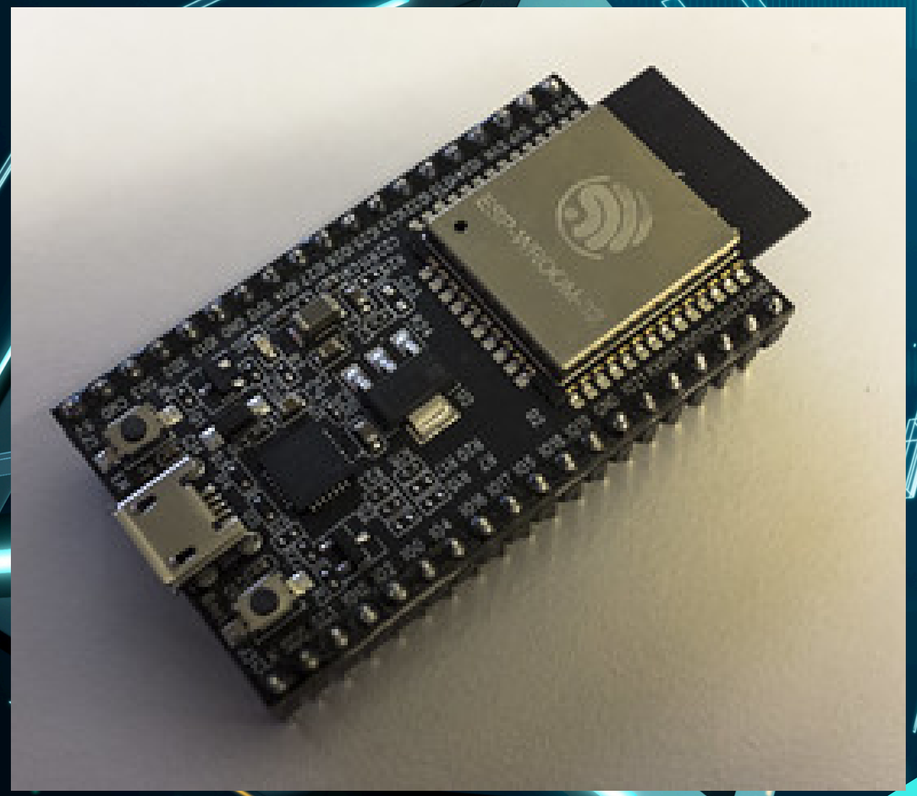


INTRODUCTION MIKROKONTROLER ESP32 DAN AKSES SENSOR

ESP32???

ESP32 adalah sebuah mikrokontroler SoC (System on Chip) yang diproduksi oleh Espressif Systems. Mikrokontroler ini dilengkapi dengan Wi-Fi 802.11 b/g/n dan Bluetooth 4.2, serta berbagai peripheral, menjadikannya ideal untuk proyek Internet of Things (IoT). ESP32 dikenal karena kemampuannya dalam memproses data, berkomunikasi, dan mengendalikan perangkat melalui jaringan nirkabel.

ESPRESSIF

CPU: Tensilica Xtensa 32-bit dual core, up to 240 MHz, 600 DMIPS.

Operating voltage: 3.3V

Memory: 448 KB ROM, 520 KB SRAM 16 KB SRAM in RTC, 1 Kbit of eFuse.

External Flash: 512 KB to (4 x 16) MiB

WiFi (802.11): b/g/n/e/i

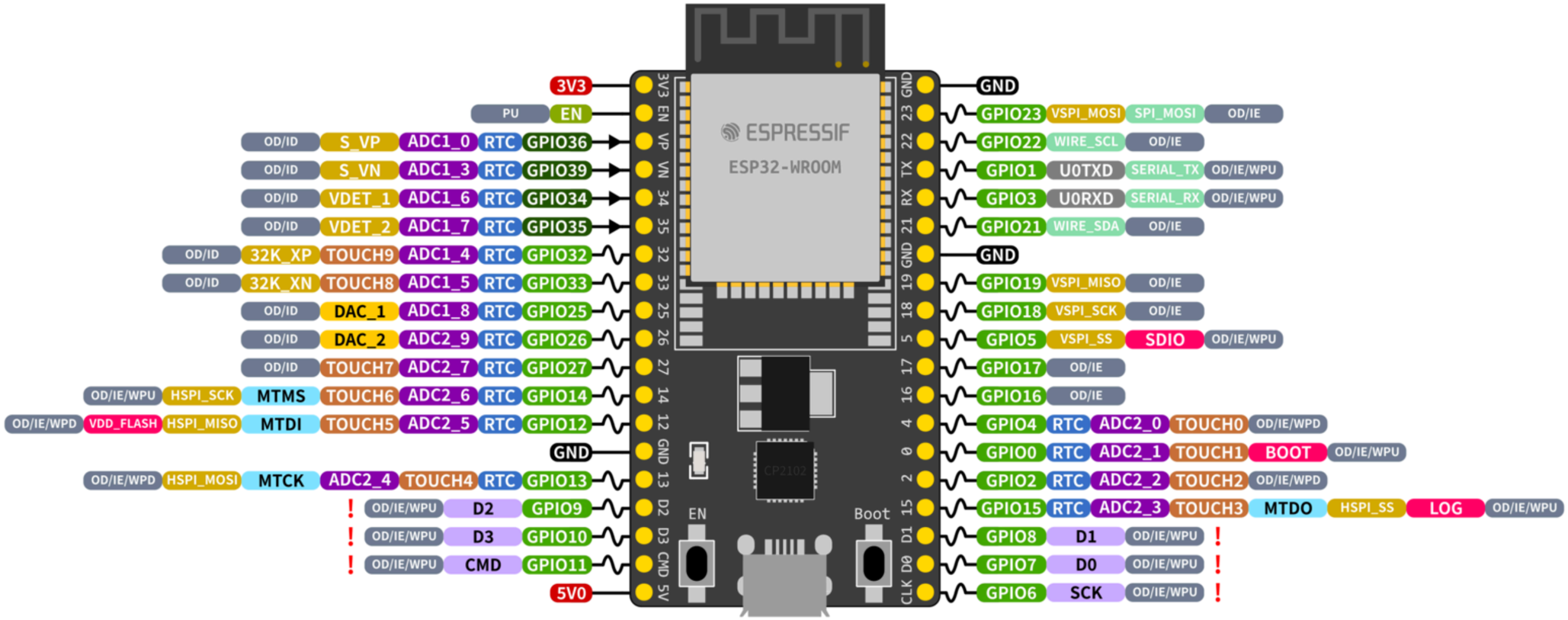
Bluetooth: v4.2 BR/EDR & BLE

Peripherals: GPIOs, PWM, ADC, DAC, I2S, UART, SPI, I2C, CAN, RMII, Cap Touch

ESP8266 VS ESP32

Specifications	ESP8266	ESP32
MCU	Xtensa Single-Core 32-bit L 106	Xtensa Dual-Core 32-bit LX6 600 DMIPS
802.11 b/g/n Wi-Fi	Yes, HT20	Yes, HT40
Bluetooth	N/A	Bluetooth 4.2 and below
Typical Frequency	80 MHz	160 MHz
SRAM	160 kBytes	512 kBytes
Flash	SPI Flash up to 16 MBytes	SPI
GPIO	17	36
Hardware / Software PWM	None / 8 Channels	1 / 16 Channels
SPI / I2C / I2S / UART	2/1/2/2	4/2/2/2
ADC	10-bit	12-bit
CAN	N/A	1
Ethernet MAC Interface	N/A	1
Touch Sensor	N/A	Yes
Temperature Sensor	N/A	Yes
Working Temperature	-40° C – 125° C	-40° C – 125° C

ESP32-DevKitC



ESP32 Specs

32-bit Xtensa® dual-core @240MHz
Wi-Fi IEEE 802.11 b/g/n 2.4GHz
BLuetooth 4.2 BR/EDR and BLE
520 KB SRAM (16 KB for cache)
448 KB ROM
34 GPIOs, 4x SPI, 3x UART, 2x I2C,
2x I2S, RMT, LED PWM, 1 host SD/eMMC/SDIO,
1 slave SDIO/SPI, TWAI®, 12-bit ADC, Ethernet

PWM Capable Pin
GPIOX GPIO Input Only
GPIOX GPIO Input and Output
DAC_X Digital-to-Analog Converter
DEBUG JTAG for Debugging
FLASH External Flash Memory (SPI)
ADCX_CH Analog-to-Digital Converter
TOUCHX Touch Sensor Input Channel
OTHER Other Related Functions
SERIAL Serial for Debug/Programming
ARDUINO Arduino Related Functions
STRAP Strapping Pin Functions

RTC RTC Power Domain (VDD3P3_RTC)
GND Ground
PWD Power Rails (3V3 and 5V)
! Pin Shared with the Flash Memory
Can't be used as regular GPIO

GPIO STATE
WPU: *Weak Pull-up (Internal)*
WPD: *Weak Pull-down (Internal)*
PU: *Pull-up (External)*
IE: *Input Enable (After Reset)*
ID: *Input Disabled (After Reset)*
OE: *Output Enable (After Reset)*
OD: *Output Disabled (After Reset)*

