

Wi-SUN - SoC Border Router

- [Introduction](#)
- [Getting Started](#)
- [Wi-SUN Border Router Commands](#)
- [Wi-SUN Border Router Settings](#)
 - [wisun Section Settings](#)
 - [GTK/LGTK Management](#)
 - [app Section Settings](#)
 - [ping Section Settings](#)
 - [wifi Section Settings](#)
- [Wi-Fi Connectivity](#)
 - [Wi-Fi Commands](#)
- [RTT traces](#)
- [Troubleshooting](#)
- [Resources](#)
- [Report Bugs & Get Support](#)

Introduction

The Wi-SUN Border Router demonstration provides a Wi-SUN Border Router implementation running entirely on the EFR32. It provides an easy and quick way to evaluate the Silicon Labs Wi-SUN stack solution without deploying an expensive and cumbersome production-grade Wi-SUN Border Router. A CLI (Command-Line Interface) is exposed to facilitate the configuration.

The Wi-SUN Border Router demonstration is delivered only in a binary format. The implementation does not scale for a production-grade Border Router maintaining several thousand Wi-SUN nodes.

Getting Started

To get started with Wi-SUN and Simplicity Studio, see [QSG181: Wi-SUN SDK Quick Start Guide](#).

The Wi-SUN Border Router demonstration is required to use the other Wi-SUN sample applications. The Wi-SUN Border Router creates a Wi-SUN network the Wi-SUN nodes can join. When part of the same network, the Wi-SUN nodes are able to exchange IP packets.

To get started with the demo, follow these steps:

- Flash the "Wi-SUN Border Router" demonstration to a compatible device.
- In Simplicity Studio, open a console on the device.

- Start the Border Router using the CLI command `wisun start_fan11` when using FAN-1.1 PHY configuration type and `wisun start_fan10` in order to use FAN-1.0 Phy configuration type.

See the associated sections in [QSG181: Wi-SUN SDK Quick Start Guide](#) if you want step-by-step guidelines for each operation.

Wi-SUN Border Router Commands

To see the available commands, enter the following command in the console:

```
wisun help
```

The list of available commands is output on the console with the associated help. Following is an extended description and examples of how to use each command.

Command	Description	Example
<code>wisun get <section>.[setting]</code>	Get a setting variable. Specifying only the section retrieves all settings of the section.	<pre>> wisun get wisun.ip_addresses > wisun get wisun > wisun get wisun.join_state > wisun get statistics > wisun get app</pre>
<code>wisun set <section>.[setting] <value></code>	Set a setting variable	<pre>> wisun set wisun.network_size test</pre>
<code>wisun start_fan10</code>	Start the border router using FAN-1.0 PHY settings	<pre>> wisun start_fan10</pre>
<code>wisun start_fan11</code>	Start the border router using FAN-1.1 PHY settings	<pre>> wisun start_fan11</pre>
<code>wisun start_explicit</code>	Start the border router using explicit PHY settings	<pre>> wisun start_explicit</pre>
<code>wisun start_ids</code>	Start the border router using radio conf ids	<pre>> wisun join_ids</pre>
<code>wisun start_custom_fsk</code>	Start the border router using custom FSK PHY settings	<pre>> wisun join_custom_fsk</pre>
<code>wisun start_custom_ofdm</code>	Start the border router using custom OFDM PHY settings	<pre>> wisun join_custom_ofdm</pre>
<code>wisun start_custom_oqpsk</code>	Start the border router using custom OQPSK PHY settings	<pre>> wisun join_custom_oqpsk</pre>
<code>wisun stop</code>	Stop the border router	<pre>> wisun_stop</pre>
<code>wisun mac_allow <mac address></code>	Add an allowed MAC address to the access list broadcast address: allow all MAC addresses unique address: allow the given MAC address	<pre>> wisun mac_allow ff:ff:ff:ff:ff:ff:ff > wisun mac_allow ffffffff</pre>

