IOT Slips

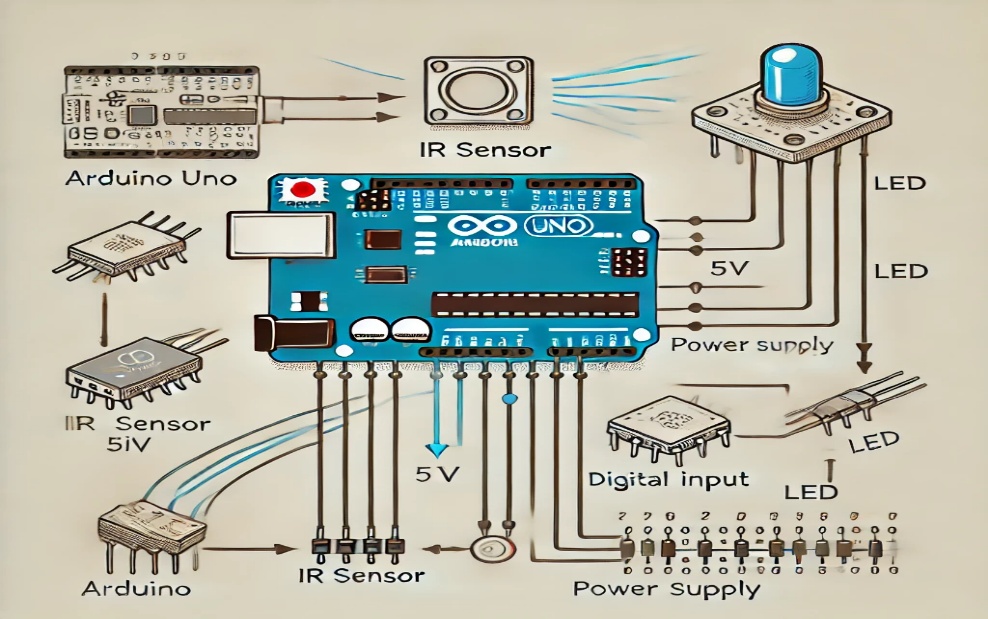
\*\* Blink LED

**a) Block Diagram / Pin Diagram**

Below is the block diagram of **Arduino Uno interfaced with an IR Sensor and LED**:

**Block Diagram Components:**

* **Arduino Uno** (Microcontroller)
* **IR Sensor** (Input device)
* **LED** (Output device)
* **Power Supply** (5V from Arduino)



1 \_\_\_\_

int LED\_PIN = 13; // The built-in LED on Arduino

void setup() {

pinMode(LED\_PIN, OUTPUT); // Set pin as output

}

void loop() {

digitalWrite(LED\_PIN, HIGH); // Turn LED on

delay(1000); // Wait 1 second

digitalWrite(LED\_PIN, LOW); // Turn LED off

delay(1000);

}

2\_\_\_\_

void setup() {

pinMode(LED\_BUILTIN, OUTPUT);

}

void loop() {

digitalWrite(LED\_BUILTIN, HIGH);

delay(1000);

digitalWrite(LED\_BUILTIN, LOW);

delay(1000);

}

**Observations on Input and Output**

**Input:**

The Arduino receives an electrical signal and controls the LED using the digitalWrite() function.

**Output:**

The LED blinks every second, turning ON and OFF as per the delay function.

**Result:** The LED successfully blinks at a 1-second interval.

**Conclusion:**

1This experiment demonstrates **basic microcontroller programming** using the **Arduino Uno**.

2It helps in understanding **GPIO pin control** for digital outputs.

