Smart Joystick

status active

---

Smart Joystick

## 📝 Table of Contents

## 🧐 About

This repo contains

- Firmware
- Circuit Diagram
- Detailed instructions

for Smart Joystick.

## Getting Started

These instructions will get you a copy of the project up and running on your system.

### Prerequisites

Things you need to install the FW.

```
- Arduino IDE
```

## Installing

A step by step series that tell you how to get the Firmware and Backend running

#### ESP32 Configuration

You should have Arduino IDE Installed

1. Add ESP32 Board to your Arduino IDE 1. In your Arduino IDE, go to File> Preferences Installing ESP32 Add-on in Arduino IDE Windows, Mac OS X, Linux open preferences 2. Enter `https://dl.espressif.com/dl/package_esp32_index.json` into the "Additional Board Manager URLs" field then, click the "OK" button: Note: if you already have the ESP32 boards URL, you can separate the URLs with a comma(each board will go to neaw line) as follows: `https://dl.espressif.com/dl/package_esp32_index.json,\n http://arduino.esp8266.com/stable/package_esp8266com_index.json`

2. Open the Boards Manager. Go to Tools > Board > Boards Manager…

3. Search for ESP32 and press install button for the ESP32 by Espressif Systems":

4. That's it. It should be installed after a few seconds.

5. In your Arduino sketchbook directory, create tools directory if it doesn't exist yet.

6. Unpack the tool into tools directory(present in libs/ESP32FS-1.0.zip) (the path will look like <home_dir>/Arduino/tools/ESP32FS/tool/esp32fs.jar).

