How to Get Started in Robotics

Summary

Robotics and automated systems are the future of the manufacturing sector. Indeed, these systems are becoming more prevalent in many factories across the country, and it is the innovative companies investing in this technology that are going to remain competitive in the days to come. You may be asking yourself “As someone who will be a member of tomorrow’s workforce, how do I develop the skills necessary to become valuable in the field of robotics? Where do I even begin?”

The good news is the answer to these questions is simple: Arduino. Arduino is an opensource programmable microcontroller that allows users to experiment with different electronics components, write code, and make things that interact with the physical world. Arduino starter kits are inexpensive and can be purchased on Amazon and from other e-tailers.

In this tutorial, we will connect a servo motor to an Arduino Mega 2560. We will write the code that changes the position of the servo randomly. Lastly, we will test the device and consider the implications of our work.

Required Materials

Arduino Mega 2560

USB A to B Cable

Male-to-Male Jumper Wires

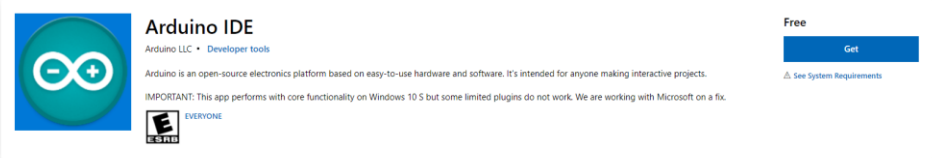
- Personal computer or laptop with: Windows 7 or newer OR Linux OR Mac with OS X 10.10 or newer.
    - Note: I'm using Windows 10 Professional in this tutorial, but any computer that meets the above requirements will work.
- Atmel Mega 2560 development board. Because the Arduino hardware is open source, lots of different manufacturers have made their own Mega 2560 boards, not just Arduino.
- One (1) USB A to B cable to connect the Mega to your computer or laptop.
- Tower Pro SG90 9-gram servo motor.
- Three (3) male-to-male jumper wires.
- Arduino Integrated Development Environment (IDE) - we'll download this.

Arduino IDE

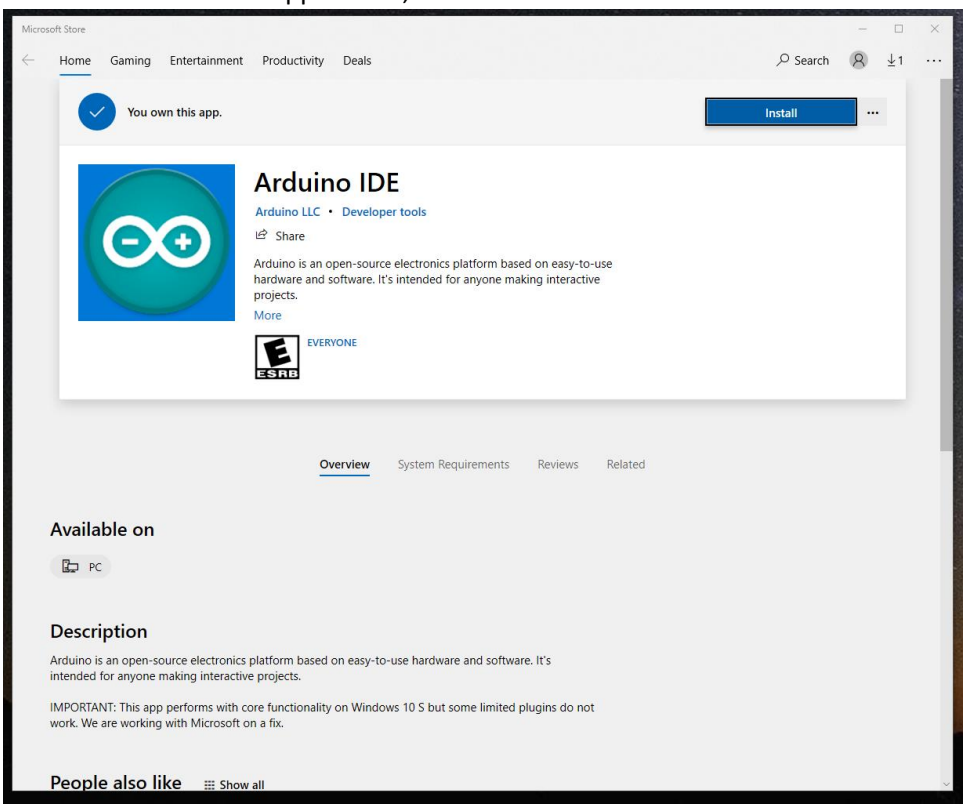
In this tutorial, we will need to write the code that controls the Mega. Once the code is written, we need to upload it to the board. To accomplish this, we need to download the Arduino Integrated Development Environment, or IDE. The Arduino IDE is completely free and available for most operating systems. The following instructions explain how to download, install and configure the IDE.

