

# How to Update Ares Firmware in OctoPi?

by [valleyman](#) » 2016-02-13 22:10

## [1. Transfer firmware package](#)

### [1.1. RAR package](#)

### [1.2. Use your computer's network file system](#)

### [1.3. Command line alternative](#)

## [2. Upload firmware to Arduino](#)

### [2.1. Move firmware to sketchbook](#)

### [2.2. Install and hack Time library](#)

### [2.3. Connect to Arduino board](#)

### [2.4. Load Ares firmware into IDE](#)

### [2.5. Adjust firmware and upload to Arduino](#)

### [2.x. For the paranoid \(That's me\)](#)

[Ares 3D printer](#) uses two micro-controller boards. The printing process is controlled by an [Arduino](#) (Mega?), and the rest of the system (user interface, file management, command input, etc.) runs on a [Raspberry Pi](#) running the [OctoPi](#) operating system. OctoPi also runs auxiliary functions such as [Arduino IDE](#) which can be used to update firmware. What is commonly referred to as the printer's "**firmware**" ([Marlin](#) in this case) runs in the Arduino. When you get a firmware update (or if you just need to adjust your firmware), you'll need to

1. Transfer the firmware to OctoPi.
2. Log onto OctoPi and upload firmware to Arduino.<sup>1</sup>

Obviously you'll need the following.

- A firmware package. Easy Arts posted some [Official Ares Firmware](#) in [RAR format](#). I think most of us received a printer with 22cm rods, so choose the package compatible with 22cm rods.
- A working network connection to your Ares, a mechanism to transfer files to OctoPi, and an RDP client such as Microsoft Remote Desktop to log on to OctoPi.

Below are steps to refresh your Ares firmware.

## 1. Transfer firmware package

### 1.1. RAR package

RAR packaging software is not included in the OctoPi that comes with Ares. I just installed one on my Mac and unpack the posted firmware package locally before the transfer.

---

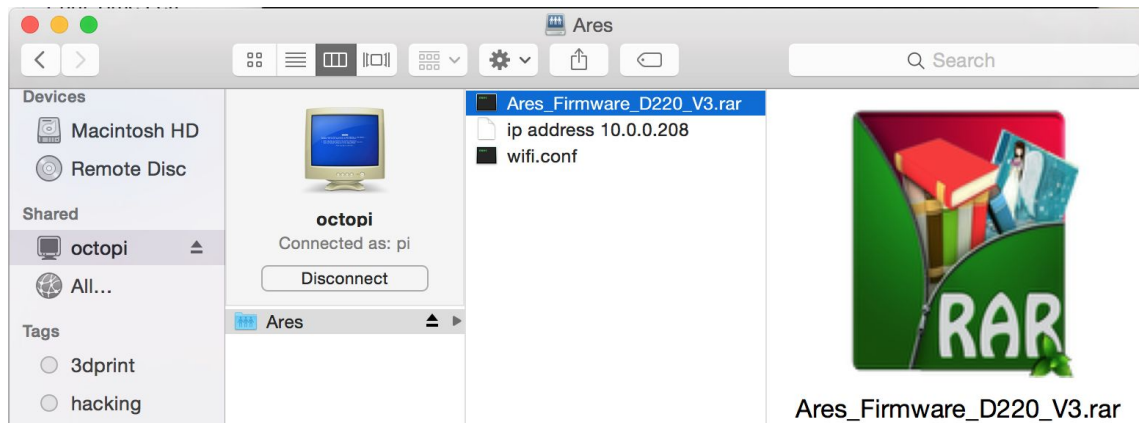
<sup>1</sup> In embedded programming, it is conventional to say "download" firmware to a micro controller, although this is not as common among Arduino programmers. To avoid confusion, we describe the "upload" from your computer to the Pi as "transfer".

But you can choose to install one on OctoPi from the network, e.g.,  
`apt get unrar-free`

Note: After you unpack the .rar file, it produces three layers of directories before getting to the actual firmware. I took the second level directory, `220_Ares_Firmware/`, which is the most identifiable to me.

### 1.2. Use your computer's network file system

To transfer, you also have several options. For many, using SMB that is natively supported by Windows, Linux and Mac is perhaps the easiest. The caveat is that the package ends up in the directory `/home/pi/Ares/`, which can be undesirable. Easy Arts' [How to connect your Ares 3D printer](#) includes screenshots about how to view contents in this directory using Windows Network Neighborhood. My [Ares \(OctoPi\) Notes for Mac Users](#) includes screenshots in Mac's Finder. Once the network drive shows, you just drag and drop.<sup>2</sup>



### 1.3. Command line alternative

On Mac and Linux, it is easy to use command line to transfer files. (That's what I eventually did after realising that OctoPi doesn't have unrar and decided not to install one on the Pi.)

```
scp -pr 220_Ares_Firmware/ pi@octopi:sketchbook/
```

Upon Enter, scp will ask for password. (If you haven't changed it, the default is "raspberrypi".)

