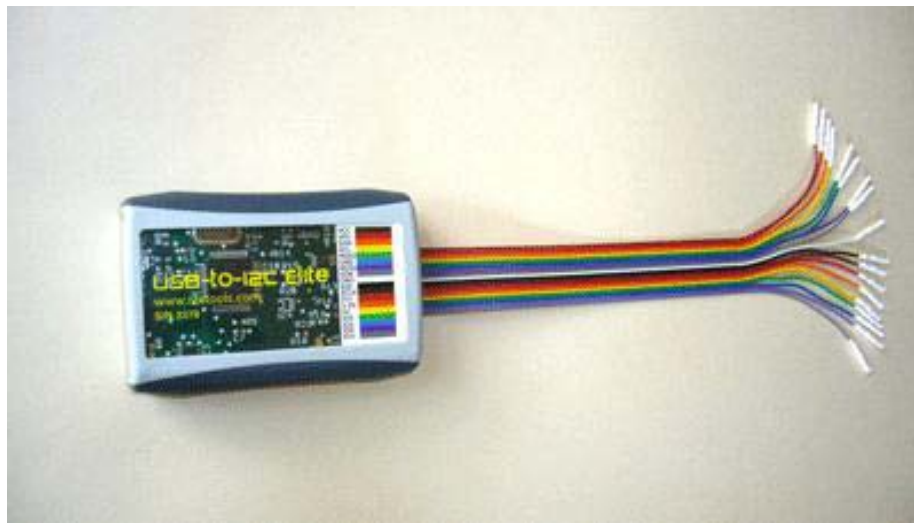


# USB-to-I2C Elite<sup>®</sup>

## Hardware User's Manual



<http://www.i2ctools.com/>

Information provided in this document is solely for use with the USB-to-I2C Elite product from SB Solutions, Inc. SB Solutions, Inc. reserves the right to make changes or improvements to this document at any time without notice. SB Solutions, Inc. assumes no liability whatsoever in the sale or use of this product, including infringement of any patent or copyright.

Reasonable efforts have been made to ensure the accuracy of the information presented. However, SB Solutions, Inc. assumes no responsibility for the accuracy of the information.

No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of SB Solutions, Inc., P.O. Box 4506, Cary, NC USA 27519-4506.

Microsoft and Windows are registered trademarks of Microsoft Corporation.

Delphi and C++ Builder are trademarks of Embarcadero Technologies.

Other brand names are trademarks or registered trademarks of their respective owners.

# Table of Contents

INTRODUCTION .....	1
USB-TO-I2C ELITE KIT CONTENTS .....	1
I2C AND SPI OPERATING SPEEDS .....	1
DISCLAIMER AND WARRANTY .....	1
GETTING STARTED .....	2
MINIMUM SYSEM REQUIREMENTS .....	2
HARDWARE POWER REQUIREMENTS .....	2
INSTALLATION .....	2
HARDWARE DESCRIPTION.....	3
CONNECTION TO USER SYSTEM.....	4
PROTECTION CIRCUITRY.....	7
USB-TO-I2C ELITE HARDWARE TO TARGET CONNECTION .....	8
USING USB-TO-I2C ELITE TO POWER A TARGET SYSTEM .....	9
FIRMWARE UPDATES .....	10
PCB LAYOUT .....	12
SOFTWARE UPDATES .....	12
TECHNICAL SUPPORT .....	12
TROUBLESHOOTING.....	12

## INTRODUCTION

The USB-to-I2C Elite Hardware connects to a standard USB port found on most IBM-compatible PCs and provides bi-directional communication with I<sup>2</sup>C devices using the I<sup>2</sup>C protocol. The Hardware is powered directly from the PC's USB port. The on-board Link LED illuminates after the USB host has successfully enumerated it.

The USB-to-I2C Elite software runs on Windows XP, Vista, and Windows 7, and is compatible with any PC hardware having a minimum of a Pentium processor and an USB port. Please see the USB-to-I2C Elite Software User's Manual for more detail on the software.

