BRAC
UNIVERSITY

Inspiring Excellence

Lab report 1 of CSE461

Submitted By:

Group 4

Tabassum Nusrat Jahan

ID- 19201027

Tahmid Chowdhury

ID- 19201115

Sadman Sakib

ID- 19201029

Mahfuza Sultana Mim

ID- 18101703

Fairuz Anika

ID- 2030146

1.1 Name of the experiment : Controlling a LED on the Raspberry Pi 4

1.2 Objective: The objective of controlling an LED on the Raspberry Pi 4 is to learn how to use the GPIO (General Purpose Input/Output) pins on the Raspberry Pi to interface with external components, in this case, an LED. By controlling the LED, we can understand the basics of working with digital outputs and gain a foundation for more complex projects involving sensors, actuators, and other electronic components.

1.3 Equipment:

- ★ Raspberry Pi 4
- ★ LED
- ★ A resistor of 200 ohms
- ★ Connecting wires (female to male or male to male)
- ★ Breadbroad

1.4 Experimental Setup:

For the experiments firstly we set up our raspberry-pi and then open a python IDE. Then according to circuits we build the circuit on a breadboard and connect through raspberry-pi.

To do experiment 1 firstly we build the circuit on a breadboard. To do this task firstly we took the power from the GPIO 17(Pin number 11) of raspberry-pi by the help of a jumper wire and then connect to the positive point of the breadboard.After that we took the Ground from the Pin number 6 of GPIO and connect by the help of a jumper wire in the negative points of breadboard. Moreover, we connect the positive side of LED light to the positive points of the breadboard which means Vcc and negative side of LED to the negative side of the breadboard. We also connected a 200 ohm resistor where one point was connected to the negative side of the LED and another point was connected with the negative side of the breadboard which means ground. After running the code the LED lit up.

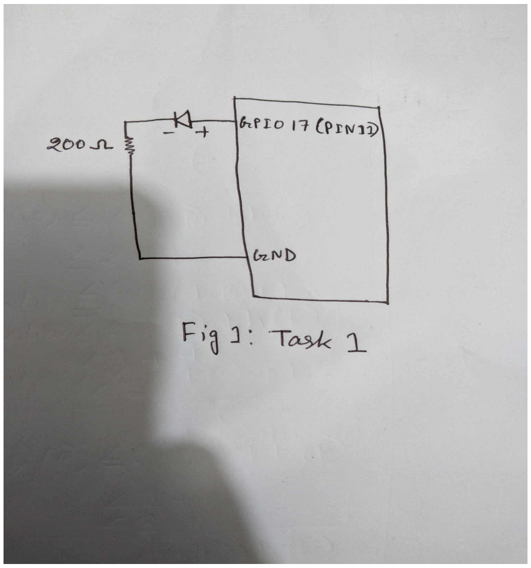

Fig1: Hand drawn circuit of Task 1

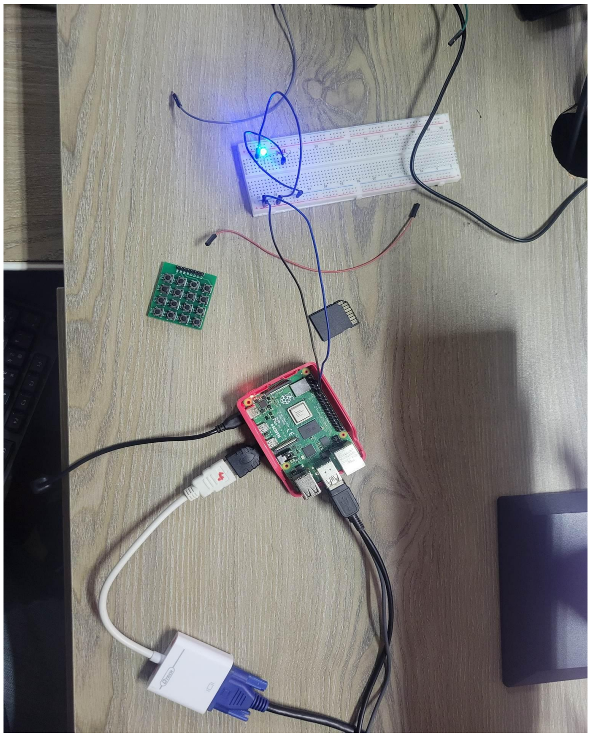

Fig2: Circuit implementation of Task 1

1.5 Code:

```
import RPi.GPIO as GPIO
Import time

LED_PIN = 17
GPIO.setmode(GPIO.BCM)
GPIO.setup(LED_PIN, GPIO.OUT)
while(True):
        GPIO.output(LED_PIN, GPIO.HIGH)
        time.sleep(1)
        GPIO.output(LED_PIN, GPIO.LOW)
        time.sleep(1)
```

1.6 Result: The demonstration of Controlling a LED on the Raspberry pi 4 was successful. When we connected the LED, Raspberry Pi 4 with jumper wires and breadboard and executed the code, the LED turned on. Also, we observed the response time which was satisfactory too.