## 2. Upload firmware to Arduino

This assumes that you know how to connect to OctoPi by using a remote desktop.<sup>3</sup> See Easy Arts connections guide and my Mac notes linked above.

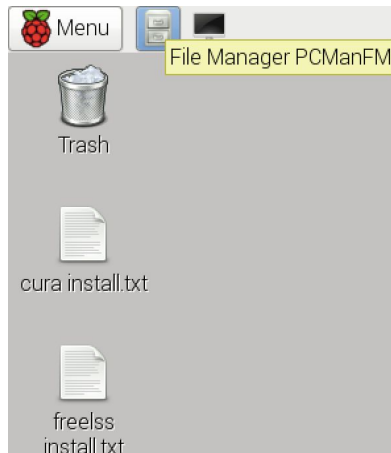
---

<sup>2</sup> The screenshot shows the .rar package. You can upload the entire unpacked firmware directory the same way - it saves you the trouble of installing unrar on the Pi.

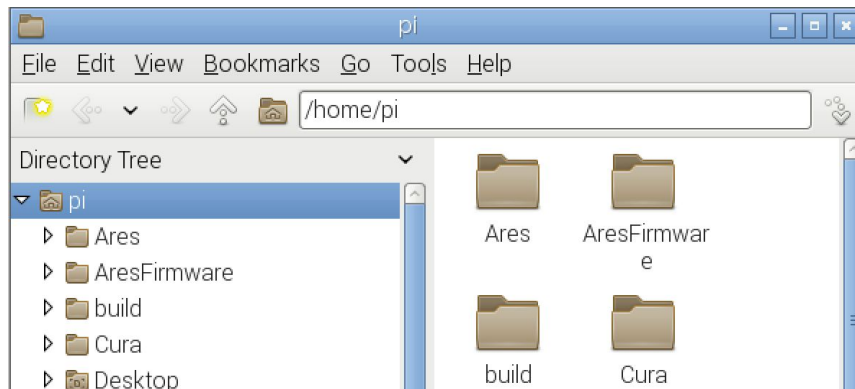
<sup>3</sup> Arduino IDE only runs in graphic mode. Although you can run its underlying tools from command line, hence from remote, most probably won't go to that length.

## 2.1. Move firmware to sketchbook<sup>4</sup>

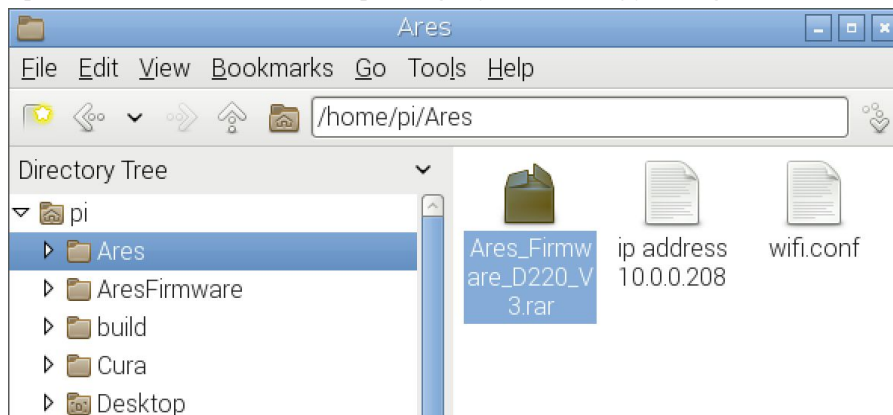
If you have transferred the firmware using your operating system's network file support, first move the package out of the Ares/ directory. Open File Manager (the cabinet on the top).



File Manager will automatically open user pi's home directory.



