

Benutzerhandbuch IoT-Raumautomation

Windisch, 14. August 2020

Autoren	Gabriel Nussbaumer und Lukas Meienberger
Hochschule	Hochschule für Technik - FHNW
Studiengang	Elektro- und Informationstechnik

Inhaltsverzeichnis

1	Installation Aktor	1
1.1	Anleitung einrichten	1
1.2	Inbetriebnahme	2
1.3	Programmierung	2
1.4	Netzwerkeinstellungen ändern	3
1.5	System Tasten	3
1.6	Topics	3
2	Installation Sensor	4
2.1	Anleitung einrichten	4
2.2	Inbetriebnahme	4
2.3	Programmierung	5
2.4	Netzwerkeinstellungen ändern	5
2.5	System Tasten	5
2.6	Topics	5
3	Installation Openhab	7
3.1	Openhab Betriebssystem aufsetzen	7
3.2	Openhab Configurieren	8
4	Installation Sprach Assistant	11
4.1	Topics	12
5	System	13
5.0.1	Zugangsdaten	13
5.0.2	Systemübersicht	13

1 Installation Aktor

Das Aktor-Bord ist die physikalische Schnittstelle zu den Elektronischen Geräten. Die Relais K1 bis K4 können 250 Volt AC und 10 Ampere schalten. An den 0-10 Volt Ausgängen darf einen maximalen Strom von 4 mA bezogen werden.

1.1 Anleitung einrichten

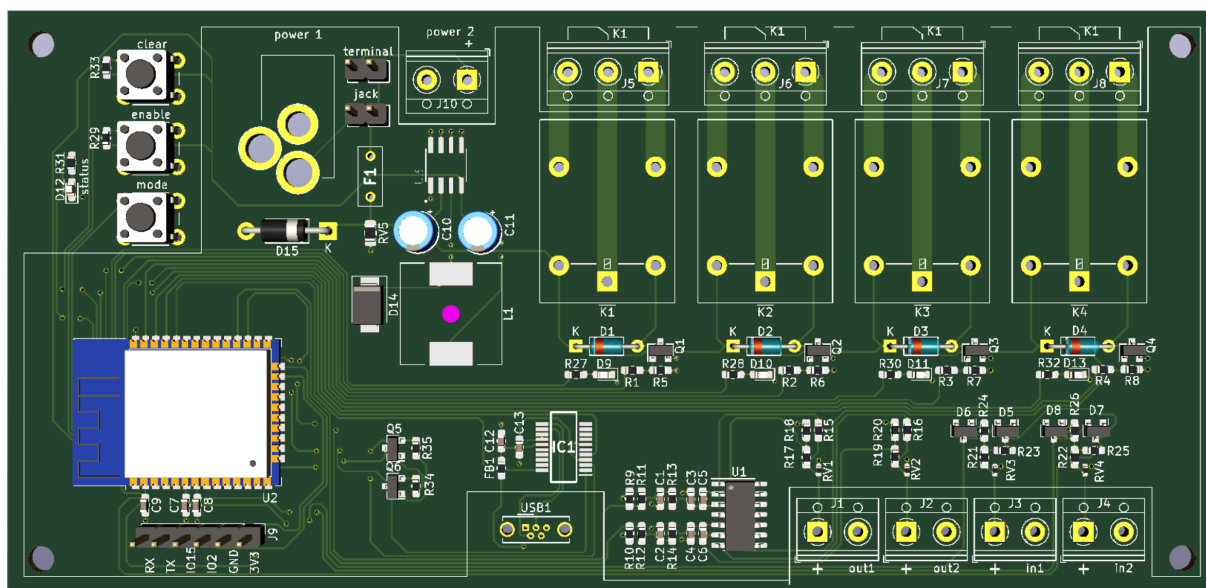


Abbildung 1.1: Aktor Bord

1. Das Gehäuse des Aktor-Bords, soll in eine Haupt- oder Unterverteilung eingebaut werden, wichtig ist zu beachten, dass sich dieser Standort innerhalb der eigenen WLAN-Reichweite befindet. Der Einbau ist im spannungslosen Zustand durchzuführen.
2. Verbraucher wie Leuchten, Beschattungs-Systeme oder Ventilationen können in Schliesser- oder Öffnerbetrieb an den Relais K1 bis K4 angeschlossen werden. Wichtig ist zu beachten, dass die maximale Spannung von 250 Volt AC und maximaler Strom von 10 Ampere nicht überschritten wird.
3. An den out1 und out2 Klemmen können Geräte mit einer 0-10 Volt Schnittstelle angeschlossen werden. Ein maximaler Strom von 4 mA darf nicht überschritten werden, damit die Spannung stabil bleibt.
4. An den In1 und In2 Klemmen können 0-10 Volt Sensoren angeschlossen werden. Werden diese Eingänge nicht verwendet, wird eine Spannung von 0.25 V ermittelt. Falls 0 V gewünscht ist, kann der '+' Eingang auf Ground bzw. dem zweiten zweiten Anschluss der Klemme angeschlossen werden.

