

HOPERF Wi SUN Node Operation Guide

Table of Contents

1 Description	3
1.1 Related Pin Introduction	3
1.2 Serial Port Tool Installation and Usage	4
2 Command Description	6
2.1 ROUTER Related Command Description	7
2.1.1 reboot	7
2.1.2 wisun_config	7
2.1.3 network_state	8
2.1.4 set_network_name	9
2.1.5 get_network_name	9
2.1.6 set_mac_addr	9
2.1.7 get_mac_addr	10
2.1.8 set_wisun_phy_configs	10
2.1.9 get_wisun_phy_configs	10
2.1.10 set_icmpv6_id	11
2.1.11 set_icmpv6_seqnum	11
2.1.12 set_hop_limit	11
2.1.13 set_mtu_size	11
2.1.14 set_efde_mode	12
2.1.15 set_pkt_uint_num	12
2.1.16 set_pkt_tail	12
2.1.17 join_multicast_group	12
2.1.18 creat_udp_socket	13
2.1.19 creat_icmpv6_socket	13
2.1.20 set_udp_dst_port_num	13
2.1.21 socket_udp_send_to	13
2.1.22 user_udp_send_to	14
2.1.23 set_user_udp_payload	14
2.1.24 socket_icmpv6_send_echo_req_to	14
2.1.25 set_gtk_req_time	14
2.1.26 set_untrusted_cert_en	15
2.1.27 set_usr_chn_mask	15
2.1.28 get_usr_channel_mask	15
2.1.29 set_fhss_timing_configure	15
2.1.30 get_fhss_timing_configure	16
2.1.31 set_fhss_unicast_channel_function	16
2.1.32 get_fhss_unicast_channel_function	16

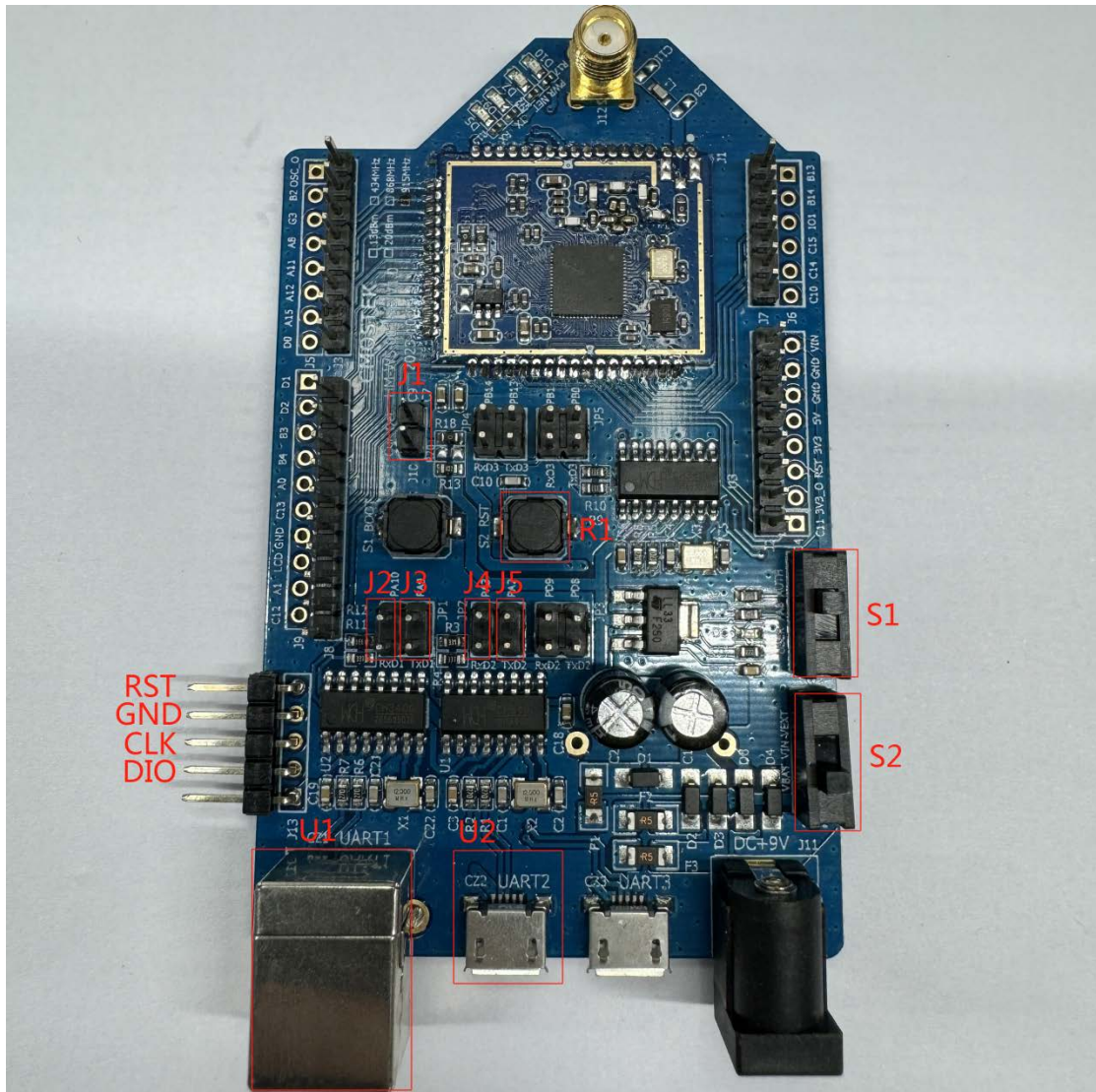
2.1.33	set_fhss_broadcast_channel_function	16
2.1.34	get_fhss_broadcast_channel_function	17
3	Procedure of UDP Data Transparent Transmit.....	17
3.1	Node Joins Network	17
3.2	Set up UDP and Transparent Data	18
4.	Revise History	19
5.	Contacts	20

1 Description

1.1 Related Pin Introduction

This document is an operation guide for the HOPERF Wi-SUN ROUTER. The HOPERF Wi-SUN ROUTER has been pre-burned by the firmware based on Wi-SUN FAN 1.0.

