

# Technische Doku

- Software Doku
- Hardware Doku
- Datenverarbeitung

# Software Doku

# Hardware Doku

## Ladeparameter

### MPPT Parameter

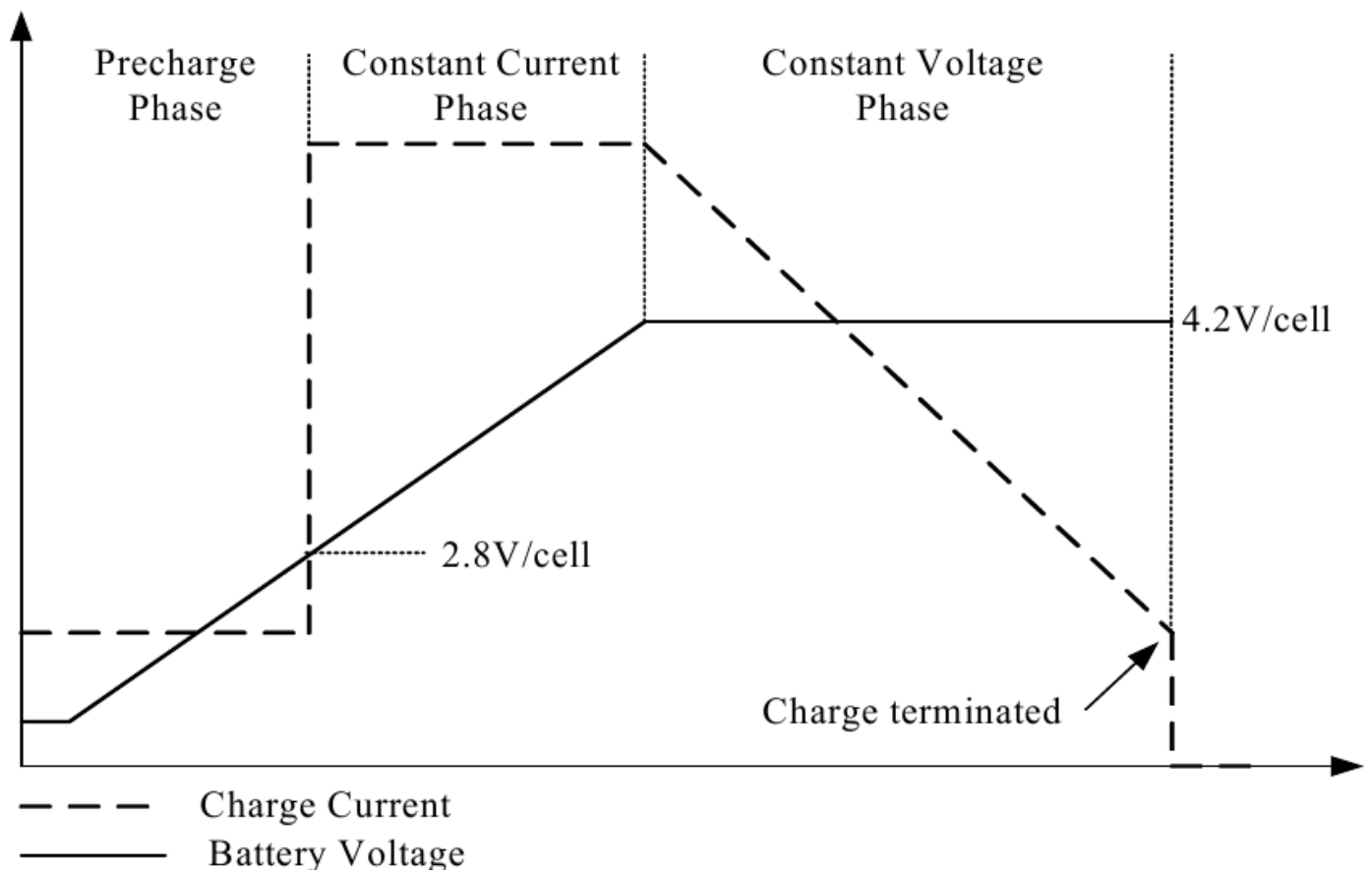
Über DIP Switches kann man einstellen bei welcher Spannung die Solarpanele betrieben werden sollen. Default=12V, bei schwachen Panels kann man auf 9V gehen (zuerst 12V abschalten, dann 9V einschalten !)

## Solar-Laderegeling

Es wird geladen bis 4.2V Spannung an der Batterie anlegen, danach ist die Ladung aus Sicht des Reglers fertig.

