

Tampines Regional Library
LearnX Community
Pi Python Introductory Course
Course Material
By
Goh Soon Seng

**PLEASE DO NOT WRITE ANYTHING ON THE PAGES
OF THIS BOOKLET. THANK YOU.**

Pi Python Introductory Course

COURSE OBJECTIVE

- Learn the basics of Python Programming
- Foster a spirit of tinkering and making

WE ENCOURAGE

- Continuous life-long learning
- Self directed learning

OUR TEACHING PLATFORM

- The Raspberry Pi Pico W
- The Thonny IDE.

Programme

Python Lesson – Session # 1

LIBRARIES

- machine
- Time
- tm1637

VARIABLES

LOOPS

- while True:

PYTHON RULE

- Indentation
- Character casing

Python Lesson – Session # 2

FUNCTIONS

CONDITIONAL STATEMENTS

- if else

LOOPS

- while True:
- for

MESSAGING

- how humans and machine communicate with each other

Python Lesson – Session # 3

MORE PYTHON CODING

PASSWORD

Python Lesson – Session # 4

REVISION & TEST

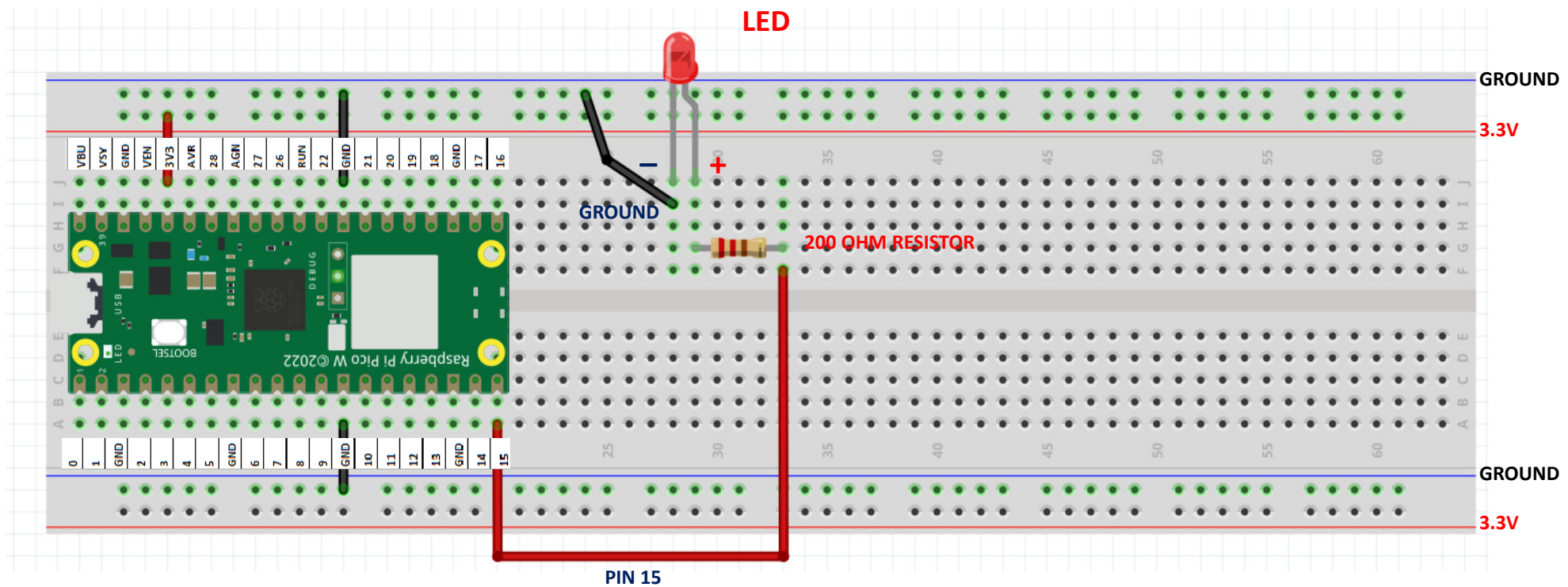
- programming the lights and siren of a patrol car

INTRODUCTION TO CHATGPT

- How to login to ChatGPT
- Getting ChatGPT to help us code
- Learning using ChatGPT

EXERCISE #1 Wiring a Light Emitting Diode (LED)

Please wire up the following circuit



TESTING OUR CIRCUIT USING THONNY SHELL

Ex 1a. Turning our Red LED on

```
>>> from machine import Pin
>>> red_led = Pin(15,Pin.OUT)
>>> red_led.on()
```

Ex 1b. Turning it off

```
>>> red_led.off()
```

Ex 1c. Blinking the red LED

Start with red led off
Wait 1 second
Turn red led on
Wait 1 second
Turn red led off

An Algorithm

Start with red led off
Wait 1 second
Turn red led on
Wait 1 second
Turn red led off

from time import sleep

Thonny - <untitled> @ 9:14
File Edit View Run Tools Help

Files ×

This computer
C: \ Pi Club \ Club Meetings 2023 \ Meeting 07 - July 02 2023
Peter
Pico
Agenda for Meeting 2 July 2023.docx
classRoboticArm.py
line_following_robot_pico_v3.py
robotwitharm.html

```
<untitled> * ×  
1 from machine import Pin  
2 from time import sleep  
3 red_led = Pin(15,Pin.OUT)  
4  
5 red_led.off()  
6 sleep(1)  
7 red_led.on()  
8 sleep(1)  
9 red_led.off()  
10
```

Save this program as ex1c.py

How to modify a program to add new features

What if I want to blink my red LED forever

```
ex1c.py x
1 from machine import Pin
2 from time import sleep
3 red_led = Pin(15,Pin.OUT)
4
5 red_led.off()
6 sleep(1)
7 red_led.on()
8 sleep(1)
9 red_led.off()
```