## USB-TO-I2C ELITE KIT CONTENTS

- ❖ USB-to-I2C Elite Hardware
- ❖ 18-pin split cable for connection to user target system
- ❖ USB cable
- ❖ Getting Started with USB-to-I2C Elite Guide (Quick Start Guide)
- ❖ CD-ROM loads the following files:
  - Software license agreement
  - USB-to-I2C Elite user Registration Form
  - USB-to-I2C Elite Software User's Manual
  - USB-to-I2C Elite Hardware User's Manual
  - USB-to-I2C Elite application
  - USB-to-SPI Elite application
  - I<sup>2</sup>C and SPI DLL
  - DLL examples for many of the most common development environments

## I2C AND SPI OPERATING SPEEDS

- ❖ I<sup>2</sup>C     maximum frequency: 1 MHz (fast mode compliant)  
          minimum frequency: 500 Hz
- ❖ SPI     maximum frequency: 15 MHz  
          minimum frequency: 5 kHz

\* the I<sup>2</sup>C specification specifies the maximum speed for fast mode as 400 kHz, however, the USB-to-I2C Elite hardware can operate at the higher 1 MHz frequency

## DISCLAIMER AND WARRANTY

Proper use of USB-to-I2C Elite 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 Elite 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<sup>2</sup>C Bus protocol. The I<sup>2</sup>C Bus specification is a good source of detailed information about the I<sup>2</sup>C Bus. The complete specification can be downloaded from the NXP Semiconductors (formerly Philips Semiconductors) website.

### Static Handling Precautions

The USB-to-I2C Elite Hardware contains CMOS 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 Elite Hardware.

## MINIMUM SYSTEM REQUIREMENTS

- ✓ Pentium processor (or equivalent), 8MB RAM, and 20MB of hard drive space
- ✓ One USB port (either 2.0 or 1.1 compatible)
- ✓ 32-bit or 64-bit versions of Windows XP, Vista, or Windows 7
- ✓ CD-ROM Drive for installation (download from website also available)

## HARDWARE POWER REQUIREMENTS

The USB-to-I2C Elite hardware obtains its power from the PC USB port.

## INSTALLATION

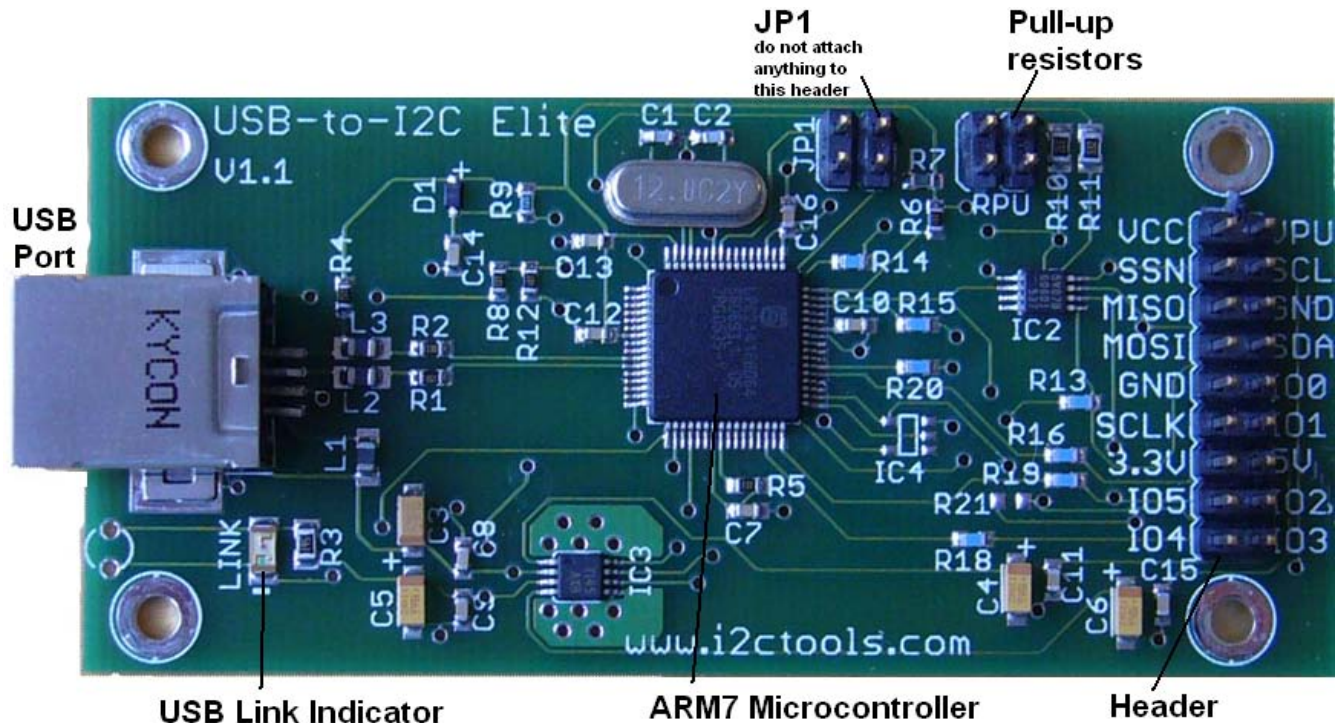
### USB-to-I2C Elite Software

- Insert the USB-to-I2C Elite installation CD-ROM into the appropriate drive.
- The installation software should automatically start
- If the installation software does not start automatically, double-click on the Setup.exe file on the CD-ROM
- Complete and send in the registration form via email

