

2015



Interfacing stepper motor with Raspberry



Author: Palani K

Introduction:

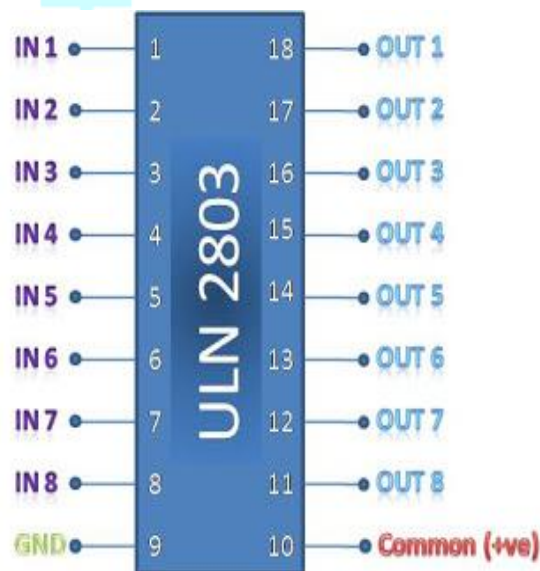
Raspberry Pi is a credit card sized computer that plugs into a computer monitor or TV, and uses standard keyboard and mouse. It's capable of doing everything you'd expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and playing games. Here we are going to do interface with stepper motor with Raspberry pi using motor driver IC ULN2803

Hardware Requirements:

1. Raspberry Pi board.
2. Stepper Motor.
3. Power supply breakout board
4. Jumper wires
5. ULN2803 IC(motor driver IC)

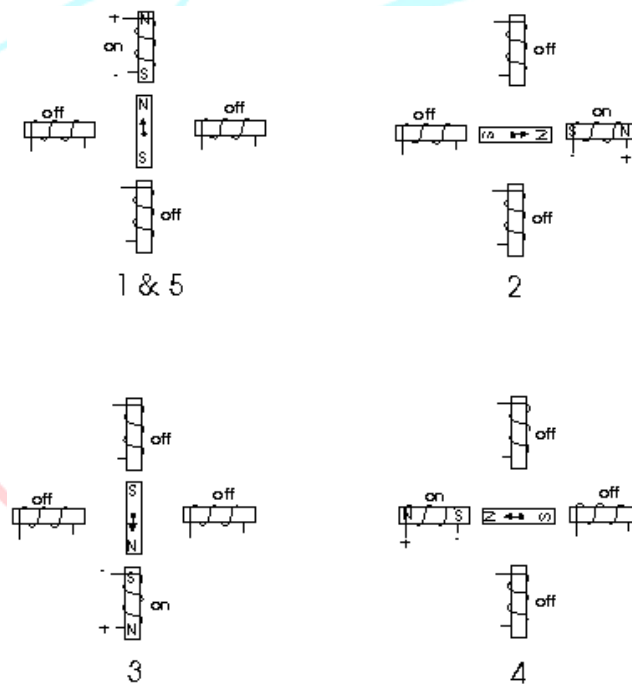
ULN2803 motor driver IC:

The ULN2803 motor driver is a high-voltage, high-current Darlington drivers comprised of eight NPN Darlington pairs. All units feature integral clamp diodes for switching inductive loads. As you can see, ULN2803 motor driver has eight input pins and eight output pins. As we are using stepper motor which has four pins, we need only four input and output pins of ULN2803.



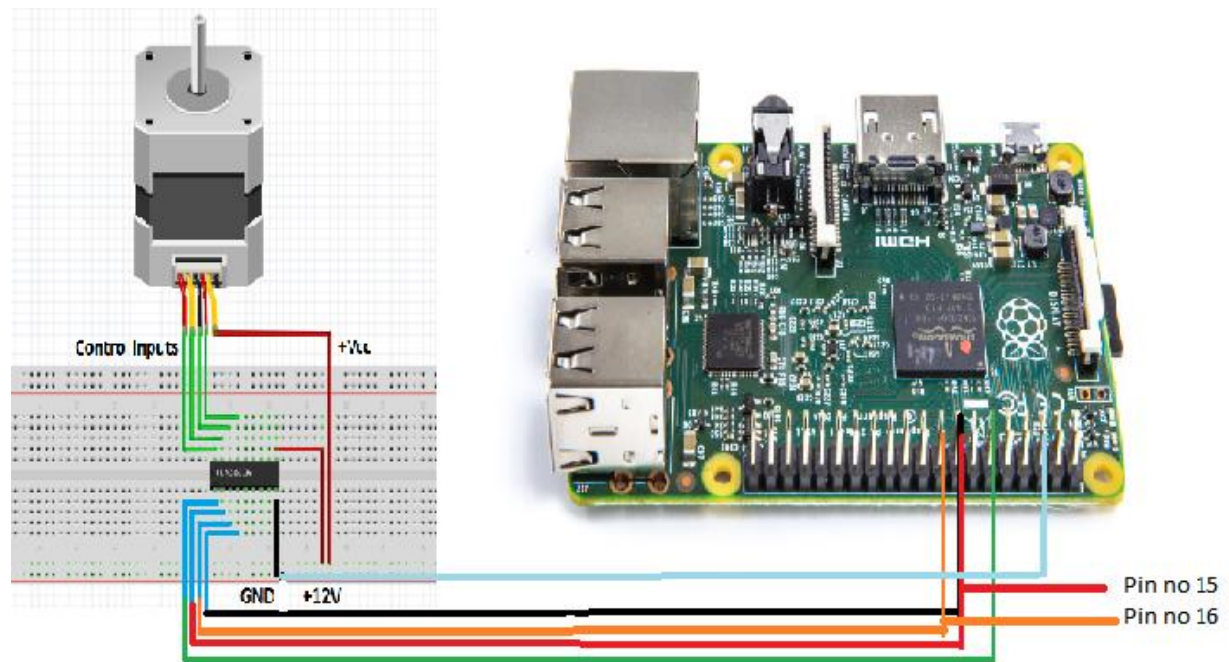
Stepper motor:

