

# MYC Funk Interface

Author DK1RI, Version V01.0, 20241218

This paper is published in <https://github.com/dk1ri> as well

**Dieses Projekt ist noch in Entwicklung und nicht getestet!!!**

## Einleitung

Die meisten MYC Geräte (funcs) haben bisher drahtgebundene Interfaces (I2C, seriell oder USB). Für einige, wie zum Beispiel den Klimasensor, ist eine Funkverbindung nötig, wenn sie weiter entfernt vom Commandrouter - oder, wenn der nicht vorhanden ist, vom Webserver oder HI - platziert werden sollen.

Dies ist die Beschreibung eines Funk Interfaces mit RFM95 oder RYFA689 Modem. Wenn die Datenmenge sehr gering ist, bietet sich diese Interfaces an.

Das Funkinterface kann nur jeweils mit einem Gerät (func) verbunden werden.

Diese Interfaces können in einem MYC System verwendet werden, aber auch unabhängig davon mit (binären) Befehlen oder mit einem Browser verwendet werden oder auch als transparente Datenstrecke über Funk.

Für den MYC Mode sind die Befehle als announcements in der Datei announcements.bas im Bascom Programm beschrieben.

Im Transparent Mode des Servers werden Daten transparent bidirektional von Server zum Client übertragen.

Die Funk relevanten Programmteile sind im Verzeichnis commonxx enthalten. Sie können so für die Serverseite und Client (funcs) verwendet werden, sofern das Funk Modem an die funcs angeschlossen werden kann und die funcs eine passende Firmware haben.

Hier werden allgemeine Aspekte des Funkinterfaces und Details zur Serverseite beschrieben. Details zur Clientseite wird bei den Geräten (funcs) beschrieben.

### *RFM95*

Es gibt fertige Module für den RFM95 LoRa mit ESP32 als Prozessor; das erscheint mir aber überdimensioniert, wenn die Funktionalität auch mit einem 10MHz ATMEGA erreicht werden kann.

Daher gibt es eine je eine Leiterplatte für den (Web / Commandrouter)-Server mit ATMEGA und eine Leiterplatte für den Anschluss an die Sensoren (funcs) als Client.

Das Server Interface kann seriell oder über USB gesteuert werden.

Zur Steuerung mit einem Browser: siehe [7].

### *RYFA689*

Weitere Details zum Einsatz des RYFA689 stehen weiter unten (Einsatz des RYFA689 Modems) und in der Zeitschrift „Funkamateure“ Heft 2024 / 12.

*Weitere Funkmodule können folgen*

## Grenzen des Einsatzes in einem MYC System

Diese Funksysteme sind für geringen Datenverkehr ausgelegt, Die Daten sind nicht verschlüsselt. Befehle, die einen Missbrauch ermöglichen, sollten bei Betrieb über Funk gesperrt werden. Die Datenpakete dürfen auch nicht zu lang sein.

## **Einbindung in das MYC System**

Details zum MYC System stehen in [3].

Auf der Serverseite ist das Funkinterface eine Funktion aber nicht als MYC Schnittstelle aktiv und muss daher mit der Individualisierung für die Serverseite abgeschaltet werden. Dies wird auch zur Unterscheidung der Server und Client (func) Seite verwendet, da der Datenfluss für Server und Client unterschiedlich ist. Als Default ist die Firmware als MYC Schnittstelle eingeschaltet (für func).

Wenn die „remote“ Konfiguration des Funkinterfaces der funcs gewünscht ist, müssen die Befehle zur Steuerung des Funkinterfaces dann auch im Befehlssatz des func erscheinen. Die HI erhält ja die Befehle der func. Ob die Befehle die func oder das Funkinterface des Servers erreichen, hängt vom Mode des Servers ab: MYC mode oder Transparent mode. Die func verarbeitet alle Eingaben über alle aktiven Schnittstellen als Befehle, auch die zu Konfiguration des Funkinterfaces.

Die func sollte die Möglichkeit anbieten, die Konfiguration seines Funkinterfaces zu sperren.

## **Umschaltung vom Transparent in den Myc Mode (Server)**

Die Umschaltung ist auf der Serverseite möglich.

Es wird der Befehl &HF8 und einer 32 Byte langen Datensequenz gesendet wird. Die soll eine versehentliche Umschaltung verhindern:

&HF8&H209d,?sz9\_+w5.,d6;t8\*<32ksy(e\$fhN6

Die Umschaltmöglichkeit kann im MYC mode deaktiviert werden. Um dann wieder in den MYC Mode zu kommen, muss man das (Server-)Interface mit JP4 neu gestartet werden.

