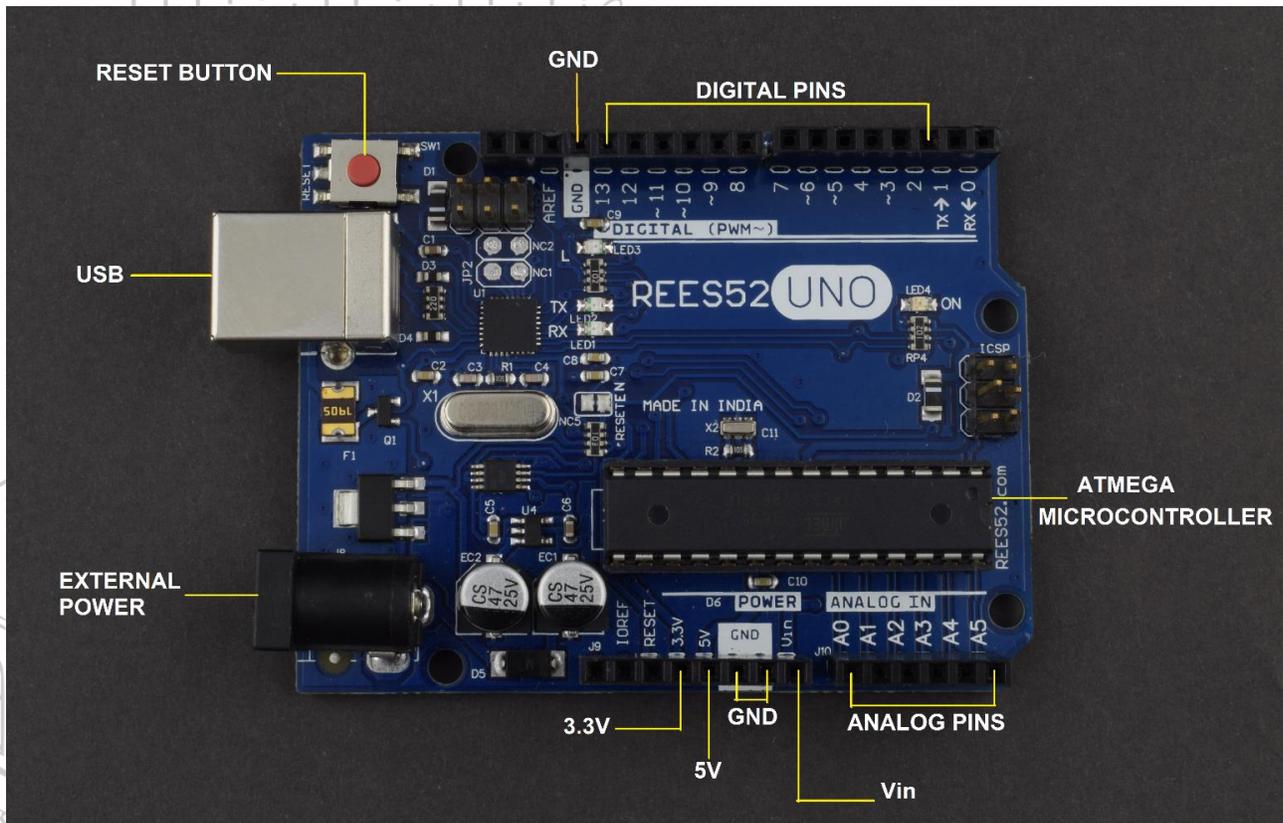


How to use Arduino Uno



Typical Application

- Xoscillo, an open-source oscilloscope
- Arduinome, a MIDI controller device that mimics the Monome
- OBDuino, a trip computer that uses the on-board diagnostics interface found in most modern cars
- Ardupilot, drone software and hardware
- Gameduino, an Arduino shield to create retro 2D video games
- ArduinoPhone, a do-it-yourself cellphone
- Water quality testing platform

Downloading / Installation

Go to www.arduino.cc to download the latest version of arduino software and select your operating system

On the Title bar Click on the Software Tab , Just scroll down once you will see this image

Download the Arduino IDE



ARDUINO 1.8.4

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

This software can be used with any Arduino board. Refer to the Getting Started page for Installation instructions.