7. Close and re-open the Arduino IDE.

8. Now copy the contents of the libs folder to the libraries directory of your Arduino

    1. If you are using windows, the libraries directory will be Documents/Arduino/libraries

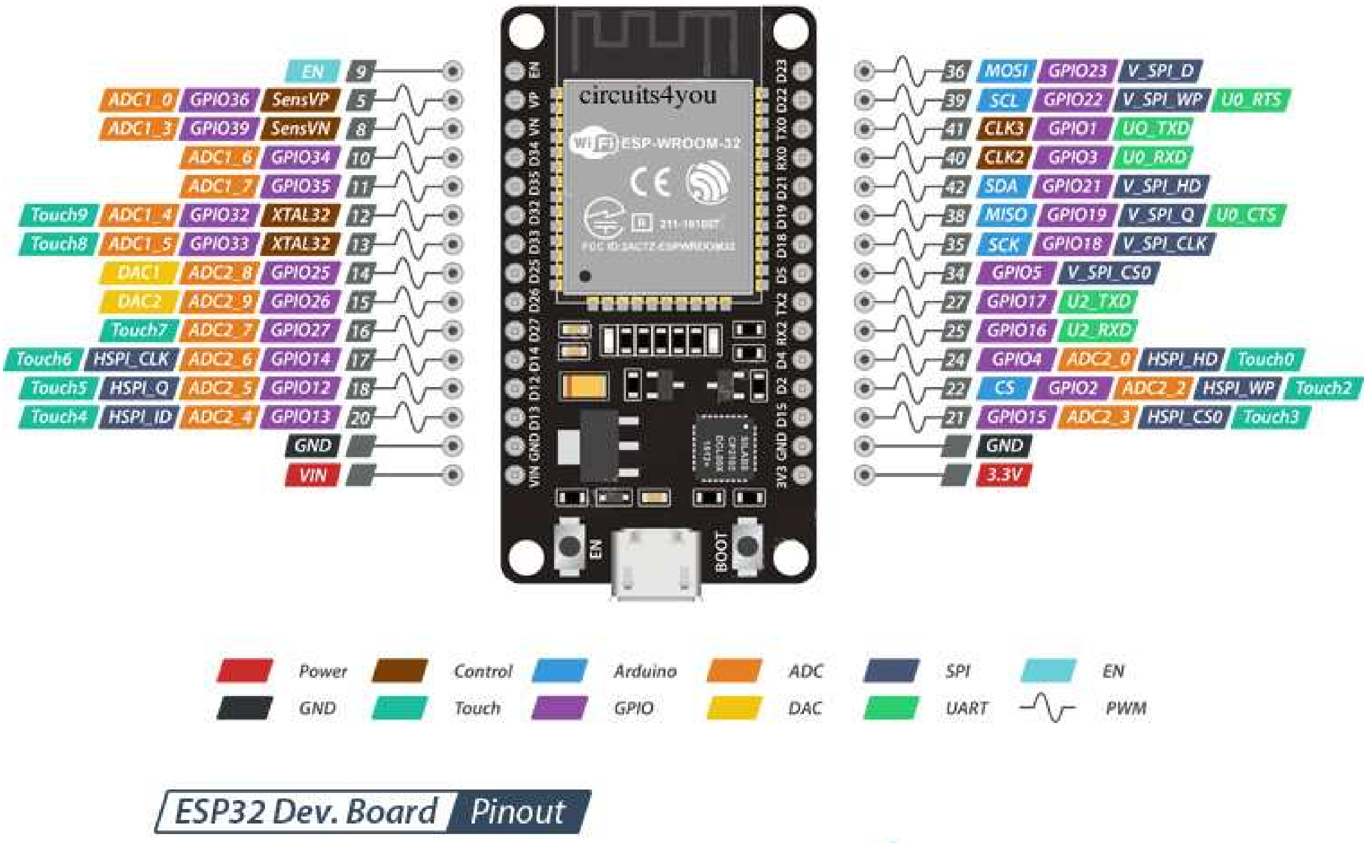
##### ESP32 Node FW Uploading

1. Select ESP32 Dev Module from Tools->Board->ESP32
2. Select the correct port from Tools->Port
3. Then open Firmware.ino file,
4. Select Tools > ESP32 Sketch Data Upload menu item. This should start uploading the files into ESP32 flash file system.
5. Now Upload the Code to your ESP32 Dev Module.
6. Your ESP32 is now ready to be used.

