

การเขียนโปรแกรมระบบไอโอที

(Programming for IoT)



จรรยา พุคมากร
และคณะ

140.-



การเขียนโปรแกรมระบบไอโอที (Programming for IoT)

กลุ่มอาชีพซอฟต์แวร์และการประยุกต์

ISBN 978-616-495-365-9

พิมพ์ครั้งที่ 1 พ.ศ. 2569

จำนวนที่พิมพ์ 3,000 เล่ม



แถมฟรี e-book
แต่ละบทเรียน
ในรูปแบบ QR Code

เรียบเรียงโดย

ขจรพงษ์ พุ่มกรไกรภพ
อลิษา อีราลาภ
ธฤต ไชยมงคล

จัดพิมพ์และจัดจำหน่าย โดย...



วังอักษร

บริษัท วังอักษร จำกัด

69/3 ถนนอรุณอมรินทร์ แขวงวัดอรุณ เขตบางกอกใหญ่ กรุงเทพฯ 10600

โทร. 0-2472-3293-4 Mobile 08-8585-1521

e-mail : wangaksorn9@gmail.com

Facebook : สำนักพิมพ์ วังอักษร

www.wangaksorn.com

Line ID : wangaksorn



บริษัท วังอักษร จำกัด ขอสงวนลิขสิทธิ์

ตามพระราชบัญญัติลิขสิทธิ์ พ.ศ. 2537 และตามพระราชบัญญัติลิขสิทธิ์เพิ่มเติม พ.ศ. 2565
ห้ามมิให้ทำการคัดลอก ตัดแปลง แก้ไขเนื้อหา หรือเผยแพร่ต่อสาธารณชน ไม่ว่ารูปแบบใด
ชื่อและเครื่องหมายการค้าอื่น ๆ ที่อ้างอิงในหนังสือเล่มนี้ เป็นสิทธิโดยชอบด้วยกฎหมายของเจ้าของแต่ละราย
บริษัทฯ มิได้อ้างความเป็นเจ้าของแต่อย่างใด



ตารางกำหนดการจัดการเรียนรู้
การเขียนโปรแกรมระบบไอโอที
(Programming for IoT)

ครั้งที่	รายการ		จำนวนชั่วโมง		
	ทฤษฎี	ปฏิบัติ	ท	ป	รวม
1	บทเรียนที่ 1 พื้นฐานระบบสมองกลฝังตัว	ใบงานที่ 1.1 การเปิด - ปิดไฟ LED ใบงานที่ 1.2 การใช้สวิตช์ควบคุม LED	1	4	5
2 - 5	บทเรียนที่ 2 สถาปัตยกรรม IoT	ใบงานที่ 2.1 การสร้างฐานข้อมูล และโฮสต์ตั้ง ใบงานที่ 2.2 การแลกเปลี่ยนข้อมูล ระหว่าง ESP8266 กับ Firebase ใบงานที่ 2.3 IoT เบื้องต้นด้วยการใช้ ESP8266 และเซนเซอร์ ใบงานที่ 2.4 การควบคุมอุปกรณ์ ผ่านอินเทอร์เน็ตอย่างง่าย	4	16	20
6 - 10	บทเรียนที่ 3 ระบบสื่อสารและโปรโตคอล	ใบงานที่ 3.1 การสื่อสารข้อมูลแบบ อนุกรมด้วย UART ใบงานที่ 3.2 การสื่อสารข้อมูลแบบ SPI ใบงานที่ 3.3 การสื่อสารข้อมูลแบบ I2C ใบงานที่ 3.4 การอ่านข้อมูลอุณหภูมิ ด้วย I2C ใบงานที่ 3.5 การสื่อสารข้อมูลผ่าน เครือข่ายอินเทอร์เน็ต ใบงานที่ 3.6 การสื่อสารข้อมูลด้วย MQTT	5	20	25



ครั้งที่	รายการ		จำนวนชั่วโมง		
	ทฤษฎี	ปฏิบัติ	ท	ป	รวม
11 - 13	บทเรียนที่ 4 การพัฒนาเว็บแอปพลิเคชัน สำหรับ IoT	ใบงานที่ 4.1 Bootstrap 5 ใบงานที่ 4.2 การสร้างหน้าเว็บไซต์ ด้วย React ใบงานที่ 4.3 การแลกเปลี่ยนข้อมูล ระหว่าง React กับ Firebase	3	12	15
14 - 15	บทเรียนที่ 5 โครงงานระบบ IoT และเอกสารประกอบ	ใบงานที่ 5.1 การออกแบบระบบ IoT ด้วย UML Modeling ใบงานที่ 5.2 การจัดทำคู่มือการใช้งาน ระบบ IoT	2	8	10
ประเมินผลสัมฤทธิ์การเรียนรู้ระดับรายวิชา					
รวม			15	60	75



คำนำ

บริษัท วังอักษร จำกัด ได้จัดทำหนังสือวิชา **การเขียนโปรแกรมระบบไอโอที (Programming for IoT)** เพื่อใช้บูรณาการและประกอบการเรียนการสอน ตามหลักสูตรอาชีวศึกษา (สอศ.) หลักสูตรส่งเสริมการเรียนรู้ (สกร.) หลักสูตรอุดมศึกษา ผู้สนใจในการพัฒนาการเรียนรู้และทักษะ

หนังสือเล่มนี้ประกอบด้วยเนื้อหาสาระ ใบบาง แบบฝึกหัดคู่ขนาน มีจุดประสงค์ในการยกระดับผู้เรียนให้เรียนทฤษฎีและปฏิบัติ เพื่อเสริมสร้างทักษะ ประสบการณ์ และการประยุกต์ใช้ได้จริง โดยเน้นกระบวนการเรียนรู้ที่มีคุณธรรม จริยธรรม จรรยาบรรณวิชาชีพ และกิจนิสัยในการทำงาน เพื่อสร้างสมรรถนะตามความต้องการของสถานประกอบการ เนื้อหาในหนังสือเล่มนี้แบ่งออกเป็น **5 บทเรียน** คือ **บทเรียนที่ 1 พื้นฐานระบบสมองกลฝังตัว** **บทเรียนที่ 2 สถาปัตยกรรม IoT** **บทเรียนที่ 3 ระบบสื่อสารและโพรโทคอลใน IoT** **บทเรียนที่ 4 การพัฒนาเว็บแอปพลิเคชันสำหรับ IoT** และ**บทเรียนที่ 5 โครงงานระบบ IoT และเอกสารประกอบ**

หนังสือเล่มนี้มีเนื้อหาที่ทันสมัย ตามแผนการพัฒนาศึกษาด้านวิชาชีพ ซึ่งสอดคล้องกับยุทธศาสตร์ชาติและแผนการศึกษาแห่งชาติ เพื่อให้ผู้เรียนมีผลลัพธ์การเรียนรู้และสมรรถนะตามมาตรฐานอาชีพ สถาบันคุณวุฒิวิชาชีพ รหัส 11309, 12303, 12307, 12309, อาชีพนักพัฒนาซอฟต์แวร์เพื่ออินเทอร์เน็ตของสรรพสิ่ง ระดับ 4

ขจรพงษ์ พุ่มมรโกธภาพ
อลิษา อีราลาภ
ธฤต ไชยมงคล



IoT สารบัญ

บทเรียนที่ 1	พื้นฐานระบบสมองกลฝังตัว.....	1
1.1	ความรู้พื้นฐานเกี่ยวกับระบบสมองกลฝังตัว.....	2
1.2	โครงสร้างและส่วนประกอบของ ESP8266.....	6
1.3	ไฟฟ้าและแหล่งจ่ายไฟฟ้า.....	8
1.4	การเขียนโปรแกรมเบื้องต้นสำหรับ ESP8266.....	10
ใบงานที่ 1.1	การเปิด - ปิดไฟ LED.....	27
ใบงานที่ 1.2	การใช้สวิตช์ควบคุม LED.....	32
บทเรียนที่ 2	สถาปัตยกรรม IoT.....	37
2.1	ความรู้เบื้องต้นเกี่ยวกับ IoT.....	38
2.2	องค์ประกอบของสถาปัตยกรรม IoT.....	39
2.3	เทคโนโลยีไฟร์เบส.....	41
2.4	การประยุกต์ใช้งานระบบ IoT.....	42
ใบงานที่ 2.1	การสร้างฐานข้อมูลและโฮสติ้ง.....	44
ใบงานที่ 2.2	การแลกเปลี่ยนข้อมูลระหว่าง ESP8266 กับ Firebase.....	55
ใบงานที่ 2.3	IoT เบื้องต้นด้วยการใช้ ESP8266 และเซนเซอร์.....	58
ใบงานที่ 2.4	การควบคุมอุปกรณ์ผ่านอินเทอร์เน็ตอย่างง่าย.....	64
บทเรียนที่ 3	ระบบสื่อสารและโพรโทคอล.....	69
3.1	โพรโทคอลใน IoT.....	70
3.2	การสื่อสารข้อมูลแบบอนุกรม.....	72
3.3	เครือข่ายอินเทอร์เน็ต.....	80
3.4	MQTT กับ IoT.....	81
ใบงานที่ 3.1	การสื่อสารข้อมูลแบบอนุกรมด้วย UART.....	93
ใบงานที่ 3.2	การสื่อสารข้อมูลแบบ SPI.....	98
ใบงานที่ 3.3	การสื่อสารข้อมูลแบบ I2C.....	103



