

# Stepper motor

## Overview



This is a step motor test experiment.

## Specification



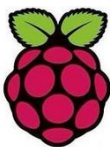
Please view "Stepper-Motor.pdf"

Path: \Public\_materials\Datasheet\ Stepper-Motor.pdf

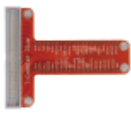



## Pin definition



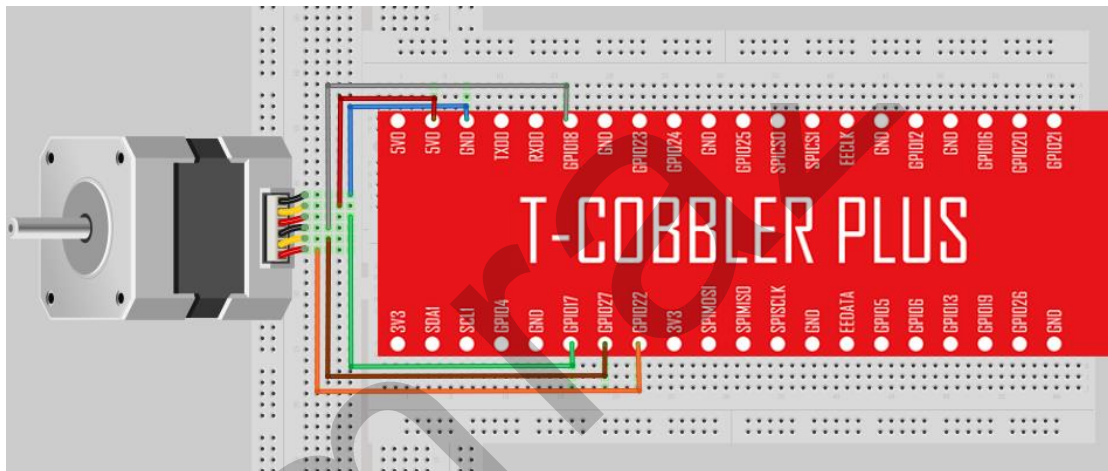
## Hardware required

Material diagram	Material name	Number
	Step motor	1
	ULN2003 step motor driver board	1
	Raspberry Pi Board	1

V1.0

	T-Cobbler Plus	1
	40P GPIO Cable	1
	Breadboard	1
	Jumper wires	Several

### Connection diagram



### Connection

RPI	Stepper Motor
GPIO17	IN1
GPIO18	IN2
GPIO27	IN3
GPIO22	IN4
"+"	5V0
"-"	GND

### Sample code

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

```
#include <wiringPi.h>
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
```

```
#define CLOCKWISE 1
#define COUNTER_CLOCKWISE 2
```

V1.0

```

#define pinA 0
#define pinB 1
#define pinC 2
#define pinD 3

void delayMS(int x);
void rotate(int* pins, int direction);

int main(void)
{
    printf( "Welcome to Smraza\n");
    printf( "Raspberry Pi Stepper motor test program\n" );
    printf( "Press Ctrl+C to exit\n" );
    int pins[4] = {pinA, pinB, pinC, pinD};
    if (-1 == wiringPiSetup()) {
        printf("Setup wiringPi failed!");
        return 1;
    }

    /* set mode to output */
    pinMode(pinA, OUTPUT);
    pinMode(pinB, OUTPUT);
    pinMode(pinC, OUTPUT);
    pinMode(pinD, OUTPUT);

    delayMS(50);    // wait for a stable status
    for (int i = 0; i < 500; i++) {
        rotate(pins, CLOCKWISE);
    }

    return 0;
}

/* Suspend execution for x milliseconds intervals.
 * @param ms Milliseconds to sleep.
 */
void delayMS(int x) {
    usleep(x * 1000);
}

/* Rotate the motor.
 * @param pins      A pointer which points to the pins number array.
 * @param direction  CLOCKWISE for clockwise rotation, COUNTER_CLOCKWISE for
counter clockwise rotation.

```

```
*/  
void rotate(int* pins, int direction) {  
    for (int i = 0; i < 4; i++) {  
        if (CLOCKWISE == direction) {  
            for (int j = 0; j < 4; j++) {  
                if (j == i) {  
                    digitalWrite(pins[3 - j], 1); // output a high level  
                } else {  
                    digitalWrite(pins[3 - j], 0); // output a low level  
                }  
            }  
        } else if (COUNTER_CLOCKWISE == direction) {  
            for (int j = 0; j < 4; j++) {  
                if (j == i) {  
                    digitalWrite(pins[j], 1); // output a high level  
                } else {  
                    digitalWrite(pins[j], 0); // output a low level  
                }  
            }  
        }  
        delayMS(4);  
    }  
}
```

**Compiling:** g++ -o stepper\_motor stepper\_motor.c -lwiringPi

**Run:** sudo ./stepper\_motor

**Tips:** Press "Ctrl+C" to exit

### Application effect

When you are running program, then you can control the relay by the screen tips.