

บทที่ 2

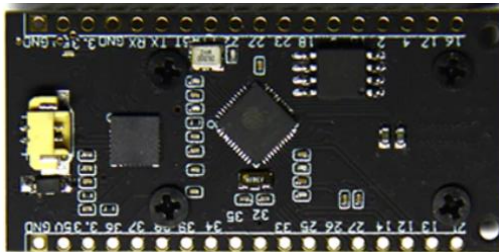
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในการจัดทำโครงการโรงเพาะเห็ดอัจฉริยะผู้จัดทำโครงการได้ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้องเพื่อใช้ในการดำเนินโครงการให้เป็นไปตามวัตถุประสงค์และขอบเขตของโครงการ

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 ไมโครคอนโทรลเลอร์ ESP32 LoRa

ESP32 LoRa เป็นบอร์ดที่รวมเอา ESP32 และโมดูล LoRa SX1278 ไว้ในบอร์ดเดียวทำให้สามารถสื่อสารแบบ Wi-Fi Bluetooth และสามารถส่งข้อมูลระยะไกลแบบ LoRa



รูปที่ 2.1 แสดงภาพของไมโครคอนโทรลเลอร์ ESP32 LoRa

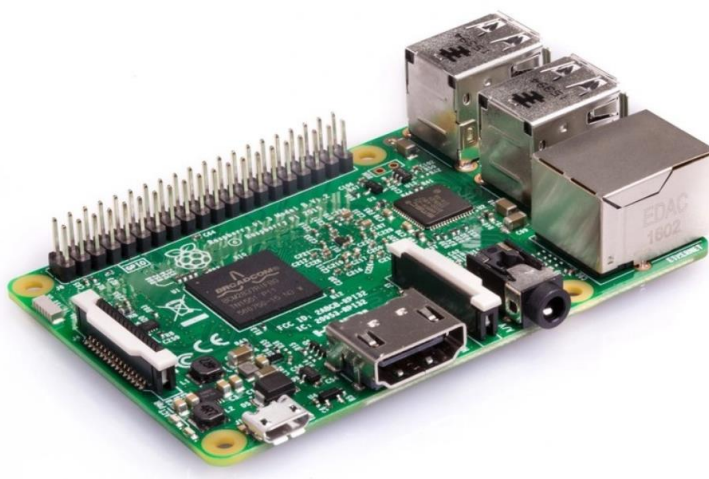
รายละเอียด ESP32 เป็นชื่อของไอซีไมโครคอนโทรลเลอร์ที่รองรับการเชื่อมต่อ WiFi และ Bluetooth 4.2 BLE ในตัว ผลิตโดยบริษัท Espressif จากประเทศจีน โดยไอซี ESP32 มีสเปคโดยละเอียด ดังนี้

- ซีพียูใช้สถาปัตยกรรม Tensilica LX6 แบบ 2 แกนสมอง สัญญาณนาฬิกา 240MHz
- มีแรมในตัว 512KB
- รองรับการเชื่อมต่อรอมภายนอกสูงสุด 16MB
- มาพร้อมกับ WiFi มาตรฐาน 802.11 b/g/n รองรับการใช้งานทั้งในโหมด Station softAP และ Wi-Fi direct
- มีบลูทูธในตัว รองรับการใช้งานในโหมด 2.0 และโหมด 4.0 BLE
- ใช้แรงดันไฟฟ้าในการทำงาน 2.6V ถึง 3V
- ทำงานได้ที่อุณหภูมิ -40°C ถึง 125°C
- นอกจากนี้ ESP32 ยังมีเซ็นเซอร์ต่าง ๆ มาในตัวด้วย ดังนี้
- วงจรกรองสัญญาณรบกวนในวงจรขยายสัญญาณ
- เซ็นเซอร์แม่เหล็ก
- เซ็นเซอร์สัมผัส (Capacitive touch) รองรับ 10 ช่อง
- รองรับการเชื่อมต่อคลิสตอล 32.768kHz สำหรับใช้กับส่วนวงจรนับเวลาโดยเฉพาะ
- ขาใช้งานต่าง ๆ ของ ESP32 รองรับการเชื่อมต่อบัสดังต่าง ๆ ดังนี้
- มี GPIO จำนวน 28 ช่อง
- รองรับ UART จำนวน 3 ช่อง
- รองรับ SPI จำนวน 2 ช่อง
- รองรับ I2C จำนวน 2 ช่อง
- รองรับ 12-bits ADC จำนวน 18 ช่อง
- รองรับ 8-bits DAC จำนวน 2 ช่อง
- รองรับ I2S จำนวน 1 ช่อง
- รองรับ PWM / Timer ทุกช่อง
- รองรับการเชื่อมต่อกับ SD-Card
- นอกจากนี้ ESP32 ยังรองรับฟังก์ชันเกี่ยวกับความปลอดภัยต่าง ๆ ดังนี้
- รองรับการเข้ารหัส WiFi แบบ WEP และ WPA/WPA2 PSK/Enterprise

รายละเอียด LoRa SX1278 ทำงานร่วมกับโปรโตคอลการสื่อสาร SPI ดังนั้นจึงสามารถใช้กับไมโครคอนโทรลเลอร์ที่รองรับ SPI ได้ จำเป็นต้องใช้ Ariel (เสาอากาศ) ร่วมกับโมดูลมิฉะนั้นอาจทำให้โมดูลเสียหายอย่างถาวร โมดูลควรใช้พลังงาน 3.3V เท่านั้นโดยมีคุณสมบัติต่างดังนี้

- ใช้แรงดันไฟฟ้าในการทำงาน 3.3V
- ความถี่ในการทำงาน 433MHz
- การสื่อสาร SPI แบบ Half-Duplex
- เทคนิคการมอดูเลต FSK, GFSK, MSK, GMSK, LoRa
- Packet size: 256 bytes
- ความไว: -148db

2.1.2 Raspberry Pi 3 Model B



รูปที่ 2.2 แสดงภาพ Raspberry Pi 3 Model B

Raspberry Pi 3 Model B เปรียบเสมือนคอมพิวเตอร์ขนาดเล็กที่สามารถเชื่อมต่อกับ จอ เมาส์ และคีย์บอร์ดได้ สามารถทำงานได้เหมือนคอมพิวเตอร์ทั่วไป เช่น การทำงานเอกสาร ดูหนัง ฟัง เพลง การใช้งานอินเทอร์เน็ต การเขียนโปรแกรม เป็นต้น

Raspberry Pi สามารถรองรับระบบปฏิบัติการ Linux ได้หลาย Distro เช่น Raspbian Ubuntu Pidora และ Arch Linux โดยติดตั้งระบบปฏิบัติการลงใน SD Card โดย Raspberry Pi สามารถนำไปประยุกต์ใช้ทำสิ่งต่างๆได้มากมาย เช่น Database Server Robot Microcontroller

รายละเอียดทางเทคนิคของ Raspberry Pi 3 Model B มีดังนี้

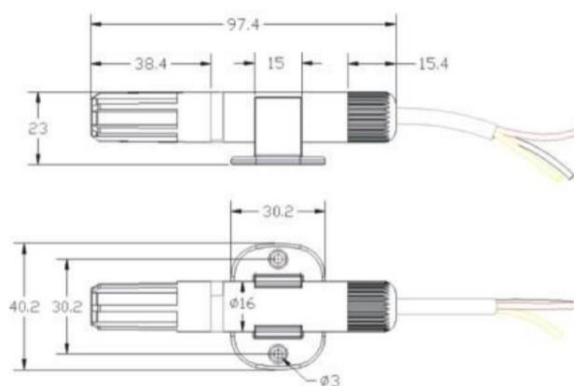
- Quad Core 1.2GHz Broadcom BCM2837 64bit CPU
- 1GB RAM
- BCM43438 wireless LAN and Bluetooth Low Energy (BLE) on board
- 100 Base Ethernet
- 40-pin extended GPIO
- 4 USB 2 ports
- 4 Pole stereo output and composite video port
- Full size HDMI
- CSI camera port for connecting a Raspberry Pi camera
- DSI display port for connecting a Raspberry Pi touchscreen display
- Micro SD port for loading your operating system and storing data
- Upgraded switched Micro USB power source up to 2.5A

2.1.3 AM2315 เซนเซอร์วัดอุณหภูมิและความชื้น



รูปที่ 2.3 แสดงภาพ AM2315 เซนเซอร์วัดอุณหภูมิและความชื้น

AM2315 คือเซ็นเซอร์วัดค่าอุณหภูมิและความชื้นคุณภาพสูงที่มีเอาต์พุตเป็นสัญญาณดิจิทัล โดยมีการสอบเทียบสัญญาณดิจิทัลเอาต์พุตแล้ว ใช้โมดูลตรวจจับอุณหภูมิและความชื้นคุณภาพสูงเพื่อให้แน่ใจว่ามีการวัดค่าที่แม่นยำและมีความทนทาน เซ็นเซอร์ประกอบด้วยตัววัดความชื้นประเภทความจุไฟฟ้า อุปกรณ์วัดอุณหภูมิที่มีความแม่นยำสูง และเชื่อมต่อกับไมโครคอนโทรลเลอร์ 8 บิตประสิทธิภาพสูง AM2315 สื่อสารด้วย I2C ผู้ใช้สามารถเชื่อมต่อ I2C bus ได้โดยตรง



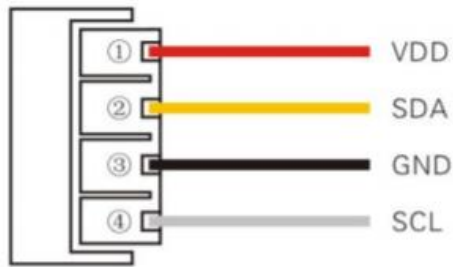
รูปที่ 2.4 แสดงภาพขนาดของ AM2315 ในหน่วยมิลลิเมตร (mm)

รายละเอียดทางเทคนิคของ AM2315 มีดังต่อไปนี้

- 3.5 to 5.5V power and I/O
- 10 mA max current use during conversion (while requesting data)
- Good for 0-100% humidity readings with 2% accuracy
- Good for -20 to 80°C temperature readings $\pm 0.1^{\circ}\text{C}$ typical accuracy
- 0.5 Hz update rate
- Body size 98mm x 16mm diameter (4" x 0.9")
- 4 wires 20 inches long
- Weight: 82.64g
- This board/chip uses I2C 7-bit address 05C.

Pin	Color	Name	Description
1	Red	VDD	Power (3.5V–5.5V)
2	Yellow	SDA	Serial data, bidirectional
3	Black	GND	Ground
4	White	SCL	Serial Clock, input

รูปที่ 2.5 แสดงภาพ AM2315 pin assignment



รูปที่ 2.6 แสดงภาพ AM2315 wiring diagram

Parameter	Condition	min	typ	max	Unit
Resolution			0.1		%RH
			16		bit
Accuracy ^[1]	25°C		± 2		%RH
Repeatability			± 0.1		%RH
Exchange		Completely interchangeable			
Response ^[2]	1/e(63%)		<5		S
Sluggish			<0.3		%RH
Drift ^[3]	Typical		<0.5		%RH/yr

รูปที่ 2.7 แสดงภาพ AM2315 relative humidity performance table

Parameter	Condition	min	typ	max	Unit
Resolution			0.1		°C
			16		bit
Accuracy			± 0.1	± 1	°C
Range		-40		125	°C
Repeat			± 0.2		°C
Exchange		Completely interchangeable			
Response	1/e(63%)		<5		S
Drift			± 0.1		°C/yr

รูปที่ 2.8 แสดงภาพ AM2315 relative temperature performance table

2.1.3.1 โปโตคอลการสื่อสาร I²C ของเซนเซอร์ AM2315

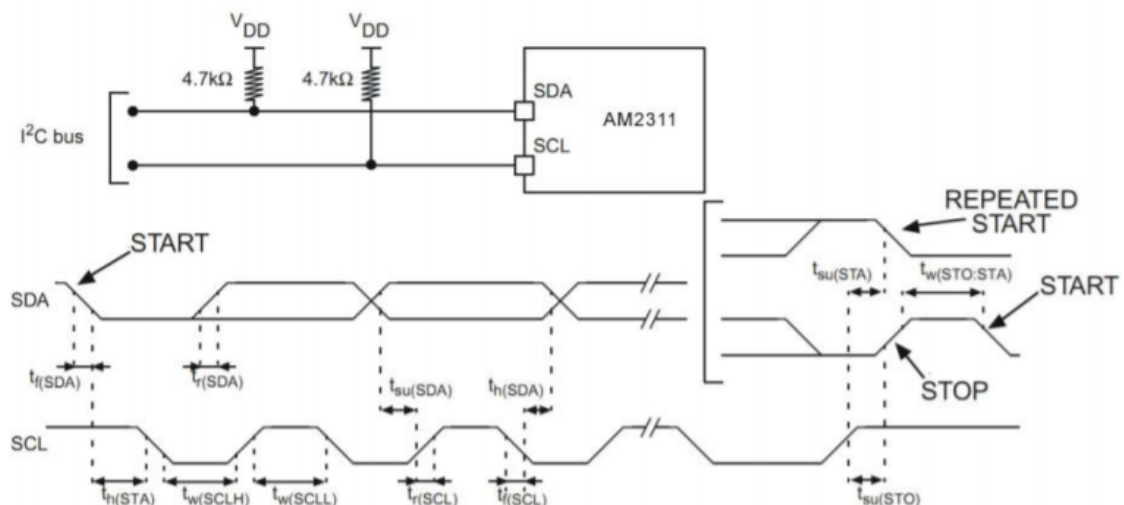
AM2315 I²c bus มีจุดเชื่อมต่อ (Interface) ที่ได้มาตรฐาน สามารถต่อกับ I²C bus ของไมโครคอนโทรลเลอร์ได้โดยตรง เซนเซอร์ AM2315 มี I2C address (SLAVE ADDRESS) คือ 0xB8

2.1.3.1.1 คำอธิบายอินเทอร์เฟซ I2C ของเซนเซอร์ AM2315

การวัดค่าอุณหภูมิและความชื้นที่แม่นยำและลดการทำงานของเซนเซอร์ เมื่อเซนเซอร์ไม่มีการทำงานเซนเซอร์จะเปลี่ยนเป็นโหมด Sleep โดยอัตโนมัติ เพื่อลดการทำงานของเซนเซอร์ ลดอุณหภูมิภายในของเซนเซอร์ เมื่อต้องการจะอ่านค่าจากเซนเซอร์ที่อยู่ในโหมด Sleep ไมโครคอนโทรลเลอร์จะต้องจะคำสั่งเพื่อปลุกการทำงานของเซนเซอร์ก่อน ดังนั้นถ้าไม่มีการอ่านค่าจากเซนเซอร์เป็นเวลานาน เมื่อจะอ่านค่าจากเซนเซอร์ต้องอ่านค่าอย่างน้อย 2 วินาที เพราะค่าที่อ่านได้วินาทีแรกจะเป็นค่าล่าสุดก่อนเซนเซอร์จะเข้าโหมด Sleep

2.1.3.1.2 คุณสมบัติอินเทอร์เฟซของ I²C

ลักษณะเฉพาะ I²C ของ AM2315 ถ้าต้องการค่าอุณหภูมิและความชื้นที่ดีที่สุด การต่อสาย I²C bus จะต้องต่อดังรูปที่ 2.9



รูปที่ 2.9 แสดงภาพการต่อวงจรกับ I²C bus ของ AM2315

2.1.3.1.3 โพรโทคอลการสื่อสาร

เซนเซอร์ AM2315 มีโปรโตคอลการสื่อสาร I²C bus ที่ได้มาตรฐานโดยอ้างอิงจาก โปรโตคอล ModBus เซนเซอร์ AM2315 ตามมาตรฐานโรงงานได้รวมเอา I²C_ModBus โดยมีรูปแบบดังต่อไปนี้

1) รูปแบบการสื่อสารข้อมูล (Communication data format)

Data format:	I ² CAdd+R/W	Function code	Data area	CRC Check ^[3]
Data length:	1Byte ^[2]	1Byte	NByte	16-bit CRC (cycle redundancy code)

รูปที่ 2.10 แสดงภาพรูปแบบการสื่อสารข้อมูลของ AM2315

2) การสื่อสารและกระบวนการถ่ายโอนข้อมูล

เมื่อมีการส่งคำสั่งการสื่อสารจากไมโครคอนโทรลเลอร์ (host) ไปยัง เซนเซอร์ และเซนเซอร์ได้รับคำสั่งแล้วจะประมวลผลคำสั่งและอ่านค่าจากนั้นจะส่งข้อมูลที่ได้กลับไปยังไมโครคอนโทรลเลอร์ (host) ข้อมูลที่ส่งกลับไปจะประกอบไปด้วย function code, ข้อมูล (data) และ CRC code

3) I²C slave address ของเซนเซอร์ AM2315

เซนเซอร์ AM2315 ทุกตัวจะมี I²C address เหมือนกันคือ 0xB8 ดังนั้น ใน I²C bus จะสามารถมีเซนเซอร์ AM2315 ได้เพียงแค่ 1 เซนเซอร์เท่านั้น

4) การสื่อสาร I²C function code

ในการสื่อสาร I²C function code คือ byte แรกของเฟรมข้อมูล (frame transmission) กฎของการสื่อสาร I²C_ModBus กำหนด function code ระหว่าง 1 ถึง 127 เช่น ไมโครคอนโทรลเลอร์ (host) ส่งความต้องการที่จะดำเนินงานผ่านทาง function code เพื่อบอกเซนเซอร์ดำเนินการตามที่ต้องการ การตอบกลับของเซนเซอร์ก็จะส่ง function code เดียวกันไปยังไมโครคอนโทรลเลอร์ (host) เช่นกัน ตัวอย่างของ function code แสดงดังรูปที่ 2.11

Function code	Definition	Operations (binary)
0x03	Read register data	Read data from one or more registers
0x10	Write multiple registers	Multiple sets of binary data is written to multiple registers

รูปที่ 2.11 แสดงภาพตัวอย่างของ function code

5) พื้นที่ข้อมูลการสื่อสาร I²C

พื้นที่ข้อมูล (data area) ประกอบด้วยข้อมูลที่ต้องการจากเซนเซอร์ ตัวอย่างของข้อมูล เช่น อุณหภูมิ ความชื้น ข้อมูลต่างๆของเซนเซอร์ ข้อมูลที่เขียน เป็นต้น ที่อยู่อ้างอิง (reference address) ตัวอย่างเช่น ไมโครคอนโทรลเลอร์ (host) บอกเซนเซอร์ว่า 03 ผ่านทาง function code ข้อมูลที่ได้จะประกอบด้วย ความยาวของข้อมูล และ ข้อมูลภายใน register

I²C_Modbus เซนเซอร์ใช้โปรโตคอลการสื่อสารแบบกำหนดเอง (custom) ไมโครคอนโทรลเลอร์ (host) ใช้คำสั่งการสื่อสาร (function code 03) สามารถอ่านข้อมูลใน register ที่แสดงในรูปที่ 2.12 ได้ทั้งหมด แต่ละ register จะมีขนาดเพียง 1 byte ถ้ามีการอ่านข้อมูลจากเซนเซอร์มากกว่า 10 register ขึ้นไป จะได้ค่า error จากเซนเซอร์

Register information	Add	Register information	Add	Register information	Add	Register information	Add
High RH	0x00	Model high	0x08	Users register a high	0x10	Retention	0x18
Low RH	0x01	Model low	0x09	Users register a low	0x11	Retention	0x19
High temp.	0x02	Version number	0x0A	Users register 2 high	0x12	Retention	0x1A
Low temp.	0x03	ID(24–31) Bit	0x0B	Users register 2 low	0x13	Retention	0x1B
Retention	0x04	ID(16–23) Bit	0x0C	Retention	0x14	Retention	0x1C
Retention	0x05	ID(8 – 15) Bit	0x0D	Retention	0x15	Retention	0x1D
Retention	0x06	ID(0 – 7) Bit	0x0E	Retention	0x16	Retention	0x1E
Retention	0x07	Status Register	0x0F	Retention	0x17	Retention	0x1F

รูปที่ 2.12 แสดงภาพ AM2315 Data register

6) รูปแบบเอาท์พุตของอุณหภูมิ (Temperature output format)

ค่าอุณหภูมิมีความละเอียด 16bit ตำแหน่งค่าอุณหภูมิมากที่สุด (Bit15) เท่ากับ 1 สำหรับค่าอุณหภูมิที่ติดลบ มีค่าเท่ากับ 0 สำหรับค่าอุณหภูมิที่เป็นบวก ค่าอุณหภูมิจะอยู่ที่ (Bit14-Bit0) เป็นตัวอักษร (string)

7) สถานะของ Register

สถานะของ Register Bit7-Bit0 bit แสดงดังรูปที่ 2.13

Status register bits	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Features	Retention							

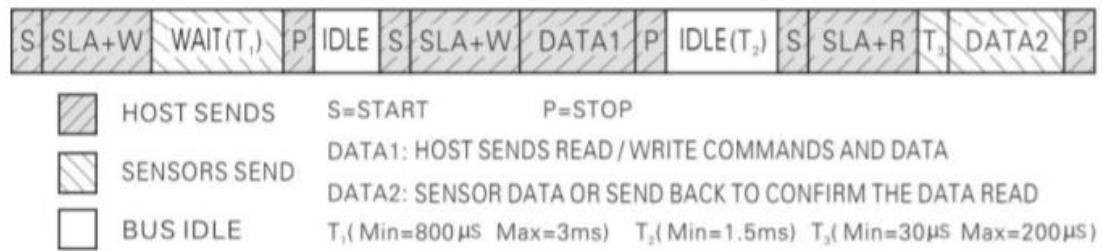
รูปที่ 2.13 แสดงภาพสถานะของ Register

2.1.3.1.4 Timing การสื่อสาร I²C

การสื่อสาร I²C ของเซนเซอร์ AM2315 แม้ว่าจะอ้างอิงมาตรฐาน I²C Timing แต่จำเป็นสำหรับโปรโตคอลของเซนเซอร์ AM2315 และ ข้อกำหนดด้านเวลา (timing) ในการสื่อสาร สำหรับการอ่านค่าจากเซนเซอร์ที่ถูกต้อง ต้องทำตามข้อกำหนดและ timing

