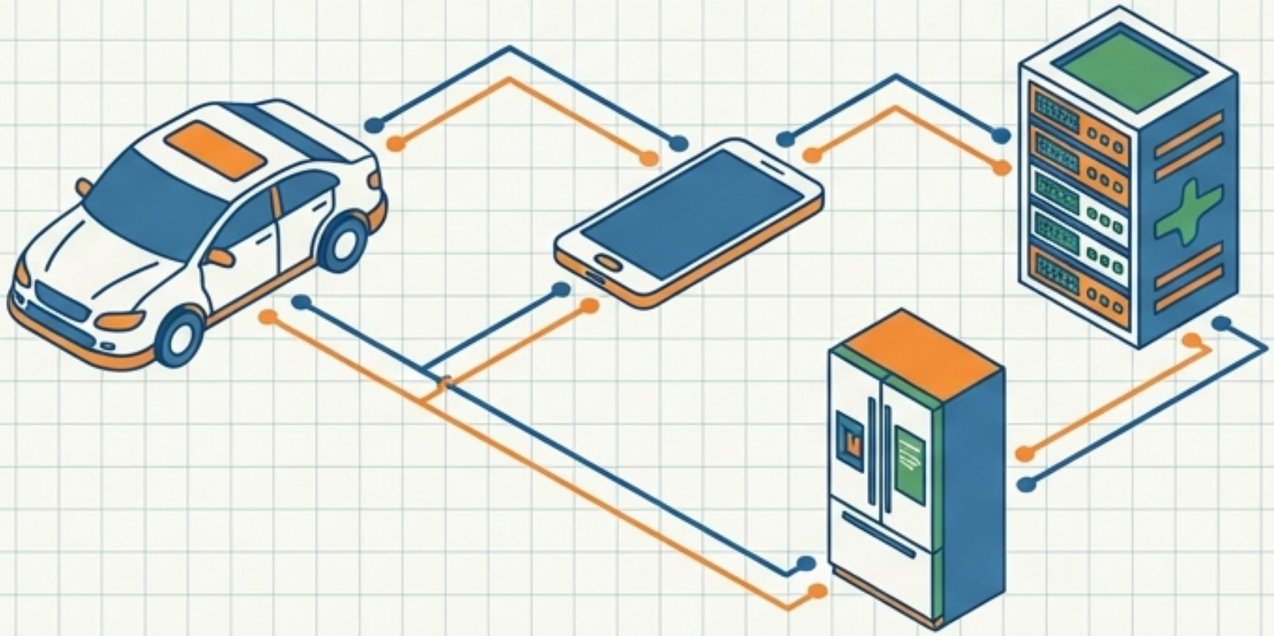
IoT Masterclass: ปลดล็อกโลกของอินเทอร์เน็ตในทุกสิ่ง

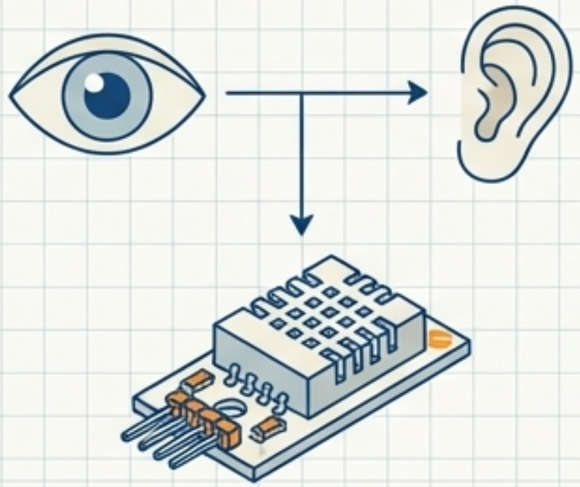
พิมพ์เขียวนักสร้างสรรค์: จากความเข้าใจพื้นฐาน
สู่การลงมือสร้างสถาปัตยกรรม IoT แบบครบวงจร



NotebookLM

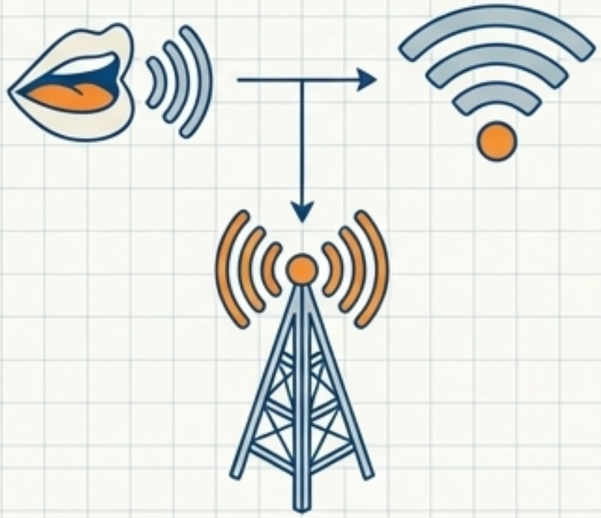
กายวิภาคของสรรพสิ่ง (What is IoT?)

