CS23E01 / EMBEDDED SYSTEMS AND INTERNET OF THINGS

- Session: July-Nov 2026 (B.E. CSE – VII sem)

Activities for Lab-9:

Topics: Using sensors, relay with Pi, using python libraries RPI.GPIO or PIGPIO, try Thonny IDE

****

1. Interfacing with sensors: Understand how sensors like Temperature, PIR or others work and how to inferface them with Pi. (Refer to the examples, references shared).

2. Demonstrate switching on and off an LED using Switch/button connected on the breadboard(You may just use some wires to simulate switch operation in case if switch is not available) alternatively through a keyboard button press.

3. Use PIR sensor to detect any motion, and light an LED.

4. Read data from any sensor and write it to a text file.

5. Using a relay operate a mini fan.

****

Some WEB references/installations as necessary

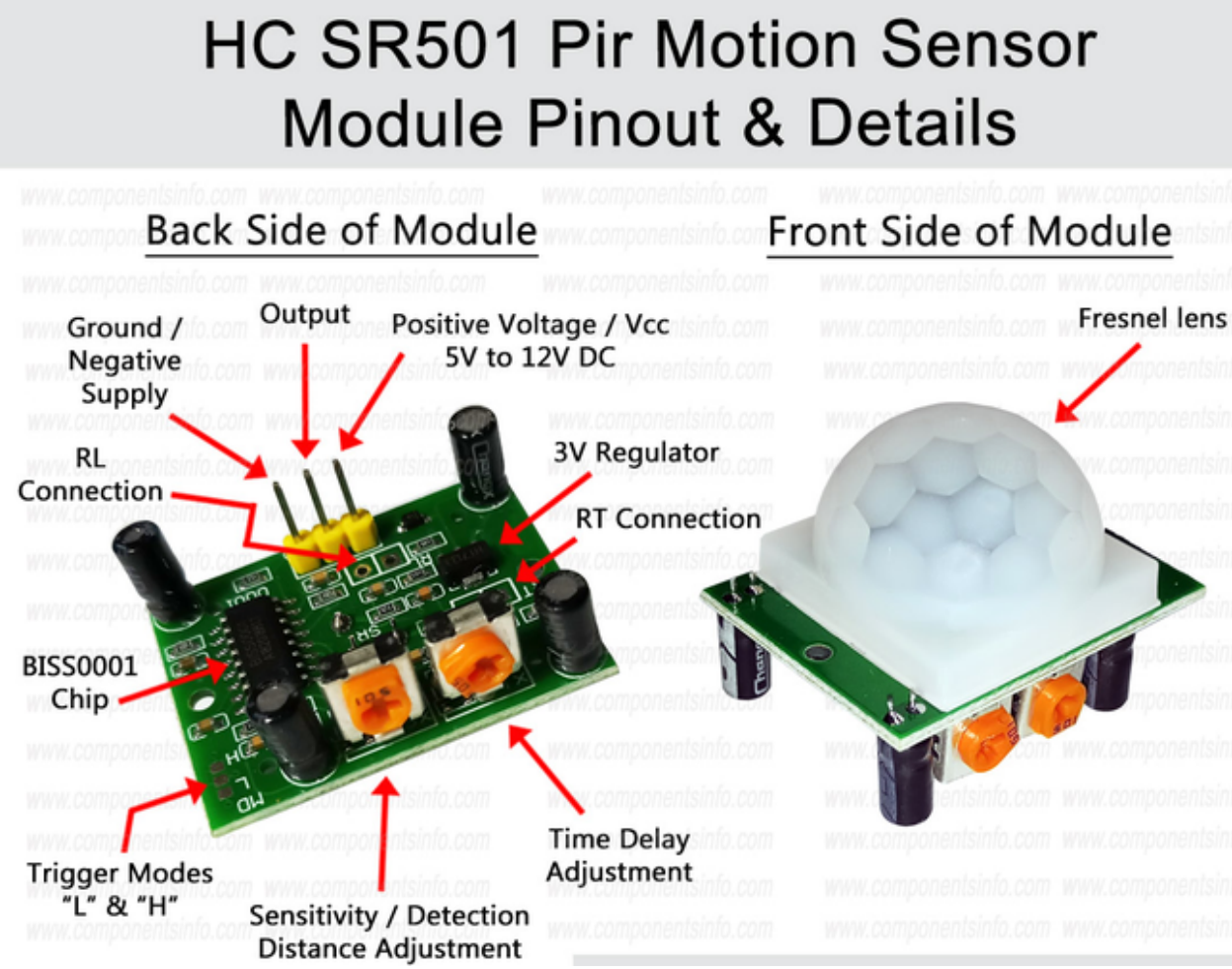
Thonny IDE: *sudo apt install thonny*

pigpio: https://abyz.me.uk/rpi/pigpio/python.html

//Check if pigpio Is installed:

```
sudo apt update
sudo apt install pigpio python3-pigpio
```

****



****

PIR Sensor Example (for Exercise No. 3)

Connections

- **PIR VCC** → 5V
- **PIR GND** → GND
- **PIR OUT** → GPIO 17
- **LED anode (+)** → GPIO 18 through a **220–330 Ω resistor**
- **LED cathode (−)** → GND

Sample Code snippet for PIR sensor using RPI.GPIO:

```
import RPi.GPIO as GPIO
import time

# GPIO pins using BCM numbering
PIR_PIN = 17
LED_PIN = 18

# Set GPIO numbering mode
GPIO.setmode(GPIO.BCM)

# Configure GPIO pins
GPIO.setup(PIR_PIN, GPIO.IN)
GPIO.setup(LED_PIN, GPIO.OUT)

# Make sure LED starts OFF
GPIO.output(LED_PIN, GPIO.LOW)

print("PIR motion detector started...")
print("Waiting for motion...")

try:
    while True:
        motion = GPIO.input(PIR_PIN)

