

# July 2016 Workshop

# DC & Servo Motors

Rusty Cain

7/27/2016



# Build the Project 1 L293NE or SN754410 Quad Half-H Driver

Get parts list from the web page [www.the-cains-group.net](http://www.the-cains-group.net)

Select: [July 2016 Workshop Servo & DC Motors](#)

Select: [Part\\_lists.txt](#) for a list of the parts.

Follow schematic drawing and build on your breadboard.

Download: [Project\\_1\\_HBridge\\_SN754410.ino](#)

Upload to Arduino Board

# L293NE or SN754410 Quadruple Half-H Driver

The L293NE/SN754410 is a very basic H-bridge.

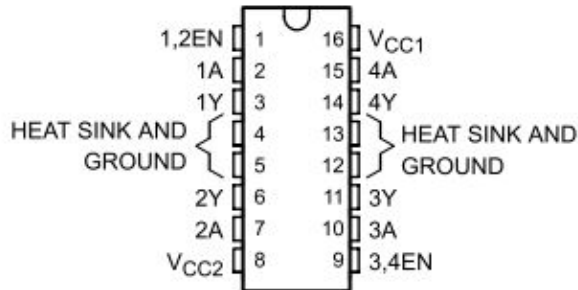
It has two bridges, one on the left side of the chip and one on the right, and can control 2 motors. It can drive up to 1 amp of current, and operate between 4.5V and 36V.

EN	1A	2A	FUNCTION
H	L	H	Turn right
H	H	L	Turn left
H	L	L	Fast motor stop
H	H	H	Fast motor stop
L	X	X	Fast motor stop

L = low, H = high, X = don't care

PIN		TYPE	DESCRIPTION
NAME	NO.		
1,2EN	1	I	Enable driver channels 1 and 2 (active high input)
<1:4>A	2, 7, 10, 15	I	Driver inputs, non-inverting
<1:4>Y	3, 6, 11, 14	O	Driver outputs
GROUND	4, 5, 12, 13	—	Device ground and heat sink pin. Connect to circuit board ground plane with multiple solid vias
V <sub>CC2</sub>	8	—	Power VCC for drivers 4.5V to 36V
3,4EN	9	I	Enable driver channels 3 and 4 (active high input)
V <sub>CC1</sub>	16	—	5V supply for internal logic translation

NE PACKAGE  
(TOP VIEW)



# L293NE or SN754410 Quadruple Half-H Driver

## How it is connected to the Arduino & Motor

Pin 1 (1,2EN) enables and disables our motor whether it is give HIGH or LOW

Pin 2 (1A) is a logic pin for our motor (input is either HIGH or LOW)

Pin 3 (1Y) is for one of the motor terminals

Pin 4-5 are for ground

Pin 6 (2Y) is for the other motor terminal

Pin 7 (2A) is a logic pin for our motor (input is either HIGH or LOW)

Pin 8 (VCC2) is the power supply for our motor.

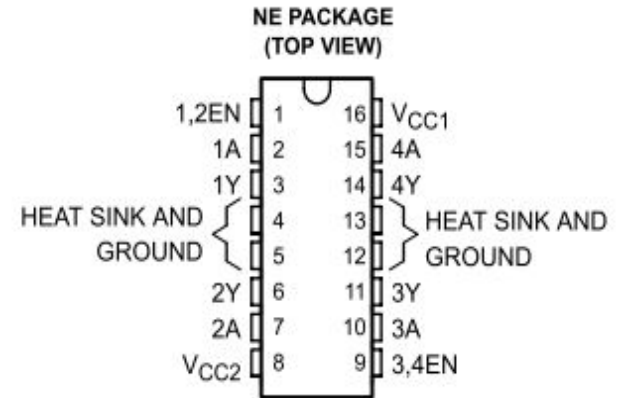
This should be given the rated voltage of your motor

