Tinkering Project

• Group Members:
– Ankit Gond
– Anshuman Gupta
– Anusha Khare
– Aryan Sharma(SPOC)
– Dikshant Parasher
– Hemang Tailor

1. Hardware Linkages

The following hardware connections were made before the Arduino code was uploaded to an ESP32 microcontroller.
Here, we go over how we linked the ESP32 to four push buttons and one joystick module.

1.1 Connections for Push Buttons

- **X Button (A):** Attach pin 23 of the ESP32 to one terminal of the push button and the other terminal to a 5V-connected pull-down resistor, usually 10k ohms.

- **Circle Button (B):** Attach the push button's first terminal to pin 22 of the ESP32 and the second terminal to a 5V-connected pull-down resistor (usually 10k ohms).

- **Triangle Button (Y):** Attach the push button's first terminal to pin 1 of the ESP32 and the second terminal to a 5V-connected pull-down resistor (usually 10k ohms).

- **Square Button (X):** Attach the push button's first terminal to pin 3 of the ESP32 and the second terminal to a 5V-connected pull-down resistor (usually 10k ohms).

1.2 Connection of the Joystick Module

We have a push button and two potentiometers for the X and Y axes on the joystick module. The subsequent

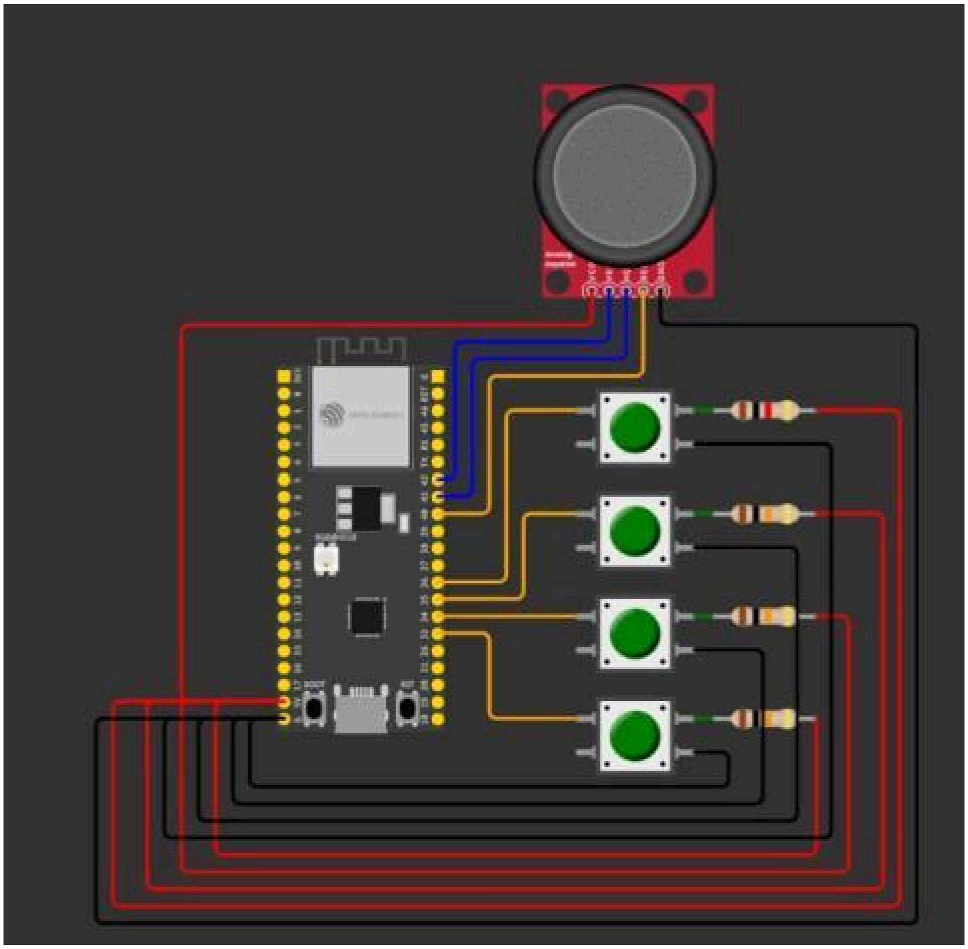
Relationships were formed.