Windows Installer

Windows ZIP file for non admin install

Windows app 

Mac OS X 10.7 Lion or newer

Linux 32 bits

Linux 64 bits

Linux ARM

Release Notes

Source Code

Checksums (sha512)

According to your operating system , like if you have windows system then choose Windows Installer.

Downloading / Installation



The screenshot displays the Arduino IDE interface. At the top, a toolbar contains icons for Verify, Upload, New, Open, and Save, with labels above them. Below the toolbar, the code editor shows the following code:

```
sketch_feb16b
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}
```

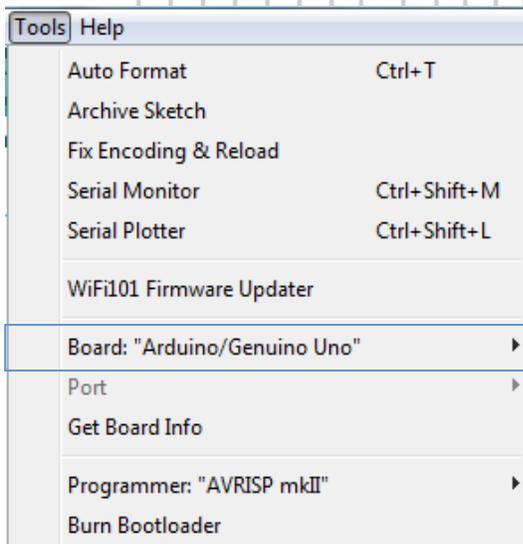
The IDE interface includes a teal header bar, a white code editor area, and a black status bar at the bottom. The status bar displays "Arduino/Genuino Uno on COM3".