Stepper motors consist of a permanent magnetic rotating shaft, called the rotor, and electromagnets on the stationary portion that surrounds the motor, called the stator. Figure 17 illustrates one complete rotation of a stepper motor. At position 17, we can see that the rotor is beginning at the upper electromagnet, which is currently active (has voltage applied to it). To move the rotor clockwise (CW), the upper electromagnet is deactivated and the right electromagnet is activated, causing the rotor to move 90 degrees CW, aligning itself with the active magnet. This process is repeated in the same manner at the south and west electromagnets until we once again reach the starting position.



TECHNETRONICS

Circuit diagram:



Connection explanation:

- ULN2803 1st pin is connected to 11th pin of Raspberry Pi.
- ULN2803 2nd pin is connected to 16th pin of Raspberry Pi.
- ULN2803 3rd pin is connected to 18th pin of Raspberry Pi.
- ULN2803 4th pin is connected to 15th pin of Raspberry Pi.
- ULN2803 18,17,16,15 pins are connected to Stepper motor
- Stepper motor is powered from 12V power supply breakout board.

Coding in Raspberry Pi:

Step 1: Raspberry pi home screen like this. Open LXTERMINAL which is available on left corner of the screen.

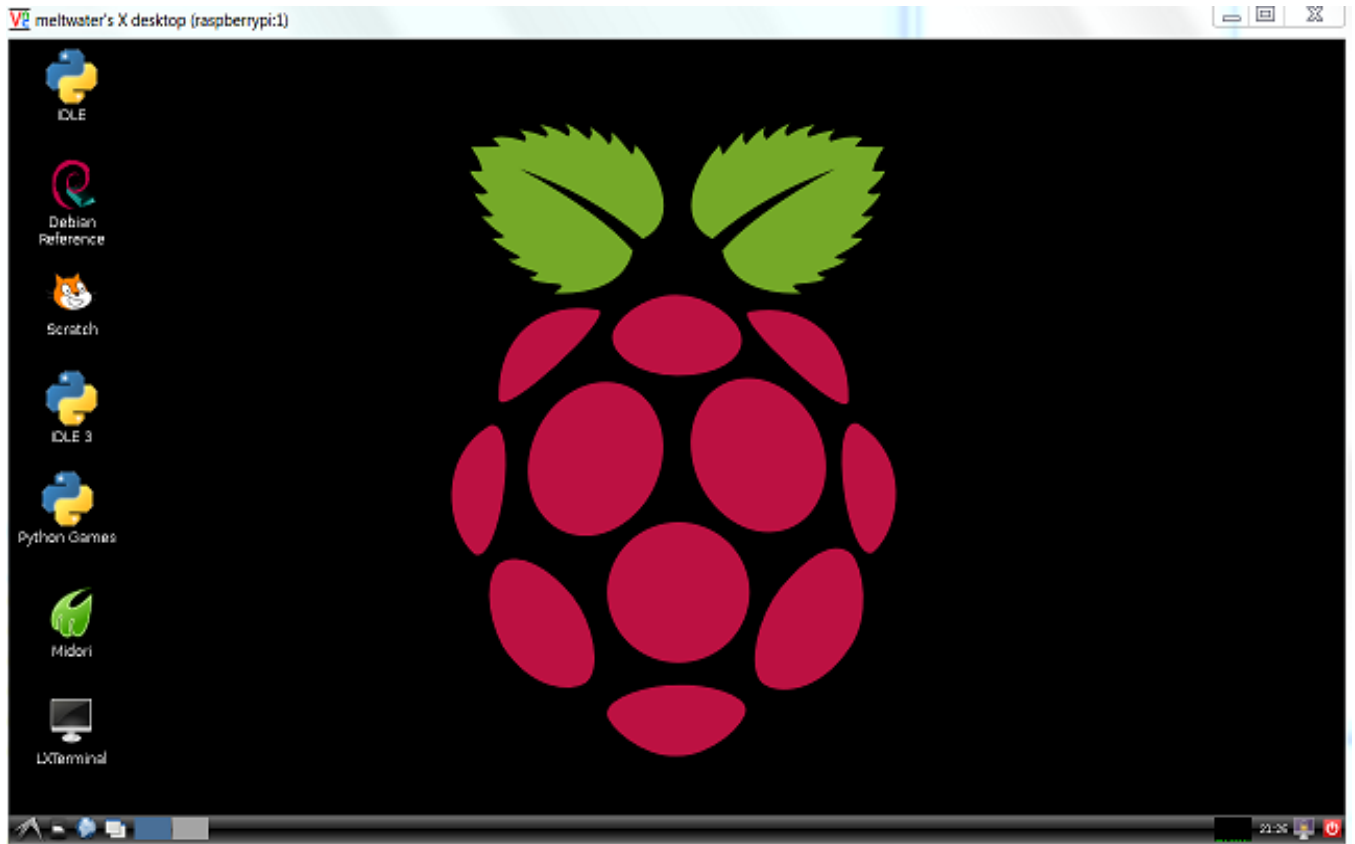


Figure 1

TENET
TECHNETRONICS

Step 2: Create a new file by using **sudo nano filename.py** command.

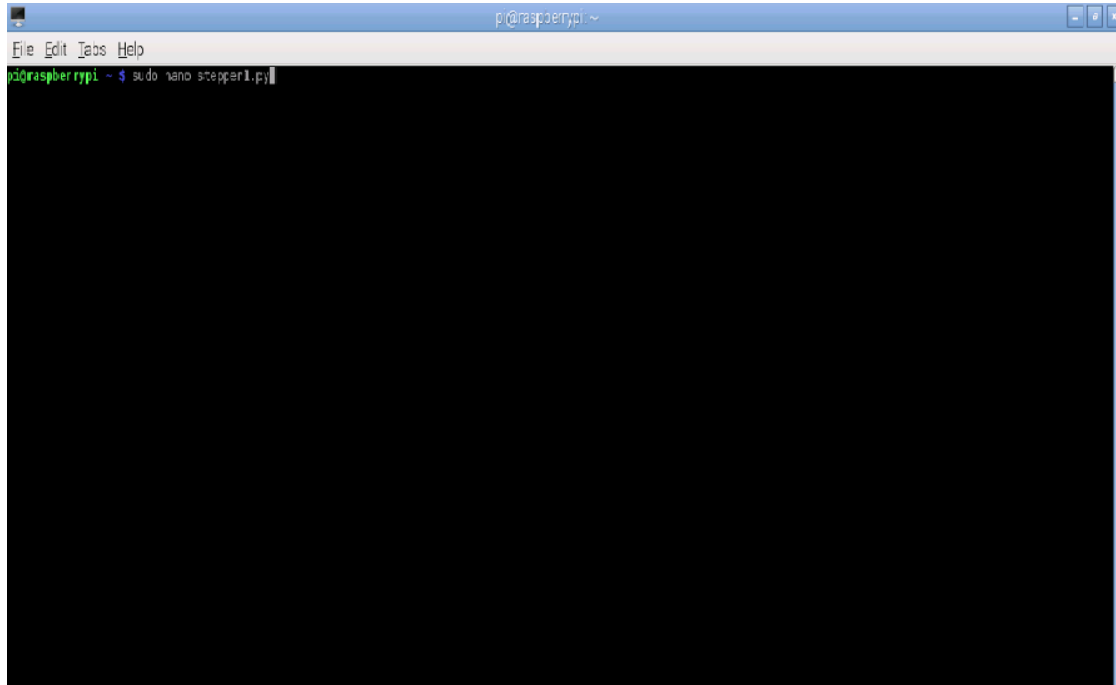


Figure2

Step 3: Hitting ENTER key will take you in the new window where you can type your code.

