

# ECE 476/676 - Homework #3

*Binary Outputs, Binary Inputs, Serial I/O - Due Monday, September 16th*

## Binary I/O

Use the buttons, LEDs, and Beeper on the Pico Breadboard Kit:

Input:

- Button GP15 and GP14

Output:

- LED GP16, GP17
- Beeper GP13

1) Write a Python program which turns on the LEDs based upon which buttons are pressed:

- GP16: Turn on if both buttons are pressed (logic AND)
- GP17: Turn on if only one button is pressed (logic XOR)

Test your code to verify it works.

Code:

```
# Problem 1

from machine import Pin
from time import sleep

A = Pin(15, Pin.IN, Pin.PULL_UP)
B = Pin(14, Pin.IN, Pin.PULL_UP)

D16 = Pin(16, Pin.OUT)
D17 = Pin(17, Pin.OUT)
Beep = Pin(13, Pin.OUT)

while(1):
    D16.value(A.value() and B.value())
    D17.value(A.value() ^ B.value())
    print(A.value(), B.value(), D16.value(), D17.value())
    sleep(0.1)
```

Testing the code:

A	B	A and B	A xor B
1	1	1	0
0	1	0	1
1	1	1	0
1	0	0	1

2) Write a Python program which counts when you press the buttons

- The counter starts at 0
- When you press and release GP14, the counter increases by 1
- When you press and release GP15, the counter increases by 10

Verify your code works

```
# Problem 2

from machine import Pin
from time import sleep

B10 = Pin(15, Pin.IN, Pin.PULL_UP)
B1 = Pin(14, Pin.IN, Pin.PULL_UP)

D16 = Pin(16, Pin.OUT)
D17 = Pin(17, Pin.OUT)
Beep = Pin(13, Pin.OUT)

Count = 0
flag = 0

while(1):
    if(B10.value() == 0):
        Count += 10
        flag = 1
        while(B10.value() == 0):
            pass
    if(B1.value() == 0):
        Count += 1
        flag = 1
        while(B1.value() == 0):
            pass
    if(flag):
        print(Count)
        flag = 0
```

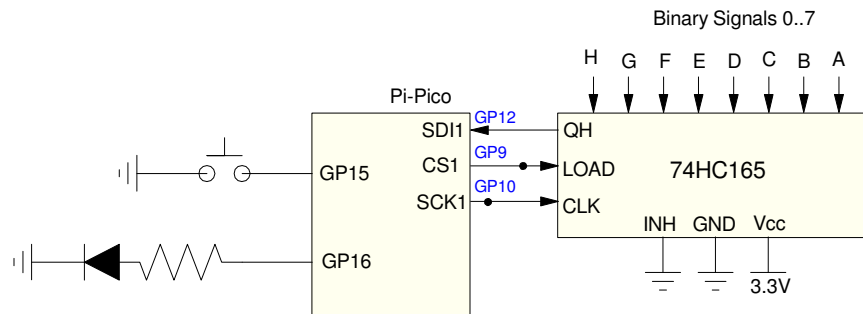
Testing the code:

- Button 14 counts by one
- Button 15 counts by ten

```
Button Count
14      1
14      2
14      3
14      4
15     14
15     24
15     34
14     35
14     36
15     46
```

## Combination Lock

Write a program where you guess the value of an 8-bit binary number, defined by a 74165 shift register



- At the start of the game, the Pico picks a random number from 0 to 255 and the LED GP16 is turned off.
- You then set the eight binary inputs to the shift register
- Once set, press GP15 button.
- The Pico then reads your guess an SPI interface.
- It then tells you if your guess was too low, too high, or correct
- If the code corectis just right, turn on the LED on GP16

3) Write a subroutine which generates a random number from 0..255

- Test your code

Code:

```
# Problem 3

from machine import Pin
from time import sleep
from random import randrange

def GetKey():
    x = randrange(256)
    return(x)

for i in range(0,5):
    key = GetKey()
    print(i, key)
```

Testing: Five trials resulted in random numbers in the range of 0..255

i	key
0	94
1	140
2	226
3	135
4	0

4) Write a subroutine which reads a 74165 shift register and returns a number from 0..255

- Test your code

Code:

```
from machine import Pin, SPI
from time import sleep, sleep_ms, sleep_us

spi = SPI(1, baudrate=10_000, polarity=1, phase=1, bits=8, sck=10, mosi=11,
miso=12)

Button = Pin(20, Pin.IN, Pin.PULL_UP)
LATCH = Pin(9, Pin.OUT)
Beeper = Pin(13, Pin.OUT)
Beeper.value(0)

def LS165():
    LATCH.value(1)
    sleep_us(10)
    LATCH.value(0)
    sleep_us(10)
    LATCH.value(1)
    # data is latched - now shift it in
    rxdata = spi.read(1, 0x42)
    return(ord(rxdata))

while(1):
    Y = LS165()
    print(Y)
    sleep(0.1)
```

Testing the code:

- Input several different numbers by moving the wires around
- Check that the reading is correct
- Also check the traces on the oscilloscope

```
0 works
255 works
1 works
128 works
```

## 5) Write a program for a combination lock program

- Test your code
- Display the actual code and your guess
- Verify your code returns the correct messages (too high, too low, correct)

```
from machine import Pin, SPI
from time import sleep, sleep_ms, sleep_us

from machine import Pin, SPI
from time import sleep, sleep_ms, sleep_us
from random import randrange

spi = SPI(1, baudrate=10_000, polarity=1, phase=1, bits=8, sck=10, mosi=11,
miso=12)

LATCH = Pin(9, Pin.OUT)
Beeper = Pin(13, Pin.OUT)
Beeper.value(0)

def LS165():
    LATCH.value(1)
    sleep_us(10)
    LATCH.value(0)
    sleep_us(10)
    LATCH.value(1)
    # data is latched - now shift it in
    rxdata = spi.read(1, 0x42)
    return(ord(rxdata))

def GetKey():
    x = randrange(256)
    return(x)

A = Pin(15, Pin.IN, Pin.PULL_UP)
B = Pin(14, Pin.IN, Pin.PULL_UP)

Key = GetKey()
flag = 0

while(flag == 0):
    print('Set the key value and press button 15')
    while(A.value() == 1):
        pass
    while(A.value() == 0):
        pass
    Guess = LS165()
    if(Guess < Key):
        print('Guess ' + str(hex(Guess)) + ' is too low')
    if(Guess > Key):
        print('Guess ' + str(hex(Guess)) + ' is too high')
    if(Guess == Key):
        print('Correct: Key = ' + str(hex(Guess)))
        flag = 1
```

Testing the code:

### Trial #1

```
Set the key value and press button 15
128 is too high
64 is too low
96 is too high
80 is too low
88 is too low
90 is too low
91 is correct
```

### Trial #2

```
Set the key value and press button 15
128 is too low
192 is too high
160 is too low
176 is too low
184 is too low
188 is correct
```

## 6) Demonstrate your code

- In-Person or with a video

