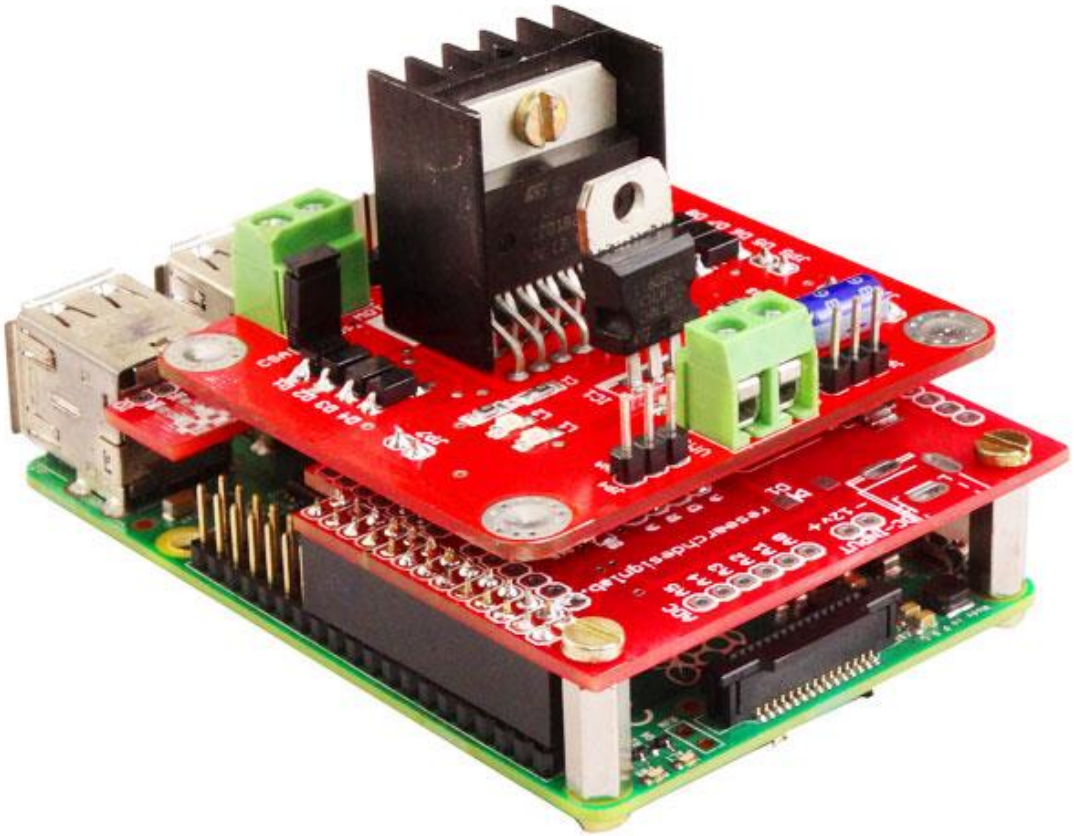
Research Design Lab



RDL L298 MOTOR DRIVER SHIELD FOR RASPBERRY-PI

Contents:

Introduction……...……3

Features…….....3

Compatibility ……...3

Package contains……...4

Internal Block Connections…….….4

Sample Python script……...….4

Introduction:

L298N is a high voltage, high current motor driver chip, with the highest working voltage of 46V, continuous operating current of 2A, and instantaneous peak current up to 3A. The chip contains two "H bridges" which are high-voltage and high current full-bridge drivers that can directly drive two DC motors.

Features:

- Driver: L298N Dual H Bridge DC Motor Driver IC
- Motor Power Supply Vs: +5 V to +35 V
- Max average current 2A
- peak current Io: 3A
- Logic Level Power Vss: +5 V ~ +7 V (Onboard 5V Regulator can be used if Motor Power is > 7.0V)
- Logic level power: 0 ~ 36mA
- Two motor direction indicator LEDs.
- Screw-terminals for power and motor connections.
- High quality PCB FR4 Grade with FPT Certified.
-

Compatibility:

