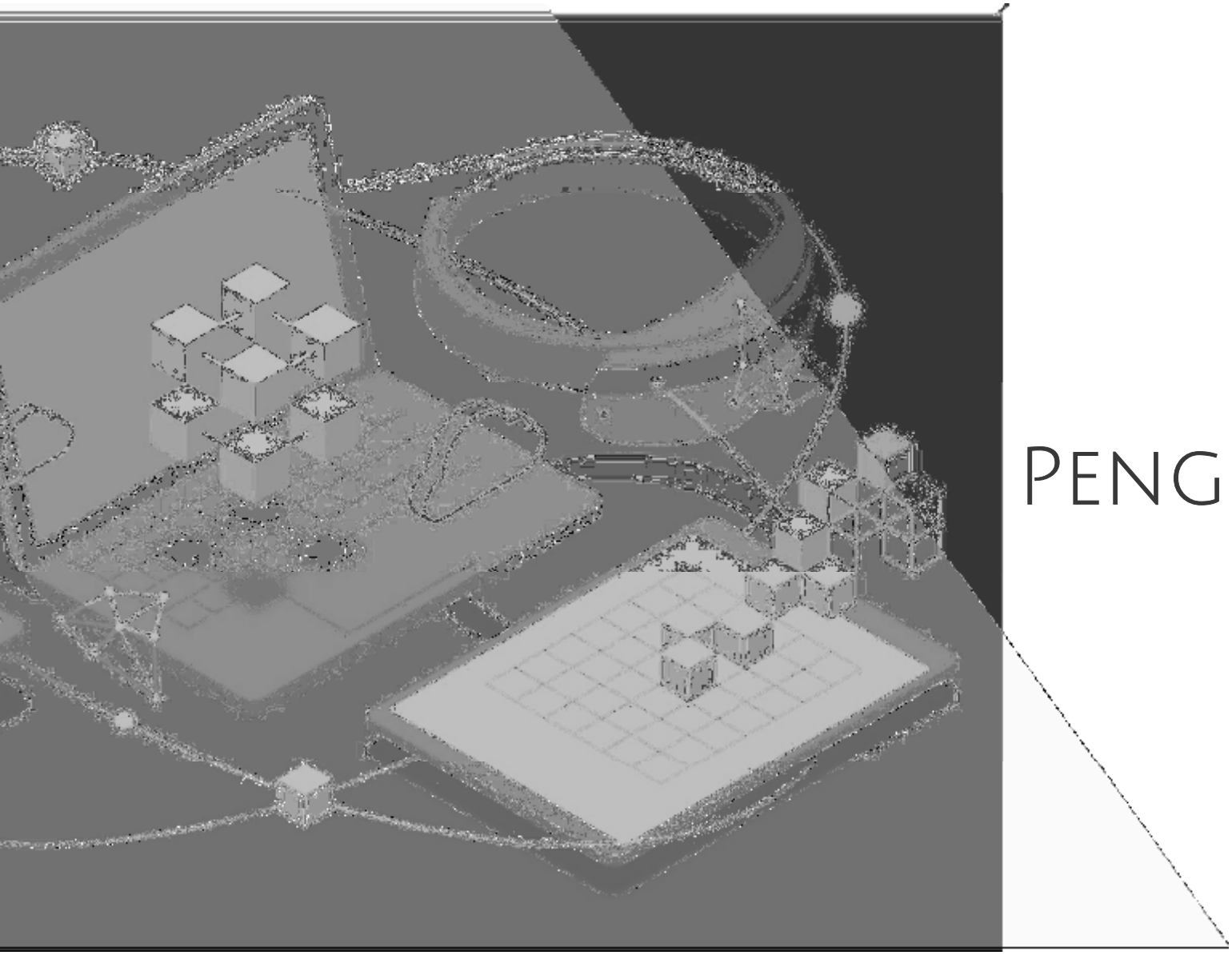
IOT ESP32

Table of Contents



01

PENGENALAN

GOAL

- Memahami sistem kerja ESP32
- Membuat program dengan Platform IO (arduino framework)
- Membuat firmware untuk membaca sensor
- Mengirimkan data ke cloud melalui MQTT

ESP32

Tentang ESP32

ESP32 Features and Specifications

- Wireless connectivity WiFi: 150.0 Mbps data rate with HT40
- Bluetooth: BLE (Bluetooth Low Energy) and Bluetooth Classic
- Processor: Tensilica Xtensa Dual-Core 32-bit LX6 microprocessor, running at 160 or 240 MHz
- ROM: 448 KB
- SRAM: 520 KB
- Low Power: ensures that you can still use ADC conversions, for example, during deep sleep.

Peripheral Input/Output:

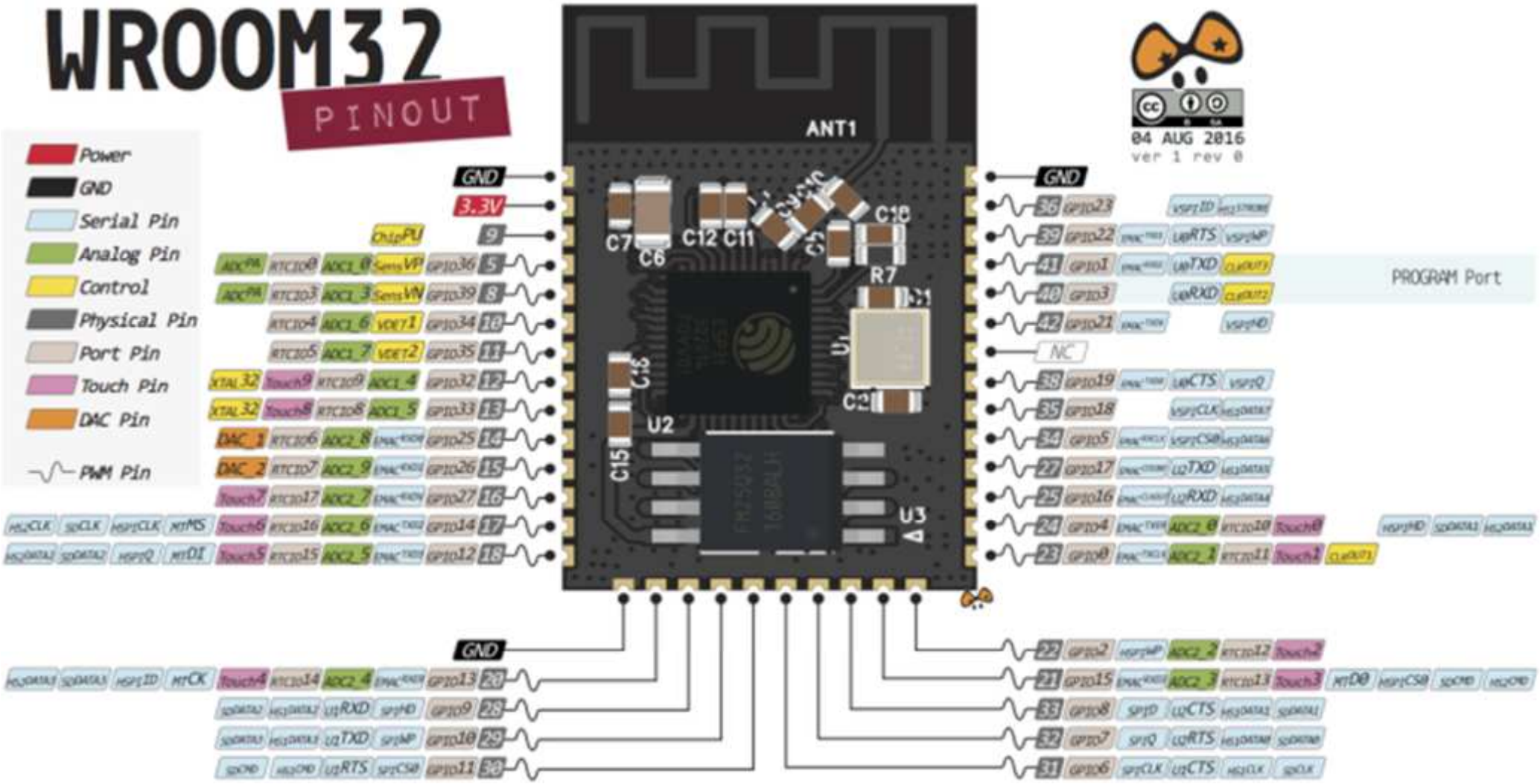
- Peripheral interface with DMA that includes capacitive touch
- ADCs (Analog-to-Digital Converter)
- DACs (Digital-to-Analog Converter)
- I²C (Inter-Integrated Circuit)
- UART (Universal Asynchronous Receiver/Transmitter)
- SPI (Serial Peripheral Interface)
- I²S (Integrated Interchip Sound)
- RMII (Reduced Media-Independent Interface)
- PWM (Pulse-Width Modulation).

