PWM and Serial Comms

Open the data port from MATLAB.

```
clear s % closes ports already open
s = serialport('COM4',9600) % Check COM#
```

```
s =
  Serialport with properties:

                Port: "COM4"
            BaudRate: 9600
    NumBytesAvailable: 0

  Show all properties, functions
```

Write a message to send the duty cycle.

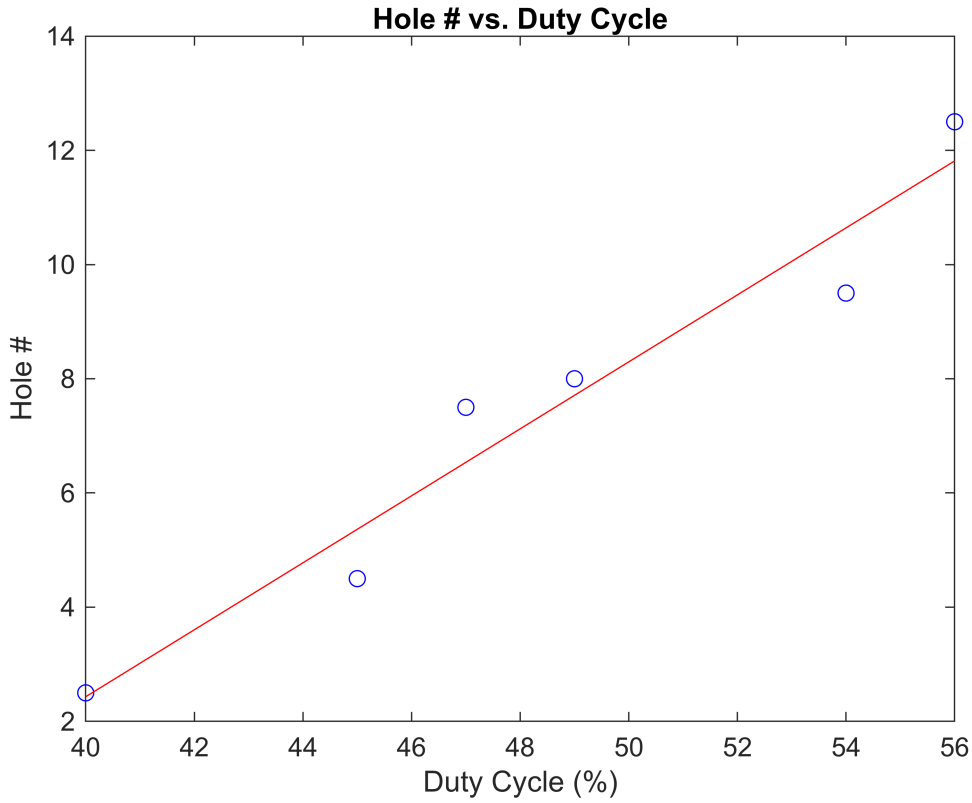
```
s.writeline("0.5"); % write duty cycle to Pico
```

In MATLAB, plot the data points of Hole # vs. Duty Cycle. The notional line or curve nearest to these points represents the load characteristic.

```
duty_cycle = [40 45 47 49 54 56]; % expiremented duty cycles
hole_num = [2.5 4.5 7.5 8 9.5 12.5]; % resultant hold number
plot(duty_cycle, hole_num,'bo') % plot points

pfit = polyfit(duty_cycle,hole_num,1); % fit line to data
pval = polyval(pfit,duty_cycle); % evaulate fitted line
hold on
plot(duty_cycle,pval, 'r') % plot fitted line
hold off

title("Hole # vs. Duty Cycle")
xlabel("Duty Cycle (%)")
ylabel("Hole #")
```


Hole # vs. Duty Cycle
Hole #
Duty Cycle (%)
14
12
10
8
6
4
2
40
42
44
46
48
50
52
54
56

```
from machine import Pin, PWM
import ttyacm

tty = ttyacm.open(1)

motor = PWM(Pin(16)) # DC motor control from GPIO16
motor.freq(1000) # set frequency to 1KHz -- DO NOT CHANGE
motor.duty_u16(32768) # 0-65535 for duty cycle range 0-100

#########################################################
# NOTE: The DC motor of the fan is NOT BRUSHED          #
# Keep the duty cycle above 35%                         #
#########################################################

# YOUR CODE HERE

# duty_cycle = 0.6 # set duty cycle

while True:
    duty_cycle = float(tty.readline()) # read duty cycle value from the serial port
    print(duty_cycle) # print out read duty cycle
    convert_dc = int(duty_cycle * 65535) # convert duty cycle to an integer 0 to 65535
    motor.duty_u16(convert_dc) # set motor pin to new duty cycle
    if duty_cycle == -1: # terminate loop if input read from serial port is -1
        break

motor.duty_u16(0) # upon termination, set motor pin to duty cycle 0
```