- Als Energieversorgung kann an den Power Klemmen eine 24 V DC Quelle angeschlossen werden. Das Board kann auch mit einem 24VDC Hohlstecker verbunden werden. Der Jumper muss entsprechend gesteckt werden, 'jack' für Hohlstecker oder "terminal" wenn die Speisung über Klemmen erfolgt.

1.2 Inbetriebnahme

Sobald das Aktor-Bord mit Spannung versorgt wird, startet der Mikrocontroller ein Access-Point und das Config-Portal des Bords. Mit einem beliebigen Gerät, kann nach einem WLAN-Netzwerk gesucht werden. Das Netzwerk hat den Namen 'Aktor' gefolgt von der 10 Stelligen Chip-ID des Mikrocontrollers.

Aktor 3249694636

WiFiManager

Configure WiFi and MQTT

Configure WiFi (No Scan)

Info

Reset

Abbildung 1.2: Ansicht Configportal Frontseite

SSID
password
mqtt server
mqtt port
board location
save

[Scan WiFi](#)

Abbildung 1.3: Parameter Config

Mit der Funktion 'Scan WiFi' werden vorhanden Netzwerke angezeigt. Wird ein eigener MQTT Server installiert, kann an dieser Stelle die IP-Adresse angegeben werden, ansonsten der URL des öffentlichen MQTT-Broker. Der Port ist Default '1883' anzugeben. Die Eingabe bei 'board location' wird verwendet um MQTT-Topics zu generieren, sie muss also eindeutig sein und sollte keine Sonderzeichen beinhalten. Werden mehrere Aktor-Boards verbaut, unterscheiden sie sich an der 'bord location'. Mit der Taste 'save' werden die Eingaben gesichert und müssen bei einem erneuten Start nicht mehr eingegeben werden. Kann sich das Aktor Bord erfolgreich ins Lokale Netzwerk anmelden, blinkt die Status LED in einem regelmässigen Zyklus. Bei jeder Zustandsänderung der Statusleuchte wird eine Messung an den Eingängen In1 und In2 durchgeführt.

1.3 Programmierung

Verschiedene Eigenschaften können mit dem Entwicklertool zusätzlich verändert werden, wenn der Programmcode des Mikrocontrollers bearbeitet wird. In der nachfolgenden Tabelle sind Default-Konfigurationen enthalten.

Bezeichnung	Variable	Wert
Zeit Intervall Messungen	NUM_SEC	10
Allgemeiner MQTT-Topic	MQTT_SERIAL_PUB	'data/aktorboard/'
Messungen für Mean Wert ADC	I	100

Werden schnelle Reaktionszeiten vom System verlangt, von Befehlseingabe bis zur Ausführung, können Funktionen direkt im Programmcode eingebunden werden. So kann eine Reaktionszeit

von 300 ms erreicht werden. In diesem Fall wird in der Funktion 'callback()' definiert, wenn topic 'data/sensorboard/location/s1' empfangen wird, wird Relais1 schalten.

1.4 Netzwerkeinstellungen ändern

Soll sich das Aktor-Board in ein anderes Netzwerk anmelden, gibt es zwei verschiedene Möglichkeiten die Einstellungen zu ändern. Falls während des Startvorgangs die 'Mode' Taste betätigt wird, öffnet der Mikrocontroller das Config-Portal. Das Config-Portal wird ebenfalls geöffnet, wenn das einst eingetragene Netzwerk nicht mehr gefunden wird und keine Verbindung hergestellt werden kann.

1.5 System Tasten

Mit der Taste 'enable' wird manuell ein Neustart durchgeführt.

Mit der Taste 'mode' wird das Config-Portal eröffnet die Taste muss beim Startvorgang mindestens 5 s Betätigt werden.

Die Taste 'clear' hat auf dem Prototyp Aktor-Board keine Funktion.

1.6 Topics

In der Nachfolgenden Tabelle sind die automatisch generierten Topics, welche weiter in Openhab verwendet werden, um festzulegen welches Relais bei welchem Befehl schaltet. Als 'location' wurde im Config-Portal in diesem Fall 'location' eingetragen.

