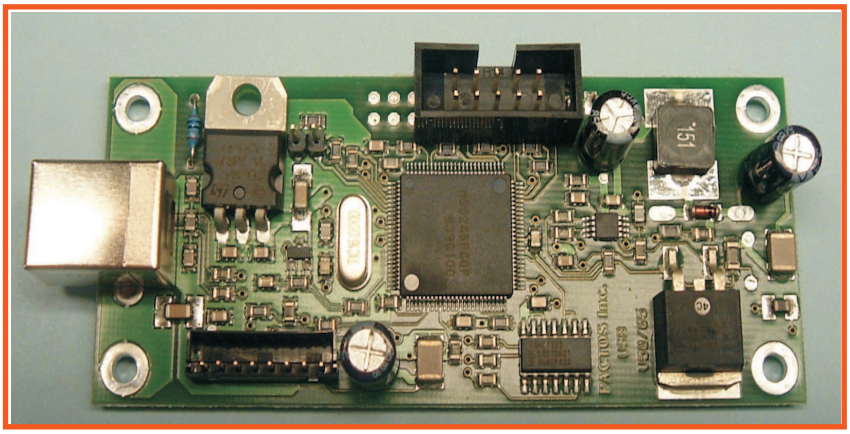
metec

INGENIEUR-AKTIENGESELLSCHAFT · MEDIZINTECHNIK · FEINWERKTECHNIK

USB-Controller



Abstract

The USB Interface connects a PC with a Braille Line assembled with metec Braille Modules (B11 or B16)
This Doku is valid for the USB-Device 01#00.

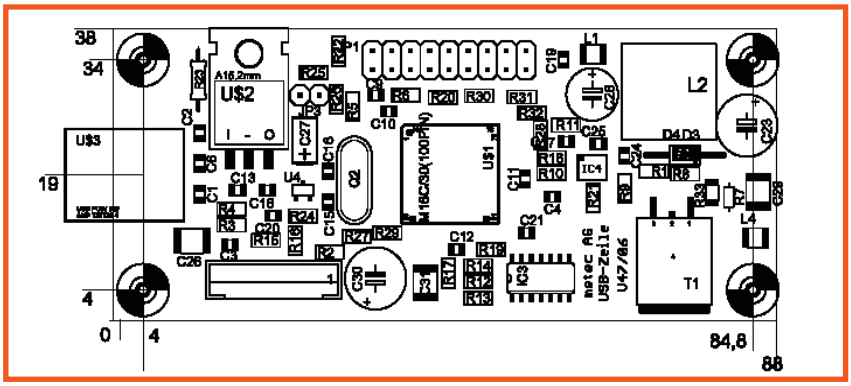
Installing the Driver

First connect the USB Interfaceto the PC. When there is no matching driver the Operating System requests one. Choose the folder where metecUSB.sys and metecUSB.inf is loacated.
The driver fits for Windows98, WindowsME, Windows2000 and WindowsXP.

Testapplication USB_tst

This application is only used to demonstrate the function. When the device is connected and the driver is loaded, the High Voltage of the device is switched on.
The message “High Voltage is On!“ is displayed.
Then you have to enter the number of modules.
In the white Line you can enter a text that is shown on the Braille Line.
In the field “CursorPosition” the Number of the switched Cursor key is shown.
If the box “Testmode” is checked the whole Line flashes between all dots on and off.
To show the use of the driver the source code of this application is also delivered. (Borland C++Builder 4.0

Dimension



E05 USB 1

METEC AG · Hasenbergstr. 31 · 70178 STUTTGART · GERMANY
TELEFON:(07 11) 66 60 3-0 · TELEFAX:(07 11) 66 60 333 · E-MAIL:contact@metec-ag.de

USB Electronic

USB Functions of the Interface

1 Starting the Communication

First the Driver Handle has to be fetched.
This is done by calling CreateFile with the Driver Name (\\.\MITSUUSB0). If there is no USB-Braille Line connected the operating system do not activate the driver. In this case CreateFile delivers INVALID_HANDLE_VALUE as result.
If the driver delivers a valid Handle you have tho use this handle to communicate with the driver via the DeviceIoControl function.
Next you have to fetch the Device Info.
(For details see source code in funktion startClick.)

2 Reading the Device ID String

This string has to be read from the Interface to check whether the device works with this application.
To read the String a Vendor Request Nr 4 with 1 data byte of the value 0 is to be sent.
As reply the device starts a bulk read on pipe 0.
(For details see source code in funktion startClick.)

3 Switch on High Voltage

To switch on the High Voltage a Vendor Request Nr 1 with 1 data byte of the value 0xef is to be sent.