REES52.COM

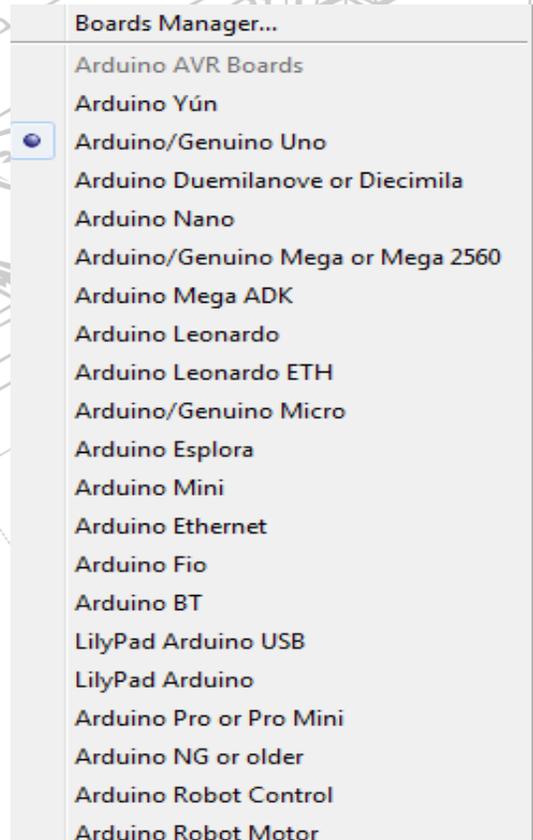
Arduino/Genuino Uno on COM3

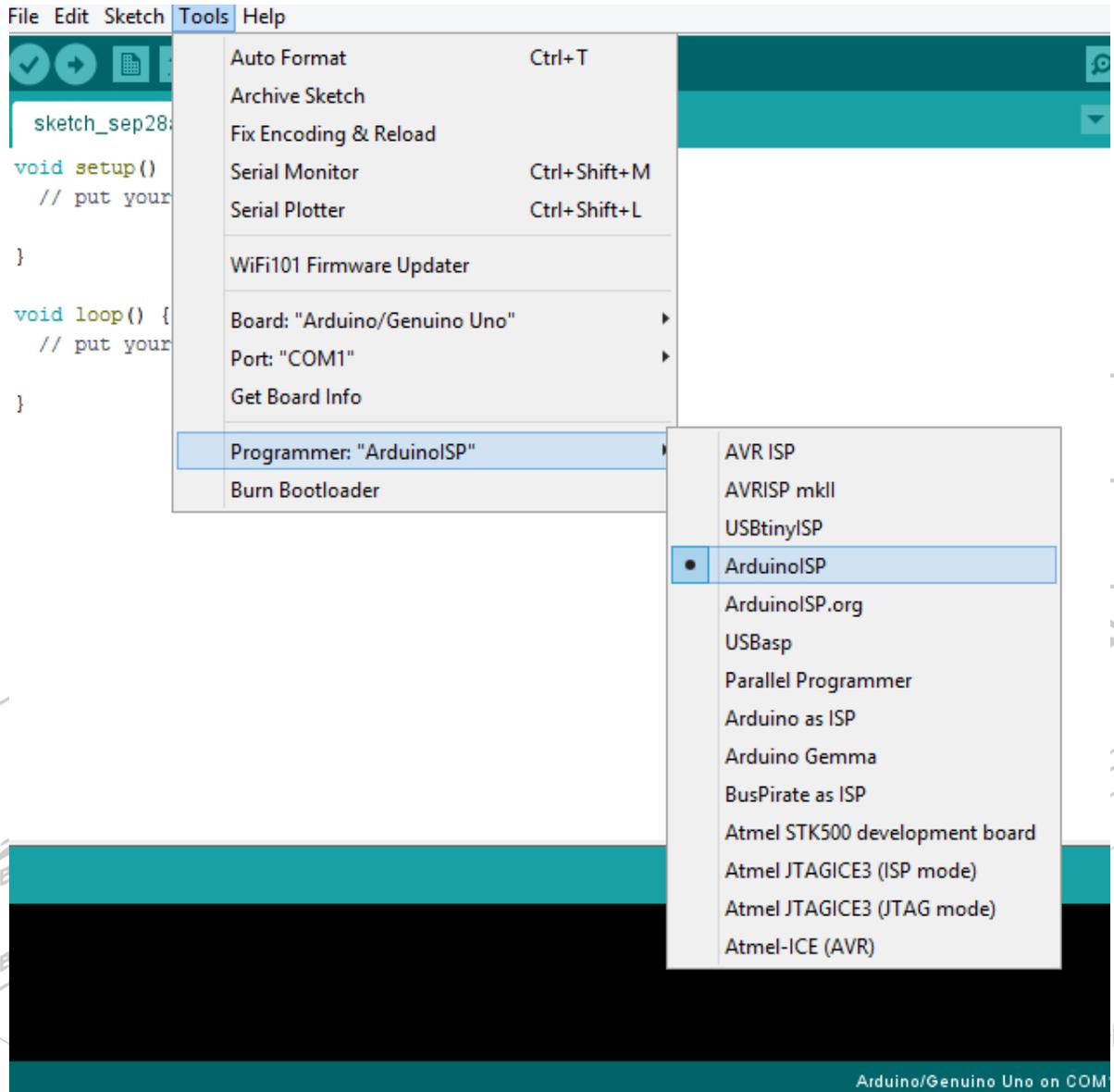
The Initial Set up

Select **Tools** menu and **Board**



Then select the type of Arduino board you want to program, in our case it's the **Arduino Uno**.





Select the programmer Arduino ISP , if this is not selected must select the Arduino ISP programmer . after connecting the Arduino must select the COM port.

