

## LoRa Fernbedienung Receiver und Sender

LoRa (LongRange) ist eine Technik mit der Funksignale über eine lange Distanz gesendet werden können. Als Sendefrequenz können 433 Mhz, 868 Mhz oder 915 MHz verwendet werden. In Europa (Deutschland) ist nur 433 MHz erlaubt. Welche Frequenzen in anderen Staaten erlaubt sind, muss vorher eruiert werden.

Im Gegensatz zu normalen 433 MHz Sender Empfängern, die eine Reichweite von ca. 10-40 m besitzen, kann eine LoRa Funkstrecke durchaus mehrere 100 m überbrücken.

Im Folgenden wird in groben Zügen der Bau eines LoRa Senders und Empfängers beschrieben. Die Beschreibung setzt Kenntnisse in Elektronik und in der Programmierung von Arduinos voraus und ist nicht für Laien gedacht.

### **Bauteile (Geschätze Kosten ca. 130 €):**

#### **2x Adafruit Feather M0 RFM96 Lora Radio 433MHz**

<https://www.adafruit.com/product/3179>

Anleitung:

<https://cdn-learn.adafruit.com/downloads/pdf/adafruit-feather-m0-radio-with-lora-radio-module.pdf>

#### **1x Akku Lithium Ion Polymer Battery - 3.7v 500mAh für Sender**

<https://www.adafruit.com/product/1578>

#### **1x Flash Speicher für Empfänger zum Speichern der angelernten SenderIDs**

**DollaTek 5Pcs W25Q64 64Mbit 8MByte Flash-Speichermodul 2.7-3.6V DataFlash SPI-Schnittstelle (Hersteller: winbond)**

[https://www.amazon.de/dp/B07PNDXGXW?ref=ppx\\_yo2ov\\_dt\\_b\\_fed\\_asin\\_title](https://www.amazon.de/dp/B07PNDXGXW?ref=ppx_yo2ov_dt_b_fed_asin_title)

Anleitung:

[https://www.winbond.com/hq/support/documentation/levelOne.jsp?\\_\\_locale=en&DocNo=DA00-W25Q64JV.1](https://www.winbond.com/hq/support/documentation/levelOne.jsp?__locale=en&DocNo=DA00-W25Q64JV.1)

Wird ein anderer Speicher verwendet, muss die Klasse StorageHandler in der Software umgeschrieben werden. Beim obigen Winbond Flash Speicher muss ein Block erst gelöscht werden, um ihn neu beschreiben zu können.

#### **9x Taster (8 Taster für Sender + 1 Taster für Empfänger)**

[https://www.amazon.de/dp/B0B3QYVMVF?ref=ppx\\_yo2ov\\_dt\\_b\\_fed\\_asin\\_title&th=1](https://www.amazon.de/dp/B0B3QYVMVF?ref=ppx_yo2ov_dt_b_fed_asin_title&th=1)

Die hier verlinkten Taster sind aus Metall und machen den Sender recht schwer. Besser wären Kunststofftaster, die eingelassen sind, also vor versehentlichem Drücken geschützt sind.

#### **Optional 8x KlinkenEinbaubuchsen oder Wago Klemmen**

Die parallel zu den Tastern angeschlossen werden können um z.B. Fußschalter anzuschließen.

[https://www.amazon.de/dp/B0CQJVHF52?ref=ppx\\_yo2ov\\_dt\\_b\\_fed\\_asin\\_title](https://www.amazon.de/dp/B0CQJVHF52?ref=ppx_yo2ov_dt_b_fed_asin_title)

#### **2x Gehäuse**

[https://www.amazon.de/dp/B0BWLW941S?ref=ppx\\_yo2ov\\_dt\\_b\\_fed\\_asin\\_title&th=1](https://www.amazon.de/dp/B0BWLW941S?ref=ppx_yo2ov_dt_b_fed_asin_title&th=1)

Beim Gehäuse muss auf die Größe geachtet werden, bei den Maßen 5,5 cm x 9,5 cm x 2,5 cm wird man für den Sender eventuell keine 8 Taster und 8 Buchsen unterbringen.