Download and Installation

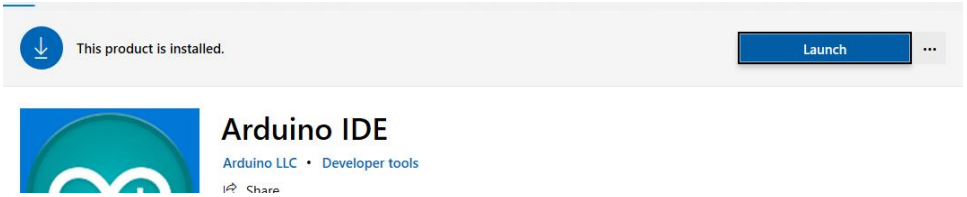
1. In a web browser, navigate to https://www.arduino.cc/en/software. On this page you will find links to download the IDE. Click on the link for your computer's operating system. If you're using Windows 10 like I am, the link will take you to the Microsoft Store in your browser.
2. In the Microsoft Store browser, click the blue "Get" button in the top-right corner of the window. This will launch the Microsoft Store application on your computer.



3. In the Microsoft Store application, click the blue “Install” button.



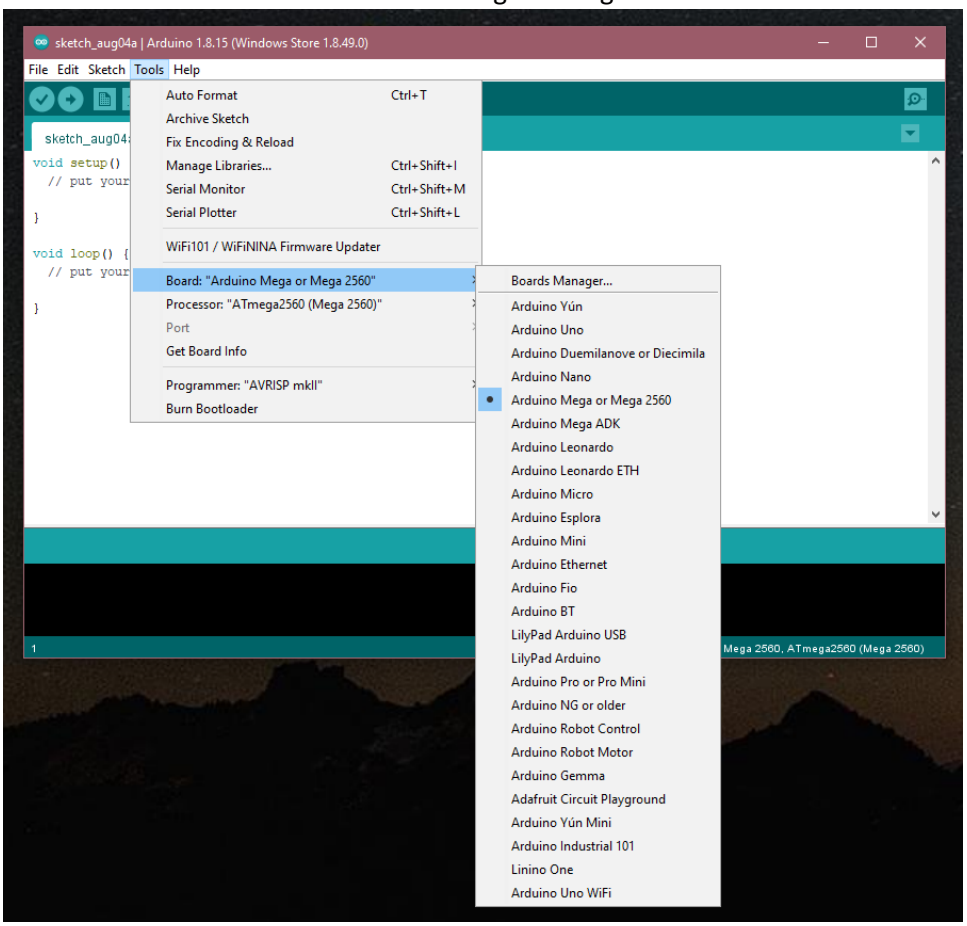
4. Once the IDE has been downloaded, the text in the blue button will change from “Install” to “Launch.” Click this button. This will launch the IDE.



