Plugin-Daten	
Autor	Michael Schlenstedt
Logo	
Status	STABLE
Version	1.0.2
Min. LB Version	2.2.2
Release Download	https://github.com/mschlenstedt/LoxBerry-Plugin-MultiIO/archive/refs/tags/LoxBerry-Plugin-MultiIO-1.0.2.zip
Beschreibung	Das Plugin kann verschiedene Hardware-Module als Ein- und Ausgänge am LoxBerry konfigurieren sowie einige Sensoren einbinden.
Sprachen	EN
Diskussion	https://www.loxforum.com/forum/projektforen/loxberry/plugins/395534-plugin-multi-io-plugin-mehr-als-512-ein-ausg%C3%A4nge-sensoren-am-loxberry

Multi-IO

Version History...

https://github.com/mschlenstedt/LoxBerry-Plugin-MultiIO/releases

---

Download

Direkter Download-Link: Siehe *Tabelle oben*

Letzter Entwicklungsstand im Repo: https://github.com/mschlenstedt/LoxBerry-Plugin-MultiIO

Funktion des Plugins

Das Plugin kann mit Hilfe der Software MQTT-IO (https://github.com/flyte/mqtt-io) verschiedene Hardware-Module am LoxBerry als Ein- und Ausgänge oder als Sensoren nutzen. Natürlich werden dabei die nativen GPIO-Ports des Raspberry unterstützt, aber auch IO-Expander Module wie das PCF8574 bzw. PCF8575 oder das MCP23017. Mit deren Hilfe können am Raspberry mehr als 512(!) Ein oder Ausgänge bereitgestellt werden. Des weiteren können verschiedene bekannte Sensoren für den Raspberry, z. B. ADS1x15 analog to digital converters, DHT11/DHT22/AM2302 oder der MCP3008 analog to digital converter integriert werden. Eine vollständige Liste der unterstützten Hardware findet sich auf der Projekt-Homepage: https://github.com/flyte/mqtt-io

Das Plugin setzt ein installiertes MQTT Gateway Plugin voraus (bis LoxBerry 2.2.x), im LoxBerry 3.0.x ist dieses bereits integriert und wird benutzt.

Über einen Watchdog wird die Funktion des Plugins permanent überwacht. Bei Problemen wird versucht das Plugin neu zu starten.

Hardware

Für einige Hardware haben wir spezielle Informationen auf einer eigenen Unterseite

zusammengefasst. Die Links findet ihr in den Unterkapiteln weiter unten.

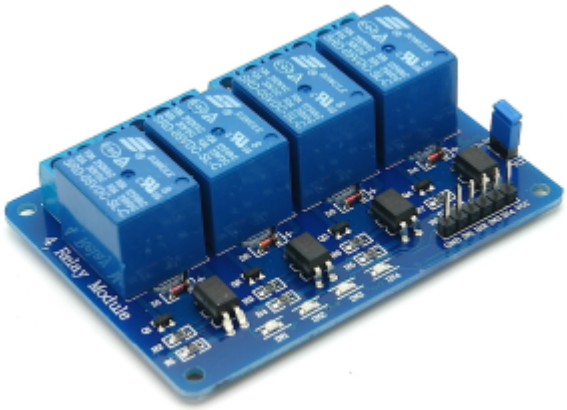
Separater I2C Bus

Nutzt ihr GPIO Module und/oder Sensoren über den I2C-Bus und verwendet ihr parallel noch andere Plugins, die Hardware am I2C Bus nutzen, empfehle ich Euch dieses Plugin auf einem separaten Bus laufen zu lassen. Auch wenn ihr Module mit identischer Adresse mehrfach verwenden wollt, müsst ihr diese auf separaten Bussen betreiben. Wie man mehrere I2C Busse auf dem Raspberry aktiviert, könnt ihr hier nachlesen: Mehrere I2C Busse an einem Raspberry

Schalten von bis zu 30V DC oder bis zu 230V AC

Spannungen von 230V sind **tödlich**! Installationen dürfen ausschließlich von Elektrofachkräften durchgeführt werden!