Blink a Led

Connect the board to the computer. In the Arduino, software go to **File -> Examples -> Basics -> Blink LED**. The code will automatically load in the window.



```

Blink $

Blink
Turns on an LED on for one second, then off for one second, repeatedly.

Most Arduinos have an on-board LED you can control. On the UNO, MEGA and ZERO
it is attached to digital pin 13, on MKR1000 on pin 6. LED_BUILTIN is set to
the correct LED pin independent of which board is used.

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
}

```

1 Arduino/Genuino Uno on COM3

Press the **Upload** button and wait until the program says **Done Uploading**. You should see the LED next to pin 13 start to blink. Note that there is already a green LED connected to most boards – you don't necessarily need a separate LED.

Troubleshooting

If you are not able to upload any program to Arduino Uno and getting this error for "BLINK"

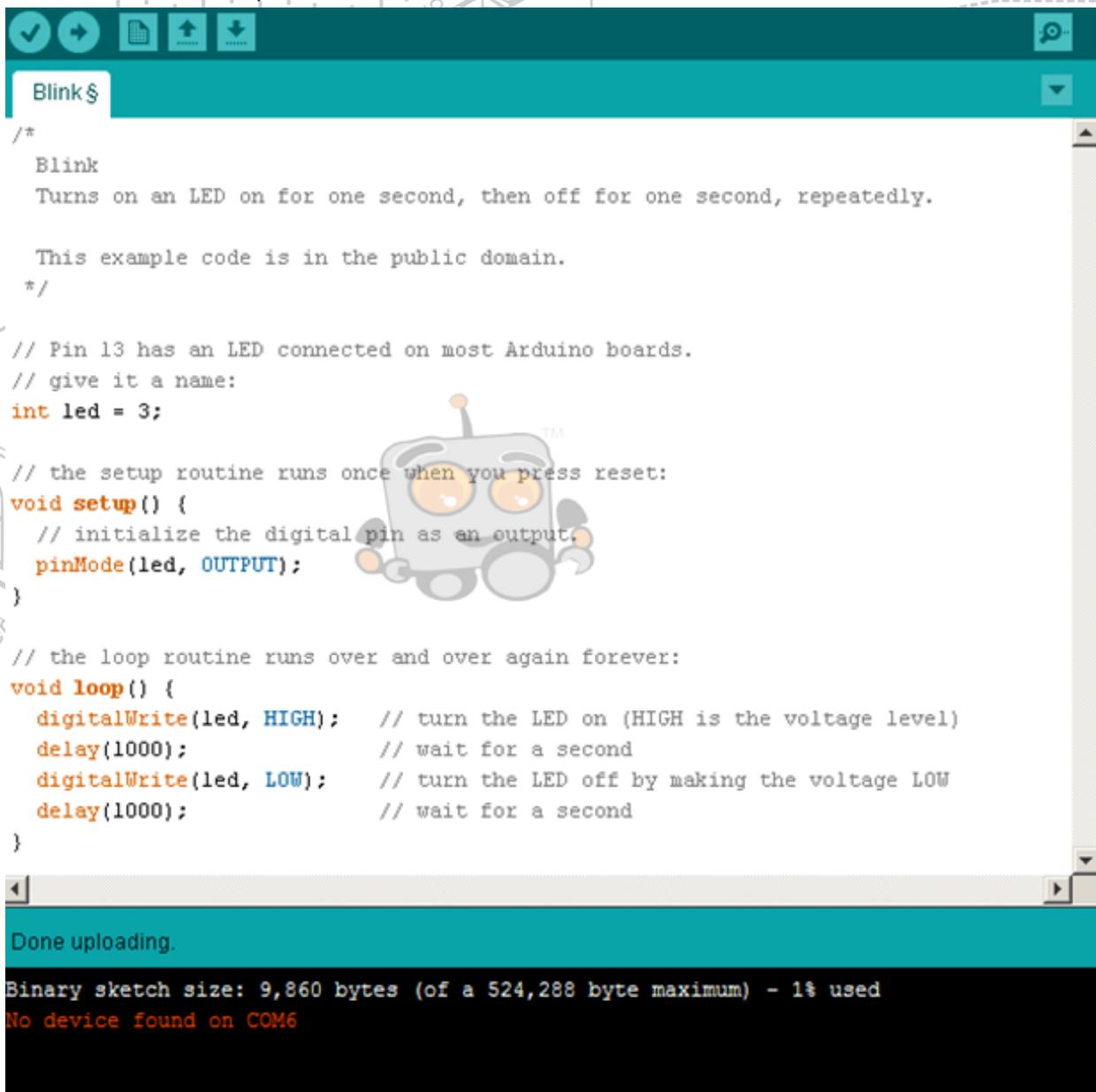
While uploading Tx and Rx blinks simultaneously and generate the message

