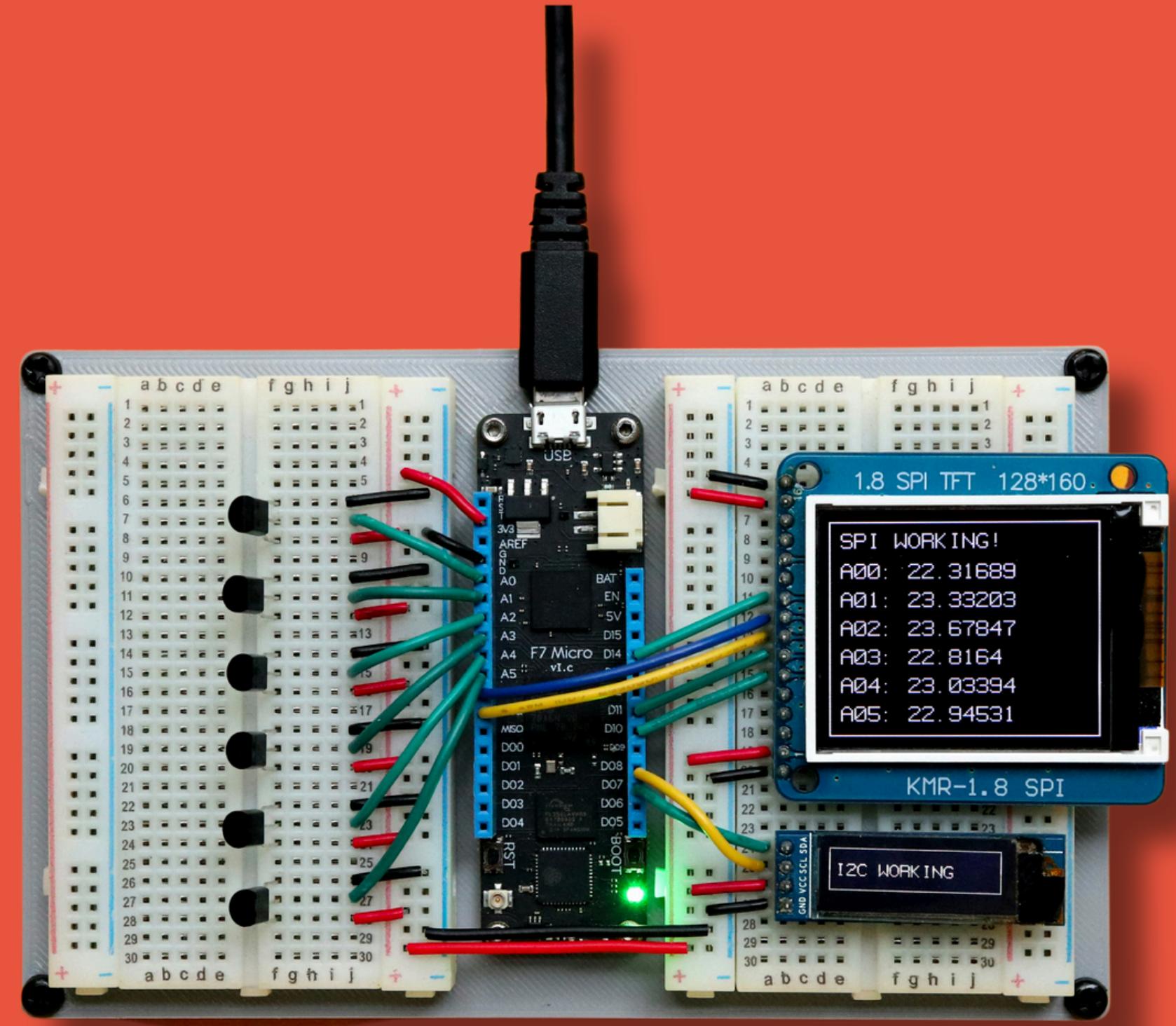


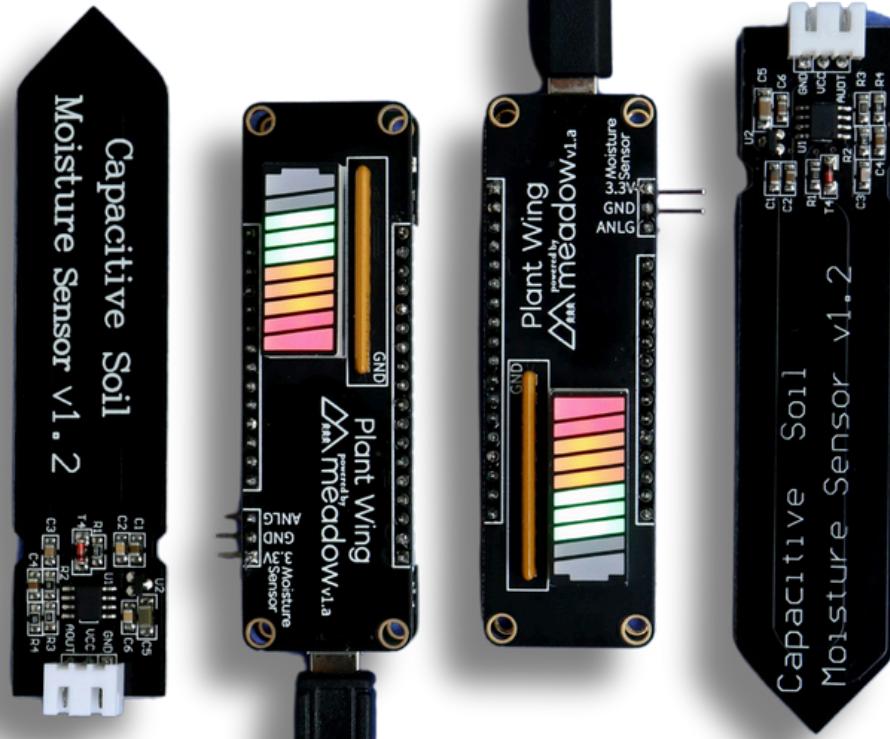
# Pengenalan Internet of Things

Mulai belajar →

BELAJAR IOT BAGI PEMULA



# Definisi IoT (Internet of Things)



**IoT adalah sebuah konsep dimana benda-benda fisik dapat saling berkomunikasi melalui jaringan internet.** Benda fisik disini adalah seluruh benda fisik berbentuk apapun.

Salah satu penerapan IoT misalnya dalam sistem pemantauan suhu dan kelembaban di gudang, sehingga dapat memastikan bahwa produk yang disimpan tetap dalam kondisi yang baik karena dapat dipantau dari manapun walau jarak.

# Pemantauan Armada



IoT dapat digunakan untuk memantau posisi, kondisi, dan performa kendaraan dalam suatu armada, sehingga dapat meningkatkan efisiensi dan produktivitas.

Misalnya pemantauan posisi kendaraan secara *realtime* dengan mengirim data GPS lewat internet.

# Sistem Pengatur Suhu



IoT dapat digunakan untuk mengintegrasikan sistem pengatur suhu rumah dengan sensor suhu dan pengontrol jarak jauh, sehingga dapat mengatur suhu secara otomatis. Seperti suhu rumah, kulkas, kompor, dl.

# Pengontrol Pintu dan Jendela



IoT dapat digunakan untuk mengintegrasikan sistem pengontrol pintu dan jendela dengan sensor gerak dan pengontrol jarak jauh, sehingga dapat membuka dan menutup pintu dan jendela secara otomatis.