Command	Description	Example
wisun mac_deny <mac address>	Add a denied MAC address to the access list broadcast address: deny all MAC addresses unique address: deny the given MAC address	> wisun mac_deny 00:01:02:03:04:05:06:07 > wisun mac_deny 0001020304050607
wisun ping <remote address>	Ping a remote host	> wisun ping fd00:7283:7e00:0:fd6f:d00:a8c0:20fe
wisun reset	Reset variables to default values	> wisun reset
wisun save	Save variables to NVM	> wisun save
wisun socket_close <socket>	Close an open socket	> wisun socket_close 3
wisun socket_list	List open sockets	> wisun socket_list
wisun socket_read <socket> <amount of bytes>	Read buffered data from a socket	> wisun socket_read 3 14
wisun socket_set_option <socket> <option> <option data>	Configure a socket	> wisun socket_set_option 3 event_mode indication
wisun socket_write <socket> <string>	Write a string to a socket	> wisun socket_write 3 "hello world"
wisun socket_writeto <socket> <remote address> <remote port> <string>	Write a string to an unconnected socket, i.e. UDP server socket	> wisun socket_writeto 3 fc00::1 5001 "hello world"
wisun tcp_client <remote address> <remote port>	Open a TCP connection to a remote host	> wisun tcp_client fd24:120b:802c:0001:705d:9179:8607:fd21 5001
wisun tcp_server <local port>	Open a TCP server port	> wisun tcp_server 5001
wisun udp_client <remote address> <remote port>	Open a UDP connection to a remote host	> wisun udp_client fd00:6172:6d00:0:3038:5115:26:27 7
wisun udp_server <local port>	Open a UDP server port	> wisun udp_server 5001
wisun socket_get_option <socket> <option>	Get a socket option	> wisun socket_get_option
wisun clear_credential_cache	Clear the wisun credential cache	> wisun clear_credential_cache
wisun reset_statistics	Reset the statistics	> wisun reset_statistics
wisun set_trace_level <string>	Set trace level for a trace group	> wisun set_trace_level all,debug [Set 64 trace groups] > wisun set_trace_level 1,none;4,info [Set 2 trace groups]
wisun set_lfn_support <lfn_limit>	configure LFN support: - [uint8] lfn_limit: Number of LFN children (10 maximum)	> wisun set_lfn_support 3

Command	Description	Example
<code>wisun mode_switch <mode></code> <code><phy_mode_id></code> <code><neighbor_mac_address></code>	configure mode switch: - [uint8] mode: 0=disabled, 1=enabled, 2=default - [uint8] PhyModeID - [string] neighbor MAC address (optional)	<code>> wisun mode_switch 1 84</code> <code>> wisun mode_switch 1 84</code> <code>01:02:03:04:05:06:07:08</code>
<code>wisun trigger <frame type></code>	Trigger a transmission of a periodic frame - [string] pas: PAN Advertisement Solicit, pa: PAN Advertisement, pcs: PAN Configuration Solicit, pc: PAN Configuration, dis: DODAG Information Solicitation, dio: DODAG Information Object	<code>> wisun trigger pas</code>
<code>wisun getsockname <socket></code>	Get local IP address of the socket	<code>> wisun getsockname 3</code>
<code>wisun getpeername <socket></code>	Get peer IP address of the socket	<code>> wisun getpeername 3</code>
<code>wisun set_dhcp_vendor_data</code> <code><enterprise number> <sub-</code> <code>option length vendor data></code>	Set DHCPv6 Vendor Data delivered to nodes	<code>> wisun set_dhcp_vendor_data 39873</code> <code>{0001000568656c6c6f}</code>
<code>wisun set_phy_sens</code> <code><phy_mode_id> <sensitiv></code>	Set the sensitivity for the given PhyModeID - [uint8] PhyModeID - [int16] sensitivity in dBm	<code>> wisun set_phy_sens 1 -70</code>
<code>wisun pan_defect_advertise</code> <code><state> <min_scan_time></code> <code><max_scan_time></code>	Set PAN Defect Advertisement parameters - [uint8] state: 0=disabled, 1=enabled - [uint32] minimum duration in seconds for switching PANs - [uint32] maximum duration in seconds for switching PANs	<code>wisun pan_defect_advertise 1 600 1800</code>

