ECE 476/676 - Homework #10

GPS & BlueTooth - Due Monday, November 18th

GPS: How Fast Can You Run?

1) (30 points): Write a Python program which measures how fast you can run using a GPS sensor.

- Record your speed with a GPS sensor
- Start new recording by pressing button GP15
- While recording, detect the maximum speed seen from the GPS sensor, and
- Display the top three speeds on graphics display

Include

- Your Python program
- Data showing it is working
- A photo of your graphics display showing your three fastest speeds

The subroutines are from Bison Academy with the GPS Speedometer. The main routine changes

- When mph > 2, start recording the maximum speed
- When mph drops below 1,
  - Save the max speed into the array Top4
  - Sort the array Top4
  - Clear the max speed

```
while(Button15.value() == 1):
    Error_Flag = 1
    while(Error_Flag == 1):
        Error_Flag = 0
        [t, x, y, v] = GPS_Read(0)
    mph = v*1.15078
    if(mph < 1):
        if(flag2 == 1):
            Top4[3] = Vmax
            Top4 = sorted(Top4, reverse = True)

        flag2 = 0
        Vmax = 0
    if(mph > 2):
        Vmax = max(mph, Vmax)
        flag2 = 1

    LCD.Number4(mph, 5, 1, 150, 50, White, Navy)
    LCD.Number4(Vmax, 5, 1, 150, 85, Pink, Navy)
    LCD.Number4(Top4[0], 5, 1, 150, 120, White, Navy)
    LCD.Number4(Top4[1], 5, 1, 150, 155, White, Navy)
    LCD.Number4(Top4[2], 5, 1, 150, 190, White, Navy)
```

