

OPTIONS

- disable Use file to load persistence, but don't write to it
- form Form a network if startup from persistence fails
- join Join a network if startup from persistence fails

delete

enable [FILE]
 start the saving the current state after the stack has started

disable
 stops the saving of persistence if it has already been started

snapshot FILE
 performs a one-time save of the current state of a started stack

swapout FILE

swapin FILE
 similar to persist startup, except the IEEE address is retained

Resume network operation from saved persistence file.

optional arguments:

- help Display this message and exit

startup_rejoin

usage: startup_rejoin

Join a network with secure rejoin commands using the startup configuration

optional arguments:

- help Display this message and exit

startup_tc_rejoin

usage: startup_tc_rejoin

Join a network with TC rejoin commands using the startup configuration

optional arguments:

- help Display this message and exit

status

status [LIMIT]

LIMIT : Limit the number of entries to print for each table.

Default: 0 (unlimited)

Display status information

optional arguments:

-help Display this message and exit

status_code

usage: status_code [STATUS]

Describe status codes returned by the ZigBee stack

positional arguments:

STATUS ZigBee Status Code

optional arguments:

-help Display this message and exit

testcase

Usage: testcase <TESTCASE>|disable

Supported test cases:

15.47

15.48

gb868_min_power

drop_rejoin_rsp

version

help: Usage: version

help: display versions of components

wait

usage: wait [-min MINUTES] [-sec SECONDS] [-ms MILLISECONDS]

Wait for the given amount of time. Stack and packet processing will still take place.

optional arguments:

- min MINUTES Time in minutes to wait
- sec SECONDS Time in seconds to wait
- ms MILLISECONDS Time in milliseconds to wait
- help Display this message and exit

wpan

Usage: wpan <TYPE> ...

Create a wpan device.

TYPE

serial PATH [NAME [BAUD]]

PATH Path to the OS device (e.g. /dev/ttyUSB0)

NAME Symbolic device name (default: wpan0)

BAUD Serial port baud rate (default: 115200)

vpan vN

vN defines the pseudo-wireless area the device belongs to.

All devices within the surrounding wireless area can communicate with each other.

(Example: 'v0')

Environment Variables

VPAN_PATH Optional path to vpan manager (default is /tmp/vpan)

Create radio devices

optional arguments:

-help Display this message and exit

zb_leave

usage: zb_leave

Leave the network.

optional arguments:

-help Display this message and exit

zb_reset

Usage: zb_reset

Leave the network.

optional arguments:

-help Display this message and exit

zcl_alarm_client

Usage: zcl_alarm_client SRC_EP <COMMANDS>

COMMANDS:

```
[-init] [-remote_address] [-remote_endpoint]
[-read] [-remote_address] [-remote_endpoint]
[-reset_alarm] [-cluster]
[-reset_all_alarms]
[-get_alarm]
[-reset_alarm_log]
```

ARGUMENTS:

SRC_EP Endpoint for the ZCL device temperature client (Positional)

-i, -init Initialize the client cluster

-r, -read read the value of the remote attribute

-a, -remote_address remote short (network) address to send messages or bind

-e, -remote_endpoint remote endpoint to send messages or bind

-m, -reset_alarm send a Reset Alarm command to the server (via binding) with provided alarm_code (uint8_t); must also specify cluster

-c, -cluster value of cluster to send the reset alarm to (uint16_t)

-l, -reset_all_alarms send a Reset All Alarms command from the client to the server (via binding)

-g, -get_alarm send a Get Alarm command from the client to the server (via binding)

-x, -reset_alarm_log send a Reset Alarm Log command from the client to the server (via binding)

OPTIONAL ARGUMENTS for [-init]:

-a, -remote_address remote short (network) address to send messages or bind

-e, -remote_endpoint remote endpoint to send messages or bind

zcl_alarm_server

Usage: `zcl_alarm_server SRC_EP <COMMANDS>`

COMMANDS:

`[-init] [-alarm_count]`

ARGUMENTS:

`SRC_EP` Endpoint for the ZCL alarm server (Positional)

`-i, -init` initialize the alarm server

`-c, -alarm_count` max number of alarms (`uint16_t`)

zcl_attr_disc

Usage: zcl_attr_disc NWKADDR ENDPT CLUSTER PROFILE START-ATTR
MAX-ATTR [server|client]

Send ZCL discover attribute command

NWKADDR : destination address

ENDPT : destination endpoint

CLUSTER : cluster ID

PROFILE : profile ID (HA = 0x0104, SE = 0x0109)

START-ATTR : starting attribute ID

MAX-ATTR : max attributes

server|client : specify direction (default: server)

Send ZCL discover attribute command

optional arguments:

-help Display this message and exit

zcl_attr_read

Usage: zcl_attr_read CLUSTER ATTR NWKADDR ENDPOINT PROFILE
[server|client]

Send ZCL read attribute command

CLUSTER : cluster ID

ATTR : attribute ID

NWKADDR : destination address

ENDPOINT : destination endpoint

PROFILE : profile ID (HA = 0x0104, SE = 0x0109)

server|client : specify direction (default: server)

Send ZCL read attribute command

optional arguments:

-help Display this message and exit

zcl_attr_report

Usage: zcl_attr_report COMMAND

config NWKADDR ENDPT CLUSTER PROFILE ATTR TYPE MIN MAX
[CHANGE]

read NWKADDR ENDPT CLUSTER PROFILE ATTR

NWKADDR : Destination Address

ENDPT : Destination Endpoint

CLUSTER : Cluster ID

PROFILE : Profile ID (e.g. HA = 0x0104, SE = 0x0109)

ATTR : Attribute ID

TYPE : Attribute Type

MIN : Min interval time (seconds)

MAX : Max interval time (seconds)

CHANGE : Reportable change. Required for analog data types.

Omitted for 'discrete' data types E.g. bit fields.