Um Ausgänge zu schalten benötigt man ein Relais Board. Üblicherweise kommen Relaisboards mit mechanischen Relais zum Einsatz. Diese gibt es üblicherweise mit 2, 3, 4, 8 oder 16 Relais. Die Relais werden dabei mit sogenannten Optokopplern an den Raspberry oder sonstigen Mikrocontrollern angeschlossen. Die Boards sind sehr flexibel - üblicherweise können sie rein von sich aus sowohl Netzspannung bis 230V mit 10A schalten (Spezifikation beachten!) als auch Niederspannung bis 30V DC schalten. Allerdings verursachen sie teilweise erhebliche Störungen am Raspberry/Mikrocontroller beim Schalten von 230V. Wenn man induktive Lasten schaltet (z. B. eine Spule in einem Magnetventil) entstehen auch deutliche Störungen beim Schalten von 24V DC.



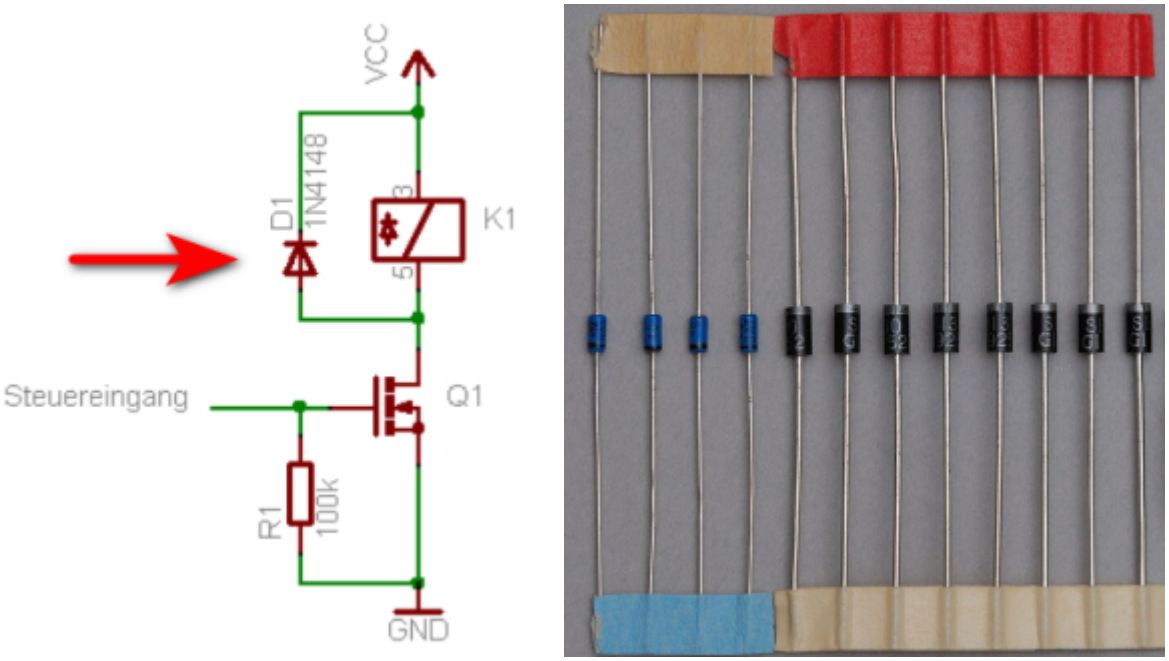
Beispiel eines Relais Boards, (c) berrybase.de

Beim Schalten hoher (induktiver) Lasten entstehen am Schaltkontakt des Relais sehr hohe Spannungen, die einen sogenannten Lichtbogen erzeugen. Diese Störungen schlagen bis auf die Spannungsversorgung und Mikrochips des Raspberry/Mikrocontrollers durch und führen hier zu erheblichen Störungen (bis hin zum Absturz des Raspberry). Schaltet man induktive Gleichstromlasten (Spule eines Magnetventils zum Beispiel) ab, bricht deren Magnetfeld zusammen. Die im Magnetfeld gespeicherte Energie kann nicht einfach verschwinden. Damit wird die Induktivität zur Energiequelle, welche sehr hohe Spannungen erzeugen kann (Prinzip der Zündspule). Ich empfehle daher dringend (aus eigener Erfahrung) diese Relais Boards nicht zum Schalten von 230V zu

verwenden (es gibt Alternativen - siehe unten) oder zu entstören (nächster Abschnitt). Beim Schalten von Gleichstrom muss häufig nicht entstört werden (wenn keine induktiven Lasten geschaltet werden). Ansonsten empfehle ich eine Schottky-Diode (Freilaufdiode) zu verbauen (siehe weiter unten). Hintergrund dazu hier:
https://www.mikrocontroller.net/articles/Relais_mit_Logik_ansteuern#Entst%C3%B6rung