# Sistem Pemantauan Peralatan

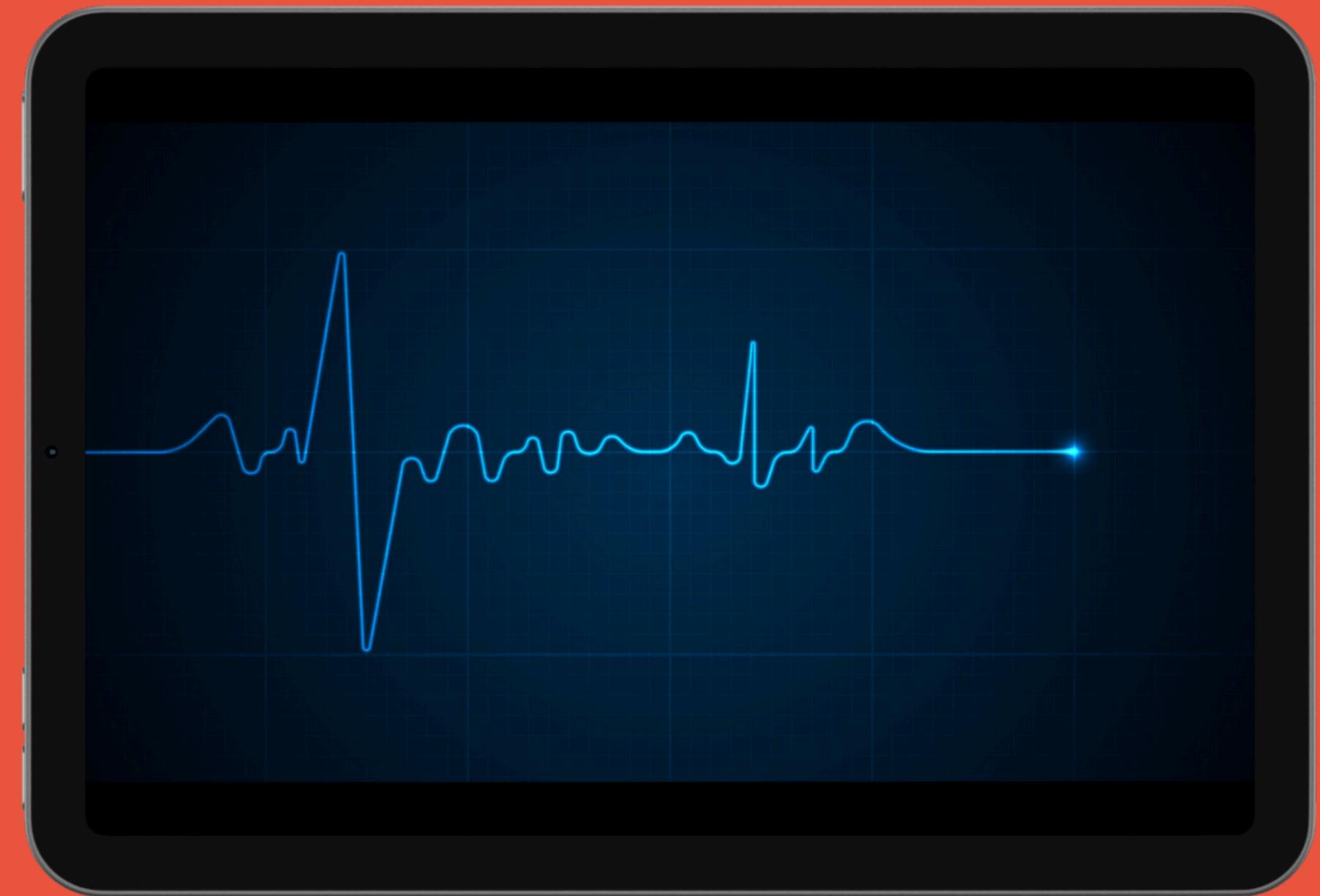


IoT dapat digunakan untuk memantau kondisi peralatan rumah seperti AC, mesin cuci, dan kulkas, dan memberikan peringatan jika perlu dilakukan perawatan atau perbaikan.

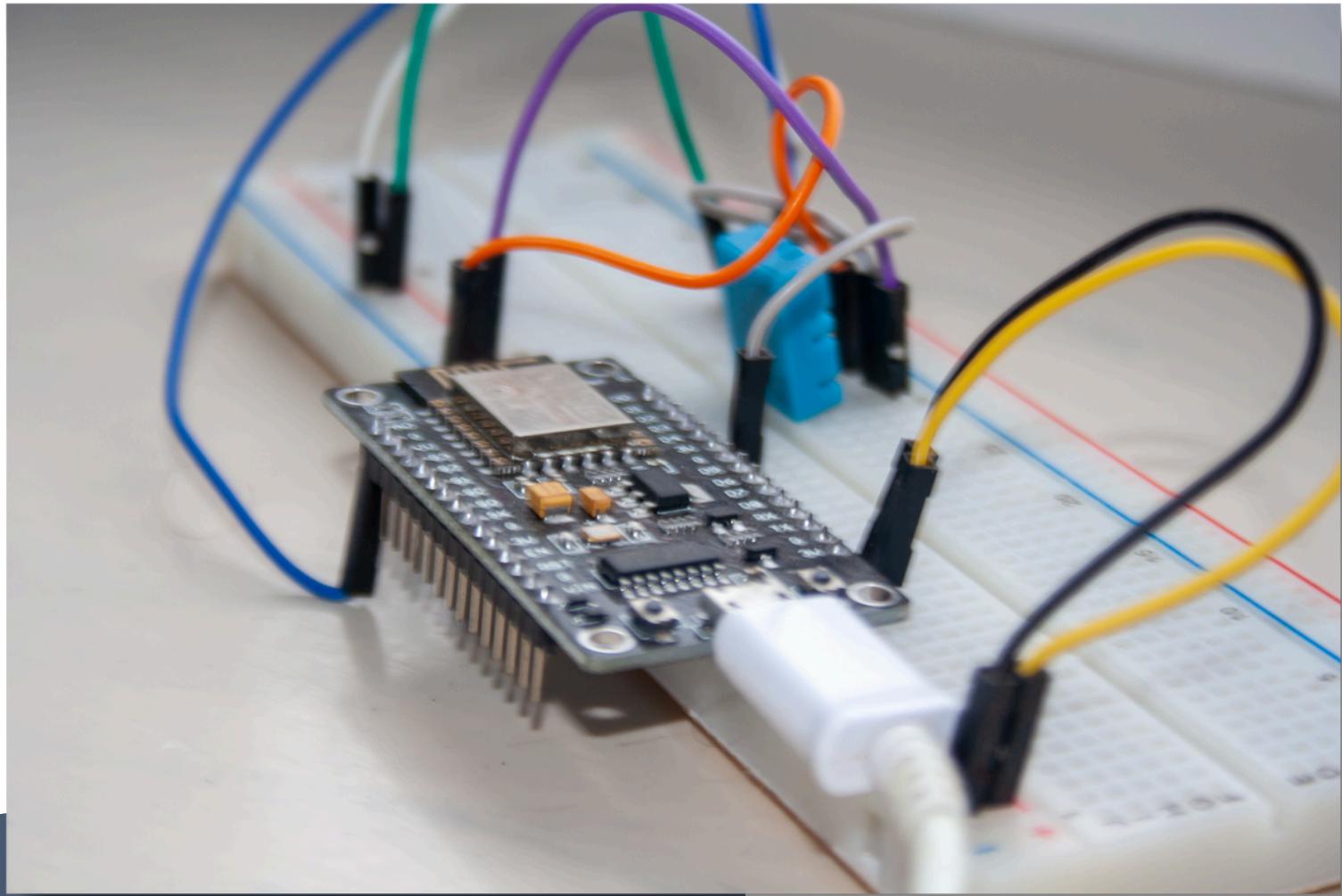


# Bentuk Sinyal dan Alur Data

Mulai belajar →



# Bentuk Sinyal



## Sinyal Digital

**Hanya terdapat 2 nilai** yaitu 0 atau 1, tinggi atau rendah, ya atau tidak, basah atau kering, bergerak atau berhenti, dan seterusnya.