2.1 Name of the experiment : Controlling a LED with a push button on the Raspberry Pi 4

2.2 Objective: The objective of controlling an LED with a push button on the Raspberry Pi 4 is to learn how to use both digital inputs and outputs on the GPIO (General Purpose Input/Output) pins. By integrating a push button into the circuit, we can control the LED based on the button's state, allowing for interactive projects and user input.

2.3 Equipment:

★ Raspberry Pi 4
★ LED
★ A resistor of 200 ohms
★ Connecting wires (female to male or male to male)
★ Breadbroad
★ Push Button Switch

2.4 Experimental Setup: To do experiment 2 firstly we build the circuit on a breadboard. To do this task firstly we took the power from the GPIO 4(Pin number 7) of raspberry-pi by the help of a jumper wire and then connect to the positive point of the breadboard.After that we took the Ground from the Pin number 6 of GPIO and connect by the help of a jumper wire in the negative points of breadboard. Moreover, we connect the positive side of LED light to the positive points of the breadboard which means Vcc and negative side of LED to the negative side of the breadboard. We also connected a 200 ohm resistor where one point was connected to the negative side of the LED and another point was connected with the negative side of the breadboard which means ground. Furthermore, we connected a Push Button Switch which R1 was connected with GPIO 27(Pin number 13) and C1 was connected with the Ground which means the ground of the breadboard. After running the code and pushing the C1 R1 switch of the Push Button the LED lit up.

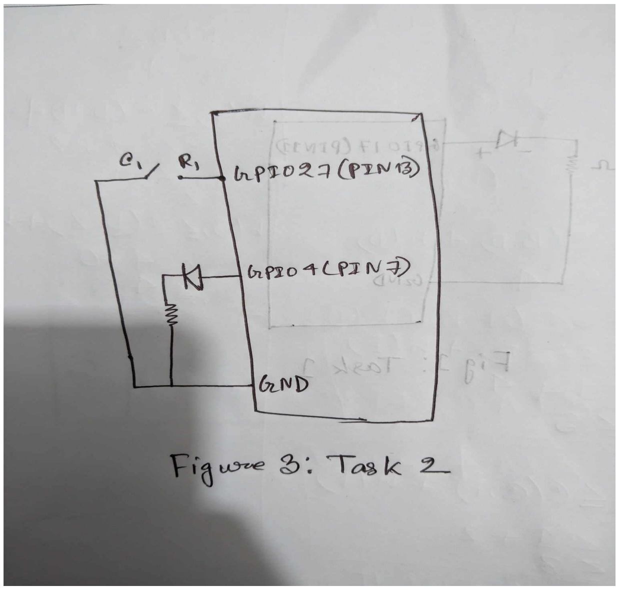

Fig 3: Hand drawn circuit of Task2

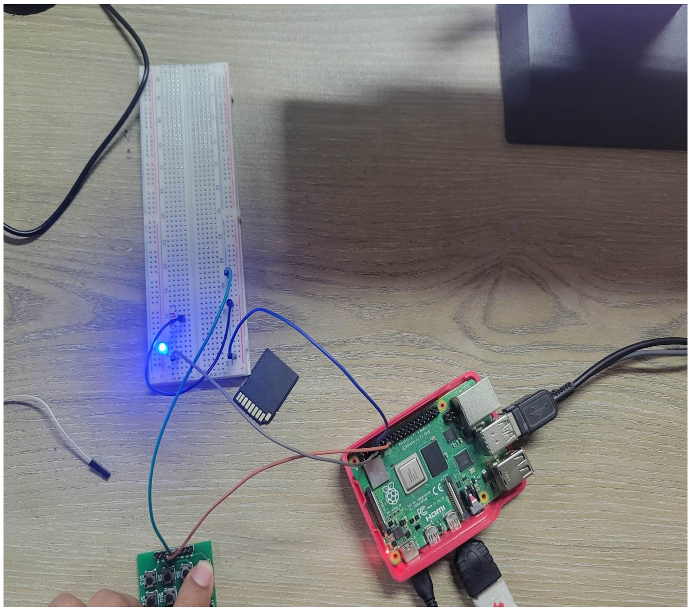

Fig 4: Circuit implementation of Task 2

2.5 Code:

```
from gpiozero import Device,LED, Button
from time import sleep

button1 = Button(27)
Led = LED(4)
while (True):
        led.off()
        button1.wait_for_press()
        led.on()
        button1.wait_for_release()
```

2.6 Result: The Experiment functioned well. We connected all the components with Raspberry Pi as per the requirement and ran the code. As a result, when we pressed the push button LED turned on and when we released the push button LED turned off. Thus, the demonstration was adequate.

