Stepper Speed Control

Overview

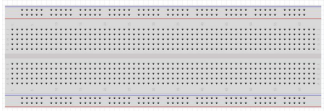
In this example, a potentiometer (or other sensor) on analog input 0 is used to control the rotational speed of a stepper motor using the Arduino Stepper Library. The stepper is controlled by with digital pins 2, 3, 4, and 5 for either unipolar or bipolar motors.

Specification

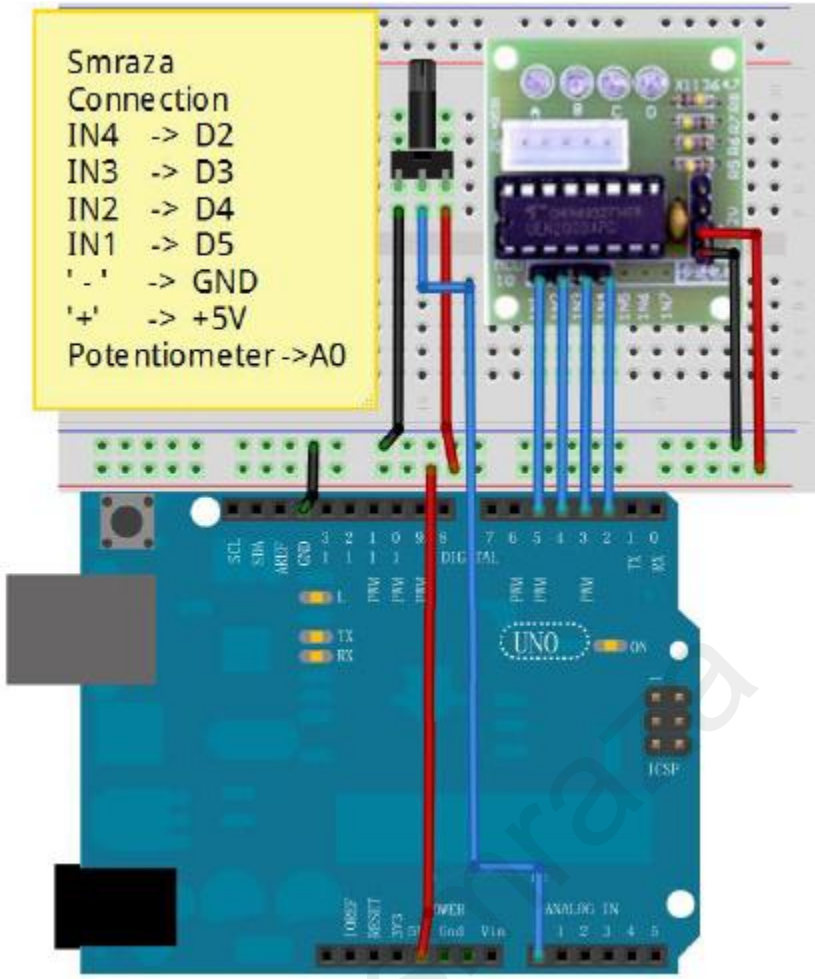
Please view "Stepper-Motor.pdf"
Path: \Public_materials\Datasheet\ Stepper-Motor.pdf

Pin definition

Hardware required

Material diagram	Material name	Number
	Step motor	1
	ULN2003 step motor driver board	1
	10KΩ potentiometer	1
	USB Cable	1
	UNO R3	1
	Breadboard	1
	Female to male Jumper	6
	Jumper wires	Several

Connection diagram



Note: The middle pin of the potentiometer is connected to the analog port 0(A0).