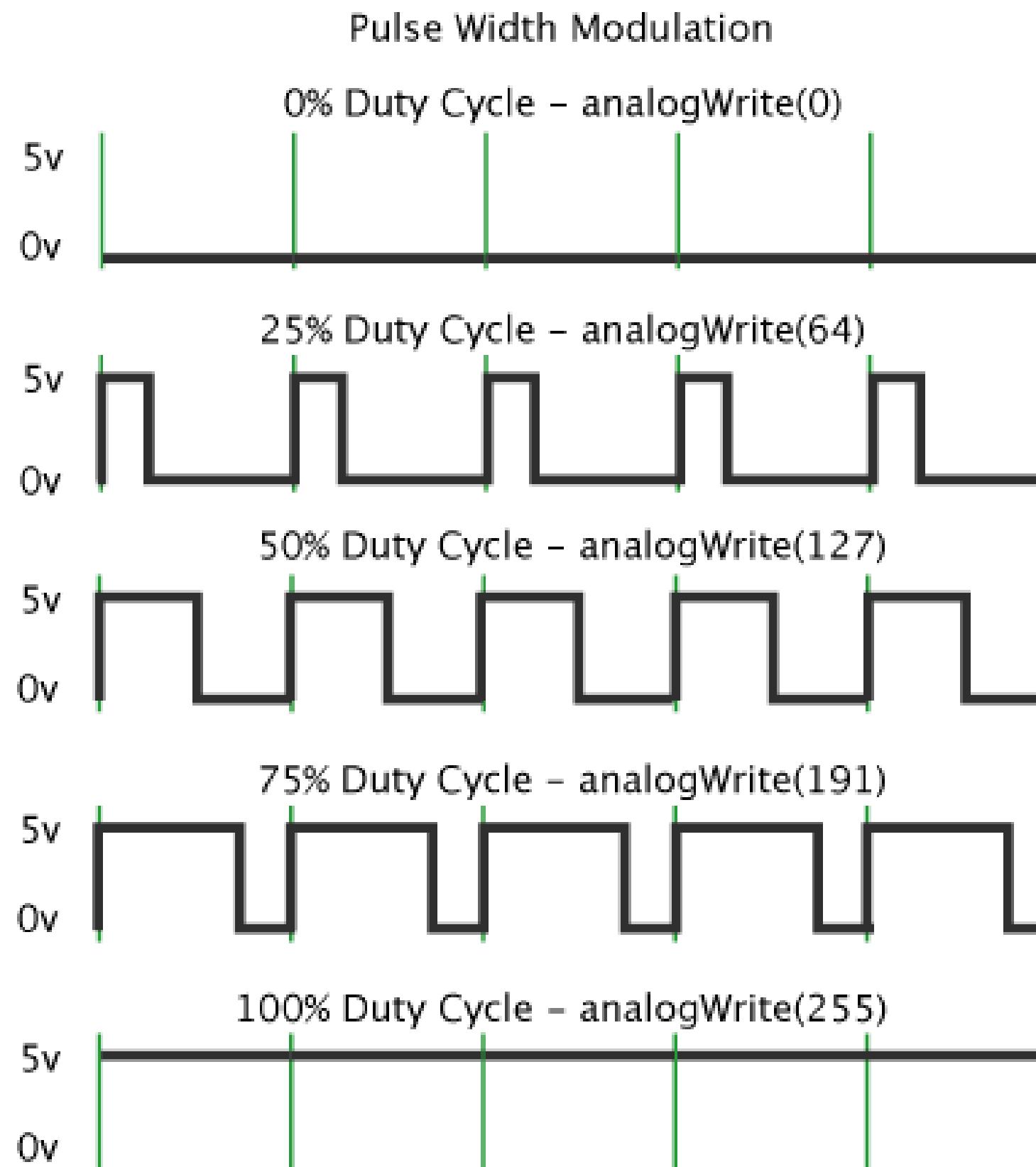
## Sinyal Analog

**Memiliki banyak nilai bukan hanya 2,** seperti suhu ruangan sebesar 45 derajat celcius, ketinggian air sungai 5 meter, intensitas cahaya 78%, dan seterusnya.

# Contoh Perbedaan Penggunaan Sinyal Digital dan Analog

Contoh	Sinyal Digital	Sinyal Analog
Lampu LED	Nyala atau Mati	Tingkat Keceran
Sensor Hujan	Basah atau Kering	Besaran Curah Hujan
Sensor Suhu	Panas atau Dingin	Tingkat Suhu
Motor DC	Berputar atau Diam	Kecepatan Putaran

# Teknik PWM (Pulse Width Modulation)



Terdapat suatu teknik dimana teknik tersebut berguna untuk menciptakan hasil berupa analog dengan cara mentransmisikan sinyal digital. Atau dengan kata lain teknik ini adalah **teknik untuk mengubah sinyal digital menjadi seolah-olah seperti sinyal analog.**

- Teknik tersebut dinamakan dengan PWM (Pulse Width Modulation).

# Input



# Proses



# Output

## Alur Data

Setiap pengolahan data untuk menjadi informasi selalu minimal terdapat 3 (tiga) elemen yaitu:

- Input
- Proses
- Output

Tanpa tiga elemen ini maka data tidak akan dapat diolah menjadi informasi.