1) ตัวอย่างลำดับการอ่านค่าจากเซนเซอร์

รูปที่ 2.14 ตัวอย่างการอ่านค่าจากเซนเซอร์ AM2315 การอ่านค่าที่ถูกต้องต้องปฏิบัติตามข้อตกลงอย่างเข้มงวด จากรูปการสื่อสารระหว่างไมโครคอนโทรลเลอร์ (host) กับเซนเซอร์ตั้งแต่เริ่มต้นจนถึงสิ้นสุดจะใช้เวลานานที่สุดคือ 3 วินาที



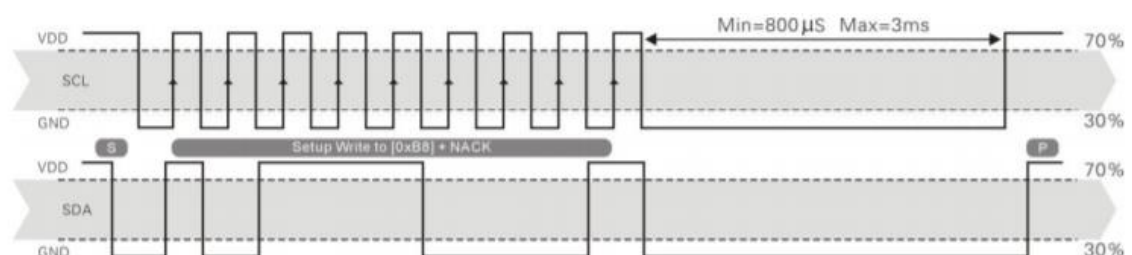
รูปที่ 2.14 แสดงภาพตัวอย่างการอ่านค่าจากเซนเซอร์ AM2315

2) I²C Read and write timing decomposition

การอ่านหรือเขียนเซนเซอร์ ต้องเป็นไปตาม 3 ขั้นตอนต่อไปนี้ มิฉะนั้นจะไม่สามารถสื่อสารกับเซนเซอร์ได้หรือไม่สามารถอ่านค่าที่ต้องการ

ขั้นตอนที่ 1 ปลุกการทำงานของเซนเซอร์ ในการลดความร้อนภายในเซนเซอร์ เซนเซอร์จะอยู่ในสถานะไม่ทำงาน ต้องปลุกการทำงานของเซนเซอร์ก่อนส่งคำสั่งอ่านและเขียน มิฉะนั้นเซนเซอร์จะไม่ตอบสนอง (respond) โดยการตอบกลับด้วยการส่ง ACK การปลุกการทำงานของเซนเซอร์ทำได้ตามคำสั่งต่อไปนี้

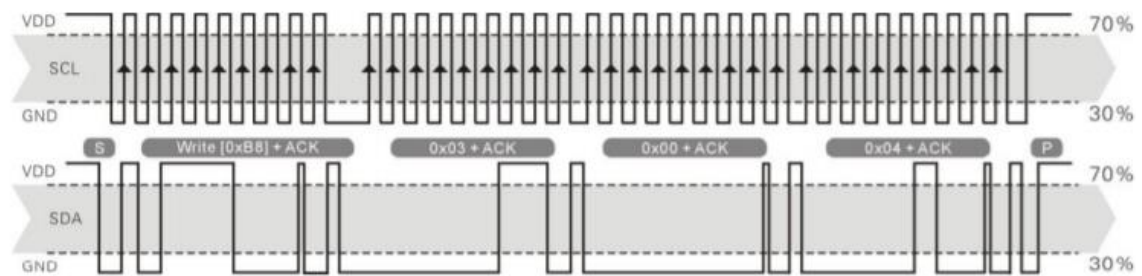
ส่งสัญญาณเริ่มต้น (start signal) ใช้กับโฮสต์ที่อยู่เริ่มต้น รอช่วงเวลาหนึ่ง แล้วส่งสัญญาณสิ้นสุด (stop signal) กล่าวคือ initial signal + 0xB8 + wait (800us-3s) + stop signal ดังรูปที่ 2.15



รูปที่ 2.15 แสดงภาพการปลุกการทำงานของเซนเซอร์

ขั้นตอนที่ 2 อ่านคำสั่งเพื่อส่งหรือส่งคำสั่งเพื่อเขียน

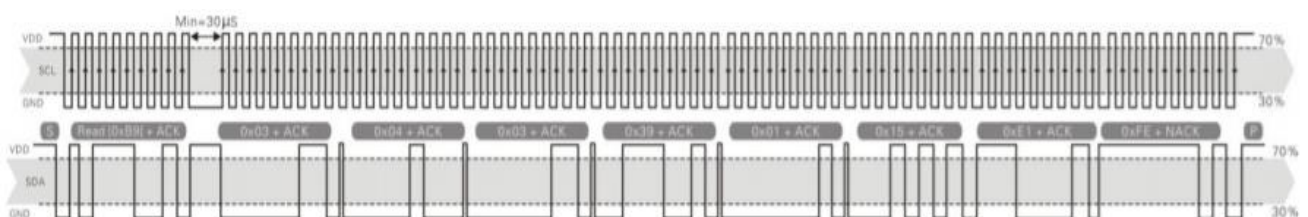
เมื่อปลุกการทำงานของเซนเซอร์แล้วจะสามารถอ่านข้อมูลได้อย่างครบถ้วนด้วยมาตรฐาน I²C timing รองรับความเร็วสูงสุด 100Kb/s การอ่านค่าอุณหภูมิและความชื้นเบื้องต้นแสดงดังรูปที่ 2.16 โดยไมโครคอนโทรลเลอร์ (host) จะส่งคำสั่งไปดังนี้ START + 0xB8 (SLA) + 0x03 (function code) + 0x00 (start address) + 0x04 (register length) + STOP



รูปที่ 2.16 แสดงภาพตัวอย่างการส่งคำสั่งเพื่ออ่านค่าอุณหภูมิและความชื้น

ขั้นตอนที่ 3 ข้อมูลอ่านกลับหรือสัญญาณยืนยัน

เมื่อไมโครคอนโทรลเลอร์ (host) ส่งคำสั่งเพื่ออ่านหรือเขียนไปแล้วต้องรอนานอย่างน้อย 1.5ms แล้วขั้นตอนต่อไปก็ส่งลำดับการอ่าน (read sequence) เพื่ออ่านค่าจากเซนเซอร์ดังรูปที่ 2.17 เมื่ออ่านค่าเสร็จแล้ว I²C address จะต้องการรอเป็นเวลาก่อนอย่างน้อย 30us ก่อนที่จะส่งสัญญาณเพื่ออ่านค่าต่อไป หรือส่งค่า error



รูปที่ 2.17 แสดงภาพตัวอย่างการค่าอุณหภูมิและความชื้น

ตัวอย่างของข้อมูลที่อ่านได้และส่งกลับไปยังไมโครคอนโทรลเลอร์ (host)

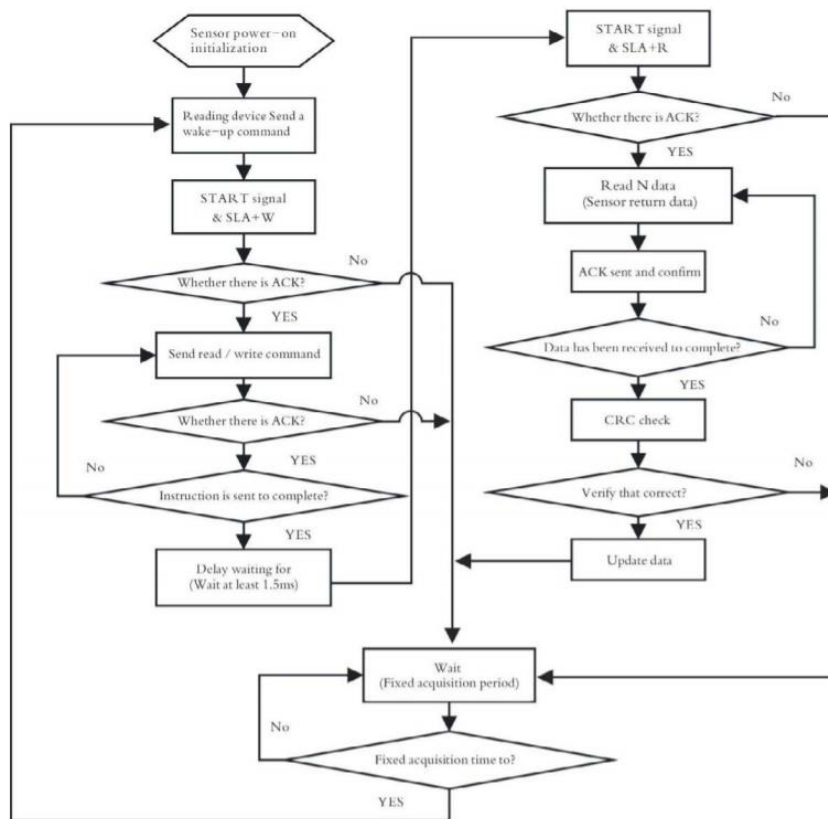
0x03(Function code) + 0x04(Data length) + 0x03(High humidity) + 0x39(Low humidity) + 0x01(High temperature) + 0x15(Low temperature) + 0xE1(CRC Low byte checksum) + 0xFE(CRC High byte checksum);

ดังนั้น $0339H = 3 \times 256 + 3 \times 16 + 9 = 825 \Rightarrow \text{Humidity} = 825/10 = 82.5\%RH$;

$0115H = 1 \times 256 + 1 \times 16 + 5 = 277 \Rightarrow \text{Temperature} = 277/10 = 27.7^{\circ}C$;

จากขั้นตอนทั้งหมดที่ผ่านมาเป็นกระบวนการอ่านค่าจากเซนเซอร์ ถ้าสามารถปฏิบัติตามขั้นตอนทั้งหมดได้ก็จะสามารถใช้งานเซนเซอร์ได้อย่างมีประสิทธิภาพ การอ่านค่าจากเซนเซอร์อย่างต่อเนื่องค่าได้จะอัปเดตทุก 2 วินาทีเป็นอย่างน้อย (minimum interval of continuous reading 2S)

2.1.3.1.5 Flow chart การทำงานของเซนเซอร์



รูปที่ 2.18 แสดงภาพ Flow chart การทำงานของเซนเซอร์

2.1.4 4-Channel Relay Module



รูปที่ 2.19 แสดงภาพของ 4-Channel Relay Module

รีเลย์เป็นสวิตช์ที่ทำงานด้วยไฟฟ้าและเช่นเดียวกับสวิตช์อื่น ๆ ที่สามารถเปิดหรือปิดได้โดยปล่อยให้กระแสไหลผ่านหรือไม่ สามารถควบคุมด้วยแรงดันไฟฟ้าต่ำเช่น 3.3V ที่ให้มาโดย Microcontroller GPIO และช่วยให้สามารถควบคุมแรงดันไฟฟ้าสูงเช่น 12V, 24V หรือแรงดันไฟหลัก (220V)

รายละเอียดทางเทคนิคของ 4-Channel Relay Module มีดังนี้

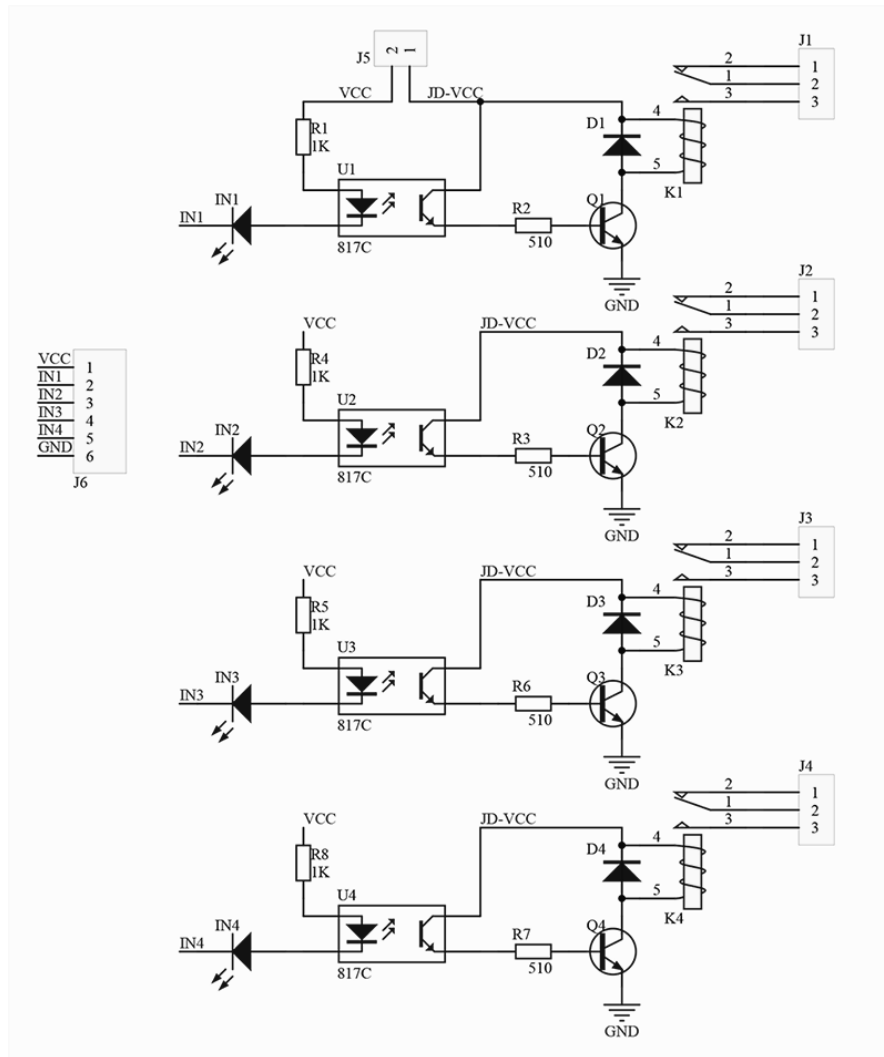
- 4-Channel Relay breakout
- Power supply range from 5V~7.5V
- Onboard Photocoupler isolation
- Equiped with high-current relay, AC250V 10A ; DC30V 10A.
- Relay Output Indicator LED
- VCC is system power source, and JD_VCC is relay power source. Ship 5V relay by default. Plug jumper cap to use
- TTL logic interface can be directly connected to microcontroller (Arduino, 8051, AVR, PIC, DSP, ARM, ARM, MSP430, etc)

จากรูปที่ 2.19 ประกอบด้วย relay จำนวน 4 ตัว แต่ละตัวมีสามช่อง common(COM), Normally Closed(NC), Normally Open(NO)

- COM เชื่อมต่อกับแหล่งจ่ายแรงดัน (220V, 12V, 24V)
- NC (Normally Closed) ใช้เมื่อต้องการให้รีเลย์ทำงานแบบ Normally Closed หมายความว่าวงจรปิด(กระแสไหล) เว้นแต่จะส่งสัญญาณจาก Microcontroller ไปยังโมดูลรีเลย์เพื่อเปิดวงจรและหยุดการไหลของกระแส
- NO (Normally Open) ใช้เมื่อต้องการให้รีเลย์ทำงานแบบ Normally Open หมายความว่าวงจรเปิด(กระแสไม่ไหล) เว้นแต่จะส่งสัญญาณจาก Microcontroller เพื่อปิดวงจร

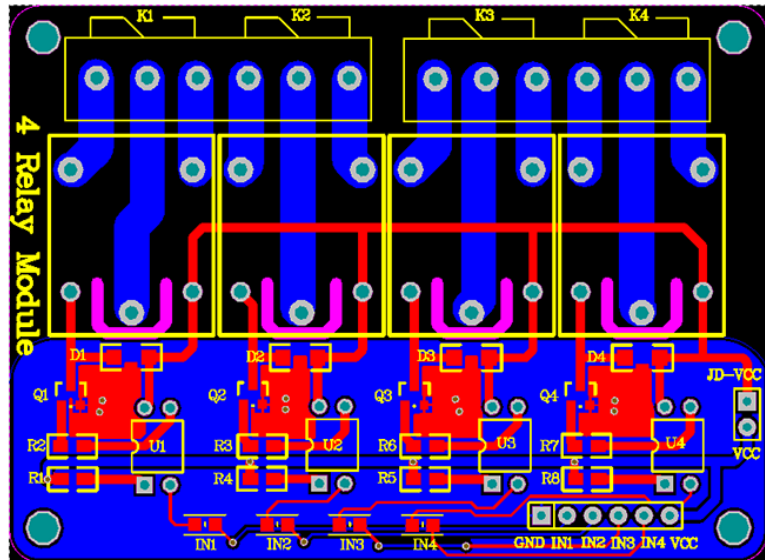
ด้านแรงดันไฟต่ำ(ต่อกับ Microcontroller) ประกอบด้วย 6 พินดังนี้

- VCC แรงดันไฟฟ้าบวก
- GND Ground
- IN1-IN4 พอร์ตสำหรับควบคุมรีเลย์



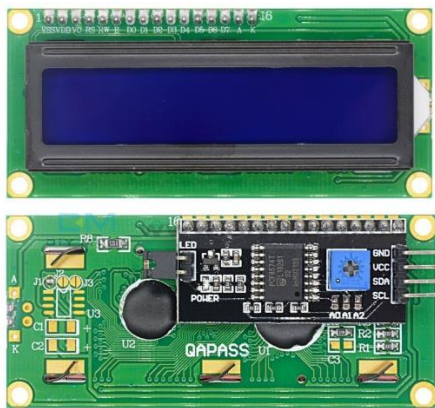
รูปที่ 2.20 แสดงภาพของ 4-channel relay schematic

จากรูปด้านบน เมื่อพอร์ตสัญญาณ(IN) อยู่ในระดับต่ำ(LOW) ไฟสัญญาณจะสว่างขึ้นและออปโตคัปเปลอร์ 817c จะแปลงแสงเป็นสัญญาณทางไฟฟ้า ออปโตคัปเปลอร์ 817c สามารถแยกสัญญาณไฟฟ้าอินพุตและเอาต์พุตออกจากกัน จากนั้นทรานซิสเตอร์จะทำงาน ขดลวดรีเลย์จะถูกกระแสไฟฟ้า และหน้าสัมผัส(Contact) ที่เปิดตามปกติ(Normally Open) ของรีเลย์จะปิดลง เมื่อพอร์ตสัญญาณ(IN) อยู่ในระดับสูง(HIGH) หน้าสัมผัส(Contact) ที่ปิดตามปกติ(Normally Close) ของรีเลย์จะปิด



รูปที่ 2.21 แสดงภาพ 4-channel relay PCB

2.1.5 จอแอลซีดี (I2C LCD)



รูปที่ 2.22 แสดงภาพของจอแอลซีดี (I2C LCD)

I2C LCD ประกอบด้วยวงจรเสริมขนาดเล็กที่ติดตั้งที่ด้านหลังของโมดูล คือชิป PCF8574 ใช้สำหรับการสื่อสาร I2C และโพเทนชิโอมิเตอร์เพื่อปรับแสงจอ (backlight) ข้อดีของ I2C LCD คือการต่อสายทำได้ง่ายใช้เพียงสองพินในการควบคุม LCD โดยทั่วไปแล้ว LCD มาตรฐานต้องการการเชื่อมต่อประมาณ 12 พินซึ่งอาจเป็นปัญหาได้หากคุณไม่มีพิน GPIO เหลือพอ

บัส I2C เป็นบัสอนุกรมที่คิดค้นโดย PHILIPS เป็นบัสอนุกรมประสิทธิภาพสูงซึ่งมีการกำหนดเวลาบัสและฟังก์ชันซิงโครไนซ์อุปกรณ์ความเร็วสูงหรือต่ำที่ระบบโฮสต์หลายระบบต้องการ บัส I2C มีเพียงสองสายสัญญาณสองทิศทาง Serial Data Line (SDA) และ Serial Clock Line (SCL)

ข้อมูลจำเพาะของ 16 × 2, 20 × 4 และ LCD ขนาดอื่น ๆ ส่วนใหญ่จะเหมือนกัน เพราะทั้งหมดใช้ตัวควบคุม HD44780 Hitachi LCD เหมือนกัน

รายละเอียดทางเทคนิคของ จอแอลซีดี (I2C LCD) มีดังต่อไปนี้

- Operating voltage 5 V
- Controller Hitachi HD44780 LCD controller
- Default address 0x27
- Screen resolution 2-lines × 16 characters
- Character resolution 5 × 8 pixels
- Module dimensions 80 × 36 × 12 mm
- Viewing area dimensions 64.5 × 16.4 mm

2.1.6 เครื่องปั๊มน้ำ (Pump)

เครื่องปั๊มน้ำ (Pump) เป็นอุปกรณ์ที่ช่วยส่งผ่านพลังงานจากแหล่งต้นกำเนิดไปยังของเหลว เพื่อให้ของเหลวเคลื่อนที่จาก ตำแหน่งหนึ่งไป ยังอีก ตำแหน่งหนึ่งที่อยู่สูงกว่า หรือในระยะทางที่ไกลออกไป โดยจุดเริ่มต้นของเครื่องปั๊มน้ำนี้มีประวัติศาสตร์ที่ยาวนานกว่า 2,000 ปีก่อนคริสตศักราช ซึ่งในช่วงเริ่มแรกมีการใช้พลังงาน ที่ได้จากมนุษย์ สัตว์ ต่อมาจึงได้ใช้พลังงานจากธรรมชาติ เช่น พลังงานจากลม และน้ำเป็นแหล่งต้นกำเนิด ซึ่งในช่วงแรกเพียง เพื่อการอุปโภคบริโภคและทำการเกษตรเท่านั้น

ในปัจจุบันเครื่องปั๊มน้ำจัดเป็นอุปกรณ์เครื่องมืออีกชนิดหนึ่งที่มีความเกี่ยวข้องกับชีวิตความเป็นอยู่ของมนุษย์อย่างมาก เป็นอุปกรณ์ ที่ช่วยจัดส่งน้ำเพื่อการอุปโภค บริโภค การเกษตร คมนาคม

อุตสาหกรรม ตลอดจนการบำบัดน้ำเสีย เพื่อรักษา สภาวะแวดล้อม ที่ดีให้กับมนุษย์ ซึ่งวิวัฒนาการของ เครื่องปั้มน้ำในปัจจุบันได้เปลี่ยนไปจากเดิม ที่ใช้พลังงานจาก แหล่งธรรมชาติมาเป็น การใช้พลังงานจาก ใอน้ำ จากเครื่องยนต์ และที่นิยมกันมากคือ การใช้พลังงานไฟฟ้า เนื่องจากความสะดวกและง่ายต่อการใช้ งาน