HOPERF Wi-SUN ROUTER DUT is shown as followed.



RST,GND,CLK,DIO are used for firmware burning upgrade.

J1, J2, J3, J4, and J5 need to be connected to pins with jumpers before the DUT is operated.

Switch S1 needs to be set to PA8_AUTH, and switch S2 needs to be set to VIN_VEXT.

Pressing R1 button can reset the HOPERF Wi-SUN ROUTER to power on and restart.

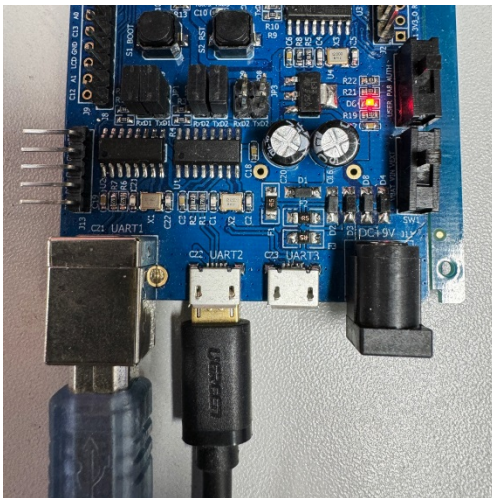
U1 is the power supply and DUT is the running log serial port, and U2 is the command serial port.

1.2 Serial Port Tool Installation and Usage

Click file of MobaXterm_21.1_Crack and double-click the installation package to install as shown below. Keep clicking Next until successfully install the software.

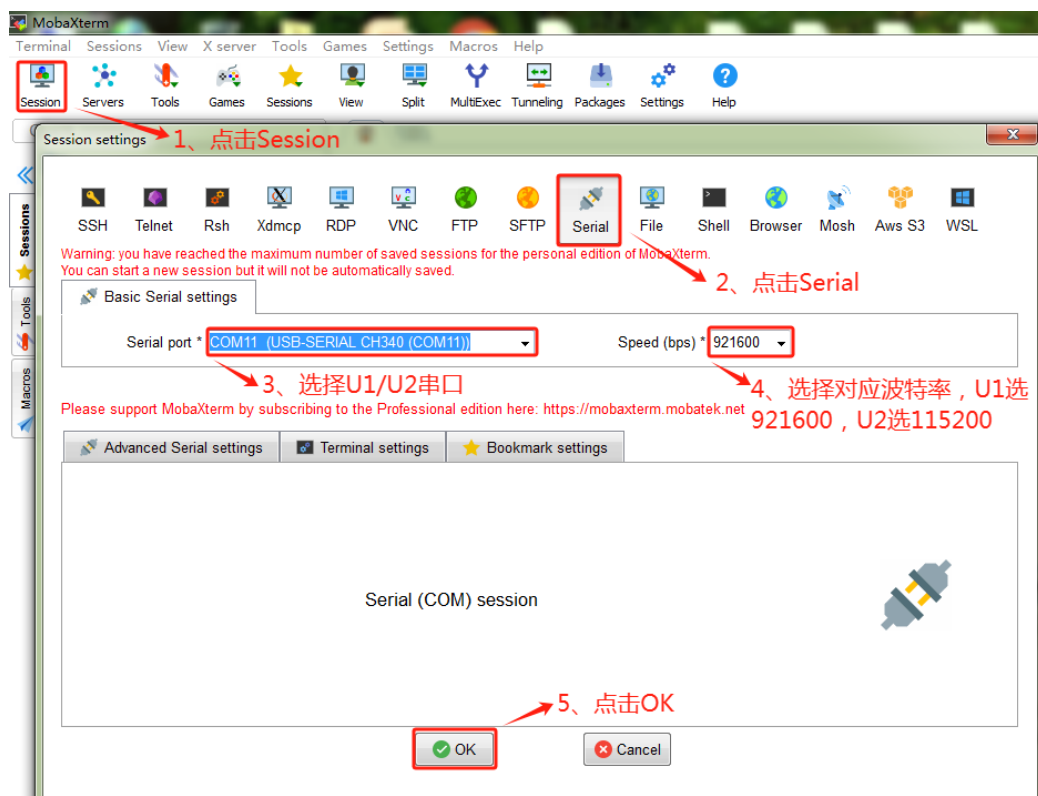
名称	修改日期	类型	大小
注册机	2024/2/27 星期...	文件夹	
MobaXterm_installer.dat	2021/4/5 星期一 ...	Probe Document	15,646 KB
MobaXterm_installer_21.1.msi	2021/4/5 星期一 ...	Windows Install...	12,004 KB

After successfully installing, set U1 and U2 as the following to connect to the PC machine and then operate MobaXterm.