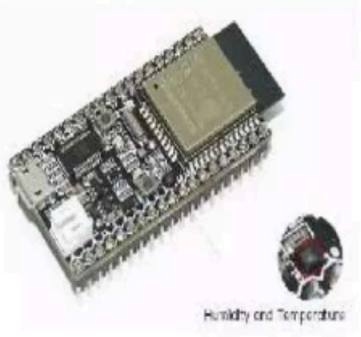
ESP32 Boards

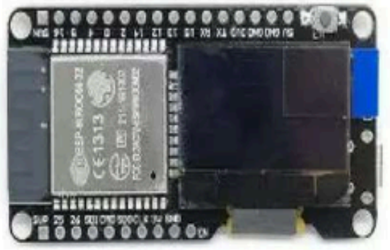
Huzzah

Hornbill

ARS01119B

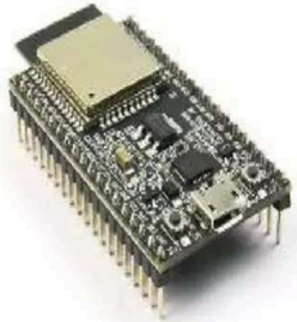
AnalogLamb ESP32





and many more…

ESP32 Boards

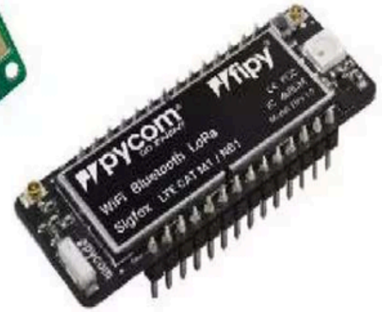


Espressif DevKit

ESP32 Things

ESP320

ESP32 N1



FiPy

Nano32

Widora Air

Pesky ESP32

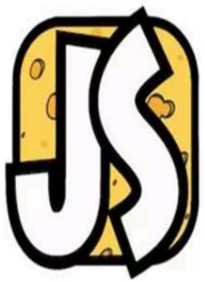
and many more…

ESP32 Development Platforms

ESP-IDF

Espruino

ARDUINO



ESP32 Development Tools / IDE

Espruino

Arduino IDE

Cara Memulai Penggunaan ESP32 dengan Arduino IDE

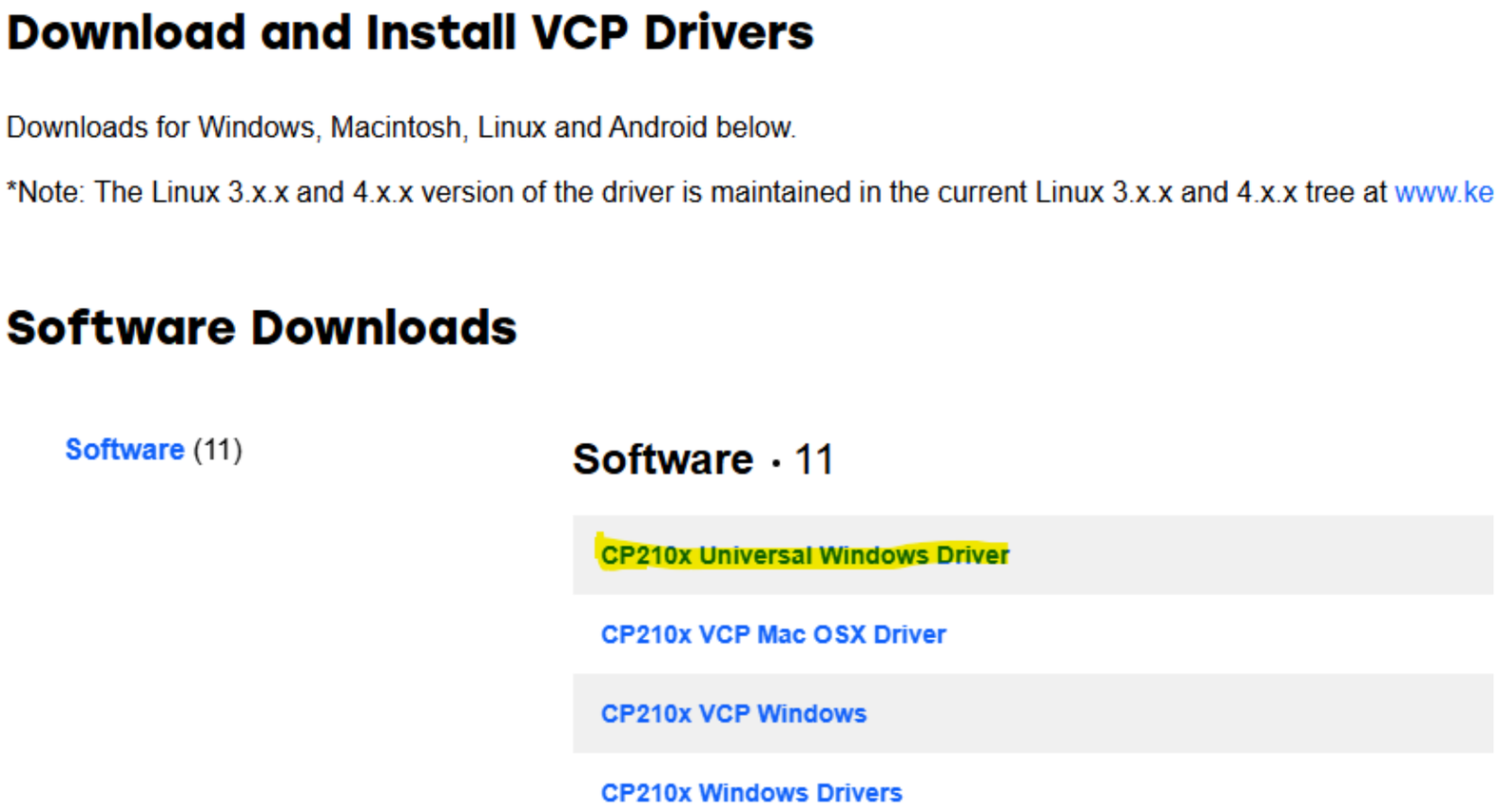
1. Mempunyai Board ESP32
2. Install Driver ESP32
3. Install Software Arduino IDE
4. Install Board ESP32
5. Porgram lalu upload


