



คู่มือการใช้งาน Farm1

- เอาต์พุต Contact 5A/NO 4 ช่อง
- อินพุตดิจิทัล 2 ช่อง
- อินพุตแอนะล็อก 2 ช่อง
- ต่ออุปกรณ์เสริมผ่าน RS485 และ I²C
- ใช้แรงดันไฟฟ้าได้ 5V ถึง 24V
- ใช้ ESP32 เป็นตัวประมวลผล
- เชื่อมต่ออินเทอร์เน็ตผ่าน WiFi
- มีวงจรนับเวลา (RTC) ในตัว
- ขนาด 110 x 120 x 40 มิลลิเมตร

Farm1 กล่องควบคุมฟาร์มระบบ IoT ใช้ไมโครคอนโทรลเลอร์ ESP32 มาพร้อม WiFi และบลูทูธ ติดตั้งบนบอร์ด Farm1 รองรับการทำงานนอกผ่านช่องเอาต์พุต 5A/220V จำนวน 4 ช่อง ต่ออินพุตดิจิทัล 2 ช่อง อินพุตแอนะล็อก 2 ช่อง และ I/O จำนวน 1 ช่อง มีช่องต่อ I2C และ RS485 ใช้งานได้กับเซ็นเซอร์ทุกประเภท มีวงจรนับเวลา (RTC) มาในตัว มาพร้อมกล่องติดตั้งบนราวปีกนก

เวอร์ชัน	รายละเอียดการแก้ไข	เผยแพร่
1.0	เพิ่มข้อมูลทั่วไปเกี่ยวกับ Farm1 และการเขียนโปรแกรมสั่งงานเบื้องต้น	18/11/2564
1.1	เพิ่มวิธีการใช้งานกับ HandySense	25/11/2564

สารบัญ

สร้างระบบควบคุมฟาร์ม IoT อย่างรวดเร็วด้วย HandySense	4
การต่อเซ็นเซอร์เข้ากับบอร์ด Farm1	4
การอัปเดตเฟิร์มแวร์ HandySense	5
การเพิ่มอุปกรณ์เข้าระบบ HandySense	7
ข้อมูลด้านฮาร์ดแวร์และวงจร	13
คุณสมบัติทางไฟฟ้า	13
ตำแหน่งขาต่อใช้งาน	14
การจ่ายไฟ	14
ช่องเอาต์พุต	16
ช่องอินพุตดิจิทัล	18
ช่องอินพุตแอนะล็อก	19
ช่อง I/O	20
ช่อง I ² C	22
ช่อง RS485	23
วงจรนับเวลา (RTC)	24
การเขียนโปรแกรมสั่งงานด้วย Arduino IDE	26
ติดตั้งโปรแกรม Arduino IDE	26
ติดตั้งไดรเวอร์ FT231X	28
การเพิ่มบอร์ด ESP32 ลงโปรแกรม Arduino IDE	31
การเชื่อมต่อ Farm1 เข้ากับเครื่องคอมพิวเตอร์	33
ทดสอบอัปโหลดโปรแกรมไฟกระพริบ	33
เขียนโปรแกรมสั่งงานช่องเอาต์พุต	36

เขียนโปรแกรมอ่านชั่งอินพุตดิจิทัล	37
เขียนโปรแกรมอ่านชั่งอินพุตแอนะล็อก	39
เขียนโปรแกรมชั่ง I/O อ่านค่าอุณหภูมิจาก DS18B20	41
เขียนโปรแกรมชั่ง I ² C อ่านค่าอุณหภูมิและความชื้นจาก SHT30	44
เขียนโปรแกรมชั่ง RS485 อ่านค่าอุณหภูมิและความชื้นจาก XY-MD02	45
เขียนโปรแกรมตั้งเวลาและอ่านค่าเวลาจากวงจรนับเวลา (RTC)	46
คำแนะนำการติดตั้ง	46
แนวทางการแก้ไขปัญหา	47

สร้างระบบควบคุมฟาร์ม IoT อย่างรวดเร็วด้วย HandySense

HandySense เป็นแพลตฟอร์มระบบควบคุมฟาร์มประกอบด้วยฮาร์ดแวร์และซอฟต์แวร์ เปิดเป็น Open Source เปิดให้ผู้สนใจนำไปพัฒนาต่อได้ บอร์ด Farm1 สามารถใช้ร่วมกับระบบ Cloud ของ HandySense ผ่านเฟิร์มแวร์ที่ปรับแต่งมาโดยเฉพาะ โดยมีข้อกำหนดสำคัญที่ต้องใช้ฮาร์ดแวร์ เซ็นเซอร์ และการต่อสาย ในรูปแบบที่ผู้ผลิตกำหนดเท่านั้น

การต่อเซ็นเซอร์เข้ากับบอร์ด Farm1

อุปกรณ์ด้านอิเล็กทรอนิกส์ที่จำเป็นต้องใช้ มีดังนี้

1. บอร์ด Farm1



2. เซ็นเซอร์วัดอุณหภูมิและความชื้น SHT30



3. เซ็นเซอร์วัดความชื้นในดิน



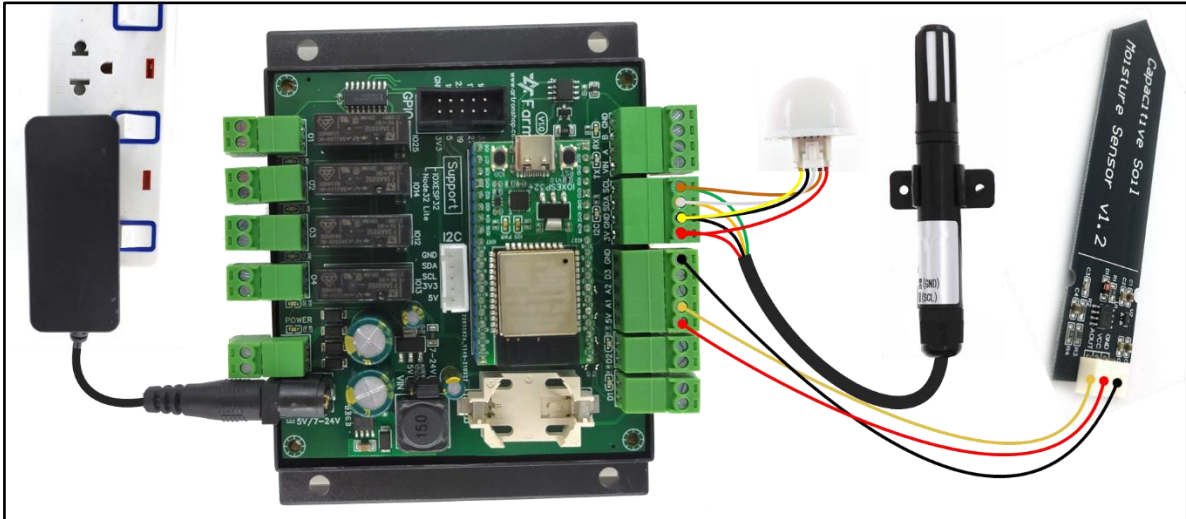
4. เซ็นเซอร์วัดแสง (BH1750)



5.อะแดปเตอร์ 12V 1A หัว DC Jack



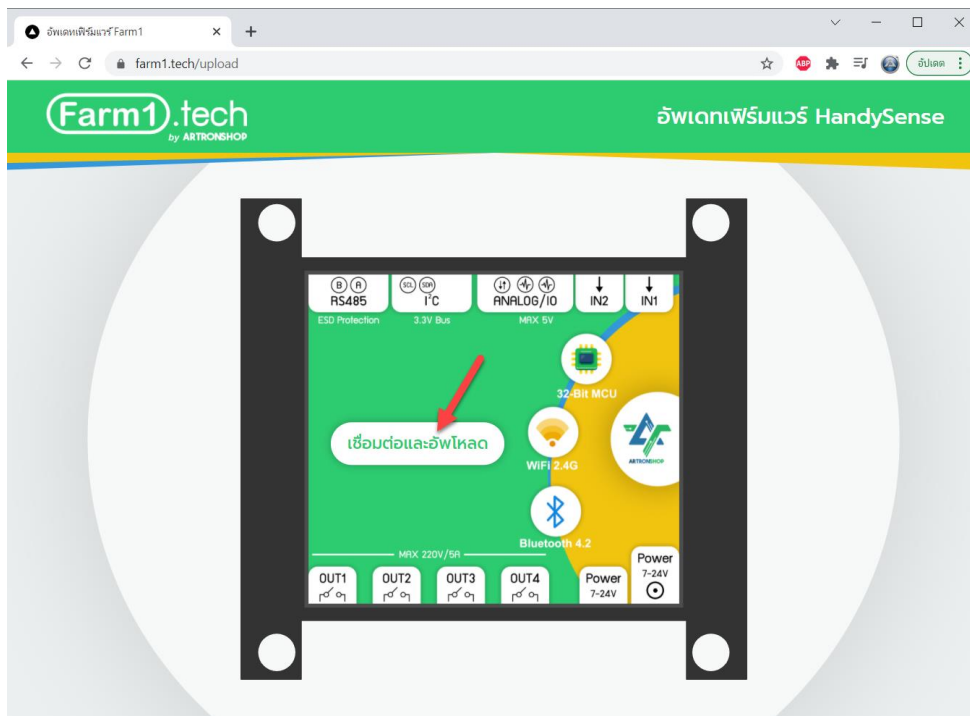
ต่ออุปกรณ์เข้าด้วยกันดังนี้



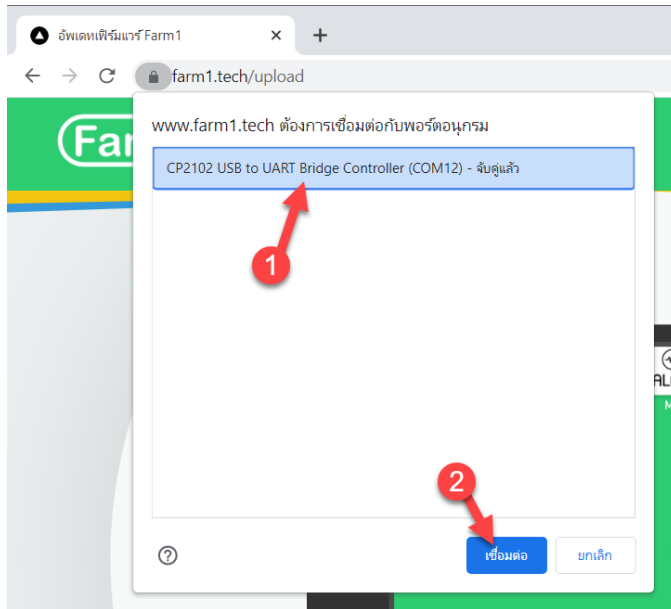
การอัปเดตเฟิร์มแวร์ HandySense

เปิดฝากล่อง Farm1 ออกมา แล้วใช้สาย USB-C เชื่อมต่อเข้ากับคอมพิวเตอร์ตามหัวข้อ [การเชื่อมต่อ Farm1 เข้ากับเครื่องคอมพิวเตอร์](#) ตัดตั้งไดรเวอร์ตามหัวข้อ [ติดตั้งไดรเวอร์ FT231X](#)

