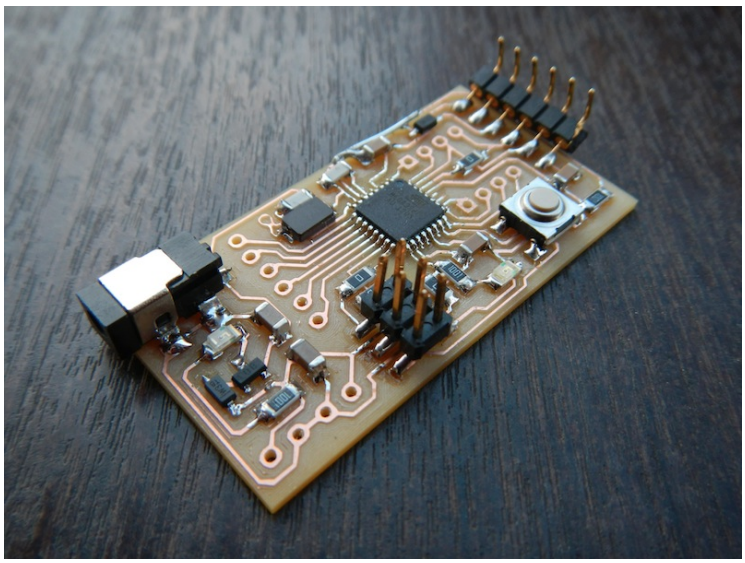
Barduino

A fully featured and FabLab compatible Arduino clone with a terrible name



Originally designed around the through-hole (28 DIP) version of the Mega168/Mega328 micro-controller in an attempt to simplify soldering for workshop students, the Barduino (please help me find a better name) has since been updated to use the SMD versions these same micro-controllers.

This board is based on the Fabian Arduino project developed by Shawn Wallace at AS220. It adds a few components that make it a little easier to program and use. These are:

- Power LED
- PIN 13 LED
- ICSP Connector
- 16 MHz Crystal
- Power Jack
- Protection Diodes
- Headers

The LED's have been added to make the debugging process easy, if the power LED does not come on during your smoke test then you definitely have a problem. On the other hand if it does come on it doesn't mean your board is without fault. The LED on pin number 13 (in the arduino IDE) allows you to quickly load the standard blink example. If this test is successful your board if good to go.

The ICSP header has been added to simplify the programing sequence but adds some bulk as it is the tallest component. If you want to keep it compact and save a few cents you can connect the 6 Pin header (ICSP) to the connector at the end of the ribbon cable that goes to your fabISP and simply touch it to the board whilst programing. Pay attention to the orientation.

The Crystal is added simply for better sensitivity though better timekeeping. The power jack is a nice addition that will save you from soldering cables straight onto the PCB and protects you from yourself when it comes to shorts.

The protection diodes allow you to power the circuit simultaneously through external power such as a 9V battery and the FTDI cable. This is particularly useful whilst programing and does away with jumpers which are a pain to be changing all the time.

Finally the headers, which are optional (and not pictured) make it easy to use this board to

prototype quickly, much like you would use a regular commercial arduino. Simply connect sensors, LED's or jumper wires straight onto the headers without the need to solder and unsolder. You can also use screw terminals which will make your connections much more reliable in the long run, this is a good idea for a final project or anything you wish work reliably for anything longer then a few hours.

Design files

These are the various design files needed to make and customize your board. If you wish to make it as is go ahead and download the two FabModules compatible PNG images. These have been fully tested.

- **barduino.traces.png**
- **barduino.outside.png**

Feel free to download and edit the eagle files for your specific needs. Maybe you want to add some SMD sensors, more LED's or simply integrate this circuit into a larger project.

- **barduino.sch**
- **barduino.brd**

This is a pinout of the final board which displays the pin numbers according to the arduino IDE conventions so it will come in handy when testing and working with your barduino. Digital Pin 4 is not routed to a header at the moment, which is not a big deal but I will do that as soon as possible (as always, feel free to do that if you want).