ใบงานที่ 3.4 การอ่านข้อมูลอุณหภูมิด้วย I2C.....	109
ใบงานที่ 3.5 การสื่อสารข้อมูลผ่านเครือข่ายอินเทอร์เน็ต.....	115
ใบงานที่ 3.6 การสื่อสารข้อมูลด้วย MQTT.....	119

บทเรียนที่ 4 การพัฒนาเว็บแอปพลิเคชันสำหรับ IoT..... 126

4.1 การพัฒนาเว็บไซต์ด้วย React.....	127
4.2 การตกแต่งเว็บไซต์.....	129
4.3 การติดตั้งแพ็คเกจ.....	131
ใบงานที่ 4.1 Bootstrap 5.....	137
ใบงานที่ 4.2 การสร้างหน้าเว็บไซต์ด้วย React.....	142
ใบงานที่ 4.3 การแลกเปลี่ยนข้อมูลระหว่าง React กับ Firebase.....	161

บทเรียนที่ 5 โครงงานระบบ IoT และเอกสารประกอบ 172

5.1 แนวคิดและขอบเขตของโครงงานระบบ IoT.....	173
5.2 การออกแบบระบบ IoT ด้วย UML Modeling.....	176
5.3 การพัฒนาและเชื่อมต่อระบบด้วย APIs.....	183
5.4 การทดสอบโปรแกรมย่อยและระบบ IoT.....	185
5.5 การจัดทำคู่มือการใช้งานระบบ IoT.....	187
ใบงานที่ 5.1 การออกแบบระบบ IoT ด้วย UML Modeling.....	190
ใบงานที่ 5.2 การจัดทำคู่มือการใช้งานระบบ IoT.....	193

บรรณานุกรม 196

บทเรียนที่

1

พื้นฐานระบบสมองกลฝังตัว



หัวข้อหลัก

- 1.1 ความรู้พื้นฐานเกี่ยวกับระบบสมองกลฝังตัว
- 1.2 โครงสร้างและส่วนประกอบของ ESP8266
- 1.3 ไฟฟ้าและแหล่งจ่ายไฟฟ้า
- 1.4 การเขียนโปรแกรมเบื้องต้นสำหรับ ESP8266



ผลลัพธ์การเรียนรู้ระดับบทเรียน

เขียนโปรแกรมควบคุมการทำงานของอุปกรณ์พื้นฐานบน ESP8266 เพื่อนำไปทดสอบ วิเคราะห์ผล และประยุกต์ใช้กับอุปกรณ์ไอโอที (IoT)



แนวคิดหลัก

การควบคุมอุปกรณ์อิเล็กทรอนิกส์ผ่านระบบสมองกลฝังตัว เป็นพื้นฐานสำคัญในการพัฒนาอุปกรณ์อัจฉริยะและระบบไอโอที (IoT) โดยการเรียนรู้วิธีการเชื่อมต่อและการเขียนโปรแกรมควบคุมอุปกรณ์ต่าง ๆ จะช่วยให้ผู้เรียนสามารถพัฒนาทักษะการทำงานกับไมโครคอนโทรลเลอร์และเซนเซอร์ได้อย่างมีประสิทธิภาพ



สมรรถนะการเรียนรู้

1. แสดงความรู้เกี่ยวกับสมองกลฝังตัวและการเขียนโปรแกรมควบคุมเบื้องต้น
2. เขียนโปรแกรม เชื่อมต่อ และตั้งค่าอุปกรณ์พื้นฐานสำหรับประยุกต์ใช้กับไอโอที (IoT)
3. ทดสอบและวิเคราะห์ผลการทำงานของโปรแกรมด้วยความละเอียดรอบคอบ



จุดประสงค์การเรียนรู้

1. อธิบายความรู้พื้นฐานของระบบสมองกลฝังตัวได้ถูกต้อง
2. อธิบายหลักการเขียนโปรแกรมเบื้องต้นสำหรับการควบคุมได้
3. วางผังงานและเขียนโปรแกรมเพื่อประยุกต์ใช้ในการควบคุมอุปกรณ์ไอโอที (IoT)
4. เขียนโปรแกรมเพื่อการควบคุมอุปกรณ์พื้นฐานบน ESP8266
5. ทดสอบและวิเคราะห์ผลการทำงานของโปรแกรมด้วยความละเอียดรอบคอบ



ระบบสมองกลฝังตัว ซึ่งเป็นระบบคอมพิวเตอร์ที่ออกแบบมาเพื่อทำงานเฉพาะด้านในอุปกรณ์ต่าง ๆ เช่น เครื่องใช้ไฟฟ้า รถยนต์ และอุปกรณ์ไอโอที (Internet of Things : IoT) ผู้เรียนจะได้ศึกษาเกี่ยวกับส่วนประกอบหลักของระบบสมองกลฝังตัว วิธีการทำงาน และความสำคัญของการใช้ระบบเหล่านี้ ในการควบคุมและตรวจสอบการทำงานของอุปกรณ์ต่าง ๆ นอกจากนี้ยังมีการปฏิบัติการเขียนโปรแกรมเบื้องต้นสำหรับไมโครคอนโทรลเลอร์ เช่น ESP8266 เพื่อเตรียมความพร้อมสำหรับการพัฒนาโครงการไอโอที (IoT) ในขั้นต่อไป



ความรู้พื้นฐานเกี่ยวกับระบบสมองกลฝังตัว

ระบบสมองกลฝังตัว (Embedded Systems) เป็นระบบคอมพิวเตอร์ที่ถูกออกแบบมาเพื่อทำหน้าที่เฉพาะด้าน โดยมักฝังอยู่ในอุปกรณ์อิเล็กทรอนิกส์ที่เราใช้ในชีวิตประจำวัน เช่น เครื่องซักผ้า โทรทัศน์ รถยนต์ และอุปกรณ์ไอโอที (Internet of Things : IoT) ระบบเหล่านี้จะมีลักษณะพิเศษที่ไม่เหมือนกับคอมพิวเตอร์ทั่วไป เพราะจะต้องทำงานในสภาพแวดล้อมที่มีข้อจำกัดทางทรัพยากรและพลังงาน การเรียนรู้เกี่ยวกับระบบสมองกลฝังตัวจะช่วยให้ผู้เรียนเข้าใจหลักการทำงาน และสามารถพัฒนาอุปกรณ์อัจฉริยะที่ตอบสนองต่อความต้องการในชีวิตประจำวันได้

1.1.1 ระบบสมองกลฝังตัวคืออะไร

ระบบสมองกลฝังตัว (Embedded Systems) คือ ระบบคอมพิวเตอร์ชนิดหนึ่งที่ถูกออกแบบมาเพื่อทำงานเฉพาะด้านในอุปกรณ์ต่าง ๆ โดยมักฝังตัวอยู่ภายในเครื่องใช้ไฟฟ้า ยานยนต์ อุปกรณ์อิเล็กทรอนิกส์ หรืออุปกรณ์ IoT ระบบสมองกลฝังตัวจะทำงานตามโปรแกรมที่ถูกกำหนดไว้ และจะทำงานในสภาพแวดล้อมที่มีข้อจำกัดด้านทรัพยากร เช่น พลังงานหรือหน่วยความจำ โดยแบ่งออกเป็น 2 กลุ่มดังนี้