### USB-to-I2C Elite 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 Elite Hardware from the packaging
- Connect the USB-to-I2C Elite hardware to a USB Port using a standard USB cable. Do not try using the hardware until the software has been installed.
- The LED on the USB-to-I2C Elite Hardware should illuminate

## HARDWARE DESCRIPTION



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

**USB Port** – this is where you plug the usb cable into the board

**USB Link Indicator** – this LED should be illuminated when the cable has been connected between the PC and the hardware.

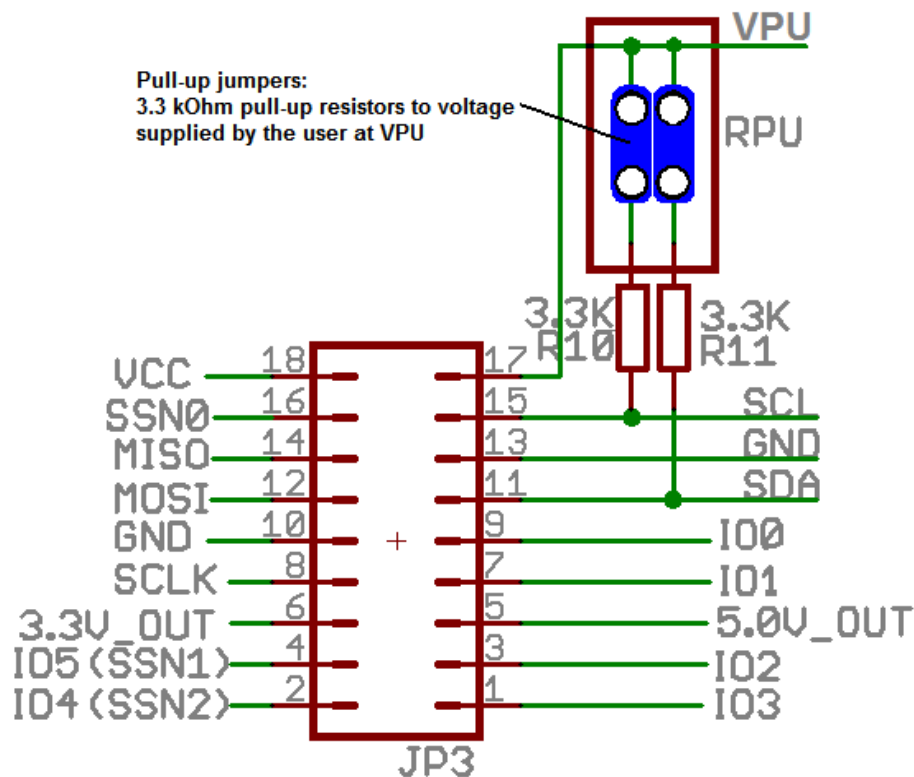
**Microcontroller** – the microcontroller is a 32-bit ARM7 device running at 60 MHz. The hardware does have an ESD protection device to reduce the possibility of damage.

**Header** – the connections to the I2C, SPI, IO, 3.3V/5V supplies are located at this header. Details can be found in the next section.

**Pull-up jumpers (RPU)** – the pull-up resistors that are located on the hardware can be disconnected from the I2C pins. The resistors are disconnected when the jumpers are removed. The jumpers should be removed if the user target system already has the minimum resistor values or if the target uses a voltage greater than 3.3V.

**Jumper J1** – these pins must be left open.

## CONNECTION TO USER SYSTEM



### JP3 Header

Header JP3 contains the connections to a user target system.

**VCC** is the 3.3V power for the USB-to-I2C Elite hardware. This supply is active whenever the hardware is plugged into a USB port. It is provided here to supply the pull-up resistors.

**VPU** – supplies the power to the I<sup>2</sup>C pull-up resistors (see RPU). The pull-up voltage should not exceed 5.5V.

**SPI Port** – the SPI signals are connected to the target system from this port. Note that these signals drive a 3.3V output.