Program Env

Arduino IDE
Espressif IDF
Micropython
JavaScript
LUA

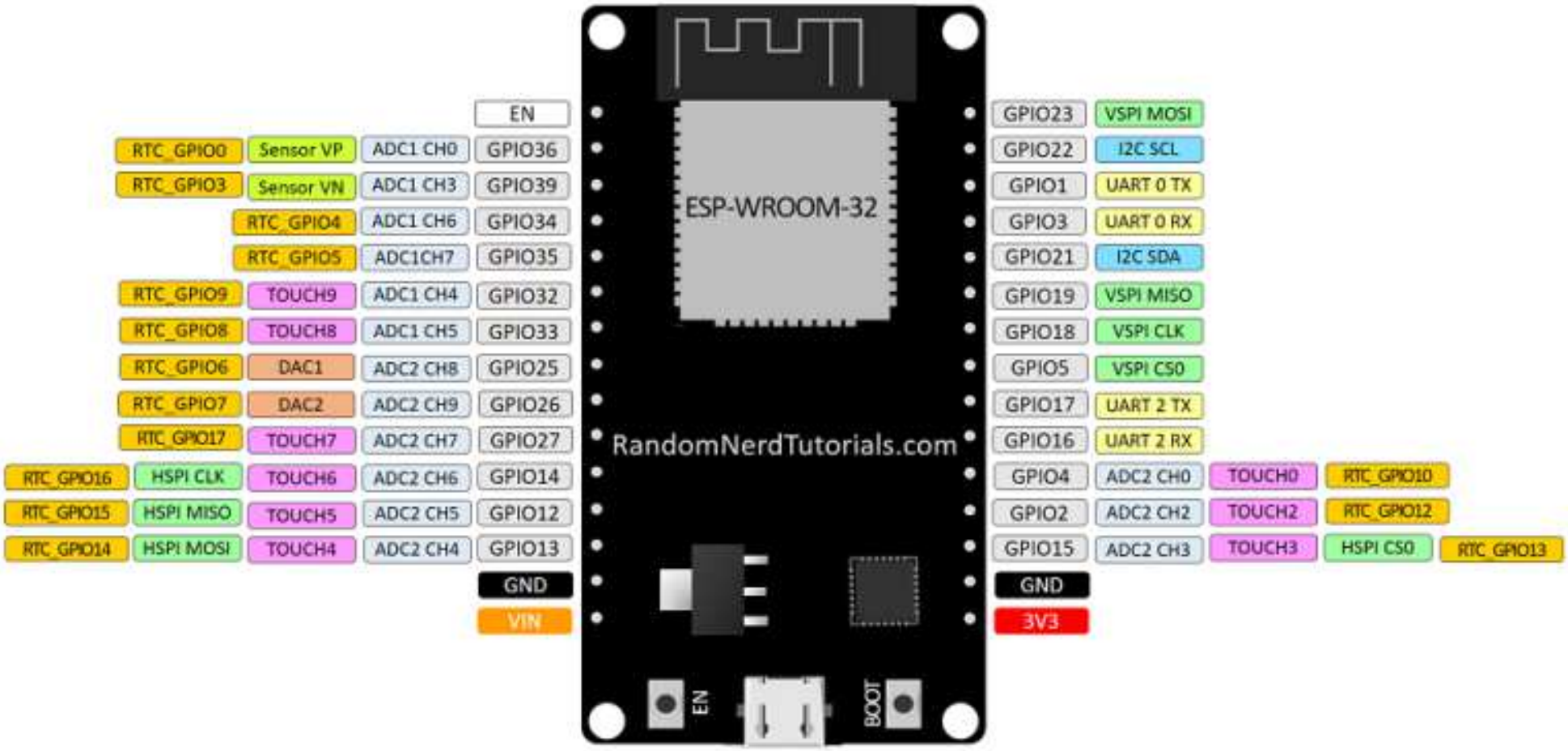
(Windows, Mac OS X and Linux)

DEVELOPMENT BOARD


WROOM32
PINOUT
Power
GND
Serial Pin
Analog Pin
Control
Physical Pin
Port Pin
Touch Pin
DAC Pin
PWM Pin
ANT1
04 AUG 2016
ver 1 rev 0
PROGRAM Port
GND
3.3V
NC

