

HIMALAYAN MAKERS GUILD

Foundation Activity 9

Servo Motors

CONTENTS AND LEARNING OUTCOMES

Students will,

1. Compare DC and Servo Motors
2. Control the angle of a servo motor

This activity should take **~1 hour (1.5 hours recommended)** to complete:

<i>Materials and Costs per Student</i>	2
<i>Lesson</i>	2
<i>Activity Overview (5 minutes)</i>	2
<i>Introduction to Servo Motors (5 minutes)</i>	3
<i>Controlling the Servo Motor with an Arduino (10 minutes)</i>	3
<i>Arduino Programming Overview and Demonstration (15 minutes)</i>	4
<i>Program and Test the Servo Motor (20 minutes)</i>	5
<i>Debrief Discussion (5 minutes)</i>	5
<i>Challenge and Explore</i>	5
<i>Frequently Asked Questions</i>	7

MATERIALS AND COSTS PER STUDENT

This activity assumes free access to computers capable of connecting to the Arduino with a USB cable and running two programs:

1. Arduino IDE: <https://www.arduino.cc/en/Main/Software>
2. BlocklyDuino: <https://github.com/BlocklyDuino/BlocklyDuino> (or a similar visual programming tool for the Arduino e.g. ArduBlock or S4A)

Both programs can be run in a web browser, or downloaded for offline use. At least one computer and Arduino microcontroller board per three students is recommended.

Assuming one kit of parts per student:

Item	Qty.	Cost per Student ¹	Expendable ²	Supplier
9V Battery Snap	1	0.16		AliExpress
Jumper cables, MM, 10cm	5	0.10	Y	AliExpress
Breadboard 400 point	1	1.49		AliExpress
Servo SG90	1	1.78		AliExpress
Arduino UNO with cable	1	6.62		AliExpress
9V Ni-Mh 450mAh	1	5.17		AliExpress
Total Cost per Student		\$15.33 CAD		AliExpress

1. Currency is CAD, 2017-06-10. Assuming one set of parts per student.

2. Likely to be broken or lost during the activity.

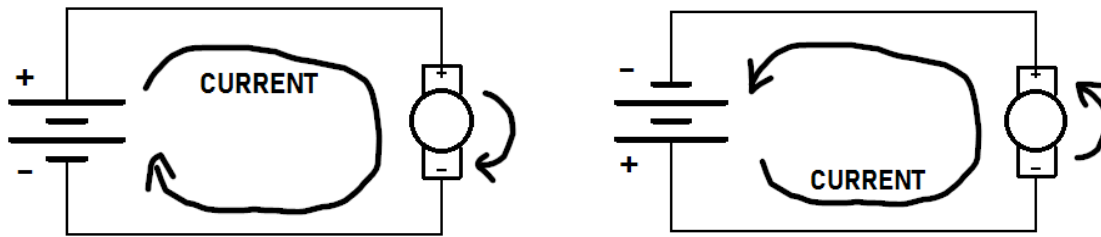
Each student should also get one printed copy of the activity handout. Check the pinout (wire colour and order) of your servo motor, as it may be different from the one used in the instructions below; adjust the lesson and handout accordingly.

LESSON

Bold text indicates directions or notes specifically for the instructor.

ACTIVITY OVERVIEW (5 MINUTES)

In the last activity we controlled the speed of a DC motor using a variable resistor called a potentiometer. By applying a voltage across the two pins of a DC motor we can make it spin continuously; if we reverse the voltage polarity, the motor will change direction.



But, what if we want a motor that does not spin continuously, but instead turns to a specific angle? This type of motor is called a servo motor.