Schalten von induktiven Lasten mit 30V DC - Mit Standardrelaisboard

Hier benötigt man eine Schottky-Diode (Freilaufdiode), die parallel zur induktiven Last verbaut wird - und zwar für die Betriebsspannung in Sperrrichtung. Auf dem folgenden Beispiel-Bild ist die Freilaufdiode mit "D1" bezeichnet (Hinweis: Auf der Diode ist ein Strich aufgedruckt, dieser entspricht dem "Strich" auf dem Symbol im Schaltplan):

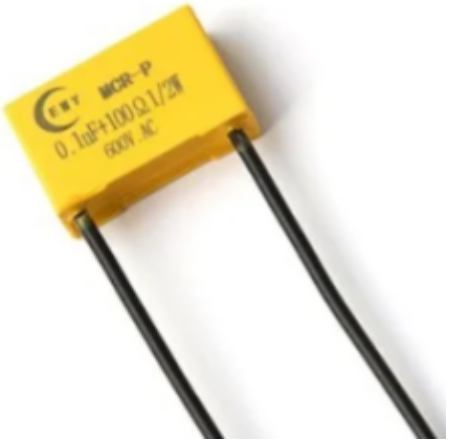


Schottky-/Freilaufdiode, (c) mikrocontroller.net und Wikipedia

Freilaufdioden sind ein Cent-Artikel, man muss nur darauf achten, dass sie die Betriebsspannung sicher sperren können. Bei 24V DC zum Beispiel eine Diode für 40V. Ich klemme sie immer direkt am Relaisausgang parallel zur eigentlich geschalteten Last.

Schalten von 230V AC - Mit Standardrelaisboard

Möchte man also dennoch ein solches Relais Board zum Schalten seiner 230V Verbraucher verwenden, kann (muss) man unbedingt ein RC-Glied (eine Kombination aus Widerstand und Kondensator) parallel zum Verbraucher schalten. Dieses RC-Glied nimmt die Störungen und Überspannungen während des Schaltvorgangs auf. So ein RC-Glied findet man auch unter dem Namen "Snubber". Wenn man Glück hat und die zu schaltenden Lasten sind nicht all zu groß, kann der Snubber die Störungen komplett beseitigen.

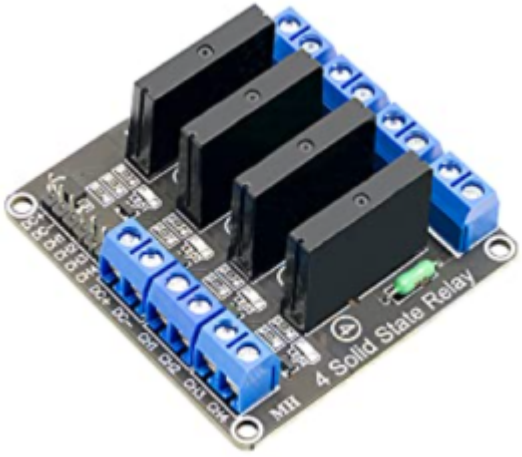


Snubber/RC-Glied, (c) Shelly

(Bessere) Lösung zum Schalten von 230V AC - Solid State Relais

Spannungen von 230V sind **tödlich**! Installationen dürfen ausschließlich von Elektrofachkräften durchgeführt werden!

Wer auf Nummer Sicher gehen möchte, der nutzt zum Schalten von 230V ausschließlich Solid State Relais (SSR). SSRs funktionieren genauso wie mechanische Relais, bestehen aber ausschließlich aus elektronischen Bauteilen und haben keinerlei Mechanik verbaut. Daher können beim Schalten auch keine Störungen durch Lichtbögen entstehen. Das Gute daran: Sie sind nicht sehr viel teurer als mechanische Relais. Üblicherweise gibt es die SSR genauso verbaut auf Boards wie die "normalen" Relaisboards auch. Aber Achtung! Diese Boards **können nur 2A schalten** und sie können **ausschließlich Wechselspannung** schalten! Schaut Euch die Spezifikation unbedingt an.