# Circuit

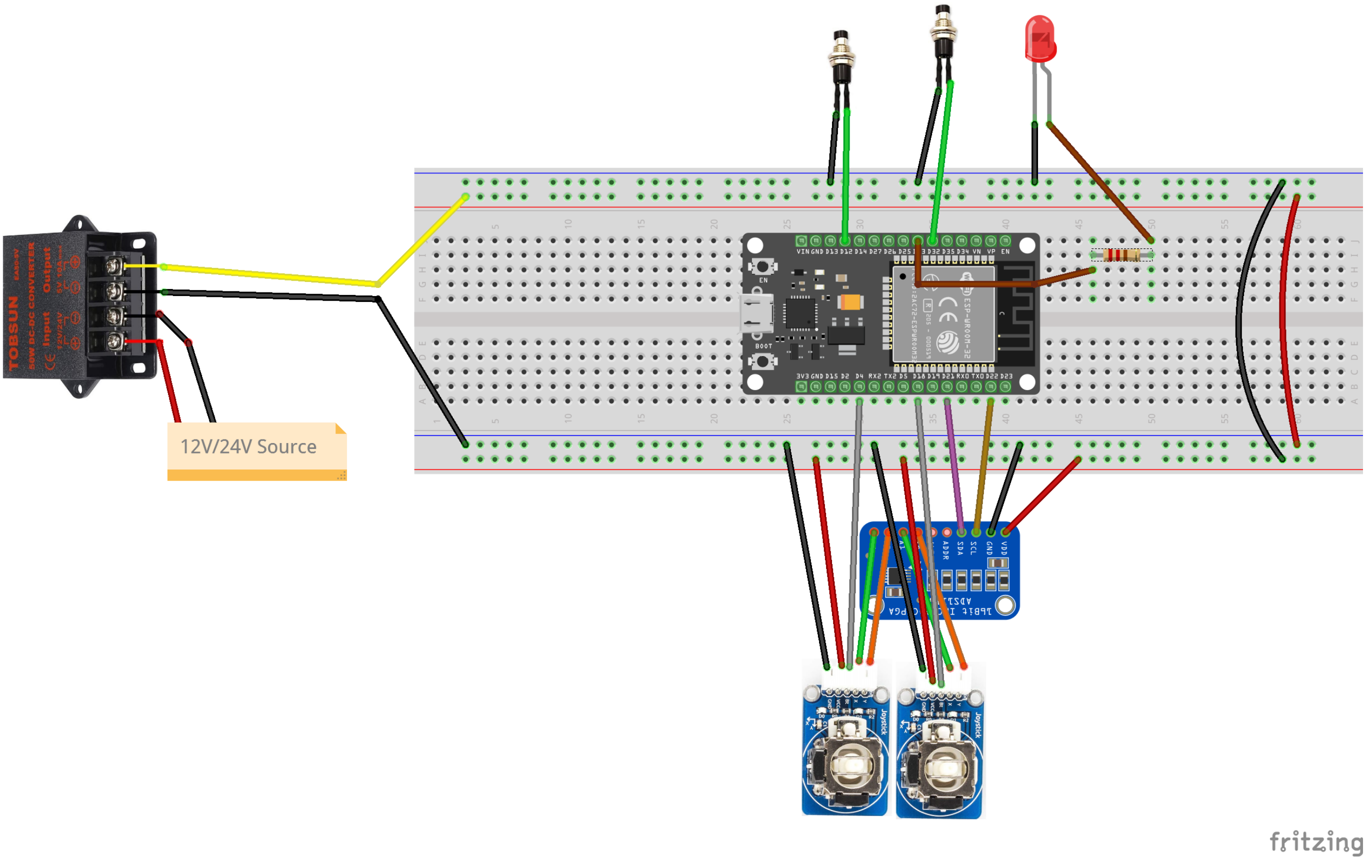
## ESP32 Dev Module Pinout

Follow the pinout diagram given below to connect different components to your TTGO LORA32 board.



## Circuit Diagram for Smart Speaker

Here's the complete circuit diagram of the system.



## Components Connections

```
Other components pin connection details
```

### ADS1115

`ADS1115 Connections with ESP32`

| ADS1115 Pins | ESP32 Dev Module Pins |
|---|---|
| `SCL` | `22` |
| `SDA` | `21` |
| `GND` | `GND` |

### Buttons

`Buttons Connections with ESP32`

| Buttons Pins | ESP32 Dev Module Pins |
|---|---|
| `BTN1_PIN1` | `12` |
| `BTN1_PIN2` | `GND` |
| `BTN2_PIN1` | `33` |
| `BTN2_PIN2` | `GND` |

### Joystick Connections

`Buttons Connections with ADS115 and ESP32`

| Joystick Pins | ADS1115 Pins |
|---|---|
| `Joystick1_Y` | `A0` |
| `Joystick1_X` | `A1` |
| `Joystick2_Y` | `A2` |
| `Joystick2_X` | `A3` |

| Joystick Pins | ESP32 Pins |
|---|---|
| `Joystick1_BT` | `18` |
| `Joystick2_BT` | `4` |

### Status RGB LED

`LED Connections`

| LED Pins | ESP32 Dev Module |
|---|---|

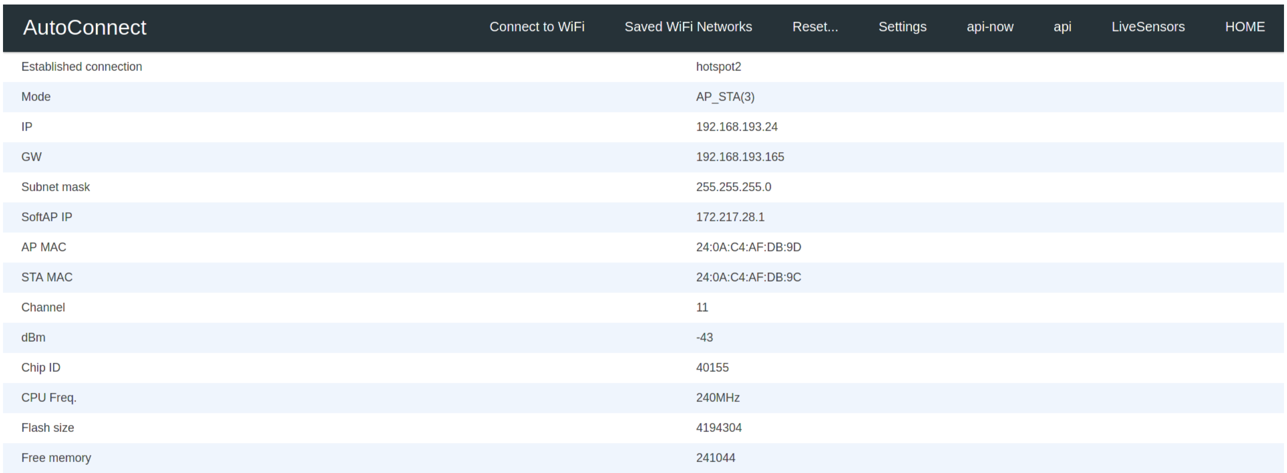
| LED Pins | ESP32 Dev Module |
|---|---|
| `Anode` | `33 via 220Ω resistor` |
| `Cathode` | `GND` |

