Arduino 101 Firmware Update

**Tables of Contents**

[Firmware Update Using JTAG 2](#_Toc434519477)

[1. Connect all wires as shown on the picture 2](#_Toc434519478)

[2. Install Flyswatter2 probe drivers: 3](#_Toc434519479)

[2.1. Windows 3](#_Toc434519480)

[2.2. Linux 3](#_Toc434519483)

[3. Flash the firmware 4](#_Toc434519484)

[Firmware Update Using USB 5](#_Toc434519485)

[1. Connect USB cable to board as shown 5](#_Toc434519486)

[2. Install drivers 6](#_Toc434519487)

[2.1. Windows 6](#_Toc434519488)

[2.2. Linux 7](#_Toc434519489)

[3. Flash firmware 8](#_Toc434519490)

# Firmware Update Using JTAG

## Connect all wires as shown on the picture



**7-17V power**

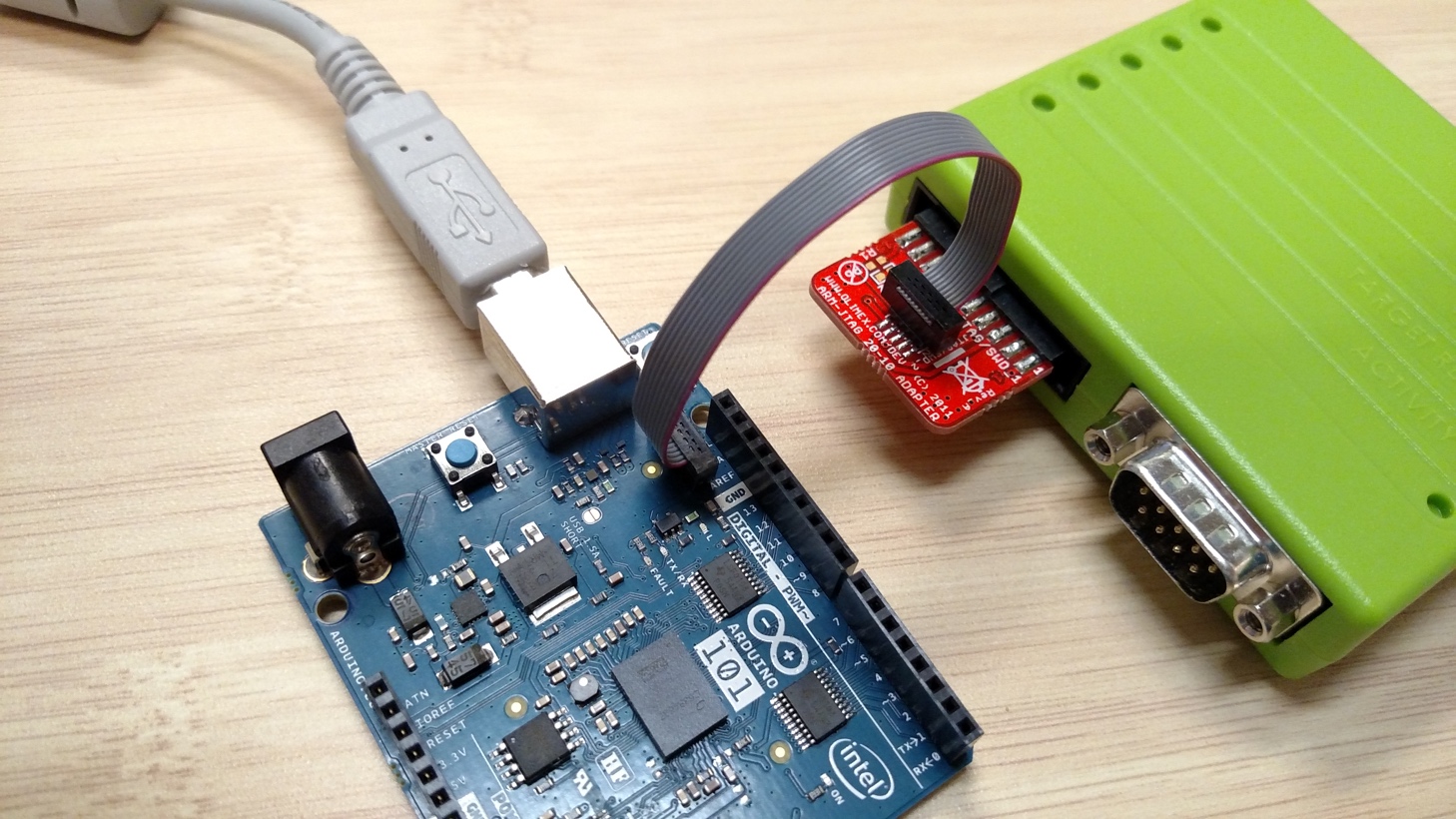
**Flyswatter2**

**Olimex 20 to 10 pin adapter with ribbon cable**

**Components listing:**

1. Windows/Linux/OSX laptop
2. Flyswatter2 JTAG probe with USB cable
3. Olimex 20 to 10 pin adapter with ribbon cable
4. Arduino 101 board with 7-17V DC power supply (or USB-B cable)