Pin 9-11 are unconnected as you are only using one motor in this lab

Pin 12-13 are for ground

Pin 14-15 are unconnected

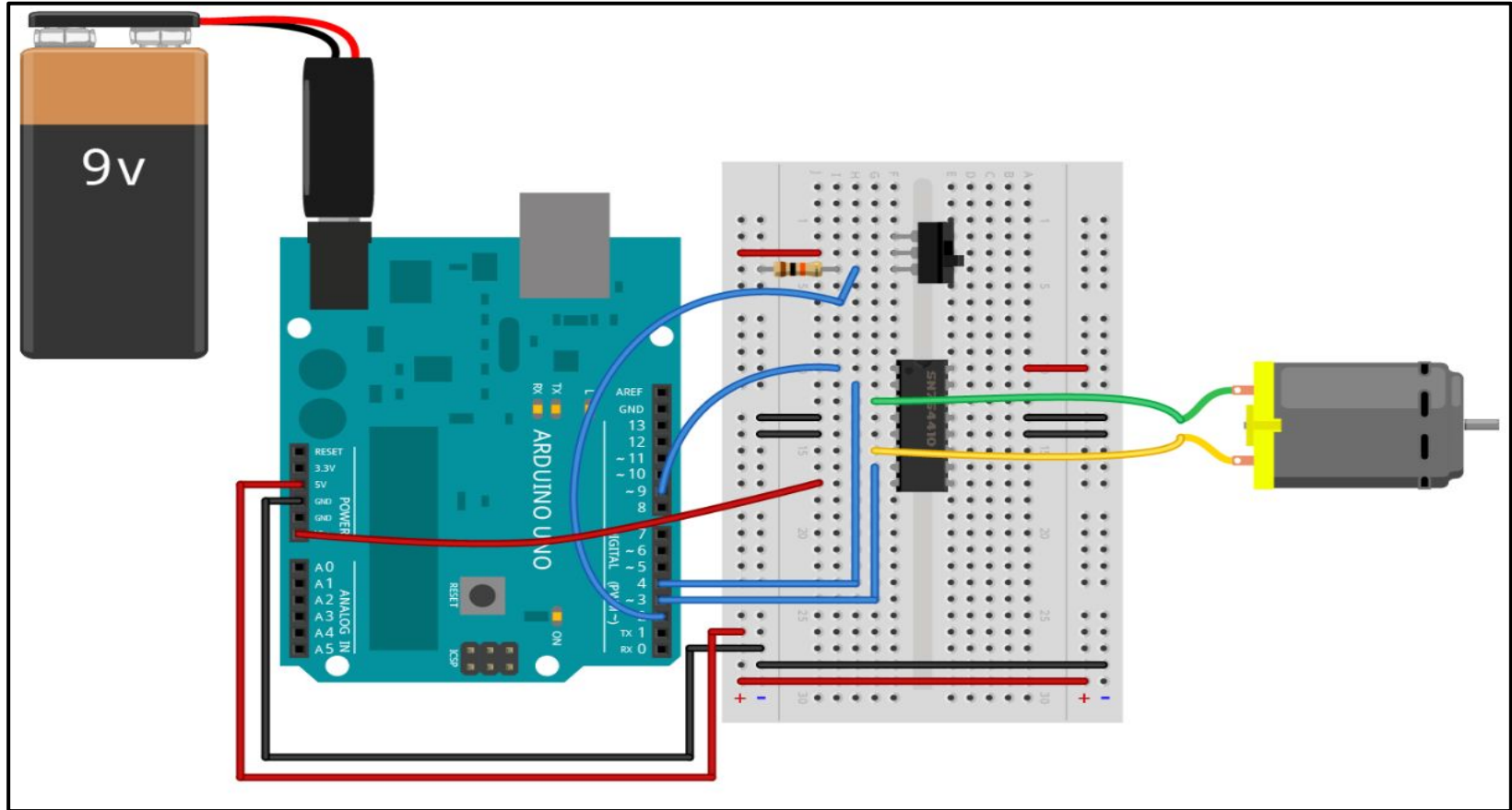
Pin 16 (VCC1) is connected to 5V