2.1.6.1 ปั้มน้ำหมอก DC12V 4.8 บาร์



รูปที่ 2.23 แสดงภาพปั้มน้ำหมอก DC12V 4.8 บาร์

รายละเอียดทางเทคนิคของ ปั้มน้ำหมอก DC12V 4.8 บาร์ มีดังต่อไปนี้

- แรงดันไฟฟ้า : DC 12V
- กำลังไฟฟ้า: 24 วัตต์
- กินกระแส : 2 แอมป์
- แรงดัน : 4.8 บาร์

- อัตราการไหลสูงสุด : 3.5 ลิตร / นาที หรือ 210 ลิตร / ชั่วโมง
- ท่อน้ำเข้า - ออก : 3 นิ้ว (3/8 นิ้ว) สามารถใช้สาย 3 นิ้ว สวมเข้ากับท่อเข้า-ออก บิมน้ำได้เลย
- ระยะดูดลึก : ไม่เกิน 1.8 เมตร
- ระบบการทำงาน : ไม่มีสวิตช์แรงดันอัตโนมัติ (NO Pressure switch) เหมือนปั้มน้ำอัตโนมัติ
- สามารถใช้งานกับแบตเตอรี่ 12V หรือแผงโซลาร์เซลล์ไม่ควรต่ำกว่า 30 วัตต์ 12V หรือ 18V
- ขนาดโดยประมาณ 15*8*7 ซม.
- น้ำหนัก 0.5 กก.
- ควรใช้งานต่อเนื่องนานประมาณ 2 ชั่วโมง พัก 15 นาที เพื่อป้องกันการไหม้ของมอเตอร์
- บิมน้ำไม่ควรโดนน้ำ เพราะจะทำให้น้ำซึมผ่านเข้าไปในมอเตอร์ ทำให้มอเตอร์ไหม้ได้
- สามารถใช้เป็นปั้มพ่นยา พ่นปุ๋ย สามารถใช้สายพ่นยาได้ 50 เมตร
- สามารถพ่นได้ไกลสูงประมาณ 4 เมตร

2.1.7 พัดลม (Fan)

พัดลม (Fan) เป็นอุปกรณ์ที่ช่วยให้เกิดการไหลของอากาศ ปัจจุบันพัดลมเป็นอุปกรณ์ที่มีใช้โดยทั่วไปในโรงงานอุตสาหกรรมและอาคารบ้านเรือน

2.1.7.1 การจำแนกพัดลมสามารถแบ่งได้เป็นประเภทใหญ่ๆ ตามลักษณะการเคลื่อนที่ของอากาศ

1) พัดลมแบบหมุนแรงเหวี่ยง (Centrifugal flow or radial fans)

พัดลมแบบแรงเหวี่ยงหรือพัดลมซึ่งมีการไหลของอากาศในแนวรัศมีจะประกอบด้วยใบพัดหมุนอยู่ภายในตัวเรือนของพัดลม (Fan house) ชุดใบพัดจะประกอบด้วยแผ่นใบเล็กๆประกอบเข้าด้วยกัน เป็นลักษณะก่อกวนความดันของอากาศจะถูกทำให้มีค่าสูงขึ้นภายในตัวเรือนของพัดลมซึ่งสามารถเพิ่มค่าให้สูงขึ้นได้ด้วยการเพิ่มขนาดความยาวของใบพัดซึ่งจะทำให้แรงเหวี่ยงหนีศูนย์กลางภายในระบบมีค่ามากขึ้นอากาศจะ

ไหลผ่านเข้าไปในท่อทางเข้าโดยมีทิศทางขนานกับแกนของใบพัดและไหลออกใน ทิศทางตั้งฉากกับแกนของเพลลาใบพัดในท่อทางออกพัดลมประเภทนี้จำแนกตามลักษณะรูปร่างของ ใบพัดเป็น 3 แบบ คือ

1.1) แบบใบพัดตรง (Straight blade หรือ Radial fans)

พัดลมชนิดนี้มีจำนวนใบน้อยที่สุดประมาณ 6 ถึง 20 ใบและใบพัดจะอยู่ในระนาบรัศมีจากเพลลา ใบพัดหมุนด้วยความเร็วรอบอย่างต่ำประมาณ 500-3000 รอบ/นาที ดังนั้นจึงเหมาะกับการทำงานที่ต้องการปริมาตร การไหลน้อยๆและมีค่าความดันของอากาศสูงๆ

1.2) แบบใบพัดโค้งไปข้างหน้า (Forward curved blade fans)

พัดลมชนิดนี้จะมีใบพัดโค้งไปข้างหน้าในทิศทางเดียวกับการหมุนชุดใบพัดจะมีจำนวนแผ่นใบพัดประมาณ 20 – 60 ใบชุดใบพัดจะมีลักษณะคล้ายกับกรงกระรอก (Squirrel cage) เพลลาใบพัดจะมี ขนาดเล็กหมุนด้วยความเร็วรอบที่สูงกว่าพัดลมชนิดใบพัดตรงการทำงานของพัดลมชนิดนี้มีเสียงเบา ที่สุดมีข้อเสียคือมีโอกาสที่มอเตอร์จะทำงานเกินกำลังและมีช่วงการทำงานของพัดลมที่ไม่เสถียรดังนั้น จึงไม่ควรใช้กับงานหรือระบบที่มีอัตราการไหลของอากาศเปลี่ยนแปลงตลอดเวลาพัดลมชนิดนี้จะให้ ค่าความดันลมและอัตราการไหลของอากาศสูงที่สุด

1.3) แบบใบพัดโค้งไปข้างหลัง (Backward curved blade fans)

พัดลมชนิดนี้จะมีใบพัดเอียงไปข้างหลังในทิศทางตรงกันข้ามกับทิศทางการหมุนของใบพัด จะมี จำนวนใบพัดประมาณ 10 –50 ใบ และเป็นพัดลมที่มีความเร็วรอบสูง ไม่ก่อให้เกิดเสียงดังเกินควรไม่ มีลักษณะที่มอเตอร์จะทำงานเกินกำลัง และไม่มีช่วงการทำงานที่ไม่เสถียรเหมาะที่จะใช้งานระบายอากาศและอากาศที่ใช้ต้องสะอาดด้วยเนื่องจากสามารถที่จะควบคุมความดันและปริมาณลมได้ง่าย พัดลมชนิดนี้จะมีราคาสูงกว่าชนิดอื่นๆเมื่อเทียบขนาดเท่ากัน

2) พัดลมแบบอากาศไหลตามแนวแกน (Axial flow fans)

พัดลมแบบนี้อากาศจะไหลขนานกับแกนของใบพัดและตั้งฉากกับระนาบการหมุนของใบพัดชุดใบพัด จะถูกติดตั้งบนแกนเพลลาขับของมอเตอร์ต้นกำลัง ซึ่งอยู่ภายในตัวพัดลมทำให้มอเตอร์สามารถระบาย ความร้อนออกไปกับอากาศที่ถูกขับเคลื่อนพัดลมชนิดนี้มีราคาถูกการทำงานของพัดลมมีเสียงดังและมีช่วงการ

ทำงานของพัดลมที่ไม่เสถียรจึงเหมาะกับการระบายอากาศ มีขนาดเล็ก เคลื่อนย้ายง่าย สามารถแบ่งได้ 2 ลักษณะคือ

2.1) พัดลมที่ให้ลมหมุนเป็นเกลียว (Tube axial fans)

พัดลมแบบอากาศไหลตามแนวแกนชนิดนี้มีโครงสร้างประกอบด้วยชุดใบพัดซึ่งหมุนอยู่ภายในท่อรูปทรงกระบอกลมที่ถูกขับเคลื่อนให้ผ่านชุดใบพัดจะหมุนเป็นเกลียวมีลักษณะการไหลแบบปั่นป่วน พัดลมชนิดนี้ให้ค่าความดันลมปานกลาง

2.2) พัดลมที่ให้ลมในแนวเส้นตรง (Vane axial fans)

พัดลมแบบอากาศไหลตามแนวแกนชนิดนี้จะมีแผ่นครีปเพื่อใช้ในการบังคับการไหลของอากาศที่ถูกขับเคลื่อนติดตั้งอยู่ภายในตัวเรือนของพัดลม บริเวณท่อทางออกบริเวณด้านหลังชุดใบพัดเพื่อช่วยให้ การไหลของอากาศที่ถูกขับเคลื่อน มีทิศทางเป็นเส้นตรงมากที่สุดซึ่งจะช่วยลดลักษณะการไหลของ อากาศปั่นป่วนลดลงและลดพลังงานสูญเสียเนื่องจากการไหลของอากาศปั่นป่วนภายในระบบให้ น้อยลงทำให้ประสิทธิภาพการใช้งานและราคาสูงกว่าพัดลมชนิด Tube axial fans

2.1.7.2 การทำงานของพัดลม

การทำงาน และกลไกที่ทำให้พัดลมกับที่หรือหมุนสายไปมาพัดลมจะทำงานได้เมื่อกระแสไฟฟ้า เข้าสู่ระบบ และเมื่อกดปุ่มเลือกให้ลมแรงหรือเร็วตามที่ผู้ใช้งานต้องการ กระแสไฟฟ้าจึงไหลเข้าสู่ ตัวมอเตอร์ ทำให้แกนมอเตอร์หมุนใบพัดที่ติดอยู่กับแกนก็หมุนตามไปด้วยจึงเกิดเป็น ลมพัดออกมา พัดลมตั้งโต๊ะจะมีราคาต่ำกว่าพัดลมตั้งพื้นและใช้พลังงานไฟฟ้าต่ำกว่า ทั้งนี้เพราะมีขนาดมอเตอร์และกำลังไฟฟ้าต่ำกว่าและพัดลมตั้งพื้นจะให้ลมแรงมากกว่า

2.1.7.3 พัดลม 12 นิ้ว 10 ใบ 12V



รูปที่ 2.24 แสดงภาพพัดลม 12 นิ้ว 10 ใบ 12V

รายละเอียดทางเทคนิคของ พัดลม 12 นิ้ว 10 ใบ 12V มีดังต่อไปนี้

- แรงดันไฟฟ้า : DC 12V
- กำลังไฟฟ้า: 80 วัตต์
- กินกระแส : 6.6 แอมป์
- ขนาด 12 นิ้ว
- 10 ใบพัด
- ใบพัดบาง น้ำหนักเบา มีตะแกรงกันการกระเด็น
- สามารถใช้ระบายความร้อนในรถยนต์ หรืองานตัดแปลงอื่นๆ

2.1.8 LoRa (Long Range)

LoRa มาจากคำว่า Long Range เป็นหนึ่งในเทคโนโลยีด้าน LPWAN (Low-Power Wide-Area Network) เครือข่ายสื่อสารแบบกว้างที่เน้นใช้พลังงานต่ำ เป็นเทคโนโลยีที่ถือว่าเป็นหัวใจสำคัญสำหรับงานทางด้าน IoT (Internet of Things) จุดเด่นของ LPWAN เทคโนโลยีได้แก่

- Secure — Bidirectional Communication
- Simple Star Network Topology (ไม่จำเป็นต้องมี network ที่ซับซ้อนเช่น Mesh หรือ Repeater)
- Low data rate
- Low cost
- Long battery life

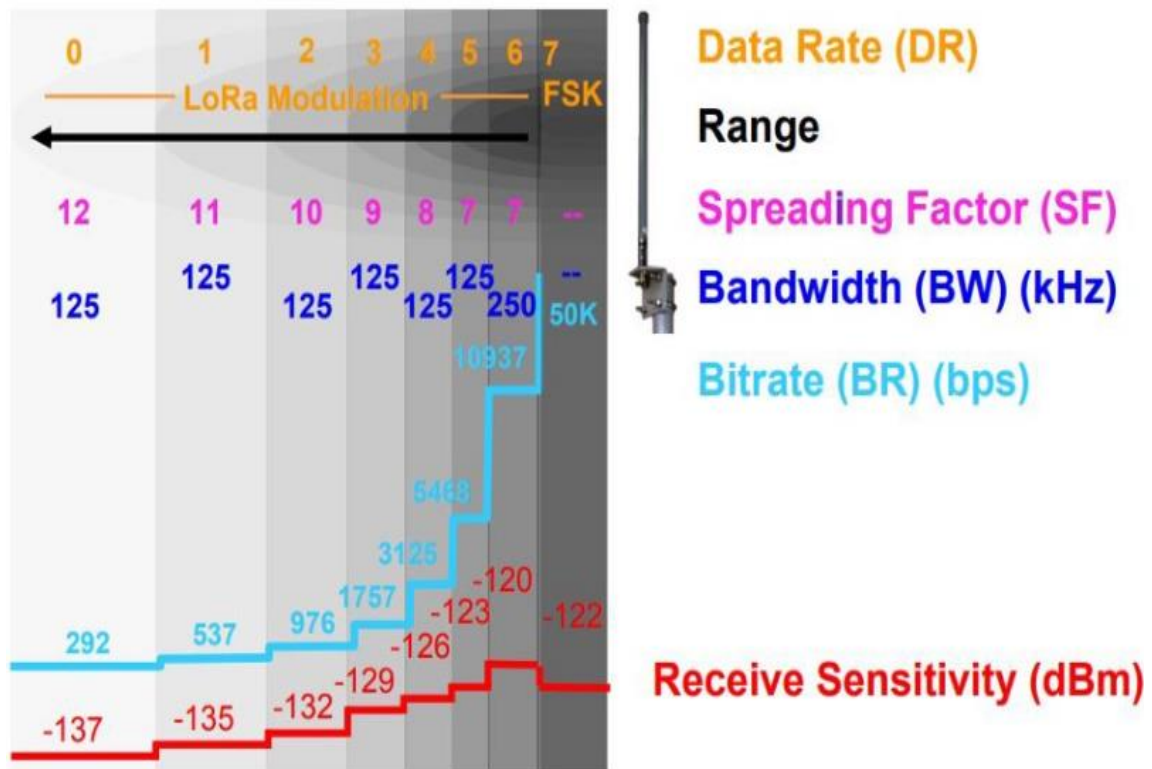
LoRa เป็นเทคโนโลยีการส่งข้อมูลแบบไร้สายในระยะไกล โดยใช้เทคนิค Proprietary Spread Spectrum technology ซึ่งรูปแบบถูกพัฒนาโดย Semtech Corporation ซึ่งมีย่านความถี่ในแต่ละภูมิภาคที่แตกต่างกันดังตารางต่อไปนี้

ตารางที่ 2.1 ความถี่ที่ใช้สำหรับ LoRa (หน่วยเป็น MHz)

ไทย	920-925
ยุโรป	867-869
อเมริกาเหนือ	902-928
จีน	470-510
เกาหลี	920-925
ญี่ปุ่น	920-925
อินเดีย	865-867

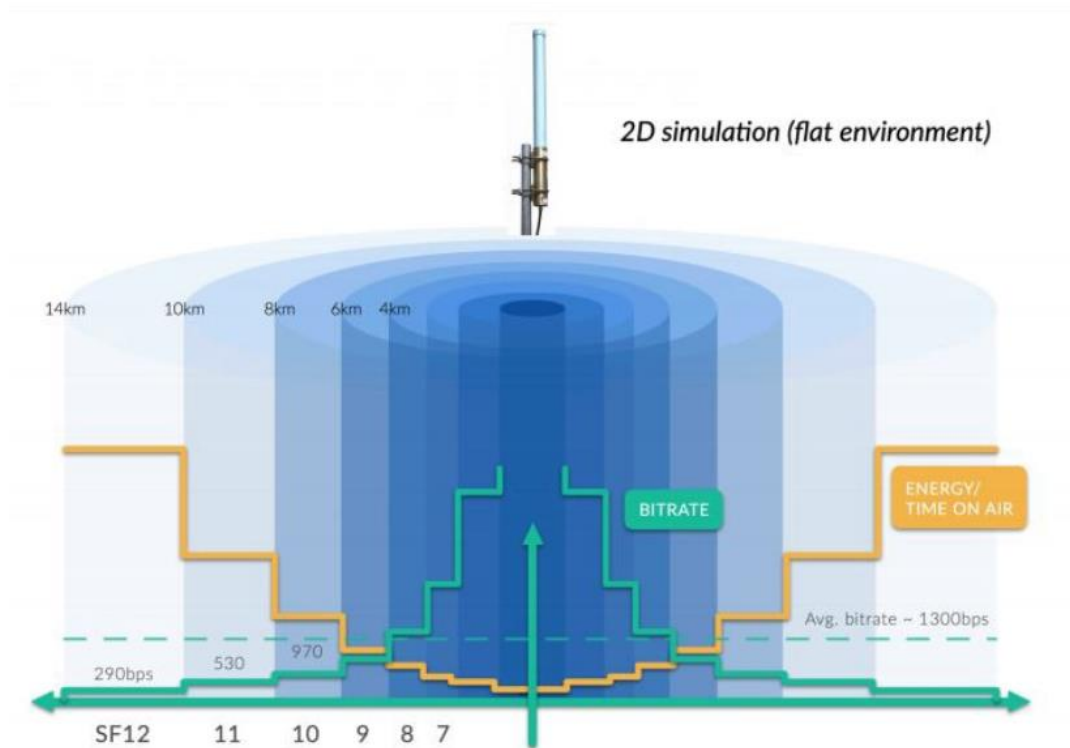
โดยประกาศจาก กสทช. ที่เกี่ยวข้องกับเทคโนโลยี LPWAN ว่าถ้ามีกำลังส่งไม่เกิน 500 มิลลิวัตต์ ได้รับยกเว้นไม่ต้องได้รับใบอนุญาตให้ มี ใช้ และนำออก ซึ่งเครื่องวิทยุคมนาคมและใบอนุญาตให้ตั้งสถานีวิทยุคมนาคม แต่ไม่ได้รับยกเว้น ใบอนุญาตให้ทำนำเข้า และค้าซึ่งเครื่องวิทยุคมนาคม

ถ้ามีกำลังส่งสูงกว่า 500 มิลลิวัตต์จะต้องได้รับใบอนุญาตวิทยุคมนาคมที่เกี่ยวข้อง และการใช้คลื่นความถี่ดังกล่าว จะต้องได้รับอนุญาตให้ใช้คลื่นความถี่ ตามมาตรา ๔๕ แห่งพระราชบัญญัติองค์กรจัดสรรคลื่นความถี่ และ กำกับกับการประกอบกิจการวิทยุกระจายเสียง วิทยุโทรทัศน์ และกิจการโทรคมนาคม พ.ศ. ๒๕๕๓ และจะต้องได้รับ ใบอนุญาตประกอบกิจการโทรคมนาคมแบบที่ สาม



รูปที่ 2.25 แสดงภาพตัวอย่างค่า Data Rate (DR)

จากรูปที่ 2.25 ตัวอย่างค่า Data Rate (DR) สังเกตจากรูป จะเห็นว่า DR เป็น 0 อุปกรณ์จะสามารถส่งข้อมูลได้ไกลที่สุด โดยสามารถส่งด้วย Bitrate ที่ต่ำที่สุดโดยการกำหนด Data Rate จะถูกกำหนดจาก Spreading Factor (SF) ตั้งแต่ 7- 12 โดยที่แบนวิธท์ ช่องสัญญาณ และค่า SF ที่ปรับได้อาจจะเปลี่ยนแปลงตาม Frequency plan ของแต่ละโซน

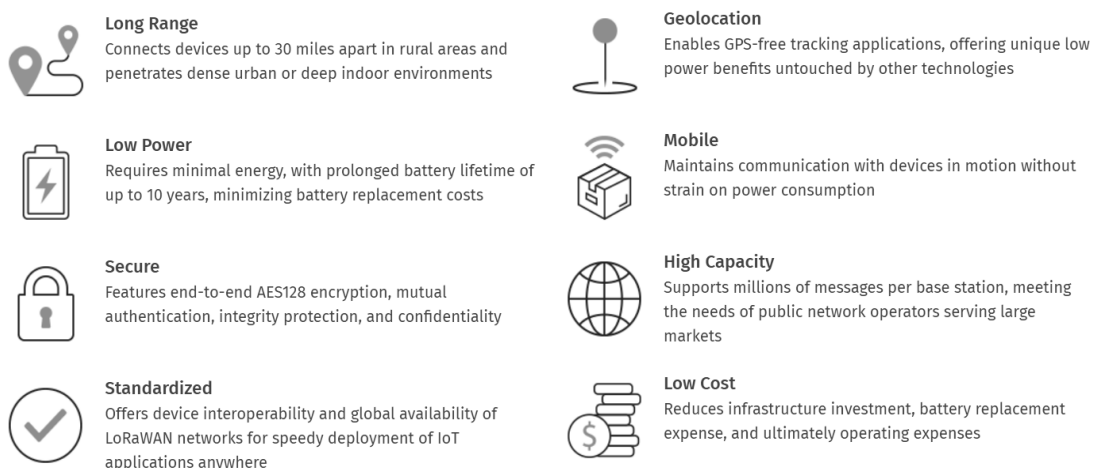


รูปที่ 2.26 แสดงภาพความสัมพันธ์ระหว่าง BITRATE และระยะห่างของอุปกรณ์กับ Gateway

จากรูปที่ 2.26 จะเห็นว่าเมื่ออุปกรณ์เข้าใกล้ gateway มากก็จะสามารถที่จะส่งข้อมูลด้วย BITRATE ที่สูงขึ้นได้ และการส่งข้อมูลจะเร็วขึ้นอีกด้วย รวมถึงพลังงานที่ใช้ในการส่งถ้าเทียบต่อขนาดของแพ็คเกจก็จะน้อยกว่าอุปกรณ์ที่อยู่ไกล Gateway ซึ่งในระดับ LoRaWAN จะมีโหมด ADR (Adaptive Data Rate) ที่เซตในแพ็คเกจการส่งข้อมูลเพื่อให้การเชื่อมต่อระหว่าง Gateway และ Device สามารถปรับ Spreading Factor แบบอัตโนมัติเพื่อประสิทธิภาพในการส่งโดยดูจากระยะการเชื่อมต่อระหว่าง Gateway และ Device โดยสามารถเลือกได้ว่าต้องที่จะ ปรับเพื่อส่งข้อมูลได้เร็วที่สุด หรือ ปรับเพื่อยืดการใช้งานแบตเตอรี่ที่ยาวนานที่สุดเป็นต้น