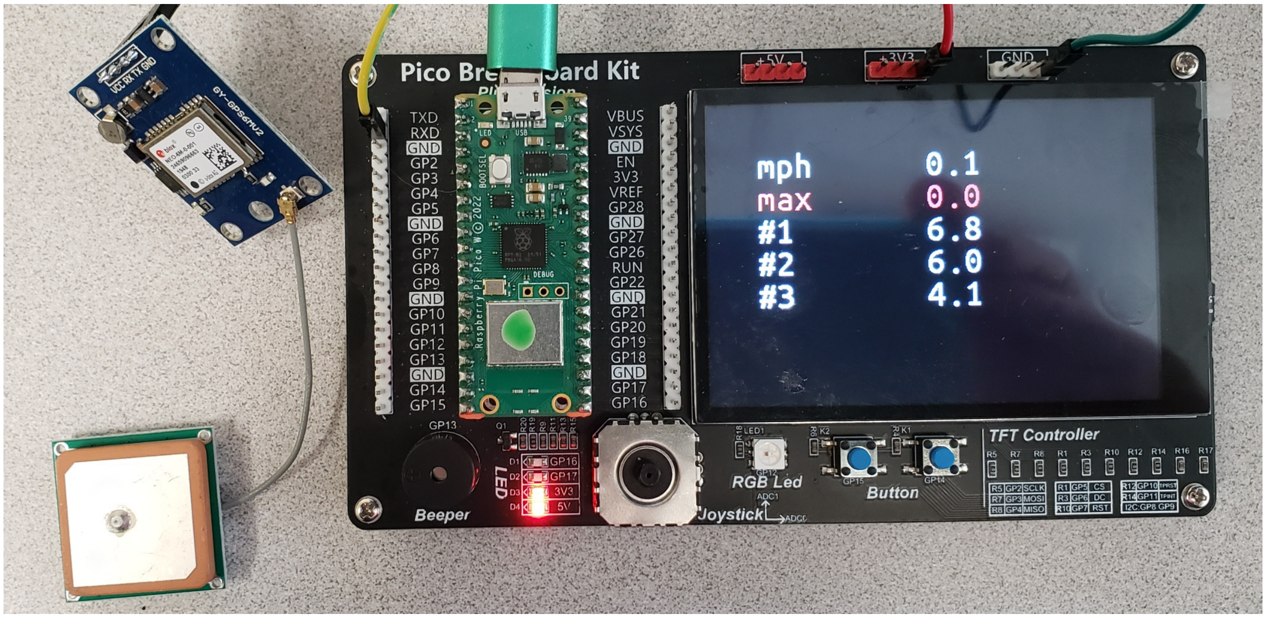
2) (10 points): Demo your program

- Short video preferred
- Photos also work

For this test

- I went outside to get a lock on the GPS sattelites,
- Walked to the sidewalk, stopped (4.1mph max speed recorded)
- Jogged back to the building and stopped (6.8mph max speed recorded), then
- Jogged back to the sidewalk (6.0mph recorded)

The resulting top three speeds are then displayed on the LCD screen:



```
mph      0.1
max      0.0
#1       6.8
#2       6.0
#3       4.1
```

BlueTooth & Motor Speed:

3) (30 Points): Write a Python program which allows you to control the speed of a DC motor using your cell phone and a bluetooth interface. Some options are:

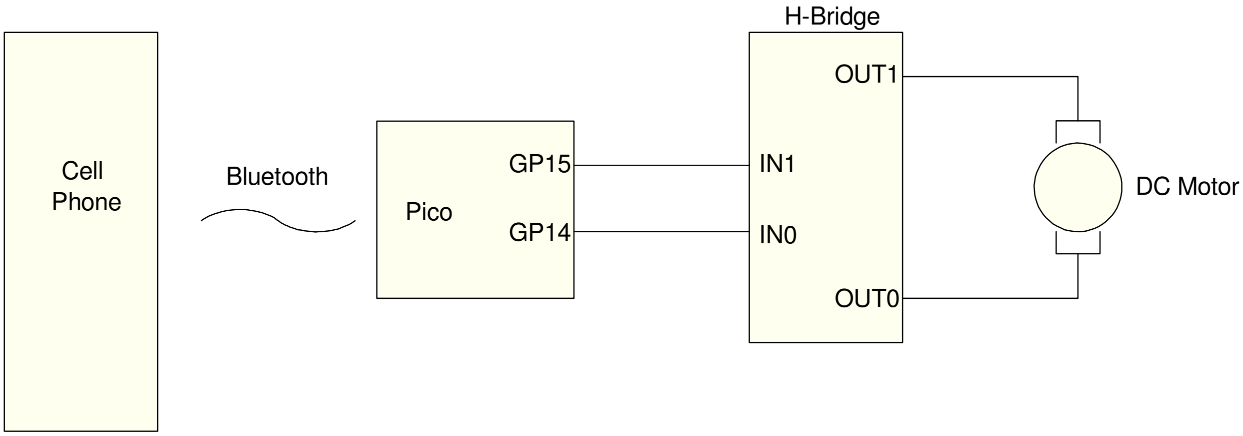
- Use PWM to vary the speed and direction of the DC motor
- Use commands to set the direction (CW or CCW) and speed (000 to 100)
- Use a long string to input both (+100, -085, etc)

When completed, you should be able to set the voltage to the motor from -100% to +100%

Inlcude:

- Your Python program
- Data showing it is working (CW, CCW, speed or votlage changes with data input)

Hardware



PWM Initialzation

- Pins 14 and 15
- 100Hz
- 0% to 100% PWM

```
from machine import Pin, PWM
import bluetooth
from ble_simple_peripheral import BLESimplePeripheral
import neopixel, LCD, time

Out1 = Pin(14, Pin.OUT)
Out2 = Pin(15, Pin.OUT)

fwd = PWM(Pin(14))
rev = PWM(Pin(15))

fwd.freq(100)
rev.freq(100)
```

on_rx() routine

- Pull out the speed as a number from -100 to 100
- Experimentation shows that parsing the fields isn't necessary

```
def on_rx(data):
    global Speed, flag
    print("Data received: ", data)
    try:
        Speed = int(data[0:4])
        flag = 1
    except:
        print('invalid data entry')
```