Beispiel eines Solid State Relais Boards, (c) az-delivery.de

Wer höhere Lasten als 2A schalten will, kann auf die folgende Bauart zurückgreifen. Diese SSR gibt es aber nur als Einzelvariante - sie kosten ca. 5 EUR/Stück. Ein bekannter von mir schaltet damit in

seiner Brauanlage eine Kochplatte mit 3500W ohne Probleme! Diese SSR gibt es üblicherweise in Varianten mit 25A oder 40A.

Beispiel eines Solid State Relais mit 40A, (c) botland.de

Konfiguration

Das Plugin bietet zwei Hauptfunktionen: **Inputs / Outputs** und **Sensoren**.

Wie der Name schon sagt können über "Inputs / Outputs" beliebig viele Ein- und Ausgänge konfiguriert werden. Diese müssen natürlich über entsprechende Hardwaremodule auch physikalisch vorhanden sein. Der Raspberry bringt schon einige Ein- und Ausgänge mit (GPIOs), weitere können über sogenannte GPIO Expander hinzugefügt werden.

Im Tab "Sensoren" können verschiedene gängige Sensoren wie z. B. Analog-Digital-Wandler, Ultraschallsensoren oder auch Luftgütesensoren an den Raspberry angebunden werden.

Input / Outputs

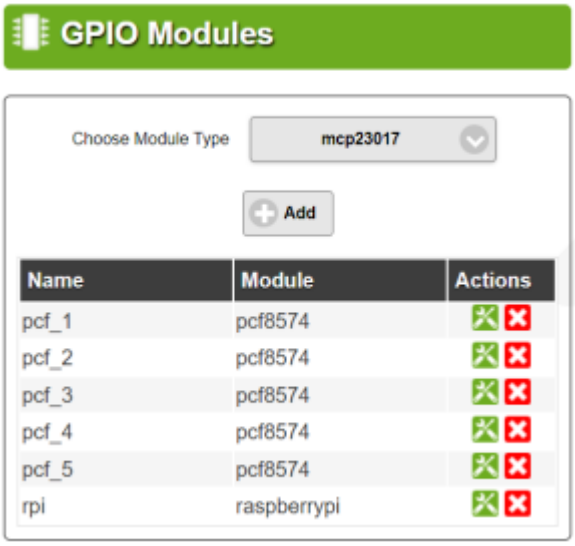
Zunächst muss ein entsprechendes GPIO Modul eingerichtet werden. Es definiert die Hardware, über die die Ein- und Ausgänge zur Verfügung gestellt werden. Ein Beispiel wäre die Raspberry GPIO Leiste und deren PINs. Das Plugin unterstützt aber noch zahlreiche weitere Möglichkeiten zusätzliche Ein- und Ausgänge zur Verfügung zu stellen. Dazu werden sogenannte GPIO Expander Module an den I2C Bus des Raspberry angeschlossen. Diese bieten meist 8 oder 16 weitere GPIOs. Durch Kombination mehrerer dieser Module und Verwendung von mehreren I2C-Bussen am Raspberry können quasi unendlich viele Ein- und Ausgänge am Raspberry angeschlossen werden.