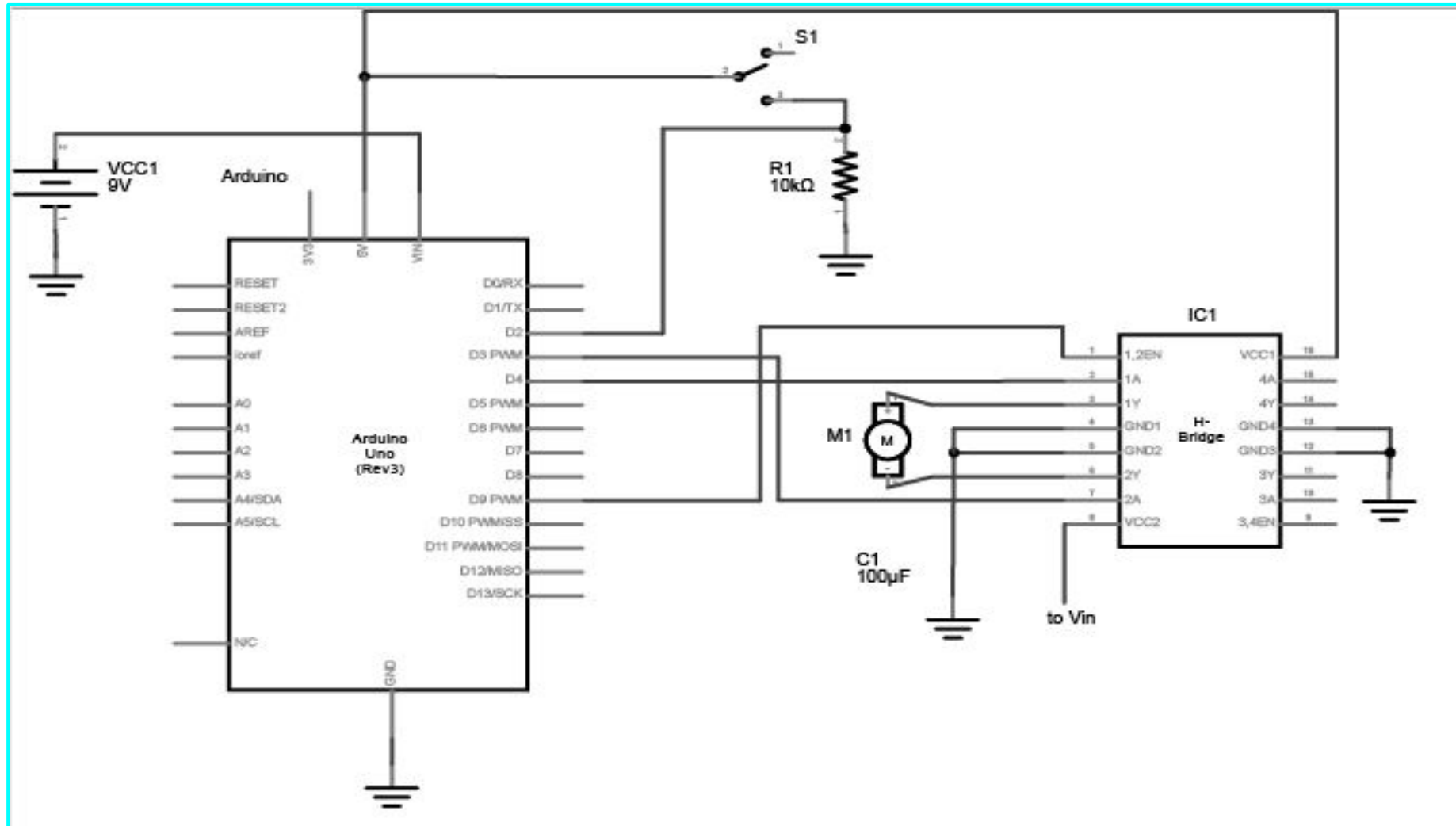


<http://itp.nyu.edu/physcomp/labs/motors-and-transistors/dc-motor-control-using-an-h-bridge/>

Let's build the project and load the code.