- **Left VRY (Vertical Axis)**: Attach this to the ESP32's pin 15.

- **Pin 4** on the ESP32 should be connected to the left VRX (Horizontal Axis).

- **Joystick Push Button**: Attach this to the ESP32's pin 19.

Additionally, the joystick has a pin for ground and voltage connections, which were made to the GND and 5V pins. pin on the ESP32, in turn.



- Red Wire: Voltage Source
- Black Wire: Ground Connection
- Orange Wire: Digital Input
- Blue Wire: Analog Input

2. Code:-

2.1.1 Defining Buttons

These buttons are defined with pins:

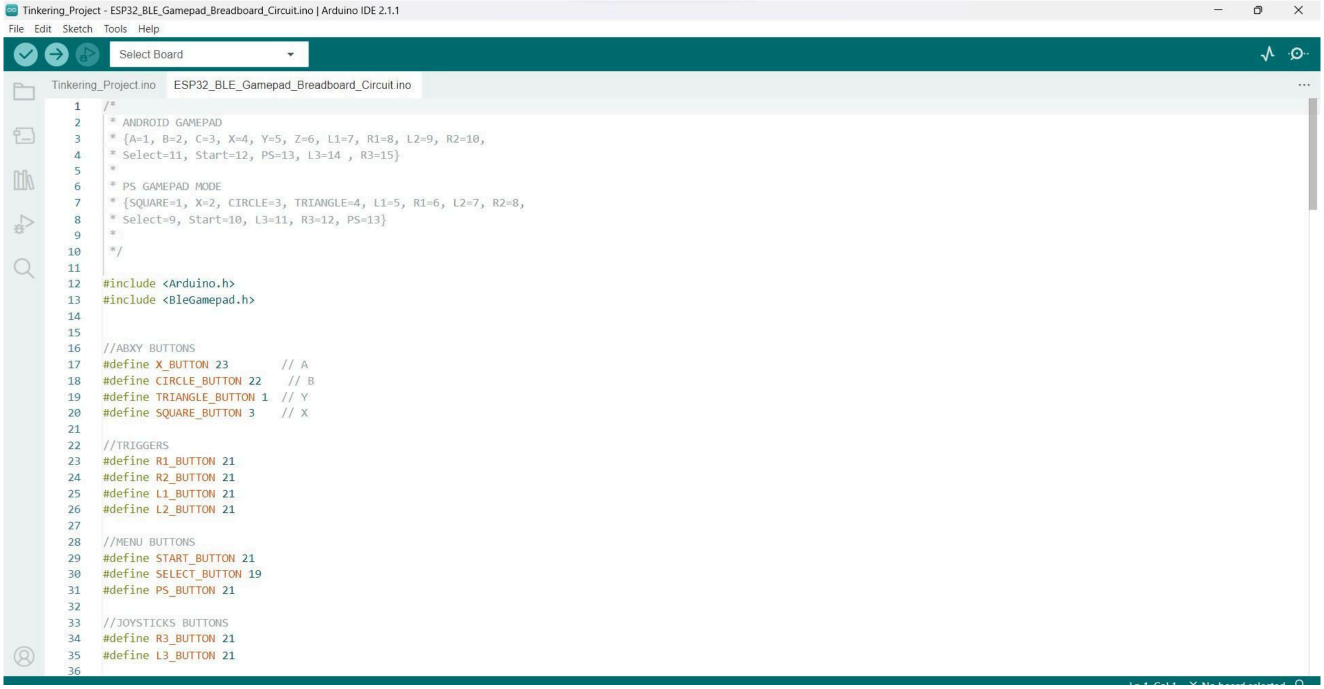
- X BUTTON: Pin 23
- CIRCLE BUTTON: Pin 22
- TRIANGLE BUTTON: Pin 1
- SQUARE BUTTON: Pin 3
- SELECT BUTTON: Pin 19