อ้างอิงจาก IEEE (2015) IoT คือเครือข่ายที่เชื่อมต่อสรรพสิ่งเข้าด้วยกันผ่านอินเทอร์เน็ต
โดยมีองค์ประกอบการทำงานเปรียบเสมือนร่างกายมนุษย์:



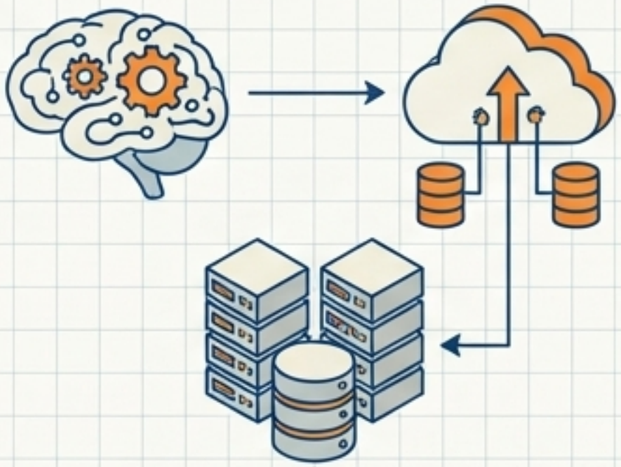
รับรู้บริบท (Sensing)

เซ็นเซอร์ทำหน้าที่เหมือนตาและหู
เพื่อเก็บข้อมูลสภาพแวดล้อม
(เช่น อุณหภูมิ, ความชื้น)



สื่อสาร (Connectivity)

ระบบสมองกลฝังตัวและโปรโตคอล
ไร้สายทำหน้าที่ส่งต่อข้อมูลที่รับรู้ได้

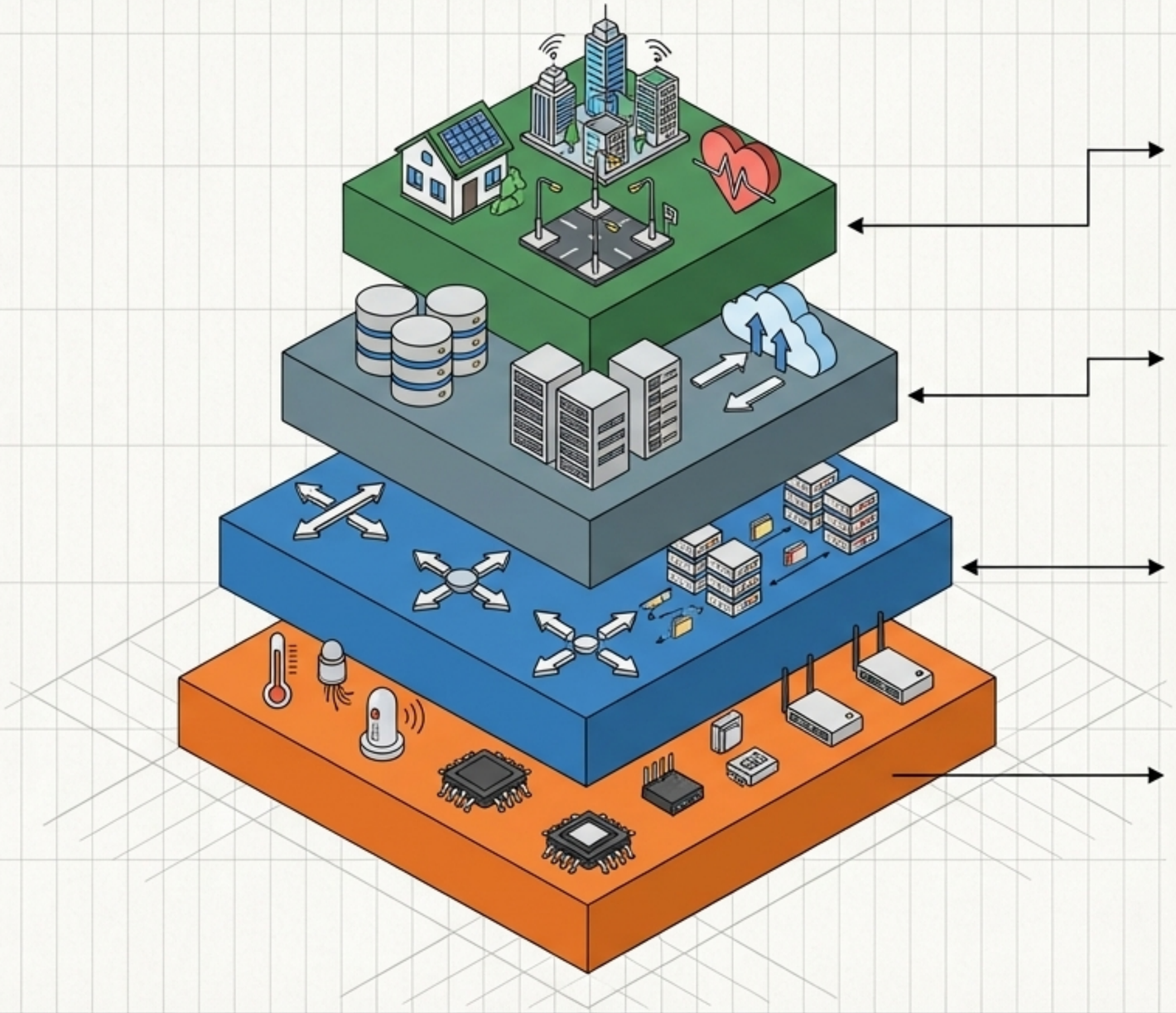


คิดวิเคราะห์ (Embedded Intelligence)

คลาวด์หรือเซิร์ฟเวอร์นำข้อมูล
มาประมวลผลและสั่งการ
(Actuation) แบบอัตโนมัติ

NotebookLM

โครงสร้าง 4 ชั้นของ IoT (IoT Layer Architecture)



Application Layer (ชั้นประยุกต์ใช้งาน)
ปลายทางที่ผู้ใช้สัมผัส เช่น Smart Home, Smart City, Smart Health

Support Layer (ชั้นสนับสนุน)
ตัวกลางที่ช่วยซ่อนความซับซ้อน (Middleware) จัดการฐานข้อมูล บริการคลาวด์ และแปลภาษาข้อมูล

Network Layer (ชั้นเครือข่าย)
ถนนส่งข้อมูล ทำหน้าที่จัดเส้นทาง (Routing) และส่งข้อมูลไปยังจุดประมวลผล