# L293NE or SN754410 Quadruple Half-H Driver





```

1 // Program the microcontroller to run the motor through the H-bridge.
2 // First set up constants for the switch pin, the two H-bridge pins,
3 // and the enable pin of the H-bridge. Use one of the analogWrite pins (3,5,6,9,10, or 11) for the enable pin.
4 const int switchPin = 2; // switch input
5 const int motor1Pin = 3; // H-bridge leg 1 (pin 2, 1A)
6 const int motor2Pin = 4; // H-bridge leg 2 (pin 7, 2A)
7 const int enablePin = 9; // H-bridge enable pin
8 const int GreenledPin =12; // Turn on Green LED
9 const int RedledPin =11; // Turn on Red LED
10 // In the setup(), set all the pins for the H-bridge as outputs, and the pin for the switch as an input.
11 // The set the enable pin high so the H-bridge can turn the motor on.
12 void setup() {
13 // set the switch as an input:
14   pinMode(switchPin, INPUT);
15 // set all the other pins you're using as outputs:
16   pinMode(motor1Pin, OUTPUT);
17   pinMode(motor2Pin, OUTPUT);
18   pinMode(enablePin, OUTPUT);
19   pinMode(GreenledPin, OUTPUT);
20   pinMode(RedledPin, OUTPUT);
21 // set enablePin high so that motor can turn on:
22   digitalWrite(enablePin, HIGH);
23 }
24 // In the main loop() read the switch.
25 // If it's high, turn the motor one way by taking one H-bridge pin high and the other low.
26 // If the switch is low, reverse the direction by reversing the states of the two H-bridge pins.
27 void loop() {
28 // if the switch is high, motor will turn in the Right direction:
29 if (digitalRead(switchPin) == HIGH) {
30   digitalWrite(motor1Pin, LOW); // set leg 1 of the H-bridge low
31   digitalWrite(motor2Pin, HIGH); // set leg 2 of the H-bridge high
32   digitalWrite(GreenledPin, HIGH); //Green ON Turn Motor Right
33   digitalWrite(RedledPin, LOW); // Red OFF
34 }
35 // if the switch is low, motor will turn in the Left direction:
36 else {
37   digitalWrite(motor1Pin, HIGH); // set leg 1 of the H-bridge high
38   digitalWrite(motor2Pin, LOW); // set leg 2 of the H-bridge low
39   digitalWrite(GreenledPin, LOW); //Green OFF Turn Motor Left
40   digitalWrite(RedledPin, HIGH); // Red ON
41 }
42 }

```



# Build the Project 2 SeeedStudio Motor Shield

Get parts list from the web page [www.the-cains-group.net](http://www.the-cains-group.net)

Select: [July 2016 Workshop Servo & DC Motors](#)

Place Motor Shield on Arduino Board

Connect 2 Motors from the Motor Pack for Arduino

Download: [Project\\_2\\_Motor\\_shield\\_demo\\_code.ino](#)

Upload to Arduino Board

# Project 2 - SeeedStudio Motor Shield

Two type of Motor Shields

Motor Shield V1.0

Motor Shield V2.0



Motor Shield V1.0



Motor Shield V2.0

URL: [http://www.seeedstudio.com/wiki/Motor\\_Shield](http://www.seeedstudio.com/wiki/Motor_Shield)

# Motor Shield V1.0 & V2.0

Motor Shield is used with the Arduino to drive two brushed DC motors or one 4-wire two-phase stepper motor.

Based on the H-bridge driver Chip L298N motor driver integrated circuit.

Requires a 6V to 15V power supply to power the motor and also includes an on-board

5V voltage regulator for powering the main Arduino board.

Additional plug and play Grove sensors can be connected to the 13 on-board Grove

# Hardware Overview V2

## 1. Channel 1 indicator, include 3 leds

- *EB* - channel 1 enable, high active
- *IN3* - status of OUT3
- *IN4* - status of OUT4

## 2. Channel 1 Sense - Please connect the left 2 pins together for normal usage.

Note that it's a high level application for sense the current, please refer to datasheet and schematic for more information.

## 3. OUTPUT - There're 2 channels, each channel has 2 output

- *Channel 0* - OUT1, OUT2
- *Channel 1* - OUT3, OUT4

## 4. Channel 0 Sense

## 5. Channel 0 indicator, include 3 leds

- *EB* - channel 0 enable, high active
- *IN1* - status of OUT1
- *IN2* - status of OUT2