2.1.2 Joysticks

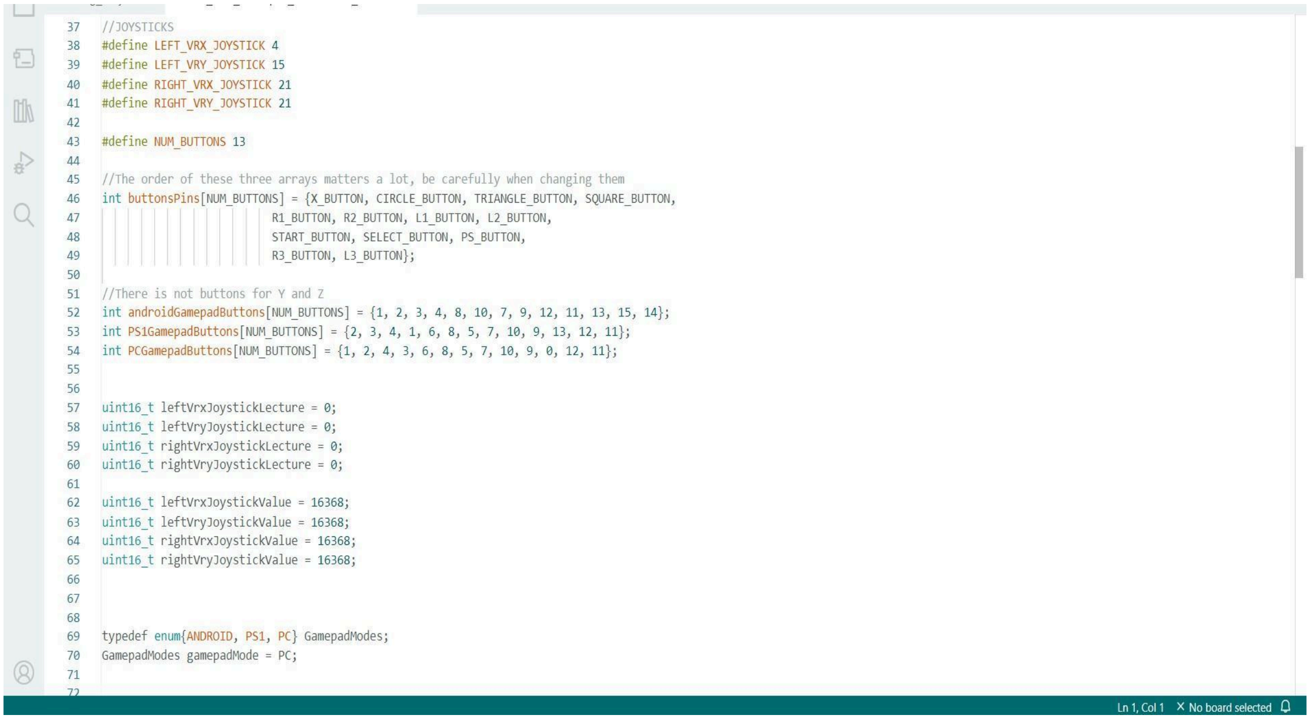
The code defines the pins for the joystick inputs:

- ‘LEFT VRX JOYSTICK‘: Pin 4
- ‘LEFT VRY JOYSTICK‘: Pin 15
- ‘RIGHT VRX JOYSTICK‘: Pin 0
- ‘RIGHT VRY JOYSTICK‘: Pin 0

Full code:-

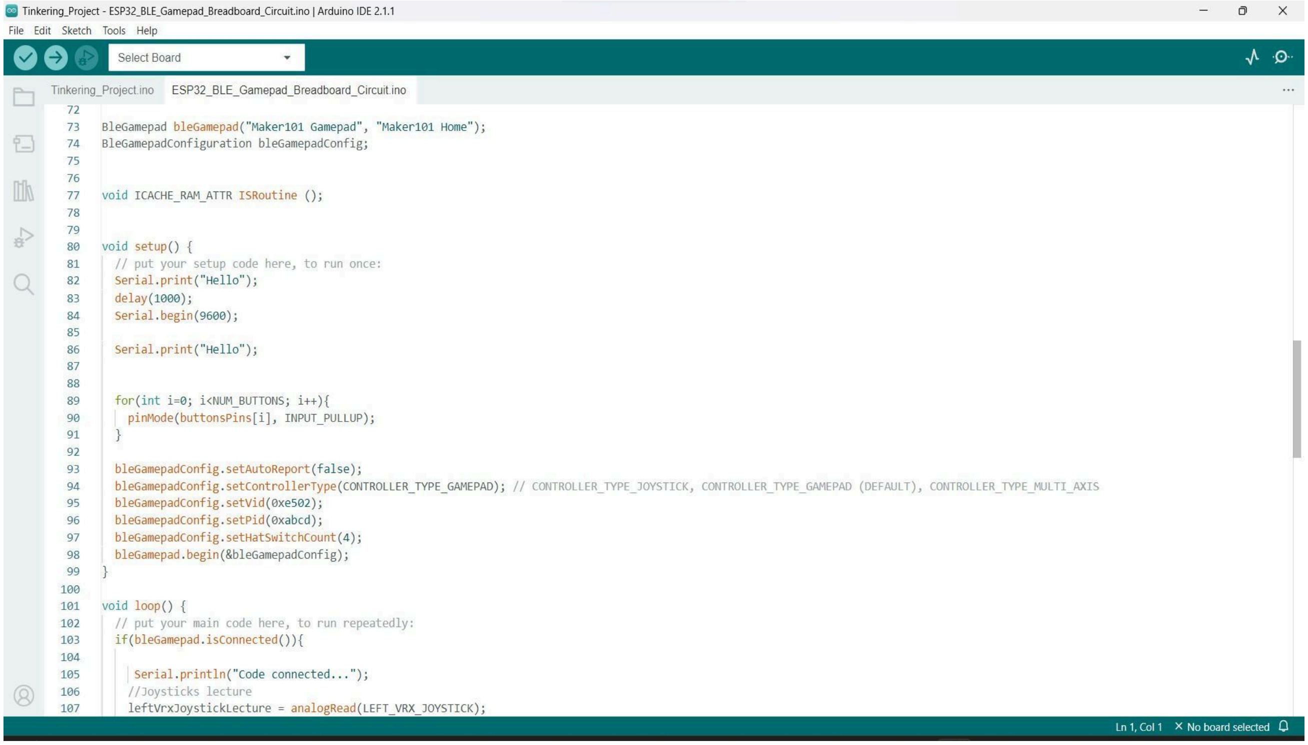

Tinkering_Project - ESP32_BLE_Gamepad_Breadboard_Circuit.ino | Arduino IDE 2.1.1
File Edit Sketch Tools Help
Select Board
Tinkering_Project.ino
ESP32_BLE_Gamepad_Breadboard_Circuit.ino

```
/*
 * ANDROID GAMEPAD
 * {A=1, B=2, C=3, X=4, Y=5, Z=6, L1=7, R1=8, L2=9, R2=10,
 * Select=11, Start=12, PS=13, L3=14 , R3=15}
 *
 * PS GAMEPAD MODE
 * {SQUARE=1, X=2, CIRCLE=3, TRIANGLE=4, L1=5, R1=6, L2=7, R2=8,
 * Select=9, Start=10, L3=11, R3=12, PS=13}
 *
 */

#include <Arduino.h>
#include <BleGamepad.h>


//ABXY BUTTONS
#define X_BUTTON 23        // A
#define CIRCLE_BUTTON 22    // B
#define TRIANGLE_BUTTON 1  // Y
#define SQUARE_BUTTON 3    // X

//TRIGGERS
#define R1_BUTTON 21
#define R2_BUTTON 21
#define L1_BUTTON 21
#define L2_BUTTON 21

//MENU BUTTONS
#define START_BUTTON 21
#define SELECT_BUTTON 19
#define PS_BUTTON 21

//JOYSTICKS BUTTONS
#define R3_BUTTON 21
#define L3_BUTTON 21

