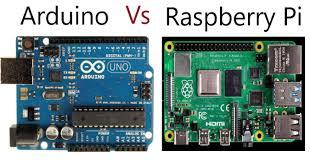
**🔬 Experiment 01 — Arduino and Raspberry Pi Communication**

**🎯 Aim**

To study the basic functionalities of Arduino and Raspberry Pi, and implement serial communication between them.

**📚 Theory**

* **Arduino** is an open-source microcontroller platform ideal for controlling electronic components using simple code and hardware.
* **Raspberry Pi** is a single-board computer that supports Python programming and external device interfacing through GPIO.
* **Serial Communication** allows devices to transmit data one bit at a time, enabling data exchange between Arduino and Raspberry Pi.



**🧰 Materials Required**

* Arduino Uno board
* Raspberry Pi (any model)
* USB cable, Breadboard, Jumper wires
* LED, 220-ohm resistor
* DHT11 sensor
* MicroSD card with Raspbian OS
* Monitor, keyboard, mouse

**🛠️ Procedure**

**Part 1: Arduino Setup**

1. **Connect LED**
   * Anode → pin 9 (via 220Ω resistor), Cathode → GND
2. **Upload Code via Arduino IDE**

void setup() {

Serial.begin(9600); // Initialize serial communication

pinMode(9, OUTPUT); // Set pin 9 as output

}

void loop() {

digitalWrite(9, HIGH); // Turn the LED on

Serial.println("LED ON");

delay(1000); // Wait for a second

digitalWrite(9, LOW); // Turn the LED off

Serial.println("LED OFF");

delay(1000); // Wait for a second

}

**Part 2: Raspberry Pi Setup**

1. Insert SD card and connect monitor, keyboard, and mouse.
2. **Install pySerial Library**

sudo apt-get update

sudo apt-get install python3-serial

1. **Create Python Script**

import serial

# Open serial port

ser = serial.Serial('/dev/ttyACM0', 9600)

while True:

try:

line = ser.readline().decode('utf-8').strip()

print(line)

except KeyboardInterrupt:

break

ser.close()

1. **Run the Script**

python3 read\_serial.py

1. Connect Arduino to Raspberry Pi via USB.

**

**⚙️ Working**

* **Arduino** turns an LED on and off while sending status messages via serial.
* **Raspberry Pi** receives and displays these messages using the Python script.

**🆚 Arduino vs Raspberry Pi (Key Differences)**

| **Feature** | **Arduino** | **Raspberry Pi** |
| --- | --- | --- |
| Type | Microcontroller | Microprocessor |
| Introduced | 2005 | 2012 |
| Architecture | 8-bit | 64-bit |
| RAM | 2 KB | 1 GB |
| Clock Speed | 16 MHz | 1.4 GHz |
| Logic Level | 5V | 3.3V |
| Internet Support | No | Yes (WiFi/Ethernet) |
| OS Requirement | Not required | Required |
| Applications | Timers, Weighing Machines | Robots, Game Servers |

**✅ Conclusion**

This experiment successfully demonstrated how to:

* Operate an Arduino to control an LED
* Set up a Raspberry Pi to read data
* Establish serial communication between the two