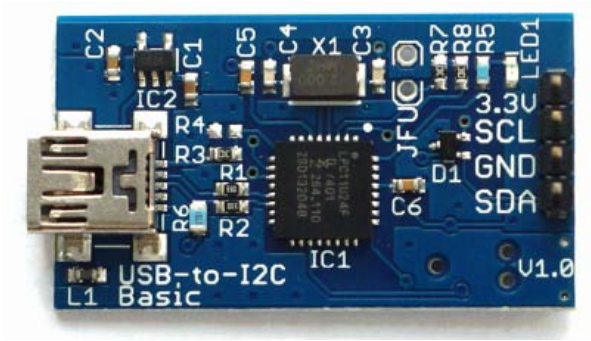
USB-to-I2C© Basic

Hardware User's Manual



http://www.i2ctools.com/

Table of Contents

INTRODUCTION

The USB-to-I2C Basic Hardware connects to a standard USB port found on most Personal Computers and provides bi-directional communication with I^2C devices using the I^2C protocol. The Hardware is powered directly from the PC's USB port.

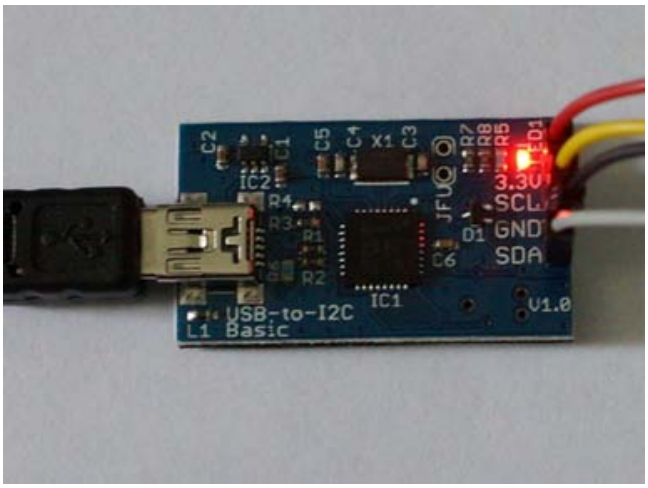


Figure 1 USB Link Indicator LED

The LED, located beside the connector, illuminates after it has successfully been enumerated by the USB host. This is shown in Figure 1.

The USB-to-I2C software runs on 32-bit and 64-bit versions of Windows XP, Vista, Windows 7, and Windows 8 and is compatible with any PC hardware having a minimum of a Pentium processor and an USB port. The software provides several Universal modes to allow communication with most I^2C devices. The USB-to-I2C software is designed to operate only with the included USB-to-I2C Basic hardware.

USB-TO-I2C BASIC KIT CONTENTS

- USB-to-I2C Basic Hardware
- USB-to-I2C software installation download includes the following files:
  - Software license agreement
  - USB-to-I2C Basic Registration Form
  - USB-to-I2C Basic application
  - USB-to-I2C Software User's Manual
  - USB-to-I2C Basic Hardware User's Manual
  - I^2C examples

DISCLAIMER AND WARRANTY

Proper use of USB-to-I2C is the sole responsibility of the user. SB Solutions, Inc. is not responsible for any damage resulting from misuse or improper installation.

SB Solutions, Inc. will, at our option, repair or replace a defective USB-to-I2C hardware within thirty (30) days of the purchase date. Return shipping is the responsibility of the user.

GETTING STARTED

Assumptions

We are assuming the user of this product has experience with the I^2C Bus protocol. The I^2C Bus specification is a good source of detailed information about the I^2C Bus. The complete specification can be downloaded from the NXP Semiconductors website.

Static Handling Precautions

The USB-to-I2C Basic hardware contains devices that can be damaged by ESD. It is recommended to use a ground strap or touching the PC case or other grounded source before unpacking or handling the USB-to-I2C Basic hardware.

