BIOT LoRa E5

Release 1.0.0

Martin Stusek

Feb 02, 2023

Chapter 1

Modules Package

1.1 modules.but_board module

1.2 modules.lora_E5 module

class `modules.lora_E5.`**`LoRaE5`** (*rx_pin*, *tx_pin*, *speed*, *debug=True*)

Bases: `object`

Construct a LoRaE5 class designed to control LoRa E5 HF module

`check_link` ( ) → str

Use to send LinkCheckReq mac command to check Link status between modem and server

Returns link status

`disable_channel` (*channel: int*) → str

Disable LoRaWAN channel

Parameters **`channel`** – channel nuber

Returns channel configuration

`enable_channel` (*channel: int*) → str

Enable LoRaWAN channel

Parameters **`channel`** – channel nuber

Returns channel configuration

`factory_reset` ( ) → str

Do factory reset

Returns received data (OK)

`get_adr` ( ) → str

Get status of ADR

Returns ADR status

`get_all_channels` ( ) → str

Query channels configuration

Returns channels configuration

`get_band_scheme` ( ) → str

Get data rate scheme for current band (EU868)

Returns data rate plan

`get_channel` (*channel: int*) → str
Get single channel configuration

Parameters **`channel`** – channel value

Returns channel configuration

`get_counters` ( ) → str
Get uplink and downlink counter values

Returns counters value

`get_dr` ( ) → str
Get current data rate

Returns data rate

`get_enabled_channels` ( ) → str
Get list of enabled channels

Returns list of active channels

`get_id` ( ) → str
Use to check the ID of the LoRaWAN module

Returns module ID

`get_port` ( ) → str
Get LoRaWAN port used for transmission

Returns port

`get_power` ( ) → str
Get TX power

Returns TX power

`get_power_map` ( ) → str
Get TX power map

Returns power map

`get_rep_limit` ( ) → str
Get maximum number of repetitions for unconfirmed messages

Returns number of repetitions

`get_ret_limit` ( ) → str
Get maximum number of retransmissions for confirmed message

Returns number of retransmissions

`get_rx2` ( ) → str
Query RXWIN2 configuration

Returns RXWIN2 configuration

`get_rx_delay` ( ) → str
Get RX windows delay

Returns RX windows delay

`get_version` ( ) → str
Check firmware version

Returns FW version

`module_reset` ( ) → str

Use to reset LoRaWAN module

Returns received data

`read_data` (*multi: bool = False*) → str | None

Read data received from module via UART

Parameters **`multi`** – defines if multiple lines will be read

Returns received data

`read_n_times` (*times: int*) → str

Read from multiple lines responses from UART

Parameters **`times`** – number or read repetitions total time defined by the *timeout* and *timeout_char*

times from *UART* object :return: received data

`send_ascii` (*data: str, confirmed: bool = False*) → str

Send LoRaWAN message in ASCII format

Parameters

- **`data`** – data to send
- **`confirmed`** – confirmed/unconfirmed message

Returns received data

`send_hex` (*data: str | None = None, confirmed: bool = False*) → str

Send LoRaWAN message in HEX format

Parameters

- **`data`** – data to send
- **`confirmed`** – confirmed/unconfirmed message

Returns received data

`send_raw_command` (*command: str*) → None

Send command directly to LoRaWAN module via UART

Parameters **`command`** – AT command to transmit

Returns None

`set_adr` (*enable: bool = False*) → str

Control ADR functionality

Parameters **`enable`** – enable/disable

Returns new ADR status

`set_channel` (*channel: int, freq: int, drmin: int, drmax: int*) → str | None

Set channel configuration

Parameters

- **`channel`** – channel value
- **`freq`** – frequency
- **`drmin`** – minimum data rate
- **`drmax`** – maximum data rate

Returns new channel configuration

`set_counters` (*up: int*, *down: int*) → str

Set uplink and downlink counter values

Parameters
- **`up`** – uplink counter value
- **`down`** – downlink counter value

Returns set value for both counters

`set_dr` (*dr: int*) → str | None

Set data rate value

Parameters **`dr`** – data rate ranging from 0 to 15

Returns new data rate value

`set_dr_band` (*band: str*) → str | None

Set data rate scheme for current band (EU868)

Parameters **`band`** – frequency band (currently only EU868 supported)

Returns data rate plan

`set_duty_cycle` (*max_dc: int*) → str

Set maximum duty cycle

Parameters **`max_dc`** – duty cycle

Returns duty cycle value

`set_etsi_duty_cycle` (*enabled: bool = True*) → str

Set ETSI defined duty cycle limitation

Parameters **`enabled`** – enable/disable

Returns duty cycle value

`set_id` (*mode: str*, *value: int*) → str | None

Use to set the ID of the LoRaWAN module

Parameters
- **`mode`** – ID to change DevAddr, DevEui, or AppEui
- **`value`** – ID value

Returns new ID

`set_key` (*key: str*, *value: int*) → str | None

Set encryption keys

Parameters
- **`key`** – type of key (NWKSKEY/APPSKEY/APPKEY)
- **`value`** – key value

Returns newly set key

`set_mode` (*mode: str*) → str | None

Set LoRaWAN operational mode

Parameters **`mode`** – operational mode (TEST, LWOTAA, LWABP)

Returns set mode

`set_port` (*port: int*) → str | None

Set LoRaWAN port

Parameters **`port`** – numeric value from 0 to 255

Returns new port

set_power (*power: int*) → str | None

Set TX power

Parameters **power** – power value

Returns set TX power

set_rep_limit (*rep: int*) → str | None

Set maximum number of repetitions for unconfirmed messages

Parameters **rep** – number of repetitions

Returns set number of repetitions

set_ret_limit (*ret: int*) → str | None

Set maximum number of retransmissions for confirmed message

Parameters **ret** – number of retransmissions

Returns set number of retransmissions

set_rx2 (*freq: int*, *dr: int*) → str | None

Set RXWIN2 configuration

Parameters

- **freq** – window frequency
- **dr** – data rate

Returns RXWIN2 new configuration

set_rx_delay (*window: str*, *delay: int*) → str | None

Set RX windows delay

Parameters

- **window** – RX window (RX1/RX2/JRX1/JRX2)
- **delay** – delay in ms

Returns RX windows configuration

test_at ( ) → str

Used to test if connection of module is OK. This is a dummy command just like other common AT modules

Returns received data

- genindex
- modindex
- search

Index