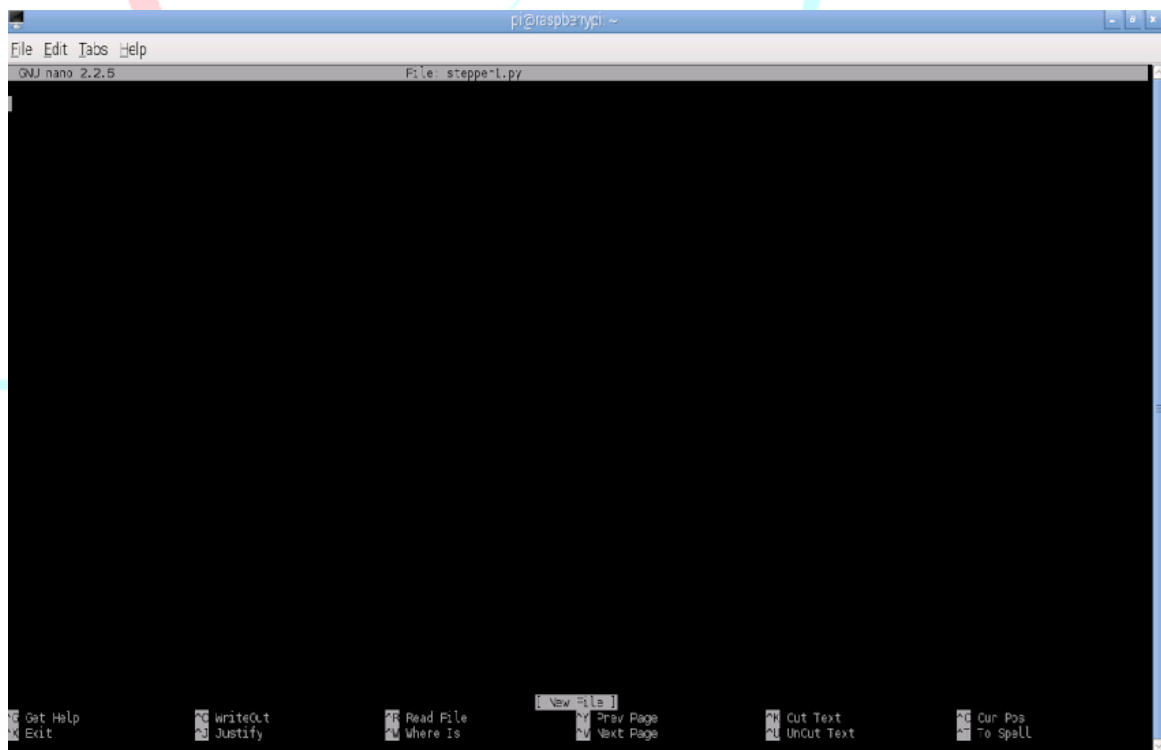
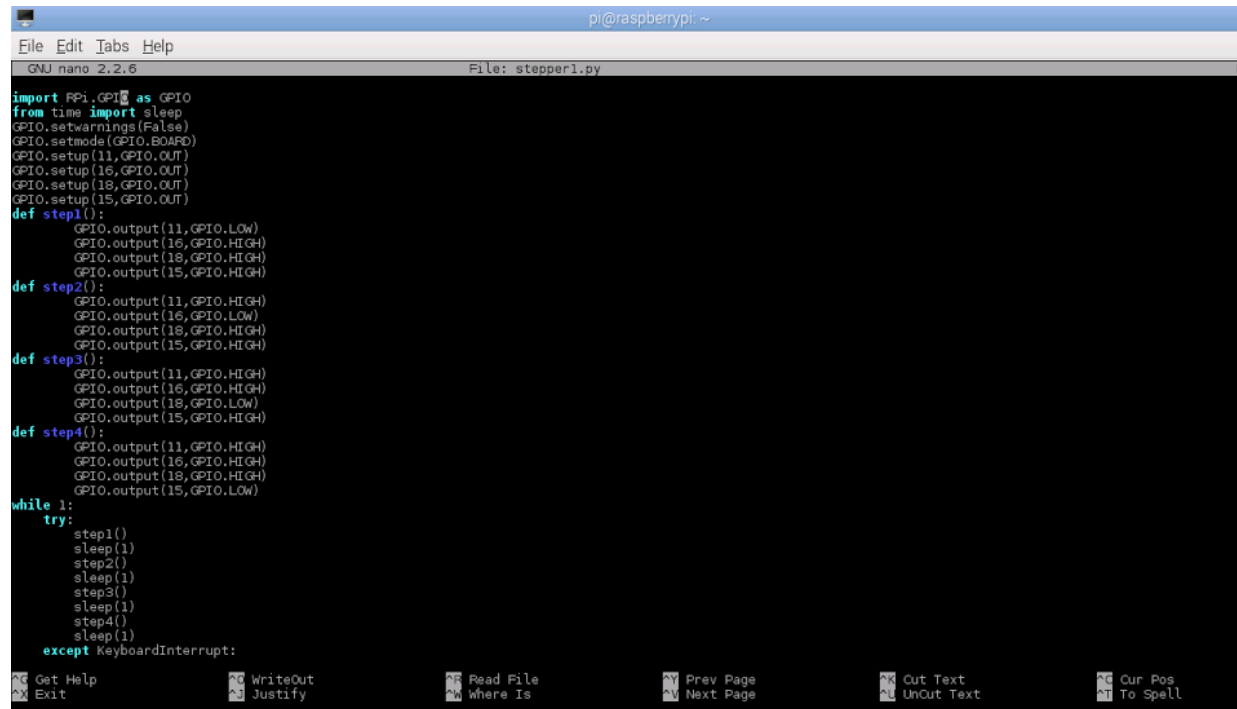