2.1.8.1 คุณสมบัติหลักของเทคโนโลยี LoRa

Key Features of LoRa Technology



รูปที่ 2.27 แสดงภาพคุณสมบัติหลักของเทคโนโลยี LoRa

จากรูปที่ 2.27 เทคโนโลยี LoRa มีคุณสมบัติต่างๆ ดังต่อไปนี้

- Long Range สามารถเชื่อมต่ออุปกรณ์ที่ห่างกันไม่เกิน 30 ไมล์ในพื้นที่ชนบทและเจาะเข้าไปในสภาพแวดล้อมในเมืองที่หนาแน่นหรือในที่ลึก
- Geolocation มีแอปพลิเคชันการระบุตำแหน่งที่ใช้พลังงานต่ำ
- Low Power ใช้พลังงานน้อยที่สุดพร้อมอายุการใช้งานแบตเตอรี่ที่ยาวนานถึง 10 ปี ลดค่าใช้จ่ายในการเปลี่ยนแบตเตอรี่
- Mobile รักษาการสื่อสารกับอุปกรณ์ที่เคลื่อนไหวโดยไม่ต้องใช้พลังงานมากเกินไป
- Secure มีการเข้ารหัส AES128 แบบ end-to-end การรับรองความถูกต้องซึ่งกันและกัน
- High Capacity รองรับข้อความนับล้านต่อสถานีฐานตอบสนองความต้องการของผู้ให้บริการเครือข่ายสาธารณะที่ให้บริการในตลาดขนาดใหญ่

- Standardized ความสามารถในการทำงานร่วมกันของอุปกรณ์และความพร้อมใช้งานทั่วโลกของเครือข่าย LoRaWAN สำหรับการปรับใช้แอปพลิเคชัน IoT ได้อย่างรวดเร็วทุกที่
- Low Cost ลดการลงทุนโครงสร้างพื้นฐานค่าใช้จ่ายในการเปลี่ยนแบตเตอรี่และค่าใช้จ่ายในการดำเนินงาน

2.1.8.2 ความแตกต่างระหว่าง LoRa และ LoRaWan

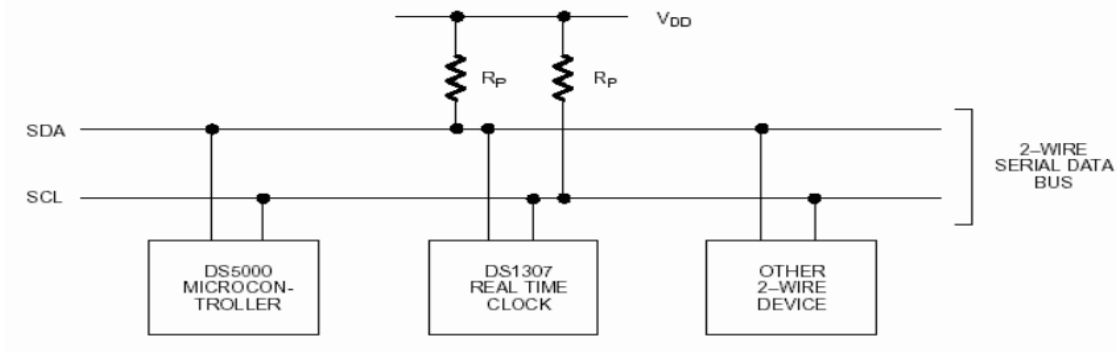
LoRa ใช้งานได้เฉพาะโปรโตคอลระดับ Link layer เหมาะอย่างยิ่งที่จะใช้ในการสื่อสารแบบ P2P (point to point) ระหว่างโหนด

LoRaWan สามารถใช้งานโปรโตคอลระดับ Network layer ทำให้สามารถสื่อสารกับอุปกรณ์อื่นที่เชื่อมต่อกันผ่าน Cloud platform

2.1.9 การเชื่อมต่ออุปกรณ์แบบ I2C (I²C)

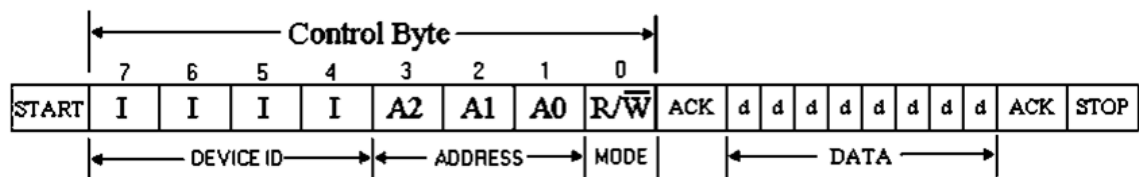
I²C หรือ I2C Bus ย่อมาจาก Inter Integrate Circuit Bus (IIC) นิยมเรียกสั้นๆว่า BUS (ไอ-แอสคว-ซี-บัส) เป็นการสื่อสารอนุกรม แบบซิงโครนัส (Synchronous) เพื่อใช้ ติดต่อสื่อสาร ระหว่าง ไมโครคอนโทรลเลอร์ (MCU) กับอุปกรณ์ภายนอก ซึ่งถูกพัฒนาขึ้นโดยบริษัท Philips Semiconductors โดยใช้สายสัญญาณเพียง 2 เส้นเท่านั้น คือ serial data (SDA) และสาย serial clock (SCL) ซึ่งสามารถเชื่อมต่ออุปกรณ์ จำนวนหลายๆ ตัว เข้าด้วยกันได้ ทำให้ไมโครคอนโทรลเลอร์ใช้พอร์ตเพียง 2 พอร์ตเท่านั้น

I²C BUS ใช้สายสัญญาณ 2 เส้น คือ SCL ,SDA สำหรับติดกับอุปกรณ์แบบ 2 ทิศทาง โดยที่ ขาสัญญาณทั้ง 2 จะต้องต่อกับตัวต้านทานแบบ pull up 2-10K เนื่องจากเอาต์พุตมีลักษณะเป็น แบบ Open Darin หรือเป็นแบบ Open Collector เพื่อให้เอาต์พุตเชื่อมต่อกันได้หลายตัว



รูปที่ 2.28 แสดงลักษณะการการเชื่อมต่ออุปกรณ์แบบ I²C BUS

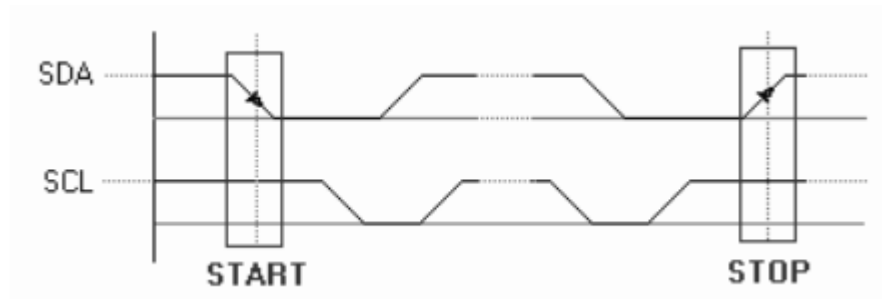
2.1.9.1 การเขียน-อ่านข้อมูลกับอุปกรณ์แบบ I²C BUS



รูปที่ 2.29 แสดงรูปแบบการเขียนและอ่านข้อมูลแบบ I²C BUS

จากรูปที่ 2.29 การรับ-ส่งข้อมูลแบบ I²C BUS ไมโครคอนโทรลเลอร์จะเริ่มต้นการส่งข้อมูลด้วยการส่ง สถานะเริ่มต้น (START Conditions) เพื่อแสดงการขอใช้บัส แล้วตามด้วย รหัสควบคุม (Control Byte) ซึ่งประกอบ ด้วยรหัส ประจำตัวอุปกรณ์ Device ID ,Device Address ,และ Mode ในการเขียนหรืออ่านข้อมูล เมื่ออุปกรณ์ รับทราบว่าไมโครคอนโทรลเลอร์ต้องการ จะติดต่อด้วยก็ต้องส่ง สถานะรับรู้ (Acknowledge) หรือแจ้งให้ไมโครคอนโทรลเลอร์รับรู้ว่าคุณสมบัติที่ส่งมาถูกต้อง และเมื่อสิ้นสุดการส่งข้อมูล MCU จะต้องส่ง สถานะสิ้นสุด (STOP Conditions) เพื่อบอกกับอุปกรณ์ว่าสิ้นสุดการใช้บัส สถานะบัสว่าง คือเมื่อบัสไม่ได้ถูกใช้งาน ทั้ง SCL และ SDA จะเป็น 1 ทั้งคู่

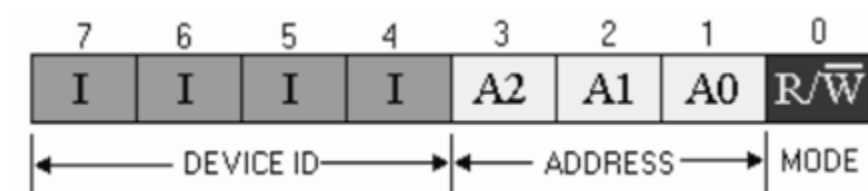
2.1.9.2 การกำหนดสถานะเริ่มต้นและสถานะสิ้นสุดของ I²C BUS (START and STOP Conditions)



รูปที่ 2.30 แสดง I²C BUS START and STOP Conditions

จากรูปที่ 2.30 ลักษณะการกำหนดสถานะเริ่มต้นและสถานะสิ้นสุดของ I²C BUS เมื่อต้องการส่งข้อมูล ไมโครคอนโทรลเลอร์จะต้องส่งสถานะเริ่มต้น (START Conditions) คือให้ SDA เปลี่ยนจาก 1 มาเป็น 0 ในขณะที่ SCL มีค่าเป็น 1 และเมื่อสิ้นสุดการการใช้บัส ไมโครคอนโทรลเลอร์จะต้องส่งสถานะสิ้นสุด (STOP Conditions) คือให้ SDA เปลี่ยนจาก 0 มาเป็น 1 ในขณะที่ SCL มีค่าเป็น 1

2.1.9.3 รหัสควบคุมของ I²C BUS (Control Byte)

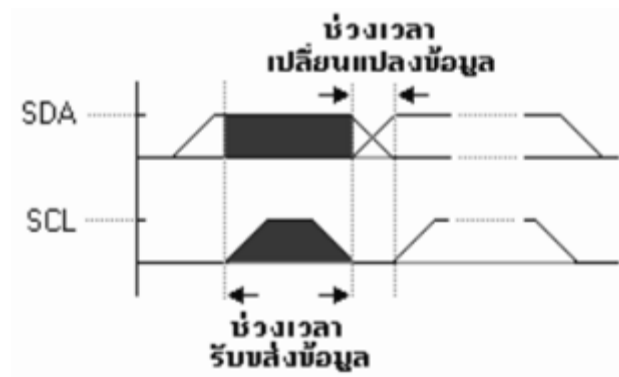


รูปที่ 2.31 แสดงรหัสควบคุมของ I²C BUS (Control Byte)

จากรูปที่ 2.31 รหัสควบคุมของ I²C BUS ประกอบด้วยรหัสประจำตัวของอุปกรณ์ (Device ID) ประกอบด้วยบิต 1-7 และบิต 0 เป็นบิตควบคุมการเขียนอ่าน รหัสประจำตัวของอุปกรณ์ประกอบด้วยรหัสประจำตัวจากผู้ผลิต Product ID 4 บิต (บิต 4-7) ที่เปลี่ยนแปลงแก้ไขไม่ได้ และ

Device Address 3 บิต (บิต 1-3) ซึ่งผู้ใช้ สามารถ กำหนด เองได้ รวมแล้วเป็นรหัส 7 บิต ใช้ระบุตัวอุปกรณ์ ที่ต่ออยู่บนบัส จะมีค่าซ้ำกันไม่ได้ บิตควบคุมการเขียนอ่าน (Mode) บิต 0 เมื่อไมโครคอนโทรลเลอร์ต้องการเขียนข้อมูลไปยังอุปกรณ์ก็กำหนดให้บิตนี้เป็น 0 และเมื่อต้องการ อ่านข้อมูล จากอุปกรณ์ ก็กำหนดให้บิตนี้เป็น 1

2.1.9.4 ช่วงเวลารับส่งบิตข้อมูลของ I²C BUS



รูปที่ 2.32 แสดงช่วงเวลารับส่งบิตข้อมูลของ I²C BUS

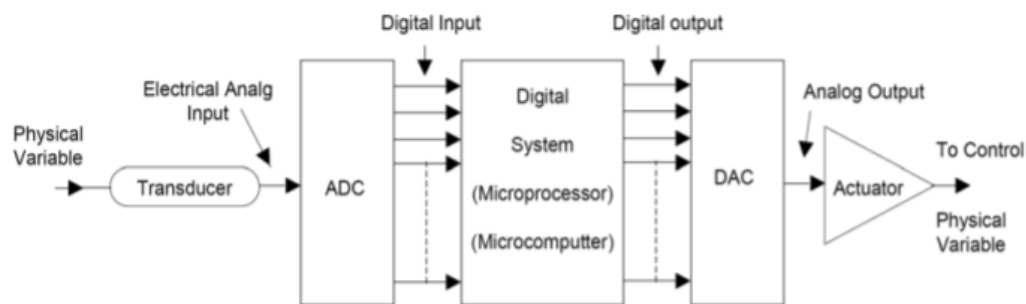
จากรูปที่ 2.32 สภาวะการรับ-ส่งข้อมูล จะกระทำในขณะที่ขา SCL เป็น 1 และสภาวะการเปลี่ยนแปลงข้อมูล จะกระทำในขณะที่ขา SCL เป็น 0

2.1.10 วงจรแปลงสัญญาณอนาล็อกเป็นดิจิตอล (Analog to Digital Converter)

การรับสัญญาณจากตัวตรวจจับ (Sensor) ที่เป็นสัญญาณอนาล็อกในรูปของแรงดันหรือกระแสที่เปลี่ยนแปลง เช่น ตัวตรวจจับอุณหภูมิ ความดัน แสง ฯลฯ มาประมวลผลด้วยตัวประมวลผลที่เป็นวงจร ดิจิตอล ไมโครโปรเซสเซอร์ หรือไมโครคอมพิวเตอร์ จะต้องมีการเปลี่ยนสัญญาณอนาล็อกจากตัวตรวจจับเหล่านั้นให้เป็นสัญญาณดิจิตอล โดยใช้วงจรที่เรียกว่า วงจร Analog to Digital Converter (ADC) ซึ่งวิธีการในการแปลงสัญญาณจากอนาล็อกเป็นดิจิตอลมีหลายวิธีด้วยกันตั้งแต่ใช้วงจรแปลงสัญญาณจากอนาล็อกเป็น Counter Ramp ADC, แบบ Linear Ramp ADC, แบบ Dual Slope ADC

หรือแบบ Successive Approximation ADC ซึ่งวงจรดังกล่าวนี้อาจอยู่ในรูปของวงจรที่ประกอบจากวงจรเปรียบเทียบแรงดัน และวงจรเข้ารหัส หรือวงจรที่ประกอบจากวงจรเปลี่ยนเทียบแรงดันวงจรนับวงจรเปลี่ยนสัญญาณจากดิจิตอลเป็นอนาล็อก รวมทั้งที่สร้างเป็นไอซีสำเร็จรูปที่ใช้สำหรับแปลงสัญญาณอนาล็อกเป็นดิจิตอลโดยเฉพาะ

วงจรแปลงสัญญาณอนาล็อกเป็นดิจิตอลเป็นวงจรที่ทำหน้าที่แปลงสัญญาณอนาล็อกที่อยู่ในรูปของแรงดันหรือกระแสที่เปลี่ยนแปลง ให้เป็นสัญญาณดิจิตอลเพื่อส่งไปยังส่วนประมวลผลที่ใช้วงจรดิจิตอลไมโครโปรเซสเซอร์ หรือไมโครคอมพิวเตอร์



รูปที่ 2.33 แสดงบล็อกไดอะแกรมของระบบควบคุมที่ประกอบด้วยวงจร ADC ตัวประมวลผลและวงจร DAC

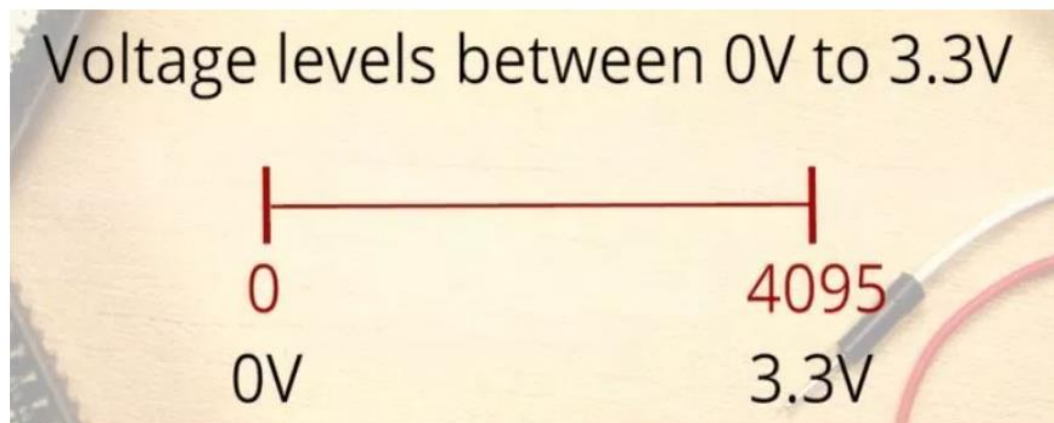
จากรูปที่ 2.33 ทรานสดิวเซอร์จะตรวจจับการเปลี่ยนแปลงทางกายภาพ (Physical Variable) เช่นการเปลี่ยนแปลงของอุณหภูมิ, ความเข้มของแสง, อัตราการไหล, ความดัน และความเร็ว แล้วทำการเปลี่ยนการเปลี่ยนแปลงทางกายภาพเหล่านั้นให้เป็นการเปลี่ยนแปลงของสัญญาณไฟฟ้าในรูปของแรงดันหรือกระแสที่ เป็นสัดส่วนโดยตรงกับการเปลี่ยนแปลงทางกายภาพนั้น ๆ อุปกรณ์ทรานสดิวเซอร์ที่สามารถตรวจจับ สัญญาณดังกล่าวได้แก่ เทอร์มิสเตอร์ (Thermister) ,โฟโต้ไดโอด (Photo Diode) , โฟโต้เซลล์ (Photo cell) , Flow Meter , Pressure Transducer หรือ Tachometer ตามลำดับ หลังจากนั้นจะส่งสัญญาณไฟฟ้าที่ได้ไปยัง วงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิตอล เพื่อทำการแปลงสัญญาณไฟฟ้างกล่าวให้เป็นสัญญาณ ดิจิตอล แล้วส่งต่อไปยังส่วนควบคุมหรือส่วนประมวลผลที่เป็นวงจรดิจิตอลไมโครโปรเซสเซอร์ หรือ ไมโครคอมพิวเตอร์ ซึ่งจะทำหน้าที่ประมวลผล

สัญญาณดิจิทัลที่อินพุต ตามเงื่อนไขของโปรแกรมที่กำหนดไว้ จากนั้นส่งผลลัพธ์ที่ได้จากการประมวลผล ในรูปของสัญญาณดิจิทัลไปที่เอาต์พุตซึ่งมีวงจรแปลงสัญญาณดิจิทัลเป็นอนาล็อกที่ทำการแปลง สัญญาณดิจิทัลให้เป็นอนาล็อกส่งไปยัง Actuator ที่ใช้ควบคุมการเปลี่ยนแปลงทางกายภาพ เช่น ควบคุมการเปิดปิดของวาล์ว ควบคุมการทำงานของ Heater หรือควบคุมทิศทางความเร็ว หรือตำแหน่ง ของมอเตอร์ เป็นต้น

2.1.10.1 การใช้งาน ADC กับไมโครคอนโทรลเลอร์ (ESP32)

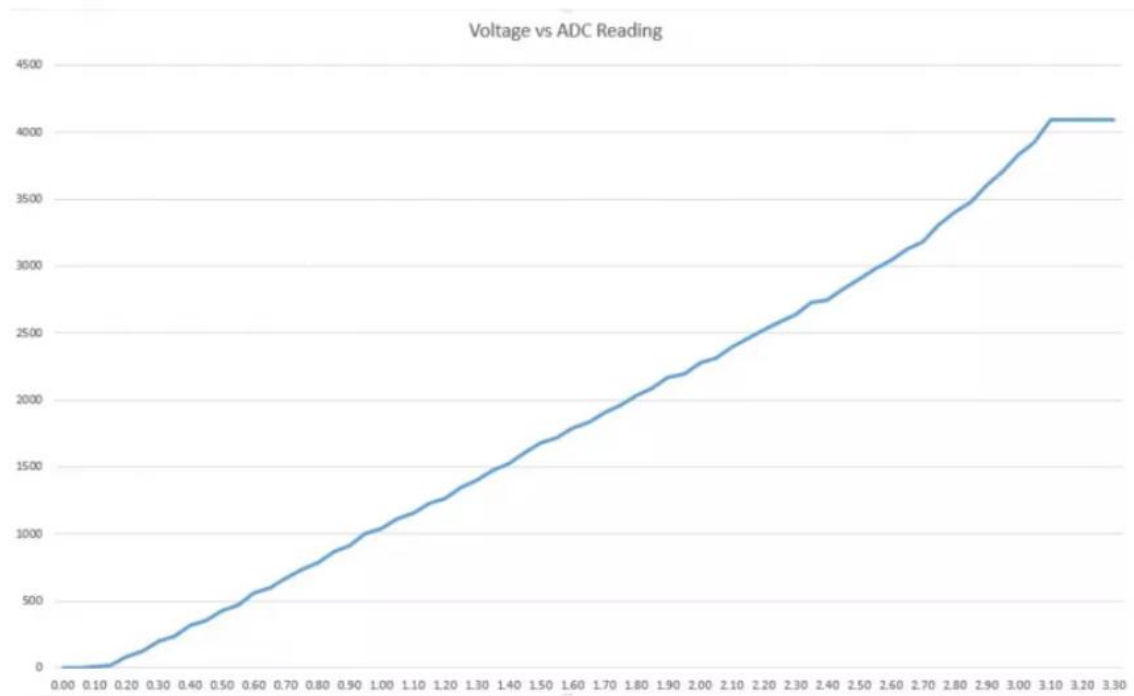
การอ่านค่าอนาล็อกด้วยไมโครคอนโทรลเลอร์ (ESP32) หมายความว่าสามารถอ่านค่า แรงดันระหว่าง 0V ถึง 3.3V ซึ่ง ESP32 สามารถอ่านค่าอนาล็อกด้วยความละเอียดสูงสุดถึง 12-bit