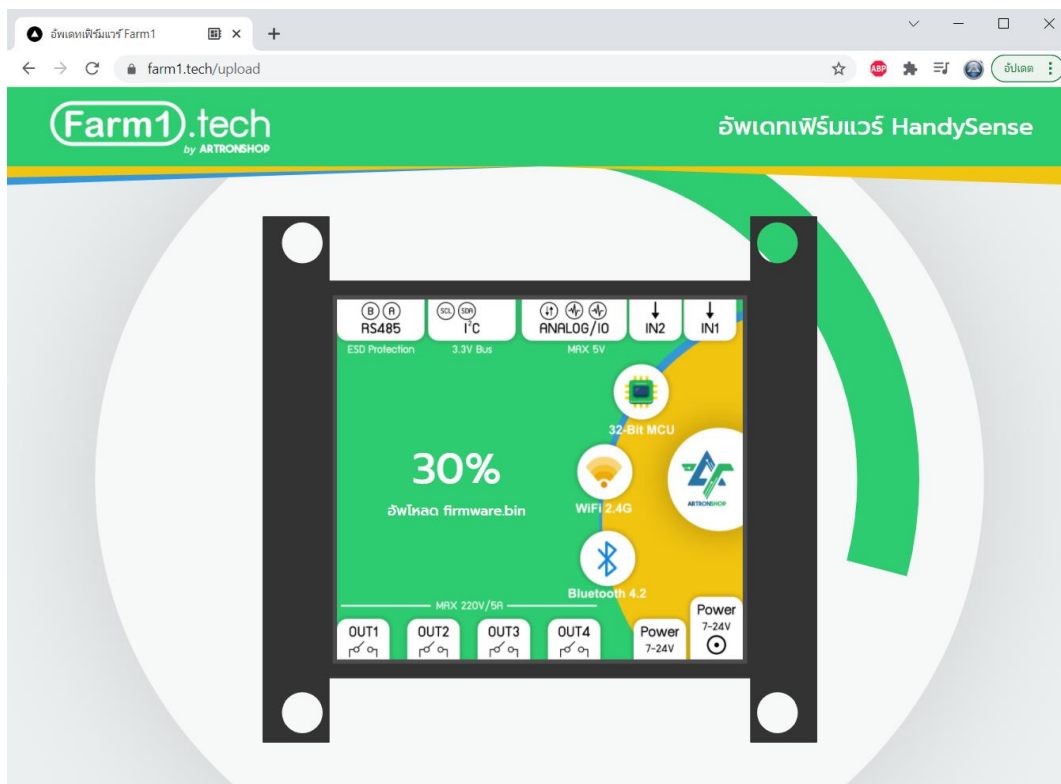
อัปเดตเฟิร์มแวร์ HandySense ให้เป็นเวอร์ชันล่าสุด โดยใช้โปรแกรม Google Chrome หรือ Microsoft Edge เวอร์ชันล่าสุด เข้าไปที่ <https://www.farm1.tech/upload> กดปุ่ม เชื่อมต่อและอัปเดต



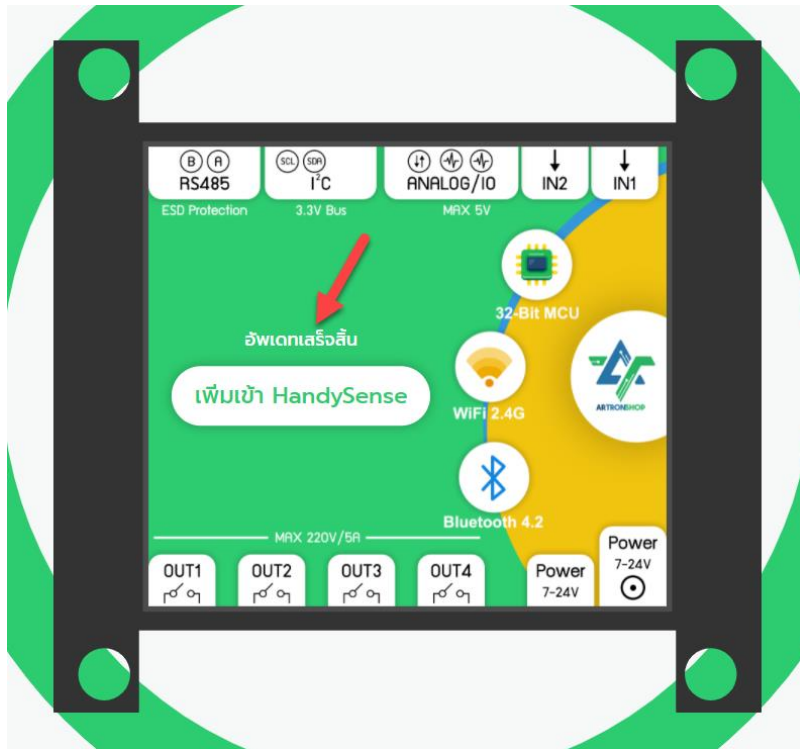
เลือกพอร์ตของบอร์ด แล้วกดปุ่ม เชื่อมต่อ



รออัปเดตเฟิร์มแวร์ชักรุ่น

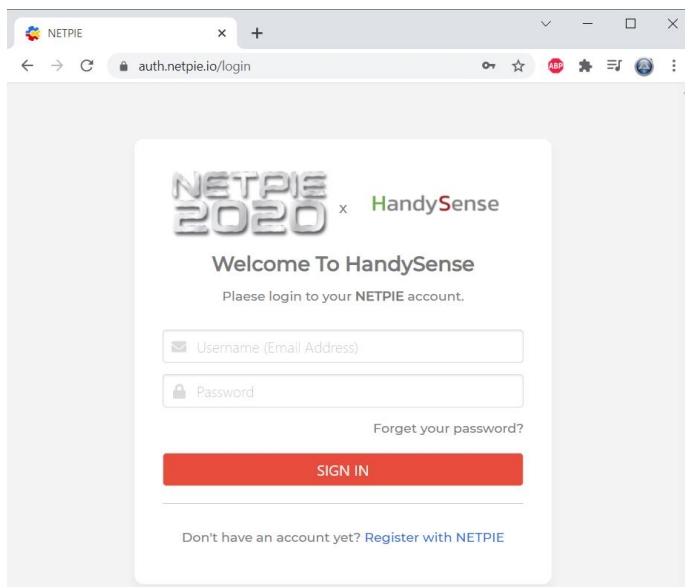


เมื่อเสร็จแล้วจะมีข้อความแจ้ง อัปเดตเฟิร์มแวร์ (ดังรูป)



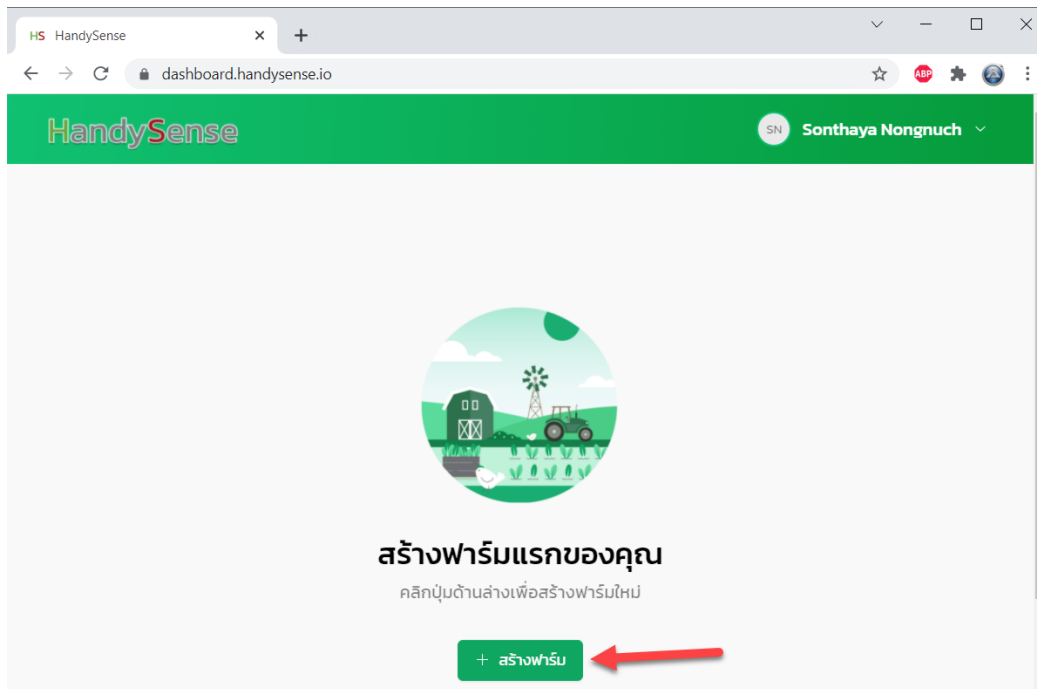
การเพิ่มอุปกรณ์เข้าระบบ HandySense

เข้าไปที่ dashboard.handysense.io จากนั้นสมัครสมาชิก หรือล็อกอินตามขั้นตอน



จากนั้นระบบจะพาเข้ามาหน้าแรก ซึ่งเป็นหน้าจัดการฟาร์ม กดปุ่ม สร้างฟาร์ม

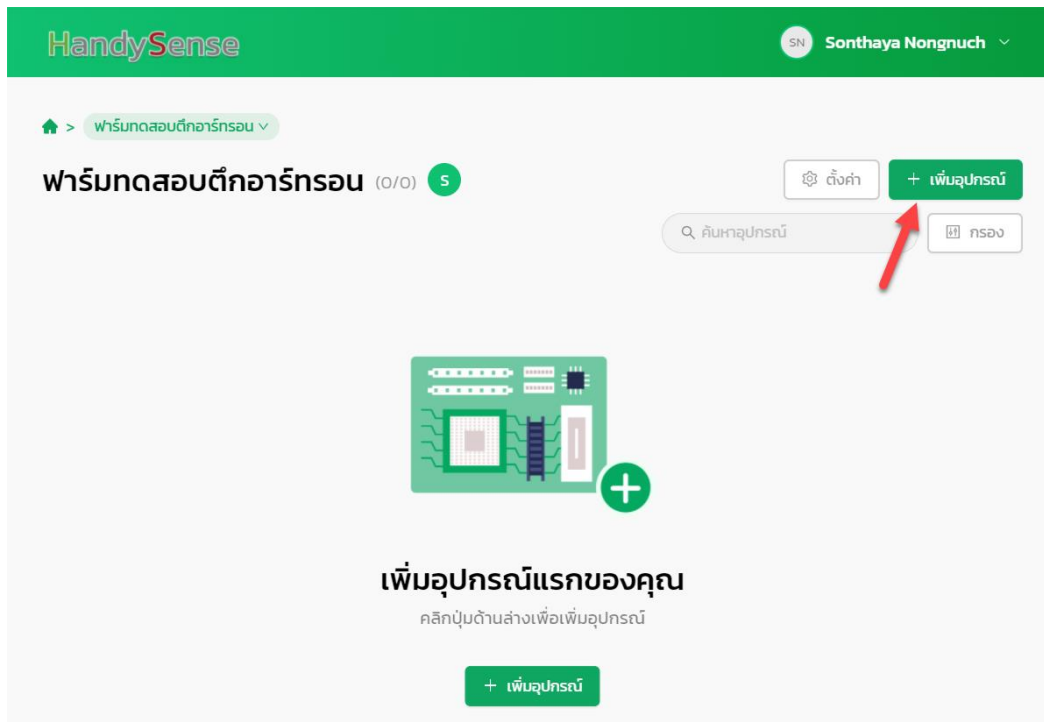
■ HandySense ออกแบบระบบให้ต้องมีฟาร์มก่อน จากนั้นจึงจะเพิ่มอุปกรณ์ลงในฟาร์มได้ ใน 1 ฟาร์ม
เพิ่มได้หลายอุปกรณ์