1

while True:

```
red_led.off()
sleep(1)
red_led.on()
sleep(1)
red_led.off()
```

2

Thonny - Raspberry Pi Pico :: /ex1c.py @ 11:1

File Edit View Run Tools Help

Files x

This computer

- C: \ Pi Club \ Club Meetings 2023 \ Meeting 07 - July 02 2023
- Peter
- Pico
- Agenda for Meeting 2 July 2023.docx
- classRoboticArm.py
- line_following_robot_pico_v3.py
- robotwitharm.html

Raspberry Pi Pico

- ex1c.py

[ex1c.py] * x

```
1 from machine import Pin
2 from time import sleep
3 red_led = Pin(15,Pin.OUT)
4
5 while True:
6     red_led.off()
7     sleep(1)
8     red_led.on()
9     sleep(1)
10    red_led.off()
11
```

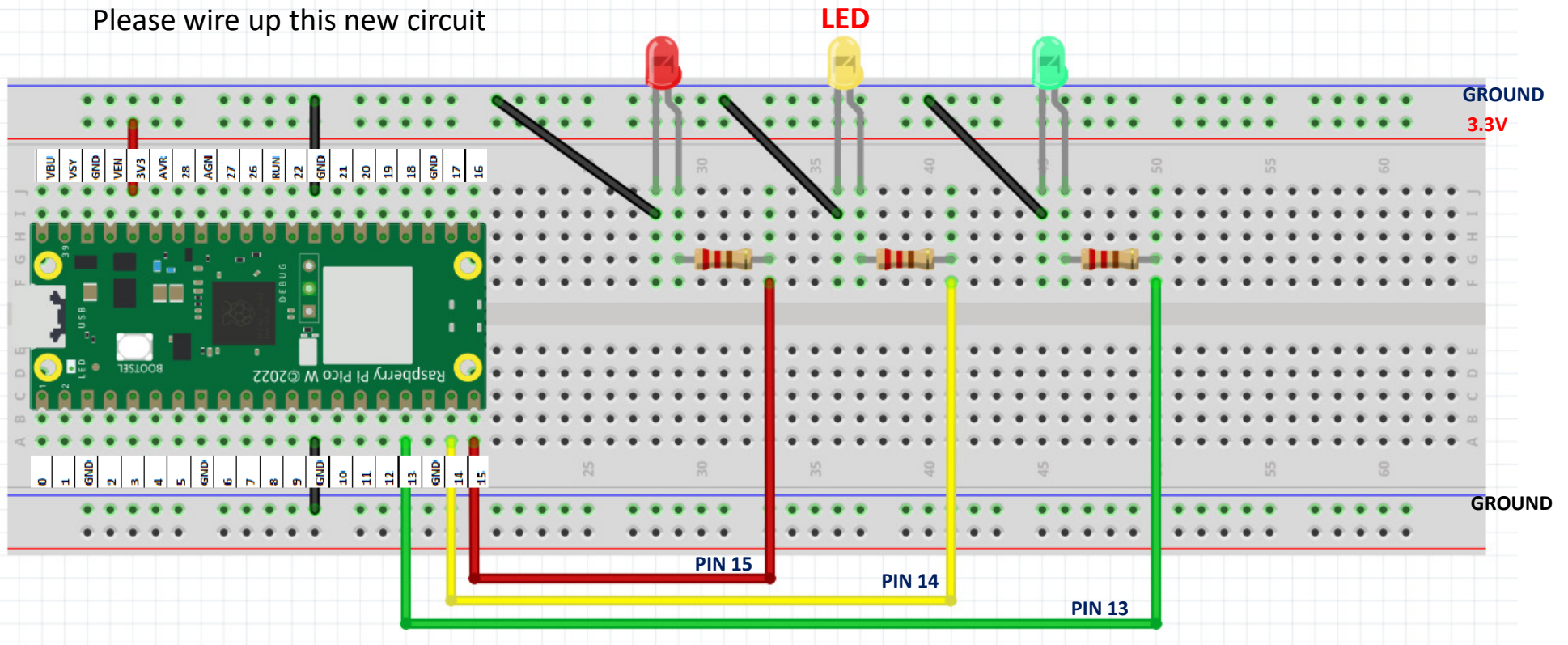
These spaces are required
And are called indentations

Now save this new program as
ex1d.py

3

EXERCISE #2 – Traffic Light

Please wire up this new circuit



EXERCISE 2 - TESTING OUR CIRCUIT

Ex 2a. Traffic Light

```
>>> from machine import Pin
>>> red_led= Pin(15,Pin.OUT)
>>> yellow_led = Pin(14,Pin.OUT)
>>> green_led=Pin(13,Pin.OUT)
```

Test the LEDs. Turn each one of them on and off

```
>>>
```

Ex 2b. Using the Example Code below, create a program for a Traffic Light System

```
from machine import Pin
from time import sleep
red_led = Pin(15,Pin.OUT)

red_led.off()
sleep(1)
red_led.on()
sleep(1)
red_led.off()
```

Modify to this

How does a traffic light work?
Green Light On
Give cars 10 seconds to cross
Green Light Off
Yellow Light On
Give 5 seconds warning
Yellow Light Off
Red Light On
Wait 10 seconds
Red Light Off

To this

A New program
implementing
the new Algorithm

EXERCISE #2b – How to program a traffic light system

Lets create this program together
Before we do this, **TEST** your circuit
Fill in the blanks and test.

