

MYC Realtime Clock mit DCF

Author DK1RI, Version V03.1, 20200429

This project can be found in <https://www.github.com/dk1ri> also.

Einleitung

Dieses Interface ist bindet das RTC DCF Modul von ELV in das MYC System ein.

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 arbeitet als Slave am I2C Bus oder kann über RS232 / USB gesteuert werden. Defaultmäßig sind alle Schnittstellen aktiv. Mit dem Initialisierungsbefehl können diese aber deaktiviert werden. Der Initialisierungsbefehl funktioniert aber immer.

Beschreibung und Bedienung

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

Die Steuerung kann über I2C, USB oder RS232 erfolgen.

Als Realtime Modul wird der RTC-DCF Modul von ELV verwendet.

Das Modul wird als Arduino Shield verwendet, dadurch wird die Leiterplatte zwar etwas größer, aber so entfällt die Pegelwandlung.

Die Kommunikation mit dem DCF Modul erfolgt über die SPI Schnittstelle.

Die MYC Befehle müssen binär über die I2C Schnittstelle oder über USB eingegeben werden.

Defaultmäßig wird der DCF Empfang eingeschaltet und die LED leuchtet, sobald die Uhrzeit gültig ist. Es dauert einige Zeit, bis die Uhrzeit richtig empfangen wird..

Die Uhrzeit wird im Format 8 Byte Unixzeit in Sekunden nach dem 1. 1. 1970 angegeben; allerdings nicht in UTC sondern in ME(S)Z angegeben. Es werden also 3600 (7200) Sekunden mehr angezeigt als die „wahre“ (UTC) Unixzeit.

Mit dem Befehl &H02<n><m>, &H3<n> können die 15 Register des Moduls beschrieben / gelesen werden. Details stehen in der Dokumentation des Moduls. Normalerweise werden diese Funktionen nicht gebraucht.

Einbindung in das MYC System

Details zum MYC System stehen in [3].

Folgende Befehle werden akzeptiert:

Announce:

'Befehl &H00

'eigenes basic announcement lesen

'basic announcement is read to I2C or output

Data "0;m;DK1RI;RTC;V03.2;1;145;1;9;1-1"

,

'Announce1:

'Befehl &H01

'liest Uhrzeit

'read time

Data "1;aa,read time;t"

,

'Announce2:

'Befehl &H02 <0..14><m>

'schreibt register

'write register

Data "2;om,write register;b;15"

,

'Announce3:

'Befehl &H03 <0..14>

'liest register

'read register

Data "3;am,,as2"

,

'Announce4:

'Befehl &HF0<n><m>

'liest announcements

'read n announcement lines

Data "240;ln,ANNOUNCEMENTS;145;9"

,

'Announce5:

'Befehl &HFC

'Liest letzten Fehler

'read last error

Data "252;aa,LAST ERROR;20,last_error"

,

'Announce6:

'Befehl &HFD

'Geraet aktiv Antwort

'Life signal

Data "253;aa,MYC INFO;b,ACTIVE"

,

'Announce7:

'Befehl &HFE :

'eigene Individualisierung schreiben

'write individualization

Data "254;ka,INDIVIDUALIZATION;20,NAME,Device 1;b,NUMBER,1;a,I2C,1;b,ADRESS,19,
{0 to 127};a,SERIAL,1"

,

'Announce8:

'Befehl &HFF :

'eigene Individualisierung lesen

'read individualization

Data "255;la,INDIVIDUALIZATION;20,NAME,Device 1;b,NUMBER,1;a,I2C,1;b,ADRESS,19,{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 Atmega8 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.

Serielle (RS232 / USB) Schnittstelle

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 mit JP7 / JP8 verlötet: der USB Stecker zeigt zum Rand. USB Platine und Interface müssen voneinander isoliert werden.

Die Stromversorgung erfolgt dann über USB.

I2C Schnittstelle

Die Default Adresse ist 19 (&H13)

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, JP2 entfallen und alternativ wird JP3 und JP4 bestückt.

Ganz ohne I2C kann auch SL1, SL2, JP3, JP4 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. 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.

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 max.

Alternativ erfolgt die Stromversorgung über USB

RTC-DCF Modul

Da der Modul mit SPI gesteuert wird, brauchen die Steckerleisten ST2 und ST4 weder beim Modul noch beim Interface bestückt werden.

Auf dem Modul müssen die 4 Lötbrücken für SPI überbrückt werden.

Der Schalter des Moduls muss auf 10xxxxxx (S1 – S8) - SPI mode - eingestellt werden.

Bestückung der Leiterplatte

Verwendung von ISP:

JP6

Mit I2C:

Siehe I2C oben.

Mit serieller Schnittstelle:

Bei Verwendung der RS232 Schnittstelle alle Bauteile bestückt. Alternativ dazu kann der USB Modul UM2102 verwendet werden. Dann darf IC2, IC4 , X1, X2 und DC1 nicht bestückt werden!

D1 und C1, C6 – C9, C11, C12, L1, L2 braucht auch nicht bestückt werden.

Die Buchsen ST1 – ST4 nehmen das RTC Modul auf und sollten daher passende Höhe haben.

ST2 und ST4 werden (elektrisch) nicht benötigt.

JP1 muss für Normalbetrieb überbrückt werden.

Anschlüsse

Power

Tip 12V

Ring GND

RS232 (Buchse)

5 GND

2 TX (PC Eingang)

3 RX (PC Ausgang)

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

Versionen

Diese Beschreibung gilt für die

Leiterplattenversion V03.1

Bascom Version V03.2

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.

Referenzen

[1] dk1ri.de/dhw/rtc_eagle.zip

[2] dk1ri.de/dhw/rtc_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/myc/common_1.10.zip