Open Ares/.<sup>5</sup> You'll see the package (or directory) alongside with wifi.conf.<sup>6</sup>

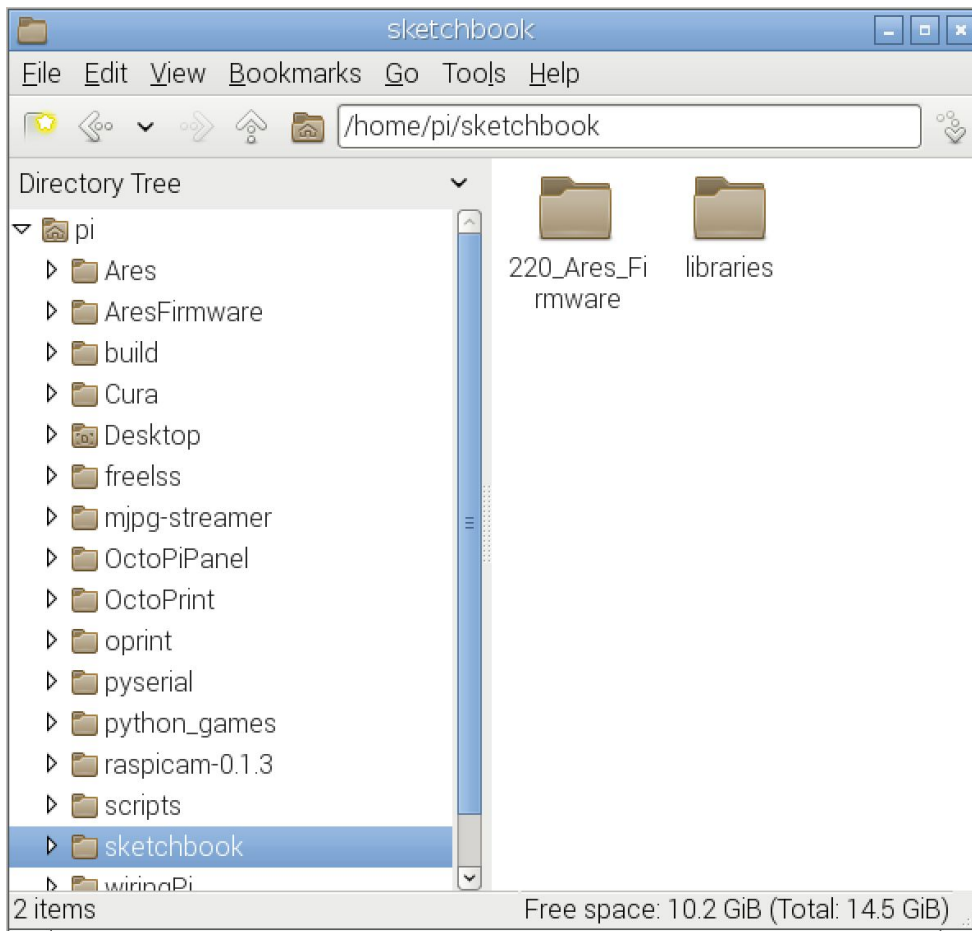


<sup>4</sup> Skip this step if you have scp'd the firmware directly into sketchbook/.

<sup>5</sup> No need to look under AresFirmware/, as it is empty.

<sup>6</sup> The following screenshot shows the RAR package instead of unpacked directory.

They don't belong there. So drag the object and throw it into the sketchbook/ directory slightly down in the tree view.



## 2.2. Install and hack Time library

Marlin requires [Arduino's Time library](#) which is not found in Ares' OctoPi installation.

In an OctoPi terminal, (alternatively ssh to Ares)

```
wget https://github.com/PaulStoffregen/Time/archive/master.zip
unzip master.zip
mv Time-Master sketchbook/libraries/Time
cd !$
ln -s Time.h time.h
```

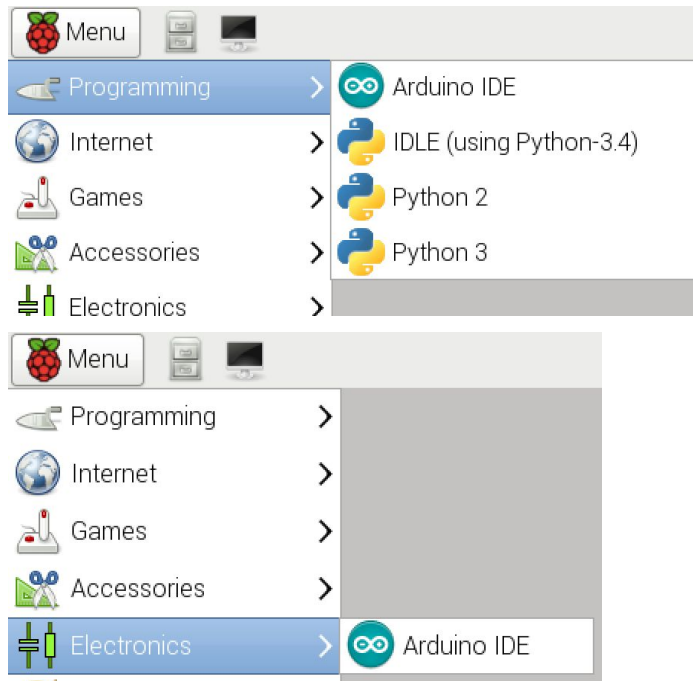
The last step is to hack around a [Marlin bug](#). In my opinion, Easy Arts should have included library and the hack in shipped printers.<sup>7</sup>

