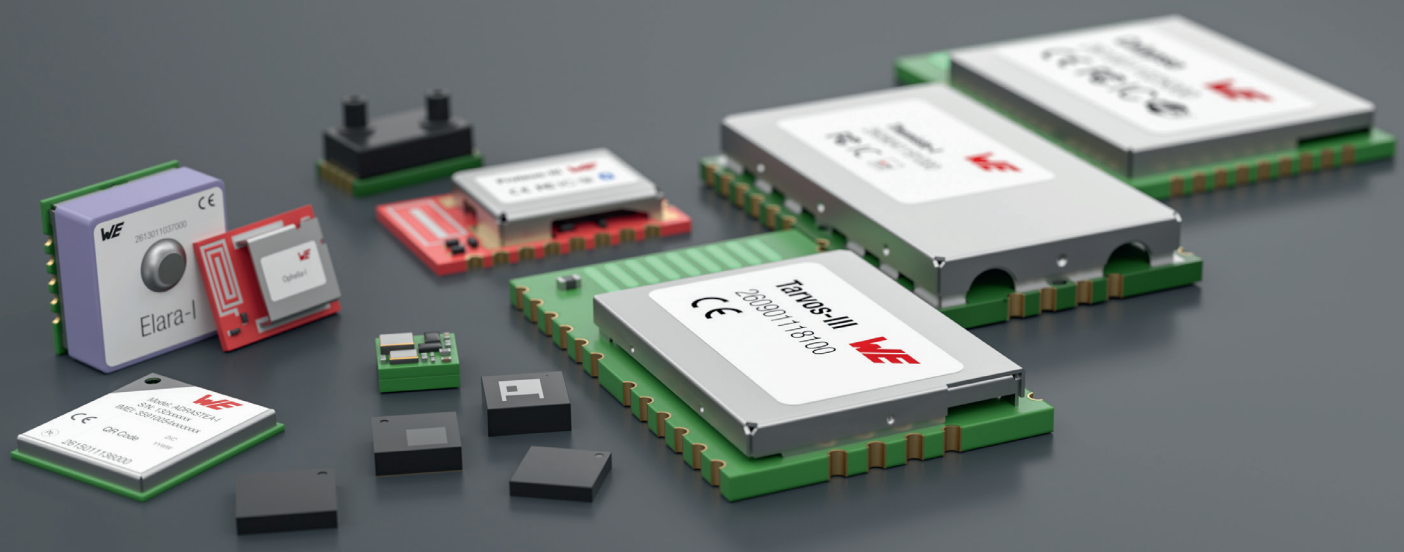

User manual

Daphnis-I

2618011181000

Version 1.3

March 19, 2025

WÜRTH ELEKTRONIK MORE THAN YOU EXPECT

****

MUST READ

Check for firmware updates

Before using the product, make sure you use the most recent firmware version, data sheet, and user manual. This is especially important for Wireless Connectivity products that were not purchased directly from Würth Elektronik eiSos. A firmware update on these respective products may be required.

We strongly recommend including the possibility of a firmware update in the customer system design.

Revision history

Manual version	Firmware version	Hardware version	Notes	Date
1.0	1.3.0	2.0	• Initial release of the manual	March 2024
1.1	1.3.0	2.0	• Updated Long-Range-WAN to LoRaWAN® • Updated Long-Range to LoRa® • Added information about *VDDA* and *VBAT* to chapter `Pinout` • Added chapter `Information for explosion protection` • Updated chapter `General labeling information` • Improved description of `AT+SEND`, `+RXINFO` and `+TXCONF` in chapter `AT commands`	September 2024

1.2	1.4.0	2.0	• Updated pins in chapter Pinout • TX current consumption updated in chapter Power consumption • New features of firmware version 1.4.0. Please refer to chapter Firmware history • Updated symbol of Daphnis-I in schematics chapter EV-Board • Corrected frequency band range in chapter Radio characteristics	March 2025
1.3	1.4.0	2.0	• Added chapter Known issues	March 2025

Abbreviations

Abbreviation	Name
ABP	Activation By Personalization
ADC	Analog to Digital Converter
CAD	Channel Activity Detection
CMOS	Complementary Metal Oxide Semiconductor
DAC	Digital to Analog Converter
DevEUI	Device Extended Unique Identifier
EUI	Extended Unique Identifier
ESD	Electrostatic Discharge
EV	Evaluation
FSK	Frequency Shift Keying
FW	Firmware
GND	Ground
GPS	Global Positioning System
GPIO	General Purpose Input Output
GUI	Graphical User Interface
I^2C	Inter Integrated Circuit
IP	Internet Protocol
JTAG	Joint Test Action Group
LBT	Listen Before Talk
LED	Light Emitting Diode
LNA	Low Noise Amplifier
MCU	Micro Controller Unit
OTAA	Over The Air Activation
P2P	Peer to Peer
PA	Power Amplifier
PC	Personal Computer
PCB	Printed Circuit Board
PCN	Product Change Notification
RAM	Random Access Memory
RF	Radio Frequency
RTC	Real Time Clock
RX	Receiver
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory

SWD	Serial Wire Debug
TCP	Transmission Control Protocol
TCXO	Temperature Compensated Crystal Oscillator
TTL	Transistor Transistor Logic
TTN	The Things Network
UART	Universal Asynchronous Receiver Transmitter
UMRF	Ultra Miniature Radio Frequency
VDD	Supply voltage
WAN	Wide Area Network
WE	Würth Elektronik eiSos GmbH & Co. KG

Contents

Overview of helpful application notes

Application note ANR008 - Wireless Connectivity Software Development Kit

http://www.we-online.com/ANR008
To ease the integration of the Würth Elektronik eiSos radio modules into an application, Würth Elektronik eiSos offers the corresponding Software Development Kit (SDK) for most commonly used host processors. This SDK contains drivers and examples in C-code to communicate with the corresponding radio module. This application note shows which SDKs are available and describes how to download and use them.

Application note ANR010 - Range estimation

http://www.we-online.com/ANR010
This application note presents the two most used mathematical range estimation models, Friis and two ray ground reflection, and its implementation in the range estimation tool of the RED-EXPERT.

Application note ANR015 - From 868 MHz To 915 MHz

http://www.we-online.com/ANR015
Due to radio regulation laws in Europe and North America different frequencies are used in the sub-GHz range for radio data transmission. Therefore Würth Elektronik eiSos offers each product operating in the European 868 MHz range an counterpart operating in the American 915 MHz range. This application note shows which modules are comparable, where they differ and what to consider when switching from 868 to 915 MHz and vice versa.

1 Introduction

Daphnis-I is a low-power, long-range transceiver module designed for LoRaWAN® and P2P applications. Daphnis-I is based on the STM32WLE5CCU6 chip.

Daphnis-I module supports two primary modes of operation:

- LoRaWAN® network end node mode.
  - In this mode, the LoRaWAN® gateway is required to use Daphnis-I. Refer to chapter 5.3.1 for more information about LoRaWAN® setup and selecting a compatible gateway.
  - Daphnis-I module complies with Class A, Class B, and Class C of LoRaWAN® 1.0.4 specifications.
- P2P mode.
  - In this mode of operation, module uses the the pre-loaded WE-ProWare radio stack and offers the ability to communicate in different topologies such as point-to-point, star and mesh. Refer to chapter 5.4 for more information about P2P.

Compact 15 x 16 x 3 mm design allows the module to fit into small-size applications. The low power consumption of the Daphnis-I module makes it suitable for battery-powered applications.

The Daphnis-I module can be configured and taken into operation using AT commands via the Application UART interface.

The Daphnis-I EV-Kit and "WE UART Terminal" tool [1] allow getting started with the module and testing its functionalities. The EV-Board can be connected to a USB port of a PC. The EV-Board also represents our reference design. For further information refer to the EV-Board manual [2].

The module comes with the declaration of conformity (CE) and the module is compliant to RoHS and REACH.

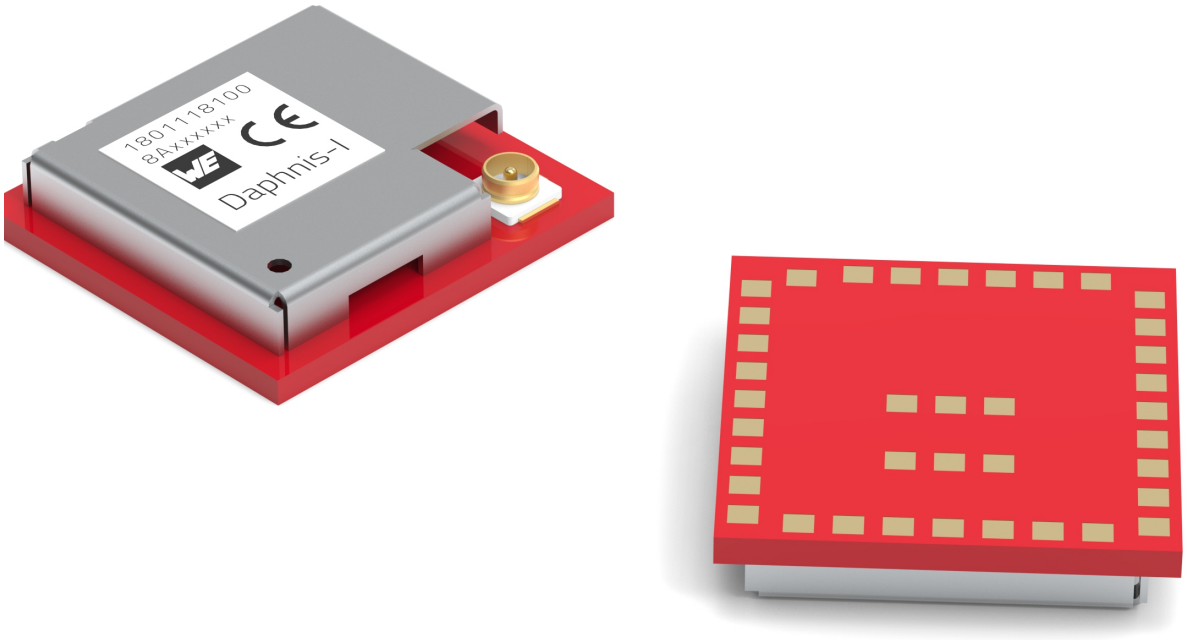

Figure 1: Daphnis-I

1.1 Block diagram

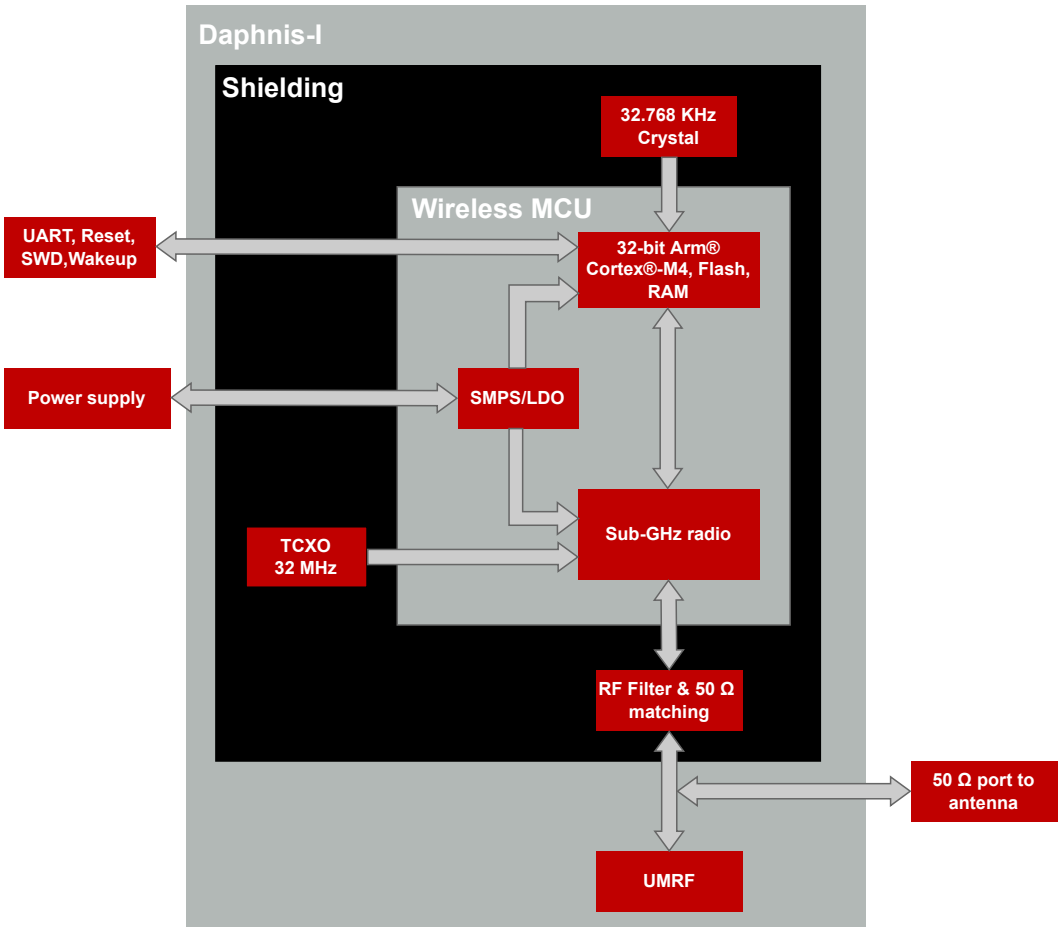

Figure 2: Block diagram

1.2 Ordering information

WE order code	Description
2618011181000	Daphnis-I radio module in T&R packaging
2618019381001	EV-Kit for Daphnis-I

Table 1: Ordering information

2 Electrical and radio specifications

Unless otherwise stated, all the values given in the manual were measured on the Daphnis-I EV-Board with the following conditions: T = 25 °C, VDD= 3.3 V, PER 1% and internal DCDC in use.

2.1 Operating conditions

Parameter	Direction	Min.	Typ.	Max.	Unit
Supply voltage (VDD)	Supply	2.0	3.3	3.6	V
Operating temperature		-40	25	85	°C

Table 2: Operating conditions

An instable supply voltage may significantly decrease the radio performance and stability.

2.2 Absolute maximum ratings

Parameter	Min.	Max.	Unit
Supply voltage (VDD)	-0.3	+3.9	V
Voltage on any digital pin	-0.3	+3.9	V
Input RF level when in RX operation mode		0	dBm
Flash endurance	10 000		Write/erase cycles

Table 3: Absolute maximum ratings

2.3 Power consumption

Description	Test conditions	Typ.	Unit
TX current consumption	50 Ω load, +14dBm output power configuration, SF12BW125	26.5	mA
RX current consumption	50 Ω load, LNA on, SF12BW125	6.9	mA
`Idle state` current consumption	RTC enabled, full SRAM and peripheral retention	1.5	µA
`Sleep mode` current consumption	RTC disabled, backup registers retained	63.9	nA

Table 4: Current consumption

`Sleep mode` and `Idle state` currents increase significantly for temperatures above 30 °C.

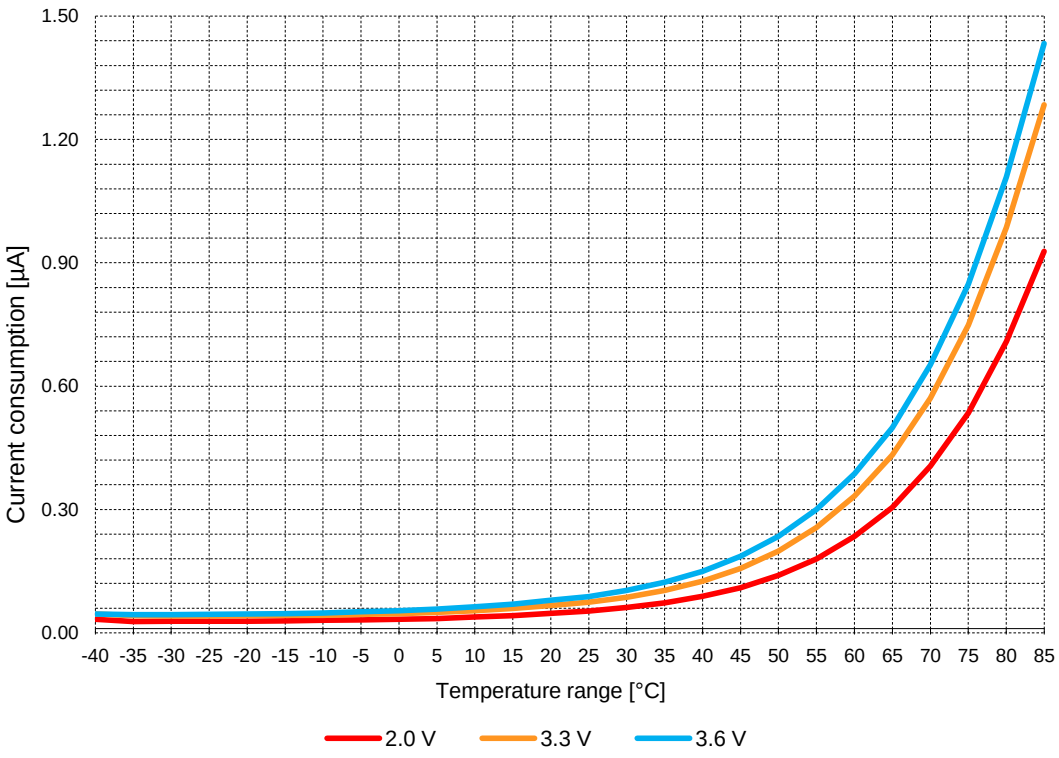

Figure 3: Current consumption during `Sleep mode`

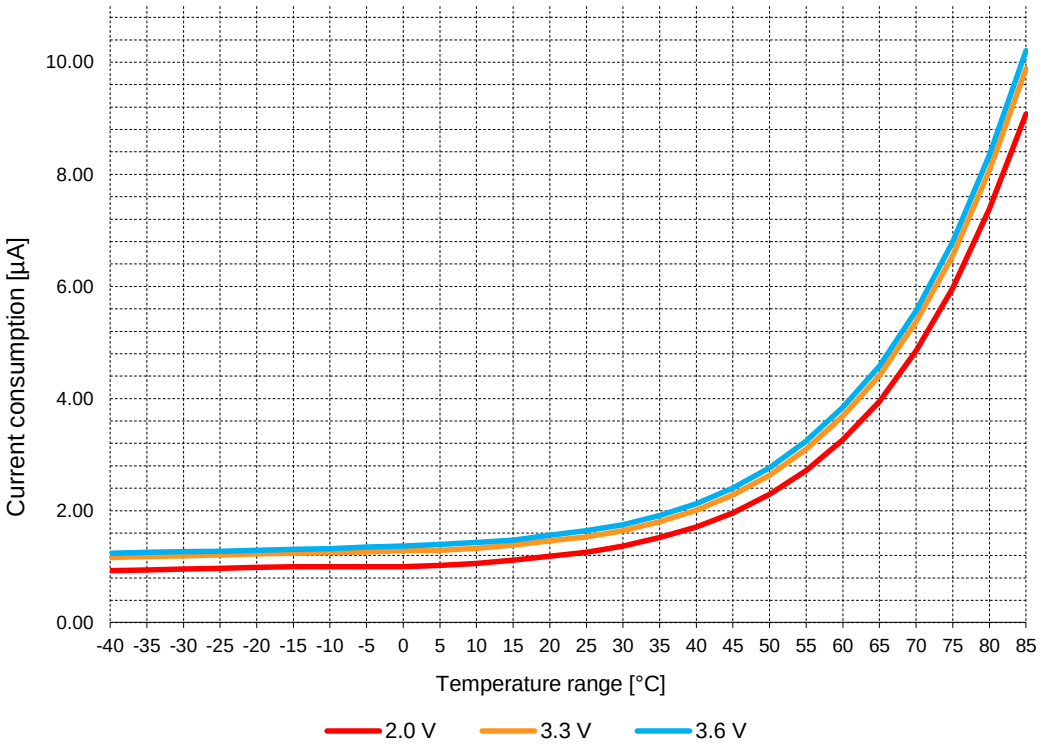

Figure 4: Current consumption during `Idle state`

2.4 Radio characteristics

The following factors and tasks are critical for the performance of the Daphnis-I module:

- External 868 MHz antenna for signal reception and transmission,
- RF trace delivering the signal from the external antenna to the RF pad of the Daphnis-I module,
- Minimization of external and environmental effects.

Unless noted otherwise, the Daphnis-I EV-Board is in static mode.

Description	Min	Typ.	Max	Unit
Frequency band	863.0		870.0	MHz
Radio data rate	0.25		50	kbit/s
RX sensitivity LoRa® (SF12BW125) LoRa® (SF7BW125) LoRa® (SF7BW250) FSK (50 kbps)		 -138.0 -124.6 -121.8 -107.5		dBm
TX power		+13.4		dBm

Table 5: Radio characteristics

RX sensitivity values measured with internal LNA activated.

2.5 Pin characteristics

Pin specifications listed in the table below. For more details refer to [3].

Parameter	Min.	Typ.	Max.	Unit
Input high-level voltage	0.7 ×VDD		VDD	V
Input low-level voltage			0.3 ×VDD	V
Output low-level voltage, 4 mA, 8 mA, 10 mA, VDD ≥2.0V			0.4	V
Output low-level voltage, 20 mA, VDD ≥2.7 V			1.3	V
Output high-level voltage, 4 mA, VDD ≥2.0 V	VDD-0.45 V			V
Output high-level voltage,CMOS port, 8 mA,VDD ≥2.7 V	VDD-0.4 V			V
Output high-level voltage,TTL port, 8 mA, VDD ≥2.7 V	2.4			V
Output high-level voltage, 20 mA, VDD ≥2.7 V	VDD-1.3 V			V
Internal pull-up resistance	25	40	55	kΩ
Internal pull-down resistance	25	40	55	kΩ

Table 6: Pin characteristics

3 Pinout

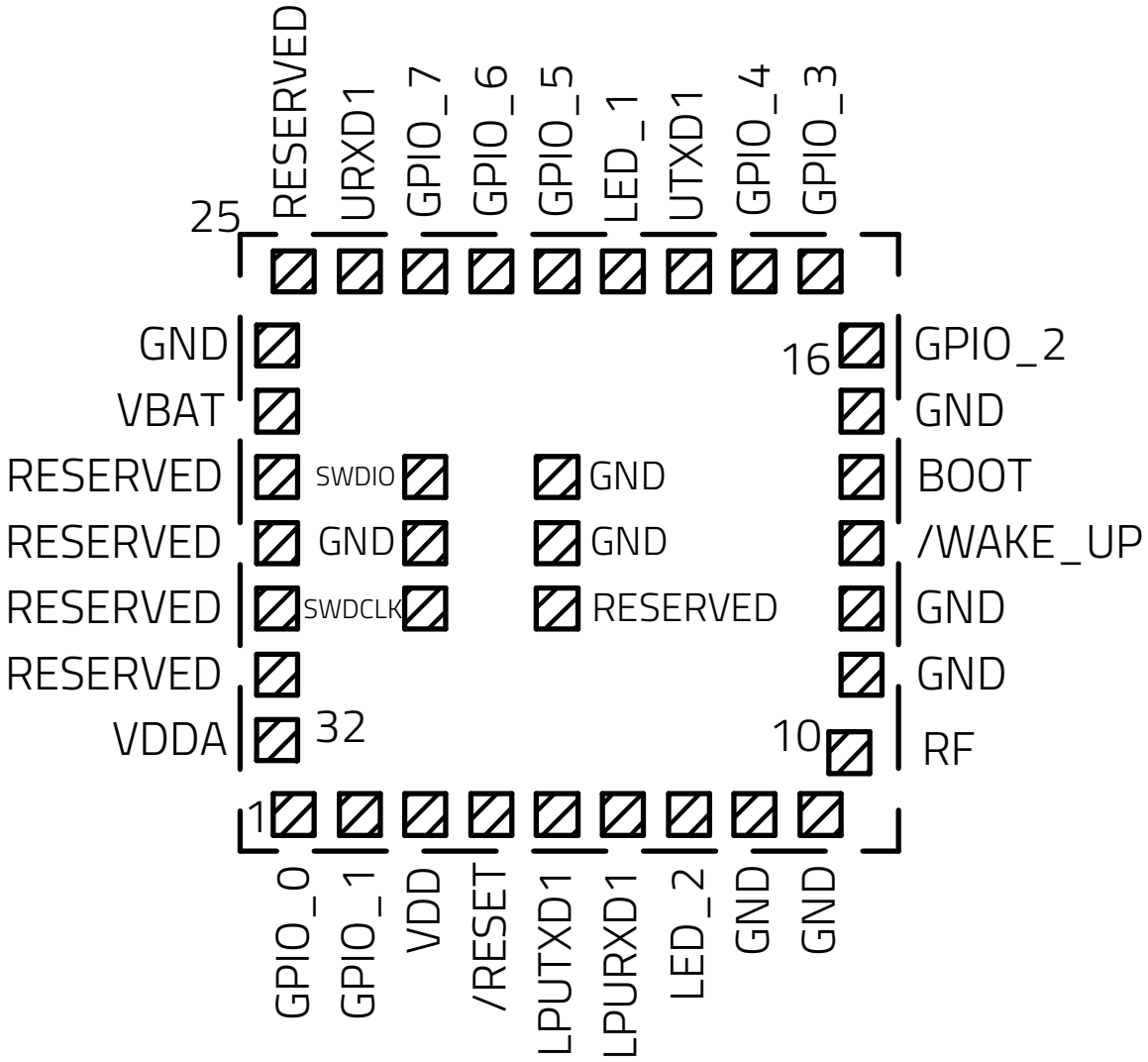

Figure 5: Pinout (top view)

Pin Number	STM32WLE5CCU6	Designation	I/O	Description
1	PB7	*GPIO_0* [1]	GPIO	Pin for local and remote GPIO access, refer to chapter *GPIO* pins
2	PB8	*GPIO_1* [1]	GPIO	Pin for local and remote GPIO access, refer to chapter *GPIO* pins
3	–	*VDD*	Supply	Power supply for module
4	–	*/RESET*	Input	Reset, refer to chapter */RESET* pin
5	PA2	*LPUTXD1*	Output	Application UART transmit
6	PA3	*LPURXD1*	Input	Application UART receive, refer to chapter *LPURXD1* pin
7	PA4	*LED_2*	Output	Refer to chapter *LED* pins
8	–	*GND*	Supply	Negative supply voltage
9	–	*GND*	Supply	Negative supply voltage
10	–	*RF*	RF	50 Ohm RF connection to transceiver, refer to chapter *RF* pin
11	–	*GND*	Supply	Negative supply voltage
12	–	*GND*	Supply	Negative supply voltage
13	PA0	*/WAKE_UP*	Input	Refer to chapter */WAKE_UP* pin
14	PH3-BOOT0	*BOOT*	Input	Refer to chapter *BOOT* pin
15	–	*GND*	Supply	Negative supply voltage
16	PA5	*GPIO_2* [1]	GPIO	Pin for local and remote GPIO access, refer to chapter *GPIO* pins
17	PA6	*GPIO_3* [1]	GPIO	Pin for local and remote GPIO access, refer to chapter *GPIO* pins
18	PB2	*GPIO_4* [1]	GPIO	Pin for local and remote GPIO access, refer to chapter *GPIO* pins
19	PA9	*UTXD1*	Output	Bootloader UART transmit
20	PA7	*LED_1*	Output	Refer to chapter *LED* pins
21	PA8	*GPIO_5* [1]	GPIO	Pin for local and remote GPIO access, refer to chapter *GPIO* pins

22	PA12	*GPIO_6* [1]	GPIO	Pin for local and remote GPIO access, refer to chapter *GPIO pins*
23	PB6	*GPIO_7* [1]	GPIO	Pin for local and remote GPIO access, refer to chapter *GPIO pins*
24	PA10	*URXD1*	Input	Bootloader UART receive, Refer to chapter *URXD1 pin*
25	PA11	*RESERVED* [2]	–	Reserved
26	–	*GND*	Supply	Negative supply voltage
27	–	*VBAT* [3]	Input	Connect to backup battery, refer to chapter *VBAT pin*
28	PA15	*RESERVED* [2]	–	Reserved
29	PB4	*RESERVED* [2]	–	Reserved
30	PB5	*RESERVED* [2]	–	Reserved
31	PB3	*RESERVED* [2]	–	Reserved
32	–	*VDDA* [3]	Supply	Positive supply voltage analog (VDDA), refer to chapter *VDDA pin*
B1	PA14	*SWDCLK*	Input	Serial wire clock (SWD Interface)
B2	PA1	*RESERVED* [2]	–	Reserved
B5	PA13	*SWDIO*	I/O	Serial wire data (SWD Interface)
B3, B4, B6	–	*GND*	Supply	Negative supply voltage

Table 7: Pinout

[1] Do not connect, if not needed.
[2] Shall be soldered and electrically unconnected.
[3] If not used, connect to *VDD*.

3.1 *VDDA* pin

The *VDDA* pin serves as the external analog power source for ADCs, DACs, the voltage reference buffer and comparators.

The functionalities associated to *VDDA* are not implemented as a part of the standard firmware. *VDDA* shall be connected to *VDD*.

3.2 *VBAT* pin

The *VBAT* can be used to power the RTC, the 32.768 kHz oscillator and backup registers from an external battery or an external super-capacitor, when VDD is not present.

The *VBAT* pin shall be connected to *VDD*, if no backup supply is connected to the pin.

3.3 */RESET* pin

The */RESET* pin of the module is an active-low logic input. */RESET* pin is internally connected to a permanent pull-up resistor. No external pull-up is needed.

- Internally pulled HIGH: Turn on module regulators and power up the module.
- Pulled LOW: Turn off module regulators and shutdown the module.

The user must ensure that the voltage level on the /RESET pin is pulled below the 0.3 x VDD level, otherwise the reset is not recognized by the module.

3.4 */WAKE_UP* pin

The host connected to this pin should always keep the state of this pin HIGH and pull it to LOW to immediately wake up the module from `Sleep mode`.

- Pulled LOW: Wake up internal MCU of module from sleep mode.
- Pulled HIGH: Allow internal MCU of module to go into sleep mode.

This pin is internally pulled-up during `Sleep mode`, otherwise it is high impedance mode.

3.5 *RF* pin

The *RF* pin provides a 50 Ω RF signal. This pin must be connected to the external antenna via 50 Ω feed line.

If the *RF* pin is used, do not connect any cable or adapter to the UMRF connector.

3.6 *BOOT* pin

The firmware update of the module over the UART interface is managed by the secure bootloader implemented as a part of the standard firmware.

The *BOOT* pin must be held HIGH during the boot-up process to activate the secure UART bootloader. Otherwise, the module starts the application firmware.

The *LED_1* will toggle to indicate that the module is in the bootloader mode and waiting for the firmware file to be sent.

Due to the internal pull-down resistor, the module by default starts in the application firmware, after a power up or reset event.

The *BOOT* pin is internally connected to a 10 kΩ pull-down resistor.

3.7 *LPURXD1* pin

This pin is internally pulled HIGH in application mode and it is in high impedance in bootloader mode.

3.8 *URXD1* pin

This pin is high impedance in application mode and it is open-drain in bootloader mode. During the bootloader mode the host MCU should guarantee that it is pulled high in idle state.

3.9 *LED* pins

The pins *LED_1* and *LED_2* are active HIGH.

Mode	*LED_1*	*LED_2*
Bootloader	Blinking (Turns ON for 3000 ms, Turns OFF for 3000 ms)	OFF
Application LoRaWAN® mode	Turns ON for 500 ms and then turns OFF whenever data has been sent to or received from the network	Turns ON when the module successfully joins the network
Application P2P mode	Indicates RF transmission	Indicates RF reception

Table 8: LED pins behavior

LED_2 is in high impedance mode in bootloader mode.These pins can source up to 8 mA, for more details refer to `Pin characteristics`.

3.10 *GPIO* pins

Pin Number	GPIO_ID	Supported functions
1	0	Input (with and without pull resistor) or output
2	1	Input (with and without pull resistor) or output
16	2	Input (with and without pull resistor) or output
17	3	Input (with and without pull resistor) or output
18	4	Input (with and without pull resistor) or output
21	5	Input (with and without pull resistor) or output
22	6	Input (with and without pull resistor) or output
23	7	Input (with and without pull resistor) or output

Table 9: GPIO pins

4 Quick start

4.1 Minimal pin connections

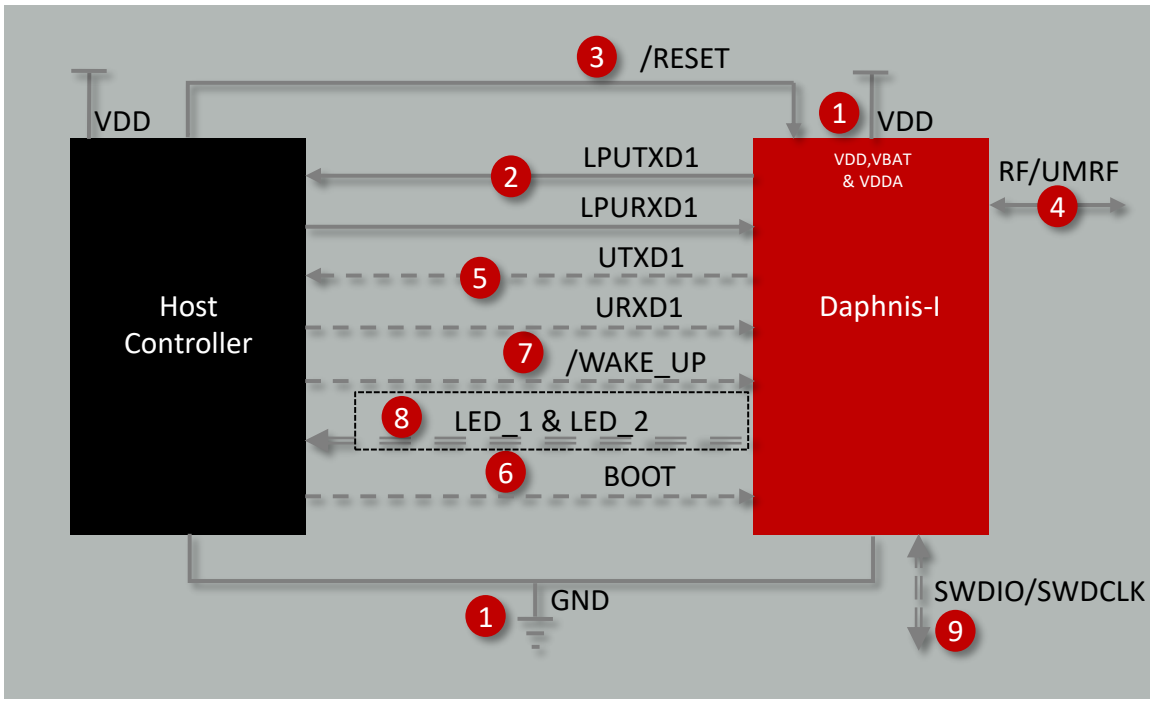

Figure 6: Minimal pin connections

The above image shows the steps to be performed to integrate the Daphnis-I into a custom end device.

1. Supply voltage and ground
   Connect pins *VDD*, *VBAT*, *VDDA* and *GND* to supply the radio module with power.

2. Application UART serial interface
   Connect the UART pins *LPUTXD1* and *LPURXD1* to the host to control the module via host. No flow control needed.

3. Reset
   Connect the */RESET* pin to the host to allow a hard reset of the module.

4. Antenna connection
   The antenna connection must be performed either via *RF* pin or via UMRF connector.

5. Bootloader UART serial interface
   Connect the UART pins *UTXD1* and *URXD1* to the host or to testpads to perform firmware updates. No flow control pins are needed.

6. Boot
   Connect the *Boot* pin to the host to set the module into bootloader mode to perform firmware updates.

7. (Optional) Wake up
   Connect the */WAKE_UP* pin to the host to wake up the module when it is in `Sleep mode`.

8. (Optional) Status indication
   Connect the *LED_1* and *LED_2* pin to the host controller to allow easy indication of the status of the module.

9. (Optional) Flash and debug interface
   In case of custom firmware development, it is recommended to additionally have the pins *SWDIO* and *SWDCLK* accessible in order to support a fail-safe update of firmware. A standard socket on the customer's PCB for connecting a flash adapter can be useful for debugging purposes (e.g. a JTAG 2*10 pin header with 2.54 mm pin-to-pin distance).

4.2 Antenna connection

Daphnis-I's external antenna connection allows the user to choose between *RF* pin or UMRF connector.

4.2.1 *RF* pin

The *RF* pin provides a 50 Ω RF signal. This pin must be connected to the external antenna via 50 Ω feed line.

If the *RF* pin is used, do not connect any cable or adapter to the UMRF connector.

4.2.2 UMRF connector

The ultraminiature RF connector placed on the top of the module can be used to connect an external antenna via a 50 Ω *UMRF*[1] *cable*.

If the connector is used, neither the RF pin shall be electrically connected, nor the pad shall be connected to an open trace. For mechanical stability it shall be connected to a solder pad.

[1] The UMRF connector is equivalent to MHF 1 and U.FL.

4.3 LoRaWAN® mode quick start example

In this example, it is demonstrated how the Daphnis-I joins a LoRaWAN® network and exchanges data with it. For this purpose a public network server provided by *TTN* is used.