MINIMUM SYSYEM REQUIREMENTS

- ✓ PC Pentium 60 processor (or equivalent), 8 MB RAM, and 16 MB of hard drive space
- ✓ One USB port (either 3.0, 2.0 or 1.1 compatible)
- ✓ 32-bit or 64-bit versions of XP, Vista, Windows 7, and Windows 8
- ✓ Internet access for downloading software: http://www.i2ctools.com/downloads.html

ADAPTER POWER REQUIREMENTS

The USB-to-I2C Basic hardware obtains its power from the computer's USB port. It should not be powered by an external power source.

INSTALLATION

USB-to-I2C Software

- Download the software installation file from the i2ctools.com website
- Unzip the file into a directory of your choice
- Double-click the unzipped installation file
- The installation software should automatically start
- Complete and send in the registration form via email

USB-to-I2C Hardware connection to computer USB port

- Neutralize any ESD (static charge) by touching the bare metal on the rear of your computer before removing the USB-to-I2C Hardware from the packaging
- Connect the USB-to-I2C Hardware to a USB Port using a standard USB cable
- The LED beside the USB port should illuminate

HARDWARE DESCRIPTION

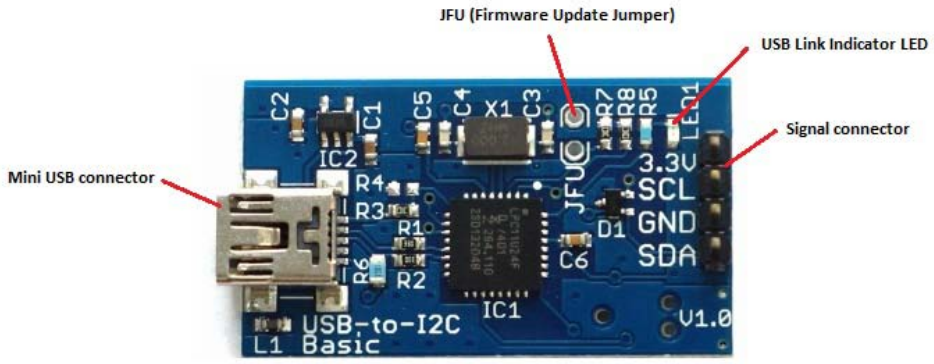


Figure 2 Hardware Description

The figure above shows the items on the hardware which you should become familiar with. Here are the details:

USB Port – this is where you plug the USB cable into the board. A cable with a mini USB connector is required.

JFU (firmware updated jumper) – short the two pins in the JFU to initiate a firmware update. The complete update procedure is detailed later in this manual.

USB LINK INDICATOR LED – the LED will illuminate when the cable has been connected between the PC and the USB-to-I2C Basic hardware.

Signal Connector

The Signal Connector contains the connections to a user target application.

Signal	Pin #	Function
3.3V	1	3.3V Supply Voltage Output (not required)
SCL	2	I2C clock
GND	3	Ground
SDA	4	I2C Data

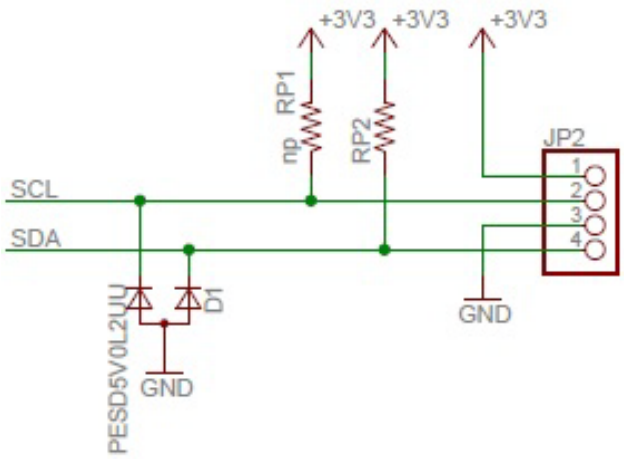


Figure 3 PCB connections

3.3V is the 3.3V power for the USB-to-I2C hardware. This supply is active whenever the hardware is plugged into a USB port. It is provided here to provide power to the pull-up resistors, if soldered to the board. It can also be used to power a user's target hardware. Do not connect this pin to your hardware if your hardware already has a power source.