---

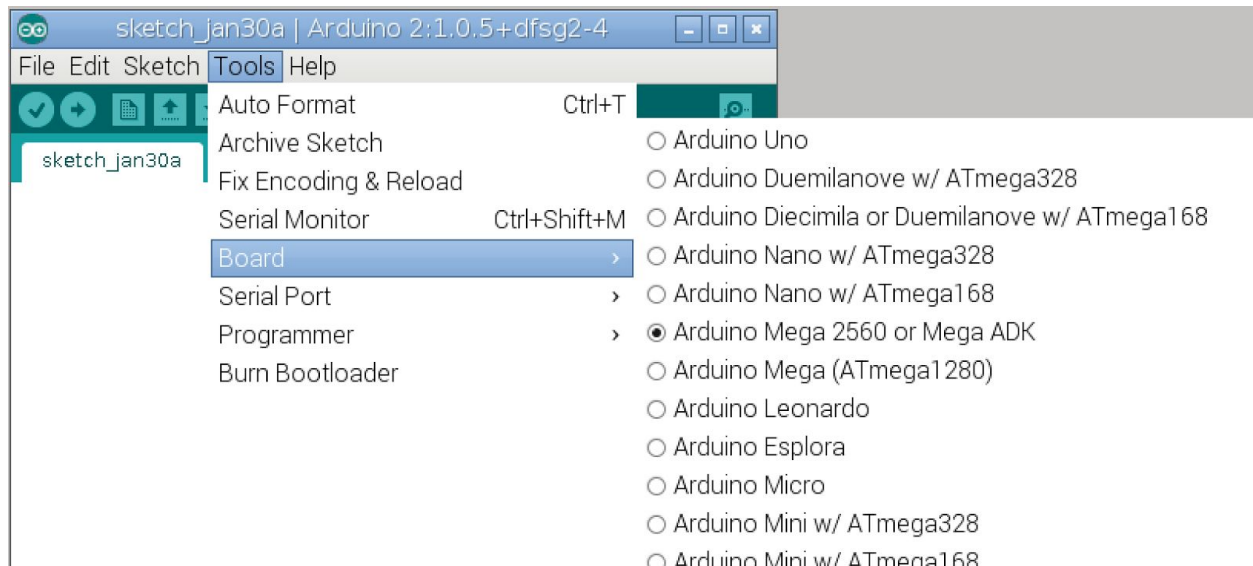
<sup>7</sup> An alternative hack as Francis did is to edit `qr_resolve.cpp` to use `Time.h` (with capital "T") instead of `time.h`. But this requires you to fix every update till the bug is fixed in the source. My hack will work always.

## 2.3. Connect to Arduino board

Arduino IDE can be found under two menus in OctoPi, first under “Programming”, then under “Electronics”. They both link to the same installation.

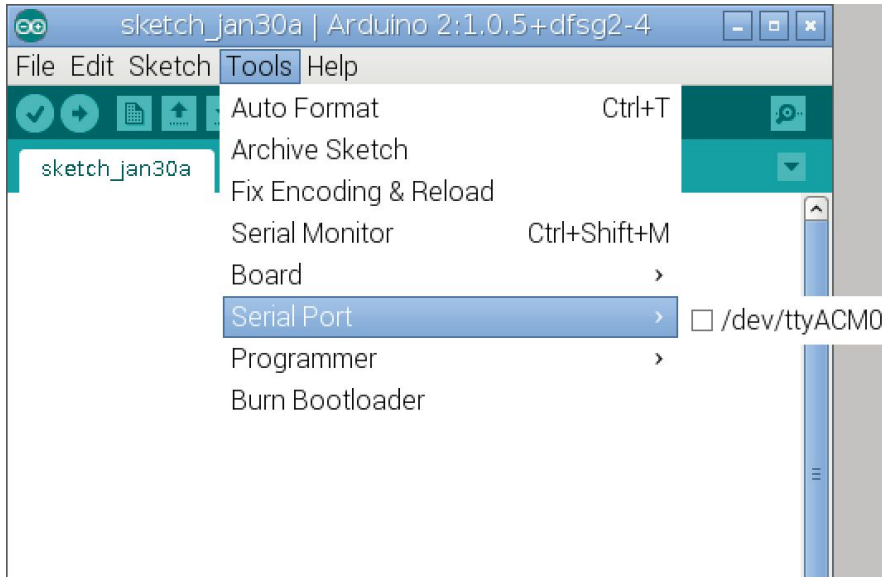


Once inside Arduino IDE, you want to first check board selection by using “Tools -> Board”. My printer’ IDE already selects “Mega 2560 or Mega ADK”, which is the correct target platform for Marlin.