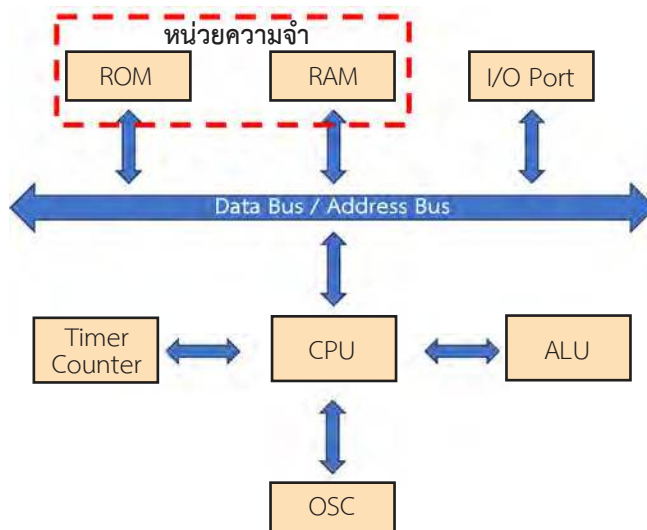
1. Industrial IoT (IIoT) เน้นการใช้งานในภาคอุตสาหกรรม โดยมุ่งเน้นที่การเพิ่มประสิทธิภาพ ลดต้นทุน ปรับปรุงการทำงานของกระบวนการผลิต และการดำเนินงานในโรงงานหรืออุตสาหกรรมต่าง ๆ IIoT ใช้เซนเซอร์ อุปกรณ์ที่เชื่อมต่อกับเครือข่ายและการวิเคราะห์ข้อมูล เพื่อให้สามารถติดตามและควบคุมการทำงานของเครื่องจักรและระบบต่าง ๆ ในอุตสาหกรรมได้แบบเรียลไทม์ เช่น การบำรุงรักษาเชิงคาดการณ์ (Predictive Maintenance) การควบคุมกระบวนการผลิตแบบเรียลไทม์ (Real-time Process Control) การจัดการซัพพลายเชนอัจฉริยะ (Smart Supply Chain Management) และระบบควบคุมพลังงาน (Energy Management Systems)

2. Commercial IoT (CIoT) มุ่งเน้นการใช้งาน IoT ในภาคธุรกิจและการพาณิชย์ รวมถึงการใช้งานในชีวิตประจำวันของผู้บริโภค อุปกรณ์ CIoT มักเกี่ยวข้องกับการปรับปรุงประสบการณ์ผู้ใช้ การเพิ่มประสิทธิภาพการดำเนินงานในสถานประกอบการ ตลอดจนการจัดการระบบในอาคารหรือ

สภาพแวดล้อมทางธุรกิจต่าง ๆ เช่น บ้านอัจฉริยะ (Smart Home) สำนักงานอัจฉริยะ (Smart Office) และการจัดการอาคาร (Building Management)

1.1.2 ส่วนประกอบหลักของระบบสมองกลฝังตัว

ระบบสมองกลฝังตัว (Embedded Systems) มีส่วนประกอบเหมือนกันกับไมโครคอนโทรลเลอร์ ประกอบด้วยส่วนประกอบหลักหลายส่วนที่ทำงานร่วมกัน เพื่อให้สามารถควบคุมและดำเนินการต่าง ๆ ได้อย่างมีประสิทธิภาพ ส่วนประกอบหลักของระบบสมองกลฝังตัวและโครงสร้างภายใน มีองค์ประกอบและรายละเอียดดังรูปที่ 1.1



รูปที่ 1.1 โครงสร้างทั่วไปของไมโครคอนโทรลเลอร์

1. **หน่วยประมวลผลกลาง (Central Processing Unit : CPU)** เป็นหัวใจหลักของระบบสมองกลฝังตัว มีหน้าที่ประมวลผลคำสั่งและข้อมูลต่าง ๆ ที่ได้รับจากโปรแกรม โดยทำหน้าที่ควบคุมการทำงานของส่วนประกอบอื่น ๆ ในระบบ ไมโครคอนโทรลเลอร์ (Microcontroller) ซึ่งมักจะรวมเอา CPU หน่วยความจำ และพอร์ต I/O ไว้ในชิปเดียวกัน ถือเป็น CPU สำหรับระบบสมองกลฝังตัว

2. **หน่วยความจำ (Memory)** หน่วยความจำในระบบสมองกลฝังตัวถูกใช้สำหรับเก็บข้อมูลและโปรแกรมที่ระบบต้องใช้ในการทำงาน แบ่งออกเป็น 2 ประเภทหลัก ดังนี้

2.1 RAM (Random Access Memory) ใช้สำหรับเก็บข้อมูลชั่วคราวที่ระบบต้องการในการประมวลผล

2.2 ROM (Read-only Memory) ใช้สำหรับเก็บโปรแกรมและข้อมูลที่ไม่เปลี่ยนแปลง เช่น เฟิร์มแวร์ (Firmware) ที่ควบคุมการทำงานของระบบ

3. **อินพุต/เอาต์พุต (Input/Output : I/O)** พอร์ตอินพุต/เอาต์พุตเป็นส่วนที่ช่วยให้ระบบสมองกลฝังตัวสามารถเชื่อมต่อกับอุปกรณ์ภายนอก เช่น เซนเซอร์ ตัวกระตุ้น (Actuator) จอแสดงผล หรือปุ่มกด อินพุตใช้สำหรับรับข้อมูลจากภายนอกเข้าสู่ระบบ ส่วนเอาต์พุตใช้สำหรับส่งข้อมูลหรือคำสั่งจากระบบไปยังอุปกรณ์ภายนอก

4. อุปกรณ์จัดเก็บข้อมูล (Storage) ระบบสมองกลฝังตัวอาจมีการจัดเก็บข้อมูลเพิ่มเติม เช่น Flash Memory หรือ EEPROM ที่ใช้ในการเก็บข้อมูลที่ต้องการเก็บรักษาเมื่อระบบปิดตัวลง การจัดเก็บข้อมูลเหล่านี้จะช่วยให้ระบบสามารถบันทึกสถานะหรือข้อมูลสำคัญเพื่อใช้ในภายหลังได้

5. ตัวควบคุมพลังงาน (Power Management) ระบบสมองกลฝังตัวมักถูกออกแบบมาให้ใช้พลังงานน้อย และสามารถทำงานได้ภายใต้แหล่งจ่ายไฟที่จำกัด เช่น แบตเตอรี่ ตัวควบคุมพลังงานทำหน้าที่จัดการและควบคุมการใช้พลังงานของระบบ เพื่อให้ระบบทำงานได้อย่างมีประสิทธิภาพและประหยัดพลังงาน

6. ระบบเครือข่ายและการสื่อสาร (Networking and Communication) ระบบสมองกลฝังตัวมักมีความสามารถในการเชื่อมต่อและสื่อสารกับอุปกรณ์อื่น ๆ ผ่านโพรโทคอลการสื่อสารต่าง ๆ เช่น UART, I2C, SPI, หรือการเชื่อมต่อไร้สาย เช่น Wi-Fi และ Bluetooth ความสามารถในการสื่อสารนี้ทำให้ระบบสมองกลฝังตัวสามารถทำงานร่วมกับอุปกรณ์อื่น ๆ ในระบบ IoT หรือระบบควบคุมอื่น ๆ ได้

7. ตัวแปลงสัญญาณ (Analog-to-digital Converter : ADC และ Digital-to-analog Converter : DAC) ในระบบสมองกลฝังตัวอาจมีความจำเป็นต้องแปลงสัญญาณจากรูปแบบดิจิทัลเป็นแอนะล็อก (DAC) หรือจากแอนะล็อกเป็นดิจิทัล (ADC) เพื่อเชื่อมต่อกับเซนเซอร์หรืออุปกรณ์ต่าง ๆ ที่ทำงานด้วยสัญญาณแอนะล็อก

8. ระบบปฏิบัติการหรือเฟิร์มแวร์ (Operating System or Firmware) เป็นซอฟต์แวร์ที่ทำหน้าที่ควบคุมการทำงานของฮาร์ดแวร์ และจัดการทรัพยากรต่าง ๆ ในระบบสมองกลฝังตัว เฟิร์มแวร์มักจะถูกบันทึกไว้ในหน่วยความจำแบบถาวร เช่น ROM หรือ Flash Memory และทำงานทันทีที่ระบบเปิดใช้งาน