TTN is a free public network server, where all members agree that everyone can use their gateways to communicate with their applications. This means that when you connect your gateway to TTN the usage of the gateway is not limited to your end nodes but to all nodes that are using TTN within the range of the gateway.

Note that depending on the LoRaWAN® network server of your choice, the steps to set it up may differ. Nevertheless, after the network is set up, the commands that the MCU has to send to the Daphnis-I to join the network and to exchange data with it, will be the same.

It is recommended to use the Daphnis-I EV-Kit and WE UART Terminal [1] PC tool to take the module into operation.

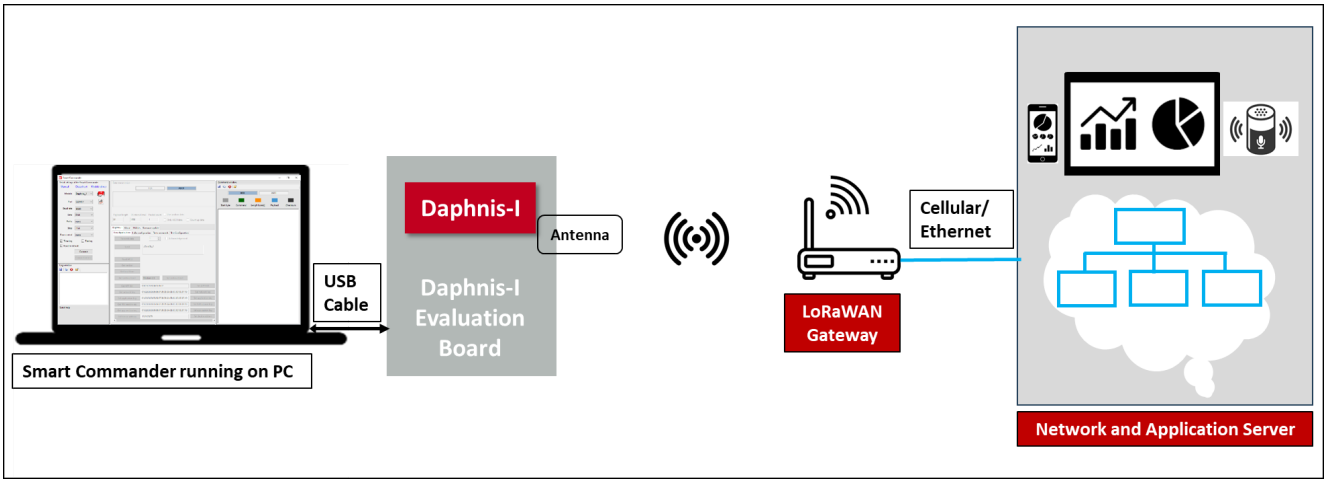

Figure 7: Daphnis-I quick start setup

4.3.1 Prerequisites

1. Daphnis-I EV-Kit.
2. Computer with a serial terminal emulator. The use of Würth Elektronik eiSos's WE UART Terminal [1] PC tool is recommended.
3. A public or private LoRaWAN® gateway with connection to the network server.
4. An account on TTN.
5. An application on a TTN public server.

4.3.2 Setting up the module

1. Connect the external antenna provided with the EV-Kit.
2. Connect the EV-Board to the PC using a USB cable. One COM port should be detected. Check the device manager to acquire the COM port names of the EV-Board. A typical name is "COMxy" in windows systems or /dev/ttyUSB0 in linux systems.

To interact with the module, it is advised to use the WE UART Terminal tool. Otherwise, a terminal program (like "PuTTY", "HTerm" or "Tera Term" for windows) has to be run and the corresponding COM port has to be opened using the default settings of the mounted Daphnis-I module.

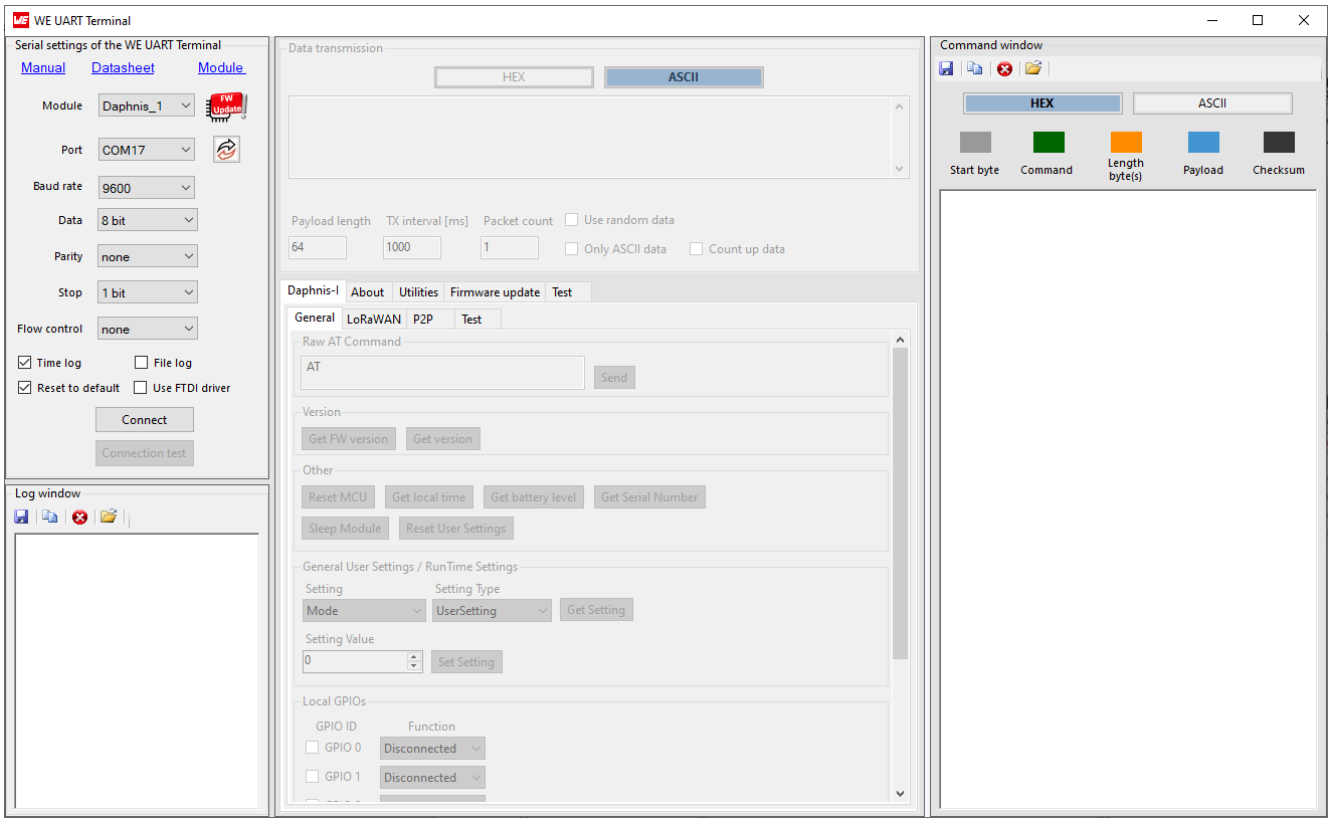

Figure 8: WE UART Terminal

3. Press the Reset button on the EV-Board to ensure a clean start-up of the module. When the Daphnis-I is ready, the output console should display the following:

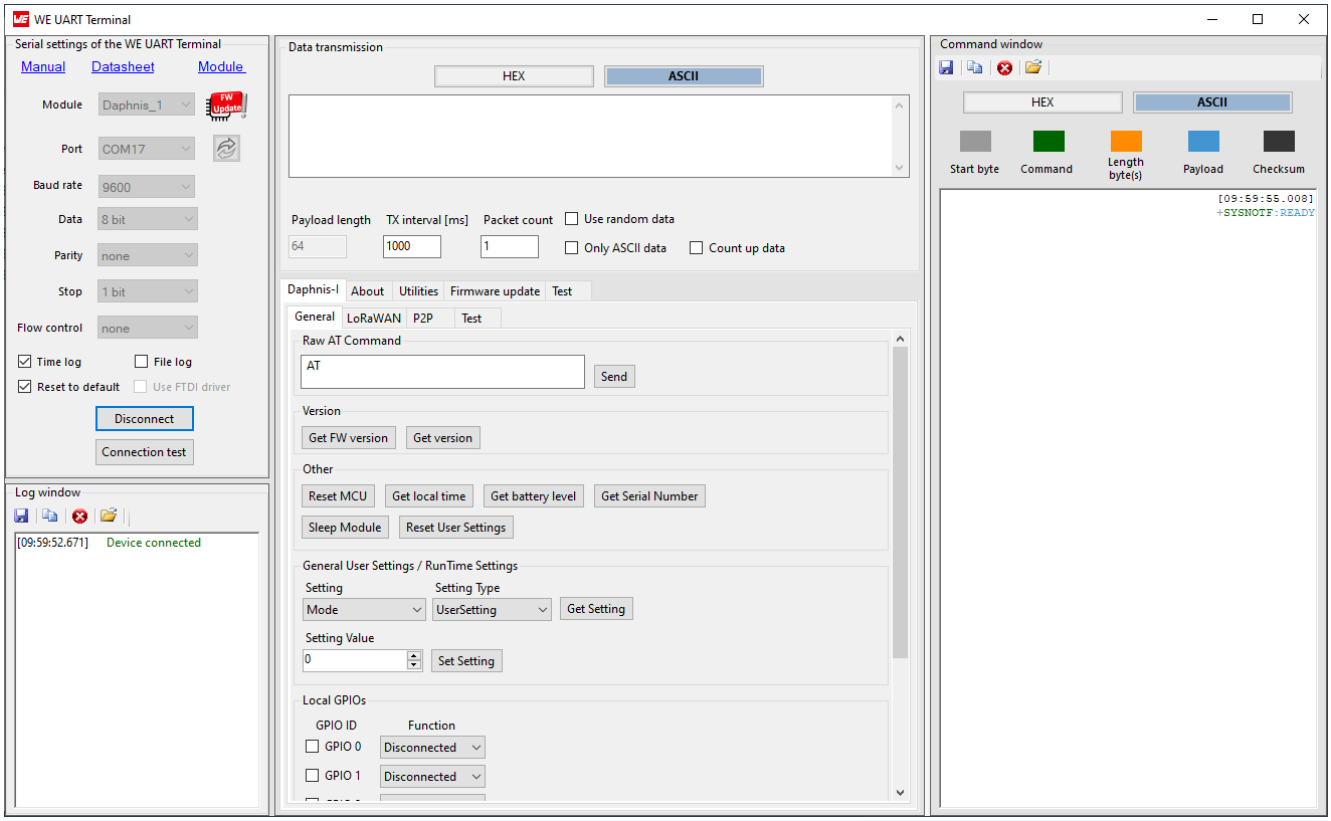

Figure 9: WE UART Terminal `+SYSNOTF` event

4. In the "General" tab, read back the `MODE` Runtime Setting by selecting "Mode" from the "Setting" dropdown and "RunTimeSetting" from the "Setting Type" dropdown and then click the "Get Setting" button. If the `MODE` isn't LoRaWAN® (Value 0). Then select "Mode" from the "Setting" dropdown, "UserSetting" from the "Setting Type" and set the "Setting Value" input to LoRaWAN® (Value 0) and click the "Set Setting". After that reset the module and wait for the `+SYSNOTF` event.

If the `MODE` is changed, then the module must be reset so that the new `MODE` is applied.

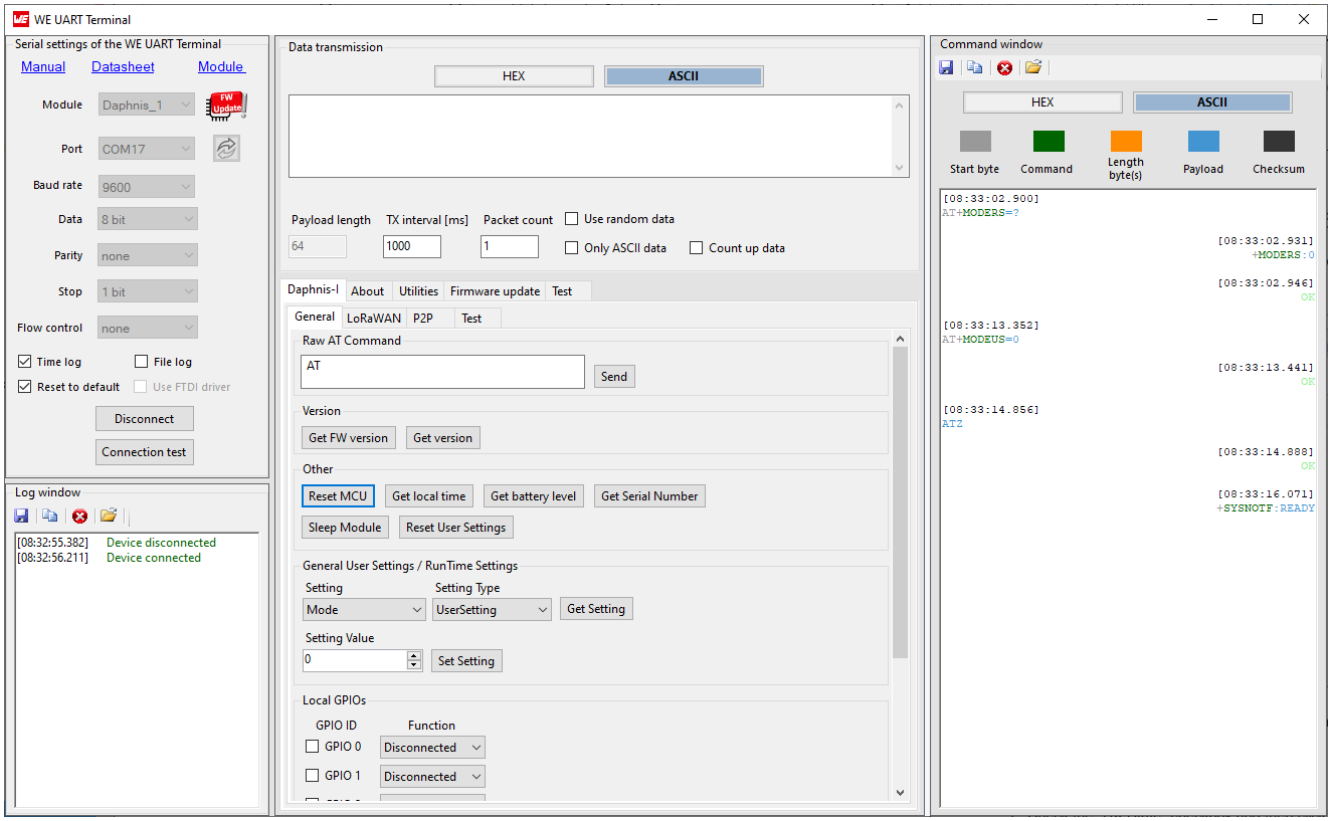

Figure 10: WE UART Terminal switch to LoRaWAN® MODE

5. As preparation for the module registration process, described in the following chapter, it may be good to know the DevEUI. Thus navigate to the "LoRaWAN" tab and request the default value of the module's DevEUI by selecting "DEUI" from the "EUI Type" dropdown and then clicking the "Get EUI" button:

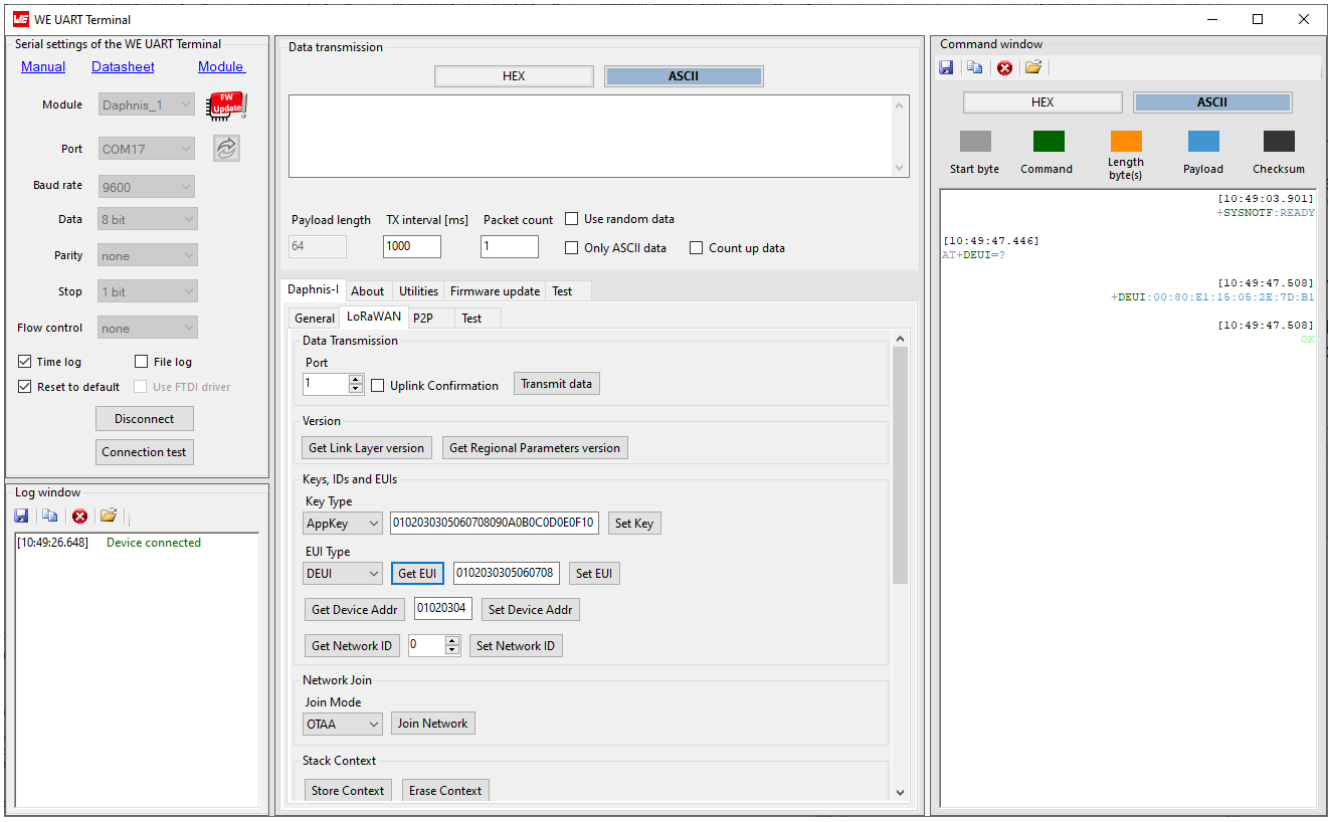

Figure 11: WE UART Terminal `AT+DEUI` command

The DevEUI is 00:80:E1:15:05:2E:7D:B1.

4.3.3 Register module to application on network server

The next step is to register the module on the network server.

TTN community (*https://www.thethingsnetwork.org/*) will be used as the network server but the steps should be similar on all network servers.

1. Enter the end device specifications manually and choose the options as specified below:

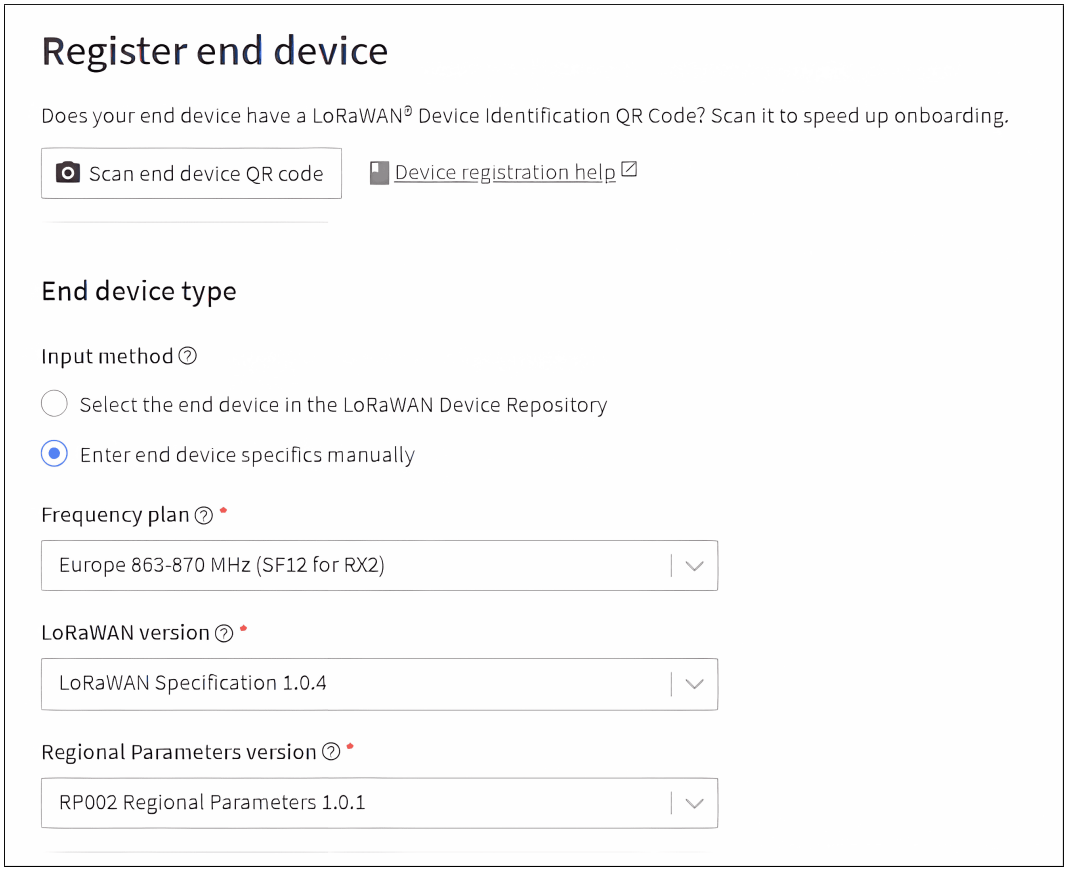

Figure 12: TTN device specifications

- Frequency plan: Europe 863-870 MHz.
- LoRaWAN® version: 1.0.4.
- Regional parameters Version: 2-1.0.1.

2. Choose the preferred activation mode:

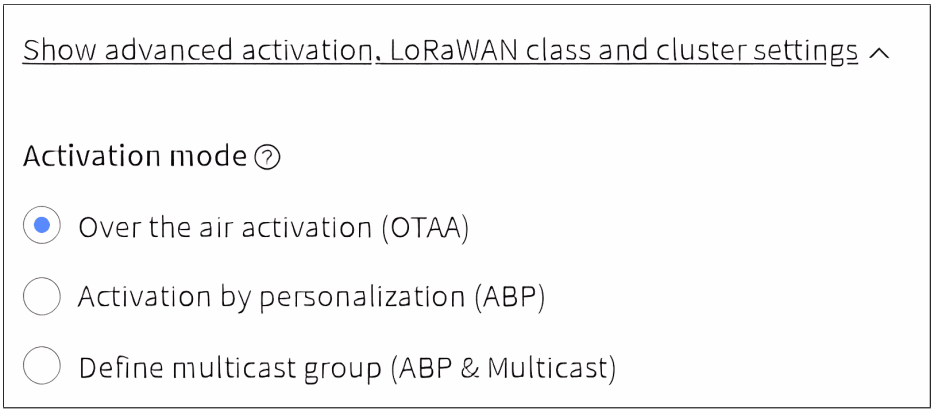

Figure 13: TTN activation mode

3. Choose the provisioning information based on the selection in previous step:

- **For OTAA joining process:**

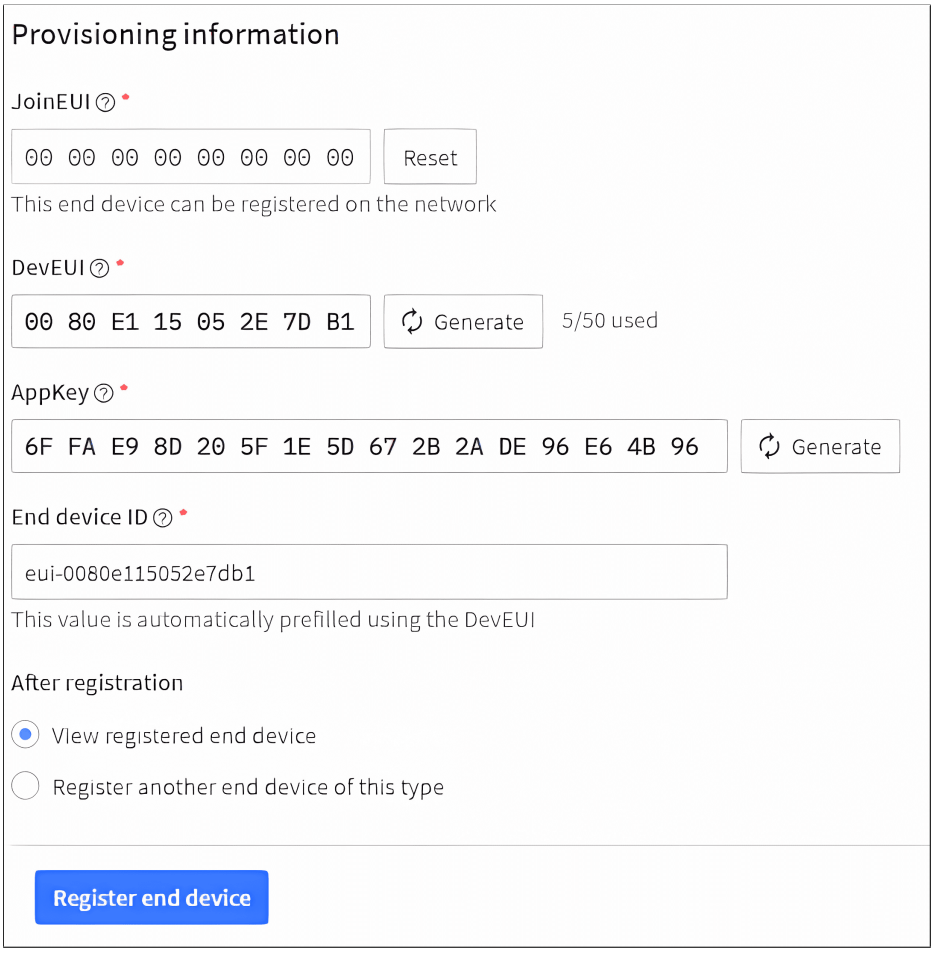

Figure 14: TTN provisioning information OTAA

– JoinEUI: set to 0s.
– DevEUI: use the default DevEUI read out in the previous chapter.
– AppKey: generate a random key.

For using OTAA during development the following option can be used so that the DevNonce on the network server resets each time on join. The Network Server uses the DevNonce of each end-device to keep track of their join requests.

– The following can be enabled to reset the DevNonce[1] on join in TTN, in the registered device under General settings, and then under Join settings.

[1] DevNonce is defined in the LoRaWAN® specification.

Figure 15: TTN OTAA reset DevNonce

– Click register end device.

- **For ABP joining process:**

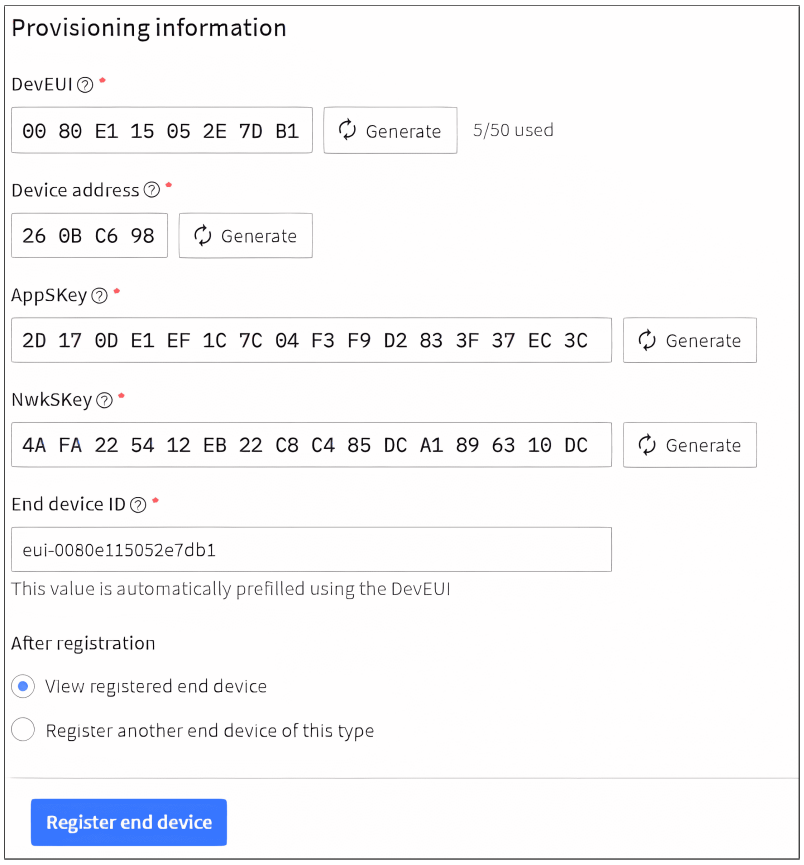

Figure 16: TTN provisioning information ABP

– DevEUI: use the default DevEUI read out in the previous chapter.
– Device Address: generate a random key.
– AppSKey: generate a random key.
– NwkSKey: generate a random key.

For using ABP during development the following option can be used so that the frame counters on the network server reset each time on join.

– The following can be enabled to reset frame counters on join in TTN, in the registered device under General settings, and then under Network layer.

Figure 17: TTN ABP reset frame counters

– Click register end device.

4.3.4 Joining the network

After the device has been registered at the network server, it can join the network.

4.3.4.1 Join via OTAA

1. Set the AppEUI/JoinEUI using `AT+APPEUI`:

```
> AT+APPEUI=00:00:00:00:00:00:00:00
< OK
```

2. Set the AppKey using `AT+APPKEY`:

```
> AT+APPKEY=3F:76:A8:2B:97:E9:44:5B:4B:8D:F8:24:5F:A4:5A:20
< OK
```

3. Set the DevEUI using `AT+DEUI`, in case the default DevEUI has not been used during device registration process on the network server:

```
> AT+DEUI=00:80:E1:15:05:2E:7D:B1
< OK
```

This step is not required if the DevEUI that was used during network registration was obtained using `AT+DEUI`.

4. Join the network using `AT+JOIN` and wait for the `+JOIN` event:

```
> AT+JOIN=1
< OK
< +JOIN:JOINED
```

To confirm that the module joined the network successfully, the following messages should appear on the TTN console of the end device.

After the network has been joined, the `AT+CS` command can be used to store the LoRaWAN® context on non-volatile memory.

Figure 18: TTN OTAA join

4.3.4.2 Join via ABP

1. Set the NwkSKey using `AT+NWKSKEY`:

```
> AT+NWKSKEY=31:ED:AC:79:D6:DB:98:E2:95:A4:B4:00:5C:01:47:1D
< OK
```

2. Set the AppSKey using `AT+APPSKEY`:

```
> AT+APPSKEY=F6:C5:4C:61:17:2B:56:2E:1C:E1:4A:3D:76:C6:1E:1D
< OK
```

3. Set the device address using `AT+DADDR`:

```
> AT+DADDR=26:0B:0D:9B
< OK
```

4. Set the DevEUI using `AT+DEUI`, in case the default DevEUI has not been used during device registration process on the network server:

```
> AT+DEUI=00:80:E1:15:05:2E:7D:B1
< OK
```

5. Join the network using `AT+JOIN` and wait for the `+JOIN` event:

```
> AT+JOIN=0
< OK
< +JOIN:JOINED
```

After the network has been joined, the `AT+CS` command can be used to store the LoRaWAN® context on non-volatile memory.

4.3.5 Data exchange with the LoRaWAN® network

4.3.5.1 Sending data to the network in unconfirmed mode

1. Join the network via OTAA 4.3.4.1 or ABP 4.3.4.2.
2. Send payload 0x1A in unconfirmed mode using AT+SEND:

```
> AT+SEND=1:0:1A
< OK
```

To confirm that the payload was received by the network successfully, the following messages should appear on the TTN console of the end device.

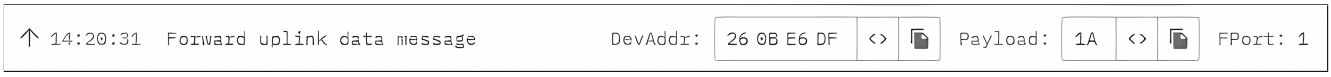

Figure 19: TTN payload received

4.3.5.2 Sending data to the network in confirmed mode

1. Join the network via OTAA 4.3.4.1 or ABP 4.3.4.2.
2. Send payload 0x1A in confirmed mode using AT+SEND and wait for +TXCONF event:

```
> AT+SEND=1:1:1A
< OK
< +TXCONF
```

To confirm that the payload was received successfully by the network, the following messages should appear on the TTN console of the end device.

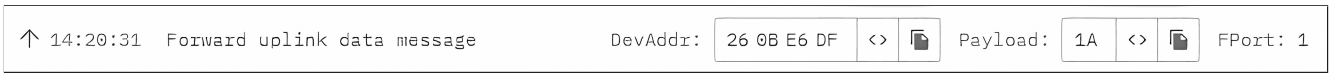

Figure 20: TTN payload received

4.3.5.3 Receiving data from the network unconfirmed mode

1. Join the network via OTAA `4.3.4.1` or ABP `4.3.4.2`.
2. Schedule downlink via TTN console:

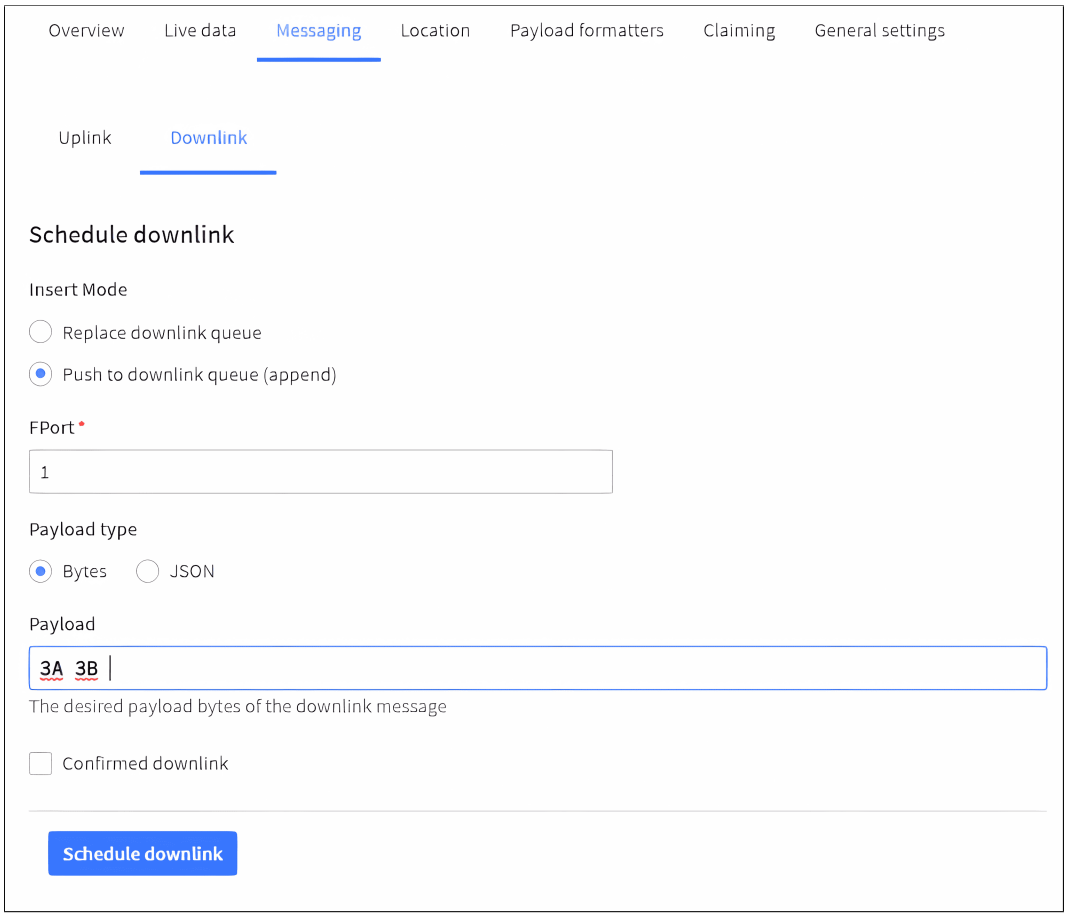

Figure 21: TTN payload schedule

To confirm that the payload was scheduled successfully, the following messages should appear on the TTN console of the end device.

Figure 22: TTN payload sent

3. The payload is received on the module depending on the selected class, in case of Class A an uplink is required first using `AT+SEND` and then wait for the `+RXDATA` event:

```
> AT+SEND=1:0:1A
< OK
< +RXDATA:1,02,3A3B
```

4.3.6 Switch the LoRaWAN® class

The default class for the Daphnis-I is LoRaWAN® Class A. It can be changed to Class B or Class C. The following lines show how to update it to Class B. For Class C, run the same steps with C instead of B.