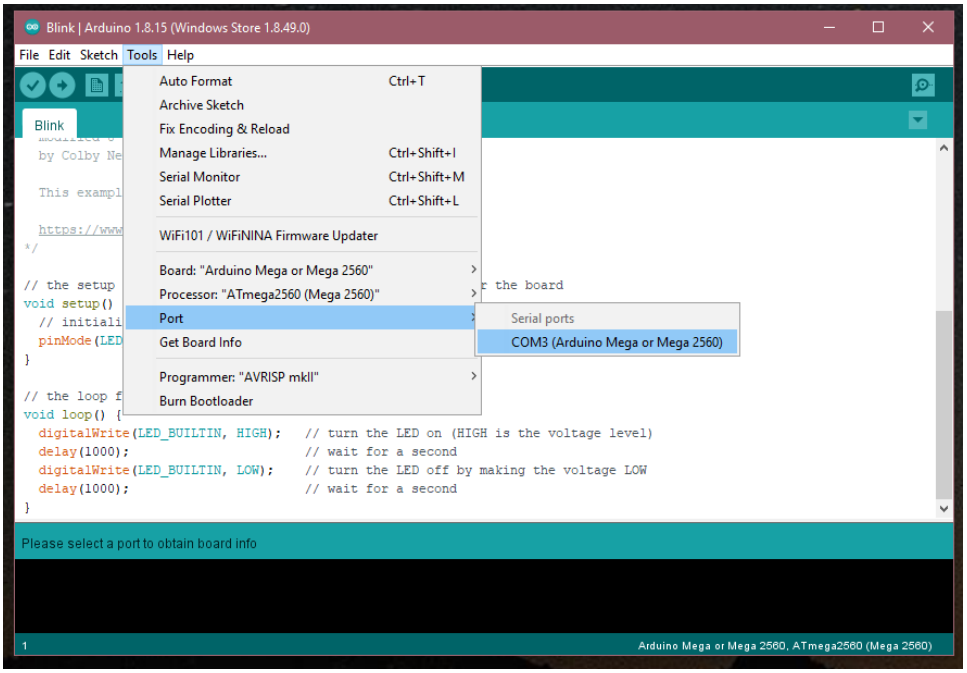
Configuring the IDE

1. Connect the Mega 2560 to your computer.
2. In the menu bar at the top of the window, click on “Tools.”

3. Click on “Board:” and select “Arduino Mega or Mega 2560.”



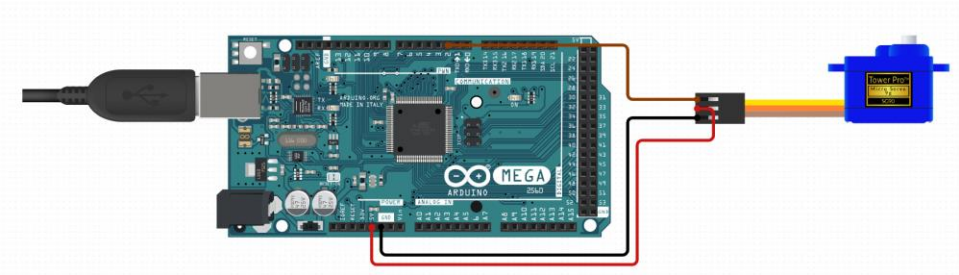
4. While still in the “Tools” menu, click on “Port” and select the COM port that your board is plugged in to.



5. That's it! You will now be able to upload the programs you write to your board.

Assembling the circuit

Arduino microcontroller development boards have many physical components and connectors, and this is especially true for the Mega. This complexity can be very overwhelming for beginners, so I have drawn a circuit diagram using the online tool at https://www.circuito.io/. Additionally, it's always a good idea to examine the datasheet for the components in the circuit. In this case, the datasheet for the Tower Pro SG90 can be found via a quick internet search; however, I have included it at the end of this document for your reference.



1. Disconnect the USB cable from the Mega. Power to the board is supplied via the USB port, and making electrical connections while power is still available can potentially damage the components.
2. Connect the orange wire to pin 2. This is the pin that will control the actual position of the servo using a technique called Pulse Width Modulation (PWM).
3. Connect the red wire to the 5V pin. This provides the +5 volts of direct current needed to power the servo.
4. Connect the brown wire to a GND pin. This provides a path to ground for the +5 volts from Step 2.
5. Reconnect the USB cable to the Mega.

