History

Version	Date	Changes	Status
01	12.07.2019	Born	
02	15.04.2020	Various Adaptions & 1st Release	
03	22.06.2020	Add Termopiler Add USB Console Handling Changed LED description	Release

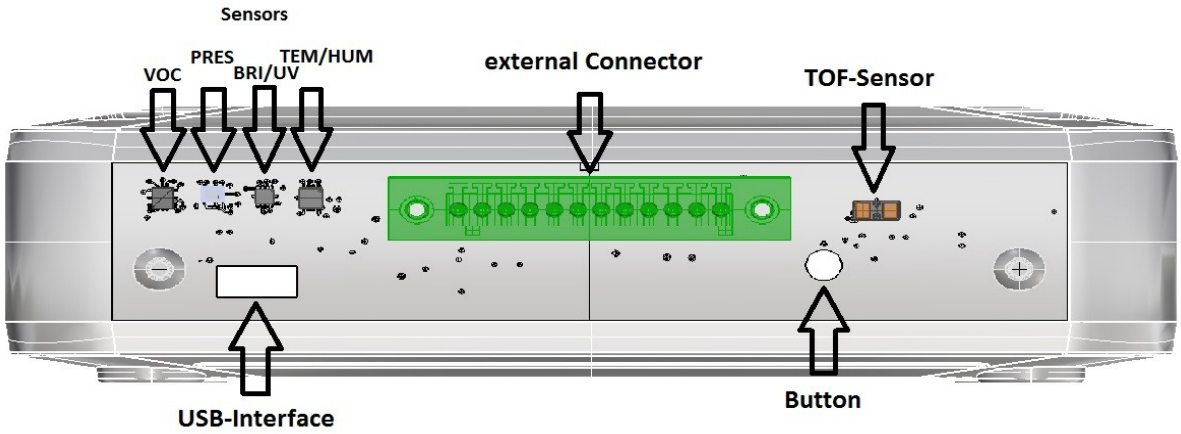
Legal

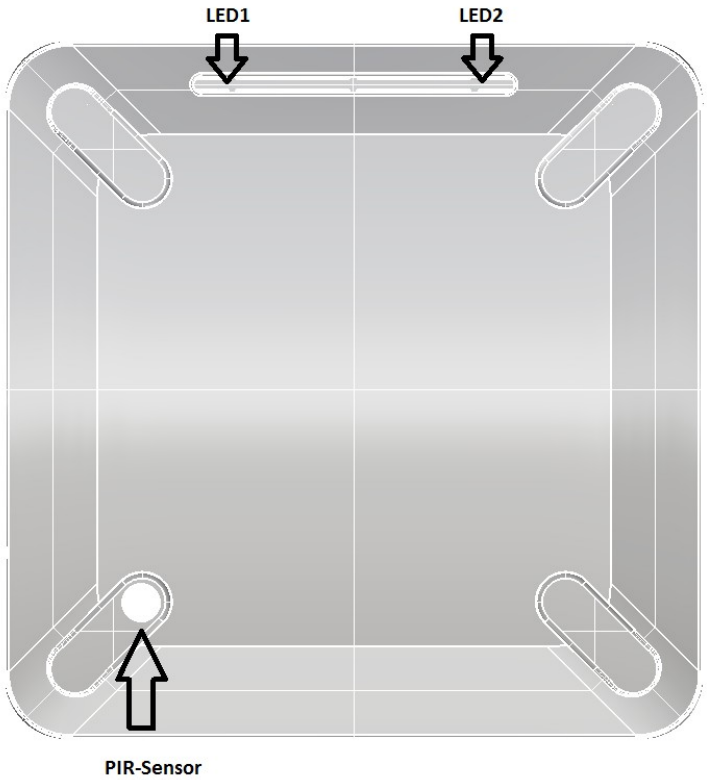
pikkerton GmbH
Wohlrabedamm 16
13629 Berlin
Germany

