ECE 476/676 - Test #2: Name _____

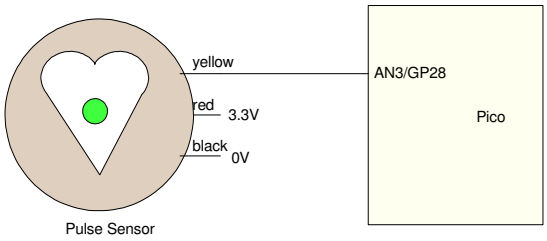
Due Sunday, March 30th at midnight. Individual Effort.

Heart Rate Sensor

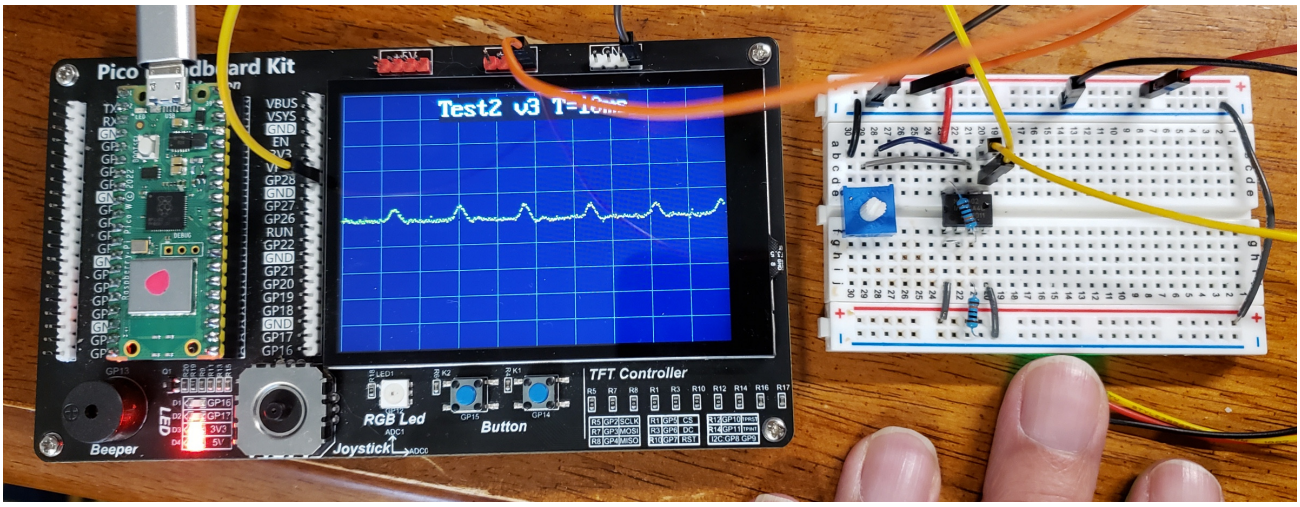
Write a python program to

- Display your heart beat in real time on the LCD display,
- Detect heartbeats, and
- Beep (or blink a light) each heartbeat

Step 1: Get the heart rate sensor to read on the Pico chip



Trying it out, no amplification was needed. Using the sample starting code, the Pico was able to read the pulse rate sensor directly (op-amp shown in photo isn't being used)



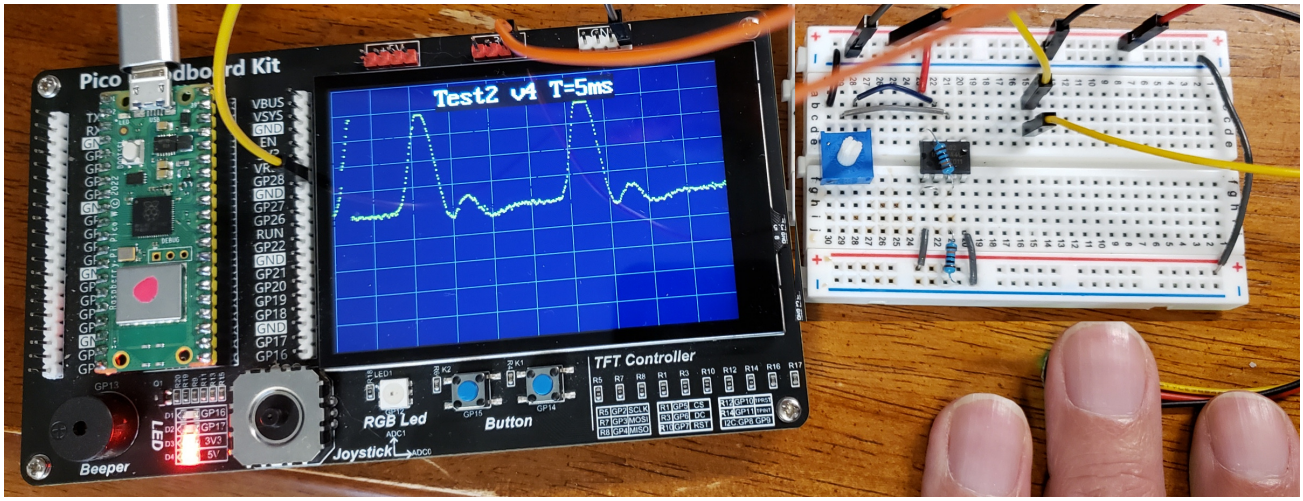
Direct connection from heart sensor to the Pico board (op-amp isn't being used)

