|  |  |
| --- | --- |
| Autor | Daniel Schären |
|  |  |
| Version | V1 |
| Datum | 21.09.2022 |
| Status | In Arbeit |
| Dokument Name | IoT Reference Testkit |
| Server Ablage |  |

A picture containing text

Description automatically generated

Inhalt

[1 Einleitung 5](#_Toc114654265)

[2 LPWAN Netze 5](#_Toc114654266)

[3 CAT M1 5](#_Toc114654267)

[4 NB-IoT 7](#_Toc114654268)

[5 eDRX (extended Discontinuous reception) 9](#_Toc114654269)

[6 PSM (Power Save Mode) 9](#_Toc114654270)

[7 Hardware 10](#_Toc114654271)

[7.1.1 M5-Core 11](#_Toc114654272)

[7.1.2 I2C Hub 11](#_Toc114654273)

[7.1.3 Meteosensoren 11](#_Toc114654274)

[7.1.4 Modem 11](#_Toc114654275)

[7.2 Schnell-Inbetriebnahme 11](#_Toc114654276)

[1 Einzelpunkte Inbetriebnahme und Funktionsweis 12](#_Toc114654277)

[1.1 SIM-Karte einsetzen 12](#_Toc114654278)

[1.2 Speisung 12](#_Toc114654279)

[1.2.1 Konfiguration 12](#_Toc114654280)

[2 Parametereinstellung 13](#_Toc114654281)

[3 Funktionen 14](#_Toc114654282)

[3.1.1 Stromaufzeichnung 14](#_Toc114654283)

[4 Dateiablage 15](#_Toc114654284)

[5 Software 15](#_Toc114654285)

[6 ThingsBoard Cloud 17](#_Toc114654286)

[7 Roules konfigurieren 19](#_Toc114654287)

[7.1 IP-Check 19](#_Toc114654288)

**OFFENE FRAGEN**

* Netzkennung CAT M1 gleich wie NB-IoT (Marco vonMoos)
* Grösse Datenpuffer Download im S-GW

PUNKTE

* Kommentare im Code
* Parameterübergabe an Thingsboard Cloud
* Stromverbrauchserkennung nur auf dem Gerät
* PSM Settings CFUN = 0
* Anzeigen der zurückgelesenen Werte

DOKUMENTATION:

* Mailadressen anpassen
* JSON Roule Nodes laden
* PARAMETER Einstellung nicht Up to Date
* CEDRX und CPSM

# Einleitung

Das IoT Test kit dient zum dauerhaften Überprüfen der Low Power Netzwerkfunktionalitäten LTE Cat M1 und NB-IoT. Das Modem unterstützt aktuell keine Cat-0, Cat-M2, NB-IoT2 (LTE) Standard.

Bei der Swisscom wird mit Stand dieser Doku die LTE Cat M1 Funktionalität über die gleichen EPGs bedient wie alle anderen normalen Mobilefunkkunden. Um insbesondere die Power Saving und eDRX-Funktionen die LTE Cat M1 mit sich bringt testen zu können, wurde dieses Test-Kit entworfen.

Die Idee dahinter. Ein einfaches IoT Device sendet in gewissen Abständen Werte an einen Server. Dieser überprüft ob die Timeoutzeiten eingehalten wurde. Ausserdem werden weitere Werte wie die IP-Adresse des IoT Kits auf Übereinstimmung überprüft.

# LPWAN Netze

Hier kurz eine Erklärung zu dem wichtigsten Parameter. Im Appendix oder auch im Internet findet man ansonsten deutlich tiefere Erklärungen.

Für NB-IoT gelten die gleichen Einstellungen. Dort sind aber in den allgemein deutlich höheren Zeiten insbesondere beim PSM erlaubt.

Der Autor findet es noch wichtig folgendes als Verständnis zu sagen 😊. Es ist gerade in der Schweiz zu beachten das Swisscom mit dem Band 20 (800Mhz) eine tiefe Frequenz für die LPWAN Anwendungen verwendet, die eigentlich gut für die Flächenversorgung geeignet ist. In städtischen Gebieten ist aber vor allem das Band 3 (1800Mhz) Netz gut ausgebaut und das Band 20 Netz ist oftmals deutlich schlechter Empfangbar. Dies sieht man immer wieder gut, wenn man ein Stadt Land Vergleich der RSSI-Werte macht. Dies hat also nichts mit einer schlechten Empfangsantenne oder dergleichen am UE zu tun.

