

no iframe



Burning the Bootloader without external AVR-Writer

Kimio Kosaka

kim.kosmac@gmail.com

2008.07.27 05:48(JST)

When we burn the bootloader of Arduino Diecimila, we connected external AVR-Writer to ICSP of Diecimila.

But I found the method to burn the bootloader by Diecimila itself without external AVR-writer. It is able to do by FT232RL BitBang Mode AVR-Writer software.

And, this method is usable on the Arduino NG, Arduino Duemilanove and w/ATMega328P too.

But, this method cannot do on the Arduino Uno.

[Japanese](#) English

This page was renewed for tutorial of "Btiband method" in GUI-Operation.

Old version tutorial in CUI-Operation is here -> [Burning the Bootloader without external AVR-writer \(CUI Operation\)](#)

How to run this "FTDI Bitbang method" with the Arduino-IDE ->[Click here](#)

1. FT232RL BitBang Mode AVR-Writer

FT232RL is an USB-Serial bridge on an Arduino Dicimila/NG/Duemilanove PCB. It has the function to manipulate each signal pin directly. It's called [BitBang Mode](#).

"avrdude-serjtag" is AVR-Writer software developed by [Mr.Suz](#). It include the function to control FT232RL BitBang Mode.

If we use "avrdude-serjtag" we can burn the bootloader by Diecimila itself.

I describe below the method. on Windows-XP.

- o **Attention!!!**

If you want use "avrdude-serjtag" on Linux or Mac OS, you must remake (patch, reconfigure and recompile) it.

There are useful projects.

- o These projects made a way to BitBang Mode AVR-Writer on Linux and Mac !!!

Of course, if you use Windows in Vmware on your Linux or Mac OS, you can run "avrdude-serjtag(windows version)".

[avrdude by FT245R/FT232R\(Linux\)](#) (Translated to English by Google.)

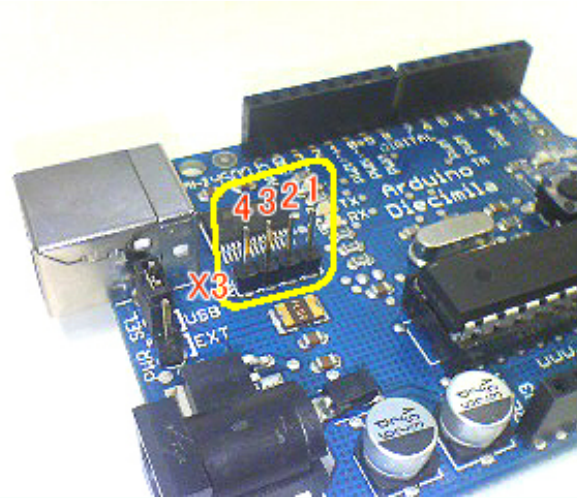
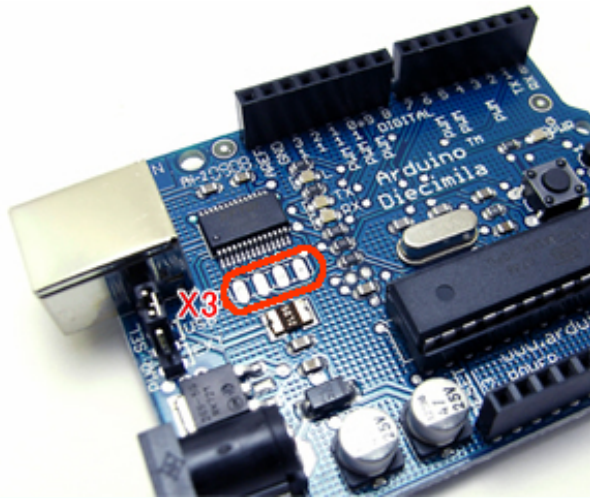
[FTDI Drivers for avrdude in Linux](#) by Coffeebot Labs

[BitBang Mode AVR-Writer on Mac](#) (Translated to English by Google.)

2. Modify the Dicimila

There are four pads written as X3 near FT232RL on a Diecimila PCB. (These pads are connected to the control pins of FT232RL.)