```



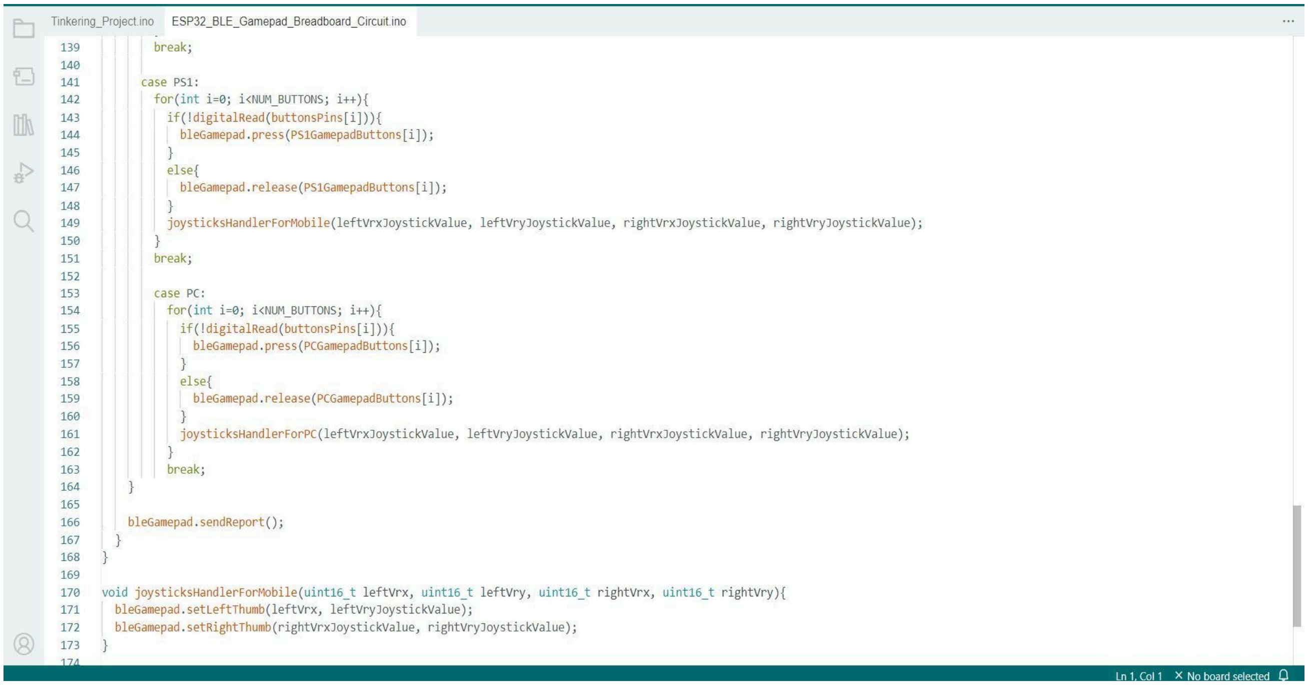
```
//JOYSTICKS
#define LEFT_VRX_JOYSTICK 4
#define LEFT_VRY_JOYSTICK 15
#define RIGHT_VRX_JOYSTICK 21
#define RIGHT_VRY_JOYSTICK 21

#define NUM_BUTTONS 13

//The order of these three arrays matters a lot, be carefully when changing them
int buttonsPins[NUM_BUTTONS] = {X_BUTTON, CIRCLE_BUTTON, TRIANGLE_BUTTON, SQUARE_BUTTON,
                                R1_BUTTON, R2_BUTTON, L1_BUTTON, L2_BUTTON,
                                START_BUTTON, SELECT_BUTTON, PS_BUTTON,
                                R3_BUTTON, L3_BUTTON};

//There is not buttons for Y and Z
int androidGamepadButtons[NUM_BUTTONS] = {1, 2, 3, 4, 8, 10, 7, 9, 12, 11, 13, 15, 14};
int PS1GamepadButtons[NUM_BUTTONS] = {2, 3, 4, 1, 6, 8, 5, 7, 10, 9, 13, 12, 11};
int PCGamepadButtons[NUM_BUTTONS] = {1, 2, 4, 3, 6, 8, 5, 7, 10, 9, 0, 12, 11};


uint16_t leftVrxJoystickLecture = 0;
uint16_t leftVryJoystickLecture = 0;
uint16_t rightVrxJoystickLecture = 0;
uint16_t rightVryJoystickLecture = 0;

uint16_t leftVrxJoystickValue = 16368;
uint16_t leftVryJoystickValue = 16368;
uint16_t rightVrxJoystickValue = 16368;
uint16_t rightVryJoystickValue = 16368;



typedef enum{ANDROID, PS1, PC} GamepadModes;
GamepadModes gamepadMode = PC;
```