The operation mode of MobaXterm software is shown in the following figure. It should be noted that the baud rates of U1 and U2 are different, and two serial ports need to create two sessions.

After successfully created the U1 Session, press R1 to reset and print the log as shown in the following figure.



```

boot 1.0V
main init
[INFO][main]: Start mesh-minimal application
[INFO][main]: Build: Feb 26 2024 16:46:44
Mesh type: WISUN
soft version: 6.15 wisun stack version: 1.0
[DBG][core]: Allocate Root Tasklet
[DBG][6lo]: P.Init

[DBG][cstr]: config set: gen, changed fields:
[DBG][cstr]: i: 0x1 v: 57:69:53:55:4e:20:50:41:4e
[DBG][cstr]: config set: phy, changed fields:
[DBG][cstr]: i: 0x1 v: 02:03
[DBG][cstr]: config set: fhss, changed fields:
[DBG][cstr]: i: 0x0 v: 0f
[DBG][core]: NS Root task Init
[DBG][cstr]: i: 0xA v: 00:00
[DBG][sck]: Socket Tasklet Generated
[DBG][cstr]: config set: fhss, changed fields:
[DBG][cstr]: i: 0x1 v: 64
[DBG][cstr]: i: 0x4 v: 90:01
[DBG][cstr]: i: 0xC v: 00:00
[DBG][cstr]: config set: timing, changed fields:
[DBG][cstr]: i: 0x0 v: 0f
[DBG][cstr]: i: 0x2 v: 3c:00
[DBG][cstr]: i: 0x6 v: 08:07
[DBG][cstr]: config set: rpl, changed fields:
[DBG][cstr]: i: 0x0 v: 0f:02:00
[DBG][cstr]: i: 0x8 v: 20:1c
[DBG][cstr]: i: 0xC v: 20:1c
[DBG][cstr]: config set: sec_prot, changed fields:
[DBG][cstr]: i: 0xC v: 0a
[DBG][cstr]: i: 0xE v: 3c:00:3c:00:0f
[DBG][cstr]: config set: mpl, changed fields:
[DBG][cstr]: i: 0x2 v: 0a
[DBG][cstr]: i: 0x6 v: 78:00
[INFO][main]: Connecting...
[INFO][cmt2310a]: cmt2310a is connecting

[INFO][cmt2310a]: update rf register configs: 2
[DBG][mlme]: SW-MAC driver support rf extension 150000 symbol/seconds 6666 ns symbol time length
[DBG][swm]: Set MAC mode to IEEE 802.15.4-2011, MTU size: 127
[DBG][WSIn]: connect()
[DBG][wllc]: Dynamic EAPOL entry max 19
[INFO][wsbs]: WS tasklet init
[INFO][wspt]: Key timers revocation lifetime: 86400, new activation time: 3600, max mismatch 3840, time to update: 82800
[INFO][wsbs]: ws fhss initialize
[DBG][wsbs]: Excluded ctrl 0, excluded channel count 0, total domain channels 64
[INFO][wsbs]: MAC address: 8c:1f:64:5e:40:00:00:06
[INFO][addr]: Address added to IF 1: fe80::8e1f:645e:4000:6
[DBG][addr]: LL64 Register OK!
[DBG][rout]: Added route:
[DBG][rout]: fe80::/64 if:1 src:'Static' id:0 lifetime:infinite
[DBG][rout]: On-link (met 128)
[DBG][rout]: Added route:
[DBG][rout]: ff00::/8 if:1 src:'Static' id:0 lifetime:infinite
[DBG][rout]: On-link (met 192)
[DBG][sck]: Socket id 0 allocated
[INFO][mntr]: Monitor rate limit incoming packets at:1024

```

After successfully created the U2 session, press R1 to reset and print the log as shown in the following figure.

```

HOPERF wisun shell
SV 3.1.3 build Feb 23 2024
exist
msh >

```

2 Command Description

After the Node is powered on, Serial port command display as the following figure.

```
HOPERF wisun shell
SV 3.1.3 build Feb 23 2024
exist
msh >|
```

Press Tab in command line can see all the commands.

```
HOPERF wisun shell commands:
version          - shell version information
reboot           - reset wisun system
wisun_config     - wisun current config
network_state    - wisun current network state
set_network_name - set wisun network name
get_network_name - get wisun network name
set_mac_addr     - set device mac address
get_mac_addr     - get device mac address
set_wisun_phy_configs - set wisun phy configs
get_wisun_phy_configs - get wisun phy configs
set_icmpv6_id    - set icmpv6 echo req pkt id
set_icmpv6_seqnum - set icmpv6 echo req pkt seqnum
set_hop_limit    - set pkt hop limit
set_mtu_size     - set pkt mtu size
set_edfe_mode    - set icmpv6 echo req or udp pkt edfe mode
set_pkt_unit_num - set icmpv6 echo req or udp pkt unit num
set_pkt_tail     - set icmpv6 echo req or udp pkt tail
join_multicast_group - join multicast group
create_udp_socket - create a udp socket and indicate port num
create_icmpv6_socket - create a icmpv6 socket
set_udp_dst_port_num - set udp dst port num
socket_udp_send_to - send udp pkt to dst addr
user_udp_send_to - user send udp pkt to dst addr
set_user_udp_payload - set user udp pkt payload
socket_icmpv6_send_echo_req_to - send icmpv6 echo req pkt to dst addr
set_gtk_req_time - set gtk req time
set_untrusted_cert_en - enable untrusted cert
set_usr_chn_mask - config user channel mask
get_usr_chn_mask - get usr channel mask
set_fhss_timing_configure - set fhss timing configure such as uc_dwell_interval bc_interval bc_dwell_interval
get_fhss_timing_configure - get fhss timing configure such as bc_dwell_interval bc_interval bc_dwell_interval
set_fhss_unicast_channel_function - set fhss unicast channel function
get_fhss_unicast_channel_function - get fhss unicast channel function
set_fhss_broadcast_channel_function - set fhss broadcast channel function
get_fhss_broadcast_channel_function - get fhss broadcast channel function
get_wisun_gtk_key - get wisun index gtk key
set_timing_parameters - set timing parameters
get_timing_parameters - get timing parameters
set_sniffer_en - enable sniffer or not
set_whls - set whls status pack value: 00 or 01
help          - HOPERF shell help

msh >|
```