- **pinout guide**

BOM

All but a few of the components are included in the FabLab Inventory and you should have in stock at your lab. The two that are not in the inventory can be easily purchased at the usual suppliers. Check PDF below for quantities. Keep in mind that you should choose either the 168 or the 328 not both. Only difference is available memory, 16KB for the 168 and 32KB for the 328. **EDIT: Looks like the 328 is now part of the inventory for this year so you might have it in your lab.**

The headers listed have 6 positions, which is not ideal but they are the only ones I managed to find at digikey. Normally I would use long 40 position headers and but them to size (6, 4 and 2). I will keeping looking but for now you should be able to use these and cut them to size, they are pricy though so you might be able to find a better alternative locally. Keep in mind you will loose one position when cutting them to size.

- Barduino BOM PDF

Inventory components

- Resistor 1K
- Resistor 0
- Capacitor 10uf
- Capacitor .1uf
- Red LED
- Green LED
- Shottky Diode
- 5V Regulator
- Power Jack
- ICSP Connector
- Button
- FTDI Header
- ATMega 328

Non-Inventory components

- Crystal 16 Mhz
- Capacitor 18pF
- 6 POS Header

TIP - *The links above are to digikey US. Remember to change your country preference on their site if you are outside the US. Seems obvious but I have had students wait over one month for parts because they assumed the site would detect their location (which it doesn't) and so parts were shipped form the US.*

Milling tips

The original intention was to be able to fit 3 barduinos inside a 3x2 inch FR1. I have not yet managed to get it small enough although it's not very far. Height wise we are below 2 inches so, good to go. Width wise it needs to be below 1 inch, which I am very close to (1.0227). For now please use the larger (6x4) FR1 stock and nest it according to your need. This is a one sided design, so a one sided stock is recommended.

The center of the micro controller tracks look funny. I had a problem with those thin tracks lifting and this is a trick I came up with to make them a little stronger. When they are short (pads) they tend to lift, by adding a little length they have more contact surface so they tend to stay put.

Stuffing tips

These SMD micro controllers can be a little scary the first time but are very easy to solder using the "gloop and subtract" technique. Apply a small amount of solder to one of the corner traces of the micro controller on the milled board. Making sure that the orientation is correct (check register circle or dot on micro controller and pinout guide) align the micro controller to the traces using your precision tweezers.

Once oriented correctly put your finger over the chip and apply a little pressure so it doesn't move. Carefully approach the tip of your soldering pencil to the pin where you previously applied the solder. Once you have managed to flow this small amount of solder to you chip leg it should

be fixed. Turn it upside down to make sure it's stuck. Check the positioning making sure all pins of the micro controller line up with the traces. You can still reposition the chip at this point, to do so, simply apply heat to the one leg that is soldered and use your tweezers to align it.

Once you feel the orientation is correct, simply apply a generous amount of solder to all pins, they will be stuck together at this point. Do no freak out. Continue to go around, making sure not to apply too much heat to any one area for too long.

Now simply use your Quickbraid to remove the excess solder, again, not lingering for too long in any one area.

Video and pictures of this process coming soon.

Programming

Once soldered perform the smoke test by connecting the FTDI cable to the board and to your computer, or alternatively, to a 9V battery through the DC Jack. If the red LED comes on, move to the next step.