Wi-SUN Border Router Settings

The command-line interface maintains a number of settings. The Wi-SUN settings are distributed in two sections: *wisun* and *app*. They can be listed by entering:

```
wisun get <section name>
```

The Wi-SUN stack settings are listed with their current state/value. Some of them can be modified by using the following command prototype:

```
wisun set <section name>.<settings name> <value>
```

To modify the network name used by the Wi-SUN Border Router, enter:

```
wisun set wisun.network_name "My Network"
```

Next time you issue the `wisun start_fan11` command in the case of a FAN-1.1 PHY, the Border Router creates a Wi-SUN network named "My Network".

wisun Section Settings

- The settings in the *wisun* section are directly related to the Wi-SUN stack behavior. A detailed settings list can be found below.
- Writable settings are valid for all PHYs unless it is specified otherwise in the last column .

Variable	R/W	Type	Values	Description	Applies to
wisun.network_name	R/W	string	up to 31 ASCII characters. Use double quotes (") to include spaces	Name of the Wi-SUN network	
wisun.phy_config_type	R	string	- FAN1.0 (0) - FAN1.1 (1) - explicit (2) - IDs (3)	Wi-SUN PHY configuration type (depends on the <code>wisun join_xxxx</code> used)	
wisun.regulatory_domain	R/W	integer	- WW (0): Worldwide - NA (1): North America - JP (2): Japan - EU (3): Europe - CN (4): China - IN (5): India - MX (6): Mexico - BZ (7): Brazil - AZ (8): Australia/New Zealand - KR (9): South Korea - PH (10): Philippines - MY (11): Malaysia - HK (12): Hong Kong - SG (13): Singapore - TH (14): Thailand - VN (15): Vietnam	Regulatory domain of the Wi-SUN network	- FAN1.0 - FAN1.1
wisun.operating_class	R/W	integer	1 to 4	Wi-SUN FAN1.0 operating class	FAN1.0
wisun.operating_mode	R/W	integer	- 0x1a (1a) - 0x1b (1b) - 0x2a (2a) - 0x2b (2b) - 0x3 (3) - 0x4a (4a) - 0x4b (4b) - 0x5 (5)	Wi-SUN FAN1.0 operating mode	FAN1.0
wisun.network_size	R/W	integer	- automatic (0) - small (default): less than 100 nodes (1) - medium: 100 to 1000 nodes (2) - large: more than 1000 nodes (3) - test (4) - certification (6)	Used to optimize network timings considering the number of expected nodes on the network.	
wisun.tx_power	R/W	integer	-45 to 20	Maximum TX power in dBm	
wisun.tx_power_ddbm	R/W	integer	-450 to 200	Maximum TX power in deci-dBm	
wisun.unicast_dwell_interval	R/W	integer	10 to 255	Unicast dwell interval in milliseconds	
wisun.broadcast_dwell_interval	R/W	integer	100 to 255	Broadcast dwell interval in milliseconds	
wisun.broadcast_interval	R/W	integer	broadcast_dwell_interval to 16777	Broadcast interval in milliseconds	