When you enter a command on the command line, press Tab to automatically adding the rest of the commands, as shown below.

```
msh >set_fh
set_fhss_timing_configure
set_fhss_unicast_channel_function
set_fhss_broadcast_channel_function
msh >set_fhss_t
set_fhss_timing_configure
msh >set_fhss_timing_configure|
```

输入set_fh后按Tab键补齐命令至set_fhss_

再输入t后按Tab键补齐命令至set_fhss_timing_configure

2.1 ROUTER Related Command Description

2.1.1 reboot

Format: reboot

Parameter: none

Function: soft reset, reset Node and rerun

For instance:

```
msh >reboot
HOPERF wisun shell
SV 3.1.3 build Dec 21 2023
exist
msh >
```

2.1.2 wisun_config

Format: wisun_config

Parameter: none

Function: check the current Node configuration

For instance:

```
msh >wisun_config
network_name:WiSUN PAN
regulatory_domain:0x1, operating_class:0x2, operating_mode:0x3
bc_channel_function:2, bc_dwell_interval:255ms, bc_fixed_channel:0, bc_interval:1020ms
uc_channel_function:2, uc_dwell_interval:15ms, uc_fixed_channel:0
wisun_regulatory_id:2
whls_status:0
msh >
```

Turn back command description:

- 1) network_name represents the current network name of the Node application
- 2) regulatory_domain specific description is shown as followed.

```
REG_DOMAIN_WW 0x00 /**< World wide */
REG_DOMAIN_NA 0x01 /**< North America */
REG_DOMAIN_JP 0x02 /**< Japan */
REG_DOMAIN_EU 0x03 /**< European Union */
REG_DOMAIN_CH 0x04 /**< China */
REG_DOMAIN_IN 0x05 /**< India */
REG_DOMAIN_MX 0x06 /**< Mexico */
REG_DOMAIN_BZ 0x07 /**< Brazil */
REG_DOMAIN_AZ 0x08 /**< Australia */
REG_DOMAIN_NZ 0x08 /**< New zealand */
REG_DOMAIN_KR 0x09 /**< Korea */
REG_DOMAIN_PH 0x0A /**< Philippines */
REG_DOMAIN_MY 0x0B /**< Malaysia */
REG_DOMAIN_HK 0x0C /**< Hong Kong */
REG_DOMAIN_SG 0x0D /**< Singapore band 866-869 */
REG_DOMAIN_TH 0x0E /**< Thailand */
REG_DOMAIN_VN 0x0F /**< Vietnam */
REG_DOMAIN_SG_H 0x10 /**< Singapore band 920-925 */
```

- 3) Specific description of operating_class is shown as followed.


```

OPERATING_CLASS_1 0x01
OPERATING_CLASS_2 0x02
OPERATING_CLASS_3 0x03
OPERATING_CLASS_4 0x04

```

4) Specific description of operating_mode is shown as followed.

```

OPERATING_MODE_1a 0x1a /**< 50, 0,5 */
OPERATING_MODE_1b 0x1b /**< 50, 1.0 */
OPERATING_MODE_2a 0x2a /**< 100, 0,5 */
OPERATING_MODE_2b 0x2b /**< 100, 1.0 */
OPERATING_MODE_3 0x03 /**< 150, 0.5 */
OPERATING_MODE_4a 0x4a /**< 200, 0.5 */
OPERATING_MODE_4b 0x4b /**< 200, 1.0 */
OPERATING_MODE_5 0x05 /**< 300, 0.5 */

```

5) Specific description of channel_function is shown as followed.

```

CHANNEL_FUNCTION_FIXED      0x00 /**< Fixed channel */
CHANNEL_FUNCTION_TR51CF     0x01 /**< TR51CF */
CHANNEL_FUNCTION_DH1CF      0x02 /**< Direct Hash */
CHANNEL_FUNCTION_VENDOR_DEFINED 0x03 /**< vendor given channel hop schedule */

```

- 6) fixed_channel refers to the channel number used by Node when channel_function uses a fixed channel
- 7) uc_dwell_interval refers to unicast dwell interval in ms.
- 8) bc_dwell_interval refers to broadcast dwell interval in ms.
- 9) bc_interval refers to broadcast interval in ms.
- 10) wisun_regulatory_id is used to quickly configure regulatory_domain, operating_class, and operating_mode specified by Node for WiSUN authentication, as defined in the following figure

```

wisun_regulatory_id:
0x00: REG_DOMAIN_EU, OPERATING_CLASS_2, OPERATING_MODE_3
0x01: REG_DOMAIN_NA, OPERATING_CLASS_1, OPERATING_MODE_1b
0x02: REG_DOMAIN_NA, OPERATING_CLASS_2, OPERATING_MODE_3
0x03: user define REG_DOMAIN, OPERATING_CLASS, OPERATING_MODE
other: REG_DOMAIN_EU, OPERATING_CLASS_2, OPERATING_MODE_3

```

- 11) whls_status indicates the whitelist status. If the whitelist status is 0, the whitelist is not enabled.