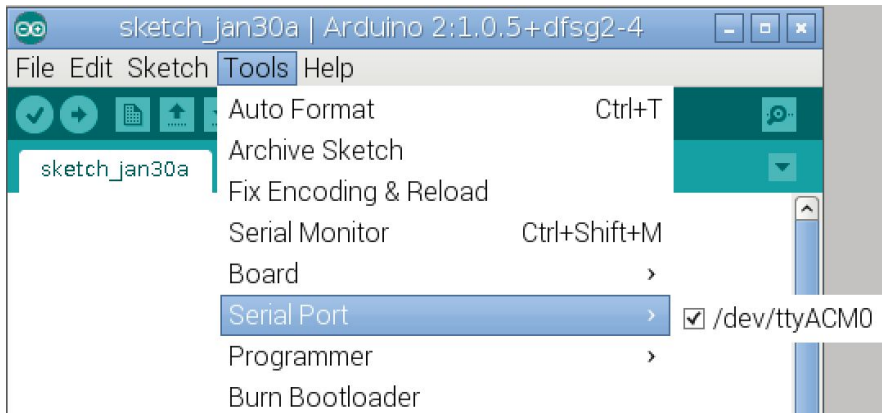


Next, check serial port. My printer’s IDE was *not* connected.

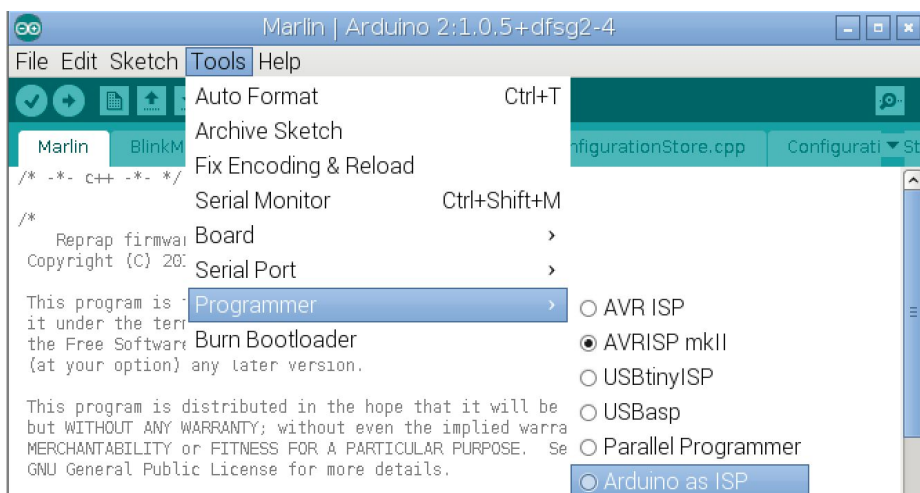
## How to Update Ares (Marlin) Firmware



As there is only one port, I check this one.



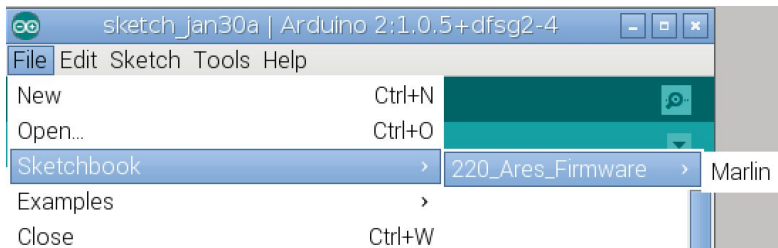
Lastly, check programmer. My printer's default selection is AVRISP mkII - which is not part of Ares.



Change to "Arduino as ISP".