1.1.3 ประโยชน์ของ IoT

1. เพิ่มประสิทธิภาพการทำงาน IoT ซึ่งจะช่วยให้งานของเครื่องจักรและระบบต่าง ๆ มีประสิทธิภาพมากขึ้น โดยการรวบรวมและวิเคราะห์ข้อมูลจากเซนเซอร์ในแบบเรียลไทม์ ทำให้สามารถปรับปรุงกระบวนการทำงาน และลดความสูญเสียในกระบวนการผลิตได้

2. ลดต้นทุน ด้วยการใช้ IoT เพื่อการบำรุงรักษาเชิงคาดการณ์ (Predictive Maintenance) และการปรับปรุงการใช้พลังงาน การดำเนินงานสามารถลดต้นทุนที่เกี่ยวข้องกับการหยุดทำงานโดยที่ไม่คาดคิดและการใช้ทรัพยากรได้

3. ปรับปรุงความปลอดภัย IoT สามารถใช้ตรวจสอบสภาพแวดล้อมการทำงาน และตรวจจับปัญหาด้านความปลอดภัยที่อาจเกิดขึ้นได้ทันที เช่น การตรวจจับการรั่วไหลของแก๊ส หรือการตรวจสอบความปลอดภัยของระบบรักษาความปลอดภัยในอาคาร

4. เพิ่มความสะดวกสบาย IoT ในชีวิตประจำวัน เช่น ในบ้านอัจฉริยะช่วยเพิ่มความสะดวกสบายให้กับผู้ใช้ โดยการควบคุมอุปกรณ์ต่าง ๆ เช่น ระบบแสงสว่าง เครื่องปรับอากาศ และเครื่องใช้ไฟฟ้าผ่านแอปพลิเคชันมือถือหรือคำสั่งเสียง

5. การตัดสินใจที่ดีขึ้นด้วยข้อมูล การเก็บข้อมูลจากอุปกรณ์ IoT ช่วยให้องค์กรสามารถวิเคราะห์และใช้ข้อมูลเพื่อการตัดสินใจที่มีพื้นฐานจากข้อเท็จจริงและข้อมูลจริง ซึ่งจะนำไปสู่การวางแผนและการดำเนินงานที่มีประสิทธิภาพมากขึ้น

6. การพัฒนาผลิตภัณฑ์และบริการใหม่ ข้อมูลที่ได้รับจาก IoT ช่วยให้องค์กรสามารถเข้าใจความต้องการของลูกค้าและปรับปรุงผลิตภัณฑ์หรือบริการให้ตรงกับความต้องการเหล่านั้น รวมถึงพัฒนาผลิตภัณฑ์ใหม่ ๆ ที่ตอบสนองต่อการใช้งานที่หลากหลาย

1.1.4 การใช้งานของ IoT

1. การบำรุงรักษาเชิงคาดการณ์ (Predictive Maintenance) ใช้เซนเซอร์ตรวจสอบการทำงานของเครื่องจักร เพื่อตรวจจับความผิดปกติและทำการบำรุงรักษาก่อนที่ปัญหาจะลุกลาม ช่วยลดการหยุดทำงานที่ไม่คาดคิด และยืดอายุการใช้งานของอุปกรณ์

2. บ้านอัจฉริยะ (Smart Home) การควบคุมอุปกรณ์ภายในบ้าน เช่น ระบบแสงสว่าง ระบบรักษาความปลอดภัย เครื่องปรับอากาศ และเครื่องใช้ไฟฟ้าผ่านแอปพลิเคชันมือถือ เพิ่มความสะดวกสบายและประหยัดพลังงาน

3. การจัดการพลังงาน (Energy Management) ระบบ IoT ช่วยในการตรวจสอบและจัดการการใช้พลังงานในอาคารหรือโรงงาน เช่น การควบคุมระบบทำความร้อนและความเย็นอย่างมีประสิทธิภาพ เพื่อลดการใช้พลังงานและค่าใช้จ่าย

4. การตรวจสอบและควบคุมในอุตสาหกรรม (Industrial Monitoring and Control) ใช้ IoT ตรวจสอบการทำงานของเครื่องจักร การควบคุมกระบวนการผลิต และการจัดการคุณภาพในสายการผลิตแบบเรียลไทม์ เพื่อเพิ่มประสิทธิภาพและลดความเสี่ยงจากข้อผิดพลาด

5. การดูแลสุขภาพ (Healthcare IoT) การติดตามสุขภาพของผู้ป่วยผ่านอุปกรณ์สวมใส่ เช่น การตรวจวัดอัตราการเต้นของหัวใจ ความดันโลหิต และระดับน้ำตาลในเลือดแบบเรียลไทม์ ช่วยให้แพทย์สามารถดูแลและรักษาผู้ป่วยได้อย่างแม่นยำและทันเวลาที่

6. ระบบการขนส่งอัจฉริยะ (Smart Transportation) การใช้ IoT ในการจัดการการจราจร ตรวจสอบตำแหน่งและสภาพของยานพาหนะ และการจัดการระบบขนส่งมวลชนแบบเรียลไทม์ เพื่อเพิ่มความสะดวกและลดการจราจรติดขัด

7. การค้าปลีกอัจฉริยะ (Smart Retail) การใช้ IoT ในร้านค้าปลีกเพื่อการจัดการสินค้าคงคลัง ระบบชำระเงินแบบอัตโนมัติ และการปรับแต่งประสบการณ์ช้อปปิ้งให้เหมาะสมกับความต้องการของลูกค้าแต่ละคน

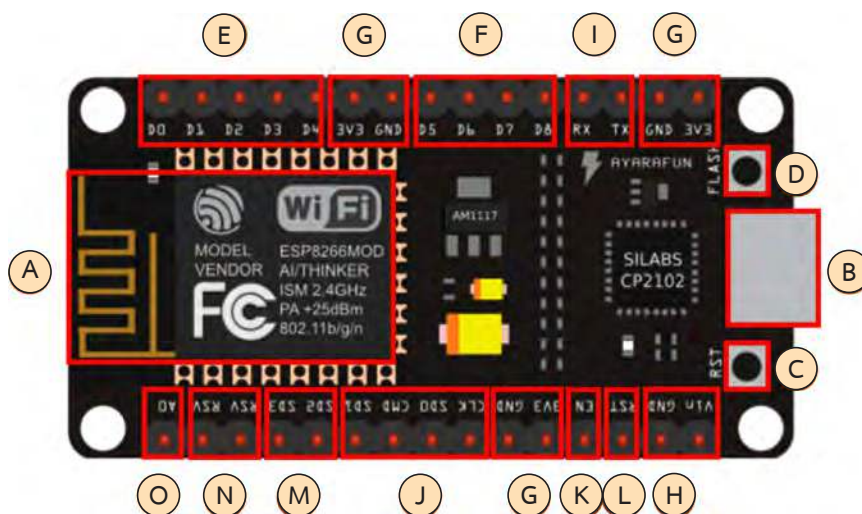
1.2 โครงสร้างและส่วนประกอบของ ESP8226

1.2.1 อะไรคือ ESP8266

ESP8266 เป็นไมโครคอนโทรลเลอร์ขนาดเล็กที่ได้รับความนิยมเป็นอย่างมาก เนื่องจากมีราคาถูก ใช้งานง่าย และที่สำคัญมาพร้อมกับระบบ Wi-Fi ที่สามารถเชื่อมต่อกับระบบเครือข่ายได้ เพื่อสื่อสารข้อมูลผ่านระบบที่เรียกว่า **อินเทอร์เน็ต** โดย ESP8266 สามารถเป็นได้ทั้ง Access Point หรือ Web Server ขนาดเล็กก็ได้ และด้วยความสามารถดังกล่าว จึงไม่แปลกใจที่ ESP8266 จะเป็นที่นิยมในการนำมาเรียนรู้ หรือการทำงานที่ต้องการการสื่อสารข้อมูลผ่านระบบเครือข่ายอินเทอร์เน็ต