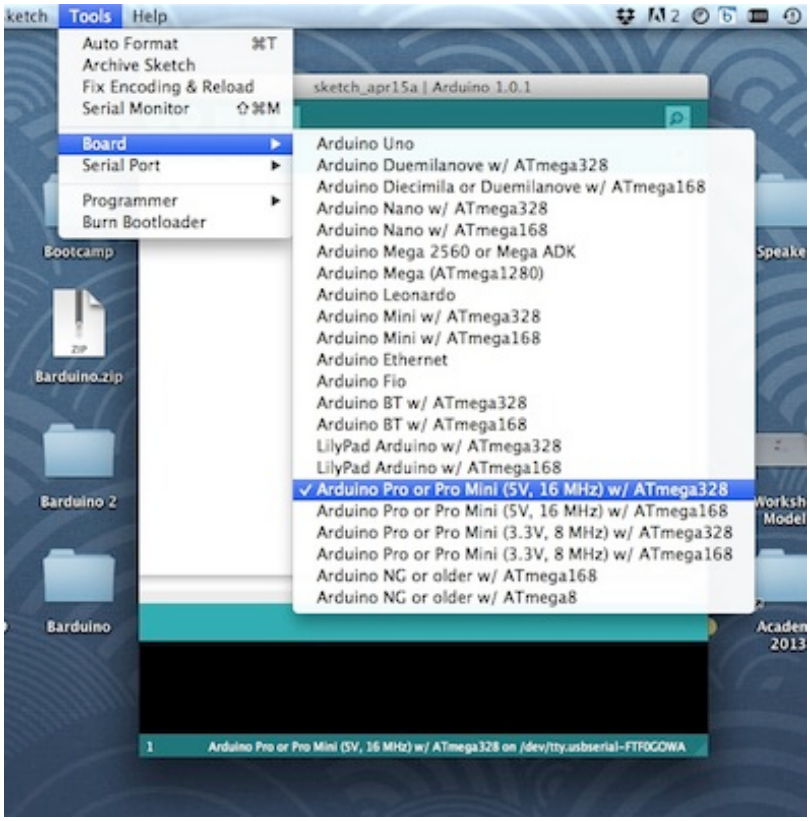
To program your board simply connect it to your fabISP (beware of orientation) and to a power supply (either one as described above). Now connect the ISP to your computer and launch the Arduino IDE. Select the correct programmer from the tools menu (fabISP or AVRISP MKII), select the board according to the micro controller you used by choosing either:

Arduino Pro or Pro Mini (5V, 16MHz) w/ ATMega 328

or

Arduino Pro or Pro Mini (5V, 16MHz) w/ ATMega 168

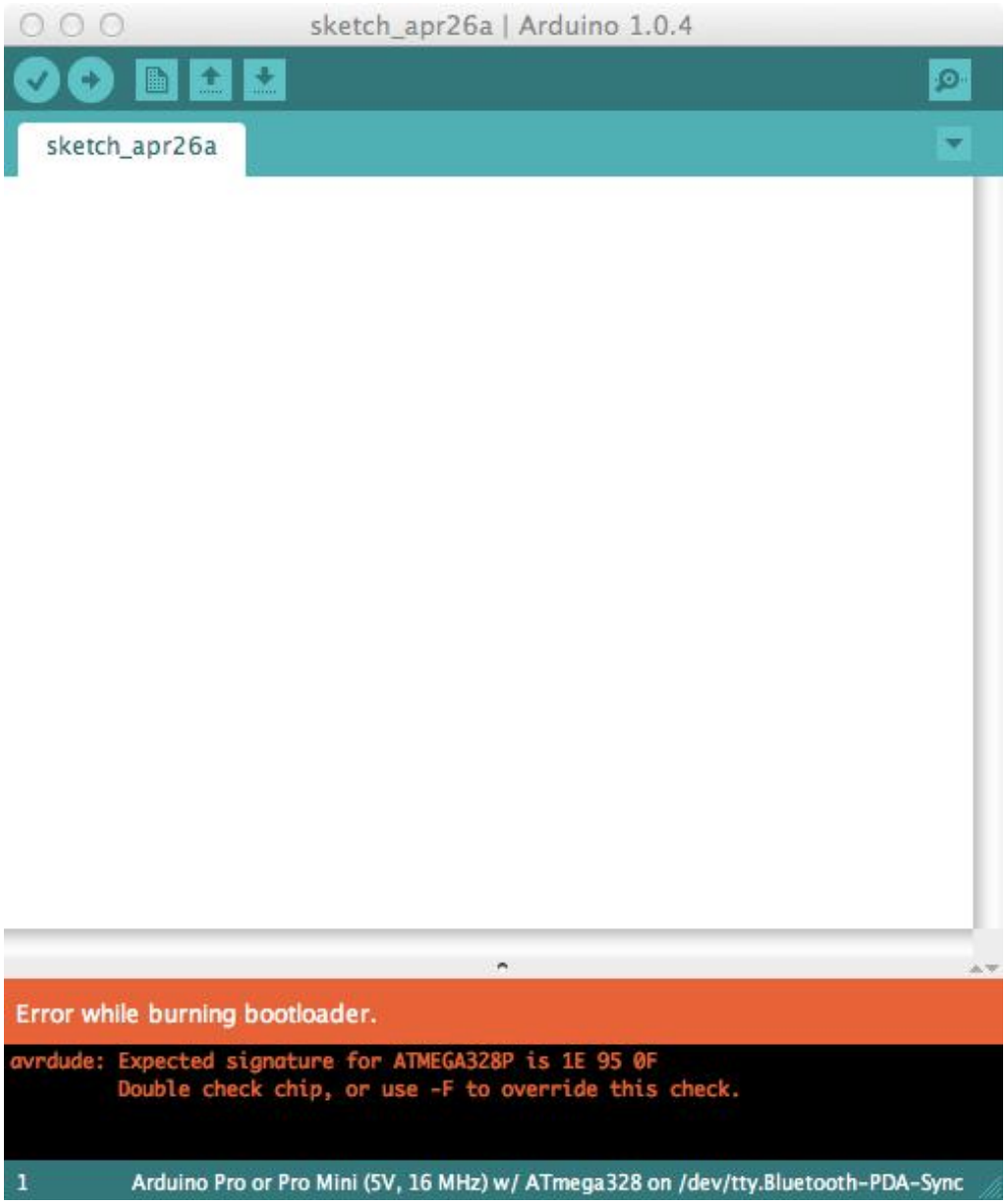
TIP - *These boards come preloaded with the IDE and there is no need to mess with the boards.txt file!*



Hopefully you got a success message (Done!) after the last operation and your barduino is ready to go. Congratulations!

To make sure, connect the FTDI cable (this should provide enough power so there's no need to plug in the battery) to the barduino and load the blink example from the arduino IDE.

It should come to life and say hi by blinking the green LED once every second. If not, don't worry, here are the solutions for a few common problems:



This is a relatively common error, and happens due to the fact that there are a few different versions of the ATMega328 and ATMega168, such as 328P or 328AU. They are almost identical and 100% interchangeable but carry different signatures.