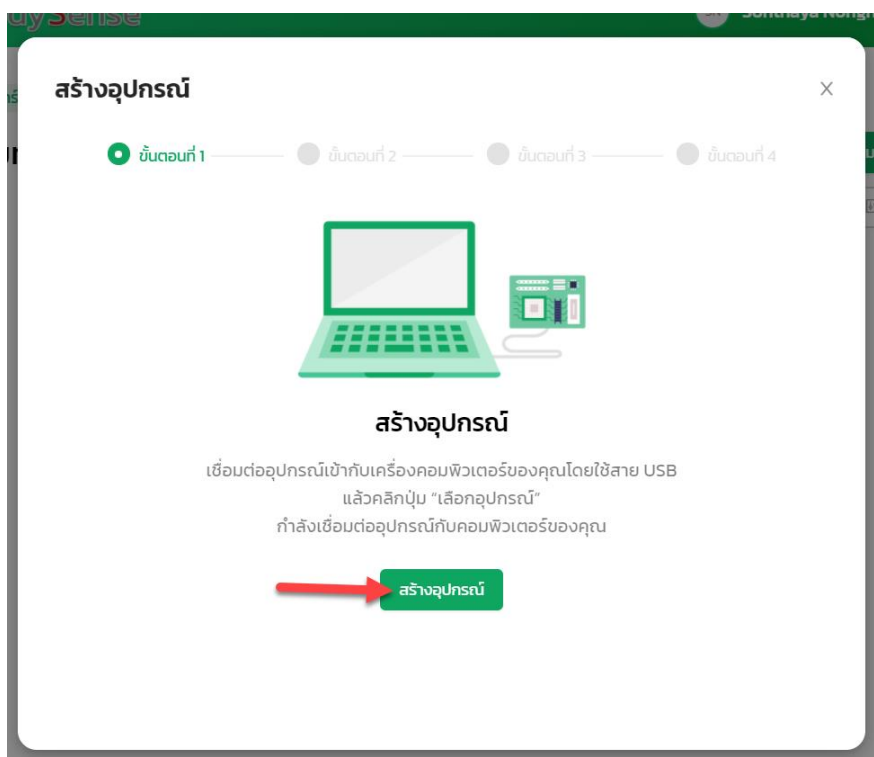
ในหน้าต่างสร้างฟาร์ม ให้ตั้งชื่อฟาร์ม (ห้ามมีเว้นวรรค) ใส่รายละเอียด (เว้นว่างได้) จากนั้นกดปุ่ม สร้าง

The screenshot shows a modal form titled 'สร้างฟาร์ม' (Create Farm). It has two input fields: 'ชื่อ' (Name) with the text 'ฟาร์มทดสอบดีอาร์ทรอน' (Farm Test ARTRON) and 'รายละเอียด' (Details) with the text 'รายละเอียด'. At the bottom, there are two buttons: 'ยกเลิก' (Cancel) and 'สร้าง' (Create). Red arrows and numbers 1, 2, and 3 point to the name field, the details field, and the 'สร้าง' button respectively.

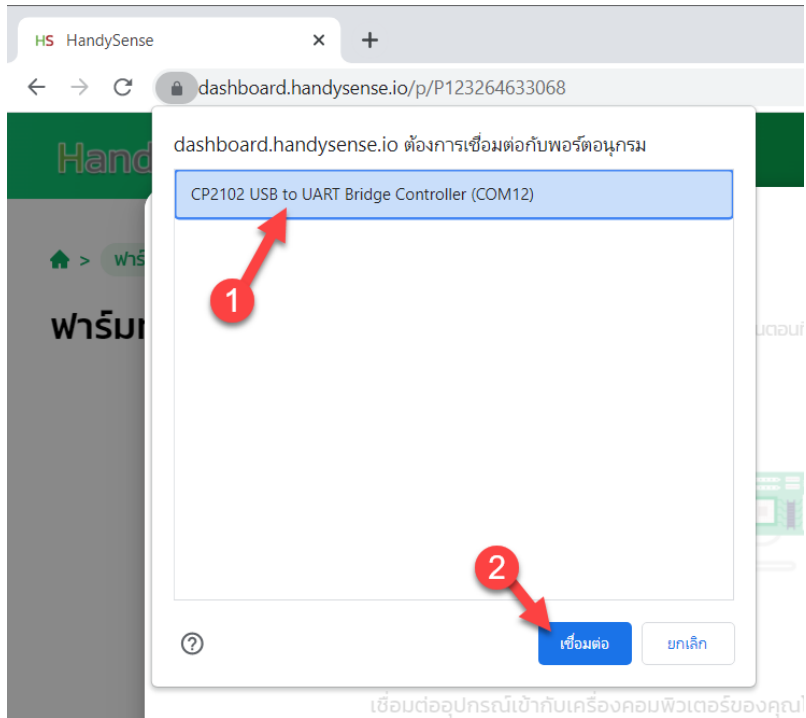
ระบบจะพาเข้ามาหน้าฟาร์มที่สร้างอัตโนมัติ ให้กดปุ่ม เพิ่มอุปกรณ์ เพื่อเริ่มเพิ่มอุปกรณ์ทันที



หน้าต่างสร้างอุปกรณ์จะแสดงขึ้นมา ในขั้นตอนที่ 1 เป็นแจ้งเตือนให้เชื่อมต่ออุปกรณ์กับเครื่องคอมพิวเตอร์ ให้กดปุ่ม สร้างอุปกรณ์

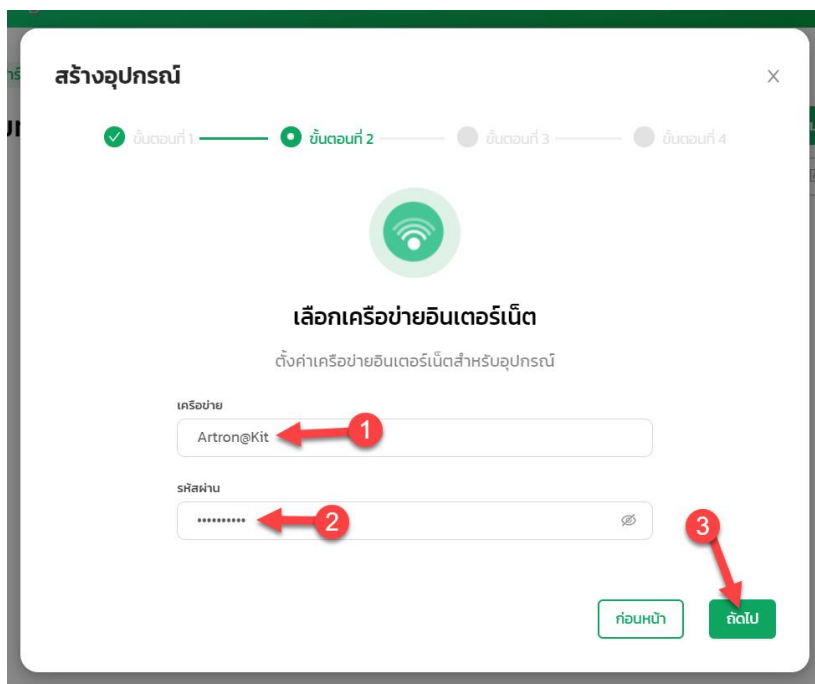


เลือกพอร์ตของอุปกรณ์ แล้วกดปุ่ม เชื่อมต่อ



ขั้นตอนที่ 2 ตั้งค่าการเชื่อมต่อ WiFi ให้อุปกรณ์ แล้วกดปุ่ม ถัดไป

- Farm1 รอรับการเชื่อมต่อ WiFi 2.4GHz เท่านั้น
- ถ้า WiFi ไม่มีรหัสผ่าน ชอว์รหัสผ่านให้เร้ากันว่าไว้



ขั้นตอนที่ 3 ตั้งชื่อชุดอุปกรณ์ ใส่รายละเอียด (เว้นว่างได้) แล้วกดปุ่ม ถัดไป

สร้างอุปกรณ์

✓ ขั้นตอนที่ 1 — ✓ ขั้นตอนที่ 2 — ● ขั้นตอนที่ 3 — ○ ขั้นตอนที่ 4

ตั้งค่ารายละเอียดอุปกรณ์

ชื่อ
โรงเพาะเห็ด

รายละเอียด
รายละเอียด

ก่อนหน้า **ถัดไป**

รอประมาณผล ใช้เวลา 2 - 5 วินาที จะแสดงหน้าขั้นตอนที่ 4 สร้างอุปกรณ์เสร็จสิ้น กดปุ่ม เสร็จสิ้น เพื่อปิดหน้าต่างนี้

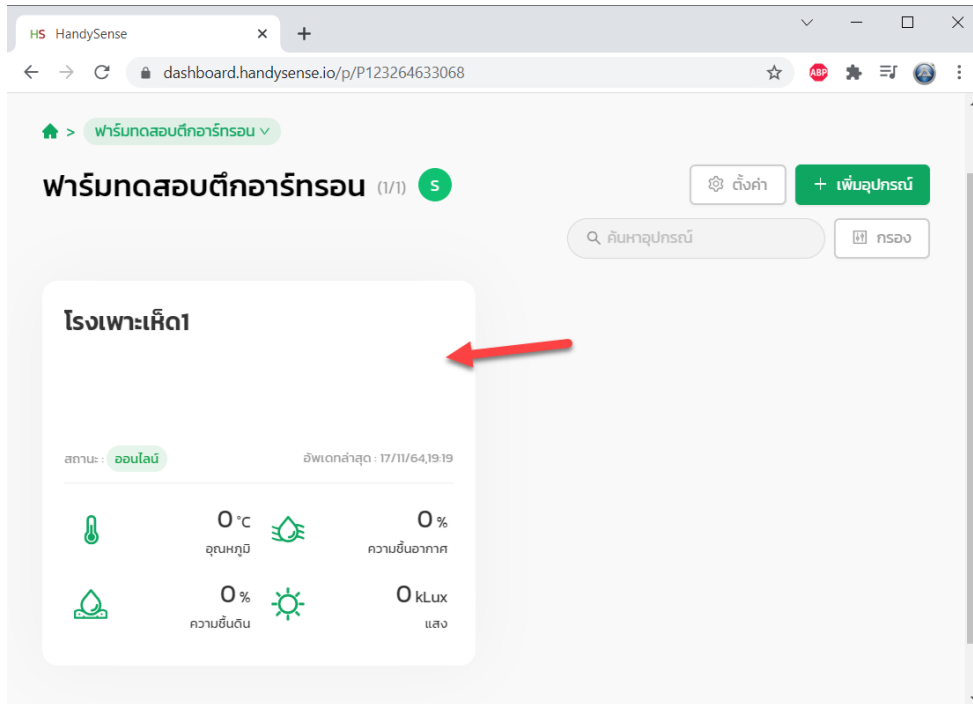
สร้างอุปกรณ์

✓ ขั้นตอนที่ 1 — ✓ ขั้นตอนที่ 2 — ✓ ขั้นตอนที่ 3 — ● ขั้นตอนที่ 4

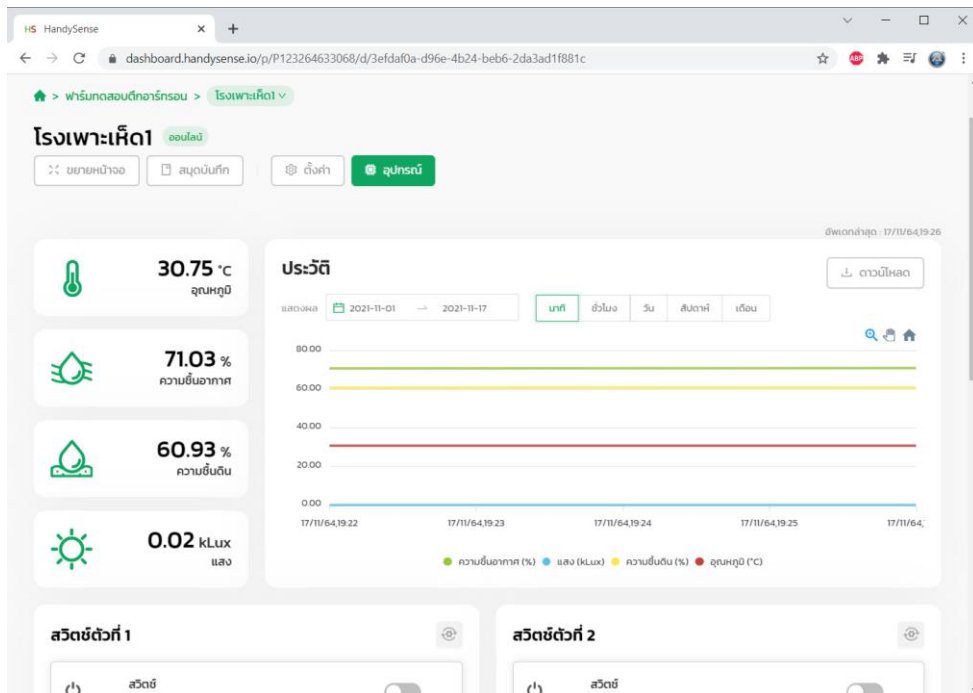
สร้างอุปกรณ์เสร็จสิ้น

เสร็จสิ้น

จะมีอุปกรณ์เพิ่มเข้ามาในฟาร์มแล้ว กดที่กล่องอุปกรณ์เพื่อดูรายละเอียด และตั้งค่าต่าง ๆ



