Experiment No. 5

Government College of Engineering, Jalgaon

(An Autonomous Institute of Government of Maharashtra)

Name :	**PRN :**
Class : T. Y. B.Tech Computer	**Academic Year :** 2024-25
Date of Performance :	**Date of Completion :**
Subject : CO357UB IOT Lab	**Course Teacher** : Miss. Prajakta Sawale

Aim: Working with Analog I/O's in Arduino.

Components Required:

- Arduino Uno R3
- Breadboard
- Potentiometer
- LED
- Resistor (220Ω or 330Ω)
- Jumper Wires **Theory:**

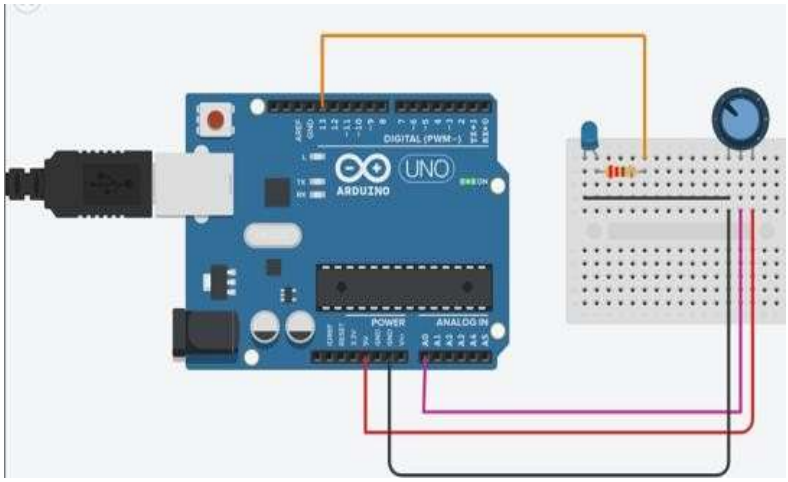
Arduino Uno has several pins that support analog input and output functionalities. Analog **inputs** are read using the ADC (Analog to Digital Converter), typically from sensors like potentiometers, and **outputs** are simulated using PWM (Pulse Width Modulation) via analogWrite().

- **Analog Input:** Pins A0 to A5 on Arduino Uno are capable of reading voltage between 0 to 5V and converting it to a digital value between 0 and 1023.
- **Analog Output (PWM):** Digital pins ~3, ~5, ~6, ~9, ~10, and ~11 support PWM, allowing analog-like output by varying the duty cycle of the signal.

Arduino has analog input pins (A0 to A5) which can read voltage levels between 0–5V using its builtin Analog-to-Digital Converter (ADC). This ADC converts analog voltage into a digital value ranging from 0 to 1023.
In this experiment:

- We use a **potentiometer** to generate variable voltage.
- The Arduino reads this voltage and controls the **blink speed** of an LED based on the analog input.

Circuit Diagram:



Circuit Connections:

- Potentiometer: ○ One end to **5V** ○ Other end to **GND** ○ Middle terminal to **A0**
- LED:
  - ○ Anode (+) to digital pin **13** via **220Ω**

resistor ○ Cathode (–) to **GND Code:**

```
void setup() {
Serial.begin(9600);
pinMode(13, OUTPUT);
}

void loop() {  int PotVal = analogRead(A0);      // Read
potentiometer value  Serial.println(PotVal);          // Print
value to Serial Monitor  digitalWrite(13, HIGH);          //
Turn ON LED  delay(PotVal);                  // Delay based on
PotVal  digitalWrite(13, LOW);           // Turn OFF LED
delay(PotVal);                  // Delay again
}
```

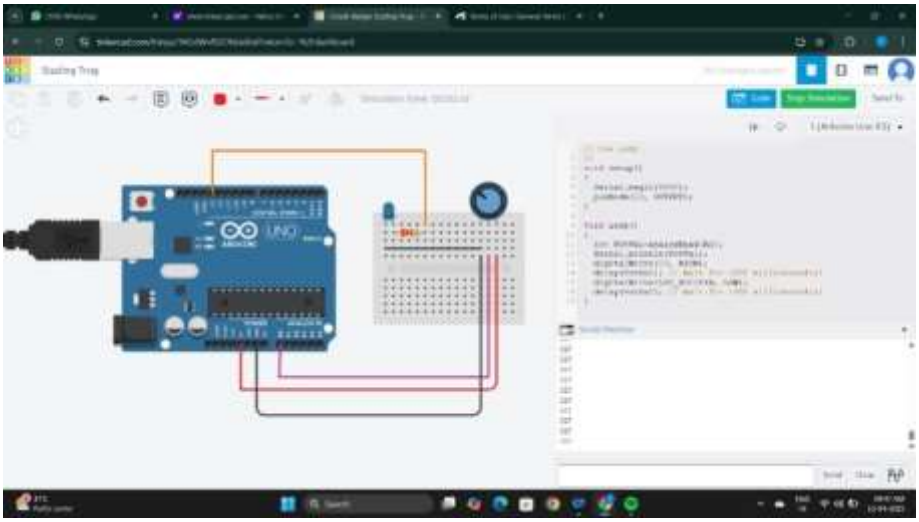
Working/Observation:

- When you rotate the **potentiometer**, the value from **analogRead(A0)** changes (0 to 1023).

- This value is printed on the **Serial Monitor**.
- The **LED blinks** faster or slower based on the potentiometer's position.
    - Low PotVal = short delay → **fast blinking**

o High PotVal = long delay → **slow blinking Result:**

The Arduino successfully reads analog input and adjusts the LED blink rate, demonstrating the use of analog input (potentiometer) and digital output (LED control).



Conclusion:

This experiment proves the use of analog input from a sensor to control an output in real-time, which is essential in embedded systems and IoT projects.

Sign of Course Teacher