Topic	Funktion
data/aktorboard/location/K1	Schaltet Relais 1
data/aktorboard/location/K2	Schaltet Relais 2
data/aktorboard/location/K3	Schaltet Relais 3
data/aktorboard/location/K4	Schaltet Relais 4
data/aktorboard/location/A1	Schaltet 0-10V Output 1
data/aktorboard/location/A2	Schaltet 0-10V Output 2
data/aktorboard/location/E1	Publish 0-10V Input 1
data/aktorboard/location/E2	Publish 0-10V Input 2

Tabelle 1.1: Generierte MQTT-Topics Aktor-Bord

Mit den Topics werden die verschiedenen Anwendungen unterschieden, wobei sich der Zustand mit der Payload unterscheidet. Bei den Relais wird zwischen 'ON' und 'OFF' unterschieden. Bei den 0-10 Volt Inputs so wie Outputs enthält die Payload den entsprechenden Wert.

2 Installation Sensor

Das Sensor-Bord ist die physikalische Schnittstelle, wo Aktionen ausgelöst werden und anschliessend von Openhab empfangen und verarbeitet werden.

2.1 Anleitung einrichten

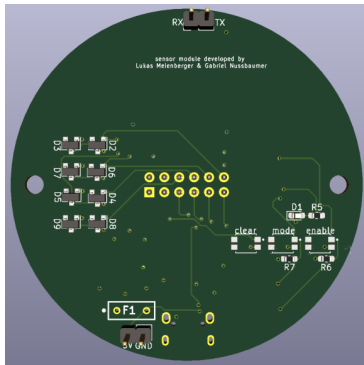


Abbildung 2.1: Sensor-Bord Rückseite

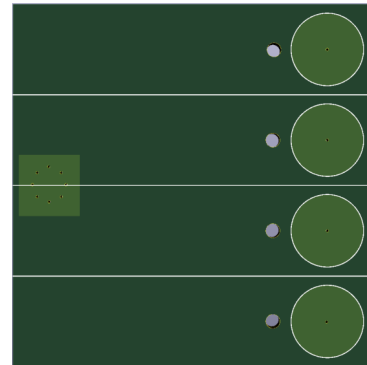


Abbildung 2.2: Front-print



Abbildung 2.3: Spannungsversorgung Rückseite

1. Die Installation ist in einem spannungslosen Zustand durchzuführen.
2. An der Rückseite von der Spannungsversorgung wird die Netzspannung 230 VAC angeschlossen siehe Abbildung 2.3.
3. Das Sensor-Bord 2.1 wird mit der Spannungsversorgung 2.3 und wird mit einem Standard Feller Gr-1 Befestigungsrahmen in eine Vorgesehene Einbaushalteröffnung eingebaut.
4. Der Frontplatte 2.2 wird mit dem Abdeckrahmen Gr-1 auf das Sensor-Bord 2.1 gesteckt. Die Beschriftung der Pins, GND zu GND ist dringend zu Beachten beim zusammenstecken, es kann auch auf den Markierungspunkt beim Stecker geachtet werden.

2.2 Inbetriebnahme

Sobald das Sensor-Board mit Spannung versorgt wird, startet das Config-Portal des Bords. Mit einem beliebigen Gerät kann im nach einem WLAN-Netzwerk gesucht werden. Das Netzwerk hat den Namen 'Sensor' gefolgt von der 10 Stelligen Chip-ID des Mikrocontrollers.

Mit der Funktion 'Scan WiFi' werden vorhandenen Netzwerke angezeigt. Wird ein eigener MQTT Server installiert, kann an dieser Stelle die IP-Adresse angegeben werden und der Port ist Default '1883' anzugeben. Die Eingabe bei 'board location' wird verwendet um MQTT-Topics zu generieren, sie muss eindeutig sein. Werden mehrere Sensor-Bords verbaut, unterscheiden sie sich an der 'bord location'. Mit dem Button 'save' werden die Eingaben gesichert und müssen bei einem erneuten Start nicht mehr eingegeben werden. Kann sich das Sensor-Board erfolgreich ins Lokale Netzwerk anmelden, blinkt die Status LED in einem regelmässigen Zyklus. Bei jeder Zustandsänderung der Statusleuchte, wird eine Temperaturmessung mit dem NTC-Widerstand durchgeführt.

2.3 Programmierung

Verschiedene Eigenschaften können mit dem Entwicklertool zusätzlich verändert werden, falls der Programmcode des Mikrocontrollers bearbeitet wird. In der nachfolgenden Tabelle sind die wichtigsten Default-Konfigurationen enthalten.