avrdude: verification error, first mismatch at byte 0x0000

0x0d != 0x0c

Avrdude verification error; content mismatch

Avrduddone "Thank you"



```

Blink$
/*
  Blink
  Turns on an LED on for one second, then off for one second, repeatedly.

  This example code is in the public domain.
  */

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 3;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}

Done uploading.
Binary sketch size: 9,860 bytes (of a 524,288 byte maximum) - 1% used
No device found on COM6

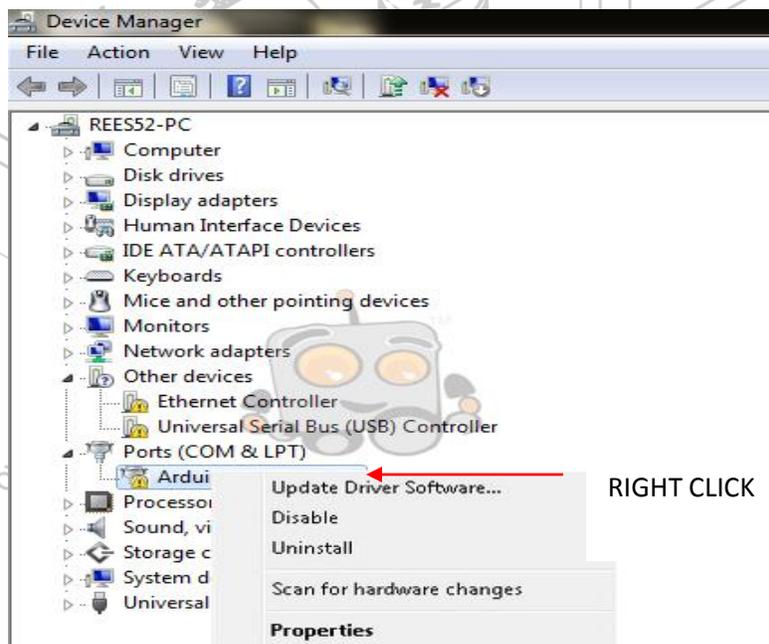
```

Suggestion

- Make sure you have the right item selected in the **Tools > Board** menu. If you have an Arduino Uno, you'll need to choose it. Also, newer Arduino Duemilanove boards come with an ATmega328, while older ones have an ATmega168. To check, read the text on the microcontroller (the larger chip) on your Arduino board.
- Check that the proper port is selected in the **Tools > Serial Port** menu (if your port doesn't appear, try restarting the IDE with the board connected to the computer). On the Mac, the serial port should be something like `/dev/tty.usbmodem621` (for the Uno or Mega 2560) or `/dev/tty.usbserial-A02f8e` (for older, FTDI-based boards). On Linux, it should be `/dev/ttyACM0` or similar (for the Uno or Mega 2560) or `/dev/ttyUSB0` or similar (for older boards).
- On Windows, it will be a COM port but you'll need to check in the Device Manager (under Ports) to see which one. If you don't seem to have a serial port for your Arduino board, see the following information about drivers.