Man kann so die Konfiguration auch im laufenden MYC System vornehmen.

Der Befehl &HF8 wird im MYC Mode des Servers und beim Client auch zum Änderung des Namens des Funkinterfaces benutzt

## **Transparent Mode (Server Interface)**

Ohne Jumper JP4 ist das Funkinterface beim Einschalten im Transparent Mode;. Dies sollte die normale Betriebsart sein.

Ob JP4 vorhanden ist, wird nur beim Einschalten geprüft.

Bei Verwendung des RYFA689 wird den Nutzdaten über die Funkstrecke wird ein Byte mit der Datenlänge vorangestellt.

## **Beschreibung (RFM95 Modul)**

Die Eagle Daten für die Leiterplatten stehen unter [1]

Die Stromversorgung der Server Interfaces erfolgt über USB oder muss über JP7 (3.3V !!!) zugeführt werden.

Es wird dringend empfohlen, bei Änderung der Firmware das Datenblatt des RFM95 zu lesen.

## **MYC Mode (Server Interface)**

Die Konfigurationsbefehle haben commandtoken direkt unterhalb der reservierten token, um eine möglichst gute Trennung zu den normalen Funktionen zu haben. Falls beim Client commandtoken mit einem Byte nicht reichen, muss man zwei Byte verwenden.

Der Funkname muss &HF8 geändert werden. Bei Server und Client muss der gleiche Name verwendet werden; es dient zu Ausfiltern fremder Datenpakete. Als Zeichen sollten nur Buchstaben,

Zahlen und „\_“ verwendet werden. Die Länge ist auf < 5 Byte begrenzt.

Die meisten Parameter des Funkmoduls werden als Defaultwerte belassen, einige werden nur bei der Initialisierung geändert. Wenn Änderungen nötig sind, muss die Firmware neu kompiliert werden.

### **Fehlermeldungen (Server Interface)**

Für den MYC Mode:

Der Befehl &HFC liefert den letzten Fehler im Format:

aktuelle Befehlsnummer - Fehler - Befehlsnummer beim Auftritt des Fehlers

Dazu werden die empfangenen Befehle von 0 bis 255 umlaufend gezählt.

Nach 254 korrekten Befehlen wird der Fehlereintrag gelöscht.

### **Reset (Server Interface)**

Ist der Reset Jumper JP5 beim Anlegen der Versorgungsspannung überbrückt, werden wieder die Defaultwerte eingelesen.

### **Watchdog (Server Interface)**

Für den MYC Mode:

Es gibt einen kompletten Hardware-reset, wenn die Hauptschleife länger als 2 Sekunde dauert.

Zusätzlich gibt es drei weitere Watchdogs, die in der vorliegenden Firmware für Tests und

„nicht\_MYC Betrieb“ nach ca 10 Sekunden ansprechen. Für „MYC Betrieb“ sollte der Wert auf 1 Sekunde gesetzt werden.

Die Befehlseingabe und Ausführung muss in dieser Zeit beendet sein. Danach werden die bereits empfangenen Daten gelöscht. Dies soll falsche Eingaben vermeiden. Mit dem &HFC "letzten Fehler" Befehl kann man Eingabefehler sehen.

### **Software (Server Interface)**

Die Steuerung übernimmt ein AVR Mikrocontroller Atmega328.

Die Software wurde in BASCOM geschrieben [2]

Um das Programm zu kompilieren, muss das Verzeichnis common\_1.14 [6] in das Verzeichnis mit dem Programm kopiert werden und das gewünschte Funkmodul bei Const Radiotype = x eingetragen werden.

Sofern vorhanden, gibt es hex Dateien für die verschiedenen Funkmodule.

### **Programmierung des Prozessors (Server Interface)**

Zur Programmierung des Prozessors ist ein 6poliger ISP Stecker (JP6 bei der RFM95 Leiterplatte) vorgesehen.

Für die verschiedenen Funkmodule gibt es jeweils eigene Hex Dateien (getrennte Kompilierung).

Die Fuses müssen möglicherweise programmiert werden (siehe Bascom Programm).

Während der Programmierung JP1, JP2, JP9 – JP11 entfernt werden. Andernfalls wird das Modem zerstört!!

### **Serielle (RS232 / USB) Schnittstelle (RFM95 Modul Server Interface)**

Schnittstellenparameter: 19k2 8N1.