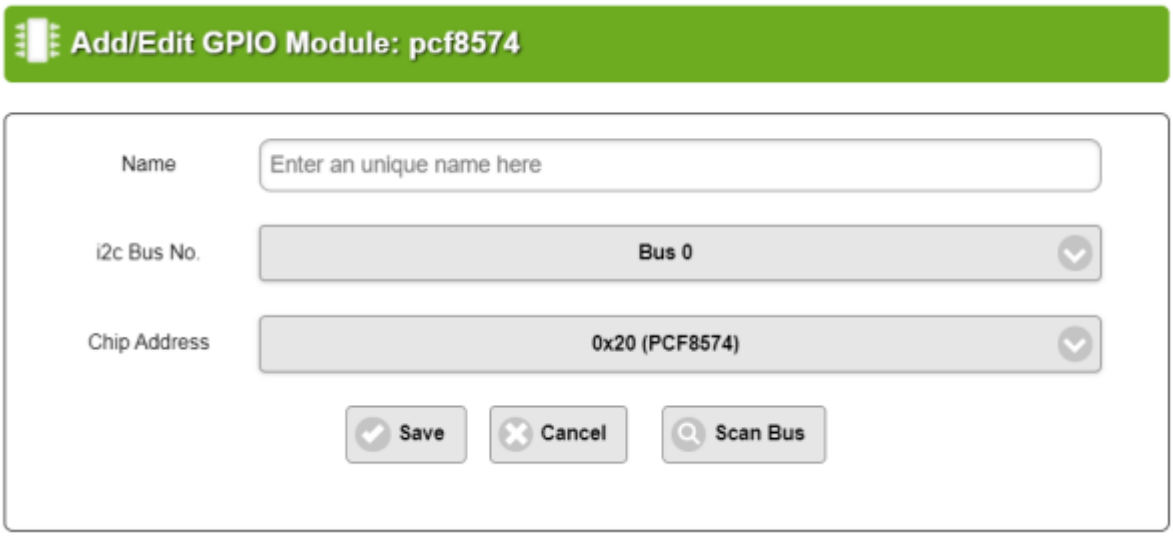
Anschließend können die am GPIO Modul zur Verfügung stehenden einzelnen Ein-/Ausgänge (PINs) konfiguriert werden.

GPIO Module

Ein neues GPIO Modul wird über den Add Button hinzugefügt. Es öffnet sich die Konfigurationsseite für das Modul, wo meist weitere Einstellungen getroffen werden müssen. Folgende GPIO-Module stehen im Plugin zur Verfügung - die Konfiguration findet ihr auf der jeweiligen Unterseite:

- MCP23017 Expander Modul
- PCF8574(A) Expander Modul
- PCF8575 Expander Modul
- Raspberry Modul