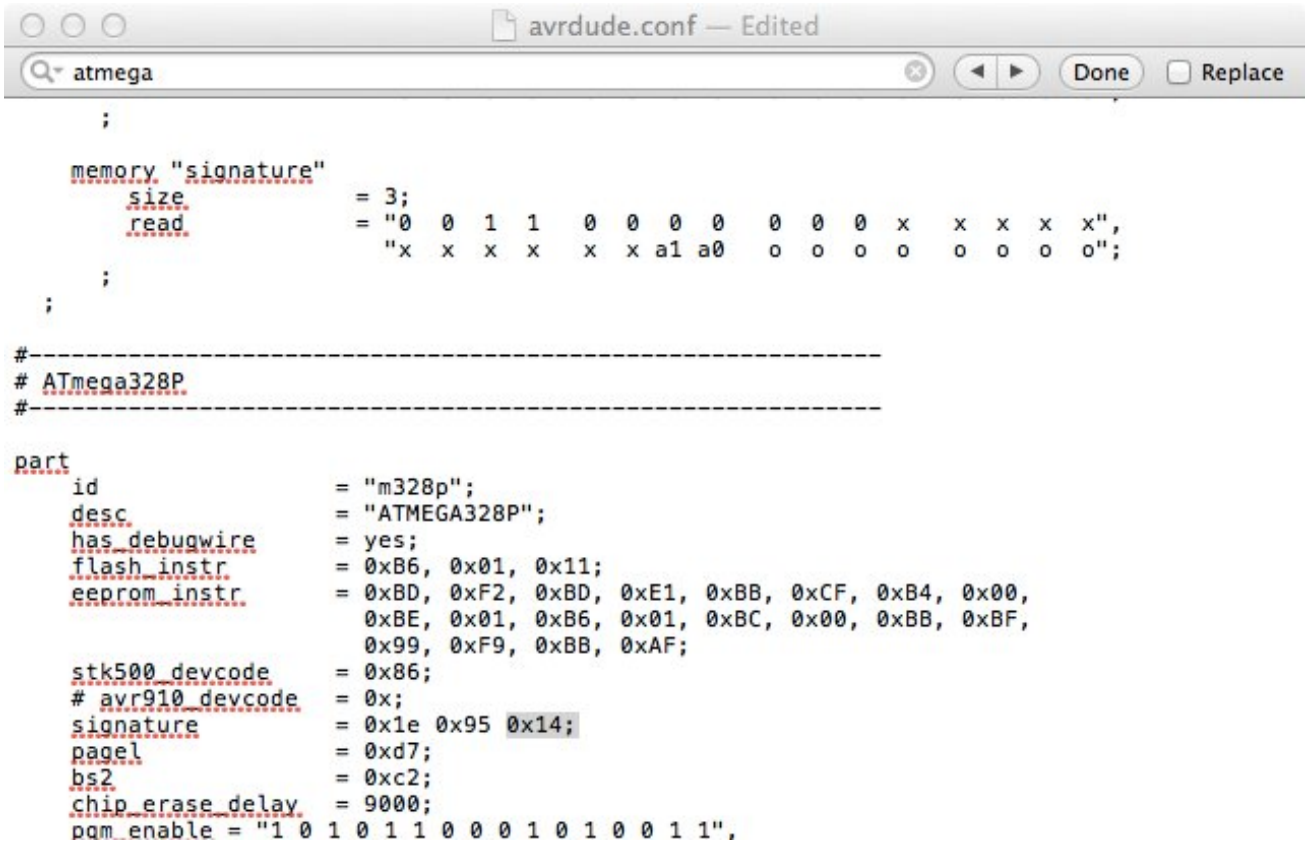
To change the signature AVRdude expects, navigate to the correct folder shown below according to your OS and open the file named **avrdude.conf** with a text editor:

Mac:
MacintoshHD/Applications/Arduino/Contents/Resources/Java/hardware/tools/avr/etc/avrdude.conf