I^2C Port – connect these two pins (SDA and SCL) and GND to your target system. Details can be found in the section titled "USB-to-I2C Hardware To Target Connection" of this document.

SCL – is the serial clock generated by the USB-to-I2C Basic hardware. Clock stretching is supported.
SDA – is the bidirectional I^2C data pin.

Pull-up resistors – although there are no pull-up resistors soldered on the USB-to-I2C Basic hardware, there are solder pads provided on the bottom side of the PCB (RP1 and RP2) if you desire to have these available on the hardware.
The resistors have not been placed on the PCB since most target boards contain pull-up resistors. Removing the pull-up resistors from the PCB also allow you to use a pull-up voltage that matches your hardware requirements, rather than being restricted to the 3.3V available on the USB-to-I2C Basic hardware.

Protection circuit –the USB-to-I2C Basic hardware incorporates ESD protection diodes connected to the SDA and SCL lines, but it does not have series protection resistors. Because Fast-mode plus allows a relatively high sink current on the I2C lines, adding series resistors would have caused a high offset voltage, so it was decided to eliminate these resistors. Additional care should be exercised when handling the hardware due to the elimination of the series protection resistors.

USB-TO-I2C BASIC HARDWARE TO TARGET CONNECTION

When connecting the USB-to-I2C Basic to user a user target system, a minimum of three (3) wires will be required. The SDA, SCL, and GND on the USB-to-I2C Basic hardware must be connected to their corresponding signals on the target.
The USB-to-I2C hardware does not have pull-up resistors, so these resistors need to available on the target hardware, or they need to be soldered to the bottom side of the PCB.

The SDA and SCL lines may be pulled up, through resistors, to a voltage between 3.3V and 5.5V. It may work at voltages down to 2.7V; however, the noise margins will be lower.

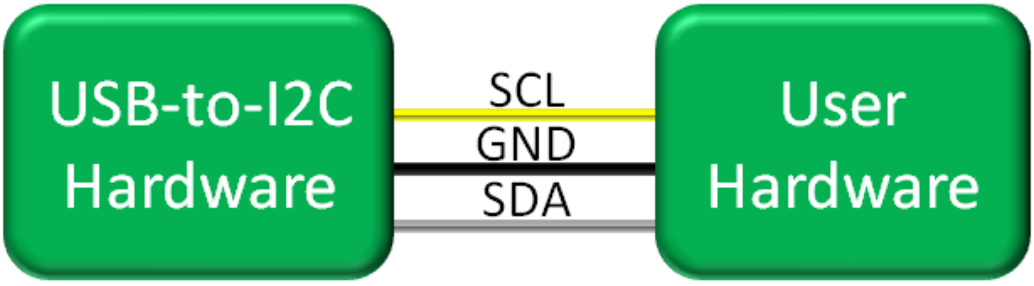


Figure 4 Minimum hardware connection between USB-to-I2C Basic and User Hardware

If you would like the USB-to-I2C Basic hardware to power your 3.3V target hardware, then you can connect the 3.3V pin on the header to your hardware.
The current consumption of the target hardware should be limited to 80mA or less.