# CAT M1

Cat M1 wurde für die M2M Anwendungen entwickelt. Es bietet zwar eine deutlich kleinere Datenrate als normales LTE, da die Bandbreite auf 1,4Mhz begrenzt ist. Die Sendeleistung bleibt aber identisch was dafür sorgt das die Reichweite deutlich höher ist. Bei der Swisscom laufen diese aktuell im Guard Band vom LTE Band 20 Netz (Schweiz) und Band 3 (Lichtenstein). Cat M1 kann sowohl in Band, das heisst es werden 1.4Mhz innerhalb des 10Mhz Blocks vom Band oder aber auch im Guardband platziert sein. Aktuell sind diese bei der Swisscom im Guardband angelegt.

|  |  |
| --- | --- |
| Frequenzband | LTE Band 20 (Schweiz)  LTE Band 3 (Lichtenstein) |
| Datenraten | Download und Upload 0,1 Kbit/s - 1 Mbit/s |
| Maximaler Pfadverlust (MCL) | 155 dB |
| Max. Abdeckung Schweiz Aussen Max. Abdeckung Schweiz Innen Anmerkung: Die Abdeckung von CAT M1 und NB-IoT ist in urbanen Gebieten vergleichbar. In ruralen Gebieten hat NB-IoT die bessere Abdeckung. | 99.9% 99.6% |
| Reichweiten Steigerung (Coverage Enhancement) | Coverage Enhancement A & B  Max. 256 – 2048 Retransmissions Frequencyhopping |
| Empfangsantenne UE (User Equipment) | Kein MIMO (Multiple Input Multiple Output) |
| Max. Batterielaufzeit | Max. 5 – 10 Jahre |
| Sprachtelefonie | Ja, VoLTE |
| Realtime Anwendungen | Ja |
| Latenz | 10 – 200ms |
| Duplex-Übertraung | Full-Duplex Frequency Division Duplex (FDD) & Time Division Duplex (TDD) |
| Max. Gerätedichte | 10'000 pro Funkzelle |
| Standard Stromsparfunktion / Discontinuous Reception (DRX) | 2.56s |
| Erweiterte Stromsparfunktion / Extended Discontinuous Reception (eDRX) | 44 Min |
| Erweiterte Stromsparfunktion / Power Saving Mode (PSM) | 413 Tage |
| Mobilität | Handover |

Diagram

Description automatically generated

Aus der Sicht des Corenetzes erfolgt der Download der Daten genau gleich wie bei einer normalen LTE-Verbindung über das S1U Interface. Ist das UE aufgrund von PSM oder eDRX nicht erreichbar, werden die Daten im S-GW Zwischengespeichert. Die Grösse des Datenpuffers wird laut ELEX über einen Parameter in der

# NB-IoT

NB-IoT ist anderes als LTE Cat M1 nicht eine Erweiterung von LTE sondern eine eigne Kategorie und kann sowohl Stand Alone im GSM Band oder im LTE Band laufen. Bei der Swisscom laufen diese gleich wie bei LTE Cat M1 aktuell im Guard Band von LTE Band 20 Netz (Schweiz) und Band 3 (Lichtenstein). NB-IoT bietet eine noch deutlich kleinere Datenrate als CAT M1 und auch kein Handover, dafür sind die Module billiger in der Herstellung, verbrauchen noch weniger Energie, da die Überwachung der Nachbarzellen entfällt. Bei der Swisscom wird für

|  |  |
| --- | --- |
| Frequenzband | LTE Band 20 800 MHz lizenziert |
| Datenraten | DL 0.4 – 30 Kbit/s UL 0.1 – 60 Kbit/s |
| Max. Pfadverlust (MCL) | 164 dB |
| Max. Abdeckung Schweiz Aussen Max. Abdeckung Schweiz Innen | 99.9 % 99.7 % |
| Reichweiten Steigerung (Coverage Enhancement) | Max. 128 / 2048 Retransmissions PSD / UL single-tone multi-tone transmission |
| Empfangsantenne UE | Kein MIMO, kein RX Diversity |
| Max. Batterielaufzeit | Bis zu 10 Jahren |
| Sprachtelefonie | Nein |
| Realtime Anwendungen | Nein |
| Latenz | 1.4 – 10s |
| Duplex-übertragung | Half-Duplex FDD only |
| Max. Gerätedichte | 10‘000 pro Funkzelle |
| Standard Stromsparfunktion - DRX | 10.24s |
| Erweiterte Stromsparfunktion - eDRX | 2.9h |
| Erweiterte Stromsparfunktion PSM | 413 Tage |
| Mobilität | Cell Reselection |

