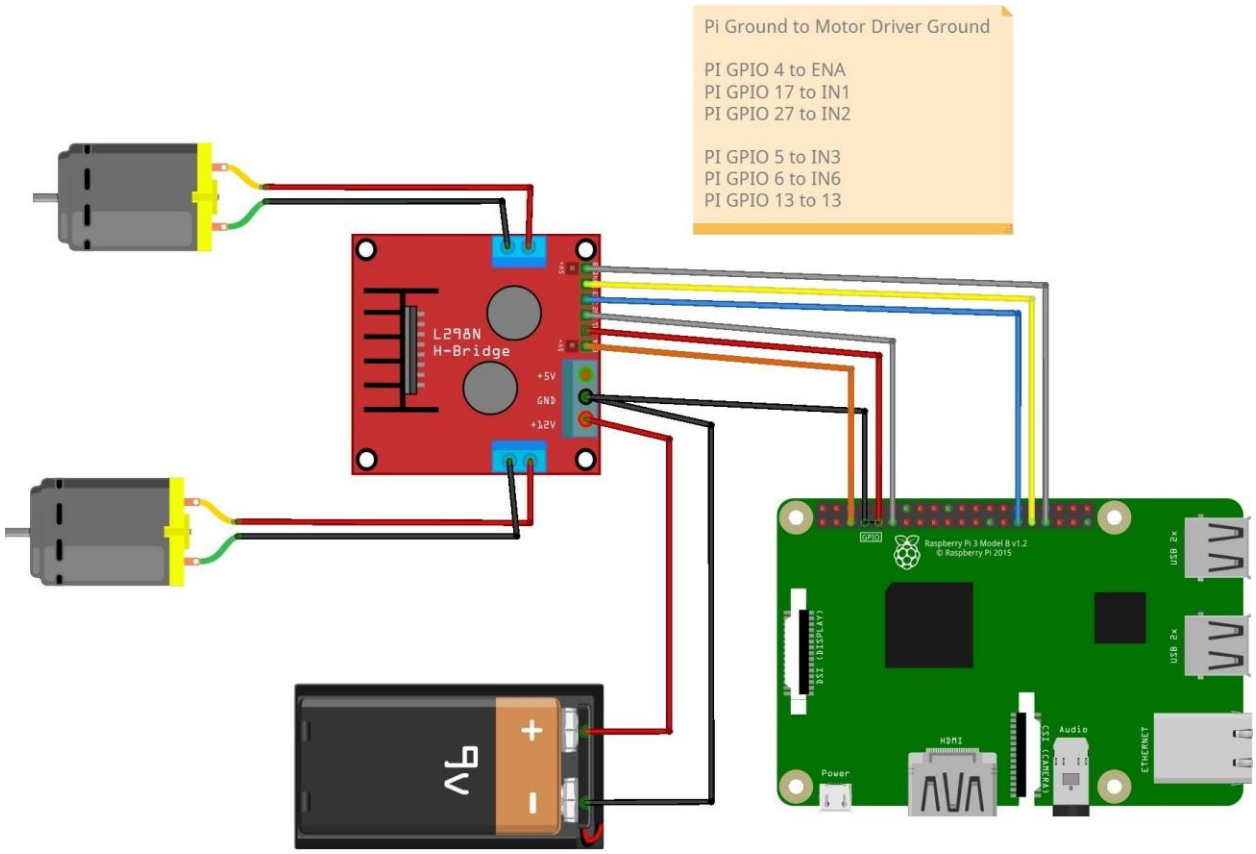
Raspberry Pi with DC Motor

Code on a Raspberry pi:

```
import RPi.GPIO as GPIO
from time import sleep

GPIO.setwarnings(False)

# Right Motor
in1 = 17
in2 = 27
en_a = 4
# Left Motor
in3 = 5
in4 = 6
en_b = 13


GPIO.setmode(GPIO.BCM)
GPIO.setup(in1,GPIO.OUT)
GPIO.setup(in2,GPIO.OUT)
GPIO.setup(en_a,GPIO.OUT)

GPIO.setup(in3,GPIO.OUT)
GPIO.setup(in4,GPIO.OUT)
GPIO.setup(en_b,GPIO.OUT)

power_a =GPIO.PWM(en_a,100)
power_b =GPIO.PWM(en_b,100)
power_b.start(75)
power_a.start(75)

GPIO.output(in1,GPIO.LOW)
GPIO.output(in2,GPIO.LOW)
GPIO.output(in4,GPIO.LOW)
GPIO.output(in3,GPIO.LOW)

# Wrap main content in a try block so we can catch the user pressing CTRL-C and run the
# GPIO cleanup function. This will also prevent the user seeing lots of unnecessary error
   messages.
try:
# Create Infinite loop to read user input
   while(True):
      # Get user Input
      user_input = input()

      # To see users input
      # print(user_input)
```

```
      if user_input == 'w':
         GPIO.output(in1,GPIO.HIGH)
         GPIO.output(in2,GPIO.LOW)

         GPIO.output(in4,GPIO.HIGH)
         GPIO.output(in3,GPIO.LOW)

         print("Forward")

      elif user_input == 's':
         GPIO.output(in1,GPIO.LOW)
         GPIO.output(in2,GPIO.HIGH)

         GPIO.output(in4,GPIO.LOW)
         GPIO.output(in3,GPIO.HIGH)
         print('Back')

      elif user_input == 'd':
         GPIO.output(in1,GPIO.LOW)
         GPIO.output(in2,GPIO.HIGH)

         GPIO.output(in4,GPIO.LOW)
         GPIO.output(in3,GPIO.LOW)
         print('Right')

      elif user_input == 'a':
         GPIO.output(in1,GPIO.HIGH)
         GPIO.output(in2,GPIO.LOW)

         GPIO.output(in4,GPIO.LOW)
         GPIO.output(in3,GPIO.LOW)
         print('Left')

      # Press 'c' to exit the script
      elif user_input == 'c':
         GPIO.output(in1,GPIO.LOW)
         GPIO.output(in2,GPIO.LOW)