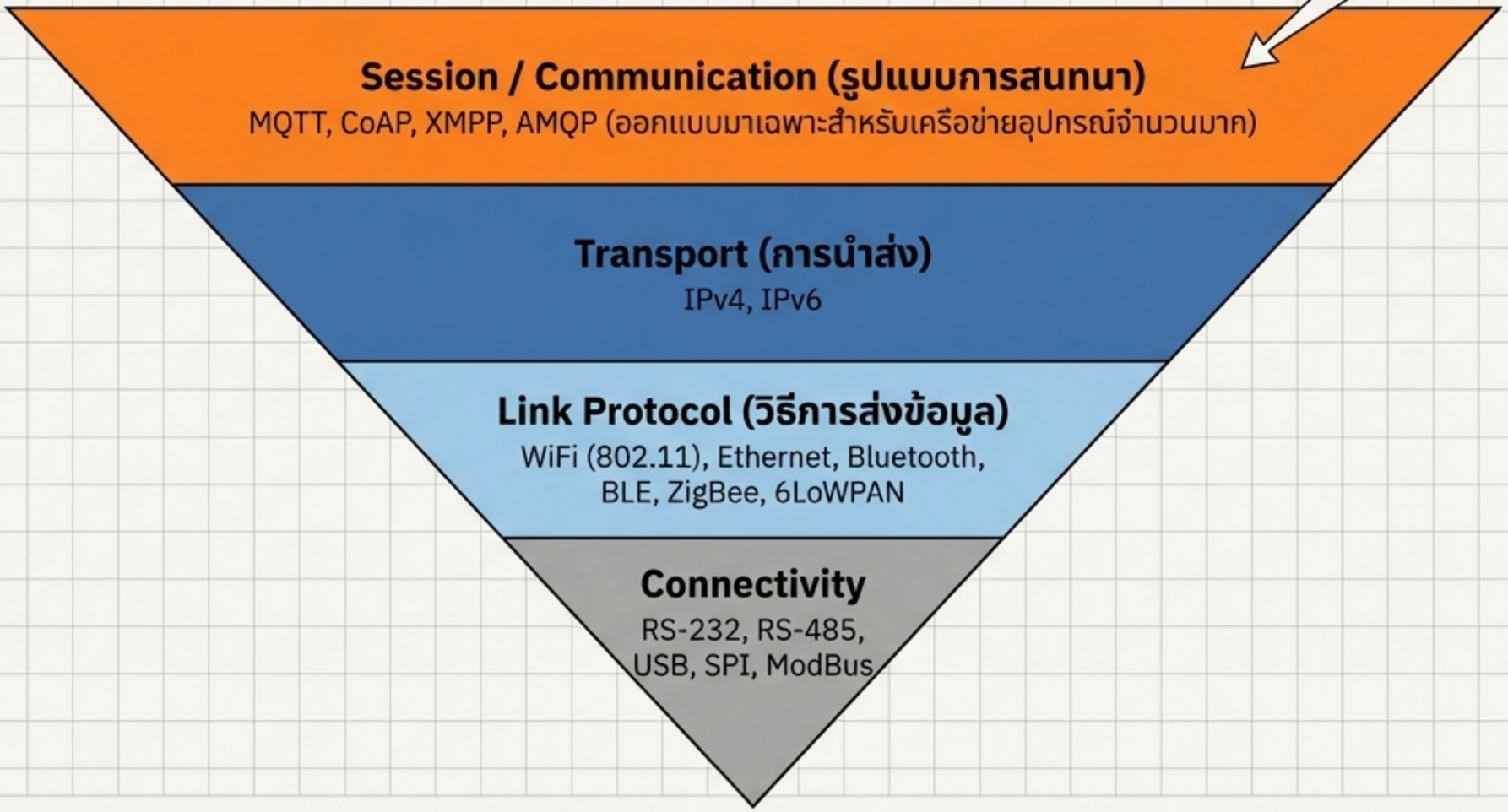
Device Layer (ชั้นอุปกรณ์)
จุดกำเนิดข้อมูล ประกอบด้วยเซ็นเซอร์และอุปกรณ์ Gateway ที่ใช้อ่านค่าและควบคุม

NotebookLM

ภาษาที่สรรพสิ่งใช้คุยกัน (Protocol Stack)

