

## Feinstaub Sensor mit Funkschnittstelle für das MYC Protokoll

Author DK1RI, Version V02.0, 20251209

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

### Definitionen

<https://dk1ri.de/myc/Definitions.txt> oder <https://dk1ri.de/myc/Definitions.pdf> (englisch)

### Einleitung

Dieses Gerät kann in einem MYC System verwendet werden, aber auch unabhängig davon mit (binären) Befehlen gesteuert werden. Die Steuerung kann seriell, USB, I2C, über ein Funkinterface erfolgen. Die Befehle sind als announcements in der Datei announcements.bas im Bascom Programm beschrieben.

Zur Steuerung mit einem Browser: <https://dk1ri.de/myc/webserver.pdf> oder <https://dk1ri.de/myc/webserver.txt>

Defaultmäßig sind alle Schnittstellen aktiv. Mit dem Initialisierungsbefehl können diese aber deaktiviert werden. Der Initialisierungsbefehl funktioniert aber immer.

Das Gerät besteht aus einer Leiterplatte mit dem Prozessor und dem Funkinterface und dem Feinstaub Sensor. Die Leiterplatte ist auch für andere Sensoren vorgesehen, auch für verschiedene Funkmodule und unterschiedliche Stromversorgung. Für eine spezielle Anwendung lässt sich die Hardware erheblich vereinfachen.

Zur Fernsteuerung über kurze Entfernungen kann ein Funkmodul angeschlossen werden:

[https://www.dk1ri.de/dhw/Wireless\\_interface.pdf](https://www.dk1ri.de/dhw/Wireless_interface.pdf) oder [https://www.dk1ri.de/dhw/Wireless\\_interface.txt](https://www.dk1ri.de/dhw/Wireless_interface.txt)

**Zur Zeit (Dezember 2025) funktioniert Lora mit RFM95 und 2,4GHz ISM mit nRF24**

Die Unterlagen zum Feinstaubsensor ohne Funkmodul:

<https://www.dk1ri.de/dhw/Feinstaubsensor.pdf> oder <https://www.dk1ri.de/dhw/Feinstaubsensor.txt>

Diese Version wird nicht weiter entwickelt.

### Beschreibung und Bedienung

#### *Allgemeines:*

Die Stromversorgung ist 7- 9V, Stromaufnahme ca. 140mA max oder über USB

Es wird der gleiche Sensor wie in dem ELV Artikel (ELV Journal 2019 Heft 6) verwendet:

Sensirion SPS30 ab Firmware 2.1.

Der Sensor ist aber anders als da nicht auf der gleichen Leiterplatte montiert, und es gibt auch kein Display.

Details zu der Funktion des Sensors können in dem ELV Artikel oder im Datenblatt des Sensors nachgelesen werden.

Der Sensor ist über die serielle Schnittstelle mit der Leiterplatte verbunden.

Der Prozessor ATMEGA1284 wird verwendet, da er 2 serielle Hardware Schnittstellen hat und 16kB RAM. Dies erlaubt die Speicherung von 765 Messwerten.

Das Messintervall beträgt 3s, 10s, 1min, 10min, 30min und 60min.

Jeder Messwert benötigt 20 Byte, es können also nur 750 Messwerte gespeichert werden (Das ELV Original hat einen zusätzlichen 4MBit Speicher und kann daher mehr Messwerte speichern).

Ist der Speicher voll, werden alte Daten überschrieben.

Beim Auslesen des Speichers kommt der neueste Messwert zuerst.  
Da das Interface auch keine Uhr hat, muss die Messzeit an Hand der auslesbaren Intervallzeit berechnet werden. Beim Ändern der Messintervallzeit , Start, Stop, Reset und Abschalten gehen die gespeicherten Daten verloren.

#### *Details:*

Die Bedienung mit dem Browser ist weitestgehend selbsterklärend.

Die Parameter zu Funkübertragung können nicht über Funk verändert werden. Zu Änderung muss JP11 überbrückt sein.

Wenn beim Start, das eingestellte Funkinterface nicht angeschlossen ist, wird das Funkinterface abgeschaltet.

Im folgenden werden auch Befehlsfolgen beschrieben, wie sie über die Schnittstellen direkt eingegeben werden. Alle Befehlsfolgen stehen auch mit dem Browser zur Verfügung.

Start der Messung: &H1D01

Das Auslesen erfolgt in Blöcken von maximal 28 Werten, beginnend mit den neuesten Daten.