Bezeichnung	Variable	Wert
Zeit Intervall Messungen	NUM_SEC	10
Allgemeiner MQTT-Topic	MQTT_SERIAL_PUB	data/sensorboard/
Offset Temperatur Messung	b_temp	0.1369856
Steigung Temperatur Messung	m_temp	0.0008155002
Treshold Trigger Touchsensor	tresh	15
Messungen für Mean Wert Temperatur	I	10
Messungen für Mean Wert Touchwert	count	15

2.4 Netzwerkeinstellungen ändern

Soll sich das Sensor-Board in ein anderes Netzwerk anmelden, gibt es zwei verschiedene Möglichkeiten die Einstellungen zu ändern. Wenn während des Startvorgangs die Clear-Taste 5s betätigt wird, eröffnet der Mikrocontroller das Config-Portal. Das Config-Portal wird ebenfalls geöffnet, wenn das einst eingetragene Netzwerk nicht mehr gefunden wird und keine Verbindung hergestellt werden kann.

2.5 System Tasten

Mit der Taste 'enable' wird manuell ein Neustart durchgeführt.

Mit der Taste 'clear' wird während dem Startvorgang das Config-Portal eröffnet.

Die Taste 'mode' hat auf dem Prototyp Sensor-Bord keine Funktion.

2.6 Topics

In der Nachfolgenden Tabelle sind die Automatisch generierten Topics, welche weiter in Openhab verwendet werden. Als location wurde im Config-Portal 'location' eingetragen.

Topic	Funktion
data/sensorboard/location/S1	Taster 1 wurde betätigt
data/sensorboard/location/S2	Taster 2 wurde betätigt
data/sensorboard/location/S3	Taster 3 wurde betätigt
data/sensorboard/location/S4	Taster 4 wurde betätigt
data/sensorboard/location/S0	Gemessene Raumtemperatur

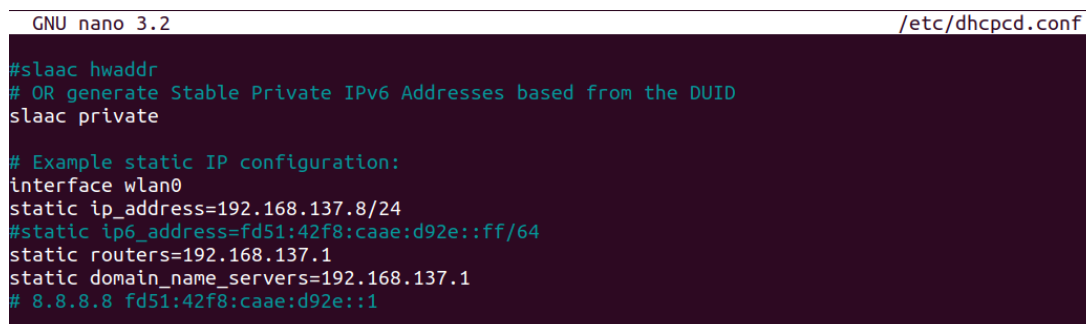
Tabelle 2.1: MQTT-Topics

3 Installation Openhab

Openhab ist die Schnittstelle, welche in diesem Projekt die verschiedenen Geräte mit einander verknüpft. Die Opensource Software wird auf dem Raspberrypi 4 mit 2 GB RAM installiert und in Betrieb genommen.

3.1 Openhab Betriebssystem aufsetzen

1. Betriebssystem Download von der offiziellen Openhab Webseite [1], anschliessend Image auf Micro SD-Card mit Balena Etcher flashen.
2. Auf SD-Card in Order 'openhabian.conf' File mit lokalen WLAN Namen 'ssid' und Passwort 'psk' erweitern.
3. SD-Card auswerfen, Raspi mit SD-Card ausstatten und in Betrieb nehmen. Ein neuer Teilnehmer wird im eigenen Netzwerk sichtbar, Gerätenahme 'openhab'.
4. Im Browser kann das Installierte Betriebssystem unter der IP Adresse und mit dem Port 8080 erreicht werden: `http://192.168.137.51:8080/` Packet Standard Installieren. Dauert 20 min.
5. Mit Konsole Openhabian, bearbeiten : `sudo ssh openhabian@192.168.137.24` .
Default Passwort 'openhabian'
6. Statische IP-Adresse vergeben: File `dhcpcd.conf` öffnen `sudo nano /etc/dhcpcd.conf` . Interface 'eth0' zu 'wlan0' ändern. Folgende Zeilen gemäss Abbildung 3.1 anpassen.



```

GNU nano 3.2 /etc/dhcpcd.conf

#slaac hwaddr
# OR generate Stable Private IPv6 Addresses based from the DUID
slaac private

# Example static IP configuration:
interface wlan0
static ip_address=192.168.137.8/24
#static ip6_address=fd51:42f8:caae:d92e::ff/64
static routers=192.168.137.1
static domain_name_servers=192.168.137.1
# 8.8.8.8 fd51:42f8:caae:d92e::1