จากนั้นรอประมาณ 2 นาที (ค่าจากเซ็นเซอร์ส่งขึ้น HandySense ทุก ๆ 2 นาที) ค่าที่วัดได้จากเซ็นเซอร์ก็จะแสดงขึ้นมาแล้ว



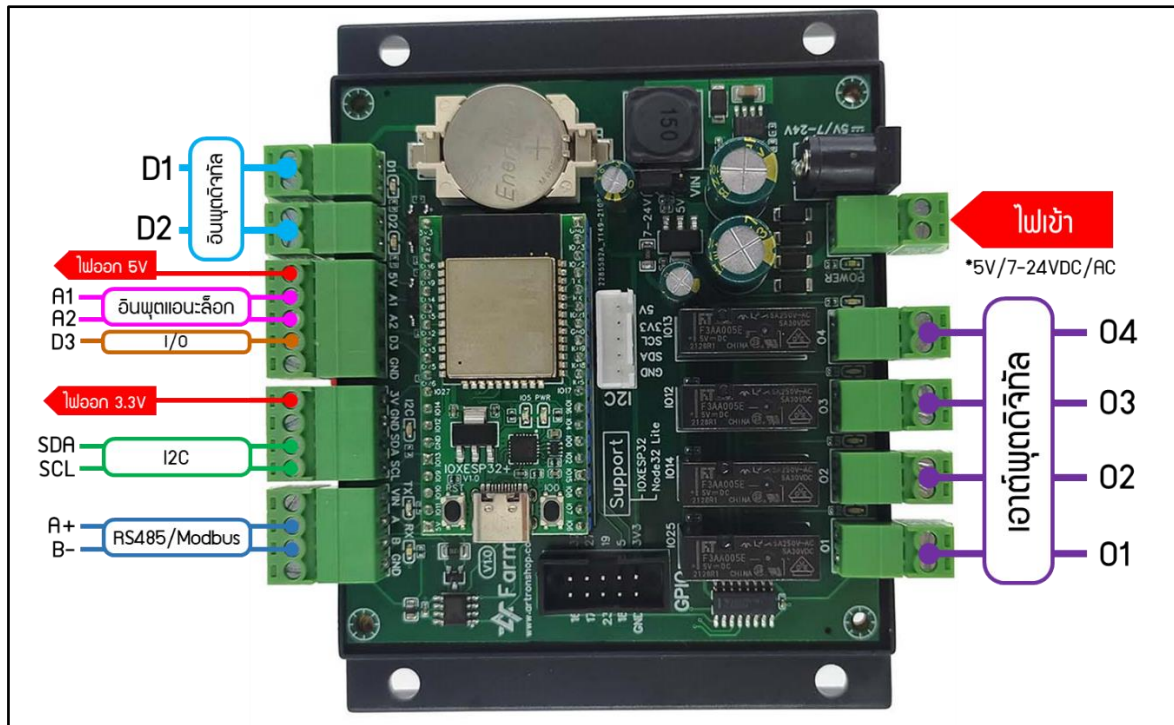
รายละเอียดการใช้งานระบบ ศึกษาเพิ่มเติมได้จากเอกสาร [คู่มือการใช้งาน Web Application: HandySense ระบบเกษตรแม่นยำ ฟาร์มอัจฉริยะ](#)

ข้อมูลด้านฮาร์ดแวร์และวงจร

คุณสมบัติทางไฟฟ้า

คุณสมบัติ	ต่ำสุด	ปกติ	สูงสุด
แรงดันไฟฟ้า (DC/AC)	5V	12V	24V
กระแสไฟฟ้า	-	-	1A
การทนกระแสไฟฟ้าชอร์ตพุท	-	5A/220V	-
แรงดันช่องอินพุตดิจิทัล	3V	-	12V
กระแสช่องอินพุตดิจิทัล	0A (LOW)	-	3mA (HIGH)
แรงดันช่องอินพุตแอนะล็อก	0V	-	9.9V
กระแสช่องอินพุตแอนะล็อก	-	-	<1mA
แรงดันช่อง I/O (D3)	0V	-	3.3V
กระแสช่อง I/O ขณะเป็นอินพุต	0A (LOW)	-	1mA (HIGH)
กระแสช่อง I/O ขณะเป็นเอาต์พุต	0A	-	10mA
แรงดันช่อง I ² C	-	3.3V	-
แรงดันช่อง VIN	แรงดันไฟเข้า - 0.6V		
แบตเตอรี่วงจรนับเวลา (RTC)	3V (CR2032)		

ตำแหน่งขาต่อใช้งาน



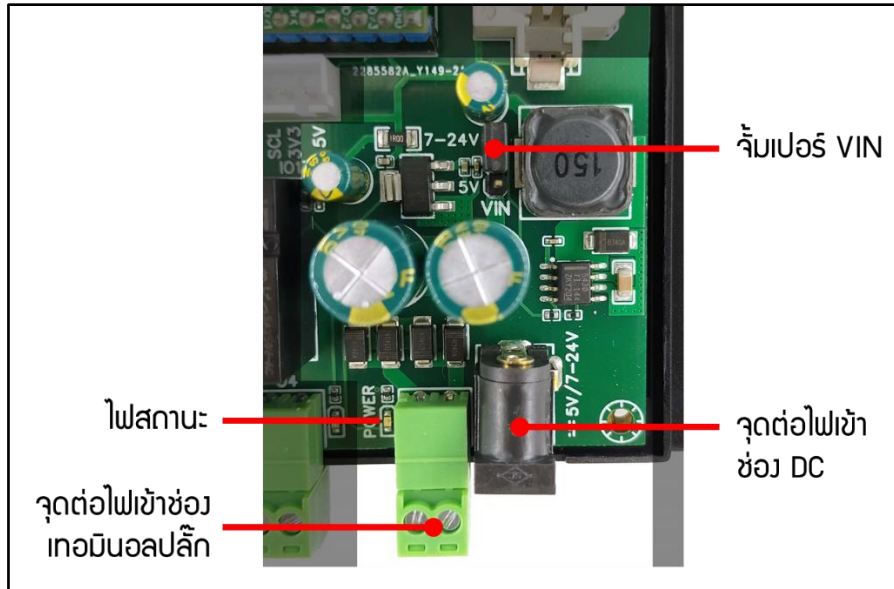
รูปที่ 1 ตำแหน่งขาต่อใช้งาน Farm1

การจ่ายไฟ

รองรับการจ่ายไฟผ่านช่อง DC / เทอมินอลปลั๊ก รองรับไฟเข้าได้ 5V และ 7V ถึง 24V (ขึ้นอยู่กับจัมเปอร์) ทั้งกระแสตรง (DC) และกระแสสลับ (AC) กระแสไฟฟ้าที่ต้องการต่ำสุด 1A

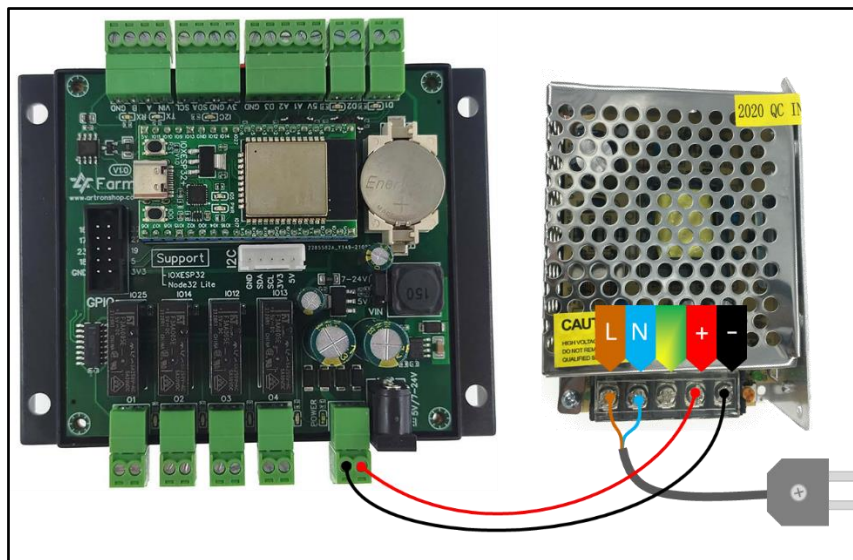
จัมเปอร์ VIN ใช้เลือกกระดบแรงดันไฟเข้า โดยหากจัมป์ที่ 5V ต้องจ่ายไฟเข้า 5V เท่านั้น (DC) หากเลือกเป็น 7-24V เลือกจ่ายไฟด้วยหม้อแปลง แบบตาออร์ โซล่าเซลล์ หรือแหล่งจ่ายไฟอื่นที่อยู่ในช่วง 7V ถึง 24V ได้ รองรับทั้งไฟฟ้้ากระแสตรงและกระแสสลับ