1. Enable Class B support in TTN for the end device and save the changes:

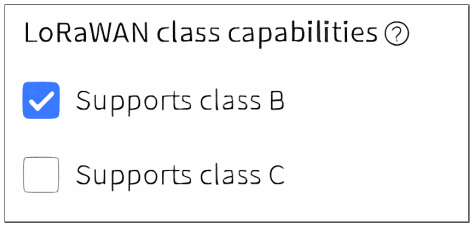

Figure 23: TTN enable Class B

2. Join the network via OTAA `4.3.4.1` or ABP `4.3.4.2`.
3. Run the switch class command `AT+CLASS`:

```
> AT+CLASS=B
< OK
```

4. Send an uplink with any payload using `AT+SEND`:

```
> AT+SEND=1:0:1A
< OK
```

5. Wait for the `+CLASSUPDATE` event:

```
< +CLASSUPDATE:B
```

6. Send an uplink with any payload using `AT+SEND`:

```
> AT+SEND=1:0:1A
< OK
```

4.4 P2P mode quick start example

This section is intended to help the user set-up and test the exchange data between two Daphnis-I modules. Minimal pin and antenna connections have to be done on both the modules, as described in sections 4.1 and 4.2. It is recommended to use the Daphnis-I EV-Kit for quick tests.

4.4.1 Prerequisites

The following hardware is required to go through the quick start example.

1. Two Daphnis-I EV-Boards
2. Computer with WE UART Terminal PC tool or any other a serial terminal emulator.

4.4.2 Setting up the module

In this example, the Daphnis-I EV-Boards are connected to a PC running the WE UART Terminal tool (Figure 24).

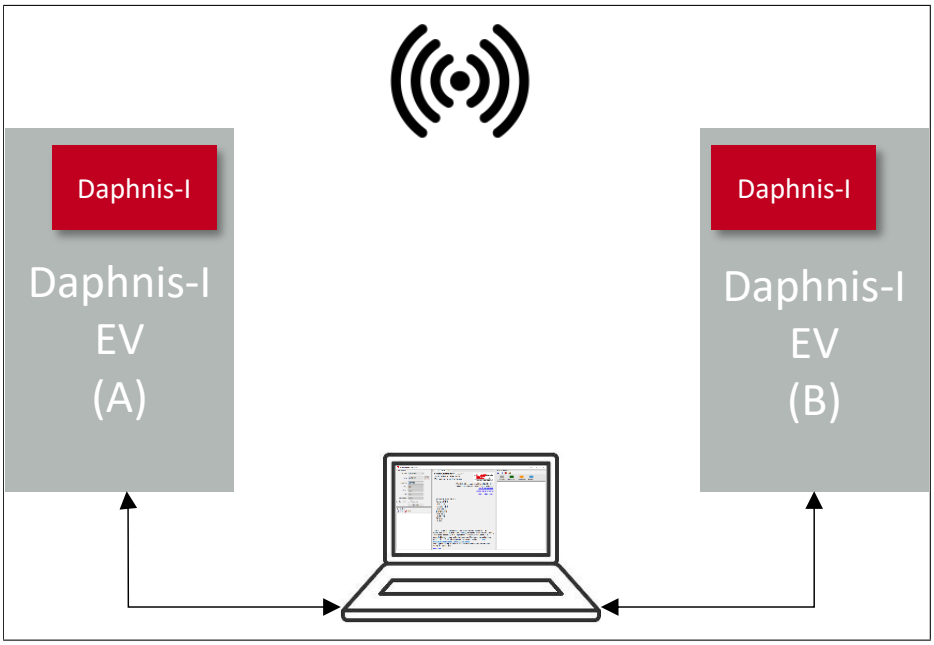

Figure 24: Set-up quick start

1. Connect the Daphnis-I EV-Boards to the laptop/PC via USB.
2. The power LED indicates that supply voltage is active.

The FTDI driver [4] for the converter IC on the EV-Board has to be installed and/or updated. On correct driver installation, the EV-Board appears as a virtual COM port.

3. Open two instances of the WE UART Terminal [1].

4. Open an instance of the serial port with COM port settings as specified in chapter 6.1.1 and perform the following steps for each Daphnis-I EV-Board in the WE UART Terminal instances.

5. By pressing the Reset button, the +SYSNOTF event appears on the command window of the WE UART Terminal.

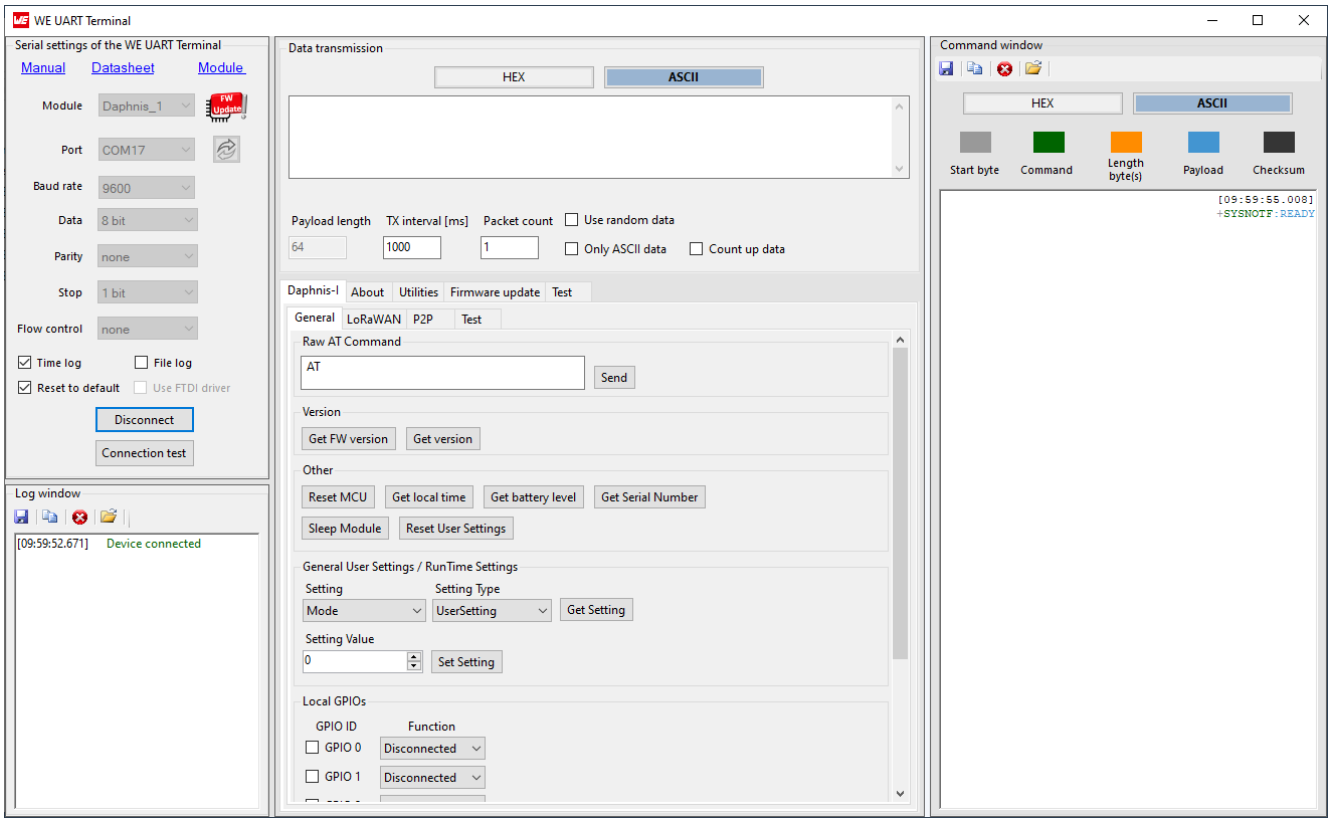

Figure 25: WE UART Terminal +SYSNOTF event

6. In the "General" tab, read back the MODE Runtime Setting by selecting "Mode" from the "Setting" dropdown and "RunTimeSetting" from the "Setting Type" dropdown and then click the "Get Setting" button. If the MODE isn't P2P (Value 1). Then select "Mode" from the "Setting" dropdown, "UserSetting" from the "Setting Type" and set the "Setting Value" input to P2P (Value 1) and click the "Set Setting". After that reset the module and wait for the +SYSNOTF event.

If the MODE is changed, then the module must be reset so that the new MODE is applied.

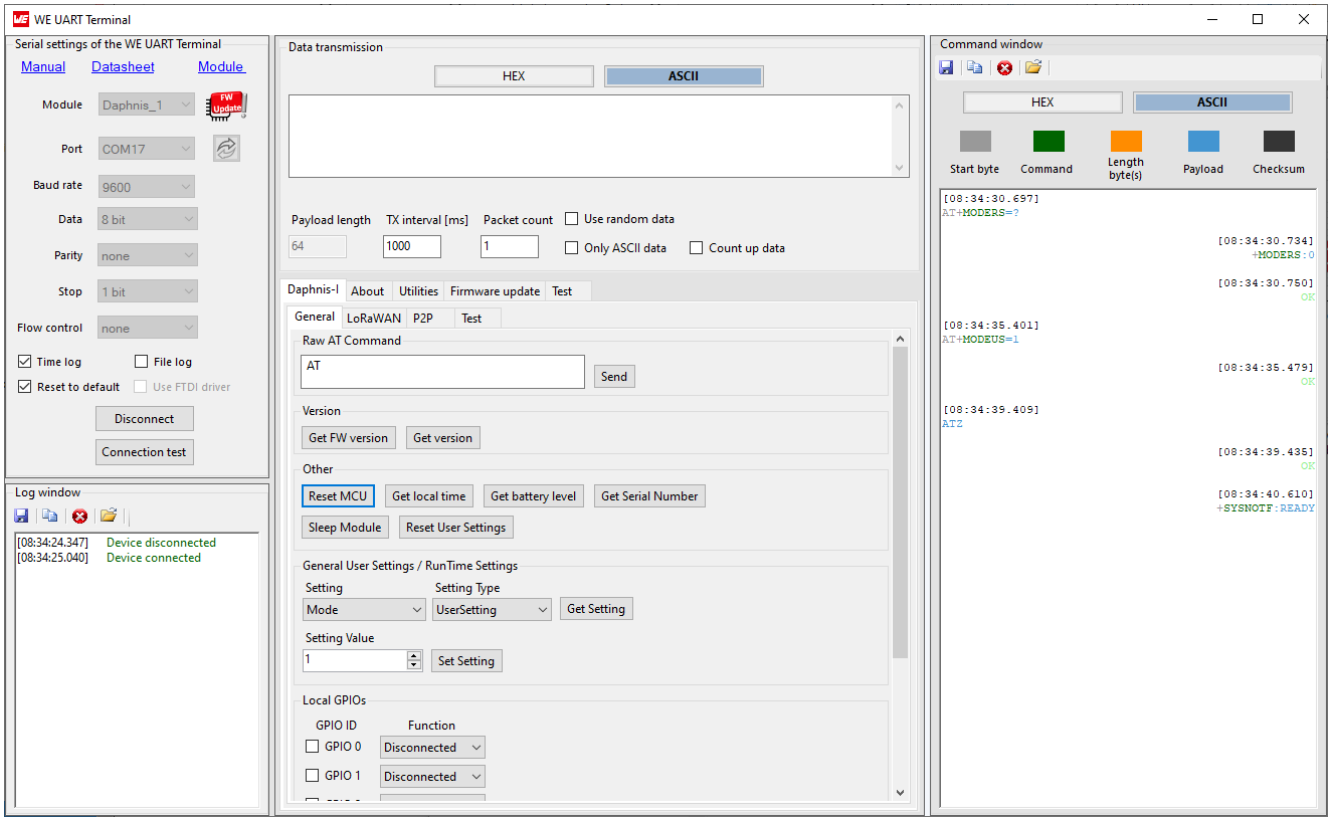

Figure 26: WE UART Terminal switch to P2P `MODE`

7. Navigate to the "P2P" tab.

4.4.3 Transmit and receive data

1. Check the "RX State" checkbox and then click the "Set RX State" button, this will trigger the `AT+P2PRX` command.
2. On the WE UART Terminal connected to Daphnis-I-EV A, enter "hello world" (or hex-equivalent of it in case that option is chosen) in the data transmission text box.
3. Choose "Broadcast" from the "Transmission Mode" dropdown and then click on the "Transmit data" button, this will trigger the `AT+P2PBROADCASTTX` command.
4. The module acknowledges with a `+P2PTXCONF` event and sends a `+P2PTXRESP` event indicating successful transmission of data.
5. On the WE UART Terminal connected to Daphnis-I-EV B, a `+P2PRXDATA` message appears containing the same data that was transmitted.
6. Repeat the above steps from Daphnis-I-EV B to test data transmission from B to A.
7. Note that the *LED_1* and *LED_2* blink to indicate data transmission and reception, respectively.

The RSSI values will differ based on the distance between the modules.

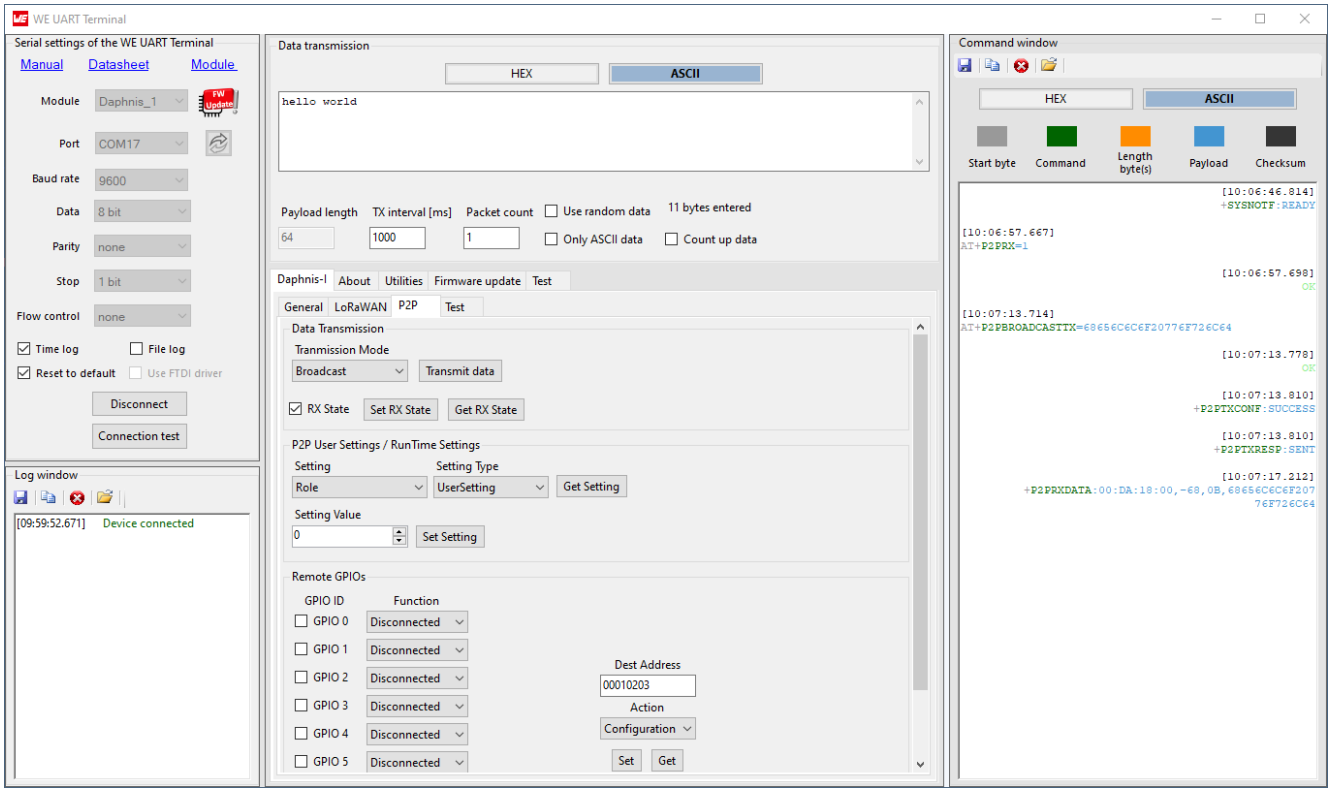

Figure 27: Module A command sequence

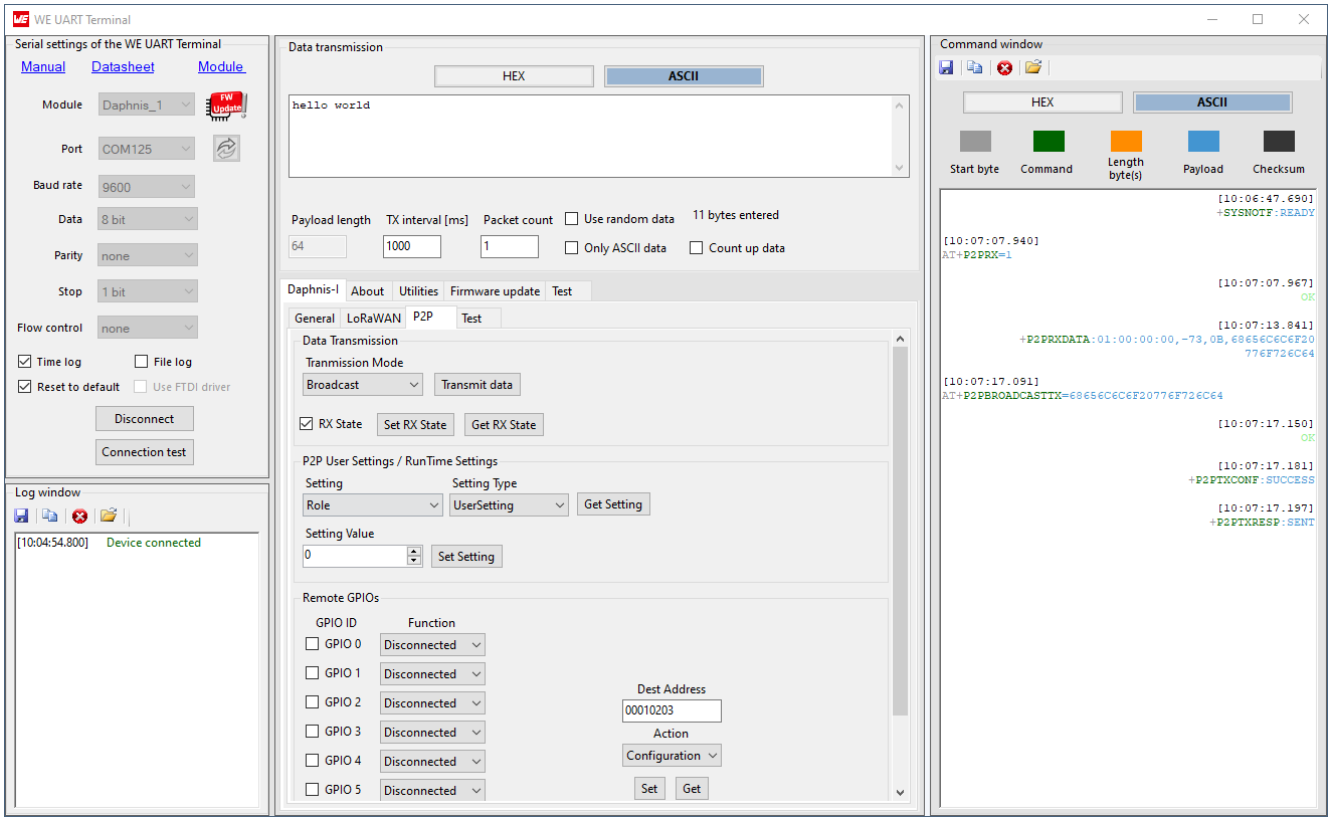

Figure 28: Module B command sequence

5 Functional description

The Daphnis-I module is intended to be used as a radio sub-system in order to provide LoRaWAN® as well as proprietary communication capabilities to the system.

Daphnis-I is a compact, high-performance LoRaWAN® and proprietary module with ultra-low power consumption.

The `Application UART` interface acts as the primary interface between the module and a host MCU. The module can be configured and operated using a set of AT commands over `Application UART`.

5.1 Module key features

Feature	Description
Physical dimensions	15 mm x 16 mm x 3 mm
Radio chipset	STM32WLE5CCU6
Supported protocol	LoRaWAN®, Proprietary
Module interfaces	UART
Supported frequency bands	EU868
Maximum power transmission	14 dBm
`Sleep mode` current consumption	63.9 nA
Operating voltage	2.0 V to 3.6 V
Operating temperature	-40 °C to 85 °C

Table 10: Module Key Features

Feature	Description
Supported LoRaWAN® specification	1.0.4 [5]
Supported regional parameters	2-1.0.1 [6]

Supported LoRaWAN® classes	`Class A`
	`Class B`
	`Class C`

Table 11: LoRaWAN® mode Key Features

5.2 General functional description

5.2.1 Secure firmware update

It is possible to perform a firmware update on the module over the `Bootloader UART`. The bootloader checks for the image signature before overwriting the exiting image. Refer to chapter `12` for further details.

5.2.2 Local digital I/O control

Daphnis-I offers several programmable digital I/O pins for use in end applications. These pins can be configured and controlled directly by the local host over `Application UART`.
Chapter `7.5` explains the options available to configure and control the digital I/O pins locally in detail.

5.3 LoRaWAN® mode functional description

5.3.1 LoRaWAN® network setup

The LoRaWAN® network consists of the end nodes, the gateways and the network server.

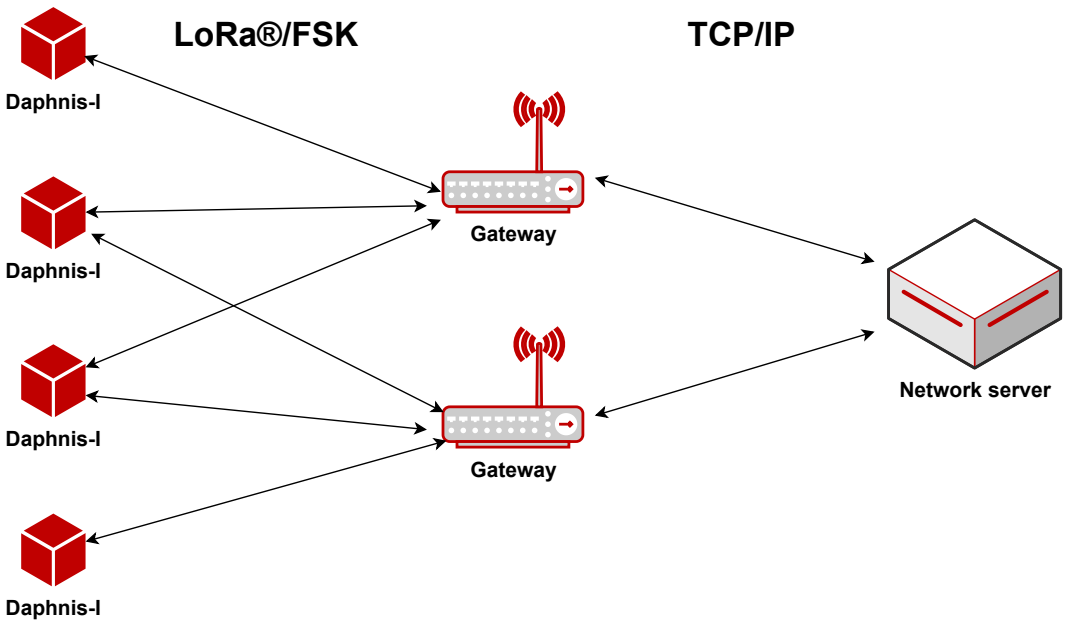

Figure 29: LoRaWAN® network setup

The end nodes represent Daphnis-I, which usually exist in end devices. Using these end nodes, data can be sent to or received from the gateways in a LoRaWAN® frame encapsulated in a LoRa® or FSK packet.

The LoRaWAN® gateways act as a bridge between the end nodes and the LoRaWAN® network server, decapsulating the LoRa® or FSK packets received within their range from the end nodes and encapsulate the resulting LoRaWAN® frames in TCP/IP packets and send

them to a LoRaWAN® network server using TCP/IP. Additionally, the gateways decapsulate the TCP/IP packets received from the LoRaWAN® network server and encapsulate the resulting LoRaWAN® frames in LoRa® or FSK packets and send them to the end nodes.

In order for the gateway to be compatible with Daphnis-I, the gateway should support EU868 regional parameters version 2-1.0.1.

Some gateways have an embedded LoRaWAN® network server. If the embedded network server is used, it should support LoRaWAN® 1.0.4 specification.

The LoRaWAN® network server is the core of the network as its responsible for security and data routing through the network. It also manages uplinks that have been sent to the network server from the gateways and removes duplicates. It also schedules downlinks on the closest gateway to the end nodes. The network server is usually connected to application servers for further processing of the data.

In order for the network server to be compatible with Daphnis-I, it needs to support EU868 regional parameters version 2-1.0.1 and LoRaWAN® 1.0.4 specification.

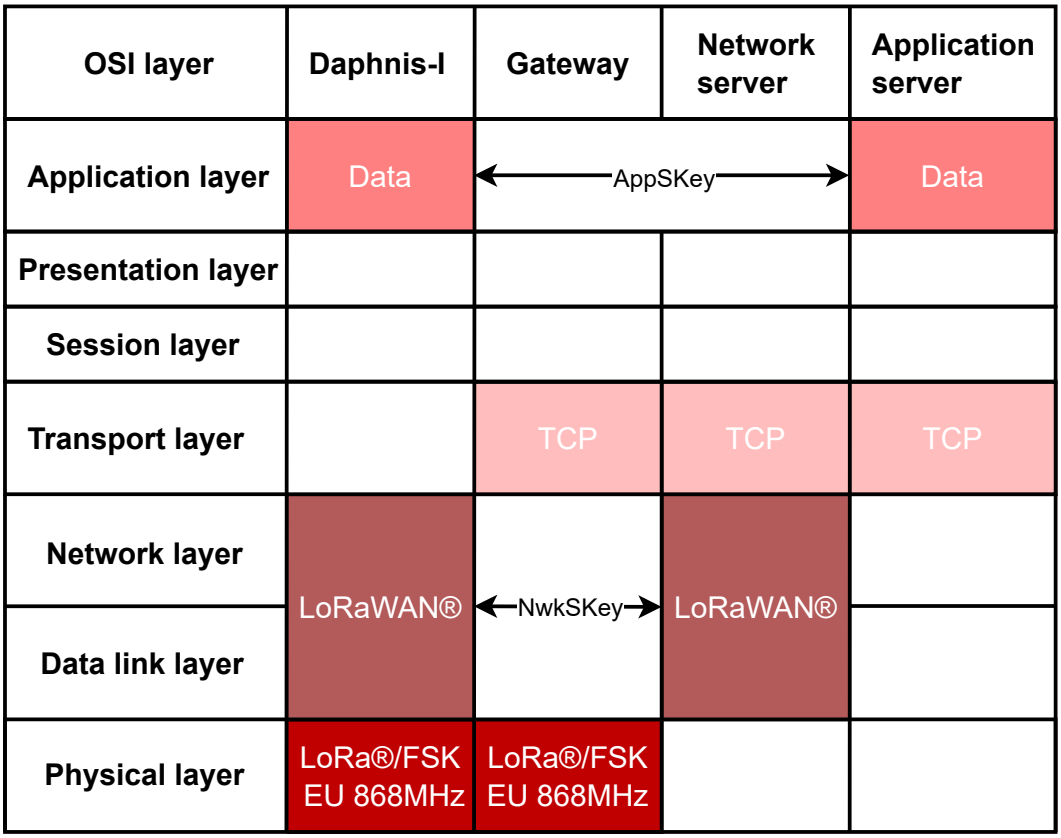

Figure 30: LoRaWAN® stack in the OSI model

5.3.2 LoRaWAN® classes

The LoRaWAN® specification defines three device types: Class A, Class B, and Class C. Daphnis-I supports all three (Class A, Class B and Class C) LoRaWAN® classes.

Class B and C are an extension to Class A.

5.3.2.1 Class A

In Class A communication is always initiated by the module, which can send an uplink message at any time. When the uplink transmission is completed, the module opens two short receiving windows (RX1 and RX2 respectively). There is a delay between the end of the uplink transmission and the start of the receive windows. If the network server does not respond during these two receive windows (RX1 and RX2), the next downlink will be available after the next uplink transmission.

The module opens the RX2 window only, when it does not receive data in RX1 (network server missed downlink transmission in RX1).

The module in this class is in `Idle state`. It is awaken by the application UART via AT commands, then it goes back to `Idle state` again. When `AT+SEND` command is triggered, the module moves into `Active state` during the transmission window and the two RX windows. When these windows are closed, the module goes back to `Idle state`.

Between the uplink transmission and the two short receiving windows, the module remains in `Idle state`.

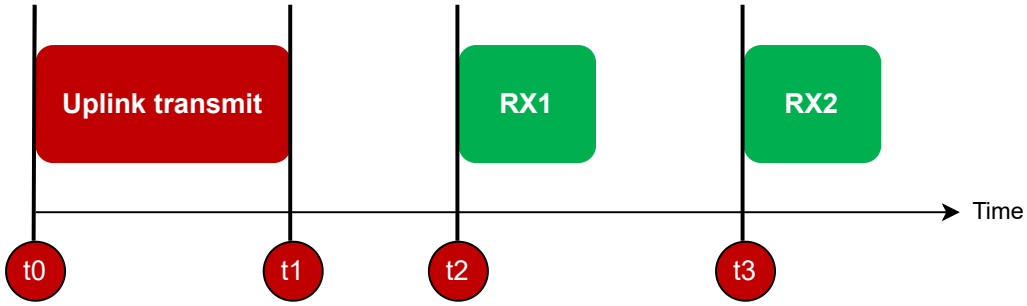

Figure 31: Class A

Time	Description
t1-t0	Transmit time on air
t2-t1	Receive (RX1) delay
t3-t1	Receive (RX2) delay

Table 12: Class A windows description

5.3.2.2 Class B

This class is an extension to `Class A`. In addition to `Class A`, the module opens periodic receive windows which are called ping slots.

The network server sends beacons through the gateway, so that the module can align its clock with the network server's clock. The module opens a receive window (ping slots) after it receives a time-synchronized beacon from the gateway.

Any of these ping slots may be used by the network server to initiate a downlink communication.

The module in this class keeps switching between `Idle state` and `Active state` during ping slots.

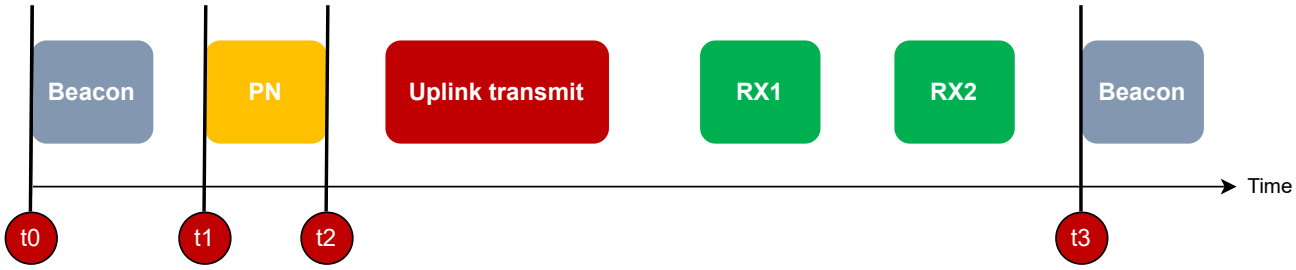

Figure 32: Class B

Time	Description
t3-t0	Beacon period
t2-t1	Ping slot

Table 13: Class B windows description

5.3.2.3 Class C

The module in this class has receive windows (RXC, RX1, RX2) which are almost always open. These receive windows close only when the module is transmitting.

This means in `Class C` module uses more power to operate than `Class A` or `Class B`, but `Class C` offers the lowest latency for communication from the server to the module.

The module in this class is always in `Active state` since there is a continuous receive window open.

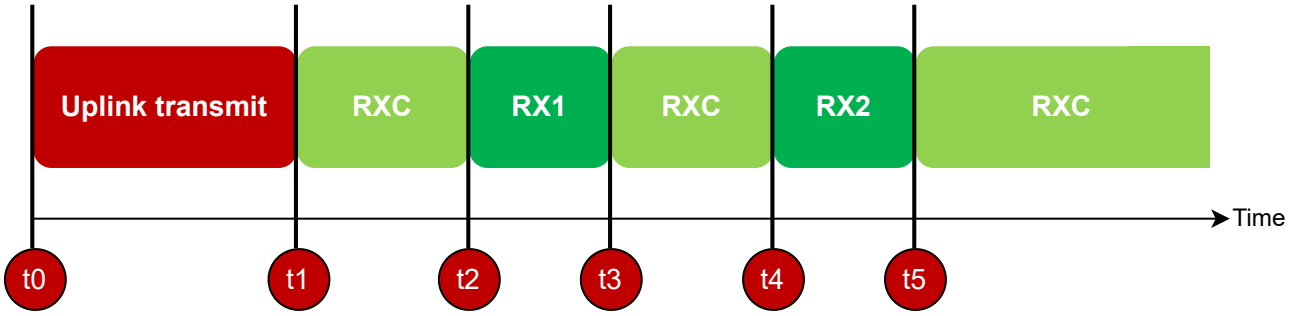

Figure 33: Class C

Time	Description
Any time except for t1-t0 or t3-t2 or t5-t4	RXC open

Table 14: Class C windows description

The default class for the Daphnis-I is LoRaWAN® Class A. It can be changed to Class B or Class C. To change the LoRaWAN® Class Refer `Switch the LoRaWAN® class.`

5.4 P2P mode functional description

5.4.1 Network addressing

In order to build a network on interconnected devices and send data to specific devices, an addressing mechanism is necessary. Daphnis-I implements addressing mechanism to support unicast, multicast and broadcast transmission of data.

- Every module can be configured to have a unique 4-byte address. By default, the source address of the module is configured to be the serial number of the module.
- Additionally, the module can be configured to transmit to a group using a 1-byte group-ID. This feature enables creation of multicast groups.
- Data commands are available to send unicast/multicast or broadcast data (see section 7.10.1).
- Automatic address resolution is implemented to filter out the packets that are not addressed to the module's configured address.
- The parameters P2PMACSRCADDR, P2PMACGRPID and P2PMACDESTADDR can be used to set the network settings.

5.4.2 Acknowledgments

In order to improve reliability in communication, the module can be configured to use the radio acknowledgment mechanism. It can be activated setting the parameter P2PMACACK accordingly. This is only valid for unicast transmission.

5.4.3 Flooding mesh: Using the repeater functionality

Daphnis-I can be run as repeater to extend the transmission range using the setting P2PROLE. A module configured as a repeater, simply retransmits the received packet after a random back-off time. A time-to-live (TTL) parameter (P2PMACTTL) can be used to set the hop limit.

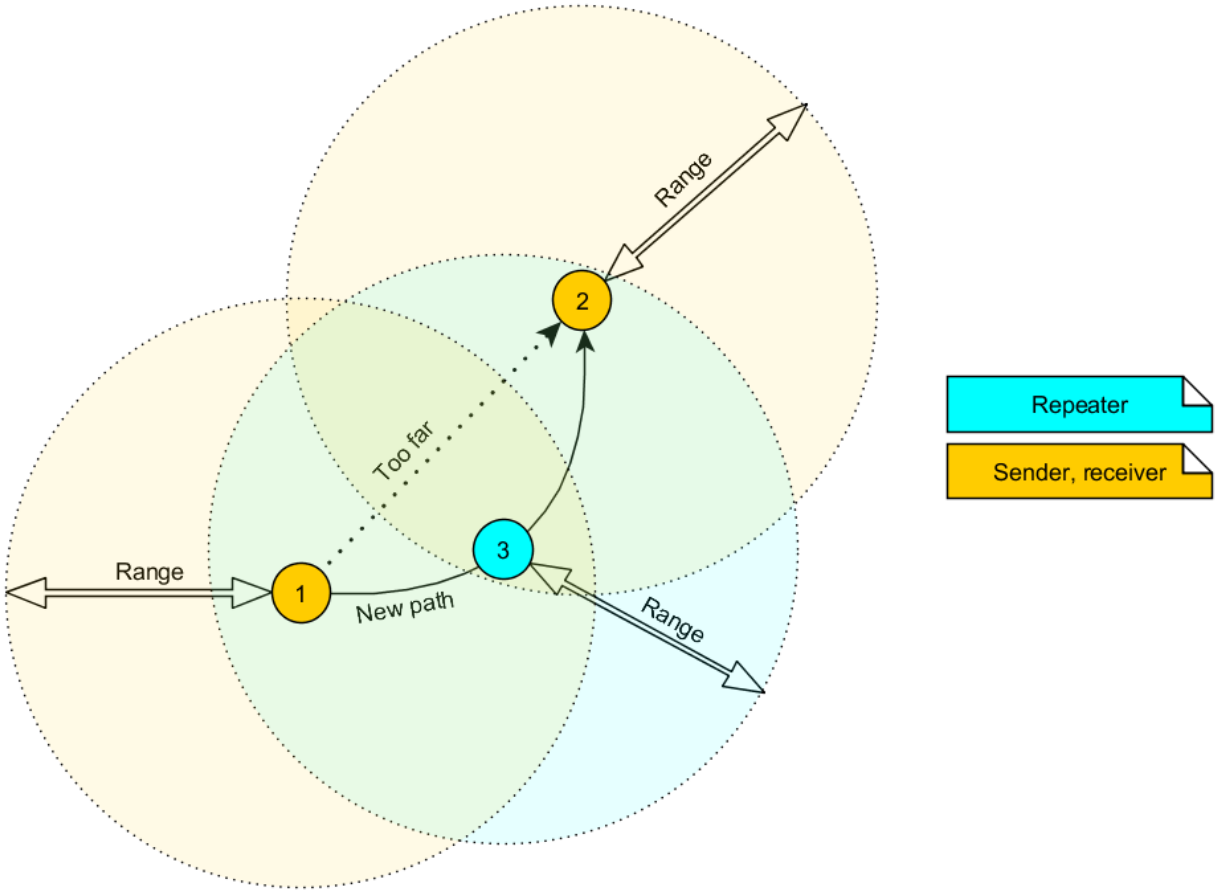

Figure 34: Range extension using several repeaters

If the module is configured as repeater, it can be simply added to an existing wireless network consisting of compatible modules. With this, the newly generated mesh network uses the so-called "flooding technique" to deliver data packets from their source to their destination device.