1.2.2 ลักษณะทางกายภาพของ ESP8266

ไมโครคอนโทรลเลอร์ ESP8266 ใช้ไมโครคอนโทรลเลอร์ ESP8266 ขนาด 32 บิต มีขนาดกว้าง 49 มม. ยาว 26 มม. มีระยะห่างระหว่างพอร์ต 2.54 มม. ความเร็ว 80 MHz เชื่อมต่อกับคอมพิวเตอร์ด้วย Micro USB แรงดันที่ใช้ในการทำงาน 3.3 โวลต์ สามารถรับแรงดันภายนอกผ่านพอร์ต Vin ระหว่าง 4.5 - 10 โวลต์ มีหน่วยความจำที่ใช้เก็บโปรแกรมที่ใช้ในการควบคุม ขนาด 4 MB มีหน่วยความจำที่ใช้เป็นพื้นที่สำหรับประมวลผล 64 KB มีพอร์ตดิจิทัลที่ใช้สำหรับสื่อสารข้อมูลกับอุปกรณ์ภายนอก 11 พอร์ต มีพอร์ตแอนะล็อกที่ใช้สำหรับสื่อสารข้อมูลกับอุปกรณ์ภายนอก 1 พอร์ต มีวงจรแปลงแอนะล็อกเป็นดิจิทัล (Analog-to-digital Converter : ADC) ที่สามารถวัดแรงดันได้ตั้งแต่ 0 - 3.3 โวลต์ มีพอร์ตสื่อสารข้อมูลแบบอนุกรม ได้แก่ UART หรือ Universal Asynchronous Receiver-transmitter จำนวน 1 ชุด SPI หรือ Serial Peripheral Interface จำนวน 1 ชุด และ I2C หรือ Inter-integrated Circuit จำนวน 1 ชุด มีโมดูล Wi-Fi มาตรฐาน 802.11 b/g/n และทนทานต่ออุณหภูมิระหว่าง -40 - 125 องศาเซลเซียส โดยมีลักษณะทางกายภาพดังรูปที่ 1.2



รูปที่ 1.2 ลักษณะทางกายภาพของ ESP8266

ข้อมูลเฉพาะและคุณสมบัติต่าง ๆ ของ ESP8266 มีดังนี้

- (A) ชิพ ESP8266 พร้อม 2.4 GHz Antenna สายอากาศย่านความถี่ 2.4 GHz
- (B) ช่องสำหรับเชื่อมต่อ ESP8266 กับคอมพิวเตอร์ ด้วยสาย Micro USB
- (C) ปุ่มรีเซ็ตบอร์ดเพื่อเริ่มทำงานใหม่
- (D) ใช้สำหรับอัปโหลดโปรแกรม
- (E) พอร์ตดิจิทัลสำหรับสื่อสารข้อมูลภายนอก ประกอบด้วย D0/GPIO16 D1/GPIO5 D2/GPIO4

D3/GPIO0 และ D4/GPIO2

- (F) พอร์ตดิจิทัลสำหรับสื่อสารข้อมูลภายนอก ประกอบด้วย D5/GPIO14 D6/GPIO12 D7/GPIO13 และ D8/GPIO15

- (G) 3V3 พอร์ตแรงดันไฟ 3.3 โวลต์ และ GND เป็นพอร์ตกราวด์

- (H) VIN พอร์ตรับแรงดันภายนอกสามารถรับแรงดัน ระหว่าง 4.5 - 10 โวลต์ได้ และ GND เป็นพอร์ตกราวด์

- (I) RX/TX เป็นพอร์ตที่ใช้รับส่งข้อมูลแบบอนุกรมกับอุปกรณ์อื่น

- (J) พอร์ตที่ใช้ในการสื่อสารข้อมูลแบบ SPI

- (K) พอร์ตที่ใช้ควบคุมการทำงาน เมื่อจ่ายไฟจะเป็นการกำหนดให้เป็น Active High

- (L) พอร์ตที่ใช้รีเซ็ตการทำงานใหม่ เมื่อต่อกับกราวด์จะเป็นการกำหนดให้เป็น Active Low

- (M) พอร์ตดิจิทัลสำหรับสื่อสารข้อมูลภายนอก ประกอบด้วย SD2/GPIO9 และ SD3/GPIO10

- (N) พอร์ตที่ถูกสงวนไว้สำหรับการใช้งานที่กำหนดไว้เท่านั้น

- (O) พอร์ตแอนะล็อกที่สามารถอ่านค่าได้ตั้งแต่ 0 - 1023

1.2.3 พอร์ต พิน หรือขา เรียกอะไรดี

เป็นเรื่องที่สร้างความสับสนให้แก่ผู้ที่เริ่มต้นศึกษาเกี่ยวกับไมโครคอนโทรลเลอร์ เนื่องจากมีหลายชื่อเรียก แล้วแต่วัฒนธรรมขององค์กรต่าง ๆ บ้างก็เรียกว่า **พอร์ต** บ้างก็เรียกว่า **พิน** หรือไม่ก็เรียกกันง่าย ๆ ว่า **ขา** ซึ่งไม่ว่าจะเรียกว่าอะไรก็ล้วนแต่มีความหมายเหมือนกัน ดังนั้น ในหนังสือเล่มนี้จะขอเรียกว่า **พอร์ต** และพอร์ตก็หมายถึง **พอร์ตอินพุตและเอาต์พุต** ที่เตรียมไว้สำหรับสื่อสารข้อมูลกับอุปกรณ์ต่าง ๆ ภายนอกนั่นเอง

1.2.4 การติดตั้งโปรแกรมสำหรับใช้งาน ESP8266

ในการพัฒนาโปรแกรมเพื่อควบคุมไมโครคอนโทรลเลอร์ ESP8266 โปรแกรมที่ใช้ในการพัฒนา คือ Arduino IDE ที่ใช้สำหรับเขียนคำสั่ง (Sketch Code) ตรวจสอบความถูกต้องของโครงสร้างภาษา (Verify/Compile) และการอัปโหลดโปรแกรมลงใน ESP8266 (Upload) ตรวจสอบความถูกต้องของโปรแกรมคำสั่งที่ใส่ลงไป ESP8266 (Check) และปรับปรุงแก้ไขคำสั่งที่ใส่ลงไปหากการทำงานไม่ถูกต้องตามที่ต้องการ

1.2.5 การเตรียมความพร้อม

ก่อนจะเริ่มการพัฒนางานต่าง ๆ ที่เกี่ยวกับ ESP8266 สิ่งแรกที่เราควรคำนึงถึง คือ การเตรียมความพร้อม ทั้งด้านฮาร์ดแวร์และซอฟต์แวร์ โดยมีรายละเอียดที่ต้องเตรียมดังนี้

1. การดาวน์โหลดและติดตั้งโปรแกรม
2. การตั้งค่าโปรแกรมให้ Arduino IDE สามารถเขียนคำสั่งเพื่อสั่งงาน ESP8266 ได้
3. ตรวจสอบและติดตั้งไดรเวอร์เกี่ยวกับ ESP8266
4. ตรวจสอบการเชื่อมต่อระหว่างคอมพิวเตอร์กับ ESP8266
5. ตรวจสอบว่า Arduino IDE สามารถเชื่อมต่อกับ ESP8266 ได้หรือไม่
6. ดาวน์โหลดและติดตั้งไลบรารี (Library) หรือชุดคำสั่งเพิ่มเติม

1.3 ไฟฟ้าและแหล่งจ่ายไฟฟ้า

ไฟฟ้าและแหล่งจ่ายไฟฟ้า เป็นปัจจัยสำคัญที่ทำให้ระบบสมองกลฝังตัวทำงานได้อย่างต่อเนื่องและมีประสิทธิภาพ การเลือกใช้แหล่งจ่ายไฟฟ้าที่เหมาะสมและการจัดการพลังงานที่ดี จะช่วยให้ระบบทำงานได้อย่างเสถียรและปลอดภัย การเรียนรู้เกี่ยวกับหลักการเบื้องต้นของไฟฟ้า แหล่งจ่ายไฟฟ้า และการป้องกันการรบกวนทางไฟฟ้า จะช่วยให้ผู้เรียนสามารถออกแบบและพัฒนาระบบสมองกลฝังตัวที่มีความน่าเชื่อถือและประสิทธิภาพสูงได้

1.3.1 พื้นฐานไฟฟ้าสำหรับระบบสมองกลฝังตัว

พื้นฐานไฟฟ้าสำหรับระบบสมองกลฝังตัว เป็นสิ่งสำคัญที่ต้องเข้าใจ เนื่องจากระบบสมองกลฝังตัว มีการทำงานที่เกี่ยวข้องกับวงจรไฟฟ้าและการจ่ายพลังงานอย่างใกล้ชิด การเข้าใจพื้นฐานไฟฟ้า จะช่วยให้สามารถออกแบบ ติดตั้ง และแก้ไขปัญหาของระบบสมองกลฝังตัวได้อย่างมีประสิทธิภาพ แรงดันไฟฟ้าเป็นแรงที่ผลักดันให้กระแสไฟฟ้าไหลผ่านวงจรไฟฟ้า มีหน่วยเป็นโวลต์ (Volt หรือ V) สำหรับระบบสมองกลฝังตัวมักจะใช้แรงดันไฟฟ้าต่ำ เช่น 3.3 โวลต์ หรือ 5 โวลต์ ซึ่งเป็นแรงดันที่เหมาะสมกับไมโครคอนโทรลเลอร์และอุปกรณ์อิเล็กทรอนิกส์อื่น