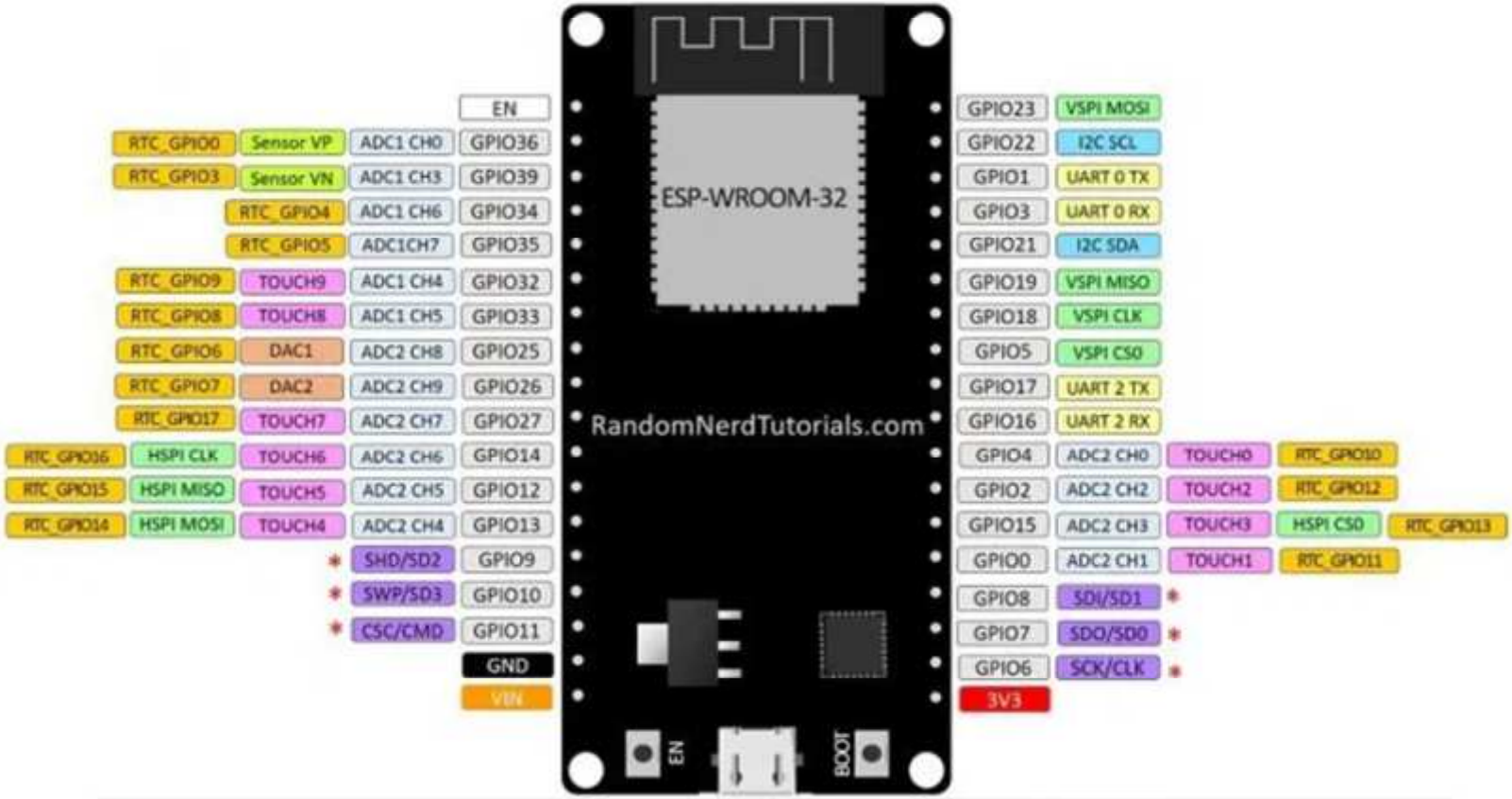
ESP32 DEVKIT V1 – DOIT

version with 30 GPIOs



ESP32 DEVKIT V1 – DOIT

version with 36 GPIOs



* Pins SCK/CLK, SDO/SD0, SDI/SD1, SHD/SD2, SWP/SD3 and SCS/CMD, namely, GPIO6 to GPIO11 are connected to the integrated SPI flash integrated on ESP-WROOM-32 and are not recommended for other uses.

ESP32 Pinout Reference

Input only pins

- GPIO 34
- GPIO 35
- GPIO 36
- GPIO 39

ESP32 has 18 x 12 bits ADC input channels (while the ESP8266 only has 1x 10 bits ADC).

ADC1_CH0 (GPIO 36) ADC1_CH1 (GPIO 37)
ADC1_CH2 (GPIO 38) ADC1_CH3 (GPIO 39)
ADC1_CH4 (GPIO 32) ADC1_CH5 (GPIO 33)
ADC1_CH6 (GPIO 34) ADC1_CH7 (GPIO 35)
ADC2_CH0 (GPIO 4) ADC2_CH1 (GPIO 0)
ADC2_CH2 (GPIO 2) ADC2_CH3 (GPIO 15)
ADC2_CH4 (GPIO 13) ADC2_CH5 (GPIO 12)
ADC2_CH6 (GPIO 14) ADC2_CH7 (GPIO 27)
ADC2_CH8 (GPIO 25) ADC2_CH9 (GPIO 26)

There are 2 x 8 bits DAC channels on the ESP32 to convert digital signals into analog voltage signal outputs. These are the DAC channels:

DAC1 (GPIO25)
DAC2 (GPIO26)

ESP32 PINOUT REFERENCE

Strapping Pins

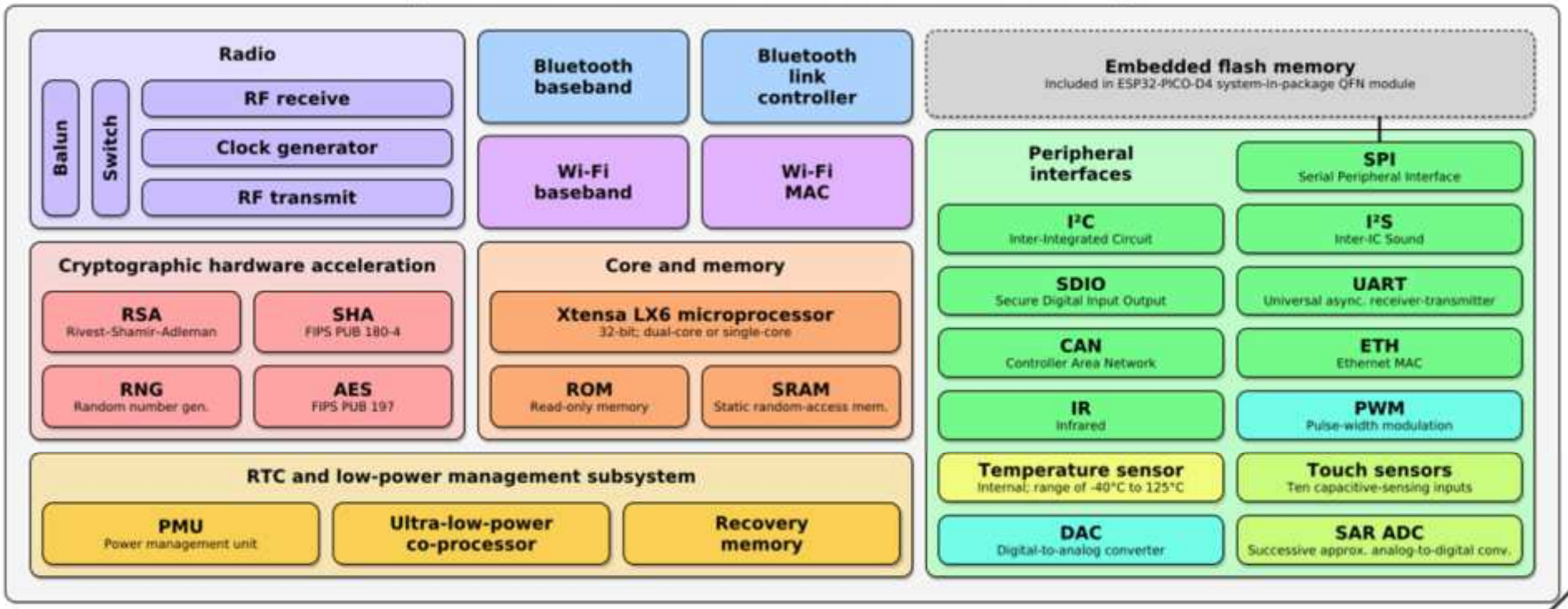
The ESP32 chip has the following strapping pins:

GPIO 0
GPIO 2
GPIO 4
GPIO 5 (must be HIGH during boot)
GPIO 12 (must be LOW during boot)
GPIO 15 (must be HIGH during boot)

Pins HIGH at Boot

GPIO 1
GPIO 3
GPIO 5
GPIO 6 to GPIO 11 (connected to the ESP32 integrated SPI flash memory – not recommended to use).
GPIO 14
GPIO 15

