Rising Data Arduino Setup:

The purpose of this document is to go from zero to an operating development environment as quickly as possible. The target audience is the teachers or lab assistants who might set up the machines; not the students themselves. Successfully completing the steps outlines herein should serve as an effective test that the environment is ready for students to use.

The document is divided into three phases:

Phase 1 – Hardware Preparation: The Arduino Pro Trinket (APT) and FTDI cables are prepared to prepared so we have a mechanism to talk to the board from the computer.

Phase 2 – Software Preparation: The computer (running Windows 7) downloads and installs the Integrated Development Environment, Plugins, Drivers, and Libraries we will need for the project. This phase assumes the computer is connected to the internet and the user has administrative privileges.

Phase 3 – Integration Testing: We connect the Arduino to the computer and send some data back and forth to verify our serial connection is working.

Note: There are two version of the Arduino Pro Trinket being sold by Adafruit, the 5V and 3.3V variants. In many instances throughout the documentation we inherited from Adafruit, the 5V version is referenced or imaged. We will always be using the 3.3V version, even when the documentation might suggest otherwise.

Phase 1 – Hardware Preparation:

1) Solder Header Pins onto Arduino:

Solder male header pins onto the sides of the Arduino that face downward
Solder male header pins along the bottom of the Arduino that face upward
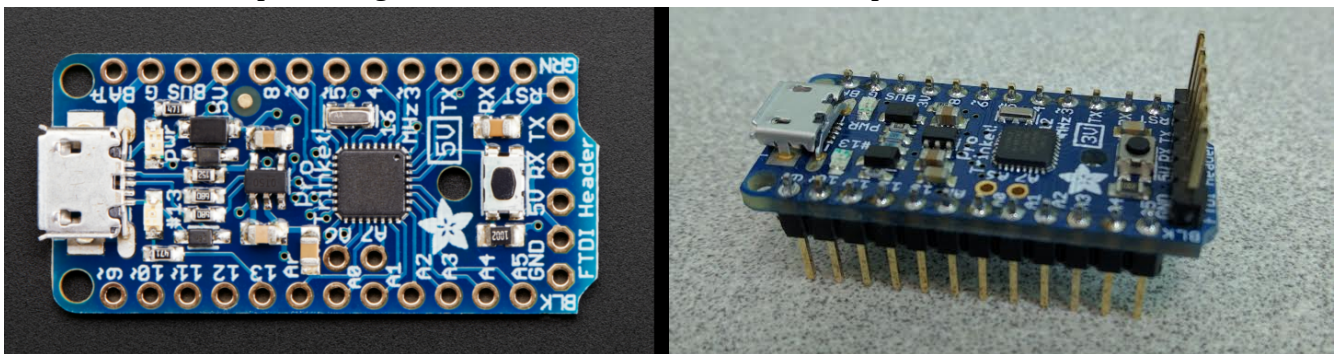
Note:
There are two non populated pins on the board (A6 and A7).

2) Pin Back Unused Lines:

We will use the black, brown, red, orange, yellow, and green wires from the FTDI cable. The unused cables can be secured back on the base using a cable tie or small zip tie to keep them out of our way.

Phase 2 – Software Preparation:

1) Arduino IDE: This is the software that lets users write and load Arduino code onto their device. As of the time of writing this document, the most recent version is 1.6.11, however any version that is 1.6.X should work properly. If you already have another version of Arduino installed, you should consider removing it to avoid confusion.
https://learn.adafruit.com/adafruit-arduino-ide-setup/arduino-1-dot-6-x-ide

2) Windows Drivers: These are the drivers that allows windows to treat the APT as a USB device.
https://learn.adafruit.com/introducing-pro-trinket/windows-setup

3) Download and install FTDI drivers: These are the drivers that allow windows to communicate with the FTDI driver. Windows has a default FTDI driver that needs to be overridden to ensure proper operation.
http://www.ftdichip.com/Support/Documents/AppNotes/AN_119_FTDI_Drivers_Installation_Guide_for_Windows7.pdf