3.1 Name of the experiment : Controlling a LED with a push button on the Raspberry Pi 4

3.2 Objective: The objective of controlling an LED with a push button on the Raspberry Pi 4 is to learn how to interface with both digital inputs (the push button) and digital outputs (the LED) using the GPIO pins. This objective combines the concepts of reading input states and controlling output states to create an interactive project.

3.3 Equipment:

- ★ Raspberry Pi 4
- ★ LED
- ★ A resistor of 200 ohms
- ★ Connecting wires (female to male or male to male)
- ★ Breadbroad
- ★ Push Button

3.4 Experimental Setup: To do experiment 3 firstly we build the circuit on a breadboard. To do this task firstly we took the power from the GPIO 27(Pin number 13) of raspberry-pi by the help of a jumper wire and then we took the Ground from the Pin number 6 of GPIO and connect by the help of a jumper wire in the negative points of breadboard. After that from that positive point of the breadboard we connect the R1 of Push Button Switch and then connect the C1 of the Push Button switch with the positive side of the LED by the help of jumper wires. Moreover, we connect the negative side of the LED to the one point of the 200 ohm resistor and another point of resistor connected with the Ground of the breadboard. After running the code and pushing the C1 R1 switch of the Push Button the LED lit up.

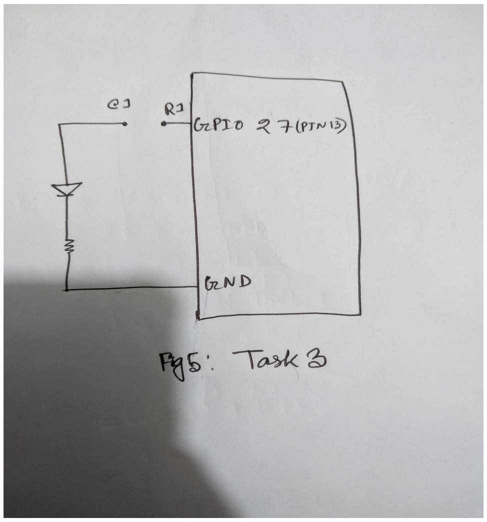

Fig 5: Hand drawn circuit of Task3

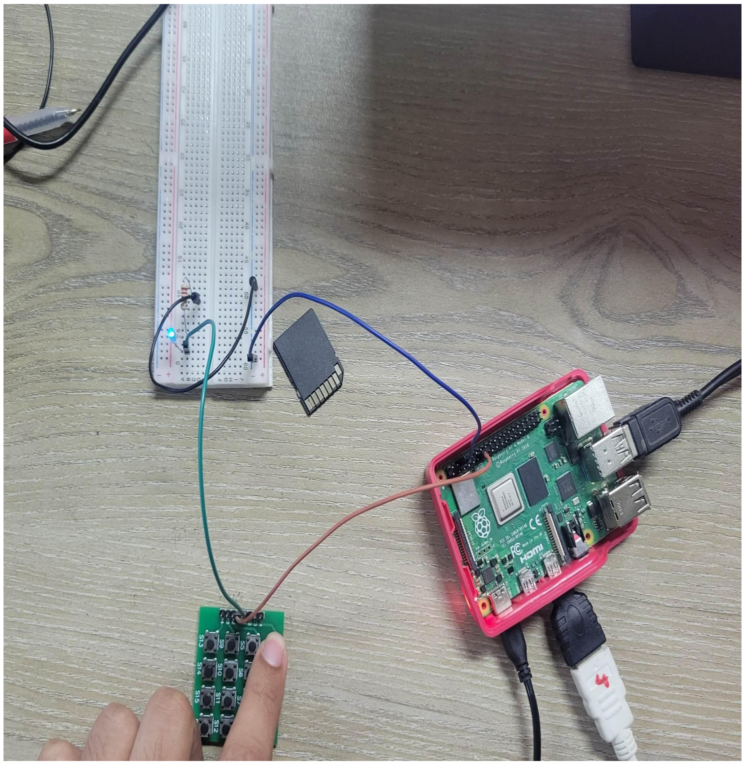

Fig 6: Circuit implementation of Task 3

3.5 Code:

```
import RPi.GPIO as GPIO
Import time

LED_PIN = 27
GPIO.setmode(GPIO.BCM)
GPIO.setup(LED_PIN, GPIO.OUT)
while(True):
GPIO.output(LED_PIN, GPIO.HIGH)
```

3.6 Result:

The experiment of Controlling a LED with a push button on the Raspberry Pi 4 was successful. The whole circuit was implemented according to the given instructions and diagrams. After connecting the LED, register, push button and implying the code, the LED lit up which was satisfactory.

4. Discussion:

To have a clear signal the connection to GND is required. The signal will be 0, if no pulse is delivered. Otherwise, it will be 1. The successful implementation of a push button mechanism with the Raspberry Pi 4 allowed researchers to effectively monitor an LED light during experimentation. Due to its ease of use, setup and programming were seamless processes that yielded reliable results. The adaptable nature of the Raspberry Pi 4 made it a suitable platform for this particular type of control system.