Figure 3

Step 4: Type your code in the prescribed window.



```
File Edit Tabs Help
GNU nano 2.2.6 File: stepper1.py

import RPi.GPIO as GPIO
from time import sleep
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BOARD)
GPIO.setup(11, GPIO.OUT)
GPIO.setup(16, GPIO.OUT)
GPIO.setup(18, GPIO.OUT)
GPIO.setup(15, GPIO.OUT)

def step1():
    GPIO.output(11, GPIO.LOW)
    GPIO.output(16, GPIO.HIGH)
    GPIO.output(18, GPIO.HIGH)
    GPIO.output(15, GPIO.HIGH)

def step2():
    GPIO.output(11, GPIO.HIGH)
    GPIO.output(16, GPIO.LOW)
    GPIO.output(18, GPIO.HIGH)
    GPIO.output(15, GPIO.HIGH)

def step3():
    GPIO.output(11, GPIO.HIGH)
    GPIO.output(16, GPIO.HIGH)
    GPIO.output(18, GPIO.LOW)
    GPIO.output(15, GPIO.HIGH)

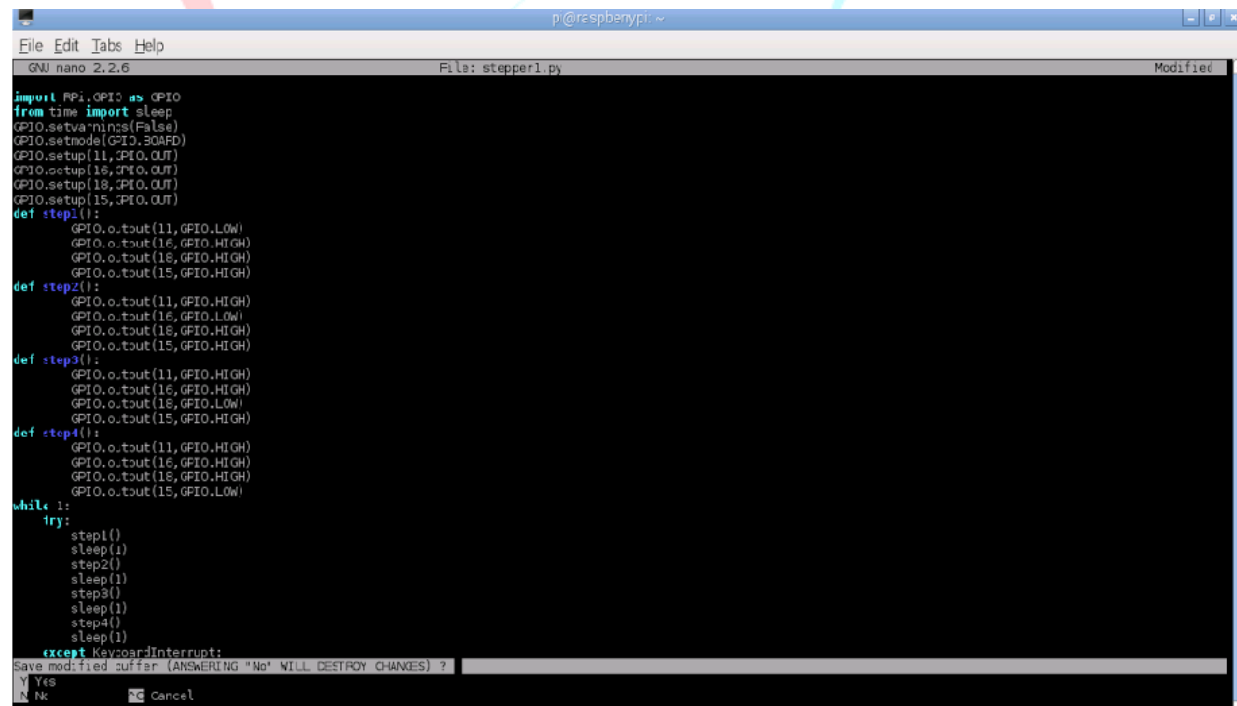
def step4():
    GPIO.output(11, GPIO.HIGH)
    GPIO.output(16, GPIO.HIGH)
    GPIO.output(18, GPIO.HIGH)
    GPIO.output(15, GPIO.LOW)

while 1:
    try:
        step1()
        sleep(1)
        step2()
        sleep(1)
        step3()
        sleep(1)
        step4()
        sleep(1)
    except KeyboardInterrupt:
        pass

Get Help WriteOut Read File Prev Page Cut Text Cur Pos
Exit Justify Where Is Next Page UnCut Text To Spell
```