2.1.3 network_state

Format: network_state

Parameter: none

Function: query the current Node access phase, network PAN ID, parent node signal strength, and parent node address

For example:


```
msh >network_state
network_state

join_state: 1
pan_id: 65535
parent_rssi : -87
parent: 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
global_address: 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
link_local_address: fe:80:00:00:00:00:00:00:8e:1f:64:5e:40:00:00:05
msh >
```

The command returned information description

join_state is shown in detailed as the following figure:

```
Wi-SUN join state 1 Discovery
Wi-SUN join state 2 Authentication
Wi-SUN join state 3 Configuration learn
Wi-SUN join state 4 RPL parent discovery
Wi-SUN join state 5 Active state
```

pan_id is defined by the border router, that is, PAN ID.

parent_rssi represents the rssi value of the current parent Node.

Parent is the link-local IP address of the parent Node of the current node

global_address is the global IP address of the current Node

link_local_address is the link-local IP address of the current node.

2.1.4 set_network_name

Format: set_network_name [param]

Parameter: network name

Function: set the network name for which the Node applies to join.

For example:

```
msh >set_network_name WiSUN PAN
set_network_name: WiSUN PAN
msh >
```

2.1.5 get_network_name

Format: get_network_name

Parameter: none

Function: get the network name for which the Node applies to join.

For example:

```
msh >get_network_name
get_network_name: WiSUN PAN
msh >
```

2.1.6 set_mac_addr

Format: set_mac_addr [param]

Parameter: MAC address

Function: set MAC address of Node

For example:

```
msh >set_mac_addr 1234567887654321
set_mac
mac_addr: 1234567887654321
msh >
```

Description: After the configuration is complete, run the reboot command to restart the configuration.

2.1.7 get_mac_addr

Format: get_mac_addr

Parameter: none

Function: get the MAC address of Node

For example:

```
msh >get_mac_addr
mac_address: 12:34:56:78:87:65:43:21
msh >
```

2.1.8 set_wisun_phy_configs

Format: set_wisun_phy_configs [param]

Parameter: set the PHY configuration ID

Function: sets which PHY configuration the Node uses

For example:

```
msh >set_wisun_phy_configs 0x2
set wisun phy configs -- regulatory domain: 0x01, operating class: 0x02, operating mode: 0x03
msh >
```

Parameter description:

```
wisun_regulatory_id:
0x00: REG_DOMAIN_EU, OPERATING_CLASS_2, OPERATING_MODE_3
0x01: REG_DOMAIN_NA, OPERATING_CLASS_1, OPERATING_MODE_1b
0x02: REG_DOMAIN_NA, OPERATING_CLASS_2, OPERATING_MODE_3
0x03: user define REG_DOMAIN, OPERATING_CLASS, OPERATING_MODE
other: REG_DOMAIN_EU, OPERATING_CLASS_2, OPERATING_MODE_3
```

2.1.9 get_wisun_phy_configs

Format: get_wisun_phy_configs

Parameter: none

Function: get the current PHY configuration of Node

For example:

```
msh >get_wisun_phy_configs
get_wisun_phy_configs -- regulatory domain: 0x01, operating class: 0x02, operating mode: 0x03
msh >
```

2.1.10 set_icmpv6_id

Format: set_icmpv6_id [param]

Parameter: ID number of ICMPv6 packet

Function: set ID number of ICMPv6 packet

For example:

```
msh >set_icmpv6_id 0x1122
icmpv6 id: 0x1122
msh >
```

2.1.11 set_icmpv6_seqnum

Format: set_icmpv6_seqnum [param]

Parameter: sequence number of ICMPv6 packet

Function: set the sequence number of ICMPv6 packet

For example:

```
msh >set_icmpv6_seqnum 0x1234
icmpv6 seqnum: 0x1234
msh >
```

2.1.12 set_hop_limit

Format: set_hop_limit [param]

Parameter: hop limit of data packet

Function: set hop limit of data packet

For example:

```
msh >set_hop_limit 0x14
hop limit: 0x14
msh >
```

2.1.13 set_mtu_size

Format: set_mtu_size [param]

Parameter: MTU value of data packet

Function: set MTU value of data packet

For example:

```
msh >set_mtu_size 0x100
mtu size: 0x100
msh >
```

2.1.14 set_efde_mode

Format: set_efde_mode [param]

Parameter: edfe mode enable bit

Function: set whether to enable edfe mode

For example:

```
msh >set_efde_mode 0x1
set edfe mode: 0x1
msh >
```

2.1.15 set_pkt_uint_num

Format: set_pkt_uint_num [param]

Parameter: packet unit number

Function: set the number of packet units to be refilled in the packet

For example:

```
msh >set_pkt_uint_num 0x10
pkt unit num: 0x10
msh >
```

description: This command is mainly used for WiSUN authentication with WiSUN qualified test requirement and the data unit value are

0x61,0x62,0x63,0x64,0x65,0x66,0x67,0x68,0x69,0x6a,0x6b,0x6c,0x6d,0x6e,0x6f,0x70,0x71,0x72,0x73,0x74,0x75,0x76,0x77,0x78,0x79,0x7a. User can be set the number of authentication data units to be repopulated in the packet according to test needs.