A screenshot of a computer

Description automatically generated with medium confidence

Bei NB-IoT werden die Daten zum UE über das S1 Interface an die MME und von dort an das UE gesendet. Dieses Vorgehen nennt man auch DAS und ist ein weiterer Grund, warum die NB-IoT Geräte viel Effizienter sind als CAT-M1. Der S1U Tunnel muss nicht aufgebaut werden und es entfallen die dafür nötigen Schritte. Randbemerkung: Dieser wird bei LTE immer auf und abgebaut, auch wenn der Bearer existiert, wenn Daten kommuniziert werden und kostet dementsprechend Energie

# eDRX (extended Discontinuous reception)

Im LTE-Netz schalten die UEs ihr Empfänger nur über einen kurzen Zeitraum ein Horchen auf den Pageing Kanälen ob für sie neue Daten vorhanden sind. Diese Zeit wird als DRX (Discontenious Recieve) Time bezeichnet und ist bei LTE auf 1.28, 2.56, 5.12 oder 10.24 Sekunden normiert. Swisscom verwendet eine Pageingtime von 1.28s oder 2.56s. Sprich alle 2.56 Sekunden können UE angepaget werden und Datenpakete überhaupt übertragen werden. In der Zwischenzeit verweilen diese im S-GW.

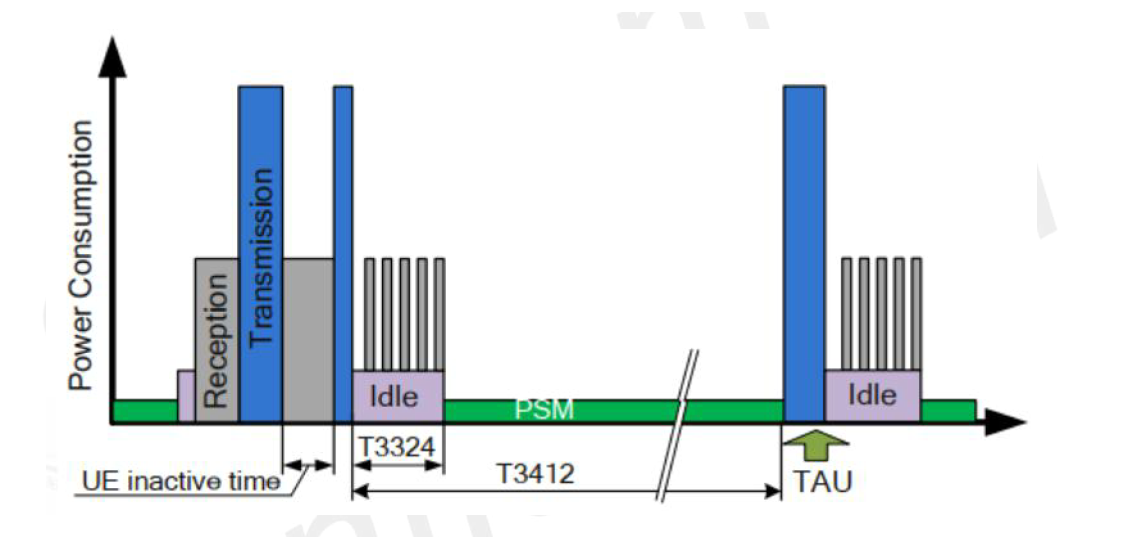
Mit Release 13 wurde dann eDRX (e für Extended) eingeführt. Mit diesem kann das Pageing für eine gewisse Zeit unterbrochen werden. Dabei gibt es denn Parameter eDRX\_Cycle und das Pageing Time Window (PTW)

Der Parameter eDRX abgezogen vom PTW gibt an wie viele DRX-Zyklen der Empfänger nicht auf empfang ist. Daher ist dieser ein Vielfaches von 1.28 Sekunden. Mit dem PTW wird gemeldet wie viele einfache DRX Cycle lang der Empfänger im normalen Pageinig Zyklus erreichbar ist.