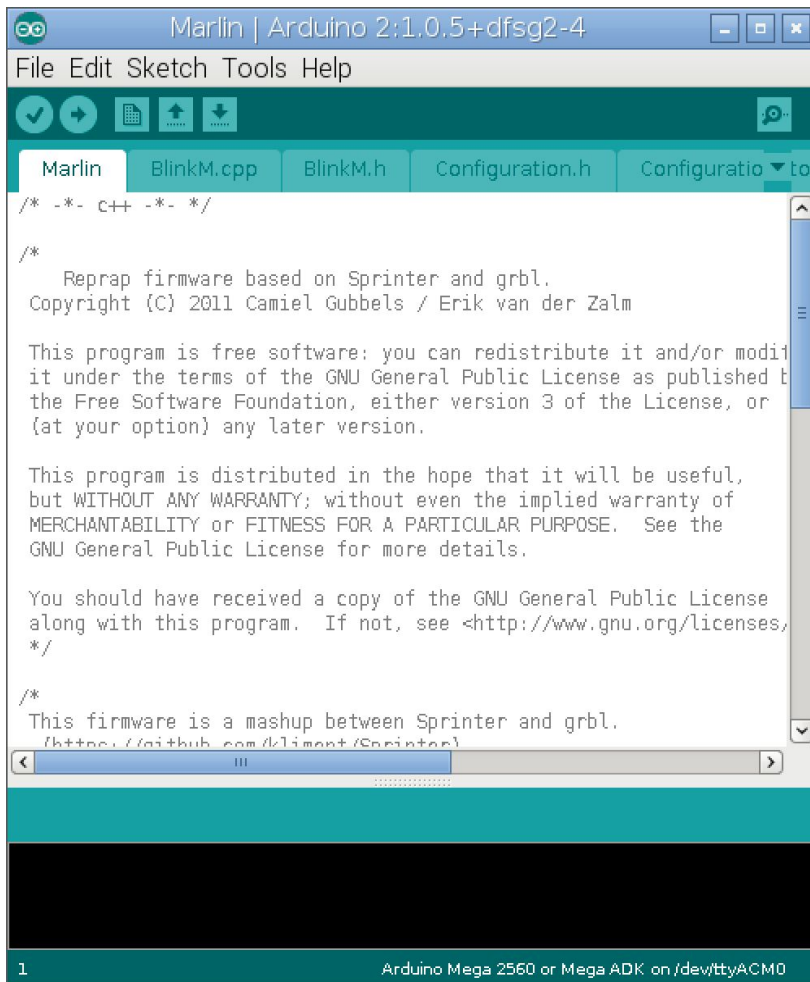
### 2.4. Load Ares firmware into IDE

With the package under `sketchbook/`, this is an easy step. Click “File -> Sketchbook”, you’ll see “220\_Ares\_Firmware” (or any other directory layer you have chosen) as a submenu. Select “Marlin” from underneath.



### 2.5. Adjust firmware and upload to Arduino

Once firmware is loaded, you can examine the various files, make changes, and so on. (You don't have to if you don't feel necessary or comfortable.)