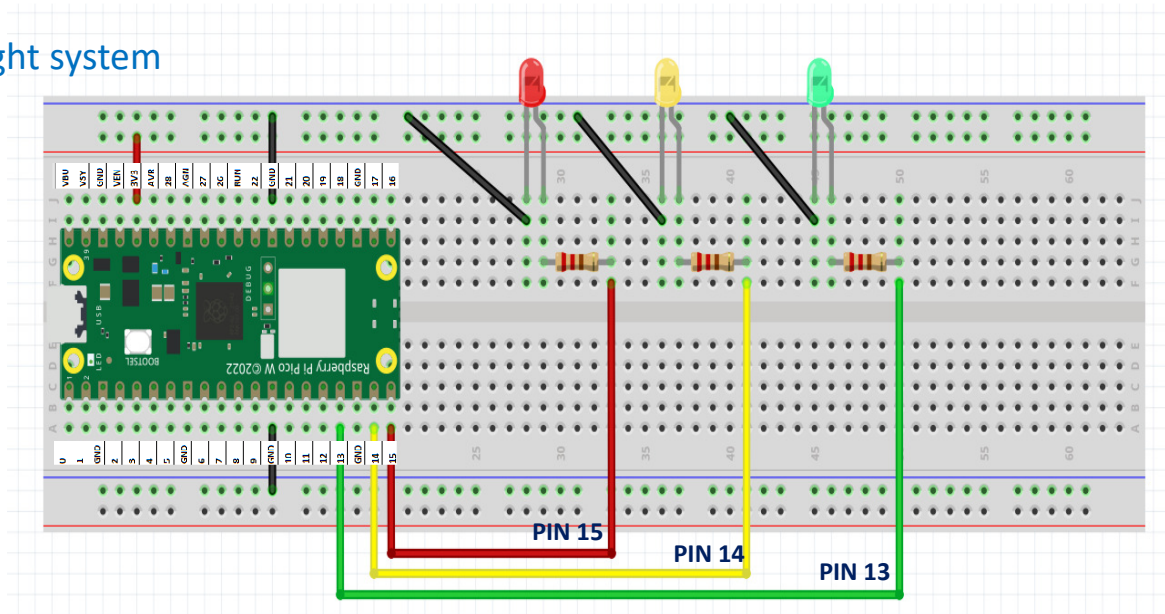
```
>>> from machine import Pin
>>> red_led = Pin(15,Pin.OUT)
>>> yellow_led=Pin( ..... )
>>> green_led=Pin( ..... )
>>> red_led.on()
>>>
```

Do this on your own

Turn off/on the RED LED

Turn on/off the YELLOW LED

Turn on/off the GREEN LED



How does a traffic light work?
Green Light On
Give cars 10 seconds to cross
Green Light Off
Yellow Light On
Give 5 seconds warning
Yellow Light Off
Red Light On
Wait 10 seconds
Red Light Off

OPEN THONNY

START A NEW PROGRAM

```
from machine import Pin  
from time import sleep  
red_led = Pin(15,Pin.OUT)
```

SAVE YOUR PROGRAM AS ex2c.py

HOW TO MAKE IT RUN FOREVER

```

1 from machine import Pin
2 from time import sleep
3 red_led=Pin(15,Pin.OUT)
4 yellow_led=Pin(14,Pin.OUT)
5 green_led=Pin(13,Pin.OUT)
6
7
8 green_led.on()
9 sleep(10)
10 green_led.off()
11 yellow_led.on()
12 sleep(5)
13 yellow_led.off()
14 red_led.on()
15 sleep(10)
16 red_led.off()

```

Library

Setup

Algorithm

Runs once

```

ex2c.py
1 from machine import Pin
2 from time import sleep
3 red_led=Pin(15,Pin.OUT)
4 yellow_led=Pin(14,Pin.OUT)
5 green_led=Pin(13,Pin.OUT)
6
7 while True:
8     green_led.on()
9     sleep(10)
10    green_led.off()
11    yellow_led.on()
12    sleep(5)
13    yellow_led.off()
14    red_led.on()
15    sleep(10)
16    red_led.off()

```

While True
make this group
of codes
run forever

INDENTATION

FUNCTIONS – GROUP A SET OF CODES AND ASSIGN IT A NAME

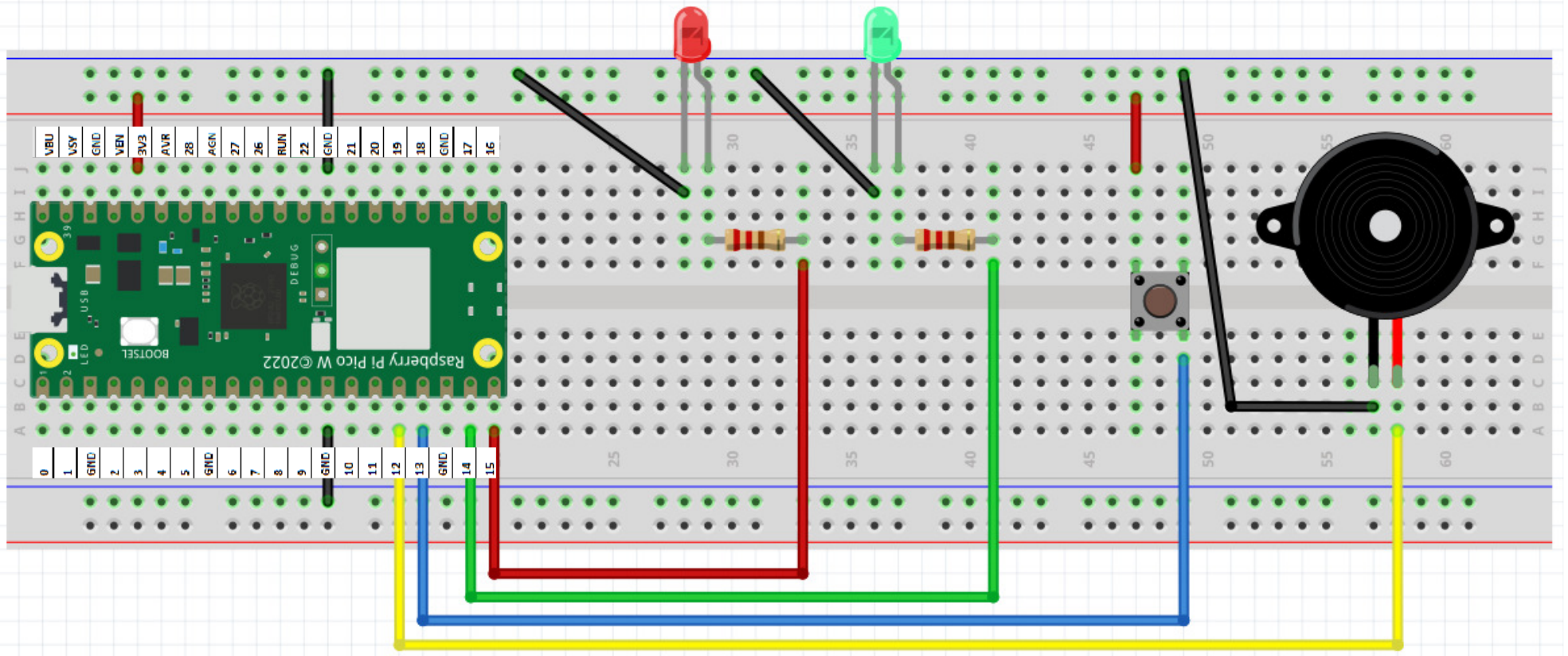
```
<untitled> * x
1 from machine import Pin
2 from time import sleep
3 red_led=Pin(15,Pin.OUT)
4 yellow_led=Pin(14,Pin.OUT)
5 green_led=Pin(13,Pin.OUT)
6
7 while True:
8     green_led.on()
9     sleep(10)
10    green_led.off()
11    yellow_led.on()
12    sleep(5)
13    yellow_led.off()
14    red_led.on()
15    sleep(10)
16    red_led.off()
17
```

```
<untitled> * x
1 from machine import Pin
2 from time import sleep
3 red_led=Pin(15,Pin.OUT)
4 yellow_led=Pin(14,Pin.OUT)
5 green_led=Pin(13,Pin.OUT)
6
7 def trafficlight():
8     green_led.on()
9     sleep(10)
10    green_led.off()
11    yellow_led.on()
12    sleep(5)
13    yellow_led.off()
14    red_led.on()
15    sleep(10)
16    red_led.off()
17
18
19 while True:
20     trafficlight()
21
```