Quelle: SIMCOM Doku over LPWAN


# PSM (Power Save Mode)

Für den Power Save Mode bei dem das Modem komplett ausgeschaltet wird, gibt es zwei wichtige Timer. Der Timer T3412 beschreibt wieviel Zeit zwischen denn TAU (Tracking Area Update) vergehen. Beim TAU meldet sich das UE Aktiv beim Netz. Während der Activezeit T3324 ist das Modem aktiv und verhält sich jenachdem ob eDRX eingeschaltet ist nach dessen Vorgaben.



Hardware

# Hardware

Diagram

Description automatically generated

BILD ERSETZEN

Das Setup besteht aus Modulen von M5 Stack. Diese können einfach zusammengesteckt und so um Funktionen erweitert werden kann. Das System besteht aktuell ausfolgenden Modulen:

* Meteosensor Sht30
* I2C Hub
* Current Sensor ADS1115
* Modem Simcom 7080 G

### M5-Core

Der Kern des Aufbaus bildet der M5Core. Dieser beinhaltet einen ESP32 Microcontroller, inkl Display und Buttons. Der M5 Core hat einen Akku mit dem das komplette Setup für ein paar Stunden weiterlaufen kann. Auf Am M5Core kann die SD-Card mit der Konfiguration eingesteckt werden. Falls diese einmal stecken bleiben sollte, was häufiger passiert, kann man das Gehäuse öffnen und sie von innen nach draussen schieben. Über I2C roter Anschluss und UART blauer Anschluss, kommuniziert der M5Core mit der Peripherie.

**Wichtig:** Das Modem wird an Port C und der I2C Bus an Port A angeschlossen

#### M5Core UART Output

Die Kommunikation zwischen Modem und Core kann man über den Virtual COM-Port am Computer sehen. Dazu denn M5 Core an einem Computer anschliessen und mit einem COM-Terminal z.B. Putty den richtigen COM-Port und als Baudrate 115200 einstellen.

### I2C Hub

Der I2C Hub dient der Vervielfachung der I2C Buses an ihm werden Currentsensor und Meteosensor angeschlossen.

### Meteosensoren

Die Meteosensoren dienen dazu Wetterdaten (Temperatur, Druck und Luftfeuchtigkeit) zu messen. Er enthält dazu die beiden Chips Sht30 und QMQ6988. Datenblätter dazu sind im Anhang zu finden.

### Modem

Als Modem kommt das SIMCOM7080G zum Einsatz. Es kann sowohl LTE CAT M1 als auch NB-IoT. Dieses basiert auf dem Qualcomm 7020. Im PSM-Mode benötigt es 3.2uA und eingeschaltet rund 10mA.

**Aktuelle Probleme:** Das Modem würde eigentlich zum Aufwecken und Synchronisieren denn Anschluss PWRCK nach aussen geführt haben. Leider ist dies bei diesem Modul nicht der Fall. Somit kann nicht sauber detektiert werden, wenn das Modem am Schlafen ist und es auch wieder aufgeweckt werden. Theoretisch würden hier auch UIC-Meldungen reichen. Diese werden jedoch teilweise nicht oder stark verzögert gesendet. Somit ist es schwer nachzuvollziehen, ob das Modem überhaupt noch Betriebsberiet ist.

## Schnell-Inbetriebnahme

Folgende Punkte sind zur Inbetriebnahme zu durchlaufen:

1. SD-Karte Konfiguration überprüfen und in den M5 Core einstecken
2. M5Core per USB-C Kabel speisen und denn roten Knopf auf der Seite betätigen
3. Die Konfiguration wird auf dem Display während 10 Sekunden angezeigt.
4. Das Modem wird gestartet und ein Messwert wird an denn Server gesendet.
5. Auf dem Dashboard sollte nun der aktuelle Messwert stehen und das Device als active angezeigt werden

# Einzelpunkte Inbetriebnahme und Funktionsweis

## SIM-Karte einsetzen

Das Modem vom System entfernen, indem die UART-Kabel und die Antenne entfernt wird. Anschliessend müssen auf der Rückseite die 3 Schrauben geöffnet werden. Die SIM-Karte im Slot eingesetzt und anschliessend das Modem wieder in das System integriert werden.

## Speisung

Das System läuft an einem USB 3.0 Port (900mA/5V) an einem Computer Problemlos. Da der M5 Stack einen Akku integriert hat läuft das System auch ohne externe Speisung einen Momentlang weiter bzw. kann dieser denn Verbrauch gut zwischenpuffern. Für den Stand Alone Betrieb sollte ein Standardnetzteil mit 1A ausreichen. M5 Stack empfiehlt ein Netzteil mit mind. 2A Ausgangsleistung.