```

Abbildung 3.1: File: dhcpcd.conf

7. Um Openhabian weiterhin mit Konsole zu erreichen müssen sich die Verschiedenen Geräten, Notebook und Raspi im gleichen IP-Adressraum befinden, eine Änderung kann mittels Adapteroptionen, Eigenschaften Internetprotokoll IPV4, folgende Adresse vergeben, durchgeführt werden.
8. Um Konfigurationen mit Visual Studio Code zu Bearbeiten wird der Ordner openHab-conf in Netzlaufwerk vom Notebook verbunden. Eingabe Ordner: '\\192.168.137.8 \openHAB-conf' verbinden, mit anderen Anmeldedaten anmelden, Username:'openhabian', Passwort: 'openhabian' wählen.
9. In Visual Studio Code Addon 'Openhab' installieren ,Ordner öffnen openHab-conf wählen. In Einstellungen, Erweiterungen, OpenHab Configuration, Host in 'settings.json' bearbeiten wählen.
`openhab.host: '192.168.137.8', 'git.autofetch': true` eingeben. File ist in der Projekt Dokumentation vorhanden. Anschliessend ist ein Neustart von Visual Studio Code erforderlich.

3.2 Openhab Configurieren

1. Openhab Bindings (Schnittstellen) einrichten in Visual Studio Code. In der Ordnerstruktur wird der Ordner 'services' geöffnet und das File 'addons.cfg' bearbeitet. 'binding =' wird ein Kommentiert, das Binding 'mqtt' wird hinzugefügt. Ebenso wird 'misc' ein Kommentiert und 'mqttbroker, openhabcloud' hinzu gefügt gemäss Abbildung 3.2, anschliessend ist ein Reboot des Raspis notwendig.

```
# A comma-separated list of bindings to install (e.g. "binding = sonos,knx,zwave")
binding = mqtt

# A comma-separated list of UIs to install (e.g. "ui = basic,paper")
#ui =

# A comma-separated list of persistence services to install (e.g. "persistence = rr
#persistence =

# A comma-separated list of actions to install (e.g. "action = mail,pushover")
#action =

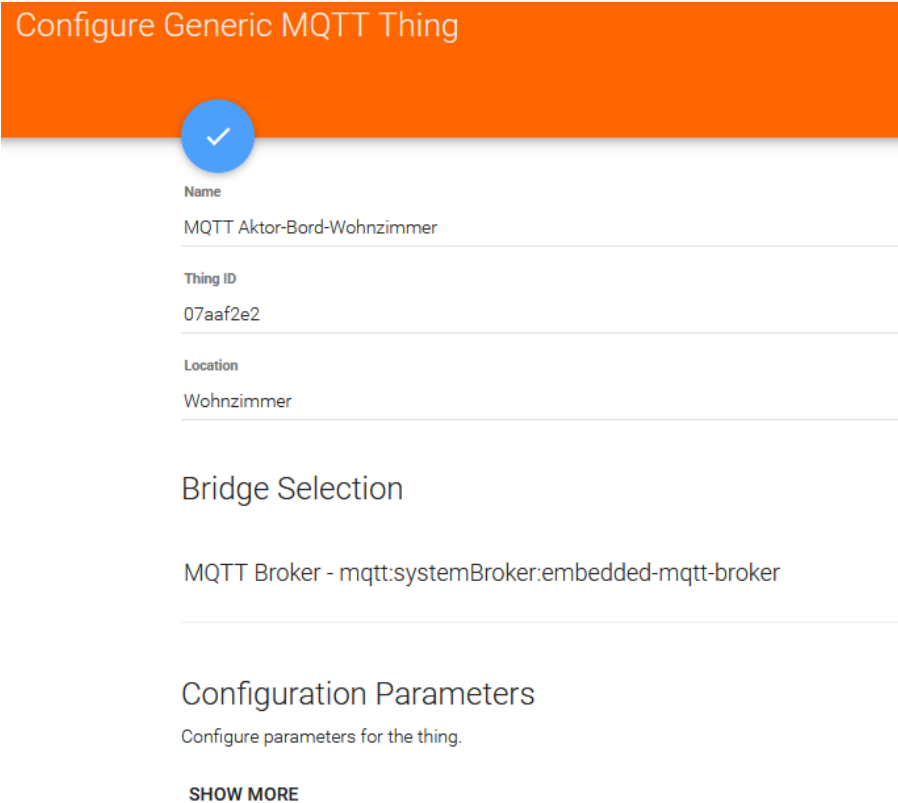
# A comma-separated list of transformation services to install (e.g. "transformatio
#transformation =

# A comma-separated list of voice services to install (e.g. "voice = marytts,freett
#voice =


# A comma-separated list of miscellaneous services to install (e.g. "misc = myopenh
misc = mqttbroker, openhabcloud
```