To communicate with a SPI slave, you require five signals and Gnd:

- SSN** – active low Slave Select 0 (also known as CS0 or Chip Select)
- MISO** – Master In Slave Out is connected to the SPI slave output
- MOSI** – Master Out Slave In is connected to the SPI slave input
- SCLK** – Serial Clock outputs the clock to the slave device
- SSN1** – (optional) active low Slave Select 1 (also known as CS1 or Chip Select)
- SSN2** – (optional) active low Slave Select 2 (also known as CS2 or Chip Select)

**I<sup>2</sup>C 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 Elite Hardware to Target Connection” of this document.

**SCL** – is the serial clock generated by the USB-to-I2C Elite hardware. Clock stretching is supported, up to the I2C timeout period (approximately 1.5 seconds).

**SDA** – is the bidirectional I<sup>2</sup>C data pin.

If a jumper is connected between VCC and VP, and pull-up resistors (see RPU description below) are connected, the voltage will be 3.3V at SDA and SCL. It is advisable to remove the jumpers when connecting the hardware to a target system,

assuming the target already has pull-up resistors. This will allow you to use the USB-to-I2C Elite hardware at the voltage of your target system.

**GPIO (IO0 to IO5)** – the USB-to-I2C Elite hardware has five pins on the header that can be used for IO. Most of these IO pins have varying functions/capabilities.

- IO0** – can be configured, using software, as an input or output. The output has the ability to drive the pin but the user can also read the pin state.
- IO1** – can be configured, using software, as an input or output. The output has the ability to drive the pin but the user can also read the pin state.
- IO2** – can be configured, using software, as an input or output. This port utilizes an open-drain structure so a pull-up resistor is required when used as an output.
- IO3** – can be configured, using software, as an input or output. The output has the ability to drive the pin but the user can also read the pin state. The default state for this pin is input.
- IO4** – can only be used as an output. It also has the capability to be used as a SPI Slave Select output (SSN2). No configuration is needed to when used as IO4 or SSN2. The user just needs to be aware that if used as a Slave Select, data should not be written to this pin via software using the IO function.
- IO5** - can only be used as an output. It also has the capability to be used as a SPI Slave Select output (SSN1). No configuration is needed to decide if it is IO5 or SSN1. The user just needs to be aware that if used as a Slave Select, data should not be written to this pin via software using the IO functions.

**INPUT PORT CHARACTERISTICS**

<b>V<sub>il</sub></b> LOW-state input voltage	<b>0.8V(max)</b>
<b>V<sub>ih</sub></b> HIGH-state input voltage	<b>2.0V(min)</b>

Note that the microcontroller is a 3.3V device but it is 5V tolerant when used as an input or when it uses an open-drain output structure.

**OUTPUT PORT CHARACTERISTICS**

<b>V<sub>oh</sub></b> HIGH-state output Voltage (typical @ 4mA)	<b>V<sub>dd</sub>-0.4V</b>
<b>V<sub>ol</sub></b> LOW-state output Voltage (typical @ 4mA)	<b>0.4V</b>

**3.3V Switched Output** – provides the user with the ability to power a low power target system at 3.3V. The USB-to-I2C Elite software has the ability to switch this output on and off. The total current provided by the USB-to-I2C Elite Hardware to the target system should not exceed 20mA. It is the user’s responsibility to ensure that excessive current is not drawn from this supply pin.

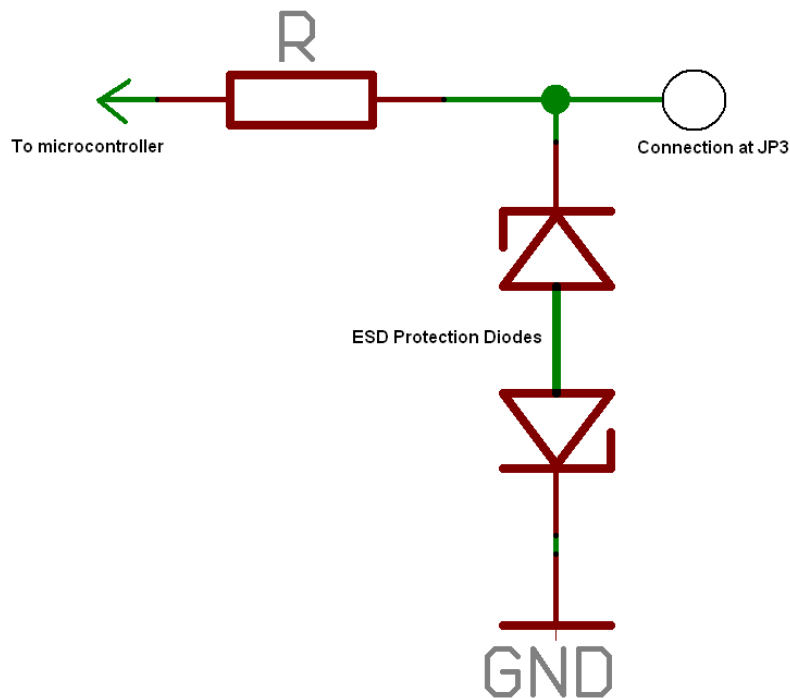
**5V Switched Output** – provides the user with the ability to power a low power target system at 5V. The USB-to-I2C Elite software has the ability to switch this output on and off. The total current provided by the USB-to-I2C Elite Hardware to the target system should



not exceed 20mA. It is the user's responsibility to ensure that excessive current is not drawn from this supply pin.