FUNCTION

EXERCISE #3 – Pedestrian Crossing

Please wire up this new circuit



TESTING OUR CIRCUIT USING THONNY SHELL

Ex 3a. Turning our Buzzer on

```
>>> from machine import Pin
>>> buzz= Pin(12,Pin.OUT)
>>> buzz.on()
```

Ex 3b. Turning it off

```
>>> buzz.off()
```

Ex 3c. Test the Red and Green LED. Make sure they can be turned on and off

```
>>>
```

HOW DOES A PEDESTRIAN CROSSING WORK?

STARTING POINT – ONLY THE RED MAN IS ON

IT WAITS FOR SOMEONE TO MAKE A REQUEST (BUTTON)

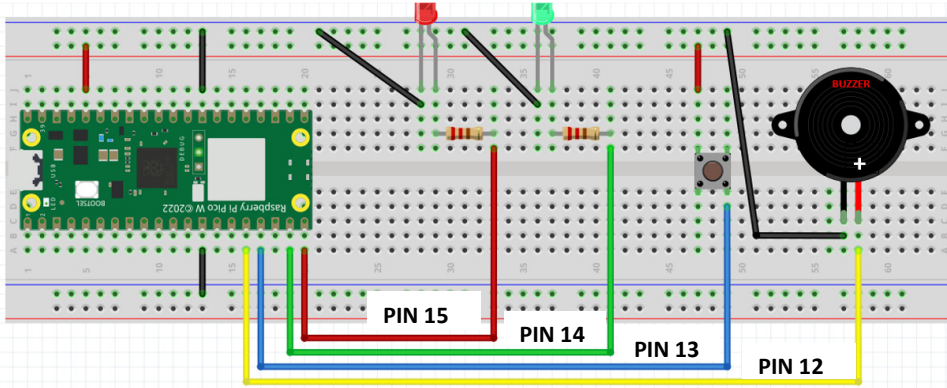
WHEN THE BUTTON IS PRESSED

THE PEDESTRIAN CROSSING SEQUENCE KICKS IN

WAIT FOR 10 SECONDS
RED LED GOES OFF
GREEN LED GOES ON
GIVE 10 SECONDS FOR PEOPLE TO CROSS
GREEN LED GOES OFF
RED LED TURNS ON



Let's do it together



WAIT FOR 10 SECONDS
RED LED GOES OFF
GREEN LED GOES ON
GIVE 10 SECONDS FOR PEOPLE TO CROSS
GREEN LED GOES OFF
RED LED TURNS ON



Save this code as ex3d.py and run it

```
from machine import Pin
from time import sleep
red_led = Pin(15,Pin.OUT)
green_led=Pin(14,Pin.OUT)
buzzer=Pin(12,Pin.OUT)
button=Pin(13,Pin.IN,Pin.PULL_DOWN)
```

```
red_led.on()
green_led.off()
```

```
while True:
    sleep(10)
    red_led.off()
    green_led.on()
    sleep(10)
    green_led.off()
    red_led.on()
```

ex3d.py

```
from machine import Pin
from time import sleep
red_led = Pin(15,Pin.OUT)
green_led=Pin(14,Pin.OUT)
buzzer=Pin(12,Pin.OUT)
button=Pin(13,Pin.IN,Pin.PULL_DOWN)
```

```
red_led.on()
green_led.off()
```

```
while True:
```

```
    sleep(10)
    red_led.off()
    green_led.on()
    sleep(10)
    green_led.off()
    red_led.on()
```

How to convert this into a function



```
1 from machine import Pin
2 from time import sleep
3 red_led=Pin(15,Pin.OUT)
4 yellow_led=Pin(14,Pin.OUT)
5 green_led=Pin(13,Pin.OUT)
6
7 def trafficlight():
8     green_led.on()
9     sleep(10)
10    green_led.off()
11    yellow_led.on()
12    sleep(5)
13    yellow_led.off()
14    red_led.on()
15    sleep(10)
16    red_led.off()
17
18
19 while True:
20     trafficlight()
```

USE THIS
EXAMPLE

Give a name for
your function.
Let's call it
greenman()