Set to zero if min = 0xffff and max = 0x0000 (default configuration)

Send ZCL configure or read reporting attribute command

optional arguments:

-help Display this message and exit

zcl_attr_reset

Usage: zcl_attr_reset

resets all clusters attributes

optional arguments:

-help Display this message and exit

zcl_attr_write

Usage: zcl_attr_write CLUSTER ATTR NWKADDR ENDPOINT PROFILE TYPE
DATA [server|client]

Send ZCL write attribute command

CLUSTER : cluster ID

ATTR : attribute ID

NWKADDR : destination address

ENDPOINT : destination endpoint

PROFILE : profile ID (HA = 0x0104, SE = 0x0109)

TYPE : attribute data type (e.g. bool = 0x10; uint8 = 0x20; uint16 = 0x21)

DATA : ascii string representation of value to write. Conversion depends on If spaces,
use quotes.

server|client : specify direction (default: server)

Send ZCL write attribute command

optional arguments:

-help Display this message and exit

zcl_ballast_config_client

Usage: `zcl_ballast_config_client SRC_EP COMMAND`

commands:

- `init`

- `destroy`

zcl_ballast_config_server

Usage: zcl_ballast_config_server SRC_EP COMMAND

commands:

init

destroy

read_attr OPTIONS

write_attr OPTIONS

options:

-attrId ID

-attrVal VALUE

zcl_basic

Usage: zcl_basic COMMAND

COMMANDS

server SRC_EP write alarm-mask MASK : writes
ZCL_BASIC_ATTR_ALARM_MASK

server SRC_EP write model NAME : writes ZCL_BASIC_ATTR_MODEL_NAME

server SRC_EP post-alarm CODE : CODE: = 0 HW, 1 = SW

client SRC_EP init :

client SRC_EP read_test NWKADDR DST_ENDPT : Follows Certification Test

client SRC_EP write_test NWKADDR DST_ENDPT : Follows Certification Test

client SRC_EP reset [NWKADDR DST_ENDPT] : If NWKADDR and
DST_ENDPT are not provided,

command is sent using binding.

ZCL basic cluster

optional arguments:

-help Display this message and exit

`zcl_calendar_client`

Usage: `zcl_calendar_client SRC_EP COMMAND`

COMMANDS:

- `init`
- `read_test`
- `write_test`
- `command_test`

zcl_calendar_server

Usage: zcl_calendar_server SRC_EP COMMAND

COMMANDS:

init

ZCL calendar server cluster

optional arguments:

-help Display this message and exit

zcl_color_client

Usage: zcl_color_client SRC_EP <COMMANDS>

COMMANDS:

- [-init]
- [-read]
- [-write] [-value]
- SUB-COMMANDS (see below)

ARGUMENTS:

- SRC_EP Endpoint for the ZCL Color Client Cluster (Positional)
- i, -init Initialize the client cluster
- read ATTR read the value of the remote attribute
- write ATTR write the remote attribute to the value VALUE
- value VALUE the value used by the set

List of sub-commands (Cluster needs to be initialized first):

Optional Arguments for ALL sub-commands:

- k, -options_mask set the options mask bitmap to 1 and override bitmap to 0
- v, -options_override set the options mask bitmap to 1 and override bitmap to 1

zcl_color_client SRC_EP -command MoveToHue [-hue -h] [-direction -d]
[-transition_time -t] [-options_mask -k] [-options_override -v]

Required Arguments for MoveToHue:

- h, -hue value of hue to move to (uint8_t)
- d, -direction direction of move (specify: shortest, longest, up, down)
- t, -transition_time transition time of command (uint16_t)

zcl_color_client SRC_EP -command MoveHue [-move_mode -m] [-rate -r]
[-options_mask -k] [-options_override -v]

Required Arguments for MoveHue:

- m, -move_mode move mode (specify: up, down, stop)
- r, -rate rate of movement (uint16_t)

zcl_color_client SRC_EP -command StepHue [-step_mode] [-step_size]
[-transition_time -t] [-options_mask -k] [-options_override -v]

Required Arguments:

- step_mode step mode (specify: up, down)
- step_size size of step (uint16_t)

`-t`, `-transition_time` transition time of command (uint16_t)

`zcl_color_client SRC_EP -command MoveToSaturation [-sat -s] [-transition_time -t] [-options_mask -k] [-options_override -v]`

Required Arguments:

`-s`, `-sat` value of saturation to move to (uint8_t)

`-t`, `-transition_time` transition time of command (uint16_t)

`zcl_color_client SRC_EP -command MoveSaturation [-move_mode -m] [-rate -r] [-options_mask -k] [-options_override -v]`

Required Arguments:

`-m`, `-move_mode` move mode (specify: up, down, stop)

`-r`, `-rate` rate of movement (uint16_t)

`zcl_color_client SRC_EP -command StepSaturation [-step_mode] [-step_size] [-transition_time -t] [-options_mask -k] [-options_override -v]`

Required Arguments:

`-step_mode` step mode (specify: up, down)

`-step_size` size of step (uint16_t)

`-t`, `-transition_time` transition time of command (uint16_t)

`zcl_color_client SRC_EP -command MoveToHueSaturation [-hue -h] [-sat -s] [-transition_time -t] [-options_mask -k] [-options_override -v]`

Required Arguments:

`-h`, `-hue` value of hue to move to (uint8_t)

`-s`, `-sat` value of saturation to move to (uint8_t)

`-t`, `-transition_time` transition time of command (uint16_t)

`zcl_color_client SRC_EP -command MoveToColor [-color_x -x] [-color_y -y] [-transition_time -t] [-options_mask -k] [-options_override -v]`

Required Arguments:

`-x`, `-color_x` value of color x to move to (uint16_t)

`-y`, `-color_y` value of color y to move to (uint16_t)

`-t`, `-transition_time` transition time of command (uint16_t)

`zcl_color_client SRC_EP -command MoveColor [-rate_x] [-rate_y] [-options_mask -k] [-options_override -v]`

Required Arguments:

`-rate_x` value of rate x to move to (uint16_t)

`-rate_y` value of rate y to move to (uint16_t)

zcl_color_client SRC_EP -command StepColor [-step_x] [-step_y] [-transition_time -t] [-options_mask -k] [-options_override -v]

Required Arguments:

- step_x value of step x to move to (uint16_t)
- step_y value of step y to move to (uint16_t)
- t, -transition_time transition time of command (uint16_t)

zcl_color_client SRC_EP -command MoveToColorTemperature [-temp] [-transition_time -t] [-options_mask -k] [-options_override -v]

Required Arguments:

- temp value of temp to move to (uint16_t)
- t, -transition_time transition time of command (uint16_t)

zcl_color_client SRC_EP -command EnhancedMoveToHue [-enhanced_hue] [-direction -d] [-transition_time -t] [-options_mask -k] [-options_override -v]

Required Arguments:

- enhanced_hue value of enhanced hue to move to (uint16_t)
- d, -direction direction of move (specify: shortest, longest, up, down)
- t, -transition_time transition time of command (uint16_t)

zcl_color_client SRC_EP -command EnhancedMoveHue [-move_mode -m] [-rate -r] [-options_mask -k] [-options_override -v]

Required Arguments:

- m, -move_mode move mode (specify: up, down, stop)
- r, -rate rate of movement (uint16_t)

zcl_color_client SRC_EP -command EnhancedStepHue [-step_mode] [-step_size] [-transition_time -t] [-options_mask -k] [-options_override -v]

Required Arguments:

- step_mode step mode (specify: up, down)
- step_size size of step (uint16_t)
- t, -transition_time transition time of command (uint16_t)

zcl_color_client SRC_EP -command EnhancedMoveToHueSaturation [-enhanced_hue] [-sat -s] [-transition_time -t] [-options_mask -k] [-options_override -v]

Required Arguments:

- enhanced_hue value of enhanced hue to move to (uint16_t)
- s, -sat value of saturation to move to (uint8_t)
- t, -transition_time transition time of command (uint16_t)

```
zcl_color_client SRC_EP -command ColorLoopSet [-update_flags] [-update_action]
[-update_direction] [-update_time] [-update_start_hue] [-action -a] [-loop_dir]
[-transition_time -t] [-start_hue] [-options_mask -k] [-options_override -v]
```

Required Arguments:

- a, -action value of action (specify: deactivate, activateField, activateAttr)
- loop_dir value of direction to move to (specify: decrement, increment)
- t, -transition_time transition time of command (uint16_t)
- start_hue value of start hue of command (uint16_t)
- update_flags value of the update flag bitmap, alternatively you can use the combination of the flags below (uint8_t)

One of more of the flags:

- update_action set the update action flag bit
- update_direction set the update direction flag bit
- update_time set the update time flag bit
- update_start_hue set the update start hue flag bit

```
zcl_color_client SRC_EP -command StopMoveStep [-options_mask -k]
[-options_override -v]
```

```
zcl_color_client SRC_EP -command MoveColorTemperature [-move_mode -m] [-rate -r]
[-min_mireds] [-max_mireds] [-options_mask -k] [-options_override -v]
```

- m, -move_mode move mode (specify: up, down, stop)
- r, -rate rate of movement (uint16_t)
- min_mireds value of minimum temperature mireds (uint16_t)
- max_mireds value of maximum temperature mireds (uint16_t)

```
zcl_color_client SRC_EP -command StepColorTemperature [-step_mode] [-step_size]
[-transition_time -t] [-min_mireds] [-max_mireds] [-options_mask -k] [-options_override
-v]
```

Required Arguments:

- step_mode step mode (specify: up, down)
- step_size size of step (uint16_t)
- t, -transition_time transition time of command (uint16_t)
- min_mireds value of minimum temperature mireds (uint16_t)
- max_mireds value of maximum temperature mireds (uint16_t)

zcl_color_server

Usage: zcl_color_server <COMMANDS>

commands:

init <ENDPOINT>

optional arguments:

- e -enhanced enable enhanced mode (during server allocation)
- h -hue enable hue and saturation support (during server allocation)
- x -xy enable xy support (during server allocation)
- t -temp enable temperature support (during server allocation)
- l -loop enable color loop support (during server allocation)
- a -all enable all color mode support (during server allocation)

`zcl_device_mgmt_client`

Usage: `zcl_device_mgmt_client SRC_EP COMMAND`

COMMANDS:

- `init`
- `read_test`
- `write_test`

ZCL Device Management Client Cluster

optional arguments:

- `-help` Display this message and exit

zcl_device_mgmt_server

Usage: zcl_device_mgmt_server SRC_EP COMMAND

COMMANDS:

init

ZCL Device Management Server Cluster

optional arguments:

-help Display this message and exit

zcl_device_temp_client

Usage: zcl_device_temp_client SRC_EP <COMMANDS?

COMMANDS:

[-init]

[-write] [-value] [-remote_address] [-remote_endpoint]

[-read] [-remote_address] [-remote_endpoint]

ARGUMENTS:

SRC_EP Endpoint for the ZCL device temperature client (Positional)

-i, -init Initialize the client cluster

-r, -read read the value of the remote attribute

-w, -write write the remote attribute to the value VALUE

-v, -value the value used by the set

-a, -remote_address remote short (network) address to send messages

-e, -remote_endpoint remote endpoint to send messages

zcl_device_temp_server

Usage: zcl_device_temp_server SRC_EP <COMMANDS>

COMMANDS:

[-init]

[-temp_update]

ARGUMENTS:

SRC_EP Endpoint for the ZCL alarm server (Positional)

-i, -init initialize the alarm server

-i, -temp_update update server cluster with a temperature sensor reading (int16, range of -200 to 200)

zcl_doorlock

Usage: zcl_doorlock <COMMANDS>

server init <ENDPOINT>

client init <ENDPOINT>

client read <NWKADDR>

client lock <NWKADDR>

client unlock <NWKADDR>

client toggle <NWKADDR>

`zcl_drlc_client`

Usage: `drlc_client SRC_EP COMMAND`

COMMANDS:

- `init`

- `get_events`

zcl_drlc_server

Usage: drlc_server SRC_EP COMMAND

COMMANDS:

init

new ID DEVICE CRITICALITY DURATION [START]

where

DEVICE

hvac

heater

water-heater

pool-pump

appliance

irrigation

commercial

residential

exterior-light

interior-light

vehicle

generator

CRITICALITY

green

emergency

outage

disconnect

cancel ID

cancelall

list

reset

read NAME

write NAME VALUE

ZCL DRLC Server Cluster

optional arguments:

-help Display this message and exit

zcl_endpoint_remove

Usage: zcl_endpoint_remove ENDPOINT

Remove an Endpoint

optional arguments:

- help Display this message and exit

zcl_events_client

Usage: zcl_events_client SRC_EP COMMAND

COMMANDS:

init

ZCL Events Client Cluster

optional arguments:

-help Display this message and exit

zcl_events_server

Usage: zcl_events_server SRC_EP COMMAND

COMMANDS:

- init
- command_test

ZCL Events Server Cluster

optional arguments:

- help Display this message and exit

zcl_fan

Usage: zcl_fan:

server init <ENDPOINT>

client init <ENDPOINT>

client read <NWKADDR> <attrIds..>

client write <NWKADDR> <attrId> <value>

zcl_groups_client

Usage: zcl_groups_client SRC_EP COMMAND

COMMANDS:

```
init
read ATTR NWKADDR ENDPT
write ATTR NWKADDR ENDPT TYPE VALUE
add NWKADDR ENDPT GROUP
view NWKADDR ENDPT GROUP
get_membership NWKADDR ENDPT [GROUP]
remove NWKADDR ENDPT GROUP
remove_all NWKADDR ENDPT
add_identify NWKADDR ENDPT GROUP
```

ZCL Groups Client Cluster

optional arguments:

```
-help Display this message and exit
```


`zcl_groups_server`

Usage: `zcl_groups_server SRC_EP COMMAND`

`init`

`destroy`

ZCL Groups Server Cluster

optional arguments:

`-help` Display this message and exit

zcl_ias_ace_device

Usage: zcl_ias_ace_device SRC_EP COMMAND

init ZONE-TYPE MANUF-CODE [trip] : Allocates the ACE-client and Zone-server clusters

 ZONE-TYPE : 2-octet. Sets ZoneState server attribute

 MANUF-CODE : 2-octet

 trip : use trip-to-pair enrollment

destroy : Removes and frees the endpoint and clusters

bind CIE_ADDR ENDPT :

ace COMMAND

 arm NWKADDR ENDPT MODE CODE ZONE :

 bypass NWKADDR ENDPT CODE ZONE :

 emergency NWKADDR ENDPT :

 fire NWKADDR ENDPT :

 panic NWKADDR ENDPT :

 get_zone_id_map NWKADDR ENDPT :

 get_zone_info NWKADDR ENDPT ZONE :

 get_panel_status NWKADDR ENDPT :

 get_zone_bypassed NWKADDR ENDPT :

zone COMMAND

 trip-enroll : Trigger Trip-to-pair

 change-status ZONE-STATUS : Change the zone's status

ZCL IAS ACE Device Cluster

optional arguments:

 -help Display this message and exit

zcl_ias_cie_device

Usage: zcl_ias_cie_device SRC_EP COMMAND

init : Creates the endpoint and clusters

destroy : Removes and frees the endpoint and clusters

ace COMMAND

code [ARM_CODE] : Configures the arm/disarm code

arm stay|night|away : Arm the system

disarm : Disarm the system

bypass on|off|not ZONE : Control the bypass of a zone

on : Enable bypass of a zone

off : Disable bypass of a zone

not : Set the zone so it's not allowed to be bypassed

zone COMMAND

server-enroll EXTADDR ENDPT

send-enroll-rsp EXTADDR ENDPT ZONE-TYPE

check-enroll EXTADDR ENDPT

delete-addr EXTADDR

normal-mode EXTADDR ENDPT

test-mode EXTADDR ENDPT SECONDS

warn COMMAND

start OPTIONS : Sends Start Warning command to binding.

-mode MODE : stop, burglar, fire, emerg, police-panic, fire-panic, emerg-panic.

Default: stop

-strobe : Enable strobe

-siren-level LEVEL : low, medium, high or very-high. Default: low

-duration SECS : Default: 0

-duty-cycle [0,100] : Default: 0

-strobe-level LEVEL : low, medium, high or very-high. Default: low

squawk OPTIONS : Sends Squawk command to binding.

-mode MODE : armed, disarmed. Default: armed

-strobe : Enable strobe

-squawk-level LEVEL : low, medium, high or very-high. Default: low

ZCL IAS CIE Device Cluster

optional arguments:

-help Display this message and exit

zcl_ias_warn_device

Usage: zcl_ias_warn_device SRC_EP COMMAND

init ZONE-TYPE MANUF-CODE [trip] : Creates the endpoint and clusters

ZONE-TYPE : 2-octet. Sets ZoneState server attribute

MANUF-CODE : 2-octet

trip : use trip-to-pair enrollment

destroy : Removes and frees the endpoint and clusters

zone COMMAND

trip-enroll : Trigger Trip-to-pair

change-status ZONE-STATUS : Change the zone's status

ZCL IAS Warning Device Cluster

optional arguments:

-help Display this message and exit

zcl_ias_zone_device

Usage: zcl_ias_zone_device SRC_EP COMMAND

init ZONE-TYPE MANUF-CODE [trip] : Creates the endpoint and clusters

ZONE-TYPE : 2-octet. Sets ZoneState server attribute

MANUF-CODE : 2-octet

trip : use trip-to-pair enrollment

destroy : Removes and frees the endpoint and clusters

zone COMMAND

trip-enroll : Trigger Trip-to-pair

change-status ZONE-STATUS : Change the zone's status

test on|off [DURATION] : Enable / Disable mode

DURATION : Only applies to 'on'

ZCL IAS Zone Device Cluster

optional arguments:

-help Display this message and exit

zcl_identify_client

Usage: zcl_identify_client SRC_EP COMMAND

init : Allocate the cluster

destroy : Frees the cluster

identify NWKADDR ENDPT SECONDS : Send IDENTIFY command to server

query NWKADDR ENDPT : Send QUERY command to server

ZCL Identify Client Cluster

optional arguments:

-help Display this message and exit

zcl_identify_server

Usage: `zcl_identify_server SRC_EP COMMAND`

`init` : Allocates the cluster

`destroy` : Frees the cluster

`identify [SECONDS]` : Put cluster into Identify mode. If SECONDS is not provided, default is 10 seconds.