LETS GET STARTED

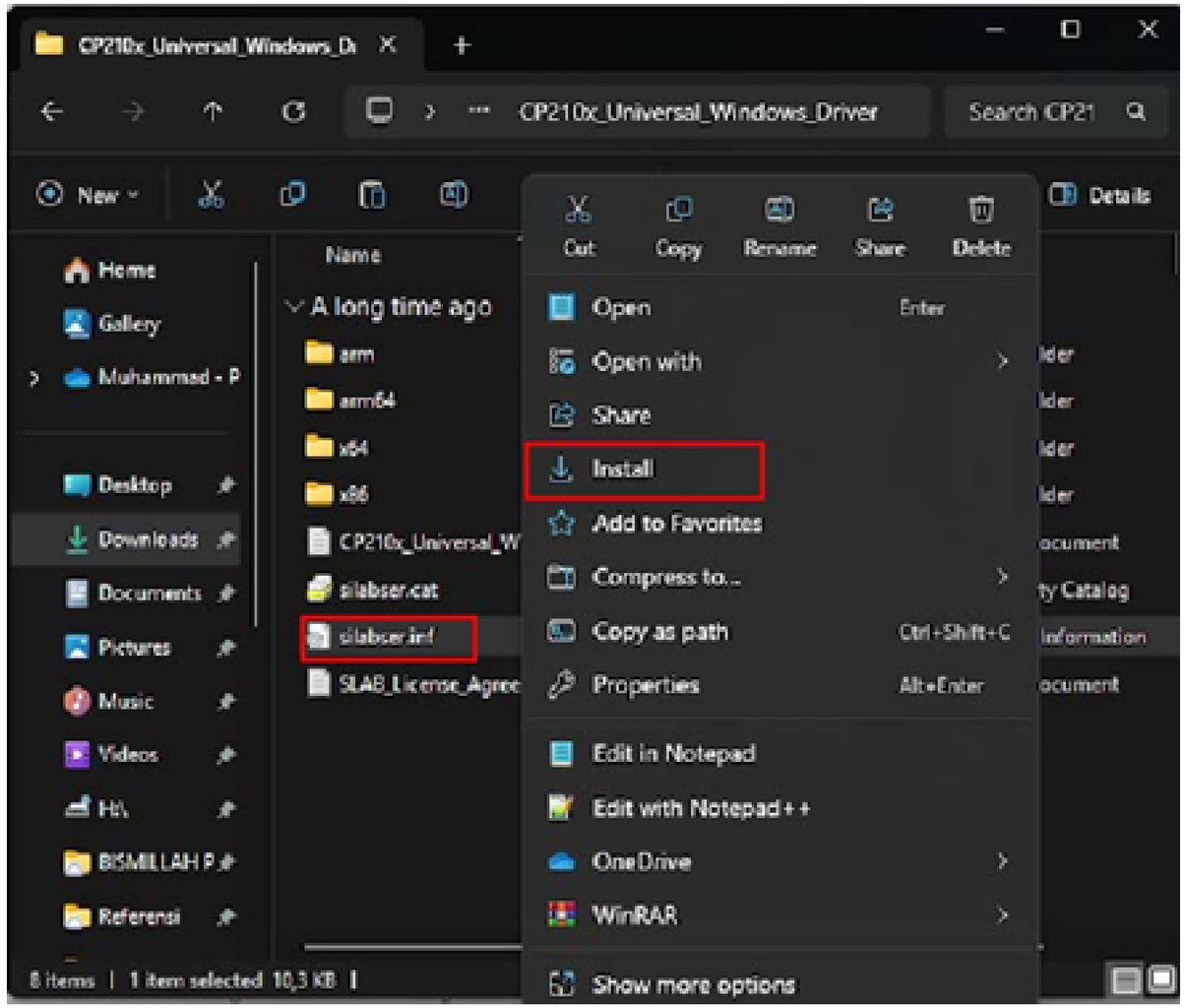
Preparation

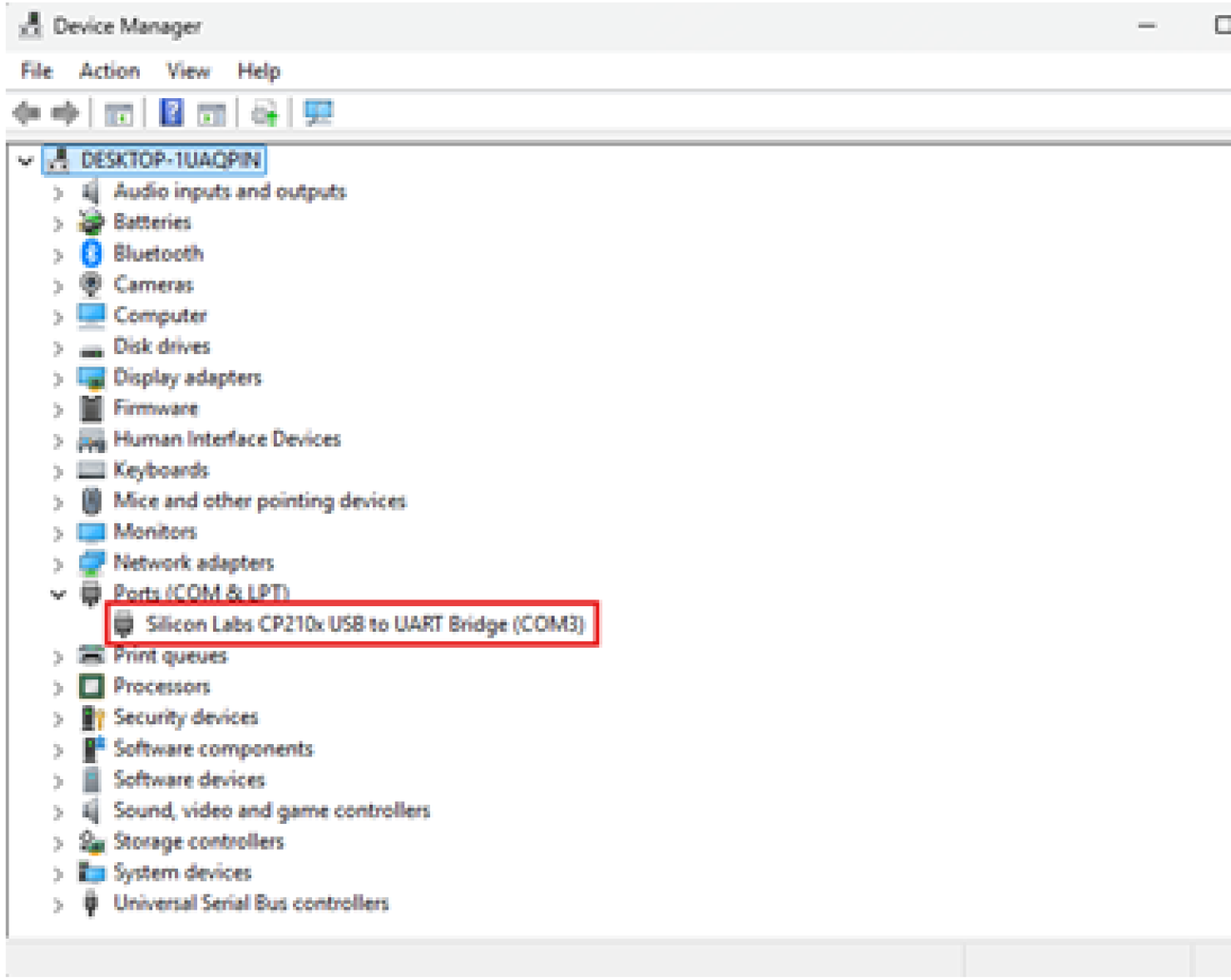
1. Siapkan koneksi internet kemudian kunjungi website resmi Arduino di www.arduino.cc/en/software.
2. Pilih versi Arduino IDE yang sesuai dengan sistem operasi komputer Anda (Windows, Mac, atau Linux) bisa menggunakan Arduino versi 2.0 atau versi 1.8.x (legacy IDE). Setelah proses download selesai kemudian install.

3. Install driver ESP32 dengan cara kunjungi website Silicon Labs https://www.silabs.com/developer-tools/usb-to-uart-bridge-vcp-drivers.

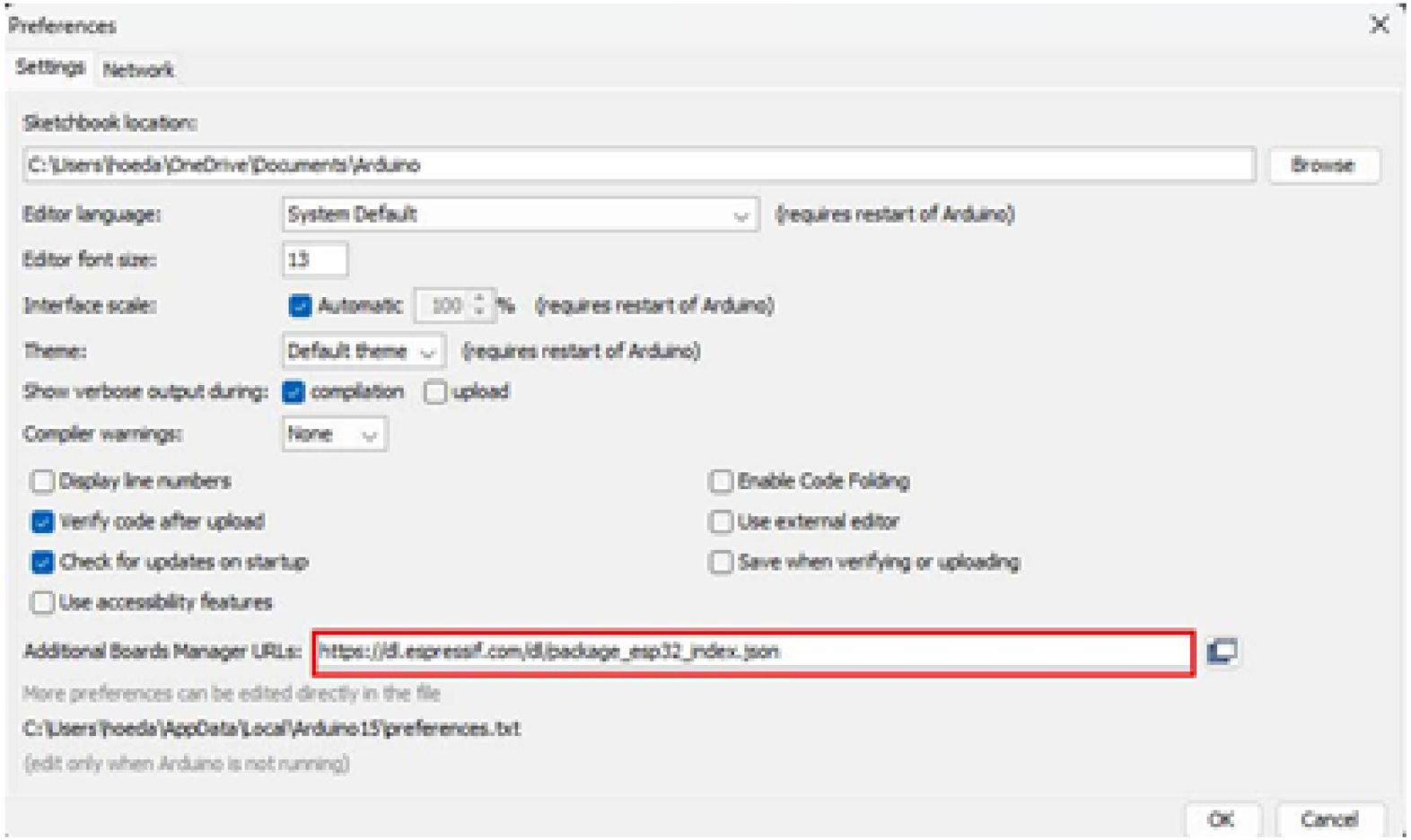


4. Ekstrak file yang sudah didownload, kemudian buka foldernya. Klik kanan pada file yang bernama silabser.inf >> Install.

