

MYC Klimasensor

Author DK1RI, Version V03.3, 20200504

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

Einleitung

Dies ist die Beschreibung eines Klimasensors mit dem Sensor BME280

Dieses Gerät kann in einem MYC System verwendet werden, aber auch unabhängig davon mit (binären) Befehlen gesteuert werden. Die Befehle sind im Kapitel „Einbindung in das MYC System“ beschrieben.

Das Interface (grosse Leiterplatte) arbeitet als Slave am I2C Bus oder kann über RS232 /USB gesteuert werden. Die kleine Leiterplatte hat nur ein USB Interface.

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

Beschreibung

Die Eagle Daten für die Leiterplatte stehen unter [1].

Die Stromversorgung ist 12V +-10%, Stromaufnahme ca. 20 max.

Die Steuerung kann über I2C, USB oder RS232 erfolgen (kleine Version: nur USB).

Die Schaltung verwendet das Sparkfun BM280 Hookup Modul (TM).

Neben dem Auslesen von Temperatur, Feuchte und Luftdruck können auch die Parameter des BME280 geändert werden. Die genauen Auswirkungen müssen im Datenblatt nachgelesen werden.

Die Default - Einstellung sollte für viele Anwendungen passen.

Ein Reset liest auch die Korrekturdaten. Dies ist nötig, wenn der Sensormodul gewechselt wird.

Einbindung in das MYC System

Details zum MYC System stehen in [3].

Folgende Befehle werden akzeptiert:

Announce0:

'Befehl &H00

'basic announcement wird gelesen

'basic announcement is read

Data "0;m;DK1RI;Klimasensor;V02.2;1;145;1;21;1-1"

,

Announce1:

'Befehl &H01

'liest Temperatur

'read temperature

Data "1;ap,read temperature;1;12500,{-40.00 to 84.99};lin;DegC"

,

Announce2:

'Befehl &H02

'liest Feuchtigkeit

'read humidity

Data "2;ap,read humidity;1;100001,{0.000 to 100.000};lin;%"

'
Announce3:
'Befehl &H03
'liest Druck
'read pressure
Data "3;ap,read pressure;1;1100001,{0.000 to 1100.000};lin;hPa"
,

Announce4:
'Befehl &H04 0 to 5
'schreibt Oversampling Feuchte
'write Oversampling humidity
Data "4;oa,oversampling humidity;b,{0,1,2,4,8,16}"
,

Announce5:
'Befehl &H05
'liest Oversampling Feuchte
'read Oversampling humidity
Data "5;aa,as4"
,

Announce6:
'Befehl &H06 0 to 5
'schreibt Oversampling Druck
'write Oversampling pressure
Data "6;oa,oversampling pressure;b,{0,1,2,4,8,16}"
,

Announce7:
'Befehl &H07
'liest Oversampling Druck
'read Oversampling pressure
Data "7;aa,as6"
,

Announce8:
'Befehl &H08
'schreibt Oversampling Temperatur
'write Oversampling Temperature
Data "8;oa,oversampling Temperatur;b,{0,1,2,4,8,16}"
,

Announce9:
'Befehl &H09
'liest Oversampling Temperatur
'read Oversampling Temperatur
Data "1;aa,as8"
,

Announce10:
'Befehl &H0A
'schreibt Pause Zeit
'write non active time
Data "10;oa,non activ time;b,{0.5,62.5,125,500,1000,10,20},ms"
,

Announce11:

'Befehl &H0B
 'liest Pause Zeit
 'read non active time
 Data "11;aa,as10"
 ',

Announce12:
 'Befehl &H0C
 'schreibt Filter
 'write Filter
 Data "12;oa,filter;b,{0,2,4,8,16}"
 ',

Announce13:
 'Befehl &H0D
 'liest Filter
 'read Filter
 Data "13;aa,as12"
 ',

Announce14:
 'Befehl &H0E
 'liest ID
 'read ID
 Data "14;aa,read ID;b"
 ',

Announce15:
 'Befehl &H0F
 'Reset
 'Reset
 'Data "15;ot,reset;0"
 ',

Announce16:
 'Befehl &HF0<n><m>
 'liest announcements
 'read m announcement lines
 Data "240;ln,ANNOUNCEMENTS;145;21"
 ',

Announce17:
 'Befehl &HFC
 'Liest letzten Fehler
 'read last error
 Data "252;aa,LAST ERROR;20,last_error"
 ',

Announce18:
 'Befehl &HFD
 'Geraet aktiv Antwort
 'Life signal
 Data "253;aa,MYC INFO;b,ACTIVE"
 ',

Announce19:
 'Befehl &HFE <n><data>
 'eigene Individualisierung schreiben

```
'write individualization
Data "254;ka,INDIVIDUALIZATION;20,NAME,Device 1;b,NUMBER,1;a,I2C,1;b,ADRESS,22,
{0 to 127};a,SERIAL,1"
'
```

Announce20:

```
'Befehl &HFF <n>
```

'eigene Individualisierung lesen

```
'read individualization
```

```
Data "255;la,INDIVIDUALIZATION;20,NAME,Device 1;b,NUMBER,1;a,I2C,1;b,ADRESS,22,{0
to 127};a,SERIAL,1;b,BAUDRATE,0,{19200};3,NUMBER_OF_BITS,8n1"
'
```

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.