#### **1x Micro JST PH 2.0 2PIN Rastermaß 2mm Steckverbinder**

um Akku über einen Ein/Ausschalter zu verbinden

[https://www.amazon.de/dp/B08RCXD67Z?ref=ppx\\_yo2ov\\_dt\\_b\\_fed\\_asin\\_title&th=1](https://www.amazon.de/dp/B08RCXD67Z?ref=ppx_yo2ov_dt_b_fed_asin_title&th=1)

**1x Ein-/Ausschalter für Sender Akku**

[https://www.amazon.de/dp/B0BTM3WQXN?ref=ppx\\_yo2ov\\_dt\\_b\\_fed\\_asin\\_title&th=1](https://www.amazon.de/dp/B0BTM3WQXN?ref=ppx_yo2ov_dt_b_fed_asin_title&th=1)

**2x Leuchtdioden mit Widerstand (1x Sender + 1x Empfänger)**

[https://www.amazon.de/dp/B08JC5CHJ7?ref=ppx\\_yo2ov\\_dt\\_b\\_fed\\_asin\\_title&th=1](https://www.amazon.de/dp/B08JC5CHJ7?ref=ppx_yo2ov_dt_b_fed_asin_title&th=1)

**1x USB C Buchse für Sender**

[https://www.amazon.de/dp/B0D5B96FBR?ref=ppx\\_yo2ov\\_dt\\_b\\_fed\\_asin\\_title&th=1](https://www.amazon.de/dp/B0D5B96FBR?ref=ppx_yo2ov_dt_b_fed_asin_title&th=1)

Um den Akku nicht ausbauen zu müssen, kann der USB Micro Anschluss des Feather M0 mit der USB C Buchse verbunden werden, hierzu kann ein USB Micro auf USB A Kabel aufgeschnitten werden:

Bei normaler Farbgebung des Kabels:

Rot auf V, (VCC) Schwarz auf G(Gnd), Grün auf D+, Weiß auf D-

Da die Buchse nur für USB 2.0 vorgesehen ist, benötigt man noch ein Kabel USB 2.0, USB C auf USB A, um den Sender per USB zu laden oder mit dem Rechner für das Aufspielen der Software zu verbinden.

**1x Vibrationsmotor für Sender**

als haptische Rückmeldung, dass der Empfänger den Tastendruck empfangen hat

[https://www.amazon.de/dp/B085G6WB64?ref=ppx\\_yo2ov\\_dt\\_b\\_fed\\_asin\\_title](https://www.amazon.de/dp/B085G6WB64?ref=ppx_yo2ov_dt_b_fed_asin_title)

**2x Antenne Draht 17,31 mm für 433 MHz (Adafruit schlägt 16,5 mm vor)**

Elektronik Kabel falls nicht vorhanden:

[https://www.amazon.de/dp/B09BFPLD2Q?ref=ppx\\_yo2ov\\_dt\\_b\\_fed\\_asin\\_title&th=1](https://www.amazon.de/dp/B09BFPLD2Q?ref=ppx_yo2ov_dt_b_fed_asin_title&th=1)



## Empfänger:

Es ist ein Taster für den Lernmodus (Learning) Pin #11 vorgesehen

Es ist eine Kontroll-LED Pin#13 vorgesehen.

Der Flashspeicher ist für das Speichern der angelernten SenderIDs vorgesehen:

Beim Flashspeicher wird

D0 mit #22 MISO und DI mit #24 MOSI, GND mit GND, VCC mit 3V Pin, SCK mit #24 SCK und CS mit Pin #A5 19 verbunden

Da der Empfänger eine USB-Verbindung mit dem Rechner kommunizieren muss, wird hier kein Akku benötigt.

## Receiver Adafruit Feather M0 RFM9xLoRa <https://www.adafruit.com/products/3178>

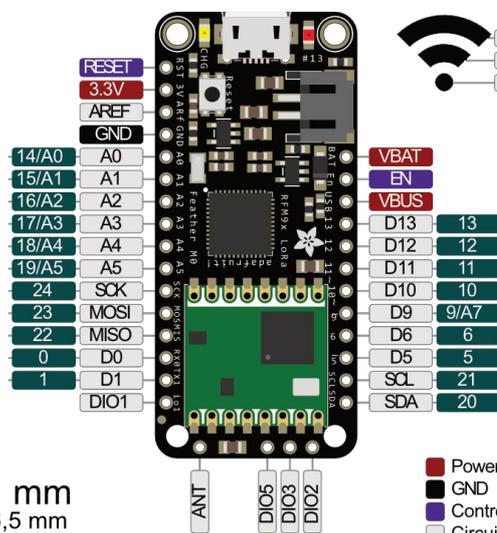
TheMicrochip(neeAtmel)SAMD21isan  
 ARMCortex-M0+runningat48MHzwith  
 32kBon-chipSRAM,256KBFlashmemory  
 andbuiltinUSB.AllGPIOis3.3V  
 in/outmaxunlessotherwisestated.  
 SERCOMscanbeusedasUART(TXon  
 SERCOMpad0or2,RXonanyypad),I2C  
 (SDAonpad0,SCLonpad1),orSPI  
 (SCKonpad1or3,MOSIonpad0or3  
 (SCKonpad1)orpad0or2(SCKonpad  
 3),MISOonanyypadremaining)

### Flash/EEPROM

A5/19 CS  
 24 SCK  
 23 MOSI (DI)  
 22 MISO (DO)

### Wire Antenna

433 MHz 17,31 mm  
 Adafruit Suggestion 16,5 mm



Als Antenne kann ein Draht verwendet werden. Bei 433 MHz sollte der Draht 17,31 mm lang sein, wobei Adafruit eine Länge von 16,5 mm vorschlägt.

