

# Code EXP 3

October 17, 2019

```
[ ]: import busio
import digitalio
import board
import adafruit_mcp3xxx.mcp3008 as MCP
from adafruit_mcp3xxx.analog_in import AnalogIn
import time

spi = busio.SPI(clock=board.SCK, MISO=board.MISO, MOSI=board.MOSI)
cs = digitalio.DigitalInOut(board.D5)

# Create an MCP3008 object
mcp = MCP.MCP3008(spi, cs)
# Create an analog input channel on the MCP3008 pin 0
channel1 = AnalogIn(mcp, MCP.P0) #left sensor
channel2 = AnalogIn(mcp, MCP.P1) # right sensor

previous_channel1_values = []
previous_channel2_values = []
difference_channel1_values = []
difference_channel2_values = []

# assuming that the voltage values decreases as it gets darker/(motion moves????
→)

while True:
    #print('Left Sensor - Raw ADC Value: ', channel1.value)
    #print('Left Sensor -ADC Voltage: ' + str(channel1.voltage) + 'V')
    #print('Right Sensor - Raw ADC Value: ', channel2.value)
    #print('Right Sensor -ADC Voltage: ' + str(channel2.voltage) + 'V')

    if len(previous_channel1_values) != 0 and len(previous_channel2_values) != 0:
→0:
        #print(abs(channel1.voltage - previous_channel1_values[-1]))
        if abs(channel1.voltage - previous_channel1_values[-1]) > 0.1:
            difference_channel1_values.append(channel1.voltage -
→previous_channel1_values[-1])
        else: difference_channel1_values.append(0)
```

```

        if abs(channel2.voltage - previous_channel2_values[-1]) > 0.1:
            difference_channel2_values.append(channel2.voltage -
previous_channel2_values[-1])
        else: difference_channel2_values.append(0)
        # move left --> difference_channel1_value: more positive
        # --> difference_channel2_value: more negative
        # move right --> difference_channel1_value: more negative
        # --> difference_channel2_value: more positive
        previous_channel1_values.append(channel1.voltage)
        previous_channel2_values.append(channel2.voltage)
        if len(difference_channel1_values) == 10 and
previous_channel2_values == 10:
            sign_difference_channel1 = []
            sign_difference_channel2 = []
            #print(difference_channel1_values)
            #print(difference_channel2_values)
            for x in range(len(difference_channel1_values)-1):
                if difference_channel1_values[x] > difference_channel1_values[x+1]:
                    sign_difference_channel1.append(-1)
                elif difference_channel1_values[x] <
previous_channel2_values[x+1]:
                    sign_difference_channel1.append(1)
                else: sign_difference_channel1.append(0)

            for x in range(len(difference_channel2_values)-1):
                if difference_channel2_values[x] > difference_channel2_values[x+1]:
                    sign_difference_channel2.append(-1)
                elif difference_channel2_values[x] <
previous_channel2_values[x+1]:
                    sign_difference_channel2.append(1)
                else: sign_difference_channel2.append(0)

            total_sign_difference_channel1 = 0
            total_sign_difference_channel2 = 0
            for y in range(len(sign_difference_channel1)):
                total_sign_difference_channel1 += y*sign_difference_channel1[y]

                total_sign_difference_channel2 += y*sign_difference_channel2[y]

            if total_sign_difference_channel1 > 0 and
previous_channel2_values < 0:
                print ("Left to Right Movement")

            elif total_sign_difference_channel1 < 0 and
previous_channel2_values > 0:

```

```
        print ("Right to Left Movement")

    else: print ("No Movement or Did not capture")

    difference_channel1_values = difference_channel1_values[1:-1]
    difference_channel2_values = difference_channel2_values[1:-1]

time.sleep(1)
```