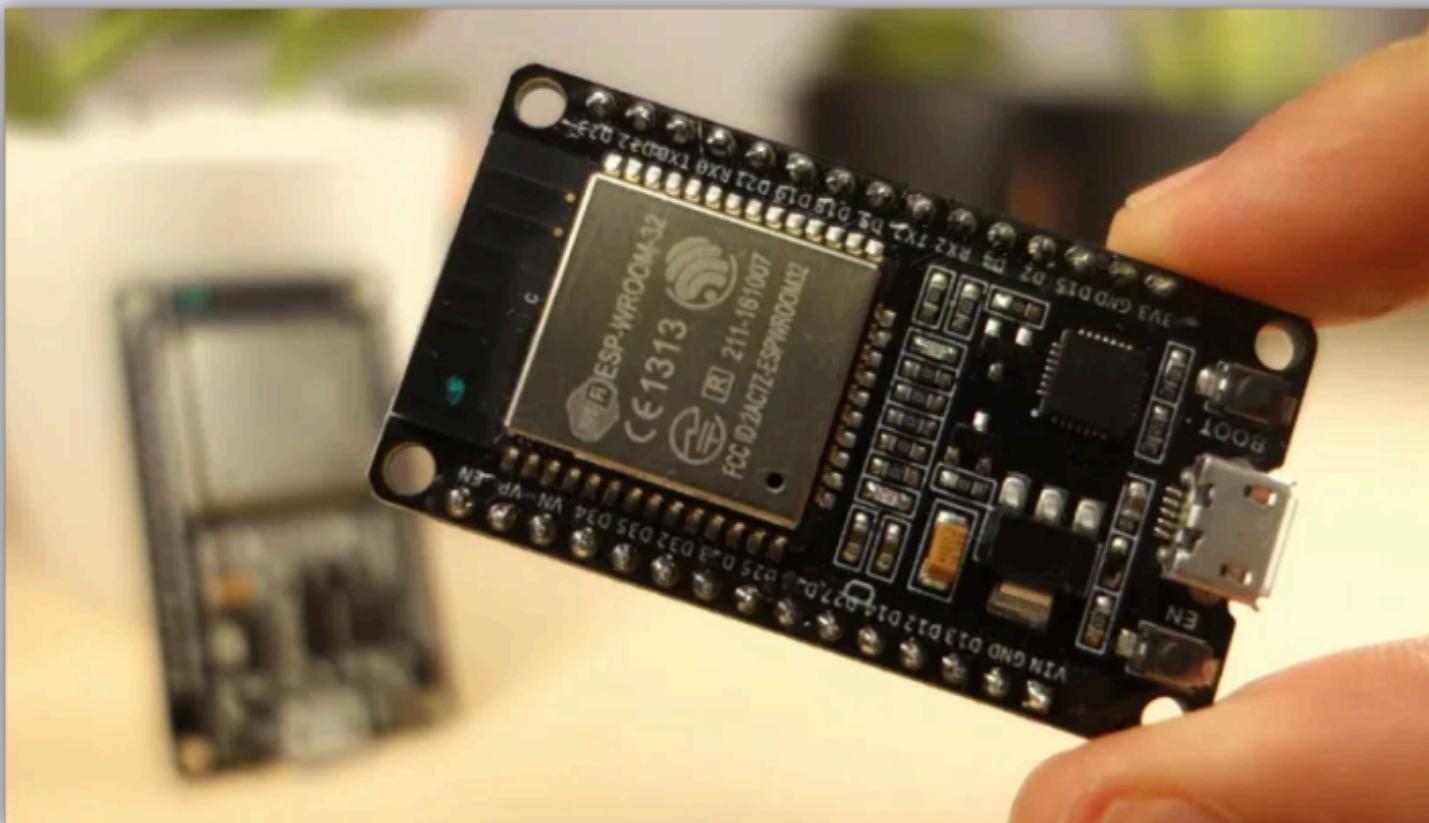
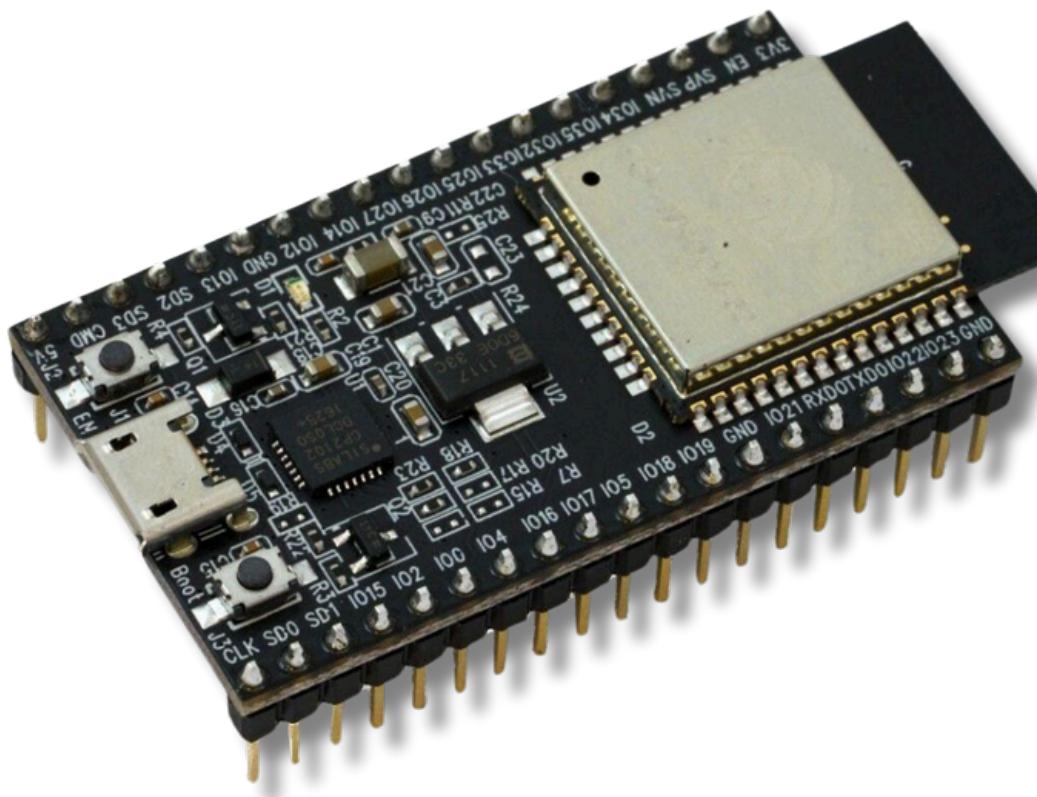
**Jika hanya ada proses dan output dimana komputer tidak akan paham apa yang harus dilakukan dan apa yang harus diproses. Jika hanya ada proses dan output maka komputer pun akan bingung apa yang harus dilakukan dan proses nya akan menghasilkan apa.**

# Mikrokontroller ESP32

Mulai belajar →

BELAJAR IOT BAGI PEMULA





## Apa Itu ESP32?

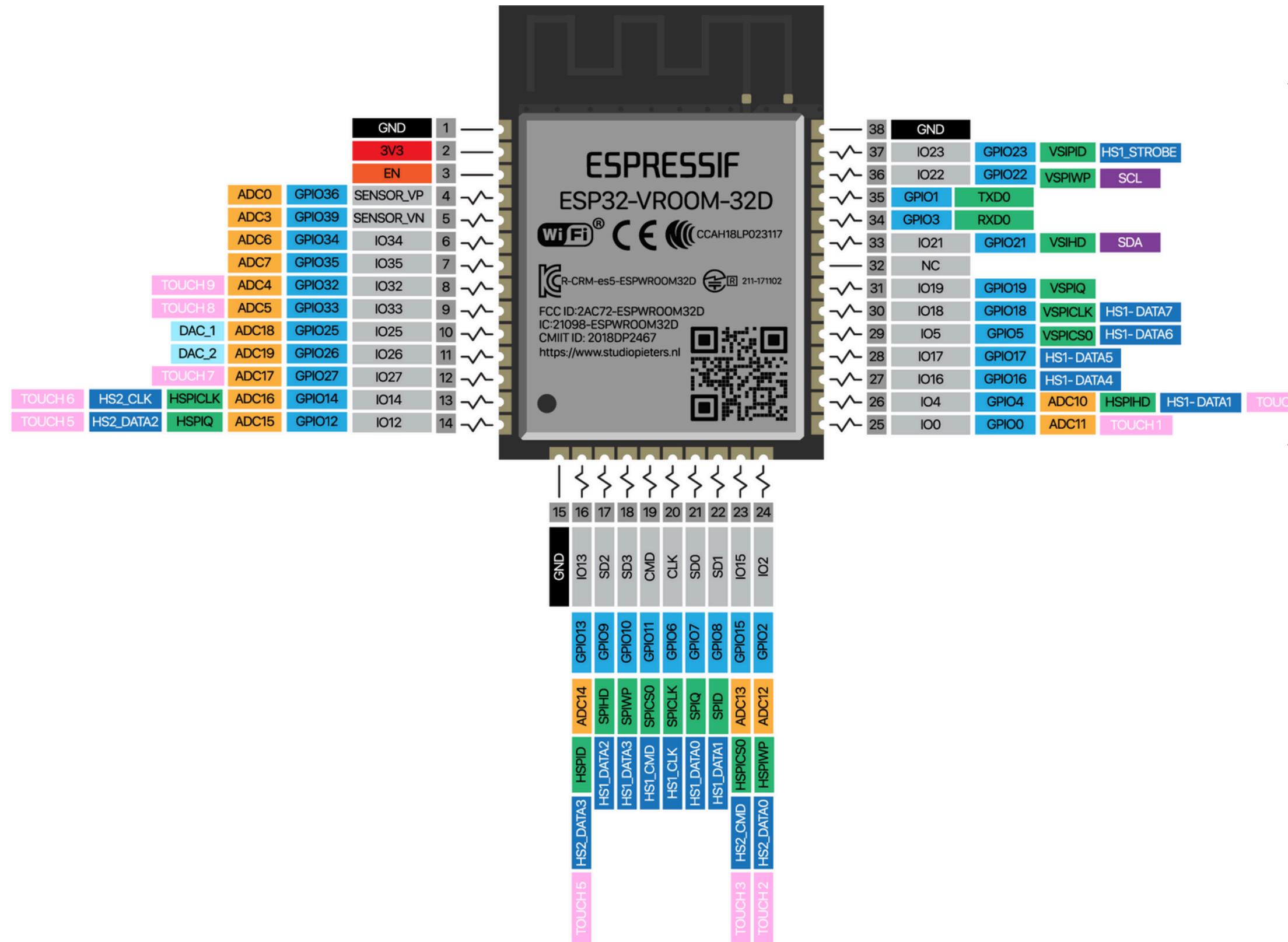
**ESP32 merupakan sebuah modul mikrokontroler yang memiliki fitur Wi-Fi dan dapat diprogram untuk berbagai aplikasi dalam proyek Internet of Things (IoT).**

