CamJam EduKit Sensors Worksheet Six

Project Intruder Alarm

Description In this project, you will use the passive inferred sensor circuit and include lights and sound to create an alarm.

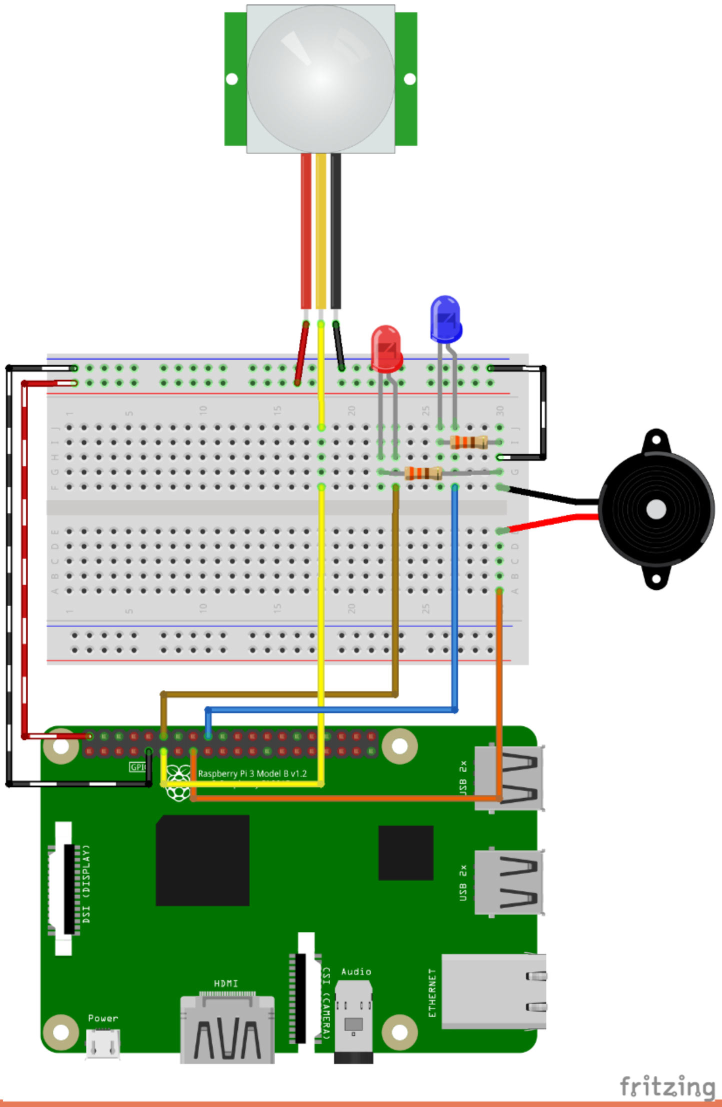
Equipment Required

- Your Raspberry Pi
- 400 point breadboard
- Passive Infrared Sensor (PIR)
- 2 x 330Ω resistors
- 9 x M/F jumper wires
- 1 x M/M jumper wires
- 1 x red LED
- 1 x blue LED
- 1 x buzzer

Building the Circuit

The Alarm circuit combines the PIR circuit and the LED/Buzzer circuit from Worksheet Two. You will be using this to create a simple movement alarm.

Build the circuit as shown.



Code

The code is based on the code in Worksheet Five, so you are going to copy that code instead of retyping it all. Start the Thonny editor and open the file `5-movement.py`. Save it immediately as `6-alarm.py`.

Edit the code so that it looks like the following.

```
# CamJam EduKit 2 - Sensors
# Worksheet 6 - Alarm

# Import Python header files
import time
from gpiozero import MotionSensor, LED, Buzzer

# Set a variable to hold the GPIO Pin identity
pir = MotionSensor(17)
red = LED(18)
blue = LED(24)
buzzer = Buzzer(22)

print("Waiting for PIR to settle")
pir.wait_for_no_motion()

print("PIR Module Test (CTRL-C to exit)")

def sound_alarm():
    # Flash the LEDs and sound buzzer three times
    for x in range(0, 3):
        buzzer.on()
        red.on()
        time.sleep(0.2)
        red.off()
        blue.on()
        time.sleep(0.2)
        blue.off()
        buzzer.off()
        time.sleep(0.2)

try:
    # Variables to hold the current and last states
    current_state = False
    previous_state = False

    # Loop until users quits with CTRL-C
    while True:
        # Read PIR state
        current_state = pir.motion_detected

        # If the PIR is triggered
        if current_state is True and previous_state is False:
            print(" Motion detected!")
            sound_alarm()

            # Record previous state
            previous_state = True
        # If the PIR has returned to ready state
        elif current_state is False and previous_state is True:
            print(" No Motion")
            previous_state = False

        # Wait for 10 milliseconds
        time.sleep(0.01)

except KeyboardInterrupt:
    print(" Quit")
```

Save the file as `6-alarm.py`.

Running the Code

Select the Run Module menu option, under the Run menu item. Alternatively, you can just press the F5 key.

When the PIR detects movement, it will flash the LEDs and sound the buzzer three times.

Challenge

Alter the circuit and code so that the alarm is only active when it is dark, by using the LDR.