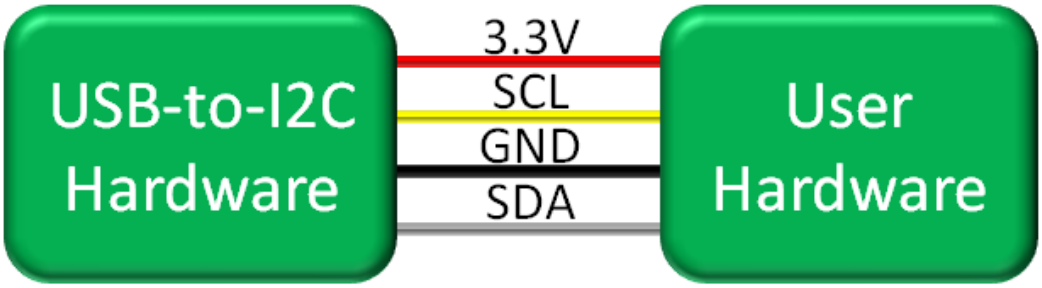


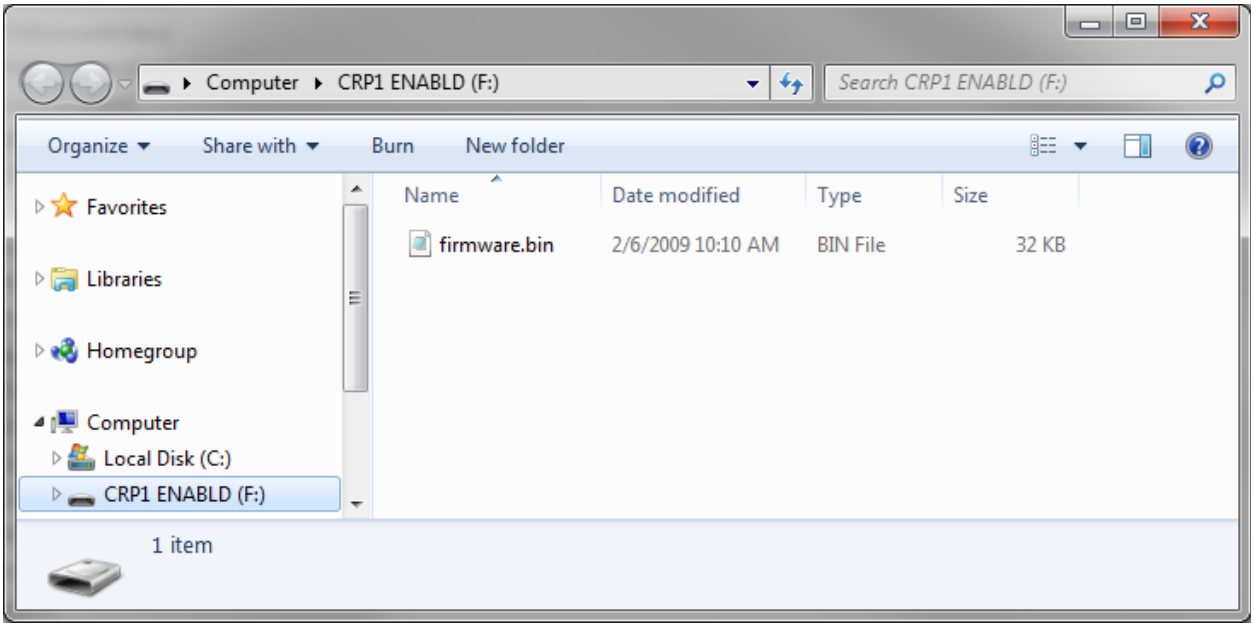
Figure 5 Connections between USB-to-I2C Basic and User Hardware, where power is provided to the User Hardware