การส่งข้อมูลจากเซ็นเซอร์ไปสร้างมูลค่าทางธุรกิจ
ต้องผ่านโปรโตคอลหลายระดับ

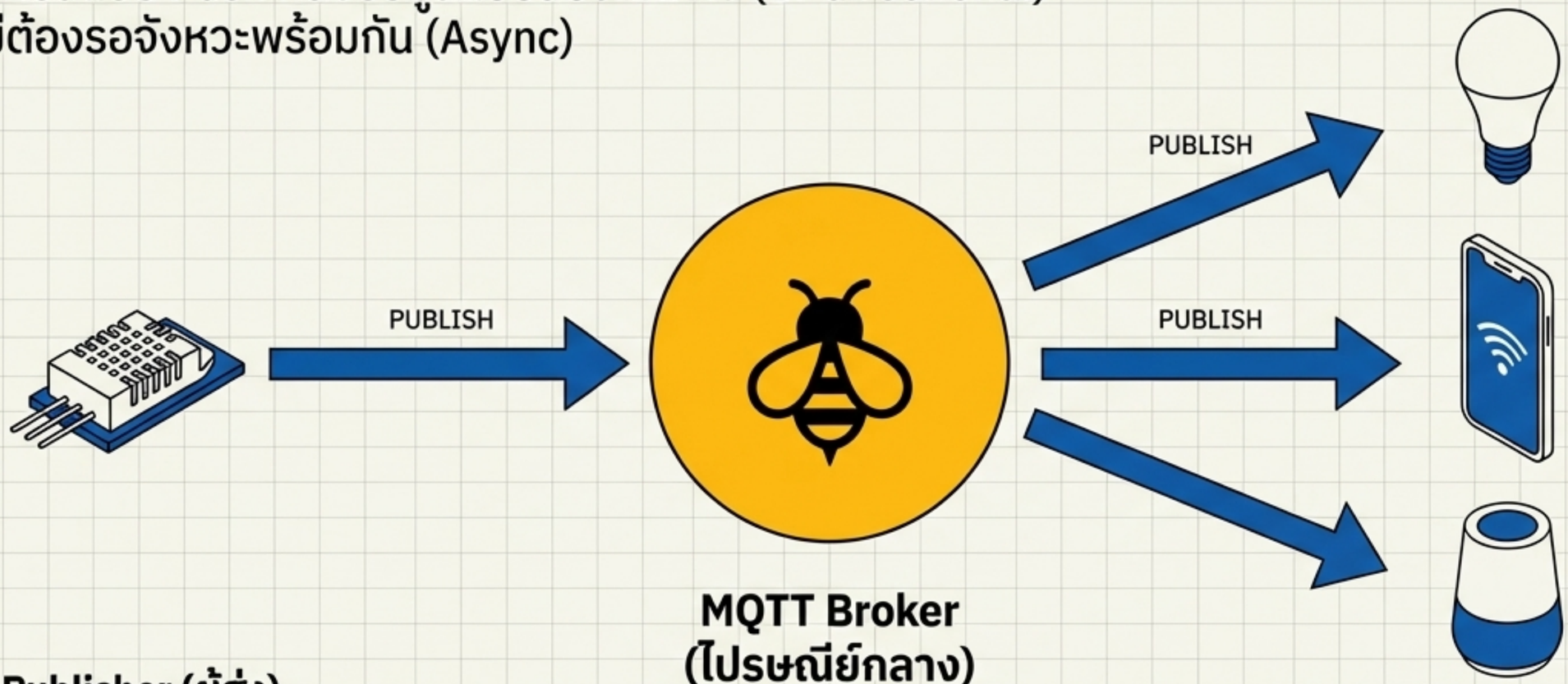
MQTT คือดาวเด่นที่สุดในยุคนี้
เนื่องจากเบาและส่งข้อมูลได้ไว



NotebookLM

MQTT: ไปรษณีย์กลางแห่งโลก IoT

โปรโตคอลแบบ Push ที่ส่งข้อมูลแบบสองทิศทาง (Bi-directional)
และไม่ต้องรอจังหวะพร้อมกัน (Async)



Publisher (ผู้ส่ง)
อุปกรณ์ (เช่น เซ็นเซอร์อุณหภูมิ)
ทำหน้าที่ Publish ข้อมูลขึ้นไป

MQTT Broker
(ไปรษณีย์กลาง)
หัวใจสำคัญ ทำหน้าที่รับข้อมูลและ
คัดกรองว่าใครควรได้รับข้อความไหน

Subscriber (ผู้รับ)
อุปกรณ์ที่ลงทะเบียน Subscribe ไว้ จะได้รับ
รับข้อมูลทันทีแบบ Push โดยไม่ต้องคอยถาม

NotebookLM

ชำแหละโครงสร้างข้อความ MQTT

topic level separator

myhome / groundfloor / livingroom / temperature

topic level topic level

Topic (ที่อยู่จัดส่ง)

ใช้คัดกรองข้อความ
คั่นแต่ละระดับชั้นด้วยเครื่องหมาย /

Wildcards (ตัวแทนการค้นหา)

- + (Single-level): แทนที่ได้แค่ 1 ระดับชั้น (เช่น scores/football/+/Texas)
- # (Multi-level): แทนที่ระดับชั้นย่อยทั้งหมดที่ตามมา (เช่น scores/#)

MQTT Packet

Payload (เนื้อหา):
ข้อมูลจริงที่ต้องการส่ง
(เช่น '32.5 องศา')

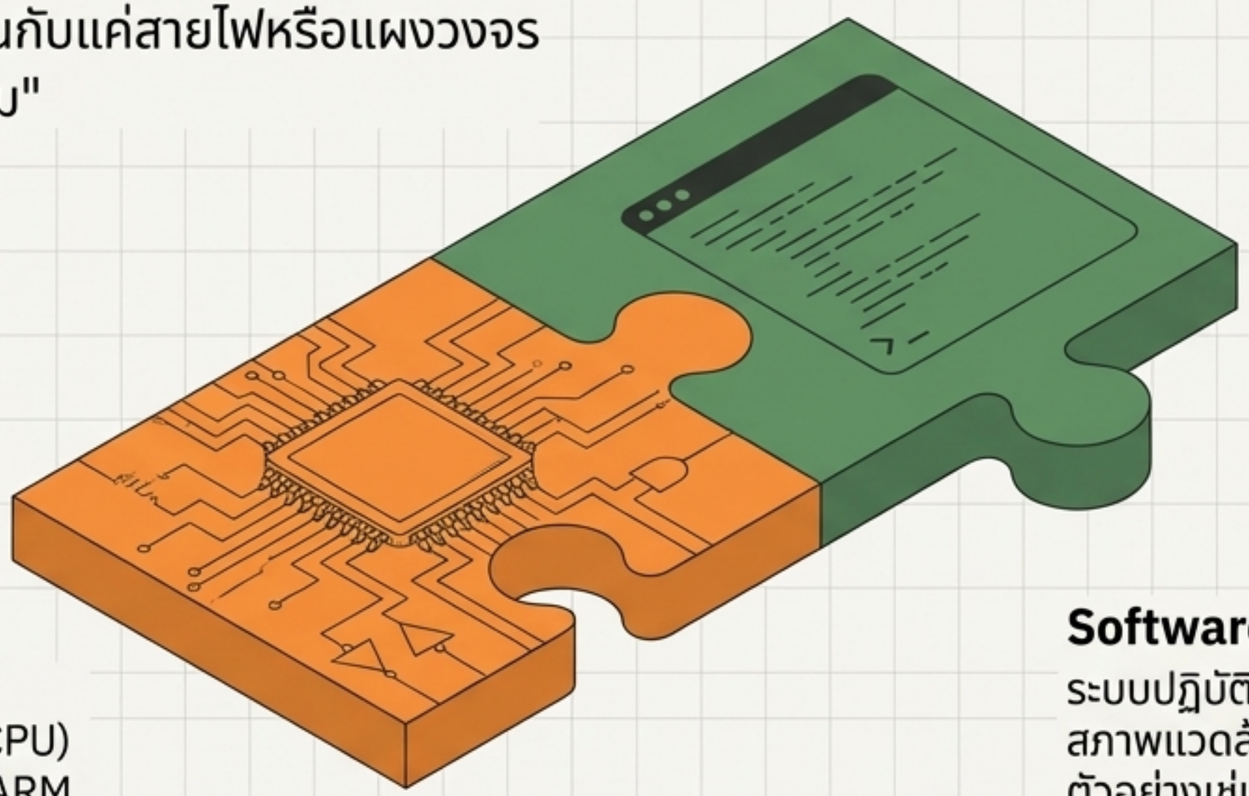
QoS (Quality of Service):
ระดับ 0, 1 หรือ 2 เพื่อการันตีว่าข้อความจะไปถึงปลายทาง

