

# Tutorial: How to Build, Program and Send Data Using the Easylogger with a USB Cable

By: Rafal Piersiak

The Easylogger reads the voltage on a port pin and sends a line of data each second to a computer using a USB cable. The Easylogger acts like a keyboard and will input data directly into an open Excel spreadsheet, text file, etc.

This tutorial will guide you in:

- Building the Easylogger
- Editing the code
- Compiling the code
- Programming the ATtiny45
- Sending ADC readings to the computer using the Easylogger USB interface

1. Download the Easylogger software:

<http://www.obdev.at/products/vusb/easylogger.html>

2. Unzip the Easylogger file

3. Read the Readme.txt

4. Go into the circuit folder and open up "Easylogger.pdf"
  - Build this circuit exactly as shown
  - J1 can be ignored if you wish

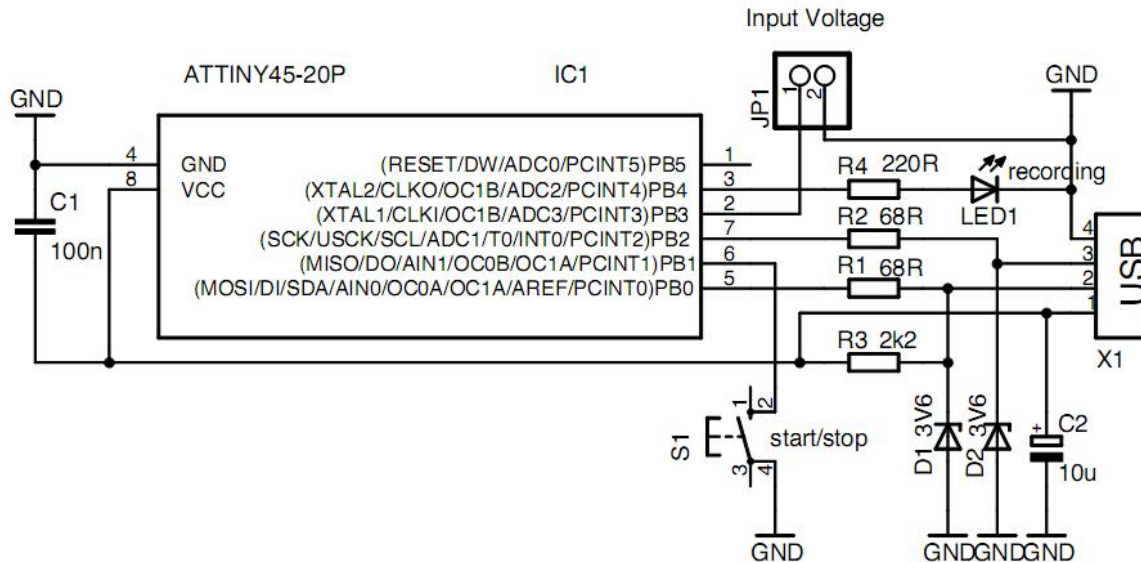


Figure 1 - EasyLogger Schematic

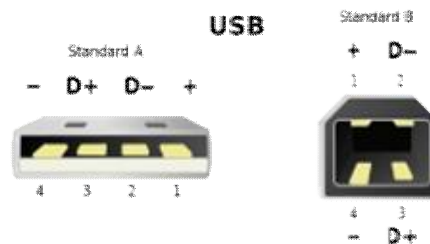


Figure 2 - USB Pin Configuration

## 5. Once the circuit is built, unplug Pin 2 (D-) and Pin 3 (D+) of the USB!!!

If this is not done, when you attempt to flash the hex file in AVR Studio 4:

- AVRISP mkII will say that Pin 5 (MOSI) and Pin 7 (SCK) are shorted.
- JTAGICE mkII will give you a huge error box saying everything is wrong.

Please Review the next page for PCB design considerations!

## 6. PCB Considerations

If you build your own USB cable, D- and D+ can be easily unplugged from a breadboard.

