ROBOTICS CLUB
UNIVERSITI MALAYSIA SABAH

CUSTOMIZABLE CONTROLLER

WORKSHOP

ROBOTICS CLUB
UNIVERSITI MALAYSIA SABAH

CONTENT

Electrical & Electronics (EE)

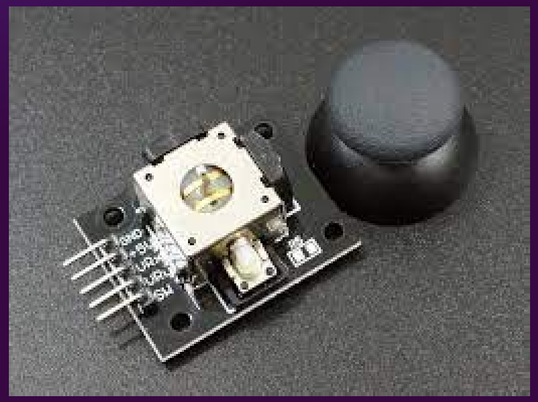
Components
Schematics
Prototyping

Programming

Initializations & Libraries
Transmitting & Receiving
Data reception

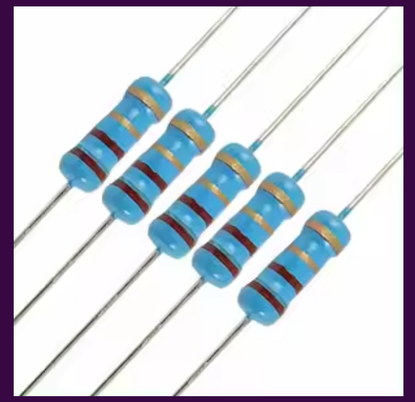
ROBOTICS CLUB
UNIVERSITI MALAYSIA SABAH