Ln 1, Col 1
✕ No board selected

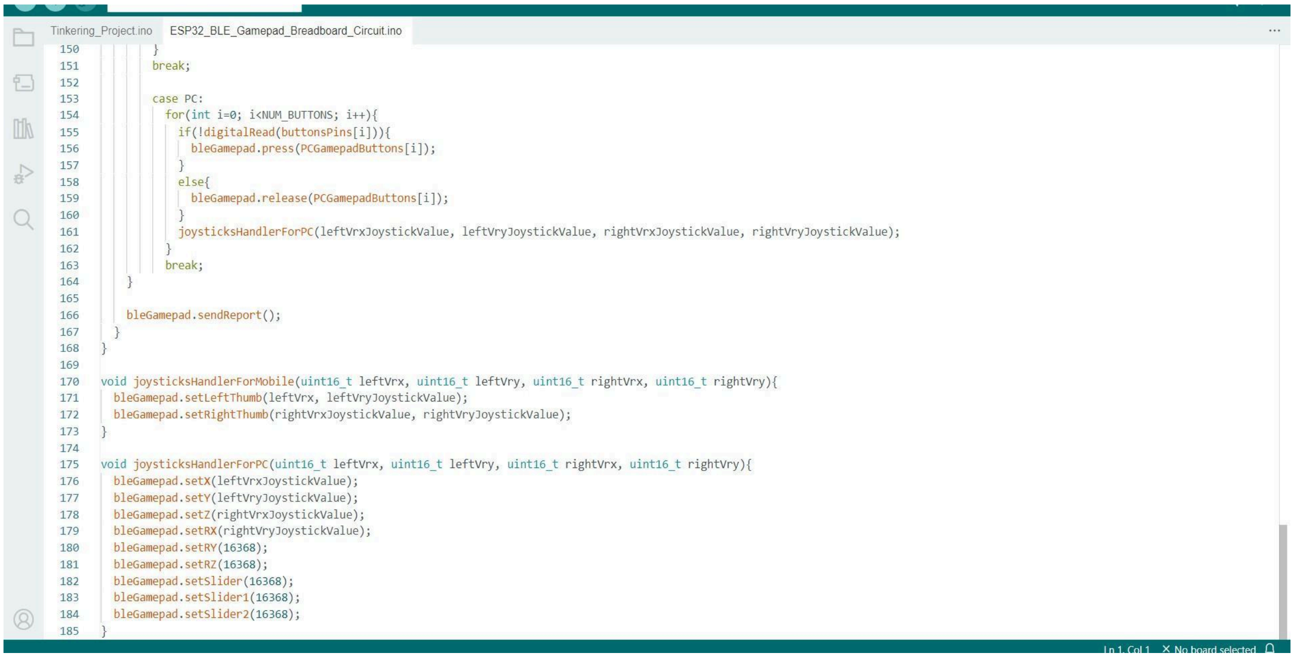


Tinkering_Project - ESP32_BLE_Gamepad_Breadboard_Circuit.ino | Arduino IDE 2.1.1
File
Edit
Sketch
Tools
Help
Select Board
Tinkering_Project.ino
ESP32_BLE_Gamepad_Breadboard_Circuit.ino

```

BleGamepad bleGamepad("Maker101 Gamepad", "Maker101 Home");
BleGamepadConfiguration bleGamepadConfig;


void ICACHE_RAM_ATTR ISRoutine ();


void setup() {
  // put your setup code here, to run once:
  Serial.print("Hello");
  delay(1000);
  Serial.begin(9600);

  Serial.print("Hello");


  for(int i=0; i<NUM_BUTTONS; i++){
    pinMode(buttonsPins[i], INPUT_PULLUP);
  }

  bleGamepadConfig.setAutoReport(false);
  bleGamepadConfig.setControllerType(CONTROLLER_TYPE_GAMEPAD); // CONTROLLER_TYPE_JOYSTICK, CONTROLLER_TYPE_GAMEPAD (DEFAULT), CONTROLLER_TYPE_MULTI_AXIS
  bleGamepadConfig.setVid(0xe502);
  bleGamepadConfig.setPid(0xabcd);
  bleGamepadConfig.setHatSwitchCount(4);
  bleGamepad.begin(&bleGamepadConfig);
}

void loop() {
  // put your main code here, to run repeatedly:
  if(bleGamepad.isConnected()){

    Serial.println("Code connected...");
    //Joysticks lecture
    leftVrxJoystickLecture = analogRead(LEFT_VRX_JOYSTICK);
```



```
        break;

      case PS1:
        for(int i=0; i<NUM_BUTTONS; i++){
          if(!digitalRead(buttonsPins[i])){
            bleGamepad.press(PS1GamepadButtons[i]);
          }
          else{
            bleGamepad.release(PS1GamepadButtons[i]);
          }
          joysticksHandlerForMobile(leftVrxJoystickValue, leftVryJoystickValue, rightVrxJoystickValue, rightVryJoystickValue);
        }
        break;

        case PC:
          for(int i=0; i<NUM_BUTTONS; i++){
            if(!digitalRead(buttonsPins[i])){