## 6. External Power Input, range 6-15V

## 7. Reset indicator - turn red when Reset button is pressed

## 8. Reset button - pressed to reset the shield and Arduino

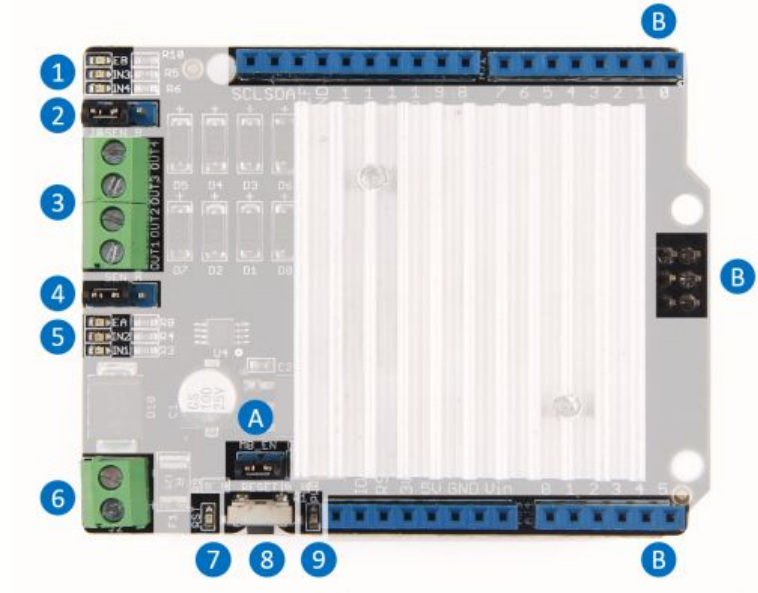
## 9. Power indicator - turn green when power in, either internal or external

### A. Power switch

Connect - Get power from Arduino

Disconnect - Get power from External sources

### B. Standard Arduino shield pin out



```

20 int pinI1=8;//define I1 interface
21 int pinI2=11;//define I2 interface
22 int speedpinA=9;//enable motor A
23 int pinI3=12;//define I3 interface
24 int pinI4=13;//define I4 interface
25 int speedpinB=10;//enable motor B
26 int speed =127;//define the speed of motor
27
28 void setup()
29 {
30     pinMode(pinI1,OUTPUT);
31     pinMode(pinI2,OUTPUT);
32     pinMode(speedpinA,OUTPUT);
33     pinMode(pinI3,OUTPUT);
34     pinMode(pinI4,OUTPUT);
35     pinMode(speedpinB,OUTPUT);
36 }
37
38 void forward()
39 {
40     analogWrite(speedpinA,speed);//input a simulation value to set the speed
41     analogWrite(speedpinB,speed);
42     digitalWrite(pinI4,HIGH);//turn DC Motor B move clockwise
43     digitalWrite(pinI3,LOW);
44     digitalWrite(pinI2,LOW);//turn DC Motor A move anticlockwise
45     digitalWrite(pinI1,HIGH);
46 }
47 void backward()//
48 {
49     analogWrite(speedpinA,speed);//input a simulation value to set the speed
50     analogWrite(speedpinB,speed);
51     digitalWrite(pinI4,LOW);//turn DC Motor B move anticlockwise
52     digitalWrite(pinI3,HIGH);
53     digitalWrite(pinI2,HIGH);//turn DC Motor A move clockwise
54     digitalWrite(pinI1,LOW);
55 }

```

```

56 void left()//
57 {
58     analogWrite(speedpinA,speed);//input a simulation value to set the speed
59     analogWrite(speedpinB,speed);
60     digitalWrite(pinI4,HIGH);//turn DC Motor B move clockwise
61     digitalWrite(pinI3,LOW);
62     digitalWrite(pinI2,HIGH);//turn DC Motor A move clockwise
63     digitalWrite(pinI1,LOW);
64 }
65 void right()//
66 {
67     analogWrite(speedpinA,speed);//input a simulation value to set the speed
68     analogWrite(speedpinB,speed);
69     digitalWrite(pinI4,LOW);//turn DC Motor B move anticlockwise
70     digitalWrite(pinI3,HIGH);
71     digitalWrite(pinI2,LOW);//turn DC Motor A move clockwise
72     digitalWrite(pinI1,HIGH);
73 }
74 void stop()//
75 {
76     digitalWrite(speedpinA,LOW);// Unenble the pin, to stop the motor.
77     digitalWrite(speedpinB,LOW);// this should be done to avoid damaging the motor.
78     delay(1000);
79 }
80 }
81
82 void loop()
83 {
84     left();
85     delay(2000);
86     stop();
87     right();
88     delay(2000);
89     stop();
90     // delay(2000);
91     forward();
92     delay(2000);
93     stop();
94     backward();
95     delay(2000);
96     stop();
97 }

```

# Build the Project 3 Servo Motor - Arduino

Get parts list from the web page [www.the-cains-group.net](http://www.the-cains-group.net)

Select: [July 2016 Workshop Servo & DC Motors](#)

Connect Servo Motor to Arduino Board using Schematic.