Bei einem Messintervall von einer Minute gibt der Befehl „read mass concentration“ mit „to send“: 5 und „start at“: 255 (&H0100FF05 )

5 Werte der Massekonzentration von vor 255 – 259 Minuten aus.

Entsprechend dem MYC Protokoll muss der Startmesswert als 2 Byte Wert eingegeben werden (00FF).

Der Grund für die Limitierung der übertragenen Werte ist die Funk Übertragung.

Der Sleep mode des Sensors wird nicht unterstützt

### **Einbindung in das MYC System**

Details zum MYC System stehen in <https://www.dk1ri.de/myc/MYC.pdf> .

Die komplette Befehlsliste steht als announcements in der Datei announcements.bas im Bascom Programm.

### **Fehlermeldungen**

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.

### **Hardware Reset**

Ist der Reset Jumper JP5 beim Anlegen der Versorgungsspannung überbrückt, werden wieder die Defaultwerte eingelesen. Dies ist hilfreich, wenn die aktuelle I2C Adresse verloren gegangen ist.

### **Watchdog**

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 „Testbetrieb“ nach ca 10 Sekunden ansprechen. Für den „nicht Test 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.

Bei einem I2C Lesebefehl müssen die Daten innerhalb dieser Zeit vom I2C Master abgeholt werden. Danach werden die Daten gelöscht. Neue Befehle können erst eingegeben werden, wenn alle Daten abgeholt wurden oder die Watchdog Zeit abgelaufen ist. Wird die RS232 / USB Schnittstelle verwendet, werden die Daten sofort ausgegeben.

Bei einem I2C BusLock (SDA pin auf 0) erfolgt auch ein I2C reset.

## **Firmware**

Die Steuerung übernimmt ein AVR Mikrocontroller Atmega1284.

Die Software wurde in BASCOM geschrieben

[https://www.dk1ri.de/dhw/Feinstaubsensor\\_w\\_bascom.zip](https://www.dk1ri.de/dhw/Feinstaubsensor_w_bascom.zip) .

Dort gibt es einen Hex Datei zur Programmierung des ATMEGA

Um das Programm zu kompilieren, muss das Verzeichnis common\_V01.14 in das Verzeichnis mit dem Programm kopiert werden. Die aktuelle gültige Version steht im Bacom Programm.

Die Firmware funktioniert nur mit der Leiterplattenversion ab Version 5.0

## **Programmierung des Prozessors**

Zur Programmierung des Prozessors ist ein 6poliger ISP Stecker JP6 vorgesehen.

Die Fuses müssen möglicherweise programmiert werden (JTAG abschalten!!). Prozessortyp und Frequenz müssen gegebenenfalls angepasst werden.

Der Jumper JP1 und sollte während der Programmierung entfernt werden; ebenso der Sensor und der Funkmodul.

## **Serielle (RS232 / USB) Schnittstelle**

Für die RS232 Schnittstelle wird IC4 und C6 – C9 bestückt. Die seriellen Daten können an JP9 abgenommen werden. 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. Die 6 pins des Verbinders ST2 sind mit den 6 Anschlusspunkten JP7 / JP8 auf dem Interface zu verbinden. USB Platine und Interface müssen voneinander isoliert werden.

Die Stromversorgung kann dann über USB erfolgen.

## **I2C Schnittstelle**

Die Default Adresse ist 28

Mit dem Befehl &HFE03<n> kann die Adresse in n (1 ... 127) geändert werden.

Pullup Widerstände R3 / R4 müssen immer bestückt werden (1k - 10k).

Mit JP2 kann festgelegt werden, ob der I2C Bus mit 3V oder 5V betrieben wird.

Wenn auf den 3V Betrieb völlig verzichtet werden soll, kann IC3 (PCA9517), R1, R2 entfallen und stattdessen wird JP3 und JP4 bestückt und überbrückt.

Ganz ohne I2C kann auch SL1, JP3, JP4 R3, R4 entfallen.

Der Gesamtwiderstand am I2C Bus sollte bei 1 bis 10 kOhm je nach Leitungslänge liegen

Mit IC3 muss R1 / R2 (<=10k) bestückt werden. Wenn auf IC3 verzichtet wird und JP3 / JP4

verwendet wird, muss berücksichtigt werden, dass R1 / R2 parallel zu R3 / R4 liegt. R1 / R2 kann also gegebenenfalls entfallen.

Der I2C Anschluss erfolgt an SL1.