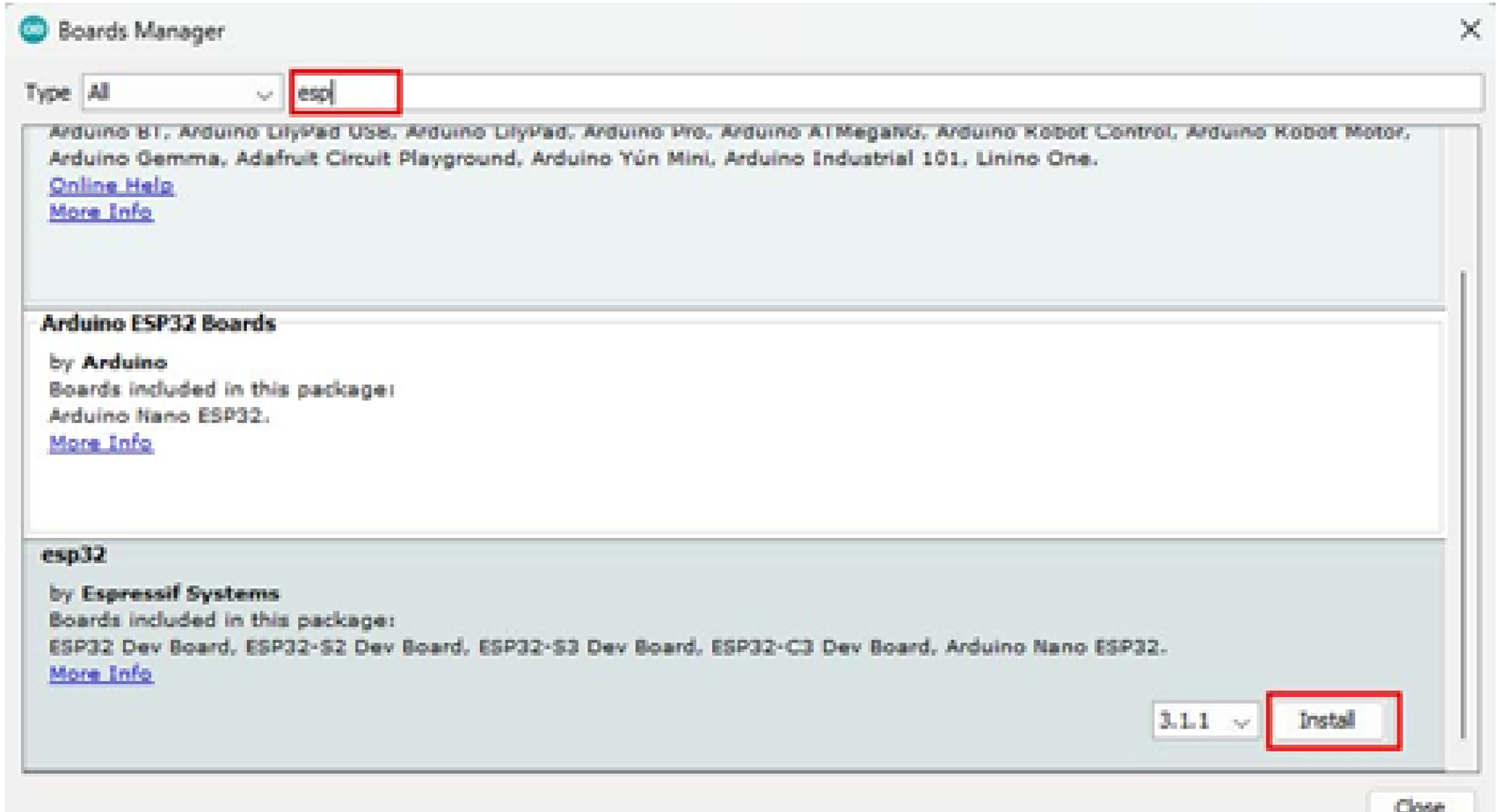




5. Selanjutnya install board ESP32 dengan cara buka software Arduino IDE >> pilih menu File >> Preference.
6. Pada kolom Aditional Boards Manager URLs masukkan link berikut: https://dl.espressif.com/dl/package_esp32_index.json, kemudian klik “OK”.



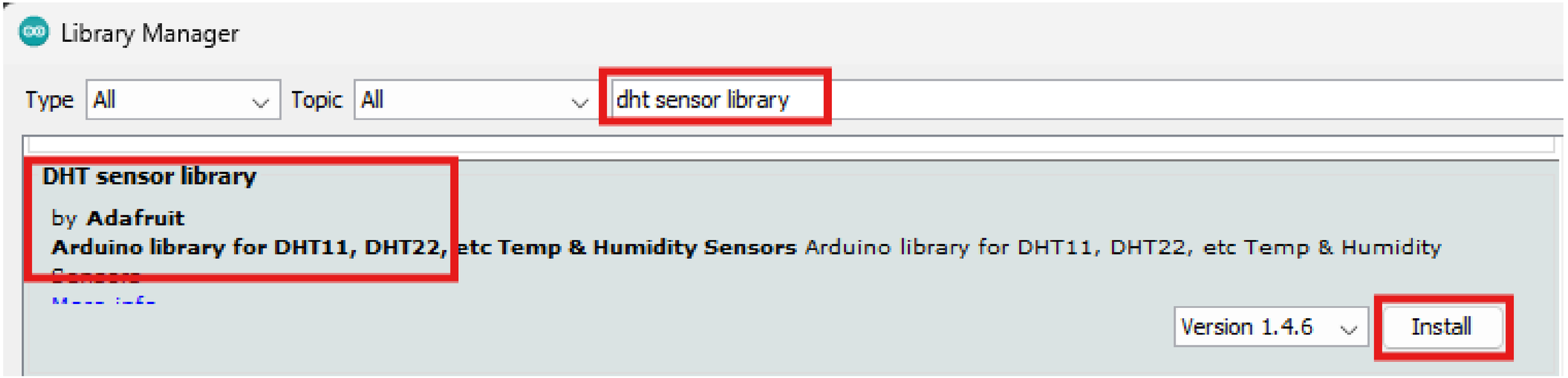
7. Selanjutnya install board ESP32 dengan cara klik Tools >> Board >> Boards Manager.
8. Kemudian pada jendela Boards Manager, ketik ESP >> Pilih esp32 by Esperssif Systems >> Install, tunggu hingga proses instalasi selesai (notes: harus terkoneksi dengan internet).



Contoh coding bisa di download di:
https://github.com/hoedaa/Coding__Sensor_Aktuator