Step 2: Change the sampling rate using timer interrupts. 5ms sampling rate was set using a timer interrupt:

```
import LCD
from machine import ADC, Pin, Timer
from time import sleep_ms

tim = Timer()
flag = 0
def tic(timer):
    global flag
    flag = 1

tim.init(period = 5, mode=Timer.PERIODIC, callback=tic)
```

Every 5ms, the flag is set. This sets the loop time in the main loop

```
while(1):
    while(flag == 0):
        pass
    flag = 0

    (etc)
```



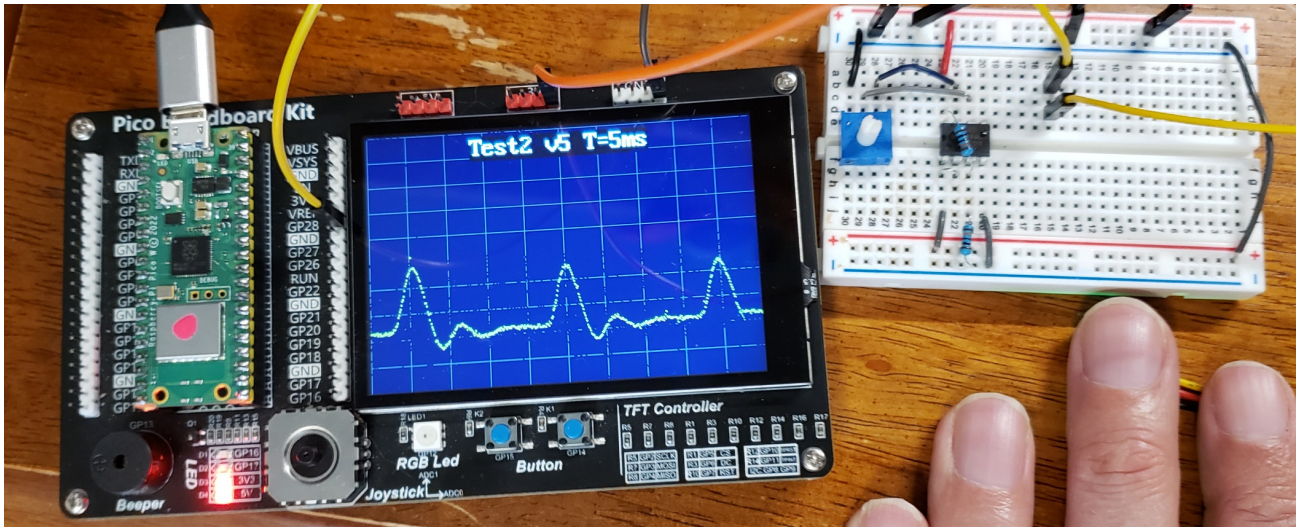
Step 3: Use edge interrupts to adjust the scaling

- GP15: scaling x2
- GP14: scaling x 1/2

```
# Edge Interrupts for x2 and /2

pin1 = Pin(15,Pin.IN,Pin.PULL_UP)
pin2 = Pin(14,Pin.IN,Pin.PULL_UP)

k = 319 / 65520

def gain_mul2(pin1):
    global k
    k = k * 2

def gain_div2(pin2):
    global k
    k = k / 2

pin1.irq(trigger=Pin.IRQ_FALLING, handler=gain_mul2)
pin2.irq(trigger=Pin.IRQ_FALLING, handler=gain_div2)
```

Comment:

- Each time GP15 is pressed, the scaling increases 2x
- Each time GP14 is pressed, the scaling decreases 2x



Pressing GP14 and GP15 changes the scaling of the graph by 2x or 1/2x respectively