If you build the EasyLogger on a PCB, you will not be able to disconnect D- and D+. This will cause problems during programming and will result in errors.

1. To rectify this problem, you must create a special USB cable that only has the power and ground wires. The D- and D+ must be removed or cut to prevent any signal from coming from the USB port to the EasyLogger. This way, you can supply power during programming using the special cable and then use a regular USB cable for transmitting data from the EasyLogger to the computer.
2. Another solution is to program the ATtiny45 on a breadboard and then solder it on after it is programmed.
3. A final solution is to find another way to supply the circuit with 5V power during programming, without connecting anything to the USB port on the EasyLogger. A power and ground pin can be added to the PCB to accommodate this method.

7. Wire the ISP interface for the ATtiny45

- The ATtiny45 can only be programmed using ISP.
- The JTAG programming method is not supported on the ATtiny45.
  - The JTAGICE mkII programmer can use ISP! Yay!

You can use these programmers:

- **AVRISP mkII** (6 Pin Header)



Figure 3 - AVRISP mkII

The little programmer, cheap, **ISP only**

- **JTAGICE mkII** (10 Pin Header)



Figure 4 - JTAGICE mkII

The big programmer, expensive, **JTAG and ISP**

This tutorial will show how to program using both programmers and the corresponding connections to the ATtiny45.

To read more on how to use the JTAGICE mkII as an ISP programmer, please read:

JTAGICE mkII Quick Start Guide  
[www.atmel.com/Images/doc2562.pdf](http://www.atmel.com/Images/doc2562.pdf)

## Wiring the AVRISP mkII to the ATtiny45

The following figure shows the pin connections for an ISP header.

**Figure 1.** ISP6PIN header pinout

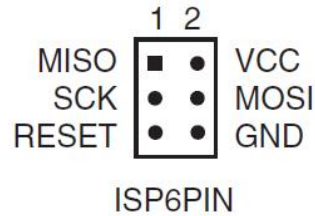


Figure 5 - Standard ISP 6 Pin Header

Wire the ISP header as described in the following table.

ISP Header Pin #	Signal Name	Connect to ATTINY45 Pin:
Pin 1	MISO	MISO (Pin 6)
Pin 2	VCC	+VCC (Pin 8)
Pin 3	SCK	SCK (Pin 7)
Pin 4	MOSI	MOSI (Pin 5)
Pin 5	RESET	/RESET (Pin 1)
Pin 6	GND	GND (Pin 4)

**Table 1 - Wiring the ISP header to the ATtiny45**

## Wiring the JTAGICE mkII to the ATtiny45

The following figure shows the pin connections for a JTAG header.

**Figure 3.** JTAG10PIN connector

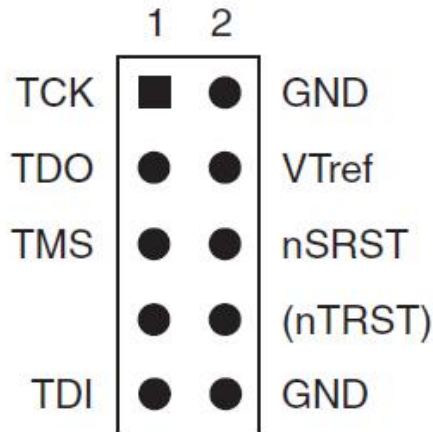


Figure 6 - Standard JTAG Pin Header

Wire the JTAG header as described in the following table.

JTAG Header Pin #	Signal Name	Connect to ATTINY45 Pin:
Pin 1	TCK	SCK (Pin 7)
Pin 2	GND	GND (Pin 4)
Pin 3	TDO	MISO (Pin 6)
Pin 4	VTref	+VCC (Pin 8)
Pin 6	nTRST	/RESET (Pin 1)
Pin 9	TDI	MOSI (Pin 5)

Table 2 - Wiring the JTAG header to the ATtiny45

8. Install the following software

To **edit, compile, and flash** code for the Easylogger, you must install:

- AVR Studio 4
- WinAVR - (AVR-GCC)
  - This is a free C compiler for AVR Studio 4

To **flash** the AVRtiny45 without playing with the code, you must install

- AVR Studio 4

This tutorial was done using:

- AVR Studio 4 (Version 4.18 Build 716)
- WinAVR (Date Code: 20100110)

If you want to try using AVR Studio 5 or IAR, I would assume putting the files in a project, compiling, and programming the device would suffice. I do not know how IAR carries out ISP programming, so some effort would be needed to figure this out.