The repeater module itself simply listens to the configured channel and forwards all received packets. In case the RSSI value of the received packet is higher than the threshold `P2PRPTHRESH`, the radio packet will not be relayed.

A random delay is used to avoid RF packet collision. To reduce traffic on the frequency channel, each repeater device checks before repetition, whether it has already sent this packet before or not. Thus, every repeater sends each packet only once. Furthermore, a time-to-live parameter can be set on the radio packet in order to restrict the number of hops.

In a network with `NumRP` repeater devices, each data packet is repeated `NumRP` times. Therefore, each packet that is sent from node A to node B forces a traffic of `NumRP+1` data packets in total on the frequency channel.

5.4.3.1 Setup of the network and repeater device

The repeater mode can be enabled by setting the P2PROLE parameter to 1.
If the module is configured as repeater, the following notes have to be considered:

1. Requirements on the network:
   a) The repeater devices have to be line-powered (no battery).
   b) Depending on the data rate, each repeater should repeat a maximum of 2-5 packets per second to give a good chance that the repeater is not busy with repeating when already a new packet arrives for repetition. More packets per second will result in more packet loss, as the collision probability is increased.
   c) If the network consists of several layers of repeaters, each layer adds additional delays to the packet transmission.
   d) To set-up the network, all participants have to use the same P2PRFCHANNEL and P2PRFPROFILE.
2. Information for the repeater device:
   a) Acknowledgements (ACK) of successfully received packets are blocked. If an ACK is requested by the sending module, the request is ignored. Furthermore, the repeater does not request any ACK, when repeating a packet.
   b) A time-to-live parameter is present in every packet. The repeater decrements the TTL by one before forwarding the packet. The packet with a TTL value of zero is not repeated.
3. Information for the sending and receiving devices:
   a) The senders should send less frequently to avoid packet collision on the frequency channel.
   b) The repeater devices do not support the feature of ACKs for the successful reception of the packets. Thus, the sender will never receive ACKs, if requested.
   c) Every repeater sends each packet only once. However, receivers can receive each packet several times (sent by different repeaters), if there are packets of different content in the network temporarily close to each other. Thus, on the side of the receiving device, a mechanism that detects and filters double packets shall be implemented.

5.4.3.2 Example network

In the example network shown above, the goal is to send a packet from device 1 to 5. Without the repeater devices, this would be impossible. The steps are as follows:

1. Sender 1 sends a packet.
   a) Repeater 2 and 3 receive and accept it at the same time.
2. Device 2 and 3 delay the packet.
   a) Repeater 3 sends the packet.
      i. Sender 1 and 6 do not accept it, since their addresses are wrong (unequal 5).

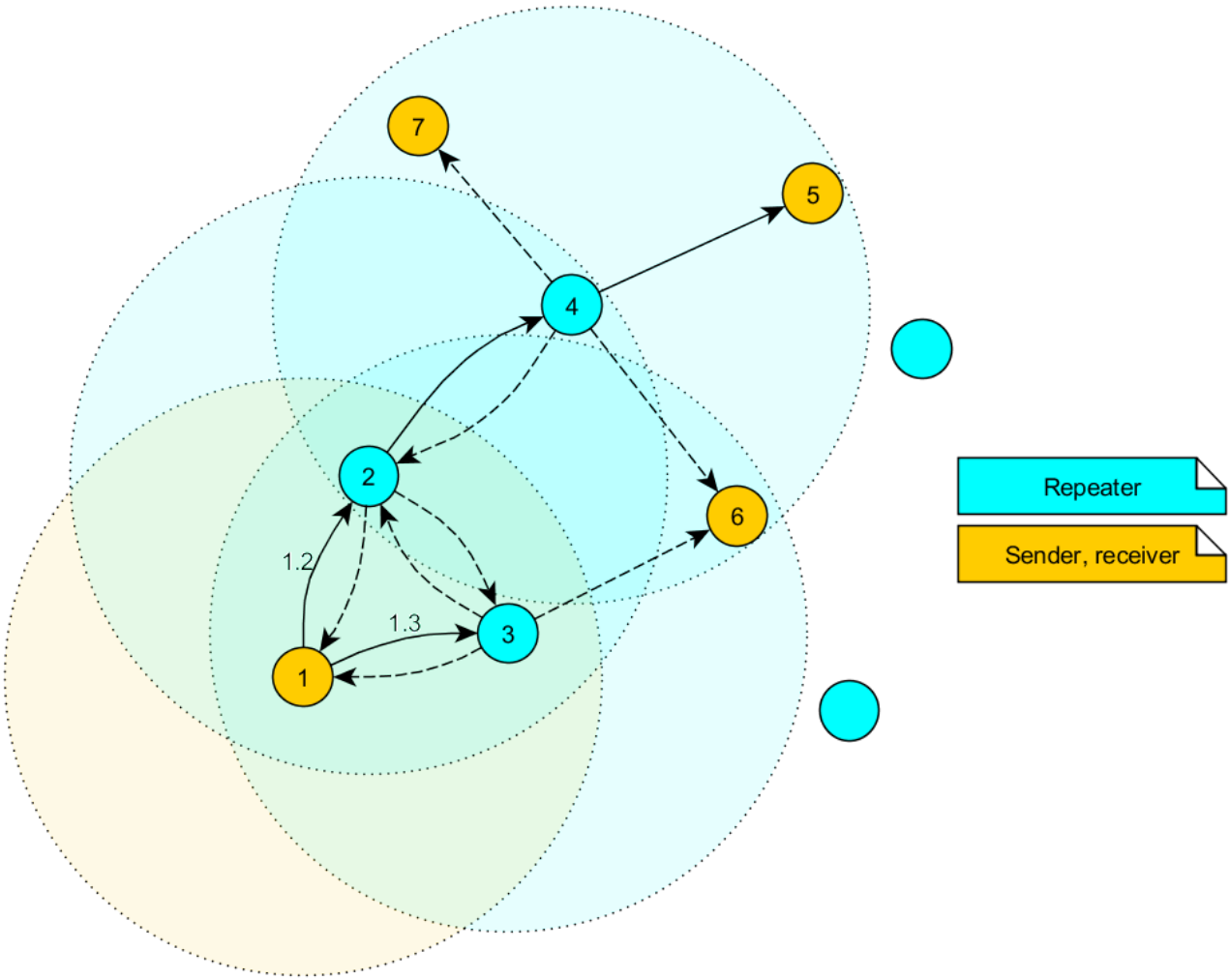

Figure 35: Example network

ii. Repeater 2 does not accept it, since it has been already received before (1.2).

b) Repeater 2 sends the packet.

i. Repeater 3 does not accept it, since it has been already received before (1.3).

ii. Sender 1 does not accept it, since its address is wrong (unequal 5).

iii. Repeater 4 receives and accepts the packet.

3. Repeater 4 delays and sends the packet.

a) Senders 6 and 7 do not accept it, since their addresses are wrong (unequal 5).

b) Repeater 2 does not accept it, since it has been already received before (1.2).

c) Receiver 5 accepts it and its successfully delivered (address equals 5).

Note that the packet forwarded by repeater 2 and 3 would collide in the frequency channel, if they wouldn't be randomly delayed (see `P2PRPNUMSLOTS`).

5.4.3.2.1 Application in parallel networks

As described above, a repeater device forwards all packets that are received before. If a network needs to have a bigger throughput of data, a parallel network can be set up, that relaxes the stress of the primary network. To do so, all sending, receiving and repeater devices of the parallel network are configured to use a new non-overlapping channel, such that the primary network is not affected at all by the traffic of the parallel network.

5.4.4 Listen Before Talk

This section describes the Listen Before Talk (LBT) feature. A radio transmitter performing an LBT, assesses the channel and proceeds with transmission only when the channel is detected as not active. This method provides a way for collision avoidance at the cost of latency and throughput.

P2PLBT is disabled by default in Daphnis-I meaning that the module transmits the data without checking if the channel is inactive.

If LBT is enabled and the specified P2PRFPROFILE uses FSK modulation then the transmitter performs a signal strength measurement on the configured channel for a predefined observation period. If the measured signal strength is below a predefined threshold, the channel is considered to be free and the packet is transmitted. In case the signal strength detected is beyond a certain threshold, the channel is considered to be busy. In this case, the host gets a channel busy status indicated.

Otherwise if LBT is enabled and the specified P2PRFPROFILE uses LoRa® modulation then the transmitter starts Channel Activity Detection (CAD) which scans the band for a predefined duration for preamble or data symbols. If there were no symbols detected during CAD, the channel is considered to be free and the packet is transmitted. In case there were symbols detected, the channel is considered to be busy. In this case, the host gets a channel busy status indicated.

In the figure 36 below, the procedure to determine if the channel is busy or free is shown.

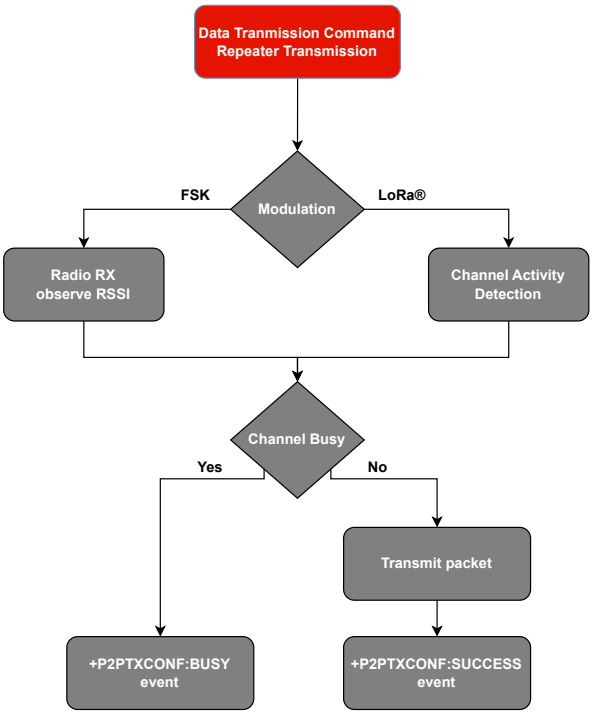

Figure 36: Listen Before Talk

5.4.5 Encryption/decryption

In order to establish a secure wireless link, the Daphnis-I supports encryption and decryption of transmitted payload.

- The Daphnis-I implements a symmetric 128 bit AES encryption.
- Encryption is disabled by default.
- Encryption/decryption of data cannot be enabled without setting the `P2PMACENCKEY`.
- The 128 bit key can be written on to the flash using the parameter `P2PMACENCKEY`.
- The parameter `P2PMACENCMODE` can be used to configure the level of security.

5.4.6 Remote digital I/O control

Daphnis-I offers several programmable digital I/O pins for use in end applications. These pins can be configured and controlled by a remote host over the radio link.
Chapter `7.10.2` explains the options available to configure and control the digital I/O pins remotely in detail.

5.5 Daphnis-I power modes

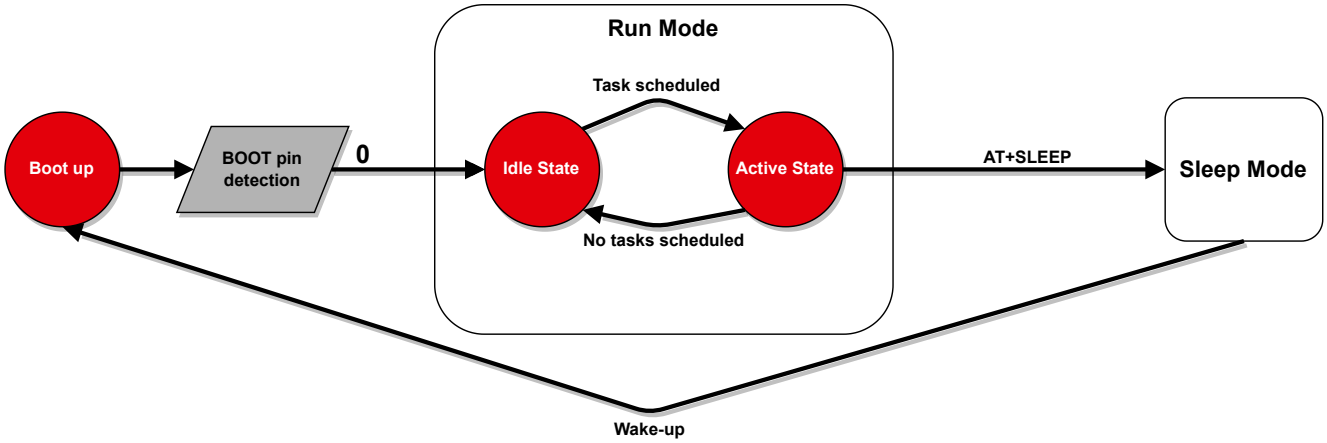

Figure 37: Power modes

5.5.1 Run mode

In this mode the module is either in `Active state` or in `Idle state`. The switching between the two states is managed automatically by the module.

5.5.1.1 Active state

The active state is when the module is parsing the received AT commands via the `Application UART`, transmitting or receiving packets.

5.5.1.2 Idle state

The idle state is when the module is currently not performing any task.

5.5.2 Sleep mode

The module enters this mode using the `AT+SLEEP` command and exits this mode by pulling low the */WAKE_UP* pin.

6 Host connection

The Daphnis-I is intended to be used as a radio module in a system, interfaced with a host MCU. The use of an industry standard UART as the primary interface ensures a very minimal requirement set on the host MCU. As a result of this, the module can be designed in with most host controllers from an 8051 to the more advanced ARM core architecture.

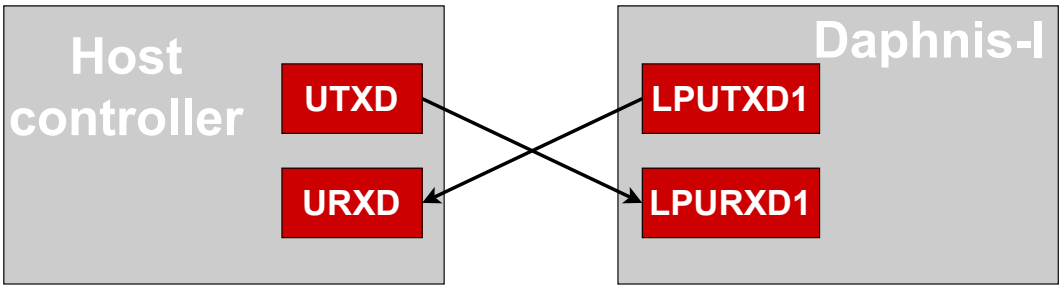

Figure 38: Host interface

6.1 UART interfaces parameters

The Daphnis-I implements two UART interfaces, one for the application and one for the bootloader with the following parameters.

6.1.1 Application UART

Parameter	Range	Standard
Baud	9600, 57600, 115200	9600
Data bits	8	8
Stop bits	1	1
Parity	none	none
Flow control	none	none

Table 15: Application UART parameters

The configuration of the `Application UART` in factory state is 9600 baud with data format of 8 data bits, no parity and 1 stop bit ("8n1"). The baud rate of the application UART can be configured using the setting `UARTBAUDRATE`. All other parameters are fixed as shown in `Application UART parameters`. This results in a user data ratio of 10 UART symbols per 8 bit.

6.1.2 Bootloader UART

Parameter	Range
Baud	115200
Data bits	8
Stop bits	1
Parity	none
Flow control	none

Table 16: Bootloader UART parameters

The configuration of the `Bootloader UART` in factory state is 115200 baud with data format of 8 data bits, no parity and 1 stop bit ("8n1"). All parameters are fixed as shown in `Bootloader UART parameters`. This results in a user data ratio of 10 UART symbols per 8 bit.

The firmware update process requires access of the `Bootloader UART`.

7 AT commands

7.1 AT command syntax and types

The command set consists of a series of short text strings for performing operations such as parameters setting and reading data from the Daphnis-I module. AT commands are sent through the UART interface.

The AT commands are sent to the Daphnis-I module with the following generic syntax:

AT+<command_name>=<param1>,<param2>,. . . ,[<paramN>]<CR><LF>

Following syntactical definitions apply for AT command syntax.

- AT+: Command line prefix for AT commands.
- <command_name>: It represents the name of the AT command.
- <param1>,<param2>: Name enclosed in <angle brackets> is a syntactical element. Brackets themselves do not appear in the command line.
- [<paramN>]: Name enclosed in [<square brackets>] is a syntactical element. Brackets themselves do not appear in the command line. And it indicates an optional value (will not always be returned or does not always need to be set).

AT commands in Daphnis-I are case sensitive.

When an AT command has more than 1 parameter, then a comma (,) is used as a delimiter to separate each parameter.

- Additional whitespaces in an AT command are not allowed. Additional whitespaces will return ERROR by the module.
- <CR><LF>: Mandatory command line termination characters <CR><LF>. For readers convenience the example sections will not display these characters.

The following command types are available:

- AT+<command_name>? provides a short help of the given command.
- AT+<command_name>=? is used to get the value of a given command.
- AT+<command_name> is used to run a command.
- AT+<command_name>=<param1>,<param2>,. . . ,[<paramN>] is used to provide a value to a command.

  Output of the commands is provided on the UART. The output format is typically:

  ```
  +<command_name>:<param1>,<param2>,...,<paramN><CR><LF>
  <return_code<CR><LF>
  ```

  Considerations:

  – <CR> and <LF> stand for carriage return and line feed.
  – When no value is returned, then only the return code will be returned.
  – Every command returns a status string, which is followed by <CR><LF>.

  Refer to chapter 7.2 for possible <return_code> values.

7.2 Return codes

On each command, the module answers with a return code to give a feedback on the requested operation status. This chapter describes the possible values of the return codes.

The return codes are valid for all operation modes unless otherwise specified.

7.2.1 OK

Description: The command ran successfully without errors.

7.2.2 AT_ERROR

Description: Generic error that is returned when the command is not implemented or an internal error occurred.

7.2.3 AT_PARAM_ERROR

Description: One or more parameters of the command are invalid.

7.2.4 AT_BUSY_ERROR

Description: The radio part of the module is currently busy with sending or receiving.

7.2.5 AT_TEST_PARAM_OVERFLOW

Description: The internal UART buffer of the module is filled and can no longer fit more commands to be processed.

7.2.6 AT_NO_NETWORK_JOINED

Description: No LoRaWAN® network is currently joined.

This error is only returned when Daphnis-I is in LoRaWAN® `MODE`.

7.2.7 AT_RX_ERROR

Description: An error was detected during the reception of the command on the `Application UART` interface, this could be due to some noise on the interface or some bytes have been missed in case the radio interrupt is taking a long time to be processed.

7.2.8 AT_DUTYCYCLE_RESTRICTED

Description: The duty cycle limit is reached.

This error is only returned when `AT+DCS` is off in LoRaWAN® `MODE`, or `P2PDCENFORCE` is on in P2P `MODE`.

7.2.9 AT_CRYPTO_ERROR

Description: An error occurred during encryption/decryption, or no encryption key was configured.

7.2.10 AT_INVALID_MODE

Description: The command cannot be executed in this `MODE`.

7.2.11 AT_INVALID_ROLE

Description: The command cannot be executed in this role. Refer to chapter `7.12.1`.

This error is only returned when Daphnis-I is in P2P `MODE`.

7.3 GPIO blocks structures

The following blocks are used by multiple commands and events that relate to GPIOs such as in chapter 7.5.8 and chapter 7.10.2.

When a GPIO block has more than 1 parameter, then a colon (:) is used as a delimiter to separate each parameter.

7.3.1 GPIO configuration block

Description: This block is used in GPIO commands that set or get the GPIO configurations.

GPIO configuration block parameters
<param1>: GPIO ID <param2>: GPIO function 0: Disconnected 1: Input 2: Output [<param3>]: GPIO value • Input 0: No pull resistor 1: Pull down resistor 2: Pull up resistor • Output 0: Low 1: High

Table 17: GPIO configuration block parameters

<param3> is an optional parameter that exists when the GPIO function is not disconnected.

7.3.2 GPIO value set block

Description: This block is used in GPIO commands that set the GPIO values.

GPIO value set block parameters
<param1>: GPIO ID <param2>: GPIO value • Output 0: Low 1: High

Table 18: GPIO value set block parameters

7.3.3 GPIO value get block

Description: This block is used in GPIO commands that get the GPIO values.

GPIO value get block parameters
<param1>: GPIO ID <param2>: GPIO status 0: Fail 1: Success [<param3>]: GPIO value • Input 0: Low 1: High • Output 0: Low 1: High

Table 19: GPIO value get block parameters

7.3.4 GPIO status block

Description: This block is used in GPIO commands to indicate the state of the GPIO pins.

GPIO status block parameters
<param1>: GPIO ID <param2>: GPIO status 0: Fail 1: Success

Table 20: GPIO status parameters

7.4 User/Runtime settings AT commands

The module's parameters also called as "User settings" are stored in flash. A local copy of the user settings also known as "Runtime settings" are stored on RAM. The "User settings" can be modified and read according to 7.4.1. The "Runtime settings" can be modified and read according to 7.4.2. The values stored in the flash are retained even after a power cycle.

The validity of the specified parameters is not verified. Incorrect values can result in device malfunction.

After the modification of the non-volatile parameters, a reset will be necessary for the changes to be applied.

When the configuration settings of the module are known in advance, an individualized firmware can be used to prevent in-field configuration. See chapter 11 for more information regarding firmware individualization.

7.4.1 Setting/Getting a User setting

Description: Get or set a User setting of Daphnis-I.

Command	Set Parameters	Get Parameters	Return Code
AT+<setting>US	Depends on the setting	None	OK AT_PARAM_ERROR AT_INVALID_MODE AT_ERROR
AT+<setting>US=?	None	Depends on the setting	OK AT_INVALID_MODE

7.4.2 Setting/Getting a Runtime setting

Description: Get or set a Runtime setting of Daphnis-I.

Command	Set Parameters	Get Parameters	Return Code
AT+<setting>RS	Depends on the setting	None	OK AT_PARAM_ERROR AT_INVALID_MODE AT_BUSY_ERROR
AT+<setting>RS=?	None	Depends on the setting	OK AT_INVALID_MODE

7.4.3 AT+RUS

Description: Restore user settings to their default values.

Command	Return Code
AT+RUS	OK AT_ERROR

Table 21: AT+RUS command

Example:

```
> AT+RUS
< OK
```

The default user settings are only applied after the module has been reset.

7.5 General AT commands

This chapter lists AT commands that are applicable to all application modes.

7.5.1 AT

Description: Check communication via UART with Daphnis-I.

Command	Return Code
AT	OK

Table 22: AT command

Example:

```
> AT
< OK
```

7.5.2 ATZ

Description: Reset Daphnis-I.

Command	Return Code
ATZ	OK

Table 23: ATZ command

Example:

```
> ATZ
< OK
```

7.5.3 AT+LTIME

Description: Get the local time.

Command	Return Value	Return Code
AT+LTIME=?	<param1>: Time following format XXhXXmXXs <param2>: Date in the following format dd/MM/yyyy	OK

Table 24: AT+LTIME command

Although this command can be used in all application modes, the returned time is only valid if you are in `Class B` or have at least once switched to `Class B` without resetting the module.

Example:

```
> AT+LTIME=?
< +LTIME:16h45m25s,15/05/2023
< OK
```

7.5.4 AT+BAT

Description: Get the battery level in mV.

Command	Return Value	Return Code
AT+BAT=?	<param1>: Battery level in mV	OK

Table 25: AT+BAT command

Example:

```
> AT+BAT=?
< +BAT:3155
< OK
```

7.5.5 AT+SLEEP

Description: Put Daphnis-I into `Sleep mode`.

Command	Return Code
AT+SLEEP	OK

Table 26: AT+SLEEP command

Example:

```
> AT+SLEEP
< OK
```

7.5.6 AT+SN

Description: Get the serial number of the module.

Command	Return Value	Return Code
AT+SN=?	<param1>: serial number 4 byte value separated by ':' in hex string format XX:XX:XX:XX	OK

Table 27: AT+SN command

Example:

```
> AT+SN=?
< +SN:30:92:45:D2
< OK
```

7.5.7 AT+FWVER

Description: Get the firmware version.

Command	Return Value	Return Code
AT+FWVER=?	<param1>: Firmware version in the following format VX.X.X	OK

Table 28: AT+FWVER command

Example:

```
> AT+FWVER=?
< +FWVER:V1.4.0
< OK
```

7.5.8 Local GPIO commands

The following commands use `GPIO blocks structures` as parameters. Refer to chapter `7.3` for more information.

7.5.8.1 AT+GPIOLCFGSET

Description: This command configures the free GPIOs of the radio module. This is necessary to allow local and remote GPIO control. As the configuration is stored in flash, it is retained after restarting the device.

The flash memory used to store these settings has a limited count of write cycles. Try to avoid performing repeated `AT+GPIOLCFGSET` as each command will use one write cycle. When the configuration of the I/O pins is known in advance, an individualized firmware can be used to prevent in-field configuration (see chapter `11`).

Command	Parameters	Return Value	Return Code

AT+GPIOLCFGSET	<param1>: `GPIO configuration block`
	...
	<paramN>: `GPIO configuration block`
		<param1>: `GPIO status block`
		...
		<paramN>: `GPIO status block`
			`OK`
			`AT_PARAM_ERROR`
			`AT_ERROR`

Table 29: AT+GPIOLCFGSET command

Example:

```
> AT+GPIOLCFGSET=1:1:1,4:2:1,6:0
< +GPIOLCFGSET:1:1,4:1,6:1
< OK
```

7.5.8.2 AT+GPIOLCFGGET

Description: This command reads the current configuration of the free GPIOs of the radio module.

Command	Return Value	Return Code
AT+GPIOLCFGGET=?	<param1>: `GPIO configuration block` ... <paramN>: `GPIO configuration block`	`OK`

Table 30: AT+GPIOLCFGGET command

Example:

```
> AT+GPIOLCFGGET=?
< +GPIOLCFGGET:0:0,1:1:1,2:0,3:0,4:2:1,5:0,6:0,7:0
< OK
```

7.5.8.3 AT+GPIOLVALUESET

Description: This command writes the free GPIOs of the local device. This command can be only run successfully if the respective pins of the local device have been configured correctly before.

Command	Parameters	Return Value	Return Code
AT+GPIOLVALUESET	<param1>: GPIO value set block ... <paramN>: GPIO value set block	<param1>: GPIO status block ... <paramN>: GPIO status block	OK AT_PARAM_ERROR

Table 31: AT+GPIOLVALUESET command

Example:

```
> AT+GPIOLVALUESET=1:1:1,4:2:1,6:0
< +GPIOLVALUESET:1:1,4:1,6:1
< OK
```

7.5.8.4 AT+GPIOLVALUEGET

Description: This command reads the free GPIOs of the local device. This command can only be run successfully, if the respective pins of the local device have been configured as output or input pins before.

Command	Parameters	Return Value	Return Code
AT+GPIOLVALUEGET	<param1>: GPIO ID ... <paramN>: GPIO ID	<param1>: GPIO value get block ... <paramN>: GPIO value get block	OK AT_PARAM_ERROR

Table 32: AT+GPIOLVALUEGET command

Example:

```
> AT+GPIOLVALUEGET=1,4
< +GPIOLVALUEGET:1:1:0,4:1:1
< OK
```

7.6 General events

This chapter lists events that are applicable to all application modes.

7.6.1 +SYSNOTF

Description: Daphnis-I ready to receive commands (boot up finished).

Event	Parameters
+SYSNOTF	<param1>: State READY

Table 33: +SYSNOTF Event

Example:

```
< +SYSNOTF:READY
```

7.7 General User/Runtime settings

This chapter lists User/Runtime settings that are applicable to all application modes. Please refer to chapter 7.4 for more information on how to use these settings.

7.7.1 UARTBAUDRATE

Description: This setting defines the UART baud rate of the `Application UART` interface.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
UARTBAUDRATE	<param1>: Baud rate **Default Value** 0 0: 9600 1: 57600 2: 115200[1]	Read/Write	Read

Table 34: UARTBAUDRATE setting

The newly selected baud rate is only applied after the module has been reset.

Example:

```
> AT+UARTBAUDRATEUS=1
< OK

> AT+UARTBAUDRATEUS=?
< +UARTBAUDRATEUS:1
< OK

> AT+UARTBAUDRATERS=?
< +UARTBAUDRATERS:0
< OK
```

7.7.2 MODE

Description: This setting defines the operation mode of Daphnis-I.

[1] Selecting this baud rate will increase the possibility of getting the `AT_RX_ERROR` error code when there is a lot of traffic in the network.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
MODE	<param1>: Operation mode **Default Value** 0 0: LoRaWAN® 1: P2P	Read/Write	Read

Table 35: MODE setting

Example:

```
> AT+MODEUS=1
< OK

> AT+MODEUS=?
< +MODEUS:1
< OK

> AT+MODERS=?
< +MODERS:0
< OK
```

7.8 LoRaWAN® AT commands

This chapter lists AT commands that are applicable to LoRaWAN® `MODE`.

7.8.1 LoRaWAN® stack context AT commands

7.8.1.1 AT+CS

Description: Store LoRaWAN® context to flash memory.

Command	Return Value	Return Code
AT+CS	<param1>: State NVM_DATA_STORED	`OK` `AT_ERROR`

Table 36: AT+CS command

Example:

```
> AT+CS
< +CS:NVM_DATA_STORED
< OK
```

7.8.1.2 AT+RFS

Description: Erase LoRaWAN® context stored in flash memory.

Command	Return Code
AT+RFS	`OK`

Table 37: AT+RFS command

Example:

```
> AT+RFS
< OK
```

This command restarts Daphnis-I after execution.

7.8.2 LoRaWAN® keys, IDs and EUIs AT commands

7.8.2.1 AT+APPEUI

Description: Get or set the application identifier (also known as join identifier).

Command	Parameters	Return Value	Return Code
AT+APPEUI	<param1>: APPEUI 8 byte value separated by ':' in hex string format XX:XX:XX:XX:XX:XX:XX:XX	None	OK AT_PARAM_ERROR AT_ERROR
AT+APPEUI=?	None	<param1>: APPEUI 8 byte value separated by ':' in hex string format XX:XX:XX:XX:XX:XX:XX:XX	OK AT_ERROR

Table 38: AT+APPEUI command

Example:

```
> AT+APPEUI=00:00:00:00:00:00:00:00
< OK

> AT+APPEUI=?
< +APPEUI:00:00:00:00:00:00:00:00
< OK
```

7.8.2.2 AT+APPKEY

Description: Set the application key.

Command	Parameters	Return Code
AT+APPKEY	<param1>: APPKEY in the following format 16 byte value separated by ':' in hex string format XX:XX:...:XX	OK AT_PARAM_ERROR AT_ERROR

Table 39: AT+APPKEY command

This key can only be set but not read back due to security reasons.

Example:

```
> AT+APPKEY=3F:76:A8:2B:97:E9:44:5B:4B:8D:F8:24:5F:A4:5A:20
< OK
```

7.8.2.3 AT+APPSKEY

Description: Set the application session key.

Command	Parameters	Return Code
AT+APPSKEY	<param1>: APPSKEY in the following format 16 byte value separated by ':' in hex string format XX:XX:...:XX	OK AT_PARAM_ERROR AT_ERROR

Table 40: AT+APPSKEY command

This key can only be set but not read back due to security reasons.

Example:

```
> AT+APPSKEY=81:CD:59:B3:0E:E8:75:3F:B7:F7:CC:FE:4B:D3:E9:66
< OK
```

7.8.2.4 AT+NWKSKEY

Description: Set the network session key.

Command	Parameters	Return Code
AT+NWKSKEY	<param1>: NWKSKEY in the following format 16 byte value separated by ':' in hex string format XX:XX:...:XX	OK AT_PARAM_ERROR AT_ERROR

Table 41: AT+NWKSKEY command

This key can only be set but not read back due to security reasons.

Example:

```
> AT+NWKSKEY=54:44:A3:15:76:CE:6A:71:BC:B7:5A:DF:B1:DB:03:D0
< OK
```

7.8.2.5 AT+DADDR

Description: Get or set the device address.

Command	Parameters	Return Value	Return Code
AT+DADDR	<param1>: DADDR 4 byte value separated by ':' in hex string format XX:XX:XX:XX	None	OK AT_PARAM_ERROR AT_ERROR
AT+DADDR=?	None	<param1>: DADDR 4 byte value separated by ':' in hex string format XX:XX:XX:XX	OK AT_ERROR

Table 42: AT+DADDR command

Example:

```
> AT+DADDR=30:92:45:D2
< OK

> AT+DADDR=?
< +DADDR:30:92:45:D2
< OK
```

7.8.2.6 AT+DEUI

Description: Get or set the device identifier.

Command	Parameters	Return Value	Return Code
AT+DEUI	<param1>: DEUI 8 byte value separated by ':' in hex string format XX:XX:...:XX	None	OK AT_PARAM_ERROR AT_ERROR
AT+DEUI=?	None	<param1>: DEUI 8 byte value separated by ':' in hex string format XX:XX:...:XX	OK AT_ERROR

Table 43: AT+DEUI command

Example:

```
> AT+DEUI=32:E4:5F:7D:50:4A:E6:8A
< OK

> AT+DEUI=?
< +DEUI:32:E4:5F:7D:50:4A:E6:8A
< OK
```

7.8.2.7 AT+NWKID

Description: Get or set the network ID.

Command	Parameters	Return Value	Return Code
AT+NWKID	<param1>: NWKID Integer value from 0 to 127	None	OK AT_PARAM_ERROR AT_ERROR
AT+NWKID=?	None	<param1>: NWKID Integer value from 0 to 127	OK AT_ERROR

Table 44: AT+NWKID command

Example:

```
> AT+NWKID=64
< OK

> AT+NWKID=?
< +NWKID:64
< OK
```

7.8.3 LoRaWAN® network AT commands

7.8.3.1 AT+LLVER

Description: Get the LoRaWAN® link layer version.

Command	Return Value	Return Code
AT+LLVER=?	<param1>: Link layer specification in the following format VX.X.X	OK

Table 45: AT+LLVER command

Example:

```
> AT+LLVER=?
< +LLVER:V1.0.4
< OK
```

7.8.3.2 AT+RPVER

Description: Get the regional parameters specification.

Command	Return Value	Return Code
AT+RPVER=?	<param1>: Regional parameters specification in the following format VX-X.X.X	OK

Table 46: AT+RPVER command

Example:

```
> AT+RPVER=?
< +RPVER:V2-1.0.1
< OK
```

7.8.3.3 AT+ADR

Description: Get or set adaptive data rate functionality.

Command	Parameters	Return Value	Return Code
AT+ADR	<param1>: ADR 0: ADR off 1: ADR on	None	OK AT_PARAM_ERROR

AT+ADR=?	None	<param1>: ADR **Default Value** 1 0: ADR off 1: ADR on	OK

Table 47: AT+ADR command

Example:

```
> AT+ADR=1
< OK

> AT+ADR=?
< +ADR:1
< OK
```

7.8.3.4 AT+DR

Description: Get or set TX data rate.

Command	Parameters	Return Value	Return Code
AT+DR	<param1>: Data rate Integer value from 0 to 7	None	OK AT_PARAM_ERROR AT_ERROR
AT+DR=?	None	<param1>: Data rate **Default Value** 0 Integer value from 0 to 7	OK AT_ERROR

Table 48: AT+DR command

This command can be used only if adaptive data rate is off.

Example:

```
> AT+DR=2
< OK

> AT+DR=?
< +DR:2
< OK
```

7.8.3.5 AT+DCS

Description: Get or set duty cycle restriction.