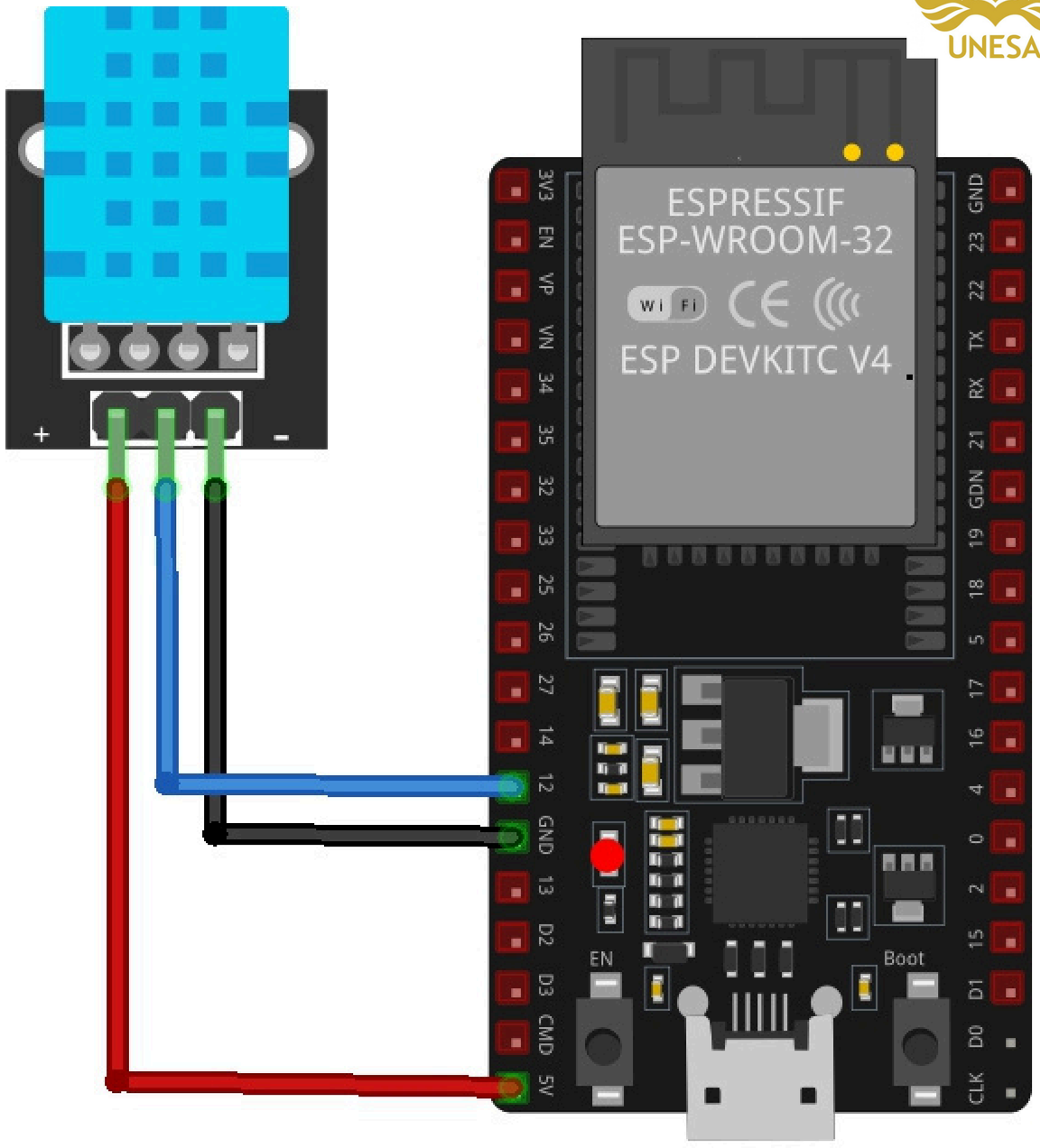
Sensor DHT11

install library sensor dht11 dengan cara:
Klik menu "Sketch" >> "Include Library" >> "Manage Libraries". Pada kotak pencarian, ketik "DHT", Cari "DHT sensor library" by Adafruit >> Klik "Install".