Retain-Flag (ความทรงจำ):
หากตั้งค่าเป็น True, Broker จะจำค่า Payload ล่าสุดไว้ เมื่อมีคนใหม่เข้ามา Subscribe ก็จะได้รับค่านี้ทันทีโดยไม่ต้องรออุปกรณ์ส่งใหม่

NotebookLM

Platform (แพลตฟอร์ม) คืออะไร?

ในโลกของ IoT เราไม่ได้ทำงานกับแค่สายไฟหรือแผงวงจร
แต่เราทำงานบน "แพลตฟอร์ม"



Hardware / ฮาร์ดแวร์

สถาปัตยกรรมคอมพิวเตอร์ หรือ
สถาปัตยกรรมหน่วยประมวลผล (CPU)
ตัวอย่างเช่น สถาปัตยกรรม x86, ARM,
หรือ ไมโครคอนโทรลเลอร์ตระกูล AVR

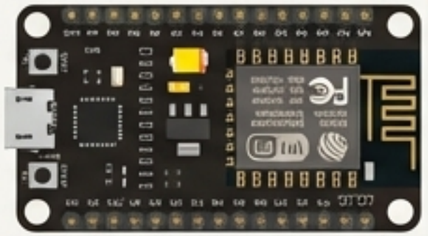
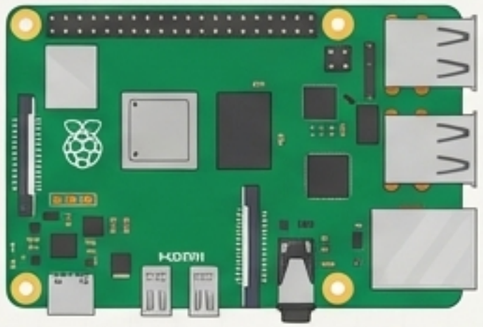
Software / ซอฟต์แวร์

ระบบปฏิบัติการ (OS) หรือ
สภาพแวดล้อมที่ใช้เขียนโปรแกรม (IDE)
ตัวอย่างเช่น Linux, Windows, หรือ
Java Platform

การทำงานร่วมกันของทั้งสองส่วนนี้คือสิ่งที่ทำให้ Application ของ IoT เกิดขึ้นได้จริง

NotebookLM

The Hardware Matrix: เลือกสมองกลให้ถูกงาน

			
	Arduino	**NodeMCU**	**Raspberry Pi**
รูปแบบ	Microcontroller Board	WiFi SoC Development Kit	Single Board Computer (SBC)
ชิปประมวลผล	ATmega328 (AVR)	ESP8266	Broadcom (ARM)
การเชื่อมต่อ	ไม่มีในตัว (ต้องต่อ Shield)	มี WiFi ภายในตัว	Ethernet LAN & WiFi
ระบบปฏิบัติการ	ไม่มี (รันโปรแกรมวนลูป)	ไม่มี (รัน Firmware/Script)	มี Linux OS (Raspbian) เต็มรูปแบบ
จุดเด่นหลัก	**ควบคุม I/O พื้นฐานได้เสถียรมาก**	**ราคาถูก เชื่อมต่อเน็ตได้ง่ายดาย**	**พลังประมวลผลสูง ทำหน้าที่เป็น MQTT Broker ได้**