Save your
program as
ex3e.py and
Run it

```

[ ex3f.py ] * ×
1  from machine import Pin
2  from time import sleep
3
4  red_led=Pin(15,Pin.OUT)
5  green_led=Pin(14,Pin.OUT)
6  buzzer=Pin(12,Pin.OUT)
7  button=Pin(13,Pin.IN,Pin.PULL_DOWN)
8
9
10
11 def blinkblink():
12     for x in range(1,11,1):
13         green_led.on()
14         buzzer.on()
15         sleep(0.5)
16         green_led.off()
17         buzzer.off()
18         sleep(0.5)
19
20 def greenman():
21     sleep(10)
22     red_led.off()
23     green_led.on()
24     sleep(10)
25     blinkblink()
26     green_led.off()
27     red_led.on()
28
29 red_led.on()
30 green_led.off()
31 while True:
32     greenman()
33

```

WAIT FOR 10 SECONDS
 RED LED GOES OFF
 GREEN LED GOES ON
 GIVE 10 SECONDS FOR PEOPLE TO CROSS

GREEN LED STARTS TO BLINK 10 TIMES (10 SECONDS)
BUZZER ALSO BLINKS 10 TIMES (10 SECONDS)

GREEN LED GOES OFF
 RED LED TURNS ON

```

def blinkblink():
    for x in range(1,11,1):
        green_led.on()
        buzzer.on()
        sleep(0.5)
        green_led.off()
        buzzer.off()
        sleep(0.5)

```

10 X

```
[ex3f.py] * ×
1 from machine import Pin
2 from time import sleep
3
4 red_led=Pin(15,Pin.OUT)
5 green_led=Pin(14,Pin.OUT)
6 buzzer=Pin(12,Pin.OUT)
7 button=Pin(13,Pin.IN,Pin.PULL_DOWN)
8
9
10
11 def blinkblink():
12     for x in range(1,11,1):
13         green_led.on()
14         buzzer.on()
15         sleep(0.5)
16         green_led.off()
17         buzzer.off()
18         sleep(0.5)
19
20 def greenman():
21     sleep(10)
22     red_led.off()
23     green_led.on()
24     sleep(10)
25     blinkblink()
26     green_led.off()
27     red_led.on()
28
29 red_led.on()
30 green_led.off()
31 while True:
32     greenman()
33
```

This program works fine.
But in the real world, it's not practical.
Why?

If no one want to cross, the program will continue to work

We want this program to run only
when someone presses the button

```
while True:
    greenman()
```



```
while True:
    if button.value()==1:
        greenman()
```

Button saves the day 😊

```
while True:  
    greenman()
```



```
while True:  
    if button.value()==1:  
        greenman()
```

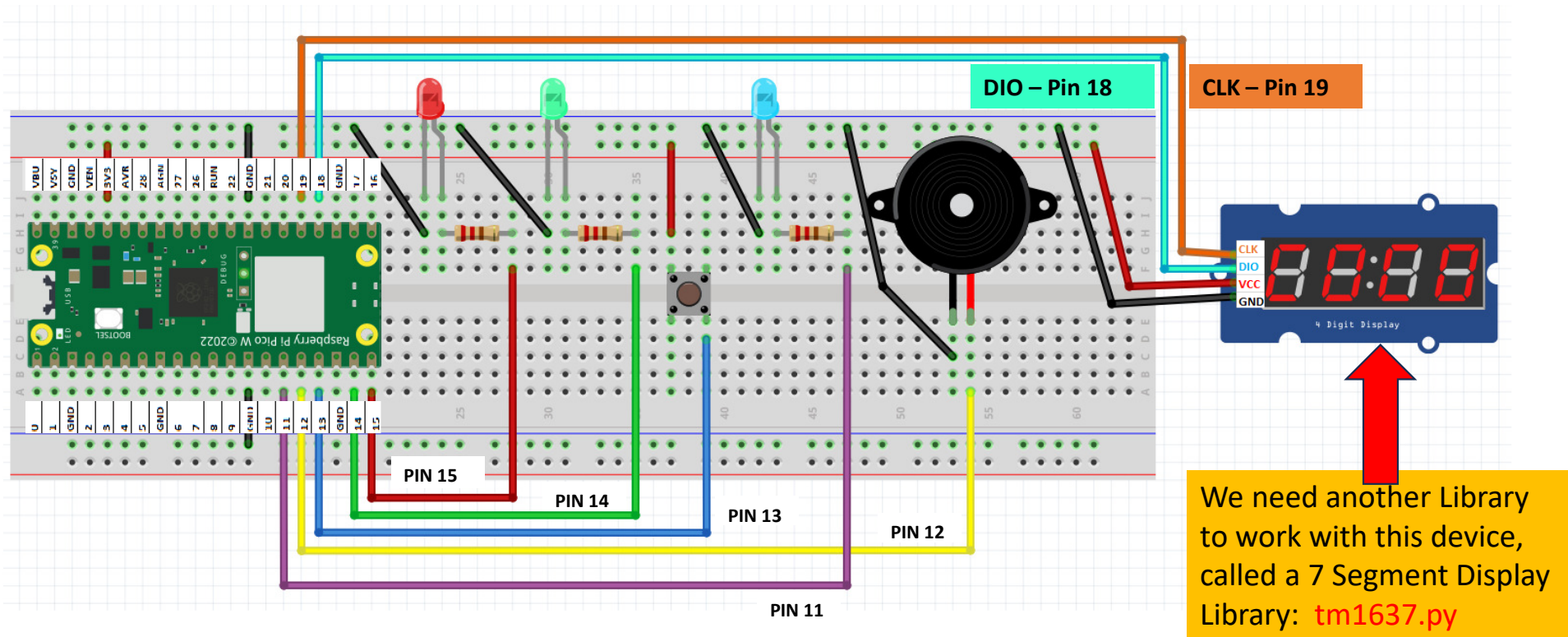
greenman() will only execute
When the button is pressed

```
[ ex3h.py ] * ×  
1  from machine import Pin  
2  from time import sleep  
3  
4  red_led=Pin(15,Pin.OUT)  
5  green_led=Pin(14,Pin.OUT)  
6  buzzer=Pin(12,Pin.OUT)  
7  button=Pin(13,Pin.IN,Pin.PULL_DOWN)  
8  
9  def blinkblink():  
10     for x in range(1,11,1):  
11         green_led.on()  
12         buzzer.on()  
13         sleep(0.5)  
14         green_led.off()  
15         buzzer.off()  
16         sleep(0.5)  
17  
18  def greenman():  
19     sleep(10)  
20     red_led.off()  
21     green_led.on()  
22     sleep(10)  
23     blinkblink()  
24     green_led.off()  
25     red_led.on()  
26  
27  red_led.on()  
28  green_led.off()  
29  while True:  
30      if button.value()==1:  
31          greenman()  
32
```