Today, we're going to:

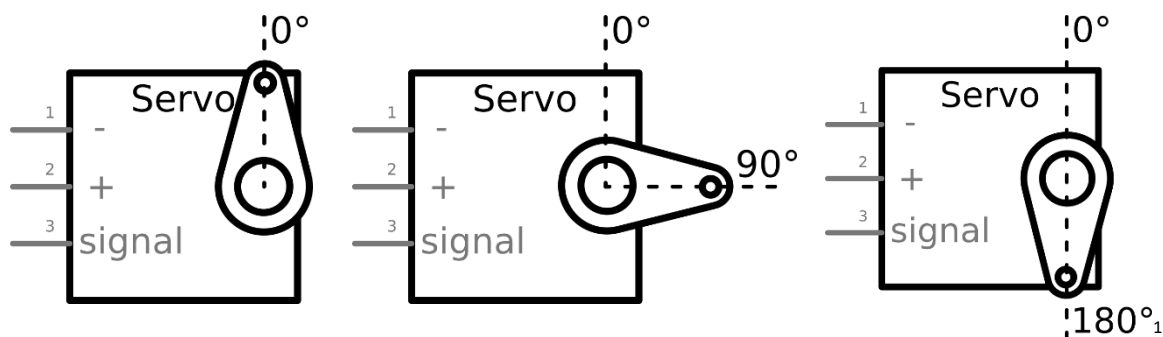
1. Learn how to control servo motors work
2. Program an Arduino UNO to make a servo motor move back and forth between two angles
3. Test the Arduino program with a servo motor

INTRODUCTION TO SERVO MOTORS (5 MINUTES)

Servo motors can be instructed to turn to a specific angle, usually from 0 to 180 degrees, by sending it a special signal. How many wires do you think a servo motor has?

A: Just like the DC motor, it needs to be connected to a potential difference (5V and GND), but it also has a 3rd wire, "signal," that we use to tell the motor what angle to turn to.

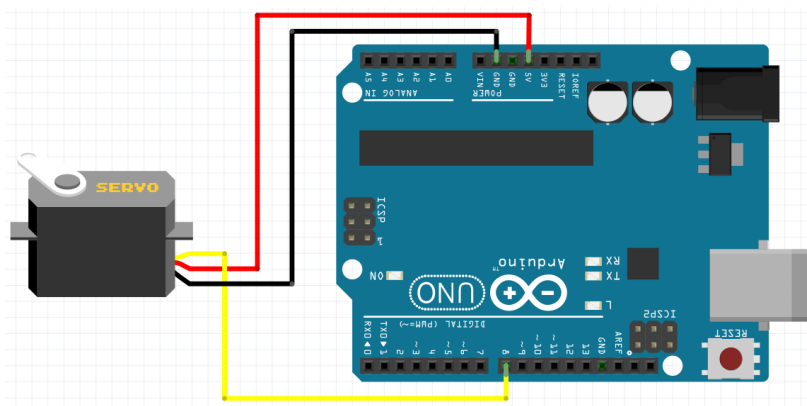
We can use an Arduino microcontroller board to send a control signal to the servomotor.



CONTROLLING THE SERVO MOTOR WITH AN ARDUINO (10 MINUTES)

Our goal is make the servo motor turn back and forth between two angles, with a pause at each angle. We will connect the servo motor to a digital output pin of the Arduino, then write a program to make the motor move back and forth. The 5V and GND pins of the Arduino will power the servo. The connection may look like this (**draw the Arduino and servo motor image below onto the board**), though we can use any of the digital output pins to control the servo (pin 8 is used here).

¹Servo and Arduino images from Fritzing.



Using BlocklyDuino, we will need two functions (blocks), one to tell the servo what angle to turn to, and another to wait so the servo motor time to move.

Servo – Degree Function

Click “**Servo**” from the left side-bar menu.
Select the top block. Set the two arguments:

- 1) **PIN#**: The output pin that is connected to the “Signal” wire of the motor
- 2) **Degree**: The degree we want the motor to turn to.




Delay Function

Click “**Control**” from the left side-bar menu.
Select the Delay block. The one argument is how many milliseconds the Arduino should wait before continuing on to the next instruction in the program.