              bleGamepad.press(PCGamepadButtons[i]);
            }
            else{
              bleGamepad.release(PCGamepadButtons[i]);
            }
            joysticksHandlerForPC(leftVrxJoystickValue, leftVryJoystickValue, rightVrxJoystickValue, rightVryJoystickValue);
          }
          break;
    }

    bleGamepad.sendReport();
  }
}

void joysticksHandlerForMobile(uint16_t leftVrx, uint16_t leftVry, uint16_t rightVrx, uint16_t rightVry){
  bleGamepad.setLeftThumb(leftVrx, leftVryJoystickValue);
  bleGamepad.setRightThumb(rightVrxJoystickValue, rightVryJoystickValue);
}
```



```
        }
        break;

        case PC:
          for(int i=0; i<NUM_BUTTONS; i++){
            if(!digitalRead(buttonsPins[i])){
              bleGamepad.press(PCGamepadButtons[i]);
            }
            else{
              bleGamepad.release(PCGamepadButtons[i]);
            }
            joysticksHandlerForPC(leftVrxJoystickValue, leftVryJoystickValue, rightVrxJoystickValue, rightVryJoystickValue);
          }
          break;
    }

    bleGamepad.sendReport();
  }
}

void joysticksHandlerForMobile(uint16_t leftVrx, uint16_t leftVry, uint16_t rightVrx, uint16_t rightVry){
  bleGamepad.setLeftThumb(leftVrx, leftVryJoystickValue);
  bleGamepad.setRightThumb(rightVrxJoystickValue, rightVryJoystickValue);
}

void joysticksHandlerForPC(uint16_t leftVrx, uint16_t leftVry, uint16_t rightVrx, uint16_t rightVry){
  bleGamepad.setX(leftVrxJoystickValue);
  bleGamepad.setY(leftVryJoystickValue);
  bleGamepad.setZ(rightVrxJoystickValue);
  bleGamepad.setRX(rightVryJoystickValue);
  bleGamepad.setRY(16368);
  bleGamepad.setRZ(16368);
  bleGamepad.setSlider(16368);
  bleGamepad.setSlider1(16368);
  bleGamepad.setSlider2(16368);
}
```

3.Final Hardware:-