Modul ini memiliki kemampuan yang sangat fleksibel dan dapat digunakan untuk mengendalikan berbagai perangkat elektronik melalui jaringan Wi-Fi.

# Spesifikasi ESP32

Spesifikasi	Detail Spesifikasi
<b>Controller</b>	<b>Dual-core Xtensa LX6 microprocessor</b> , masing-masing <b>berkecepatan hingga 240 MHz</b> . Mendukung mode sleep <b>ultra-low power</b> dan memiliki kemampuan pemrosesan yang kuat untuk aplikasi IoT.
<b>Wi-Fi (802.11 b/g/n)</b>	Mendukung protokol <b>Wi-Fi 802.11 b/g/n</b> untuk koneksi nirkabel yang stabil dan cepat. Dapat berfungsi sebagai station, access point, atau kedua-duanya (station dan access point) untuk memfasilitasi komunikasi dalam jaringan lokal atau koneksi ke internet.
<b>Bluetooth</b>	Mendukung <b>Bluetooth Classic dan Bluetooth Low Energy (BLE)</b> untuk komunikasi nirkabel dengan perangkat lain yang kompatibel dengan Bluetooth, seperti smartphone, speaker, atau perangkat IoT lainnya.
<b>Kapasitas Memori</b>	<b>Flash memory hingga 16MB</b> untuk menyimpan program aplikasi dan data. <b>SRAM hingga 520KB</b> untuk menjalankan program dan menyimpan data sementara yang dibutuhkan oleh aplikasi.
<b>GPIO (General Purpose Input/Output)</b>	Mendukung sejumlah pin <b>GPIO</b> yang dapat diatur sebagai <b>input atau output</b> . Ini memungkinkan ESP32 untuk berinteraksi dengan berbagai perangkat eksternal seperti sensor, aktuator, atau modul lainnya.
<b>ADC (Analog-to-Digital Converter)</b>	Memiliki <b>12-bit ADC</b> (Analog-to-Digital Converter) yang memungkinkan pembacaan nilai analog dari sensor atau perangkat eksternal. Ini memungkinkan ESP32 untuk mendeteksi dan memonitor perubahan lingkungan atau kondisi fisik dalam proyek IoT.

# Pin GPIO Pada ESP32



## Yang Aman Digunakan Untuk Input:

- IO2
- IO4
- IO5
- IO13 sampai IO39

## Yang Aman Digunakan Untuk Output:

- IO2
- IO4
- IO5
- IO12 sampai IO33

## I2C:

- IO22: SCL
- IO21: SDA

## SPI:

- IO6: SCK/CLK
- IO7: SDO/D0
- IO8: SDI/SD1
- IO9: SHD/SD2
- IO10: SWP/SD3
- IO11: CSC/CMD

## PWM:

Semua pin output dapat digunakan sebagai PWM kecuali 34 dan 39

# Development Board dan Production Board



## Development Board

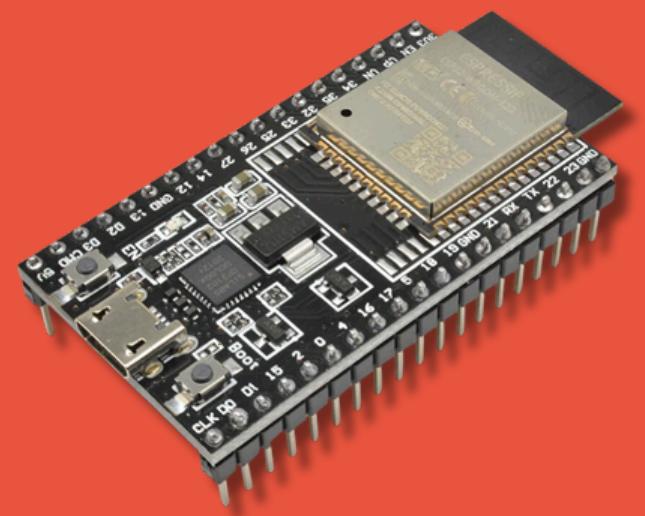
Papan rangkaian yang dirancang khusus **untuk memudahkan pengembangan dan prototyping perangkat elektronik**.

Development board digunakan dalam tahap awal pengembangan proyek **untuk eksperimen, debugging, dan pengujian**.

## Production Board

Versi final dari sebuah perangkat elektronik **yang telah melewati tahap pengembangan dan pengujian menggunakan development board**.

Production board dirancang **untuk diproduksi secara massal** dan digunakan dalam skala yang lebih luas.