4) Download the LSM9DS1 Library for 9DoF: These drivers allow us to communicate with the Sparkfun 9 degrees of freedom board without having to write a lot of low level code. First, download the following library:
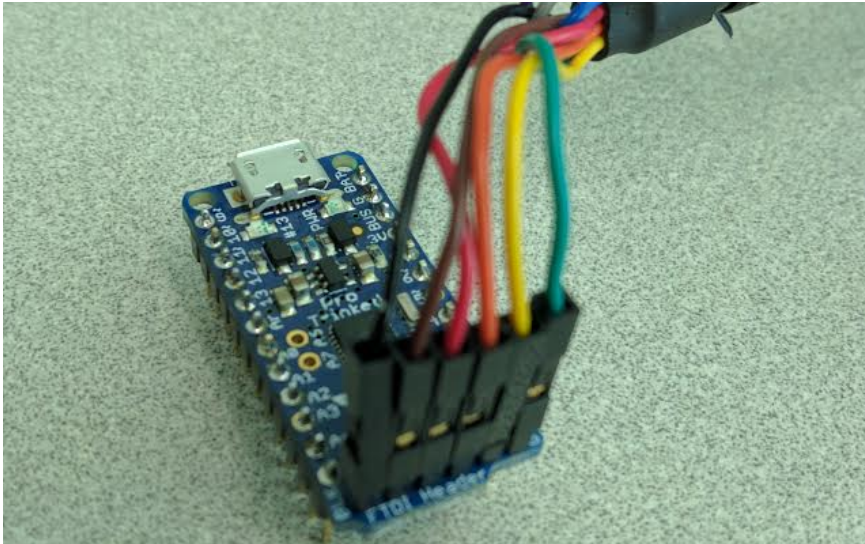https://github.com/sparkfun/SparkFun_LSM9DS1_Arduino_Library/archive/master.zip

And then follow these steps to import the library into the Arduino IDE:
https://learn.sparkfun.com/tutorials/installing-an-arduino-library#installing-a-library---windows

Phase 3 – Integration Testing:

1) FTDI Pins: Connect FTDI Pins to header pins on the APT as shown in the image below:
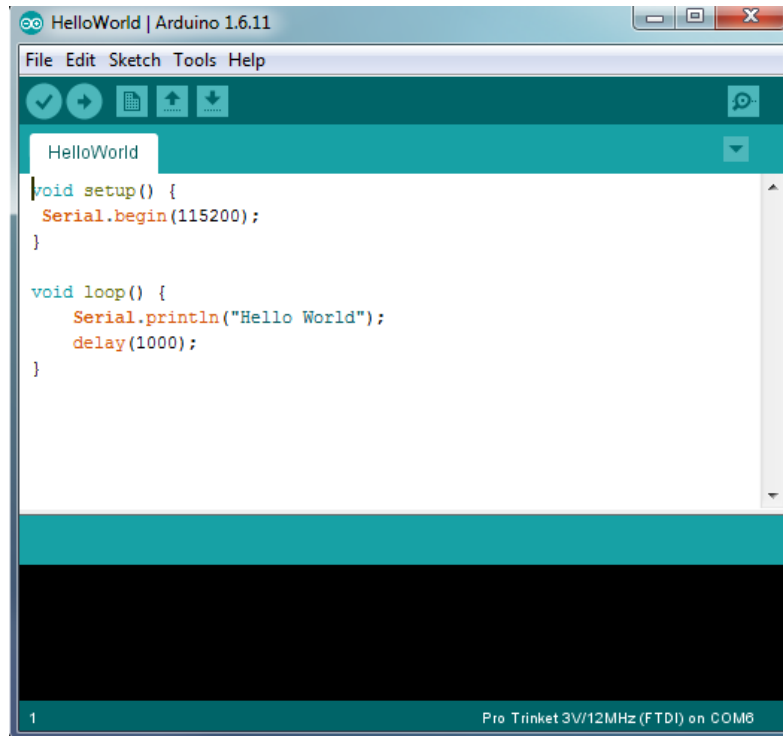

Note: Physicists might find it easy to remember the order as Black, Brown, [R,O,Y,G] when going from left to right.

2) FTDI USB: Connect the USB side of the cable to your computer's USB port.
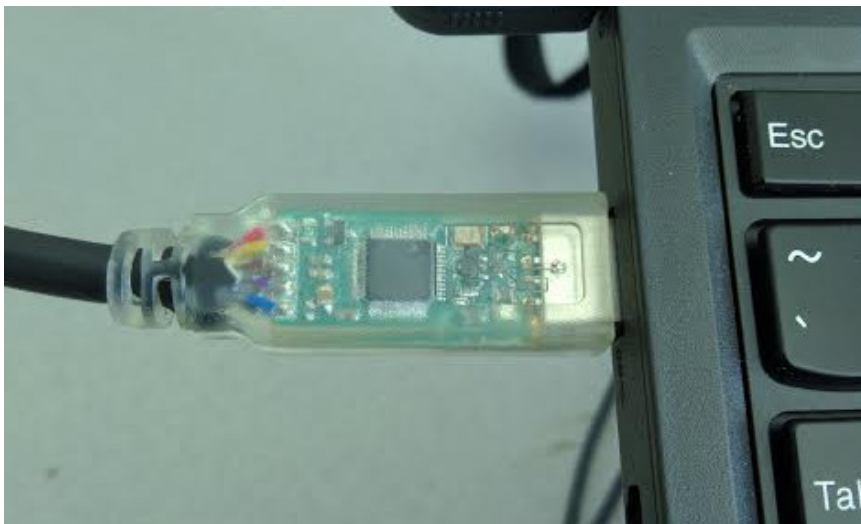
3) Open Sketch: Open the sketch found at *RisingData/src/HelloWorld/HelloWorld.ini*


This very simple program is designed to communicate a simple message over the serial connection. Compiling, loading, and running this sketch will show us that our serial communications are working correctly.