Drivers

- On Windows 7 (particularly the 64-bit version), you might need to go into the Device Manager and update the drivers for the Uno or Mega 2560.



Just right click on the device (the board should be connected to your computer), and point Windows at the appropriate .inf file again. The .inf is in the drivers/ directory of the Arduino software (not in the FTDI USB Drivers sub-directory of it).

- If you get this error when installing the Uno or Mega 2560 drivers on Windows XP: "The system cannot find the file specified"
- On Linux, the Uno and Mega 2560 show up as devices of the form /dev/ttyACM0. These are not supported by the standard version of the RXTX library that the Arduino software uses for serial communication. The Arduino software download for Linux includes a version of the RXTX library patched to also search for these /dev/ttyACM* devices. There's also [an Ubuntu package](#) (for 11.04) which includes support for these devices. If, however, you're using the RXTX package from your distribution, you may need to symlink from /dev/ttyACM0 to /dev/ttyUSB0 (for example) so that the serial port appears in the Arduino software

Run

```
sudo usermod -a -G tty yourUserName
sudo usermod -a -G dial out yourUserName
```

Log off and log on again for the changes to take effect.

Access to the Serial Port

- On Windows, if the software is slow to start or crashes on launch, or the Tools menu is slow to open, you may need to disable Bluetooth serial ports or other networked COM ports in the Device Manager. The Arduino software scans all the serial (COM) ports on your computer when it starts and when you open the Tools menu, and these networked ports can sometimes cause large delays or crashes.
- Check that you're not running any programs that scan all serial ports, like USB Cellular Wi-Fi Dongle software (e.g. from Sprint or Verizon), PDA sync applications, Bluetooth-USB drivers (e.g. BlueSoleil), virtual daemon tools, etc.
- Make sure you don't have firewall software that blocks access to the serial port (e.g. ZoneAlarm).
- You may need to quit Processing, PD, vvvv, etc. if you're using them to read data over the USB or serial connection to the Arduino board.
- On Linux, you might try running the Arduino software as root, at least temporarily to see if fixes the upload.

Physical Connection

- First make sure your board is on (the green LED is on) and connected to the computer.
- The Arduino Uno and Mega 2560 may have trouble connecting to a Mac through a USB hub. If nothing appears in your "Tools > Serial Port" menu, try plugging the board directly to your computer and restarting the Arduino IDE.
- Disconnect digital pins 0 and 1 while uploading as they are shared with serial communication with the computer (they can be connected and used after the code has been uploaded).
- Try uploading with nothing connected to the board (apart from the USB cable, of course).
- Make sure the board isn't touching anything metallic or conductive.
- Try a different USB cable; sometimes they don't work.

Auto reset

- If you have a board that doesn't support auto-reset, be sure that you are resetting the board a couple of seconds before uploading. (The Arduino Diecimila, Duemilanove, and Nano support auto-reset as do the LilyPad, Pro, and Pro Mini with 6-pin programming headers).
- However, note that some Diecimila were accidentally burned with the wrong bootloader and may require you to physically press the reset button before uploading.
- However, on some computers, you may need to press the reset button on the board after you hit the upload button in the Arduino environment. Try different intervals of time between the two, up to 10 seconds or more.
- If you get this error: [VP 1]Device is not responding correctly. Try uploading again (i.e. reset the board and press the download button a second time).

Boot loader

- Make sure there's a bootloader burned on your Arduino board. To check, reset the board. The built-in LED (which is connected to pin 13) should blink. If it doesn't, there may not be a bootloader on your board.
- What kind of board you have. If it's a Mini, LilyPad or other board that requires extra wiring, include a photo of your circuit, if possible.
- Whether or not you were ever able to upload to the board. If so, what were you doing with the board before / when it stopped working, and what software have you recently added or removed from your computer?
- The messages displayed when you try to upload with verbose output enabled. To do this, hold down the shift key while clicking on the upload button in the toolbar.