EE Components



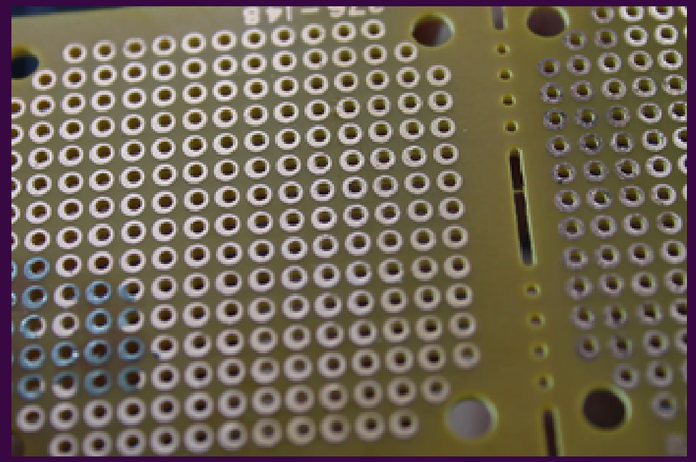
Joystick

Tactile buttons



Resistors 4.7k

Microcontroller



Perfboard/donut board

Soldering supplies

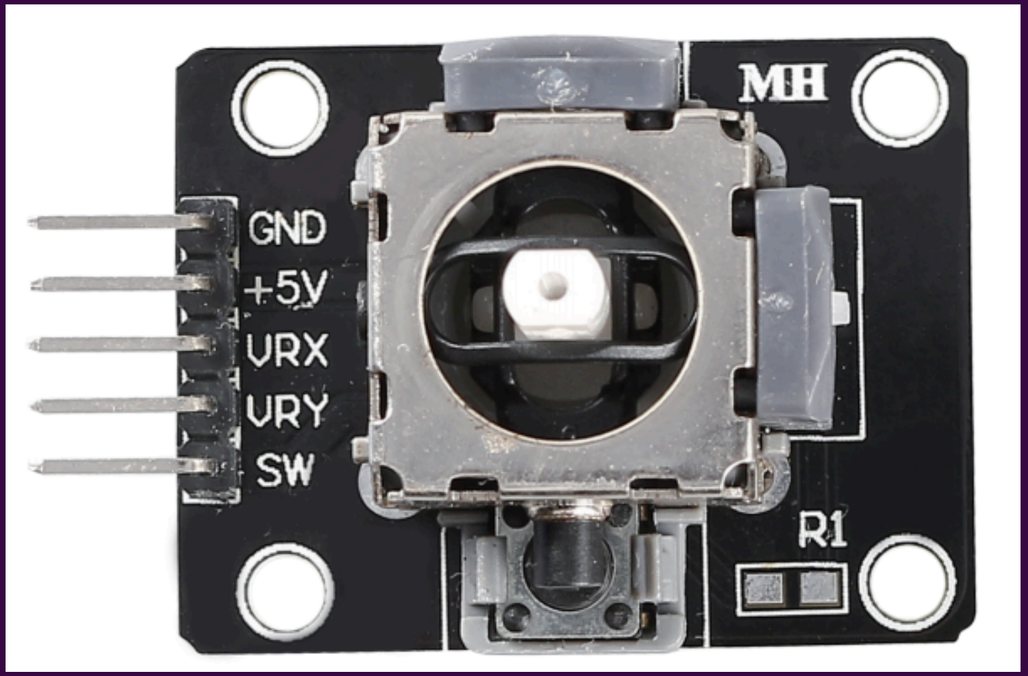
ROBOTICS CLUB
UNIVERSITI MALAYSIA SABAH

EE

Schematic Connections

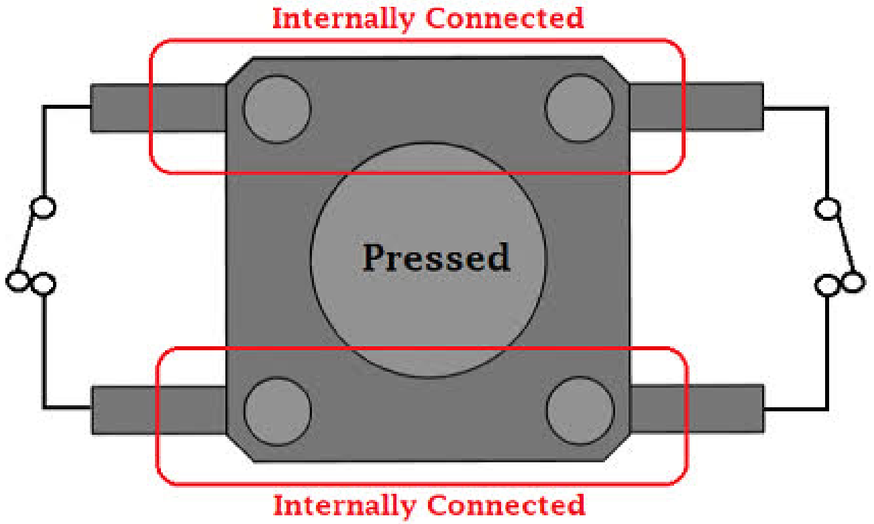
MUST refer which pins can receive input & which can give output