Variable	R/W	Type	Values	Description	Applies to
wisun.chan_plan_id	R/W	integer	- Europe: 32-37 - North America: 1-5 - Brazil: 1-5 - Japan: 21-24	Wi-SUN channel plan ID [uint8]	FAN1.1
wisun.phy_mode_id	R/W	integer	- Europe: FSK without FEC: 1/3/5 FSK FEC: 17/19/21 OFDM: 84-86 - North America: FSK without FEC: 2/3/5/6/8 FSK FEC: 18/19/21/22/24 OFDM: 34-38 / 51-54 / 68-70 / 84-86 - Brazil: FSK without FEC: 2/3/5/6/8 FSK FEC: 18/19/21/22/24 34-38 / 51-54 / 68-70 / 84-86 - Japan: FSK without FEC: 2/4/5/7/8 FSK FEC: 18/20/21/23/24 OFDM: 51-54 / 68-70 / 84-86	Wi-SUN FAN1.1 PHY mode ID	- FAN1.1 - explicit - join by IDs
wisun.ch0_frequency	R/W	integer		Explicit channel plan: Central frequency of the first channel in kHz	explicit
wisun.channel_spacing	R/W	integer	- 100 kHz (0) - 200 kHz (1) - 400 kHz (2) - 600 kHz (3) - 250 KHz (4) - 800kHz (5) - 1200KHz (6)	Explicit channel plan: Channel spacing	explicit
wisun.number_of_channels	R/W	integer		Explicit channel plan: Number of channels	explicit
wisun.protocol_id	R/W	integer		Radioconf protocol ID [uint16]	join by IDs
wisun.channel_id	R/W	integer		Radioconf channel ID [uint16]	join by IDs
wisun.ip_addresses	R	list of IPv6 addresses		List of all IP addresses assigned to the device	
wisun.border_router	R	list of IPv6 addresses		List of known IPv6 addresses of the border router	
wisun.parents	R	list of IPv6 addresses		List of known IPv6 addresses of the parents	
wisun.neighbors	R	list of IPv6 addresses		List of known IPv6 addresses of the RPL neighbors	
wisun.state	R	integer	- initialized (0) - operational (1)	State of the border router	
wisun.mac_address	R	MAC address		MAC address to use	
wisun.allowed_channels	R/W	string		Allowed channel ranges (e.g. 0-54,57-60,64,67-68)	
wisun.fec	R/W	bool	- disable FEC (0) - enable FEC (1)	Use FEC flag	FAN1.0
wisun.regulation	R/W	string	- None (0) - ARIB (1)	Regional regulation	
wisun.regulation_warning_threshold	R/W	integer	- threshold value (0-100) - disable (-1)	Transmission power warning threshold in percent (-1 to disable)	
wisun.regulation_alert_threshold	R/W	integer	- threshold value (0-100) - disable (-1)	Transmission power alert threshold in percent (-1 to disable)	