## 9. To **edit** and **compile** code:

Edit:

Open up (using notepad, AVR Studio, IAR or anything else) "main.c", found in the firmware folder of the unzipped Easylogger file.

Do any edits you wish and **save** the file

Common Edits:

In function "evaluateADC"

- Comment out "value += value + (value >> 1); /\* value = value \* 2.5 for output in mV \*/"
  - This will cause the Easylogger to output the reading stored in the register after a read, (value of 0-1023)

In function "adcInit"

- You can alter "ADMUX = UTIL\_BIN8(1001, 0011);"
  - "1001" Vref = 2.56V
  - "0011" ADC3
- To select Vref = VCC
  - "0000" Vref = VCC
  - Like this "ADMUX = UTIL\_BIN8(0000, 0011);"



Compile:

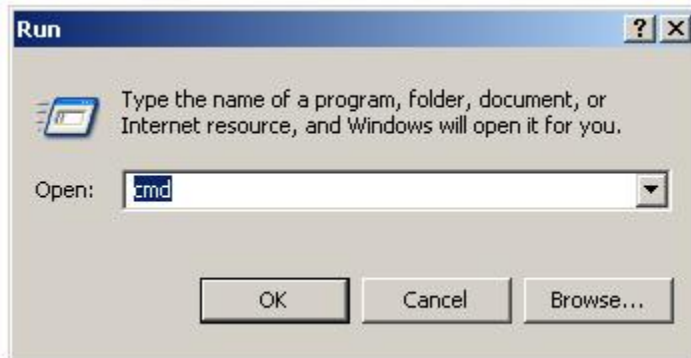
## Keep the entire Easylogger folder in a filepath with no spaces!!!!

Ex: C:\Easylogger\firmware

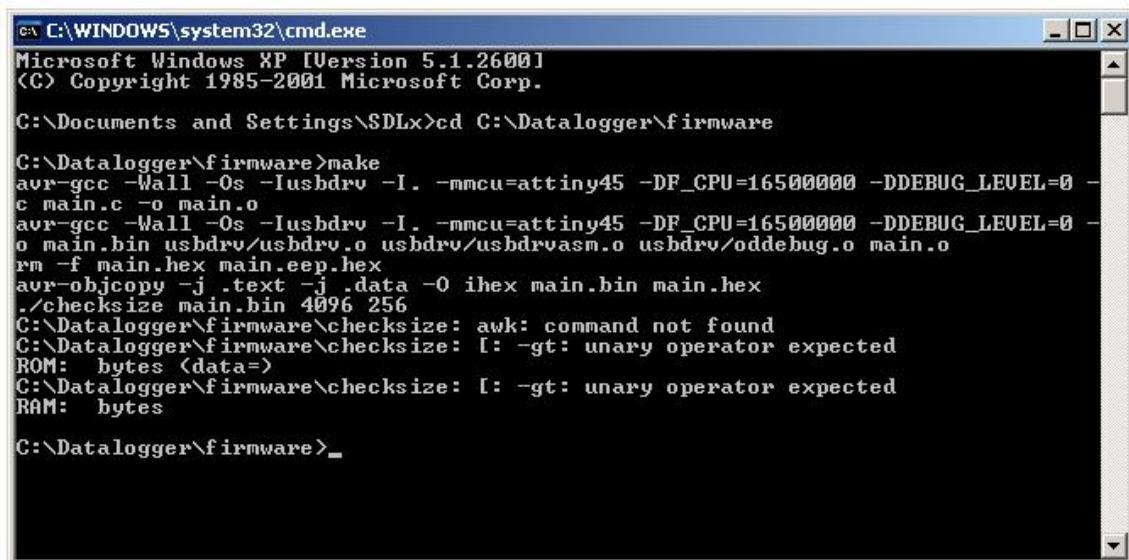
Spaces will cause the command prompt to freak out.