Amend program here and
save it as ex3h.py and RUN

EXERCISE# 4 – Full Fledge Pedestrian Crossing

Please wire up this circuit



EXERCISE 4a – THE 7 SEGMENT DISPLAY


```
>>> from machine import Pin
>>> import tm1637
>>> tm = tm1637.TM1637(clk=Pin(19), dio=Pin(18))
```

```
>>> tm.show("help")
```

```
>>> tm.number(1234)
```

```
>>> tm.temperature(24)
```

```
>>> tm.show(" " * 4)
```



In python " " * 4 means 4 SPACES
This will clear the display.

Make sure there is a SPACE in between the quotation marks

Exercise 4a. Display the following
PICO
95
70 Degrees Celcius
Clear the display

COUNT DOWN DISPLAY

```
from machine import Pin
from time import sleep
import tm1637
tm = tm1637.TM1637(clk=Pin(19), dio=Pin(18))
for x in range(10,-1,-1):
    tm.number(x)
    sleep(1)
```

Save this program as ex4b.py and RUN it.

Try Counting it from 0 to 20.

Use this Example

```
for x in range(1,11,1):
    green_led.on()
    buzzer.on()
    sleep(0.5)
    green_led.off()
    buzzer.off()
    sleep(0.5)
```


ADDING COUNTER
TO PEDESTRIAN
CROSSING
blinkblink()

Besides the green man
flashing and the buzzer
beeping
the counter will show
how much time left
for crossing

```
[ex3h.py] x
1 from machine import Pin
2 from time import sleep
3 import tm1637
4 red_led=Pin(15,Pin.OUT)
5 green_led=Pin(14,Pin.OUT)
6 buzzer=Pin(12,Pin.OUT)
7 button=Pin(13,Pin.IN,Pin.PULL_DOWN)
8 tm = tm1637.TM1637(clk=Pin(19), dio=Pin(18))
9 def blinkblink():
10     for x in range(10,-1,-1):
11         tm.number(x)
12         green_led.on()
13         buzzer.on()
14         sleep(0.5)
15         green_led.off()
16         buzzer.off()
17         sleep(0.5)
18 def greenman():
19     sleep(10)
20     red_led.off()
21     green_led.on()
22     sleep(10)
23     blinkblink()
24     green_led.off()
25     red_led.on()
26
27 red_led.on()
28 green_led.off()
29 while True:
30     if button.value()==1:
31         greenman()
```

Save this amended program as
ex4c.py and RUN

```

[ ex3j.py ] x
1  from machine import Pin
2  from time import sleep
3  import tm1637
4  red_led=Pin(15,Pin.OUT)
5  green_led=Pin(14,Pin.OUT)
6  buzzer=Pin(12,Pin.OUT)
7  button=Pin(13,Pin.IN,Pin.PULL_DOWN)
8  tm = tm1637.TM1637(clk=Pin(19), dio=Pin(18))
9  antispam_led=Pin(11,Pin.OUT)
10
11 def blinkblink():
12     for x in range(10,-1,-1):
13         tm.number(x)
14         green_led.on()
15         buzzer.on()
16         sleep(0.5)
17         green_led.off()
18         buzzer.off()
19         sleep(0.5)
20
21 def check():
22     if antispam_led.value() == 1:
23         pass
24     else:
25         antispam_led.on()
26         greenman()
27

