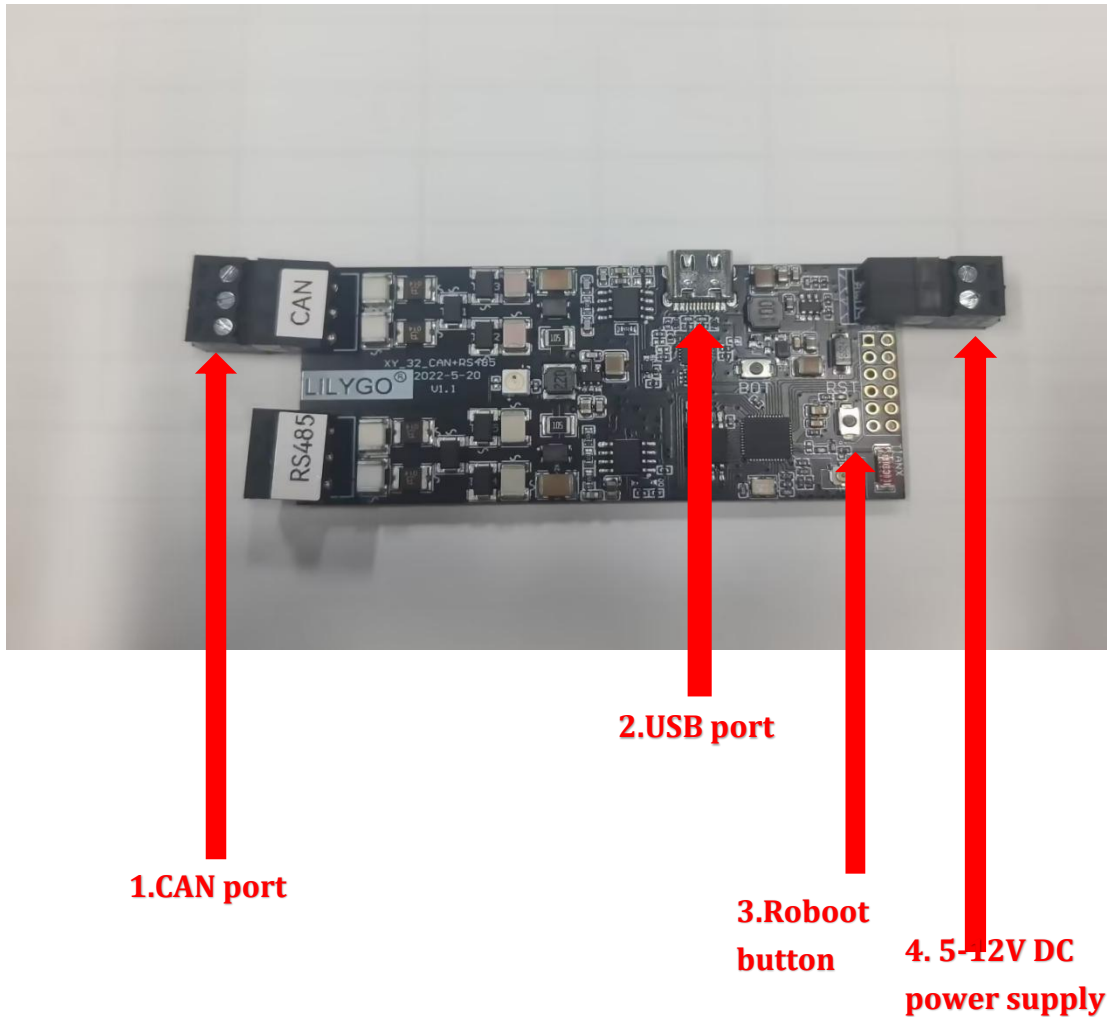


ESP32 Layout Description PINMAP



1. CAN Interface

Connects to the robotic arm for CAN communication. The back of the board is labeled with CAN-H (yellow wire) and CAN-L (white wire).

2. USB Interface

Used for uploading the program to the ESP32 board (Type-C interface). You may keep this connection for powering the board via your PC after uploading the code. If you use a DC power supply (see “power supply interface” below) to power on the board, this USB cable can be unplugged after the program is uploaded.

3. Reset Development Board

Pressing this button will reset the program burned into the development board.

4. Power Supply Interface

Used to power the development board, requires a voltage supply of 5~12V. (Note:

ESP32 Instructions (Robotic Arm Version)

1. Upload the code required to control the robotic arm into the corresponding development board, ensure all connections are correct, and turn on the power.
2. Modify the Wi-Fi connection setup for the ESP32 in the code to ensure it is on the same WiFi as the control computer.

```
const char *ssid = "Picasso_OMNI3"; // should be connected Wi-Fi's name
const char *password = "Picasso1234"; // should be connected Wi-Fi's password
const char *hostname = "espressif";
IPAddress local_IP(192, 168, 1, 211); // static ip address to be assigned to ESP32 board
IPAddress gateway(192, 168, 1, 0); // Gateway of the connected network
IPAddress subnet(255, 255, 255, 0); // subnet code of connected network

WebSocketsServer webSocket = WebSocketsServer(81); // websocket port
uint16_t status;
```

3. Modify the IP address in the webpage code (.html file) to control the dual arms separately. The IP needs to be set differently for each arm and consistent with the burned code. Save the changes after modification.

```
function initializeWebSocket() {
    ws1 = new WebSocket('ws://192.168.1.211:81'); // Create a WebSocket connection to IP address 192.
    ws2 = new WebSocket('ws://192.168.1.212:81'); // Create a WebSocket connection to IP address 192.
```

4. Open the HTML webpage file to enter the following interface:

Device Control

Robot arm 01 (IP: 192.168.1.211)

Enable

Disable

Set zero point

Prepare

Raise arm

Wave arm

Action 4

Action 5

Action 6

Action 7

Action 8

Action 9

ID 1 Position: 0

Speed: 0

ID 2 Position: 0

Speed: 0

ID 3 Position: 0

Speed: 0

ID 4 Position: 0

Speed: 0

ID 5 Position: 0

Speed: 0

ID 6 Position: 0

Speed: 0

ID 7 Position: 0

Speed: 0

Send all commands

Start reading data

Stop reading data

Robot arm 02 (IP: 192.168.1.212)

Enable

Disable

Set zero point

Prepare

Raise arm

Wave arm

Action 4

Action 5

Action 6

Action 7

Action 8

Action 9

ID 1 Position: 0

Speed: 0

ID 2 Position: 0

Speed: 0

ID 3 Position: 0

Speed: 0

ID 4 Position: 0

Speed: 0

ID 5 Position: 0

Speed: 0

ID 6 Position: 0

Speed: 0

ID 7 Position: 0

Speed: 0

Send all commands

Start reading data

Stop reading data

Connect to WebSocket

Simultaneous control (robot arm 01 & robot arm 02)

Prepare

Raise arm

Wave arm

Robot arm 01 data display

Robot arm 02 data display

2.

1.

1. After entering the page, click 'Connect' first, then use other functions.
2. This text box is used to enter the motor ID, which is only used to set the zero point.

It is highly recommended to set the zero point after sending the broadcast command to the desired position before using the robotic arm.