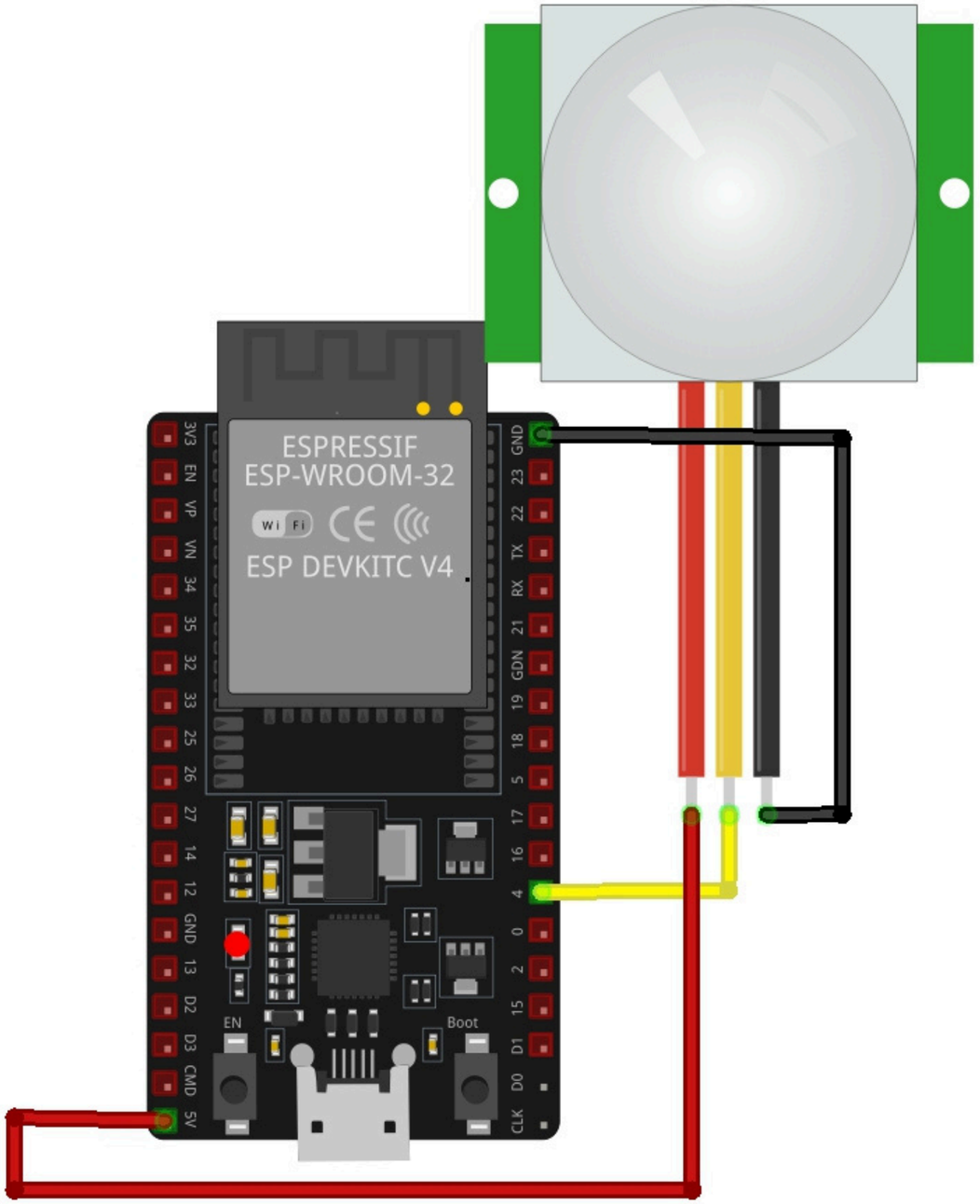
Wiring Sensor DHT11

DHT 11	ESP32
+	5V
-	GND
OUT	12



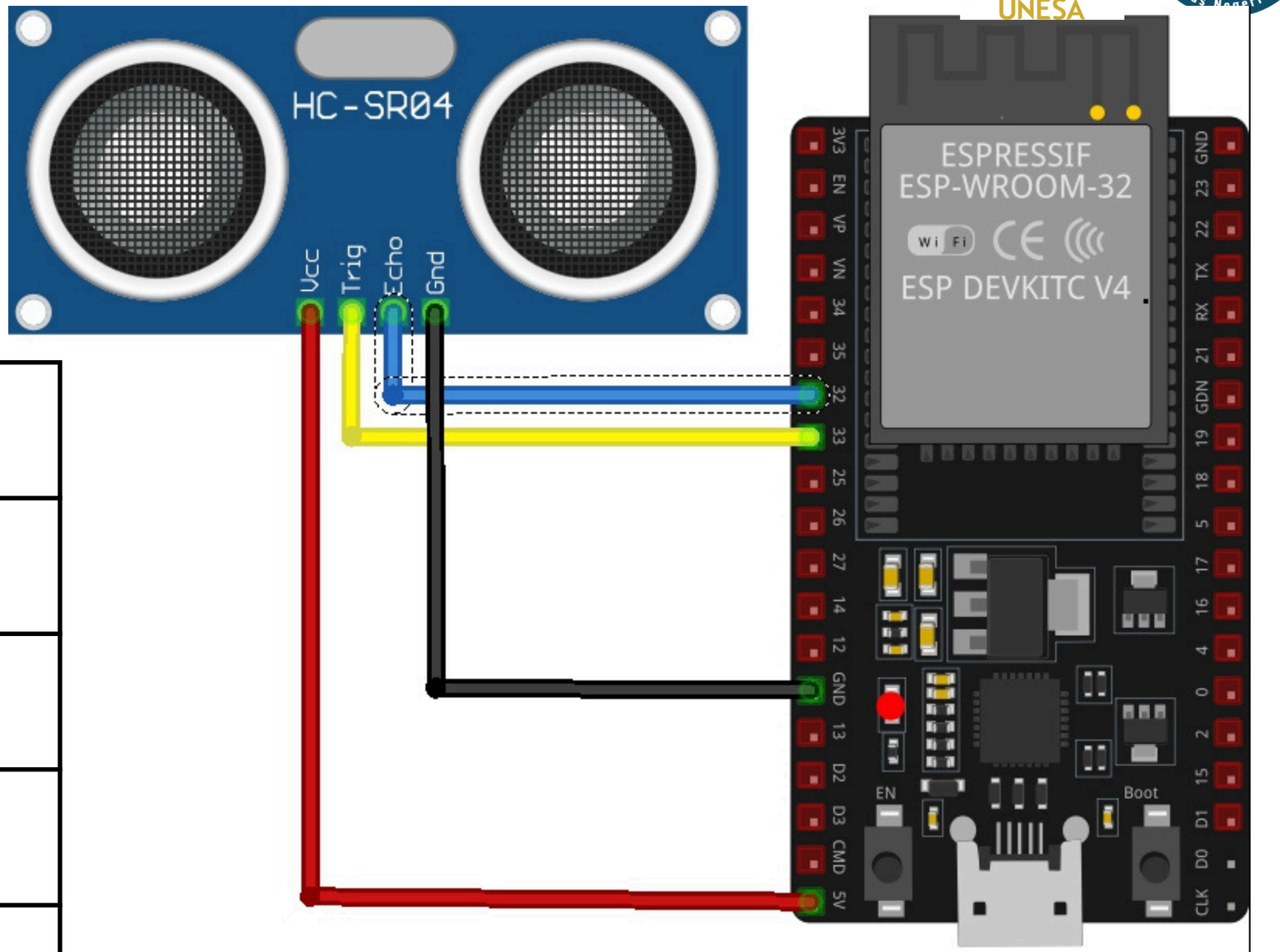
Wiring Sensor PIR

PIR	ESP32
+	5V
-	GND
OUT	4



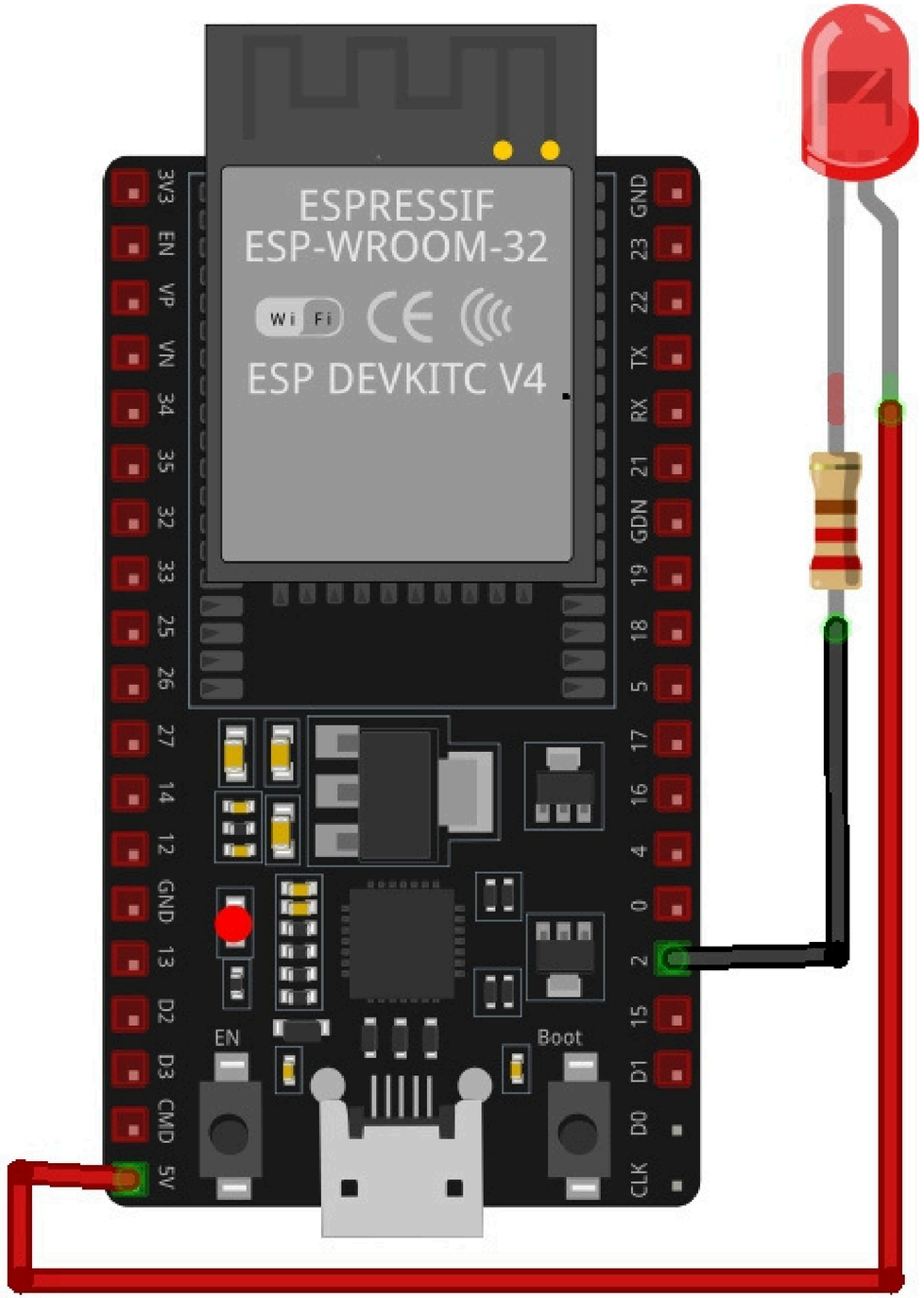
Wiring Sensor Ultrasonik HC-SR04

HC-SR04	ESP32
+	5V
-	GND
TRIG	33
ECHO	32



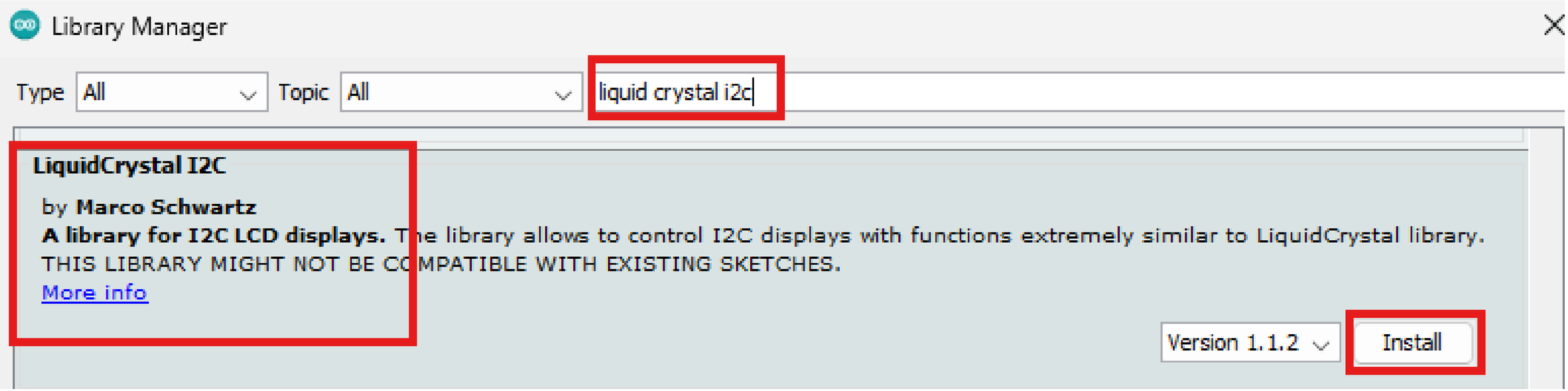