Um Buslocks zu vermeiden, wird circa alle 200ms geprüft, ob das SDA Signal auf „0“ liegt.

Ist das 50 mal hintereinander der Fall, wird die I2C Schnittstelle neu gestartet.

## Browser Schnittstelle

Es gibt einen (Windows) Webserver, an den die FU direkt oder über die Funkschnittstelle angeschlossen wird. Die Bedienung erfolgt mit einem Browser, der auf den Webserver zugreift. Details dazu stehen in <https://dk1ri.de/myc/webserver.pdf> oder <https://dk1ri.de/myc/webserver.txt> . Ein Bildschirm Bild und nötige Daten für dieses Device stehen in [https://dk1ri.de/w\\_dat.htm](https://dk1ri.de/w_dat.htm) ,

## Remote Bedienung (über Funk)

Zur Überbrückung kurzer Entfernungen kann eine Funkschnittstelle angeschlossen werden. Je nach verwendeter Funkschnittstelle muss die FU entsprechend programmiert werden (JP11 überbrückt). Es muss zumindest das gewünschte Funkmodul, Name und Frequenz eingetragen angepasst werden. Danach entfernt man den Jumper für den normalen Betrieb.

## SMD

Die Leiterplatte ist teilweise mit SMD bestückt

## Stromversorgung

Die Stromversorgung erfolgt mit 7 -12V oder 5V über JP10 oder eine USBmicro Buchse. Bei 12V Versorgung wird JP10, D1, und C1 und IC2 bestückt. Wenn 3V benötigt werden, wird der Regler IC5 verwendet. Dies ist ein 78xx kompatibler Schaltregler. Alternativ erfolgt die Stromversorgung über USB. **IC2 und IC5 dürfen dann nicht bestückt werden!** Wenn wegen hoher Ströme die 7-12V Stromversorgung und auch USB benötigt wird, dürfen JP8 Pin 1 und 2 nicht mit dem USB Modul verbunden werden.

## Bestückung der Leiterplatte

[https://www.dk1ri.de/dhw/Wireless\\_interface\\_eagle.zip](https://www.dk1ri.de/dhw/Wireless_interface_eagle.zip)

Die Leiterplatte ist für verschiedene Konfigurationen ausgelegt und wird daher normalerweise nur teilweise bestückt. Für genau definierte Anwendungen können die Leiterplatten verkleinert werde

Verwendung von ISP:

JP6

Mit I2C:

Siehe I2C oben.

Mit serieller Schnittstelle:

Siehe Serielle Schnittstelle oben

Externe Stromversorgung:

Siehe Stromversorgung oben.

Außer den bereits erwähnten Bauteilen wird immer bestückt: Q1, IC1, U1, C2 – C5, C10 - C12, JP1, JP5, JP11, JP32, JP33 - JP37

Der Sensirion Modul wird an JP37 (Stiftleiste) über Kabel (mit Buchse) angeschlossen.

## **Anschlüsse**

JP 15 (12V / 5V Stromversorgung)

1	GND
2	12V
3	5V

JP 9 (RS232)

1	GND
2	RX (PC Ausgang)
3	TX (PC Eingang)

I2C (SL1, SL2)

1	GND
2	SCL
3	SDA

Feinstaubsensor (JP37) Drahtanschluss

JP37 1 5V	Sensirion Pin1 5V
JP37 2 TX	Sensirion Pin2 RX
JP37 3 GND	Sensirion Pin5 GND
JP37 4 RX	Sensirion Pin3 TX
	Sensirion Pin4 nicht verbunden

## **Jumper**

JP1	Power
JP2	I2C: 3V (1-2) 5V (2-3) Umschaltung
JP3	SDA Überbrückung (ohne IC3)
JP4	SCL Überbrückung (ohne IC3)
JP5	Reset
JP6	ISP
JP7/8	Anschluss für USB Modul
JP35	Pin1 und 2 überbrücken
JP36	Pin1 und 2 überbrücken

## **Versionen**

Diese Beschreibung gilt für die Leiterplattenversion (wireless interface) V05.0

[https://www.dk1ri.de/dhw/Wireless\\_interface\\_eagle.zip](https://www.dk1ri.de/dhw/Wireless_interface_eagle.zip)

Bascom Version V03.0

[https://www.dk1ri.de/dhw/Feinstaubsensor\\_w\\_bascom.zip](https://www.dk1ri.de/dhw/Feinstaubsensor_w_bascom.zip) .

## **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 Geafahr; es wird keinerlei Garantie 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.