```

```

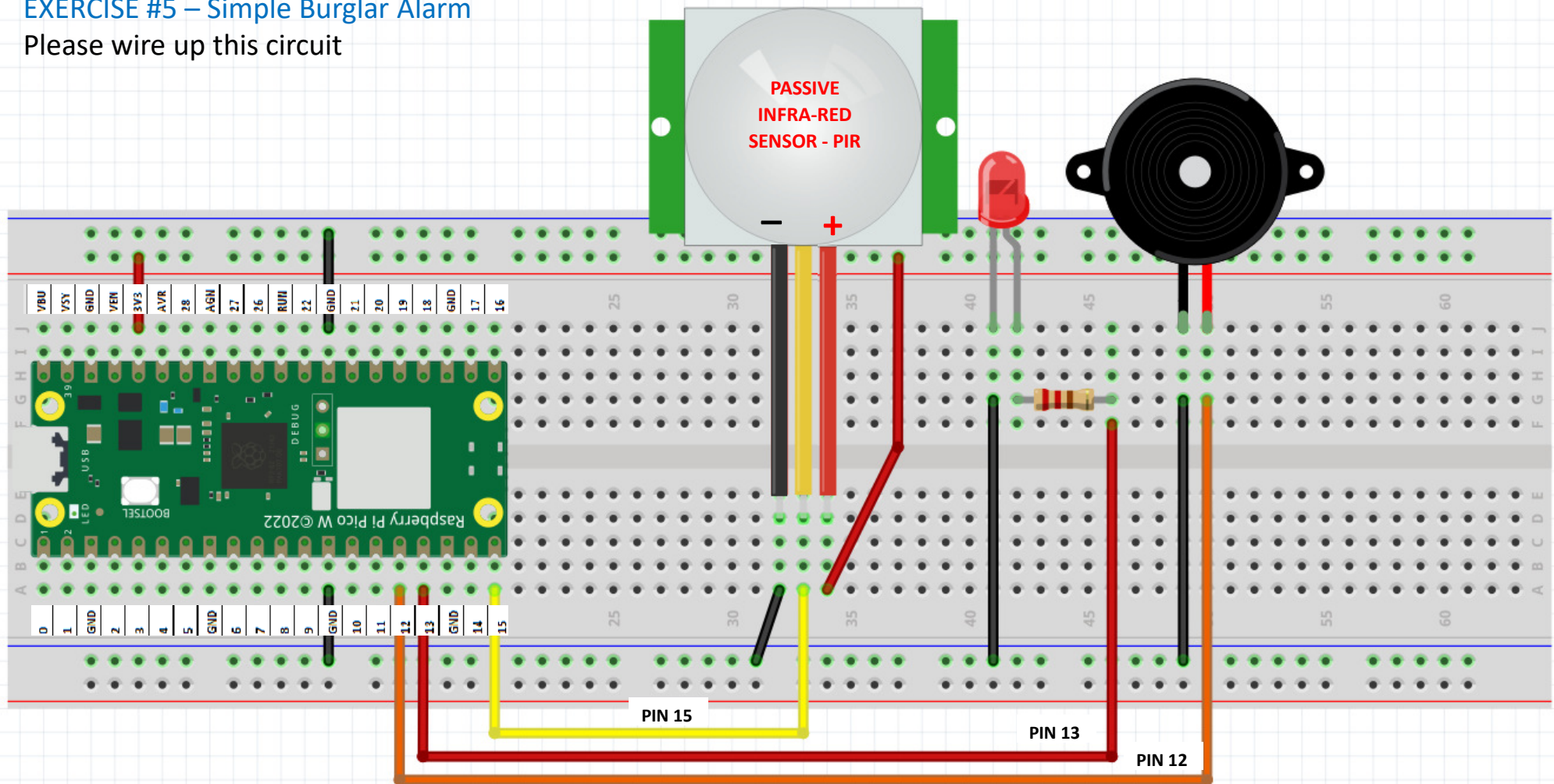
28 def greenman():
29     sleep(10)
30     red_led.off()
31     green_led.on()
32     sleep(10)
33     blinkblink()
34     green_led.off()
35     red_led.on()
36     antispam_led.off()
37
38 red_led.on()
39 green_led.off()
40 antispam_led.off()
41 while True:
42     if button.value()==1:
43         check()
44

```

Save this program as
ex4d.py and RUN

EXERCISE #5 – Simple Burglar Alarm

Please wire up this circuit

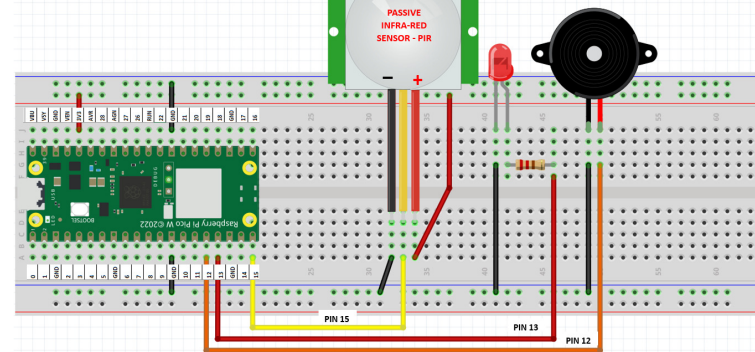


HOW DOES A PIR WORK

- In summary, the HC-SR501 PIR sensor is a device that can detect motion by sensing changes in the infrared heat patterns around it.
 - It's like having a pair of eyes that can "see" heat, and when it detects movement, it can trigger other devices or actions based on that detection.
- Actually, it works like a Button
- How a basic Burglar Alarm works
 - Let's look at the diagram

- We set the Burglar Alarm On
 - at Night (ARM)
- We turn it off in the morning (DISARM)
- To DISARM a password is needed

EXERCISE #5 – Simple Burglar Alarm
Please wire up this circuit



CODE EXAMPLE

```

18 def greenman():
19     sleep(10)
20     red_led.off()
21     green_led.on()
22     sleep(10)
23     blinkblink()
24     green_led.off()
25     red_led.on()
26
27 red_led.on()
28 green_led.off()
29 while True:
30     if button.value()==1:
31         greenman()