### Konfiguration

Die Konfiguration wird über Parameter auf der SD-Karte vorgenommen. Auf dieser muss ein File mit dem Namen CONFIGURATION.txt vorhanden sein. Wird kein File mit diesem Namen gefunden wird eine entsprechende Fehlermeldung ausgegeben. Das aktuelle Format mit den Parametern, die dieses File enthalten muss, findet man «Appendix» im Projekt Ordner. Wichtig ist das die Parameter durchnummeriert sind und das am Ende des Files ein END steht.

# Parametereinstellung

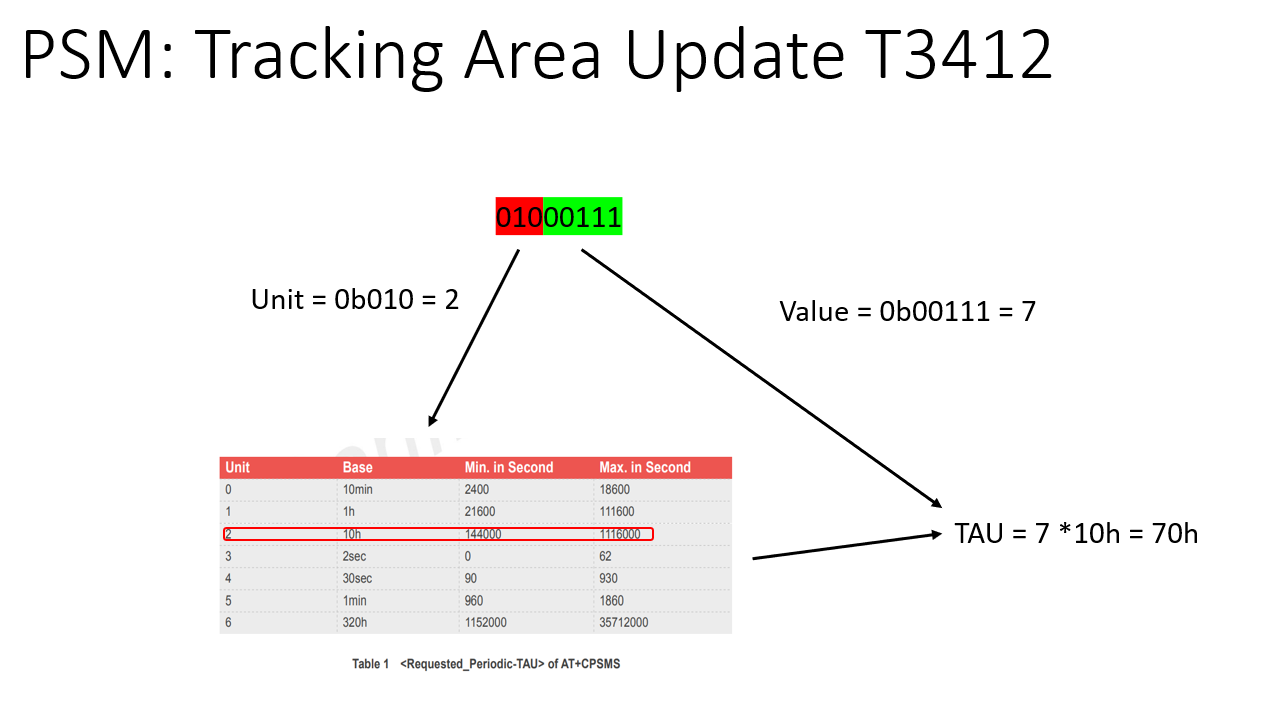
Auf dem CONFIGURATION.txt File auf der SD-Card können aktuell folgende Parameter eingestellt werden. Das Referenzfile befindet sich im Appendix Ordner