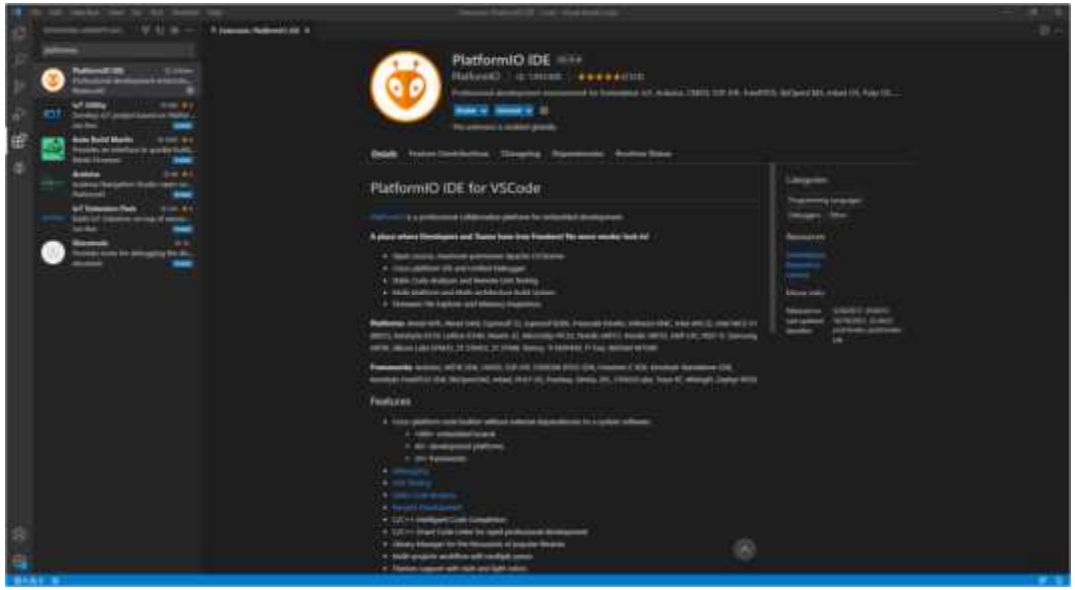
Espressif ESP32 Wi-Fi & Bluetooth Microcontroller — Function Block Diagram



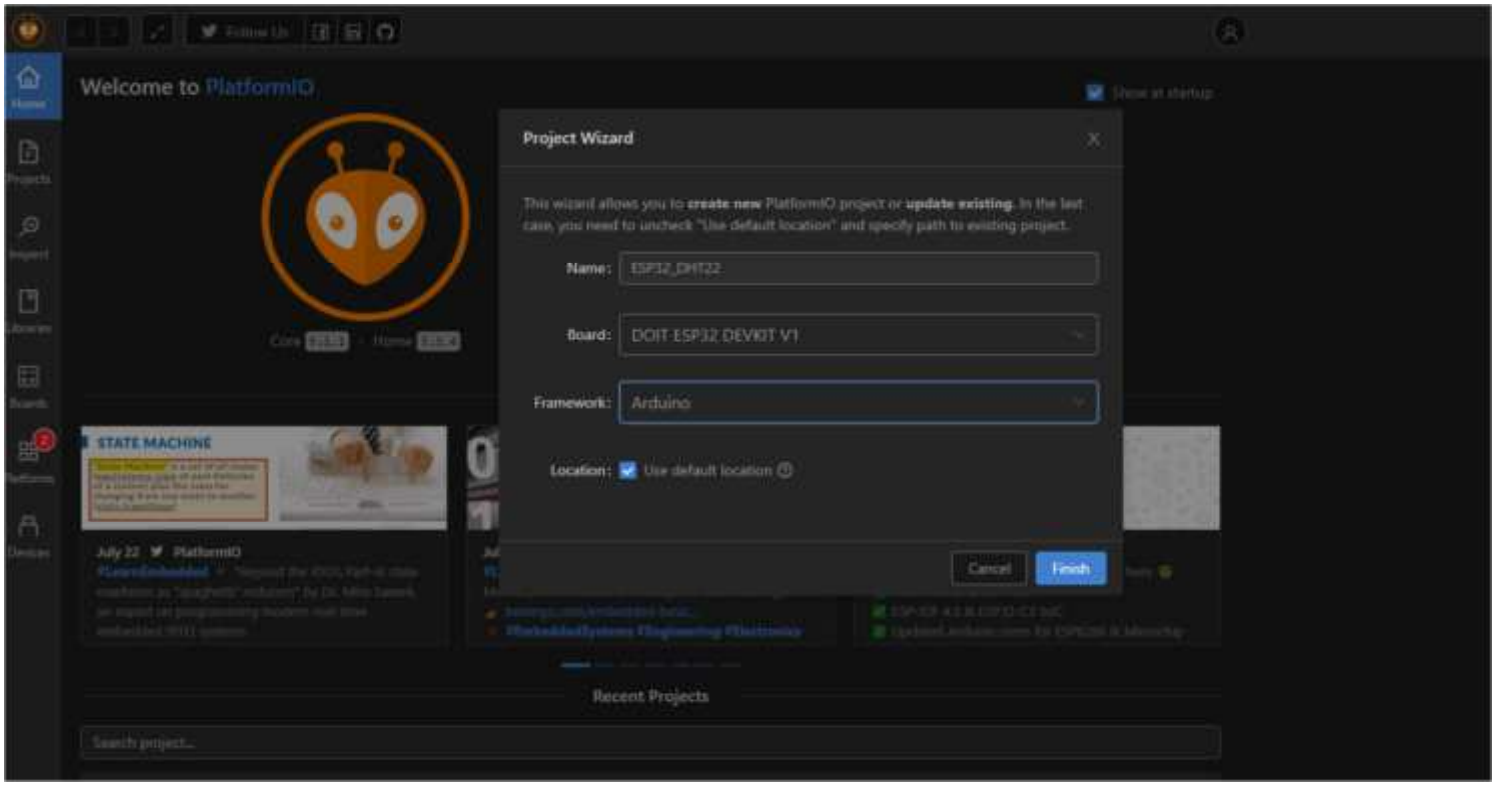
Platform IO

PLATFORM IO IDE

PlatformIO IDE berjalan diatas VSCode sebagai official extentions
Pada menu Extention Manager pada sidebar IDE VScode– search platformIO – pilih install



