

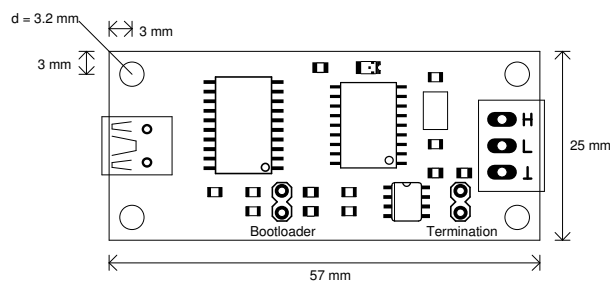
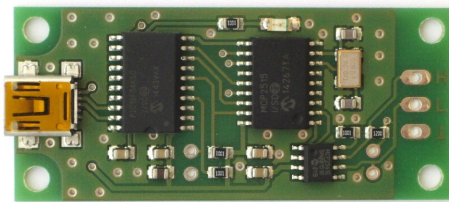
---

# USBtin EB – USB to CAN interface board

## Datasheet

---

USBtin is an open source USB to CAN interface. It can monitor CAN buses and transmit CAN messages. All common CAN baud rates are supported. USBtin implements the USB CDC class and creates a virtual COM port on the host computer. Via this serial port, the interface is controlled by means of simple commands.



### Main features

- USB powered (no isolation between USB and CAN ground)
- Control over any serial terminal program with simple serial protocol
- Presets for common baud rates (10k ... 1M)
- Timestamping of incoming messages
- Listen-only and active mode
- CAN ID filtering, e.g. for high-loaded busses
- 120 Ohm termination via jumper
- Open source GUI and Java Library available

## Connection

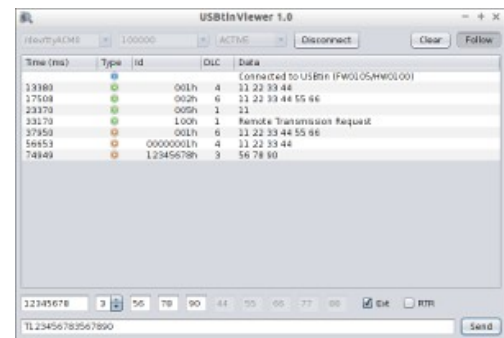
USBtin EB has a mini-USB connector for communication with the host computer via a standard mini-USB cable. The CAN bus is routed to three pins (CON2): CAN-H, CAN-L and ground. The spacing of these three pins is 3.5mm and fits for standard RM3.5 connectors, e.g. Reichelt „AKL 182-03“ or Conrad „729906-62“.

## Driver installation

On Linux, MacOS and Windows >8 systems there is no need for drivers, the CDC is automatically detected and mapped to an virtual com port „/dev/ttyACMx“ or „COMx“. For Windows <8, there is a driver file available (<http://www.fischl.de/usbtin/#drivers>).

## Software

USBtin is controlled by means of simple commands via the virtual serial port. This commands can be sent with any serial terminal program. There is also USBtinViewer, a simple multi-platform GUI. For own developments, there exists USBtinLib, an open source Java Library. Further information about the software and a list of serial commands can be found on <http://www.fischl.de/usbtin/>.



## Termination

120 Ohm termination can be activated via the termination jumper (JP2, 2.54mm) or via the solder jumper on the bottom of the board (JP3).

## LED signaling

Flashing: USBtin is idle, in configuration mode and can channel is closed.  
Permanent on: CAN channel is open.

