

# LAB 14

## STEPPER MOTOR

---

### OBJECTIVES:

- To interface a stepper motor to the STM32F103.
- To write a program to control the angle and direction of stepper motor rotation by the user.

### REFERENCES:

- Mazidi & Naimi “The STM32F103 Arm Microcontroller and Embedded Systems,” Chapter 14.

### MATERIALS:

- Keil IDE or any other IDE for STM32
- Blue pill or any other STM32F10x trainer board
- ST-Link V2
- Stepper motor
- ULN2003 driver chip
- Ohmmeter

### ACTIVITY 1

Using your trainer board, build the circuit shown in Chapter 14 of the textbook. Then write and run a program to rotate it clockwise continuously. Use the following steps to figure out the stepper motor leads.

- 1) Use an ohmmeter to measure the resistance of leads. This should identify the COM leads and A through D winding leads.
- 2) The common wires are connected to the positive side of the motor's power supply. In many motors, +5V is sufficient.
- 3) The four leads of the stator winding are controlled by four pins of the microcontroller port. However, since the microcontroller might lack sufficient current to drive the stepper motor Windings and in order to protect microcontroller from EMF, we must use a driver such as the ULN2003 to energize the stator. Instead of the ULN2003, we could have used transistors as drivers, as shown in Chapter 14. However, notice that if transistors are used as drivers, we must also use diodes to take care of inductive current generated when the coil is turned off. One reason that the ULN2003 is preferable to the use of transistors as drivers is the fact that the ULN2003 has an internal diode to take care of back EMF. Change the delay to see the speed of rotation.

# LAB 14

## STEPPER MOTOR

---

### ACTIVITY 2

After making sure that Activity 1 works, write and run a program to rotate clockwise 180 degrees, stop for a second; and then go counter clockwise to its original position.

### ACTIVITY 3

After making sure that Activity 1 works, write and run the following:

1. Connect a buffered switch to PB0 and use it for clockwise or counter clockwise. Use PB0=0 for CW and PB0=1 for CCW. If you have set up STM32 serial communication, use the PC to send letters R and L for clockwise and counter clockwise, respectively.
2. After the user has made a choice, the motor will rotate accordingly.