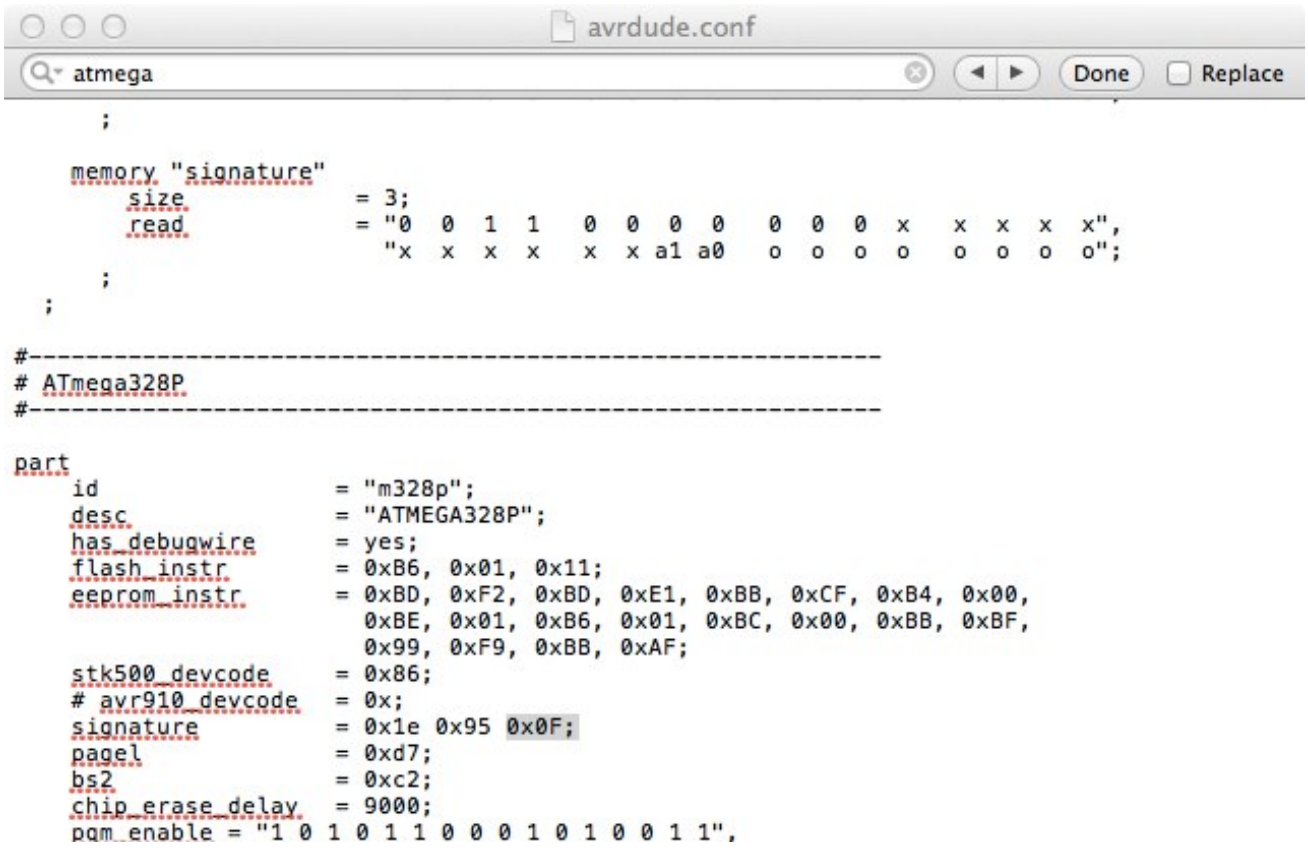
TIP - *On OSX an "application" is really a folder. To view inside it simply right click (secondary click) on the application icon and choose "show package contents" from the menu.*

Windows: ?/Arduino-1.x.x/hardware/tools/avr/etc/avrdude.conf

Search for the name of the micro controller that was mentioned on the error message. In the case above, 328P.



Here we can see the expected signature for the 328P ends with the 0x14 bit. Because we had used a 328AU the signature on our chip ends with the bit 0x0F. We can just go ahead and make that change and save the file.



You should now be able to burn the bootloader correctly. This is a quick fix and it might have you running back and forth changing the avrdude.conf as you try to program different boards. A better long term suggestion would be to add another entry on the avrdude.conf file for the 328AU and 168AU and change the boards.txt file to reflect this. I will work on this and try to host alternate versions of both those files to make it easier in the future to deal with this.

Here is a little cheat sheet for the signatures:

- 328 0x1E 95 **0F**
- 328P 0x1E 95 **14**
- 168 0x1E 94 **06**
- 168P 0x1E 94 **0B**

The other common error message talks about AVR Dude and STK500 (image coming soon), this means that you have some bad soldering and should go back and physically check your board using some sort of magnification, specially around the micro controller.

Congratulations!
Luciano

If you have any questions please email me on luciano at fablabbcn.org