



Smart Water Meter Monitoring

status active

Smart Water Meter Monitoring

Table of Contents

- [About](#)
- [Getting Started](#)
- [Circuit](#)
- [Server Details](#)
- [MQTT Topic Details](#)
- [Usage](#)
- [List Of Components](#)
- [Built Using](#)
- [Authors](#)

About

This repo contains

- Backend
- Firmware
- Detailed instructions

for Smart Smart Water Meter Monitoring.

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
2. In your Arduino IDE, go to File> Preferences Installing ESP32 Add-on in Arduino IDE Windows, Mac OS X, Linux open preferences
3. 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 new line) as follows:
https://dl.espressif.com/dl/package_esp32_index.json,
http://arduino.esp8266.com/stable/package_esp8266com_index.json
4. Open the Boards Manager. Go to Tools > Board > Boards Manager...
5. Search for ESP32 and press install button for the ESP32 by Espressif Systems":
6. That's it. It should be installed after a few seconds.
7. In your Arduino sketchbook directory, create tools directory if it doesn't exist yet.
8. 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).
9. Close and re-open the Arduino IDE.
10. 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

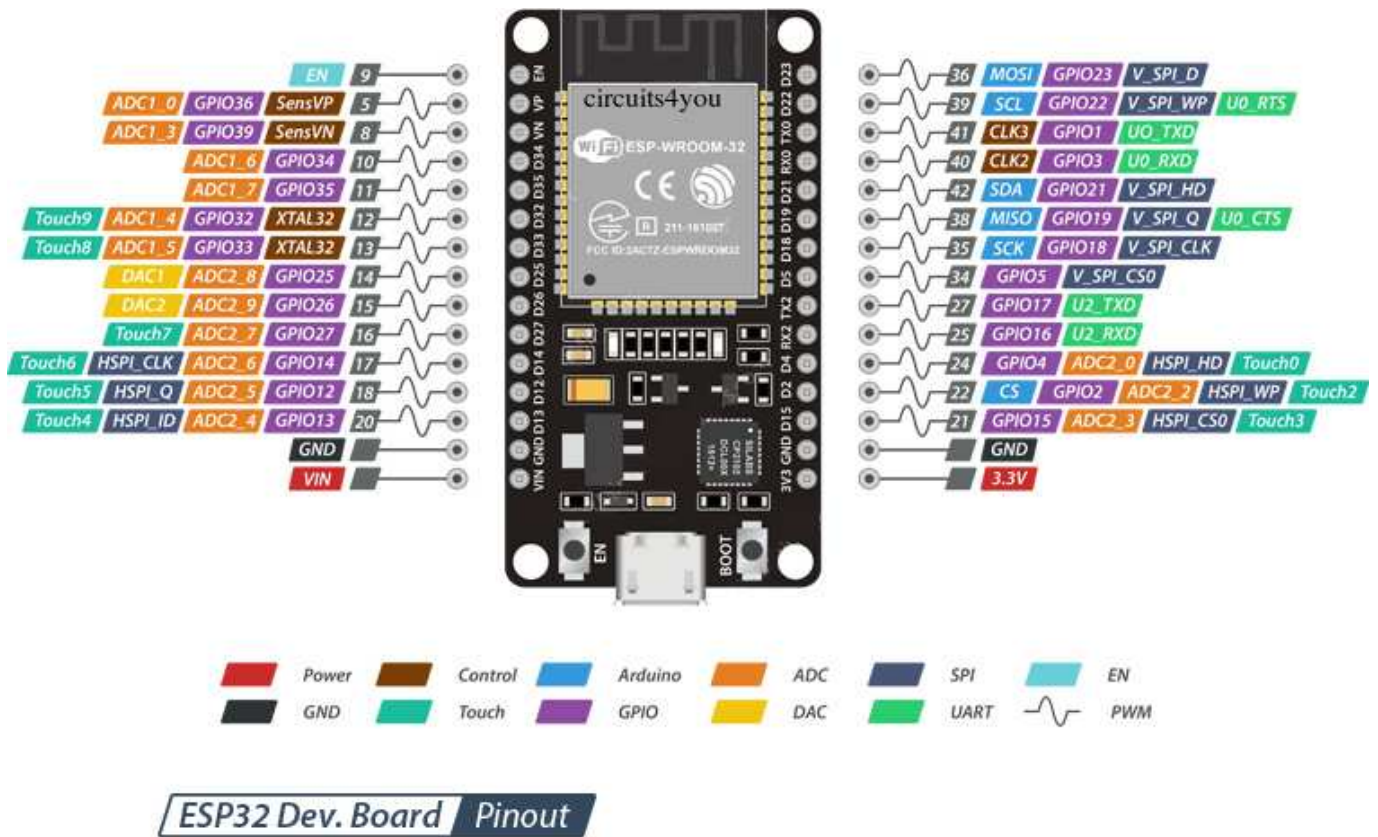
ESP32 Node FW Uploading

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 ESP32 Dev Module board.