## Software

Die Software wurde mit Visual Studio Code und PlatformIO geschrieben:

**Visual Studio Code:** <https://code.visualstudio.com/>

**PlatformIO:** Visual Studio Code bietet PlatformIO als Plugin an.

Ansonsten: <https://platformio.org/>

Die Software für den Sender und die Software für den Empfänger kann unter [www.liveshowsoftware.de](http://www.liveshowsoftware.de) unter dem Menüpunkt 'Requisiten- / Hardwaresteuerung' heruntergeladen werden.

Die Sender-Software muss auf den Sender übertragen werden und die Empfänger-Software auf den Empfänger.

Falls ein anderer Flash/EEPROM, Speicher als oben angegeben, verwendet wird, muss in der Empfänger-Software die Klasse 'StorageHandler' umgeschrieben werden!!!

Die Pins für den LoRa Baustein werden in der Klasse LoraHandler festgelegt.

### Funktionsweise:

*Der Empfänger wird über ein USB-Kabel an den Rechner angeschlossen. Beim Start blinkt die LED des Empfängers 3 mal. Sind keine Sender angelernt, so blinkt die LED daraufhin dauerhaft. Die LED leuchtet kurz auf, wenn eine Nachricht empfangen wurde.*

*Der Sender besitzt eine eigene Stromversorgung und muss eingeschaltet werden.*

*Direkt nach dem Einschalten wird die Batterie des Senders geprüft, ist die Spannung noch über 3.6 Volt, so blinkt die LED des Senders 3 mal, der Vibrationsmotor vibriert 3 mal..*

*Ist die Spannung unter 3.6 Volt, so blinkt die LED des Senders nicht und der Akku muss geladen werden!*

Jeder Sender besitzt eine eigene ID (dies ist eine Umrechnung der Seriennummer). Der Empfänger akzeptiert nur Nachrichten von Sendern, die angelernt wurden – siehe weiter unten.

Für das Drücken und für das Loslassen eines Tasters am Sender wird jeweils eine Nachricht an den Empfänger gesendet. Somit wird das Drücken und Loslassen eines Tasters unterschieden.

Wird eine Taste des Senders gedrückt, wird eine Nachricht an den Empfänger gesendet. Der Empfänger sendet eine Kopie der Nachricht zurück an den Sender, als Bestätigung für den Empfang, die Status LED des Senders leuchtet auf und der optionale Vibrationsmotor vibriert.

Wird der Taster losgelassen, wird eine Nachricht an den Empfänger gesendet. Der Empfänger sendet eine Kopie der Nachricht zurück an den Sender, als Bestätigung für den Empfang, die Status LED des Senders leuchtet auf und der optionale Vibrationsmotor vibriert.

Die Status LED des Senders leuchtet nur auf, wenn die Bestätigung des Empfängers ankommt, also die Nachricht vom Empfänger auch empfangen wurde. In allen anderen Fällen bleibt die Status-LED dunkel.

Damit der Empfänger einen Sender akzeptiert, muss ein Sender angelernt werden. Es können mehrere Sender angelernt werden. Hierzu hat jeder Sender eine eigene SenderID, dies ist im Prinzip die umgerechnete Seriennummer des Bausteins.

1. Am Empfänger die Lerntaste drücken und loslassen

- Lernmodus an: Die Status LED blinkt.  
(Wird die Lerntaste nochmals gedrückt und losgelassen, wird der Lernmodus wieder ausgeschaltet).
2. Am Sender eine beliebige Taste drücken und wieder loslassen.  
Empfängt der Empfänger die Nachricht, wird die ID des Senders akzeptiert und das Blinken der Status LED hört auf.  
Die Sender ID wird dauerhaft auf dem Empfänger gespeichert.
  3. Auf diesem Weg können bis zu 20 Sender angelernt werden.
  4. Um die Liste der akzeptierten Sender auf dem Empfänger zu löschen, muss die Lerntaste länger als 3 Sekunden gedrückt werden, bis das Blinken der Status LED aufhört und diese dauerhaft leuchtet. Daraufhin können Sender neu angelernt werden