Text

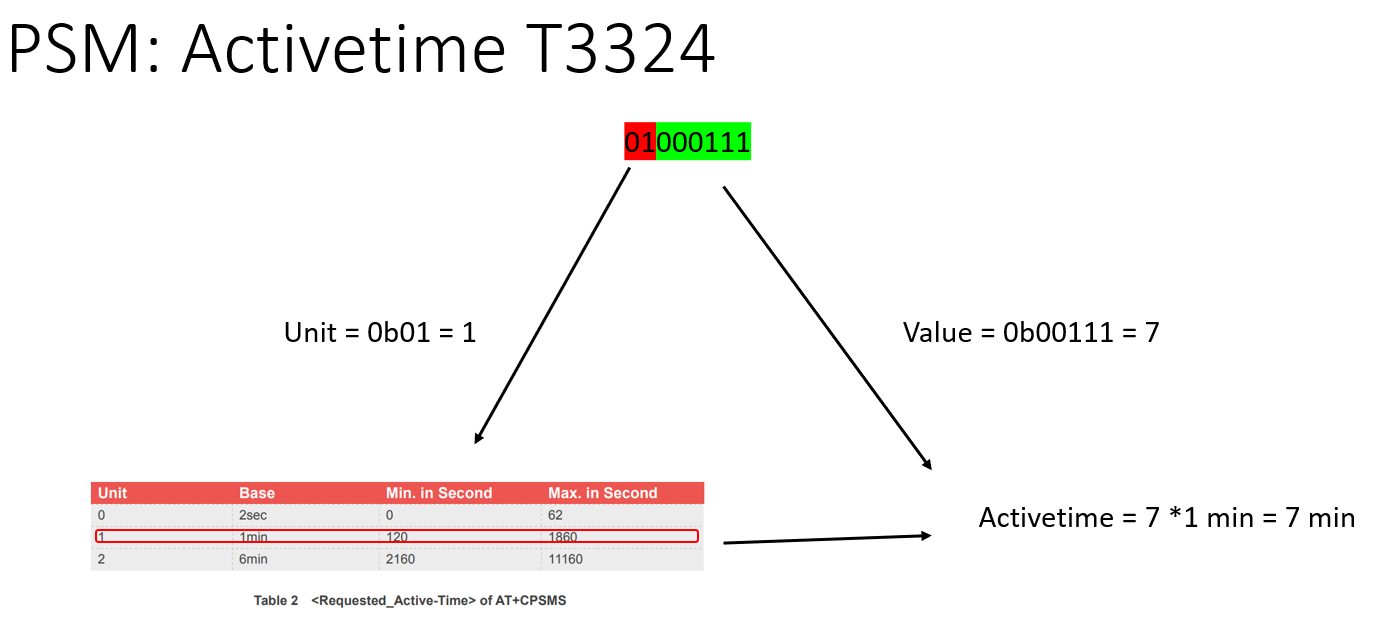
Description automatically generated

|  |  |
| --- | --- |
| eDRX | Dieser Parameter beschreibt die Anzahl DRX Cyclen für eDRX |
| PageingWindow | Mit dem Wert PageingWindow (PTW) kann der |
| TAU\_3412\_Unit | Unit bzw die Base des TAU Timers (siehe unten) |
| TAU\_3412\_Value | Value für das Tracking Area Update |
| Activetime T3324\_Unit | Unit bzw. Base für den Activetimer |
| Activetime T3324\_Value | Value für den Activetimer |
| APN | APN auf den zugegriffen wird |
| RAT | Radio Access Technologie  7 = LTE CAT M1  8 = NB-IoT |
| Mode | Wahl anderer Betriebsmodi. Aktuell nur auf 0 (Synchron mit TAU) |

Das Tracking Area Update wird in Binärer Form am Modem übergeben. Für die Konfiguration existieren die Werte TAU\_Unit und TAU\_Value. Die Unit oder Base ist die Grundzeit und mit der Value wird dann ein Vielfaches dieser Grundzeit gewählt. Die zu jeder Unit gehörende Base ist in der Tabelle untern ersichtlich:



Gleich verhält es sich mit der Activetime, wobei hier die Base nur aus den obersten Zwei Bits besteht.



# Funktionen

### Stromaufzeichnung

Über denn ADS1115 wird alle 50ms der aktuelle Stromverbrauch gemessen und in ein Buffer abgelegt. In regelmässigen Abständen von 1 Sekunde wird vom Strom ein Mittelwert gebildet. Ist der Threashold zu hoch kann ein Alarm gesetzt werden. Dieser muss auch für einen bestimmten Wert konfiguriert werden.