**Make sure to attach ribbon cable in a correct way, as shown on the photo!**



**Make sure to attach ribbon cable in a correct way!**

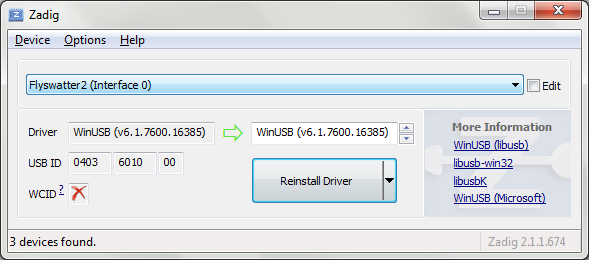
**If you do not have an A/C adapter, you can also supply power through USB-B port**

**Pin-1**

**Pin-1**

## Install Flyswatter2 probe drivers:

### Windows



* + 1. Plug in Flyswatter2 probe to the host
    2. Download and extract the latest firmware release
    3. Go to bin\ directory, run "zadig\_2.1.1.exe".
    4. Options > List all devices.
    5. Select your probe (Flyswatter2), pick WinUSB and hit Reinstall Driver;  
       do it for Interface 0 and Interface 1.
    6. Close zadig and **REPLUG THE PROBE**



### Linux

By default, non-root users won’t have access to the JTAG pods connected via USB. You must grant write access to the proper /dev/bus/usb entry every time a device is connected to be able to run OpenOCD using a non-root account. The process can be automated by adding an udev rule. Simply create a text file in the rules directory:

$ sudo vim /etc/udev/rules.d/99-openocd.rules

The IDs depend on the JTAG device. For example, for the Flyswatter2\* and the Olimex-ARM-USB-OCD-H, the rules file must have the following content:

SUBSYSTEM=="usb", ATTR{idVendor}=="0403", ATTR{idProduct}=="6010", MODE="0666"

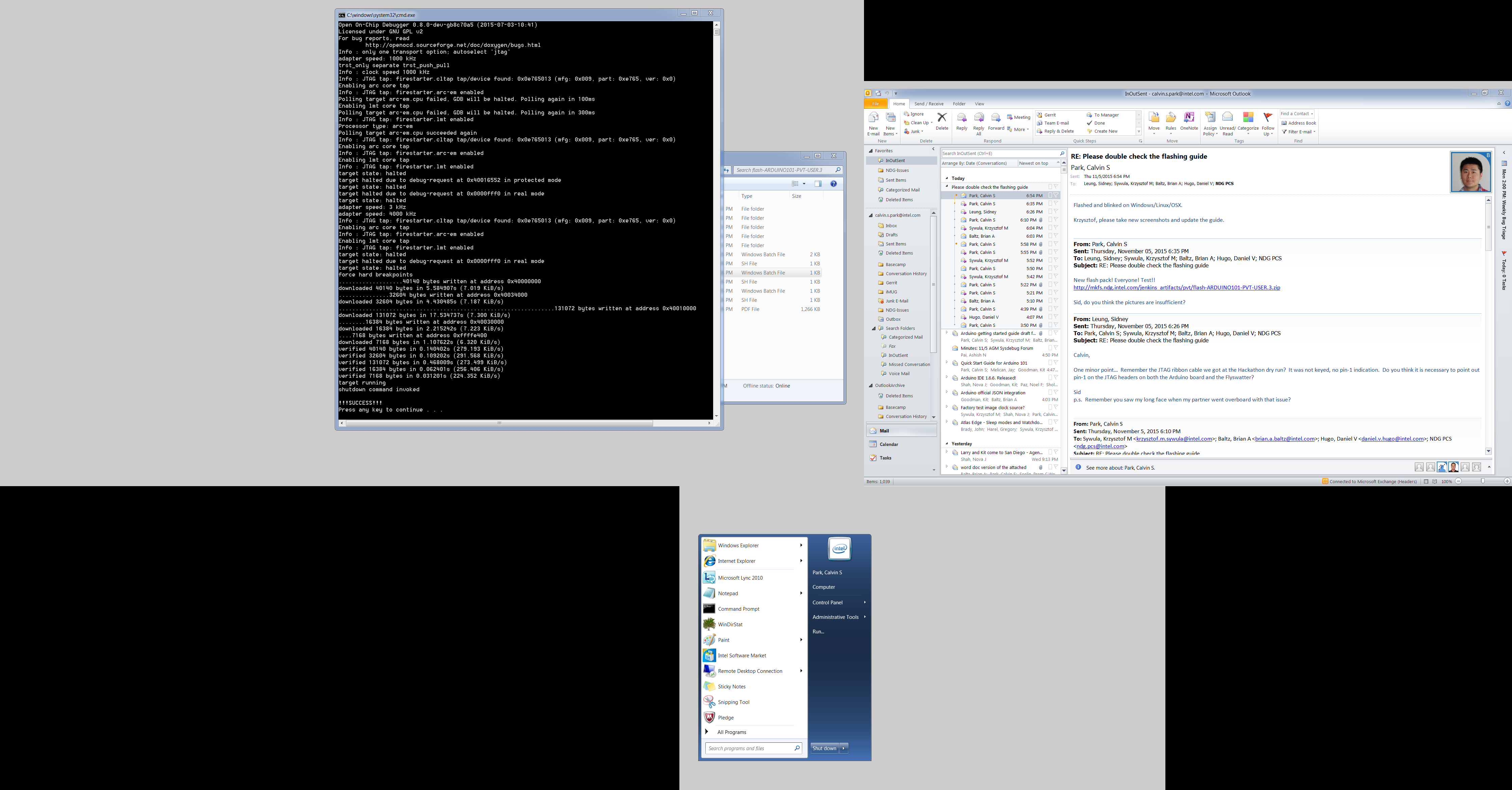
SUBSYSTEM=="usb", ATTR{idVendor}=="15ba", ATTR{idProduct}=="002b", MODE="0666"