Download: [Project\\_3\\_Servo.ino](#)

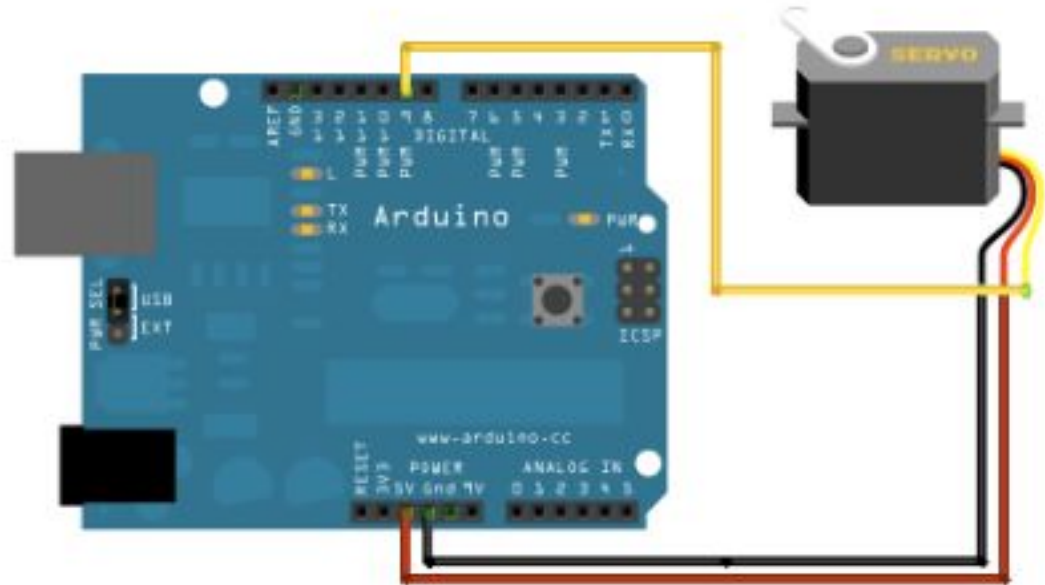
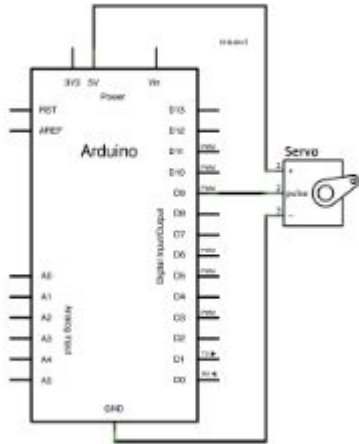
Upload to Arduino Board

\*Optional: Connect Oscilloscope to pin 9 and watch waveform.

# Project 3 Servo Motors with Arduino

Sweeps the shaft of a RC servo motor back and forth across 180 degrees.

Code Location: <http://www.arduino.cc/en/Tutorial/Sweep>



```
9 #include <Servo.h>
10 Servo myservo; // create servo object to control a servo
11 // twelve servo objects can be created on most boards
12 int pos = 0; // variable to store the servo position
13
14 void setup() {
15     myservo.attach(9); // attaches the servo on pin 9 to the servo object
16 }
17
18 void loop() {
19     for (pos = 0; pos <= 180; pos += 1) { // goes from 0 degrees to 180 degrees
20         // in steps of 1 degree
21         myservo.write(pos); // tell servo to go to position in variable 'pos'
22         delay(15); // waits 15ms for the servo to reach the position
23     }
24     for (pos = 180; pos >= 0; pos -= 1) { // goes from 180 degrees to 0 degrees
25         myservo.write(pos); // tell servo to go to position in variable 'pos'
26         delay(15); // waits 15ms for the servo to reach the position
27     }
28     /* Example:
29     for (pos = 220; pos >= 80; pos -= 20) { // goes from 180 degrees to 0 degrees
30         myservo.write(pos); // tell servo to go to position in variable 'pos'
31         delay(25); // waits 15ms for the servo to reach the position
32     }*/
33 }
```