Open up Command Prompt

- Click Start, Find "Run"
- Type in CMD



- This will pop up



- Type in "cd filepath (filepath = Ex: C:\Datalogger\firmware)"
  - Press enter key
- Type in "make"
  - Press enter key
- You should get the following file:
  - main.hex

**To recompile the files again, you need to delete:**

- main.bin
- main.o
- main.hex

Follow the command prompt procedure again after the files have been deleted.

10. To flash the code:

### **Unplug Pin 2 (D-) and Pin 3 (D+) of the USB!!!**

If this is not done, when you attempt to flash the hex file in AVR Studio 4:

- AVRISP mkII will say that Pin 5 (MOSI) and Pin 7 (SCK) are shorted.
- JTAGICE mkII will give you a huge error box saying everything is wrong.

Open up AVR Studio 4

Create New Project:

- Create New Project
- Select “**AVR GCC**” under project type
  - You may need WinAVR to see AVR GCC as an option.
- You can probably choose “**Atmel AVR Assembler**” also.
  - The project type should not matter.
- Write a **Project Name**
- Check the **Create Folder** Box
- Uncheck the **Create Initial File** box
- Select a **Location**
- Press Next

Select Debug Platform and Device:

- Select **JTAGICE mkII**
  - If you are using AVRISP mkII, select JTAGICE mkII anyway.
- Select **ATtiny45**
- Press Finish

To ensure that AVR Studio 4 connects to the programmer:

- 1) Connect the programmer header to the board**
- 2) Connect the programmer to the USB port**
- 3) Connect the Easylogger to the USB port**

Press the  button on the toolbar

Select one of the following **Platforms**:

- JTAGICE mkII
- AVRISP mkII

Select Auto for the **Port**

Press Connect

To Program the Fuse:

Click the **Fuses** Tab

For **EXTENDED**, type in 0xFF

For **HIGH**, type in 0xDD

For **LOW**, type in 0xE1

**SUT\_CKSEL** should say what is seen in the following figure.

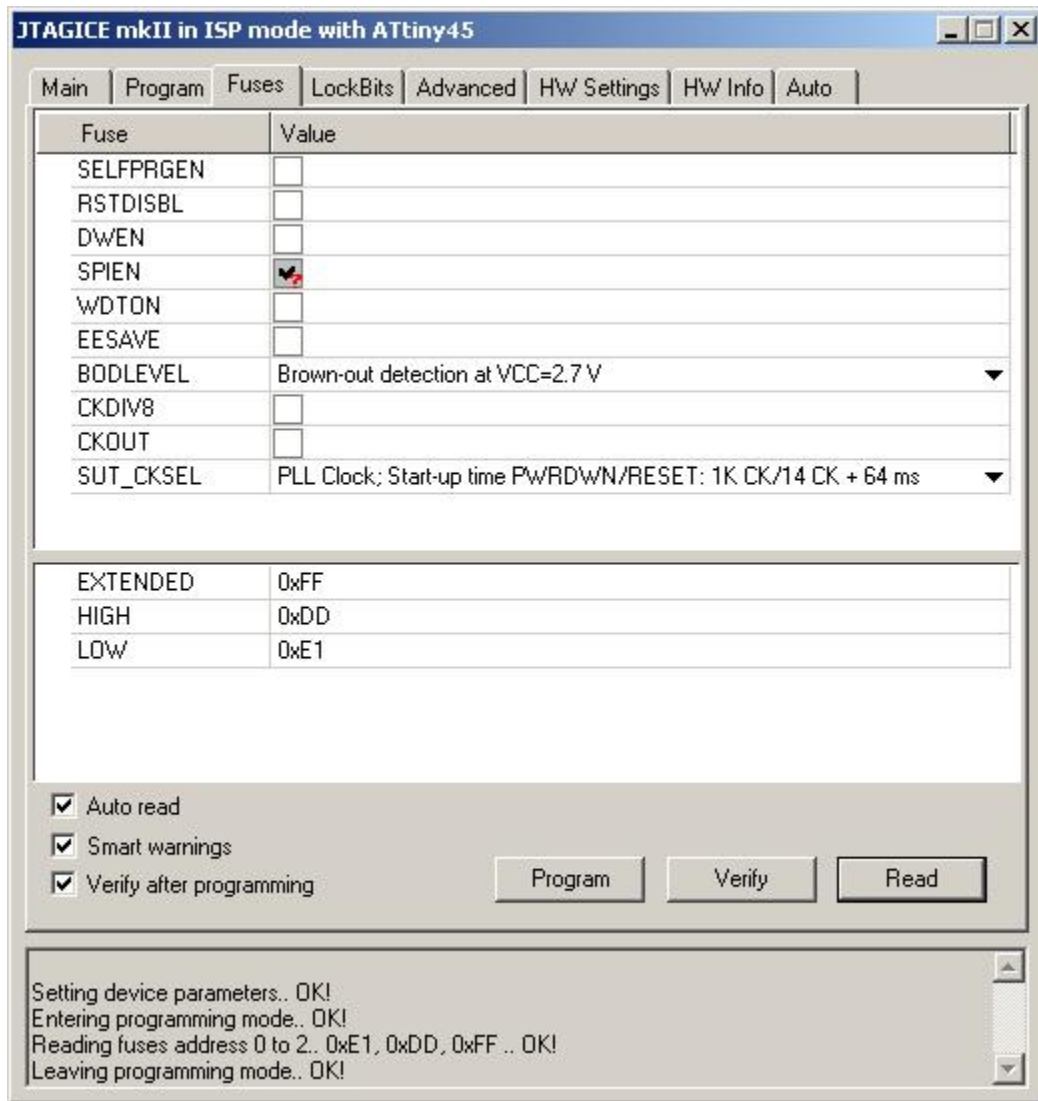


Figure 7 - Fuse Configuration

Press the **Program** button

Check the bottom of the window and check that everything is "OK!"

To Program the ATtiny45

Click the **Program** Tab

In the Flash section, find you main.hex file location

Press the **Program** button

Check the bottom of the window and check that everything is “OK!”

