

# Practical Robotics Projects with Arduino (CSE 4571)

## Lab Assignment No – 08

### Bluetooth Communication

Submission Date: \_\_\_\_\_

Branch: CSE		Section:
Name	Registration No.	Signature

Department of Computer Science and Engineering  
Institute of Technical Education and Research (Faculty of Engineering)  
**Siksha 'O' Anusandhan (Deemed to be University)**  
**Bhubaneswar, Odisha-751030.**

# Aim:

To interface an HC-05/HC-06 Bluetooth module with Arduino UNO for wireless serial communication between the Arduino and a Bluetooth-enabled device.

## Objectives:

### 1) To gain familiarity with the HC-05/HC-06 module and Arduino UNO serial communication.

- Learn the pin configuration (VCC, GND, TXD, RXD, EN/KEY) of the HC-05/HC-06 module.
- Understand the role of baud rate, pairing code, and serial communication protocol.

### 2) To establish a basic wireless link between Arduino UNO and a smartphone/PC.

- Connect the HC-05/HC-06 module to the Arduino UNO (using SoftwareSerial).
- Write a simple Arduino sketch to send and receive data over Bluetooth.
- Pair the module with a smartphone/PC using a Bluetooth terminal app and verify successful two-way communication.

### 3) To implement bidirectional communication and LED control.

- Build an external circuit with an LED connected to Arduino digital pin 6.
- Modify the Arduino program to turn the LED ON when the smartphone sends the character 1, and OFF when it sends 0.
- Send acknowledgment messages back to the smartphone (e.g., “LED ON” / “LED OFF”).

### 4) To Evaluate communication performance.

- Test and record the effective communication range of the Bluetooth module.
- Measure response latency between sending a command (from phone) and Arduino execution.

## Pre-Lab Questionnaire:

- 1) Name all the pins of the HC-05/HC-06 module and their functions.
- 2) Why is a voltage divider required between Arduino TX and HC RX?
- 3) What is the default baud rate of the HC-05/HC-06 module?
- 4) Explain the difference between HC-05 and HC-06 modules.
- 5) What is Software Serial in Arduino, and why is it used in this experiment?
- 6) What is the purpose of pairing a Bluetooth device before communication?
- 7) How does Arduino interpret commands sent from a smartphone?
- 8) How can you verify if the HC-05/HC-06 module is powered and ready for pairing?
- 9) List the factors that can affect Bluetooth communication range.
- 10) Explain the role of the Serial Monitor in this experiment.

## Answers to Pre-Lab Questions



# Components/Equipment Required:

Sl. No.	Name of the Component / Equipment	Specification	Quantity
1)	Arduino UNO R3	16MHz	1
2)	Arduino UNO cable	USB Type A to Micro-B	1
3)	HC-05 / HC-06 Bluetooth Module	Bluetooth 2.0 SPP, 3.3V TTL logic, default baud rate 9600 bps	1
4)	Resistors (carbon type)	220Ω / 2.2kΩ/ 1 kΩ	1 each
5)	LED	Any colour of your choice	1
6)	Breadboard	840 Tie points	1
7)	Smartphone with Bluetooth / PC with Bluetooth	Android/iOS device or Windows PC, for testing wireless serial communication	1
8)	Bluetooth Terminal App	Android/iOS app like “Serial Bluetooth Terminal” or “Bluetooth Terminal	1
9)	Jumper Wire	-----	As per requirement

## Objective 1

To Gain familiarity with the HC-05/HC-06 module and Arduino UNO serial.