Fon +49 (0) 30 3300724 -0
Fax +49 (0) 30 3300724 -24
Website www.pikkerton.de

Table of Contents

1 Hardware / Device





1.1 Sensors and power supply recommendations

For following sensor and communication options a permanent power supply is recommended:

- WLAN Communication
- Laser based people counting
- VOC Sensor

1.2 Power Supply

The XBS-200 can be supplied via various internal and external sources:

- Internal: with 1.5V / 3.6V AA batteries,
- External: via the 5V USB interface or a separate 12-24V input

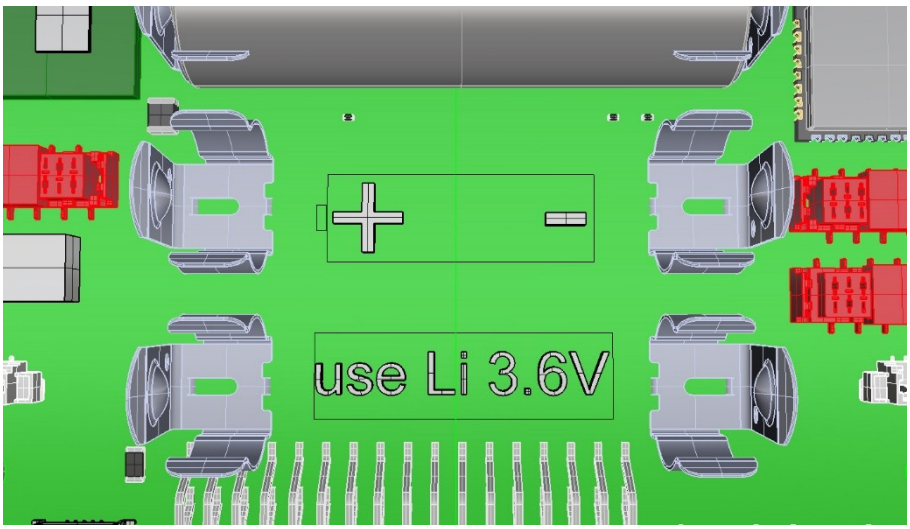
1.2.1 External Power Supply

- USB
  A 5V USB power supply is required. The connector on the device side is Mini-USB-B.
  For devices with LTE modem in GSM mode, a 2.0 Amp power supply unit is required, otherwise a 1.0 Amp model is sufficient.
  For configuration, the XBS-200 can be supplied from the PC via the USB connection. Please note the higher current consumption in GSM mode. The 2.0Amp. would have to be provided by the PC in this case, otherwise the XBS-200 doesn’t boot properly.

- External power supplies
  The XBS-200 can be supplied with an external voltage of 12-24V via the green connector. This option is not available in all hardware configurations.

1.2.2 Battery Operation

The XBS-200 can be equipped with commercially available 1.5V AA cells (Mignon, IEC: R6 / R14505), but 3.6V AA lithium cells are recommended. The hardware configuration determines which cells can be used and is marked with a sticker under the battery holders.
An incorrect battery configuration can damage the device.

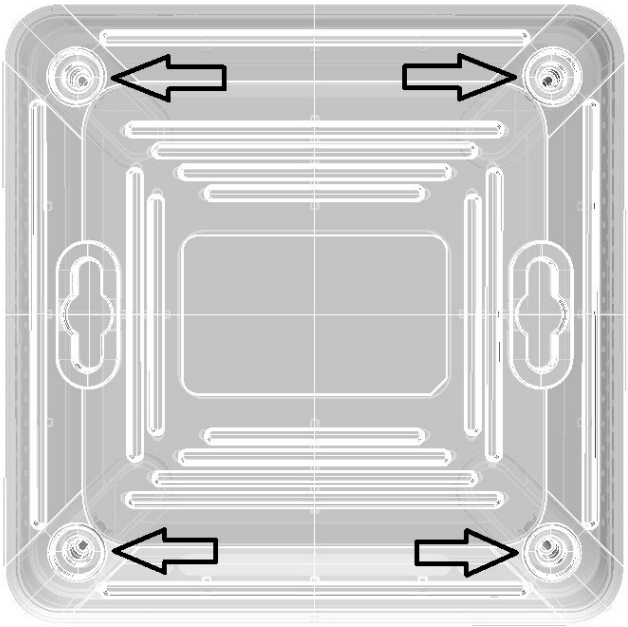


Possible battery configurations:

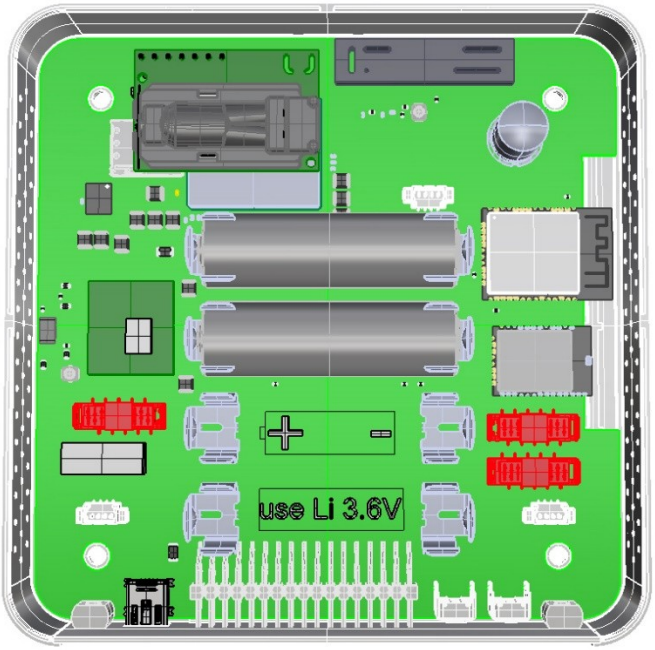
Battery type:	Qty. of cells:	Used socket:
1.5V	2	BATT1,BATT3
1.5V	4	BATT1, BATT2, BATT3, BATT4
3.6V	1	BATT1
3.6V	2	BATT1, BATT2
3.6V	3	BATT1, BATT2, BATT3
3.6V	4	BATT1, BATT2, BATT3, BATT4

Battery replacement:

Open the device by removing the 4 screws on the underside of the device. A TORX screwdriver size T8 is required.



The housing can now be dismantled into the lower part with boards, frame and cover.
Now replace the batteries and pay attention to correct polarity! The polarity is printed on the circuit board under the batteries. Reverse polarity can lead to the destruction of the device, the batteries and the risk of injury!



As soon as the batteries are inserted, the XBS-200 starts. The assembly is done in reverse order.
After the device has been reassembled, please reset the device.

1.3 Reset

The XBS-200 can be restarted using the button or a software command.

- **Button**
  The device has a button with which measurements and a reset can be triggered. To initiate a reset, press the button for at least 10 seconds. All LEDs on the device go out. The XBS-200 restarts, the blue LED1 lights up during the boot phase.

- **Software**
  The device can be restarted via the console using the reset command

  ```
  console> reset
  ```

  The XBS-200 acknowledges this with:

  ```
  Restarting XBS-200!
  ```

  The XBS-200 restarts, the blue LED1 lights up during the boot phase.

- **USB**
  If the XBS-200 is operated with batteries, a reset is necessary to activate the USB interface. USB is deactivated in battery mode to save power.
  Connect the USB cable to the PC and the XBS-200 and trigger a reset with the button. After the reset, the XBS-200 starts in USB mode. If the USB cable is removed, the XBS-200 automatically triggers a reset to restart in battery mode.

1.4 LEDs

The XBS is equipped with 2 RGB-LEDs.

	Solid On	**Flashing**
LED1 Red		Could not send data to broker (MQTT)
LED1 Green		Data sending OK (MQTT)
LED1 Blue	Booting	
LED2 Red	USB console is active	
LED2 Green	Button pressed	
LED2 Blue		

1.5 Pushbutton

A short push onto the button starts a new measurement interval, regardless of the TXT or MSI cycle. All sensor values are going to be transmitted immediately. Pressing the button longer than 10s causes a hardware reset / restart.