Doku zum Laderegler in Obercloud/Projekte/connecting-peaks/Sensor\_Elektronik/DataSheets/CN3791-CONSONANCE.pdf

Ladekurve:



Über den DIP Switch kann man die LEDs einschalten (ziehen auch Strom !), dann sieht man den Betrieb:

$\overline{\text{CHRG}}$ pin	$\overline{\text{DONE}}$ pin	State Description
Low(the red LED on)	High Impedance(the green LED off)	Charging
High Impedance(the red LED off)	Low(the green LED on)	Charge termination
Pulse (the red LED blinking)	Pulse (the green LED on or blinking)	Battery not present
High Impedance(the red LED off)	High Impedance(the green LED off)	There are 2 possible reasons: <ul style="list-style-type: none"> <li>the voltage at the VCC pin below the UVLO level or</li> <li>the voltage at the VCC pin below <math>V_{\text{BAT}}</math></li> </ul>

## Batterie

Sie hält lange die Spannung, danach fällt sie abrupt ab.

Die Default-Einstellungen in der Config-Datei (unter 3.5V in Tiefschlaf gehen) kommen von ConnectingPeaks v1, damit laufen die Geräte stabil seit 2 Jahren → beibehalten

## Laden über USB

Wenn USB angeschlossen ist, kann das Lyligo Board bis 500mA Strom ziehen, und über das kleine Kabel das Board versorgen und somit die Batterie laden. Über Solar fließen eher 100mA.

## Undervoltage Lock

Wenn die Spannung zu tief senkt, geht der Lyligo in den Tiefschlaf, kann auch nicht mehr durch den Reset-Knopf am Board oder über den Magneten aufgeweckt werden. Lösung : USB einstecken um eine saubere Stromversorgung sicherzustellen, auf dem Mainboard Switch SW1 auf OFF, Lyligo aus dem Stecker ziehen, damit das Lyligo Board sich komplett zurücksetzt.

## DIP Switches

Switch Label	Funktion
6V, 12V, 15V, 18V, 24V	Operationspunkt vom MPPT Regler definieren – siehe MPPT Parameter
LEDS	Aktiviert die Status-LEDs der <b>Solar</b> -Laderegelung – siehe Solar-Laderegelung
Boot	## klären mit Tom## Lyligo Modul versucht zu booten weg möglich, geht nicht in Deep Sleep

Sowohl LEDS als auch Boot sollten aus sein wenn die Geräte geliefert werden, um zu vermeiden dass die Batterien tiefentladen werden und kaputt gehen. LEDS aus reduziert zusätzlich den Stromverbrauch.

## Abschaltung & Tiefschlaf

Verschiedene Spannungslevel können in der Konfiguration angegeben werden, ab denen andere Schlafzeiten verwendet werden um die Batterie zu schonen.

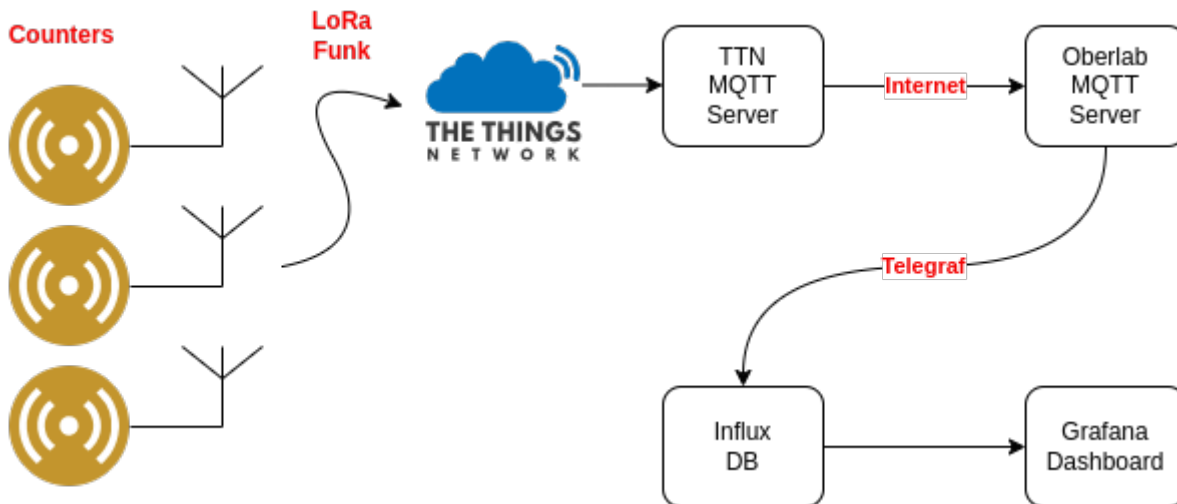
Spannung zwischen	und	Schlafzeit
max	low_power_slowdown1_mV	normal_sleep_time_s
low_power_slowdown1_mV	low_power_slowdown2_mV	slowdown1_sleep_time_s
low_power_slowdown2_mV	low_power_slowdown3_mV	slowdown2_sleep_time_s
low_power_slowdown3_mV	0	slowdown2_sleep_time_s



# Datenverarbeitung

Diese Seite beschreibt den Datenfluss, von den Sensoren bis zum finalen Dashboard.

## Übersicht



## Datentransfer Sensor → TTN

Die Messwerte der einzelnen Counter werden über das LoRa Funkprotokoll an das The Things Network (TTN) übermittelt.