1.3.2 การเลือกและการใช้งานแหล่งจ่ายไฟฟ้า

การเลือกและการใช้งานแหล่งจ่ายไฟฟ้าสำหรับระบบสมองกลฝังตัว เป็นขั้นตอนสำคัญที่มีผลโดยตรงต่อการทำงานของระบบ เนื่องจากแหล่งจ่ายไฟฟ้าที่ไม่เหมาะสม อาจทำให้ระบบทำงานไม่เสถียรหรือเกิดความเสียหายต่ออุปกรณ์ต่าง ๆ ในระบบได้ ต่อไปนี้เป็นแนวทางในการเลือกและการใช้งานแหล่งจ่ายไฟฟ้าสำหรับระบบสมองกลฝังตัว

1. พิจารณาความต้องการของระบบ (Power Requirements)

1.1 แรงดันไฟฟ้า (Voltage) ตรวจสอบว่าอุปกรณ์สมองกลฝังตัวของคุณต้องการแรงดันไฟฟ้าเท่าใด (เช่น 3.3 โวลต์ 5 โวลต์ หรือ 12 โวลต์) การเลือกแหล่งจ่ายไฟฟ้าที่มีแรงดันไฟฟ้าที่เหมาะสมเป็นสิ่งสำคัญ เพื่อให้แน่ใจว่าอุปกรณ์จะทำงานได้อย่างถูกต้อง

1.2 กระแสไฟฟ้า (Current) คำนวณกระแสไฟฟ้าที่ระบบต้องการโดยรวม ซึ่งอาจต้องพิจารณาจากอุปกรณ์ทั้งหมดที่ต่ออยู่ในวงจร แหล่งจ่ายไฟฟ้าควรมีกระแสไฟฟ้าเพียงพอที่จะรองรับอุปกรณ์ทั้งหมด มิฉะนั้นอุปกรณ์อาจทำงานไม่เสถียรหรือแหล่งจ่ายไฟฟ้าอาจร้อนเกินไปและเสียหายได้

2. ประเภทของแหล่งจ่ายไฟฟ้า (Types of Power Supply)

2.1 แบตเตอรี่ (Battery) เป็นแหล่งจ่ายไฟฟ้าแบบพกพาที่มักใช้ในระบบที่ต้องการเคลื่อนที่หรือใช้งานในพื้นที่ที่ไม่มีแหล่งจ่ายไฟฟ้าคงที่ แบตเตอรี่มีหลายประเภท เช่น แบตเตอรี่ลิเธียมไอออน (Li-ion) แบตเตอรี่ลิเธียมโพลีเมอร์ (Li-Po) และแบตเตอรี่ AA

2.2 อะแดปเตอร์ (AC to DC Adapter) ใช้ในกรณีที่มีแหล่งจ่ายไฟฟ้า AC (เช่น ไฟฟ้าบ้าน) และต้องการแปลงเป็นไฟฟ้า DC ที่เหมาะสมกับอุปกรณ์สมองกลฝังตัว อะแดปเตอร์ควรเลือกตามแรงดันไฟฟ้าและกระแสไฟฟ้าที่ระบบต้องการ

2.3 แหล่งจ่ายไฟฟ้าแบบสวิตซิง (Switching Power Supply) แหล่งจ่ายไฟฟ้าชนิดนี้มีขนาดเล็กและประสิทธิภาพสูง สามารถปรับแรงดันไฟฟ้าและกระแสไฟฟ้าได้ตามต้องการ เหมาะสำหรับใช้งานกับระบบที่ต้องการความเสถียรในการจ่ายไฟฟ้า

3. การควบคุมแรงดันไฟฟ้า (Voltage Regulation) ช่วยให้แรงดันไฟฟ้าที่ส่งไปยังอุปกรณ์สมองกลฝังตัวมีความเสถียรและสม่ำเสมอ แม้ว่าแรงดันไฟฟ้าจากแหล่งจ่ายไฟฟ้าจะมีการเปลี่ยนแปลง ตัวควบคุมแรงดันไฟฟ้าที่นิยมใช้ ได้แก่ LM7805 สำหรับแรงดันไฟฟ้า 5 โวลต์ หรือ LM1117 สำหรับแรงดันไฟฟ้า 3.3 โวลต์

4. การป้องกันวงจร (Circuit Protection)

4.1 ฟิวส์ (Fuse) ใช้ฟิวส์ในวงจรเพื่อป้องกันการใช้กระแสไฟฟ้าเกิน หากมีการใช้กระแสไฟฟ้าเกิน ฟิวส์จะขาดและตัดการจ่ายไฟฟ้า ทำให้วงจรและอุปกรณ์ได้รับการป้องกัน

4.2 ไดโอดกั้นไฟฟ้าย้อน (Diode) ใช้ไดโอดเพื่อป้องกันไฟฟ้าย้อนกลับ ในกรณีที่แหล่งจ่ายไฟฟ้าถูกเชื่อมต่อแบบกลับหัว ซึ่งอาจทำให้อุปกรณ์เสียหายได้

5. การจัดการพลังงาน (Power Management) ระบบสมองกลฝังตัวบางประเภทอาจต้องการการจัดการพลังงานที่ดี เช่น การปรับระดับการใช้งานพลังงานในโหมดต่าง ๆ เพื่อประหยัดพลังงาน แหล่งจ่ายไฟฟ้าที่มีการจัดการพลังงานที่ดี สามารถยืดอายุการใช้งานของแบตเตอรี่และลดการใช้พลังงานในระบบได้

6. การต่อสายดิน (Grounding) การต่อสายดินอย่างเหมาะสมเป็นสิ่งสำคัญในการป้องกันการเกิดสัญญาณรบกวนและความผิดพลาดในระบบ การเชื่อมต่อจุดต่าง ๆ ในวงจรเข้ากับจุดสายดินเดียวกัน (Single Point Grounding) จะช่วยลดปัญหานี้

7. การทดสอบและตรวจสอบ (Testing and Verification) หลังจากเลือกและติดตั้งแหล่งจ่ายไฟฟ้าแล้ว ควรทำการทดสอบระบบเพื่อให้แน่ใจว่าแหล่งจ่ายไฟฟ้าสามารถรองรับการทำงานของอุปกรณ์ทั้งหมดได้อย่างปลอดภัยและมีประสิทธิภาพ การตรวจสอบแรงดันไฟฟ้าและกระแสไฟฟ้าด้วยมัลติมิเตอร์เป็นขั้นตอนที่ควรทำอย่างละเอียด

8. การกระจายพลังงาน (Power Distribution) สำหรับระบบที่มีการใช้งานหลายอุปกรณ์ ควรมีการจัดการการกระจายพลังงานอย่างเหมาะสม เพื่อให้แน่ใจว่าแต่ละส่วนของระบบได้รับพลังงานที่เพียงพอ การใช้บอร์ดจ่ายไฟฟ้า (Power Distribution Board) หรือการแยกพอร์ตจ่ายไฟฟ้า จะช่วยให้การกระจายพลังงานเป็นไปอย่างมีประสิทธิภาพ