Remove the solder of these pads and insert a pin-header. And soldering it.



3. Downloading

- i. "avrdude-serjtag" FTDI BitBang AVR-Writer

[serjtag-0.3.zip](#) mirror site

[serjtag-0.3.tar.gz](#) mirror site

- ii. [serjtag-0.3.zip](#) original site (<http://suz-avr.sblo.jp/archives/20070621-1.html>)

[serjtag-0.3.tar.gz](#) original site (<http://suz-avr.sblo.jp/archives/20070621-1.html>)

- iii. configure-file for avrdude-serjtag

[avrdude.conf](#) (**Update:** included chip-parameter of ATmega328P,168P and 88P)

- iv. "avrdude-GUI (yuki-lab.jp Version)" GUI wrapper for avrdude

[avrdude-GUI-1.0.5.zip](#) mirror site

[avrdude-GUI-1.0.5.zip](#) original site (<http://yuki-lab.jp/hw/avrdude-GUI/index.html>)

- v. avrdude-GUI (yuki-lab.jp Version) require Microsoft .NET Framework 2.0.

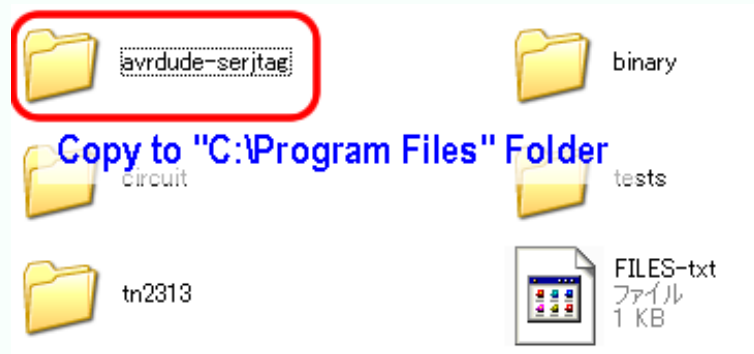
When .NET Framework 2.0 is not installed. Download it from [here \(Microsoft .NET Framework 2.0 download page\)](#)

and install it.

4. Installing

i. avrdude-serjtag

- Extract serjtag-03.zip.
- Copy "avrdude-serjtag" folder into the "C:\Program Files" folder.



- Delete "src" folder in the "avrdude-serjtag" folder.

ii. avrdude.conf

- Copy(overwrite) "avrdude.conf" into the "C:\Program Files\avrdude-serjtag\binary" folder.
- This modified "avrdude.conf" has setting-scripts of "FTDI BitBang AVR-Writer" for Diecimila below.

```
#arduino diecimila

programmer

id="diecimila";

desc = "FT232R Synchronous BitBang";

type = ft245r;

miso = 3; # CTS X3(1)

sck = 5; # DSR X3(2)

mosi = 6; # DCD X3(3)
```

```
reset = 7; # RI X3(4)  
;
```