**Jumper RPU** allows the user to connect 3.3Kohm pull-up resistors, which are connected between the I<sup>2</sup>C pins (SDA and SCL) and the VP pull-up voltage. The jumpers should be removed when connecting the USB-to-I2C Elite hardware to a target system.

## PROTECTION CIRCUITRY



Both a series resistor and ESD protection diodes protect most of the pins on the header. The resistor is either 56 ohms or 47 ohms. This value can be raised to improve the protection, however, doing this will raise the low voltage level ( $V_{ol}$ ) seen on the bus. The user can change these resistors but they are responsible for any damage to the hardware. The microcontroller is a 3.3V device that is over-voltage tolerant. Pins configured as inputs or open-drain structures can have voltages up to 5.5V applied to them. Pins configured as outputs should not be subjected to voltages above 3.3V unless current limiting resistors are used to ensure excess current does not flow.

We have found that most damage to the hardware occurs when users connect the pins on the header directly to a supply voltage. When this occurs, for example on the I2C bus pins, a high current will flow that will damage the device when the output is low. This failure type usually shows up as a SDA or SCL pin that will no longer go high. Replacement of the microcontroller will be required if this occurs. The series resistors were inserted to minimize the probability of this occurring, but it will not eliminate this failure, depending upon the voltage applied and the duration.

## USB-TO-I2C ELITE HARDWARE TO TARGET CONNECTION

When connecting the USB-to-I2C Elite Hardware to an I<sup>2</sup>C target system, a cable with three (3) wires will be required. The SDA, SCL, and GND on the USB-to-I2C Elite Hardware must be connected to their corresponding signals on the target.

The USB-to-I2C Elite Hardware has 3.3K ohm pull-up resistors connected to the on-board 3.3V supply. If the target system is running at 3.3V, then the pull-ups can remain connected; otherwise, the pull-ups should be removed from the system by removing the two jumpers at RPU. If the jumpers are left in the circuit, ensure that the total I<sup>2</sup>C sink current of 3mA is not exceeded when the target pull-ups and USB-to-I2C Elite pull-ups are taken into consideration.

The SDA and SCL lines may be pulled up to a voltage range between 3.0V and 5.5V. It may work at voltages down to 2.5V; however, the noise margins will be lower.



USB-to-I2C Elite Hardware connected to a self-powered target

## USING USB-TO-I2C ELITE TO POWER A TARGET SYSTEM

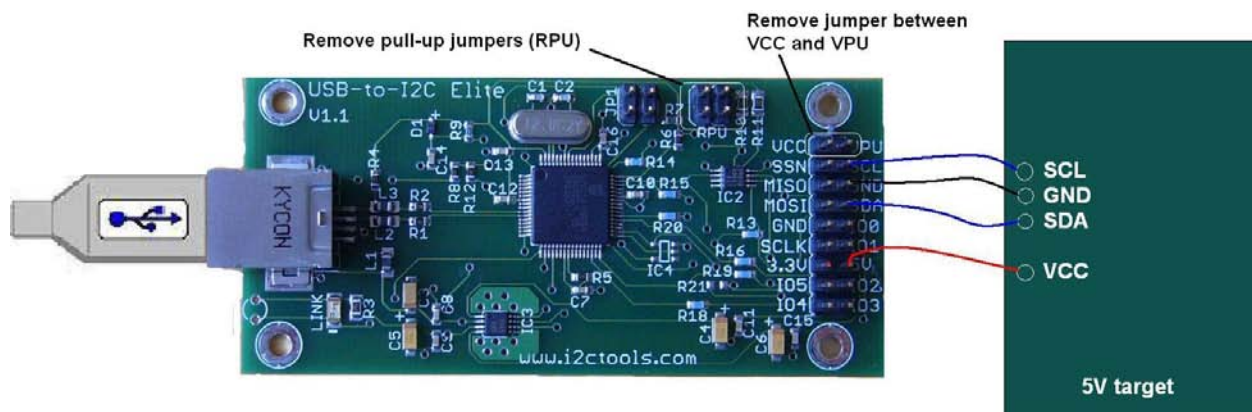
When connecting the USB-to-I2C Elite Hardware to an I<sup>2</sup>C target system, the target system can be self-powered or it may be powered from the 3.3V or 5V connectors on the hardware. The power outputs can be turned on and off using the USB-to-I2C Elite software or DLL. To turn on the power, select the appropriate item under the Options menu. The status bar at the bottom of the USB-to-I2C Elite software will indicate if a power output is enabled. Note that the 3.3V and 5V power outputs are disabled upon hardware reset.