Figure4

After entering your code press the CTRLX to save your code. And it will prompt you that save mod at the bottom of the window. Press Y and hit enter key



```
File Edit Tabs Help
GNU nano 2.2.6 File: stepper1.py Modified

import RPi.GPIO as GPIO
from time import sleep
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BOARD)
GPIO.setup(11, GPIO.OUT)
GPIO.setup(16, GPIO.OUT)
GPIO.setup(18, GPIO.OUT)
GPIO.setup(15, GPIO.OUT)

def step1():
    GPIO.output(11, GPIO.LOW)
    GPIO.output(16, GPIO.HIGH)
    GPIO.output(18, GPIO.HIGH)
    GPIO.output(15, GPIO.HIGH)

def step2():
    GPIO.output(11, GPIO.HIGH)
    GPIO.output(16, GPIO.LOW)
    GPIO.output(18, GPIO.HIGH)
    GPIO.output(15, GPIO.HIGH)

def step3():
    GPIO.output(11, GPIO.HIGH)
    GPIO.output(16, GPIO.HIGH)
    GPIO.output(18, GPIO.LOW)
    GPIO.output(15, GPIO.HIGH)

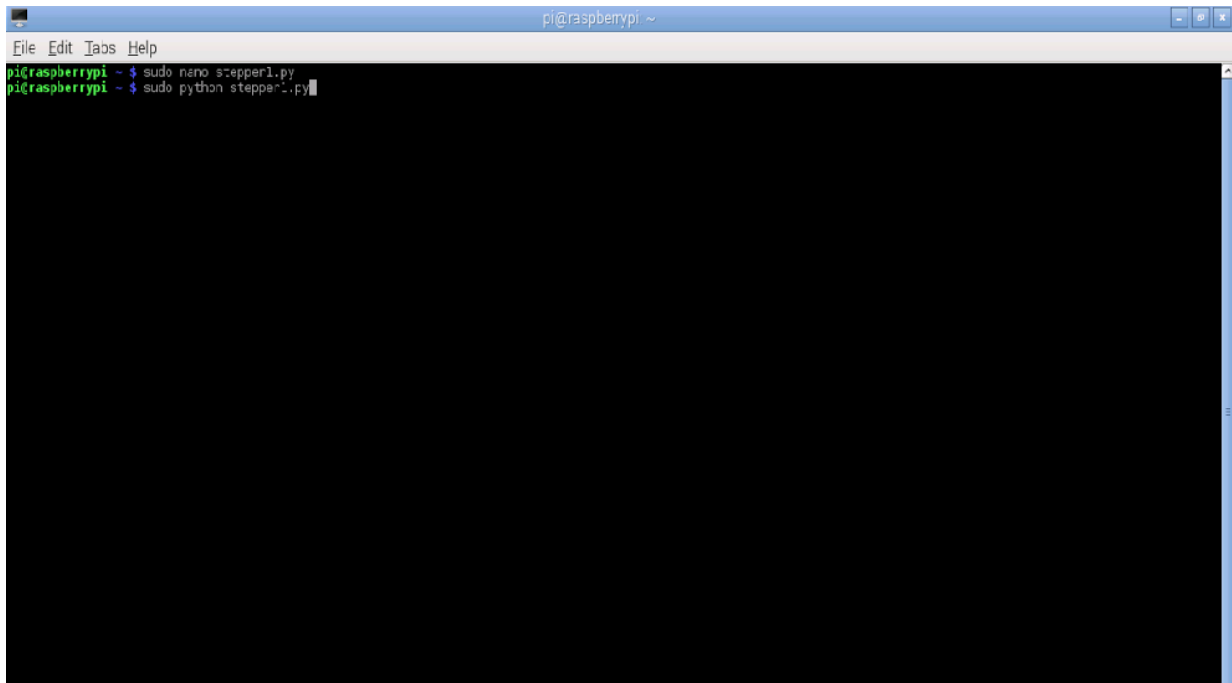
def step4():
    GPIO.output(11, GPIO.HIGH)
    GPIO.output(16, GPIO.HIGH)
    GPIO.output(18, GPIO.HIGH)
    GPIO.output(15, GPIO.LOW)

while 1:
    try:
        step1()
        sleep(1)
        step2()
        sleep(1)
        step3()
        sleep(1)
        step4()
        sleep(1)
    except KeyboardInterrupt:
        pass

Save modified buffer (ANSWERING "No" WILL DESTROY CHANGES) ?
Y Yes
N No Cancel
```

Figure5

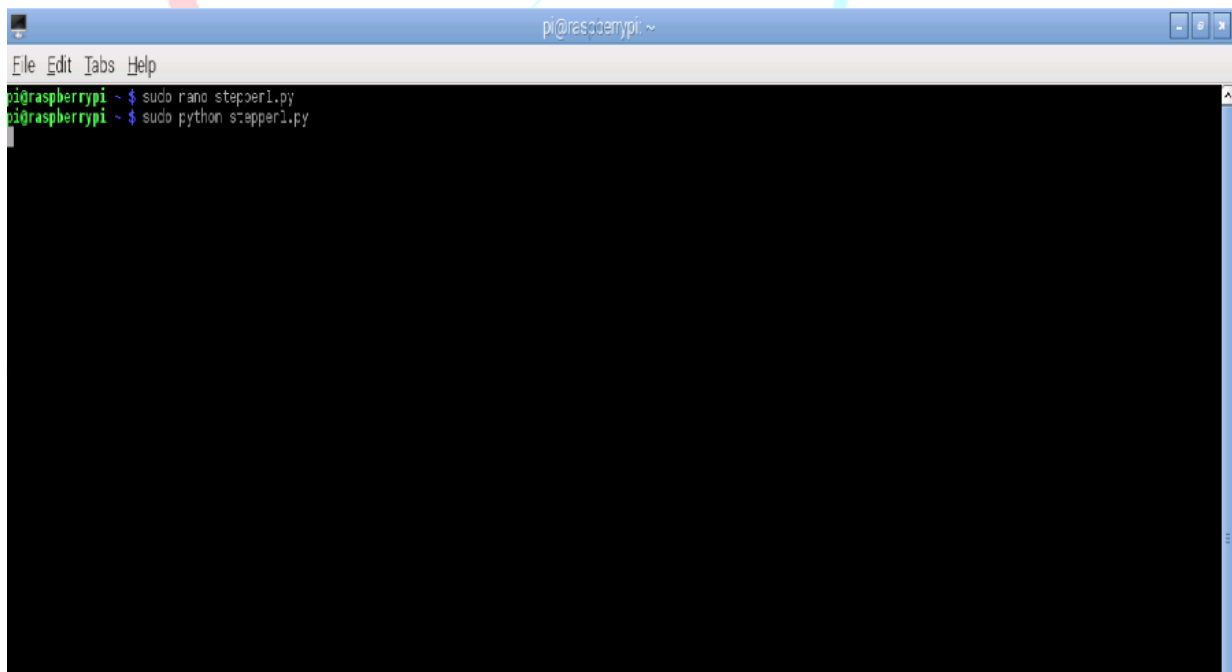
Step 5: Run the code by using **sudo python filename.py** command. Hit enter key