         GPIO.output(in4,GPIO.LOW)
         GPIO.output(in3,GPIO.LOW)
         print('Stop')

# If user press CTRL-C
except KeyboardInterrupt:
  # Reset GPIO settings
  GPIO.cleanup()
  print("GPIO Clean up")
```

Circuit diagram:



If you want to turn your motors when connected with power, you can choose this code:

```
import RPi.GPIO as GPIO
from time import sleep

GPIO.setwarnings(False)

# Right Motor
in1 = 17
in2 = 27
en_a = 4
# Left Motor
in3 = 5
in4 = 6
en_b = 13

# Set up GPIO mode
GPIO.setmode(GPIO.BCM)

# Set motor control pins as output
GPIO.setup(in1, GPIO.OUT)
GPIO.setup(in2, GPIO.OUT)
GPIO.setup(en_a, GPIO.OUT)

GPIO.setup(in3, GPIO.OUT)
GPIO.setup(in4, GPIO.OUT)
GPIO.setup(en_b, GPIO.OUT)

# Initialize PWM for speed control
q = GPIO.PWM(en_a, 100)  # Right motor speed control
p = GPIO.PWM(en_b, 100)  # Left motor speed control
p.start(75)  # Start with 75% speed
q.start(75)  # Start with 75% speed

# Initialize motor states (ensure motors are stopped initially)
GPIO.output(in1, GPIO.LOW)
GPIO.output(in2, GPIO.LOW)
GPIO.output(in4, GPIO.LOW)
GPIO.output(in3, GPIO.LOW)

try:
    # Start motors in forward direction when the Raspberry Pi is powered on
    print("Motors starting, moving forward...")

    # Right motor forward
    GPIO.output(in1, GPIO.HIGH)
    GPIO.output(in2, GPIO.LOW)
```

```
    # Left motor forward
    GPIO.output(in4, GPIO.HIGH)
    GPIO.output(in3, GPIO.LOW)

    # Keep the motors running indefinitely
    while True:
        sleep(1)  # Just a delay to keep the motors running, loop will run
indefinitely

# Handle cleanup if power is lost or user stops the program
except KeyboardInterrupt:
    # If user presses CTRL-C to stop the program
    print("Stopping motors and cleaning up GPIO...")
    GPIO.cleanup()

finally:
    # If power is disconnected or any error occurs, stop the motors and cleanup
    print("GPIO cleanup on power disconnection...")
    GPIO.cleanup()
```