Variable	R/W	Type	Values	Description	Applies to
wisun.trace_filter.000-031	R/W	hexadecimal	(Refer to sl_wisun_types.h / sl_wisun_trace_group_t for a list of all 37 trace groups)	RTT Trace filter enable bitfield for first 32 groups	
wisun.trace_filter.032-063	R/W	hexadecimal	(Refer to sl_wisun_types.h / sl_wisun_trace_group_t for a list of all 37 trace groups)	RTT Trace filter enable bitfield for last 32 groups	
wisun.lfn_profile	R/W	integer	- Test (0) - Balanced (1) - Eco (2)	Wi-SUN LFN profile	FAN1.1
wisun.rx_phy_mode_ids	R/W	integer	list of phy_mode_ids	List of PhyModeId to advertise in POM-IE	FAN1.1
wisun.crc_type	R/W	integer	- no CRC (0) - 2-byte CRC (1) - 4-byte CRC (2)	CRC for custom FSK/OFDM/OQPSK PHY	
wisun.preamble_length	R/W	integer		Preamble length in bits for custom FSK/OQPSK PHY	
wisun.stf_length	R/W	integer		Short Training Field length in symbols for custom OFDM PHY	
wisun.async_channel_mask	R	xx:....xx	- Not masked (0) - Masked (1)	Excluded channel mask applied to async frames	
wisun.unicast_channel_mask	R	xx:....xx	- Not masked (0) - Masked (1)	Excluded channel mask applied to unicast frequency hopping	
wisun.broadcast_channel_mask	R	xx:....xx	- Not masked (0) - Masked (1)	Excluded channel mask applied to broadcast frequency hopping	
wisun.neighbor_table_size	R/W	integer	1 to 254	Size of the neighbor table, excluding temporary entries and RPL parents	
wisun.keychain	R/W	integer	- automatic (0) - built-in (1) - NVM (2)	Keychain to use for device credentials	
wisun.keychain_index	R/W	integer	0 to 255	Device credential index to use for built-in keychain	
wisun.gtk1	R/W	string	GTK as hex, 'none' to disable	GTK1 to install	
wisun.gtk2	R/W	string	GTK as hex, 'none' to disable	GTK2 to install	
wisun.gtk3	R/W	string	GTK as hex, 'none' to disable	GTK3 to install	
wisun.gtk4	R/W	string	GTK as hex, 'none' to disable	GTK4 to install	
wisun.gak1	R	string		GAK1 in use	
wisun.gak2	R	string		GAK2 in use	
wisun.gak3	R	string		GAK3 in use	
wisun.gak4	R	string		GAK4 in use	
wisun.lgtk1	R/W	string	LGTK as hex, 'none' to disable	GTK to install	
wisun.lgtk2	R/W	string	LGTK as hex, 'none' to disable	GTK to install	
wisun.lgtk3	R/W	string	LGTK as hex, 'none' to disable	GTK to install	
wisun.lfn_gak1	R	string		LGAK1 in use	
wisun.lfn_gak2	R	string		LGAK2 in use	
wisun.lfn_gak3	R	string		LGAK3 in use	
wisun.ipv6_prefix	R/W	string	IPv6 address	IPv6 prefix for DODAG	
wisun.dhcpv6_server	R/W	string	IPv6 address	IPv6 address of the external DHCPv6 server	

GTK/LGTK Management

By default, Wi-SUN Border Router demo is configured to use a fixed GTK1 and LGTK1 to simplify testing and make it easier to decrypt captured PTI packets in Wireshark. To utilize a randomly generated (L)GTK, the corresponding wisun setting entry must be set to none before the first network start.

app Section Settings

The settings in the *app* section relate to the application options. A detailed setting list is available below.

Variable	R/W	Type	Values	Description
app.printable_data_length	R/W	integer	0: received socket data is not printed 1 - 64: amount of characters per line	If enabled, received socket data is printed
app.printable_data_as_hex	R/W	integer	0: print received socket data as ASCII 1: print received socket data as hex	Output type for received socket data
app.autostart	R/W	bool	0: disable border router start after reset 1: enable border router start after reset	Enable or disable autostart
app.pti_state	R/W	bool	0: disable PTI 1: enable PTI	Disable or enable PTI

ping Section Settings

The settings in the *ping* section relate to the ping command. A detailed setting list is available below.

Variable	R/W	Type	Values	Description
ping.identifier	R/W	integer	0 - 65535	Identifier to aid in matching ICMPv6 Echo Requests and Replies
ping.sequence_number	R/W	integer	0 - 65535	Sequence number to aid in matching ICMPv6 Echo Requests and Replies
ping.packet_length	R/W	integer	8 - 2048	Length of ICMPv6 Echo Request packet in bytes, including the header
ping.pattern	R/W	string		Repeating pattern used to fill ICMPv6 Echo Request payload

wifi Section Settings

The settings in the *wifi* section relate to the wifi connectivity. This section is only available when application is compiled with Wi-Fi connectivity component. A detailed setting list is available below.