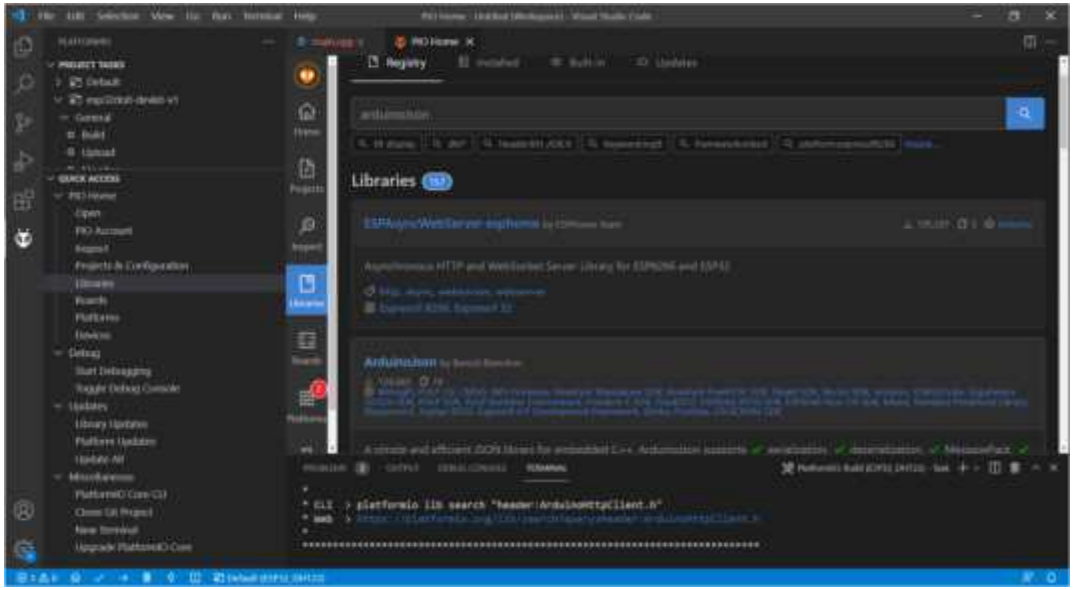
MEMBUAT PROJECT BARU



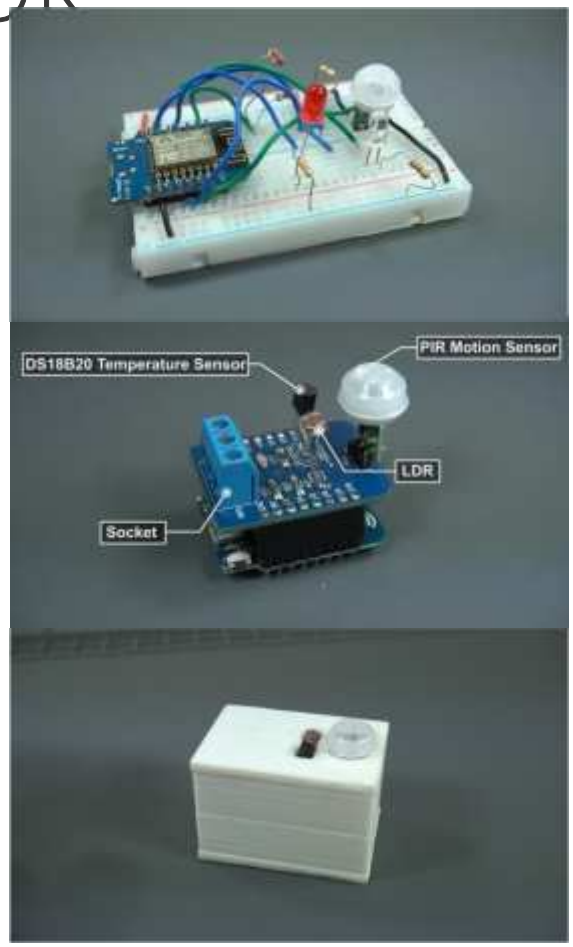
Membaca sensor

Tentang sensor

INSTALL LIBRARY UNTUK SENSOR



Sensor



SHIELD OR BOARD CUSTOM

SENSOR

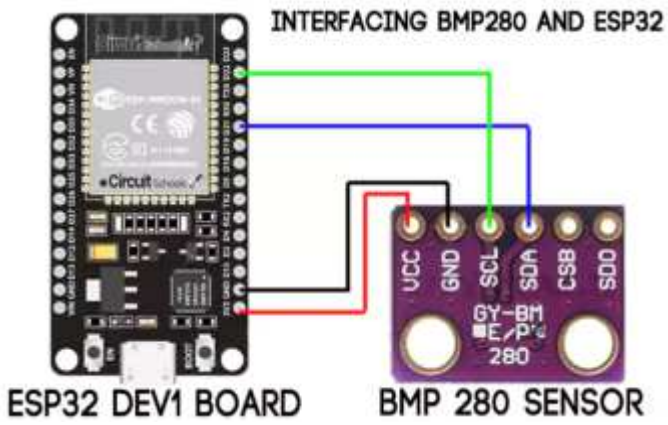
```
typedef struct
{
bool (Init)(const SensorConfig_t * const Config);
bool (Read)(const SensorObj_t * const, SensorData_t * const SensorData);
bool (*Write)(const SensorObj_t * const, SensorData_t * const
SensorData);
} Sensor_t;
```

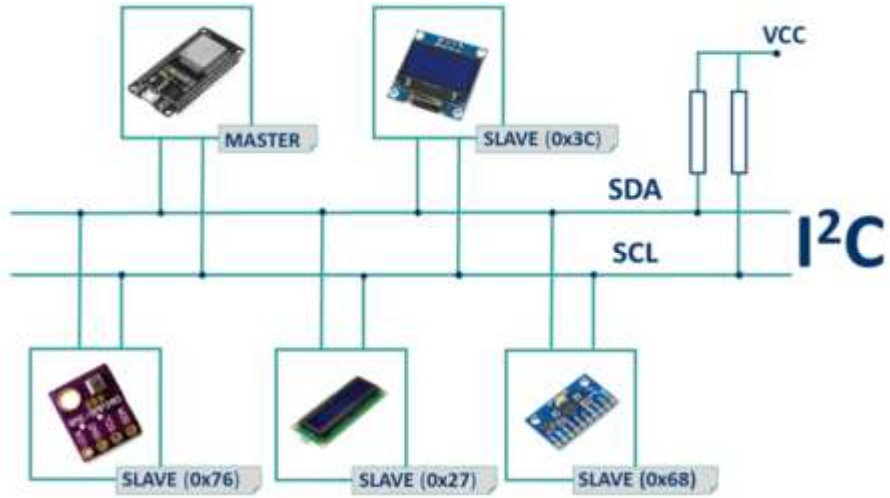
```
const Sensor_t Analog =
{
Adc_Init,
Adc_Read,
Adc_Write
};
```

```
const Sensor_t Gyro =
{
Gyro_Init,
Gyro_Read,
Gyro_Write
};
```

```
Analog.Init(AdcConfig);
Gryo.Init(GyroConfig);
```

SENSOR TEMPERATURE, HUMIDITY AND PRESSURE.





SDA	SDA (default is GPIO 21)
SCL	SCL (default is GPIO 22)

```
#for using different wire
Wire.begin(I2C_SDA, I2C_SCL);
```

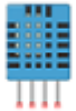
BMP280 Test

PROBLEMS 181 OUTPUT DEBUG CONSOLE TERMINAL

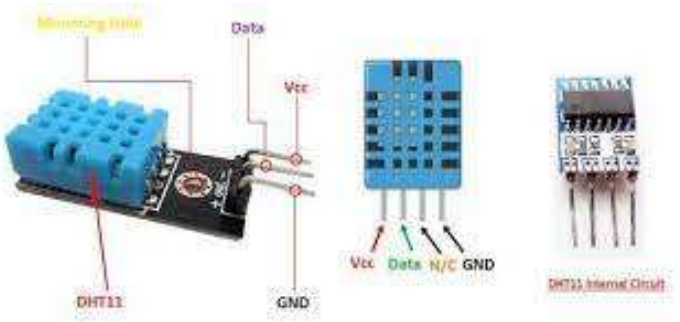
```
Temperature = 27.98 *C
Pressure = 92511.44 Pa
Approx altitude = 761.07 m

Temperature = 27.97 *C
Pressure = 92511.39 Pa
Approx altitude = 761.08 m
```

DHT11/DHT22 Temperature and Humidity

	DHT11 	DHT22 
Temperature range	0 to 50 ºC +/-2 ºC	-40 to 80 ºC +/-0.5ºC
Humidity range	20 to 90% +/-5%	0 to 100% +/-2%
Resolution	Humidity: 1% Temperature: 1ºC	Humidity: 0.1% Temperature: 0.1ºC
Operating voltage	3 – 5.5 V DC	3 – 6 V DC
Current supply	0.5 – 2.5 mA	1 – 1.5 mA
Sampling period	1 second	2 seconds
Price	$1 to $5	$4 to $10