Command	Parameters	Return Value	Return Code

AT+DCS	<param1>: DCS
	0: DCS off
	1: DCS on
		None	`OK`
			`AT_PARAM_ERROR`
			`AT_ERROR`

AT+DCS=?	None	<param1>: DCS **Default Value** 1 0: DCS off 1: DCS on	`OK` `AT_ERROR`

Table 49: AT+DCS command

Resetting the module or entering `Sleep mode` will cause the module to reset the calculated time on air.

Example:

```
> AT+DCS=1
< OK

> AT+DCS=?
< +DCS:1
< OK
```

7.8.3.6 AT+JN1DL

Description: Get or set the join accept delay between the end of the join request TX and the first join receive window (RX1).

Command	Parameters	Return Value	Return Code
AT+JN1DL	<param1>: Delay in ms Integer value	None	OK AT_PARAM_ERROR AT_ERROR
AT+JN1DL=?	None	<param1>: Delay in ms **Default Value** 5000 Integer value	OK AT_ERROR

Table 50: AT+JN1DL command

Example:

```
> AT+JN1DL=7000
< OK

> AT+JN1DL=?
< +JN1DL:7000
< OK
```

7.8.3.7 AT+JN2DL

Description: Get or set the join accept delay between the end of the join request TX and the second join receive window (RX2).

Command	Parameters	Return Value	Return Code
AT+JN2DL	<param1>: Delay in ms Integer value	None	OK AT_PARAM_ERROR AT_ERROR
AT+JN2DL=?	None	<param1>: Delay in ms **Default Value** 6000 Integer value	OK AT_ERROR

Table 51: AT+JN2DL command

Example:

```
> AT+JN2DL=8000
< OK

> AT+JN2DL=?
< +JN2DL:8000
< OK
```

7.8.3.8 AT+RX1DL

Description: Get or set the delay between the end of the TX and the first receive window (RX1).

Command	Parameters	Return Value	Return Code
AT+RX1DL	<param1>: Delay in ms Integer value	None	OK AT_PARAM_ERROR AT_ERROR
AT+RX1DL=?	None	<param1>: Delay in ms **Default Value** 1000 Integer value	OK AT_ERROR

Table 52: AT+RX1DL command

Example:

```
> AT+RX1DL=2000
< OK

> AT+RX1DL=?
< +RX1DL:2000
< OK
```

7.8.3.9 AT+RX2DL

Description: Get or set the delay between the end of the TX and the second receive window (RX2).

Command	Parameters	Return Value	Return Code
AT+RX2DL	<param1>: Delay in ms Integer value	None	OK AT_PARAM_ERROR AT_ERROR
AT+RX2DL=?	None	<param1>: Delay in ms **Default Value** 2000 Integer value	OK AT_ERROR

Table 53: AT+RX2DL command

Example:

```
> AT+RX2DL=3000
< OK

> AT+RX2DL=?
< +RX2DL:3000
< OK
```

7.8.3.10 AT+RX2DR

Description: Get or set the data rate of the second receive window (RX2).

Command	Parameters	Return Value	Return Code
AT+RX2DR	<param1>: Data rate Integer value from 0 to 7	None	OK AT_PARAM_ERROR AT_ERROR
AT+RX2DR=?	None	<param1>: Data rate **Default Value** 0 Integer value from 0 to 7	OK AT_ERROR

Table 54: AT+RX2DR command

Example:

```
> AT+RX2DR=4
< OK

> AT+RX2DR=?
< +RX2DR:4
< OK
```

7.8.3.11 AT+RX2FQ

Description: Get or set the frequency of the second receive window (RX2).

Command	Parameters	Return Value	Return Code
AT+RX2FQ	<param1>: Frequency in Hz Integer value	None	OK AT_PARAM_ERROR AT_ERROR
AT+RX2FQ=?	None	<param1>: Frequency in Hz **Default Value** 869525000 Integer value	OK AT_ERROR

Table 55: AT+RX2FQ command

Example:

```
> AT+RX2FQ=869535000
< OK

> AT+RX2FQ=?
< +RX2FQ:869535000
< OK
```

7.8.3.12 AT+TXP

Description: Get or set the transmit power according to the regional parameters.

Command	Parameters	Return Value	Return Code
AT+TXP	<param1>: Transmit power Integer value from 0 to 7	None	OK AT_PARAM_ERROR AT_ERROR
AT+TXP=?	None	<param1>: Transmit power **Default Value** 0 Integer value from 0 to 7	OK AT_ERROR

Table 56: AT+TXP command

Example:

```
> AT+TXP=4
< OK

> AT+TXP=?
< +TXP:4
< OK
```

7.8.3.13 AT+PGSLOT

Description: Get or set the periodicity factor of the ping slot.

Command	Parameters	Return Value	Return Code
AT+PGSLOT	<param1>: Periodicity factor Integer value from 0 to 7	None	OK AT_PARAM_ERROR AT_ERROR
AT+PGSLOT=?	None	<param1>: Periodicity factor **Default Value** 4 Integer value from 0 to 7	OK AT_ERROR

Table 57: AT+PGSLOT command

Ping slot periodicity is calculated as $Periodicity = 2^{<param1>}$ seconds.

Example:

```
> AT+PGSLOT=5
< OK

> AT+PGSLOT=?
< +PGSLOT:5
< OK
```

7.8.3.14 AT+JOIN

Description: Join the LoRaWAN® network.

Command	Parameters	Return Code
AT+JOIN	<param1>: Join mode 0: ABP 1: OTAA	OK AT_PARAM_ERROR

Table 58: AT+JOIN command

This command triggers the +JOIN event.

Example:

```
> AT+JOIN=0
< OK

> AT+JOIN=1
< OK
```

7.8.3.15 AT+LINKC

Description: Piggyback LinkCheckReq MAC command on the next uplink.

Command	Return Code
AT+LINKC	OK AT_PARAM_ERROR

Table 59: AT+LINKC command

This command affects the output of the +RXINFO event.

Example:

```
> AT+LINKC
< OK
```

7.8.3.16 AT+SEND

Description: Send a packet to the LoRaWAN® network.

This command triggers the `+TXCONF` event when sending in confirmed mode.

Command	Parameters	Return Code
AT+SEND	<param1>: Frame port Integer value from 0 to 255 <param2>: Uplink confirmation 0: Unconfirmed 1: Confirmed <param3>: Payload in hex string format Max length is up to 242 bytes depending on the data rate XXXX...XX	OK AT_PARAM_ERROR AT_DUTYCYCLE_RESTRICTED AT_NO_NETWORK_JOINED AT_BUSY_ERROR AT_CRYPTO_ERROR AT_ERROR

Table 60: AT+SEND command

Example:

```
> AT+SEND=1:0:1A2B
< OK

> AT+SEND=1:1:1A2B
< OK
< +TXCONF
```

7.8.3.17 AT+CLASS

Description: Get or set Daphnis-I LoRaWAN® class.

Command	Parameters	Return Value	Return Code
AT+CLASS	<param1>: Class A: Class A B: Class B C: Class C	None	OK AT_ERROR AT_PARAM_ERROR AT_NO_NETWORK_JOINED
AT+CLASS=?	None	<param1>: Class A: Class A B0,B1,B2: Switch Class A to Class B in progress B: Class B C: Class C	OK

Table 61: AT+CLASS command

The class can be changed only after a network has been joined. In order to switch to Class B or Class C, the module needs to be in Class A.

This command triggers the +CLASSUPDATE event.

Example:

```
> AT+CLASS=C
< OK

> AT+CLASS=?
< +CLASS:C
< OK
```

7.8.4 LoRaWAN® certification AT commands

7.8.4.1 AT+CERTIF

Description: Start LoRaWAN® certification mode.

Command	Parameters	Return Code
AT+CERTIF	<param1>: Join mode 0: ABP 1: OTAA	OK AT_PARAM_ERROR AT_ERROR

Table 62: AT+CERTIF command

This command triggers +JOIN event.

Example:

```
> AT+CERTIF=0
< OK

> AT+CERTIF=1
< OK
```

7.9 LoRaWAN® events

This chapter lists events that are applicable to LoRaWAN® `MODE`.

7.9.1 LoRaWAN® stack context events

7.9.1.1 +CS

Description: LoRaWAN® context storage event.

Event	Parameters
+CS	<param1>: NVM_DATA_STORED NVM_DATA_RESTORED

Table 63: +CS Event

Example:

```
< +CS:NVM_DATA_RESTORED
```

7.9.2 LoRaWAN® network events

7.9.2.1 +JOIN

Description: Join LoRaWAN® network event.

Event	Parameters
+JOIN	<param1>: Join status JOINED: Successfully joined the network JOIN_FAILED: Failed to join the network

Table 64: +JOIN Event

Example:

```
< +JOIN:JOINED
```

7.9.2.2 +RXDATA

Description: Data received event.

Event	Parameters
+RXDATA	<param1>: Application port Integer value from 1 to 223 <param2>: Size in bytes in hex format Up to 242 bytes <param3>: Payload in hex string format XXXX...XX

Table 65: +RXDATA Event

Example:

```
< +RXDATA:223,01,65
```

7.9.2.3 +RXINFO

Description: Receive window slot information event. This event indicates the receive window slot at which the module received data, confirmation or other internal messages.

<param6> and <param7> are optional and only available after the AT+LINKC command was executed and an uplink was sent.

Event	Parameters
+RXINFO	<param1>: Receive slot window RX_1: Received on first Class A window RX_2: Received on second Class A window RX_C: Received on continuous Class C window RX_B: Received on ping Class B window <param2>: Frame port Integer value from 0 to 255 <param3>: Data rate Integer value from 0 to 7 <param4>: RSSI Received signal strength indicator Integer value 1 byte <param5>: SNR Signal-to-noise ratio Integer value 1 byte [<param6>]: Demodulated margin indicating the link margin in db of the last successfully received LinkCheckReq MAC command Integer value from 0 to 254 [<param7>]: Number of gateways that successfully received the last LinkCheckReq MAC command Integer value 1 byte

Table 66: +RXINFO Event

Example:

```
< +RXINFO:RX_1,0,5,−25,9

< +RXINFO:RX_C,223,0,−17,5

< +RXINFO:RX_1,0,5,−25,9,21,1
```

7.9.2.4 +TXCONF

Description: Uplink confirmation event. This event indicates that the network server successfully received the uplink, when the uplink was sent in confirmed mode.

Event
+TXCONF

Table 67: +TXCONF Event

Example:

```
< +TXCONF
```

7.9.2.5 +CLASSUPDATE

Description: Successfully switch to class event.

Event	Parameters
+CLASSUPDATE	<param1>: Class A: Class A B: Class B C: Class C

Table 68: +CLASSUPDATE Event

Example:

```
< +CLASSUPDATE:C
```

7.9.2.6 +BEACONINFO

Description: Received beacon information event.

Event	Parameters
+BEACONINFO	<param1>: Beacon window RX_BC <param2>: Data rate Integer value from 0 to 7 <param3>: RSSI Received signal strength indicator Integer value 1 byte <param4>: SNR Signal-to-noise ratio Integer value 1 byte <param5>: Frequency in Hz Integer value <param6>: Timestamp in seconds GPS Epoch Integer value <param7>: Info descriptor Integer value 1 byte, refer to [5] Section 13.3 <param8>: Info first 3 bytes in hex string format XXXXXX, refer to [5] Section 13.3 <param9>: Info second 3 bytes in hex string format XXXXXX, refer to [5] Section 13.3

Table 69: +BEACONINFO Event

Example:

```
< +BEACONINFO:RX_BC,3,−45,12,869525000,1368197120,0,000000,000000
```

7.9.2.7 +BEACONLOST

Description: No beacon was received in the past 120 minutes event. It will also revert to Class A from Class B and try to acquire a new beacon.

Event
+BEACONLOST

Table 70: +BEACONLOST Event

Example:

```
< +BEACONLOST
```

7.9.2.8 +BEACONNOTRECEIVED

Description: The beacon receive window did not receive any beacons within the beacon period.

Event
+BEACONNOTRECEIVED

Table 71: +BEACONNOTRECEIVED Event

Example:

```
< +BEACONNOTRECEIVED
```

7.10 P2P AT commands

This chapter lists AT commands that are applicable to P2P MODE.

7.10.1 P2P data commands

7.10.1.1 AT+P2PUNICASTTX

Description: This command can be used for unicast data transmission. The module sends data to the default destination address stored in the user settings parameter P2PMACDESTADDR. A payload length of maximum 224 bytes can be transmitted per packet.
When the data is processed by the module a +P2PTXCONF is output on the UART. Additionally, a +P2PTXRESP will follow as soon as the data has been transmitted over the air.
The receiving Daphnis-I will get a +P2PRXDATA message containing the transmitted payload data.

Command	Parameters	Return Code
AT+P2PUNICASTTX	<param1>: Payload in hex string format XXXX...XX	OK AT_BUSY_ERROR AT_PARAM_ERROR AT_DUTYCYCLE_RESTRICTED AT_CRYPTO_ERROR AT_INVALID_ROLE AT_ERROR

Table 72: AT+P2PUNICASTTX command

Example:

```
> AT+P2PUNICASTTX=1A2B
< OK
```

7.10.1.2 AT+P2PUNICASTEXTX

Description: This command can be used to transmit data to the destination address sent along with the packet. A payload length of maximum 224 bytes can be transmitted per packet.
When the data is processed by the module a +P2PTXCONF is output on the UART. Additionally, a +P2PTXRESP will follow as soon as the data has been transmitted over the air.
The receiving Daphnis-I will get a +P2PRXDATA message containing the transmitted payload data.

Command	Parameters	Return Code
AT+P2PUNICASTEXTX	<param1>: Destination address 4 byte value separated by ':' in hex string format XX:XX:XX:XX <param2>: Payload in hex string format XXXX...XX	OK AT_BUSY_ERROR AT_PARAM_ERROR AT_DUTYCYCLE_RESTRICTED AT_CRYPTO_ERROR AT_INVALID_ROLE AT_ERROR

Table 73: AT+P2PUNICASTEXTX command

Example:

```
> AT+P2PUNICASTEXTX=01:02:03:04,1A2B
< OK
```

7.10.1.3 AT+P2PMULTICASTTX

Description: This command provides the multicast data transmission to a group of modules configured with the same P2PMACGRPID. The module uses the default multicast group ID stored in the user settings parameter, P2PMACGRPID. A payload length of maximum 224 bytes can be transmitted per packet.
When the data is processed by the module a +P2PTXCONF is output on the UART. Additionally, a +P2PTXRESP will follow as soon as the data has been transmitted over the air.
The receiving Daphnis-I will get a +P2PRXDATA message containing the transmitted payload data.

Command	Parameters	Return Code
AT+P2PMULTICASTTX	<param1>: Payload in hex string format XXXX...XX	OK AT_BUSY_ERROR AT_PARAM_ERROR AT_DUTYCYCLE_RESTRICTED AT_CRYPTO_ERROR AT_INVALID_ROLE AT_ERROR

Table 74: AT+P2PMULTICASTTX command

Example:

```
> AT+P2PMULTICASTTX=1A2B
< OK
```

7.10.1.4 AT+P2PMULTICASTEXTX

Description: This command can be used to multicast data to a group ID specified in the packet. A payload length of maximum 224 bytes can be transmitted per packet.
When the data is processed by the module a `+P2PTXCONF` is output on the UART. Additionally, a `+P2PTXRESP` will follow as soon as the data has been transmitted over the air.
The receiving Daphnis-I will get a `+P2PRXDATA` message containing the transmitted payload data.

Command	Parameters	Return Code
AT+P2PMULTICASTEXTX	<param1>: Group ID Integer value from 0 to 255 <param2>: Payload in hex string format XXXX...XX	OK AT_BUSY_ERROR AT_PARAM_ERROR AT_DUTYCYCLE_RESTRICTED AT_CRYPTO_ERROR AT_INVALID_ROLE AT_ERROR

Table 75: AT+P2PMULTICASTEXTX command

Example:

```
> AT+P2PMULTICASTEXTX=01:02:03:04,1A2B
< OK
```

7.10.1.5 AT+P2PBROADCASTTX

Description: This command provides the simple broadcast data transmission. A payload length of maximum 224 bytes can be transmitted per packet.
When the data is processed by the module a `+P2PTXCONF` is output on the UART. Additionally, a `+P2PTXRESP` will follow as soon as the data has been transmitted over the air.
The receiving Daphnis-I will get a `+P2PRXDATA` message containing the transmitted payload data.

Command	Parameters	Return Code
AT+P2PMULTICASTTX	<param1>: Payload in hex string format XXXX...XX	OK AT_BUSY_ERROR AT_PARAM_ERROR AT_DUTYCYCLE_RESTRICTED AT_CRYPTO_ERROR AT_INVALID_ROLE AT_ERROR

Table 76: AT+P2PBROADCASTTX command

Example:

```
> AT+P2PBROADCASTTX=1A2B
< OK
```

7.10.1.6 AT+P2PRX

Description: This command sets and gets the receiving state of the radio.

Command	Parameters	Return Value	Return Code
AT+P2PRX	<param1>: Radio receive state 0: Radio receiving off 1: Radio receiving on	None	`OK` `AT_PARAM_ERROR`
AT+P2PRX=?	None	<param1>: Radio receive state **Default Value** 0 0: Radio receiving off 1: Radio receiving on	`OK`

Table 77: AT+P2PRX command

Example:

```
> AT+P2PRX=1
< OK
```

7.10.2 P2P remote GPIO commands

The following commands use `GPIO blocks structures` as parameters. Refer to chapter `7.3` for more information.

7.10.2.1 AT+P2PGPIORCFGSET

Description: This command configures the free GPIOs of the addressed remote device. This is necessary to allow local and remote GPIO control. As the configuration is stored in flash, it is retained after restarting the device. This command has to be addressed to a specific remote device with the 4-byte destination address.

The flash memory used to store these settings has a limited count of write cycles. Try to avoid performing repeated AT+GPIOLCFGSET as each command will use one write cycle. When the configuration of the I/O pins is known in advance, an individualized firmware can be used to prevent in-field configuration (see chapter 11).

Command	Parameters	Return Code
AT+P2PGPIORCFGSET	<param1>: Destination address 4 byte value separated by ':' in hex string format XX:XX:XX:XX <param2>: GPIO configuration block ... <paramN>: GPIO configuration block	OK AT_BUSY_ERROR AT_PARAM_ERROR AT_DUTYCYCLE_RESTRICTED AT_CRYPTO_ERROR AT_INVALID_ROLE AT_ERROR

Table 78: AT+P2PGPIORCFGSET command

Example:

```
> AT+P2PGPIORCFGSET=01:02:03:04,1:1:1,4:2:1,6:0
< +P2PGPIORCFGSET:01:02:03:04,1:1,4:1,6:1
< OK
```

7.10.2.2 AT+P2PGPIORCFGGET

Description: This command reads the current configuration of the free GPIOs of the addressed remote device.

Command	Parameters	Return Code
AT+P2PGPIORCFGGET	<param1>: Destination address 4 byte value separated by ':' in hex string format XX:XX:XX:XX	OK AT_BUSY_ERROR AT_PARAM_ERROR AT_DUTYCYCLE_RESTRICTED AT_CRYPTO_ERROR AT_INVALID_ROLE AT_ERROR

Table 79: AT+P2PGPIORCFGGET command

Example:

```
> AT+P2PGPIORCFGGET=01:02:03:04
< OK
```

7.10.2.3 AT+P2PGPIORVALUESET

Description: This command writes the free GPIOs of the addressed remote device. This command can be only run successfully if the respective pins of the remote device have been configured correctly in advance.

Command	Parameters	Return Code

AT+P2PGPIORVALUESET	<param1>: Destination address
	4 byte value separated by ':' in hex string format XX:XX:XX:XX
	
	<param2>: `GPIO value set block`
	...
	<paramN>: `GPIO value set block`
		`OK`
		`AT_BUSY_ERROR`
		`AT_PARAM_ERROR`
		`AT_DUTYCYCLE_RESTRICTED`
		`AT_CRYPTO_ERROR`
		`AT_INVALID_ROLE`
		`AT_ERROR`

Table 80: AT+P2PGPIORVALUESET command

Example:

```
> AT+P2PGPIORVALUESET=01:02:03:04,1:1:1,4:2:1,6:0
< OK
```

7.10.2.4 AT+P2PGPIORVALUEGET

Description: This command reads the free GPIOs of the addressed remote device. This command can only be run successfully if the respective pins of the remote device have been configured as output or input pins in advance.

Command	Parameters	Return Code
AT+P2PGPIORVALUEGET	<param1>: Destination address 4 byte value separated by ':' in hex string format XX:XX:XX:XX <param2>: GPIO ID ... <paramN>: GPIO ID	OK AT_BUSY_ERROR AT_PARAM_ERROR AT_DUTYCYCLE_RESTRICTED AT_CRYPTO_ERROR AT_INVALID_ROLE AT_ERROR

Table 81: AT+P2PGPIORVALUEGET command

Example:

```
> AT+P2PGPIORVALUEGET=01:02:03:04,1,4
< OK
```

7.11 P2P events

This chapter lists events that are applicable to P2P `MODE`.

7.11.1 P2P data events

7.11.1.1 +P2PTXCONF

Description: Data confirmation event. This event is sent by the module in response to any of the data transmit request messages.

Event	Parameters
+P2PTXCONF	<param1>: Status SUCCESS: Successfully sent out the packet BUSY: Channel is detected as busy (generated when `P2PLBT` is enabled) ERROR: Error when transmitting

Table 82: +P2PTXCONF Event

Example:

```
< +P2PTXCONF:SUCCESS
```

7.11.1.2 +P2PTXRESP

Description: Data response event. This event is output on the UART as soon as the data transmission is complete.

Event	Parameters
+P2PTXRESP	<param1>: Status SENT: Successfully sent out the packet. ACK: Sent packet was acknowledged. NACK: Sent packet was not acknowledged. [<param2>]: NACK error code 0: Timeout 1: NACK received 2: Internal Error

Table 83: +P2PTXRESP Event

Example:

```
< +P2PTXRESP:SENT

< +P2PTXRESP:ACK

< +P2PTXRESP:NACK,1
```

7.11.1.3 +P2PTXTIME

Description: Data transmission time event. This event is sent by the module when a data transmission occurs and it contains the time on air of the sent packet.

Event	Parameters
+P2PTXTIME	<param1>: Time on air in ms 4 byte unsigned decimal value

Table 84: +P2PTXTIME Event

Example:

```
< +P2PTXTIME:34
```

7.11.1.4 +P2PRXDATA

Description: Data reception event. This event indicates the reception of a valid data packet on the radio.

Event	Parameters
+P2PRXDATA	<param1>: P2P mac source address 4 byte value separated by ':' in hex string format XX:XX:XX:XX <param2>: RSSI value in dBm Integer value <param3>: Payload length in bytes in hex format <param4>: Payload in hex string format XXXX...XX

Table 85: +P2PRXDATA Event

Example:

```
< +P2PRXDATA:01:02:03:04,−70,02,1A2B
```

7.11.2 P2P remote GPIO events

7.11.2.1 +P2PGPIORCFGSET

Description: Remote GPIO set configuration confirmation event. This event indicates the status of setting the remote GPIO configuration of the remote device.

Event	Parameters
+P2PGPIORCFGSET	<param1>: P2P mac source address 4 byte value separated by ':' in hex string format XX:XX:XX:XX <param2>: RSSI value in dBm Integer value <param3>: Command status 0: Fail 1: Success 2: Configuration blocked <param4>: `GPIO status block` ... <paramN>: `GPIO status block`

Table 86: +P2PGPIORCFGSET Event

Example:

```
< +P2PGPIORCFGGET:01:02:03:04,−70,1,0:1,5:1

< +P2PGPIORCFGSET:01:02:03:04,−70,2
```

7.11.2.2 +P2PGPIORCFGGET

Description: Remote GPIO get configuration confirmation event. This event indicates the status of getting the remote GPIO configuration of the remote device.

Event	Parameters
+P2PGPIORCFGGET	<param1>: P2P mac source address 4 byte value separated by ':' in hex string format XX:XX:XX:XX <param2>: RSSI value in dBm Integer value <param3>: Command status 1: Success <param4>: `GPIO configuration block` ... <paramN>: `GPIO configuration block`

Table 87: +P2PGPIORCFGGET Event

Example:

```
< +P2PGPIORCFGGET:01:02:03:04,−70,1,0:0,1:1:1,2:0,3:0,4:2:1,5:0,6:0,7:0
```

7.11.2.3 +P2PGPIORVALUESET

Description: Remote GPIO set value confirmation event. This event indicates the status of setting the remote GPIO value of the remote device.

Event	Parameters
+P2PGPIORVALUESET	<param1>: P2P mac source address 4 byte value separated by ':' in hex string format XX:XX:XX:XX <param2>: RSSI value in dBm Integer value <param3>: Command status 1: Success <param4>: `GPIO status block` ... <paramN>: `GPIO status block`

Table 88: +P2PGPIORVALUESET Event

Example:

```
< +P2PGPIORVALUESET:01:02:03:04,−70,1,0:1,5:1
```

7.11.2.4 +P2PGPIORVALUEGET

Description: Remote GPIO get value confirmation event. This event indicates the status of getting the remote GPIO value of the remote device.

Event	Parameters
+P2PGPIORVALUEGET	<param1>: P2P mac source address 4 byte value separated by ':' in hex string format XX:XX:XX:XX <param2>: RSSI value in dBm Integer value <param3>: Command status 1: Success <param4>: `GPIO value get block` ... <paramN>: `GPIO value get block`

Table 89: +P2PGPIORVALUEGET Event

Example:

```
< +P2PGPIORVALUEGET:01:02:03:04,−70,1,1:1:0,4:1:1
```

7.11.2.5 +P2PGPIORCFGCHANGED

Description: Remote GPIO configuration changed event. This event indicates that the remote device has configured the free GPIOs of the radio module.

Note that only the GPIOs are part of this message, that have been configured successfully. Failed attempts of GPIO configurations will not be indicated by this message.

Event	Parameters
+P2PGPIORCFGCHANGED	<param1>: P2P mac source address 4 byte value separated by ':' in hex string format XX:XX:XX:XX <param2>: RSSI value in dBm Integer value <param3>: `GPIO configuration block` ... <paramN>: `GPIO configuration block`

Table 90: +P2PGPIORCFGCHANGED Event

Example:

```
< +P2PGPIORCFGCHANGED:01:02:03:04,-70,0:2:1,1:1:1
```

7.11.2.6 +P2PGPIORVALUECHANGED

Description: Remote GPIO value changed event. This event indicates that the remote device has set the values the free GPIOs of the radio module.

Note that only the GPIOs are part of this message, that have been configured successfully. Failed attempts of GPIO configurations will not be indicated by this message.

Event	Parameters
+P2PGPIORVALUECHANGED	<param1>: P2P mac source address 4 byte value separated by ':' in hex string format XX:XX:XX:XX <param2>: RSSI value in dBm Integer value <param3>: `GPIO value get block` ... <paramN>: `GPIO value get block`

Table 91: +P2PGPIORVALUECHANGED Event

Example:

```
< +P2PGPIORVALUECHANGED:01:02:03:04,−70,1:1:0,4:1:1
```

7.12 P2P User/Runtime settings

This chapter lists User/Runtime settings that are applicable to P2P MODE. Please refer to chapter 7.4 for more information on how to use these settings.

7.12.1 P2PROLE

Description: This setting defines the role of Daphnis-I when its in P2P MODE.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PROLE	<param1>: P2P role **Default Value** 0 0: Transceiver 1: Repeater	Read/Write	Read

Table 92: P2PROLE setting

Example:

```
> AT+P2PROLEUS=1
< OK

> AT+P2PROLEUS=?
< +P2PROLEUS:1
< OK

> AT+P2PROLERS=?
< +P2PROLERS:0
< OK
```

7.12.2 P2PDCTRACKER

Description: This setting, when enabled, will output the event +P2PTXTIME when the module transmits data on the radio. This message contains the time needed to transmit the radio data, which can be used by the host to track its duty cycle.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PDCTRACKER	**Default Value** 0 <param1>: P2P DC tracker state 0: Duty cycle tracker state off 1: Duty cycle tracker state on	Read/Write	Read

Table 93: P2PDCTRACKER setting

Example:

```
> AT+P2PDCTRACKERUS=1
< OK

> AT+P2PDCTRACKERUS=?
< +P2PDCTRACKERUS:1
< OK

> AT+P2PDCTRACKERRS=?
< +P2PDCTRACKERRS:0
< OK
```

7.12.3 P2PDCENFORCE

Description: This setting, when enabled, will internally keep track of the time on air of the packets sent and will prevent the module from sending more packets when the duty cycle limit is reached.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PDCENFORCE	<param1>: P2P DC enforce state **Default Value** 1 0: Duty cycle enforce state off 1: Duty cycle enforce state on	Read/Write	Read

Table 94: P2PDCENFORCE setting

Example:

```
> AT+P2PDCENFORCEUS=1
< OK

> AT+P2PDCENFORCEUS=?
< +P2PDCENFORCEUS:1
< OK

> AT+P2PDCENFORCERS=?
< +P2PDCENFORCERS:0
< OK
```

7.12.4 P2PRFPROFILE

Description: This setting determines the modulation and coding used on the physical layer of the module.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PRFPROFILE	<param1>: P2P radio profile **Default Value** 5 Integer value from 0 to 7	Read/Write	Read

Table 95: P2PRFPROFILE setting

param1	Radio Profile
0	LoRa® (SF12BW125)
1	LoRa® (SF11BW125)
2	LoRa® (SF10BW125)
3	LoRa® (SF9BW125)
4	LoRa® (SF8BW125)
5	LoRa® (SF7BW125)
6	LoRa® (SF7BW250)
7	FSK (50 kbps)

Example:

```
> AT+P2PRFPROFILEUS=2
< OK

> AT+P2PRFPROFILEUS=?
< +P2PDCENFORCEUS:2
< OK

> AT+P2PRFPROFILERS=?
< +P2PRFPROFILERS:7
< OK
```

7.12.5 P2PTXPOWER

Description: This setting determines the output power in dBm of the module.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PTXPOWER	<param1>: P2P tx power in dBm **Default Value** 14 Integer value from 0 to 14	Read/Write	Read/Write

Table 96: P2PTXPOWER setting

Example:

```
> AT+P2PTXPOWERUS=13
< OK

> AT+P2PTXPOWERUS=?
< +P2PTXPOWERUS:13
< OK

> AT+P2PTXPOWERRS=5
< OK

> AT+P2PTXPOWERRS=?
< +P2PTXPOWERRS:5
< OK
```

7.12.6 P2PRFCHANNEL

Description: This parameter defines the radio channel and hence the frequency of operation of the radio module. The dependence between channel and frequency is as follows:

$$Channel_{RF} = \frac{Frequency_{RF} - 863\ MHz}{0.050\ MHz} \quad (1)$$

or

$$Frequency_{RF} = 863\ MHz + Channel_{RF} \times 0.050\ MHz \quad (2)$$

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PRFCHANNEL	<param1>: P2P radio channel **Default Value** 64 Integer value from 0 to 140	Read/Write	Read/Write

Table 97: P2PRFCHANNEL setting

Example:

```
> AT+P2PRFCHANNELUS=13
< OK

> AT+P2PRFCHANNELUS=?
< +P2PRFCHANNELUS:13
< OK

> AT+P2PRFCHANNELRS=5
< OK

> AT+P2PRFCHANNELRS=?
< +P2PRFCHANNELRS:5
< OK
```

7.12.7 P2PLBT

Description: This value determines if the listen before talk is enabled or disabled. Enabling LBT makes the module perform a detection (busy/free) before transmission. Refer to chapter 5.4.4 for further details.

P2PRFPROFILE 6 does not support LBT.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PLBT	<param1>: P2P LBT state **Default Value** 0 0: LBT is not enabled 1: LBT is enabled	Read/Write	Read

Table 98: P2PLBT setting

Example:

```
> AT+P2PLBTUS=1
< OK

> AT+P2PLBTUS=?
< +P2PLBT:1
< OK

> AT+P2PLBTRS=?
< +P2PLBTRS:0
< OK
```

7.12.8 P2PMACSRCADDR

Description: This setting contains the source address of the radio module. The address is included in the radio packets transmitted by the module in P2P mode and has to be unique in a network of devices. If not configured (i.e. if set to the default value), the module uses its serial number (refer to `AT+SN`) as its source address.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PMACSRCADDR	<param1>: P2P mac source address **Default Value** FF:FF:FF:FF 4 byte value separated by ':' in hex string format XX:XX:XX:XX	Read/Write	Read

Table 99: P2PMACSRCADDR setting

Example:

```
> AT+P2PMACSRCADDRUS=01:02:03:04
< OK

> AT+P2PMACSRCADDRUS=?
< +P2PMACSRCADDRUS:01:02:03:04
< OK

> AT+P2PMACSRCADDRRS=?
< +P2PMACSRCADDRRS:05:06:07:08
< OK
```

7.12.9 P2PMACDESTADDR

Description: This setting contains the default destination address of the radio module. The address is used by default as the destination address, when sending data using `AT+P2PUNICASTTX`.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PMACDESTADDR	<param1>: P2P mac destination address **Default Value** FF:FF:FF:FF 4 byte value separated by ':' in hex string format XX:XX:XX:XX	Read/Write	Read

Table 100: P2PMACDESTADDR setting

Example:

```
> AT+P2PMACDESTADDRUS=01:02:03:04
< OK

> AT+P2PMACDESTADDRUS=?
< +P2PMACDESTADDRUS:01:02:03:04
< OK

> AT+P2PMACDESTADDRRS=?
< +P2PMACDESTADDRRS:05:06:07:08
< OK
```

7.12.10 P2PMACGRPID

Description: This setting contains the default destination group ID of the radio module. This group ID is used by default as the multicast address, when sending data using `AT+P2PMULTICASTTX`.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PMACGRPID	<param1>: P2P mac group ID **Default Value** 0 Integer value from 0 to 255	Read/Write	Read

Table 101: P2PMACGRPID setting

The value 255 is a special value which will trigger any compatible receiver to interpret this frame to forward it to its host.

Example:

```
> AT+P2PMACGRPIDUS=4
< OK

> AT+P2PMACGRPIDUS=?
< +P2PMACGRPIDUS:4
< OK

> AT+P2PMACGRPIDRS=?
< +P2PMACGRPIDRS:7
< OK
```

7.12.11 P2PMACENCMODE

Description: This parameter determines the level of encryption on the module in P2P mode.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PMACENCMODE	<param1>: P2P mac encryption mode **Default Value** 0	Read/Write	Read

Table 102: P2PMACENCMODE setting

param1	Transmission	Reception
0	Unencrypted	Unencrypted and encrypted packets (with matching key).
1	Encrypted	Unencrypted and encrypted packets (with matching key).
2	Unencrypted	Encrypted packets only (unencrypted and non-decryptable packets are discarded).
3	Encrypted	Encrypted packets only (unencrypted and non-decryptable packets are discarded).

Example:

```
> AT+P2PMACENCMODEUS=1
< OK

> AT+P2PMACENCMODEUS=?
< +P2PMACENCMODEUS:1
< OK

> AT+P2PMACENCMODERS=?
< +P2PMACENCMODERS:2
< OK
```

7.12.12 P2PMACENCKEY

Description: This setting contains the key used for encryption/decryption of the user payload. A key consisting of 16 times 0x00 is considered as "no key set". Thus, write 16 times 0x00 to delete your key.

Setting Name	Set Parameters	Get Parameters	User setting permission	Runtime setting permission
P2PMACENCKEY	<param1>: P2P mac encryption key 16 byte value separated by ':' in hex string format XX:XX:...:XX	<param1>: P2P mac encryption key state **Default Value** 0 0: Encryption key is not configured 1: Encryption key is configured	Read/Write	Read

Table 103: P2PMACENCKEY setting

Example:

```
> AT+P2PMACENCKEYUS=01:01:01:01:01:01:01:01:01:01:01:01:01:01:01:01
< OK

> AT+P2PMACENCKEYUS=?
< +P2PMACENCKEYUS:1
< OK

> AT+P2PMACENCKEYRS=?
< +P2PMACENCKEYRS:0
< OK
```

7.12.13 P2PMACTTL

Description: This defines the time-to-live value of the packets transmitted over the radio. The time-to-live packet defines the maximum number of hops that a packet can make before being discarded. A limited number of hops prevents indefinite forwarding of a packet over a network with multiple repeaters.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PMACTTL	<param1>: P2P mac TTL **Default Value** 3 Integer value from 0 to 255	Read/Write	Read

Table 104: P2PMACTTL setting

Example:

```
> AT+P2PMACTTLUS=4
< OK

> AT+P2PMACTTLUS=?
< +P2PMACTTLUS:4
< OK

> AT+P2PMACTTLRS=?
< +P2PMACTTLRS:7
< OK
```

7.12.14 P2PMACACK

Description: This setting when enabled, will prompt the receiver module to automatically send a wireless acknowledgment ("ACK"). Please note that sending acknowledgments additionally increases the traffic and will have influence on the duty-cycle, which can be crucial for CE compliance.