ESP32 DevKitC



NodeMCU-32S



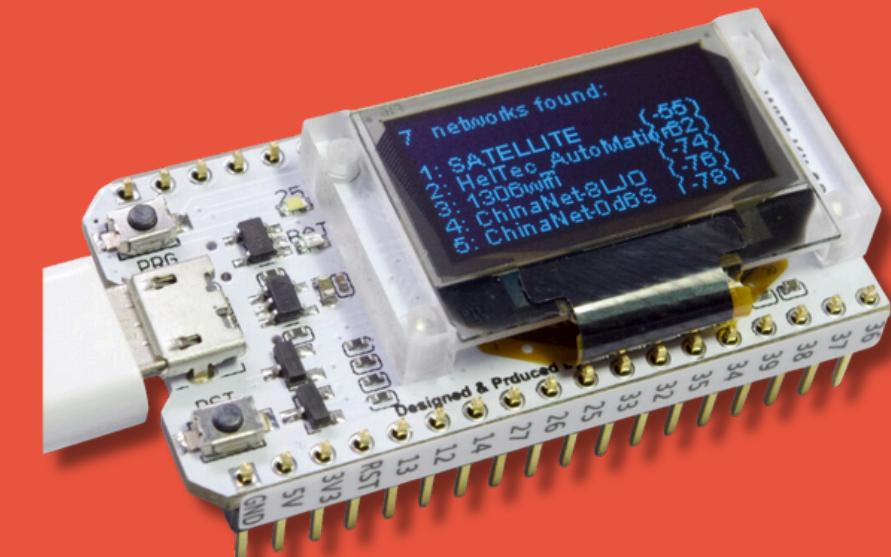
Adafruit HUZZAH32



SparkFun ESP32 Thing



Wemos D1 Mini ESP32

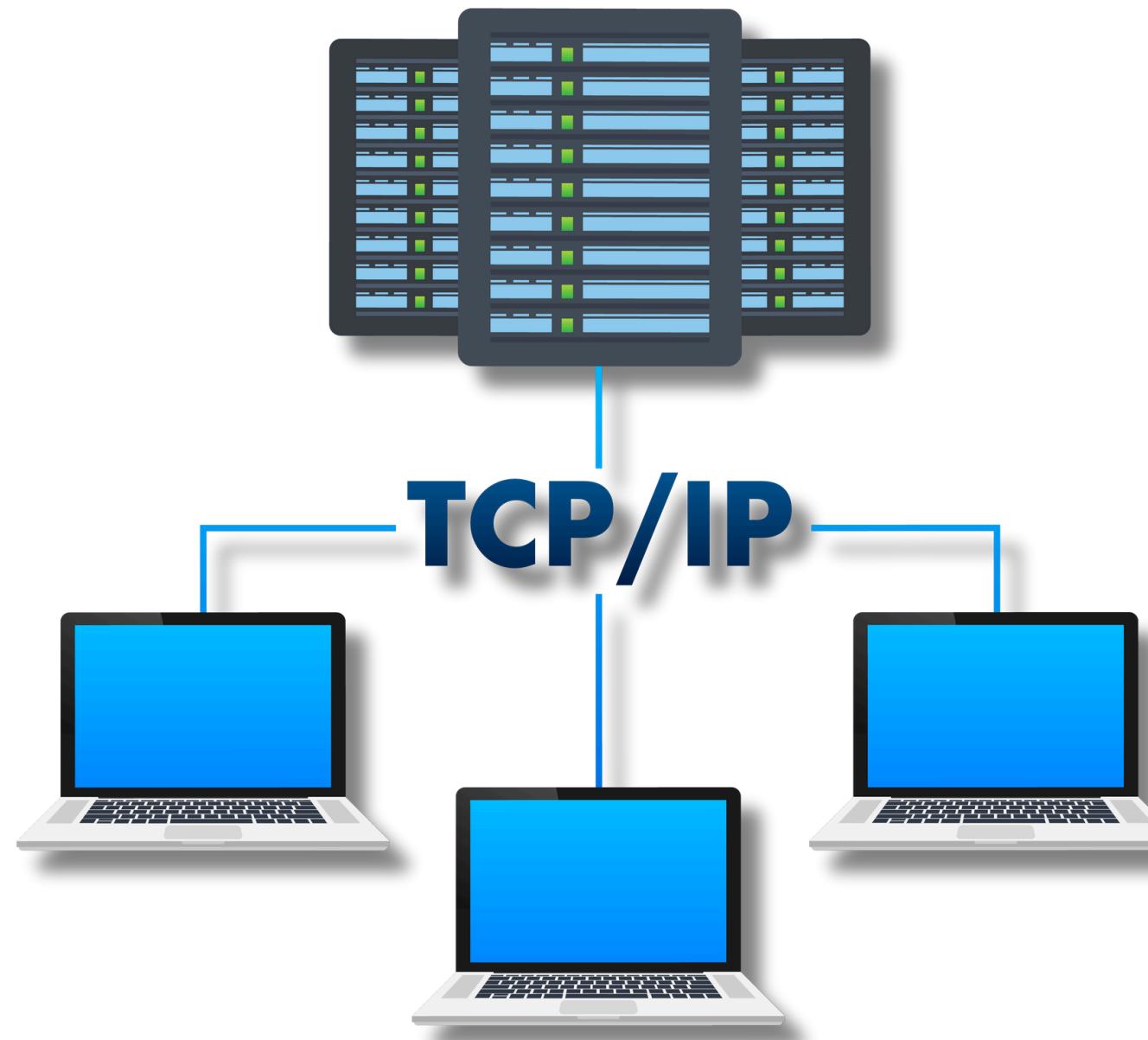


Heltec WiFi Kit 32

# Protokol Komunikasi MQTT

Mulai belajar →





# Protokol Komunikasi

Untuk memudahkan memahami mengenai protokol komunikasi maka dapat kita sebut bahwa protokol adalah aturan.

Bayangkan ketika akan melakukan upacara kemerdekaan maka ada banyak langkah-langkah dan aturan yang harus dilakukan. Aturan inilah yang kita sebut sebagai protokol upacara.

## Contoh Protokol Komunikasi

# HTTP (Hyper Text Transfer Protocol)

HTTP://



HTTP adalah salah satu protokol yang paling sering digunakan. Saat kita berselancar di internet ataupun membuka salah satu website, maka **untuk membuka website tersebut menggunakan protokol HTTP.**

## Contoh Protokol Komunikasi

# SMTP (Simple Mail Transfer Protocol)

SMTP://

SMTP adalah protokol **untuk mengirim dan menerima email**, dengan menyediakan perintah-perintah seperti "MAIL FROM" dan "RCPT TO" yang digunakan untuk menentukan pengirim dan penerima email. SMTP juga menyediakan mekanisme untuk menentukan subjek email dan menyertakan file-file lampiran dalam email.

## Contoh Protokol Komunikasi

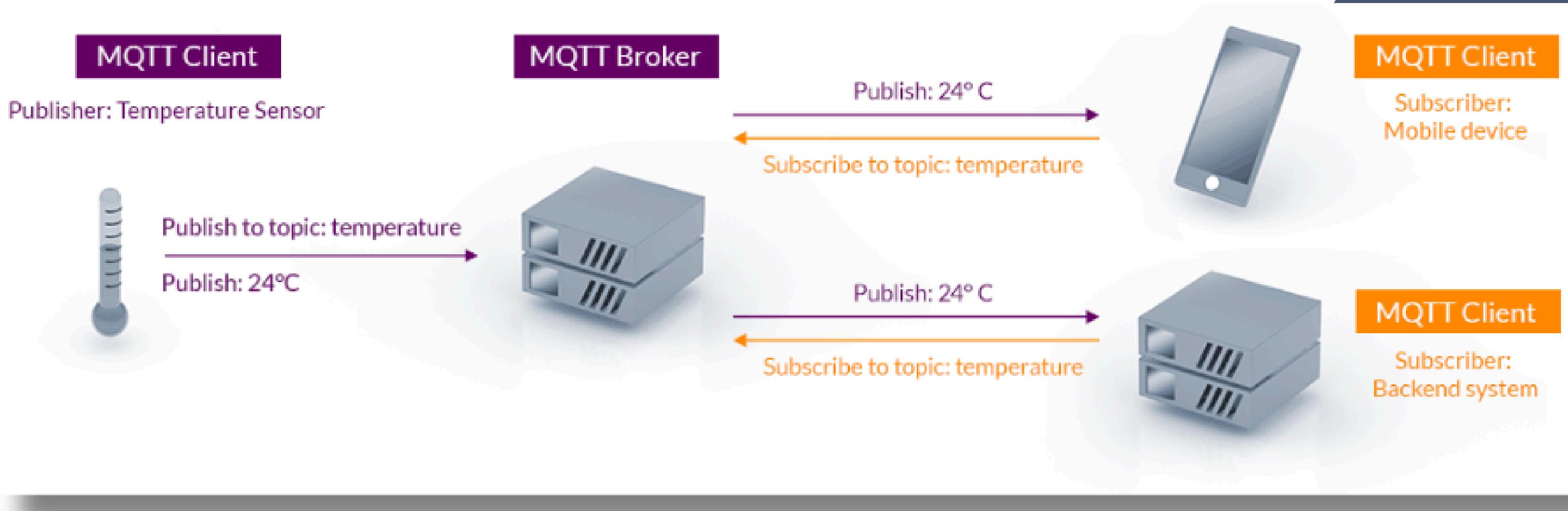
### RTSP (Real-Time Streaming Protocol)

RTSP://

Protokol jaringan yang digunakan **menggunakan** live stream dalam jaringan komputer.