NotebookLM

เจาะลึก Arduino: ผู้บุกเบิกโลก Open-Source

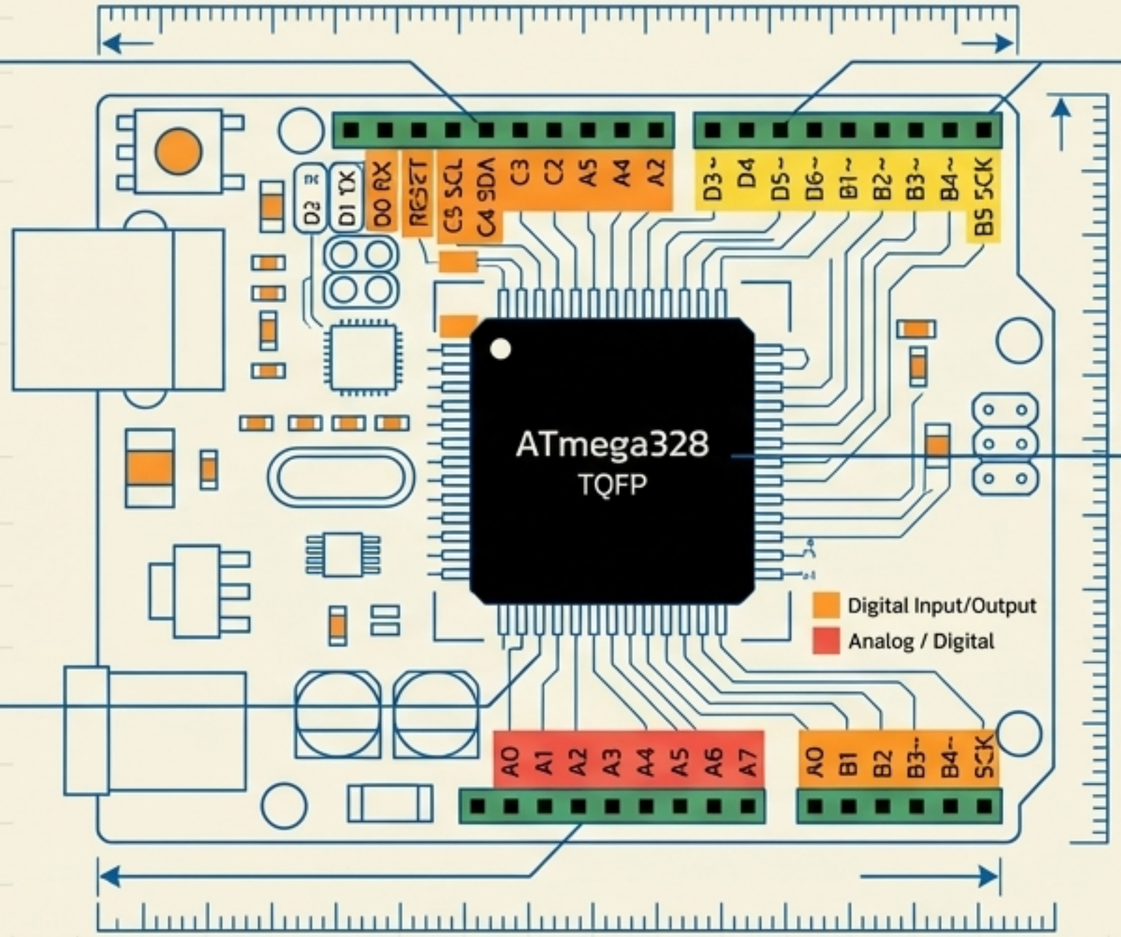
แพลตฟอร์มยอดฮิตที่ประกอบด้วยบอร์ด (AVR) และ IDE ที่เขียนด้วยภาษา Java (รองรับทั้ง Windows, Mac, Linux)

Digital Input/Output

ขารับส่งสัญญาณดิจิทัล (0 หรือ 1) สำหรับสั่งเปิด/ปิดอุปกรณ์

PWM (Pulse Width Modulation)

ขาที่สามารถจำลองสัญญาณอนาล็อกได้ เช่น ใช้หรี่ไฟ LED หรือควบคุมความเร็วมอเตอร์



ATmega328

หัวใจหลักในการประมวลผลคำสั่ง

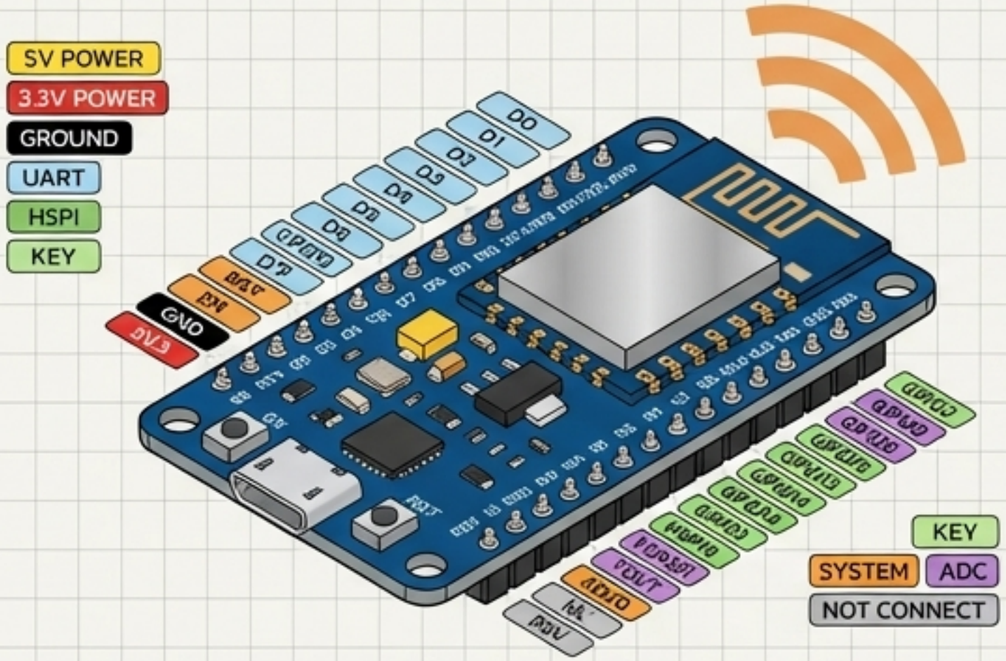
Analog Input

สำหรับอ่านค่าจากเซ็นเซอร์ที่มีความละเอียดสูงแบบต่อเนื่อง

NotebookLM

NodeMCU: พลังแห่งการเชื่อมต่อไร้สาย

Hardware Specs



- ใช้โมดูลชิป ESP8266 ที่มี WiFi ในตัว
- มี USB-TTL (IC CP2102) มาบนบอร์ด เสียบสาย micro-USB โหลดโปรแกรมได้ทันที
- มีขา A0 สำหรับรับอินพุตแรงดัน Analog (ADC ขนาด 10 บิต)