When the `P2PMACACK` setting is enabled, after a packet transmission that uses ACKs is transmitted receiving will start regardless of the state of `AT+P2PRX`. However after the ACK is either received or it times out, the state of `AT+P2PRX` is checked again and applied.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PMACACK	<param1>: P2P MAC acknowledgment **Default Value** 1 0: Acknowledgments are not enabled 1: Acknowledgments are enabled	Read/Write	Read

Table 105: P2PMACACK setting

Example:

```
> AT+P2PMACACKUS=1
< OK

> AT+P2PMACACKUS=?
< +P2PMACACKUS:1
< OK

> AT+P2PMACACKRS=?
< +P2PMACACKRS:0
< OK
```

7.12.15 P2PRPTHRESH

Description: This setting determines the threshold used, when the repeater role is used. In case the repeater is enabled, the radio frame is only relayed if its RSSI value is below this threshold. Otherwise, the message is discarded.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PRPTHRESH	<param1>: P2P repeater threshold in dBm **Default Value** 0 Integer value from -140 to 0	Read/Write	Read

Table 106: P2PRPTHRESH setting

Example:

```
> AT+P2PRPTHRESHUS=-100
< OK

> AT+P2PRPTHRESHUS=?
< +P2PRPTHRESHUS:-100
< OK

> AT+P2PRPTHRESHRS=?
< +P2PRPTHRESHRS:-20
< OK
```

7.12.16 P2PRPNUMSLOTS

Description: This value contains the number of time slots to be used for the packet repetition. When using several repeater devices in a single network, repeated data packets may collide on the radio channel, when all repeater devices send the received packet at the same time. To avoid collisions, the frequency channel is divided in `P2PRPNUMSLOTS` time slots, where each repeater chooses a certain slot randomly.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PRPNUMSLOTS	<param1>: P2P repeater number of slots **Default Value** 32 Integer value from 1 to 255	Read/Write	Read

Table 107: P2PRPNUMSLOTS setting

This calculation assumes exactly one radio frame over the whole duration of the repeating, i.e. a low throughput scenario. With higher throughput requirements, the Poisson distribution of frames together with the selected channel access scheme, needs to be included in the calculation.

The smallest number of time slots that is needed, depends on the network structure and the number of repeaters used. Assuming there are `NumRP` repeater devices in the range of a sending device and there is only one radio packet present over the entire duration of all repeating, the probability of at least two of the same repeated packet collides can be calculated by:

$$1 - \frac{RP\_NumSlots!}{RP\_NumSlots^{NumRP} \times (RP\_NumSlots - NumRP)!}$$

Common values are:

NumRP	P2PRPNUMSLOTS	Collision probability
2	32	3.1%
3	32	9.2%
4	32	17.7%
5	64	14.8%
6	64	21.5%
7	128	15.4%

Example:

```
> AT+P2PRPNUMSLOTSUS=32
< OK

> AT+P2PRPNUMSLOTSUS=?
< +P2PRPNUMSLOTSUS:32
< OK

> AT+P2PRPNUMSLOTSRS=?
< +P2PRPNUMSLOTSRS:16
< OK
```

7.12.17 P2PCMDRESPTIMEOUT

Description: This setting configures the timeout of radio command responses for the following commands: `AT+P2PGPIORCFGSET`, `AT+P2PGPIORCFGGET`, `AT+P2PGPIORVALUESET` and `AT+P2PGPIORVALUEGET`.

When the `P2PCMDRESPTIMEOUT` setting is enabled, after a radio command is transmitted receiving will start regardless of the state of `AT+P2PRX`. However after the radio command response is either received or it times out, the state of `AT+P2PRX` is checked again and applied.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PCMDRESPTIMEOUT	<param1>: P2P command response timeout in ms **Default Value** 3000 0: Command response timeouts are disabled 20 . . . 65535: Command response timeouts are enabled with the specified timeout.	Read/Write	Read

Table 108: P2PCMDRESPTIMEOUT setting

Example:

```
> AT+P2PCMDRESPTIMEOUTUS=5000
< OK

> AT+P2PCMDRESPTIMEOUTUS=?
< +P2PCMDRESPTIMEOUTUS:5000
< OK

> AT+P2PCMDRESPTIMEOUTRS=?
< +P2PCMDRESPTIMEOUTRS:2000
< OK
```

7.12.18 P2PGPIORBLOCKCFG

Description: The Daphnis-I module allows configuration and control of some of its digital I/O pins remotely. This parameter, when enabled, blocks remote configuration of these digital I/O pins.

Setting Name	Set & Get Parameters	User setting permission	Runtime setting permission
P2PGPIORBLOCKCFG	<param1>: P2P GPIO remote block configuration state **Default Value** 1 0: P2P remote gpio configuration is blocked 1: P2P remote gpio configuration is not blocked	Read/Write	Read

Table 109: P2PGPIORBLOCKCFG setting

Example:

```
> AT+P2PGPIORBLOCKCFGUS=1
< OK

> AT+P2PGPIORBLOCKCFGUS=?
< +P2PGPIORBLOCKCFG:1
< OK

> AT+P2PGPIORBLOCKCFGRS=?
< +P2PGPIORBLOCKCFGRS:0
< OK
```

7.13 Radio test AT commands

This chapter lists AT commands to test the radio interface regardless of the specified application mode.

7.13.1 AT+TTONE

Description: Start RF tone test.

Command	Return Code
AT+TTONE	OK AT_BUSY_ERROR AT_ERROR

Table 110: AT+TTONE command

Example:

```
> AT+TTONE
< OK
```

7.13.2 AT+TRSSI

Description: Start RF RSSI tone test.

Command	Return Value	Return Code
AT+TRSSI	<param1>: RSSI Integer value 1 byte	OK AT_BUSY_ERROR AT_ERROR

Table 111: AT+TRSSI command

Example:

```
> AT+TRSSI
< +TRSSI:-114
< OK
```

7.13.3 AT+TCONF

Description: Get or set the test configuration.

When setting the test configuration, unused parameters still need to be defined as shown in the example.

Command	Parameters	Return Value	Return Code
AT+TCONF	<param1>: Frequency in Hz Integer value <param2>: Transmit power in dBm Integer value from 0 to 14 <param3>: Bandwidth • FSK in Hz Integer value from 4800 to 467000 • LoRa® in kHz 0: 7.8125 1: 15.625 2: 31.25 3: 62.5 4: 125 5: 250 6: 500 • BPSK in Hz Not used • MSK in Hz Integer value from 4800 to 467000 <param4>: • FSK data rate in bps Integer value from 600 to 300000 • LoRa® spreading factor Integer value from 5 to 12	None	OK AT_PARAM_ERROR

AT+TCONF	• BPSK data rate in bps Integer value from 1 to 1000 • MSK data rate in bps Integer value from 100 to 300000 <param5>: Coding rate • FSK Not used • LoRa® 4/5 4/6 4/7 4/8 • BPSK Not used • MSK Not used <param6>: Low-noise amplifier 0: off 1: on <param7>: PA boost 0: off 1: on <param8>: Modulation 0: FSK 1: LoRa® 2: BPSK (TX) 3: MSK <param9>: Payload length in bytes Integer value from 1 to 255	None	OK AT_PARAM_ERROR

AT+TCONF	<param10>: Frequency deviation • FSK frequency in MHz Integer value from 600 to 200000 • LoRa® Not used • BPSK Not used • MSK Not used <param11>: Low DR optimization • FSK Not used • LoRa® 0: 0ff 1: on 2: Auto (1 when SF11 or SF12, 0 otherwise) • BPSK Not used • MSK Not used <param12>: Gaussian BT product • FSK, MSK 0: No filter applied 1: BT = 0.3 2: BT = 0.5 3: BT = 0.7 4: BT = 1 • LoRa® Not used • BPSK Not used	None	OK AT_PARAM_ERROR

AT+TCONF=?	None	<param1>: Frequency in Hz Integer value <param2>: Transmit power in dBm Integer value from 0 to 14 [<param3>]: Bandwidth • FSK Integer value • LoRa® 0(7812) 1(15625) 2(31250) 3(62500) 4(125000) 5(250000) 6(500000) • BPSK Not returned • MSK Integer value <param4>: • FSK data rate in bps Integer value • LoRa® spreading factor Integer value from 5 to 12 • BPSK data rate in bps Integer value • MSK data rate in bps Integer value	OK

AT+TCONF=?	None	[<param5>]: Coding rate • FSK NA • LoRa® 1(=4/5) 2(=4/6) 3(=4/7) 4(=4/8) • BPSK Not returned • MSK NA [<param6>]: Low-noise amplifier • FSK, LoRa®, MSK 0: off 1: on • BPSK Not returned [<param7>]: PA boost • FSK, LoRa®, MSK 0: off 1: on • BPSK Not returned [<param8>]: Modulation • FSK FSK • LoRa® LoRa® • BPSK Not returned • MSK MSK	OK

AT+TCONF=?	None	[<param9>]: Payload length in bytes • FSK, LoRa®, MSK Integer value • BPSK Not returned [<param10>]: Frequency deviation • FSK Integer value • LoRa® NA • BPSK Not returned • MSK FSK_DR/4 [<param11>]: Low DR optimization • FSK NA • LoRa® 0: 0ff 1: on 2: Auto (1 when SF11 or SF12, 0 otherwise) • BPSK Not returned • MSK NA [<param12>]: Gaussian BT product • FSK, MSK 0: No filter applied 1: BT = 0.3 2: BT = 0.5 3: BT = 0.7 4: BT = 1	OK

AT+TCONF=?	None	• LoRa® Not used • BPSK Not used	OK

Table 112: AT+TCONF command

Example:

```
> AT+TCONF=868000000:14:50000:50000:4/5:0:0:0:16:25000:2:3 // FSK
< OK

> AT+TCONF=868000000:14:4:12:4/5:0:0:1:16:25000:2:3 // LoRa®
< OK

> AT+TCONF=868000000:14:10000:1000:4/5:0:0:2:16:25000:2:3 // BPSK
< OK

> AT+TCONF=868000000:14:10000:10000:4/5:0:0:3:16:25000:2:3 // MSK
< OK

> AT+TCONF=?
< +TCONF:868000000,14,4(125000),12,1(=4/5),0,0,LORA,16,NA,2,NA
< OK
```

7.13.4 AT+TTX

Description: Start RF TX test with the specified number of packets.

This command will only respond after all packets have been sent.

Command	Parameters	Return Value	Return Code
AT+TTX	<param1>: Number of packets Integer value	<param1>: Success rate Integer value from 0 to 100	OK AT_BUSY_ERROR AT_ERROR

Table 113: AT+TTX command

Example:

```
> AT+TTX=3
< +TTX:100
< OK
```

7.13.5 AT+TRX

Description: Start RF RX test with the specified number of expected packets (to be used with `AT+TTX`).

This command will only respond after all packets have been either received or timed out.

Command	Parameters	Return Value	Return Code

AT+TRX	<param1>: Number of packets
	Integer value
		<param1>: Success rate
		Integer value from 0 to 100
		
		<param2>: Average RSSI
		Integer value
		
		<param3>: Average SNR
		Integer value
		
		<param4>: Average Sensitivity
		Integer value
			`OK`
			`AT_BUSY_ERROR`
			`AT_ERROR`

Table 114: AT+TRX command

Example:

```
> AT+TRX=3
< +TRX:100,-7,5,-2
< OK
```

7.13.6 AT+TTH

Description: Start RF TX hopping test.

Command	Parameters	Return Value	Return Code
AT+TTH	<param1>: Start frequency in Hz Integer value <param2>: Stop frequency in Hz Integer value <param3>: Frequency delta in Hz Integer value <param4>: Number of packets Integer value	<param1>: Success rate Integer value from 0 to 100	OK AT_BUSY_ERROR AT_ERROR

Table 115: AT+TTH command

Example:

```
> AT+TTH=868000000,868500000,100000,6
< +TTH:100
< OK
```

7.13.7 AT+TOFF

Description: Stop RF tone test.

Other radio tests once started cannot be stopped using this command.

Command	Return Code
AT+TOFF	OK AT_BUSY_ERROR AT_ERROR

Table 116: AT+TOFF command

Example:

```
> AT+TOFF
< OK
```

7.14 Legacy AT commands

This chapter lists AT commands that are still available to be used however its recommended to use the non legacy commands.

7.14.1 AT+UARTBAUDRATE

Description: Get or set the UART baud rate of Daphnis-I.

This legacy command is replaced by the setting `UARTBAUDRATE`.

The newly selected baud rate is only applied after the module has been reset.

Command	Parameters	Return Value	Return Code
AT+UARTBAUDRATE	<param1>: Baud rate 0: 9600 1: 57600 2: 115200[1]	None	`OK` `AT_PARAM_ERROR`
AT+UARTBAUDRATE=?	None	<param1>: Baud rate **Default Value** 0 0: 9600 1: 57600 2: 115200[1]	`OK`

Table 117: AT+UARTBAUDRATE command

Example:

```
> AT+UARTBAUDRATE=0
< OK

> AT+UARTBAUDRATE=?
< +UARTBAUDRATE:0
< OK
```

[1] Selecting this baud rate will increase the possibility of getting the `AT_RX_ERROR` error code when there is a lot of traffic in the network.

7.14.2 AT+VER

Description: Get the firmware version, LoRaWAN® link layer and regional parameters specifications.

This legacy command is replaced by the AT commands: `AT+FWVER`, `AT+LLVER` and `AT+RPVER`.

Command	Return Value	Return Code
AT+VER=?	<param1>: Firmware version in the following format VX.X.X <param2>: Link layer specification in the following format VX.X.X <param3>: Regional parameters specification in the following format VX-X.X.X	`OK`

Table 118: AT+VER command

Example:

```
> AT+VER=?
< +VER:V1.3.0,V1.0.4,V2-1.0.1
< OK
```

8 Timing parameters

In this chapter, the timing parameters of the module are listed and described in detail.

8.1 Power up

After powering the module, the */RESET* pin shall be held low for another Δt_1 of 1 ms after the *VDD* is stable to ensure a safe start-up. The module will send a `+SYSNOTF` event (+SYSNOTF:READY) to indicate "ready for operation" after Δt_2 of about 1125 ms from releasing the */RESET* pin.

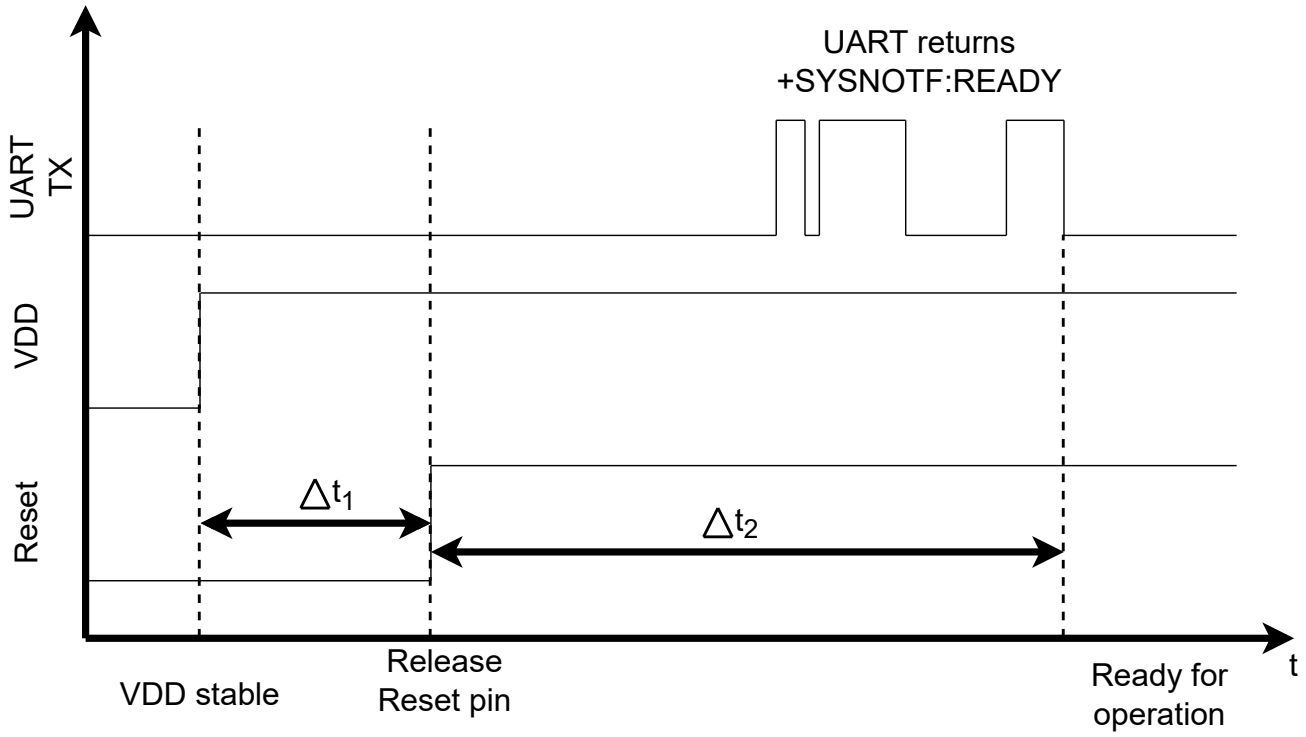

Figure 39: Power up

8.2 P2P data throughput measurements

The host micro-controller sends a data transmission request with one of the command AT+P2PBROADCASTTX. The module buffers the data and acknowledges the same over UART with a +P2PTXCONF event. A data transmission is triggered on completion of which the host receives a +P2PTXRESP event. The figure 40 shows the sequence of data transmission.

The width of the boxes in the following figures does not represent the duration it takes to send the data over UART, it is just to show the sequence of the messages.

8.2.1 Test conditions

Condition	Value
Host MCU	STM32 Nucleo-F401RE
UARTBAUDRATE [bps]	115200
P2PDCENFORCE	Off
Data transmission command	AT+P2PBROADCASTTX
Packet count	100
Payload length [Bytes]	224

Table 119: Test conditions

8.2.2 Test result

This chapter contains the result of the test according to the conditions specified in 8.2.1. The throughput is calculated as the following:

$$Throughput\ [kbps] = (8 \times Payload\ length)/(T_3 - T_0) \quad (3)$$

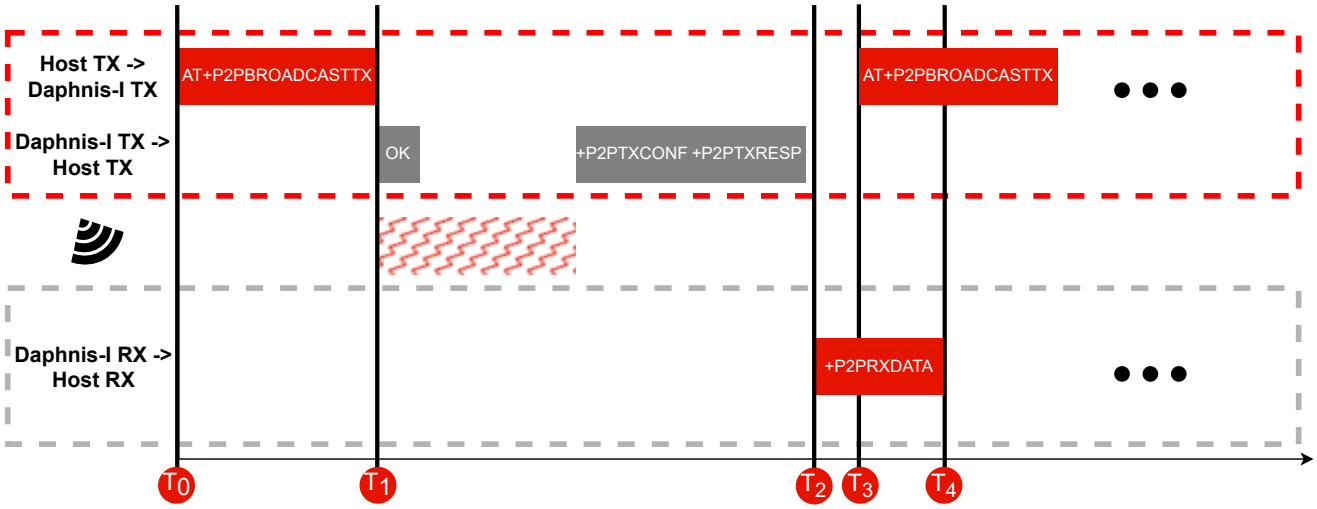

Figure 40: Command sequence when transmitting data

P2PRFPROFILE	T_1-T_0 [ms] (UART RX)	T_3-T_0 [ms] (Host TX Period)	T_4-T_2 [ms] (UART TX)	T_4-T_0 [ms] (End-to-end delay)	Average throughput [kbps]
0	40.28	8421.22	41.68	8448.14	**0.21**
1	40.28	4649.86	41.68	4701.10	**0.38**
2	40.28	2148.94	41.68	2192.93	**0.83**
3	40.28	1204.94	41.68	1246.26	**1.47**
4	40.28	702.36	41.68	742.27	**2.53**
5	40.28	426.87	41.68	464.69	**4.16**
6	40.28	241.16	41.68	279.90	**7.36**
7	40.28	96.75	41.68	134.70	**18.47**

Table 120: Maximum throughput timings, packet error rate = 0%

9 Radio parameters

9.1 P2P radio parameters

The default radio parameters are determined by the values of P2PRFPROFILE, P2PRFCHANNEL and P2PTXPOWER in the user settings. These non-volatile parameters can be modified using Setting/Getting a User setting. To modify their volatile counterparts refer to chapter Setting/Getting a Runtime setting.

STOP

The parameter must be chosen with prudence to reach good functionality and compliance with valid regulatory requirements such as the EN 300 220 in the European Union or the FCC in the United States of America. The user of this module is solely responsible for adhering to all applicable statutory regulations.

The module is tested through an accredited test lab to conform to the requirements of the RED. The module complies to receiver category 2.

This means, that mounted on its EV-board it complies to the requirements of the RED. However, conformance of the end-device depends also but not limited to the radiated power of the end-device. Which strongly depends on the selected antenna, the pcb design, the wiring to the antenna and the quality of the power supply. Thus, it is highly recommended that the end-device manufacturer verifies the radiated power characteristic on the end application.
An important aspect to comply with the radio regulatory is to adhere to the requirements of the duty cycle. As defined in EN 300 220, the duty cycle is the ratio expressed as a percentage of the cumulative duration of transmission T_{on} within an observation time interval of T_{obs}. $DC = (\frac{T_{on}}{T_{obs}})_{F_{obs}}$ on an observation bandwidth F_{obs}. Unless otherwise specified, T_{obs} is 1 hour and the observation bandwidth F_{obs} is the operational frequency band.

STOP

The setting P2PDCENFORCE can be enabled to constrain the duty cycle. However, the user is fully responsible for the compliance of the duty cycle and implementing according mechanisms in the end-device. The setting P2PDCTRACKER can be enabled to trigger the +P2PTXTIME which returns the consumed time, when data has been transmitted on radio. This can be used to track the duty cycle.

The frequency channels of the module can be selected from a 50 kHz grid. Not all channels are permissible, depending on the selected profile, output power and antenna (radio profile, radio TX power and radio channel).

STOP

Depending on the chosen radio profile the channel spacing declared in table 121, chapter 9.1.1 has to be applied.

9.1.1 Channel assignment

Sub Band	Channel	Frequency [MHz]	Profile 0-5	Profile 6	Profile 7
	Required channel spacing		150 kHz	275 kHz	125 kHz
Band K 863 MHz - 865 MHz $\leq$ 14 dBm The whole band except for audio & video applications limited to 300 kHz duty cycle $\leq$ 0.1% or PSA	0	863.00	Band limit		
	1	863.05	-	-	-
	2	863.10	K	-	K
	3	863.15	*	K	*
	4	863.20	*	*	*
	5	863.25	K	*	K
	6	863.30	*	*	*
	7	863.35	*	*	*
	8	863.40	K	*	K
	9	863.45	*	K	*
	10	863.50	*	*	*
	11	863.55	K	*	K
	12	863.60	*	*	*
	13	863.65	*	*	*
	14	863.70	K	*	K
	15	863.75	*	K	*
	16	863.80	*	*	*
	17	863.85	K	*	K
	18	863.90	*	*	*
	19	863.95	*	*	*
	20	864.00	K	*	K
	21	864.05	*	K	*
	22	864.10	*	*	*
	23	864.15	K	*	K
	24	864.20	*	*	*
	25	864.25	*	*	*
	26	864.30	K	*	K
	27	864.35	*	K	*
	28	864.40	*	*	*
	29	864.45	K	*	K
	30	864.50	*	*	*
	31	864.55	*	*	*
	32	864.60	K	*	K
	33	864.65	*	K	*
	34	864.70	*	*	*
	35	864.75	K	*	K
	36	864.80	*	*	*
	37	864.85	*	*	*
	38	864.90	K	-	K
	39	864.95	-	-	-
	40	865.00	Band limit		

Band L 865 MHz - 868 MHz $\leq$ 14 dBm -4.5 dBm/100 kHz The power density can be increased to +6.2 dBm/100 kHz if the band of operation is limited to 865 MHz to 868 MHz The whole band except for audio & video applications limited to 300 kHz duty cycle $\leq$ 1% or PSA Other usage restrictions: DSSS and any techniques other than FHSS.	41	865.05	-	-	-
	42	865.10	L	-	L
	43	865.15	*	L	*
	44	865.20	*	*	*
	45	865.25	L	*	L
	46	865.30	*	*	*
	47	865.35	*	*	*
	48	865.40	L	*	L
	49	865.45	*	L	*
	50	865.50	*	*	*
	51	865.55	L	*	L
	52	865.60	*	*	*
	53	865.65	*	*	*
	54	865.70	L	*	L
	55	865.75	*	L	*
	56	865.80	*	*	*
	57	865.85	L	*	L
	58	865.90	*	*	*
	59	865.95	*	*	*
	60	866.00	L	*	L
	61	866.05	*	L	*
	62	866.10	*	*	*
	63	866.15	L	*	L
	64	866.20	*	*	*
	65	866.25	*	*	*
	66	866.30	L	*	L
	67	866.35	*	L	*
	68	866.40	*	*	*
	69	866.45	L	*	L
	70	866.50	*	*	*
	71	866.55	*	*	*
	72	866.60	L	*	L
	73	866.65	*	L	*
	74	866.70	*	*	*
	75	866.75	L	*	L
	76	866.80	*	*	*
	77	866.85	*	*	*
	78	866.90	L	*	L
	79	866.95	*	L	*
	80	867.00	*	*	*
	81	867.05	L	*	L
	82	867.10	*	*	*
	83	867.15	*	*	*
	84	867.20	L	*	L
	85	867.25	*	L	*
	86	867.30	*	*	*
	87	867.35	L	*	L

		88	867.40	*	*	*
		89	867.45	*	*	*
		90	867.50	L	*	L
		91	867.55	*	L	*
		92	867.60	*	*	*
		93	867.65	L	*	L
		94	867.70	*	*	*
		95	867.75	*	*	*
		96	867.80	L	*	L
		97	867.85	*	L	*
		98	867.90	*	-	*
		99	867.95	-	-	-
Band M 868 MHz - 868.6 MHz $\leq$ 14 dBm The whole band except for audio& video applications limited to 300 kHz duty cycle $\leq$ 1% or PSA		100	868.00	Band limit		
		101	868.05	-	-	-
		102	868.10	M	-	M
		103	868.15	*	M	*
		104	868.20	*	*	*
		105	868.25	M	*	M
		106	868.30	*	*	*
		107	868.35	*	*	*
		108	868.40	M	*	M
		109	868.45	*	M	*
		110	868.50	*	-	*
		111	868.55	-	-	-
		112	868.60	Band limit		
		113	868.65	Out of band		
Band N 868.7 MHz - 869.2 MHz $\leq$ 14 dBm The whole band except for audio & video applications limited to 300 kHz duty cycle $\leq$ 0.1% or PSA		114	868.70	Band limit		
		115	868.75	-	-	-
		116	868.80	N	-	N
		117	868.85	*	N	*
		118	868.90	*	*	*
		119	868.95	N	*	N
		120	869.00	*	N	*
		121	869.05	*	*	*
		122	869.10	N	-	N
		123	869.15	-	-	-
		124	869.20	Band limit		
		125	869.25	Out of band		
		126	869.30			
		127	869.35			
Band O 869.4 MHz - 869.65 MHz $\leq$ 14 dBm duty cycle $\leq$ 0.1% or PSA	Band P $\leq$ 27 dBm duty cycle $\leq$ 10% or PSA	128	869.40	Band limit		
		129	869.45	-	-	-
		130	869.50	P	-	P
		131	869.55	*	-	*
		132	869.60	-	-	-
		133	869.65	Band limit		
Band Q 869.7 MHz - 870 MHz	Band R	134	869.70	Band limit		
		135	869.75	-	-	-

≤ 7 dBm No duty cycle	≤ 14 dBm duty cycle ≤ 1% or PSA	136	869.80	Q/R	-	Q/R
		137	869.85	*	Q/R	*
		138	869.90	*	-	*
		139	869.95	-	-	-
		140	870.00	Band limit		

Table 121: Channel assignment

The EN 300 220 allows a specific tx duration when the standard complying PSA method is implemented in the module firmware. The standard Daphnis-I firmware does not implement PSA. With PSA a transmission time of 100 s/h (3.6%) is possible.

K, L, M, N, P, Q, R means that the channel is allowed corresponding to the appropriate EN 300 220 operational frequency band requirements.

- means, that the channel is not allowed.

***** means, that the channel in general is allowed, but the above mentioned channel spacing must be fulfilled. In general allowed means, that the occupied channel fits into the appropriate frequency band and meets the requirement of out of band emissions and unwanted emissions in the spurious domain.

10 Power consumption optimization

This sections specifies the recommended configurations to optimize the power consumption of the radio module in the respective `MODE`.

10.1 LoRaWAN® mode optimization

- `AT+ADR`: This command should be enabled as it will allow the LoRaWAN® network server to determine the best data rate to be used by the module depending on its distance from the LoRaWAN® gateway.
- `AT+TXP`: The lowest TX output power should be chosen depending on the furthest distance between the module and the LoRaWAN® gateway to reduce transmission current consumption.
- `AT+CLASS`: The selected LoRaWAN® class should be `Class A` as this class will cause the module to only open the receive windows after a transmission occurs.

10.2 P2P mode optimization

- `AT+P2PRX`: This command should be disabled when a module is not expected to receive packets.
- `P2PRFPROFILE`: The fastest profile should to be chosen depending on the furthest distance between the modules to minimize the time on air durations when transmitting the packet.
- `P2PLBT`: This setting should be turned off as turning it on will internally cause the module to start receiving before a transmission.
- `P2PTXPOWER`: The lowest TX output power should be chosen depending on the furthest distance between the modules to reduce transmission current consumption.
- `P2PMACACK`: This setting should be turned off as turning it on will internally cause the module to start receiving after a transmission waiting for an acknowledgment or a timeout occurs.
- `P2PRPTHRESH`: This setting should be set to a threshold value depending on the network configuration which will cause packets received from close by modules to be discarded by the repeater.
- `P2PCMDRESPTIMEOUT`: This setting should be disabled and the reception of the radio command responses should be handled manually by the command `AT+P2PRX`.

11 Custom firmware

11.1 Custom configuration of standard firmware

The configuration of the standard firmware includes adoption of the non-volatile settings to customer requirements and creating a customized product based on the standard product.
This variant will result in a customer exclusive module with a unique ordering number. It will also freeze the firmware version to a specific and customer tested version and thus results in a customer exclusive module with a unique ordering number.
Further scheduled firmware updates of the standard firmware will not be applied to this variant automatically. Applying updates or further functions require a customer request and release procedure.

11.2 Customer specific firmware

A customer specific firmware may include "Custom configuration of standard firmware" plus additional options or functions and tasks that are customer specific and not part of the standard firmware.
Further scheduled firmware updates of the standard firmware will not be applied to this variant automatically. Applying updates or further functions require a customer request and release procedure.
This also results in a customer exclusive module with a unique ordering number.
An example for this level of customization are functions like host-less operation where the module will perform data generation (e.g. by reading a SPI or I^2C sensor) and cyclic transmission of this data to a data collector, while sleeping or being passive most of the time.
Also replacing UART with SPI as host communication interface is classified as a custom specific option.
Certification critical changes need to be re-evaluated by an external qualified measurement laboratory. These critical changes may occur when e.g. changing radio parameters, the channel access method, the duty-cycle or in case of various other functions and options possibly used or changed by a customer specific firmware.

11.3 Customer firmware

A customer firmware is a firmware written and tested by the customer himself or a 3rd party as a customer representative specifically for the hardware platform provided by a module.
This customer firmware (e.g. in form of an Intel hex file) will be implemented into the module's production process at our production site.
This also results in a customer exclusive module with a unique ordering number.
The additional information needed for this type of customer firmware, such as hardware specific details and details towards the development of such firmware are not available for the public and can only be made available to qualified customers.

The qualification(s) and certification(s) of the standard module cannot be applied to this customer firmware solution without a review and verification.

11.4 Contact for firmware requests

Please contact your Business Development Manager (BDM) or *WCS@we-online.com* for quotes regarding these topics.

12 Firmware update

The firmware on the Daphnis-I consists of two components:

- The bootloader that implements the secure firmware update over UART.
- The application firmware that provides the module functionality.

Only the application firmware can be updated over the UART interface (see section 12.1)

12.1 Firmware update using the Daphnis-I UART bootloader

The firmware update of the module over the UART interface is managed by the secure bootloader implemented as a part of the standard firmware. The update process can only be performed in the bootloader mode. If the firmware update process is interrupted, the application firmware will be invalidated and the module will always go to bootloader mode, regardless of the *BOOT* pin level. Once a successful update occurs, the module will boot depending on the level of the *BOOT* pin. A connection of *UTXD1* and *URXD1* over a suiting level and/or interface converter (e.g. FTDI cable or converter) is required. Refer to chapter Minimal pin connections for the pins needed for updating through the Daphnis-I bootloader.

The firmware update process requires exclusive access on the following pins of the Daphnis-I: *UTXD1*, *URXD1*.
The user is required to use the *BOOT* pin and */RESET* pin during this process.

A stable power supply is assumed to be available during any firmware update process. Battery operated firmware updates are not recommended.

- The module can be switched to the bootloader mode by holding the *BOOT* pin HIGH during the boot-up process to activate the secure UART bootloader.
- The *LED_1* pin will toggle to indicate that the module is in the bootloader mode and waiting for the firmware file to be sent.
- The secure UART bootloader runs at a fixed baud rate of 115200 Baud.

12.1.1 Update using Daphnis-I updater tool for Windows PC

The updater tool as well as the encrypted firmware update package (sfb) file is only available on request by contacting support [7].

Requirements for running the PC tool:

- Windows 10
- .NET 6.0 or newer (download available via Microsoft Homepage and in the Windows Features section)
- A folder with write and read access for the current user.

Steps to update the firmware using the tool:

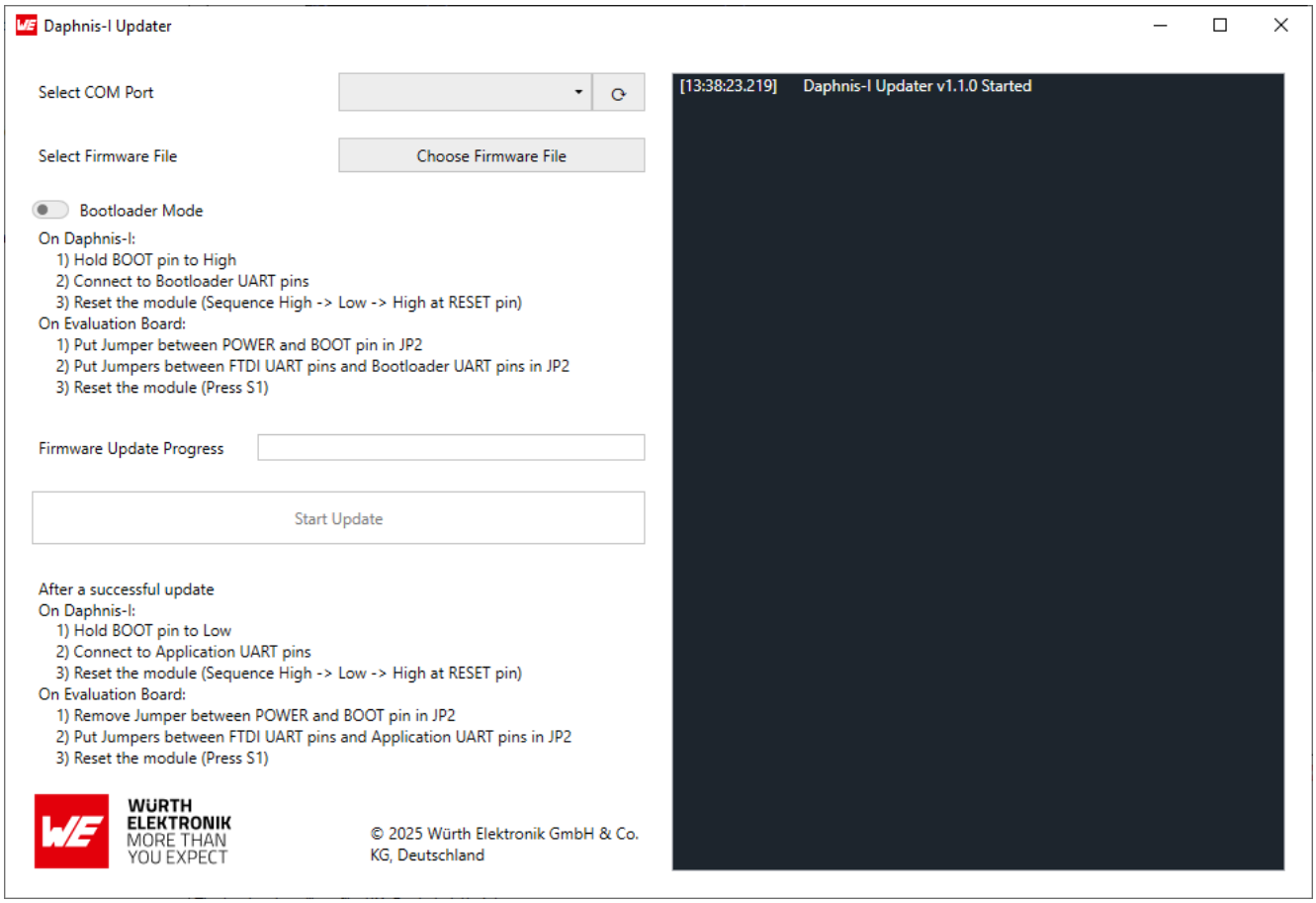

Figure 41: Daphnis-I Updater

1. Unzip the contents of the updater tool zip to any folder on your hard drive with read and write access. Network paths are not allowed.
2. Execute the GUI tool named Daphnis-I Updater.exe.
3. Connect the module to the PC and select the COM port of Daphnis-I.
4. Choose the encrypted and signed firmware update image package (sfb) file.
5. Put the module in bootloader mode this can be done by holding the *BOOT* high, connect to the bootloader UART pins (*UTXD1* and *URXD1*) and reset the module.
6. The start update button should now be enabled. Click it to start the update.