### Circuit / Schematic Diagram

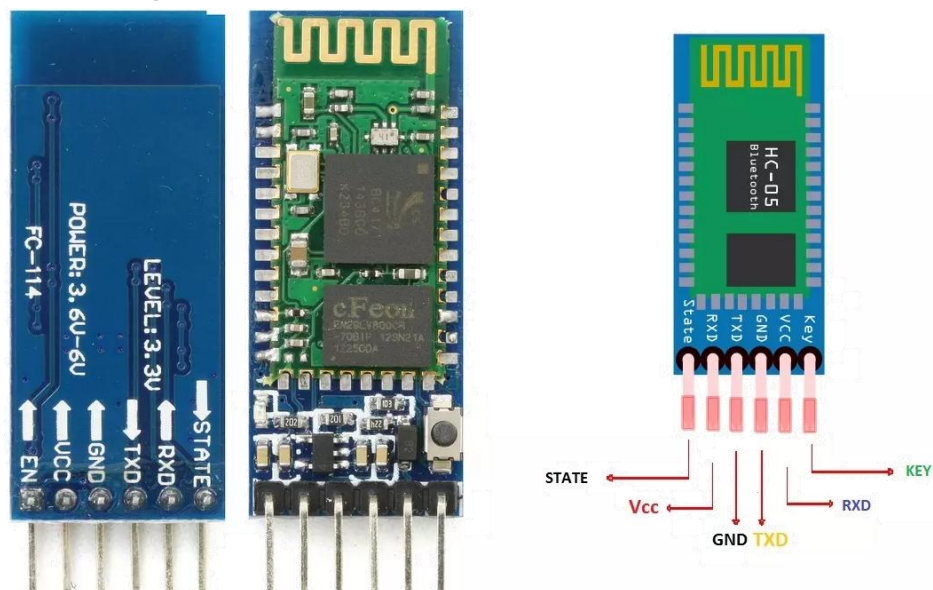


Figure 1: HC-05/HC-06 module Pinout

## Objective 2

To establish a basic wireless link between Arduino UNO and a smartphone/PC.

### Circuit / Schematic Diagram

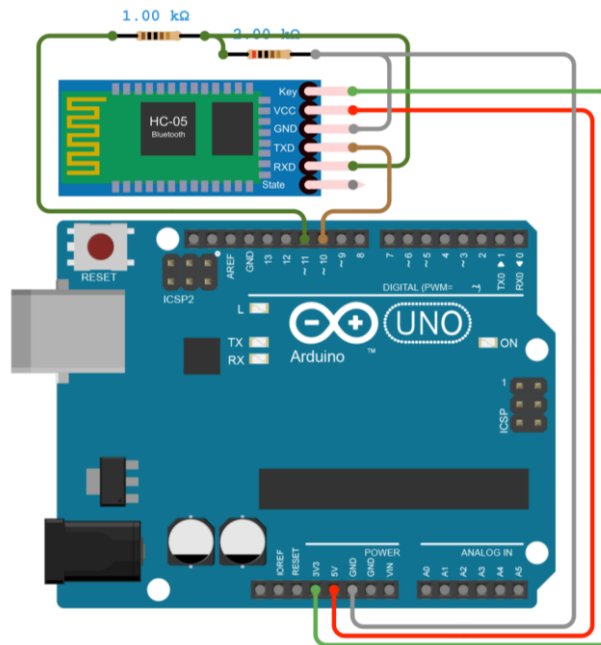


Figure 2: Interface between Arduino UNO and HC-05/HC-06 module

### Code

```
#include <SoftwareSerial.h>
SoftwareSerial BT(10, 11); // RX, TX

void setup() {
  Serial.begin(9600);
  BT.begin(9600);
  Serial.println("Ready to communicate. Type on Serial Monitor or Bluetooth app.");
}

void loop() {
  if (Serial.available()) {
    char data = Serial.read();
    BT.write(data);
  }
  if (BT.available()) {
    char data = BT.read();
    Serial.write(data);
  }
}
```