หลอดแอลอีดี POWER ติดสว่างเป็นสีเขียวเมื่อได้รับไฟฟ้าจากแหล่งจ่ายไฟ



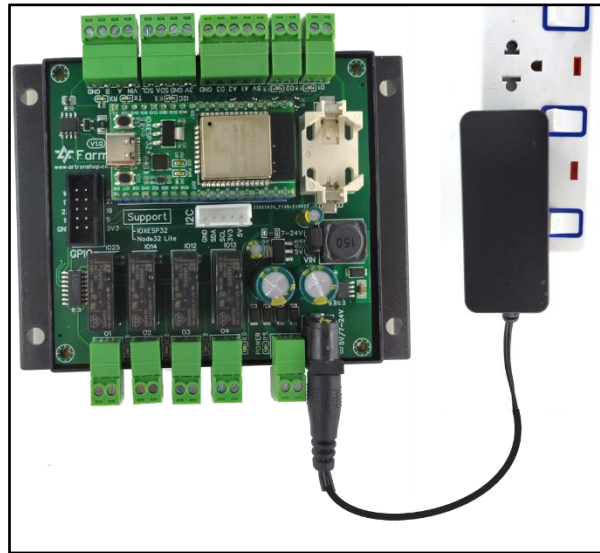
รูปที่ 2 วงจรส่วนภาคจ่ายไฟ

ตัวอย่างการจ่ายไฟให้บอร์ด Farm1 ด้วยสวิทช์เพาเวอร์ซัพพลาย 12V แสดงดังรูปที่ 3



รูปที่ 3 การจ่ายไฟให้บอร์ด Farm1 ด้วยสวิทช์เพาเวอร์ซัพพลาย 12V

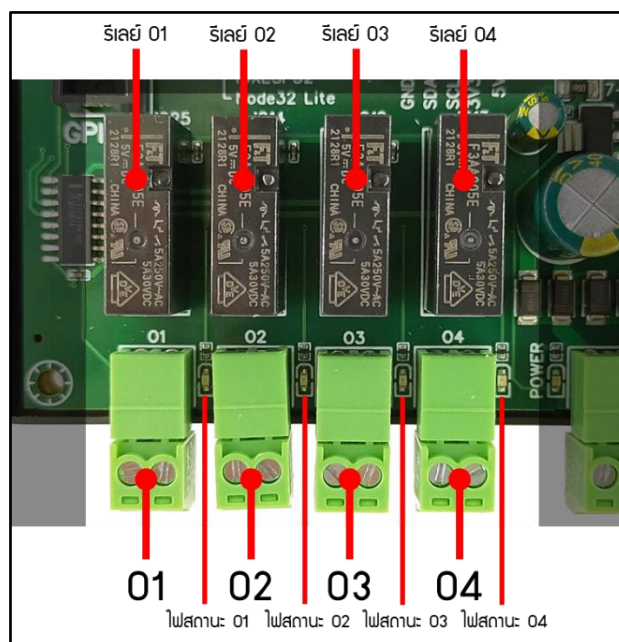
ตัวอย่างการจ่ายไฟให้บอร์ด Farm1 ด้วยอะแดปเตอร์ 9V แสดงดังรูปที่ 4



รูปที่ 4 การจ่ายไฟให้บอร์ด Farm1 ด้วยอะแดปเตอร์ 9V

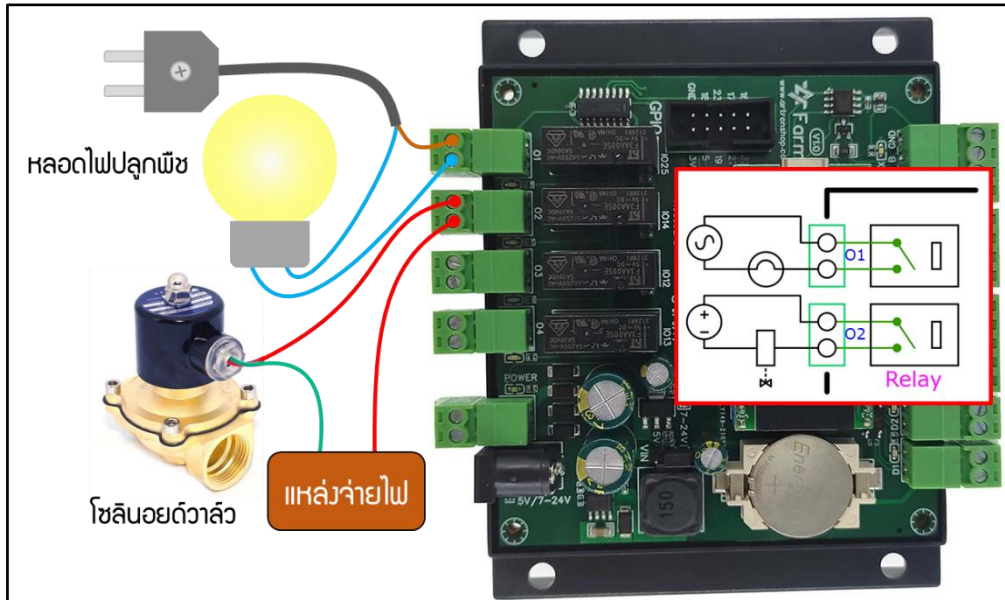
ช่องเอาต์พุต

ช่องเอาต์พุตประกอบด้วย 01 ถึง 04 เป็นเอาต์พุตแบบหน้าคอนเทครีเลย์ NO ทนกระแสได้สูงสุด 5A ที่ 220V มีหลอดแอลอีดีแสดงสถานะการทำงาน โดยจะติดสว่างเป็นสีส้มเมื่อหน้าคอนเทคต่อ (รีเลย์ทำงาน)



รูปที่ 5 ส่วนประกอบภาคเอาต์พุต

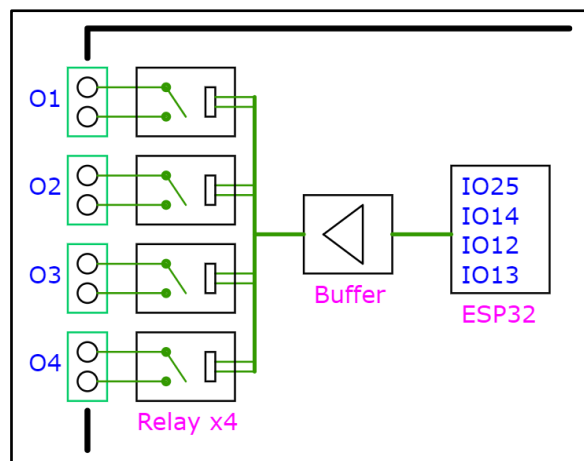
ตัวอย่างการต่อวงจรเอาต์พุตเข้ากับหลอดไฟและโซลีนอยด์วาล์ว แสดงดังรูปที่ 6



รูปที่ 6 การต่อวงจรเอาต์พุตเข้ากับหลอดไฟและโซลีนอยด์วาล์ว

■ การต่อวงจรเอาต์พุตเข้ากับอุปกรณ์กำลังสูง ต้องต่อผ่านแมกเนติก (Magnetic Contactor) เท่านั้น

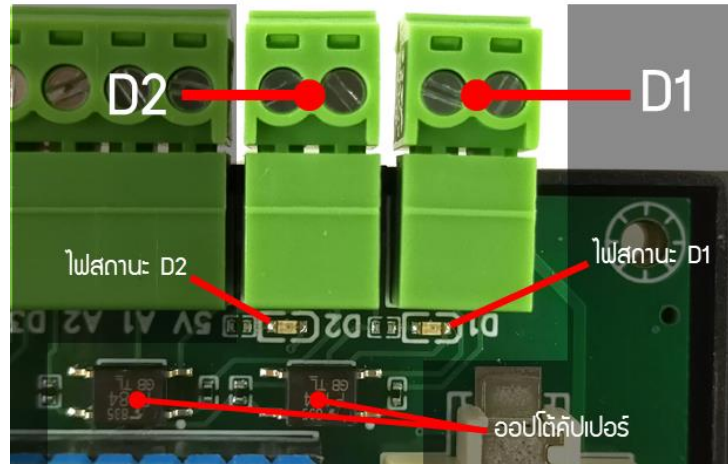
รีเลย์ของวงจรเอาต์พุตต่อเข้ากับ ESP32 ผ่านขา 25, 14, 12 และ 13 ตามลำดับ บล็อกไดอะแกรมส่วนเอาต์พุตดิจิทัลแสดงดังรูปที่ 7



รูปที่ 7 บล็อกไดอะแกรมส่วนเอาต์พุตดิจิทัล

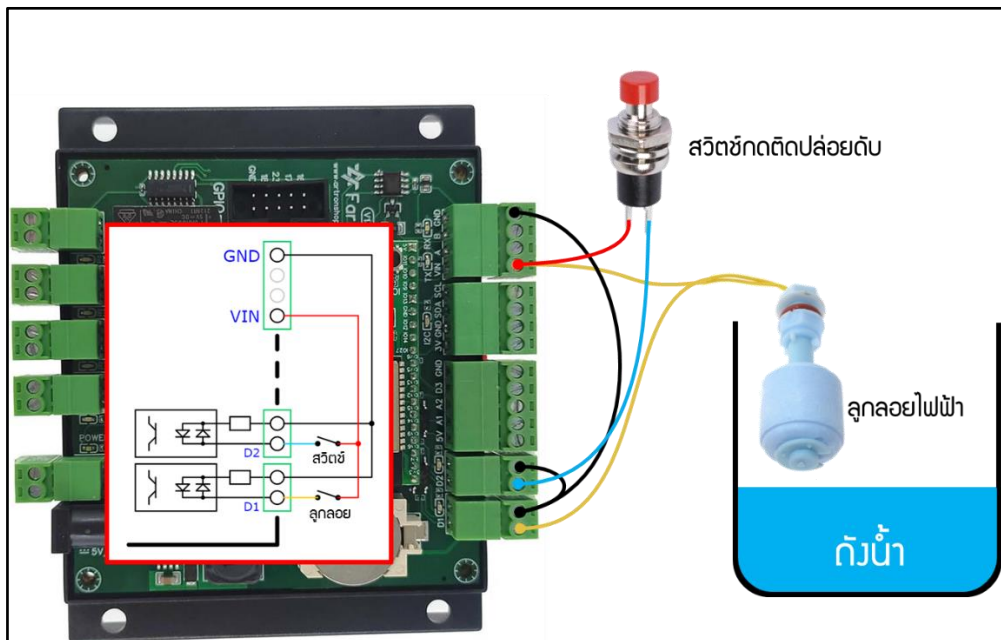
ช่องอินพุตดิจิทัล

ช่องอินพุตดิจิทัลประกอบด้วย D1 และ D2 รองรับไฟเข้าได้ 3V ถึง 12V คั่นระหว่างสัญญาณเข้ากับ ESP32 ด้วยออปโตคัปเปอร์ มีหลอดแอลอีดีสีส้ม ติดเมื่อได้รับลอจิก 1



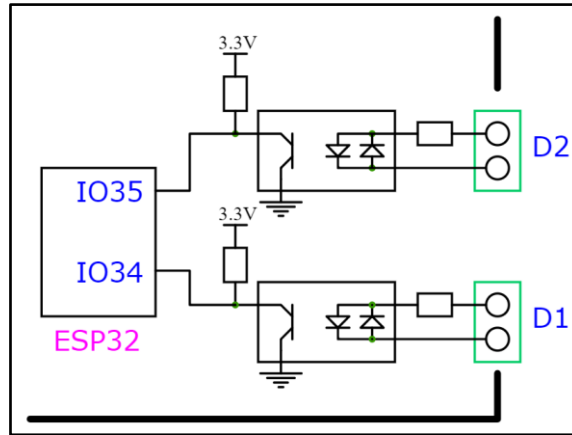
รูปที่ 8 ส่วนประกอบภาคอินพุตดิจิทัล

ตัวอย่างการต่อสวิทซ์ และลูกลอยเข้ากับช่องอินพุต แสดงด้รูปที่ 9



รูปที่ 9 การต่อสวิทซ์ และลูกลอยเข้ากับช่องอินพุต

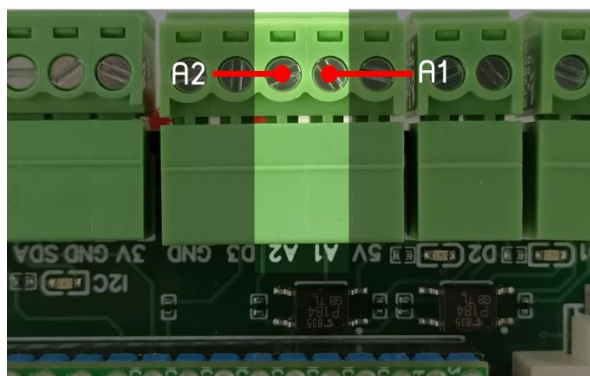
ช่องอินพุตดิจิทัล เชื่อมต่อกับ ESP32 ผ่านออปโตคอปเปอร์ที่ขา 34, 35 ตามลำดับ ทำงานแบบ Active LOW เมื่อมีสัญญาณอินพุตเข้ามา สัญญาณเข้า ESP32 จะเป็น LOW และหากไม่มีสัญญาณอินพุตเข้ามา สัญญาณเข้า ESP32 จะเป็น HIGH บล็อกไดอะแกรมส่วนอินพุตดิจิทัล แสดงดังรูปที่ 10