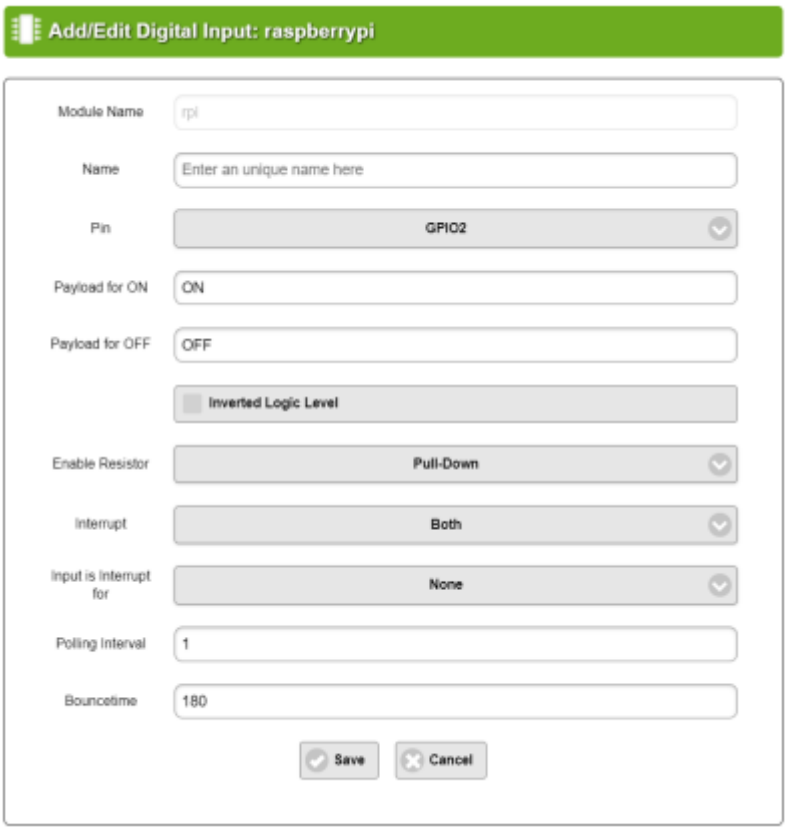


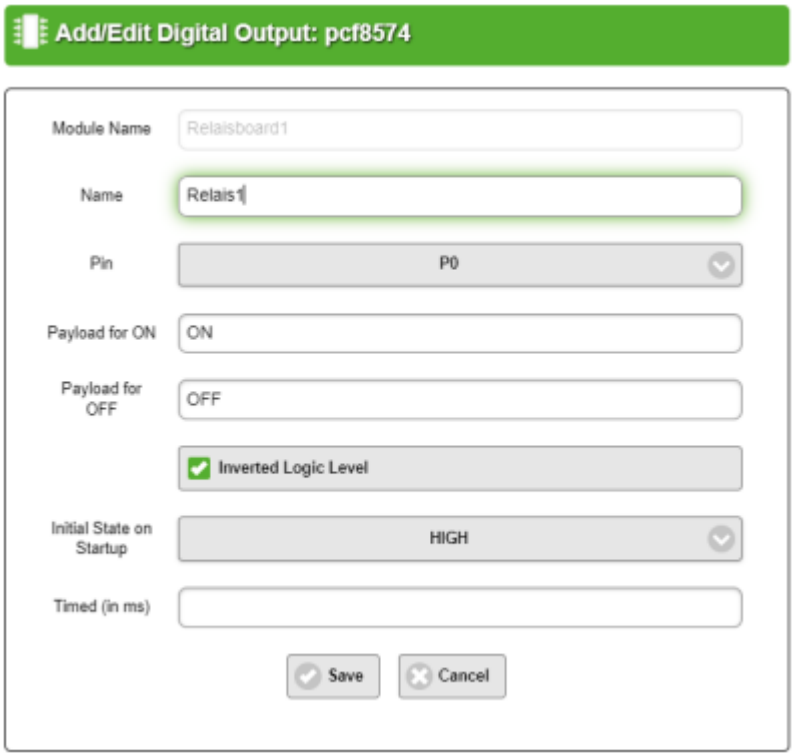


Ein- / Ausgänge

Nachdem ein GPIO Modul eingerichtet wurde, können für dieses Modul Ein- und Ausgänge konfiguriert werden. Jeder GPIO kann dabei entweder als Eingang oder als Ausgang eingerichtet werden. Die Einrichtung unterscheidet sich je nach GPIO Modul - weitere Details findet ihr auf den Unterseiten der jeweiligen GPIO Module:

- MCP23017 Expander Modul
- PCF8574(A) Expander Modul
- PCF8575 Expander Modul
- Raspberry Modul





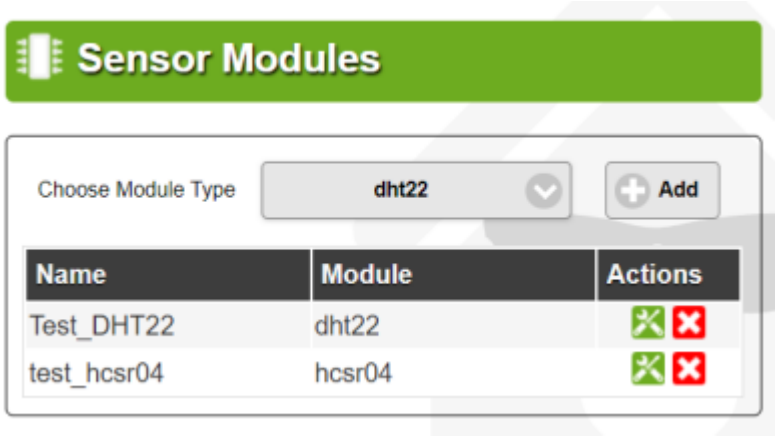
Sensors

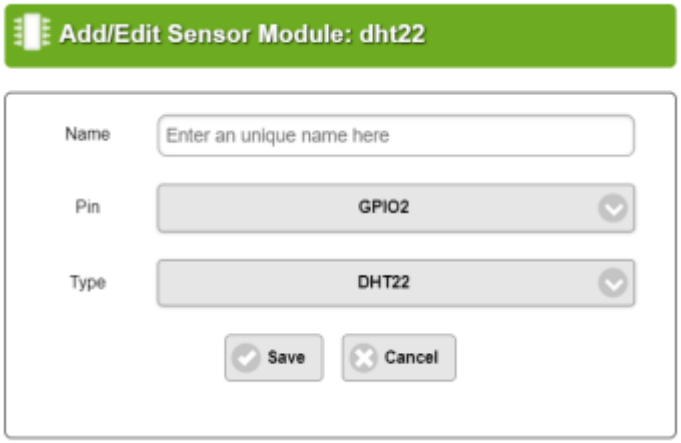
Bei den Sensoren folgt dass Plugin der Logik wie bei den Ein-/Ausgängen: Zunächst wird ein Sensor-Modul angelegt, welches die angeschlossene Hardware beschreibt und konfiguriert. Anschließend werden pro Sensormodul entsprechende Eingänge angelegt, die die Sensordaten beschreiben. Je nach Modul bzw. Sensor können das mehrere Eingänge (z. B. Luftfeuchtigkeit und Temperatur) sein oder auch nur ein Eingang.