Figure 3: Interface of Arduino UNO with HC-05/HC-06 module

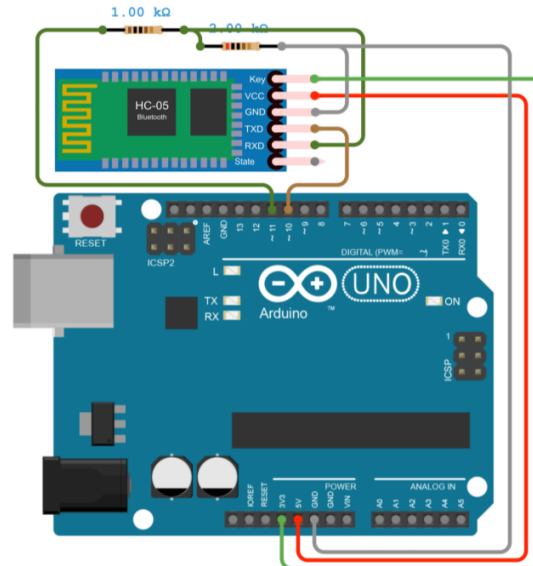
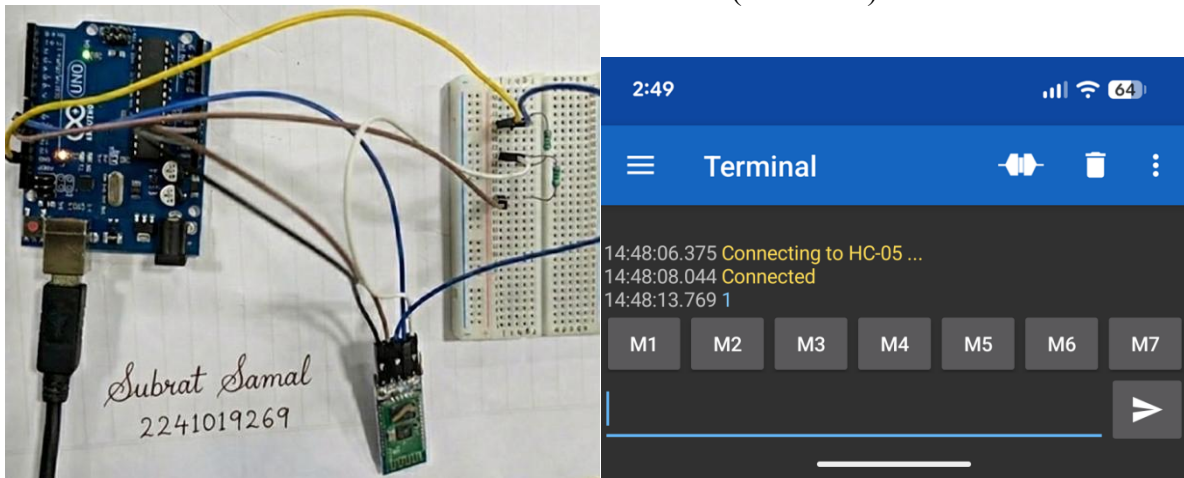


Figure 4: Interface of Arduino UNO with HC-05/HC-06 module (Hardware)



## Objective 3

To implement bidirectional communication and LED control.

### Circuit / Schematic Diagram

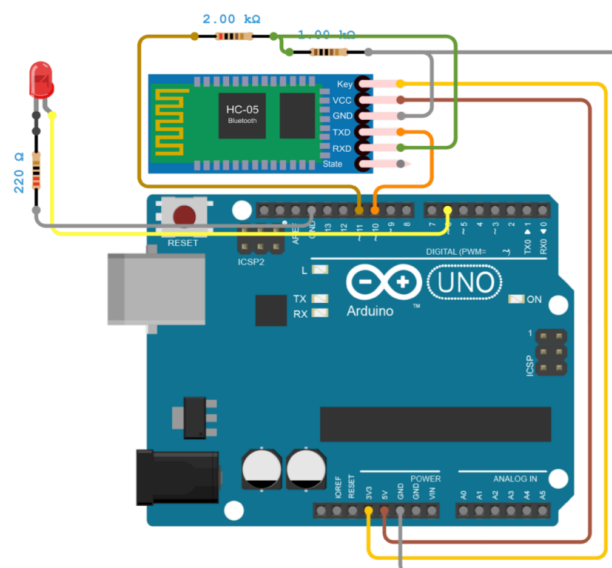


Figure 5: Interface between Arduino UNO and HC-05/HC-06 module

### Code

*PRACTICAL ROBOTIC PROJECTS USING ARDUINO (CSE 4571)*

*Servo motor control:-To Connect a potentiometer to Arduino UNO to control servo motor position and display the angle on a Display device.*

```

#include <SoftwareSerial.h>
SoftwareSerial BT(10, 11); // RX, TX
int led = 6;
void setup() {
  pinMode(led, OUTPUT);
  Serial.begin(9600);
  BT.begin(9600);
  Serial.println("Send '1' to turn ON LED, '0' to turn OFF.");
}
void loop() {
  if (BT.available()) {
    char cmd = BT.read();
    Serial.print("Received: ");
    Serial.println(cmd);
    if (cmd == '1') {
      digitalWrite(led, HIGH);
      BT.println("LED ON");
      Serial.println("LED turned ON");
    }
    else if (cmd == '0') {
      digitalWrite(led, LOW);
      BT.println("LED OFF");
      Serial.println("LED turned OFF");
    }
    else {
      BT.println("Invalid Command");
    }
  }
}

```

Figure 6: Interface HC-06 and Arduino UNO for bidirectional communication and LED control and confirmation in serial monitor window