จากนั้นแรงดันไฟฟ้าที่วัดได้จะถูกกำหนดให้เป็นค่าระหว่าง 0 ถึง 4095 โดยที่ 0 V สอดคล้อง กับ 0 และ 3.3 V สอดคล้องกับ 4095 แรงดันไฟฟ้าระหว่าง 0 V ถึง 3.3 V จะได้รับค่าที่สอดคล้องกัน ระหว่าง



รูปที่ 2.34 แสดงภาพความละเอียดและแรงดันที่ ESP32 สามารถอ่านได้

ค่า ADC ไม่ใช่เชิงเส้น (Non-linear) ตามหลักการแล้วเมื่อใช้พิน ESP32 ADC อ่านค่าคาดว่าจะ เป็นแบบเชิงเส้น (Linear) แต่ในการใช้งานจริงค่าที่ได้จะเป็นดังรูปด้านล่าง

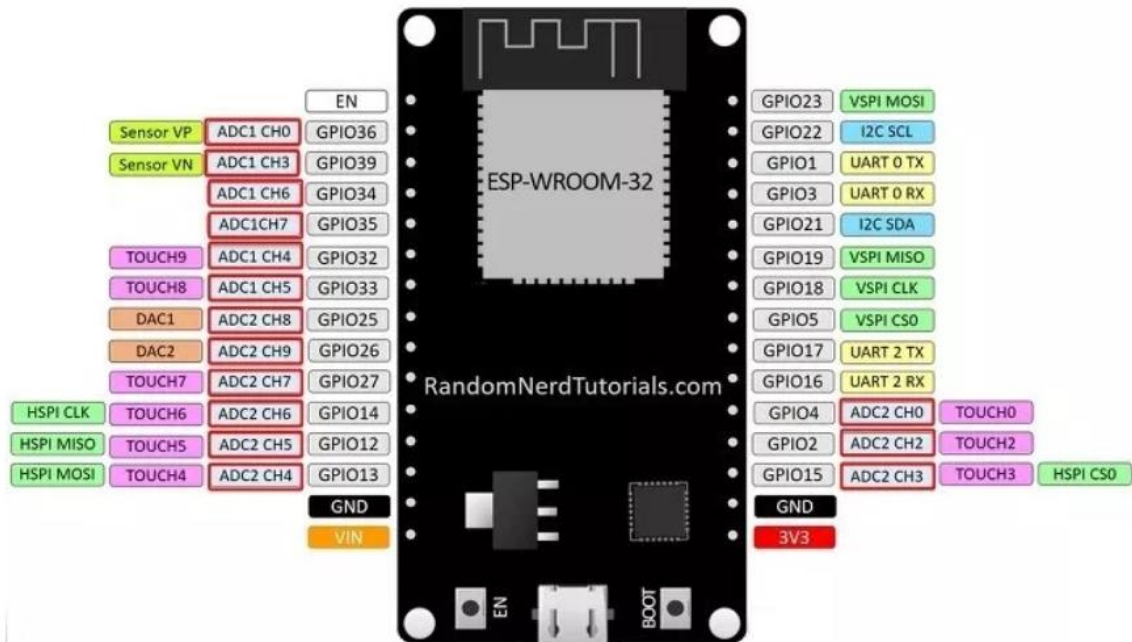


รูปที่ 2.35 แสดงภาพค่าอนาล็อกที่อ่านได้เมื่อใช้ ESP32

จากรูปที่ 2.35 จะเห็นว่า ESP32 จะไม่สามารถแยกความแตกต่างจาก 3.2V ถึง 3.3V กล่าวคือค่าความละเอียดที่ได้จะเป็นค่าเดียวกันคือ 4095 เช่นเดียวกันในระดับแรงดันที่ต่ำมากๆ จาก 0V ถึง 0.1V ค่าความละเอียดที่ได้จะเป็นค่าเดียวกันคือ 0

ฟังก์ชัน `analogRead()` การอ่านอินพุตอะนาล็อกด้วย ESP32 โดยใช้ Arduino IDE ทำได้ง่ายเพียงแค่ใช้ฟังก์ชัน `analogRead()` โดยการกำหนดขา GPIO ที่จะใช้อ่านค่าอนาล็อก `analogRead(GPIO)`; ESP32 รองรับการวัดใน 18 ช่องสัญญาณ บอร์ด DEVKIT V1 DOIT มีเพียง 15 ตัวเท่านั้น (เวอร์ชันที่มี 30 GPIOs)

ESP32 DEVKIT V1 - DOIT



รูปที่ 2.36 แสดงภาพ ESP32 Pinout Reference

ไม่สามารถใช้พิน ADC2 เมื่อใช้ Wi-Fi ดังนั้นหากใช้ Wi-Fi และมีปัญหาในการรับค่าจาก ADC2 GPIO อาจพิจารณาใช้ ADC1 GPIO แทนซึ่งจะช่วยแก้ปัญหาได้

ฟังก์ชันที่มีประโยชน์อื่นๆ ฟังก์ชันขั้นสูงสำหรับใช้กับพิน ADC ที่สามารถเป็นประโยชน์ในการทำงาน ESP32 ADC

- `analogReadResolution(resolution)` การกำหนดความละเอียดในการอ่านค่าอนาล็อก สามารถกำหนดความละเอียดได้ตั้งแต่ 9-bit (0-511) ถึง 12-bit ค่าความละเอียดเริ่มต้นคือ 12-bit
- `analogSetWidth(width)` มีวัตถุประสงค์การใช้งานเหมือน ฟังก์ชัน `analogReadResolution(resolution)`
- `analogSetCycles(cycles)` กำหนดจำนวนรอบต่อตัวอย่าง (sample) ค่าเริ่มต้นคือ 8 (0-255)

- analogSetSamples(samples) กำหนดจำนวนตัวอย่าง (samples) ในช่วง ค่าเริ่มต้นคือ 1 ตัวอย่าง (sample) มีผลในการเพิ่มความไว (sensitivity)
- analogSetClockDiv(attenuation) ตั้งค่าตัวแบ่งสำหรับสัญญาณนาฬิกา ADC ค่าเริ่มต้นคือ 1 ช่วง (1-255)
- analogSetAttenuation(attenuation) ตั้งค่าการลดทอนอินพุตสำหรับพิน ADC ทั้งหมด ค่าเริ่มต้นคือ ADC_11db การตั้งค่า ADC_0db คือไม่มีการลดทอนอินพุต (1V input = ADC reading of 1088) การตั้งค่า ADC_2_5db ตั้งค่าการลดทอนเป็น 1.34 (1V input = ADC reading of 2086) การตั้งค่า ADC_6db ตั้งค่าการลดทอน 1.5 (1V input = ADC reading of 2975) การตั้งค่า ADC_11db sets an attenuation of 3.6 (1V input = ADC reading of 3959)
- analogSetPinAttenuation(pin, attenuation) ตั้งค่าการลดทอนอินพุตสำหรับพินที่ระบุ ค่าเริ่มต้นคือ ADC_11db ค่าการลดทอนจะเหมือนกันจากฟังก์ชันก่อนหน้านี้
- adcAttachPin(pin) แแนบ (Attach) พินเข้ากับ ADC (ล้าวมดอะนาล็อกอื่น ๆ ที่อาจเปิดอยู่) ส่งคืนผลลัพธ์เป็นค่า TRUE หรือ FALSE
- adcStart(pin), adcBusy(pin) and resultadcEnd(pin) เริ่มการแปลง ADC บน บัส (bus) ของพินที่แนบ (Attach) มาตรวจสอบว่าการแปลงบนบัส ADC ของพิน กำลังทำงานอยู่หรือไม่ (ส่งกลับค่า TRUE หรือ FALSE) รับผลลัพธ์ของการแปลง: ส่งกลับจำนวนเต็ม 16 บิต

2.1.11 อุณหภูมิ (Temperature)

คือการวัดค่าเฉลี่ยของพลังงานจลน์ของอนุภาคในสารใด ๆ ซึ่งสอดคล้องกับความร้อนหรือ เย็นของสารนั้น ในอดีตมีแนวคิดเกี่ยวกับอุณหภูมิเกิดขึ้นเป็น 2 แนวทาง คือตามแนวทางของหลักอุณหพลศาสตร์ และตามการอธิบายเชิงจุลภาคทางฟิสิกส์เชิงสถิติ แนวคิดทางอุณหพลศาสตร์นั้น ถูกพัฒนาขึ้นโดยลอร์ดเคลวิน โดยเกี่ยวข้องกับการวัดในเชิงมหภาค ดังนั้นคำจำกัดความอุณหภูมิในเชิงอุณหพลศาสตร์ในเบื้องต้น จึงระบุเกี่ยวกับค่าตัวแปรต่าง ๆ ที่สามารถตรวจวัดได้จากการสังเกต ส่วนแนวทางของฟิสิกส์เชิงสถิติจะให้ความสนใจในเชิงลึกยิ่งกว่าอุณหพลศาสตร์ โดยอธิบายถึงการสะสมจำนวนอนุภาคขนาดใหญ่ และตีความพารามิเตอร์ต่าง ๆ ในอุณหพลศาสตร์ (เชิงมหภาค) ในฐานะค่าเฉลี่ยทางสถิติของพารามิเตอร์ของอนุภาคในเชิงจุลภาค

ในการศึกษาฟิสิกส์เชิงสถิติ สามารถตีความค่านิยามอุณหภูมิในอุณหพลศาสตร์ว่า เป็นการวัดพลังงานเฉลี่ยของอนุภาคในแต่ละองศาอิสระในระบบอุณหพลศาสตร์ โดยที่อุณหภูมินั้นสามารถมองเป็นคุณสมบัติเชิงสถิติ ดังนั้นระบบจึงต้องประกอบด้วยปริมาณอนุภาคจำนวนมากเพื่อจะสามารถบ่งบอกค่าอุณหภูมิอันมีความหมายที่นำไปใช้ประโยชน์ได้ ในของแข็ง พลังงานนี้พบในการสั่นไหวของอะตอมของสสารในสภาวะสมดุล ในแก๊สอุดมคติ พลังงานนี้พบในการเคลื่อนไหวไปมาของอนุภาคโมเลกุลของแก๊ส

2.1.11.1 ความร้อนและอุณหภูมิ

สสารทั้งหลายประกอบด้วย อะตอมรวมตัวกันเป็นโมเลกุล การเคลื่อนที่ของอะตอม หรือการสั่นของโมเลกุล ทำให้เกิดรูปแบบของพลังงานจลน์ ซึ่งเรียกว่า “ความร้อน” (Heat) เราพิจารณาพลังงานความร้อน (Heat energy) จากพลังงานทั้งหมดที่เกิดขึ้นจากการเคลื่อนที่ของอะตอมหรือโมเลกุลทั้งหมดของสสาร

อุณหภูมิ (Temperature) หมายถึง การวัดค่าเฉลี่ยของพลังงานจลน์ซึ่งเกิดขึ้นจากอะตอมแต่ละตัว หรือแต่ละโมเลกุลของสสาร เมื่อเราใส่พลังงานความร้อนให้กับสสาร อะตอมของมันจะเคลื่อนที่เร็วขึ้น ทำให้อุณหภูมิสูงขึ้น แต่เมื่อเราลดพลังงานความร้อน อะตอมของสสารจะเคลื่อนที่ช้าลง ทำให้อุณหภูมิลดต่ำลง

หากเราต้มน้ำด้วยถ้วยและหม้อบนเตาเดียวกัน จะเห็นได้ว่าน้ำในถ้วยจะมีอุณหภูมิสูงกว่า แต่จะมีพลังงานความร้อนน้อยกว่าในหม้อ เนื่องจากปริมาณความร้อนขึ้นอยู่กับมวลทั้งหมดของสสาร แต่อุณหภูมิเป็นเพียงค่าเฉลี่ยของพลังงานในแต่ละอะตอม ดังนั้นบรรยากาศชั้นบนของโลก (ชั้นเทอร์โมสเฟียร์) จึงมีอุณหภูมิสูง แต่มีพลังงานความร้อนน้อย เนื่องจากมีมวลอากาศอยู่อย่างเบาบาง

2.1.11.2 หน่วยของอุณหภูมิและการแปลงหน่วย

การแปลงหน่วยอุณหภูมิ คือการแปลงข้อมูลอุณหภูมิในหน่วยอุณหภูมิหนึ่ง ให้กลายเป็นอีกหน่วยอุณหภูมิหนึ่ง อาทิ เช่น 0 องศาเซลเซียส เป็น 32 องศาฟาเรนไฮต์ ซึ่งต้องใช้สูตรทางคณิตศาสตร์ในการแก้

องศาเซลเซียส (อังกฤษ: Celcius, สัญลักษณ์: °C) ตั้งตามชื่อของนักดาราศาสตร์ชาวสวีเดน นายแอนเดอร์ เซลเซียส (Anders Celsius) เป็นคนแรกที่เสนอระบบที่ใกล้เคียงกับระบบนี้ (องศาเซนติเกรด) นี้ ในปี พ.ศ. 2285 (ค.ศ. 1742) กำหนดอุณหภูมิจุดเยือกแข็งของน้ำคือ 0 องศา และจุดเดือดของน้ำคือ 100 องศาเซลเซียส ที่ระดับความดันบรรยากาศมาตรฐาน ในปัจจุบันองศาเซลเซียสใช้กับ

แพร่หลายทั่วโลกในชีวิตประจำวัน ยกเว้นประเทศสหรัฐอเมริกา และ ประเทศไลบีเรีย รวมถึงประเทศที่ใช้บริการทางอุตุนิยมวิทยา และ ดินแดนโพ้นทะเลของสหราชอาณาจักรฯ เท่านั้นที่นิยมใช้หน่วยองศาฟาเรนไฮต์ แต่องศาเซลเซียสใช้ในวงการวิทยาศาสตร์ทั่วโลกอย่างเป็นสากล

การแปลงหน่วยอุณหภูมิจากองศาฟาเรนไฮต์ (Fahrenheit) เป็นองศาเซลเซียส (Celcius)

จากสมการ

$$^{\circ}\text{F} = ^{\circ}\text{C} \times 9/5 + 32$$

จะได้

$$^{\circ}\text{C} = (^{\circ}\text{F} - 32) \times 5/9$$

การแปลงหน่วยอุณหภูมิจากเคลวิน (kelvin) เป็นองศาเซลเซียส (Celcius)

จากสมการ

$$\text{K} = ^{\circ}\text{C} + 273.15$$

จะได้

$$^{\circ}\text{C} = \text{K} - 273.15$$

องศาฟาเรนไฮต์ (อังกฤษ:Fahrenheit, สัญลักษณ์: $^{\circ}\text{F}$) คือหน่วยมาตรวัดอุณหภูมิชนิดหนึ่ง ตั้งชื่อตามนักฟิสิกส์ชาวเยอรมัน กาเบรียล ฟาเรนไฮต์ (1686-1736) ผู้เสนอระบบมาตรวัดอุณหภูมินี้เมื่อปี ค.ศ.1724 โดยที่ค่าสเกลองศาฟาเรนไฮต์มีจุดอ้างอิงต่ำสุด 0°F เป็นอุณหภูมิจุดเยือกแข็งของ น้ำ น้ำแข็ง และ กลี้อแอมโมเนียมคลอไรด์ องศาฟาเรนไฮต์เป็นมาตรวัดอุณหภูมิแรกที่มีการใช้งานกันอย่างแพร่หลาย ในปัจจุบัน องศาฟาเรนไฮต์ใช้จุดแข็งและจุดเดือดของน้ำเป็นจุดอ้างอิง โดยมีจุดเยือกแข็งอยู่ที่ 32°F และ มีจุดเดือดที่ 212°F โดยมีระยะห่างระหว่างจุดเยือกแข็งกับจุดเดือดของน้ำคือ 180 องศา ความดันบรรยากาศมาตรฐาน ณ ระดับน้ำทะเล ฟาเรนไฮต์เป็นมาตรวัดอุณหภูมิอย่างเป็นทางการในสหรัฐอเมริกา และประเทศที่ใกล้ชิดกับสหรัฐอเมริกาในมหาสมุทรแปซิฟิกตะวันตก ดินแดนหมู่เกาะที่เป็นดินแดนโพ้นทะเลของสหราชอาณาจักร และ ประเทศไลบีเรีย

การแปลงหน่วยอุณหภูมิจากเคลวิน (kelvin) เป็นองศาฟาเรนไฮต์ (Fahrenheit)

จากสมการ

$$K = ({}^{\circ}\text{F} + 459.67) \times 5/9$$

จะได้

$${}^{\circ}\text{F} = K \times 9/5 - 459.67$$

การแปลงหน่วยอุณหภูมิจากองศาเซลเซียส (Celcius) เป็นองศาฟาเรนไฮต์ (Fahrenheit)

จากสมการ

$${}^{\circ}\text{C} = ({}^{\circ}\text{F} - 32) \times 5/9$$

จะได้

$${}^{\circ}\text{F} = {}^{\circ}\text{C} \times 9/5 + 32$$

เคลวิน (อังกฤษ: kelvin, สัญลักษณ์: K) เป็นหน่วยวัดอุณหภูมิหนึ่ง และเป็นหน่วยพื้นฐานหนึ่งในเจ็ดของระบบเอสไอ นิยามให้เท่ากับ 1/273.16 เท่าของอุณหภูมิเทอร์โมไดนามิกของจุดสามสถานะของน้ำ เคลวินตั้งชื่อเพื่อเป็นเกียรติแก่นักฟิสิกส์และวิศวกรชาวอังกฤษ วิลเลียม ทอมสัน บารอนที่หนึ่งแห่ง เคลวิน (William Thomson, 1st Baron Kelvin) เคลวิน เป็นหน่วยของหน่วยวัดอุณหภูมิหนึ่งที่ลอร์ดเคลวิน ได้พัฒนาคิดสเกลขึ้นใหม่ โดยหาความสัมพันธ์ระหว่างอุณหภูมิและความเร็วของอิเล็กตรอนที่เคลื่อนที่รอบนิวเคลียส โดยสังเกตว่าถ้าให้ความร้อนกับสสารมากขึ้น อิเล็กตรอนจะมีพลังงานมากขึ้น ทำให้เคลื่อนที่มีความเร็วมากขึ้น ในทางกลับกันถ้าลดความร้อนให้กับสสาร อิเล็กตรอนก็จะมีพลังงานน้อยลง ทำให้การเคลื่อนที่ลดลง และถ้าสามารถลดอุณหภูมิลงจนถึงจุดที่อิเล็กตรอนหยุดการเคลื่อนที่ ณ จุดนั้น จะไม่มีอุณหภูมิหรือพลังงานในสสารเลย และจะไม่มีแผ่รังสีความร้อนจากวัตถุ จึงเรียกอุณหภูมิ ณ จุดนี้ว่า ศูนย์สัมบูรณ์ (0 K) อุณหภูมิเพิ่ม 1 K มีค่าเท่ากับ 1°C

การแปลงหน่วยอุณหภูมิจากองศาเซลเซียส (Celcius) เป็นเคลวิน (kelvin)

จากสมการ

$$^{\circ}\text{C} = \text{K} - 273.15$$

จะได้

$$\text{K} = ^{\circ}\text{C} + 273.15$$

การแปลงหน่วยอุณหภูมิจากองศาฟาเรนไฮต์ (Fahrenheit) เป็นเคลวิน (kelvin)

จากสมการ

$$^{\circ}\text{F} = \text{K} \times 9/5 - 459.67$$

จะได้

$$\text{K} = (^{\circ}\text{F} + 459.67) \times 5/9$$

2.1.12 ความชื้น (Humidity)

ความชื้น (Humidity) หมายถึง ปริมาณไอน้ำที่มีอยู่ในอากาศ เมื่อน้ำได้รับความร้อนจะเปลี่ยนสถานะจากของเหลวกลายเป็นไอลอยขึ้นในอากาศ เรียกว่า การระเหย ซึ่งความร้อนที่ใช้ในการทำให้น้ำระเหยกลายเป็นไอ เรียกว่า ความร้อนแฝง (Latent Heat) เมื่ออากาศเย็นลงไอน้ำจะเริ่มกลั่นตัวเป็นละอองและคายความร้อนแฝงออกมาด้วย กาศจะได้รับไอน้ำมากหรือน้อยขึ้นอยู่กับอุณหภูมิ ดังนั้นอุณหภูมิจึงเป็นตัวกำหนดปริมาณไอน้ำในอากาศ อากาศที่มีอุณหภูมิสูงจะรับไอน้ำได้มากกว่าอากาศที่มีอุณหภูมิต่ำ ถ้าอากาศไม่สามารถรับไอน้ำได้เรียกว่า ไอน้ำอิ่มตัว (Saturate)

2.1.12.1 ความชื้นในอากาศ

ความชื้นในอากาศสามารถแบ่งออกเป็น 4 ประเภท คือ

1) ความชื้นสัมบูรณ์ (Absolute Humidity) เป็นน้ำหนักของไอน้ำที่มีอยู่จริงในปริมาตรของอากาศจำนวนหนึ่งคำนวณได้จากน้ำหนักของไอน้ำต่อหนึ่งหน่วยปริมาตรของอากาศ หรือกล่าวอีกนัยหนึ่งว่าความชื้นสัมบูรณ์คือความหนาแน่นของไอน้ำในอากาศ หน่วยที่ใช้มักเป็นกรัมต่อลูกบาศก์เมตรความชื้นสัมบูรณ์ไม่นิยมใช้ในทางอุตุนิยมวิทยาเพราะเมื่ออากาศลอยตัวขึ้นหรือจมตัวลงจะทำให้ปริมาตรของ

อากาศเปลี่ยนแปลงเนื่องจากบริเวณรอบๆ ความกดอากาศจะเปลี่ยนแปลงแม้ว่าไอน้ำที่มีอยู่ในอากาศจะคงที่

2) ความชื้นจำเพาะ (Specific Humidity) คือน้ำหนักหรือความชื้นที่มีอยู่ในอากาศ (Q) เป็นอัตราส่วนระหว่างน้ำหนักไอน้ำ (Mv) ต่อน้ำหนักของอากาศชื้น น้ำหนักของไอน้ำ (Mv) รวมกับน้ำหนักของอากาศ (Ma) มักใช้เป็นกรัมของน้ำต่อ 1 กิโลกรัมของอากาศชื้น ดังสมการ

$$Q = Mv / (Mv + Ma)$$

ความชื้นจำเพาะของอากาศจะมีค่าคงที่ เมื่ออากาศขยายตัวหรือหดตัว โดยที่ความชื้นจะไม่เปลี่ยนแปลง แม้ว่าปริมาตรของอากาศจะขยายตัวหรือหดตัวก็ตาม