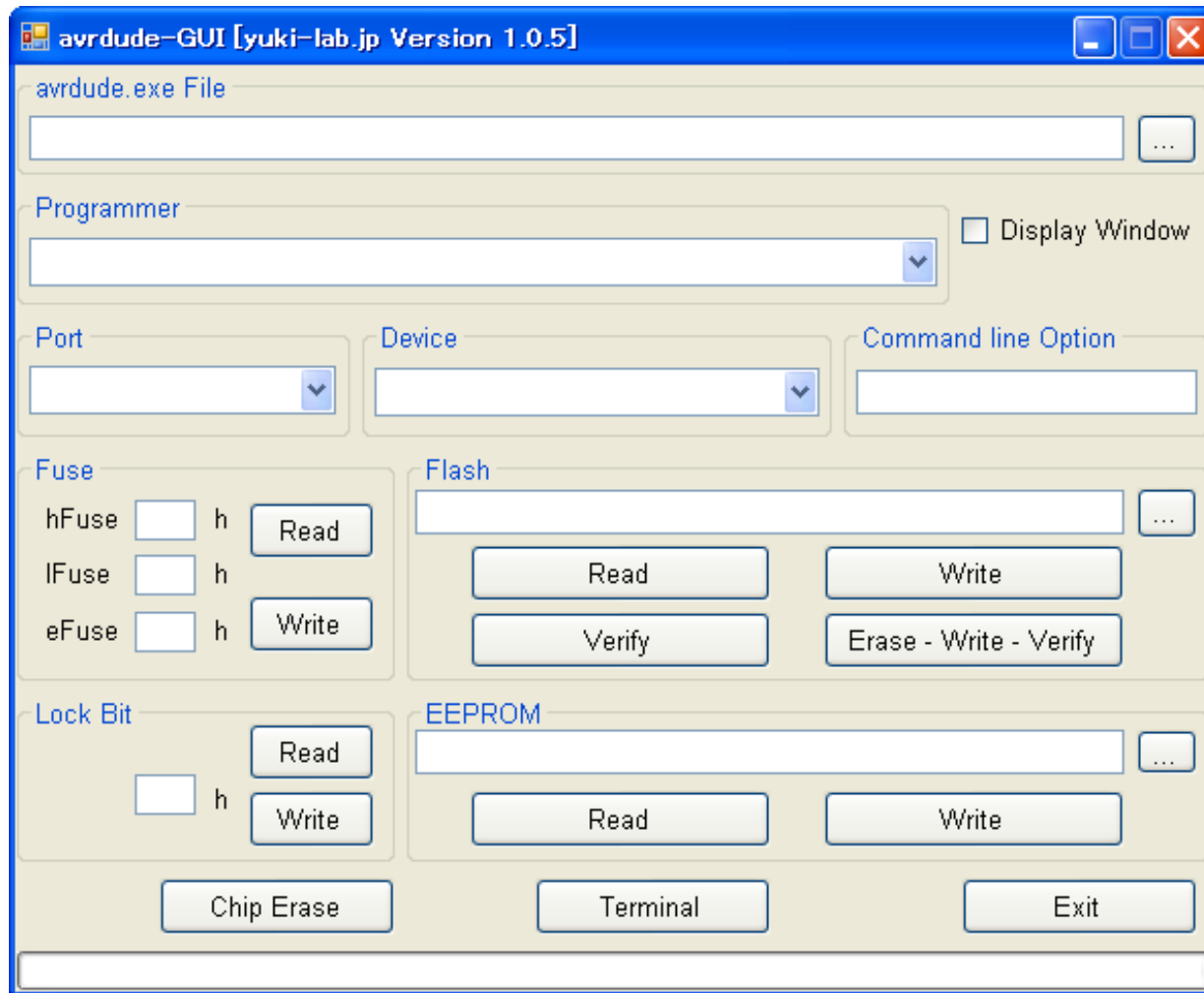
5. avrdude-GUI (yuki-lab.jp Version)

- Extract avrdude-GUI-1.0.5.zip.
- Copy "avrdude-GUI-1.0.5" folder into "C:\Program Files" folder.
- avrdude-GUI (yuki-lab.jp Version) require Microsoft .NET Framework 2.0.

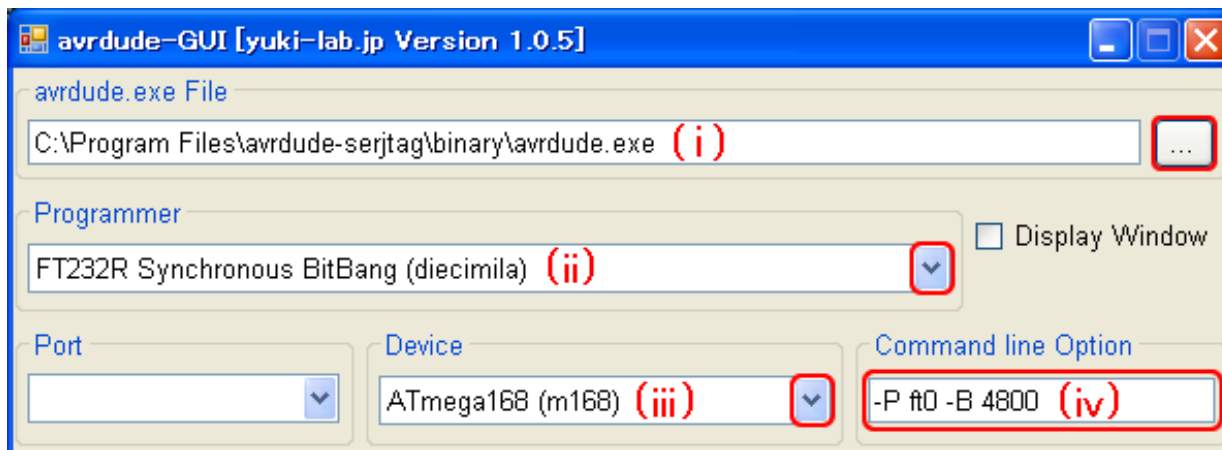
When .NET Framework 2.0 is not installed. Download it from [here](#) and install it.