Wiring Blink LED

LED	ESP32
+ (Anoda)	5V
- (Katoda)+Resistor 220 ohm	GND



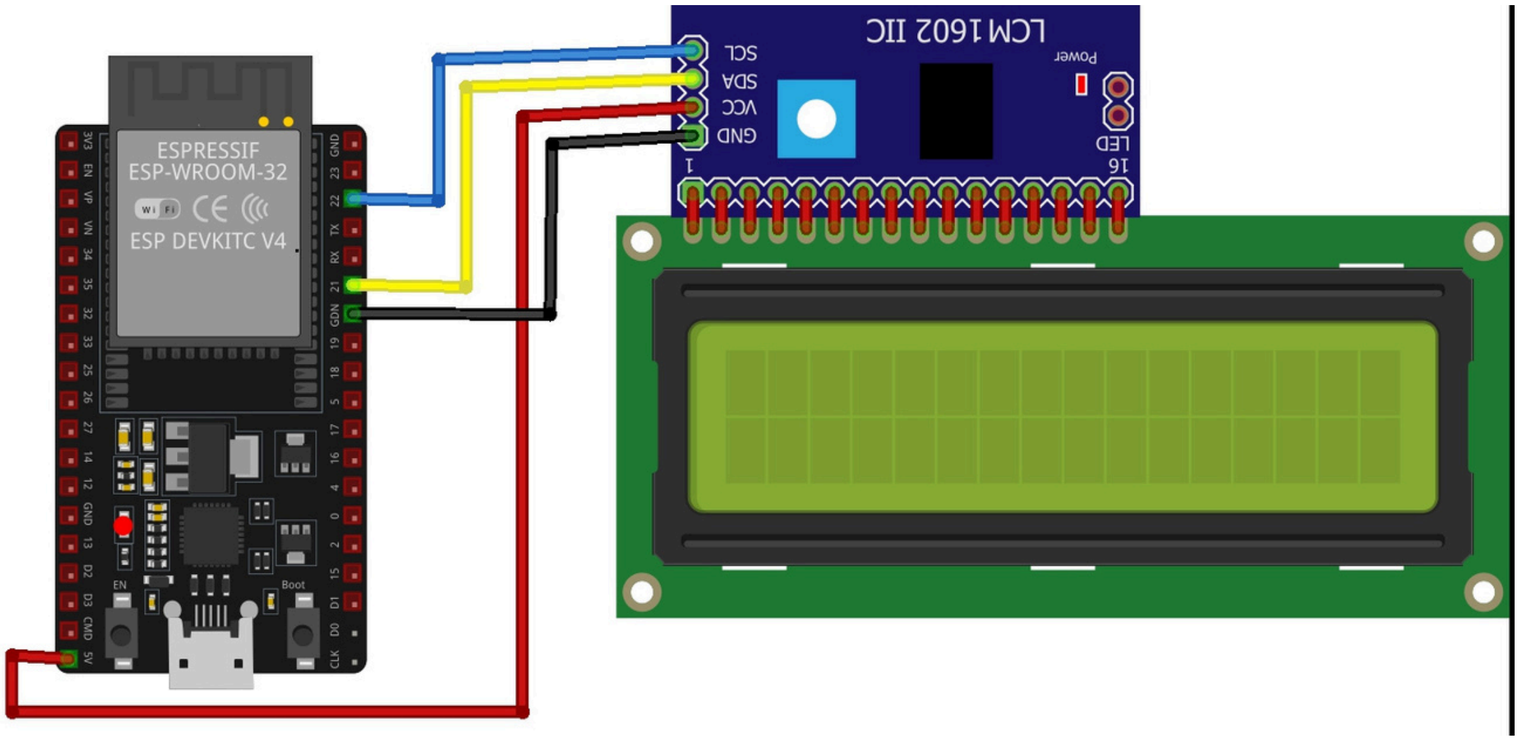
LCD I2C

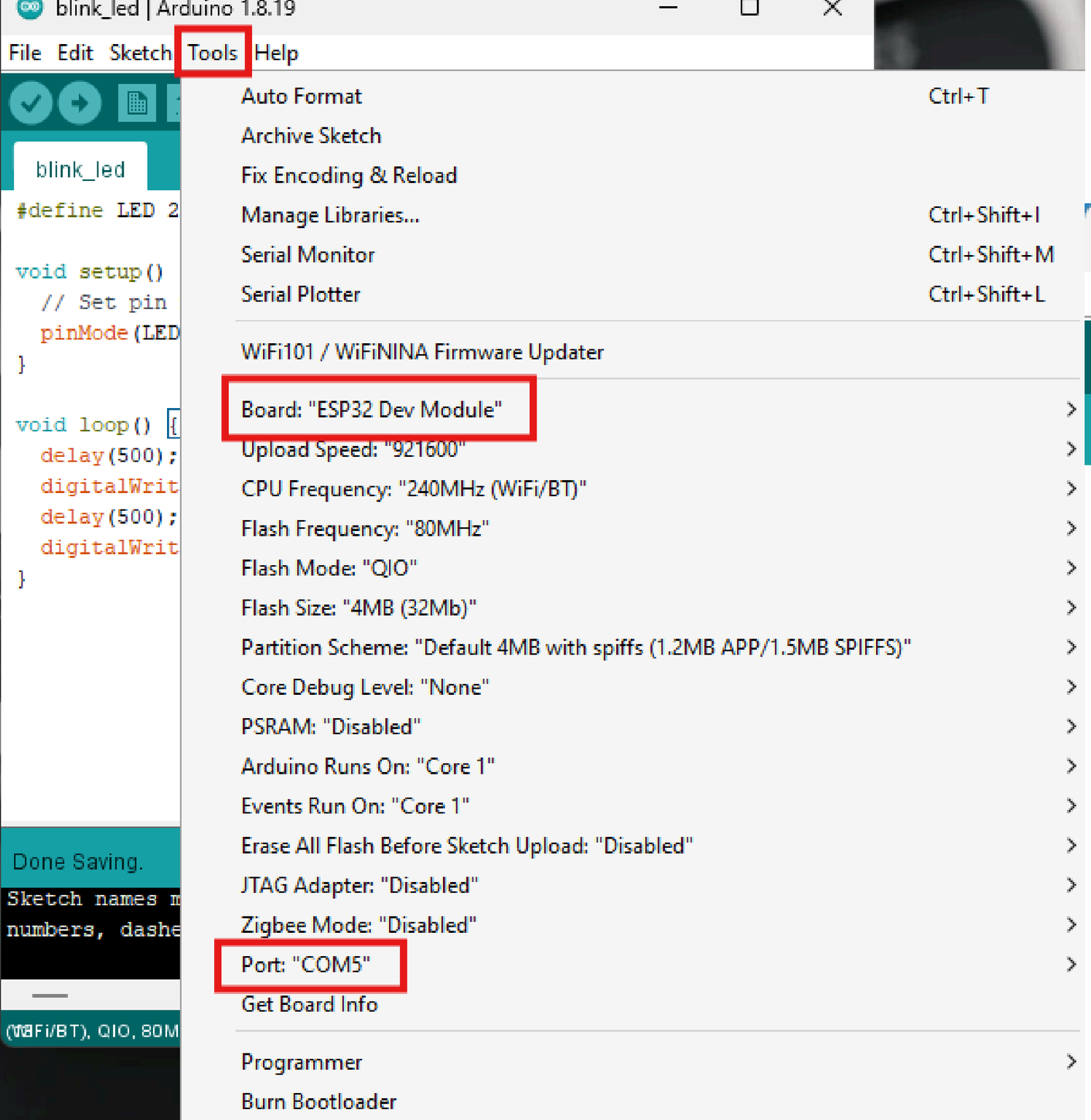
install library LCD I2C dengan cara:
Klik menu "Sketch" >> "Include Library" >> "Manage Libraries". Pada kotak pencarian, ketik "LiquidCrystal I2C". Cari "LiquidCrystal I2C". Klik "Install".