- Raspberry Pi
- Raspberry Pi 2
- Raspberry Pi Model B+
- Raspberry Pi zero

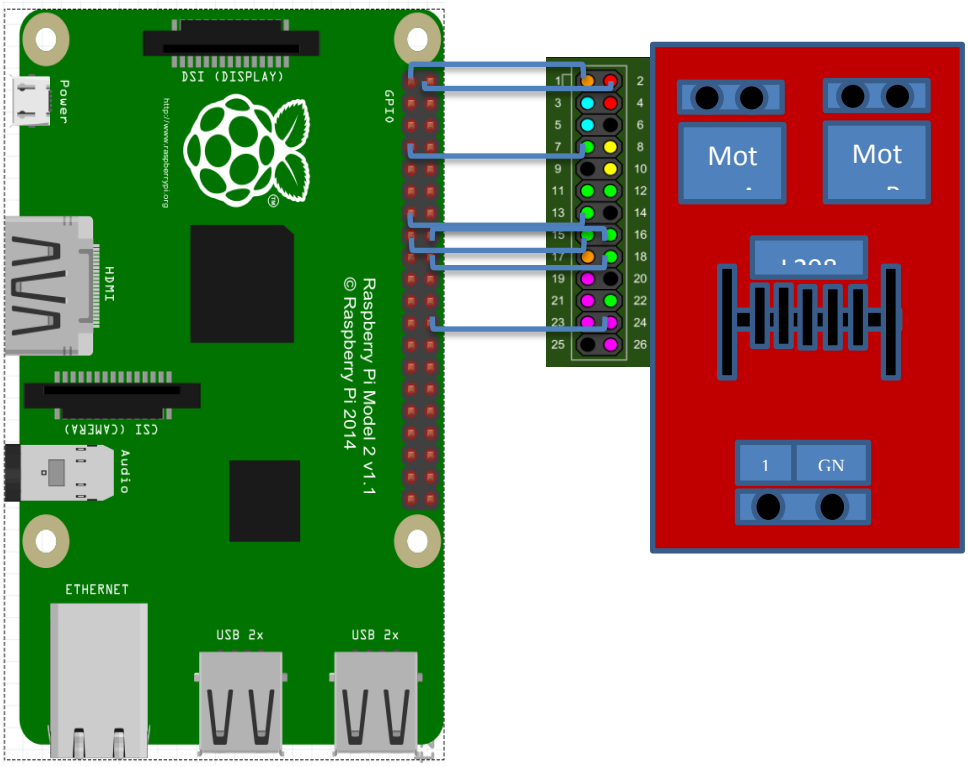
Package contains:

- L298N Motor Driver Shield Compatible to Raspberry Pi

Note1 : The Raspberry Pi in the picture above is just for reference , it does not include in the package.

Note2: Base board is only used for soldering the L298N shield and establishing the respective connectivity with Raspberry Pi I/O pins.

Internal Block Connections:

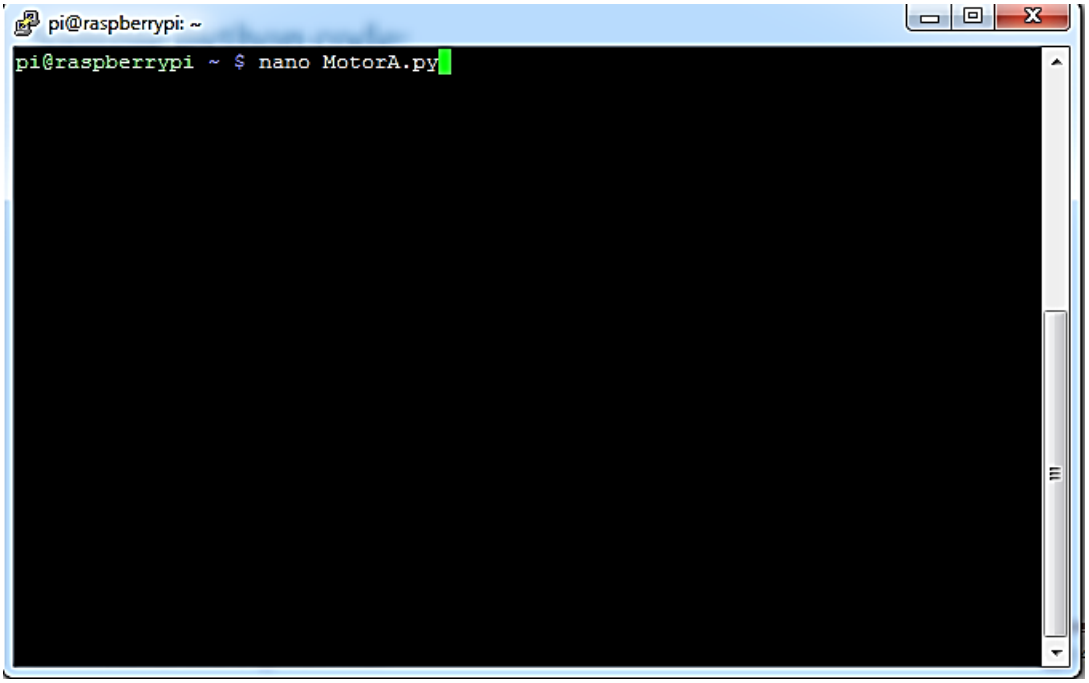


Sample python code:

Below is the sample code for motor A

open nano editor by typing following command

nano MotorA.py



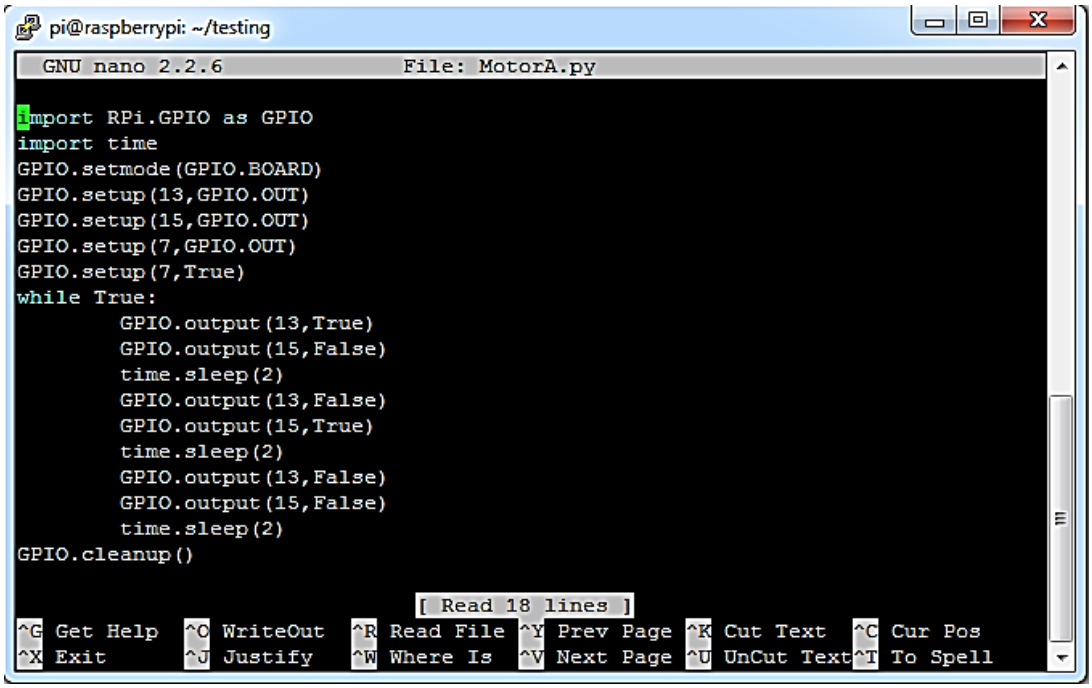
Inside nano editor enter the following command

```
import RPi.GPIO as GPIO
import time                                  } Importing necessary packages
GPIO.setmode(GPIO.BOARD)
GPIO.setup(13,GPIO.OUT)                      } here Motor A is connected to pin no
GPIO.setup(15,GPIO.OUT)                      } 13 and 15 of Raspberry so making
                                             } That pin as output mode
GPIO.setup(7,GPIO.OUT)                       } enable is connected to pin 7, making
GPIO.output(7,True)                          } that pin high
while True:
    GPIO.output(13,True)                     } rotating Motor A in one direction
    GPIO.output(15,False)                    }
    time.sleep(2)
```

```
    GPIO.output(13,False)
    GPIO.output(15,True)
    time.sleep(2)
    GPIO.output(13,False)
    GPIO.output(15,False)
    time.sleep(2)
GPIO.cleanup()
```