RTSP merupakan salah satu protokol untuk melakukan live stream. Pada ranah IoT misalnya untuk camera drone dan CCTV.

# MQTT



MQTT (Message Queue Telemetry Transport) adalah **protokol komunikasi yang digunakan untuk mentransfer data mesin ke mesin (M2M)**. MQTT digunakan untuk mengirim dan menerima data dari perangkat IoT yang terhubung ke jaringan internet.

# Alasan Menggunakan MQTT Untuk IoT

## Dibanding Protokol Lain



### Sumberdaya yang dibutuhkan sangat sedikit

Dengan konsumsi sumberdaya yang sedikit maka bisa **menghemat bandwidth dan tentu hemat energi.**

### Komunikasi bi-directional

Komunikasi bidirectional artinya **komunikasi bisa dilakukan dua arah**. Satu perangkat dapat menjadi pengirim data sekaligus dapat menjadi penerima data.

### Jutaan perangkat dapat terhubung sekaligus

MQTT sangat ringan sehingga dengan spesifikasi server yang kecil sekalipun **dapat menangani jutaan perangkat secara bersamaan.**

### Dapat diandalkan pada jaringan yang buruk

Banyak perangkat IoT terhubung melalui jaringan seluler yang tidak dapat diandalkan. **MQTT memiliki fitur retain message sehingga mengurangi waktu untuk menghubungkan kembali klien dengan server.**

# Cara Kerja

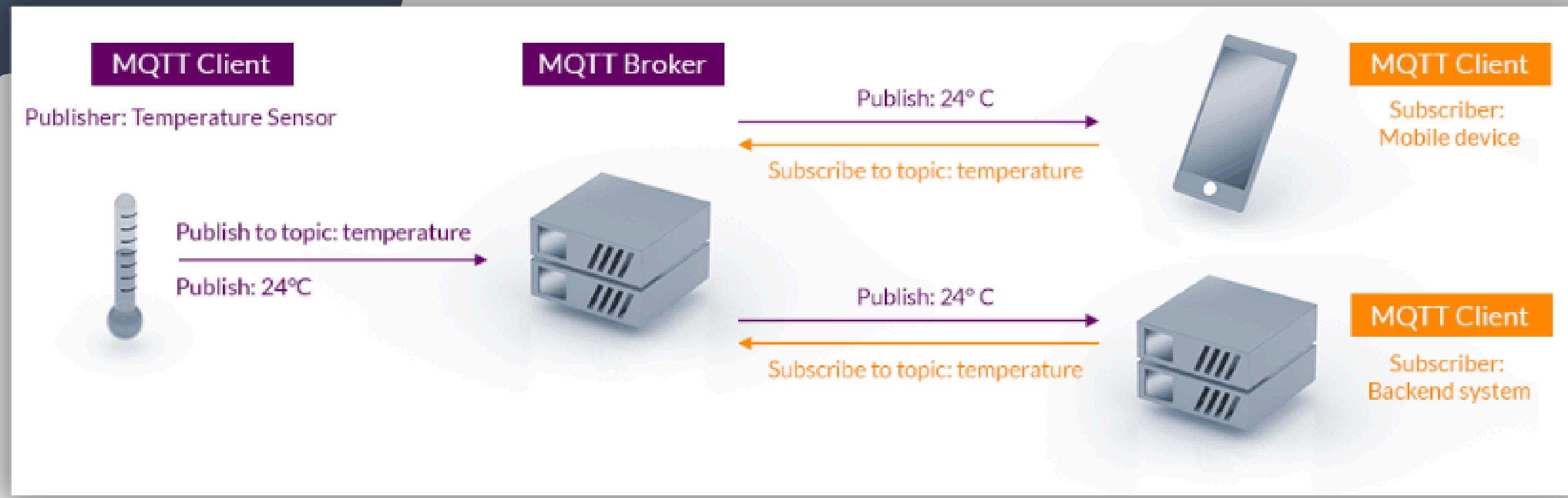
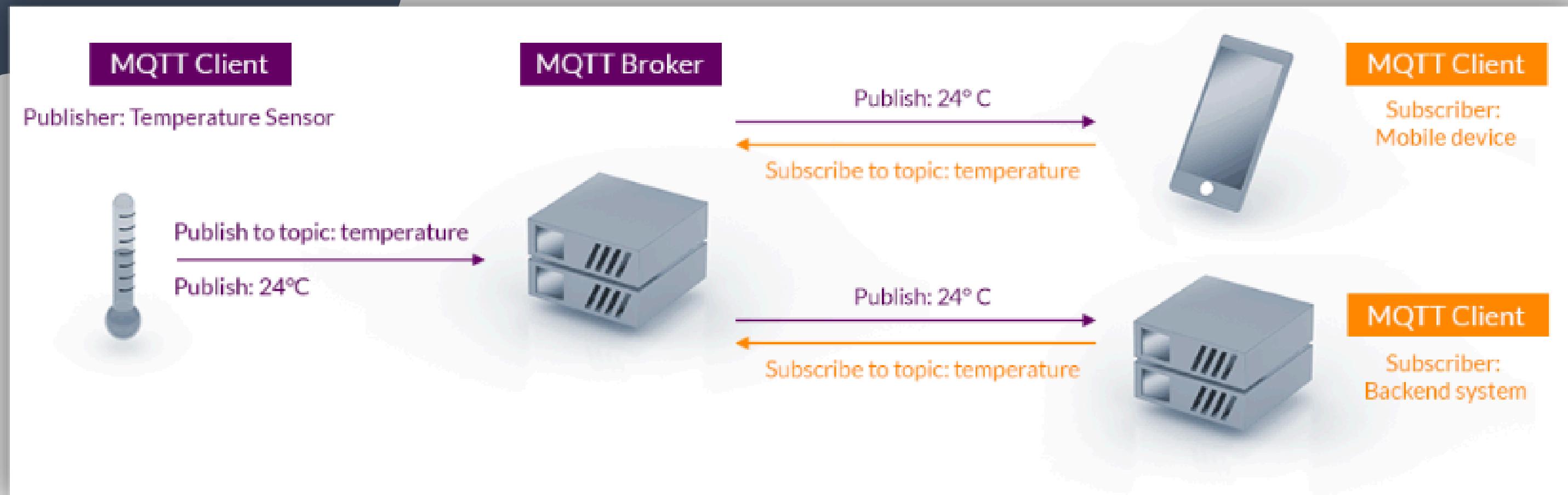