\*\* Buzzer ON/OFF

(a) Block Diagram / Pin Diagram

**Board:** Arduino Uno  
**Sensor:** IR Sensor  
**Output Device:** Buzzer  
**Power Supply:** 5V from Arduino

1\_\_\_

const int buzzerPin = 5; // Connect the piezo buzzer to pin 5

const int tickFrequency = 1000; // 1000 Hz frequency

// Define the duration for the "tick" sound (in milliseconds)

const int tickDuration = 200; // 200 ms duration

void setup() {

// Set the buzzer pin as an output

pinMode(buzzerPin, OUTPUT);

}

void loop() {

tone(buzzerPin, tickFrequency, tickDuration);

delay(1000); // 1-second gap before next tick

}

2\_\_\_\_\_\_\_

#define BUZZER\_PIN 9 // Connect the buzzer to digital pin 9

void setup() {

pinMode(BUZZER\_PIN, OUTPUT);

}

void loop() {

digitalWrite(BUZZER\_PIN, HIGH); // Turn ON the buzzer

delay(1000); // Wait 1 second

digitalWrite(BUZZER\_PIN, LOW); // Turn OFF the buzzer

delay(1000); // Wait 1 second

}

**Observations:**

* **Input:** The Arduino sends a **tone signal (1000 Hz)** to the buzzer through **pin 5**.
* **Output:** The buzzer **beeps ("tick" sound) for 200 milliseconds every 1 second**.

**Result:**

The buzzer successfully **produces a periodic "tick" sound** at **1000 Hz** for **200 ms** every **1 second**.

**Conclusion:**

* The experiment demonstrates **basic GPIO control** using Arduino.
* The **tone() function** effectively generates sound of a specific frequency and duration.
* This setup can be used in **alarms, timers, and notifications** in embedded systems.

\*\* Toggle 2 LED

**a) Block Diagram / Pin Diagram**

**Board:** Arduino Uno  
**Interface Device:** IR Sensor / Temperature Sensor / Camera (Based on Examiner's Assignment)  
**Output Device:** Two LEDs

1\_\_\_\_\_\_\_\_

const int led1 = 13; // Built-in LED on pin 13

const int led2 = 2; // External LED on pin 2

void setup() {

pinMode(led1, OUTPUT);

pinMode(led2, OUTPUT);

}

void loop() {

digitalWrite(led1, HIGH); // Turn ON LED 1

digitalWrite(led2, LOW); // Turn OFF LED 2

delay(1000); // Wait 1 second

digitalWrite(led1, LOW); // Turn OFF LED 1

digitalWrite(led2, HIGH); // Turn ON LED 2

delay(1000); // Wait 1 second

}

**Observations**

**Input:**

The Arduino / Raspberry Pi sends alternating signals to two LEDs using GPIO pins.

**Output:**

**LED 1 turns ON, LED 2 turns OFF** → Waits 1 second.

**LED 1 turns OFF, LED 2 turns ON** → Waits 1 second.

This process **repeats continuously**.

**Result & Conclusion**

**Result:**

The two LEDs **successfully toggle every 1 second** as programmed.

**Conclusion:** This experiment demonstrates **basic GPIO control**, alternating LED blinking, and its practical application in **signaling systems, indicators, and visual alerts**.

2\_\_\_\_\_\_\_\_\_\_\_\_\_

const int led1 = 5; // First LED connected to pin 5

const int led2 = 6; // Second LED connected to pin 6

void setup() {

pinMode(led1, OUTPUT);

pinMode(led2, OUTPUT);

}

void loop() {

digitalWrite(led1, HIGH); // Turn ON LED 1

digitalWrite(led2, LOW); // Turn OFF LED 2

delay(1000); // Wait 1 second

digitalWrite(led1, LOW); // Turn OFF LED 1

digitalWrite(led2, HIGH); // Turn ON LED 2

delay(1000); // Wait 1 second

}

**Observations:**

* **Input:** Arduino alternates signals between two LEDs connected to **pin 5 and pin 6**.
* **Output:**
  + **LED 1 turns ON, LED 2 turns OFF** → Waits 1 second.
  + **LED 1 turns OFF, LED 2 turns ON** → Waits 1 second.
  + Repeats indefinitely.

**Result:**

* Both LEDs successfully **toggle every 1 second**, blinking alternately.

**Conclusion:**

* Demonstrates **GPIO control** for multiple outputs.
* Useful for **signaling systems, indicators, and alternating blink patterns**.