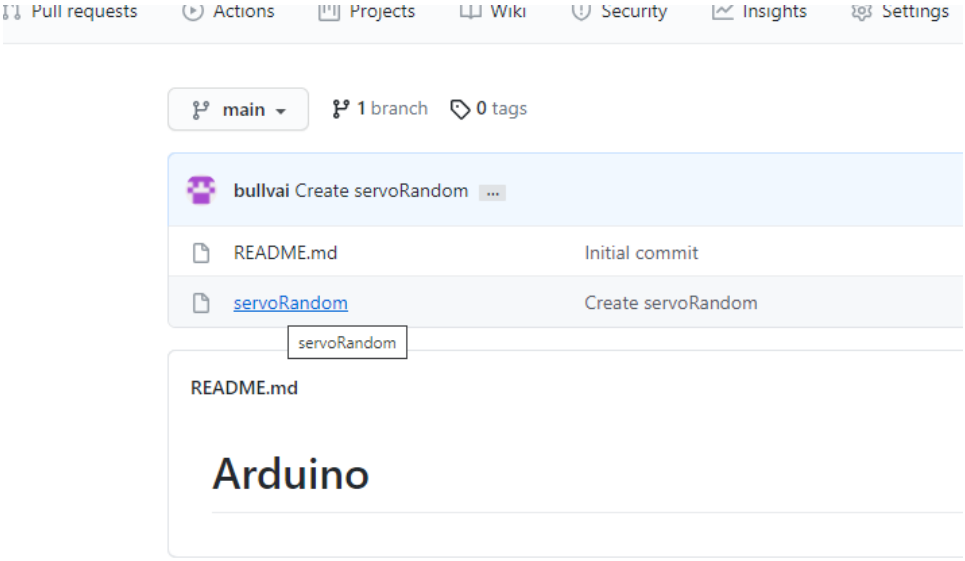
Writing the code

When you first open the IDE, you'll see two functions: setup() and loop(). The functions have comments explaining their purpose. We are going to replace this code with the code that will control our servo.

```
1 void setup() {
2   // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7   // put your main code here, to run repeatedly:
8
9 }
```

1. In a web browser, go to my GitHub repository for this project at https://github.com/bullvai/Arduino.git.

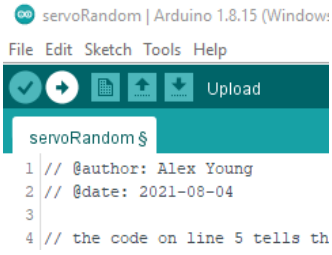
2. Click on the “servoRandom” link.



3. Copy the code and paste it into your Arduino IDE.
4. In the menu bar of the IDE, click “Save” and name your file anything you like. To understand how the code works, read the comments, which are denoted by lines starting with a double

backslash (//). When you're done, it should look like the image below.

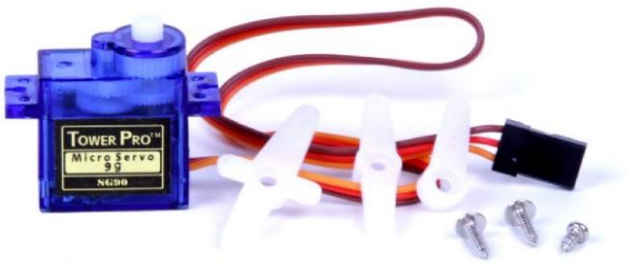
5. To upload this code to the Mega 2560, press the "upload" button in the toolbar.



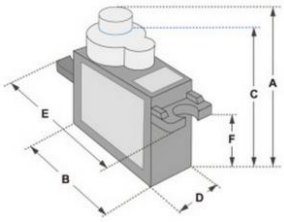
Conclusion

Congratulations! Once the code is uploaded, you will see the servo move to random positions between 0° and 180° and at a random interval between 0.5 seconds and 1.5 seconds. This is your first step into the field of embedded systems, which is a major component of robotics. From here, you can explore other projects to control more complex components or use sensors to detect information about the environment. Now, you may be wondering what you can do with the system you just created. One suggestion is to turn it into an automated cat toy by attaching a laser pointer to the servo!

SERVO MOTOR SG90 DATA SHEET



Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.



Dimensions & Specifications
A (mm) : 32
B (mm) : 23
C (mm) : 28.5
D (mm) : 12
E (mm) : 32
F (mm) : 19.5
Speed (sec) : 0.1
Torque (kg-cm) : 2.5
Weight (g) : 14.7
Voltage : 4.8 - 6

Position "0" (1.5 ms pulse) is middle, "90" (~2ms pulse) is middle, is all the way to the right, "-90" (~1ms pulse) is all the way to the left.