LED Blinking using Raspberry Pi

Introduction:

The LED blinking project is one of the simplest and best ways to get started with Raspberry Pi and Python programming. It helps beginners understand how to control external components using the Raspberry Pi's GPIO (General Purpose Input/Output) pins. By turning an LED on and off in a loop, we learn the basics of hardware interaction, which is an essential part of embedded systems and IoT projects.

Components Required:

- Raspberry Pi (any model with GPIO pins)
- LED (any color)
- Resistor (220Ω to 330Ω, to protect the LED)
- Breadboard (for easy connections)
- Jumper wires (Male-to-Female for GPIO connection)
- Power supply (5V micro-USB adapter for Raspberry Pi)

Circuit Connection:

1. Insert the LED into the breadboard (longer leg = positive, shorter leg = negative).
2. Connect a 220Ω or 330Ω resistor in series with the LED's anode.
3. Wire to Raspberry Pi:
    - Connect the longer leg (through the resistor) to GPIO Pin 17 (or any other GPIO pin).
    - Connect the shorter leg to a GND pin on the Raspberry Pi.
4. Power up the Raspberry Pi and run the script to blink the LED.

Applications:

- Used in IoT projects to signal system status.
- Helps in learning GPIO programming and controlling external hardware.
- Can be used as an alert system for notifications.
- Forms the base for more advanced automation and robotics projects.

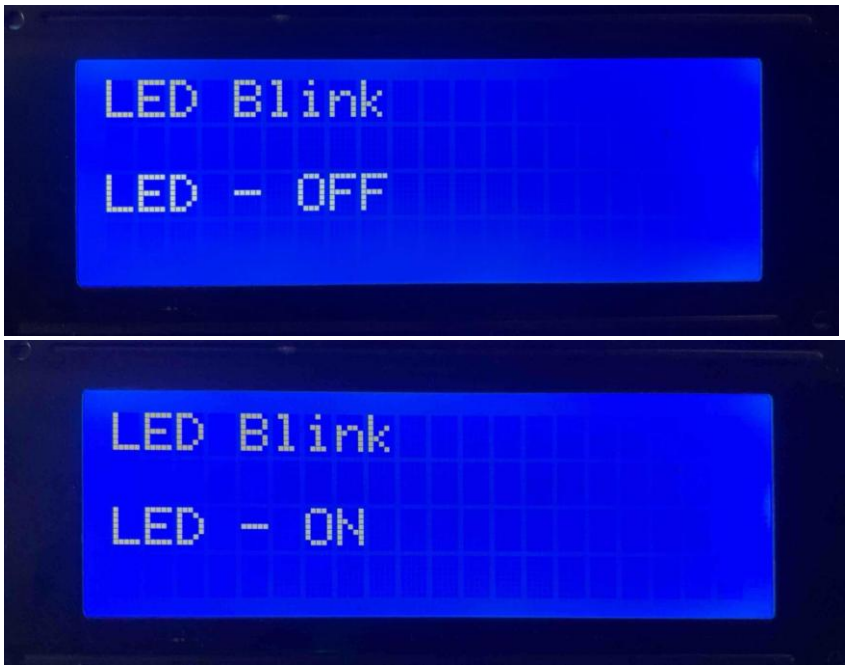
Learnings:

- Understand basic GPIO pin usage on Raspberry Pi.
- Learn how to connect LEDs and resistors correctly.
- Gain knowledge of controlling hardware using Python.
- Improve your circuit building and debugging skills.
- Learn how to power and operate Raspberry Pi for embedded projects.

Conclusion:

The LED blinking project is a beginner-friendly way to start working with Raspberry Pi and electronics. It lays the foundation for more advanced projects like home automation, sensor-based systems, and IoT applications. Once you master LED control, you can expand your learning by connecting multiple LEDs, sensors, or even motors to your Raspberry Pi.

Outcome:



Program:

```
from RPLCD.i2c import CharLCD
import RPi.GPIO as GPIO
import time
lcd = CharLCD('PCF8574', 0x27)

GPIO.setwarnings(False)

LED_Pin = 26

GPIO.setmode(GPIO.BCM)
GPIO.setup(LED_Pin, GPIO.OUT)

while True:
    lcd.clear()
    lcd.cursor_pos = (0, 0)  # First row
    lcd.write_string(f'LED Blink')
    lcd.cursor_pos = (2, 0)  # First row
```

```
lcd.write_string(f'LED - ON')
GPIO.output(LED_Pin, GPIO.HIGH)
time.sleep(1)

lcd.clear()
lcd.cursor_pos = (0, 0)  # First row
lcd.write_string(f'LED Blink')
lcd.cursor_pos = (2, 0)  # First row
lcd.write_string(f'LED - OFF')
GPIO.output(LED_Pin, GPIO.LOW)
time.sleep(1)
```