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

„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.

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.

Software

Die Steuerung übernimmt ein AVR Mikrocontroller Atmega168 oder größer.

Das aktuelle Bascom Programm verwendet einen Atmega328P.

Die Software wurde in BASCOM geschrieben [2]

Um das Programm zu kompilieren, muss das Verzeichnis common_1.10 [6] in das Verzeichnis mit dem Programm kopiert werden

Programmierung des Prozessors

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

Die Fuses müssen möglicherweise programmiert werden (siehe Bascom Programm) !! Prozessortyp und Frequenz müssen gegebenenfalls angepasst werden.

Der Jumper J1 sollte während der Programmierung entfernt werden.

Die Programmierung darf nur mit entferntem Sensor durchgeführt werden. Da die Programmierung meist mit 5V erfolgt, würde der Sensor zerstört!!

Serielle (RS232 / USB) Schnittstelle

Schnittstellenparameter: 19k2 8N1.

Alternativ (große Version) 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. Die Stromversorgung erfolgt dann über USB.

I2C

Die I2C Schnittstelle ist nur auf der großen Leiterplatte vorhanden. Die Default Adresse ist 22 (&H16).

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

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

Da der Prozessor mit 3.3V arbeitet, muss auch der interne I2C Bus mit dieser Spannung arbeiten.

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

Wenn auf den 5V Betrieb völlig verzichtet werden soll, kann IC3 (PCA9517), R1, R2, R5, R6, JP2 entfallen und alternativ wird JP3 und JP4 bestückt.

Ganz ohne I2C kann auch SL1, SL2, JP2, JP3, JP4, R1, R2, R5, R6 entfallen.

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

Mit IC3 muss R1 / R2 ($\leq 10k$) bestückt werden.

SL1 und SL2 sind parallel geschaltet. Ein Anschluss kann zur Weitergabe des I2C Signals an das nächste Gerät verwendet werden.

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.

Bei Bestückung mit der USB Schnittstelle muss die Stromversorgung darüber angeschlossen werden, auch wenn nur I2C verwendet werden soll.

SMD

Die Leiterplatte ist teilweise mit SMD bestückt.

Stromversorgung

Die Stromversorgung ist 12V +-10%, Stromaufnahme ca. 20mA ohne USB oder über USB.

Prozessor und BME280 werden mit 3.3V versorgt. Der Quarz darf daher nicht mehr als 10MHz haben.

Bestückung der Leiterplatte

„große“ Version:

Verwendung von ISP:

JP6

Mit I2C:
Siehe I2C oben.

Mit serieller Schnittstelle:

Bei Verwendung der RS232 Schnittstelle wird IC2, IC4 und C6 – C9 bestückt. Alternativ dazu kann der USB Modul UM2102 verwendet werden. Dann darf IC2 und DC1 nicht bestückt werden! X1, D1 und C1, R8, R9 braucht auch nicht bestückt werden.

Der USB Modul wird isoliert auf der Bestückungsseite angebracht,.
Der Sensormodul wird mit einer Stifteleiste versehen. Die 6 polige Buchsenleiste ist auf der Oberseite. Nach dem Aufstecken des Sensormoduls liegt der Modul zur Leiterplattenmitte, der Sensor liegt außen / oben.

„kleine“ Version:

alle Teile müssen bestückt werden.

Der Quarz muss eine kleine Version sein (HC48/U-S).

Anschlüsse

Power

Tip 12V

Ring GND

I2C (SL1, SL2)

1 GND

2 SCL

3 SDA

Jumper

JP1 Power

JP2 I2C: 3V/5V Umschaltung

JP3 SDA Überbrückung (ohne IC3)

JP4 SCL Überbrückung (ohne IC3)

JP5 Reset

JP6 ISP

JP7/JP8 Anschluss für USB Modul

JP9/JP10 Klimasensor

Versionen

Diese Beschreibung gilt für die

Leiterplattenversion V02.1

Bascom Version V03.3

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/Klimasensor_eagle.zip
- [2] dk1ri.de/dhw/Klimasensor_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_1.10.zip