ECE 476/676 - Homework #4

Timing, Analog I/O, Motors with Binary Inputs - Due Monday, September 23rd

Analog I/O: Electronic Trombone

Write a Python program to turn your Pico board into an electronic trombone

- When the analog input (joystick) is all the way left, the Pico outputs 220Hz
- When the analog input is all the way right, the Pico outputs 440Hz
- The frequency is proportional to the voltage inbetween

When button GP14 is pressed, output a square wave with the frequency determined by the joystick position

• When released, no sound is output

1) Give the resulting Python program

```
from time import sleep_ms
from machine import Pin, PWM, ADC
Spkr = PWM(Pin(0))
Spkr.freq(500)
a2d0 = ADC(0)
B0 = Pin(15, Pin.IN, Pin.PULL_UP)
while(1):
    if(B0.value() == 0):
        while(B0.value() == 0):
            a0 = a2d0.read_u16()
            f = 220 + (a0 * 220) // 65535
            Spkr.freq(f)
            Spkr.duty_u16(32000)
            Spkr.duty_u16(0)
```

2) Test your code at

- 220Hz (analog input = 0V)
- 440Hz (analog input = 3.3V)
 inbetween (check the voltage and frequency)

Plays when GP15 is pressed



3.3V plays 440Hz



0.0V plays 220Hz

Proportional inbetween

3) Demonstrate your electronic trombone

(see video)

Electronic Sunflower

Write a Python program so that a digital servo motor points at the sun (or some other light source)4) Design hardware so that the voltage is a maximum (0-3.3V) when pointing at a light source



5) Write a Python subroutine to read the voltage from problem #4

• Check that the number returned is a maximum when pointing at the light source)

```
from machine import Pin, PWM, ADC
from time import sleep_ms, sleep
a2d2 = ADC(2)
Vmax = 0
while(1):
    V = k*a2d2.read_u16()
    if(V > Vmax):
        Vmax = V
    print(V, Vmax)
```

In the shell window

V	Vmax					
0	0					
0.8866	0.8866					
1.6939	1.6939					
2.3500	2.3500					
2.7961	2.7961					
2.9925	2.9925					
2.9215	2.9925					
2.5896	2.9925					
2.0264	2.9925					
1.2821	2.9925					
0.4234	2.9925					

- 6) Write a Python subroutine which controls the angle of a DC servo motor
 - Check that you can control the angle of the servo



The motor moves from min to max angle



500ns pulse (min angle)

	RUN	CH1	DC 1X 1V/div	CH2 DC 1X 5¥/div	50005	3/diu	move slow	Т	iormal <mark>111 </mark>		CTRL
											RIIN7 Stop
											AIITO SET
1										Ţ	T CII RSOR
											V CII RSOR
											MFAS URES
											SAVF PIC
<mark>1</mark> Γreq∶a	***	1 T	im +: ***	1 Tim-	: ***						SAVE

250ms pulse (max angle)

7) Write a Python program that searches for what angle maximizes the output of the light sensor

```
from machine import Pin, PWM, ADC
from time import sleep_ms, sleep
M1 = Pin(0, Pin.OUT)
M1 = PWM(Pin(0))
M1.freq(100)
M1.duty_ns(900)
a2d2 = ADC(2)
PW = 1_{500}000
dP = 2000
k = 3.3 / 65535
print('Home Position')
M1.duty_ns(PW)
sleep(1)
Lmax = 0
flaq = 0
a2 = 32000
T = 100
n = 0
PWmax = 1_{500}000
flag = 0
while(1):
    if (flag == 0):
        PW0 = 500_{000}
        PW1 = 2 500 000
        flaq = 1
    else:
        PWO = max(500 \ 000, \ PWmax - 200 \ 000)
        PW1 = min(2_{500}, PWmax + 200_{000})
    while(PW > PW0):
        PW -= dP
        M1.duty_ns(PW)
        sleep_ms(5)
    PWmax = PW0
    Vmax = 0
    for PW in range (PW0, PW1, 2000):
        M1.duty_ns(PW)
        sleep_ms(5)
        V = k*a2d2.read u16()
        if(V > Vmax):
            PWmax = PW
            Vmax = V
    while(PW > PWmax):
        PW -= dP
        M1.duty_ns(PW)
        sleep_ms(5)
    print('max at PW = ', PW, ' Vmax = ', Vmax)
    sleep_ms(2000)
```

8) Demonstrate your program.

Every 5 seconds

- The motor swings back and forth
- While swinging, it records the pulse width that resulted in the maximum voltage
- It then points to the direction that resulted in the brightest light (max voltage)

MPY:	SC	oft	rek	poot				
Home Position								
max	at	ΡW	=	1140000	Vmax =	0.9613732		
max	at	ΡW	=	1032000	Vmax =	0.9355916		
max	at	ΡW	=	908000	Vmax =	0.9492882		
max	at	ΡW	=	860000	Vmax =	0.9565392		
max	at	ΡW	=	852000	Vmax =	0.9581506		



Demo: Motor swings back and forth, ending up pointing at the light source (note: H-bridge on PCB isn't being used - remnant of previous experiments)