- Setting


Open the "C:\Program Files\avrdude-GUI-1.0.5" folder. And double click the "avrdude-GUI.exe" to run it.



"avrdude-GUI" settings is as below from i to iv.



i. avrdude.exe File area

- o Click on the .
- o Open "C:\Program Files\avrdude-serjtag\binary" folder.
- o Select "avrdude.exe", and click on the [Open(O)] button.

ii. Programmer area

- o Click on the .
- o Select "FT232R Synchronous BitBang (diecimila)".

iii. Device area

- o Click on the .
- o Select "ATmega168 (m168)".

iv. Command line Option area

- o Click the text-box. And key in below.

-P ft0 -B 4800

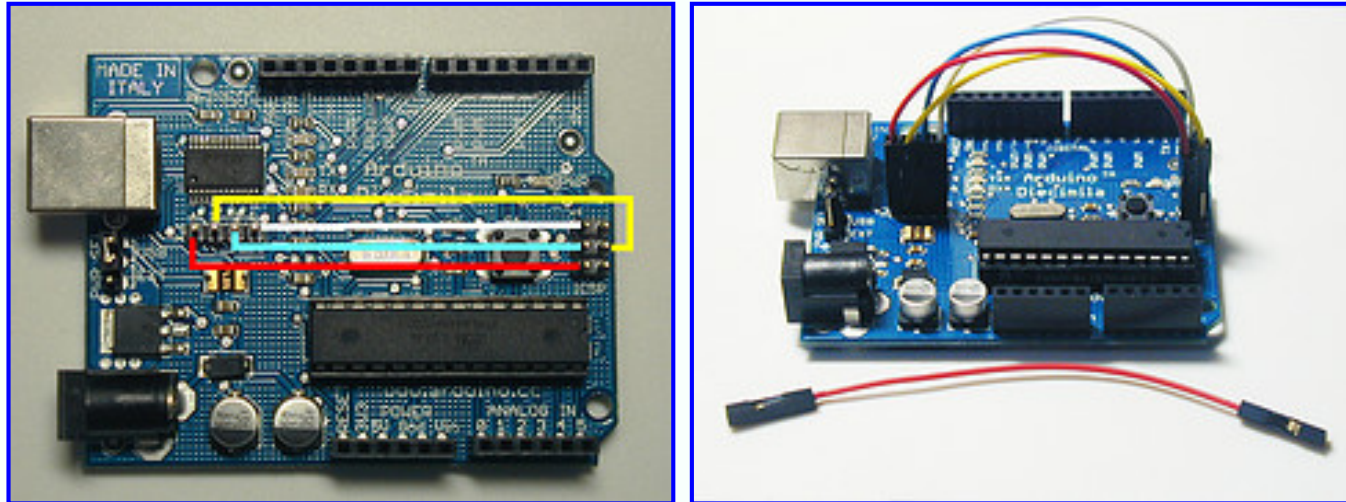
"-P ft0" is Bitbang-Port settings. "-B 4800" is slow clock mode settings.

Attention !!! Must make the Port area the blanks.