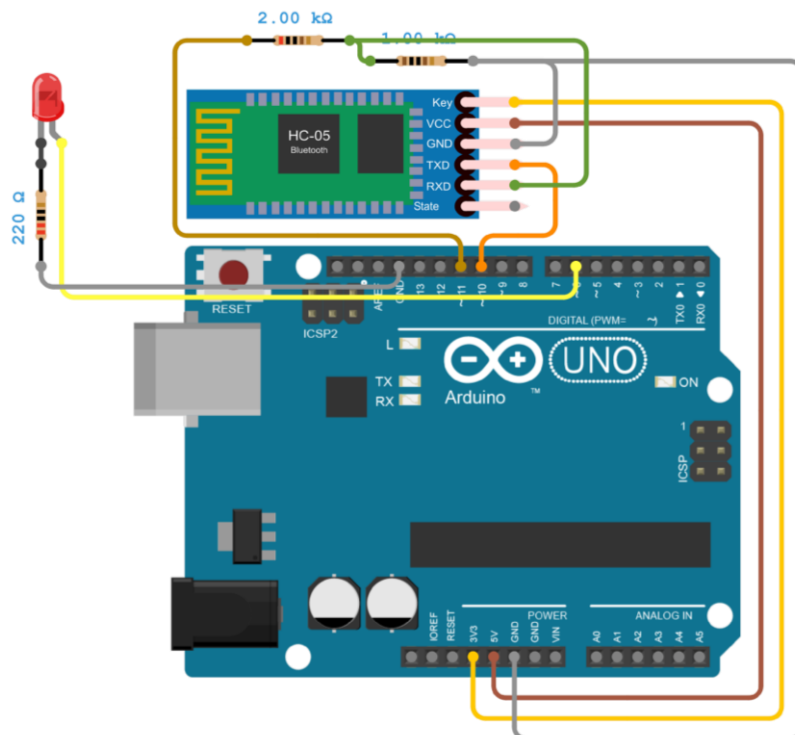
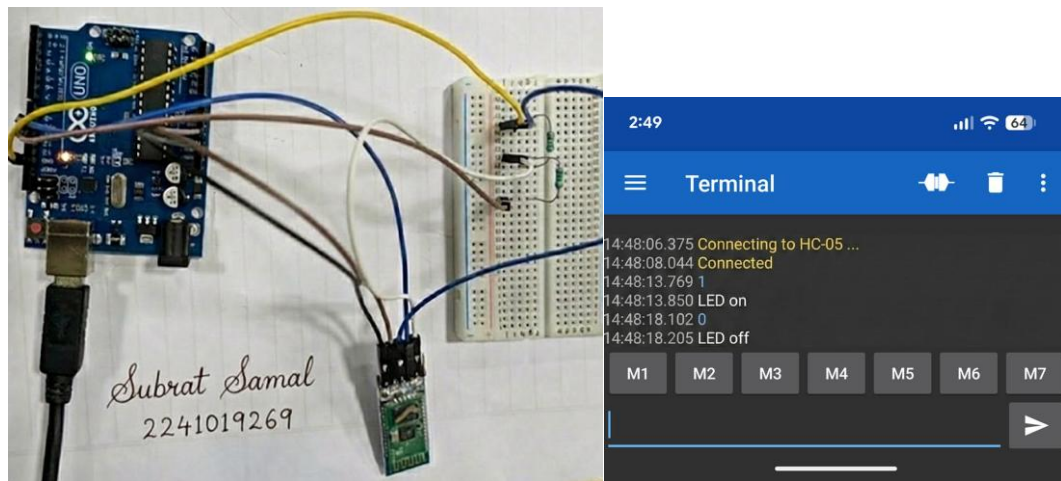


Figure 7: Interface HC-06 and Arduino UNO for bidirectional communication and LED control and confirmation in serial monitor window (Hardware)





## Objective 4

To Evaluate communication performance.