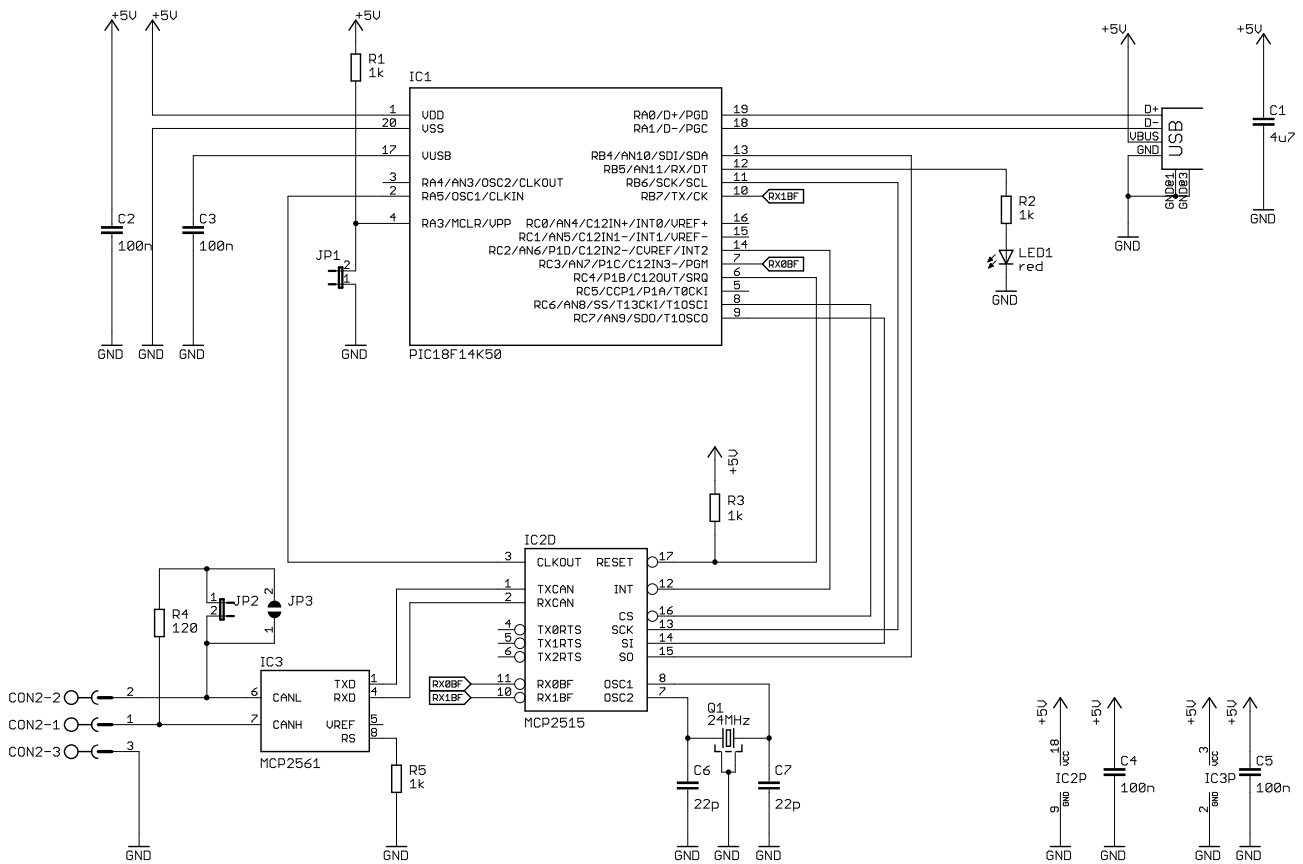
## Firmware update

1. Short bootloader jumper (JP1, 2.54mm) and connect USBtin to host USB
2. Download bootloader tool „mhidflash“ from <https://code.google.com/p/mhidflash/>
3. Start firmware upload with command line command:  
mhidflash -write USBtin\_firmware\_vX.X.hex -reset
4. Unplug USBtin from host and remove bootloader jumper (JP1)

## Throughput

The serial protocol limits the bandwidth that can be used by USBtin to send and receive CAN messages. Firmware v1.5 was tested successfully receiving 100% load on 250 kBaud bus (~2000 can messages per second). Testconditions: Standard 11bit id, 8 data bytes, timestamping disabled. Testsystem: Lubuntu 14.04 on Intel(R) Core(TM)2 Duo CPU E8400@3.00GHz with 4 GB ram.

Schematics diagram



Evaluation board notice

USBtin EB is an evaluation board intended for use for engineering development or evaluation purposes in laboratory environment only. It is not considered as an end user application. If you intend to use it in an end-product, you have to ensure in your own responsibility that the device meets the relevant regulations (e.g. CE, FCC).