To use the USB-to-I2C Elite hardware in a 3.3V environment, you will need to connect the SCL, SDA, GND and 3.3V to your target system. The pull-up resistors may be left in the circuit or they may be removed by removing the jumpers at RPU. If using the on-board pull-up resistors, you should connect a jumper between VP and VCC or apply an appropriate voltage to VP.



**USB-to-I2C Elite Hardware used to power a 3.3V target application**

Note that if an I<sup>2</sup>C bus voltage other than 3.3V is used, for example 5V, the user will need to configure the hardware as shown below. The minimum number of connections to the target will be four wires: SCL, SDA, GND, and 5V. The jumper between VCC and VP must be removed. The pull-up resistors may be disconnected by removing the jumpers at RPU or they may be left in the circuit but the target voltage must be connected to the VP pin.



**USB-to-I2C Elite Hardware used to power a 5V target application**

Important note: the total current consumed by the USB-to-I2C Elite hardware and the target system should be limited to 100mA by the user when using the power outputs. The USB-to-I2C Elite

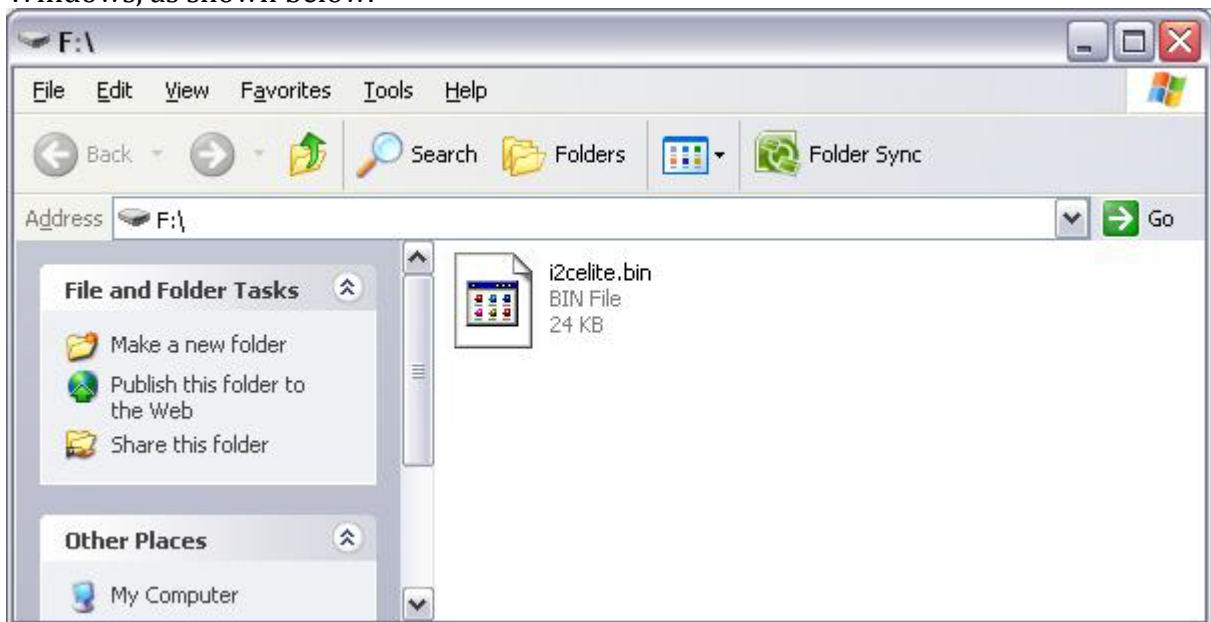
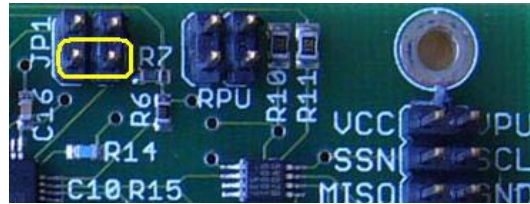
hardware uses about 80mA so the target system can consume a maximum of 20mA. Exceeding this value will violate the USB spec, but should not damage the on-board regulator.