3) ความชื้นสัมพัทธ์ (Relative Humidity) คืออัตราส่วนระหว่างน้ำหนักของไอน้ำที่มีอยู่จริงที่อุณหภูมิและความกดดันหนึ่งต่อน้ำหนักของไอน้ำอิ่มตัวที่อุณหภูมิและความกดดันนั้น คิดเป็นค่าร้อยละ ตัวอย่างเช่น อากาศ 1 ลูกบาศก์เมตร เมื่ออุณหภูมิ 30 องศาเซลเซียสมีไอน้ำอยู่ 9 กรัม และในอุณหภูมินั้นอากาศอิ่มตัวมีไอน้ำอยู่ 30 กรัม ความชื้นสัมพัทธ์เท่ากับ $(100 \times 9) / 30$ เท่ากับ 30% ความชื้นสัมพัทธ์เป็นวิธีวัดความชื้นในอากาศที่ใช้มากที่สุด การเปลี่ยนแปลงความชื้นสัมพัทธ์จะไม่ทำให้ปริมาณไอน้ำที่มีอยู่ในอากาศเปลี่ยนแปลง แต่อุณหภูมิจะเปลี่ยนแปลง และถ้าอุณหภูมิเปลี่ยนแปลงความชื้นสัมพัทธ์จะเปลี่ยนแปลงด้วย

4) อุณหภูมิจุดน้ำค้าง (Dew Point Temperature) คืออุณหภูมิซึ่งอากาศถูกทำให้เย็นลง (ความกดอากาศคงที่) ถึงอุณหภูมิหนึ่งที่ไอน้ำจุดอิ่มตัวพอดี อุณหภูมิของจุดน้ำค้างจะเป็นเท่าใดก็ได้ขึ้นอยู่กับจำนวนไอน้ำที่มีอยู่จริงในอากาศ ถ้าอากาศมีไอน้ำมากอุณหภูมิของจุดน้ำค้างจะสูง แต่ถ้าไอน้ำมีน้อยอุณหภูมิของจุดน้ำค้างจะต่ำ ถ้าอุณหภูมิของอากาศลดต่ำกว่าจุดน้ำค้างจะมีการกลั่นตัวในรูปของหยดน้ำ เช่น ในฤดูร้อนแก้วน้ำที่ใส่น้ำแข็งตั้งทิ้งไว้ความชื้นของอากาศจะรวมกันเป็นหยดน้ำเกาะอยู่รอบนอกแก้วน้ำ เนื่องจากอุณหภูมิของแก้วน้ำที่ใส่น้ำแข็งจะต่ำกว่าอุณหภูมิจุดน้ำค้างที่อยู่โดยรอบ อุณหภูมิของจุดน้ำค้างจะบอกถึงความสะดวกสบายของมนุษย์ในช่วงที่มีอากาศอุ่นและชื้นได้ดีกว่าความชื้นสัมพัทธ์ ยกเว้นผู้ที่เคยชินกับอากาศร้อนชื้น คนส่วนใหญ่รู้สึกที่อากาศชื้นไม่สะดวกสบายเมื่ออุณหภูมิจุดน้ำค้างสูงกว่า 17 องศาเซลเซียส ในขณะที่บางคนอาจไม่รู้สึกละสบายตัว เมื่ออุณหภูมิจุดน้ำค้างสูงกว่า 20 องศาเซลเซียส

2.1.12.2 ความชื้นสัมพัทธ์ของประเทศไทยในแต่ละฤดู

ประเทศไทยตั้งอยู่ในเขตร้อนใกล้เส้นศูนย์สูตร จึงมีอากาศร้อนชื้นปกคลุมเกือบตลอดปี เว้นแต่บริเวณที่ลึกเข้าไปในแผ่นดิน ตั้งแต่ภาคกลางขึ้นไป ความชื้นสัมพัทธ์จะลดลงชัดเจน ในช่วงฤดูหนาวและฤดูร้อน โดยเฉพาะฤดูร้อนจะเป็นช่วงที่มีความชื้นสัมพัทธ์ลดลงต่ำที่สุดในรอบปี ในบริเวณดังกล่าวมีความชื้นสัมพัทธ์เฉลี่ยตลอดปีที่ 72-74 เปอร์เซ็นต์และจะลดลงเหลือ 62-69 เปอร์เซ็นต์ ในช่วงฤดูร้อน ดังแสดงข้อมูลตามตารางที่ 2.ป

ตารางที่ 2.2 สถิติความชื้นสัมพัทธ์เฉลี่ย (%) ของประเทศไทยในช่วงฤดูกาลต่างๆ

ภาค	ฤดูหนาว	ฤดูร้อน	ฤดูฝน	ตลอดปี
เหนือ	73	62	81	74
ตะวันออกเฉียงเหนือ	69	65	80	72
กลาง	71	69	79	73
ตะวันออก	71	74	81	76
ใต้ฝั่งตะวันออก	81	77	78	79
ใต้ฝั่งตะวันตก	77	76	84	80

2.1.13 ตัวต้านทานตรวจสอบกระแส (Current Sense Resistors)

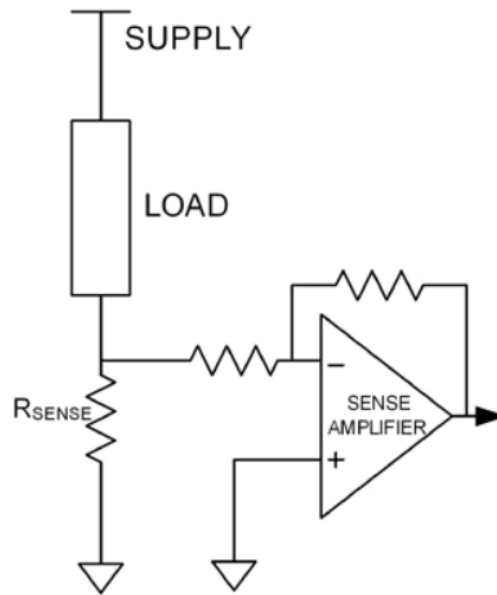
การตรวจสอบกระแสโดยการวัดแรงดันตกคร่อมตัวต้านทานเป็นวิธีที่สามารถทำได้ง่าย แต่ต้องพิจารณาปัจจัยต่างๆก่อนนำมาใช้งาน เช่น การเลือกค่าความต้านทาน ขนาดของตัวต้านทาน ความร้อน การวัดการไหลของกระแสมีพารามิเตอร์ที่สำคัญที่ต้องจัดการเพื่อให้ระบบทำงานอย่างมีประสิทธิภาพ วิธีที่ใช้กันทั่วไปในการตรวจสอบการไหลของกระแสทำได้โดยการนำตัวต้านทานที่มีค่าความต้านทานต่ำ ต่ออนุกรมกับโหลดแล้ววัดแรงดันตกคร่อมตัวต้านทานโดยใช้กฎของโอห์ม

$$V = IR$$

การใช้ตัวต้านทานในการตรวจสอบกระแสเป็นวิธีพื้นฐานที่มีประสิทธิภาพ แต่ยังมีปัญหาด้านการออกแบบรายละเอียดปลีกย่อยอีกมากมายที่ต้องคำนึงถึง เช่น ตำแหน่งของตัวต้านทานตรวจสอบกระแส ค่าความต้านทาน ขนาดของตัวต้านทาน และข้อพิจารณาเชิงกลอีกหลายประการ

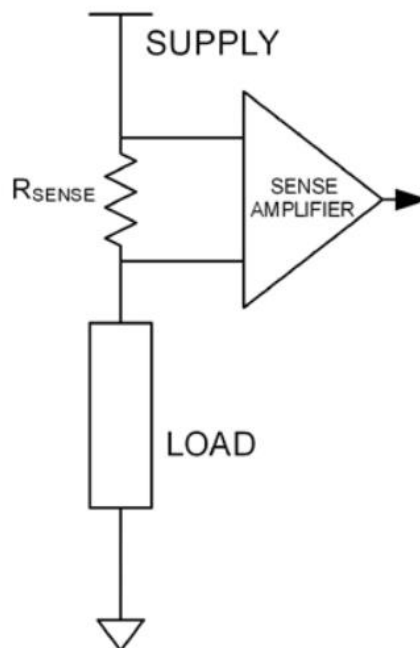
2.1.13.1 ตำแหน่งการวางตัวด้านทานตรวจสอบกระแส

ตำแหน่งการวางตัวด้านทานตรวจสอบกระแสมีการวางอยู่ 2 แบบคือ การตรวจจับด้านต่ำ (Low-side Sensing) และ การตรวจจับด้านสูง (High-side Sensing) ในการตรวจจับด้านต่ำจะวางตัวด้านทานตรวจสอบกระแสระหว่างโหลด (Load) และ กราวด์ (Ground) ดังรูปที่ 2.ป ซึ่งช่วยให้สามารถใช้วงจรขยายสัญญาณวัดแรงดันตกคร่อมตัวด้านทานตรวจสอบกระแส (Current Sense Resistors) และ กราวด์ (Ground) ได้ง่าย



รูปที่ 2.37 แสดงภาพการวางตัวด้านทานตรวจสอบกระแสแบบการตรวจจับด้านต่ำ (Low-side Sensing)

แบบที่ 2 เป็นการตรวจจับด้านสูง (High-side Sensing) โดยการวางตัวด้านทานตรวจสอบกระแส (Current Sense Resistors) ระหว่างแหล่งจ่าย (Power) และ โหลด (Load) ดังรูปที่ 2.ป การตรวจจับแบบนี้จะใช้วงจรขยายสัญญาณแบบ differential amplifier หรือ instrumentation amplifier ในการวัดแรงดันตกคร่อมตัวด้านทานตรวจสอบกระแส (Current Sense Resistors)



รูปที่ 2.38 แสดงภาพการวางตัวต้านทานตรวจสอบกระแสแบบการตรวจจับด้านสูง
(High-side Sensing)

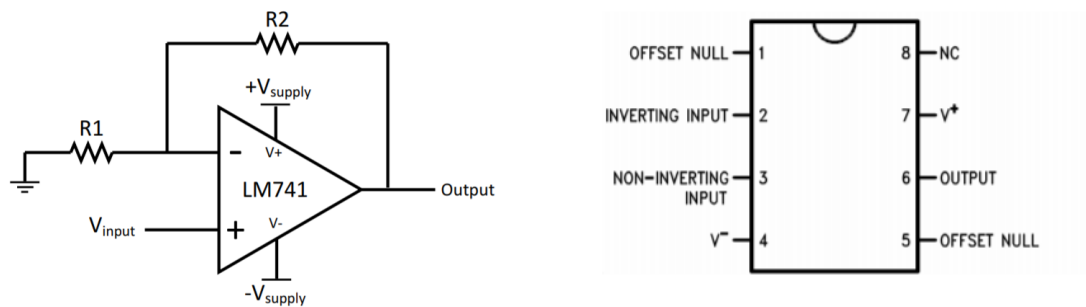
2.1.13.2 การเลือกตัวต้านทานตรวจสอบกระแส

ในทางปฏิบัติโดยทั่วไปการเลือกตัวต้านทานตรวจสอบกระแส (Current Sense Resistors) เมื่อนำตัวต้านทานไปต่อกับโหลด (Load) แรงดันตกคร่อมตัวต้านทานตรวจสอบกระแส (Current Sense Resistors) ไม่ควรเกิน 100mV หรือต่ำกว่า เพื่อไม่ให้แบ่งแรงดันจากโหลดมากเกินไป ค่าความต้านทานควรมีค่าประมาณ 10-m Ω หรือต่ำกว่าจากรูปที่ 2.ป แสดงภาพตัวอย่างของตัวต้านทานตรวจสอบกระแส (Current Sense Resistors) ที่มีค่าความต้านทาน 0.2-m Ω



รูปที่ 2.39 แสดงภาพตัวต้านทานตรวจสอบกระแสขนาด 0.2-mΩ

2.1.14 วงจรออปแอมป์ (Op-Amp Circuit)

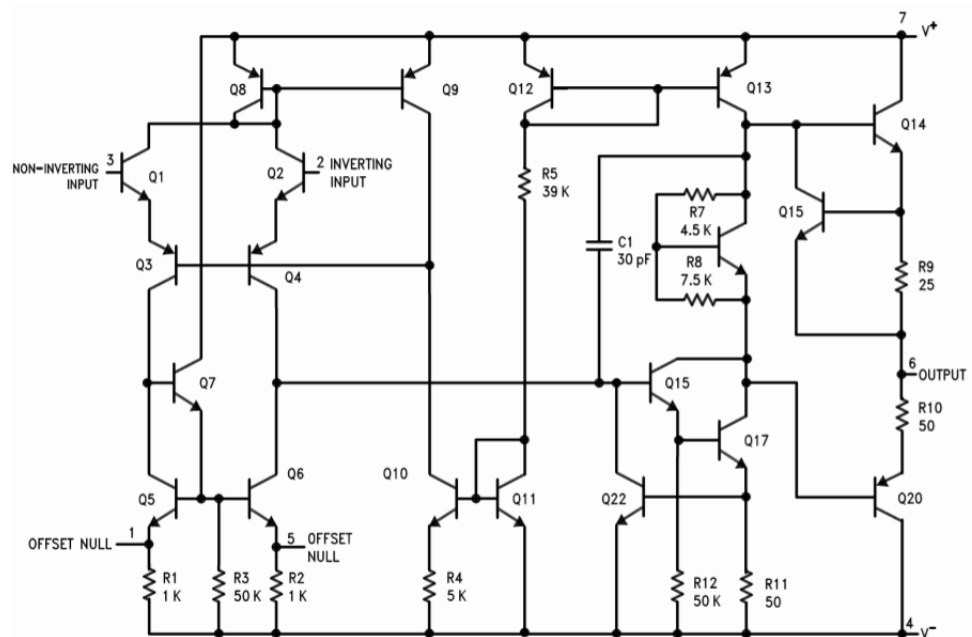


รูปที่ 2.40 แสดงภาพวงจรออปแอมป์ (Op-Amp Circuit)

2.1.14.1 คุณสมบัติของออปแอมป์

ออปแอมป์ (Op-Amp) เป็นชื่อย่อสำหรับเรียกของวงจรขยายที่มาจากคำว่า Operational Amplifier เป็นวงจรขยายแบบต่อตรง (Direct coupled amplifier) ที่มีอัตราขยายสูงมากใช้การป้อนกลับแบบลบไปควบคุมลักษณะการทำงาน ทำให้ผลการทำงานของวงจรไม่ขึ้นกับพารามิเตอร์ภายในของออปแอมป์ วงจรภายในประกอบด้วยวงจรขยายที่ต่อเนื่องกันคือวงจรขยายดิฟเฟอเรนเชียลด้าน

ทางเข้า วงจรขยายดิฟเฟอเรนเชียลภาคที่สอง วงจรเลื่อนระดับและวงจรขยายกำลังด้านทางออก สัญลักษณ์ที่ใช้แทนออปแอมป์จะเป็นรูปสามเหลี่ยม ไอซีออปแอมป์เป็นไอซีที่แตกต่างไปจากลิเนียร์ไอซี ทั่วๆ ไปคือไอซีออปแอมป์มีขาอินพุท 2 ขาเรียกว่าขาเข้าไม่กลับเฟส(Non-Inverting Input) หรือขา (+) และขาเข้ากลับเฟส (Inverting Input) หรือขา (-) ส่วน ทางด้านออกมีเพียงขาเดียว เมื่อสัญญาณป้อนเข้า ขาไม่กลับเฟสสัญญาณทางด้านออกจะมีเฟสตรงกับทางด้านเข้า แต่ถ้าป้อนสัญญาณเข้าที่ขาเข้ากลับเฟส สัญญาณทางออกจะมีเฟสต่างไป 180 องศา จากสัญญาณทางด้านเข้า



รูปที่ 2.41 แสดงแผนภาพบล็อกการทำงานของออปแอมป์

คุณลักษณะของ OP-AMP ในอุดมคติ (Ideal Op-Amp)

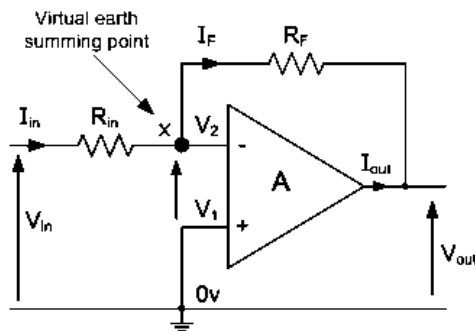
- อัตราขยายของ Op-Amp แบบวงจรรอบเปิด (Open Loop Gain) มีค่าสูงมากจนเป็นอนันต์ $AV_{OL} = \text{infinity}$
- Z_{in} มีค่าสูงมากจนถือได้ว่าเป็น infinity
- Z_{out} มีค่าต่ำมากจนถือได้ว่าเป็น 0
- อัตราขยายของ Op-Amp ไม่ขึ้นกับความถี่
- เมื่อ $V_{in} = 0$ จะได้ V_{out} เป็น 0 ด้วย

2.1.14.2 วงจรขยายสัญญาณแบบกลับเฟส (Inverting Amplifier)

ในวงจรขยายออปแอมป์นั้นสามารถที่จะกำหนดอัตราการขยายของวงจรได้โดยใช้วงจรเนกาทีฟฟีดแบ็ค (Negative Feedback) เมื่อเราป้อนสัญญาณเข้าทางขากลับเฟส (ขา-) แรงดันด้านทางออกจะมีมุมเฟสต่างไปจากแรงดันทางเข้า 180 องศา ซึ่งมีลักษณะตรงข้าม สัญญาณตรงกันข้ามนี้จะถูกป้อนกลับผ่าน R_f เข้ามายังขาอินเวอร์ตอีกครั้งหนึ่ง ตรงจุดนี้จะทำให้สัญญาณเกิดการหักล้างกัน อัตราการขยายก็จะลดลง ถ้าตัวต้านทานที่เป็นตัวป้อนกลับมีค่ามาก จะทำให้สัญญาณป้อนกลับมีขนาดเล็ก อัตราการขยายออกจึงสูง ถ้าตัวต้านทานที่ป้อนกลับมีค่าน้อยสัญญาณป้อนกลับไปได้มาก อัตราการขยายก็จะลดลง ฉะนั้นอัตราส่วนของความต้านทาน R_{in} และ R_f จะเป็นตัวกำหนดอัตราการขยายของวงจรโดยไม่ขึ้นกับอัตราการขยายของออปแอมป์ ซึ่งสามารถหาอัตราการขยายแรงดันได้จากสูตร

$$\text{Gain} = \frac{V_{out}}{V_{in}} = -\frac{R_f}{R_{in}}$$

$$V_{out} = -\frac{R_f}{R_{in}} \times V_{in}$$



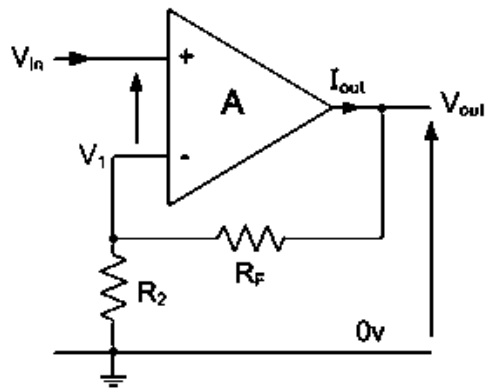
รูปที่ 2.42 แสดงภาพของวงจรขยายสัญญาณแบบกลับเฟส

2.1.14.3 วงจรขยายแบบไม่กลับเฟส (Non-inverting amplifier)

วงจรขยายนี้เป็นวงจรขยายอีกแบบหนึ่งที่ต้องการเฟสในการขยายเป็นเฟสเดียวกัน ดังนั้นการป้อนสัญญาณอินพุตจึงต้องป้อนเข้าที่ขาอินพุตไม่กลับเฟส (+) ซึ่งเมื่อขยายออกที่เอาต์พุตแล้วจะได้สัญญาณเอาต์พุตที่มีเฟสเหมือนเดิม ดังนั้นในวงจรขยายแบบไม่กลับเฟสนี้การป้อนกลับเพื่อลดอัตราการขยายจึงยังคงต้องป้อนไปยังขาอินเวอร์ต (-) เพื่อให้เกิดการหักล้างของสัญญาณกันภายในตัวไอซีออปแอมป์ โดยสามารถหาอัตราการขยายของวงจรได้

$$\frac{V_{out}}{V_{in}} = A_{(v)} = 1 + \frac{R_F}{R_2}$$

$$V_{out} = \left(1 + \frac{R_F}{R_2}\right)V_{in}$$

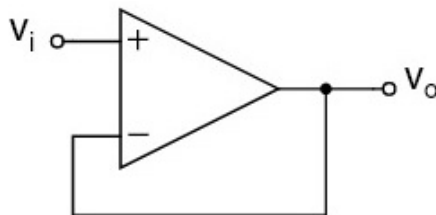


รูปที่ 2.43 แสดงภาพของวงจรขยายแบบไม่กลับเฟส

2.1.14.4 วงจรขยายสัญญาณแบบตามแรงดัน (Voltage Follower (Buffer))