Code Logic

การเชื่อมต่อ WiFi ทำได้ง่ายๆ ผ่าน ESP8266WiFi.h

```
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print('.');
}
```

สถานะจะเปลี่ยนจาก WL_IDLE_STATUS เป็น WL_CONNECTED เมื่อสำเร็จ

NotebookLM

Raspberry Pi: คอมพิวเตอร์ฉบับกระเป๋า

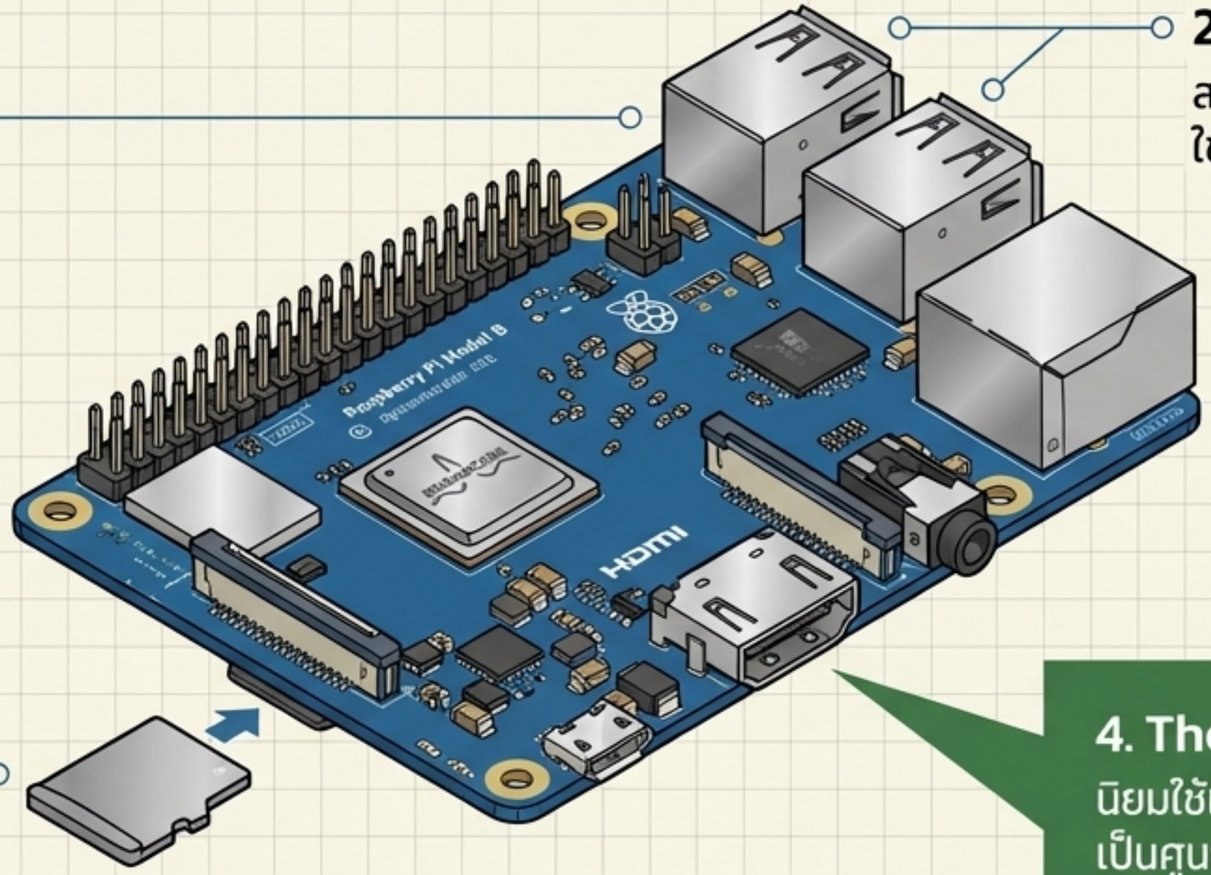
แตกต่างจากไมโครคอนโทรลเลอร์ธรรมดา Raspberry Pi คือ
คอมพิวเตอร์ขนาดเท่าบัตรเครดิตที่รันระบบปฏิบัติการเต็มรูปแบบ

2. USB & HDMI

สามารถต่อเมาส์ คีย์บอร์ด และจอภาพ
ใช้งานได้เหมือนคอมพิวเตอร์ตั้งโต๊ะ

3. Ethernet & WiFi

รองรับการต่อสาย LAN
และเครือข่ายไร้สาย (รุ่นใหม่)



1. MicroSD Card

ทำหน้าที่เป็นฮาร์ดดิสก์สำหรับ
ติดตั้ง OS (เช่น Raspbian/Linux)

4. The Heavyweight

นิยมใช้เป็น Local MQTT Broker เพื่อ
เป็นศูนย์กลางคัดกรองข้อมูลของอุปกร
ณ์ทั้งหมดในบ้านก่อนส่งขึ้น Cloud

NotebookLM

Headless Setup: การตั้งค่าแบบไร้จอภาพ

หากไม่มีจอ HDMI เราสามารถตั้งค่าผ่าน Network ได้ด้วย 4 ขั้นตอน:

Write

ใช้โปรแกรม Win32 Disk Imager นำไฟล์ Image ของ OS เขียนลงใน SD Card

Config IP

ถอด SD Card มาเสียบใน PC แก้ไขไฟล์ cmdline.txt โดยเติม ip=192.168.10.XX ต่อท้าย (ห้ามขึ้นบรรทัดใหม่)

