

PLCbase.org

Arduino Motor + NeoPixel Control

Beginner Guide

Arduino Motor + NeoPixel Control - Beginner Guide

1. What this program does - in one sentence

This program controls:

- DC motor (forward and backward, with smooth speed ramp),
- and 3 RGB LEDs that show motor state by colors.

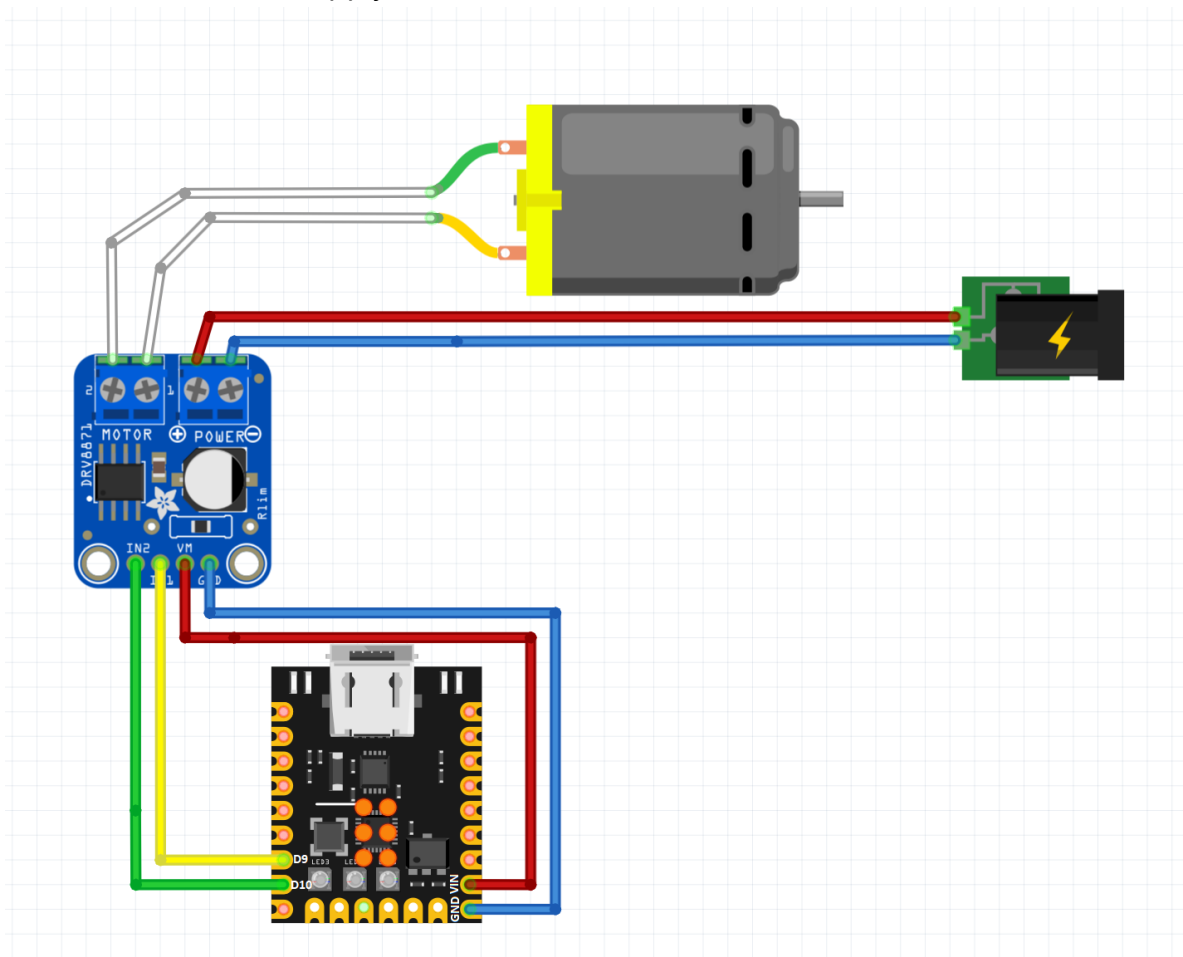
Green = ramping

Red = full speed

Blue = slowing down

2. What hardware is assumed

- Arduino board with 3 WS2812B (NeoPixel) LEDs on pin D2
- Motor driver (connected to pins D9 and D10)
- External 12V supply for Arduino and motor driver



3. Library used

```
#include <Adafruit_NeoPixel.h>
```

This library lets Arduino control addressable RGB LEDs (WS2812B).
Without it, you cannot easily set LED colors.