Abbildung 3.2: File: addons.cfg

2. Openhab mit Things (Geräten) verbinden. Things werden im Webinterface hinzugefügt und Openhab ist unter URL <IP-Adresse>:8080 erreichbar. 'Button Paper UI' wählen, in der Inbox befindet sich der MQTT Broker, welcher mit einem Klick auf den Bestätigungsbutton hinzugefügt wird.
3. Mit dem Menüpunkt 'Configuration', 'Thing' ist der MQTT Broker sichtbar, mit den blauen Add-Button werden weitere MQTT-Things hinzugefügt, 'ADD MANUALLY', 'Generic MQTT Thing' wählen, Parameter gemäss Abbildung 3.3 wählen. Dieser Vorgang wird für jedes Bord durchgeführt ein Bord wird als Gerät also als Thing bezeichnet.



Configure Generic MQTT Thing



Name
MQTT Aktor-Bord-Wohnzimmer

Thing ID
07aaf2e2

Location
Wohnzimmer

Bridge Selection

MQTT Broker - mqtt:systemBroker:embedded-mqtt-broker

Configuration Parameters

Configure parameters for the thing.

SHOW MORE

Abbildung 3.3: Configure MQTT-Thing

- In 'Configuration', 'Thing' ist das nun erstellte Gerät aufgetaucht, dies wählen und Button 'Channels add' wählen, Channel type 'On/OFF Switch' eingeben, Channel id beliebig eingeben, darf keine Leerschläge enthalten, Entsprechende MQTT Configurationen eingeben, Topic siehe Tabelle 1.1. Für jede Funktion welche in einem Thing (Gerät) enthalten sind, sprich pro MQTT-Topic wird ein Channel gemäss Abbildung 3.4 erstellt.

Add channel

Please select a channel type for the new channel:

Channel type

On/Off Switch

Channel id

LichtWohnzimmer

Label

Licht Wohnzimmer

Channel configuration

Others

MQTT State Topic

An MQTT topic that this thing will subscribe to, to receive the state. This can be left empty, the channel will be stateless command-only channel.

Custom On/Open Value

ON

A number (like 1, 10) or a string (like 'enabled') that is additionally recognised as on/open state. You can use this parameter for a second keyword, next to ON (OPEN respectively on a Contact).

MQTT Command Topic

data/aktorboard/location/K1

An MQTT topic that this thing will send a command to. If not set, this will be a read-only switch.

Custom Off/Closed Value

OFF

A number (like 0, -10) or a string (like 'disabled') that is additionally recognised as off/closed state. You can use this parameter for a second keyword, next to OFF (CLOSED respectively on a Contact).

SHOW MORE

CANCEL

SAVE

Abbildung 3.4: Configure Channel

5. In Visual Studio Code in Ordner 'items' neue Datei erstellen mit Endung .items und öffnen, wird nun das Addon Openhab geöffnet sind zwei Things ersichtlich. Im Tropicdown Menü ist das zuvor erstellte item 'Licht Wohnzimmer' ersichtlich und kann in .items Datei importiert werden.
6. Weitere Dateien werden angelegt und Konfiguriert. Für die Benutzeroberfläche wird im Ordner 'Sitemaps' eine Datei mit Endung .sitemap angelegt ebenso wird für alle automatischen Vorgängen im Ordner 'rules' eine Datei mit Endung .rules angelegt. Informationen um Items, Rules, oder Sitemaps zu initialisieren sind unter Dokumentation [2] vorhanden.
7. Fils aus der Projekt-Dokumentation können übernommen werden, dabei muss der Channel Link und Item-Bezeichnungen angepasst werden.

4 Installation Sprach Assistant

Der Home Assistant ist die Schnittstelle, welche den Google Assistant mit Openhab verbindet. Die Kommunikation zwischen Homeassistant und Openhab findet über MQTT statt. Die Homeassistant Software wird auf ein weiteres Raspberry Pi installiert. Die Verbindung zwischen Home Assistant und Google Assistant basiert auf einer Öffentlichen Cloud gemäss Abbildung 5.1.