ZCL Identify Server Cluster

optional arguments:

`-help` Display this message and exit

zcl_illum_meas_client

Usage: zcl_illum_meas_client SRC_EP COMMAND

commands:

[-init]

[-destroy]

arguments:

-c, -init Initialize the client cluster

-d, -destroy destroy the client cluster

-e, -ep required with all commands

zcl_illum_meas_server

Usage: zcl_illum_meas_server SRC_EP COMMAND

commands:

- [-init]
- [-read_attr]
- [-write_meas] [value]
- [-destroy]

arguments:

- s, -init Initialize the server cluster
- r, -read_attr Read and display all the server attributes
- w VALUE Write illuminance measured attribute
- e, -ep required with all commands
- d, -destroy destroy the server cluster

zcl_level_client

Usage: zcl_level_client <COMMANDS>

commands:

```
init <ENDPOINT> [PROFILEID]
move_level [-ep] [-nwkaddr] [-level] [-time] [-onoff] [-mask] [-override]
move [-ep] [-nwkaddr] [-mode] [-rate] [-onoff] [-mask] [-override]
step [-ep] [-nwkaddr] [-mode] [-size] [-time] [-onoff] [-mask] [-override]
stop [-ep] [-nwkaddr] [-onoff]
```

arguments:

- e, -ep endpoint
- n, -nwkaddr destination network address
- l, -level level
- r, -rate rate
- t, -time transition time
- m, -mode move mode (0 up, 1 down)
- s, -size step size
- o, -onoff send command with onOff command
- a, -mask set the options mask bit to true
- v, -override set the options override bitmap to have the ExecuteIfOff bit set

zcl_level_server

Usage: zcl_level_server <COMMANDS>

commands:

 init <ENDPOINT> [PROFILEID]

arguments:

zcl_messaging_client

Usage: zcl_messaging_client SRC_EP COMMAND

Messaging cluster client commands where COMMAND is one of:

init

getlast

Request the server to send us DisplayMessage

confirm [ID] [ANSWER] [MESSAGE]

Send user confirmation

ID (optional) id of the message being confirmed

if omitted the last received message id received is used

ANSWER (optional) either “yes” or “no”

MESSAGE (optional) is a text string response

if a MESSAGE is provided, answer must also be provided

getcancel [TIME]

Request server to send us CancelAllMessages

TIME is the earliest implementation time

if omitted the current time is used

show [PIN]

Show the local client current message

if message is protected PIN must be correct or message will not be shown

may be omitted when the current message is not protected

ZCL Messaging Client Cluster

optional arguments:

-help Display this message and exit

zcl_messaging_server

Usage: messaging_server SRC_EP COMMAND

COMMANDS:

init

new ID MESSAGE TRANSMISSION IMPORTANCE CONFIRM DURATION
[START]

where

TRANSMISSION

normal

both

interpan

IMPORTANCE

low

medium

high

critical

CONFIRM 0|1

DURATION minutes

START relative time from now in seconds

show : display current message on local server.

clear : remove current message from local server.

cancel ID : send a CANCEL command to binding.

cancelall : send a CANCEL_ALL command to binding

ZCL Messaging Server Cluster

optional arguments:

-help Display this message and exit

zcl_metering_client

usage: zcl_metering_client ENDPT COMMAND [-e ENDPT] [-n NWKADDR]
[-attr_name ATTR_NAME] [-attr_value ATTR_VALUE]

ZCL Metering Client Cluster

positional arguments:

ENDPT Source endpoint

COMMAND commands: init, write

optional arguments:

-e, -dst-ep ENDPT Destination endpoint

-n, -nwkaddr NWKADDR nwkaddr

-attr_name ATTR_NAME attribute name

-attr_value ATTR_VALUE attribute value

-help Display this message and exit

zcl_metering_server

usage: zcl_metering_server ENDPT COMMAND

ZCL Metering Server Cluster

positional arguments:

ENDPT Source endpoint

COMMAND commands: init, write

optional arguments:

-help Display this message and exit

zcl_occupancy

Usage: zcl_occupancy <COMMANDS>

server init <ENDPOINT>

client init <ENDPOINT>

client read <NWKADDR>

client write <NWKADDR>

zcl_onoff_client

Usage: zcl_onoff_client SRC_EP COMMAND
init
destroy
send on|off|toggle [NWKADDR ENDPOINT]

ZCL OnOff Client Cluster

optional arguments:

-help Display this message and exit

`zcl_onoff_server`

Usage: `zcl_onoff_server SRC_EP COMMAND`

`init`

`destroy`

ZCL OnOff Server Cluster

optional arguments:

`-help` Display this message and exit

zcl_ota_client

Usage: zcl_ota_client <COMMANDS>

COMMANDS:

[-init] [-filename] [-endpoint] [-image_block_delay]
[-discover]
[-query_next_image] [-manufacturer_code] [-image_type][-file_version] [-wait]
[-image_block_start]

ARGUMENTS:

-c, -init Initialize the client cluster
-f, -filename FILE filename for the extracted image
-e, -endpoint EP endpoint to use
-p, -client_image_block_delay client config to reset the delay between image block requests to more than 50ms
-q, -query_next_image check server for firmware updates
-m, -manufacturer_code MFG manufacturer code to use for request
-i, -image_type IT image_type to use for request
-v, -file_version CV current or requested firmware version
-w, -wait check_for_update and wait for result
-l, -image_block_start download image from server with multiple Image Block requests

zcl_ota_server

Usage: zcl_ota_server <COMMANDS>

COMMANDS:

```
[-init] [-endpoint] [-end_current_time] [-end_upgrade_time]
[-register]
[-show]
[-deregister]
[-image_notify] [-nwk_addr] [-endpoint] [-query_jitter] [-manufacturer_code]
[-image_type] [-file_version]
```

