DAILY INTERNSHIP REPORT

DAY 11

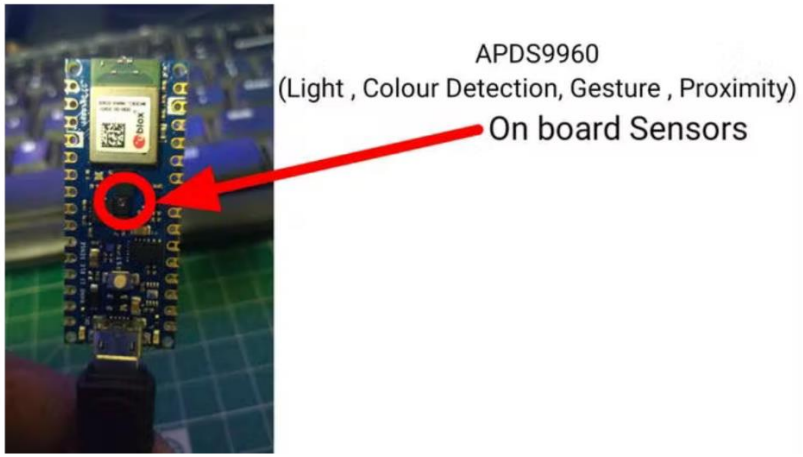
MINI PROJECT: Arduino Nano BLE 33 Sense Game Controller

Introduction

In this project, I made a game controller by using Arduino Nano BLE 33 Sense which has onboard proximity sensor. I played video games by using the Gesture Sensor (APDS9960)

Arduino Nano 33 BLE sense Board has 8 onboard sensor including Gesture Sensor

This Project has **no circuit diagram** because I used an onboard Gesture sensor.



Goals and objectives

- Get acquainted with the onboard sensors on Arduino Nano BLE 33 Sense
- Explore the Arduino_ APDS9960 library which includes the Gesture Sensor feature.
- Exploring how python can be used alongside Arduino
- Exploring the use if Arduino Nano BLE 33 Sense as a keyboard on its own.

Hardware and Software Required

- An Arduino Nano 33 BLE Sense board

- A Micro USB cable to connect the Arduino board to your desktop machine.
- To program the board, I used the Arduino IDE.
- Any compatible IDE such as Microsoft Visual Studio Code
- BlueStacks (Optional)
- Game preferably games like subway surfers.

Procedure

1) First Connect Arduino BLE Sense Board to Your PC or Laptop By using Type B Charging Cable.

2) Now Open Arduino IDE.

3) Then Open Tools --> Manage Libraries --> Search (APDS9960) and Install the Library.

4) After Installing library Go to Files --> Examples --> Arduino APDS9960 -->

Gesture Sensor.

5) Select the Board type and Port Number.

6) UPLOAD the Program.

7) Open the python script.

8) Install The required packages like pyserial and pynput (pip install pyserial, pip install pynput) run this command in the command prompt in administrator mode. These packages are used to make a connection between and Arduino and python and the pynput is used to control and monitor input devices like keyboard and mouse.

9) After executing the python file. open notepad brings the Arduino BLE board close to any object the words **W, A, S, D** will be typed in the note pad without using the keyboard.

10) Don't Forgot to enter the Arduino Port Number.

11) Now run the code in the IDE.

12) For Playing games I used BlueStacks. Go to Control option Select Swipe select the buttons and click the words W, A, S, D. Now the words are assigned for playing games and save the changes

13) Change the Control settings to enable the Arduino act as the controller.

CODE USED FOR ADUINO

```
#include <Arduino_APDS9960.h>
```

```
void setup() {
  Serial.begin(9600);
  while (!Serial);

  if (!APDS.begin()) {
    Serial.println("Error initializing APDS9960 sensor!");
  }

  // for setGestureSensitivity(..) a value between 1 and 100 is required.
  // Higher values makes the gesture recognition more sensible but less accurate
  // (a wrong gesture may be detected). Lower values makes the gesture recognition
  // more accurate but less sensible (some gestures may be missed).
  // Default is 80
  //APDS.setGestureSensitivity(80);


}
void loop() {
  if (APDS.gestureAvailable()) {
    // a gesture was detected, read and print to serial monitor
    int gesture = APDS.readGesture();

    switch (gesture) {
      case GESTURE_UP:
        Serial.println("UP");
        break;

      case GESTURE_DOWN:
        Serial.println("DOWN");
        break;
```