4. Pin definitions

```
#define LED_PIN 2  
#define NUM_LEDS 3  
#define MOTOR_IN1 9  
#define MOTOR_IN2 10
```

Meaning:

- Pin D2 → all LEDs data line
- 3 LEDs in chain
- Pin D9 and D10 → motor driver inputs

5. Creating LED object

```
Adafruit_NeoPixel leds(NUM_LEDS, LED_PIN,  
NEO_GRB + NEO_KHZ800);
```

This tells library:

- How many LEDs
- On which pin
- What LED type (WS2812B uses GRB order and 800 kHz signal)

6. setup() — runs once when Arduino starts

```
void setup() {
```

Increase PWM frequency

```
TCCR1B = TCCR1B & 0b11111000 | 0x01;
```

This changes Arduino's internal timer so PWM on D9/D10 runs at ~31 kHz.

Reason: Motor runs quieter and smoother.

Set motor pins as outputs

```
pinMode(MOTOR_IN1, OUTPUT);  
pinMode(MOTOR_IN2, OUTPUT);
```

Initialize LEDs

```
leds.begin();  
leds.setBrightness(60);  
leds.show();
```

- `begin()` prepares the LED library
- `setBrightness(60)` limits LED brightness (0..255)
- `show()` sends initial frame to LEDs

7. `loop()` — runs forever

Arduino repeats `loop()` endlessly.

Inside this loop your motor:

1. Accelerates forward
2. Runs forward full speed
3. Slows down
4. Accelerates backward
5. Runs backward full speed
6. Slows down

Then repeats forever.

8. Forward acceleration

```
digitalWrite(MOTOR_IN1, LOW);  
for (int i=130; i<200; i++)  
{  
    analogWrite(MOTOR_IN2, i);  
    leds.setPixelColor(0, leds.Color(0,255,0));  
    leds.show();  
    delay(25);  
}
```

Explanation:

- `MOTOR_IN1 = LOW` → choose forward direction
- `MOTOR_IN2` gets PWM value from 130 → 199
- That slowly increases motor speed
- LED #0 lights green during ramp-up

9. What a "for" loop is?

for (initialization; condition; increment)

In your case:

for (int i = 130; i < 200; i++)

Breakdown:

1) `int i = 130;`

Create variable i.

Set it to 130.

This runs once at loop start.

2) `i < 200;`

Condition check.

If true → run loop body.

If false → exit loop.

3) `i++`

After each loop pass, increase i by 1.

What this does physically

You are sending PWM values:

130 → 131 → 132 → ... → 199

Each step waits 25 milliseconds.

So motor power ramps smoothly upward over:

*$(200 - 130) * 25ms = 70 * 25ms = 1750ms \approx 1.75$
seconds*

That's controlled acceleration.

10. LEDs control

Why LED numbering starts at 0?

Because computers count from zero, not from one. This comes from how memory works. Inside the Arduino, your LED strip is stored like an array:

leds[0]

leds[1]

leds[2]

...

The first element in any array is index 0. So:

setPixelColor(0, ...)

means the first LED on the strip.

setPixelColor(1, ...)

means second LED.

What `leds.Color(0,255,0)` means?

RGB = Red, Green, Blue Each number is brightness from 0 to 255.

Color(R, G, B)

So:

Color(0,255,0)

means:

- Red = 0 → off
- Green = 255 → full brightness
- Blue = 0 → off

Result: pure green light.

Examples:

- `Color(0,0,0)` → off
- `Color(255,0,0)` → red
- `Color(0,0,255)` → blue
- `Color(255,255,255)` → white
- `Color(255,255,0)` → yellow
- `Color(128,0,128)` → purple

11. Forward full speed

```
analogWrite(MOTOR_IN2, 255);  
leds.setPixelColor(0, leds.Color(255,0,0));  
leds.show();  
delay(500);
```

- Motor runs full speed
- LED #0 turns red
- Waits 0.5 seconds

12. Forward slowdown

```
for (int i=200; i>= 130; i--)  
{  
    analogWrite(MOTOR_IN2, i);  
    leds.setPixelColor(0, leds.Color(0,0,255));  
    leds.show();  
    delay(25);  
}
```

- PWM decreases
- Motor slows down
- LED #0 turns blue

Then LED #0 is turned off.