4) Select Device: Go to **Tools→Board** and select the Arduino Pro Trinket 3V/12Mhz (FTDI) option.
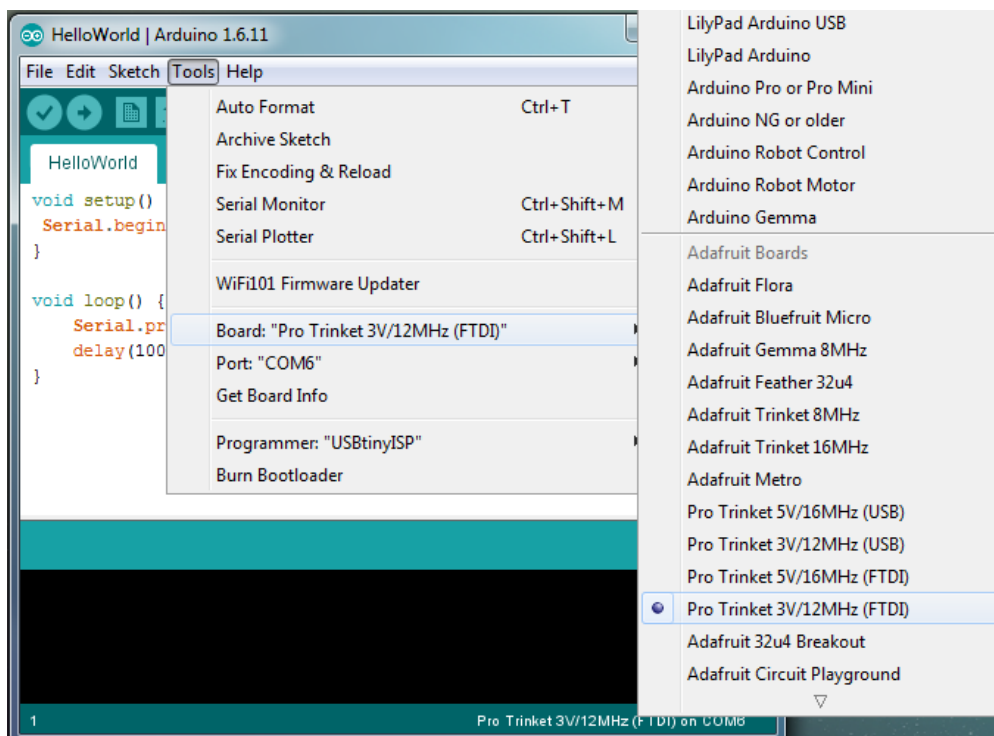
4) Select COM Port: Go to **Tools→Port** and select the listed COM port. Unless you have multiple devices connected, there will only be one in the list, through the number assigned to it is likely to be different than the one depicted below.
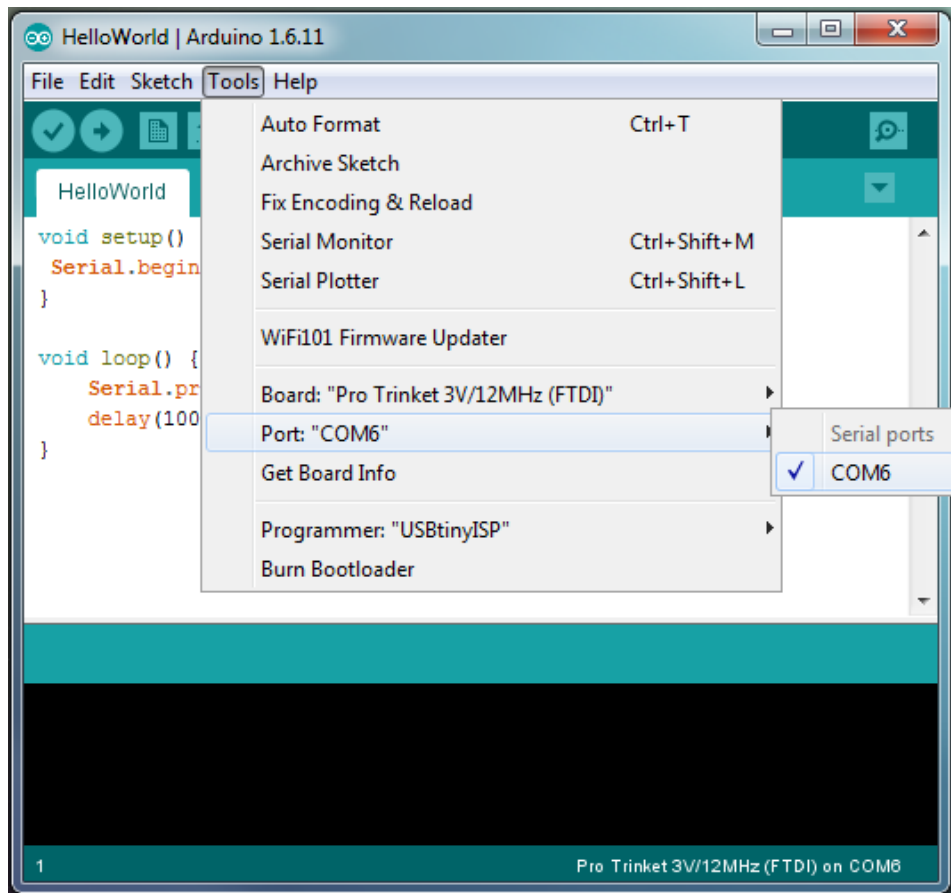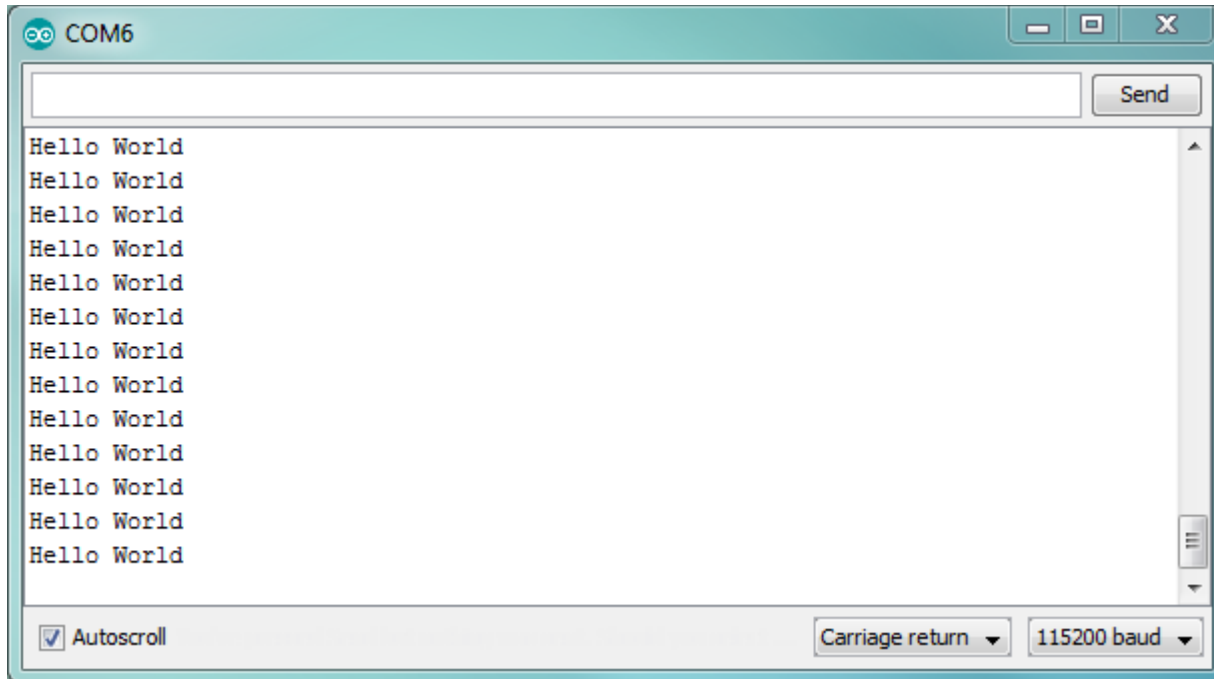

5) Compile and Upload: Compile and upload the code to your device by pressing ⊕ When finished, you should see the following at the bottom of your window:

6) Open the Serial Monitor: Open the serial monitor by pressing

Make sure you change the baud rate (bottom right corner) from the default value of 9600 to the new value of 115200. You should see this next being written to the screen at a rate of once per second. This means everything is working and we are ready to start developing with our environment.