rotating motor A in opposite direction

stopping Motor A rotation



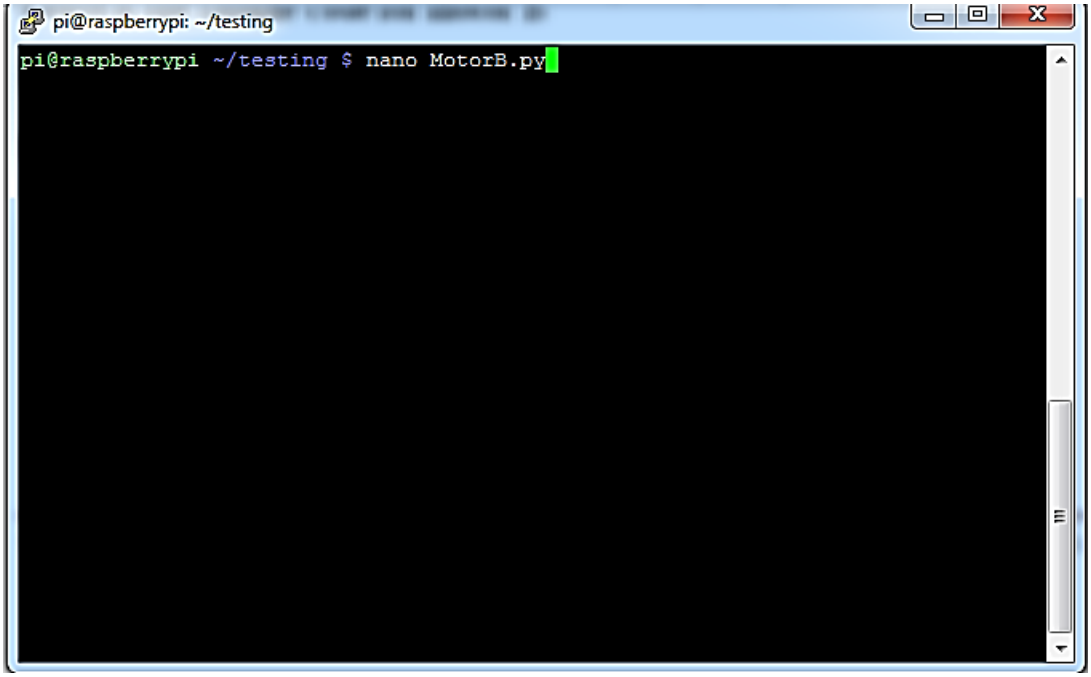
Run the above code by following command

sudo python MotorA.py

Below is the sample code for motor B

open nano editor by typing following command

nano MotorB.py



Inside nano editor enter the following command

```
import RPi.GPIO as GPIO
import time                                  } Importing necessary packages
GPIO.setmode(GPIO.BOARD)
GPIO.setup(16,GPIO.OUT)                      } here MotorB is connected to pin no
GPIO.setup(18,GPIO.OUT)                      } 16 and 18 of Raspberry so making
                                             } That pin as output mode
GPIO.setup(24,GPIO.OUT)                      } enable is connected to pin 24, making
GPIO.output(24,True)                         } that pin high
while True:
    GPIO.output(16,True)                     } rotating Motor B in one direction
    GPIO.output(18,False)                    }
    time.sleep(2)
```

GPIO.output(16,False) } rotating motor B in opposite direction

GPIO.output(18,True)

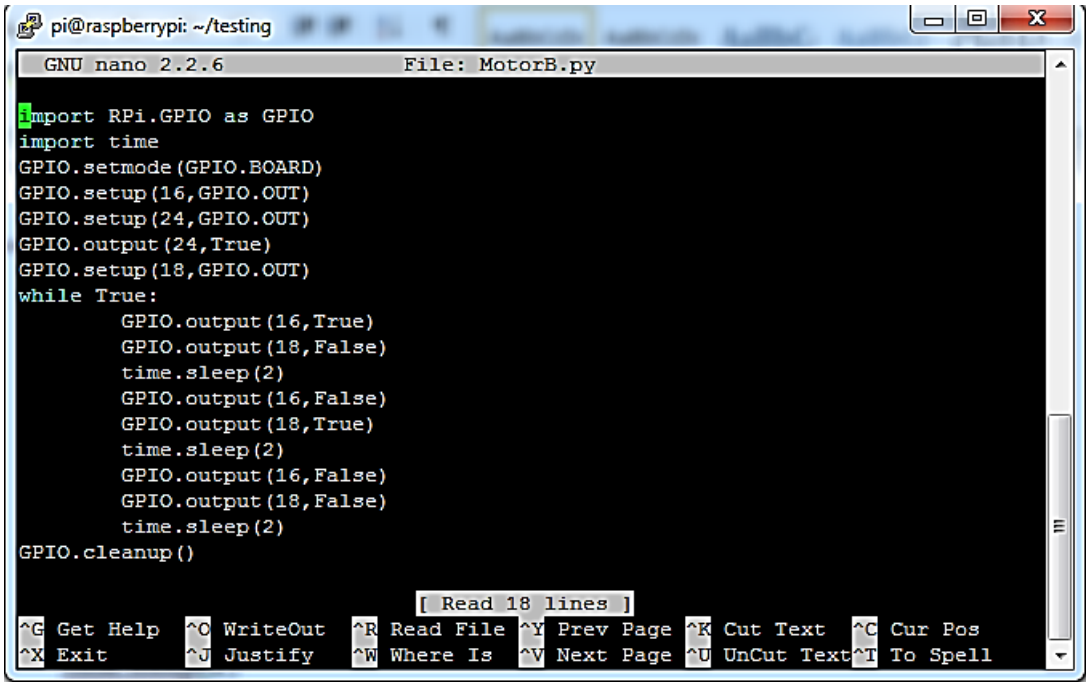
time.sleep(2)

GPIO.output(16,False) } stopping Motor B rotation

GPIO.output(18,False)

time.sleep(2)

GPIO.cleanup()



Run the above code by following command

sudo python MotorB.py