Joystick	Analog input pin
Button	Analog input pin




esp32io.com

esp32io.com

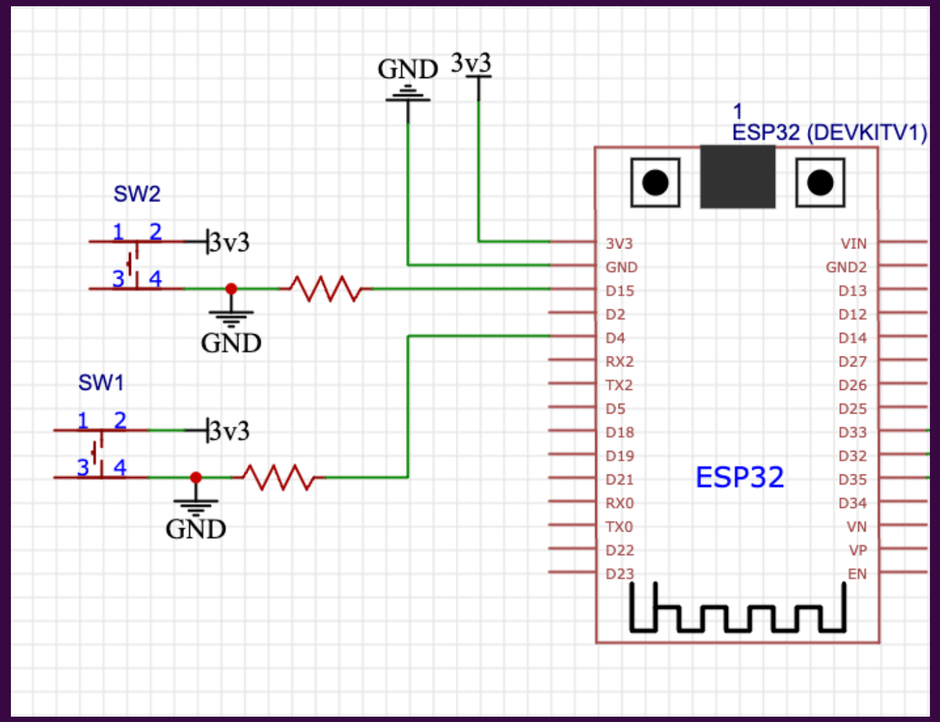

Internally Connected
Pressed
Internally Connected

ROBOTICS CLUB
UNIVERSITI MALAYSIA SABAH

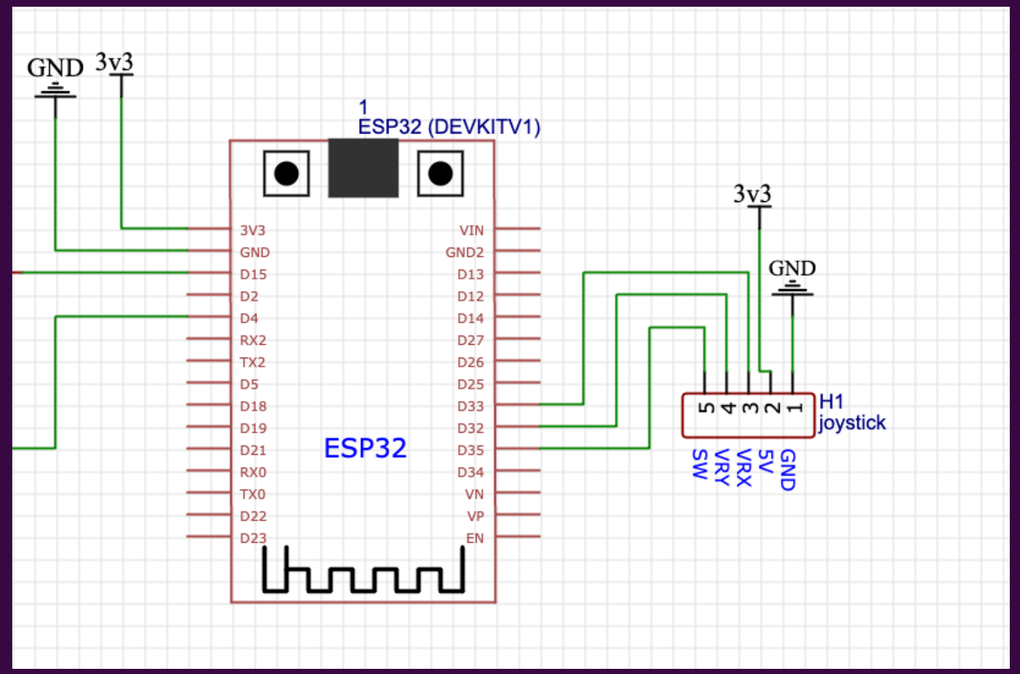
EE

Schematic Connections

Switch schematic