*33 is also connected to the internal LED of ESP32 Dev Module*

## Usage

```
! Ready for testing
```

1. Power on your ESP32, it will present you with an AP named `SmartJ` (while `SmartJ` can be changed in the portal)
2. Default captive portal password `12345678AP` which can be changed in captive portal.
3. Connect to the ESP32 access point and open the web-browser and navigate to the link `http://smartj.local/_ac`. This link will work on most of the operating systems but if your operating system is not allowing to open it, you may want to check the captive portal IP Address from the serial monitor and can use that IP address inplace of the above mentioned URL.
4. The default access IP Address is `http://192.168.4.1/_ac`
5. You will be presented with a main dashboard as shown below(based on your device)



| | |
|---|---|
| Established connection | hotspot2 |
| Mode | AP_STA(3) |
| IP | 192.168.193.24 |
| GW | 192.168.193.165 |
| Subnet mask | 255.255.255.0 |
| SoftAP IP | 172.217.28.1 |
| AP MAC | 24:0A:C4:AF:DB:9D |
| STA MAC | 24:0A:C4:AF:DB:9C |
| Channel | 11 |
| dBm | -43 |
| Chip ID | 40155 |
| CPU Freq. | 240MHz |
| Flash size | 4194304 |
| Free memory | 241044 |

6. Once connected to a WiFi network, you can again access the captive portal using same URL or the IP Address from the Serial monitor.
7. The data is published to the MQTT Topic `smartj/{hostname}` while the hostname is the one which you can define in Settings page of the captive portal.

### Changing Timezone

```diff
+ Only for developers
```

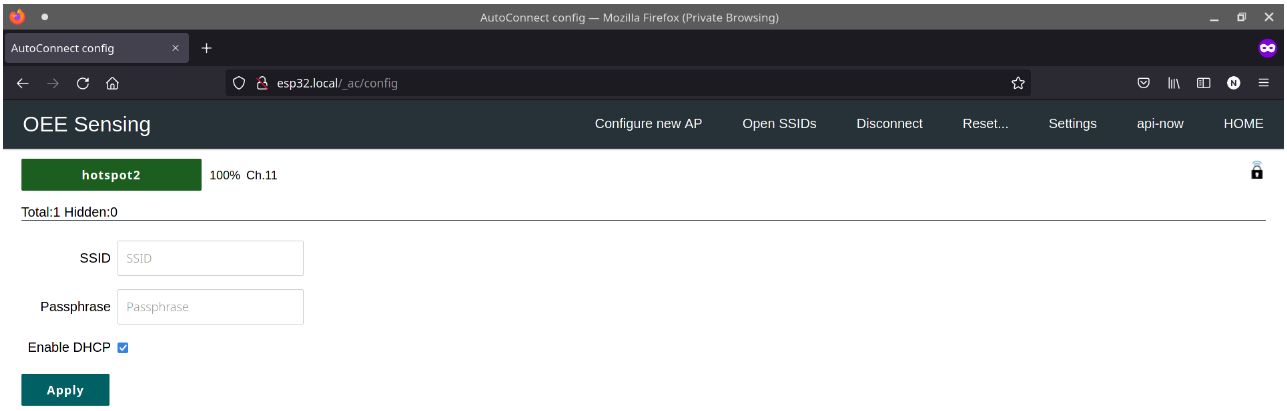
1. Open Settings tab
2. Enter timezone string from https://en.wikipedia.org/wiki/List_of_tz_database_time_zones 'TZ database name' column.
3. Click Save&Start