2.1.16 set_pkt_tail

Format: set_pkt_tail [param]

Parameter: data of the packet tail

Function: This command is mainly used for WiSUN authentication. According to WiSUN authentication test requirements, the data tail is 10 bytes. User can set the data tail in the authentication packet as required.

For example:

```
msh >set_pkt_tail 11223344556677889900
pkt tail is: 11223344556677889900
msh >
```

2.1.17 join_multicast_group

Format: join_multicast_group [param]

Parameter: multicast address

Function: Adds the Node to a specified multicast group

For example:

```
msh >join_multicast_group 11111111111111112222222222222222
set multicast addr: 11111111111111112222222222222222
msh >
```

2.1.18 creat_udp_socket

Format: creat_udp_socket [param]

Parameter: UDP socket number

Function: set a UDP socket with the specified port number

For example:

```
msh >creat_udp_socket 0x1234
udp port num: 0x1234
msh >
```

2.1.19 creat_icmpv6_socket

Format: creat_icmpv6_socket

Parameter: none

Function: set up ICMPv6 socket

For example:

```
msh >creat_icmpv6_socket
msh >
```

2.1.20 set_udp_dst_port_num

Format: set_udp_dst_port_num [param]

Parameter: port number

Function: Set the port number of the destination IP address for UDP transmission

For example:

```
msh >set_udp_dst_port_num 0x4321
dst udp port num: 0x4321
msh >
```

2.1.21 socket_udp_send_to

Format: socket_udp_send_to [param]

Parameter: IP address

Function: The destination IP address for UDP transmission is specified. This command is mainly used for WiSUN authentication. According to WiSUN authentication test requirements, the packet consists of data unit and data tail and cannot be freely configured by users.

For example:

```
msh >socket_udp_send_to 11111111111111112222222222222222
send udp packet to dst addr: 11111111111111112222222222222222
msh >
```

2.1.22 user_udp_send_to

Format: user_udp_send_to [param]

Parameter: IP address

Function: Set the destination IP address of the user for UDP transmission.

For example:

```
msh >user_udp_send_to fd006868686800001034567887654321
send udp packet to dst addr: fd006868686800001034567887654321
msh >
```

2.1.23 set_user_udp_payload

Format: set_user_udp_payload [param]

Parameter: user UDP transmit data packet

Function: set the user UDP transmit data packet

For example:

```
msh >set_user_udp_payload 1234567890
payload is: 1234567890 len is 5 byte
msh >
```

2.1.24 socket_icmpv6_send_echo_req_to

Format: socket_icmpv6_send_echo_req_to[param]

Parameter: IP address

Function: set the specified ICMPv6 transmission destination IP address. This command is mainly used for WiSUN authentication. According to WiSUN authentication test requirements, the packet consists of data unit and tail data and cannot be freely configured by users

For example:

```
msh >socket_icmpv6_send_echo_req_to 11223344556677881122334455667788
send icmpv6 packet to dst addr: 11223344556677881122334455667788
msh >
```

2.1.25 set_gtk_req_time

Format: set_gtk_req_time [param]

Parameter: GTK_MAX_MISMATCH

Function: set the GTK_MAX_MISMATCH time in s. This command is used for WiSUN authentication, which is for updating the corresponding GTK time.

For example:

```
msh >set_gtk_req_time 0x3c
gtk_request_imin:10, gtk_request_imax:60, gtk_max_mismatch:60
msh >
```

2.1.26 set_untrusted_cert_en

Format: set_untrusted_cert_en [param]

Parameter: untrusted certificate enable bit

Function: sets Node to whether uses untrusted certificates. This command is mainly used for WiSUN authentication and can be used to change untrusted certificates.

For example:

```
msh >set_untrusted_cert_en 0x1
untrusted_cert_en:0x1
msh >
```

2.1.27 set_usr_chn_mask

Format: set_usr_chn_mask [param1] [param2] [param3] [param4] [param5] [param6] [param7] [param8]

Parameter: mask the corresponding channel by bit domain

Function: set Node to mask the corresponding channel among 256 channels

For example:

```
msh >set_usr_chn_mask 0x11111111 0x2 0x3 0x4 0x5 0x6 0x7 0x8
fhss_channel_mask: 0x11111111 0x00000002 0x00000003 0x00000004 0x00000005 0x00000006 0x00000007 0x00000008
msh >
```

2.1.28 get_usr_channel_mask

Format: get_usr_channel_mask

Parameter: none

Function: get the mask information of 256 channels of the Node.