DHT 11


Data
Vcc
DHT11
GND
Vcc Data N/C GND
DHT11 Internal Circuit

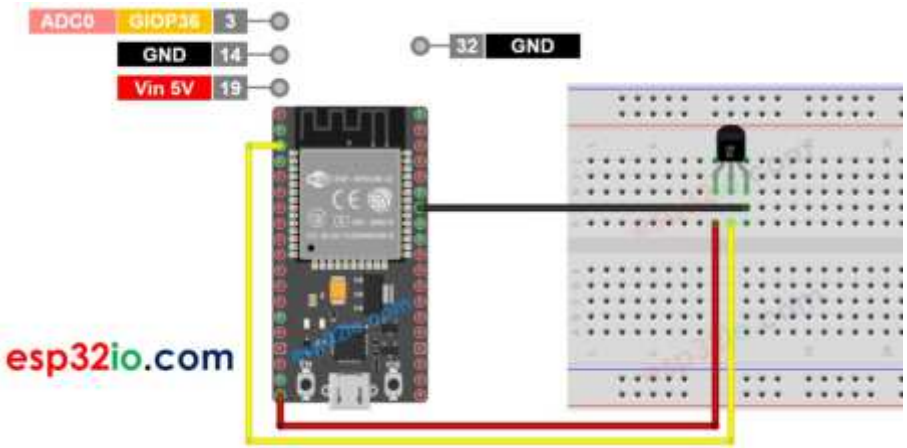

1 3.3V
33 GIOP21
32 GND
esp32io.com


DHT11
VCC
DATA
NC
GND
esp32io.com
DHT22
VCC
DATA
NC
GND
DHT11
VCC
DATA
GND
esp32io.com
DHT22
VCC
DATA
GND

DHT 11 Test

```
Failed to read from DHT sensor!
Failed to read from DHT sensor!
Failed to read from DHT sensor!
Failed to read from DHT sensor!
Humidity: 69.00%  Temperature: 27.50°C 81.50°F  Heat index: 29.62°C 85.32°F
Humidity: 66.00%  Temperature: 27.40°C 81.32°F  Heat index: 29.15°C 84.47°F
Failed to read from DHT sensor!
Humidity: 69.00%  Temperature: 27.40°C 81.32°F  Heat index: 29.45°C 85.01°F
Humidity: 65.00%  Temperature: 27.30°C 81.14°F  Heat index: 28.90°C 84.03°F
Humidity: 65.00%  Temperature: 27.30°C 81.14°F  Heat index: 28.90°C 84.03°F
Humidity: 65.00%  Temperature: 27.40°C 81.32°F  Heat index: 29.06°C 84.30°F
```

LM35 Sensor Temperature



TEST LM35

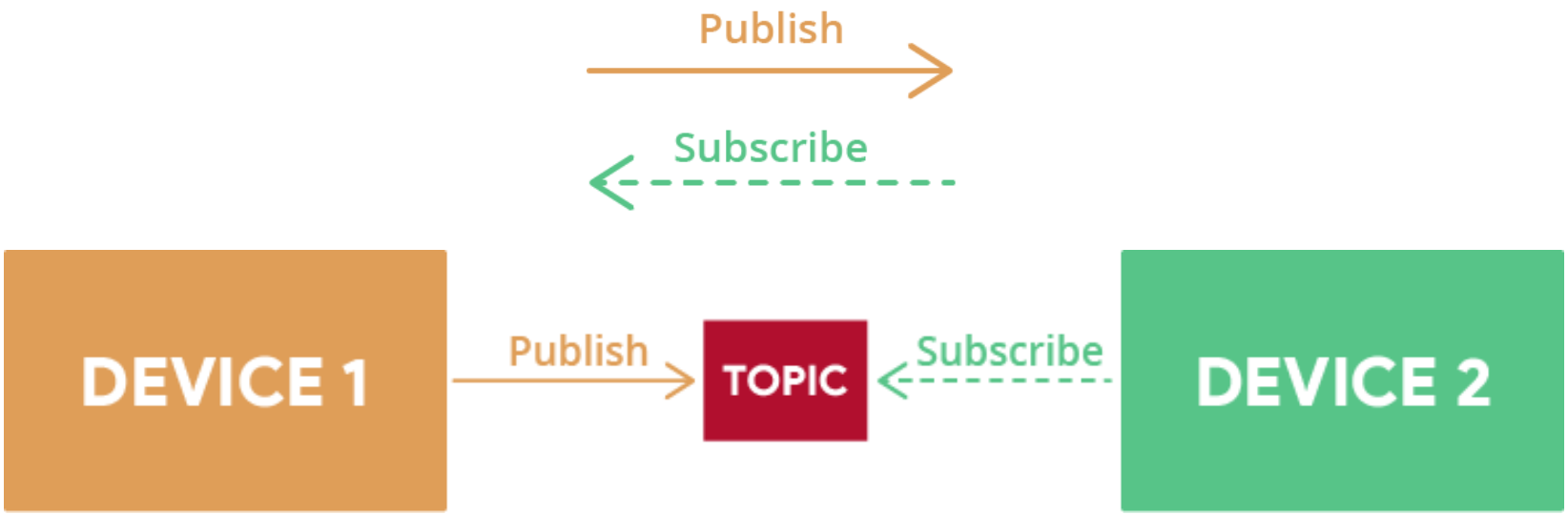
```
void loop() {
  // read the ADC value from the temperature sensor
  int adcVal = analogRead(PIN_LM35);
  // convert the ADC value to voltage in millivolt
  float milliVolt = adcVal * (ADC_VREF_mV / ADC_RESOLUTION);
  // convert the voltage to the temperature in °C
  float tempC = milliVolt / 10;
  // convert the °C to °F
  float tempF = tempC * 9 / 5 + 32;
```

```
Temperature: 32.23°C  ~  90.01°F
Temperature: 31.74°C  ~  89.14°F
Temperature: 30.94°C  ~  87.69°F
Temperature: 30.21°C  ~  86.38°F
Temperature: 29.65°C  ~  85.37°F
Temperature: 29.57°C  ~  85.22°F
Temperature: 29.65°C  ~  85.37°F
Temperature: 31.34°C  ~  88.41°F
Temperature: 33.52°C  ~  92.33°F
Temperature: 35.77°C  ~  96.39°F
Temperature: 40.36°C  ~  104.65°F
Temperature: 43.59°C  ~  110.46°F
```