GPIOs for connection

The following 10 GPIOs are available to be used

- 12, 19, 18, 5, 25, 26, 4, 15, 27, 14

Server Details

Specifications

- Recommended Specs are
 - 2 GB RAM
 - 2 Core CPU or vCPU
 - 50 GB Disk Space
 - Ubuntu Server 20.04

List of Packages installed on server

- Mosquitto Broker
- NodeRED
- ufw
- Caprover

Server Links

- MQTT Broker Link: hivemq.com
- Frontend Link: <https://nodered-proxy.production.wrapdrive.tech/ui/#/1/1>

Backend

- Backend is based on NodeRED and it is being managed by Caprover. It starts automatically on server start.

MQTT Topic Details

Topics List

Logs

- `smart-wm/macAddress/gallons` (all connected water meter gallons information is sent to this topic) READ-ONLY

Usage

- Upload the code to your ESP32.
- Open the dashboard to monitor the parameters.
- Power on your ESP32, it will present you with an AP named `SmartWM-abc` (while `SmartWM` can be changed in the portal and `abc` is a unique id for each esp32)
- Default captive portal password `12345678AP` which can be changed in captive portal.
- Connect to the ESP32 access point and open the web-browser and navigate to the link `http://esp32.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.
- The default access IP Address is `http://192.168.4.1/_ac` 8. You will be presented with a main dashboard as shown below(based on your device)

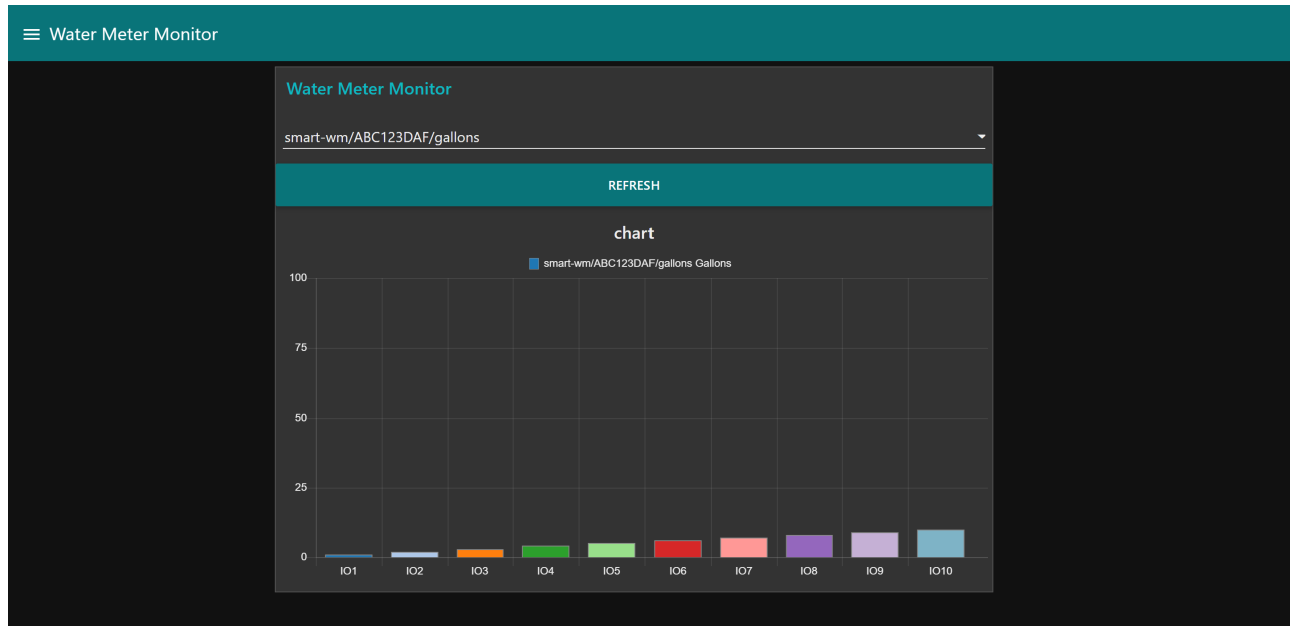
Smart Tanning Device		Connect to WiFi	Saved WiFi Networks	Disconnect	Reset...	Settings	HOME
Established connection	N/A						
Mode	AP_STA(6)						
IP	0.0.0.0						
GW	0.0.0.0						
Subnet mask	0.0.0.0						
SoftAP IP	192.168.4.1						
AP MAC	A4:CF:12:44:30:01						
STA MAC	A4:CF:12:44:30:00						
Channel	1						
dBm	0						
Chip ID	48						
CPU Freq	240MHz						
Flash size	16777216						
Free memory	246008						

- Once connected to a WiFi network, you can again access the captive portal using same URL or the IP Address from the Serial monitor.
- The data is published to the MQTT Topic `SmartWM/{hostname}` while the hostname is the one which you can define in Settings page of the captive portal.
- You can open settings page with following default credentials

1. User: **AP Name (SmartWM)**

2. Password: **admin**

10. Now you can open the dashboard and can select the active device from dropdown to see its total gallons on all 10 GPIO Pins



List of Components

Following components are used to make this project

1. ESP32 Dev Module ○ https://www.amazon.com/ESP32-WROOM-32-Development-ESP-32S-Bluetooth-Arduino/dp/B084KWNMM4/ref=sr_1_10?dchild=1&keywords=esp32+dev+module&qid=1631286335&sr=8-10

Built Using

- [NodeRED](#) - Programming Tool
- [Arduino](#) - Embedded Framework and IDE - For Sensor Node Design

Authors

- [@Nauman3S](#) - Development and Deployment