2 File System

There is a file system on the XBS which contains two partitions.

2.1 Secure Partition

This partition is used to store general settings and script files.

2.2 R/W Partition

This partition is used for log files and as file upload storage.

3 Measurement

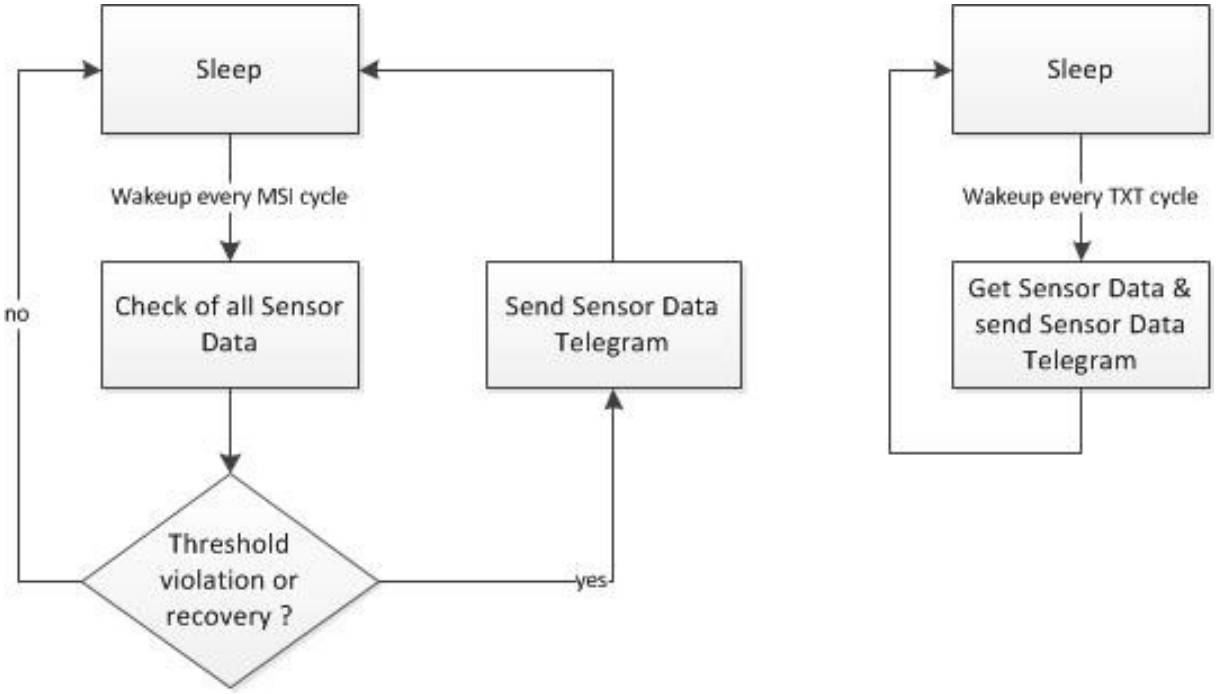
The firmware differentiates between two main cycles / intervals:

- TXT
- MSI

A TXT interval forces the controller subsystem to wakeup, to take fresh sensor values and send these data immediately to the respective receiver. The controller is going into deep sleep state afterwards.

The MSI forces also the controller subsystem to wakeip, to take fresh sensor values and qualify whether pre-configured thresholds are violated or not. If none of the thresholds are violated, the controller will enter deep sleep mode without sending anything. The thresholds can be either MIN or MAX values or also DELTA values. The delta values are generated between the respective last and the actual measurement and are positive [abs()]. If the threshold violation is healed, a 2nd telegram will be sent in order to notify the receiver of this event as well.

A short MSI cycle with a long TXT cycle can therefore combine both, almost realtime data in case of emergency or the need for alerting and longest battery lifetime.