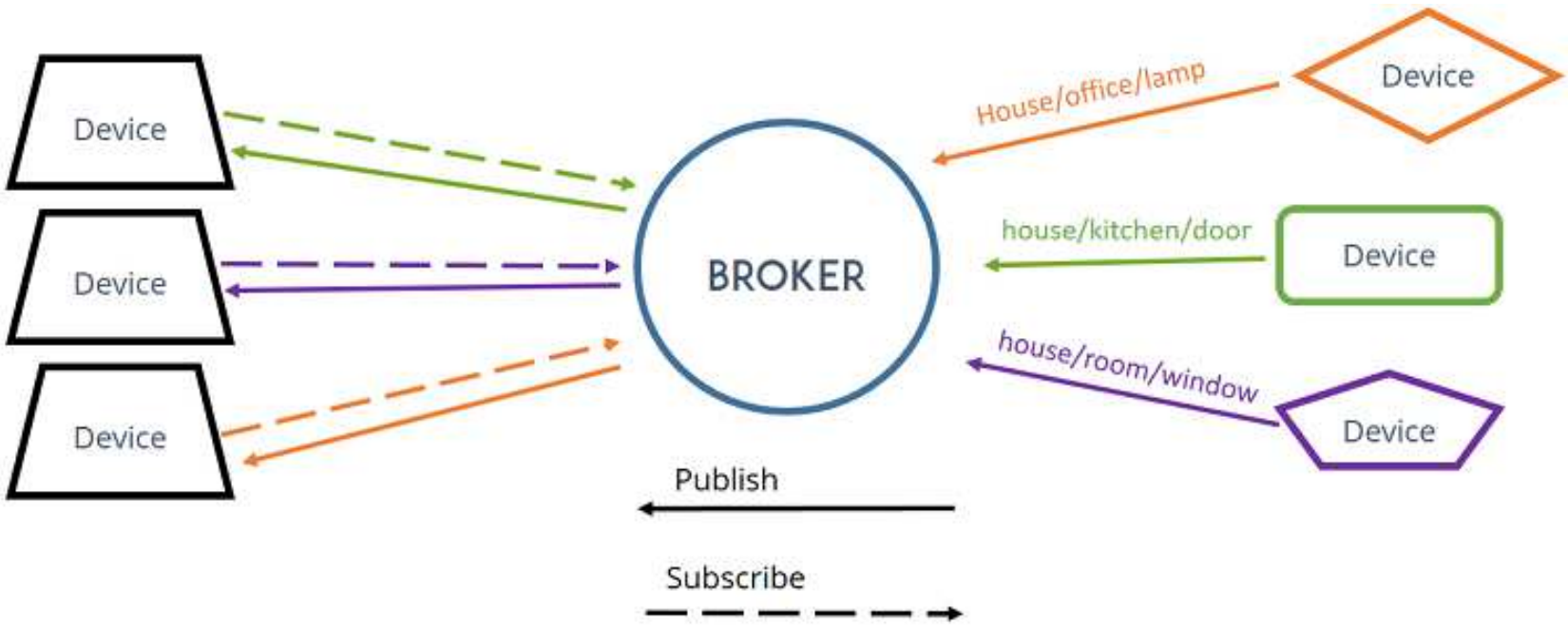
MQTT

MQTT

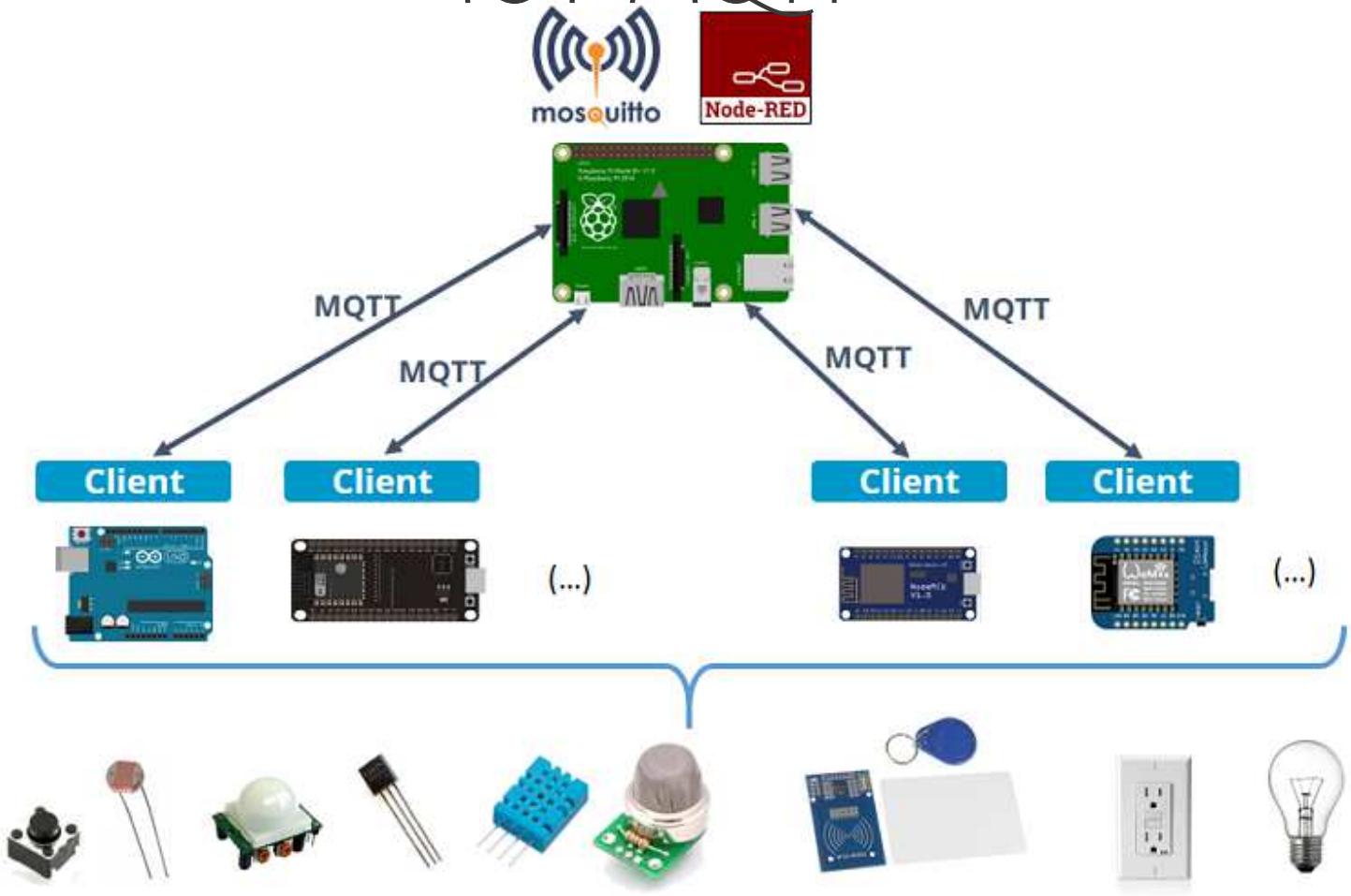
MQTT stands for Message Queuing Telemetry Transport.



MQTT Broker


Device
Device
Device
BROKER
House/office/lamp
Device
house/kitchen/door
Device
house/room/window
Device
Publish
Subscribe

IOT MQTT


mosquitto
Node-RED
MQTT
MQTT
MQTT
MQTT
Client
Client
Client
Client
(...)
(...)