Variable	R/W	Type	Values	Description
wifi.ssid	R/W	string	up to 32 ASCII characters. Use double quotes (") to include spaces	Name of the Wi-Fi network
wifi.security	R/W	integer	- WPA2_TKIP (2): WPA2 security with TKIP encryption - WPA2_CCMP (3): WPA2 security with CCMP encryption	Security type
wifi.passphrase	R/W	string	up to 32 ASCII characters.	Pre-Shared Key
wifi.wlan_state	R	string	- connected - disconnected	WLAN state
wifi.channel_number	R	integer	1 to 14	Channel number of connected Access-Point
wifi.mac_address	R	string		MAC Address
wifi.ipv6_address	R	string		IPv6 Address

Wi-Fi Connectivity

Wi-Fi Backhaul Connectivity is based on the SiWx91x™ chipset and the WiSeConnect™ SDK v3.x.

Follow the [Getting Started Guide](#) to configure the SiWN917 as a Network Co-Processor (NCP) and install the component *Wi-Fi connectivity for Wi-SUN Border Router* to enable additional features in the Wi-SUN Border Router application. This document is limited to additional commands and settings, a more detailed documentation can be found here [SoC Border Router with Wi-Fi Backhaul](#).

Wi-Fi Commands

To see the available commands, enter the following command in the console:

```
> wifi help
```

Following is an extended description of each command.

Command	Description	Example
wifi connect	Connect the Border Router to the Wi-Fi Access Point	> wifi connect
wifi disconnect	Disconnect the Border Router from the Wi-Fi Access Point	> wifi disconnect

To connect the Wi-SUN Border Router to the Wi-Fi Access Point, select the security type with *wifi* settings, enter the SSID and the PSK credential.

RTT traces

To control RTT traces, set the trace level using `wisun set_trace_level [group_id],[trace level];`

All RTT trace groups and trace levels are listed in `sl_wisun_types.h/sl_wisun_trace_group_t`

```
typedef enum {
    SL_WISUN_TRACE_GROUP_MAC      = 0,      ///< Mac
    SL_WISUN_TRACE_GROUP_NW       = 1,      ///< Network
    SL_WISUN_TRACE_GROUP_LLC      = 2,      ///< LLC
    SL_WISUN_TRACE_GROUP_6LO      = 3,      ///< 6lowpan
    SL_WISUN_TRACE_GROUP_IPV6     = 4,      ///< IPV6
    . . .
    SL_WISUN_TRACE_GROUP_WSIE     = 35,     ///< Wi-SUN IE
    SL_WISUN_TRACE_GROUP_CONFIG   = 36,     ///< Configuration
    // 36 to 63 reserved for future used
    SL_WISUN_TRACE_GROUP_INT      = 63,     ///< Internal usage
    SL_WISUN_TRACE_GROUP_COUNT    = 64      ///< Max number of trace group in this enum
} sl_wisun_trace_group_t;

. . .
typedef enum {
    ///< No trace
    SL_WISUN_TRACE_LEVEL_NONE     = 0,
    ///< Error only
    SL_WISUN_TRACE_LEVEL_ERROR    = 1,
    ///< Warning + error
    SL_WISUN_TRACE_LEVEL_WARN     = 2,
    ///< Info + warning + error
    SL_WISUN_TRACE_LEVEL_INFO     = 3,
    ///< Debug + info + warning + error
    SL_WISUN_TRACE_LEVEL_DEBUG    = 4,
} sl_wisun_trace_level_t;
```

Example: to set 'Network' to 'Debug' and 'IPV6' to 'Warning', use:

`wisun set_trace_level 1,0;4,4;` or `wisun set_trace_level 1,none;4,debug;`

Troubleshooting

Before programming the radio board mounted on the WSTK, ensure the power supply switch is in the AEM position (right side), as shown.



Resources

- [Wi-SUN Stack API documentation](#)
- [AN1330: Wi-SUN Mesh Network Performance](#)
- [AN1332: Wi-SUN Network Setup and Configuration](#)
- [AN1364: Wi-SUN Network Performance Measurement Application](#)
- [QSG181: Wi-SUN Quick-Start Guide](#)
- [UG495: Wi-SUN Developer's Guide](#)

Report Bugs & Get Support

You are always encouraged and welcome to ask any questions or report any issues you found to us via [Silicon Labs Community](#).