**Close the window by pressing the X**

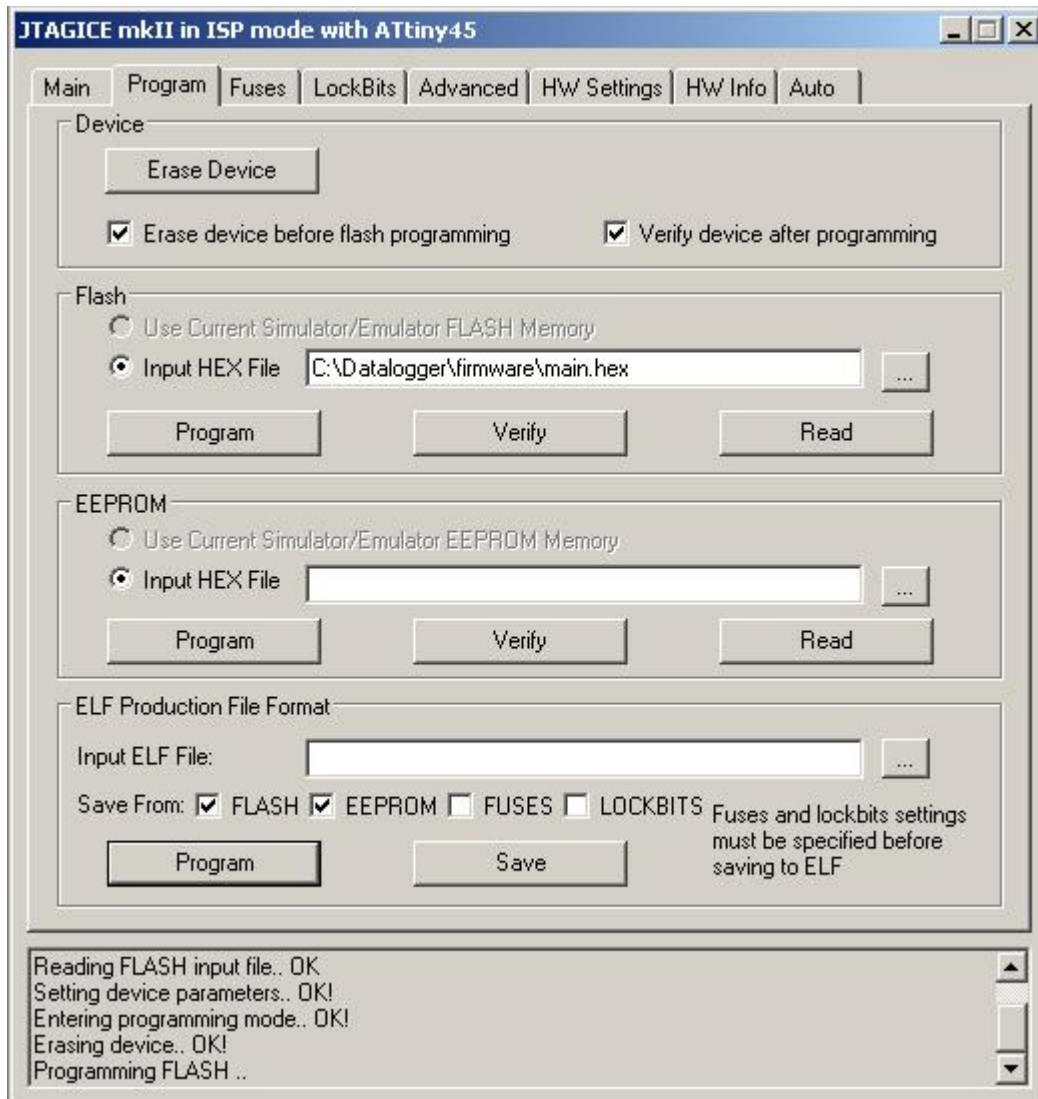


Figure 8 - Program Configuration

11. Try sending data to the computer

Disconnect the programmer header from the Easylogger

**Connect Pin 2 (D-) and Pin 3 (D+) of the USB**

Windows should identify it automatically

- A balloon will pop up saying Easylogger
- Another balloon will say Human Interface Device

Press the pushbutton on the Easylogger

- The LED should turn on

That is it! Enjoy your awesome Easylogger!

## Appendix:

Download the Easylogger software:

<http://www.obdev.at/products/vusb/easylogger.html>

JTAGICE mkII Quick Start Guide

[www.atmel.com/Images/doc2562.pdf](http://www.atmel.com/Images/doc2562.pdf)

USB Pin Out

[http://en.wikipedia.org/wiki/Universal\\_Serial\\_Bus](http://en.wikipedia.org/wiki/Universal_Serial_Bus)

Relevant Websites and Forums (Tips)

<http://imakeprojects.com/Projects/avr-tutorial/>

<http://codeandlife.com/2012/01/29/avr-attiny-usb-tutorial-part-3/>

<http://davehillier.wordpress.com/category/attiny45/>

<http://forums.obdev.at/viewtopic.php?f=8&t=5976>

<http://forums.obdev.at/viewtopic.php?f=8&t=4467>

<http://www.ruinelli.ch/how-to-use-v-usb-on-an-attiny85>

<http://forums.adafruit.com/viewtopic.php?f=24&t=16208>

This tutorial was done using:

- EasyLogger.2008-02-28.zip
- AVR Studio 4 (Version 4.18 Build 716)
- WinAVR (Date Code: 20100110)
- Windows XP Professional
- JTAGICE mkII and AVRISP mkII