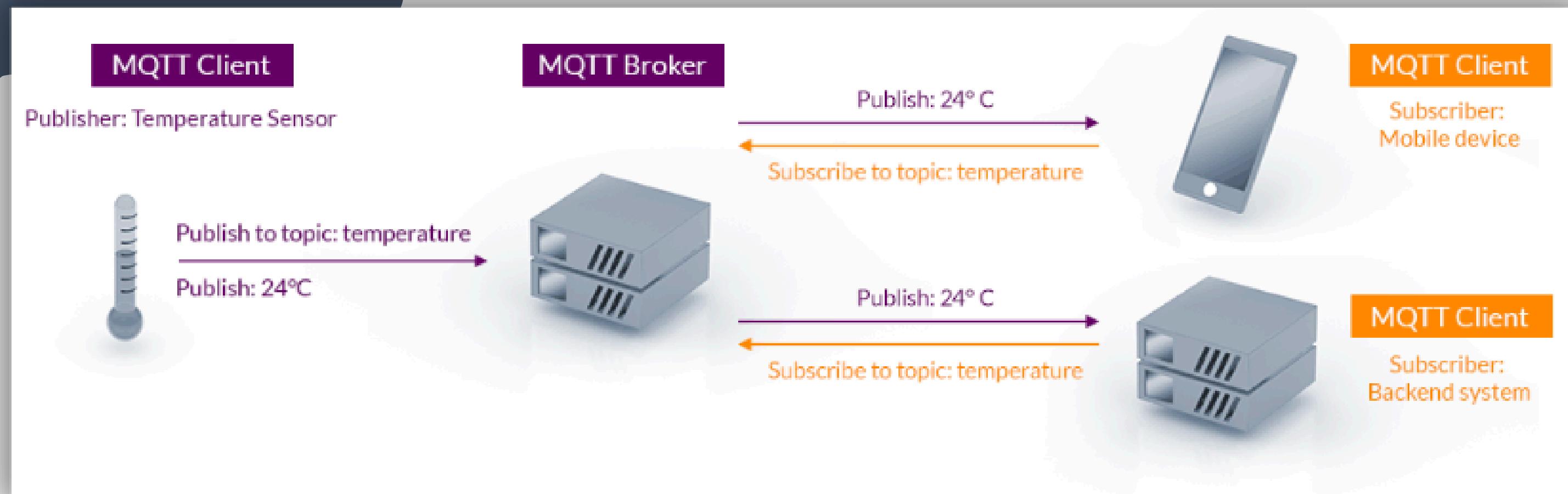
Diagram diatas akan kita gunakan untuk mempelajari bagaimana MQTT bekerja.

# Publish dan Subscribe



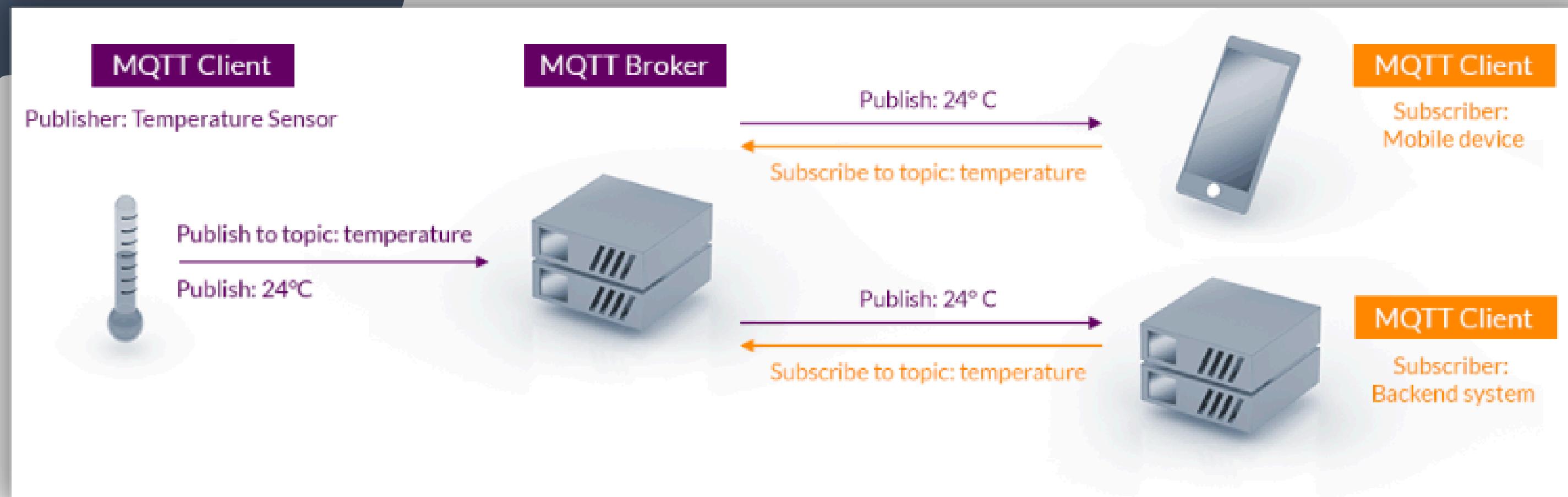
Perangkat atau client yang mengirim data disebut dengan **publisher** dan perangkat atau client yang menerima data disebut dengan **subscriber**.

# Broker



Client tidak bisa berkomunikasi dengan client lain secara langsung, oleh karena itu membutuhkan jembatan. **Server yang menjadi jembatan komunikasi ini disebut dengan Broker.**

# Topic



**Topic adalah penyaringan data yang dikirim.** Setiap kali client melakukan publish data maka data tersebut harus menyertakan topic. Setelah dipublish, selanjutnya broker akan mengirim data tersebut ke semua client yang melakukan subscribe pada topic yang sama.



# Level Pada Topic

gedung1/lantai4/ruang\_tunggu/suhu

Level 1

Level 2

Level 3

Level 4

Topic pada MQTT ditulis dengan huruf atau angka dan dipisahkan dengan simbol garis miring.

Garis miring digunakan untuk penanda pemisah level. Satu garis miring maka satu level.



# Wildcard

## Melakukan subscribe untuk banyak sub-level topic

### Single Level Wildcard

Subscribe seluruh topic yang ada pada  
pada satu level

Menggunakan Simbol +

gedung1/+ruang\_tunggu/+

gedung1/lantai1/ruang\_tunggu/suhu  
gedung1/lantai2/ruang\_tunggu/lembab  
gedung1/lantai3/ruang\_tunggu/lampu  
gedung1/lantai4/ruang\_tunggu/api

### Multi Level Wildcard

Subscribe topic pada seluruh level  
dibawahnya

Menggunakan Simbol #

gedung1/lantai4/#

gedung1/lantai4/ruang\_tunggu/suhu  
gedung1/lantai4/toilet/lembab  
gedung1/lantai4/toilet/suhu  
gedung1/lantai4/ruang\_arsip/api

# QoS (Quality of Service)

Jaminan data bisa terkirim dan tidak terjadi duplikasi.

At most once

**QoS 0**

Data dikirim tanpa ada jaminan data diterima.

- Data dapat hilang atau tiba dalam urutan yang acak.
- Pengirim tidak peduli data berhasil terkirim atau tidak.

At least once

**QoS 1**

Jika broker tidak menerima konfirmasi dari penerima, maka data akan dikirim ulang.

- Memastikan bahwa setiap data setidaknya diproses satu kali.
- Dapat terjadi duplikasi pengiriman data

Exactly Once

**QoS 2**

Setiap data dikirim dengan jaminan diterima paling sedikit sekali dan tidak lebih dari sekali.

- Proses pengiriman dan konfirmasi dilakukan dengan menggunakan tiga tahap.
- Tidak terjadi duplikasi data.

# Client ID (Identifier)

**Identifikasi unik yang diberikan kepada perangkat yang terhubung ke broker MQTT.**

- Client ID digunakan oleh broker untuk mengidentifikasi klien yang terhubung dan untuk menjaga keadaan sesi (session state) klien.
- Dengan Client ID Ini memungkinkan broker untuk mengidentifikasi klien yang terhubung kembali dan memulihkan keadaan sesi sebelumnya jika diperlukan.
- Digunakan oleh broker agar mengetahui data diterima dari perangkat mana dan harus dikirim ke perangkat mana.

# Retain Message

Retain message adalah fitur pada protokol MQTT yang memungkinkan pesan yang dipublikasikan ke sebuah topik **untuk disimpan (retained)** oleh broker.

Sehingga **data terakhir yang dipublish tetap akan disimpan dan akan dikirim ke subscriber baru walaupun publisher sudah offline.**