Main Loop

- When you receive a bluetooth command
  - Print the speed received
  - Set the PWM based upon the speed
  - 100 = 65535 / 0
  - -100 = 0 / 65535
  - Proportional inbetween

```
kV = 65535/100

while(1):
    if sp.is_connected():
        sp.on_write(on_rx)
    if(flag):
        print('Speed = ',Speed)
        LCD.Text2(str(Speed) + '  ', 300, 50, Yellow, Black)
        if(Speed > 0):
            fwd.duty_u16(int(Speed*kV))
            rev.duty_u16(0)
        else:
            fwd.duty_u16(0)
            rev.duty_u16(int(-Speed*kV))
        flag = 0
```

Shell Window

- showing that it receives commands ranging from -100 to +100

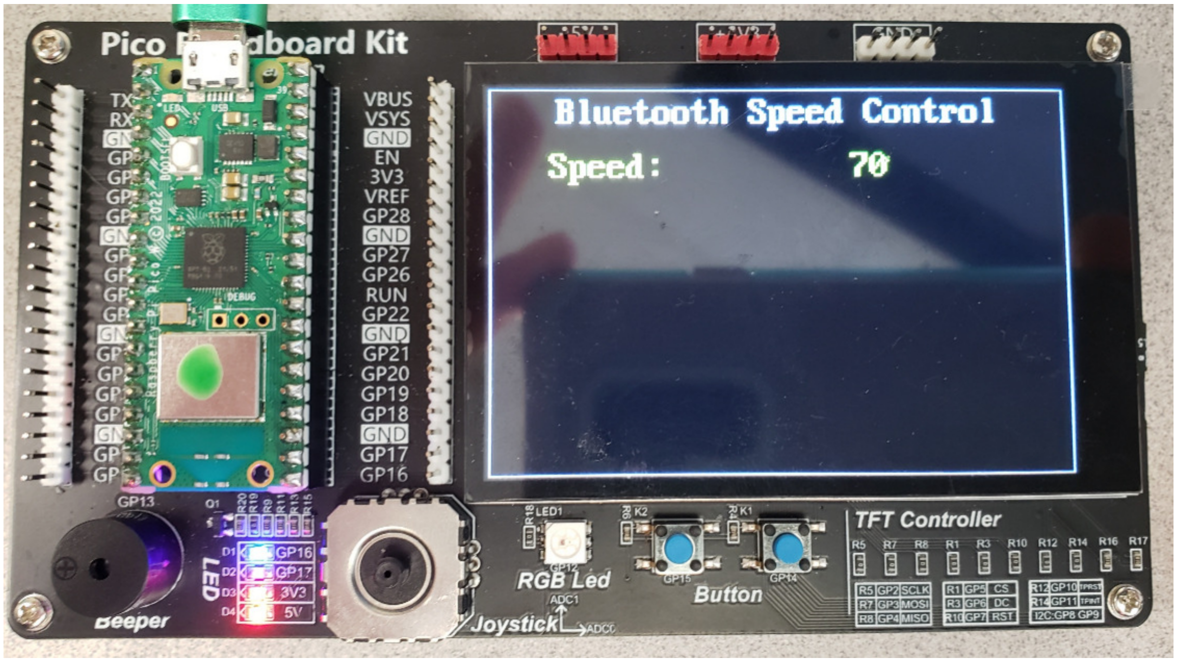
```
Starting advertising
New connection 64
Data received:  b'70\r\n'
Speed =  70
Data received:  b'7\r\n'
Speed =  7
Data received:  b'0\r\n'
Speed =  0
Data received:  b'-70\r\n'
Speed =  -70
Data received:  b'-40\r\n'
Speed =  -40
```

4) (10 points); Demo your program

- Short video preferred
- Photos also work

Connecting to a DC motor and an H-bridge resulted in

- Type in 50 <cr>
  - Motor spins clockwise
- Type in 30 <cr>
  - Motor spins slower clockwise
- Type in 0 <cr>
  - Motor stops
- Type in -50 <cr>
  - Motor spins counter-clockwise



Bluetooth Speed Control

Speed: 70