1. Download des Images für die Installation auf dem eigenen Raspi von der offiziellen Home Assistant [3] Webseite.
2. Image auf Micro SD-Card mit balenaEtcher [4] schreiben.
3. Ein USB-Stick, FAT32 formatieren und ihn als 'CONFIG' benennen, ein Ordner mit dem Namen 'network' erstellen, darin eine Text Datei erstellen, in der die Wlan Konfigurationen enthalten sind, wie im Beispiel [5] beschrieben oder in der Projekt-Dokumentation enthalten ist.
4. USB-Stick und Micro SD-Card in Raspi einsetzen und in Betrieb nehmen, dauert ca 20 min. Der USB Stick ist nur bei der ersten Inbetriebnahme notwendig um Netzkonfigurationen zu Initialisieren.
5. Im Browser wird mit dem URL: `http://homeassistant.local:8123` die Benutzeroberfläche des Homeassistenten angezeigt.
6. Durch anwählen des Benutzernamen öffnet sich das Profil, 'Erweiterter Modus' wählen.
7. Menüpunkt Supervisor anwählen und in Add-on store 'File editor' installieren, 'Start on boot' und 'show in sidebar' wählen
8. Mit File Editor Datei '/config/configuration.yaml' bearbeiten: Mqtt-Message Publish ermöglichen durch hinzufügen von Broker IP-Adresse und Port, siehe Abbildung 4.1.
9. Nach jeder Änderung mit dem "File editor ist ein Neustart des Server notwendig, dies kann bei 'Einstellungen', 'Serversteuerung', 'Serververwaltung' Button 'neu Starten' durchgeführt werden.

```

/config/configuration.yaml
1 |
2 | # Configure a default setup of Home Assistant (frontend, api, etc)
3 | default_config:
4 |
5 | # Text to speech
6 | tts:
7 |   - platform: google_translate
8 |
9 | # Example configuration.yaml entry to enable the cloud component
10 | cloud:
11 |
12 |   group: !include groups.yaml
13 |   automation: !include automations.yaml
14 |   script: !include scripts.yaml
15 |   scene: !include scenes.yaml
16 |
17 | mqtt:
18 |   - broker: 192.168.137.8
19 |     port: 1883
20 |
21 | # Example configuration.yaml entry
22 | switch:
23 |   - platform: mqtt
24 |     name: Licht Wohnzimmer
25 |     command_topic: "data/sensorboard/homeassisant/S1"
26 |     icon: mdi:lightbulb
27 |
28 |   - platform: mqtt
29 |     name: Licht Esszimmer
30 |     command_topic: "data/sensorboard/homeassisant/S2"
31 |     icon: mdi:lightbulb
32 |
33 |   - platform: mqtt
34 |     name: Licht Büro
35 |     command_topic: "data/sensorboard/homeassisant/S3"
36 |     icon: mdi:lightbulb
37 |
38 |
39 |   - platform: mqtt
40 |     name: Licht Aussen
41 |     command_topic: "data/sensorboard/homeassisant/S4"
42 |     icon: mdi:lightbulb

```

Abbildung 4.1: Configfile Home Assistant

10. Hinzu fügen von Schalter, welche mit Sprachbefehl Mqtt-Message generieren. Der Name des Schalters ist ebenso die Bezeichnung für den Sprachbefehl.
11. In den 'Einstellungen' Home Assistant Cloud anmelden, Konto erstellen. Mit Google Home App auf Smartphone mit 'Gerät verbinden', Home Assistant wählen und mit Cloud Benutzerkonto anmelden.

4.1 Topics

In Nachfolgenden Tabelle 4.1 sind die vom Home Assistant generierten Topics, welche weiter in Openhab verwendet werden.

Topic	Funktion
data/sensorboard/homeassisant/S1	Sprachbefehl von Google Assistant empfangen
data/sensorboard/homeassisant/S2	Sprachbefehl von Google Assistant empfangen
data/sensorboard/homeassisant/S3	Sprachbefehl von Google Assistant empfangen
data/sensorboard/homeassisant/S4	Sprachbefehl von Google Assistant empfangen

Tabelle 4.1: MQTT-Topics Home Assistant

5 System

5.0.1 Zugangsdaten

In der nachfolgenden Tabelle sind die Zugangsdaten, welche in diesem Projekt verwendet wurden.