        if motion == GPIO.HIGH:
            print("Motion detected!")

            # Blink LED 5 times
            for _ in range(5):
                GPIO.output(LED_PIN, GPIO.HIGH)
                time.sleep(0.2)

                GPIO.output(LED_PIN, GPIO.LOW)
                time.sleep(0.2)

            # Wait until motion signal goes LOW
```

```
            while GPIO.input(PIR_PIN) == GPIO.HIGH:
                time.sleep(0.1)

            print("Motion stopped. Waiting...")

        time.sleep(0.1)

except KeyboardInterrupt:
    print("\nExiting...")

finally:
    GPIO.output(LED_PIN, GPIO.LOW)
    GPIO.cleanup()
```

Sample Code snippet for PIR sensor using PIGPIO:

```
import pigpio
import time
# GPIO pins
PIR_PIN = 17
LED_PIN = 18
# Connect to the pigpio daemon
pi = pigpio.pi()
if not pi.connected:
   raise RuntimeError("Could not connect to pigpio daemon")
# Configure GPIO
pi.set_mode(PIR_PIN, pigpio.INPUT)
pi.set_mode(LED_PIN, pigpio.OUTPUT)
# Make sure LED starts OFF
pi.write(LED_PIN, 0)
```

```
print("PIR motion detector started...")
print("Waiting for motion...")
try:
    while True:
        motion = pi.read(PIR_PIN)
        if motion:
            print("Motion detected!")
            # Blink LED 5 times
            for _ in range(5):
                pi.write(LED_PIN, 1)
                time.sleep(0.2)
                pi.write(LED_PIN, 0)
                time.sleep(0.2)
            # Wait until motion signal goes LOW
            while pi.read(PIR_PIN):
                time.sleep(0.1)
            print("Motion stopped. Waiting...")
        time.sleep(0.1)
except KeyboardInterrupt:
    print("\nExiting...")
finally:
    pi.write(LED_PIN, 0)
    pi.stop()
        pi.stop()
**************************************************************************
```