Boot

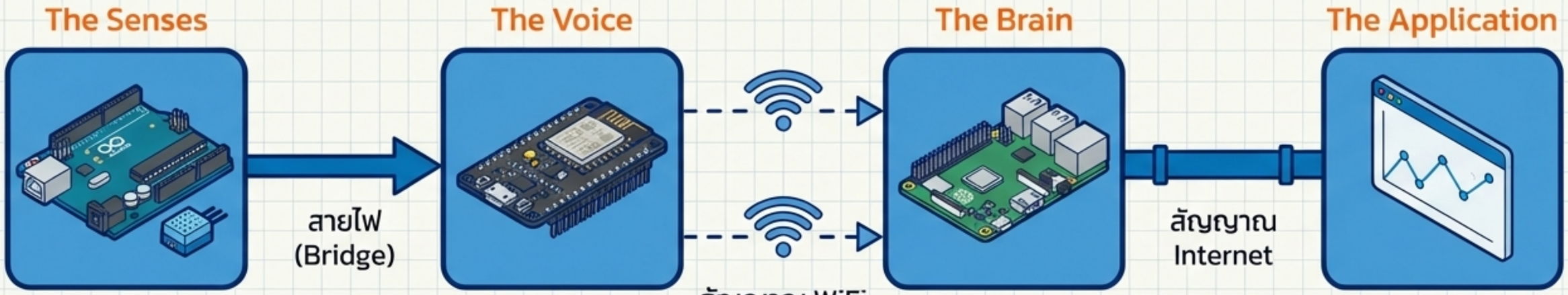
นำ SD Card ใส่กลับเข้า Pi เสียบสาย LAN เข้ากับ PC และเสียบ Adapter จ่ายไฟ

Verify

เปิด CMD ในคอมพิวเตอร์ ใช้คำสั่ง ping 192.168.10.XX เพื่อทดสอบว่าเชื่อมต่อกับ Raspberry Pi สำเร็จหรือไม่

NotebookLM

บทสรุปสถาปัตยกรรม: นำสรรพสิ่งมาเชื่อมต่อกัน



Arduino
(Sensor/Actuator)

อ่านค่า Analog
จากเซ็นเซอร์ /
และควบคุมไฟ LED

NodeMCU
(WiFi Gateway)

อ่านค่าจาก Arduino
และใช้ PubSubClient
เพื่อ Publish
ข้อมูลขึ้นเครือข่าย WiFi

Raspberry Pi
(Local MQTT Broker)

ทำหน้าที่เป็นศูนย์กลางรับส่งข้อมูล
(และมีปุ่มกดเพื่อ Publish คำสั่ง
กลับไปเปิด LED ที่ Arduino)

Web Browser

นำค่า Analog ที่
Subscribe ได้
ไปแสดงผลเป็น Real-Time
Graph ให้ผู้ใช้ดู

NotebookLM

Knowledge Check: ทบทวนพื้นฐาน & โปรโตคอล

Q: IoT คืออะไร อธิบายง่ายๆ ได้ว่าอย่างไร?

A: สรรพสิ่งที่สื่อสารและเชื่อมต่อกันผ่านอินเทอร์เน็ต มีเซ็นเซอร์รับข้อมูล (ตา/หู) และสามารถส่งข้อมูลสั่งการได้แบบอัตโนมัติ

Q: ทำไม MQTT ถึงเหมาะกับ IoT?

A: เป็นโปรโตคอลที่เบามาก ส่งข้อมูลแบบ Push ได้ไวแบบ Asynchronous และออกแบบมาเพื่อจัดการอุปกรณ์จำนวนมากๆ ได้ดีเยี่ยม

Q: เครื่องหมาย + และ # ใน MQTT Wildcard ต่างกันอย่างไร?

A: + แทนค่าแค่ 1 ระดับชั้น (Single level) ส่วน # แทนทค่าทุกระดับชั้นย่อยที่ตามมาทั้งหมด (Multi-level)

Q: Retain-Flag ใน MQTT มีไว้ทำไม?

A: สั่งให้ Broker จำค่าล่าสุดไว้ เมื่อมีอุปกรณ์ใหม่เข้ามา Subscribe ก็จะได้รับค่านี้ทันที ไม่ต้องรอเซ็นเซอ์ปเดตข้อมูลรอบใหม่

NotebookLM

Knowledge Check: ฮาร์ดแวร์ & การตั้งค่า

Q: Platform ในทาง IoT หมายถึงอะไร?

A: ไม่ใช่แค่ชิ้นส่วนอิเล็กทรอนิกส์ แต่คือการทำงานร่วมกันของ Hardware (เช่น สถาปัตยกรรม CPU) และ Software (เช่น ระบบปฏิบัติการหรือ IDE)

Q: NodeMCU ต่างจาก Arduino พื้นฐานอย่างไร?

A: NodeMCU มีโมดูล WiFi (ESP8266) ฝังมาในตัว สามารถเชื่อมต่ออินเทอร์เน็ตได้ทันทีโดยไม่ต้องต่อโล่ (Shield) อุปกรณ์เสริม

Q: ถ้าไม่มีจอภาพ จะตั้งค่า IP ให้ Raspberry Pi ได้อย่างไร?

A: ใช้วิธี Headless โดยถอด SD Card มาเสียบคอมพิวเตอร์ แล้วแก้ไขไฟล์ cmdline.txt เพื่อกำหนดค่า IP Address ก่อนนำไปเสียบสาย LAN

พิมพ์เขียวของคุณพร้อมแล้ว... ได้เวลาลงมือสร้างสรรค์โลก IoT ของคุณเอง!

NotebookLM