## FIRMWARE UPDATES

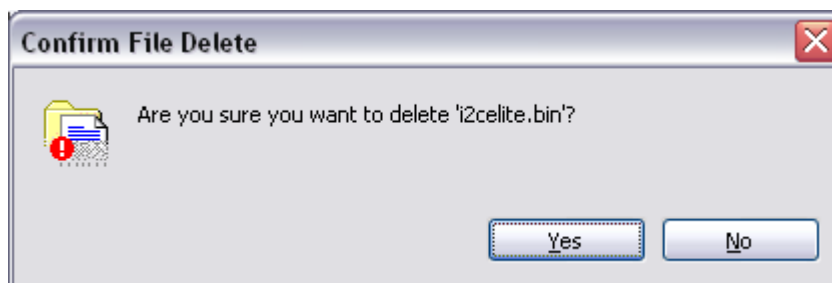
The firmware in the USB-to-I2C Elite hardware can be updated by the user. This feature is available for hardware containing firmware v2.0 or later. No special software is required for the update.

The firmware can be updated by using the following procedure.

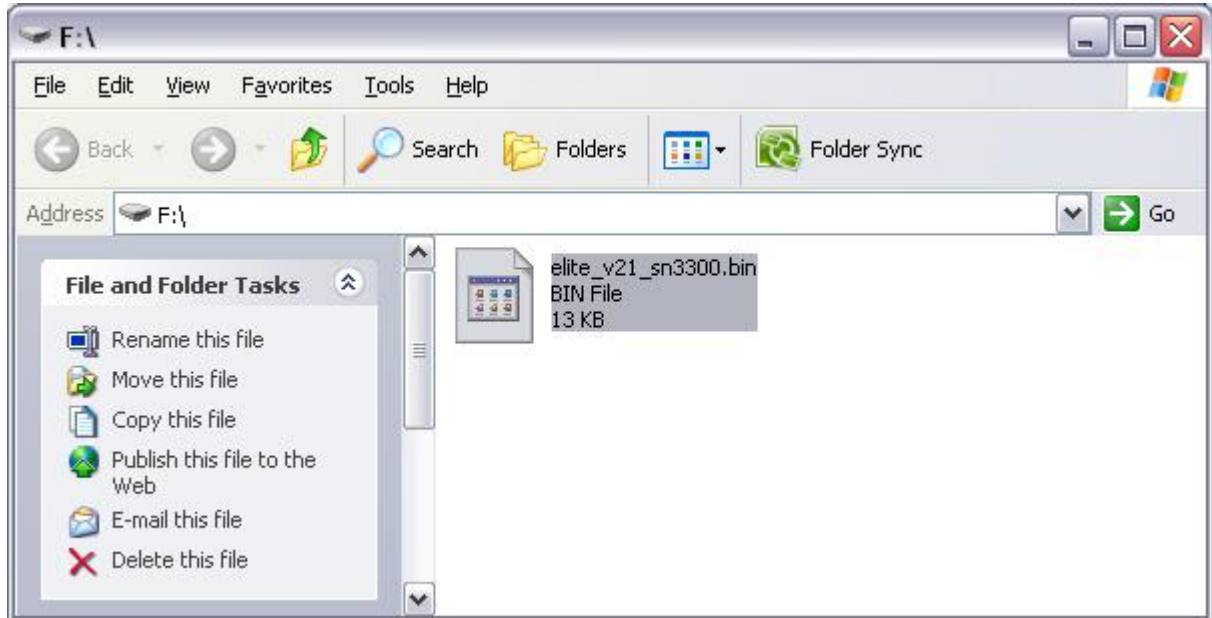
1. Disconnect the Elite hardware from the USB port. It should also be disconnected from any user target hardware.
2. Place a jumper across the bottom two pins of jumper JP1 as shown in the picture.
3. Plug the Elite hardware into the computer's USB port.
4. The USB-to-I2C Elite hardware will be displayed as a mass storage device in Windows, as shown below.