ARDUINO PROGRAMMING OVERVIEW AND DEMONSTRATION (15 MINUTES)

On the board, review the 9 steps to programming the Arduino. As you go through the steps, see if the students are able to guess the next step.

1. Open BlocklyDuino program and the Arduino IDE program.
2. Write instructions for the Arduino microcontroller using blocks in BlocklyDuino.
3. Click the “**Arduino**” tab in BlocklyDuino, select the code, and copy it
4. Go to the Arduino IDE and delete any code already there.
5. Paste the code into the Arduino IDE.
6. Make sure the Arduino is connected to the computer using a USB cable
7. Click “**Tools**” on the top menu bar in the Arduino IDE, and make sure that “**Arduino UNO**” is selected under “**Board**”.
8. Click “**Tools**” on the top menu bar in the Arduino IDE, go to “**Port**”, and select the port that appears there after the Arduino is connected.
9. Click the arrow button  to upload the program to the Arduino

Have the students gather around a computer with the Arduino UNO and servo motor

[Example Program](#)

connected and ready to be programmed (if available, a projector can be used to show the students your screen). Follow steps 1 – 9, writing a program to make the motor move from 0 degrees to 180 degrees, waiting 2 seconds at each position. The program may look like this in BlocklyDuino, if the servo is connected to pin 8:



After the program is loaded onto the Arduino, it will work using power provided by the computer through the USB cable. However, we can also unplug the USB cable and power the Arduino from a 9V battery, since the program is loaded into the Arduino's memory.

Demonstrate the Arduino controlling the servo motor with power from the 9V battery.

PROGRAM AND TEST THE SERVO MOTOR (20 MINUTES)

Your task is to program the Arduino so that the motor:

- 1) moves from 0 to 90 degrees
- 2) 2) waits one second
- 3) 3) rotates back to 0 degrees
- 4) 4) waits another second, then repeats.

Give each student (or group of students) the materials have them start working on one of the computers.

DEBRIEF DISCUSSION (5 MINUTES)

Encourage a discussion among the students for them to share their thoughts on the activity.

Today we used an Arduino microcontroller board to make a servo motor turn between 0 and 90 degrees. We can now use servo motors when we want a motor that will turn to a specific angle, rather than just spin continuously like a DC motor. Why is this important? What applications does this have? Some possible answers include:

- Servo motors can be used to control the position of a robotic arm
- They can steer remote control boats and planes by moving the angle of control rudder, or for steering a remote control car

What worked? What didn't work? Why didn't it work? What could we do next, or how could we make the circuit better?

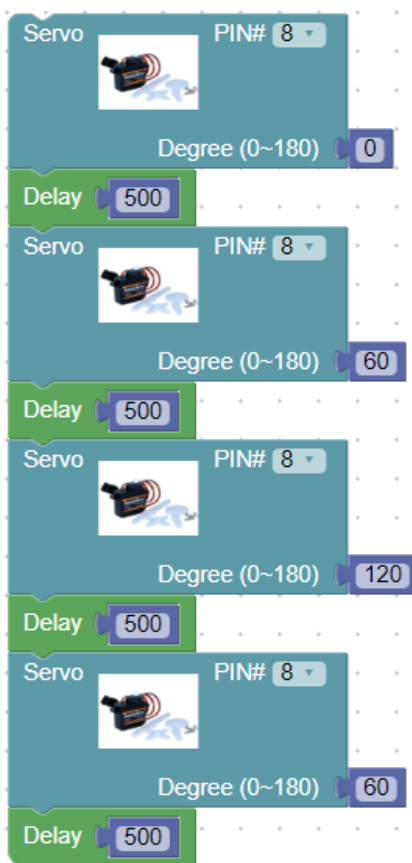
CHALLENGE AND EXPLORE

If a student completes the lesson early, evaluate their understanding by asking them to try the following:

- What would happen if we have zero delay between the angles? Try it out!
A: With no delay, the motor doesn't have time to move to the next angle, so it gets stuck in one position.
- Can you make the motor turn between three different angles, with a 1/2 second pause between each one? A:

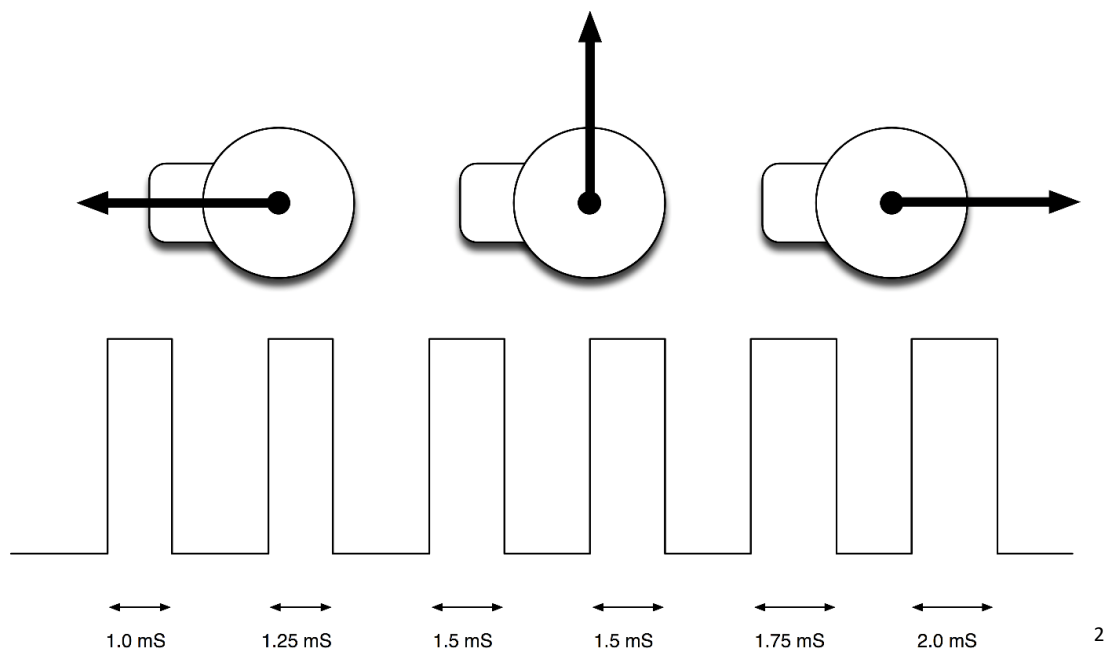


- Can you adapt that program so that the servo motor turns back and forth between the three angles, rather than jumping from the last to the first? For example, can you make it turn 0, 60, 120, 60, then repeat?



FREQUENTLY ASKED QUESTIONS

- Why is the code not compiling properly?
 - Make sure that all code was deleted from the Arduino IDE before pasting the new code from BlocklyDuino. Also, make sure that all the code was properly copied from BlocklyDuino!
- Why is the code is not uploading successfully?
 - Make sure the Arduino is connected, and that the correct COM port is selected (step 6-8 in §Programming the Arduino)
- Why isn't my servo motor not moving?
 - A: have you successfully uploaded the code using the USB cable?
 - A: does the pin number you used in your code match the pin number where the signal wire of the servo motor is connected?
 - A: is the (+) wire of the servo motor connected to 5V on the Arduino, and the (-) wire of the servo motor connected to GND on the Arduino?
- What is the signal the Arduino is sending to control the servo motor?
 - The signal that controls the servo is a periodic voltage pulse (repeating many times each second), where the pulse-width (how long it lasts) determines the angle of the motor.



²by Simon Monk from Adafruit, CC-BY-SA-3.0<https://learn.adafruit.com/assets/2313>