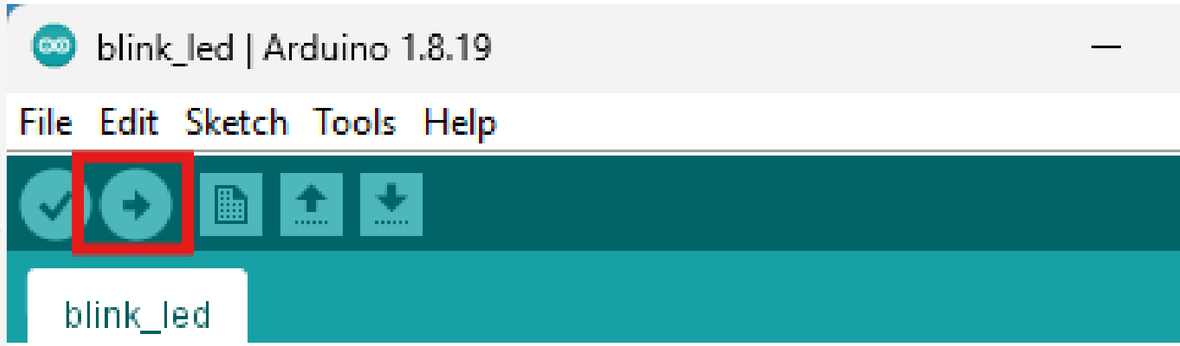


Wiring LCD I2C

LCD I2C	ESP32
+	5V
-	GND
SCL	22
SDA	21



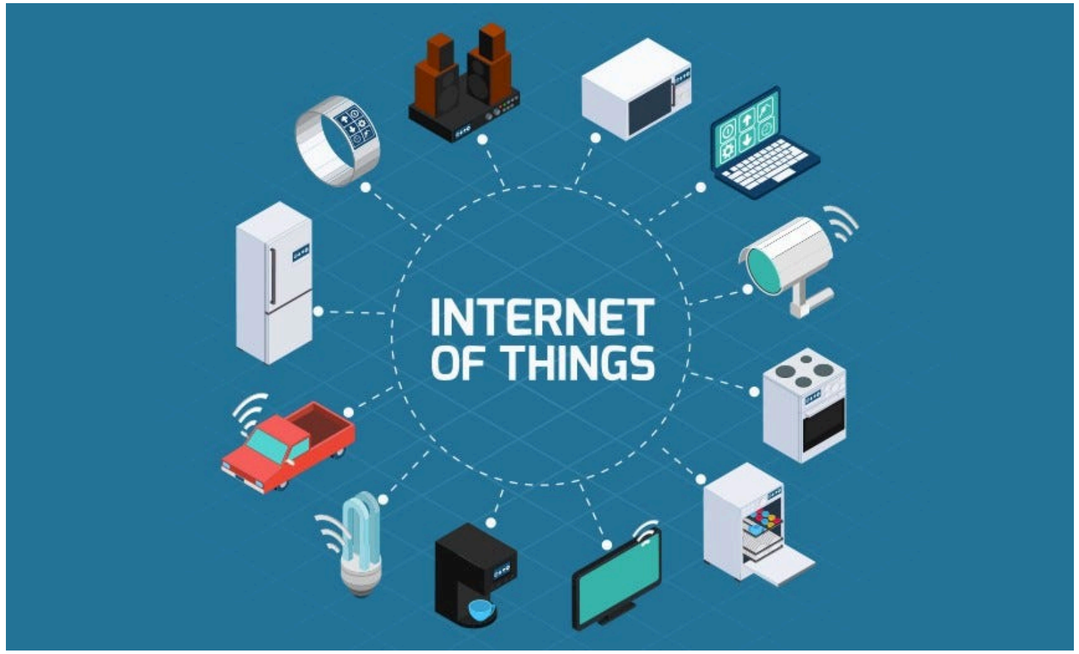




ctrl+U

Upload program

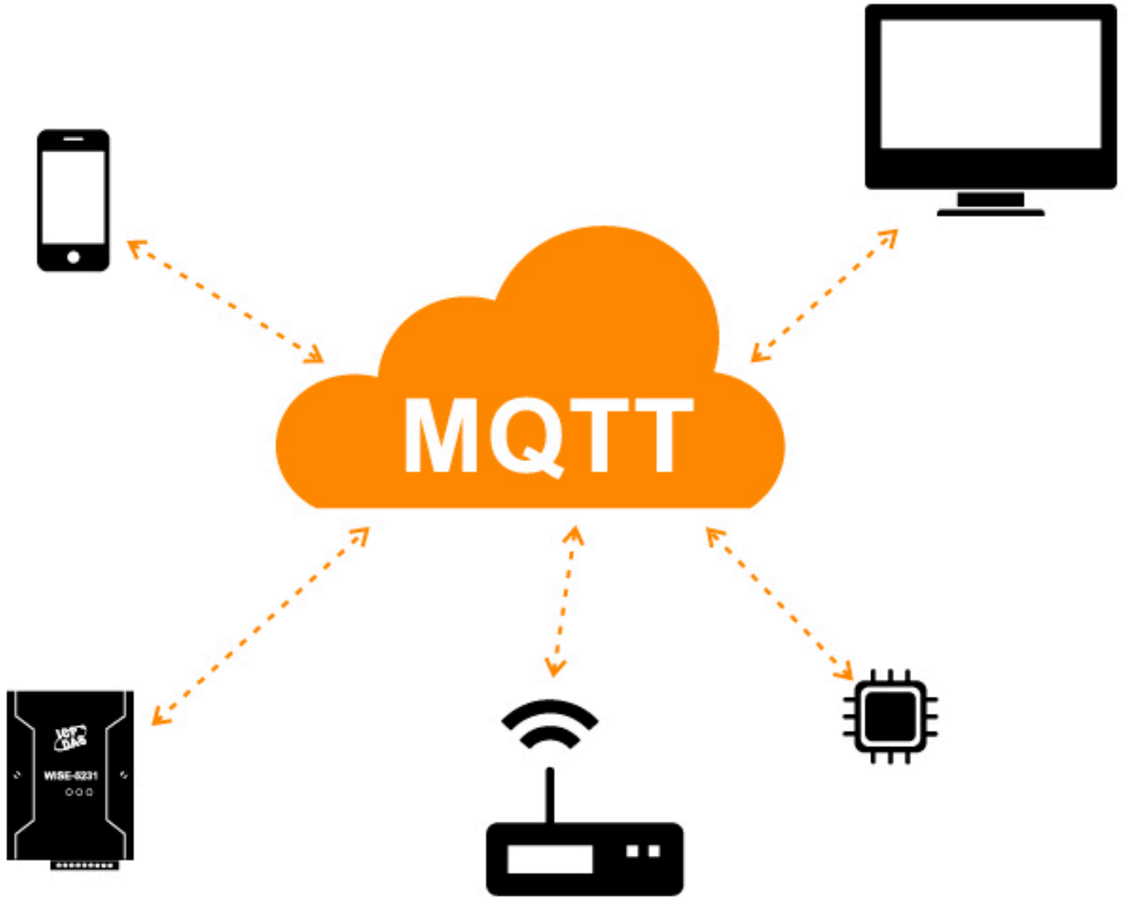
Apa itu IoT???



Internet of Things (IoT) adalah konsep teknologi yang memungkinkan perangkat fisik terhubung ke internet dan bertukar data secara otomatis.

Protokol Komunikasi IoT

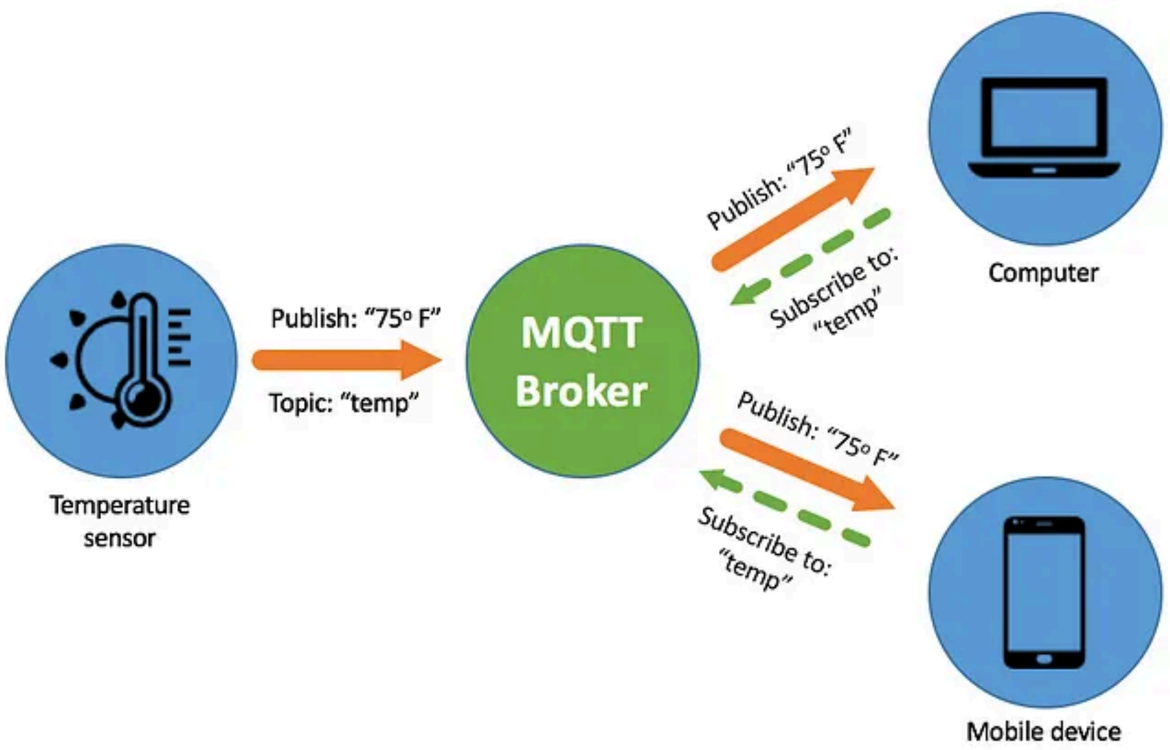
Aturan dan cara yang memungkinkan perangkat IoT untuk saling berkomunikasi. Protokol ini sangat penting karena perangkat IoT memiliki berbagai karakteristik, seperti kemampuan pemrosesan yang terbatas, ruang penyimpanan yang kecil, dan berbagai jenis konektivitas. Karena itu, protokol komunikasi harus dirancang agar efisien dalam penggunaan energi dan bandwidth.



MQTT adalah singkatan dari Message Queuing Telemetry Transport, sebuah protokol komunikasi yang ringan dan terbuka, dirancang untuk komunikasi antar perangkat dengan sumber daya terbatas, seperti perangkat IoT (Internet of Things). Protokol ini menggunakan model publish-subscribe, di mana perangkat (publisher) mengirimkan pesan ke topik tertentu, dan perangkat lain (subscriber) berlangganan topik tersebut untuk menerima pesan.

Kenapa harus MQTT???

1. Ringan & Efisien: MQTT berukuran kecil, hemat sumber daya, cocok untuk mikrokontroler, dan hemat bandwidth (data mulai dari 1 kb).
2. Dua Arah: Mendukung komunikasi dari perangkat ke cloud/server dan sebaliknya.
3. Tahan Jaringan Lemah: Tetap efisien di jaringan lambat berkat jejak data kecil dan sesi yang persisten.
4. Pengiriman Andal: Menyediakan QoS 0, 1, dan 2 untuk pengiriman pesan sesuai kebutuhan keandalan.
5. Keamanan: Mendukung enkripsi TLS dan autentikasi modern seperti OAuth.
6. Skalabilitas Tinggi: Bisa menangani jutaan perangkat dengan dukungan skalabilitas vertikal dan horizontal.



1. Publisher
Publisher mengirimkan pesan atau perintah dengan topik tertentu ke Broker

2. Subscriber
Subscriber menerima pesan dari Broker sesuai dengan topik yang di-subscribe

3. Broker
Broker menerima pesan dari Publisher dan meneruskannya ke Subscriber yang men-subscriber topik sesuai dengan pesan tersebut

4. Topic
Setiap pesan yang ada pada konsep Publish dan Subscribe memiliki kategori yang dinamakan dengan Topic