|  |  |  |  |
| --- | --- | --- | --- |
| **Course** | INFO-6101 [Integrated Systems and Microcontrollers] |  |  |
| **Lab#4** | **Control and Rotate a Servo Motor** |  |  |
| **Due Date** | See Due Date on FOL (Submission Folder) |  |  |
| **Submitting** | See instructions below |  |  |
| **Professor** | **Vinnie Moraes** |  |  |
| **Students’ names** |  | **Students’ #s** |  |

# How will my Lab be marked?

This Lab accounts for 1/9 of your Lab marks will be marked as follows:

**Evaluation will take place after class, after the deadline.**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **RUBRIC** | | | | | | | | | | | | |
| 1 out of 10 | 3 out of 10 | | | 3 out of 10 | | | 3 out of 10 | | | | **Final**  **Mark** | **Feedback** |
|  |  |  |  |  |  |  |  |  |  |  | **Max is 10** |
| **Attempt** | **Hardware** | | | **Software** | | | **Operation** | | | |  |
| **Report (grammar, organization, spelling).** | **Correct Components used** | **Circuit connections** | **Neatness & Clearness** | **Commands** | **Structure** | **Code** |  | **Power On** | **Function** | **Output & results** |  |
|  |  | | |  | | |  | | | |  |  |

# How should I submit my Assignment?

This is a group assignment; however, **each one of you** will submit **only one document** (Word file) and upload it to the proper submission folder.

## Please include your code (C programs **as text, inside of the Word file**) as well as snapshots (and pictures) of your circuit(s) and snapshots (and pictures) of inputs and outputs (**operation**). For more details, please check the information I already gave you about that.

Please make sure that the file you submit has a cover page showing the full name and student ID number of each group member as well as the code and name of the course, code and name of the program, the instructor’s name, date, and the lab name (e.g., “Lab xyz”). Also, please add a references page if that is the case. For more details, please check the information I already gave you about that.

Please ensure that you submit your lab in time since we deduct 15% of your potential marks for each incomplete day late (e.g., 1 minute late or 23h late will cause a deduction of 15%; 24h1min will cause a deduction of 30%, and so on).

# Experiment Description

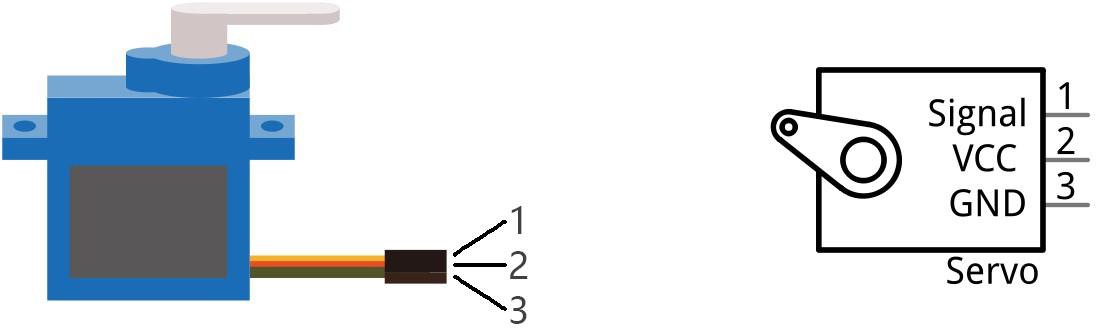
Using Tinkercad and the Arduino kit to control and rotate a servo motor through C programs.

# Hardware Connections, Components & Sketches

## Component knowledge

Servo Motor

Servo is an auto-control system, consisting of DC motor, reduction gear, sensor, and control circuit. Usually, it can rotate in the range of 180 degrees. Servo can output larger torque and is widely used in model airplane, robot and so on. It has three lines, including two for electric power line positive (2-VCC, red), negative (3GND, brown), and the signal line (1-Signal, orange).



We use 50Hz PWM signal with a duty cycle in a certain range to drive the servo. The lasting time 0.5ms- 2.5ms of PWM single cycle high level corresponds to the servo angle 0 degrees - 180 degree linearly.

Part of the corresponding values are as follows in the table below:

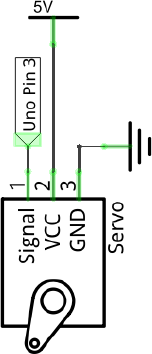
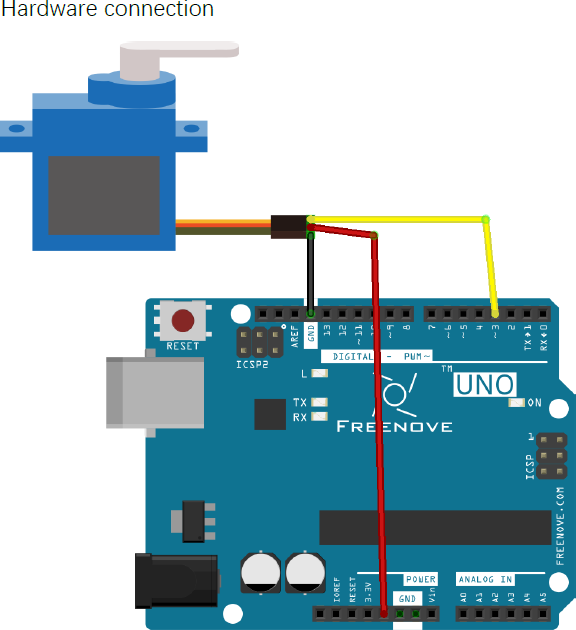
When you change the servo signal, servo will rotate to the designated position.

|  |  |
| --- | --- |
| High level time | Servo angle |
| 0.5ms | 0 degrees |
| 1ms | 45 degrees |
| 1.5ms | 90 degrees |
| 2ms | 135 degrees |
| 2.5ms | 180 degrees |

## Circuit

Use D3 of UNO to drive the servo.

**Pay attention to the color of servo lead wire: VCC (red), GND (brown), and signal line (orange). The wrong connection can cause damage to servo.**



Now, write the code to control servo, making it sweep in the motion range continuously.

Servo uses the Servo library; Servo library provides the Servo class that controls it. The Servo class must be instantiated before you use it.

#include <Servo.h>

Servo myservo; // create servo object to control a servo

int pos = 0; // variable to store the servo position

int servoPin = 3; // define the pin of servo signal line

void setup() {

myservo.attach(servoPin); // attaches the servo on servoPin to the servo object

}

void loop() {

for (pos = 0; pos <= 180; pos += 1) {

// goes from 0 degrees to 180 degrees in steps of 1 degree

myservo.write(pos); // tell servo to go to position in variable "pos" delay(15); // waits 15ms for the servo to reach the position

}

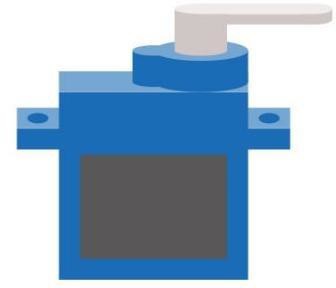
for (pos = 180; pos >= 0; pos -= 1) {

// goes from 180 degrees to 0 degrees (step 1) myservo.write(pos); // tell servo to go to position in variable "pos" delay(15); // waits 15ms for the servo to reach the position

}

}

Verify and upload the code, the servo starts to sweep continuously.



**Sketch 2.1**

Write a code to rotate the servo to go from 0 to 90 degrees (step 1 degree), then stop for 4 seconds, and finally go back from 90 to 0 degrees (step 2 degrees). Then it should wait for 7 seconds. This should be repeated continuously.