Step 4: Detect heart beats

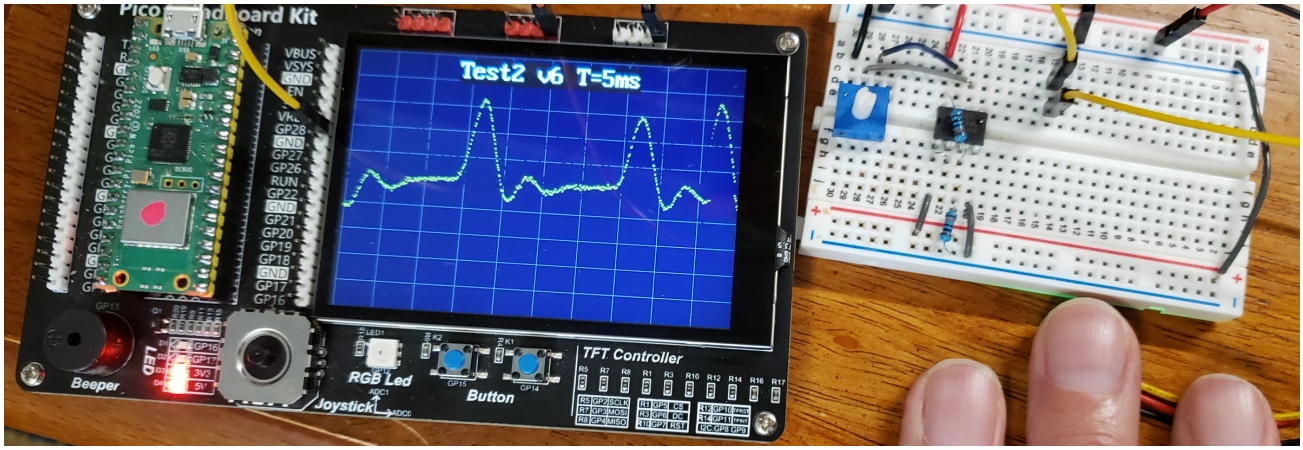
When a trace is complete, recompute, the max, the min, 30% and 70% thresholds

During a trace beep when you cross 70% after being below 30%. Use a flag (Beep_Flag) to indicate when you cross the 70% threshold (and print 'beep')

```
while(1):
    while(flag == 0):
        pass
    flag = 0

    y = a2d0.read_u16() - a0
    LCD.Line(X, 0, X, 319, Black)

    Y[X] = 165 - k*y
    if(Y[X] < Y30):
        Beep_Flag = 0
    if(Y[X] > Y70):
        if(Beep_Flag == 0):
            print('beep')
            Beep()
        Beep_Flag = 1

    LCD.Line(X, Y[X-1], X, Y[X], Yellow)

    X += 1
    if(X > 479):
        X = 0
        Ymax = Y[0]
        Ymin = Y[0]
        for i in range(1,480):
            Ymax = max(Y[i], Ymax)
            Ymin = min(Y[i], Ymin)
        Y30 = 0.3*(Ymax-Ymin)+Ymin
        Y70 = 0.7*(Ymax-Ymin)+Ymin
```