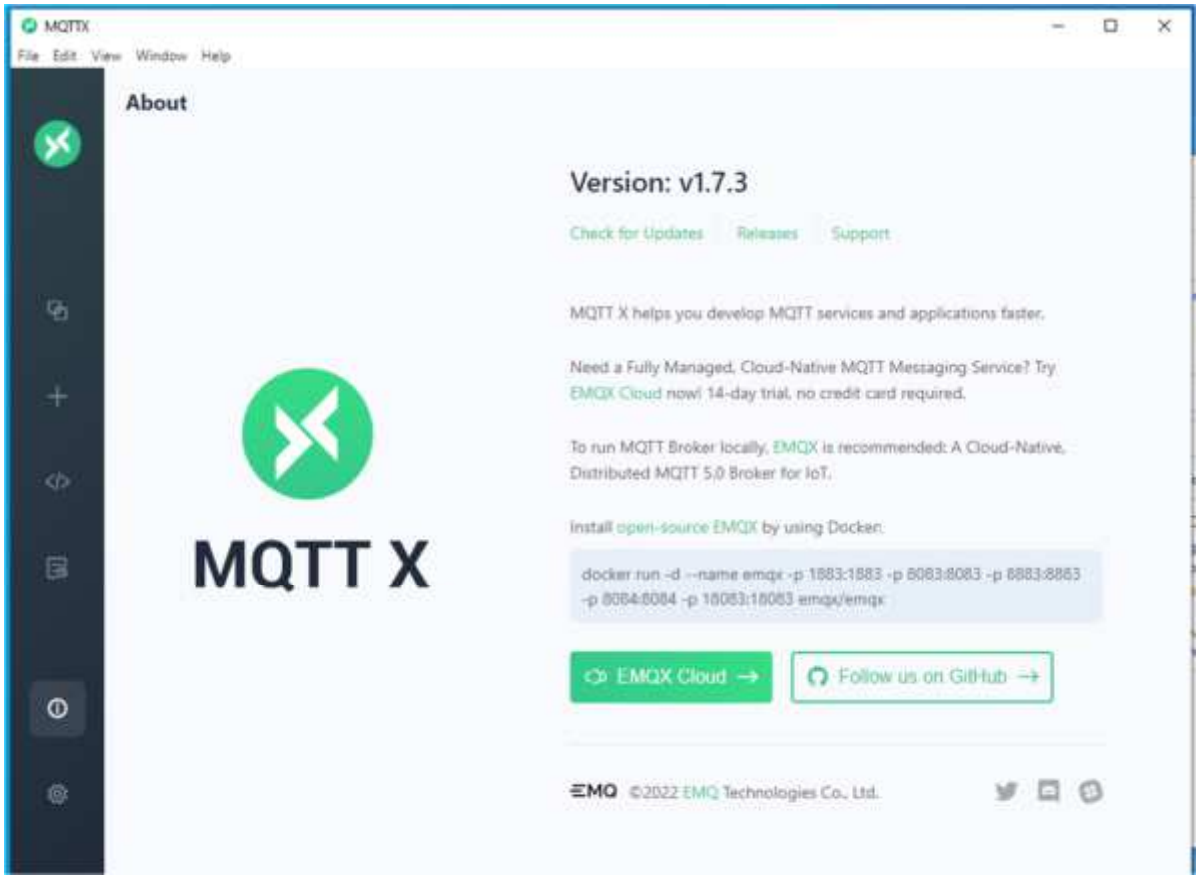
ESP MQTT

```
// WiFi
const char *ssid = "TPLINK"; // Enter your WiFi name
const char *password = "TPLINK32";  // Enter WiFi password

// MQTT Broker
const char *mqtt_broker = "broker.emqx.io";
const char *topic = "esp32/sensor/";
const char *mqtt_username = "emqx";
const char *mqtt_password = "public";
const int mqtt_port = 1883;
```

```
lib_deps =
    ;adafruit/DHT sensor library@^1.4.3
    ;adafruit/Adafruit Unified Sensor@^1.1.5
    knolleary/PubSubClient@^2.8
```

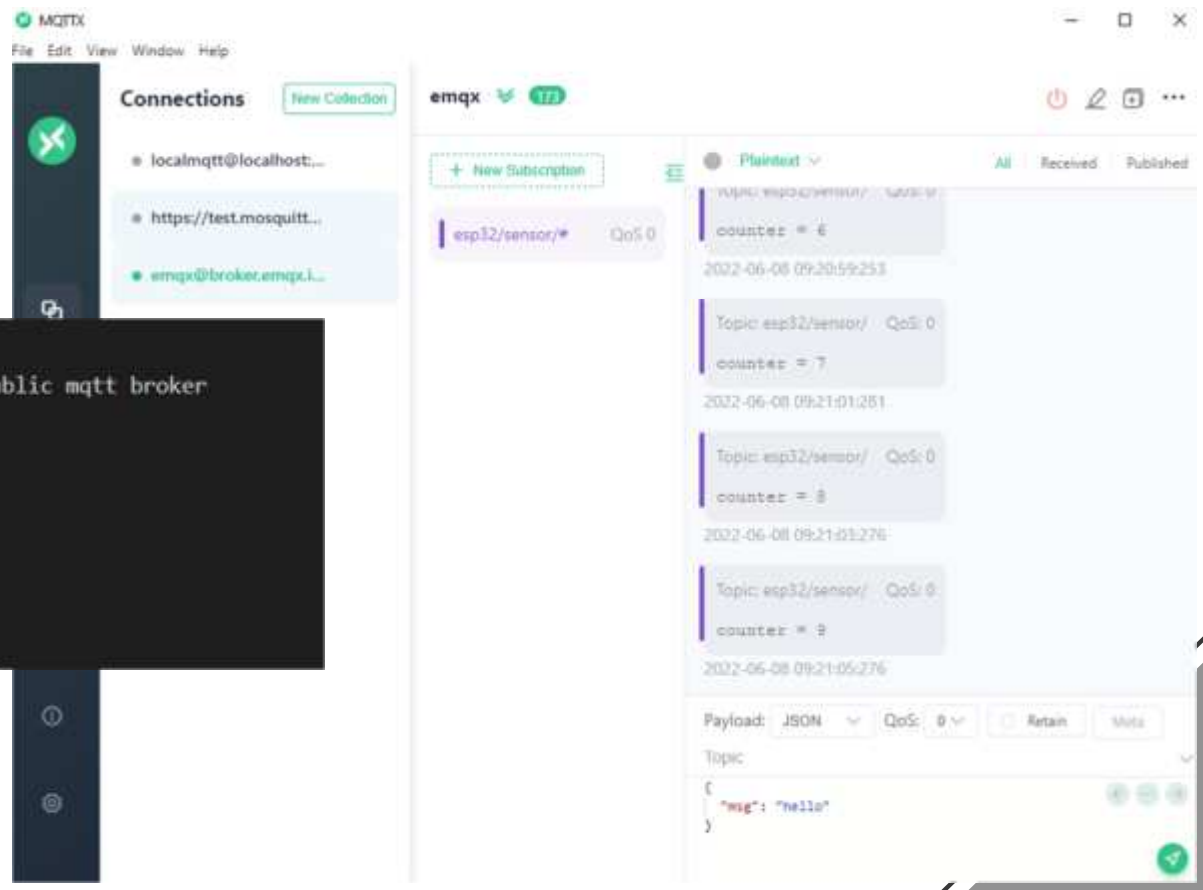
MQTT Client


MQTTX
File Edit View Window Help
About
Version: v1.7.3
Check for Updates
Releases
Support
MQTT X helps you develop MQTT services and applications faster.
Need a Fully Managed, Cloud-Native MQTT Messaging Service? Try EMQX Cloud now! 14-day trial, no credit card required.
To run MQTT Broker locally, EMQX is recommended: A Cloud-Native, Distributed MQTT 5.0 Broker for IoT.
Install open-source EMQX by using Docker:
docker run -d --name emqx -p 1883:1883 -p 8083:8083 -p 8883:8883 -p 8084:8084 -p 18083:18083 emqx/emqx
MQTT X
EMQX Cloud →
Follow us on GitHub →
EMQ ©2022 EMQ Technologies Co., Ltd.

TEST MQTT

FROM ESP32

```
Connecting to WiFi..
Connected to the WiFi network
The client esp32-client-7C:9E:BD:48:61:CC connects to the public mqtt broker
Public emqx mqtt broker connected
Message arrived in topic: esp32/sensor/
Message:counter = 0
-----------------------
Message arrived in topic: esp32/sensor/
Message:counter = 1
-----------------------
Message arrived in topic: esp32/sensor/
Message:counter = 2
```



THANKS

Do you have any question?

hasbiida@gmail.com