Identifier

Sensor	LWM2M	MQTT Topic	Unit
Temperature	/3303/0/5700	xbs-200/<Serial>/TEM	°C
Humidity	/3304/0/5700	xbs-200/<Serial>/HUM	%
Brightness	/3301/0/5700	xbs-200/<Serial>/BRI	Lux
UV-Index	/3301/1/5700	xbs-200/<Serial>/UVI	%
Air Pressure	/3315/0/5700	xbs-200/<Serial>/PRES	Pa
CO2	/3325/0/5700	xbs-200/<Serial>/CO2	ppm
TVOC	/3325/1/5700	xbs-200/<Serial>/TVOC	mg/m^3
ECO2	/3325/2/5700	xbs-200/<Serial>/ECO2	ppm
CONTACT State	/3200/0/5500	xbs-200/<Serial>/ CONTACT	
PULSE Counter	/3200/0/5501	xbs-200/<Serial>/PC	
PIR	/3200/1/5501	xbs-200/<Serial>/MOVE	
Termopiler Foreground Temperature	na	xbs-200/<Serial>/TPF	1/10°C
Termopiler Background Temperature	na	xbs-200/<Serial>/TPB	1/10°C
Termopiler Quality	na	xbs-200/<Serial>/TPQ	

4 Configuration via the USB Console

This description is based on an Ubuntu System with minicom installed.

```
Ubuntu 16.04.6 LTS
minicom Version 2.7 (kompiliert am Nov 15 2018)
```

Also tested with

```
Ubuntu 18.04.4 LTS
minicom Version 2.7.1 (kompiliert am Aug 13 2017)
```

Main features of the console:

- File upload
- Rename/Move files from r/w to secure partition
- Cat files
- Show hexdumps of a file
- Execute bg96 start scripts
- Check sensor values
- Activate/deactivate sensors
- Dump settings
- Configure msi/txt
- Delete log files
- Show system information

The command 'help' or '?' prints a short list of all known commands.

4.1 Starting the Console

The USB console is disabled by default.

If the XBS is powered off, push the button first and then plug in the USB Cable. If the red LED turns on (shortly after the blue one) the button can be released. Now the system is in USB console mode.

If the system is already running, reset the system by pushing the button for longer than 10s. The green LED turns of if the system is in reset state. Now release the button and push it again very quickly. Hold it until the green and red LED are turned on.

Please reset the device after the console is no longer needed. A running UBS console can cause imprecise temperature measuring.

4.1.1 Settings of the Serial Connection

Device: `/dev/ttyACM<x>`

Connection: 115200 8N1

Connect with minicom and press enter to see if the console is working properly. It happens sometimes that the USB-driver could not connect to the XBS. If the dmesg reports an error please reboot the Linux system. Then the connection work properly again.

4.1.2 Command Overview

To get an overview to all commands simply type `?` or `help` and press enter.

```
console> ?

exec_at <script name>
sensors
bg <timeout>,<cmd>
led <test>
dump
cat <filename>
log
get <setting> ...
set <setting>=<value>
hex <filename>
ls [-l]
reset
info
rm [-sec] <filename>
sec <filename>
fs -check, fs -format [-sec/-norm]
mv <old name> <new name>
write <filename> <filesize>
```

4.1.3 File Upload

Start the minicom in this directory where the file for upload is located.

Enter the command 'write' the filename for xbs and the file size in Bytes. The filename can be different from the name on the linux host system.

`console> write at.txt 607`

1. press <crtg>+<a> then <s>
2. from the upload menu select xmodem
3. select the file from the Linux file system

4.1.4 Rename/Move Files

Rename files:

```
console> mv old.txt new.txt
```

Move files to the secure-partition:

```
console> sec file.txt
```

4.1.5 Cat files

Useful to check Log-Files:

```
console> cat system.log
```

4.1.6 Show hexdumps of a file

```
console> hex system.log
```

4.1.7 Check sensor values

Displays the current sensor data for all existing and activated sensors.

```
console> sensors
       BRI: Sensor not active

       UVI: Sensor not active

       PRES: 101839.695 Pa

       TEM: 25.528 Â°C

       HUM: 25.993 %

       ECO2: 400.000 ppm

       TVOC: 0.000 ppm

       ACC: Sensor NOT FOUND!!!!!!!!!
```

