

MYC Realtime Clock mit DCF

Author DK1RI, Version V02.1, 20180118

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.

Das Interface arbeitet als Slave am I2C Bus oder mit USB Interface.

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

Beschreibung

Die Eagle Daten der Leiterplatte stehen unter [1].

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 Stromversorgung ist 12V, Stromaufnahme ca. 20mA mA oder über USB

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.

Die Details zu den Registern des DCF Moduls müssen in der Original Dokumentation des Moduls nachgelesen 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.. Der Befehl &HFD liefert solange eine 1. Wenn die Zeit stimmt, ist die Antwort 4.

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.

Befehle

Zu Details zum MYC Protokoll und zur Bedienung siehe [3] und [4] (aktuell).

FolgendtProgramm:

```
'announce text
```

```
,
```

```
Announce0:
```

```
'Befehl &H00
```

```
'eigenes basic announcement lesen
```

```
'basic announcement is read to I2C or output
```

```
Data "0;m;DK1RI;RTC;V02.0;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,read register;b;15"

,

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,RS232,1;a,USB,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,RS232,1;b,BAUDRATE,0, {19200};3,NUMBER_OF_BITS,8n1;a,USB,1"

,

I2C

Die Default Adresse ist 19 (&H13)

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

Pullup Widerstände für den I2C Bus (R8/R9) können bei Bedarf bestückt werden. Der

Gesamtwiderstand am Bus sollte zwischen 1 und 10 kOhm liegen.

Wenn Geräte am I2C Bus nur 3.3V vertragen (zB der Raspberry), dürfen die Pullup Widerstände nicht bestückt werden.

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.

Reset

Ist der Reset Jumper JP4 beim Anlegen der Versorgungsspannung überbrückt, werden wieder die Defaultwerte eingelesen. Dies ist hilfreich, wenn die aktuelle I2C Adresse verloren gegangen ist. Der Reset hat eine Verzögerungszeit von 1 Sekunde.

Watchdog

Die Befehlseingabe und Ausführung muss in weniger als 1 Sekunde 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 Lesebefehl müssen die Daten innerhalb von 10 Sekunden vom I2C Master abgeholt werden – wenn die I2C Schnittstelle gerade verwendet wird. Danach werden die Daten gelöscht.

Diese Zeit kann mit dem Wert Tx_factor im Bascom Programm geändert werden. Neue Befehle können erst eingegeben werden, wenn alle Daten abgeholt wurden. Wird die RS232 / USB Schnittstelle verwendet, werden die Daten sofort ausgegeben.

Es gibt einen kompletten Reset, wenn die Hauptschleife länger als 2 Sekunde dauert, zum Beispiel, wenn die I2C Schnittstelle nicht korrekt arbeitet.

Software

Die Steuerung übernimmt ein AVR Mikrocontroller Atmega8 oder größer.

Die Software wurde in BASCOM geschrieben [2].

Programmierung des Prozessors

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

Um der Prozessor von der Stromversorgung der übrigen Schaltung zu trennen, muss der Jumper JP1 entfernt werden.

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

USB Schnittstelle

Der USB Modul UM2102 wird plan auf der Rückseite mit JP3 /JP6 verlötet.

USB Platine und Interface müssen voneinander isoliert werden.

Die Stromversorgung erfolgt dann über USB.

X3 kann nur bestückt werden, wenn der USB Modul mit Abstand auf der Rückseite verlötet wird (wird nicht empfohlen).

SMD

Die Leiterplatte ist teilweise mit SMD bestückt. Bei den nötigen Bauteilen sind das aber nur relativ großen Bauteile (1206 und grösser).

Stromversorgung

Die Stromversorgung ist 12V, Stromaufnahme ca. 20mA max oder über den USB Modul.

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

Die Buchsen ST1 –und ST3 nehmen das RTC Modul auf und sollten daher passende Höhe haben. Möglicherweise muss ein weiterer Sockel aufgesteckt werden, da die Stromversorgungsbuchse das höchste Bauteil ist.

ST2 und ST4 werden (elektrisch) nicht benötigt.

JP1 muss für Normalbetrieb überbrückt werden.

JP5 ist nur bei Verwendung von ISP nötig

JP2 ist optional und erleichtert Tests.

Bei Verwendung der USB Moduls wird *nicht* bestückt:

X1, D1, L1, L2 C1, C8, DC1, X3 (5V und 3,3V liefert der USB Modul).

Bei Montage der Antenne in der Originalposition ist möglicherweise kein Empfang des DCF77 Signals möglich (Die rote LED des RTC Moduls muss nach einiger Zeit leuchten). Die Befestigung der Antenne in einiger Entfernung hilft.

Anschlüsse

Power

Tip 12V

Ring GND

I2C (2 x 3,5mm Klinke Stereo)

Sleeve GND

Ring SDA

Tip SCL

Versionen

Diese Beschreibung gilt für die
Leiterplattenversion V01.2
Bascom Version V02.0

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 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.zip
- [3] dk1ri.de/myc/MYC.pdf
- [4] dk1ri.de/myc/Description.pdf (englisch)
- [5] dk1ri.de/myc/Definitions.pdf (englisch)