รูปที่ 10 บล็อกไดอะแกรมส่วนอินพุตดิจิทัล

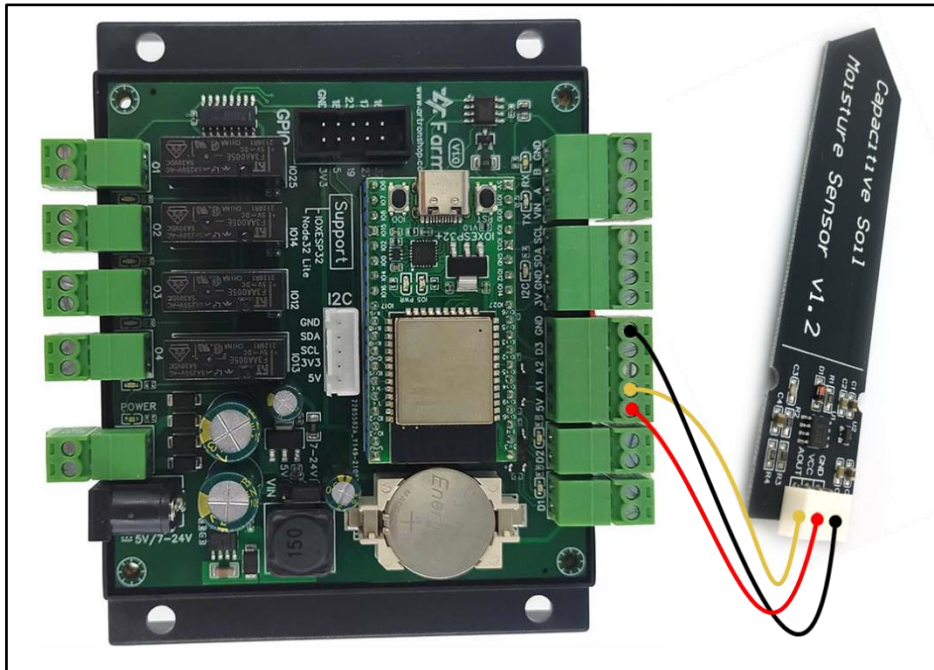
ช่องอินพุตแอนะล็อก

ช่องอินพุตแอนะล็อกประกอบด้วย A1 และ A2 รองรับไฟขาได้ 0V ถึง 9.9V ใช้ต่อเข้ากับเซ็นเซอร์ที่ให้อาต์พุตเป็นแรงดันไฟฟ้า เช่น เซ็นเซอร์วัดความชื้นในดิน, เซ็นเซอร์วัดระดับน้ำ, เซ็นเซอร์วัดความขุ่นของน้ำ, เซ็นเซอร์วัดความเป็นกรด-ด่าง (PH) เป็นต้น



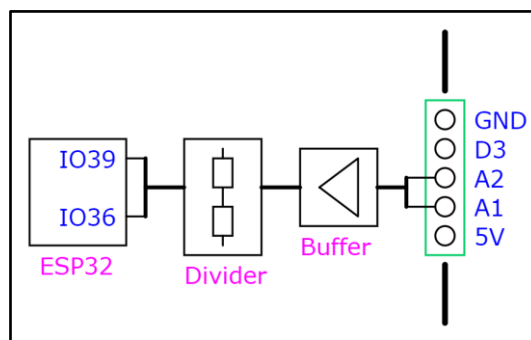
รูปที่ 11 ส่วนประกอบภาคอินพุตแอนะล็อก

ตัวอย่างการต่อเซ็นเซอร์วัดความชื้นในดินเข้ากับช่องอินพุตแอนะล็อก แสดงด้วยรูปที่ 12



รูปที่ 12 การต่อเซ็นเซอร์วัดความชื้นในดินเข้ากับช่องอินพุตแอนะล็อก

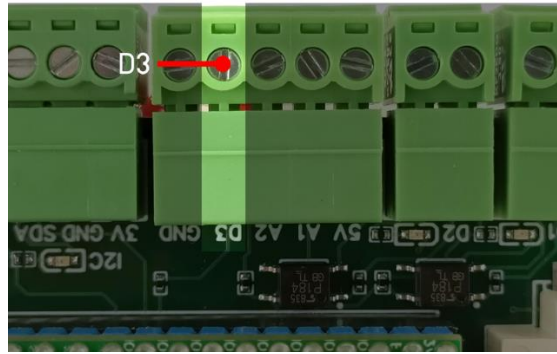
อินพุตแอนะล็อกผ่านวงจรบัพเฟอร์จากนั้นเข้าวงจรแบ่งแรงดันเพื่อให้แรงดันไม่เกิน 3.3V ทำให้ ESP32 สามารถรับแรงดันสูงได้ ช่องแอนะล็อกต่อกับ ESP32 ขา 36, 39 ตามลำดับ บล็อกไดอะแกรมส่วนอินพุตแอนะล็อก แสดงด้วยรูปที่ 12



รูปที่ 13 บล็อกไดอะแกรมส่วนอินพุตแอนะล็อก

ช่อง I/O

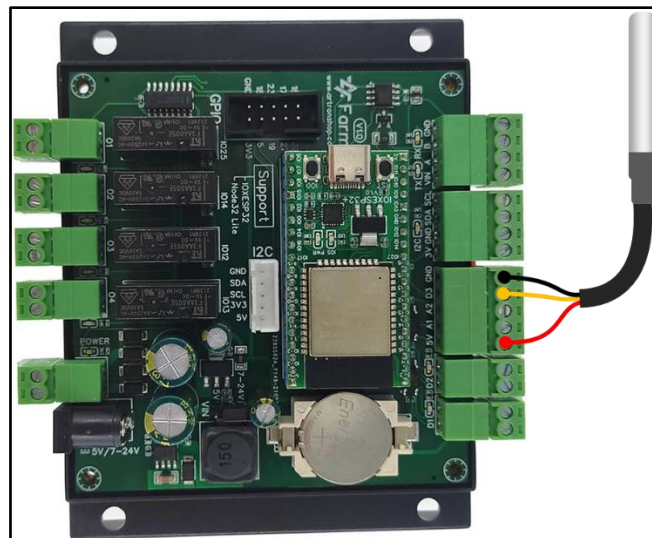
ช่อง I/O คือ D3 รองรับแรงดันไฟเข้าสูงสุด 3.3V ใช้ต่อกับกับเซ็นเซอร์/อุปกรณ์ภายนอกสื่อสารด้วยสายเส้นเดียว เช่น เซ็นเซอร์วัดอุณหภูมิและความชื้น DHT11/DHT22, เซ็นเซอร์วัดอุณหภูมิแบบกันน้ำ DS18B20, หลอดแอลอีดี RGB เบอร์ WS2812, เซอร์โวมอเตอร์ เป็นต้น



รูปที่ 14 ส่วนประกอบภาค I/O

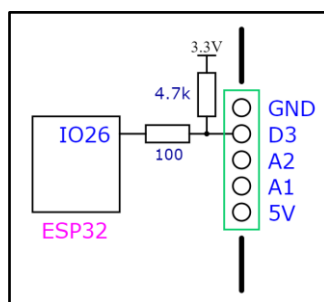
ตัวอย่างการต่อเซ็นเซอร์อุณหภูมิหัวกันน้ำ DS18B20 เข้ากับช่อง I/O แสดงด้วยรูปที่ 15

- DS18B20 ทำงานแบบ 1-wire ตามข้อกำหนด ต้องต่อตัวต้านทาน Pull-up ด้วย



รูปที่ 15 การต่อเซ็นเซอร์อุณหภูมิหัวกันน้ำ DS18B20 เข้ากับช่อง I/O

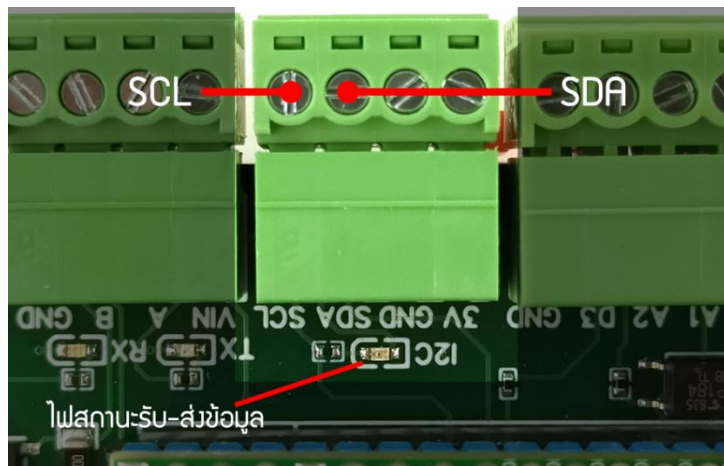
ช่อง I/O ต่อตรงกับ ESP32 ขา 26 คั่นด้วยตัวต้านทาน 100Ω เพื่อป้องกันกระแสไหลเกิน พร้อมตัวต้านทาน 4.7 kΩ เป็น Pull-up บล็อกไดอะแกรมส่วน I/O แสดงด้วยรูปที่ 16



รูปที่ 16 บล็อกไดอะแกรมส่วน I/O

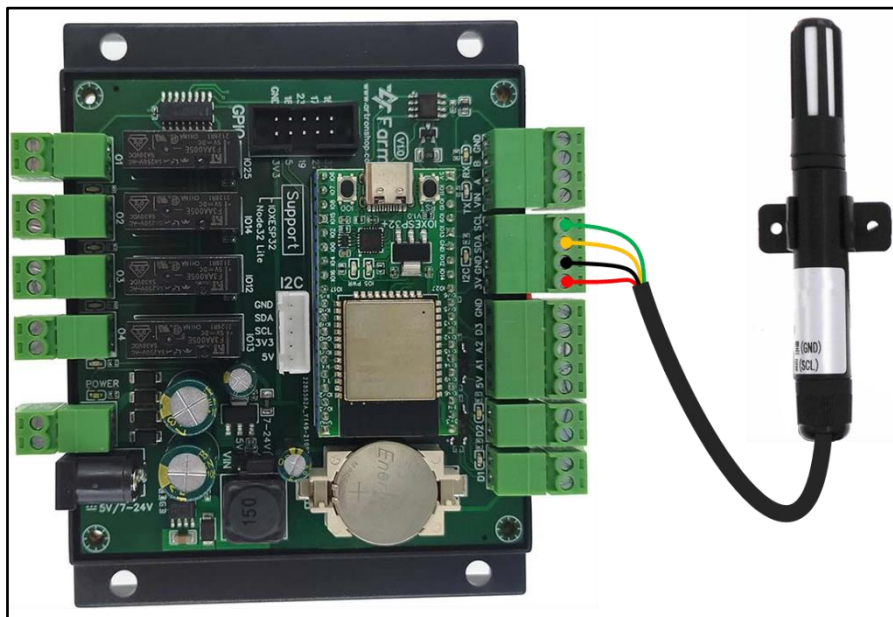
ช่อง I²C

ช่อง I²C ประกอบด้วย SCL และ SDA ใช้ต่อกับอุปกรณ์ I²C เช่น เซ็นเซอร์วัดอุณหภูมิและความชื้น SHT30, เซ็นเซอร์วัดความกดอากาศ BMP280, เซ็นเซอร์แสง BH1750 เป็นต้น ใช้งานได้ด้วยอุปกรณ์ที่ใช้แรงดันไฟฟ้า 3.3V มีหลอดแอลอีดีที่ระบุสถานะรับ-ส่งข้อมูล