Not existing sensors are marked with : `Sensor NOT FOUND!!!!!!!!!`
Not activated sensors are marked with : `Sensor not active.`

If sensors are activated / deactivated, a reset is necessary to make the apply changes. After a reset the sensors-command shows updated informations.Activate/deactivate sensors

The existing sensors can be activated and deactivated via the console, the changes are permanent and are saved in the sensors.txt.

```
       console> set pres_enable=on

       pres_enable"="on": OK
```

The command is confirmed by the XBS-200 with the return of the set value.

The list of the supported sensors varies depending on the firmware version and not all sensors are available in all hardware variants.

All editable values can be displayed using the DUMP command (see dump settings).

4.1.8 Dump settings

All variables can be displayed using the DUMP command. Variable names are required, for example, to activate / deactivate individual sensors or to change system settings. The displayed values correspond to the default values and not the current values.

```
console> dump
```

example 1:

..

```
pres_enable,onoff,on,Enables the air pressure sensor
```

..

Variable “pres_enable”, type: onoff, default-value: on.

(see activate/deactivate sensors)

example 2:

..

```
serial,const_string,XSB200010000,Serial
```

..

Cannot be changed, recognizable by “const” (constant).

4.1.9 Configure MSI/TXT

The measurement (MSI) and transmission (TXT) interval can be configured via the console, the changes are permanent and are saved in the settings.txt.

```
console> set msi=30
"msi"="30": OK
console> set txt=600
"txt"="600": OK
```

The value is given in minutes. The command is confirmed by the XBS-200 with the return of the set value.

4.1.10 Delete log files

After installation, after changing the battery or changing the configuration, it may make sense to delete the Log-Files in order not to unnecessarily inflate the files with false alarms.

```
console> rm system.log
Delete file successful
```

LOG files in the R/W partition can be safely deleted without affecting the functionality of the device.

4.1.11 Show system information

Displays information such as device type, serial number and firmware version.

```
console> info
PID:                    XBS-200
Serial Number:          XBS200010012
Firmware Version:       0.13_nb-iot
Build Date:              08:01:01 - Apr  7 2020
Prozessor ID:           0x461
Prozessor Name:         STM32L496ZGT
Prozessor Rev:          0x2000
Prozessor UUID:         0x3932211 0x1464029189 0x540227912
```

5 WAN Modem (Quectel BG96)

5.1 Start Scripts

After a reset/power on, the XBS starts the BG96 and checks whether it answers to an AT command.

The BG96 is configured by a script. This script has one command per line. The comand has two parameters seperated by a comma. The line ending encoding can be UNIX, MAC and Windows. If a line starts with an '#' the line will be ignored. Here is an example for NB-IOT with 1nce:

```
# Script for connecting with 1nce.de via nb-iot
#
#
# No left free spaces after the command is allowed. The sytax:
# <timeout [ms]>,<at command><cr><lf>
#
# At least a <cr> or <lf> must be placed at the end of a line.
# Emplty lines are allowed.
#
# Max script file size 3000kB
# NB1 only, no M1
300,AT+QCFG="iotopmode",1,1
# use LTE technology with GSM fallback
300,AT+QCFG="nwscanmode",0,1
# Network Searching Sequence NB1>GSM>M1
300,AT+QCFG="nwscanseq",030102,1
# for 1nce Europe
300,AT+QCFG="band",0,0,80,1
# connect to 1once nb
180000,AT+COPS=1,2,"26201",9
# activate context 1 for internet communication
180000,AT+QICSGP=1,1,"iot.1nce.net"
```

The firmware expects a running internet connection on context id 1 (see the last command for e.g. 1nce).

The name of the script in the XBS must be `at.txt`

It can be located in the secure and r/w partition.