Dienst	Benutzername	Passwort	IP-Adresse
WLAN	LA7722	490f2490	
Openhab	openhabian	openhabian	192.168.137.8
Home Assistant	uuserdue	uuserdue	
Cloud Nabu Casa	Uuserdue@gmail.com	uuserdue	

5.0.2 Systemübersicht

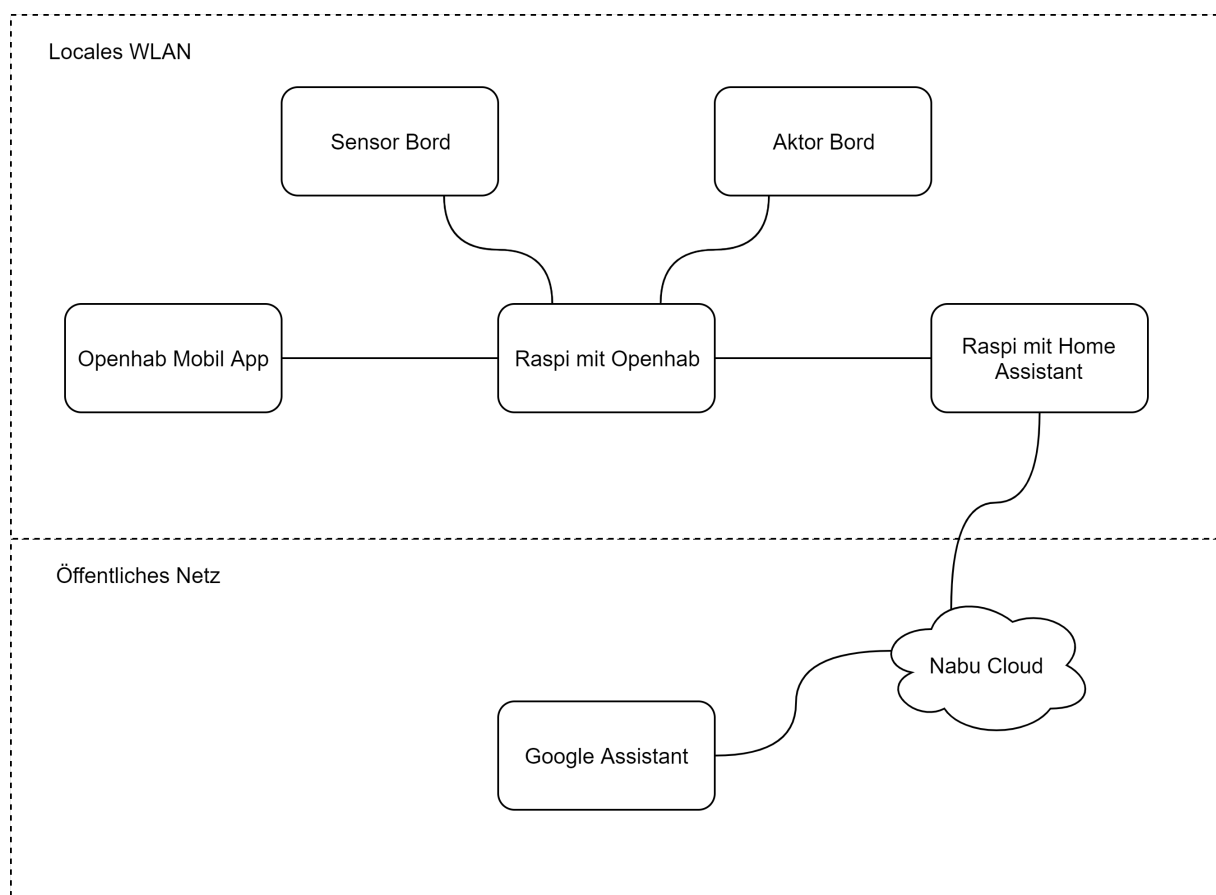


Abbildung 5.1: Systemübersicht

In der Systemübersicht ist zu erkennen, dass das gesamte System im lokalen Netz funktioniert mit der Ausnahme vom Sprachassistent. Dies erhöht die Zuverlässigkeit und die Sicherheit der Smart-Home Lösung.

Literatur

- [1] (). Download openHAB, [Online] Available: <https://www.openhab.org/download/> (Abrufdatum 10. Aug. 2020).
- [2] (). Introduction, [Online] Available: <https://www.openhab.org/docs/> (Abrufdatum 6. Aug. 2020).
- [3] H. Assistant. (). Installing Home Assistant, [Online] Available: <https://www.home-assistant.io/hassio/installation/> (Abrufdatum 11. Aug. 2020).
- [4] (). balenaEtcher - Home, [Online] Available: <https://www.balena.io> (Abrufdatum 23. Dez. 2019).
- [5] (). Home-assistant/operating-system, [Online] Available: <https://github.com/home-assistant/operating-system> (Abrufdatum 11. Aug. 2020).