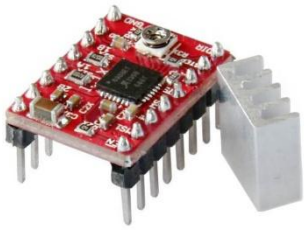
SYNACORP TECHNOLOGIES SDN. BHD. (1310487-K)
No.25 Lorong 1/SS3, Bandar Tasek Mutiara,
14120 Simpang Ampat, Penang, Malaysia.
T: +604.586.0026 F: +604.586.0026
www.synacorp.my | Email: sales@synacorp.com.my

A4988 Stepper Motor Driver Module



Introduction:

The A4988 is a micro stepping driver for controlling bipolar stepper motors which has built-in translator for easy operation. This means that we can control the stepper motor with just 2 pins from our controller, or one for controlling the rotation direction and the other for controlling the steps.

The Driver provides five different step resolutions: full-step, haft-step, quarter-step, eight-step and sixteenth-step. Also, it has a potentiometer for adjusting the current output, over-temperature thermal shutdown and crossover-current protection.

Its logic voltage is from 3 to 5.5 V and the maximum current per phase is 2A if good addition cooling is provided or 1A continuous current per phase without heat sink or cooling.

Components:

- Stepper Motor (NEMA17)
- A4988 Stepper Motor Driver Module
- Arduino Board
- Breadboard
- Jump Wires
- USB Type-B cable
- Power Supply (8-35v)
- 100µF Capacitor

Objective:

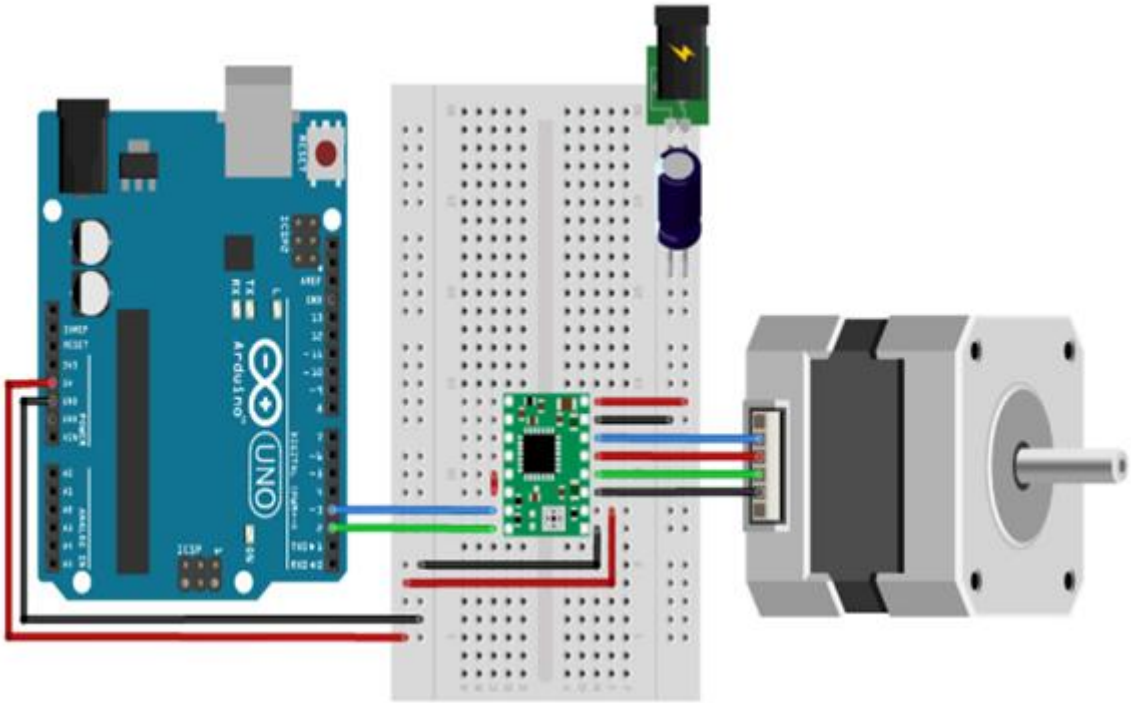
- To controlling bipolar stepper motors which has built-in translator for easy operation.

Procedure:

Step 1: Build a circuit.

The connection between the A4988 Stepper Motor Driver Module, Arduino Uno Board and the Stepper Motor (NEMA17)

A4988 Stepper Motor Driver Module	Arduino Uno
VDD	5V
GND	GND
STEP	Pin ˜3
DIR	Pin 2



A4988 Stepper Motor Driver Module	Stepper Motor (NEMA17)
VMOT	8 – 35V
GND	GND
2B	Stepper Motor
2A	Stepper Motor
1A	Stepper Motor
1B	Stepper Motor

- The motor power supply is connected to GND and VMOT (top right).
- The two coils of the stepper motor are connected to A1, A2 and B1, B2 (see below).
- The GND pin (lower right) is connected to the ground pin of the microcontroller and VDD is connected to 5V.
- The STP (step) and DIR (direction) pin are connected to digital pin 3 and 2 respectively. You can choose a different digital pin if you want, but these are the ones I used for this tutorial and the example code.
- The SLP pin is an active low input. Meaning, pulling this pin low puts the driver in sleep mode, minimizing the power consumption. RST is also an active low input. When pulled low, all STEP inputs are ignored until you pull it high. If you are not using the pin, you can connect it to the adjacent SLP/SLEEP pin to bring it high and enable the driver.
- The EN (enable) pin can be left disconnected, it is pulled low by default. When this pin is set high the driver is disabled.

Step 2: Insert the sample programming provided below by copy and paste it into Arduino IDE.

```
/*Example sketch to control a stepper motor with A4988 stepper motor driver and Arduino without a library*/
// Define stepper motor connections and steps per revolution:
#define dirPin 2
#define stepPin 3
#define stepsPerRevolution 200
void setup() {
  // Declare pins as output:
  pinMode(stepPin, OUTPUT);
  pinMode(dirPin, OUTPUT);
}
void loop() {
  // Set the spinning direction clockwise:
  digitalWrite(dirPin, HIGH);
  // Spin the stepper motor 1 revolution slowly:
  for (int i = 0; i < stepsPerRevolution; i++) {
    // These four lines result in 1 step:
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(2000);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(2000);
```

```
  }
  delay(1000);
  // Set the spinning direction counterclockwise:
  digitalWrite(dirPin, LOW);
  // Spin the stepper motor 1 revolution quickly:
  for (int i = 0; i < stepsPerRevolution; i++) {
    // These four lines result in 1 step:
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(1000);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(1000);
  }
  delay(1000);
  // Set the spinning direction clockwise:
  digitalWrite(dirPin, HIGH);
  // Spin the stepper motor 5 revolutions fast:
  for (int i = 0; i < 5 * stepsPerRevolution; i++) {
    // These four lines result in 1 step:
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(500);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(500);
  }
  delay(1000);
  // Set the spinning direction counterclockwise:
  digitalWrite(dirPin, LOW);
  //Spin the stepper motor 5 revolutions fast:
  for (int i = 0; i < 5 * stepsPerRevolution; i++) {
    // These four lines result in 1 step:
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(500);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(500);
  }
  delay(1000);
}
```