Joystick schematic

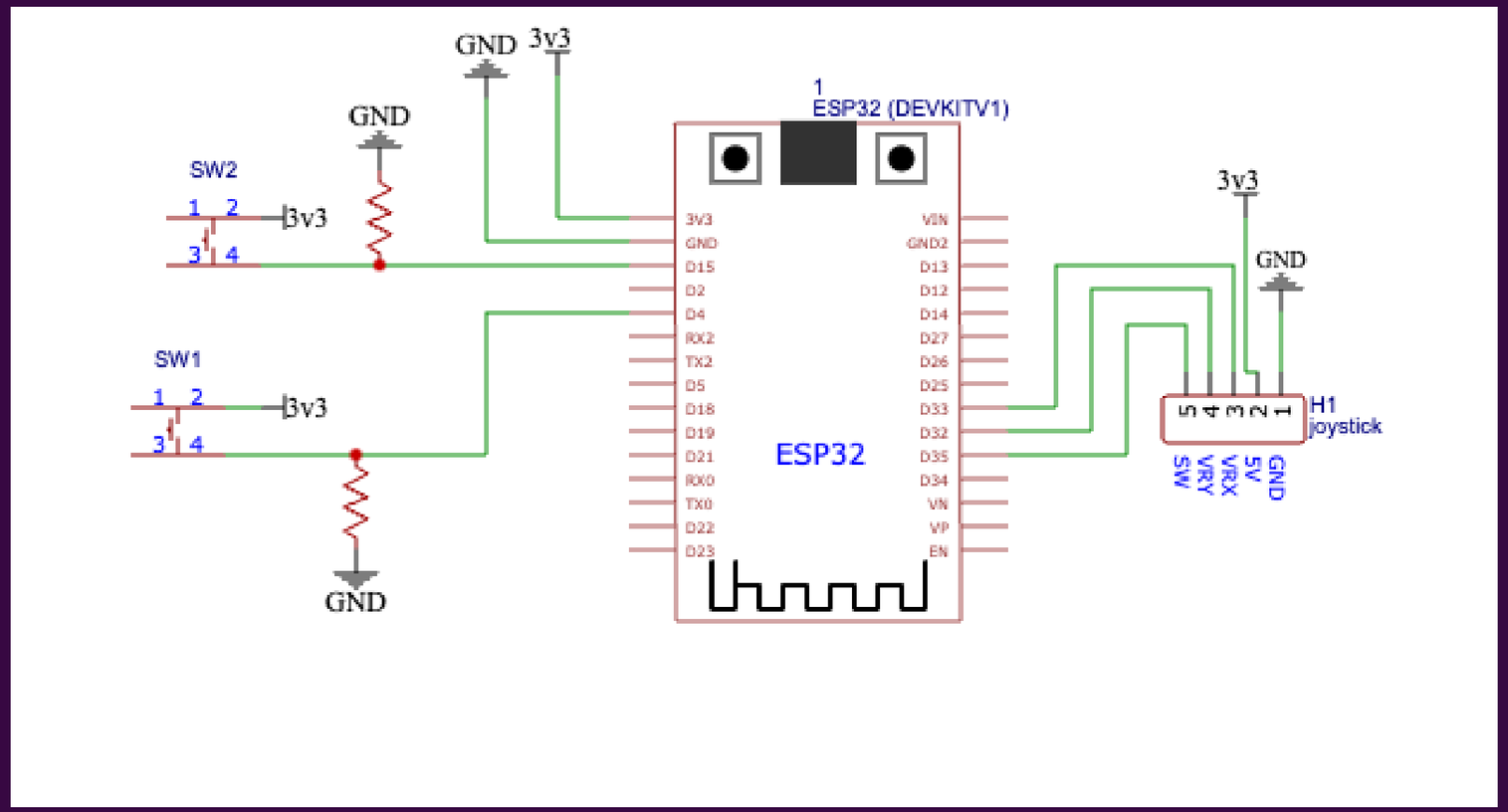


ROBOTICS CLUB
UNIVERSITI MALAYSIA SABAH

EE

Prototyping

FINAL SCHEMATIC

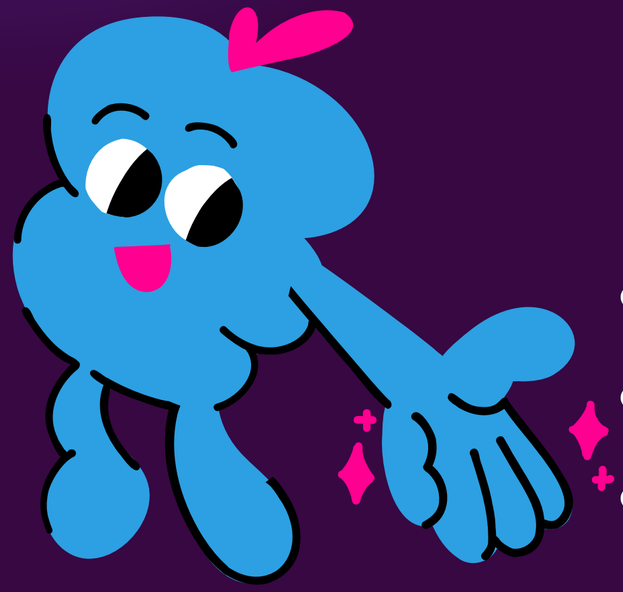


SOLDERING TIME!

(after breadboard)

PROGRAMMING

ROBOTICS CLUB
UNIVERSITI MALAYSIA SABAH



WHAT DO YOU NEED?