ARGUMENTS:

```
-s, -init initialize the OTA server
-r, -read read client attribute, from server
-e, -endpoint EP endpoint to use
-n, -nwk_addr NA nwk address of client, used for read attributes
-v, -file_version CV current or requested firmware version
-h, -hardware_version HW (optional) hardware version of device
-m, -manufacturer_code MFG manufacturer code to use for request
-i, -image_type IT image_type to use for request
-g, -register FN specify a file to register as an image
-x, -deregister remove registered image
-b, -show show currently loaded image
-d, -server_image_block_delay (optional) standalone server command to trigger a
WAIT_FOR_DATA response on the server
-c, -end_current_time (optional) value of Current Time to use in Upgrade End
Response
-u, -end_upgrade_time (optional) value of Upgrade Time to use in Upgrade End
Response
-y, -image_notify send an image notify command
-j, -query_jitter value of query jitter for image notify command
```

zcl_power

Usage: zcl_power <COMMANDS>

server init <ENDPOINT>

client init <ENDPOINT>

client read <NWKADDR>

client write <NWKADDR>

`zcl_prepay_client`

Usage: `zcl_prepay_client SRC_EP COMMAND`

COMMANDS:

- `init`
- `read_test`
- `write_test`
- `command_test`

ZCL Prepay Client Cluster

optional arguments:

- `-help` Display this message and exit

zcl_prepay_server

Usage: `zcl_prepay_server SRC_EP COMMAND`

COMMANDS:

`init`

zcl_price_client

Usage: zcl_price_client SRC_EP COMMAND

COMMANDS:

- init
- read_test [NWKADDR ENDPOINT]
- write_test [NWKADDR ENDPOINT]
- command_test [NWKADDR ENDPOINT]

ZCL Price Client Cluster

optional arguments:

- help Display this message and exit

zcl_price_server

Usage: zcl_price_server SRC_EP COMMAND

COMMANDS:

init

ZCL Price Server Cluster

optional arguments:

-help Display this message and exit

`zcl_sandbox_client`

Usage: `zcl_sandbox_client SRC_EP COMMAND`

COMMANDS:

`init`

`read_test [NWKADDR ENDPOINT]`

`write_test [NWKADDR ENDPOINT]`

`command_test [NWKADDR ENDPOINT]`

ZCL Sandbox Client Cluster

optional arguments:

`-help` Display this message and exit

zcl_sandbox_server

Usage: zcl_sandbox_server SRC_EP COMMAND

COMMANDS:

- init
- destroy

ZCL Sandbox Server Cluster

optional arguments:

- help Display this message and exit

zcl_scenes_client

Usage: zcl_scenes_client SRC_EP COMMAND

Note: commands are send to bindings, unless DSTADDR ENDPT are provided.

init : Allocates the cluster

destroy : Frees the cluster

read_test : Follows Certification Test

write_test : Follows Certification Test

add GROUP SCENE TRANSITION NAME [EXTENSION]
TRANSITION : seconds

enh-add GROUP SCENE TRANSITION NAME [EXTENSION]
TRANSITION : tenth of a second

view GROUP SCENE

enh-view GROUP SCENE

remove GROUP SCENE

remove-all GROUP

store GROUP SCENE [DSTADDR ENDPT]

recall GROUP SCENE [TRANSITION]
TRANSITION : tenth of a second

get-membership GROUP

copy GROUP_FROM SCENE_FROM GROUP_TO SCENE_TO

copy-all GROUP_FROM GROUP_TO

ZCL Scenes Client Cluster

optional arguments:

-help Display this message and exit

`zcl_scenes_server`

Usage: `zcl_scenes_server SRC_EP COMMAND`

init : Allocates the cluster

destroy : Frees the cluster

ZCL Scenes Server Cluster

optional arguments:

-help Display this message and exit

zcl_seclevel

Usage: zcl_seclevel nwk|link

Configure the default ZCL SE minimum security level for incoming ZCL messages.

optional arguments:

- help Display this message and exit

zcl_temperature_server

Usage: zcl_temperature_server SRC_ENDPT COMMAND

init

values : read values of Temperature Measurement attributes

measured VALUE : value needs to be between minMeasuredValue and maxMeasuredValue

zcl_therm

Usage: zcl_therm:

```
server init <ENDPOINT>  
client init <ENDPOINT>  
client read <NWKADDR> <attrIds..>  
client write <NWKADDR> <attrId> <value>  
client cmd <NWKADDR> <attrId> <value>
```

zcl_time_client

Usage: zcl_time_client SRC_EP COMMAND

init : Allocates the cluster

destroy : Frees the cluster

sync : Discover remote master Time Server and synchronize local Time Server.

read_test NWKADDR DST_ENDPT : Follows Certification Test

write_test NWKADDR DST_ENDPT : Follows Certification Test

ZCL Time Client Cluster

optional arguments:

-help Display this message and exit

zcl_time_server

Usage: `zcl_time_server SRC_EP COMMAND`

init : Allocates the cluster

destroy : Frees the cluster

ZCL Time Server Cluster

optional arguments:

-help Display this message and exit

zcl_touchlink

Usage: zcl_touchlink COMMAND

COMMANDS

config ENDPT [OPTIONS]

-target : Sets ZCL_TL_FLAGS_IS_TARGET and
ZCL_TL_ZBINFO_TYPE_ROUTER

-ed : Sets ZCL_TL_ZBINFO_TYPE_END_DEVICE and clears
ZCL_TL_ZBINFO_TYPE_ROUTER

-persist FILE : Sets ZCL_TL_FLAGS_USE_PERSIST and defines the file
path to

load/save persistent data from.

start : Start Touchlink

reset : Reset Touchlink

ZCL Touchlink Initiator and Target Cluster Commands

optional arguments:

-help Display this message and exit

zcl_tp2

Usage: zcl_tp2 COMMAND

COMMAND:

init [PROFILE] Create the TP2 endpoints (0x01, 0xA0, 0xF0)

buffer DSTMODE DSTADDR LENGTH [PROFILE] [-aps_sec]

DSTMODE:

nwk short address mode

grp group address mode

none use binding table (DSTADDR is ignored, set it to 0)

-aps_sec Use APS encryption

-noack Disable APS ACK (for unicast, already disabled for bcst).

counted NWKADDR LENGTH NUM_PKTS DELAY_MS [OPTIONS]

-frag allow the packet to be fragmented.