7. The log on the right should show a green success message, if the update was successful or a red failure message if the update failed.

8. If the update succeeds hold the *BOOT* low, connect to the application UART pins (*LPUTXD1* and *LPURXD1*) and reset the module.

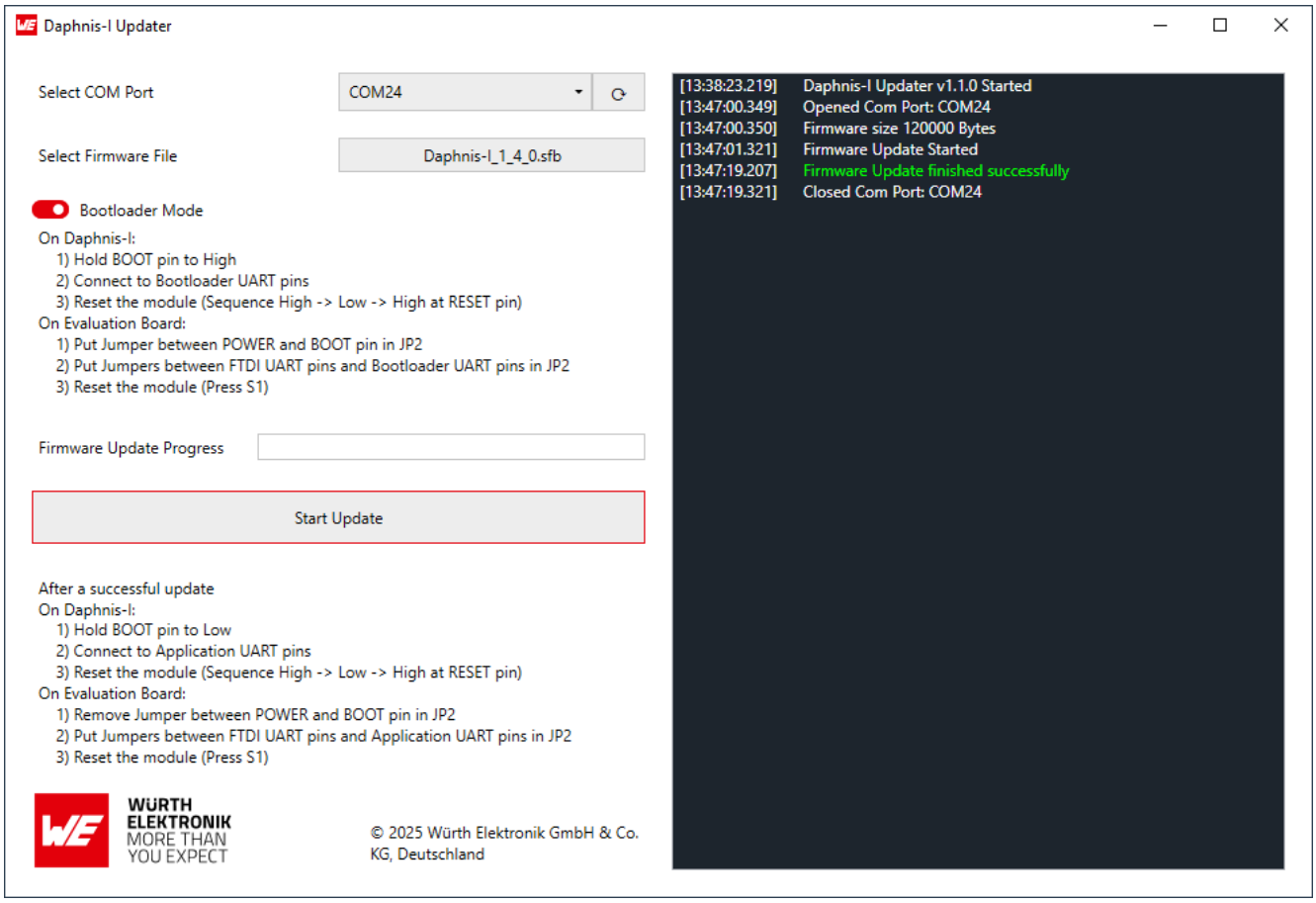

Figure 42: Daphnis-I Updater success

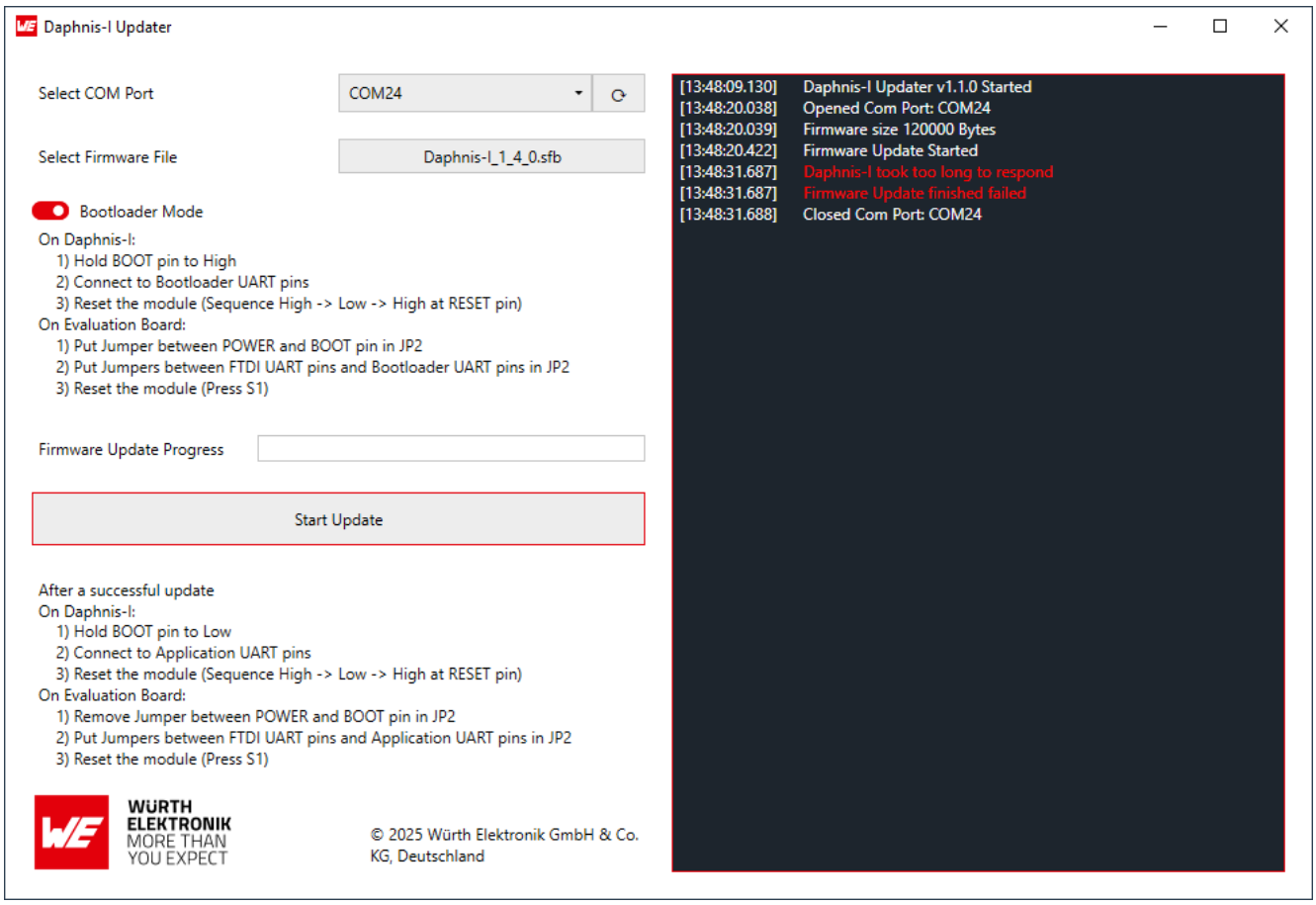

Figure 43: Daphnis-I Updater failure

The update will take some time. Make sure the command window stays top-most and is in focus during the entire update procedure. Watch carefully for any message displayed in the log window of the tool and keep the log in case of errors or if you require support.

The firmware update image used for this update process needs to be signed by Würth Elektronik eiSos, which means that the bootloader can't be used to install custom firmwares.

A firmware update will revert to the default factory settings of Daphnis-I.

The process and tools described here were tested and verified on a 64 bit Windows 10 (build 19044.2965) machine using a Daphnis-I EV-Board.

12.1.2 Update via host

In case you want to enable your own host to perform UART firmware updates, the bootloader of Daphnis-I implements the YMODEM protocol. This protocol must be implemented in the host. Refer to the chapter `Bootloader UART` for more information regarding the UART interface. The firmware update image of Daphnis-I itself may take up to 190 KB of memory in the host.

13 Firmware history

The Daphnis-I firmware is based on the STM32CubeWL software development kit (SDK) from STMicroelectronics with the corresponding features. A list of the versions of different components used for the current Daphnis-I firmware version is shown below.

Description	Version
STM32CubeWL	1.3.0
LoRaMac-node	4.6.0
LoRaWAN®	1.0.4
Regional Parameters	2-1.0.1

13.1 Release notes

Version 1.3.0 "Release"

- First release of the product.
- LoRaWAN® version: 1.0.4
- Regional parameters version: 2-1.0.1

Version 1.4.0 "Release"

- Added GPIO control capabilities.
- Added peer to peer (P2P) mode with extended AT commands and User/Runtime Settings. Refer to chapters `P2P data commands` and `P2P User/Runtime settings` respectively.

13.2 Known issues

Index	Details	Affected versions
KI-001	**Description:** The LoRaWAN® network isn't automatically joined after the LoRaWAN® context is restored. **Affected functions**: LoRaWAN® mode. **Workaround:** A manual join request using the command `AT+JOIN` after the LoRaWAN® context is restored is needed.	$\leq$ 1.4.0

14 Hardware history

Version 2.0 "Release"

- Initial release.

15 Design in guide

15.1 Advice for schematic and layout

For users with less RF experience it is advisable to closely copy the relating EV-Board with respect to schematic and layout, as it is a proven design. The layout should be conducted with particular care, because even small deficiencies could affect the radio performance and its range or even the conformity.
The following general advice should be taken into consideration:

- A clean, stable power supply is strongly recommended. Interference, especially oscillation can severely restrain range and conformity.
- Variations in voltage level should be avoided.
- LDOs, properly designed in, usually deliver a proper regulated voltage.
- Blocking capacitors and a ferrite bead in the power supply line can be included to filter and smoothen the supply voltage when necessary.

No fixed values can be recommended, as these depend on the circumstances of the application (main power source, interferences etc.).

The use of an external reset IC should be considered if one of the following points is relevant:

- The slew rate of the power supply exceeds the electrical specifications.
- The effect of different current consumptions on the voltage level of batteries or voltage regulators should be considered. The module draws higher currents in certain scenarios like start-up or radio transmit which may lead to a voltage drop on the supply. A restart under such circumstances should be prevented by ensuring that the supply voltage does not drop below the minimum specifications.
- Voltage levels below the minimum recommended voltage level may lead to malfunction. The reset pin of the module shall be held on LOW logic level whenever the VDD is not stable or below the minimum operating Voltage.
- Special care must be taken in case of battery powered systems.

- Elements for ESD protection should be placed on all pins that are accessible from the outside and should be placed close to the accessible area. For example, the RF-pin is accessible when using an external antenna and should be protected.
- ESD protection for the antenna connection must be chosen such as to have a minimum effect on the RF signal. For example, a protection diode with low capacitance such as the 8231606A or a 68 nH air-core coil connecting the RF-line to ground give good results.
- Placeholders for optional antenna matching or additional filtering are recommended.
- The antenna path should be kept as short as possible.

Again, no fixed values can be recommended, as they depend on the influencing circumstances of the application (antenna, interferences etc.).

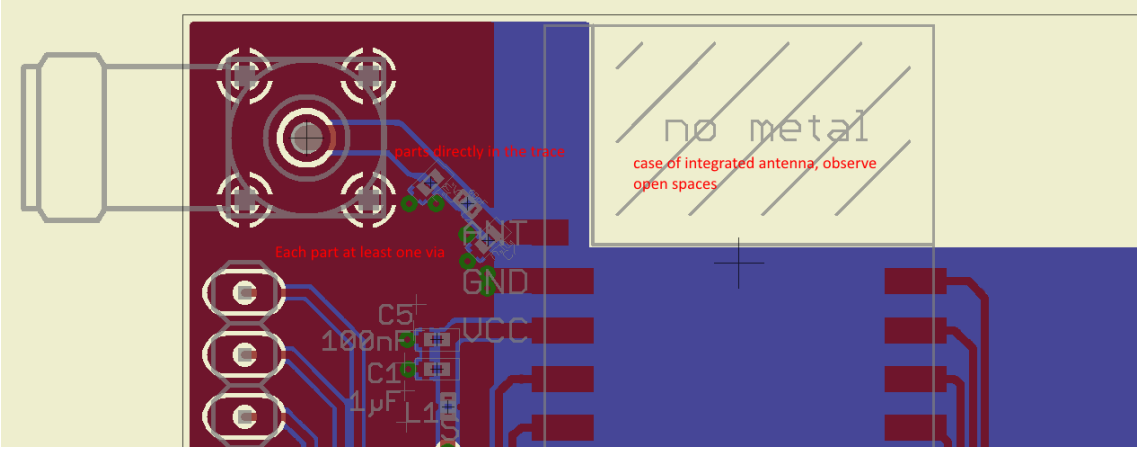

Figure 44: Layout

- To avoid the risk of short circuits and interference there should be no routing underneath the module on the top layer of the baseboard.
- On the second layer, a ground plane is recommended, to provide good grounding and shielding to any following layers and application environment.
- In case of integrated antennas it is required to have areas free from ground. This area should be copied from the EV-Board.
- The area with the integrated antenna must overlap with the carrier board and should not protrude, as it is matched to sitting directly on top of a PCB.
- Modules with integrated antennas should be placed with the antenna at the edge of the main board. It should not be placed in the middle of the main board or far away from the edge. This is to avoid tracks beside the antenna.

- Filter and blocking capacitors should be placed directly in the tracks without stubs, to achieve the best effect.
- Antenna matching elements should be placed close to the antenna / connector, blocking capacitors close to the module.
- Ground connections for the module and the capacitors should be kept as short as possible and with at least one separate through hole connection to the ground layer.
- ESD protection elements should be placed as close as possible to the exposed areas.

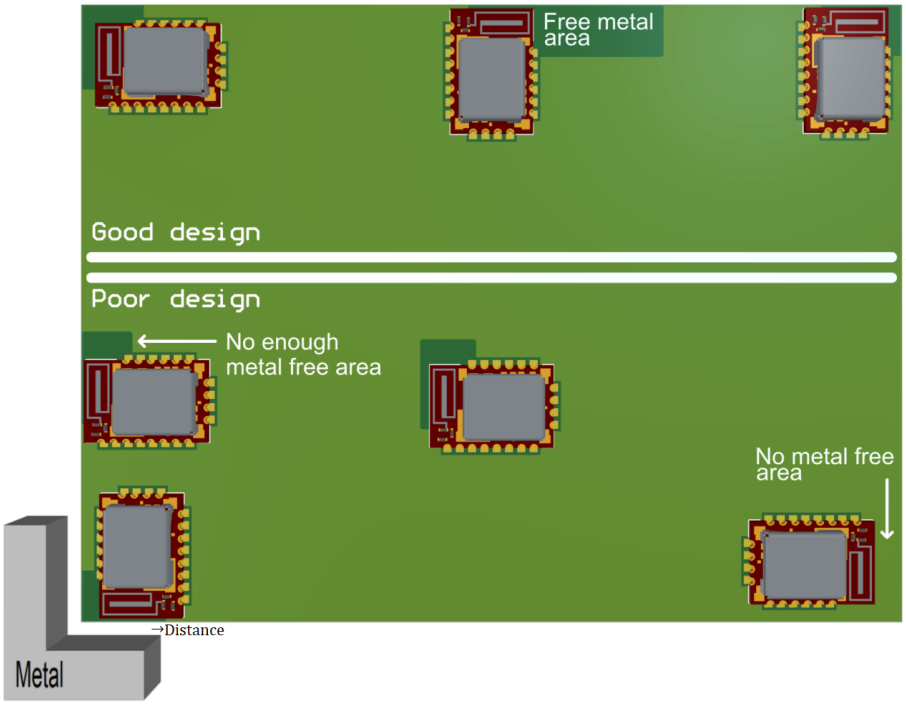

Figure 45: Placement of the module with integrated antenna

15.2 Designing the antenna connection

The antenna should be connected with a 50 Ω line. This is needed to obtain impedance matching to the module and avoids reflections. Here we show as an example how to calculate the dimensions of a 50 Ω line in form of a micro strip above ground, as this is easiest to calculate. Other connections like coplanar or strip line are more complicated to calculate but can offer more robustness to EMC. There are free calculation tools available in the internet.

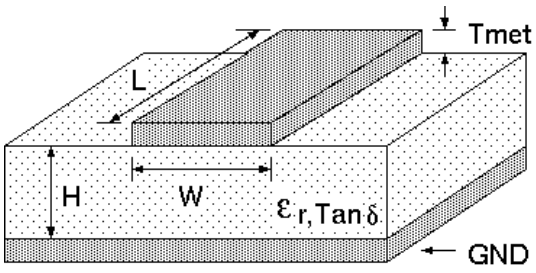

Figure 46: Dimensioning the antenna connection as micro strip

The width W for a micro strip can be calculated using the following equation:

$$W = 1.25 \times \left(\frac{5.98 \times H}{e^{\frac{50 \times \sqrt{\epsilon_r} + 1.41}{87}}} - T_{met} \right) \quad (4)$$

Example:
A FR4 material with ε_r = 4.3, a height H = 1000 µm and a copper thickness of T_{met} = 18 µm will lead to a trace width of W ~ 1.9 mm. To ease the calculation of the micro strip line (or e.g. a coplanar) many calculators can be found in the internet.

- As rule of thumb a distance of about 3×W should be observed between the micro strip and other traces / ground.
- The micro strip refers to ground, therefore there has to be the ground plane underneath the trace.
- Keep the feeding line as short as possible.

15.3 Antenna solutions

There exist several kinds of antennas, which are optimized for different needs. Chip antennas are optimized for minimal size requirements but at the expense of range, PCB antennas are optimized for minimal costs, and are generally a compromise between size and range. Both usually fit inside a housing.
Range optimization in general is at the expense of space. Antennas that are bigger in size, so that they would probably not fit in a small housing, are usually equipped with a RF connector. A benefit of this connector may be to use it to lead the RF signal through a metal plate (e.g. metal housing, cabinet).
As a rule of thumb a minimum distance of λ /10 (which is 3.5 cm @ 868 MHz and 1.2 cm @ 2.44 GHz) from the antenna to any other metal should be kept. Metal placed further away will not directly influence the behavior of the antenna, but will anyway produce shadowing.

Keep the antenna as far as possible from large metal objects to avoid electromagnetic field blocking.

In the following chapters, some special types of antenna are described.

15.3.1 Wire antenna

An effective antenna is a λ /4 radiator with a suiting ground plane. The simplest realization is a piece of wire. It's length is depending on the used radio frequency, so for example 8.6 cm 868.0 MHz and 3.1 cm for 2.440 GHz as frequency. This radiator needs a ground plane at its feeding point. Ideally, it is placed vertically in the middle of the ground plane. As this is often not possible because of space requirements, a suitable compromise is to bend the wire away from the PCB respective to the ground plane. The λ/4 radiator has approximately 40 Ω input impedance. Therefore, matching is not required.

15.3.2 Chip antenna

There are many chip antennas from various manufacturers. The benefit of a chip antenna is obviously the minimal space required and reasonable costs. However, this is often at the expense of range. For the chip antennas, reference designs should be followed as closely as possible, because only in this constellation can the stated performance be achieved.

15.3.3 PCB antenna

PCB antenna designs can be very different. The special attention can be on the miniaturization or on the performance. The benefits of the PCB antenna are their small / not existing (if PCB space is available) costs, however the EV of a PCB antenna holds more risk of failure than the use of a finished antenna. Most PCB antenna designs are a compromise of range and space between chip antennas and connector antennas.

15.3.4 Antennas provided by Würth Elektronik eiSos

Besides the radio modules Würth Elektronik eiSos provides various antennas tailored for the different frequency bands. The recommended single external antennas are shown in the subsequent chapters.

In case integrated multilayer chip antennas are needed because of space limitations, please refer to
https://www.we-online.com/en/components/products/WE-MCA.

15.3.4.1 2600130086 - Hermippe-III dipole antenna

Well suited for applications where the RF is lead through a metal wall that could serve as ground plane to the antenna.

Figure 47: Hermippe-III dipole antenna

Specification	Value
Frequency range [MHz]	855 – 915
VSWR (free space,without ground plane)	$\leq$ 2.0
Polarisation	Linear
Impedance [Ω]	50±5
Connector	SMA (Male)
Dimensions (L x d) [mm]	50±3 x 7.92±0.2
Weight [g]	4.5
Operating temp. [°C]	-40 – +85

15.3.4.2 2600130081 - Hyperion-I dipole antenna

Figure 48: Hyperion-I dipole antenna

Ideally suited for applications where no ground plane is available.

Specification	Value
Center frequency [MHz]	868
Frequency range [MHz]	853 – 883
Wavelength	λ /2
VSWR	≤ 2.0
Impedance [Ω]	50
Connector	SMA (Male)
Dimensions (L x d) [mm]	142 x 10
Peak gain [dBi]	-2.3
Operating temp. [℃]	-30 – +80

15.3.4.3 2600130082 - Hyperion-II magnetic base antenna

Well suited for applications where the RF is lead through a metal wall that could serve as ground plane to the antenna.

Figure 49: Hyperion-II magnetic base antenna with 1.5 m antenna cable

The 2600130082 is an antenna in form of $\lambda/4$ and therefore needs a ground plane at the feeding point.

Specification	Value
Frequency range [MHz]	824 – 894
VSWR	$\leq$ 2.0
Polarisation	Vertical
Impedance [Ω]	50±5
Connector	SMA (Male)
Dimensions (L x d) [mm]	89.8 x 27
Weight [g]	50±5
Operating temp. [℃]	-30 – +60

16 Reference design

Daphnis-I was tested and certified on the corresponding Daphnis-I EV-Board. For the compliance with the EU directive 2014/53/EU Annex I, the EV-Board serves as reference design. This is no discrepancy due to the fact that the EV-Board itself does not fall within the scope of the EU directive 2014/53/EU Annex I as the module is tested on the EV-Board, which is also the recommended use.

Further information concerning the use of the EV-Board can be found in the manual of the Daphnis-I EV-Board.

16.1 EV-Board

16.1.1 Schematic

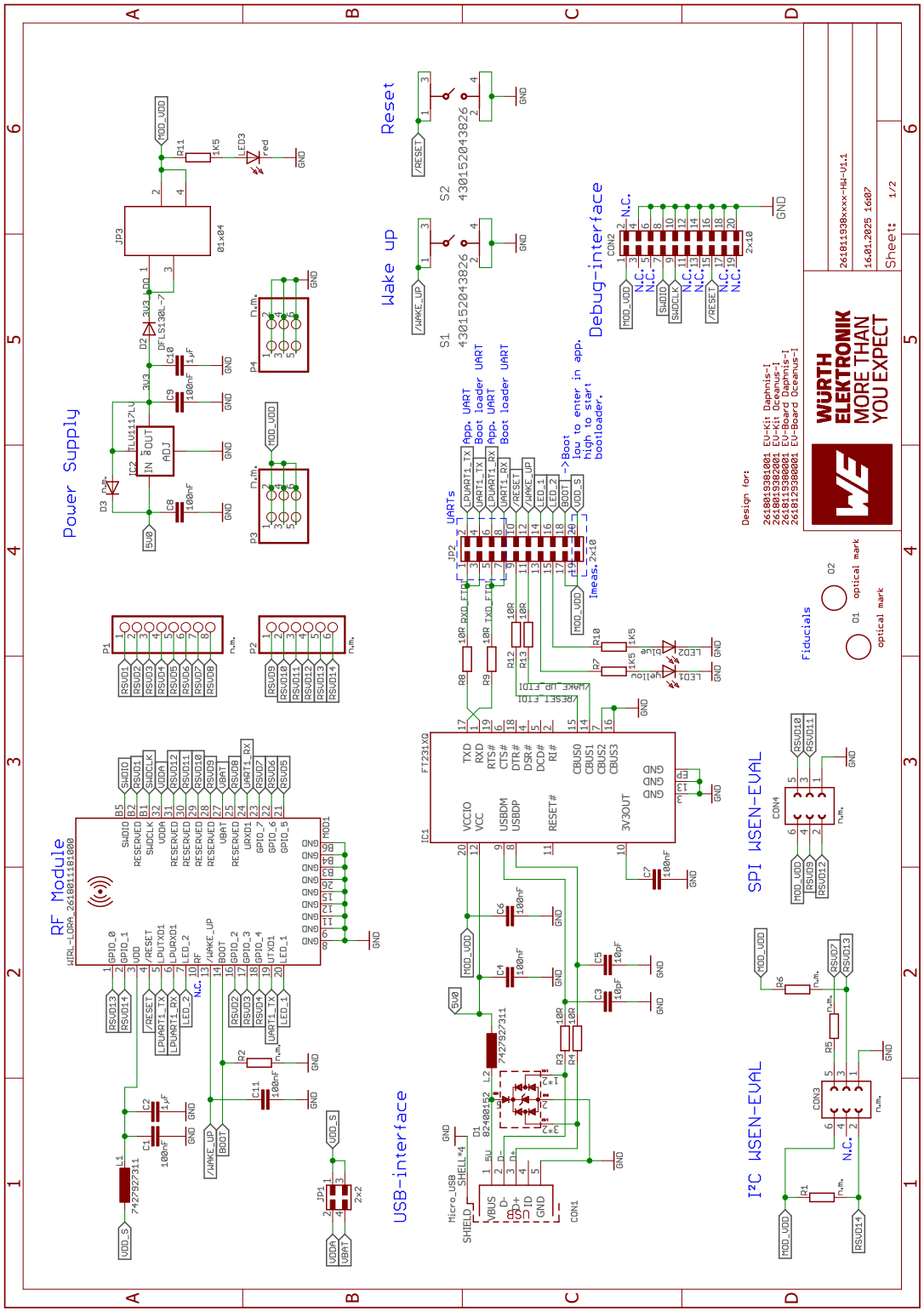

Figure 50: Reference design: Schematic diagram

16.1.2 Layout

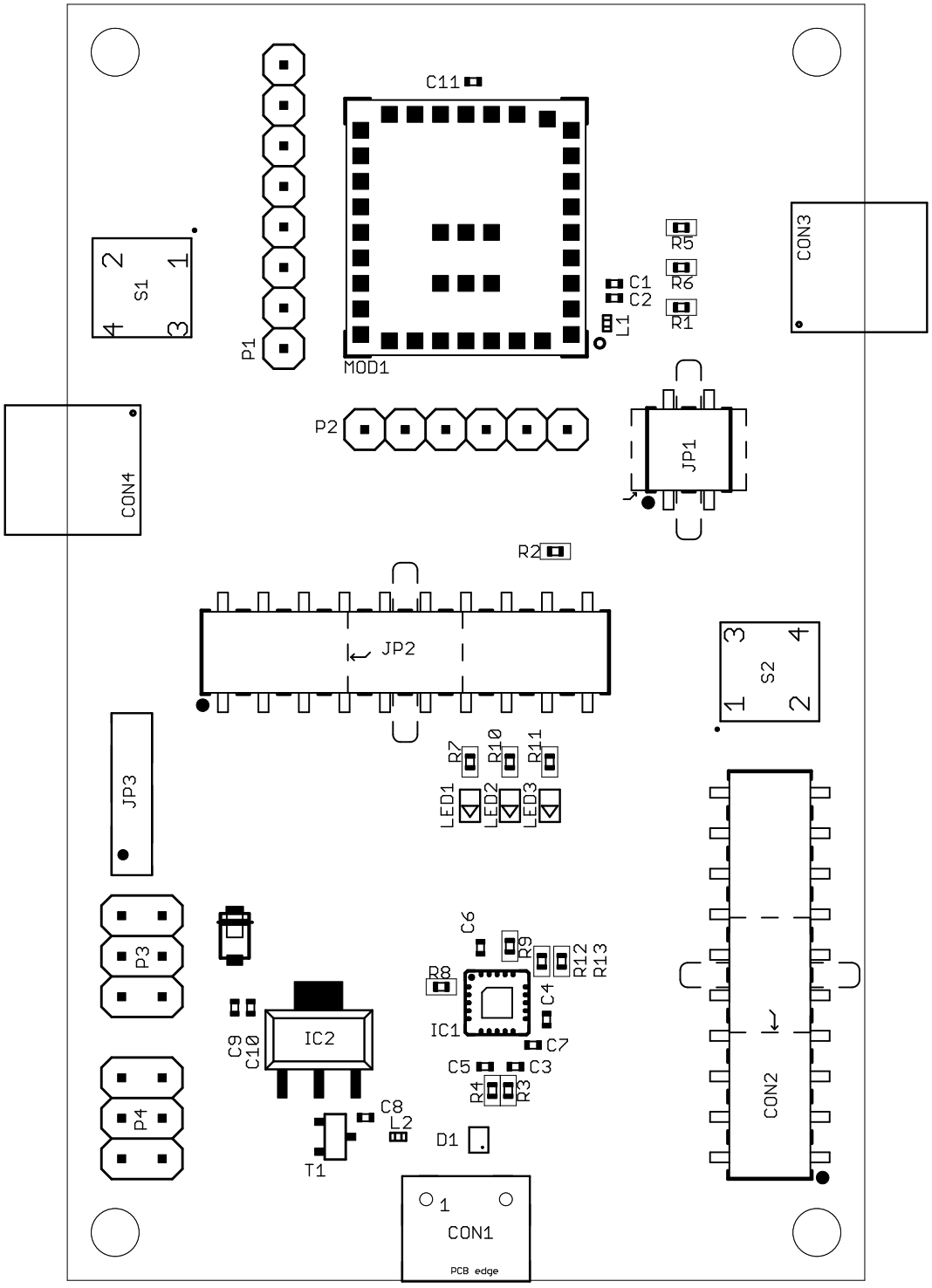

Figure 51: Reference design: Assembly diagram

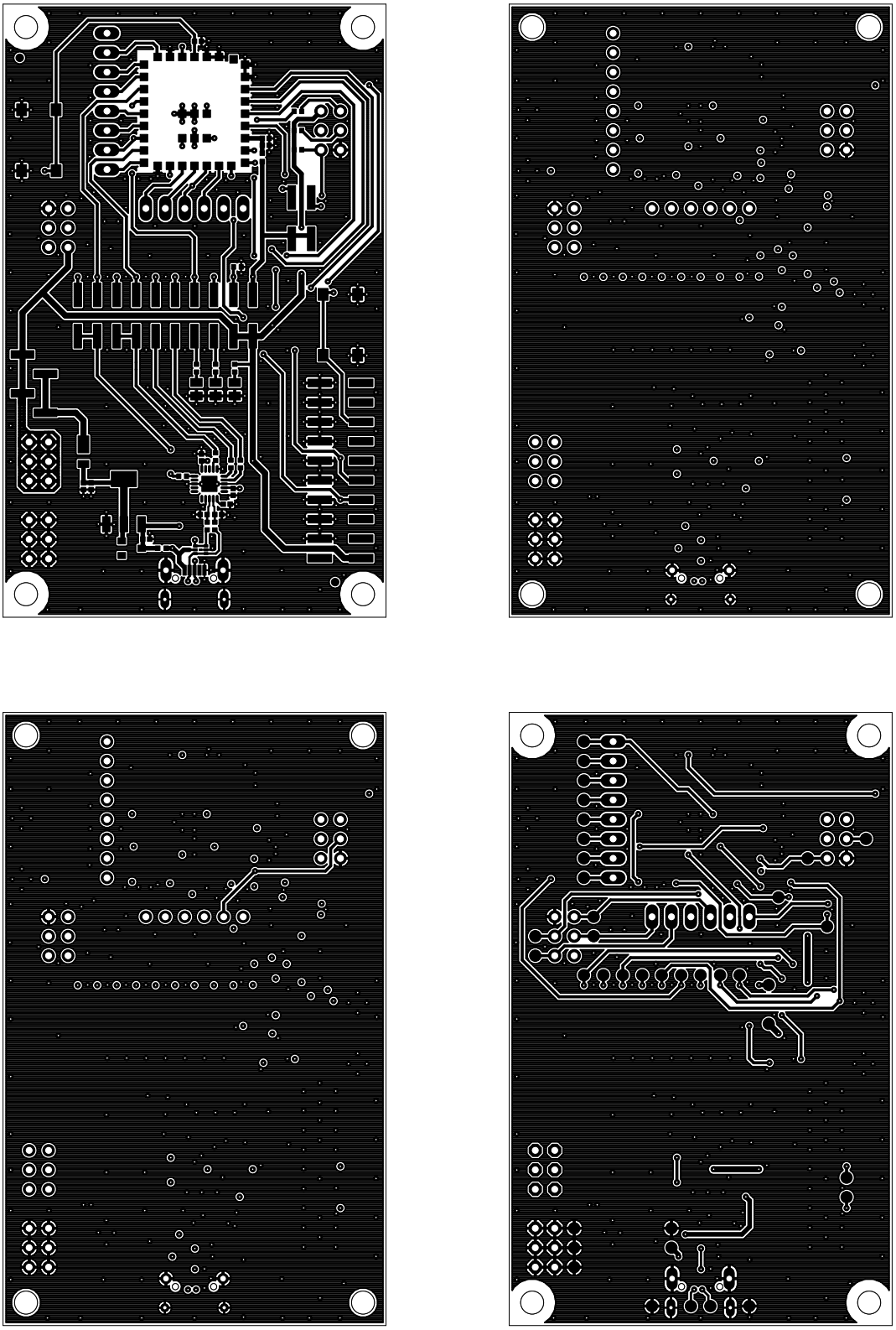

Figure 52: Top layer (upper left), second layer (upper right), third layer (bottom left), fourth layer (bottom right)

17 Manufacturing information

17.1 Moisture sensitivity level

This wireless connectivity product is categorized as JEDEC Moisture Sensitivity Level 3 (MSL3), which requires special handling.

More information regarding the MSL requirements can be found in the IPC/JEDEC J-STD-020 standard on *www.jedec.org*.

More information about the handling, picking, shipping and the usage of moisture/reflow and/or process sensitive products can be found in the IPC/JEDEC J-STD-033 standard on *www.jedec.org*.

17.2 Soldering

17.2.1 Reflow soldering

Attention must be paid on the thickness of the solder resist between the host PCB top side and the modules bottom side. Only lead-free assembly is recommended according to JEDEC J-STD020.

Profile feature		Value
Preheat temperature Min	$T_{S\ Min}$	150 °C
Preheat temperature Max	$T_{S\ Max}$	200 °C
Preheat time from $T_{S\ Min}$ to $T_{S\ Max}$	t_S	60 - 120 seconds
Ramp-up rate (T_L to T_P)		3 °C / second max.
Liquidous temperature	T_L	217 °C
Time t_L maintained above T_L	t_L	60 - 150 seconds
Peak package body temperature	T_P	260 °C
Time within 5 °C of actual peak temperature	t_P	20 - 30 seconds
Ramp-down Rate (T_P to T_L)		6 °C / second max.
Time 20 °C to T_P		8 minutes max.