### RSSI Werte

Die RSSI Werte werden während einem TAU ausgelesen und anschliessend an denn Server geschickt

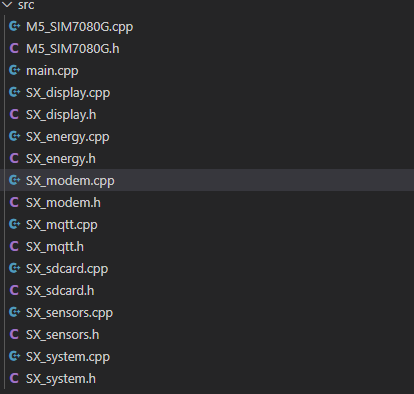
# Dateiablage

Anleitungen befinden sich in Doc da nicht einfach so von SIMCOM herunterladbar. Weiter befindet sich unter Thingsboard die nötigen JSON Files die man auf Thingsboard importieren kann um dort ein

# Software

|  |  |
| --- | --- |
| Die Software ist in Arduino geschrieben. Als Entwicklungsumgebung wird Platformio verwendet. Platformio automatisiert viele Schritte und kann somit schnell auf einem anderen Computer eingerichtet werden. Arduino ist eine einfache schnell zu versehende Programmiersprache. | |
| Text  Description automatically generated  Symbolbild | Platformio läuft innerhalb vom Editor Visual Studio Code: https://code.visualstudio.com/. Daher muss dieses zuerst installiert werden. |
| Graphical user interface, text  Description automatically generated | Anschließend kann Platformio über die Extension hinzugefügt werden. Nach der Installation Visual Studio Code einmal Schließen und wieder öffnen. |
| A picture containing text  Description automatically generated  Symbolbild | Der Sourcecode inkl. Anleitung liegt auf Git. Damit auf Änderung reagiert werden kann wird diese auch dort gepflegt.  Git installieren: [Git - Downloading Package (git-scm.com)](https://git-scm.com/download/win) oder Swisscom eOrder |
| Text  Description automatically generated with medium confidence | Das Projekt an einem Ort auf dem Computer auschecken. Am besten an einem Ort auf dem normalen Userverzeichnis, damit es keine Problem mit OneDrive und Privilegen gibt. Zum Beispiel : C:\Users\TABSCDAV\NoShare  Dazu im entsprechenden Verzeichnis Rechtskicken und Git Bash Here öffnen. |
|  | Nun kann das Projekt geöffnet werden  Hier eine kurze Erklärung zu Platformio: Platformio ist eine IDE mit der auf diverse Plattformen vor allem in Arduino programmiert werden kann. In der Datei Platformio.ini sind die Librarys eingebunden die benötigt werden. Die Libraries werden auf eine gewisse Version eingefroren damit das Projekt immer gleich erstellt wird dazu unter lip-deps, die entsprechende Library mit einem @ auf einen bestimmten Stand bringen. Der Ordner libs wird darauf beim Bauen der Software mit der richtigen Library geladen. |

Alle Funktionen, die erweitert wurden, tragen das Präfix SX. Folgende SX Dateien (src) sind in der Source vorhanden



* SX\_display => Ansteuerung des Displays
* SX\_energy => Stromverbrauchserfassung
* SX\_modem => Ansteuerung des Modems (ausser MQTT)
* SX\_sdcard => Lesen der Konfiguration von der SD-Karte
* SX\_sensors => Die Meteodaten auslesen
* SX\_system => Timer und Callbackfunktionen

# ThingsBoard Cloud

Diagram

Description automatically generated

Das Backend auf der anderen Seite stellt Thingsboard dar. Eine einfache IoT Plattform. In dieser können sogenannte Roule Nodes erstellt werden, mit derer Hilfe auf ein bestimmtes Ereignis (Event genannt) reagiert werden kann. Die Nodes erhalten Eingangsinformationen und leiten daraus eine allfällige Aufgabe ab.

Konkret für dieses Projekt existiert ein Roulenode der aufgrund eines «Device Connectivity Event» (INACTIVITY\_TIMEOUT) ein Script ausführt. Dieses Script füllt eine einfache Mail Nachricht mit den Infos weitersendet

Diagram

Description automatically generated

# Roles konfigurieren

Die Roles können in Thin

## IP-Check

Der IP-Check vergleicht die alte mit der neuen

Dashboard

Graphical user interface, application

Description automatically generated

https://thingsboard.cloud/dashboardGroups/6e2eed10-12f5-11ed-bfd5-090d09a08671/ad8b1f50-130a-11ed-bfd5-090d09a08671