- Testing

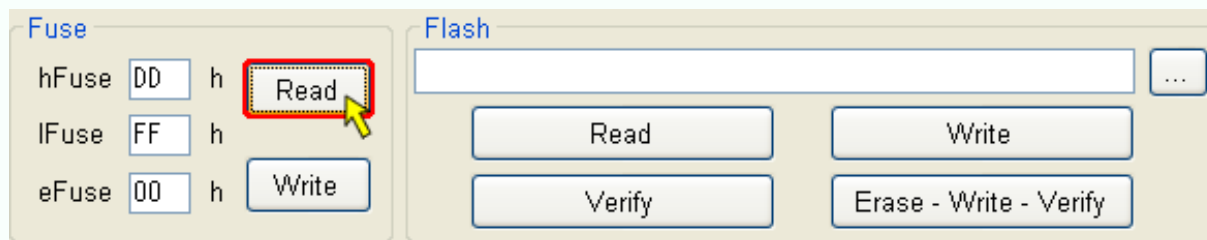
- i. Wiring

Connect the pins of X3 and the pins of ICSP by wires. Please see the photograph below. (click to enlarge)



[another wiring](#)

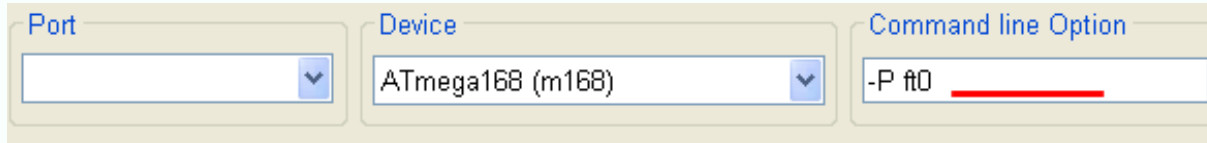
- Connect Diecimila and a PC by a USB cable.
- Run the avrdude-GUI.
- Read the Fuse-bits.
- Click on the [Read] button of Fuse area.



- If the fuse-bits are read, it is working.

When an error is displayed, it is not working. confirm wiring and setting of avrdude-GUI.

- Run it by fast clock mode.
- Delete "-B 4800" of Command line Option area.



The screenshot shows the AVR-Writer software interface. It has three main sections: 'Port', 'Device', and 'Command line Option'. The 'Port' section has a dropdown menu with a blue arrow pointing down. The 'Device' section has a dropdown menu with 'ATmega168 (m168)' selected. The 'Command line Option' section has a text box containing '-P ft0' followed by a red horizontal line.

- Read the Fuse-bits. again.

If the fuse-bits are read, it is working by fast clock mode.

- Burning the bootloader.
 - i. Set the new ATmega168-chip.
 - Disconnect USB-cable from the Diecimila.
 - Remove ATmega168-chip from the Diecimila-PCB.
 - Set the new-chip onto the Diecimila-PCB.
 - Connect USB-cable to the Diecimila again.
 - ii. Writing the Fuse-Bits.
 - Set slow clock mode.

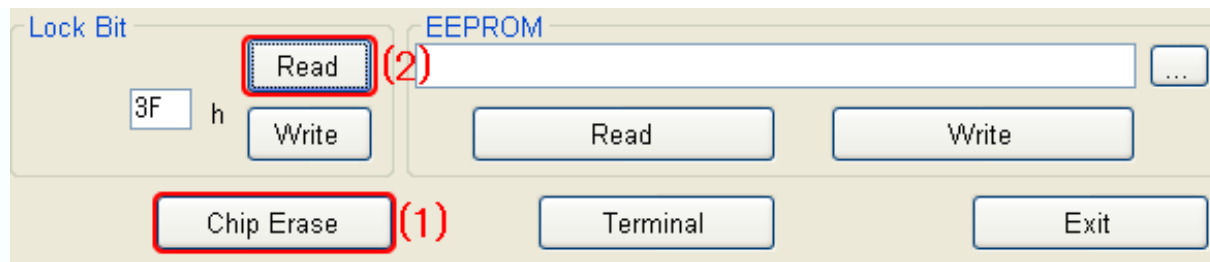
Key in "-B 4800" to text-box of Command line Option area.

Attention !!! A virgin chip cannot run by fast clock mode.