Table 123: Classification reflow soldering profile, Note: refer to IPC/JEDEC J-STD-020E

It is recommended to solder this module on the last reflow cycle of the PCB. For solder paste use a LFM-48W or Indium based SAC 305 alloy (Sn 96.5 / Ag 3.0 / Cu 0.5 / Indium 8.9HF / Type 3 / 89%) type 3 or higher.

The reflow profile must be adjusted based on the thermal mass of the entire populated PCB, heat transfer efficiency of the reflow oven and the specific type of solder paste used. Based on the specific process and PCB layout the optimal soldering profile must be adjusted and verified. Other soldering methods (e.g. vapor phase) have not been verified and have to be validated

by the customer at their own risk. Rework is not recommended.

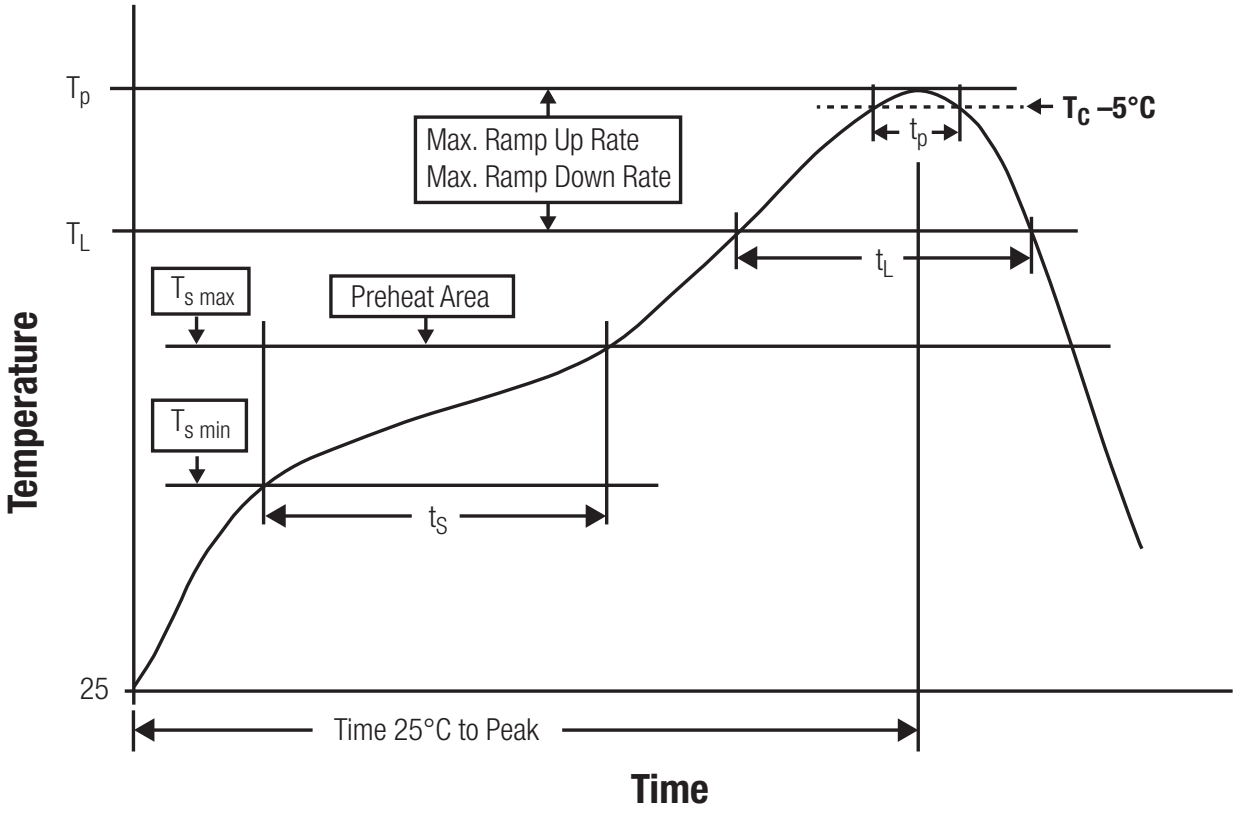

Figure 53: Reflow soldering profile

After reflow soldering, visually inspect the board to confirm proper alignment

17.2.2 Cleaning

Do not clean the product. Any residue cannot be easily removed by washing. Use a "no clean" soldering paste and do not clean the board after soldering.

- Do not clean the product with water. Capillary effects can draw water into the gap between the host PCB and the module, absorbing water underneath it. If water is trapped inside, it may short-circuit adjoining pads. The water may also destroy the label and ink-jet printed text on it.
- Cleaning processes using alcohol or other organic solvents may draw solder flux residues into the housing, which won't be detected in a post-wash inspection. The solvent may also destroy the label and ink-jet printed text on it.
- Do not use ultrasonic cleaning as it will permanently damage the part, particularly the crystal oscillators.

17.2.3 Potting and coating

- If the product is potted in the customer application, the potting material might shrink or expand during and after hardening. Shrinking could lead to an incomplete seal, allowing contaminants into the component. Expansion could damage components. We recommend a manual inspection after potting to avoid these effects.
- Conformal coating or potting results in loss of warranty.
- The RF shield will not protect the part from low-viscosity coatings and potting. An undefined amount of coating and potting will enter inside the shielding.
- Conformal coating and potting will influence the parts of the radio front end and consequently influence the radio performance.
- Potting will influence the temperature behaviour of the device. This might be critical for components with high power.

17.2.4 Other notations

- Do not attempt to improve the grounding by forming metal strips directly to the EMI covers or soldering on ground cables, as it may damage the part and will void the warranty.
- Always solder every pad to the host PCB even if some are unused, to improve the mechanical strength of the module.
- The part is sensitive to ultrasonic waves, as such do not use ultrasonic cleaning, welding or other processing. Any ultrasonic processing will void the warranty.

17.3 ESD handling

This product is highly sensitive to electrostatic discharge (ESD). As such, always use proper ESD precautions when handling. Make sure to handle the part properly throughout all stages of production, including on the host PCB where the module is installed. For ESD ratings, refer to the module series' maximum ESD section. For more information, refer to the relevant chapter 2. Failing to follow the aforementioned recommendations can result in severe damage to the part.

- the first contact point when handling the PCB is always between the local GND and the host PCB GND, unless there is a galvanic coupling between the local GND (for example work table) and the host PCB GND.
- Before assembling an antenna patch, connect the grounds.
- While handling the RF pin, avoid contact with any charged capacitors and be careful when contacting any materials that can develop charges (for example coaxial cable with around 50-80 pF/m, patch antenna with around 10 pF, soldering iron etc.)
- Do not touch any exposed area of the antenna to avoid electrostatic discharge. Do not let the antenna area be touched in a non ESD-safe manner.
- When soldering, use an ESD-safe soldering iron.

17.4 Safety recommendations

It is your duty to ensure that the product is allowed to be used in the destination country and within the required environment. Usage of the product can be dangerous and must be tested and verified by the end user. Be especially careful of:

- Use in areas with risk of explosion (for example oil refineries, gas stations).
- Use in areas such as airports, aircraft, hospitals, etc., where the product may interfere with other electronic components.

It is the customer's responsibility to ensure compliance with all applicable legal, regulatory and safety-related requirements as well as applicable environmental regulations. Disassembling the product is not allowed. Evidence of tampering will void the warranty.

- Compliance with the instructions in the product manual is recommended for correct product set-up.
- The product must be provided with a consolidated voltage source. The wiring must meet all applicable fire and security prevention standards.
- Handle with care. Avoid touching the pins as there could be ESD damage.

Be careful when working with any external components. When in doubt consult the technical documentation and relevant standards. Always use an antenna with the proper characteristics.

Würth Elektronik eiSos radio modules with high output power of up to 500 mW generate a large amount of heat while transmitting. The manufacturer of the end device must take care of potentially necessary actions for his application.

18 Product testing

18.1 Würth Elektronik eiSos in-house production tests

To achieve a high quality standard, Würth Elektronik eiSos follows a philosophy of supplying fully tested radio modules. At the end of the production process, every unit undergoes an optical inspection. Here the quality of soldering, edge castellation and edge milling is monitored.

If this has been passed, the radio modules are handed over to the automatic test equipment for the electrical characterization. This includes:

- Voltage and current tests to ensure proper electrical performance
- RF characteristics (frequency, spectrum, TX power) measurement and calibration
- Radio communication tests
- Firmware and serial number programming
- Host interface communication tests

The automated testing process is logged for internal quality control. The gained measurement data of each unit is analysed to detect defective parts and investigate the corresponding root cause. Defective radio modules are discarded, in order to guarantee a 100% failure-free delivery to customers.

18.2 EMS production tests

The rigorous in-series production testing ensures that EMS don't need to duplicate firmware tests or measurements. This streamlines the process and eliminates the need for additional testing over analogue and digital interfaces during device production. When it comes to device testing, the ideal focus should be on module assembly quality:

- All module pins are soldered properly on the base PCB
- There are no short circuits
- The mounting process did not damage the module
- The communication between host and radio module is working
- The antenna is connected properly

Simple "Go/No go" tests, like checking the RSSI value, give already a hint if the power supply and antenna have been connected properly.

In addition to such standard testing procedures, radio module integrators have the flexibility to perform additional dedicated tests to thoroughly evaluate the device. Specific tests they can consider are:

- Measure module current consumption in a specified operating state. Deviations from expected results (compared to a "Golden Device") can signal potential issues.

- Perform functional tests, including communication checks with the host controller and verification of interfaces.
- Assess fundamental RF characteristics (modulation accuracy, power levels, spectrum). Verify that the device meets expected performance standards.

19 Physical specifications

19.1 Dimensions

Dimensions
15.0 * 16.0 * 3.00 mm

Table 124: Dimensions

Tolerances: see chapter 19.3

19.2 Weight

Weight
1.21 g

Table 125: Weight

Tolerance: ± 0.15 g

19.3 Module drawing

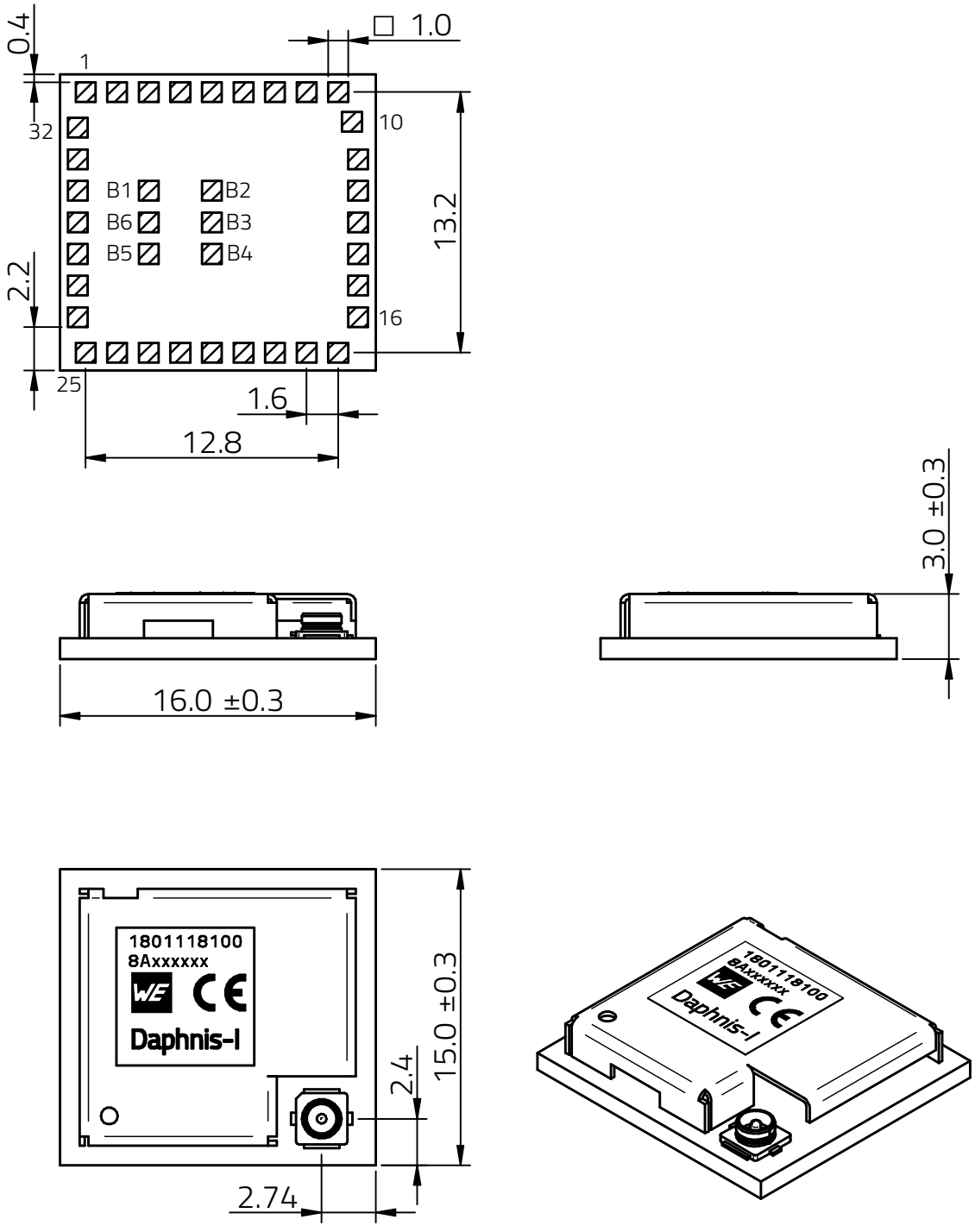

Figure 54: Module dimensions [mm]

19.4 Footprint

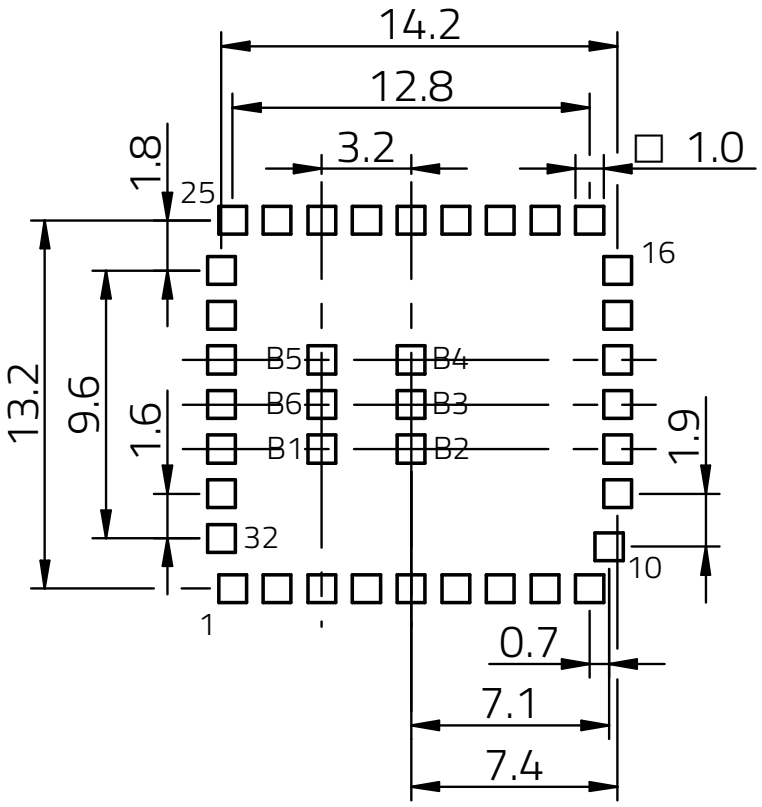

Figure 55: Footprint and dimensions [mm]

20 Marking

20.1 Lot number

The 15 digit lot number is printed in numerical digits as well as in form of a machine readable bar code. It is divided into 5 blocks as shown in the following picture and can be translated according to the following table.

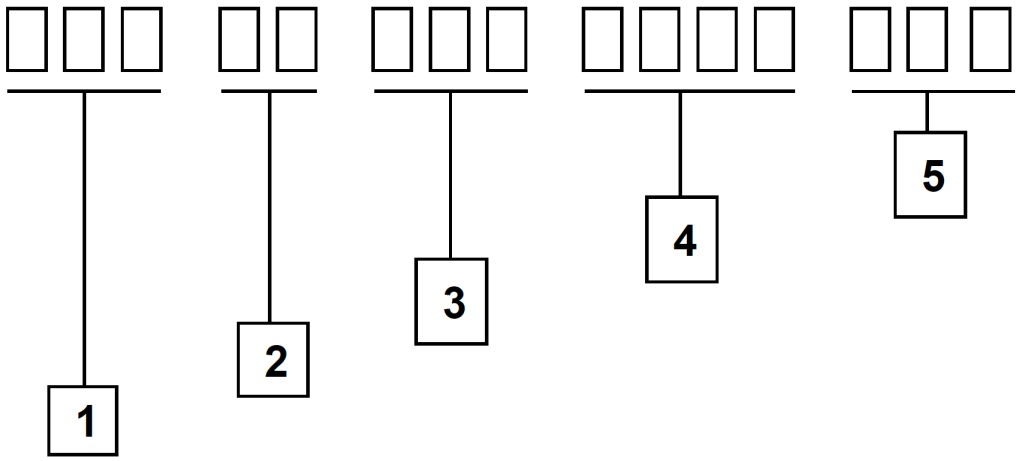

Figure 56: Lot number structure

Block	Information	Example(s)
1	eiSos internal, 3 digits	438
2	eiSos internal, 2 digits	01
3	Radio module hardware version, 3 digits	V2.4 = 024, V12.2 = 122
4	Date code, 4 digits	1703 = week 03 in year 2017, 1816 = week 16 in year 2018
5	Radio module firmware version, 3 digits	V3.2 = 302, V5.13 = 513

Table 126: Lot number details

As the user can perform a firmware update the printed lot number only shows the factory delivery state. The currently installed firmware can be requested from the module using the corresponding product specific command. The firmware version as well as the hardware version are restricted to show only major and minor version not the patch identifier.

20.2 General labeling information

Labels of Würth Elektronik eiSos radio modules include several fields. Besides the manufacturer identification, the product's *WE* order code, serial number and certification information are placed on the label. In case of small labels, additional certification marks are placed on the label of the reel.
The information on the label are fixed. Only the serial number changes with each entity of the radio module. For Daphnis-I the label is as follows:

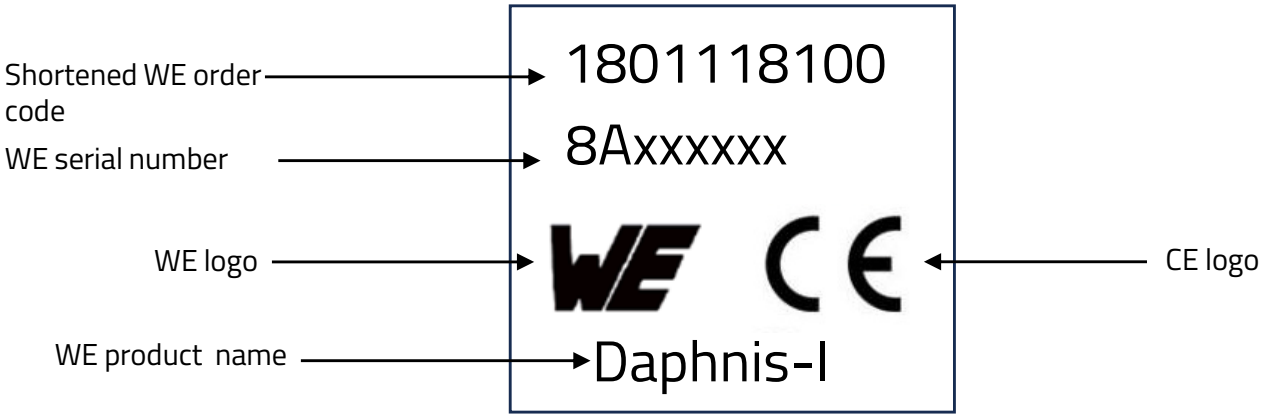

Figure 57: Label of the Daphnis-I

21 Information for explosion protection

In case the end product should be used in explosion protection areas, the following information can be used:

- The module itself is unfused.
- The maximum power of the module is 14 dBm for external antenna.
- The total amount of capacitance of all capacitors is 15.24 µF.
- The total amount of inductance of all inductors is 15.144 µH.

22 References

[1] Würth Elektronik. WE UART Terminal PC tool (Smart Commander). https://www.we-online.de/wcs-software.

[2] Würth Elektronik. DAPHNIS-I EVALUATION BOARD MANUAL. https://we-online.com/katalog/en/manual/2618119380001.

[3] STMicroelectronics. Web page: STM32WLE5CC. https://www.st.com/en/microcontrollers-microprocessors/stm32wle5cc.html.

[4] FTDI virtual COM port driver. https://ftdichip.com/drivers/vcp-drivers/.

[5] LoRa Alliance©. LoRaWAN® L2 1.0.4 Specification. https://resources.lora-alliance.org/technical-specifications/ts001-1-0-4-lorawan-l2-1-0-4-specification.

[6] LoRa Alliance©. LoRaWAN® RP002-1.0.1 LoRaWAN Regional Parameters. https://resources.lora-alliance.org/technical-specifications/rp2-1-0-1-lorawan-regional-parameters.

[7] Würth Elektronik Support. https://we-online.com/wireless-connectivity/support.

23 Regulatory compliance information

23.1 Important notice EU

The use of RF frequencies is limited by national regulations. The Daphnis-I has been designed to comply with the RED directive 2014/53/EU of the European Union (EU).
The Daphnis-I can be operated without notification and free of charge in the area of the European Union. However, according to the RED directive, restrictions (e.g. in terms of duty cycle or maximum allowed RF power) may apply.

Since the module is a built-in equipment its power class according to EN 62368-1 must be specified in the end product.

23.2 Important notice UKCA

The UK's government has laid legislation to continue recognition of current EU requirements for a range of product regulations, including the CE marking. The Radio Equipment Regulation 2017/1206 is within the scope of this announcement, among others.
Consequently, the Daphnis-I can be sold and utilized in the UK with the CE marking, without the need of UKCA declaration of conformity or UKCA marking.

Source: *https://www.gov.uk/guidance/ce-marking*

23.3 Conformity assessment of the final product

The Daphnis-I is a subassembly. It is designed to be embedded into other products (products incorporating the Daphnis-I are henceforward referred to as "final products").
It is the responsibility of the manufacturer of the final product to ensure that the final product is in compliance with the essential requirements of the underlying national radio regulations.
The conformity assessment of the subassembly Daphnis-I carried out by Würth Elektronik eiSos does not replace the required conformity assessment of the final product.

23.4 Exemption clause

Relevant regulation requirements are subject to change. Würth Elektronik eiSos does not guarantee the accuracy of the before mentioned information. Directives, technical standards, procedural descriptions and the like may be interpreted differently by the national authorities. Equally, the national laws and restrictions may vary with the country. In case of doubt or uncertainty, we recommend that you consult with the authorities or official certification organizations of the relevant countries. Würth Elektronik eiSos is exempt from any responsibilities or liabilities related to regulatory compliance.

Notwithstanding the above, Würth Elektronik eiSos makes no representations and warranties of any kind related to their accuracy, correctness, completeness and/or usability for customer applications. No responsibility is assumed for inaccuracies or incompleteness.

23.5 EU Declaration of conformity

CE

EU DECLARATION OF CONFORMITY

Radio equipment: **2618011181000**

The manufacturer: Würth Elektronik eiSos GmbH & Co. KG
Max-Eyth-Straße 1
74638 Waldenburg

This declaration of conformity is issued under the sole responsibility of the manufacturer.

Object of the declaration: 2618011181000

The object of the declaration described above is in conformity with the relevant Union harmonisation legislation Directive 2014/53/EU. Following harmonised norms or technical specifications have been applied:

EN 300 220-1 V3.1.1 (2017-02)
EN 300 220-2 V3.1.1 (2017-02)
EN 301 489-1 V2.2.3 (2019-11)
EN 301 489-3 V2.3.2 (2023-01)
EN 62368-1: 2014/AC: 2015/A11: 2017/AC:2017
EN 62311 : 2010
2011/65/EU with its amending Annex II EU 2015/863.

i.A. G. Erhardt

Trier, 7th of July 2023
Place and date of issue

24 Important notes

The following conditions apply to all goods within the wireless connectivity and sensors product range of Würth Elektronik eiSos GmbH & Co. KG:

General customer responsibility
Some goods within the product range of Würth Elektronik eiSos GmbH & Co. KG contain statements regarding general suitability for certain application areas. These statements about suitability are based on our knowledge and experience of typical requirements concerning the areas, serve as general guidance and cannot be estimated as binding statements about the suitability for a customer application. The responsibility for the applicability and use in a particular customer design is always solely within the authority of the customer. Due to this fact, it is up to the customer to evaluate, where appropriate to investigate and to decide whether the device with the specific product characteristics described in the product specification is valid and suitable for the respective customer application or not. Accordingly, the customer is cautioned to verify that the documentation is current before placing orders.

Customer responsibility related to specific, in particular safety-relevant applications
It has to be clearly pointed out that the possibility of a malfunction of electronic components or failure before the end of the usual lifetime cannot be completely eliminated in the current state of the art, even if the products are operated within the range of the specifications. The same statement is valid for all software source code and firmware parts contained in or used with or for products in the wireless connectivity and sensor product range of Würth Elektronik eiSos GmbH & Co. KG. In certain customer applications requiring a high level of safety and especially in customer applications in which the malfunction or failure of an electronic component could endanger human life or health, it must be ensured by most advanced technological aid of suitable design of the customer application that no injury or damage is caused to third parties in the event of malfunction or failure of an electronic component.

Best care and attention
Any product-specific data sheets, manuals, application notes, PCNs, warnings and cautions must be strictly observed in the most recent versions and matching to the products revisions. These documents can be downloaded from the product specific sections on the wireless connectivity and sensors homepage.

Customer support for product specifications
Some products within the product range may contain substances, which are subject to restrictions in certain jurisdictions in order to serve specific technical requirements. Necessary information is available on request. In this case, the Business Development Engineer (BDM) or the internal sales person in charge should be contacted who will be happy to support in this matter.

Product improvements
Due to constant product improvement, product specifications may change from time to time. As a standard reporting procedure of the Product Change Notification (PCN) according to the JEDEC-Standard, we inform about major changes. In case of further queries regarding the PCN, the Business Development Engineer (BDM), the internal sales person or the technical support team in charge should be contacted. The basic responsibility of the customer as per section 24 and 24 remains unaffected.
All software like ¨wireless connectivity SDK¨, ¨Sensor SDK¨ or other source codes as well as all PC software tools are not subject to the Product Change Notification information process.

Product life cycle
Due to technical progress and economical evaluation, we also reserve the right to discontinue production and delivery of products. As a standard reporting procedure of the Product Termination Notification (PTN) according to the JEDEC-Standard we will inform at an early stage about inevitable product discontinuance. According to this, we cannot ensure that all products within our product range will always be available. Therefore, it needs to be verified with the Business Development Engineer (BDM) or the internal sales person in charge about the current product availability expectancy before or when the product for application design-in disposal is considered. The approach named above does not apply in the case of individual agreements deviating from the foregoing for customer-specific products.

Property rights
All the rights for contractual products produced by Würth Elektronik eiSos GmbH & Co. KG on the basis of ideas, development contracts as well as models or templates that are subject to copyright, patent or commercial protection supplied to the customer will remain with Würth Elektronik eiSos GmbH & Co. KG. Würth Elektronik eiSos GmbH & Co. KG does not warrant or represent that any license, either expressed or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, application, or process in which Würth Elektronik eiSos GmbH & Co. KG components or services are used.

General terms and conditions
Unless otherwise agreed in individual contracts, all orders are subject to the current version of the "General Terms and Conditions of Würth Elektronik eiSos Group", last version available at *www.we-online.com*.

25 Legal notice

Exclusion of liability
Würth Elektronik eiSos GmbH & Co. KG considers the information in this document to be correct at the time of publication. However, Würth Elektronik eiSos GmbH & Co. KG reserves the right to modify the information such as technical specifications or functions of its products or discontinue the production of these products or the support of one of these products without any written announcement or notification to customers. The customer must make sure that the information used corresponds to the latest published information. Würth Elektronik eiSos GmbH & Co. KG does not assume any liability for the use of its products. Würth Elektronik eiSos GmbH & Co. KG does not grant licenses for its patent rights or for any other of its intellectual property rights or third-party rights.
Notwithstanding anything above, Würth Elektronik eiSos GmbH & Co. KG makes no representations and/or warranties of any kind for the

provided information related to their accuracy, correctness, completeness, usage of the products and/or usability for customer applications. Information published by Würth Elektronik eiSos GmbH & Co. KG regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof.

Suitability in customer applications
The customer bears the responsibility for compliance of systems or units, in which Würth Elektronik eiSos GmbH & Co. KG products are integrated, with applicable legal regulations. Customer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of Würth Elektronik eiSos GmbH & Co. KG components in its applications, notwithstanding any applications-related information or support that may be provided by Würth Elektronik eiSos GmbH & Co. KG. Customer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences lessen the likelihood of failures that might cause harm and take appropriate remedial actions. The customer will fully indemnify Würth Elektronik eiSos GmbH & Co. KG and its representatives against any damages arising out of the use of any Würth Elektronik eiSos GmbH & Co. KG components in safety-critical applications.

Trademarks
AMBER wireless is a registered trademark of Würth Elektronik eiSos GmbH & Co. KG. All other trademarks, registered trademarks, and product names are the exclusive property of the respective owners.

Usage restriction
Würth Elektronik eiSos GmbH & Co. KG products have been designed and developed for usage in general electronic equipment only. This product is not authorized for use in equipment where a higher safety standard and reliability standard is especially required or where a failure of the product is reasonably expected to cause severe personal injury or death, unless the parties have executed an agreement specifically governing such use. Moreover, Würth Elektronik eiSos GmbH & Co. KG products are neither designed nor intended for use in areas such as military, aerospace, aviation, nuclear control, submarine, transportation (automotive control, train control, ship control), transportation signal, disaster prevention, medical, public information network etc. Würth Elektronik eiSos GmbH & Co. KG must be informed about the intent of such usage before the design-in stage. In addition, sufficient reliability evaluation checks for safety must be performed on every electronic component, which is used in electrical circuits that require high safety and reliability function or performance. By using Würth Elektronik eiSos GmbH & Co. KG products, the customer agrees to these terms and conditions.

26 License terms

These License terms will take effect upon the purchase and usage of the Würth Elektronik eiSos GmbH & Co. KG wireless connectivity products. You hereby agree that these license terms are applicable to the product and the incorporated software, firmware and source codes (collectively, "Software") made available by Würth Elektronik eiSos in any form, including but not limited to binary, executable or source code form. The software included in any Würth Elektronik eiSos wireless connectivity product is purchased to you on the condition that you accept the terms and conditions of these license terms. You agree to comply with all provisions under these license terms.

Limited license
Würth Elektronik eiSos hereby grants you a limited, non-exclusive, non-transferable and royalty-free license to use the software and under the conditions that will be set forth in these license terms. You are free to use the provided software only in connection with one of the products from Würth Elektronik eiSos to the extent described in these license terms. You are entitled to change or alter the source code for the sole purpose of creating an application embedding the Würth Elektronik eiSos wireless connectivity product. The transfer of the source code to third parties is allowed to the sole extent that the source code is used by such third parties in connection with our product or another hardware provided by Würth Elektronik eiSos under strict adherence of these license terms. Würth Elektronik eiSos will not assume any liability for the usage of the incorporated software and the source code. You are not entitled to transfer the source code in any form to third parties without prior written consent of Würth Elektronik eiSos.
You are not allowed to reproduce, translate, reverse engineer, decompile, disassemble or create derivative works of the incorporated software and the source code in whole or in part. No more extensive rights to use and exploit the products are granted to you.

Usage and obligations
The responsibility for the applicability and use of the Würth Elektronik eiSos wireless connectivity product with the incorporated firmware in a particular customer design is always solely within the authority of the customer. Due to this fact, it is up to you to evaluate and investigate, where appropriate, and to decide whether the device with the specific product characteristics described in the product specification is valid and suitable for your respective application or not.
You are responsible for using the Würth Elektronik eiSos wireless connectivity product with the incorporated firmware in compliance with all applicable product liability and product safety laws. You acknowledge to minimize the risk of loss and harm to individuals and bear the risk for failure leading to personal injury or death due to your usage of the product.
Würth Elektronik eiSos' products with the incorporated firmware are not authorized for use in safety-critical applications, or where a failure of the product is reasonably expected to cause severe personal injury or death. Moreover, Würth Elektronik eiSos' products with the incorporated firmware are neither designed nor intended for use in areas such as military, aerospace, aviation, nuclear control, submarine, transportation (automotive control, train control, ship control), transportation signal, disaster prevention, medical, public information network etc. You shall inform Würth Elektronik eiSos about the intent of such usage before design-in stage. In certain customer applications requiring a very high level of safety and in which the malfunction or failure of an electronic component could endanger human life or health, you must ensure to have all necessary expertise in the safety and regulatory ramifications of your applications. You acknowledge and agree that you are solely responsible for all legal, regulatory and safety-related requirements concerning your products and any use of Würth Elektronik eiSos' products with the incorporated firmware in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by Würth Elektronik eiSos. YOU SHALL INDEMNIFY WÜRTH ELEKTRONIK EISOS AGAINST ANY DAMAGES ARISING OUT OF THE USE OF WÜRTH ELEKTRONIK EISOS' PRODUCTS WITH THE INCORPORATED FIRMWARE IN SUCH SAFETY-CRITICAL APPLICATIONS.

Ownership

The incorporated firmware created by Würth Elektronik eiSos is and will remain the exclusive property of Würth Elektronik eiSos.

Firmware update(s)
You have the opportunity to request the current and actual firmware for a bought wireless connectivity product within the time of warranty. However, Würth Elektronik eiSos has no obligation to update a modules firmware in their production facilities, but can offer this as a service on request. The upload of firmware updates falls within your responsibility, e.g. via ACC or another software for firmware updates. Firmware updates will not be communicated automatically. It is within your responsibility to check the current version of a firmware in the latest version of the product manual on our website. The revision table in the product manual provides all necessary information about firmware updates. There is no right to be provided with binary files, so called "firmware images", those could be flashed through JTAG, SWD, Spi-Bi-Wire, SPI or similar interfaces.

Disclaimer of warranty
THE FIRMWARE IS PROVIDED "AS IS". YOU ACKNOWLEDGE THAT WÜRTH ELEKTRONIK EISOS MAKES NO REPRESENTATIONS AND WARRANTIES OF ANY KIND RELATED TO, BUT NOT LIMITED TO THE NON-INFRINGEMENT OF THIRD PARTIES' INTELLECTUAL PROPERTY RIGHTS OR THE MERCHANTABILITY OR FITNESS FOR YOUR INTENDED PURPOSE OR USAGE. WÜRTH ELEKTRONIK EISOS DOES NOT WARRANT OR REPRESENT THAT ANY LICENSE, EITHER EXPRESS OR IMPLIED, IS GRANTED UNDER ANY PATENT RIGHT, COPYRIGHT, MASK WORK RIGHT, OR OTHER INTELLECTUAL PROPERTY RIGHT RELATING TO ANY COMBINATION, MACHINE, OR PROCESS IN WHICH THE WÜRTH ELEKTRONIK EISOS' PRODUCT WITH THE INCORPORATED FIRMWARE IS USED. INFORMATION PUBLISHED BY WÜRTH ELEKTRONIK EISOS REGARDING THIRD-PARTY PRODUCTS OR SERVICES DOES NOT CONSTITUTE A LICENSE FROM WÜRTH ELEKTRONIK EISOS TO USE SUCH PRODUCTS OR SERVICES OR A WARRANTY OR ENDORSEMENT THEREOF.

Limitation of liability
Any liability not expressly provided by Würth Elektronik eiSos shall be disclaimed.
You agree to hold us harmless from any third-party claims related to your usage of the Würth Elektronik eiSos' products with the incorporated firmware, software and source code. Würth Elektronik eiSos disclaims any liability for any alteration, development created by you or your customers as well as for any combination with other products.

Applicable law and jurisdiction
Applicable law to these license terms shall be the laws of the Federal Republic of Germany. Any dispute, claim or controversy arising out of or relating to these license terms shall be resolved and finally settled by the court competent for the location of Würth Elektronik eiSos registered office.

Severability clause
If a provision of these license terms is or becomes invalid, unenforceable or null and void, this shall not affect the remaining provisions of the terms. The parties shall replace any such provisions with new valid provisions that most closely approximate the purpose of the terms.

Miscellaneous
Würth Elektronik eiSos reserves the right at any time to change these terms at its own discretion. It is your responsibility to check at Würth Elektronik eiSos homepage for any updates. Your continued usage of the products will be deemed as the acceptance of the change.
We recommend you to be updated about the status of new firmware and software, which is available on our website or in our data sheet and manual, and to implement new software in your device where appropriate.
By ordering a product, you accept these license terms in all terms.

List of Figures

List of Tables

Contact
Würth Elektronik eiSos GmbH & Co. KG
Division Wireless Connectivity & Sensors

Max-Eyth-Straße 1
74638 Waldenburg
Germany

Tel.: +49 651 99355-0
Fax.: +49 651 99355-69
www.we-online.com/wireless-connectivity