-radius NUM configure the radius.

-noack Disable APS ACK (for unicast, already disabled for bcst).

-nonblock Run in the background, don't block command.

reset NWKADDR

retrieve NWKADDR

status

Note: a 'counted' packet length of 79 (not including FCS), results in a PPDU length of 127 bytes (max per frame).

ZCL Test Profile 2 Commands

optional arguments:

-help Display this message and exit

`zcl_tunnel_client`

Usage: `zcl_tunnel_client SRC_EP COMMAND`

COMMAND:

`init` PROTOCOL MANUF MTU

`connect` EUI64

`close`

`send` LENGTH

zcl_tunnel_server

Usage: zcl_tunnel_server SRC_EP COMMAND

COMMAND:

init

destroy

add PROTOCOL MANUF MTU

timeout TIMEOUT

zcl_water_content_meas_client

Usage: zcl_water_content_meas_client SRC_EP CLUSTER_ID COMMAND

commands:

[-init]

[-destroy]

[-clustId] [ID]

arguments:

-c, -init Initialize the client cluster

-d, -destroy destroy the client cluster

-e, -ep required with all commands

-l ID Cluster ID value.

zcl_water_content_meas_server

Usage: zcl_water_meas_server SRC_EP CLUSTER_ID COMMAND

commands:

- [-init]
- [-read_attr]
- [-write_meas_value] [value]
- [-clustId] [ID]
- [-destroy]

arguments:

- s, -init Initialize the server cluster
- r, -read_attr Read and display all the server attributes
- w VALUE Write Water content measured attribute
- l ID Cluster ID value.
- e, -ep required with all commands
- d, -destroy destroy the server cluster

zcl_window_client

Usage: zcl_window_client <COMMANDS>

init <ENDPOINT>

up <NWKADDR>

down <NWKADDR>

stop <NWKADDR>

zcl_window_server

Usage: `zcl_window_server <COMMANDS>`
 `init <ENDPOINT>`
 `mode`

`zdo_active_ep_req`

usage: `zdo_active_ep_req` DSTADDR NWKADDR

Send a ZDO Active_EP_req command to a remote device.

positional arguments:

DSTADDR Destination network address

NWKADDR Network address of interest

optional arguments:

-help Display this message and exit

zdo_bind_req

Usage: zdo_bind_req <SRC_IEEE64> <SRC_EP> <CLUSTER> [ext
<DST_IEEE64> <DST_EP>] | [group <DST_GROUP>] [-bcast]

Either of 'ext' or 'group' addressing must be defined.

ZDO Bind Request

optional arguments:

-help Display this message and exit

`zdo_complex_desc_req`

usage: `zdo_complex_desc_req` NWKADDR

Send a ZDO Complex_desc_req to a remote device.

positional arguments:

NWKADDR Destination network address

optional arguments:

`-help` Display this message and exit

zdo_complex_desc_set

usage: `zdo_complex_desc_set [-test]`

Configure or clear the local device's complex descriptor

optional arguments:

- test configure descriptor as described in TP/ZDO/BV-10
- help Display this message and exit

zdo_device_annce

usage: zdo_device_annce [-c NWKADDR] [-n NWKADDR] [-e EXTADDR] [-a CAPABILITY]

Send a ZDO device announcement

optional arguments:

- c, -change NWKADDR Change local address to NWKADDR prior to announcing
- n, -nwk NWKADDR Network address to announce
- e, -ext EXTADDR Extended address to announce
- a, -alias CAPABILITY Send an aliased announcement with the given CAPABILITY flags
- help Display this message and exit

zdo_discovery_cache_req

usage: `zdo_discovery_cache_req` NWKADDR

Send a ZDO Discovery Cache request (Deprecated command - for GU use only)

positional arguments:

NWKADDR Destination network address for the request

optional arguments:

`-help` Display this message and exit

`zdo_end_bind_req`

Usage: `zdo_end_bind_req` <SRC_EP> <PROFILE> in <CLUSTER_ID...> out
<CLUSTER_ID...>

Send a ZDO End_Device_Bind_req to the coordinator

optional arguments:

-help Display this message and exit

zdo_ieee_addr_req

usage: `zdo_ieee_addr_req DSTADDR NWKADDR [START_INDEX]`

Send a ZDO IEEE_Addr_req to a remote device

positional arguments:

DSTADDR Destination address for the command

NWKADDR Network address of interest

START_INDEX Start index for extended request types

optional arguments:

-help Display this message and exit

zdo_match_bind

Usage: zdo_match_bind SRC_EP CLUSTER PROFILE client|server

Perform match descriptor request and perform binding on first discovered endpoint.

optional arguments:

- help Display this message and exit

zdo_match_desc_req

Usage: zdo_match_desc_req DSTADDR NWKADDR PROFILE [in <CLUSTER_ID...>]
[out <CLUSTER_ID...>]

DSTADDR : Destination address for the command

NWKADDR : Network address of interest

PROFILE : Profile ID

in / out : At least one 'in' or 'out' cluster must be defined.