Sample code

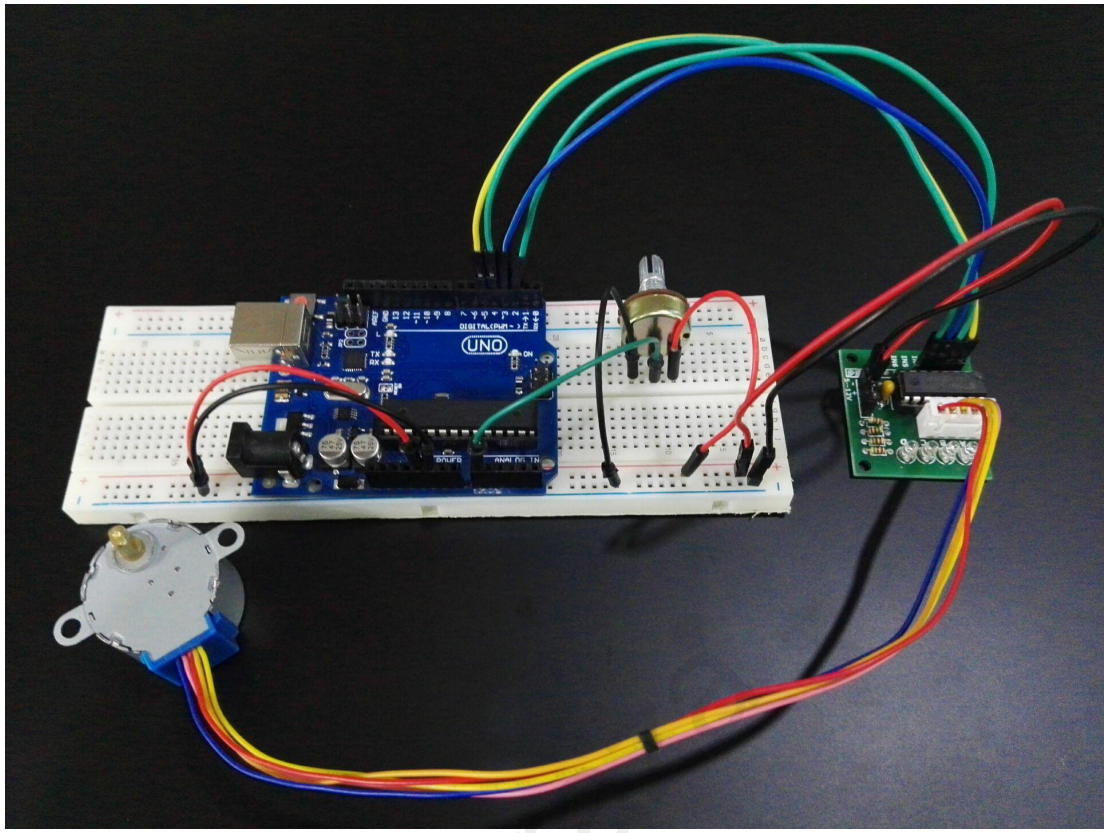
Note: sample code under the **Sample code** folder

```
#include <Stepper.h>
const int stepsPerRevolution = 200; // change this to fit the number of steps per
revolution
// for your motor

// initialize the stepper library on pins 8 through 11:

Stepper myStepper(stepsPerRevolution, 2, 3, 4, 5);
int stepCount = 0; // number of steps the motor has taken
void setup() {
    // nothing to do inside the setup
}
void loop() {
    // read the sensor value:
    int sensorReading = analogRead(A0);
    // map it to a range from 0 to 100:
    int motorSpeed = map(sensorReading, 0, 1023, 0, 100);
    // set the motor speed:
    if (motorSpeed > 0) {
        myStepper.setSpeed(motorSpeed);
        // step 1/100 of a revolution:
        myStepper.step(stepsPerRevolution / 100);
    }
}
```

Example picture



Language reference

Note：click on the following name to jump to the web page.
If you fail to open, use the Adobe reader to open this document.
Stepper myStepper = Stepper(steps, pin1, pin2, pin3, pin4)
stepper.setSpeed()
stepper.step()

Application effect

The motor will rotate in a clockwise direction. The higher the potentiometer value, the faster the motor speed. Because setSpeed() sets the delay between steps, you may notice the motor is less responsive to changes in the sensor value at low speeds.

****

* About Smraza:
* We are a leading manufacturer of electronic components for Arduino and Raspberry Pi.
* Official website: http://www.smraza.com/
* We have a professional engineering team dedicated to providing tutorials and support to help you get started.
* If you have any technical questions, please feel free to contact our support staff via email at support@smraza.com
* We truly hope you enjoy the product, for more great products please visit our
Amazon US store: http://www.amazon.com/shops/smraza
Amazon CA store: https://www.amazon.ca/shops/AMIHZKLK542FQ
Amazon UK store: http://www.amazon.co.uk/shops/AVEAJYX3AHG8Q
Amazon DE store: http://www.amazon.de/shops/AVEAJYX3AHG8Q
Amazon FR store: http://www.amazon.fr/shops/AVEAJYX3AHG8Q
Amazon IT store: http://www.amazon.it/shops/AVEAJYX3AHG8Q
Amazon ES store: https://www.amazon.es/shops/AVEAJYX3AHG8Q
****