5.2 Testing the script

The scripts can be tested via the console. In this case the script name must be given as parameter, so that different scripts from `at.txt` can be tested.

```
console> exec_at at-test.txt
```

After the execution is finished, there is a log-file available, consisting of the commands and the answers of the BG96 module. Here is the log from the `at.txt` above:

```
console> cat atCommand.log
AT+QCFG="iotopmode",1,1
OK
AT+QCFG="nwscanmode",0,1
OK
AT+QCFG="nwscanseq",030102,1
OK
AT+QCFG="band",0,0,80,1
OK
AT+COPS=1,2,"26201",9
OK
```

If the script is working properly and should be used as start script, rename it to '`at.txt`' and move it to the secure partition. The script name can be changed by the setting `qt_start_script`

```
console> mv at-test.txt at.txt
OK
console> ls
->Secured files:
settings.txt
->Unsecured files:
at.txt
console> sec at.txt
secure file success
console> ls
->Secured files:
at.txt
settings.txt
->Unsecured files:
console>
```

6 Settings

The settings of the XBS are managed by text based configuration files.

The line ending must be in UNIX style and must not contain empty lines.

The setting files must be places in the secure partition.

The syntax of the settings is:

```
<id>,<type>,<setting>,<description><0x0a>
```

6.1 Common settings in the configuration files

settings.txt

serial	const_string	Serial
variant	const_int	Variante
hw_version	const_int	Hardware Version
low_bat	int	Low battery alert threshold [mV]
msi	int	Measurement Intervall [s] 0 = off
txt	int	Sending Intervall [s]
dot	int	Device online time after last packet (in psm mode) 0..120 [s]
max_logfile_size	const_int	max logfile size (Bytes)
bg96_enable	onoff	Enables the bg96 module
bg96_psm	onoff	Enables psm of the bg96 module
wifi_enable	onoff	Enables the wifi module
ble_enable	onoff	Enables the ble module
wmbus_enable	onoff	Enables the Wireless-MBus module
wmbus_tmode	onoff	on: T1-Mode Chan 11; off: S1-Mode Chan 10
qt_start_script	string	Name of the at command start script
qt_wake_script	string	Name of the at command wake script
lwm2m_enable	onoff	Lwm2m On/Off
lwm2m_queue_enable	onoff	Lwm2m Queued Mode On/Off
lwm2m_online_time	int	Client online time in queued mode [s]
lwm2m_server	string	Lwm2m Server URL
lwm2m_server_short_id	int	Short ID of the Lwm2m Server
mac_ap	const_string	MAC-Address of Access Point
mac_st	const_string	MAC-Address of Station

esp_update	onoff	Update the ESP via Cloud
esp_max_payload	const_int	Max TCP Payload of the WLAN module
esp_update_ping	string	Ping for Update to check Web Connectivity
esp_max_rf	const_int	Max. RF Power [0..11]

mqtt.txt

m_client_enable	onoff	Custom MQTT-Client Enable
m_broker_address	string	Brokers Host Address
m_broker_port	int	Brokers Host Port
m_client_id	string	MQTT Client ID
m_user	string	MQTT Username
m_passwd	string	MQTT Passwort
m_send_base_topic	string	Topic to send
m_nct	int	Break between two connections[min]
m_ct	int	Onlinetime before disconnecting [min]