## API Endpoints and HTML URLS

`API Endpoints`

| Endpoint | Description |
|---|---|
| `/api-now` | `API: live sensor readings in JSON format` |
| `/api` | `API: sensors data in JSON format` |
| `/LiveSensors` | `HTML PAGE: Live Sensor Data` |
| `/data` | `HTML PAGE: Historical Sensor Data` |
| `/mqtt_settings` | `HTML PAGE: Settings. Default username: AP Name, Default Password: admin` |
| `/_ac` | `HTML PAGE: Main Captive portal page` |
| `/` | `HTML PAGE: Historical Sensor Data` |

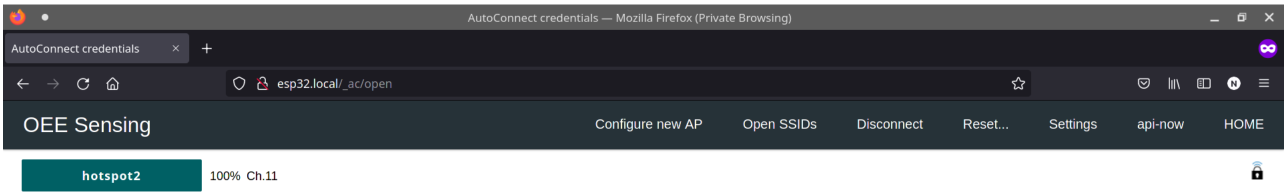
1. **Connect to WiFi** tab allows searching of nearby WiFi APs and adding them to the ESP32.