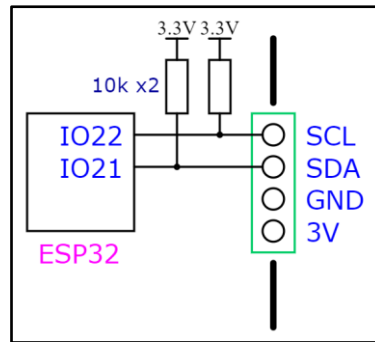
รูปที่ 17 ส่วนประกอบภาค I²C

การต่อกับเซ็นเซอร์ จำเป็นจะต้องต่อขา SCL และ SDA ให้ถูกต้อง ตัวอย่างการต่อเซ็นเซอร์วัดอุณหภูมิและความชื้น SHT30 แสดงด้วยรูปที่ 18



รูปที่ 18 การต่อเซ็นเซอร์วัดอุณหภูมิและความชื้น SHT30 เข้ากับช่อง I²C

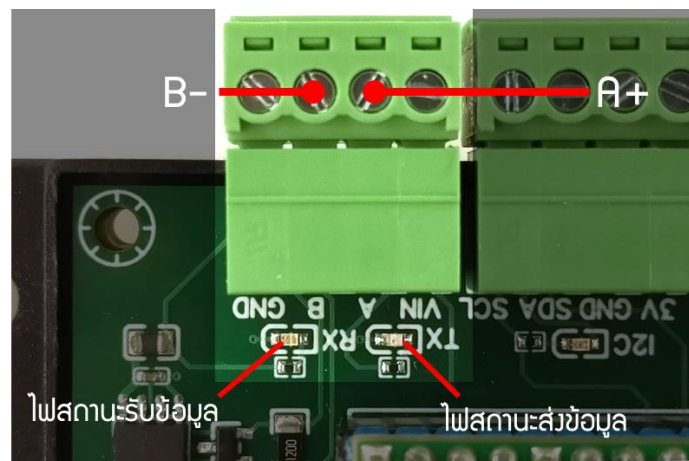
ช่อง I2C ต่อตรงกับ ESP32 ขา 22 (SCL), 21 (SDA) โดยมีตัวต้านทาน Pull-up 10kΩ ต่อไว้ให้แล้ว
บล็อกไดอะแกรมส่วน I²C แสดงดังรูปที่ 19



รูปที่ 19 บล็อกไดอะแกรมส่วน I2C

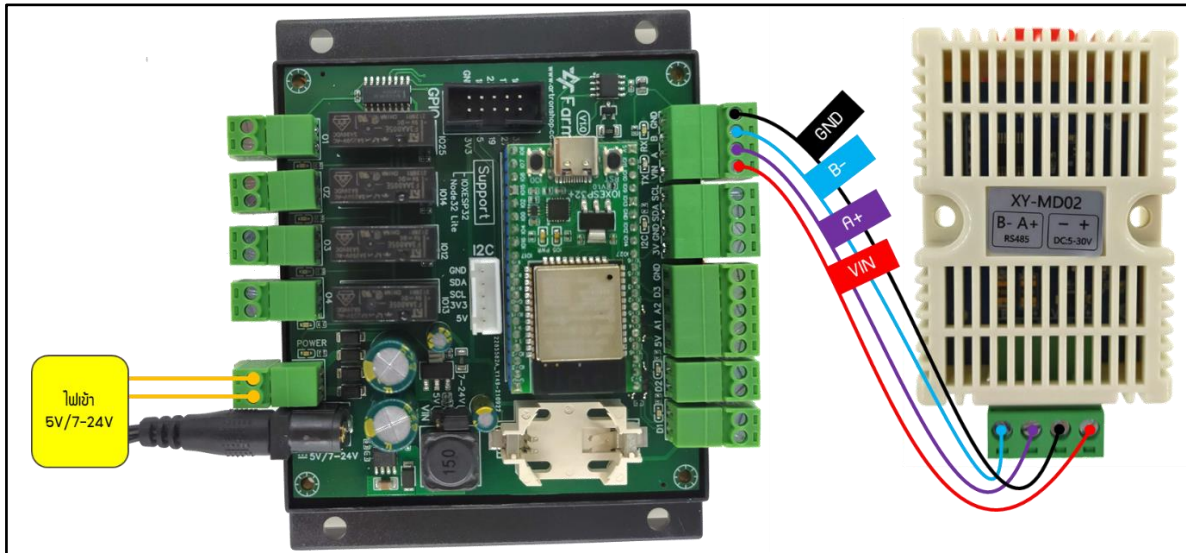
ช่อง RS485

ช่อง RS485 ประกอบด้วย A+ และ B- ใช้ต่อกับเซ็นเซอร์ RS485 หรือ Modbus RTU เช่น เซ็นเซอร์วัดความชื้นในดิน เซ็นเซอร์อุณหภูมิ เซ็นเซอร์วัดระดับน้ำฝน เซ็นเซอร์วัดความเร็วลม เป็นต้น
ช่อง RS485 บนบอร์ดมี Resistor Termination ค่าความต้านทาน 120Ω มาให้แล้ว พร้อม TVS ป้องกันแรงดันไฟฟ้าในบัสสูงเกินกำหนด



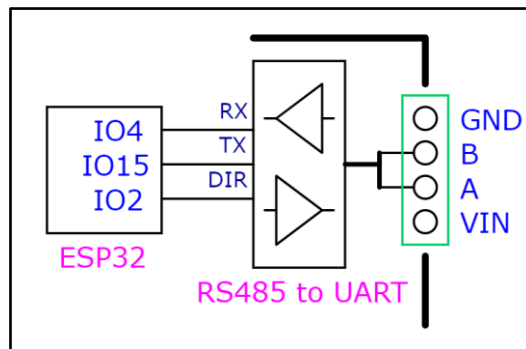
รูปที่ 20 ส่วนประกอบภาค RS485

ตัวอย่างการต่อเซ็นเซอร์วัดอุณหภูมิและความชื้น XY-MD02 เข้ากับช่อง RS485 แสดงดังรูปที่ 21



รูปที่ 21 การต่อเซ็นเซอร์วัดอุณหภูมิและความชื้น XY-MD02 เข้ากับ Fame1

ช่อง RS485 ต่อกับ ESP32 ผ่านวงจรแปลง RS485 เป็น UART เข้าที่ขา 4 (RX), 15 (TX) และ 2 (DIR) บล็อกไดอะแกรมส่วน RS485 แสดงดังรูปที่ 22



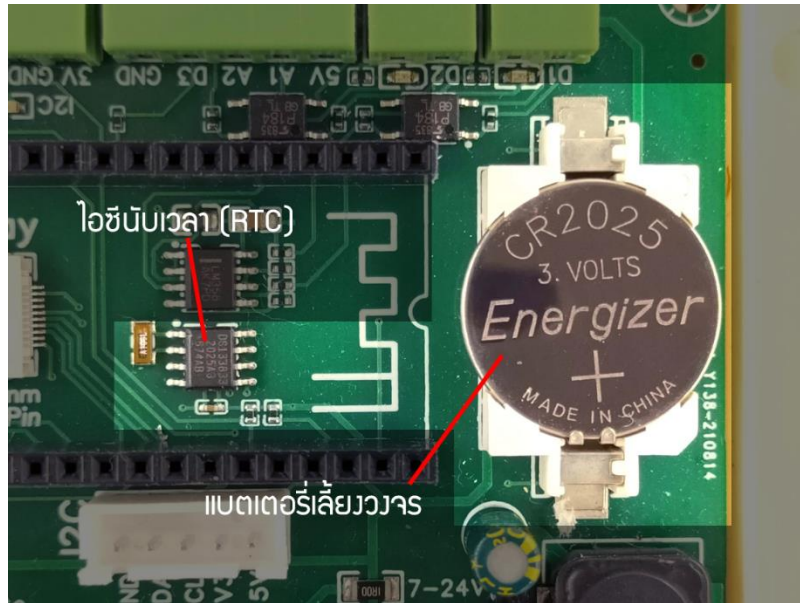
รูปที่ 22 บล็อกไดอะแกรมส่วน RS485

วงจรนับเวลา (RTC)

วงจรนับเวลาต่อตรงกับ ESP32 ผ่านช่อง I²C ใช้นับเวลาเมื่อไม่มีอินเตอร์เน็ต หรือเมื่อไม่มีไฟเลี้ยง ใช้ประกอบการเขียนโปรแกรมที่ต้องการให้ทำงานตามเวลา เช่น ส่งค่าที่วัดได้ในฟาร์มขึ้นระบบคลาวด์ตามเวลา เปิด-ปิดไฟตามเวลา เปิด-ปิดปั๊มน้ำตามเวลา เป็นต้น

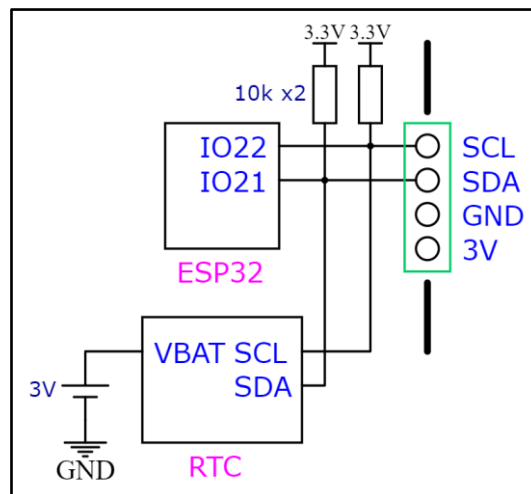
วงจรนับเวลา (RTC) จำเป็นต้องใช้ถ่าน CR2032 จ่ายไฟเลี้ยงให้วงจร โดยไฟจากถ่านจะถูกใช้เมื่อไม่มีแรงดันจากแหล่งจ่ายไฟจ่ายเข้ามา ทำให้วงจรส่วนนับเวลา (RTC) ยังมีไฟเลี้ยงอยู่ตลอด ทำให้วงจรยัง

ทำงาน นับเวลาอยู่เรื่อย ๆ จดจำค่าเวลาได้ และพร้อมให้เรียกใช้ค่าเวลาเมื่อวารส่วนประมวลผลกลับมาทำงานอีกครั้ง



รูปที่ 23 ส่วนประกอบวารนับเวลา (RTC)

บล็อกไดอะแกรมส่วนวารนับเวลา (RTC) แสดงตัวรูปที่ 24



รูปที่ 24 บล็อกไดอะแกรมส่วนวารนับเวลา (RTC)

การเขียนโปรแกรมสั่งงานด้วย Arduino IDE

Farm1 ใช้ไมโครคอนโทรลเลอร์ ESP32 ในการประมวลผล รองรับเครื่องมือพัฒนาทั้ง ESP-IDF และ Arduino IDE ในเอกสารนี้จะแนะนำตัวอย่างการเขียนโปรแกรมแต่ละส่วนด้วย Arduino IDE

ติดตั้งโปรแกรม Arduino IDE

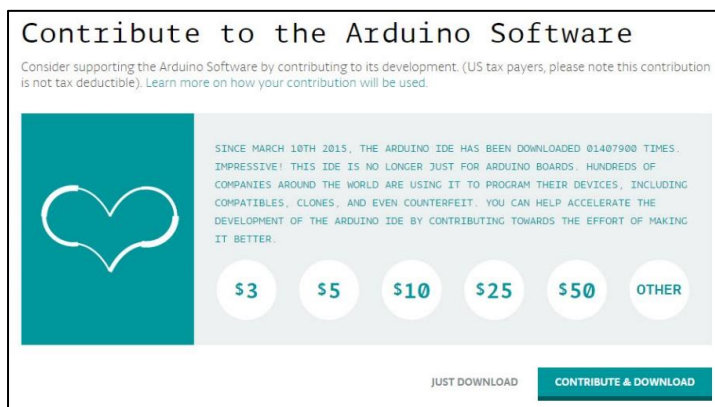
โปรแกรม Arduino IDE เป็นหนึ่งในเครื่องมือที่ใช้พัฒนาโปรแกรมให้ Farm1 ซึ่งประกอบไปด้วย Text Editor ใช้แก้ไขโค้ดโปรแกรม, Compiler และ Uploader

