DAIOT PRACTICAL 1

Aim:

To study the basics of Arduino and Raspberry Pi and establish serial communication between them.

Theory:

Arduino is an open-source microcontroller platform ideal for hardware control using simple C/C++ programming.

Raspberry Pi is a mini-computer running Linux, suitable for applications requiring OS-level features and programming (e.g., P ython).



Serial Communication allows data exchange between Arduino and Raspberry Pi via USB/serial ports.

Materials:

* Arduino Uno
* Raspberry Pi (with Raspbian OS)
* USB cable, Breadboard, Jumper wires
* LED, 220Ω resistor
* Monitor, keyboard, mouse (for Raspberry Pi)
* DHT11 sensor (optional)

Procedure:

1. Arduino Side:

* Connect an LED to digital pin 9 with a 220Ω resistor to GND.
* Upload this code using Arduino IDE:

void setup() {

Serial.begin(9600);

pinMode(9, OUTPUT);

}

void loop() {

digitalWrite(9, HIGH); Serial.println("LED ON"); delay(1000);

digitalWrite(9, LOW); Serial.println("LED OFF"); delay(1000);

}

2. Raspberry Pi Side:

* Install serial library:

sudo apt-get update

sudo apt-get install python3-serial

* Run Python script:

import serial

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

while True:

try:

print(ser.readline().decode('utf-8').strip())

except KeyboardInterrupt:

break

ser.close()

3. Connect Arduino to Raspberry Pi via USB.

* Open terminal and run the Python script to view messages sent by Arduino.

|  |  |  |
| --- | --- | --- |
| Feature | Arduino | Raspberry Pi |
| Year Introduced | 2005 | 2012 |
| Architecture | Microcontroller (8-bit) | Microprocessor (64-bit) |
| Programming | C/C++ | Python, etc. |
| OS Support | No | Yes(Linux) |
| I/O Power | High | Lower |
| Internet Support | No | Yes(WiFi/Ethernet) |
| Cost | Lower | Higher |

Conclusion:

This experiment demonstrates basic LED control via Arduino and how to receive data using Raspberry Pi through serial communication. It builds foundational skills for integrating microcontrollers with SBCs.