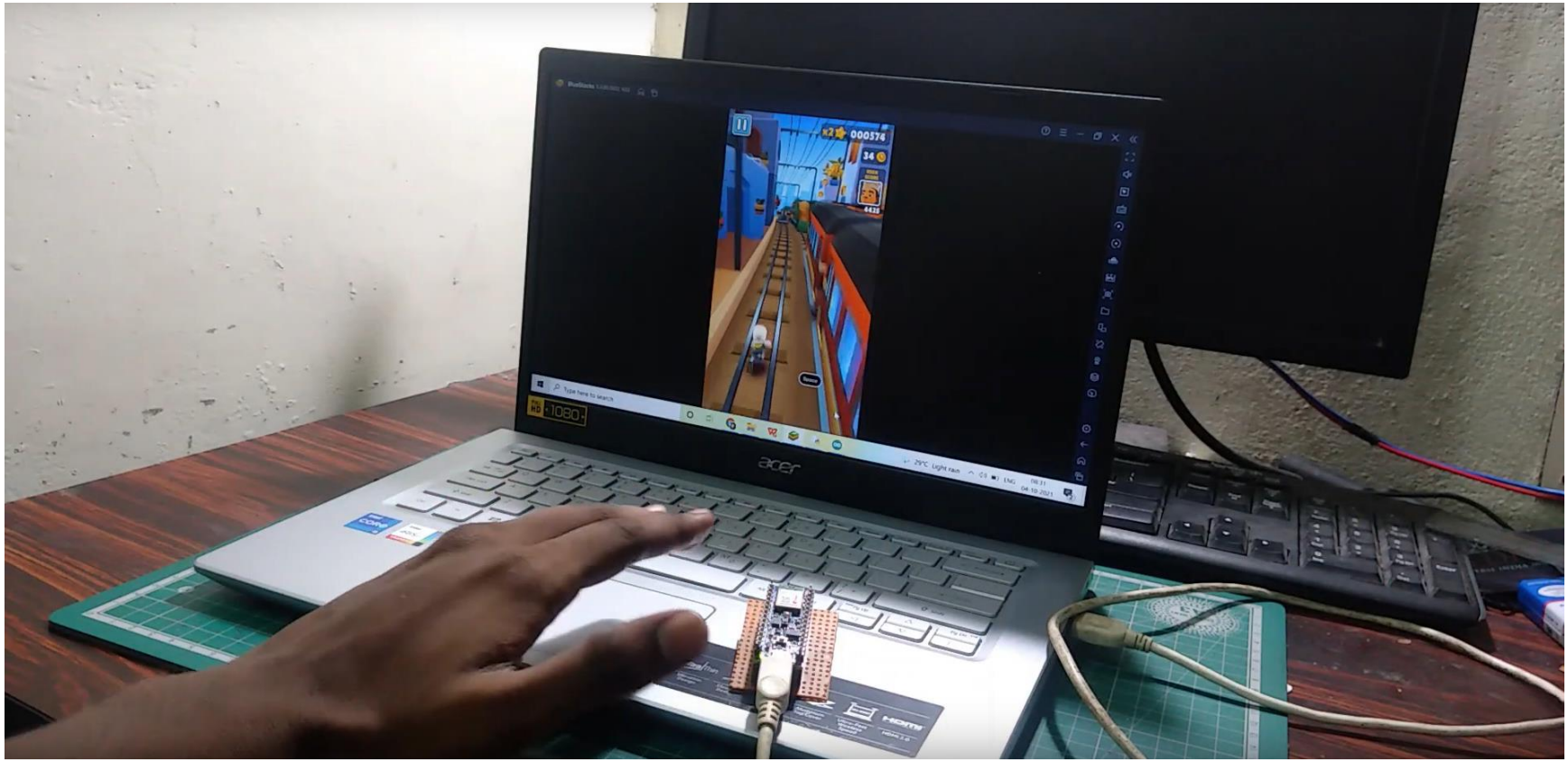
```
      case GESTURE_LEFT:
        Serial.println("LEFT");
        break;

      case GESTURE_RIGHT:
        Serial.println("RIGHT");
        break;

      default:
        // ignore
        break;
    }
  }
}
```

CODE USED FOR PYTHON

```
import serial                                          #Install Pyserial Packages

from pynput.keyboard import Key, Controller            #Install Pynput Packages

ser = serial.Serial('Enter the port number here', 9600)                    #Enter Arduino Port Number

keyboard = Controller()

while True:
    data = ser.readline()

    if data.decode().strip() == "LEFT":
        keyboard.press("a")
        keyboard.release("a")
```

```
    if data.decode().strip() == "RIGHT":
        keyboard.press("d")
        keyboard.release("d")


    if data.decode().strip() == "UP":
        keyboard.press("w")
        keyboard.release("w")


    if data.decode().strip() == "DOWN":
        keyboard.press("s")
        keyboard.release("s")
```

RESULTS



REFERENCES

[1] https://www.hackster.io/prabeenraj01/arduino-nano-ble-33-sense-game-controller-8a9927

[2] https://create.arduino.cc/projecthub/prabeenr2/play-subway-surfers-using-gesture-sensor-9c6f1e?ref=part&ref_id=108462&offset=62