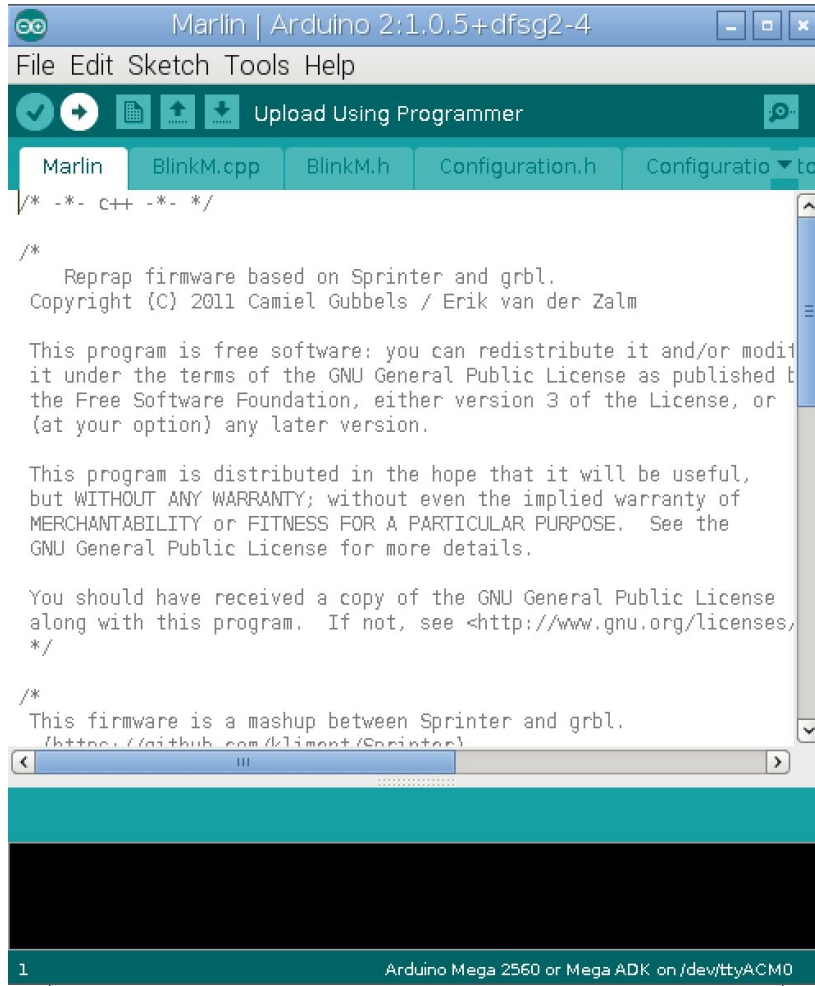


## How to Update Ares (Marlin) Firmware

Before upload, you must stop OctoPrint.

```
sudo /etc/init.d/octoprint stop
```

Now firmware upload is just one click away - by using the right arrow icon. Or you can watch it compile first by clicking the check sign icon.



That's all. You've refreshed your Ares firmware. Now restart OctoPrint

```
sudo /etc/init.d/octoprint start
```

or restart the whole thing; you can power cycle Ares as I did, or just reboot OctoPi.

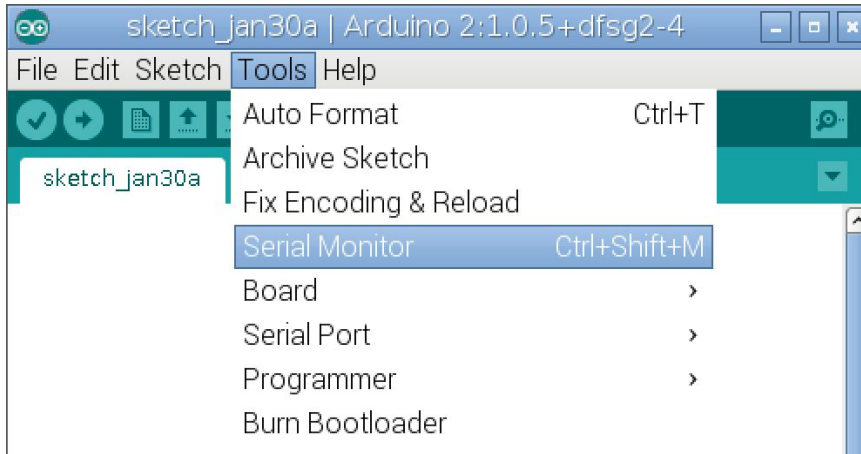
```
sudo reboot now
```

## 2.x. For the paranoid (That's me)

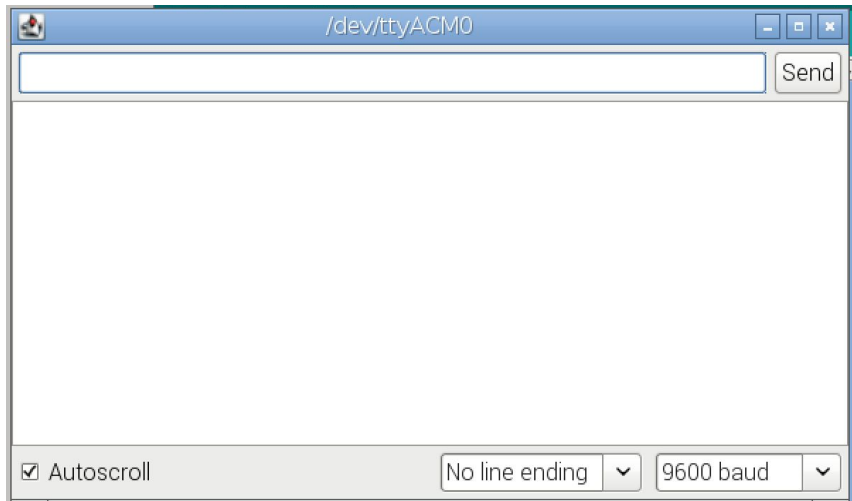
How do I know that the IDE actually connects to the printer's Arduino? I had to have some kind of confirmation before attempting a firmware update. So I opened its Serial Monitor.



## How to Update Ares (Marlin) Firmware



As is customary, default speed in serial monitor is 9600 baud.