5. Select the i2celite.bin file and delete it. You will be asked if you would like to delete the file. Choose Yes.



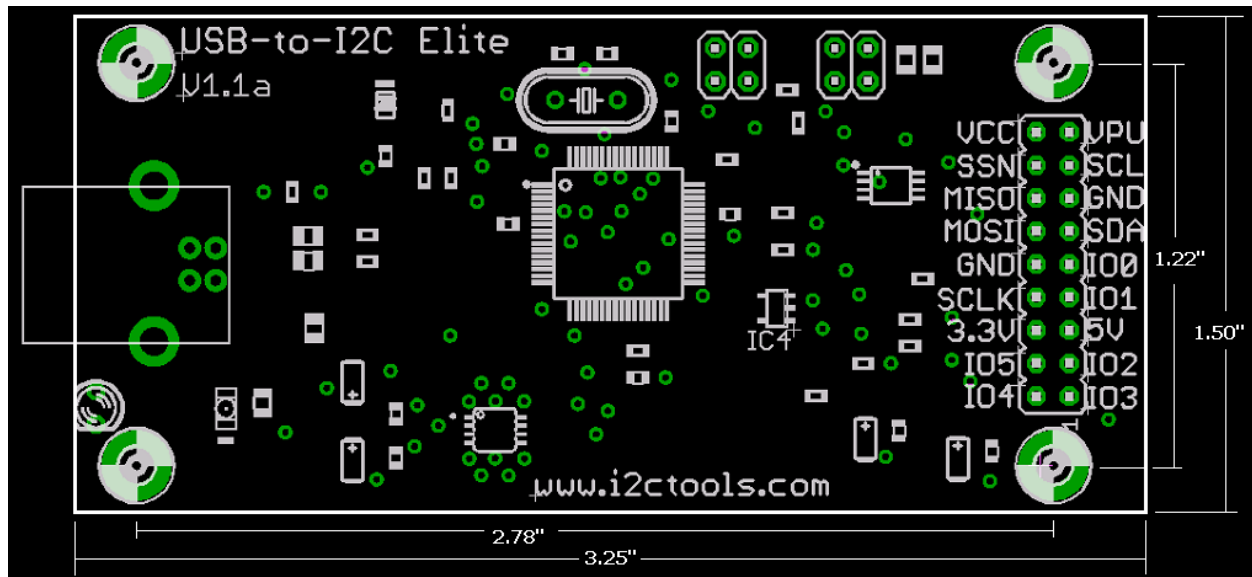
- Place the new firmware update into the folder.



- Disconnect the Elite hardware from the USB port.
- Remove the jumper from JP1.
- You can now use the USB-to-I2C Elite 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 Elite software.



## PCB LAYOUT



The PCB dimensions are shown above.  
The PCB dimensions are 3.25" x 1.50"  
The mounting holes are 2.78" x 1.22"

## SOFTWARE UPDATES

The USB-to-I2C Elite software has a built-in update feature that 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 Elite updates can also be downloaded manually from the following website: <http://www.i2ctools.com/updates>  
Check our website periodically for update announcements and information.

## TECHNICAL SUPPORT

Technical Support for USB-to-I2C Elite is available via an email to [support@i2ctools.com](mailto:support@i2ctools.com).

## TROUBLESHOOTING

- If you have problems with the software installation, ensure that you have Admin Privileges.
- If the installer application doesn't launch when the CD-ROM is inserted, then manually browse the CD-ROM and find the Setup.exe file in the root directory of the CD. Double-click on the Setup.exe file to start the installation process.
- The Adapter has 3.3k $\Omega$  pull-up resistors to 3.3V on the I<sup>2</sup>C bus lines. Ensure that additional pull-up resistors in the user target system do not bring the pull-up

resistance below 1.6k $\Omega$  for 5V systems or 1.1K $\Omega$  for 3.3V systems. The on-board pull-up resistors can be removed from the system by removing the pull-up resistor jumpers RPU (see Hardware Description above).

- The USB-to-I2C Elite version 5 and later software is only supported on Windows XP, Vista, and Windows 7.
- USB-to-I2C Elite monitors the communications on the I<sup>2</sup>C 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.
- Keep the original USB-to-I2C Elite installation CD-ROM in the event that the software needs to be re-installed. Future USB-to-I2C Elite updates from our Website may require a previous installation from the original media.
- Check for new versions of the software at <http://www.i2ctools.com/downloads.html>
- The firmware for the Elite hardware can be upgraded by the user, if the firmware loaded on the Elite hardware is version 2.0 or later. We will also upgrade the firmware for you if you have an earlier revision. The hardware will need to be returned to us for version 1.x. Please visit the i2ctools website to find the latest firmware information.

If all else fails, email a description of the problem you are having to us at [support@i2ctools.com](mailto:support@i2ctools.com).

Note that all technical support requests must begin with an email to this email address.

We are interested in receiving feedback from our customers. Is there is a feature that should be added to make this tool better? Please send your requests and comments to [support@i2ctools.com](mailto:support@i2ctools.com).