1.4

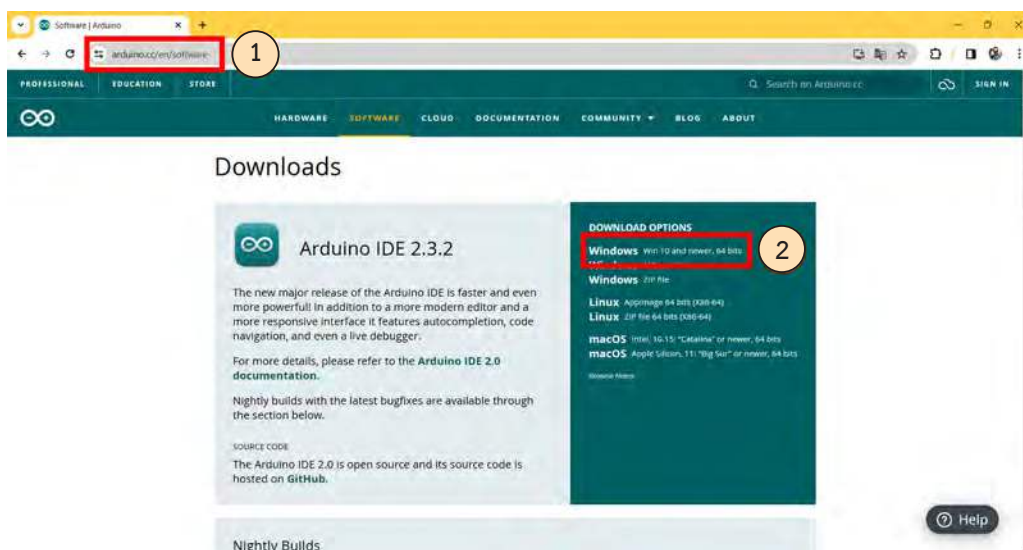
การเขียนโปรแกรมเบื้องต้นสำหรับ ESP8266

ESP8266 เป็นไมโครคอนโทรลเลอร์ที่มีความสามารถในการเชื่อมต่อกับเครือข่าย Wi-Fi ทำให้เป็นที่นิยมในการพัฒนาโครงการ IoT การเขียนโปรแกรมเบื้องต้นสำหรับ ESP8266 จะช่วยให้ผู้เรียนสามารถควบคุมการทำงานของอุปกรณ์ต่าง ๆ ผ่านการเชื่อมต่อ Wi-Fi ได้ ผู้เรียนจะได้เรียนรู้การเขียนโค้ดเบื้องต้น การอัปโหลดโค้ดเข้าสู่ ESP8266 และการทดสอบการทำงานของโปรแกรม การเรียนรู้เหล่านี้จะเป็นพื้นฐานสำคัญในการพัฒนาโครงการ IoT ที่มีความซับซ้อนมากขึ้นในอนาคต

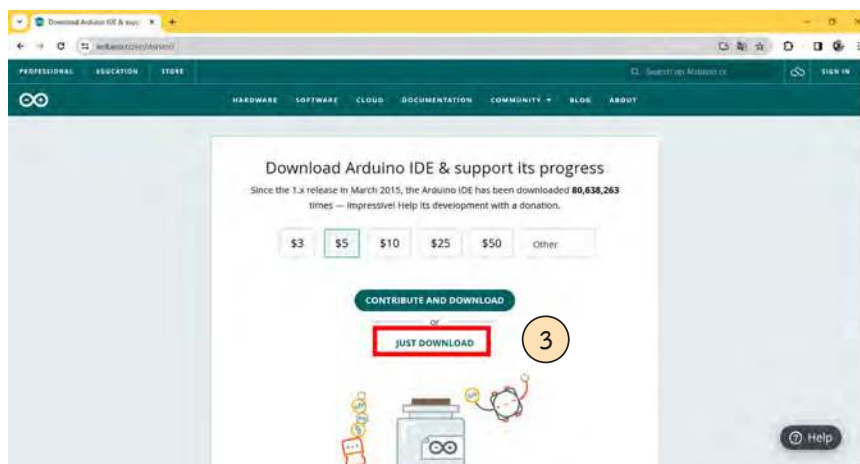
1.4.1 การติดตั้งและการใช้งาน Arduino IDE

การดาวน์โหลดและติดตั้งโปรแกรมสามารถทำได้ตามขั้นตอนดังต่อไปนี้

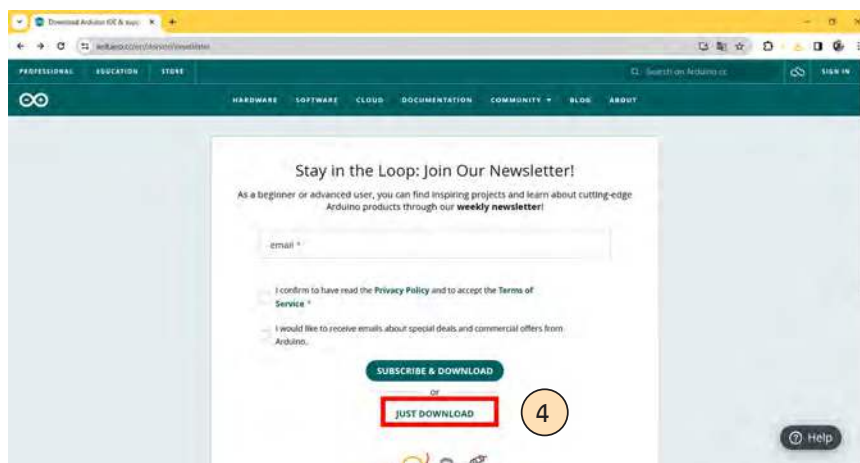
1. ไปที่เว็บไซต์ <https://www.arduino.cc/en/software>
2. เลือกดาวน์โหลดเวอร์ชันให้ตรงกับระบบปฏิบัติการที่ใช้