For example:

```
msh >get_usr_chn_mask
usr_channel_mask: 0x11111111 0x00000002 0x00000003 0x00000004 0x00000005 0x00000006 0x00000007 0x00000008
msh >
```

2.1.29 set_fhss_timing_configure

Format: set_fhss_timing_configure [param1] [param2] [param3]

Parameter1: unicast dwelling time

Parameter2: broadcast interval

Parameter3: unicast dwelling time

Function: set the Node unicast time

For example:

```
msh >set_fhss_timing_configure 0xf 0x190 0x64
fhss timing configure: uc_dwll_interval 15 ms, bc_interval 400 ms, bc_dwll_interval 100 ms
msh >
```

2.1.30 get_fhss_timing_configure

Format: get_fhss_timing_configure

Parameter: none

Function: get the Node unicast broadcast time

For example:

```
msh >get_fhss_timing_configure
fhss timing configure: uc_dwell_interval 15 ms, bc_interval 400 ms, bc_dwell_interval 100 ms
msh >
```

2.1.31 set_fhss_unicast_channel_function

Format: set_fhss_unicast_channel_function [param1] [param2] [param3]

Parameter1: channel_function of Node unicast

Parameter2: Channel number used by Node unicast on fixed channel

Parameter3: unicast dwelling time

Function: set the Node unicast configuration

For example:

```
msh >set_fhss_unicast_channel_function 0x1 0x0 0xff
unicast channel configure: channel_function: 0x01 fixed_channel: 0x0000 dwell_interval: 255 ms
msh >
```

2.1.32 get_fhss_unicast_channel_function

Format: get_fhss_unicast_channel_function

Parameter: none

Function: get the Node configuration information

For example:

```
msh >get_fhss_unicast_channel_function
unicast channel configure: channel_function: 0x01 fixed_channel: 0xffff dwell_interval: 255 ms
msh >
```

2.1.33 set_fhss_broadcast_channel_function

Format: set_fhss_broadcast_channel_function [param1] [param2] [param3] [param4]

Parameter1: channel_function of the Node broadcast

Parameter2: channel number used for fixed channel Node broadcast

Parameter3: broadcast dwelling time

Parameter4: broadcast interval time

Function: set the Node broadcast configuration

For example:

```
msh >set_fhss_broadcast_channel_function 0x1 0x0 0xff 0x3fc
broadcast channel function: channel_function: 0x01 fixed_channel: 0x0000 dwell_interval: 255 ms broadcast_interval: 1020 ms
msh >
```

2.1.34 get_fhss_broadcast_channel_function

Format: get_fhss_broadcast_channel_function

Parameter: none

Function: get the Node broadcast configuration

For example:

```
msh >get_fhss_broadcast_channel_function  
broadcast channel function: channel_function: 0x01 fixed_channel: 0xffff dwell_interval: 255 ms broadcast_interval: 1020 ms  
msh >
```

3 Procedure of UDP Data Transparent Transmit

3.1 Node Joins Network

Power on a Node and connect it to a specified Rank, enable the serial ports of log and command.

After completing the hardware connection, update Node configurations before connecting to the network through the serial port command. The steps are as follows:

1. Operate get_network_name to check the specified network name of the Node. If network_name is WiSUN PAN, go to step 3, otherwise go to step 2.
2. Operate set_network_name, that is, the specific current Node name of the WiSUN PAN command.
3. Operate network_state command to check the current network state of Node, if join_state is set to 1, go to Step 5; otherwise, go to Step 4.
4. Run reboot command to reset the Node and then perform Step 1 again.
5. Run the set_wisun_phy_configs 0x2 command to set the Node physical layer Frequency to 902.4 MHz, channel spacing to 400kHz, and data rate to 150kbps.
6. Run the set_fhss_unicast_channel_function 0x2 0x0 0xff command to set the unicast channel to direct hash channel mode. The channel number applied in the unicast fixed channel mode is 0. Set the unicast residence time to 255ms.
7. Run the set_fhss_broadcast_channel_function 0x2 0x0 0xff 0x3fc command to set the broadcast channel function to the direct hash channel mode, In fixed channel mode, set the channel number to 0, broadcast residence time to 255 ms, and broadcast interval to 1020 ms.
8. After configuration complete, run the reboot command to reset the Node and run with the configuration. Then wait for Node to enter the network.

The configuration diagram of the command serial port is as follows:

```

msh >get_ne
get_network_name
msh >get network_name
get_network_name: WiSUN PAN
msh >netwo
network_state
msh >network_state
network_state

join_state: 1
pan_id: 65535
parent_rssi : -87
parent: 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
global_address: 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
link_local_address: fe:80:00:00:00:00:00:00:8e:1f:64:5e:40:00:00:06
msh >set_wis
set_wisun_phy_configs
msh >set_wisun_phy_configs 0x2
set_wisun_phy_configs -- regulatory domain: 0x01, operating class: 0x02, operating mode: 0x03
msh >set_fhss_u
set_fhss_unicast_channel_function
msh >set_fhss_unicast_channel_function 0x2 0x0 0xff
unicast channel configure: channel_function: 0x02 fixed_channel: 0x0000 dwell_interval: 255 ms
msh >set_fhss_br
set_fhss_broadcast_channel_function
msh >set_fhss_broadcast_channel_function 0x2 0x0 0xff 0x3fc
broadcast channel function: channel_function: 0x02 fixed_channel: 0x0000 dwell_interval: 255 ms broadcast_interval: 1020 ms
msh >reboot
HOPERF wisun shell
SV 3.1.3 build Feb 23 2024
exist
msh >

```

3.2 Set up UDP and Transparent Data

After a Node is successfully connected to the network, you can see that the Node prints its IP address through the log serial port.

```

[INFO][main]: get ip address
[DBG ][main]: Connected. IP = fd00:6868:6868:0:1034:5678:8765:4321
[INFO][wsbs]: received ADVERT Src:8e:1f:64:5e:40:00:f2:0b panid:438e rssi:-38
[DBG ][rplu]: Timed parent selection
[INFO][rplu]: Time 00:04:12.0 (252.0) RPL memory usage 408