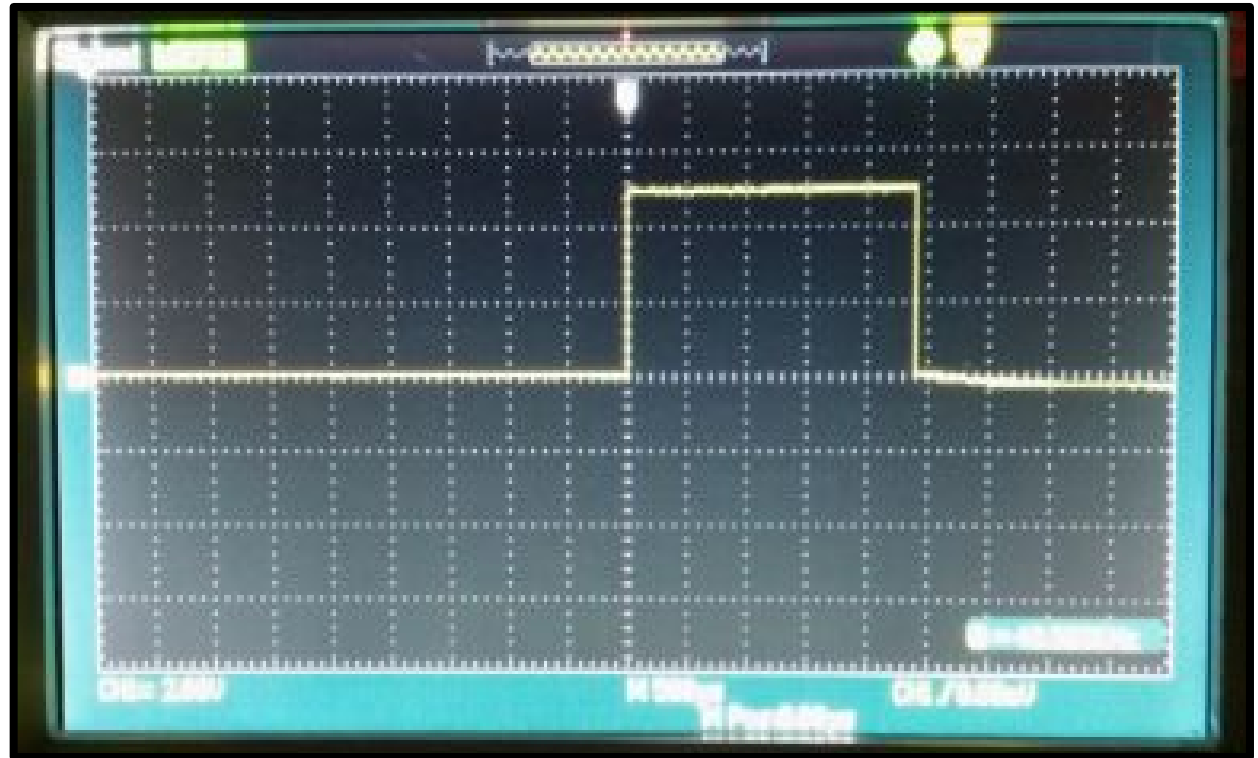


```
for (pos = 0; pos <= 180; pos += 1) { // goes from 0 degrees to 180 degrees
    myservo.write(pos); // in steps of 1 degree
    delay(15); // tell servo to go to position in variable 'pos'
                // waits 15ms for the servo to reach the position
```

PWM is used to drive motor  
pos = numbers of steps  
delay = servo to reach position

**Question:**

- If pos = 2?
- If pos =25?
- Id delay=20?
- If delay =120?



# Build the Project 4 Servo Motor - Wemos

Get parts list from the web page [www.the-cains-group.net](http://www.the-cains-group.net)

Select: [July 2016 Workshop Servo & DC Motors](#)

Connect Servo Motor to Arduino Board using Schematic.

Download: [Project\\_4\\_WeMos\\_Servo\\_Web.ino](#)

Upload to Wemos Board

\*Optional: Connect Oscilloscope to pin 9 and watch waveform.

# Project 4 -Servo Motors with Wemos D1 Mini

Presented by - Jim Merkle

