

Practical Report

For IoT Practical



January 1, 2022

Darshan Ramjiyani (DSP)

DOCS, KSKV Kachchh University

* **4.10 – Serial Communication - Move the Mouse Cursor**

Arduino to interact with an application on your computer by moving the

mouse cursor. Perhaps you want to move the mouse position in response to Arduino information**.**

* **Arduino Code:**

const int buttonPin = 2;

const int baudRate = 9600;

const int potXPin = 4;

const int potYPin = 5;

void setup()

{

        /\* Setting pins for input. \*/

        pinMode(buttonPin, INPUT);

        digitalWrite(buttonPin, HIGH);

        /\* Established Serial Communication. \*/

        Serial.begin(baudRate);

        /\* Wait until Serial Communication not established. \*/

        while (!Serial)

        {

        }

        Serial.println("Connected.");

}

void loop()

{

        int x = (512 - analogRead(potXPin)) / 4;

        int y = (512 - analogRead(potYPin)) / 4;

        Serial.print("Data,");

        Serial.print(x, DEC);

        Serial.print(",");

        Serial.print(y, DEC);

        Serial.print(",");

        if (digitalRead(buttonPin) == LOW)

                Serial.print(1);

        else

                Serial.print(0);

        Serial.println(",");

        delay(50);

}

* **Python Code:**

from time import sleep

import pyautogui

import serial

import sys

PORT: str = "com9"

BAUD\_RATE: int = 9600

CONN: serial.Serial = None

pyautogui.FAILSAFE = False

def main():

    setup()

    while True:

        loop()

def setup():

    global BAUD\_RATE, CONN, PORT

    print("[Python] : Connecting Dwaidh terminal to Arduino. Please wait...")

    while True:

        try:

            CONN = serial.Serial(PORT, BAUD\_RATE)

            print(

                f"[Python] : Dwaidh terminal connected to the Arduino via {PORT} at {BAUD\_RATE} bps.")

            break

        except serial.SerialException as e:

            print(

                "[Python] : Can not connect to the port. Try again in 2 mininutes. ", e.args)

            from time import sleep

            sleep(2)

    while True:

        receive\_data: str = CONN.readline().decode("ascii")

        if len(receive\_data) > 0:

            if "Connected" in receive\_data:

                print("[Python] : Arduino has successfully verify the connection.")

                break

            else:

                print(f"[Arduino]: {receive\_data}")

        continue

def loop():

    while True:

        receive\_data: str = CONN.readline().decode("ascii")

        if len(receive\_data) > 0:

            if "Data" in receive\_data:

                x\_coordinate = int(receive\_data.split(",")[1])

                y\_coordinate = int(receive\_data.split(",")[2])

                print(f"[Python] : Moving mouse to ({x\_coordinate},{y\_coordinate})")

                sleep(1)

                try:

                    pyautogui.moveTo(x\_coordinate, y\_coordinate, 3, pyautogui.easeInQuad)

                except KeyboardInterrupt as e:

                    sys.exit()

                sleep(5)

            else:

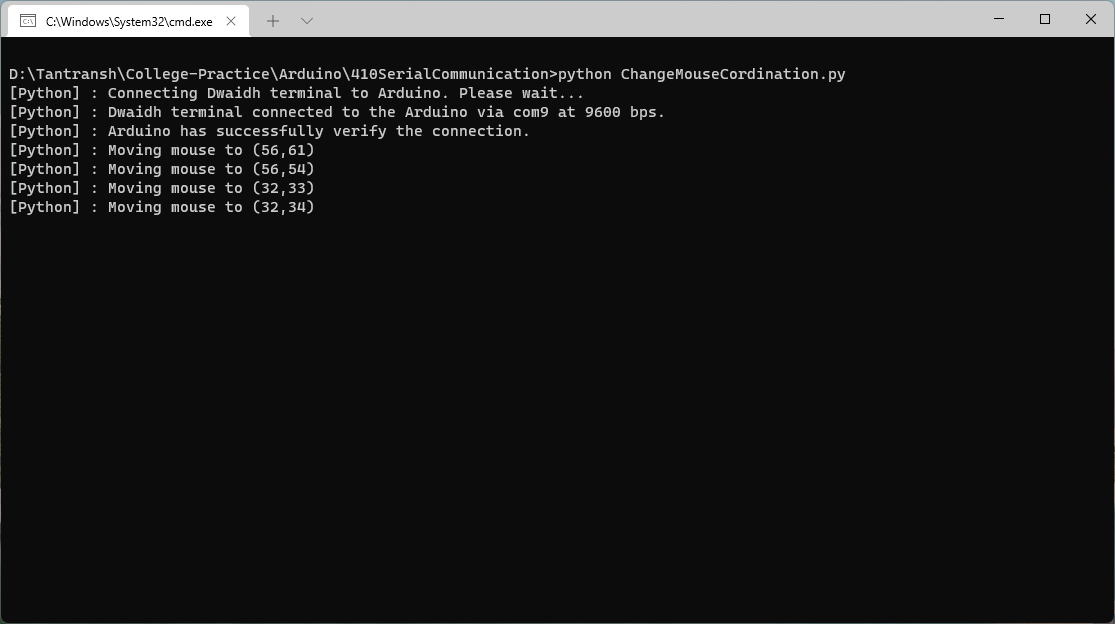
                print(f"[Arduino]: {receive\_data}")

                sleep(5)

if \_\_name\_\_ == "\_\_main\_\_":

    main()

* **Output:**

****