**Alternativ** zur RS232 Schnittstelle kann die USB Platine UM2102 von ELV verwendet werden. Die USB Platine wird plan auf der Oberseite der Interfaces verlötet: der USB Stecker zeigt zum Rand. USB Platine und Interface müssen voneinander isoliert werden.

### **SMD (RFM95 Modul Server Interface)**

Die Leiterplatte ist teilweise mit SMD bestückt.

### **Stromversorgung (RFM95 Modul Server Interface)**

Die Stromversorgung erfolgt über USB oder über JP7 JP8 (3.3V !!)  
Mit einer späteren Hardwareversion können auch Adafruit Funkmodule verwendet werden. Die Versorgungsspannung muss dann auf 5V umgesteckt werden (JP?)

### **Bestückung der Leiterplatte (RFM95 Modul Server Interface)**

Verwendung von ISP:

JP6

Mit serieller Schnittstelle:

Die serielle Schnittstelle hat keinen Pegelwandler (nur 3,3V Pegel)

### **Jumper (RFM95 Modul Server Interface)**

JP1	Power
JP5	Reset
JP6	ISP
JP7/JP8	Anschluss für USB Modul

### **RFM95 Modul Client Interface**

Der Sensor / Client (func) muss für den Funk-Modul vorbereitet sein (Hardware und Software). Der Client wird über den 8 poligen Anschluss mit dem Funk Interface verbunden.

Wenn das Funkinterface des Client konfiguriert werden sollen, müssen entsprechende Befehle auch beim Client vorhanden sein. In jedem Fall sollte der der Funkname änderbar sein. Die Konfiguration des Client kann über die Funkschnittstelle vorgenommen werden. Es wird aber empfohlen, eine serielle Schnittstelle zu verwenden und das Ändern der Funkmodulparameter mit Hilfe eines Jumpers zu verhindern.

Weitere Details sind bei den func zu finden.

Zur Zeit sind folgende func für die Funkschnittstelle vorbereitet:

Feinstaubsensor

### **Einsatz des RYFA689 Moduls**

Die Verwendung dieses Funkmoduls ist in der Zeitschrift Funkamateure 2024 / 12 beschrieben. Es gibt fertige Leiterplatten mit Firmware; ich habe daher keine Leiterplatte entwickelt. Die originale Firmware ist mir (bisher) nicht bekannt.

Für den Einsatz in einem MYC System oder MYC Clients muss aber die MYC Firmware (Server) verwendet werden.

Bei Verwendung auf der Serverseite mit dem Webserver muss die Variante 1 mit einem USB →

seriell Wandler ergänzt werden.

Auf der func (Sensor) Seite wird die Variante 2 verwendet. Funcs, die die Funkschnittstelle verwenden, haben eine passende Schnittstelle, an die auch die LoRa Module angeschlossen werden.

## Versionen

Diese Beschreibung gilt für die  
Leiterplattenversion V01.0  
Bascom Version V01.0

## Copyright

Die Ideen in diesem Dokument unterliegen der GPL (Gnu Public Licence V2) soweit keine früheren, anderen Rechte betroffen sind.

Die Verwendung der Unterlagen erfolgt auf eigene Gefahr; es wird keinerlei Garantie / Gewährleistung / Produkthaftung übernommen.

The ideas of this document can be used under GPL (Gnu Public License V2) as long as no earlier other rights are affected.

The usage of this document is on own risk, there is no warranty.

## Referenzen

- [1] [dk1ri.de/dhw/LoRa\\_interface\\_eagle.zip](https://dk1ri.de/dhw/LoRa_interface_eagle.zip)
- [2] [dk1ri.de/dhw/LoRa\\_interface\\_bascom.zip](https://dk1ri.de/dhw/LoRa_interface_bascom.zip)
- [3] <https://www.dk1ri.de/myc/MYC.pdf>
- [4] <https://dk1ri.de/myc/Description.txt> oder <https://dk1ri.de/myc/Description.pdf> (englisch)
- [5] <https://dk1ri.de/myc/Definitions.txt> oder <https://dk1ri.de/myc/Definitions.pdf> (englisch)
- [6] [https://dk1ri.de/dhw/common\\_1.13.zip](https://dk1ri.de/dhw/common_1.13.zip)
- [7] <https://dk1ri.de/myc/webserver.pdf> oder <https://dk1ri.de/myc/webserver.txt>
- [8] [https://dk1ri.de/w\\_dat.htm](https://dk1ri.de/w_dat.htm)
- [9] <https://sander-electronic.de>