

## DTMF Empfänger

Author: DK1RI

Version V03.0 20190806

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

### Einleitung

*Da das IC MT8870 kaum noch lieferbar ist, wird dieses Board nicht weiterentwickelt.  
Der Empfänger mit Software Decoder ist in Planung.*

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.

Dieses Interface ist eine Konverter von DTMF Signalen auf MYC Protokoll über I2C (Slave) / RS232 / USB

Im „Nicht-MYC“ Mode arbeitet das Interface als einfacher DTMF - ASCII Wandler.

### Beschreibung

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

Die Stromversorgung ist 7- 15V, Stromaufnahme ca. 20 max.

Die Steuerung kann über I2C, USB oder RS232 erfolgen.

Die Befehle zur Steuerung müssen binär eingegeben werden; er erfolgt keine Wandlung.

Die DTMF-Daten werden als normale Zeichen ausgegeben.

Im MYC Mode werden die DTMF Codes als String ausgegeben, der mit der Länge des Strings beginnt.

In den Nicht-MYC Mode kommt man mit dem Befehl &HEE01. Dann werden die DTMF Codes als ASCII Zeichen auf der seriellen Schnittstelle ausgegeben.

Zur Rückkehr vom Nicht-MYC Mode in den MYC mode muss an der seriellen Schnittstelle &H20 oder an der I2C Schnittstelle &HEE00 eingegeben werden.

### Einbindung in das MYC System

Details zum MYC System stehen in [3].

Folgende Befehle werden akzeptiert:

nnounce0:

'Befehl &H00

'eigenes basic announcement lesen

'basic announcement is read to I2C or output

Data "0;m;DK1RI;DTMF receiver;V04.0;1;145;1;9;1-1"

,

Announce1:  
'Befehl &H01  
'liest den DTMF-Speicher  
'read DTMF buffer  
Data "1;aa,DTMF buffer;252,{0 to 9,\*,#,A to D}"  
'

Announce2:  
'Befehl &HEE  
'schaltet MYC / no\_MYC mode  
'switches MYC / no\_MYC mode  
Data "238;ka,no\_myc;a"  
'

Announce3:  
'Befehl &HEF  
'liest MYC / no\_MYC mode  
'read MYC / no\_MYC mode  
Data "239;la,as238"  
'

Announce4:  
'Befehl &HF0<n><m>  
'liest announcements  
'read m 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,16,{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,16,{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 Atmega8.

Die Software wurde in BASCOM geschrieben [2]

Um das Programm zu kompilieren, muss das Verzeichnis common\_1.7 [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 verlötet: der USB Stecker zeigt zum Rand. Die mittleren 4 pins des Verbinders ST2 sind mit dem 4 Anschlusspunkten JP7 auf dem Interface zu verbinden. USB Platine und Interface müssen voneinander isoliert werden.

Die Stromversorgung erfolgt dann über USB.

I2C , SPI und RS232 / USB können nicht gleichzeitig verwendet. Der Befehlsbuffer wird gelöscht, wenn die Schnittstelle gewechselt wird.

## **I2C Schnittstelle**

Die Default Adresse ist 16.

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.

Bei anderer I2C Spannung als 3V kann R5 / R6 angepasst werden.

Wenn auf den 3V 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, 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.

## **SMD**

Die Leiterplatte ist teilweise mit SMD bestückt. Bei den nötigen Bauteilen sind das aber nur relativ große Bauteile.

## **Stromversorgung**

Die Stromversorgung ist 7- 15V, Stromaufnahme ca. 20mA max.

Alternativ erfolgt die Stromversorgung über USB

## **DTMF Eingang**

Der DTMF Eingang ist kapazitiv entkoppelt und entspricht der Norm

## **Bestückung der Leiterplatte**

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 nicht bestückt werden! X1, D1 und C1 braucht auch nicht bestückt werden.

JP8, LED1, R11 sind nur für Tests und brauchen nicht bestückt werden.

R48/ R9 sollen zusammen 71,5KOhm ergeben. Eine Möglichkeit, diesen Wert zu erreichen, ist die Verwendung von 82k und 560k . Dies ergibt 71,52 KOhm

## **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

### DTMF (LP SL4)

1 GND  
3 DTMF Eingang

## **Jumper**

JP1 Power  
JP2 I2C: 3V/5V Umschaltung  
JP3 SDA Überbrückung (ohne IC3)  
JP4 SCL Überbrückung (ohne IC3)  
JP5 Reset  
JP6 ISP, pin10  
JP7 Anschluss für USB Modul

## **Versionen**

Diese Beschreibung gilt für die  
Leiterplattenversion V04.1  
Bascom Version V04.0

## **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/dtmf\\_rx\\_eagle.zip](https://www.dk1ri.de/dhw/dtmf_rx_eagle.zip)
- [2] [dk1ri.de/dhw/dtmf\\_rx\\_bascom.zip](https://www.dk1ri.de/dhw/dtmf_rx_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\\_1.7.zip](https://dk1ri.de/myc/common_1.7.zip)