```

```

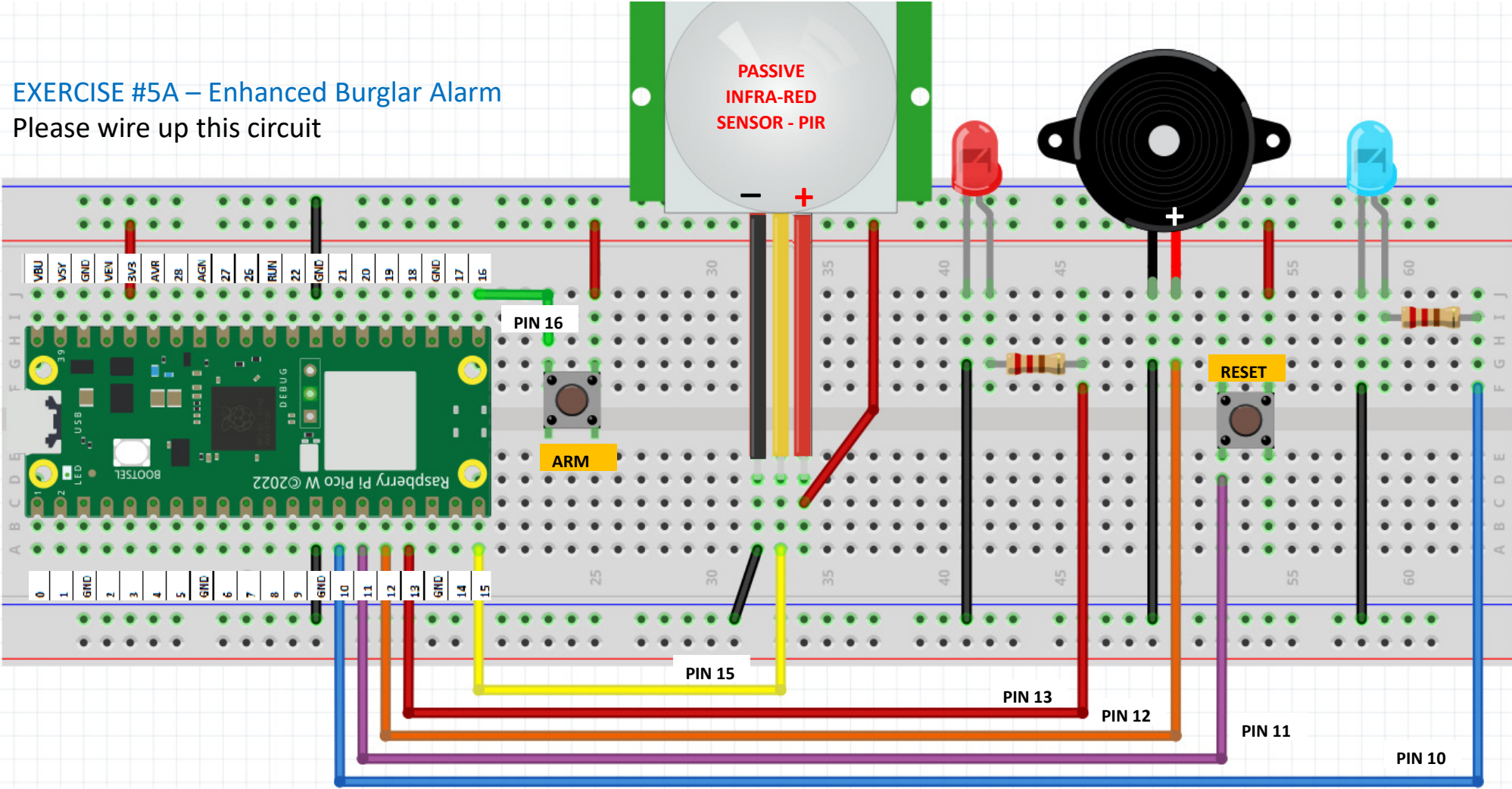
[ ex5a.py ] * ×
1  from machine import Pin
2  from time import sleep
3  sensor_pir = Pin(15,Pin.IN)
4  led = Pin(13,Pin.OUT)
5  buzz = Pin(12,Pin.OUT)
6
7  def alarm():
8      led.on()
9      buzz.on()
10     sleep(.5)
11     led.off()
12     buzz.off()
13     sleep(.5)
14
15 while True:
16     if sensor_pir.value()==1:
17         alarm()
18

```

Save this as ex5a.py and RUN

EXERCISE #5A – Enhanced Burglar Alarm

Please wire up this circuit



CODE FOR THE ENHANCED BURGLAR ALARM

```
ex5b.py x
1 from machine import Pin
2 from time import sleep
3 sensor_pir = Pin(15,Pin.IN)
4 led = Pin(13,Pin.OUT)
5 buzz = Pin(12,Pin.OUT)
6
7 arm_button=Pin(16,Pin.IN, Pin.PULL_DOWN)
8 armed_led = Pin(10,Pin.OUT)
9 disarm button = Pin(11,Pin.IN, Pin.PULL_DOWN)
10 password='12345'
11
12 def alarm():
13     for x in range(1,11,1):
14         led.on()
15         buzz.on()
16         sleep(.5)
17         led.off()
18         buzz.off()
19         sleep(.5)
```

Save program as ex5b.py and RUN

```
22 def armburglaralarm():
23     if armed_led.value() == 0:
24         armed_led.on()
25     else:
26         pass
27
28 def disarmburglaralarm():
29     if armed_led.value() == 0:
30         pass
31     else:
32         pwd=input('Enter Disarm Password ' )
33         if pwd == password:
34             armed_led.off()
35         else:
36             pass
37
40 armed_led.off()
41 buzz.off()
42 while True:
43     if arm_button.value() == 1:
44         armburglaralarm()
45     if disarm_button.value() ==1:
46         disarmburglaralarm()
47     if sensor_pir.value()==1 and armed_led.value()==1:
48         print('detected')
49         alarm()
50
```

TEST – Building the Flashing Lights and Sound of a Patrol Car

Putting knowledge gained to use

Use your blue , red LED and a Buzzer to make a noisy flashing lights on a patrol car

Choose any pin you wish

Wire up the circuit and write the python code to make it work



HINT

Material

- 1 x Red LED
- 1 x Blue LED
- 1 x Buzzer

Python Code

machine and time library
while True:
Red LED on and off
Blue LED on and off
Sleep in between on and off

```
1 from machine import Pin
2 from time import sleep
3 red_led=Pin(15,Pin.OUT)
4 yellow_led=Pin(14,Pin.OUT)
5 green_led=Pin(13,Pin.OUT)
6
7 def trafficlight():
8     green_led.on()
9     sleep(10)
10    green_led.off()
11    yellow_led.on()
12    sleep(5)
13    yellow_led.off()
14    red_led.on()
15    sleep(10)
16    red_led.off()
17
18
19 while True:
20     trafficlight()
```

USE THIS
EXAMPLE

Save your program
As **patrolcar.py**
And RUN

```
[ patrolcar.py ] ×
1  from machine import Pin
2  from time import sleep
3  red_led=Pin(17,Pin.OUT)
4  blue_led=Pin(11,Pin.OUT)
5  buzz=Pin(12,Pin.OUT)
6
7  def flashing():
8      red_led.on()
9      buzz.on()
10     sleep(.3)
11
12     red_led.off()
13     buzz.off()
14
15     blue_led.on()
16     buzz.on()
17     sleep(.3)
18
19     blue_led.off()
20     buzz.off()
21
22 while True:
23     flashing()
24
```


Send a message



Red LED Pin 13, Blue LED Pin 10, Buzzer Pin 12. write a micropython program for the flashing lights of a patrol car with buzzer blazing

use from machine import Pin. Do not use toggle

the lights are flashing together and not alternating

flash faster