sensors.txt

bri_uv_enable	onoff	Enables the brightness/uv sensor
bri_lt	float	Lower Threshold for BRI [0.0 lux .. 10000.0 lux]
bri_ht	float	Upper Threshold for BRI
bri_delta	float	Delta Threshold for BRI
uv_lt	float	Lower Threshold for the UV-Index [0.0% .. 100.0%]
uv_ht	float	Upper Threshold for the UV-Index [0.0% .. 100.0%]
uv_delta	float	Delta Threshold for the UV-Index [0.0% .. 100.0%]
co2_enable	onoff	Enables the CO2 sensor - not implemented yet
co2_lt	int	Lower Threshold for CO2 [400ppm .. 60000ppm]
co2_ht	int	Upper Threshold for CO2 [400ppm .. 60000ppm]
co2_delta	float	Delta Threshold for CO2 [0ppm .. 60000ppm]
eco2_enable	onoff	Enables the eCO2 sensor
eco2_lt	float	Lower Threshold for eCo2 [400.0ppm .. 10000.0ppm]
eco2_ht	float	Upper Threshold for eCo2 [400.0ppm .. 10000.0ppm]
eco2_delta	float	Delta Threshold for eCo2 [0.0ppm .. 10000.0ppm]
tvoc_lt	float	Lower Threshold for eCo2 [0.0 .. 10000.0]
tvoc_ht	float	Upper Threshold for eCo2 [0.0 .. 10000.0]
tvoc_delta	float	Delta Threshold for eCo2 [0.0 .. 10000.0]

pres_enable	onoff	Enables the air pressure sensor
pres_lt	float	Lower Threshold for Air Pressure [0.0Pa .. 1000000.0Pa]
pres_ht	float	Upper Threshold for Air Pressure [0.0Pa .. 1000000.0Pa]
pres_delta	float	Delta Threshold for Air Pressure [0.0Pa .. 1000000.0Pa]
tem_hum_enable	onoff	Enables the temperature/humidity sensor
tem_lt	float	Lower Threshold for Temperature [-40.0Ã‚Â°C .. 125.0Ã‚Â°C]
tem_ht	float	Upper Threshold for Temperature [-40.0Ã‚Â°C .. 125.0Ã‚Â°C]
tem_delta	float	Delta Threshold for Temperature [-40.0Ã‚Â°C .. 125.0Ã‚Â°C]
hum_lt	float	Lower Threshold for Humidity [0.0% .. 100.0%]
hum_ht	float	Upper Threshold for Humidity [0.0% .. 100.0%]
hum_delta	float	Delta Threshold for Humidity [0.0% .. 100.0%]
mov_enable	onoff	Enables the moving sensor (PIR)
mov_qt	int	Quietness time for Move Messages[s]
pc_enable	onoff	Enables the pulse counting
pc_value	int	Last pulse counter value
contact_enable	onoff	Enables the contact monitoring
pc_contact_dbt	int	Debounce time of the pulse counter [ms]

wlan.txt

ip_ap	ip	IP Address of of Grannyguard
ip_st	ip	(station mode) IP Address
sn_ap	ip	(AP mode) Subnet Mask
sn_st	ip	(station mode) Subnet Mask
gw_ap	ip	(AP mode) Gateway
gw_st	ip	(station mode) Gateway
dhcp_ap	onoff	(AP mode) DHCP On/Off
dhcp_st	onoff	(station mode) DHCP On/Off
ssid_ap	ascii	(AP mode) SSID of own Access Point
ssid_st	ascii	(station mode) SSID of Access Point to connect to
pw_ap	ascii	(AP mode) Password of own Access Point
pw_st	ascii	(station mode) Password of remote Access Point
enc_ap	int	(AP mode)(Encryption) 0: Off, 2: WPA_PSK, 3: WPA2_PSK, 4: WPA_WPA2_PSK

chan_ap	int	(AP mode)channel ID
maxcon_ap	int	(AP mode)maximum number of Stations to which AP can be connected
ssid_hidden_ap	int	(AP mode)0: SSID is broadcasted, 1: Hidden

6.2 Change settings via the console

The settings can also be read and modified by the serial console. Here is an example how to disable the bg96 LTE module:

```
console> get bg96_enable
bg96_enable = "on"

console> set bg96_enable=off
"bg96_enable"="off": OK

console> get bg96_enable
bg96_enable = "off"

console>
```

6.3 Remarks

6.3.1 lwm2m_server

The lwm2m server setting has to follow the format:

`coap://<IP or URL>:<port>`

6.3.2 m_broker_address

The broker address can be an IP address or an URL.

6.3.3 m_ct and m_nct

These are variables for further use. Don't change these settings.