- 2 ESP32(RECEIVER & TRANSMITTER)
- ARDUINO IDE
- ESP32 MAC ADDRESS
- 2 SKETCH

SETUP YOUR IDE

setp 1 : setup esp32 borad

arduinio -> File -> preference -> paste the link

setp 2 : install ESP32 board library

arduinio -> board manager

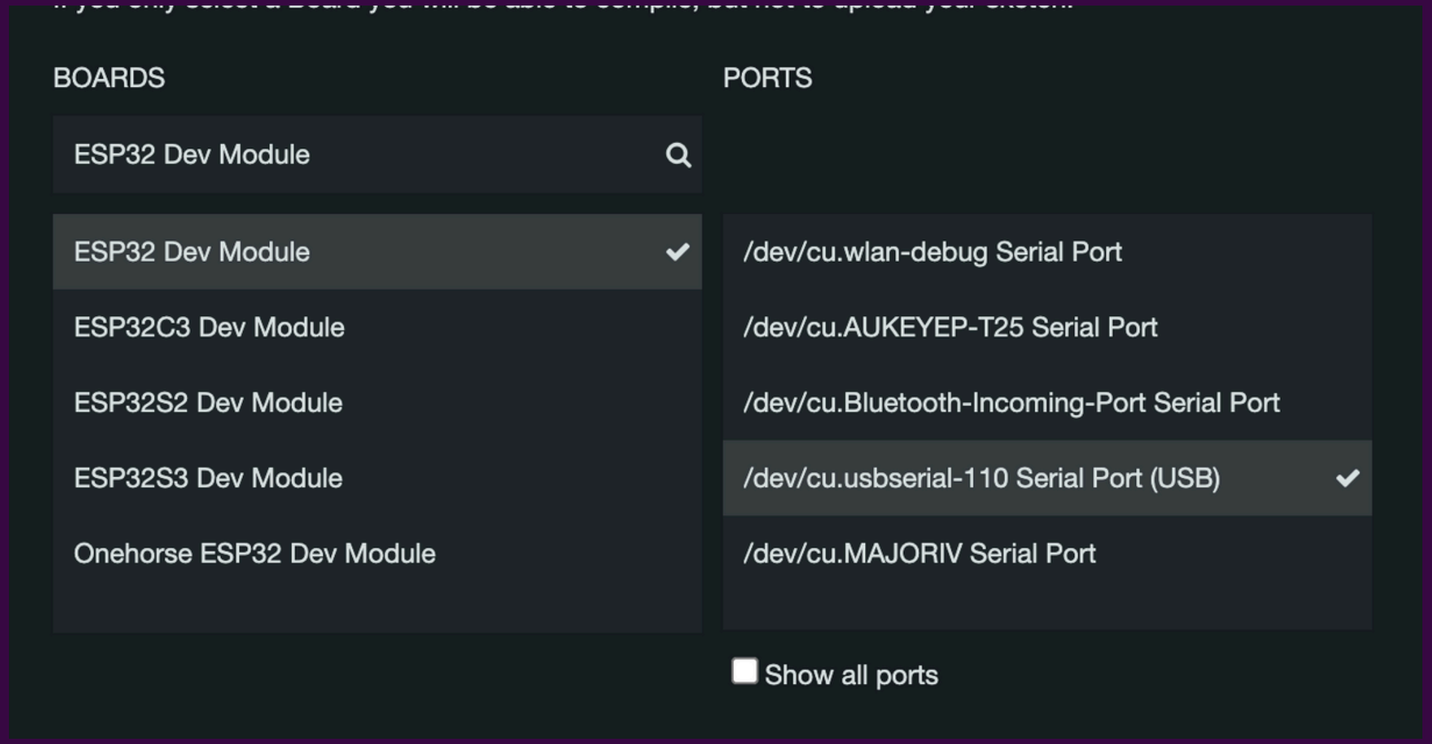
- seacrh 'esp32' by espressif system

setp 3 : download "ezButton" library at library manager

```
#include "WiFi.h"
#include "esp_now.h"
#include <ezButton.h>

```

setp 4: select the right port and “ESP32 Dev Module” board



CUSTOMIZABLE CONTROLLER

PROGRAMMING

Initializations & libraries

ROBOTICS CLUB
UNIVERSITI MALAYSIA SABAH

LIBRARIES

```
#include "WiFi.h"
#include "esp_now.h"
#include <ezButton.h>
```

Controller sketch (TRANSMIT)

PIN DEFINITIONS

```
#define VRX ?
#define VRY ?
#define SW  ?
#define redB ?
#define whiteB ?
```

```
ezButton button(SW);
```

RECEIVER ADDRESS (HAVE TO CHECK)

```
uint8_t ReceiverAddress[] = {0x08, 0xD1, 0xF9, 0x3B, 0x0B, 0xA8};
```

STEP 2
(TRANSMIT)

PROGRAMMING

Setup & Algorithm

ROBOTICS CLUB
UNIVERSITI MALAYSIA SABAH

1. get the receiver ESP's mac address

esp_macAddress.ino

```
1   #include "WiFi.h"
2   void setup() {
3     Serial.begin(115200);
4     WiFi.mode(WIFI_MODE_STA);
5     Serial.println(WiFi.macAddress());
6   }
7
8   void loop() {