Im Oberlab Account bei TTN ist die Applikation *oberlab-counter-sandbox* dafür eingerichtet worden. DeviceIDs für insgesamt 15 Geräte (5 jeweils für Schwaz/Geretsried/Miesbach) wurden angelegt. Die Liste ist unter `/Projekte/connecting-peaks/Sensor_Code/TTN_EIDs.ods` in der OberCloud abgelegt.

Der Payload Formatter setzt die 10 Bytes Daten in folgende felder um:

```
data.wifi = (input.bytes[1] << 8) + input.bytes[0];
data.ble = (input.bytes[3] << 8) + input.bytes[2];
data.battery = (input.bytes[5] << 8) + input.bytes[4];
data.wifi_new = (input.bytes[7] << 8) + input.bytes[6];
```

```
data.ble_new = (input.bytes[9] << 8) + input.bytes[8];
```

Die fertigen Datenpakete stehen anschließend am **MQTT** Server von TTN zur Verfügung und können prinzipiell dort unter dem Topic `v3/oberlab-counter-sandbox@ttn/` abgegriffen werden.

# Ober-MQTT Server

Der zentrale MQTT Server vom Oberlab spiegelt die Sensordaten jedoch nochmal in's Oberlab, wo sie unter dem Topic `connecting_peaks/sandbox/` verfügbar sind.

## Ablage in der Datenbank

Die Abfrage vom MQTT Server und die Übermittlung in die Influx Datenbank erfolgt über den Dienst **Telegraf**. Folgender Teil der `telegraf.conf` ist dafür relevant (*wobei die Renames nicht vollständig sind - geht aber trotzdem...*):

```
[[inputs.mqtt_consumer]]
  ## Topics that will be subscribed to.
  topics = [
    "connecting_peaks/sandbox/#",
  ]
  data_format="json_v2"
  ## Enable extracting tag values from MQTT topics
  ## _ denotes an ignored entry in the topic path
[[inputs.mqtt_consumer.topic_parsing]]
  topic = "connecting_peaks/sandbox/devices/+/+"
  measurement = "_/_/_/measurement"
  tags = "_/_/_/device/_"
[[inputs.mqtt_consumer.json_v2]]
[[inputs.mqtt_consumer.json_v2.object]]
  path = "[@this]"
  timestamp_key = "epoch"
  timestamp_format = "unix"
[inputs.mqtt_consumer.json_v2.object.renames]
  battery = "uplink_message.decoded_payload.battery"
  ble = "uplink_message.decoded_payload.ble"
  wifi = "uplink_message.decoded_payload.wifi"
```

Die Daten werden in die InfluxDB Datenbank *sandbox* abgelegt.

# Visualisierung in Grafana

Grafana dient der Anzeige der Daten aus der InfluxDB. Somit können nur Datensätze angezeigt werden, die die gesamte Übertragungskette LoRa→TTN→MQTT→MQTT→Telegraf→InfluxDB durchlaufen sind.

Es können verschiedene Dashboards erstellt werden, um unterschiedliche Darstellungen anzubieten.

Das Hauptdashboard ist unter

<https://connectingpeaks.oberlab.de/d/FN8cHh8Vz/sandbox?orgId=1&refresh=1m&from=now-24h&to=now> öffentlich erreichbar.

Eine Vor-Selektierung der Sensoren kann in die URL eingebunden werden, z.B. für Schwaz:

<https://connectingpeaks.oberlab.de/d/FN8cHh8Vz/sandbox?orgId=1&refresh=1m&var-Sensors=schwaz-pxc-01&var-Sensors=schwaz-pxc-02&var-Sensors=schwaz-pxc-03&from=now-24h&to=now>

Folgende Messgrößen sind in den jeweiligen Graphen ersichtlich:

- WIFI: Anzahl der erfassten WiFi Geräte
- BLE: Anzahl der erfassten Bluetooth Geräte
- WIFI New: Anzahl der erfassten WiFi Geräte, die an diesem Tag noch nicht gezählt wurden
- BLE New: Anzahl der erfassten Bluetooth Geräte, die an diesem Tag noch nicht gezählt wurden
- WIFI new 24h: Gesamt Zahl der WiFi Geräte, die an dem Tag erfasst wurden
- BLE New 24h: Gesamt Zahl der Bluetooth Geräte, die an dem Tag erfasst wurden
- Voltage: Spannung der internen Batterie – Paxcounter läuft im Normalbetrieb solange die Spannung über 3.7V bleibt

Ein Punkt auf dem jeweiligen Graphen entspricht einer übertragenen Messung. Die Kurven der 3 Counter sind farblich unterschiedlich dargestellt.

# Projektspezifische Dienste

Die Dienste Telegraf, InfluxDB und Grafana laufen in einer projektspezifischen VM ( *pconnectingpeaks*) auf dem Oberlab-Server. Damit ergibt sich eine logische Abgrenzung zu anderen Oberlab-Projekten.