วงจรขยายสัญญาณแบบตามแรงดัน จะมีแรงดันทางด้าเอาต์พุตเท่ากับแรงดันทางด้า

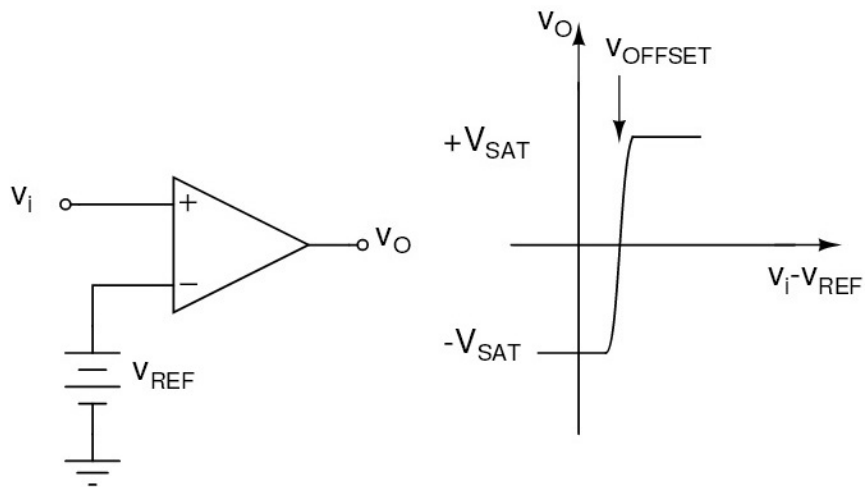
อินพุต



รูปที่ 2.44 แสดงวงจรขยายสัญญาณแบบตามแรงดัน

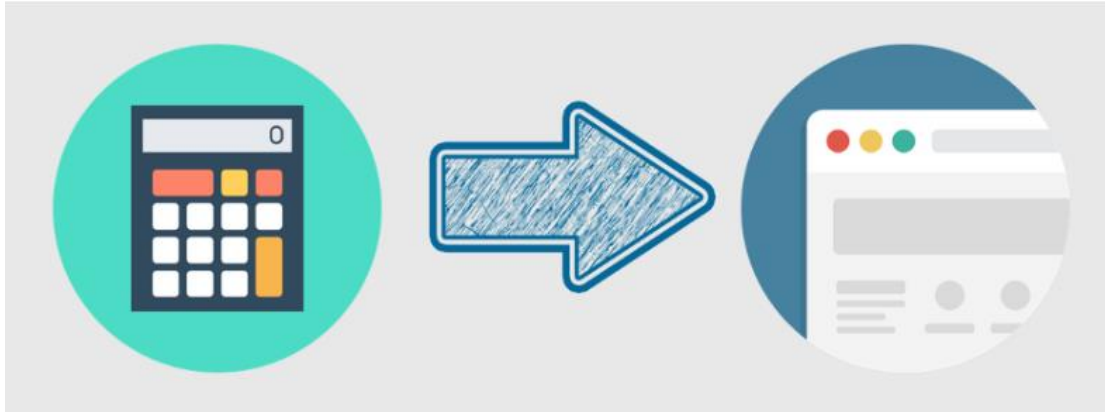
2.1.14.5 วงจรเปรียบเทียบแรงดัน (Comparator)

วงจรเปรียบเทียบแรงดันแสดงตามรูปที่ 2.ป เป็นวงจรแบบไม่เป็นเชิงเส้น ซึ่งจะให้เอาต์พุตอิมิต์ออกมาที่ประมาณ 90% ของแหล่งจ่ายไฟทั้งด้านบวก และด้านลบ ขั้วของเอาต์พุตขึ้นอยู่กับเครื่องหมายของผลต่างอินพุต $v_i - v_{REF}$



รูปที่ 2.45 แสดงวงจรเปรียบเทียบแรงดัน

2.1.15 เว็บแอปพลิเคชัน (Web Application)



รูปที่ 2.46 แสดงภาพตัวอย่างเว็บแอปพลิเคชัน

เว็บแอปพลิเคชัน (Web Application) คือเว็บที่นำเอา โปรแกรมประยุกต์ (Application) มา รวมไว้ในเว็บโดยใช้ภาษาคอมพิวเตอร์เชิงโปรแกรมมิ่งเช่น PHP, NodeJS, Python, .NET, Java ฯลฯ เช่น การพัฒนาเว็บแอปพลิเคชันให้สามารถคำนวณได้เหมือนเครื่องคิดเลข

เว็บไซต์ (Website) — เป็นเว็บไซต์ที่ไว้นำเสนอข้อมูลความรู้ แต่ไม่สามารถคำนวณ หรือ ทำงานอย่างโปรแกรมคอมพิวเตอร์ได้

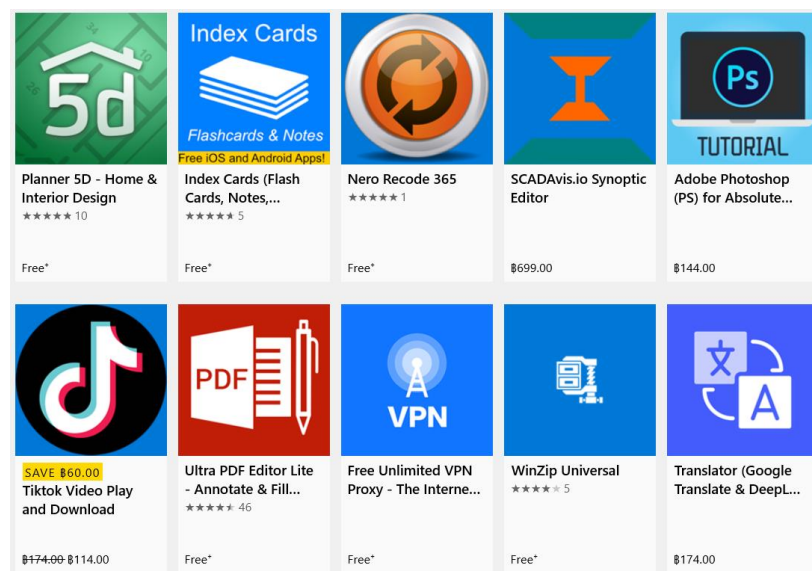
เว็บแอปพลิเคชัน (Web Application) — เป็นเว็บที่นำเอาโปรแกรมคอมพิวเตอร์มาใช้ใน เว็บซึ่งเชื่อมต่อฐานข้อมูลเรียบร้อยแล้ว สามารถเปลี่ยนข้อมูลที่แสดงในเว็บได้อย่างง่ายดายโดยไม่ต้องแก้ไข HTML, CSS และ Javascript จะเห็นตัวอย่างได้ในเว็บต่าง ๆ เว็บขายของ ฯลฯ

จุดเด่นของเว็บแอปพลิเคชัน (Web Application)

- Web Application เหมาะกับองค์กรขนาดเล็กเพราะมีค่าใช้จ่ายต่ำ และคิดค่าใช้จ่ายตามจำนวนการใช้งานจริง
- การใช้งานในองค์กรทำได้ง่าย เพียงแค่มีเว็บเบราว์เซอร์ซึ่งเป็นสิ่งพื้นฐานในคอมพิวเตอร์ปัจจุบันแทบทุกเครื่องก็ใช้งานได้

- ข้อมูลจัดเก็บที่เดียว ง่ายต่อการจัดการ และไม่เกิดความซ้ำซ้อน
- ไม่ต้องการเครื่องคอมพิวเตอร์ประสิทธิภาพสูงซึ่งมีราคาแพง
- อยู่ที่ไหนก็ทำงานได้เพราะสามารถล็อกอินเข้าใช้งานได้เลยไม่ต้องติดตั้งโปรแกรม
- ไม่ต้องมีบุคลากรด้านเทคนิคเป็นของตัวเอง เพราะผู้ให้บริการดูแลเซิร์ฟเวอร์และการบำรุงรักษาเองทั้งหมด
- ส่วน 多くใช้ได้หลากหลายแพลตฟอร์มทั้ง Windows, Linux และ Mac ทำให้องค์กรสามารถเลือกใช้บางเครื่องเป็น Linux ได้เพื่อลดค่าใช้จ่ายด้านลิขสิทธิ์
- เชื่อมต่อกับเว็บแอปหรือบริการออนไลน์อื่นๆได้ง่าย

2.1.16 วินโดว์แอปพลิเคชัน (Windows Application)



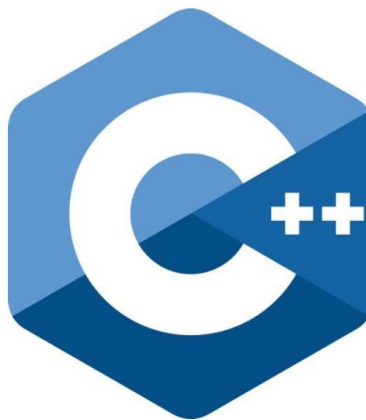
รูปที่ 2.47 ตัวอย่างวินโดว์แอปพลิเคชัน

Windows Application หรือ WinAPP เป็นโปรแกรมประเภท GUI (Graphic User Interface) เรียกง่าย ๆ ก็คือ โปรแกรมประเภทนี้เป็นโปรแกรมที่จะติดต่อหรือตอบสนองกับผู้ใช้ (User) ผ่านทางระบบภาพ Graphic เพื่อให้มีการใช้งานที่ง่ายและเข้าใจได้ง่ายยิ่งขึ้นนั่นเอง ในสมัยก่อนนั้น

โปรแกรมประเภทนี้ จะต้องผ่านการเขียนโปรแกรมขึ้นมาด้วยข้อความหรือ Text จากโปรแกรมอย่างเช่น Notepad หรือ Editplus แต่หากเป็นในปัจจุบัน มีโปรแกรมที่สามารถใช้วิธีคลิกวาง ลากวางให้สามารถจัดการสร้างสรรค์โปรแกรมได้ง่ายยิ่งขึ้น แต่ถึงอย่างไรก็ควรที่จะมีการศึกษาในเรื่องของหลักการพัฒนาโปรแกรมไว้บ้าง เพื่อความสะดวกในการแก้ไขและจัดการโปรแกรมในภายหลัง

2.1.17 ซอฟต์แวร์ที่เกี่ยวข้อง

2.1.17.1 ภาษา C++



รูปที่ 2.48 ภาษา C++

C++ เป็นภาษาคอมพิวเตอร์เพื่อวัตถุประสงค์ทั่วไป ซึ่งสามารถเขียนโปรแกรมได้ทั้งแบบออบเจ็ค และการเขียนแบบปกติทั่วไป และยังมีเครื่องมืออำนวยความสะดวกในการจัดการและเข้าถึงระดับหน่วยความจำนอกจากนี้มันยังถูกนำไปใช้ในการเขียนโปรแกรมแบบต่างๆ มากมาย เช่น โปรแกรมคอมพิวเตอร์ ระบบฝังตัว (Embedded) ไมโครคอนโทรลเลอร์ เว็บเซิร์ฟเวอร์ การพัฒนาเกม และแอปพลิเคชันที่ต้องการประสิทธิภาพอย่างสูง

ภาษา C++ เป็นภาษาที่ถูกออกแบบมาในการเขียนโปรแกรมระบบ ซึ่งมีประสิทธิภาพและความยืดหยุ่นในการออกแบบโปรแกรมสูง C++ เป็นภาษาที่ต้องคอมไพล์ก่อนที่จะนำไปใช้งาน ซึ่งสามารถ

พัฒนาได้ในหลายๆ แพลตฟอร์ม ซึ่งได้รับการสนับสนุนโดยองค์กรต่างๆ ที่ประกอบไปด้วย Free Software Foundation (FSF's GCC) LLVM Microsoft Intel และ IBM

C++ นั้นถูกกำหนดให้เป็นภาษาที่เป็นมาตรฐานโดย International Organization for Standardization (ISO) ซึ่งเวอร์ชันล่าสุดนั้นเผยแพร่ในธันวาคม 2014 คือ ISO/IEC 14882:2014 หรือที่รู้จักกันในชื่อของ C++14 โดยที่ภาษา C++ ได้เริ่มกำหนดมาตรฐานครั้งแรกในปี 1998 คือ ISO/IEC 14882:1998 ภาษา C++ ถูกพัฒนาโดย Bjarne Stroustrup ที่ Bell Labs ตั้งแต่ปี 1979 ซึ่งในตอนแรกเป็นส่วนขยายของภาษา C โดยที่เขาต้องการที่จะพัฒนาภาษาที่มีประสิทธิภาพและยืดหยุ่นเหมือนกับภาษา C และยังมีคุณสมบัติใหม่ที่สูงกว่าสำหรับพัฒนาโปรแกรม

Bjarne Stroustrup นักวิทยาศาสตร์คอมพิวเตอร์ชาวเดนมาร์ก ได้สร้างภาษา C++ ขึ้นในปี 1979 โดยเขาเริ่มจาก "C with Classes" ซึ่งเป็นภาษาก่อนหน้าของภาษา C++ แรงจูงใจสำหรับการสร้างภาษาใหม่นั้นมีต้นกำเนิดมาจากการประสบการณ์ในการเขียนโปรแกรมสำหรับงานวิจัยในการศึกษาระดับปริญญาเอกของเขา ในขณะที่ Stroustrup เริ่มต้นการทำงานที่ AT&T Bell Labs เขามีปัญหาในการวิเคราะห์ UNIX kernel ซึ่งเกี่ยวกับ distributed computing จากการจดจำในประสบการณ์ปริญญาเอกของเขา Stroustrup ตั้งใจว่าจะเพิ่มความสามารถให้ภาษา C กับคุณสมบัติที่เหมือนภาษา Simula เขาเลือกภาษา C เพราะว่ามันเป็นภาษาเขียนโปรแกรมเพื่อวัตถุประสงค์ทั่วไป ที่ทำงานเร็ว สะดวกใช้งานง่าย และใช้กันอย่างแพร่หลาย จนกระทั่งในปี 2011 มาตรฐานของ C++11 ได้ถูกเผยแพร่ โดยการเพิ่มคุณสมบัติใหม่เข้ามามากมาย รวมทั้งการเพิ่มเติมขนาดของไลบรารีมาตรฐาน และให้ความสะดวกแก่โปรแกรมเมอร์ภาษา C++ เป็นอย่างมาก

คอมไพเลอร์คือโปรแกรมคอมพิวเตอร์หรือกลุ่มของโปรแกรมที่แปลงซอสโค้ดที่เขียนขึ้นในภาษา C++ ไปเป็นภาษาคอมพิวเตอร์ (Target language) หลังจากที่ทำการแปลงแล้วจะได้ข้อมูลในรูปแบบของฐานสอง (Binary) ที่เรียกกันว่า Object code เหตุผลที่ต้องแปลงโปรแกรมจากภาษาเขียนโปรแกรมไปเป็นภาษาเครื่องโดยคอมไพเลอร์ก็เพื่อสร้างโปรแกรมที่สามารถทำงานได้ (Executable program) คอมพิวเตอร์สามารถเข้าใจแค่ภาษาเครื่อง ภาษาที่ประกอบไปด้วยตัวเลข 1 และ 0 เราจำเป็นต้องใช้คอมไพเลอร์เพื่อแปลงโปรแกรมที่เราเขียนไปเป็นภาษาเครื่องที่让คอมพิวเตอร์สามารถทำงานได้คอมไพเลอร์ช่วยให้โปรแกรมเมอร์พัฒนาโปรแกรมของพวกเขาได้อย่างง่ายดายในการเขียนโปรแกรมด้วยภาษาระดับสูง อย่างเช่น ภาษา C++

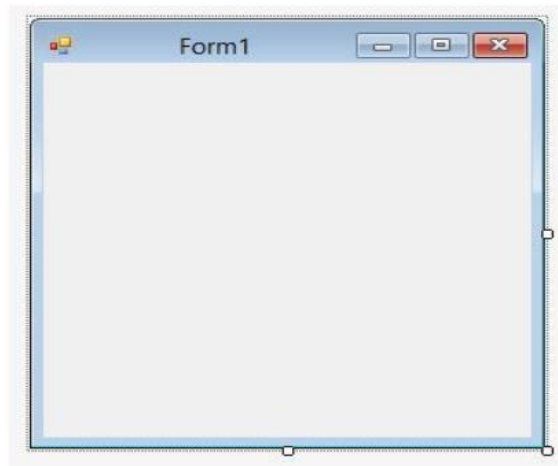
2.1.17.2 ภาษา Visual Basic.NET



รูปที่ 2.49 ภาษา Visual Basic.NET

Visual Basic .NET (VB.NET) เป็นภาษารุ่นที่สามในการเขียนโปรแกรมแบบ event-driven programming (การเขียนโปรแกรมที่ขึ้นกับเหตุการณ์) ซึ่งมาพร้อมกับเครื่องมือพัฒนาจาก Microsoft เปิดตัวครั้งแรกในปี 1991 และได้รับการพัฒนาให้ดีขึ้นจนถึงปี 2008 โดย ภาษา Visual Basic นั้นถูกออกแบบมาเพื่อให้ง่ายต่อการเรียนรู้และง่ายต่อการใช้งาน ภาษา Visual Basic นั้นถูกพัฒนามาจากภาษา Basic ภาษาเขียนโปรแกรมที่เข้าใจง่ายสำหรับผู้เริ่มต้น

ภาษา Visual Basic สนับสนุนการพัฒนาโปรแกรมแบบ rapid application development (RAD) และ graphical user interface (GUI) การเข้าถึงฐานข้อมูล และอื่นๆ ที่ทำงานภายใต้ .NET Framework เวอร์ชันล่าสุดของ Visual Basic นั้นสนับสนุนการเขียนโปรแกรมเชิงวัตถุอย่างเต็มรูปแบบ และยังถูกออกแบบมาเพื่อช่วยให้ผู้เรียนสามารถเรียนรู้ได้อย่างรวดเร็ว โปรแกรมเมอร์สามารถสร้างโปรแกรม GUI อย่างง่ายไปจนถึงซับซ้อน การเขียนโปรแกรมใน VB นั้นสามารถออกแบบได้โดยการลากและวางส่วนประกอบต่างๆ ของตัวควบคุมหรือฟอร์ม การระบุแอตทริบิวต์เพิ่มเติมสำหรับส่วนประกอบเหล่านั้น และเขียนโค้ดเพิ่มเติมสำหรับการทำงาน เพราะว่า VB ได้มีการกำหนดค่าแอตทริบิวต์และฟังก์ชันพื้นฐานในการทำงานของโค้ด โปรแกรมเมอร์จึงไม่จำเป็นต้องเขียนโค้ดเป็นจำนวนมากในการพัฒนาโปรแกรม ข้างล่างนี้เป็นตัวอย่างของหน้าจอแบบฟอร์มที่วางปล่าวใน Visual Studio 2019 ใน ภาษา Visual Basic



รูปที่ 2.50 ตัวอย่างของหน้าออกแบบฟอร์มที่ว่างเปล่าใน Visual Studio 2019

VB 1.0 นั้นถูกแนะนำครั้งแรกในปี 1991 การออกแบบหน้าต่างโดยการลากและวางนั้นได้รับการสืบทอดมาจากตัวสร้าง Prototype ที่พัฒนาโดย Alan Cooper และบริษัทของเขาชื่อว่า Tripod หลังจากนั้น Microsoft ได้ทำการติดต่อกับ Cooper เพื่อพัฒนา Tripod สำหรับระบบฟอร์มที่สามารถเขียนโปรแกรมได้ ภายใต้โค้ดที่มีชื่อว่า Ruby (ซึ่งไม่มีความสัมพันธ์กับภาษา Ruby ในปัจจุบัน) ซึ่ง Tripod ไม่ได้รวมเกี่ยวกับภาษาเขียนโปรแกรมทั้งหมด Microsoft จึงตัดสินใจที่จะรวม Ruby กับ Basic เพื่อสร้างภาษา Visual Basic โดย Interfaces ของ Ruby เป็นตัวสร้างส่วนที่มองเห็นได้ด้วยสายตา (Visual) ของ Visual Basic และได้นำไปรวมกับ "EB" Embedded BASIC engine ที่ออกแบบโดย Microsoft Ruby ยังให้ความสามารถในการเชื่อมโยงกับการไหลตลับไมโครชิปแบบไดนามิกส์ ที่สนับสนุนการควบคุมเพิ่มเติมที่ภายหลังกลายมาเป็น VBX interface

2.1.17.3 ภาษา PHP



รูปที่ 2.51 แสดงสัญลักษณ์ของภาษา PHP

PHP (พีเอชพี) ย่อมาจากคำว่า PHP Hypertext Preprocessor ภาษา PHP เป็นโปรแกรมภาษาคอมพิวเตอร์ระดับสูง ที่สามารถใช้งานในระบบเครือข่ายอินเทอร์เน็ตได้ มีความสามารถสูง และมีผู้นิยมใช้เป็นจำนวนมาก อีกทั้งยังสามารถดาวน์โหลดมาใช้ได้ฟรี จากเว็บไซต์ <http://php.net/>

PHP เป็นภาษาสคริปต์ (Scripting Language) คำสั่งต่างๆ จะเก็บในรูปแบบของข้อความ (Text) สามารถใช้งานอิสระหรืออาจเขียนแทรกอยู่ในภาษา HTML ก็ได้ แต่ในการใช้งานจริงมักใช้งานร่วมกับภาษา HTML ดังนั้นการเขียนโปรแกรมนี้ต้องมีความรู้ด้านภาษา HTML เป็นอย่างดี อย่างไรก็ตาม เราสามารถใช้โปรแกรมประยุกต์มาช่วยอำนวยความสะดวกในการสร้างงานได้ เช่น Macromedia Dreamweaver หรือโปรแกรมประเภท Editor เช่น EditPlus ฯลฯ โปรแกรมเหล่านี้จะช่วยจำแนกคำสั่ง คำสั่ง คำทั่วไป ตัวแปร ฯลฯ ให้มีสีต่างกันเพื่อสะดวกในการสังเกต และมีตัวเลขบอกบรรทัดทำให้สะดวกในการแก้ไข

PHP คือ ภาษาคอมพิวเตอร์ Server-Side Script ซึ่งใช้ในการจัดทำเว็บไซต์และสามารถประมวลผลออกมาในรูปแบบ HTML โดยมีรากฐานโครงสร้างคำสั่งมาจากภาษา ภาษาซี ภาษาจาวา และ ภาษาเพิร์ล เป้าหมายหลักของภาษาPHP คือให้นักพัฒนาเว็บไซต์สามารถเขียนเว็บเพจ ที่มีความตอบโต้ได้อย่างรวดเร็ว

ผู้ให้กำเนิด PHP มีชื่อว่า รัสมัส เลอร์ดอร์ฟ (Rasmus Lerdorf) โดยเริ่มต้นเขียนสคริปต์ Perl CGI ใส่ไว้ในโฮมเพจประวัติส่วนตัว และเห็นว่าการเขียน CGI ด้วย Perl มีความยุ่งยาก จึงได้เขียนโปรแกรมขึ้นใหม่ด้วยภาษา C ที่สามารถแยกส่วนที่เป็นภาษาHTML ออกจากส่วนที่เป็นภาษา C เพื่อแยกประมวลผล แล้วทำการสร้างโค้ด HTML ขึ้นใหม่ โดยตั้งชื่อโปรแกรมนี้นว่า Personal Home Page Tools (PHP-Tools) และได้เริ่มแจกจ่ายโค้ดออกไปในลักษณะฟรีแวร์ ต่อมาจึงได้เริ่มเปิดให้ผู้สนใจเข้าร่วมปรับปรุงและพัฒนา จนกลายเป็นภาษา PHP ในปัจจุบัน

