

## CamJam EduKit Sensors Worksheet Six

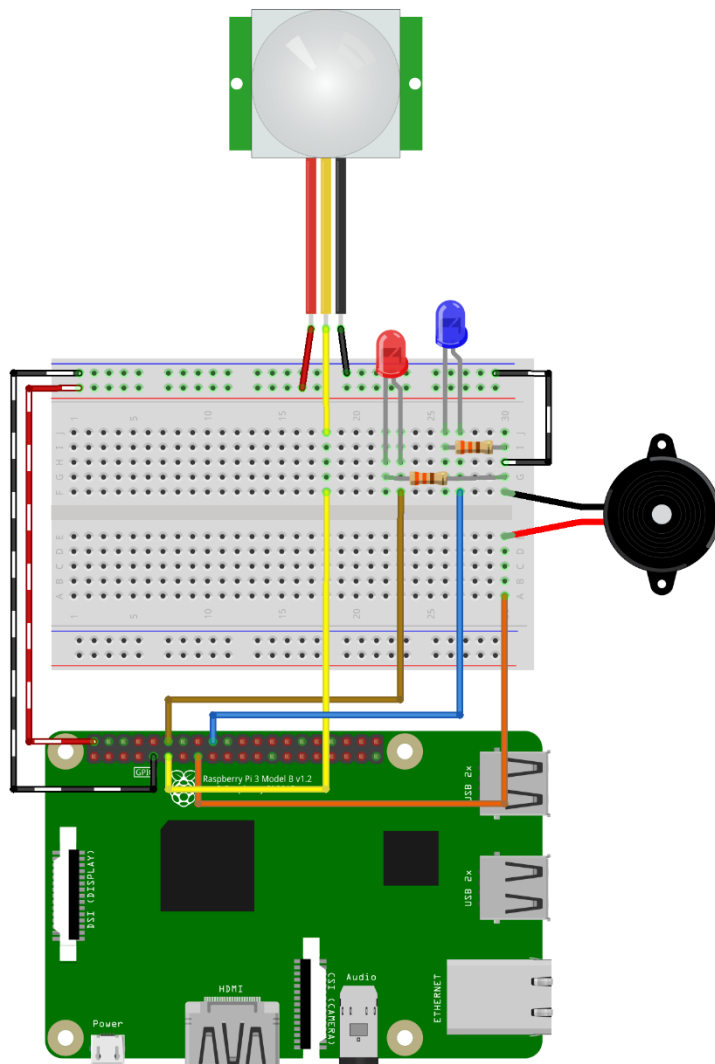
**Project** Intruder Alarm

**Description** In this project, you will use the passive infrared sensor circuit to include lights and sound.

### Equipment Required

- Your Raspberry Pi
- 400 Point Breadboard
- Passive Infrared Sensor
- 2 x 330  $\Omega$  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.

fritzing

## 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 IDLE3 editor and open the file 5-PIR.py from the EduKitSensors directory. Save it immediately as 6-Alarm.py.

Edit the code so that it looks like the following.

```
# CamJam EduKit 2 - Sensors (GPIO Zero)
# Worksheet 6 - Alarm

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

# 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)")

# Variables to hold the current and last states
currentstate = False
previousstate = False

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

        # If the PIR is triggered
        if currentstate == True and previousstate == False:
            print("    Motion detected!")
            # Flash lights 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)

            # Record previous state
            previousstate = True
```

```
# If the PIR has returned to ready state
elif currentstate == False and previousstate == True:
    print("    No Motion")
    previousstate = 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.