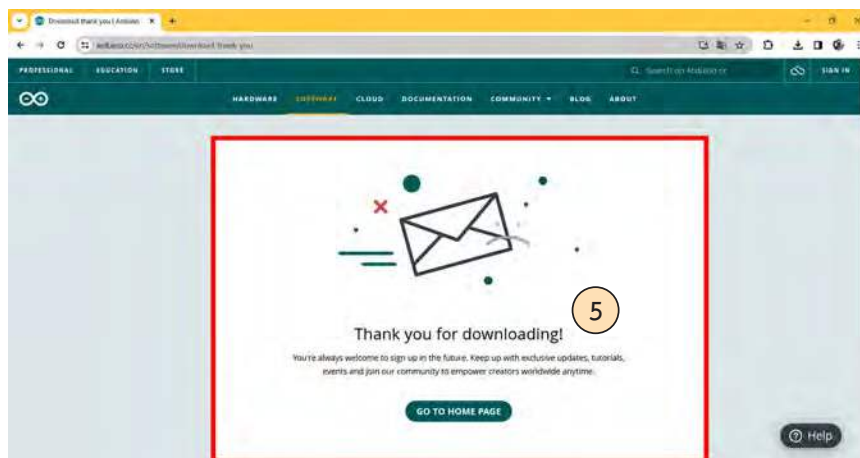
3. คลิกที่ JUST DOWNLOAD



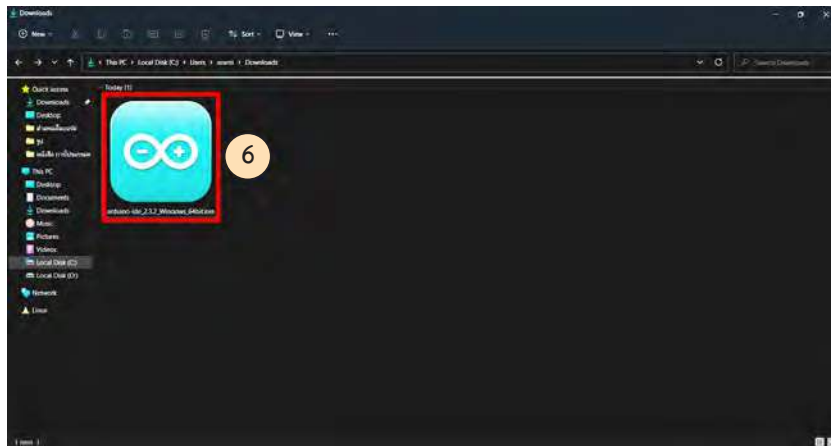
4. คลิกที่ JUST DOWNLOAD อีกครั้ง



5. เมื่อปรากฏหน้านี้ก็แสดงว่าสามารถโหลดได้ พร้อมเลือกที่เก็บตัวลงโปรแกรม



6. เมื่อดาวน์โหลดเสร็จสิ้น จะปรากฏโปรแกรมดังรูป

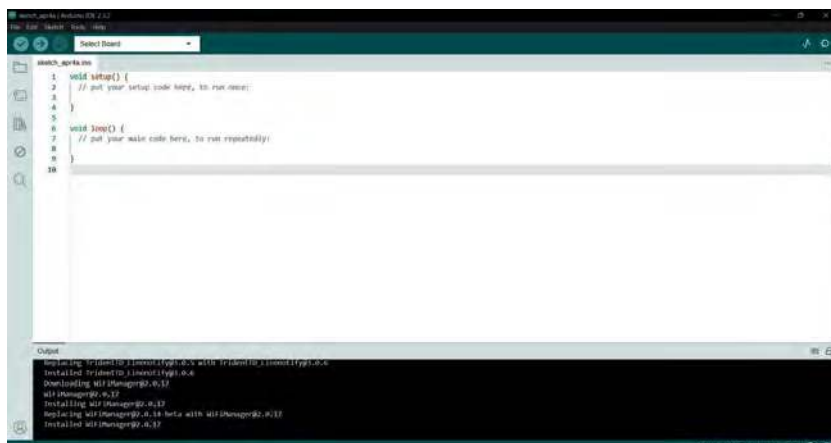


7. ดับเบิลคลิก arduino-ide_2.3.2_Windows_64bit.exe เพื่อติดตั้งโปรแกรม

8. เมื่อติดตั้งเสร็จสิ้น คลิก Finish

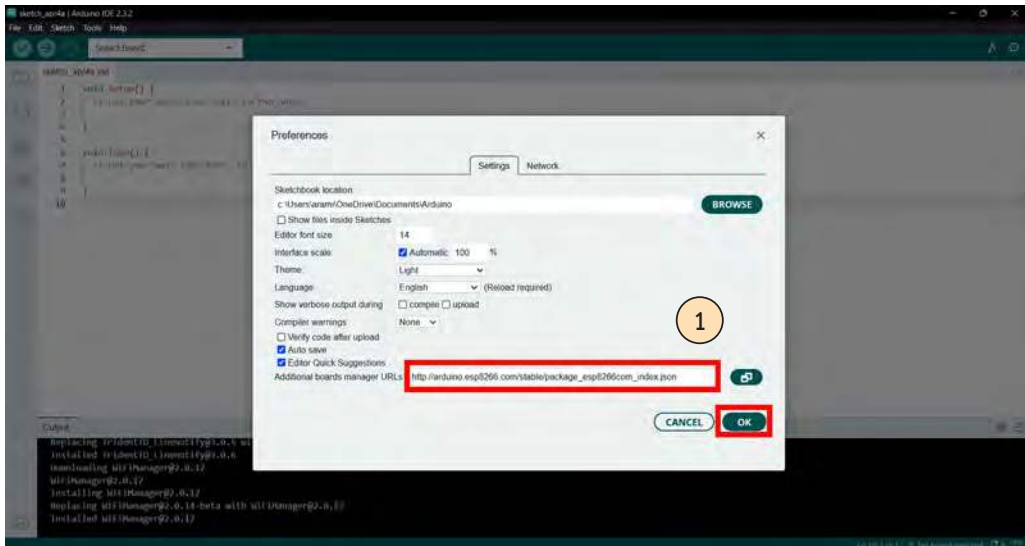


9. จะปรากฏหน้าต่างโปรแกรม Arduino IDE ที่พร้อมเขียนคำสั่ง แต่โปรแกรมจะยังไม่สามารถเขียนคำสั่งเพื่อควบคุม ESP8266 ได้

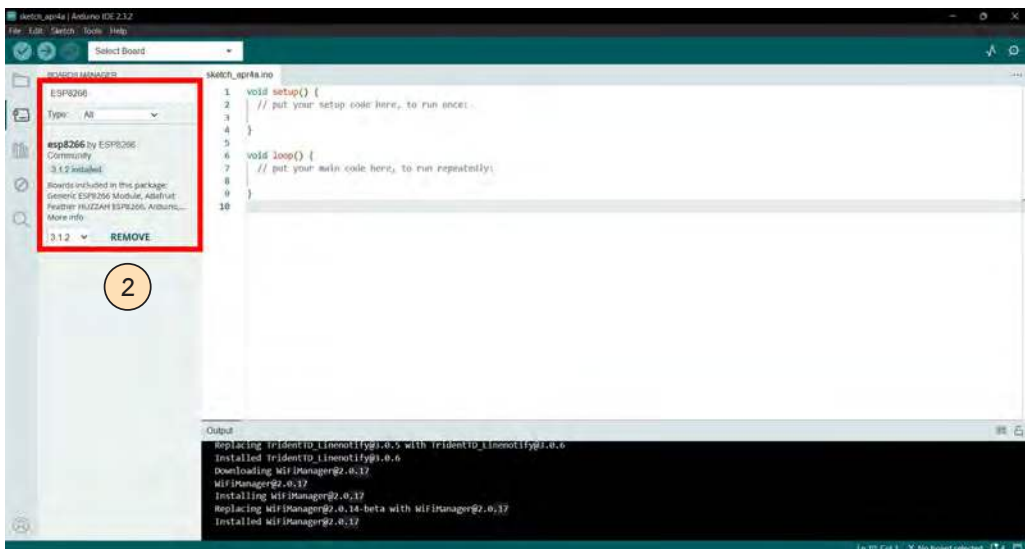


1.4.2 การตั้งค่าโปรแกรม

1. ไปที่เมนู File > Preferences... > Additional Boards Manager URLs จากนั้นใส่ http://arduino.esp8266.com/stable/package_esp8266com_index.json ลงไป แล้วคลิก OK

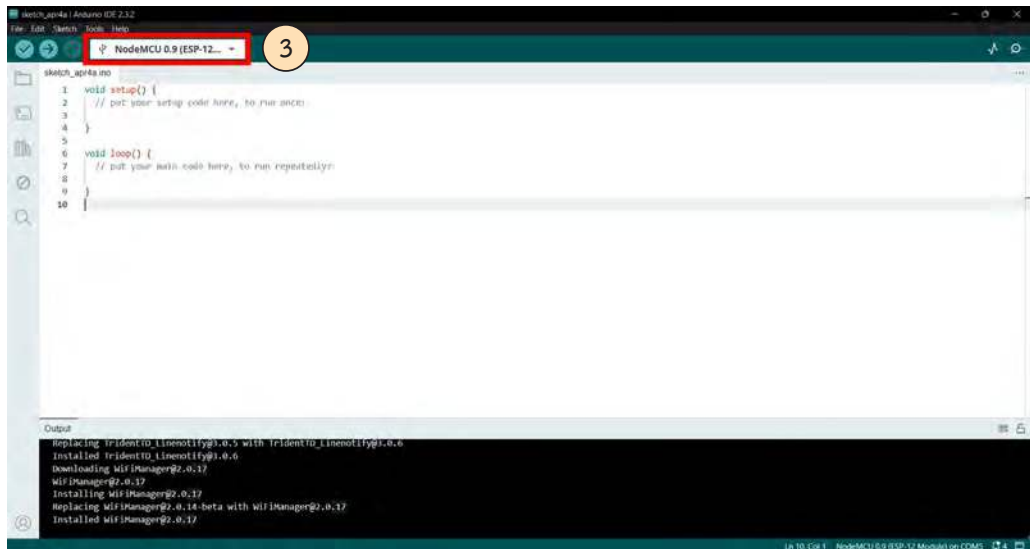


2. Tools > Board > Boards Manager... และค้นหา esp8266" จากนั้นคลิก Install เพื่อติดตั้งเมนู ESP8266 ใน Arduino IDE



3. เลือกบอร์ด ESP8266 ที่เมนู Tools > Board > ESP8266 Boards (3.1.2) > NodeMCU 0.9 (ESP-12 Module)

4. เลือกพอร์ตที่เมนู Tools > Port > ComXX โดยที่ XX หมายถึงตัวเลขใด ๆ เนื่องจาก ESP8266 แต่ละตัวจะมีเลข Com Port เป็นของตัวเอง



1.4.3 การทดสอบเบื้องต้น

ก่อนทำการทดสอบ จะต้องรู้จักคำสั่งต่าง ๆ ที่ใช้ตรวจสอบความถูกต้องของโครงสร้างภาษา (Verify/Compile) และการอัปโหลดโปรแกรมลงใน ESP8266 (Upload) ตรวจสอบความถูกต้องของโปรแกรมคำสั่งที่ใส่ลงไป (Check) และการตรวจสอบผลการทำงานด้วยหน้าจอ Serial Monitor หรือแบบกราฟ โดยมีรายละเอียดดังนี้

1. Verify/Compile คือ การตรวจสอบความถูกต้องของโครงสร้างและรูปแบบของภาษา
2. Upload คือ การอัปโหลดโปรแกรมลงใน ESP8266
3. เปิด Serial Monitor เพื่อตรวจสอบการทำงานจากคำสั่งที่ใส่ลงไป ESP8266
4. เปิดกราฟ เพื่อตรวจสอบการทำงานจากคำสั่งที่ใส่ลงไป ESP8266
5. ส่วนของการพิมพ์ชุดคำสั่ง