```

After the specific Node adds to network successfully, the current network state can be check at the network_state command. If join_state is set to 5, you can return information to know the global and link-local IP address of the current Node, which can be used as the destination address for other devices to start UDP transparent transmission. Then create_udp_socket creates a UDP socket and binds it to the specified port number.

```

msh >network_state
network_state

join_state: 5
pan_id: 4245
parent_rssi : -38
parent: fe:80:00:00:00:00:00:00:8c:1f:64:5e:40:00:f2:0b
global_address: fd:00:68:68:68:00:00:10:34:56:78:87:65:43:21
link_local_address: fe:80:00:00:00:00:00:10:34:56:78:87:65:43:21
msh >create_udp_socket 0x1111
udp port num: 0x1111
msh >

```

After the source Node is successfully connected to the network, the current network status of the Node can be check by network_state command. If join_state is set to 5, you can return information to know the global and link-local IP addresses of the Node. The UDP socket is also created with create_udp_socket and bound to the specified port number.

```

msh >network_state
network_state

join_state: 5
pan_id: 4245
parent_rssi : -64
parent: fe:80:00:00:00:00:10:34:56:78:87:65:43:21
global_address: fd:00:68:68:68:68:00:00:28:91:6d:43:4f:bf:59:34
link_local_address: fe:80:00:00:00:00:00:28:91:6d:43:4f:bf:59:34
msh >create_udp_socket 0x5555
udp port num: 0x5555
msh >

```

Run the set_udp_dst_port_num command on the source Node to set the specific port number of the UDP Node. Then user can run the set_user_udp_payload command to set the transparent UDP packet content. Finally, user_udp_send_to command will transmit the UDP packet to the specified device with IP address.

```

msh >set_udp_dst_port_num 0x1111
dst udp port num: 0x1111
msh >set_user_udp_payload 112233445566
payload is: 112233445566 len is 6 byte
msh >user_udp_send_to fd00:6868:6868:0000:1034:5678:8765:4321
send udp packet to dst addr: fd00:6868:6868:0000:1034:5678:8765:4321
msh >

```

After the destination Node successfully receives the UDP packet from the source Node, the destination node prints the following information on the log serial port:

```

[DBG ][rplu]: Timed parent selection
[INFO][rplu]: Time 00:31:26.0 (1886.0) RPL memory usage 408
[INFO][rplu]: RPL Domain 0x2000DB38
[INFO][rplu]: RPL Instance 1
[INFO][rplu]: -----
[INFO][rplu]: DODAG fd00:6868:6868:0:8c1f:645e:4000:f20b
[INFO][rplu]:   G=1 MOP=1 Prf=0
[INFO][rplu]: fd00:6868:6868::/ 64 lifetime:0 flags:--
[INFO][rplu]:   Version 241
[INFO][rplu]:   *Current version* Rank=0188
[INFO][rplu]:   1*00c4 0144 ff fe80::8c1f:645e:4000:f20b%1 (fd00:6868:6868:0:8c1f:645e:4000:f20b)
[INFO][rplu]: DIO trickle Imin=328, Imax=1311, k=0
[INFO][rplu]:   I=1311, now=601, t=1151, c=1
[INFO][rplu]: DAO Targets:
[INFO][rplu]:   fd00:6868:6868:0:1034:5678:8765:4321/128 ff seq=240 (pub)
[WARN][wsmg]: recv udp socket pkt
[WARN][wsmg]: src_add fd00:6868:6868:0:2891:6d43:4fbf:5934
[WARN][wsmg]: src_port 0x5555
[WARN][wsmg]: pkt_len 6
[WARN][wsmg]: val[0] is 0x11
[WARN][wsmg]: val[1] is 0x22
[WARN][wsmg]: val[2] is 0x33
[WARN][wsmg]: val[3] is 0x44
[WARN][wsmg]: val[4] is 0x55
[WARN][wsmg]: val[5] is 0x66
[INFO][wsbs]: received ADVERT Src:2a:91:6d:43:4f:bf:59:34 panid:1095 rssi:-64
[INFO][wsbs]: Send PAN configuration
[INFO][rplc]: transmit dio, rank: 188
[INFO][wsbs]: Send PAN advertisement
[INFO][fhss]: TX slot length: 54ms

```

4. Revise History

Version No.	Chapter	Description	Date
0.1	All	Initial version	2024/08/19

5. Contacts

Shenzhen Hope Microelectronics Co., Ltd.

Address: 30th floor of 8th Building, C Zone, Vanke Cloud City, Xili Sub-district, Nanshan, Shenzhen, GD,
P.R. China

Tel: +86-755-82973805 / 4001-189-180

Fax: +86-755-82973550

Post Code: 518052

Sales: sales@hoperf.com

Website: www.hoperf.com

Copyright. Shenzhen Hope Microelectronics Co., Ltd. All rights are reserved.

The information furnished by HOPERF is believed to be accurate and reliable. However, no responsibility is assumed for inaccuracies and specifications within this document are subject to change without notice. The material contained herein is the exclusive property of HOPERF and shall not be distributed, reproduced, or disclosed in whole or in part without prior written permission of HOPERF. HOPERF products are not authorized for use as critical components in life support devices or systems without express written approval of HOPERF. The HOPERF logo is a registered trademark of Shenzhen Hope Microelectronics Co., Ltd. All other names are the property of their respective owners.