FIRMWARE UPDATES

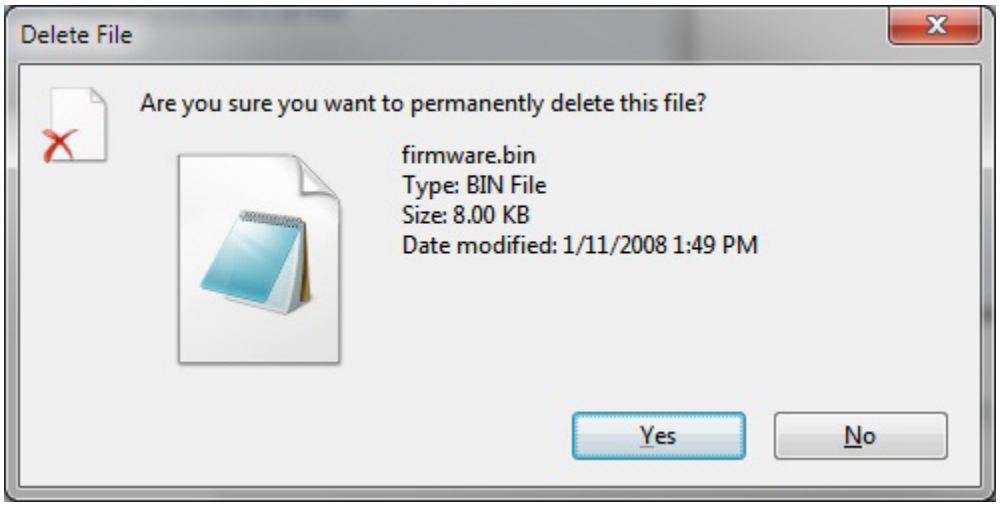
The firmware in the USB-to-I2C hardware can be updated by the user. No special software is required for the update.
The firmware can be updated by using the following procedure.

1. Disconnect the USB-to-I2C Basic hardware from the USB port. It should also be disconnected from any user target hardware.
2. Place a jumper across the bottom two pins of jumper JFU (see figure in the Hardware Description section of this document).
3. Plug the USB-to-I2C Basic hardware into the computer's USB port.

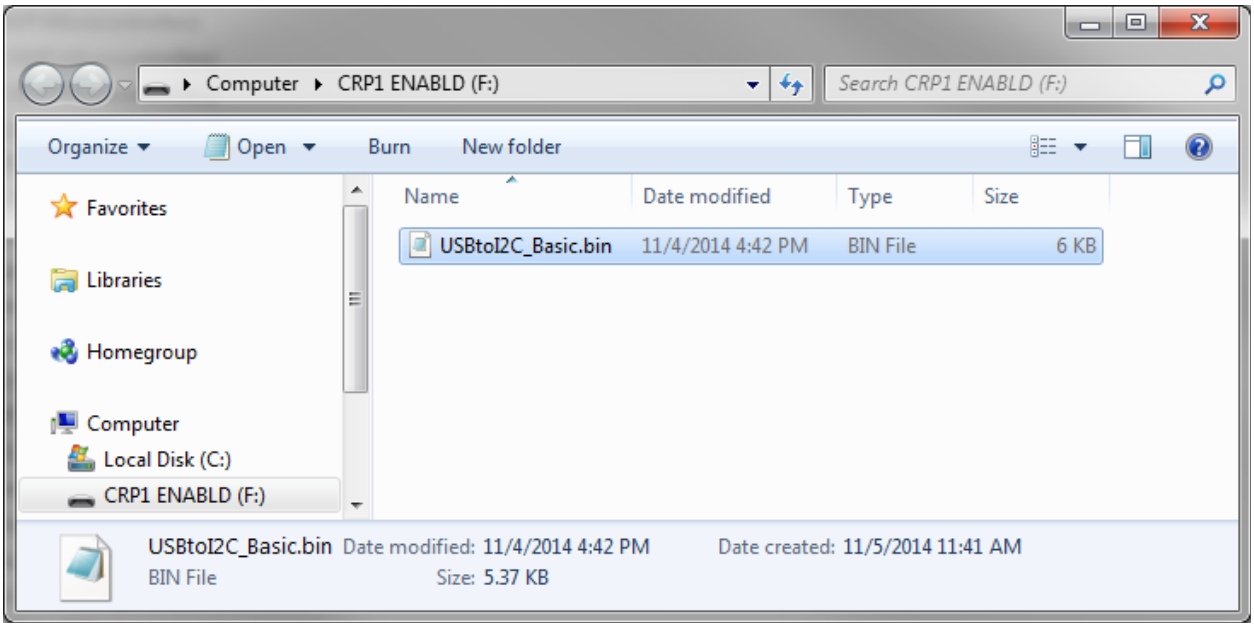
4. The USB-to-I2C Basic hardware will be displayed as a mass storage device in Windows, as shown below. Note that the contents of CRP1 will always be displayed as “firmware.bin”.