 - Erase

(1) Click on the [Chip Erase] button.

(2) Click on the [Read] button of Lock Bit area. The value 3F of Lock Bit should be displayed.



- o Writing the Fuse-Bits.

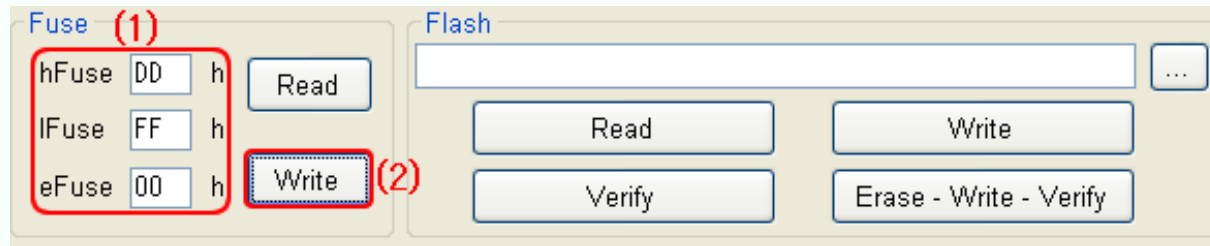
(1) Set the fuse-bits as below. (These settings are for only ATmega168.)

hfuse = DD

lfuse = FF

efuse = 00

(2) Click on the [Write] button of Fuse area.




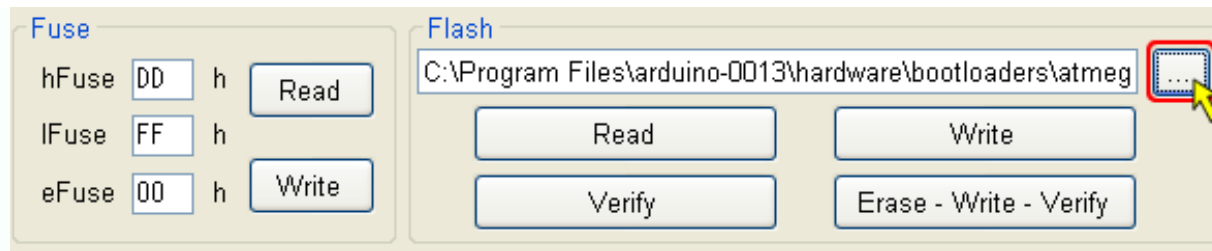
- o Switch to fast clock mode.

Delete "-B 4800" Command line Option area. And read the Fuse-Bits.

When do not read the fuse bits, probably, you had a mistake to set the Fuse-Bits. Return to slow clock mode. And write the Fuse-Bits again precisely.

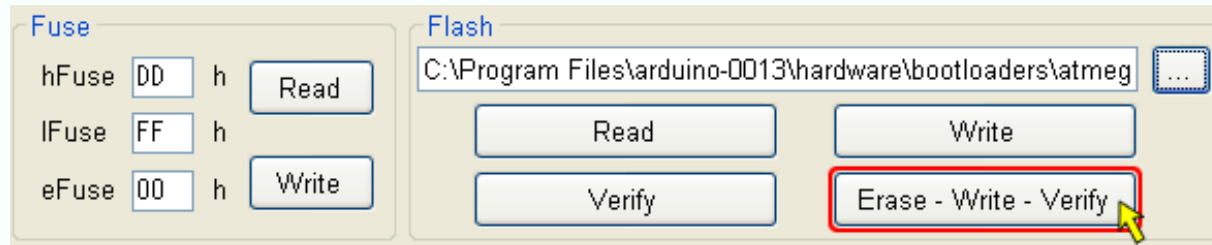
iii. Preparing Bootloader hex-file.

- o Click on the  of the Flash area.
- o Open the "C:\Program Files\arduino-0013\hardware\bootloaders\atmega168" folder.
- o Select "ATmegaBOOT_168_diecimila.hex", and click on the [Open(O)] button.



iv. Burning

- o Click on the [Erase Write Verify] button of Flash area.
- o Wait until this working will be finished.