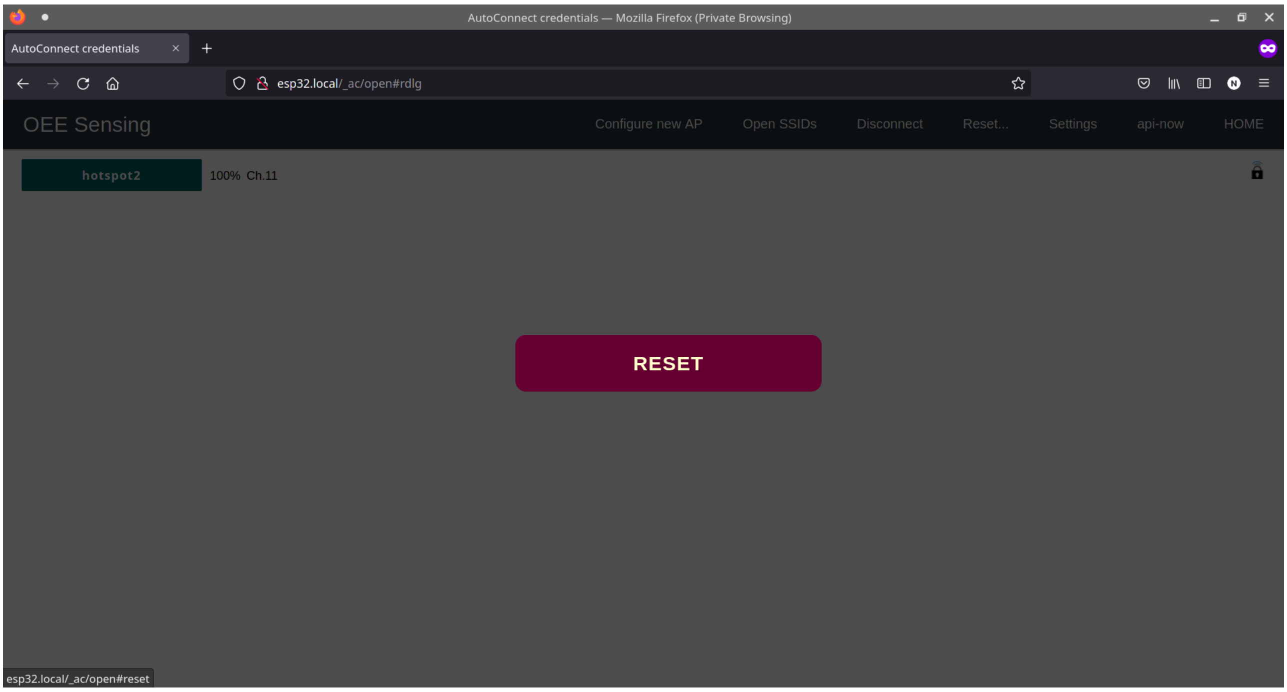
hotspot2 100% Ch.11

Total:1 Hidden:0

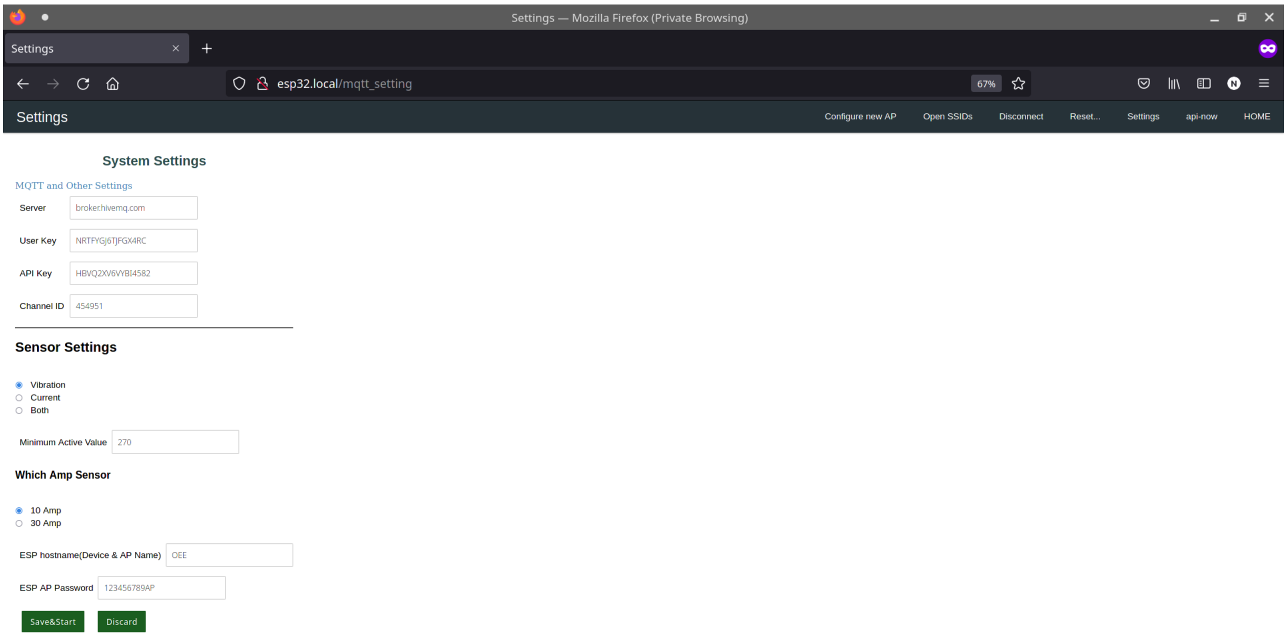
2. **Saved WiFi Networks** tab allows connecting to the saved access points.



3. **Reset...** tab allows reseting of the device to factory settings.



4. **Settings** tab contains settings related to MQTT and sensors.



System Settings

MQTT and Other Settings

Server: broker.hivemq.com

User Key: NRTFYGJ6TJFGX4RC

API Key: HBVQ2XV6VYBI4582

Channel ID: 454951

Sensor Settings

● Vibration
○ Current
○ Both

Minimum Active Value: 270

Which Amp Sensor

● 10 Amp
○ 30 Amp

ESP hostname(Device & AP Name): OEE

ESP AP Password: 123456789AP

Save&Start Discard

5. **HOME** tab shows sensor data acquired live.

## Web App

```
+ For future use
```

### Dashboard Link:

You can access the webapp with following test acccount credentials

- Email Address: `test@test.com`
- Password: `test`

### Dashboard Screenshots

## Smartphone App

```
+ For future use
```

### Smartphone App Link:

You can access the Smartphone app with following test acccount credentials

- Email Address: `test@test.com`
- Password: `test`

### Smartphone App Screenshots

## List of Components

```
+ For future use; not a comprehensive list
```

Following components are used to make this project

1. ESP32 Dev Kit Module
2. Micro USB Cable
3. 2x Joystick Modules
4. Jumper Wires

## ⛏️ Built Using

- Arduino - Embedded Framework and IDE - For Sensor Node Design

## 🎥 Demo Videos

```
+ For future use
```

## ✍️ Authors

- @Nauman3S - Development and Deployment