Script Window

```
beep
beep
beep
```

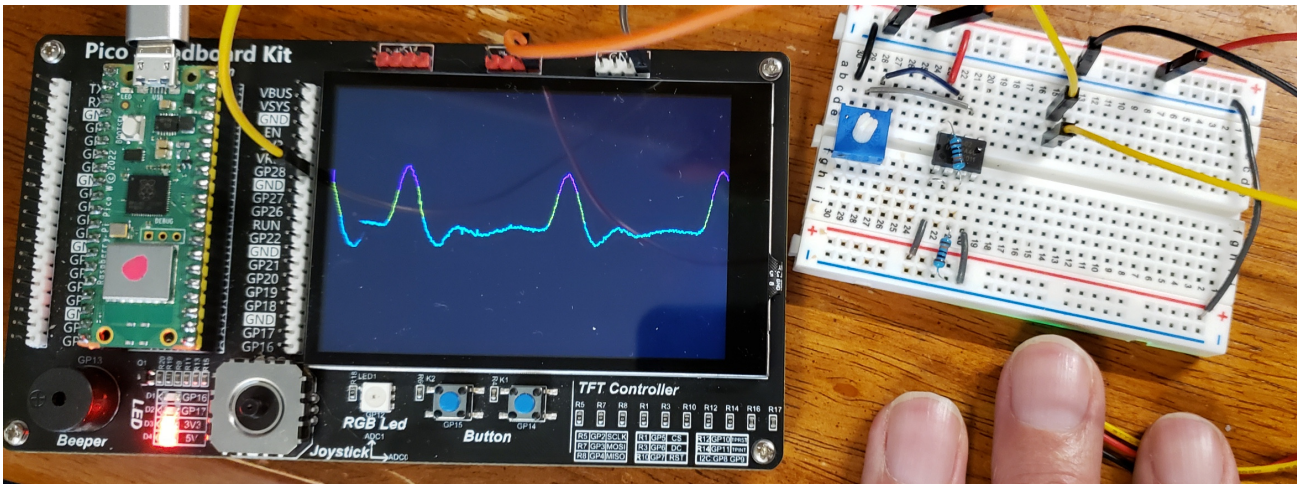


Print 'beep' each time the pulse goes above 70%

Step 5: Beep & blink a light for 50ms using an interrupt

- Beep() turns on the beeper and sets up a timer interrupt 50ms in the future
- beep_off() is an interrupt service routine that then turns off the beeper and LED

```
Beeper = Pin(13, Pin.OUT)
LED = Pin(16, Pin.OUT)
beep = Timer()

def beep_off(timer):
    Beeper.value(0)
    LED.value(0)

def Beep():
    Beeper.value(1)
    LED.value(1)
    beep.init(period = 50, mode=Timer.ONE_SHOT, callback=beep_off)
```



Final Code: Pulse shown on LCD.
Color indicates region: pink (>70%), yellow (30-70%), blue (<30%)
Light blinks and beeper sounds each time you cross 70%
Buttons GP15 and GP14 allow you to adjust the scaling (2x & 1/2x)

Fional Code:

- Red: Periodic timer interrupt. Sets the sampling rate to 5ms
- Blue: On-shot timer interrupt. Sets the duration of the beeper to 20ms
- Green: Edge interrupt. Change the scaling when you press GP14 or GP15
- Purple: Detect pulses. Beep each time you go above 70%

```
# Test #2: Sp25.  Heart Rate Sensor

import LCD
from machine import ADC, Timer, Pin
from time import sleep_ms

Beeper = Pin(13, Pin.OUT)
LED = Pin(16, Pin.OUT)

tim = Timer()
flag = 0
def tic(timer):
    global flag
    flag = 1

tim.init(period = 5, mode=Timer.PERIODIC, callback=tic)

beep = Timer()
def beep_off(timer):
    Beeper.value(0)
    LED.value(0)

def Beep():
    Beeper.value(1)
    LED.value(1)
    beep.init(period = 20, mode=Timer.ONE_SHOT, callback=beep_off)

# Edge Interrupts for x2 and /2

pin1 = Pin(15,Pin.IN,Pin.PULL_UP)
pin2 = Pin(14,Pin.IN,Pin.PULL_UP)

k = 319 / 65520

def gain_mul2(pin1):
    global k
    k = k * 2

def gain_div2(pin2):
    global k
    k = k / 2

pin1.irq(trigger=Pin.IRQ_FALLING, hanbdler=gain_mul2)
pin2.irq(trigger=Pin.IRQ_FALLING, handler=gain_div2)

a2d0 = machine.ADC(2)

Navy = LCD.RGB(0,0,10)
Yellow = LCD.RGB(150,150,0)
Cyan   = LCD.RGB(0,150,150)
Magenta   = LCD.RGB(200,0,200)
Grey = LCD.RGB(50,50,50)
White = LCD.RGB(200,200,200)
Black = LCD.RGB(0,0,0)
```

```
LCD.Init()
LCD.Clear(Black)
for i in range(0,11):
    LCD.Line(0, i*32, 479, i*32, Grey)
    LCD.Line(i*48, 0, i*48, 319, Grey)
LCD.Title('Test2 v6 T=5ms', White, Black)

Y = [0]*480

X = 0
a0 = a2d0.read_u16()

Ymax = 320
Ymin = 0
Y30 = 0.3*(Ymax-Ymin)+Ymin
Y70 = 0.7*(Ymax-Ymin)+Ymin
Beep_Flag = 0

while(1):
    while(flag == 0):
        pass
    flag = 0

    y = a2d0.read_u16() - a0
    LCD.Line(X, 0, X, 319, Black)

    Y[X] = 165 - k*y
    if(Y[X] < Y30):
        Beep_Flag = 0
    if(Y[X] > Y70):
        if(Beep_Flag == 0):
            print('beep')
            Beep()
        Beep_Flag = 1

    if(Y[X] > Y70):
        LCD.Line(X, Y[X-1], X, Y[X], Magenta)
    elif(Y[X] > Y30):
        LCD.Line(X, Y[X-1], X, Y[X], Yellow)
    else:
        LCD.Line(X, Y[X-1], X, Y[X], Cyan)

    X += 1
    if(X > 479):
        X = 0
        Ymax = Y[0]
        Ymin = Y[0]
        for i in range(1,480):
            Ymax = max(Y[i], Ymax)
            Ymin = min(Y[i], Ymin)
        Y30 = 0.3*(Ymax-Ymin)+Ymin
        Y70 = 0.7*(Ymax-Ymin)+Ymin
```