5. Select the firmware.bin file and delete it. You will be asked if you would like to delete the file. Choose Yes.

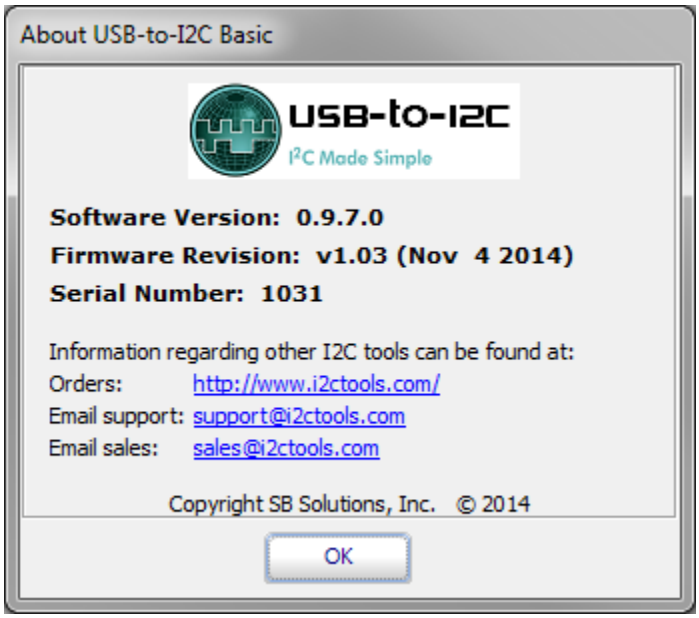


6. Place the new firmware update into the folder.



7. Disconnect the USB-to-I2C hardware from the USB port.
8. Remove the jumper from JFU.
9. Plug the hardware back into the PC. It should now contain the new firmware.

You can now use the USB-to-I2C hardware with updated firmware. You can check to see which firmware revision is loaded by clicking on About in the Help menu in the USB-to-I2C Basic software.



SOFTWARE UPDATES

The USB-to-I2C software has a built-in update feature which allows you to check for updates whenever your PC has an internet connection. This feature can be found under the Options menu. USB-to-I2C updates can also be downloaded manually from the following website: http://www.i2ctools.com/manuals-and-downloads/
Check our website periodically for update announcements and information.

TECHNICAL SUPPORT

Technical Support for USB-to-I2C is available via an email to support@i2ctools.com.

TROUBLESHOOTING

- The USB-to-I2C Basic hardware does not have pull-up resistors connected to the I²C bus lines. The on-board pull-up resistors RP1 and RP2 can be soldered to the PCB, but the pull-up voltage is limited to 3.3V.

- USB-to-I2C monitors the communications on the I²C bus for proper operation of connected peripherals; any errors on the bus are detected and reported by the software. Bus communication is stopped if errors are detected and can be resumed when the (hardware) problem is corrected and the transmission retried.

- If you get a Timeout message when using the software, make sure that there is a pull-up connected to the SDA and SCL pins. The Timeout occurs when either, or both, of the I2C pins are held low for more than 1.0 seconds. If pull-ups are connected, remove the connection to the target system. Reset the hardware by removing the USB cable and reinserting it. If you still get a timeout error, then the microcontroller may have been damaged. Contact support@i2ctools.com for further assistance.

If all else fails, email a description of the problem you are having to us at support@i2ctools.com.
Note that all technical support requests must begin with an email to this email address.