A terminal window titled 'pi@raspberrypi ~' with a menu bar (File, Edit, Tabs, Help). The command history shows 'pi@raspberrypi ~ \$ sudo nano stepper1.py' followed by 'pi@raspberrypi ~ \$ sudo python stepper1.py'. The terminal area is mostly black, indicating the program is running. The cursor is at the end of the second command line.

```
pi@raspberrypi ~  
File Edit Tabs Help  
pi@raspberrypi ~ $ sudo nano stepper1.py  
pi@raspberrypi ~ $ sudo python stepper1.py
```

Figure6

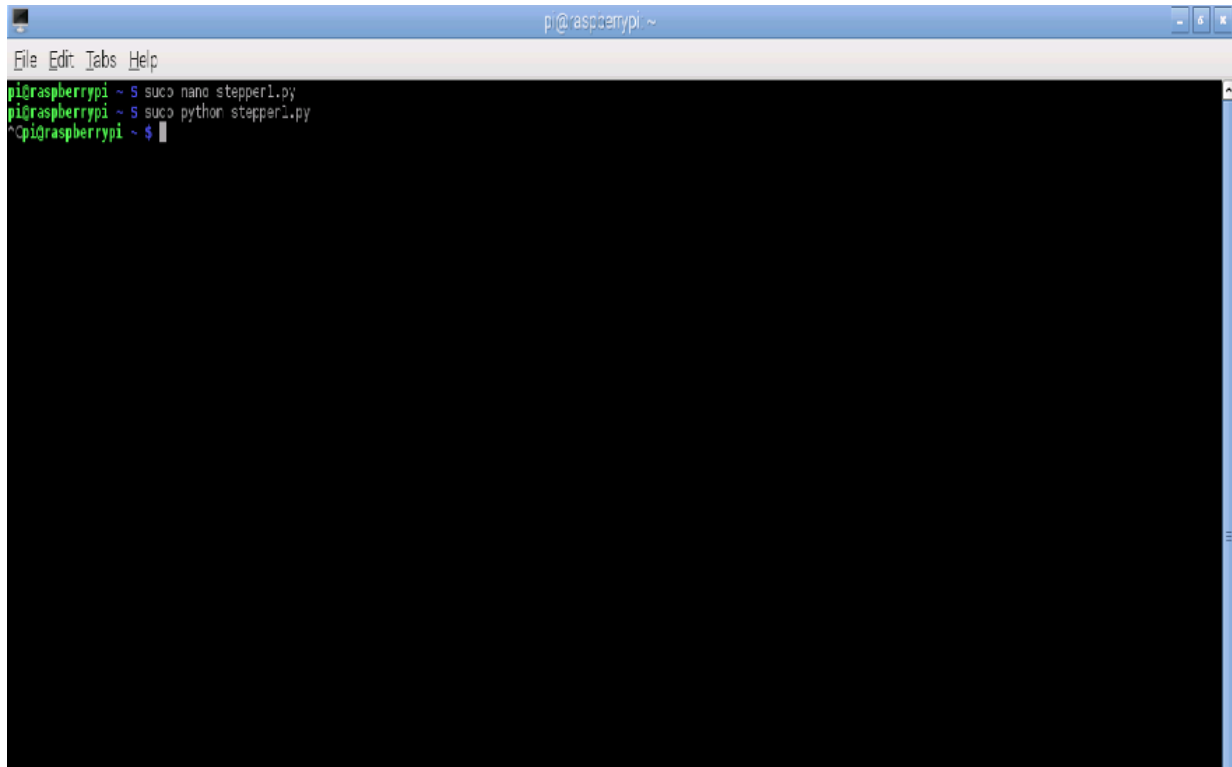
Step 6: On hitting ENTER key, your program will start to run and find the window as given below. And the cursor blinking in the below of your command.