4 Setting the Length of the Braille Line

Because the modules are connected together in form of a shiftregister it is important to set the correct length of the Braille Line.
This is done by sending a vendor Request 0x40 with the length in the only one byte of data.

5 Sending the Braille Pattern

In the source code the Braille Pattern of a 80 character line sends via the USB Bus using a 10ms interrupt.
Every interrupt sends a block of 8 characters.
The function WrZeileMod(int Block) do this.
To do this a Vendor Request Nr 10+ BlockNr with the 8 byte data is to be sent.
Because the max. size of the Braille Line is 80 characters Vendor Requests 10 ..17 are possible.
For the relation between the characters and the Braille Pattern the included table “chrdef.c” is used.
To adapt the Braille Pattern to the Pin Order of the modules the function HardwareChange is used.
(For details see source code in funktion Timer1Timer, WrZeileMod and HardwareChange.)
Remember that it is important for proper function that the correct length of the line is set.

METEC AG · Hasenbergstr. 31 · 70178 STUTTGART · GERMANY
TELEFON:(07 11) 66 60 3-0 · TELEFAX:(07 11) 66 60 333 · E-MAIL:contact@metec-ag.de

USB Electronic

USB Functions Keys

6 Reading the Cursor Position

The Cursor position can be readed by a vendor Request with the Number 0x80. The Requests returnes one Byte of date with the number of the pressed cursor key. The rear key comes with an offset of 100.
(For details see source code in funktion Timer1Timer.)
Remember that it is important for proper function that the correct length of the line is set.

Some additional information are coming when 8 bytes are read.
Byte 0 Routing keys
Byte 1 number of Modules (recognized by Firmware) (since Version 031215)
Byte 2 Additional Key Inputs (Bit 6=Key1, Bit 4= Key2, Bit 2=Key3)
(For 6Key Version: Bit 3=Key 4, Bit 1=Key5, Bit 0=Key6)
Keys are readed with vendor request 0x80

Byte 2Bit 0 S6 Key right down *1
Bit 1 S5 Key right center *1
Bit 2 S3 Key left down
Bit 3 S4 Key right up *1
Bit 4 S2 Key left center
Bit 5
Bit 6 S1 Key left up
Bit 7

Byte 3Bit 0
Bit 1
Bit 2 S7 Cursorkey left *2
Bit 3 S8 Cursorkey up *2
Bit 4 S9 Cursorkey right *2
Bit 5
Bit 6 S10 Cursorkey down *2
Bit 7

*1 only in 6 Key and Cursorkey version
*2 only in Cursorkey version

3 Key Version
The left and right Keys are connected by hardware. There are only 3 bits for the 6 keys. There are no cursorkeys.
Device String: "Device: <TestZeile> ….

6 Key Version
All 6 keys are represented in Bits. There are no cursorkeys.
Device String "Device: <6Key> ….

METEC AG · Hasenbergstr. 31 · 70178 STUTTGART · GERMANY
TELEFON:(07 11) 66 60 3-0 · TELEFAX:(07 11) 66 60 333 · E-MAIL:contact@metec-ag.de

USB Electronic

USB Function Pinning

Cursorkey Version:
Same as 6 Key Version but with additional Cursorkeys.
Device String: "Device: <10Key>……

3 Key Version

Pin		Pin	
1	+5V	2	GND
3	P17 (3,3V)	4	P16 (input key1)
5	P15 (3,3V)	6	P14 (input key2)
7	P13 (3,3V)	8	P12 (input key3)
9	P11	10	P10

The 3 keys connects as following:

key1	pin1	pin4
key2	pin5	pin6
key3	pin7	pin8

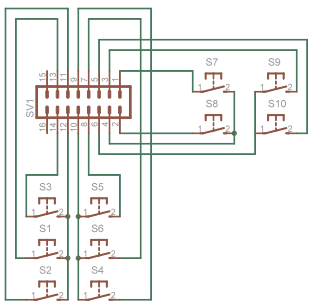
6 Key Version

Pin		Pin	
1	+5V	2	GND
3	P17 (input key3)	4	P16 (input key1)
5	P15 (3,3V)	6	P14 (input key2)
7	P13 (3,3V)	8	P12 (input key4)
9	P11 (input key5)	10	P10 (input key6)

The 6 keys connects as following:

key1	pin5	pin4
key2	pin5	pin6
key3	pin5	pin3
key4	pin7	pin8
key5	pin7	pin9
key6	pin7	pin8

Cursorkey Version



METEC AG · Hasenbergstr. 31 · 70178 STUTTGART · GERMANY
TELEFON:(07 11) 66 60 3-0 · TELEFAX:(07 11) 66 60 333 · E-MAIL:contact@metec-ag.de