Sensor Module

Ein neues Sensor Modul wird über den Add Button hinzugefügt. Es öffnet sich die Konfigurationsseite für das Modul, wo meist weitere Einstellungen getroffen werden müssen. Folgende Sensor-Module stehen im Plugin zur Verfügung - die Konfiguration findet ihr auf der jeweiligen Unterseite:

- ADS1x15 Analog zu Digital Konverter
- ADXL345 3-Achsen Accelerometer bis ±16g
- AHT20 Temperatur- und Humidity Sensor
- AS3935 Blitzsensor
- BH1750 Licht- / Helligkeitssensor
- BME280 Temperatur-, Pressure und Humidity Sensor
- BME680 Temperatur-, Pressure-, Humidity- und Gas-Sensor
- BMP085: Barometrischer Luftdruck/Temperatur/Höhen Sensor
- DHT11/DHT22/AM2302 Temperatur- und Humidity Sensor
- ENS160: Digitaler Multi-Gas Sensor
- FLOWSENSOR: Pulse Durchfluss-Sensoren
- FREQUENCYCOUNTER: Frequenzzähler
- HC-SR04 Ultraschall-Entfernungssensor
- INA219 Stromsensor
- LM75: Temperatur Sensor
- MCP3xxx: Analog Digital Wandler
- MHZ19: Infrarot CO2 Sensor
- PMS5003: Partikelsensor
- SHT4x: Temperatur- und Luftfeuchtigkeitssensoren
- TSL2561 Licht- / Helligkeitssensor
- VEML7700 Licht- / Helligkeitssensor