2.1.17.4 ภาษา JavaScript



รูปที่ 2.52 แสดงสัญลักษณ์ของ JavaScript

JavaScript คือ ภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ตที่กำลังได้รับความนิยมอย่างสูง JavaScript เป็น ภาษาสคริปต์เชิงวัตถุ (ที่เรียกกันว่า "สคริปต์" (script) ซึ่งในการสร้างและพัฒนาเว็บไซต์ (ใช้ร่วมกับ HTML) เพื่อให้เว็บไซต์ของเราดูมีการเคลื่อนไหว สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะ "แปลความและดำเนินงานไปที่ละคำสั่ง" (interpret) หรือเรียกว่า อ็อบเจ็กโอเรียนเตด (Object Oriented Programming) ที่มีเป้าหมายในการออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ต สำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้ โดยทำงานร่วมกับ ภาษา HTML และภาษา Java ได้ทั้งทางฝั่งไคลเอนต์ (Client) และทางฝั่งเซิร์ฟเวอร์ (Server)

JavaScript ถูกพัฒนาขึ้นโดย เน็ตสเคปคอมมิวนิเคชันส์ (Netscape Communications Corporation) โดยใช้ชื่อว่า Live Script ออกมาพร้อมกับ Netscape Navigator2.0 เพื่อใช้สร้างเว็บเพจ โดยติดต่อกับเซิร์ฟเวอร์แบบ Live Wire ต่อมาเน็ตสเคปจึงได้ร่วมมือกับ บริษัทซันไมโครซิสเต็มส์ปรับปรุงระบบของบราวเซอร์เพื่อให้สามารถติดต่อกับภาษาจาวาได้ และได้ปรับปรุง LiveScript ใหม่เมื่อ ปี 2538 แล้วตั้งชื่อใหม่ว่า JavaScript JavaScript สามารถทำให้ การสร้างเว็บเพจ มีลูกเล่น ต่าง ๆ มากมาย และยังสามารถโต้ตอบกับผู้ใช้ได้อย่างทันที เช่น การใช้เมาส์คลิก หรือ การกรอกข้อความในฟอร์ม เป็นต้น

เนื่องจาก JavaScript ช่วยให้ผู้พัฒนา สามารถสร้างเว็บเพจได้ตรงตามความต้องการ และมีความน่าสนใจมากขึ้น ประกอบกับเป็นภาษาเปิด ที่ใครก็สามารถนำไปใช้ได้ ดังนั้นจึงได้รับความนิยมเป็นอย่างสูง มีการใช้งานอย่างกว้างขวาง รวมทั้งได้ถูกกำหนดให้เป็นมาตรฐานโดย ECMA การทำงานของ JavaScript จะต้องมีการแปลความคำสั่ง ซึ่งขั้นตอนนี้จะถูกจัดการโดยบราวเซอร์ (เรียกว่าเป็น client-side script) ดังนั้น JavaScript จึงสามารถทำงานได้ เฉพาะบนบราวเซอร์ที่สนับสนุน ซึ่งปัจจุบันบราวเซอร์เกือบทั้งหมดก็สนับสนุน JavaScript แล้ว อย่างไรก็ตาม สิ่งที่ต้องระวังคือ JavaScript มีการพัฒนาเป็นเวอร์ชันใหม่ๆออกมาด้วย ดังนั้น ถ้านำโค้ดของเวอร์ชันใหม่ ไปรันบนบราวเซอร์รุ่นเก่าที่ยังไม่สนับสนุน ก็อาจจะทำให้เกิด error ได้

การทำงานของ JavaScript เกิดขึ้นบนบราวเซอร์ (เรียกว่าเป็น client-side script) ดังนั้นไม่ว่าคุณจะใช้เซิร์ฟเวอร์อะไร หรือที่ไหน ก็ยังคงสามารถใช้ JavaScript ในเว็บเพจได้ ต่างกับภาษาสคริปต์อื่น เช่น Perl, PHP หรือ ASP ซึ่งต้องแปลความและทำงานที่ตัวเครื่องเซิร์ฟเวอร์ (เรียกว่า server-side script) ดังนั้นจึงต้องใช้บนเซิร์ฟเวอร์ ที่สนับสนุนภาษาเหล่านี้เท่านั้น อย่างไรก็ตาม จากลักษณะดังกล่าวก็ทำให้ JavaScript มีข้อจำกัด คือไม่สามารถรับและส่งข้อมูลต่างๆ กับเซิร์ฟเวอร์โดยตรง เช่น การอ่านไฟล์จากเซิร์ฟเวอร์ เพื่อนำมาแสดงบนเว็บเพจ หรือรับข้อมูลจากผู้ชม เพื่อนำไปเก็บบนเซิร์ฟเวอร์ เป็นต้น ดังนั้นงานลักษณะนี้ จึงยังคงต้องอาศัยภาษา server-side script อยู่ (ความจริง JavaScript ที่ทำงานบนเซิร์ฟเวอร์เวอร์ก็มี ซึ่งต้องอาศัยเซิร์ฟเวอร์ที่สนับสนุนโดยเฉพาะเช่นกัน แต่ไม่เป็นที่นิยมนัก)

2.1.17.4.1 Ajax (Asynchronous JavaScript and XML)



รูปที่ 2.53 แสดงสัญลักษณ์ของ Ajax

Ajax (ย่อมาจาก Asynchronous JavaScript and XML) เป็นเทคนิคการเขียนโปรแกรมแบบหนึ่ง ไม่ใช่ภาษาโปรแกรมใหม่อย่างที่บางคนเข้าใจจากชื่อเต็มของ Ajax จะสังเกตว่ามีตัวย่อ 2 ตัวคือ (Asynchronous) JavaScript และ XML ก็คือ การใช้ JavaScript แบบ Asynchronous (แบบคำสั่งไม่เป็นลำดับขั้นหรือไม่ sync. กัน) และใช้ XML โดยมีหลักการทำงาน 2 ประเด็น คือ การ update หน้าจอแบบบางส่วน และการติดต่อสื่อสารกับ Server โดยใช้หลักการ Asynchronous ทำให้ผู้ใช้ไม่ต้องหยุดการทำงาน เพื่อรอการประมวลผลจาก Server รวมถึงการโหลดและการรีเฟรชหน้าจอ ของเบราว์เซอร์ทางฝั่ง Client มีการใช้ Ajax โดยการเพิ่มเลเยอร์ระหว่าง user browser กับ server ทำให้ผู้ใช้สามารถทำงานได้โดยไม่ต้องรอให้ Client ติดต่อไปยัง Server รวมถึงการโหลดและการรีเฟรชหน้าจอทั้งหมดด้วย ดังนั้นผู้ใช้สามารถใช้งาน application ได้อย่างมีประสิทธิภาพมากขึ้น

2.1.17.5 ภาษา HTML (Hyper Text Markup Language)



รูปที่ 2.54 แสดงสัญลักษณ์ของภาษา HTML

HTML ย่อมาจาก Hyper Text Markup Language คือภาษาคอมพิวเตอร์ที่ใช้ในการแสดงผลของเอกสารบน website หรือที่เราเรียกกันว่าเว็บเพจ ถูกพัฒนาและกำหนดมาตรฐานโดยองค์กร World Wide Web Consortium (W3C) และจากการพัฒนาทางด้าน Software ของ Microsoft ทำให้ภาษา HTML เป็นอีกภาษาหนึ่งที่ใช้เขียนโปรแกรมได้ หรือที่เรียกว่า HTML Application

HTML เป็นภาษาประเภท Markup สำหรับการการสร้างเว็บเพจ โดยใช้ภาษา HTML สามารถทำโดยใช้โปรแกรม Text Editor ต่างๆ เช่น Notepad, Editplus หรือจะอาศัยโปรแกรมที่เป็น

เครื่องมือช่วยสร้างเว็บเพจ เช่น Microsoft FrontPage, Dream Weaver ซึ่งอำนวยความสะดวกในการสร้างหน้า HTML ส่วนการเรียกใช้งานหรือทดสอบการทำงานของเอกสาร HTML จะใช้โปรแกรม web browser เช่น Microsoft Internet Explorer (IE), Mozilla Firefox, Safari, Opera, และ Netscape Navigator Google Chrome เป็นต้น

2.1.17.6 ภาษา CSS (Cascading Style Sheet)



รูปที่ 2.55 แสดงสัญลักษณ์ของภาษา CSS

CSS ย่อมาจาก Cascading Style Sheet มักเรียกโดยย่อว่า "สไตล์ชีต" คือภาษาที่ใช้เป็นส่วนของการจัดรูปแบบการแสดงผลเอกสาร HTML โดยที่ CSS กำหนดกฎเกณฑ์ในการระบุรูปแบบ (หรือ "Style") ของเนื้อหาในเอกสาร อันได้แก่ สีของข้อความ สีพื้นหลัง ประเภทตัวอักษร และการจัดวางข้อความ ซึ่งการกำหนดรูปแบบ หรือ Style นี้ใช้หลักการของการแยกเนื้อหาเอกสาร HTML ออกจากคำสั่งที่ใช้ในการจัดรูปแบบการแสดงผล กำหนดให้รูปแบบของการแสดงผลเอกสาร ไม่ขึ้นอยู่กับเนื้อหาของเอกสาร เพื่อให้ง่ายต่อการจัดรูปแบบการแสดงผลของเอกสาร HTML โดยเฉพาะในกรณีที่มีการเปลี่ยนแปลงเนื้อหาเอกสารบ่อยครั้ง หรือต้องการควบคุมให้รูปแบบการแสดงผลเอกสาร HTML มีลักษณะของความสม่ำเสมอทั่วกันทุกหน้าเอกสารภายในเว็บไซต์เดียวกัน โดยกฎเกณฑ์ในการกำหนดรูปแบบ (Style) เอกสาร HTML ถูกเพิ่มเข้ามาครั้งแรกใน HTML 4.0 เมื่อปีพ.ศ. 2539 ในรูปแบบของ CSS level 1 Recommendations ที่กำหนดโดย องค์กร World Wide Web Consortium หรือ W3C

ประโยชน์ของ CSS มีดังต่อไปนี้

- CSS มีคุณสมบัติมากกว่า tag ของ html เช่น การกำหนดกรอบให้ข้อความ รวมทั้งสี รูปแบบของข้อความที่กล่าวมาแล้ว
- CSS นั้นกำหนดที่ต้นของไฟล์ html หรือตำแหน่งอื่น ๆ ก็ได้ และสามารถมีผล กับเอกสารทั้งหมด หมายถึงกำหนด ครั้งเดียวจุดเดียวก็มีผลกับการแสดงผลทั้งหมด ทำให้เวลาแก้ไขหรือปรับปรุงทำได้สะดวก ไม่ต้องไล่ตามแก้ tag ต่างๆ ทั่วทั้งเอกสาร
- CSS สามารถกำหนดแยกไว้ต่างหากจาก ไฟล์เอกสาร html และสามารถนำมาใช้รวม กับเอกสารหลายไฟล์ได้ การแก้ไขก็แก้เพียง จุดเดียวก็มีผลกับเอกสารทั้งหมด

CSS กับ HTML / XHTML นั้นทำหน้าที่คนละอย่างกัน โดย HTML / XHTML จะทำหน้าที่ในการวางโครงร่างเอกสารอย่างเป็นรูปแบบ ถูกต้อง เข้าใจง่าย ไม่เกี่ยวข้องกับการแสดงผล ส่วน CSS จะทำหน้าที่ในการตกแต่งเอกสารให้สวยงาม เรียกได้ว่า HTML /XHTML คือส่วน coding ส่วน CSS คือส่วน design

2.1.17.7 โปรแกรม Ngrok



รูปที่ 2.56 แสดงสัญลักษณ์ของโปรแกรม Ngrok

Ngrok เป็น Tool Open Source พัฒนาโดย GitHub ซึ่งอำนวยความสะดวกให้บุคคลอื่นสามารถเข้าใช้งาน Website หรือ Application ที่กำลังทำงานอยู่บนเครื่อง Localhost นั้นเอง อีกทั้ง Ngrok ยังมีหลากหลายช่องสัญญาณที่มีความปลอดภัยในการรับและส่งข้อมูลจากเครื่องผู้ใช้ไปจนถึงเครื่อง Localhost

โดยบุคคลอื่นสามารถเข้าใช้งาน Website หรือ Application กำลังทำงานอยู่บนเครื่อง Localhost ผ่านทาง URL ของทาง Ngrok โดยที่ทาง Ngrok จะทำการสุ่มสร้าง URL ขึ้นมา และ URL ที่ได้มานั้น จะทำการเปลี่ยนไปทุกครั้งเมื่อมีการปิดหรือเปิดใช้งาน Ngrok

อีกทั้ง Ngrok ยังมีเว็บอินเตอร์เฟส ซึ่งสามารถใช้ตรวจสอบหรือ Monitor ผ่านทาง URL <http://127.0.0.1:4040> ซึ่งสามารถใช้ตรวจสอบการรับส่งข้อมูล Http ทั้งหมด ไม่ว่าจะเป็นการรับส่งข้อมูลแบบ Get หรือแบบ Post, การขอ Request การส่ง Response, Traffic รวมถึง Webhook ที่เข้ามาใช้งานบนเครื่อง Localhost

ระบบปฏิบัติการ (Operating System) ที่ Ngrok รองรับมีดังต่อไปนี้

- Mac OS X
- Linux
- Mac (32-bit)
- Windows (32-bit)
- Linux (ARM)
- Linux (ARM64)
- Linux (32-bit)
- FreeBSD (64-bit)
- FreeBSD (32-bit)

2.1.17.8 ระบบฐานข้อมูล (Database)



รูปที่ 2.57 แสดงสัญลักษณ์ของระบบฐานข้อมูล

Database หรือ ฐานข้อมูล คือ กลุ่มของข้อมูลที่ถูกเก็บรวบรวมไว้ โดยมีความสัมพันธ์ซึ่งกันและกัน โดยไม่ได้บังคับว่าข้อมูลทั้งหมดนี้จะต้องเก็บไว้ในแฟ้มข้อมูลเดียวกันหรือแยกเก็บหลาย ๆ แฟ้มข้อมูล

ระบบฐานข้อมูล (Database System) คือ ระบบที่รวบรวมข้อมูลต่าง ๆ ที่เกี่ยวข้องกันเข้าไว้ด้วยกันอย่างมีระบบมีความสัมพันธ์ระหว่างข้อมูลต่าง ๆ ที่ชัดเจน ในระบบฐานข้อมูลจะประกอบด้วยแฟ้มข้อมูลหลายแฟ้มที่มีข้อมูล เกี่ยวข้องสัมพันธ์กันเข้าไว้ด้วยกันอย่างเป็นระบบและเปิดโอกาสให้ผู้ใช้สามารถใช้งานและดูแลรักษาป้องกันข้อมูลเหล่านี้ ได้อย่างมีประสิทธิภาพ โดยมีซอฟต์แวร์ที่เปรียบเสมือนสื่อกลางระหว่างผู้ใช้และโปรแกรมต่าง ๆ ที่เกี่ยวข้องกับการใช้ฐานข้อมูล เรียกว่า ระบบจัดการฐานข้อมูล หรือ DBMS (data base management system) มีหน้าที่ช่วยให้ผู้ใช้เข้าถึงข้อมูลได้ง่ายสะดวกและมีประสิทธิภาพ การเข้าถึงข้อมูลของผู้ใช้อาจเป็นการสร้างฐานข้อมูล การแก้ไขฐานข้อมูล หรือการตั้งคำถามเพื่อให้ได้ข้อมูลมา โดยผู้ใช้ไม่จำเป็นต้องรับรู้เกี่ยวกับรายละเอียดภายในโครงสร้างของฐานข้อมูล

ประโยชน์ของ ฐานข้อมูล มีดังต่อไปนี้

- ลดการเก็บข้อมูลที่ซ้ำซ้อน ข้อมูลบางชุดที่อยู่ในรูปของแฟ้มข้อมูลอาจมีปรากฏอยู่หลาย ๆ แห่ง เพราะมีผู้ใช้ข้อมูลชุดนี้หลายคน เมื่อใช้ระบบฐานข้อมูลแล้วจะช่วยให้ความซ้ำซ้อนของข้อมูลลดน้อยลง
- รักษาความถูกต้องของข้อมูล เนื่องจากฐานข้อมูลมีเพียงฐานข้อมูลเดียว ในกรณีที่มีข้อมูลชุดเดียวกันปรากฏอยู่หลายแห่งในฐานข้อมูล ข้อมูลเหล่านี้จะต้องตรงกัน ถ้ามีการแก้ไขข้อมูลนี้ทุก ๆ แห่งที่ข้อมูลปรากฏอยู่จะแก้ไขให้ถูกต้องตามกันหมดโดยอัตโนมัติด้วยระบบจัดการฐานข้อมูล
- การป้องกันและรักษาความปลอดภัยให้กับข้อมูลทำได้อย่างสะดวก การป้องกันและรักษาความปลอดภัยกับข้อมูลระบบฐานข้อมูลจะให้เฉพาะผู้ที่เกี่ยวข้องเท่านั้นซึ่งก่อให้เกิดความปลอดภัย(security) ของข้อมูลด้วย

2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 ไอโอทีแพลตฟอร์มโรงเพาะเห็ด

ขันติชัย, ชยุต (2560) ได้ศึกษา ออกแบบ และสร้างไอโอทีแพลตฟอร์มสำหรับใช้ในโรงเพาะเห็ด โดยใช้ Raspberry Pi 3 Model B ซึ่งเป็น Microcontroller มาใช้วัดค่าอุณหภูมิและ ความชื้นจากเซนเซอร์ภายในโรงเรือน จากนั้นจึงส่งข้อมูลอุณหภูมิและความชื้นที่วัดได้ไปยัง Platform และเก็บข้อมูลอุณหภูมิและความชื้นลงบนฐานข้อมูลโดยใช้ MySQL เป็นทั้งฐานข้อมูล และตัวจัดการฐานข้อมูล นอกจากนั้นยังมีระบบ Web Application เพื่อใช้แสดงผลค่าอุณหภูมิและ ความชื้นที่ถูกส่งมาจาก Raspberry Pi หรือประวัติของค่าอุณหภูมิและความชื้นที่เก็บไว้ในฐานข้อมูล โดย Protocol ที่เป็นตัวกลางในการรับส่งข้อมูลระหว่าง Raspberry Pi, Platform และ Web Application คือ MQTT Protocol และใช้ร่วมกับ NodeJS ซึ่งเป็น Service หลักที่ใช้ภายใน Platform

2.2.2 IOT for Smart Farm: A case study of the Lingzhi Mushroom Farm at Maejo University

Oran, Anukit (2560) ได้นำเทคโนโลยีไอโอทีมาใช้ในการเพาะเห็ดหลินจือ โดยวัดค่าความชื้นในโรงเพาะเห็ดหลินจือ และแสดงผลบนโทรศัพท์และคอมพิวเตอร์ผ่าน NETPIE การควบคุมการทำงานของสปริงเกอร์และปั๊มหมอกเป็นแบบอัตโนมัติและแจ้งเตือนสถานะการทำงานของสปริงเกอร์และปั๊มหมอกผ่าน แอปพลิเคชัน LINE

2.2.3 การประยุกต์ใช้เทคโนโลยีไอโอทีควบคุมฟาร์มอัจฉริยะในโรงเรือนเพาะเห็ดนางฟ้า

วีรศักดิ์, สุรพงษ์, รัฐสิทธิ์ (2561) ได้ออกแบบระบบควบคุมอุณหภูมิและความชื้นสำหรับโรงเรือนเพาะเห็ดด้วย ไมโครคอนโทรลเลอร์โดยมีการศึกษาค้นคว้าข้อมูลที่เกี่ยวข้องต่อการเพาะเห็ดตลอดจนการออกแบบโครงสร้างโรงเรือนที่เหมาะสมโดยแบ่งการทดสอบออก 2 ส่วนคือการทดสอบในส่วนของการควบคุมและการทดสอบผลผลิต ของดอกเห็ดในโรงเรือนที่มีการควบคุมอุณหภูมิและความชื้นโดยนำก้อนเห็ดนางรมและเห็ดนางฟ้ามาทดสอบจำนวน 300 ก้อนและเปรียบเทียบประสิทธิภาพโรงเรือนเพาะเห็ดที่มีการควบคุมอุณหภูมิและความชื้นที่สร้างขึ้นกับโรงเรือน โดยประยุกต์ใช้ระบบไอโอทีที่ใช้เซนเซอร์วัดประกอบด้วย วัดอุณหภูมิ ความชื้นสัมพัทธ์ในโรงเรือนเห็ดนางฟ้าและควบคุมการเปิดปิดปั๊มน้ำให้สปริงเกอร์และพ่นหมอกแบบอัตโนมัติ และเซอร์วิสที่ใช้ในการ ส่งข้อมูลขึ้นระบบอินเทอร์เน็ตคือ NETPIE และเซอร์วิสย่อยคือ NETPIE freeboard ในการแสดงสถานะความชื้น และเวลาแบบเรียลไทม์ (real time) และ NETPIE FEED ในการบันทึกข้อมูลความชื้นและเวลาและการดึงข้อมูลมาใช้งานคือ

Node.JS ผ่านเซิร์ฟเวอร์ NETPIE REST API มาเป็นไฟล์ CSV ในส่วนแสดงสถานะการทำงานของการทำงานของน้ำแบบสปริงเกอร์และพ่นหมอกแบบอัตโนมัติผ่านมือถือ

2.2.4 IOT BASED DESIGN IMPLEMENTATION OF MUSHROOM FARM MONITORING USING ARDUINO MICROCONTROLLERS & SENSORS

Parvati, Megha (2561) ในงานวิจัยฉบับนี้นำเสนอระบบการตรวจสอบและควบคุมสิ่งแวดล้อมเพื่อตรวจสอบและควบคุมสภาพแวดล้อมในฟาร์มเห็ด ช่วยให้ผู้ใช้สามารถตรวจสอบอุณหภูมิ ความชื้น ความเข้มข้นของก๊าซคาร์บอนไดออกไซด์และความเข้มของแสงในฟาร์มเห็ดบนอุปกรณ์ Android โดยใช้แพลตฟอร์มออนไลน์ thing Speak ส่วนของตัวควบคุมสภาพแวดล้อมในฟาร์มเห็ดจะควบคุมสภาพแวดล้อมให้เป็นไปตามที่ผู้ใช้กำหนด ข้อมูลสถานะของสภาพแวดล้อมในฟาร์มเห็ดจะถูกส่งไปแสดงผลผ่าน ESP8266 WiFi modem