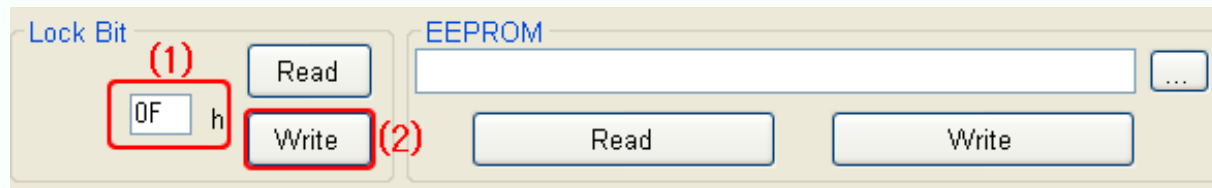
v. Writing the Lock-Bit.

- o (1) Set the Lock-Bit as below.

Lock = 0F

- o (2) Writing

Click on the [Write] button of Lock Bit area.



With the above burning of the bootloader is the end.

If you want to burn the bootloader onto

- **ATmega8** [\[Click here\]](#)

- **ATmega328P** [\[Click here\]](#)

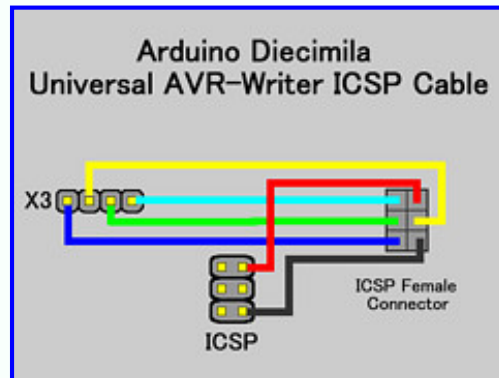
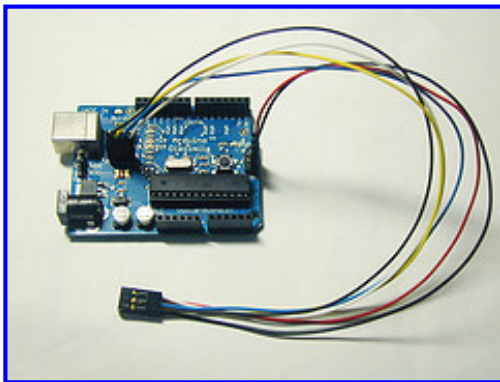
- Confirming.

- Disconnect USB-cable from the Diecimila.
- Remove the wires of ICSP and X3.
- Connect a USB cable to Diecimila again.
- Push the reset button of Diecimila.
- Start Arduino-IDE
- Upload sample sketch "Blink".

And it will be run.

- Diecimila become AVR-Writer

We can use Diecimila as an universal AVR-Writer if we use avrdude-serjtag,avrdude-GUI and the ICSP cable such as the photograph below. (click to enlarge)



Thanks

Very thanks Mr.Suz who developed BitBang Mode AVR-Writer software "avrdude-serjtag", and Mr.Yuki who developed GUI-wrapper of avrdude.exe.

Links

[avrdude on FT245R/FT232R by Suz AVR-Research](#) --> [\(Translation to English by Google\)](#)

[avrdude on FT245R/FT232R\(Linux\) by Suz AVR-Research](#) -->[\(Translation to English by Google\)](#)

[FTDI Drivers for avrdude in Linux by Coffeebot Labs](#)

[avrdude-GUI \(yuki-lab.jp Version\)](#) --> [\(Translation to English by Google\)](#)

[BitBang Mode AVR-Writer on Mac](#) --> [\(Translation to English by Google\)](#)

[Another Hack] [Arduino meets Processing via WiFi](#). How to make cheap WiFi environment for Arduino by FON's router.

[\[Return\]](#)

update 2009.03.20 avrdude.conf (update on 03.18 failed. update again)

update 2009.03.19 [about ATmega328P](#)

update 2009.03.18 avrdude.conf (add device data script for ATmega88P,ATmega328P)

renew 2009.03.07

update 2008.11.07

upload 2008.07.27 05:48(JST)