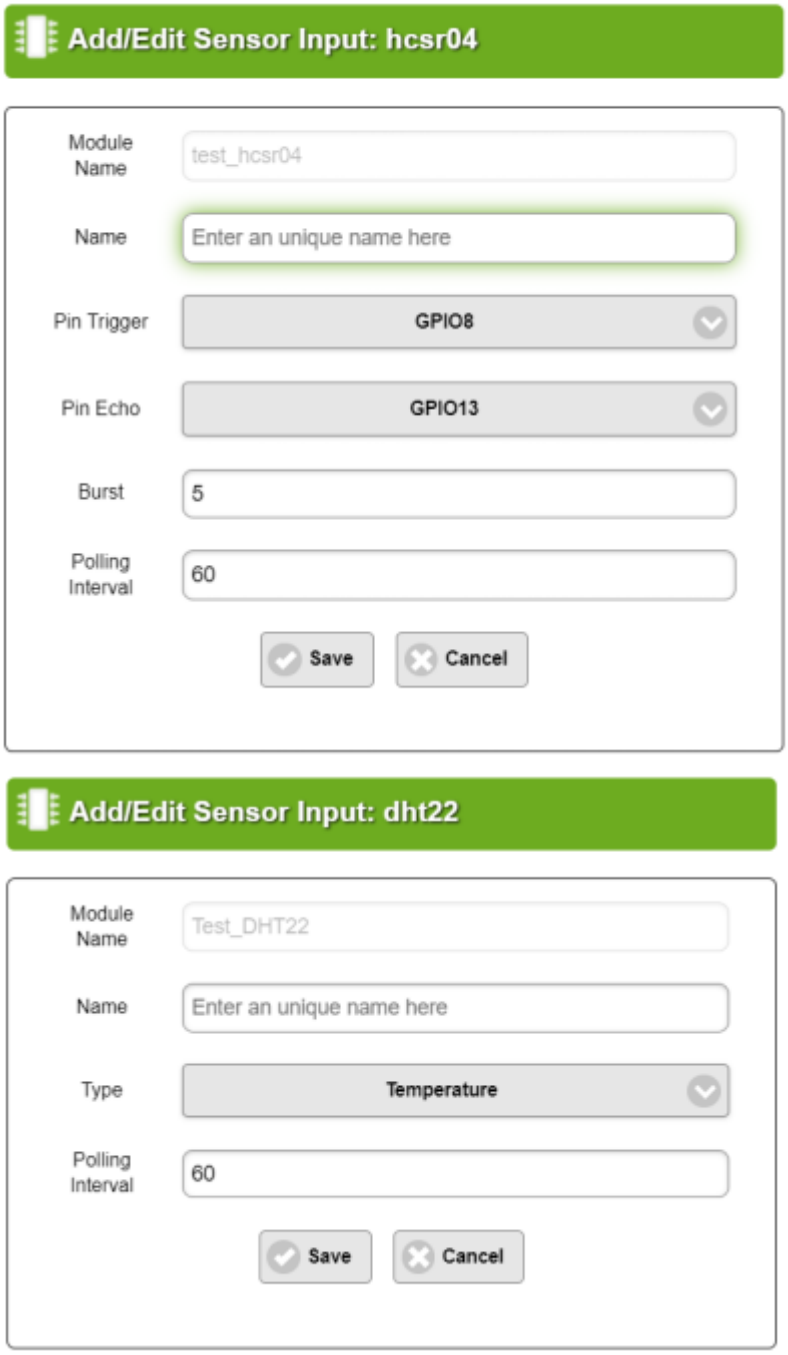




Sensor Eingänge

Nachdem ein Sensor-Modul eingerichtet wurde, können für dieses Modul Eingänge konfiguriert werden. Je nach Sensor kann es mehrere Eingänge geben (z. B. Temperatur und Luftfeuchtigkeit). Die Einrichtung unterscheidet sich je nach Sensor Modul - weitere Details findet ihr auf den Unterseiten der jeweiligen Sensor Module:

- ADS1x15 Analog zu Digital Konverter
- ADXL345 3-Achsen Accelerometer bis ±16g
- AHT20 Temperatur- und Humidity Sensor
- AS3935 Blitzsensor
- BH1750 Licht- / Helligkeitssensor
- BME280 Temperatur-, Pressure und Humidity Sensor
- BME680 Temperatur-, Pressure-, Humidity- und Gas-Sensor
- BMP085: Barometrischer Luftdruck/Temperatur/Höhen Sensor
- DHT11/DHT22/AM2302 Temperatur- und Humidity Sensor
- ENS160: Digitaler Multi-Gas Sensor
- FLOWSENSOR: Pulse Durchfluss-Sensoren
- FREQUENCYCOUNTER: Frequenzzähler
- HC-SR04 Ultraschall-Entfernungssensor
- INA219 Stromsensor
- LM75: Temperatur Sensor
- MCP3xxx: Analog Digital Wandler
- MHZ19: Infrarot CO2 Sensor
- PMS5003: Partikelsensor
- SHT4x: Temperatur- und Luftfeuchtigkeitssensoren
- TSL2561 Licht- / Helligkeitssensor
- VEML7700 Licht- / Helligkeitssensor



Roadmap

- [x] MQTT Settings (Topic) (zugewiesen an Michael Schlenstedt)
- [x] Alle Sensormodule integrieren (zugewiesen an Michael Schlenstedt)
- [ ] Deutsche Übersetzung (zugewiesen an Michael Schlenstedt)

Links

- Unterstütze Hardware und Sensoren: https://mqtt-io.app/#/?id=supported-hardware

- MQTT-IO Anleitung: https://mqtt-io.app/#/

Fragen stellen und Fehler melden

Im Loxforum in diesem
Thread: https://www.loxforum.com/forum/projektforen/loxberry/plugins/395534-plugin-multi-io-plugin-mehr-als-512-ein-ausg%C3%A4nge-sensoren-am-loxberry

From:
https://wiki.loxberry.de/ - **LoxBerry Wiki - BEYOND THE LIMITS**

Permanent link:
https://wiki.loxberry.de/plugins/multi_io/start?rev=1717228042

Last update: **2024/06/01 09:47**