Der Empfänger kann auf zwei Arten als Fernbedienung der liveSHOW Software verwendet werden:

### **Als Tastatur (HID)**

Der Empfänger funktioniert, wie eine externe Tastatur (HID).  
Wird am Sender z.B. die Taste A gedrückt und die Nachricht erreicht den Empfänger, so ist dies als ob auf einer Tastatur der Buchstabe A gedrückt wird, Wird die Taste A losgelassen, entspricht dies dem Loslassen des Buchstabens A auf der Tastatur.  
Die einzelnen Sender werden hierbei nicht unterschieden, es gibt nur die Tasten A – H,  
In der liveSHOW Software kann dies mit der 'Maus-Tastatur-Fernbedienung' verwendet werden.

### **Mit der liveSHOW\_RXTX Software**

*(siehe auch Hilfe in der liveSHOW\_RXTX Software*

In der liveSHOW-RXTX Software kann ein USB-Gerät (seriell) angelegt und mit dem Empfänger verbunden werden. Ist der Empfänger einmal als serielles Gerät verbunden, so wird die Tastatur (HID) Funktion ausgeschaltet! Erst nachdem der Empfänger ausgesteckt und wieder eingesteckt wurde, ist die Tastatur (HID) Steuerung wieder aktiv.

Die liveSHOW\_RXTX Software kann mit der liveSHOW Software verbunden werden.  
In der liveSHOW\_RXTX Software werden nun alle Fernbedienungs-Befehle und in einem weiteren Karteikartenreiter, alle Trigger der liveSHOW Software aufgelistet

### **Fernbedienen:**

Wird ein Fernbedienungsbehl in der liveSHOW\_RXTX Software angeklickt, kann dort das Lernen eingeschalten werden.

Wird nun z.B. der Taster A des Senders betätigt und dies vom Empfänger registriert, so erscheinen in der liveSHOW\_RXTX Software zwei Kommandos:

LoraSender(SenderID)\_A\_Down für das Drücken des Tasters A und  
LoraSender(SenderID)\_A\_Up für das Loslassen des Tasters A.

Es kann nun entschieden werden, ob der ausgewählte Fernbedienung-Befehl beim Drücken oder beim Loslassen des Tasters ausgeführt werden soll. Das jeweilige andere Kommando kann gelöscht werden.

Dadurch, dass die *SenderID* mitgeschickt wird, können hier die einzelnen Sender unterschieden werden.

## **Besonderheit Code umstellen:**

Besitzt die liveSHOW Software Trigger, so werden diese in der liveSHOW\_RXTX Software unter 'Trigger' aufgelistet.

Diesen Triggern können Kommandos zugewiesen werden, die an den Empfänger gesendet werden können.

Wobei der Empfänger nur einen Befehl versteht:

Normalerweise beginnen die Taster aller Sender mit dem Buchstaben A, der nächste Taster sendet den Buchstaben B, usw..

Wird für einen Trigger z.B. das Kommando

LoraSender(SenderID)\_I

eingestellt, so werden im Empfänger die Tasten des Senders mit der (*SenderID*) neu interpretiert, so als ob die erste Taste des Senders mit dem Buchstaben 'I' beginnen würde. Der nächste Taster entspräche dann dem Buchstaben 'J', usw.

Hier werden nur Großbuchstaben verwendet, Sonderzeichen und Umlaute dürfen nicht verwendet werden, da diese je nach Tastaturlayout am Rechner nicht immer gleich sind. Somit bleiben die Zahlen ab 0 und die 26 Buchstaben des Alphabets übrig.

Anstelle von 'I' kann auch ein anderer Buchstabe verwendet werden, der letztmögliche Buchstabe ist 'S', da sonst für den 8-ten Taster kein Buchstabe übrigbleibt.

Setzt man den Trigger in die Timeline oder in ein Jingle der liveSHOW Software und wird der Trigger abgespielt, wird an dieser Stelle der Empfänger auf die oben beschriebene Art umgestellt. So können mit einem Sender je nach Ablauf der Show völlig andere Fernbedienungs-Befehle verwendet werden.

Im Gegensatz zur Liste der akzeptierten SenderIDs, wird das Umstellen des Codes im Empfänger nicht dauerhaft gespeichert. Wird die USB-Verbindung des Empfängers getrennt, ist wieder der originale Code eingestellt. Es muss erst wieder der Trigger abgespielt werden, um den Code zu ändern.

Diese Technik eignet sich auch dafür, den Sender kurzzeitig abzuschalten, indem man über einen Trigger einen Buchstaben-Code einstellt, der in der RXT Software nicht verwendet wird.