9
10  }
11
```

CUSTOMIZABLE CONTROLLER

STEP 3
(TRANSMIT)

PROGRAMMING
Setup & Algorithm

ROBOTICS CLUB
UNIVERSITI MALAYSIA SABAH

3. create a new sketch , to transmit data from controller to the esp32

```
#include "WiFi.h"
#include "esp_now.h"
#include <ezButton.h>

#define VRX 35
#define VRY 32
#define SW 17
#define redB 2
#define whiteB 15
ezButton button(SW);

int stateRed;
int stateWhite;
int stateSwitch;
int valueVRX=0;
int valueVRY=0;

//address of the receiver esp32
uint8_t ReceiverAddress[] = {0x08, 0xD1, 0xF9, 0x3B, 0x0B, 0xA8};

//here all the variable that will be send to esp
typedef struct struct_message {
  int redValue;
  int whiteValue;
  int valueX;
  int valueY;
  int swValue;
}struct_message;
struct_message senderData;

esp_now_peer_info_t peerInfo;

// function that will indicate the message was sent to the receiver esp
void OnDataSent(const uint8_t *mac_addr, esp_now_send_status_t status){
  Serial.print("\r\nLast Packaet Send Status: \t");
  Serial.println(status == ESP_NOW_SEND_SUCCESS ? " DELIVERY SUCCESS" : "Delivery Fail");
 //delay(1000);
}
```

```
40  void setup() {
41    Serial.begin(115200);
42    pinMode(redB, INPUT);
43    pinMode(whiteB, INPUT);
44    pinMode(VRX, INPUT);
45    pinMode(VRY, INPUT);
```

void setup:

- SETUP THE PIN (INPUT)
- INITIALIZE THE ESP COMMUNICATION
- CALL THE SENT DATA FUNCTION
- REGISTER PEER ADDRESS
- CHECK IF FAIL TO PEER

```
48  WiFi.mode(WIFI_STA);
49
50  if(esp_now_init() != ESP_OK){
51    Serial.println("Error initializing ESP_NOW");
52    return;
53  }
54  esp_now_register_send_cb(OnDataSent);
55
56    // Register peer
57    memcpy(peerInfo.peer_addr, ReceiverAddress, 6);
58    peerInfo.channel = 0;
59    peerInfo.encrypt = false;
60
61
62    if (esp_now_add_peer(&peerInfo) != ESP_OK) {
63      Serial.println("Failed to add peer");
64      return;
65    }
```

```
72    stateSwitch = button.getState();
73    senderData.swValue = stateSwitch;
74  //stateRed is the value from the button, wi
75    stateRed = digitalRead(redB);
76    senderData.redValue = stateRed;
77
78    stateWhite = digitalRead(whiteB);
79    senderData.whiteValue = stateWhite;
80
81    valueVRX = analogRead(VRX);
82    senderData.valueX = valueVRX;
83    valueVRY = analogRead(VRY);
84    senderData.valueY = valueVRY;
```

void loop:

- SENDING THE INPUT DATA TO THE RECEIVER ESP BY USING STRUCT “SENDERDATA”
- THE DATA PASS DOWN TO THE RESULT FUNCTION TO BE SENT TO THE RECEIVER ESP32.
- CHECK IF THE TRANSMITING IS SUCCESS

```
93  //this function is the one sending data to the receiver esp.
94   esp_err_t result = esp_now_send(ReceiverAddress, (uint8_t *) &senderData, sizeof(senderData));
95
96   if(result == ESP_OK){
97     Serial.println("Sent with Success");
98    //delay(1000);
99   }
100  else {
101    Serial.println("Error sending data");
102    //delay(1000);
103  }
```

PROGRAMMING

Initializations & libraries

ROBOTICS CLUB
UNIVERSITI MALAYSIA SABAH

LIBRARIES

```
#include "WiFi.h"
#include "esp_now.h"
#include <ESP32Servo.h>
```

Receiver sketch

PIN DEFINITIONS

```
#define m1 27
#define m2 26
#define m3 14
#define m4 12
#define enA 13
#define enB 25
#define servo 4
```

RECEIVER ADDRESS : {0X08, 0XD1, 0XF9, 0X3B, 0X0B, 0XA8};

```
Servo myservo;
//make sure all the variable here same with sender code
typedef struct struct_message {
  int redValue;
  int whiteValue;
  int valueX;
  int valueY;
  int swValue;
}struct_message;

//reading data message
struct_message readingData;

int vrx = 0;
int vry = 0;
int turn_left;
int turn_right;
int val;
int c = 0;
```

- REWRITE THE STRUCT VARIABLE FROM THE TRANSIMITTER CODE
- ADD ON VARIABLE TO READ THE TRAMISTTING DATA
- USE READINGDATA TO GET DATA FROM CONTROLLER

```
//function that receive incomming data from the sender esp.
void OnDataRecv(const uint8_t * mac, const uint8_t *incomingData, int len){
  memcpy(&readingData, incomingData, sizeof(readingData));

  vry = readingData.valueY;
  vrx = readingData.valueX;
  turn_left = readingData.whiteValue;
  turn_right = readingData.redValue;
  val = readingData.swValue;
```

```
05  void setup() {
06    //setup all the mode here
07    Serial.begin(115200);
08    pinMode(m1, OUTPUT);
09    pinMode(m2, OUTPUT);
10    pinMode(m3, OUTPUT);
11    pinMode(m4, OUTPUT);
12    pinMode(enA, OUTPUT);
13    pinMode(enB, OUTPUT);
14    myservo.attach(servo);
15
16    myservo.write(0);
17    WiFi.mode(WIFI_STA);
18    if (esp_now_init() != ESP_OK) {
19      Serial.println("Error initializing ESP-NOW");
20      return;
21    }
22    esp_now_register_recv_cb(OnDataRecv);
23  }
```

void setup:

- SETUP THE PIN (OUTPUT)
- CALL THE RECEIVE DATA FUNCTION TO READ THE DATA

```
25  void loop() {
26    // the main loop will be empty since all the receive data loopiing at the sender esp
27  }
```

- LET THE VOID LOOP EMPTY

ROBOTICS CLUB
UNIVERSITI MALAYSIA SABAH

THE END

visit this github to view full project

Thank You