A terminal window titled 'pi@raspberrypi ~' with a menu bar (File, Edit, Tabs, Help). The command history shows 'pi@raspberrypi ~ \$ sudo nano stepper1.py' followed by 'pi@raspberrypi ~ \$ sudo python stepper1.py'. The terminal area is mostly black, indicating the program is running. The cursor is at the end of the second command line, blinking.

```
pi@raspberrypi ~  
File Edit Tabs Help  
pi@raspberrypi ~ $ sudo nano stepper1.py  
pi@raspberrypi ~ $ sudo python stepper1.py
```

Figure7

Step 7: Program can be terminated by clicking CTRL+C button .Find the window below.



The image shows a terminal window titled 'pi@raspberrypi ~'. The window has a menu bar with 'File', 'Edit', 'Tabs', and 'Help'. The terminal output shows the following commands and their results:

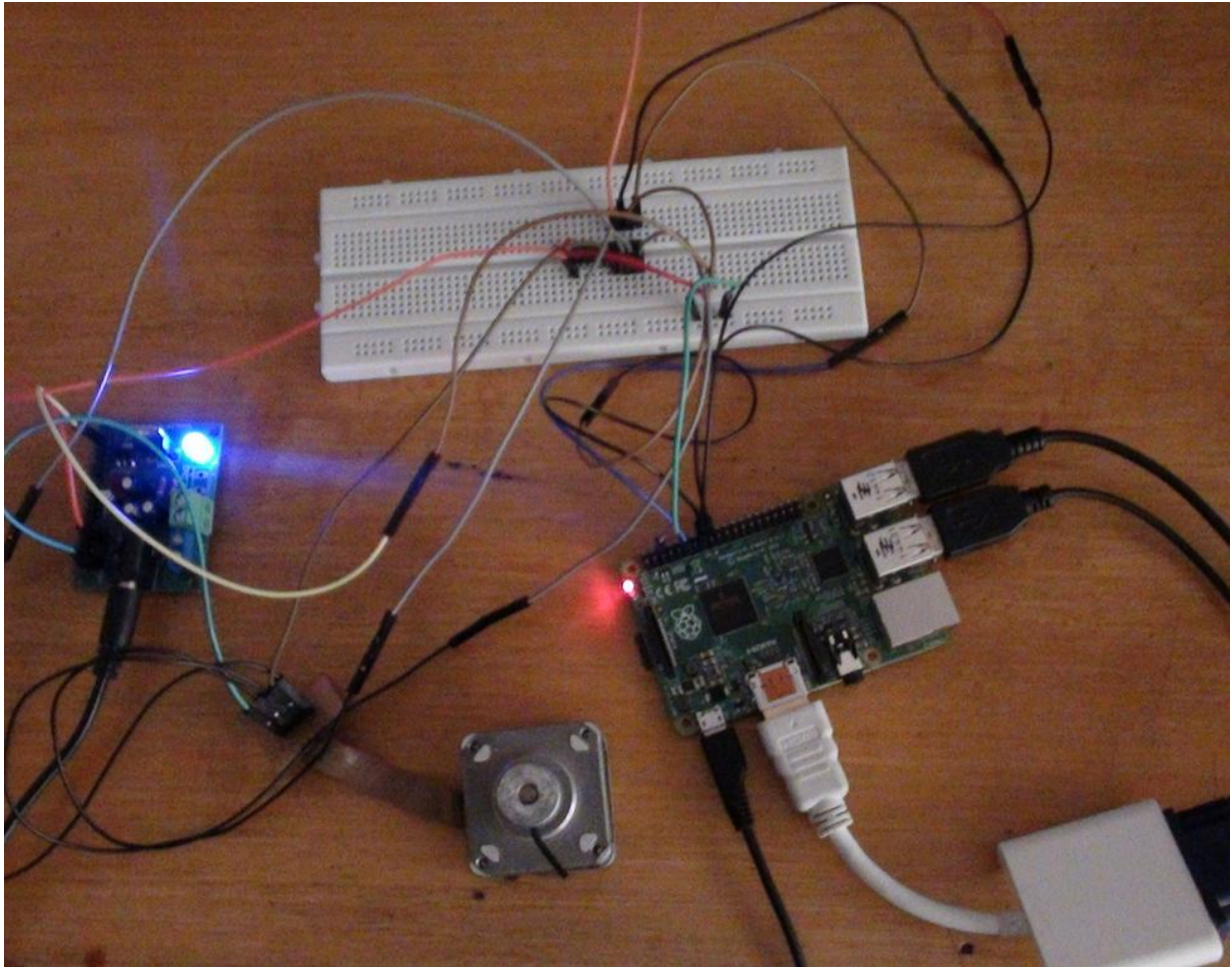
```
pi@raspberrypi ~ $ sudo nano stepper1.py
pi@raspberrypi ~ $ sudo python stepper1.py
pi@raspberrypi ~ $
```

The terminal window is mostly black, indicating that the program has been terminated and the screen is blank.

Figure8

TENET
TECHNETRONICS

OUTPUT:



For more information please visit: www.tenettech.com

For technical query please send an e-mail: info@tenettech.com