Send a ZDO Match_desc_req to a remote device

optional arguments:

-help Display this message and exit

`zdo_mgmt_bind_req`

usage: `zdo_mgmt_bind_req` NWKADDR [START_INDEX]

Send a ZDO Mgmt_Bind_req to a remote device

positional arguments:

NWKADDR Destination network address

START_INDEX Specify a start index

optional arguments:

-help Display this message and exit

zdo_mgmt_leave_req

usage: `zdo_mgmt_leave_req` [-rejoin] [-remove] NWKADDR EXTADDR

Send a ZDO Mgmt_Leave_req to a remote device

positional arguments:

NWKADDR Destination network address

EXTADDR Extended address of device requested to leave

optional arguments:

-rejoin Sets the 'rejoin' flag

-remove Sets the 'remove children' flag

-help Display this message and exit

`zdo_mgmt_lqi_req`

usage: `zdo_mgmt_lqi_req` NWKADDR [START_INDEX]

Send a ZDO Mgmt_Lqi_req to a remote device

positional arguments:

NWKADDR Destination network address

START_INDEX Start index for extended request types

optional arguments:

-help Display this message and exit

zdo_mgmt_nwk_enh_update_req

Usage: zdo_mgmt_nwk_enh_update_req <TYPE> <ARGS>

COMMANDS

scan NWKADDR PAGE:MASK DURATION

channel_switch NWKADDR UPDATE-ID PAGE:CHANNEL [PAGE:CHANNEL...]

if NWKADDR is our own address we silently change (naughty)

if NWKADDR is 0xFFFF we broadcast the change (nice)

channel_mask NWKADDR PAGE:MASK [PAGE:MASK...]

Send a ZDO Mgmt_Nwk_Enh_update_req to a remote device

optional arguments:

-help Display this message and exit

zdo_mgmt_nwk_joining_list

Usage: zdo_mgmt_nwk_joining_list <COMMAND>

COMMANDS

bcast ; broadcast a ZDO NWK_Joining_List_rsp

req [<START_INDEX>] ; Send a ZDO NWK_Joining_List_req to the coordinator (0x0000)

Send ZDO Mgmt_Nwk_Joininglist message to remote devices

optional arguments:

-help Display this message and exit

zdo_mgmt_nwk_update_req

Usage: zdo_mgmt_nwk_update_req COMMAND

COMMANDS

scan NWKADDR MASK DURATION

channel_switch NWKADDR UPDATE-ID CHANNEL

channel_mask NWKADDR MASK

if NWKADDR is our own address we silently change (naughty)

if NWKADDR is 0xFFFF we broadcast the change (nice)

Frequency agility commands using the ZDO Mgmt_Nwk_Update_req message

optional arguments:

-help Display this message and exit

`zdo_mgmt_permit_joining_req`

usage: `zdo_mgmt_permit_joining_req` DURATION [NWKADDR]

Send a ZDO Mgmt_Permit_Joining_req to a remote device

positional arguments:

DURATION Permit join duration in seconds

NWKADDR Destination network address for the request

optional arguments:

-help Display this message and exit

zdo_mgmt_unsolicit_enh_update

usage: `zdo_mgmt_unsolicit_enh_update` NWKADDR

Send a ZDO MGMT NWK Unsolicited Enhanced Update Notify)

positional arguments:

NWKADDR Destination network address for the request

optional arguments:

-help Display this message and exit

`zdo_node_desc_req`

usage: `zdo_node_desc_req` [`-no-security`] NWKADDR

Send a ZDO Node_desc_req to a remote device

positional arguments:

NWKADDR Destination network address for the request

optional arguments:

`-no-security` Sends the Node_desc_req without security.

`-help` Display this message and exit

zdo_not_supported_req

usage: `zdo_not_supported_req` NWKADDR

Sends an unsupported ZDP command (0x00ff) to a remote device

positional arguments:

NWKADDR Destination network address for the request

optional arguments:

-help Display this message and exit

`zdo_nwk_addr_req`

usage: `zdo_nwk_addr_req` [-no-security] DSTADDR EXTADDR [START_INDEX]

Send a ZDO Nwk_Addr_req to a remote device

positional arguments:

DSTADDR Destination address for the command (typically 0xfffd broadcast)

EXTADDR The extended address to search for on the network

START_INDEX Start index for extended request types

optional arguments:

-no-security Sends a ZDO NWK_Addr_req without security.

-help Display this message and exit

zdo_power_desc_req

usage: `zdo_power_desc_req` NWKADDR

Send a ZDO Power_desc_req to a remote device

positional arguments:

NWKADDR Destination network address for the request

optional arguments:

-help Display this message and exit

`zdo_simple_desc_req`

usage: `zdo_simple_desc_req` DSTADDR NWKADDR ENDPT

Send a ZDO Simple_desc_req to a remote device

positional arguments:

DSTADDR Destination network address

NWKADDR Network address of interest

ENDPT Remote endpoint of interest

optional arguments:

-help Display this message and exit

zdo_system_server_disc_req

usage: zdo_system_server_disc_req MASK

Broadcast a ZDO System_Server_discovery_req to the network.

positional arguments:

MASK Bitmask of server types to discover

optional arguments:

-help Display this message and exit

zdo_unbind_req

Usage: zdo_unbind_req <SRC_IEEE64> <SRC_EP> <CLUSTER> [ext
<DST_IEEE64> <DST_EP>] | [group <DST_GROUP>] [-bcast]

Either of 'ext' or 'group' addressing must be defined.

ZDO Unbind Request

optional arguments:

-help Display this message and exit

zdo_user_desc_req

usage: `zdo_user_desc_req` NWKADDR

Send a ZDO User_desc_req to a remote device

positional arguments:

NWKADDR Destination network address for the request

optional arguments:

-help Display this message and exit

`zdo_user_desc_set`

usage: `zdo_user_desc_set` [-test] [-string STRING] NWKADDR

Send a ZDO User_desc_set to a remote device

positional arguments:

NWKADDR Destination network address for the request

optional arguments:

- test Sets descriptor to {0x11, 0x22, 0x33, 0x44, 0x55, 0x66, 0x77, 0x88}
- string STRING Sets descriptor to STRING, can be up to 16 characters
- help Display this message and exit

zgp_commands

Usage: zgp_command
add

Send a ZGP GP Command

optional arguments:

- c, -command CMD Greenpower command ID
- n, -nwk_addr NWK destination address
- o, -options options bitmap
- gpid GPD ID
- e, -endpoint Endpoint
- i, -index Index
- g, -groupid Group ID
- d, -deviceid Device ID
- f, -frame_counter GPD security frame counter
- k, -key GPD key
- a, -alias assigned alias
- r, -radius Forwarding Radius
- help Display this message and exit

zgp_pairing

Usage: zgp_pairing
add

Send a ZGP GP Pairing Command

optional arguments:

-help Display this message and exit