13. Backward motion

Same logic, but now using MOTOR_IN1 for PWM and MOTOR_IN2 set LOW.

LED #3 is used instead of LED #0 to show backward direction.

Green → red → blue again.

14. Key concepts you must understand

- `analogWrite(pin, value)` → PWM out (0..255)
- `digitalWrite(pin, LOW/HIGH)` → binary out (0/1)
- `setPixelColor(index, r,g,b)` → sets LED color in memory
- `show()` → sends colors to LEDs
- `for(...)` → repeats gradual changes

```

#include <Adafruit_NeoPixel.h> // Library for controlling WS2812B (NeoPixel) LEDs

#define LED_PIN 2 // Pin to which all 3 LEDs (D2) are connected (from documentation on arduino board)
#define NUM_LEDS 3 // Number of LEDs in the strip (we have 3)
#define MOTOR_IN1 9 // Pin for forward moving
#define MOTOR_IN2 10 // Pin for backward moving

Adafruit_NeoPixel leds(NUM_LEDS, LED_PIN, NEO_GRB + NEO_KHZ800); //definition of the LED strip

void setup() {

    TCCR1B = TCCR1B & 0b11111000 | 0x01; // Increase the PWM frequency on D9 and D10 to ~31 kHz

    pinMode(MOTOR_IN1, OUTPUT); //Set motor pins as outputs
    pinMode(MOTOR_IN2, OUTPUT); //Set motor pins as outputs

    leds.begin(); //prepares the LED library
    leds.setBrightness(60); //limits LED brightness (0..255)
    leds.show(); // initial frame to LEDs
}

void loop() {

    // ramp up forward
    digitalWrite(MOTOR_IN1, LOW); //turn off motor

    for (int i=130; i<200; i++) //Cycle, motor speed changes from 130 to 200
    {
        analogWrite(MOTOR_IN2, i); //turn on motor
        leds.setPixelColor(0, leds.Color(0, 255, 0)); // LED #0 turns green
        leds.show(); //Update LEDs
        delay(25); //delay 25 milisecond
    }

    // forward full speed for 0.5 second
    analogWrite(MOTOR_IN2, 255); // Motor runs forward at full speed
    leds.setPixelColor(0, leds.Color(255, 0, 0)); // LED #0 turns red
    leds.show(); //Update LEDs
    delay(500); //delay 500 milisecond or 0,5 second

    // ramp down forward
    for (int i=200; i>=130; i--) //Cycle, motor speed changes from 200 to 130
    {
        analogWrite(MOTOR_IN2, i); //turn on motor
        leds.setPixelColor(0, leds.Color(0, 0, 255)); // LED #0 turns blue
        leds.show(); //Update LEDs
        delay(25); //delay 25 milisecond
    }
    leds.setPixelColor(0, 0); // LED #0 turn off
    leds.show(); //Update LEDs

    // ramp up backward
    digitalWrite(MOTOR_IN2, LOW); //turn off motor

```



```

for (int i=130; i<200; i++)    //Cycle, motor speed changes from 130 to 200
{
    analogWrite(MOTOR_IN1, i);    //turn on motor
    leds.setPixelColor(2, leds.Color(0, 255, 0));    // LED #3 turns green
    leds.show();    //Update LEDs
    delay(25);    //delay 25 milisecond
}

// backward full speed for 0.5 second
analogWrite(MOTOR_IN1, 255);    // Motor runs backward at full speed
leds.setPixelColor(2, leds.Color(255, 0, 0));    // LED #3 turns red
leds.show();    //Update LEDs
delay(500);    //delay 500 milisecond or 0,5 second

// ramp down backward
for (int i=200; i>= 130; i--)    //Cycle, motor speed changes from 200 to 130
{
    analogWrite(MOTOR_IN1, i);    //turn on motor
    leds.setPixelColor(2, leds.Color(0, 0, 255));    // LED #3 turns blue
    leds.show();    //Update LEDs
    delay(25);    //delay 25 milisecond
}
leds.setPixelColor(2, 0);    // LED #3 turn off
leds.show();    //Update LEDs
}

```

Arduino project can be loaded at download section of:



Use your phone camera to scan or use direct link:

<https://bit.ly/49LN62W>

<https://bit.ly/3NKL0MX>

Thank you for purchase

Have a great day!

info@plcbase.org

PLCbase.org