**IoT 206: Microcontroller and Interfacing IoT Lab**

**Experiment-5:** Introduction to Arduino Board and Observation of LED Blinking.

**Introduction to Arduino Board:**

Arduino is an open-source electronics platform that consists of both hardware and software components. The platform is designed to simplify the process of creating interactive projects and prototypes by providing an easy-to-use development environment. The key component of the Arduino ecosystem is the Arduino board, which is a microcontroller board based on various microcontrollers, such as the popular Atmel AVR series.

The components of the Ardunio Uno board are:

1. Microcontroller: The brain of the Arduino board, responsible for executing the program and controlling connected devices.

2. Digital and Analog Pins: These pins allow the microcontroller to interact with the external world. Digital pins can be configured as inputs or outputs, while analog pins can read analog signals.

3. Power Supply: The Arduino board can be powered via USB, an external power supply, or batteries, depending on the model.

4. Clock Oscillator: Provides the timing for the microcontroller's operations.

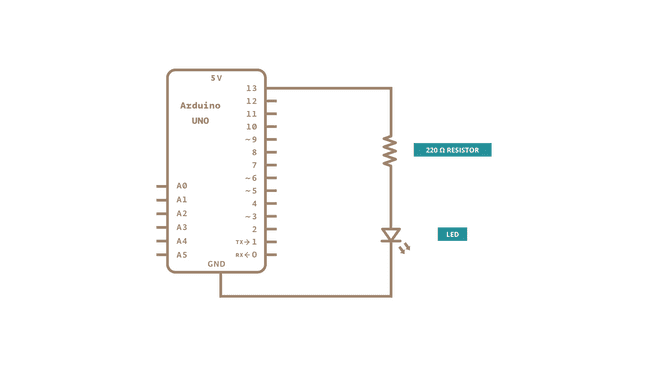
5. Reset Button: Resets the microcontroller, restarting the program execution.

6. Voltage Regulator: Regulates the incoming voltage to ensure a stable power supply for the components.

7. USB Interface: Enables communication between the Arduino board and a computer for programming and serial communication.

**Observation of LED Blinking:**

One of the simplest and most common introductory projects in the Arduino world is the LED blinking example. This project helps beginners understand the basics of programming and how to control external hardware using the Arduino board.



Schematic diagram for LED blinking

**LED Blinking Code:**

void setup() {

// Initialize the digital pin as an output.

pinMode(LED\_BUILTIN, OUTPUT);

}

void loop() {

// Turn the LED on (HIGH) for 1 second.

digitalWrite(LED\_BUILTIN, HIGH);

delay(1000);

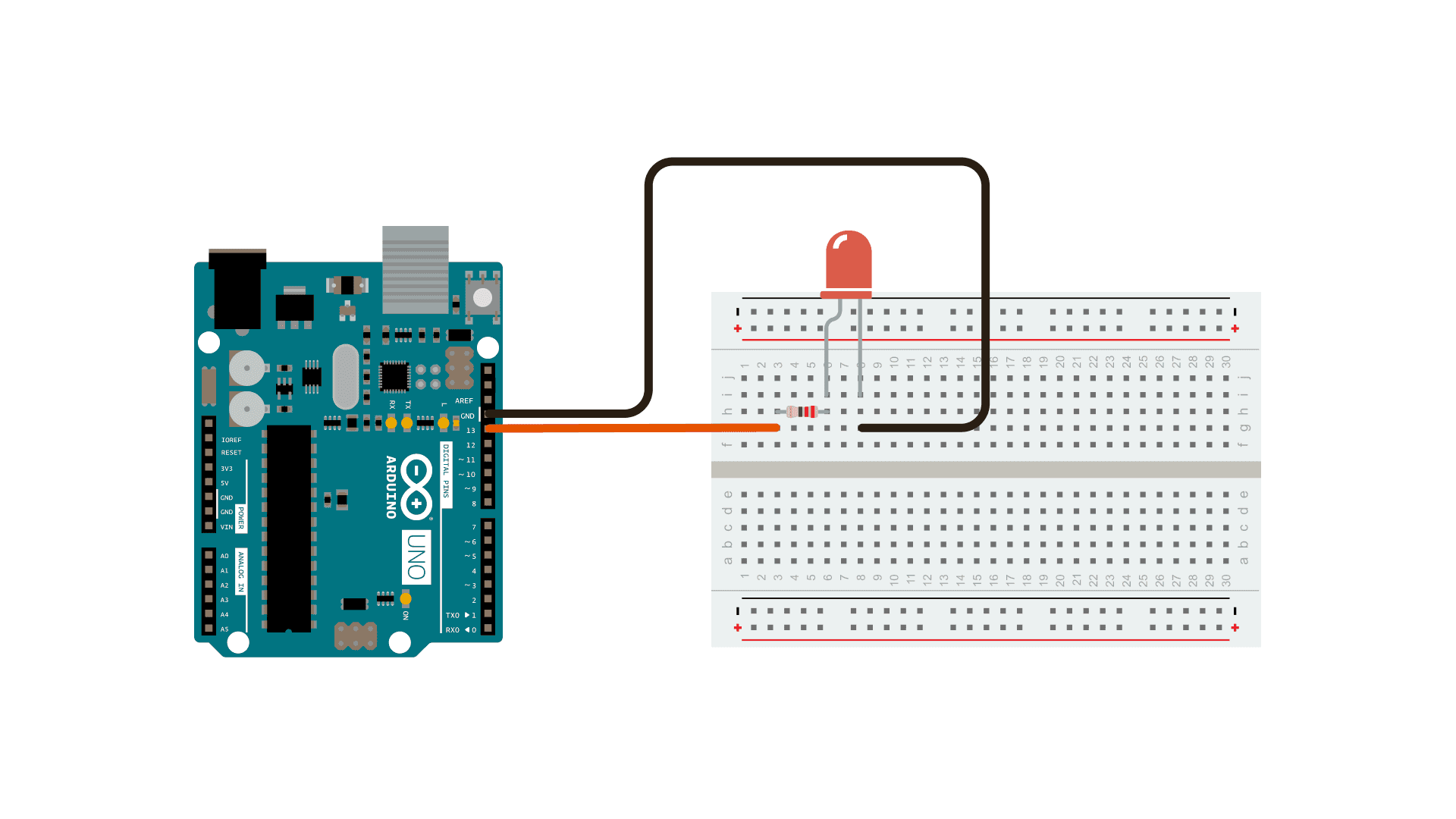
// Turn the LED off (LOW) for 1 second.

digitalWrite(LED\_BUILTIN, LOW);

delay(1000);

}

**LED on and off from Serial Monitor:**



// Define the pin number for the LED

int ledPin = 13;

void setup() {

// Initialize serial communication

Serial.begin(9600);

// Set the LED pin as an output

pinMode(ledPin, OUTPUT);

// Initially turn off the LED

digitalWrite(ledPin, LOW);

// Prompt the user to enter commands

Serial.println("Enter 'on' to turn the LED on, 'off' to turn it off:");

}

void loop() {

// Check if data is available to read from serial

if (Serial.available() > 0) {

// Read the incoming byte

char command = Serial.read();

// Check the command received

if (command == 'on') {

digitalWrite(ledPin, HIGH); // Turn the LED on

Serial.println("LED is turned on.");

} else if (command == 'off') {

digitalWrite(ledPin, LOW); // Turn the LED off

Serial.println("LED is turned off.");

} else {

Serial.println("Invalid command. Enter 'on' to turn the LED on, 'off' to turn it off:");

}

}

}