## Circuit / Schematic Diagram

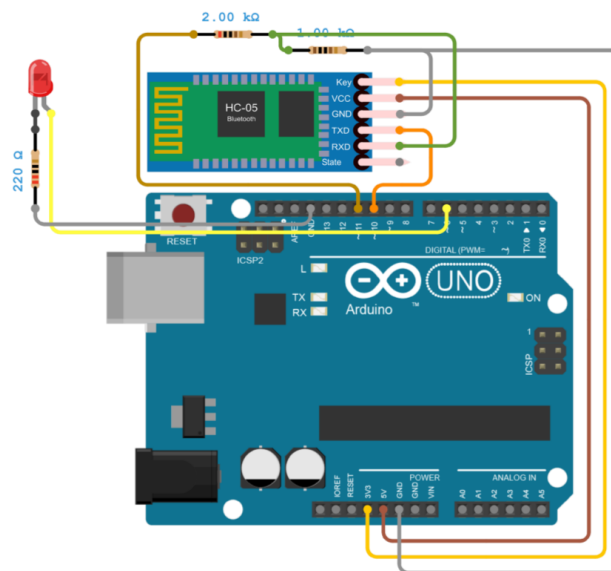
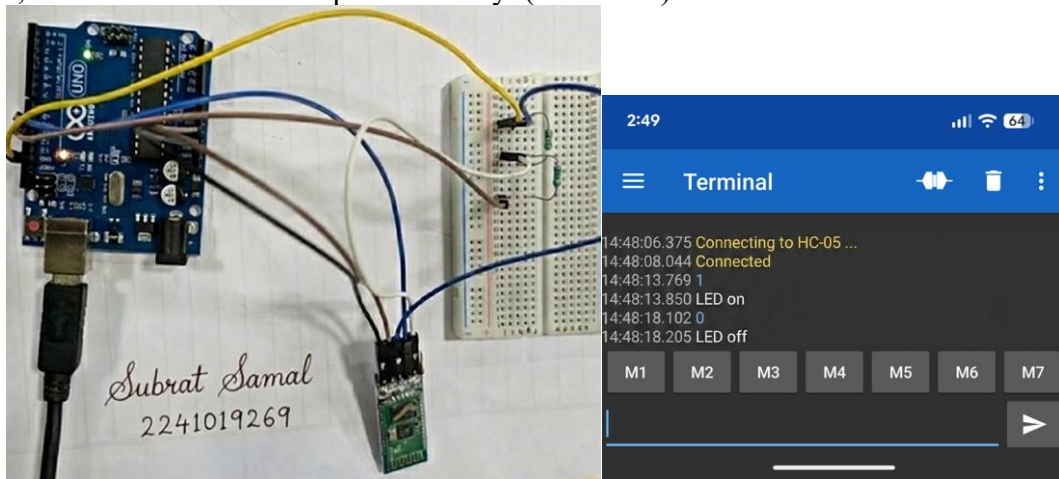


Figure 8: Establish connection with the Serial Monitor and Mobile Phone

### Code

```
#include <SoftwareSerial.h>
SoftwareSerial BT(10, 11); // RX, TX
int led = 6;
void setup() {
  pinMode(led, OUTPUT);
  Serial.begin(9600);
  BT.begin(9600);
  Serial.println("Send '1' to turn ON LED, '0' to turn OFF.");
}
void loop() {
  if (BT.available()) {
    char cmd = BT.read();
    Serial.print("Received: ");
    Serial.println(cmd);
    if (cmd == '1') {
      digitalWrite(led, HIGH);
      BT.println("LED ON");
      Serial.println("LED turned ON");
    }
    else if (cmd == '0') {
      digitalWrite(led, LOW);
      BT.println("LED OFF");
      Serial.println("LED turned OFF");
    }
    else {
      BT.println("Invalid Command");
    }
  }
}
```

Figure 9: Test, record and Measure response latency (Hardware)



## Conclusion

## Precautions

## Post Experiment Questionnaire:

- 1) How did you establish bidirectional communication between Arduino and a smartphone?
- 2) Describe the steps to control an LED using Bluetooth commands.
- 3) What was the maximum reliable communication range achieved during the experiment?
- 4) How did you implement forwarding between Serial Monitor and Bluetooth?
- 5) What challenges did you face while pairing the HC-05/HC-06 module?
- 6) Explain how the Arduino code processes different commands (1 = ON, 0 = OFF).
- 7) How would you modify the setup to control multiple LEDs with different commands?
- 8) Compare the observed LED behavior when using Serial Monitor vs. smartphone commands.

## Answers to Post-Lab Questions

---

**(Signature of the Faculty)**

**Date:** \_\_\_\_\_

---

**(Signature of the Student)**

**Name:** \_\_\_\_\_

**Registration No.:** \_\_\_\_\_

**Branch:** \_\_\_\_\_

**Section** \_\_\_\_\_