(See drivers/rules.d/99-openocd.rules)

## Flash the firmware

* + 1. In the extracted flash pack directory, run the flashing script:
  + **Windows:** Execute (double-click) **flash\_jtag.bat** in order to flash production image.
  + **Linux or OSX:** Run **flash\_jtag.sh** in order to flash production image

Below is how a successful flash looks like



# Firmware Update Using USB

## Connect USB cable to board as shown



**Components listing:**

1. Windows/Linux/OSX Laptop
2. USB cable
3. Arduino 101 board
4. 7-17V **DC** power supply (or USB-B cable)

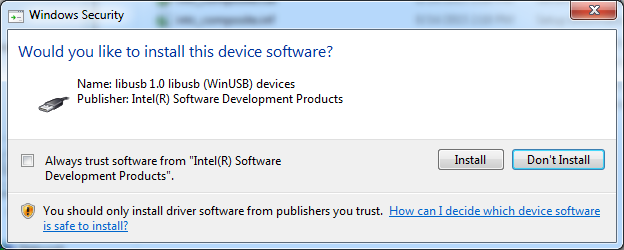
## Install drivers

### Windows

1. Download and extract the latest firmware release
2. In the extracted flash pack directory, go to drivers\Windows\ directory and run
   * + dpinst-amd64.exe on 64-bit Windows
     + dpinst-x86.exe on 32-bit Windows



1. Click Next



1. Click **Install**



1. Click **Finish**

### Linux

The DFU device can be set up for use by regular users by editing a text file in the rules directory.  
Enter **ONE RULE PER LINE**. Newline characters are not allowed.

$ sudo vi /etc/udev/rules.d/99-dfu.rules

# Arduino 101 in DFU Mode

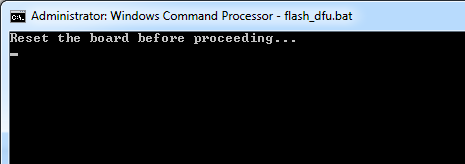
SUBSYSTEM=="tty", ENV{ID\_REVISION}=="8087", ENV{ID\_MODEL\_ID}=="0ab6", MODE="0666", ENV{ID\_MM\_DEVICE\_IGNORE}="1", ENV{ID\_MM\_CANDIDATE}="0"

SUBSYSTEM=="usb", ATTR{idVendor}=="8087", ATTR{idProduct}=="0aba", MODE="666", ENV{ID\_MM\_DEVICE\_IGNORE}="1"

(See drivers/rules.d/99-dfu.rules)

## Flash firmware

1. In the extracted flash pack directory, run the flashing script:
   * **Windows:** Execute (double-click) **flash\_dfu.bat**
   * **Linux or OSX:** Run **flash\_dfu.sh**



* + Press the **reset** button on the board to start the flash process

1. Below is how a successful **DFU** flash looks like