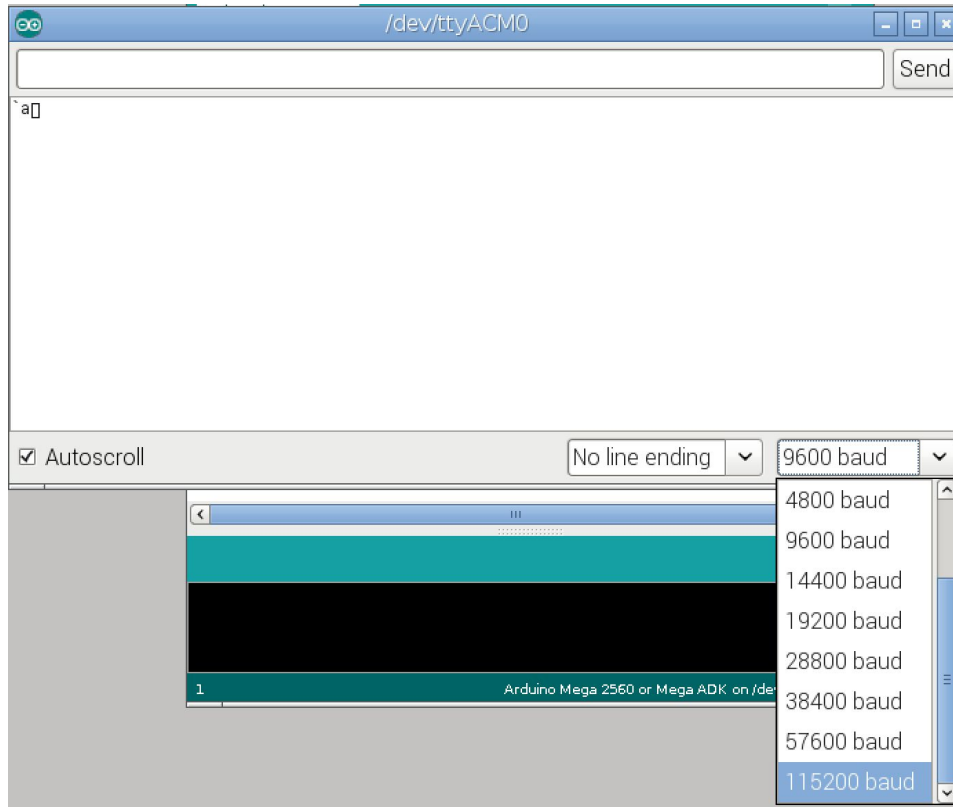


At this rate, you won't see anything even if OctoPrint<sup>8</sup> is running. (Well, if it is not running, you won't see anything, anyway, unless you program Arduino to say something at this baud rate.) So I changed it to the highest rate, 115200 baud as indicated in OctoPrint.

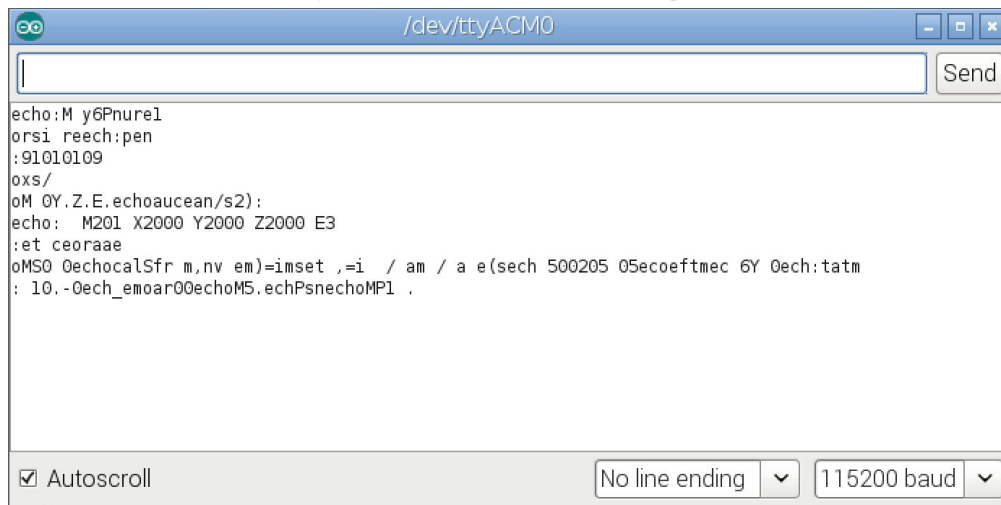
---

<sup>8</sup> [OctoPrint](#) is the Web based user interface for 3D printing that is the main application of OctoPi.

## How to Update Ares (Marlin) Firmware



Do not be discouraged by the gibberish that you may see at first. After selection, close the monitor and reopen it. The screen may be blank. Then, you can go back to your OctoPrint and send some command, any command that affects the printer. You should see some readable text.



Remember: This is just to verify that the IDE is connected to the Arduino. When you are printing, it is a bad idea to have serial monitor snooping on the line. So, close your monitor. Better yet, close Arduino IDE all together.