ขั้นตอนการติดตั้งโปรแกรม Arduino IDE มีดังนี้

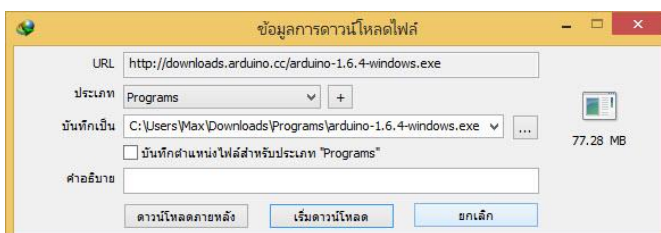
เข้าไปที่ <http://www.arduino.cc/en/Main/Software> เลือก OS ให้ตรงกับเครื่องคอมพิวเตอร์



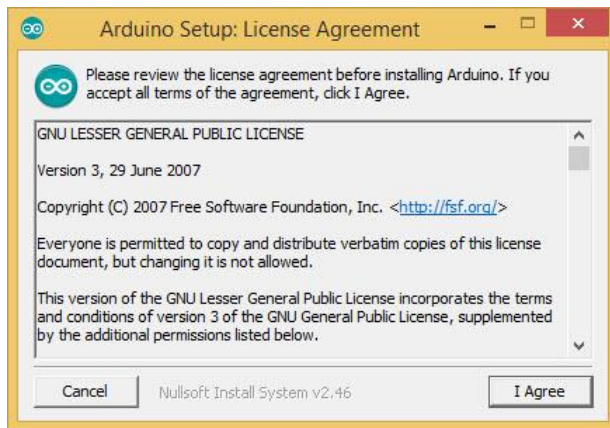
กดปุ่ม JUST DOWNLOAD เพื่อเริ่มดาวน์โหลดโปรแกรม



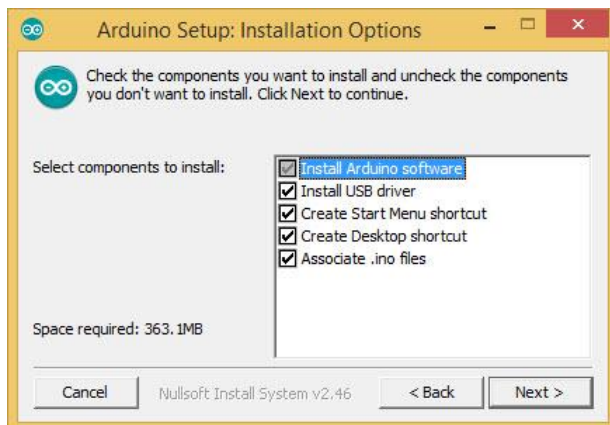
รอกว่าการดาวน์โหลดจะเสร็จสิ้น แล้วจึงทำขั้นตอนถัดไป



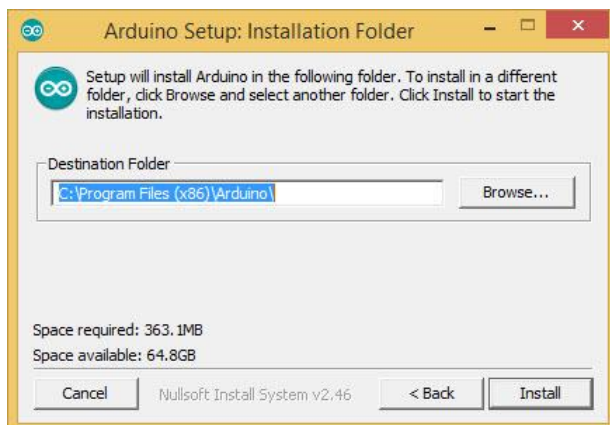
เมื่อดาวน์โหลดเสร็จแล้วให้เปิดไฟล์ติดตั้งขึ้นมา แล้วกดปุ่ม I Agree เพื่อยืนยันยอมรับเงื่อนไขการใช้งานโปรแกรม



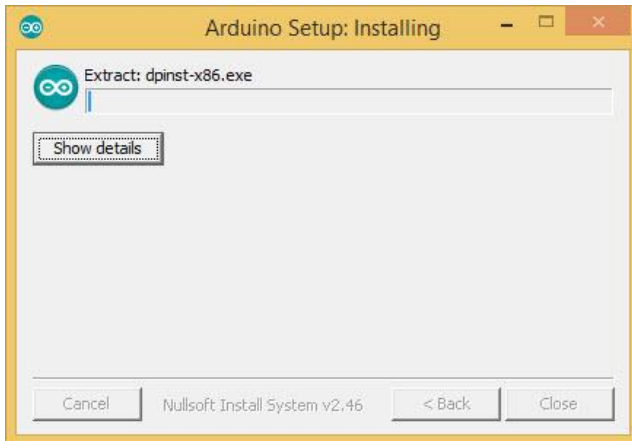
คลิกปุ่ม Next > เพื่อติดตั้งทั้งหมด



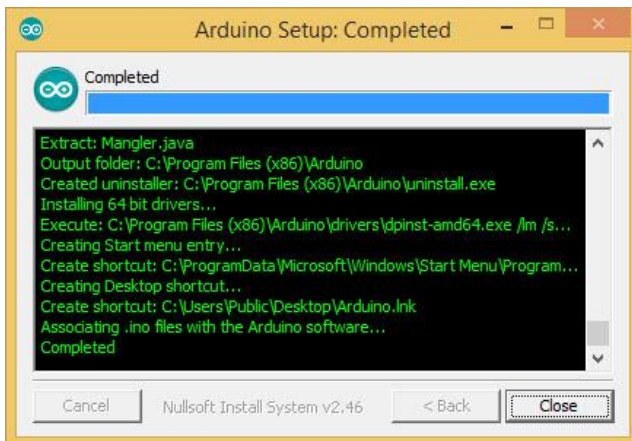
เลือกโฟลเดอร์ติดตั้งโปรแกรม หากไม่ต้องการแก้ไขคลิกปุ่ม Install



รอกันว่าโปรแกรมจะติดตั้งเสร็จสิ้น



เมื่อขึ้นคำว่า Completed ให้กดปุ่ม Close เพื่อปิดหน้าต่างติดตั้งโปรแกรมได้เลย



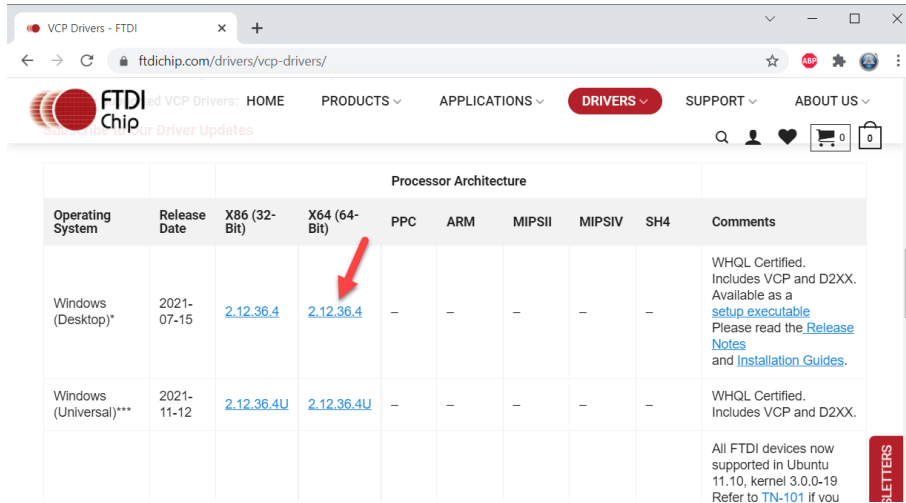
หน้าเดสทอปก็จะมีไอคอนโปรแกรม Arduino ขึ้นมาแล้ว



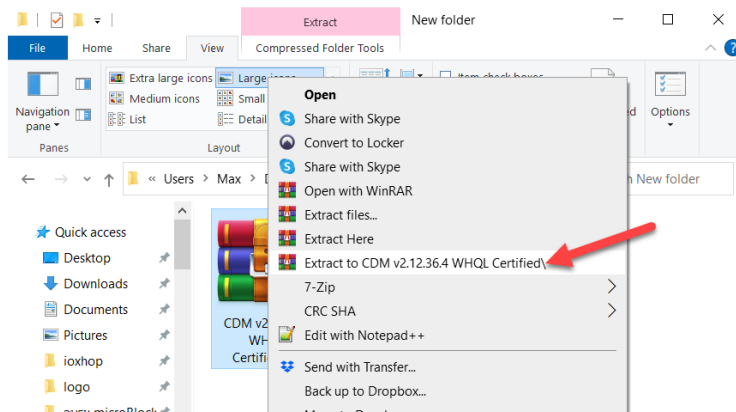
ติดตั้งไดร์เวอร์ FT231X

บอร์ด IOXESP32+ เป็นบอร์ดที่ใช้พัฒนา ESP32 เสียบอยู่บน Farm1 มีวงจรฮับโหลดโปรแกรมบนบอร์ดผ่านชิปแปลง USB เป็น UART เบอร์ FT231X ซึ่งจำเป็นต้องติดตั้งไดร์เวอร์ก่อนจึงจะฮับโหลดโปรแกรมเข้าได้

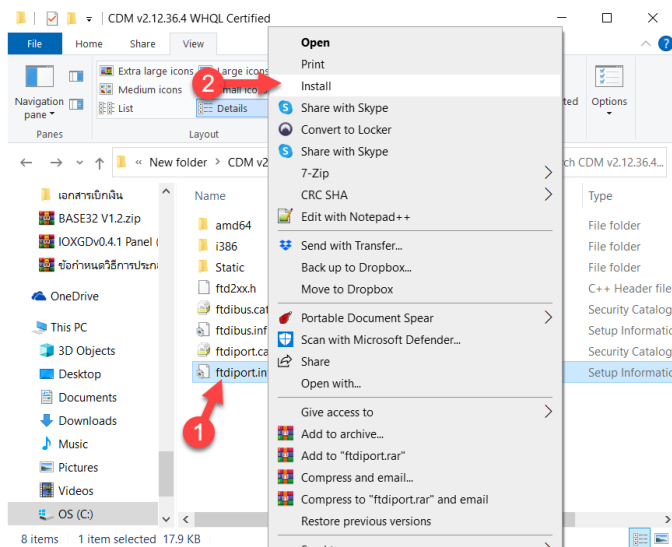
เข้าไปที่ <https://ftdichip.com/drivers/vcp-drivers/> กดดาวน์โหลดไฟล์ตาม OS ของเครื่องคอมพิวเตอร์



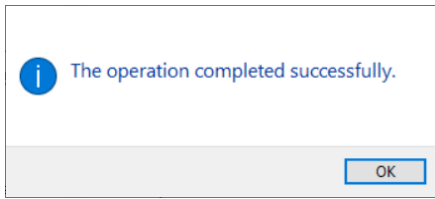
จะได้ไฟล์มา CDM vxxxxx WHQL Certified.zip ให้แตกไฟล์ด้วยโปรแกรม WinRAR, 7-Zip



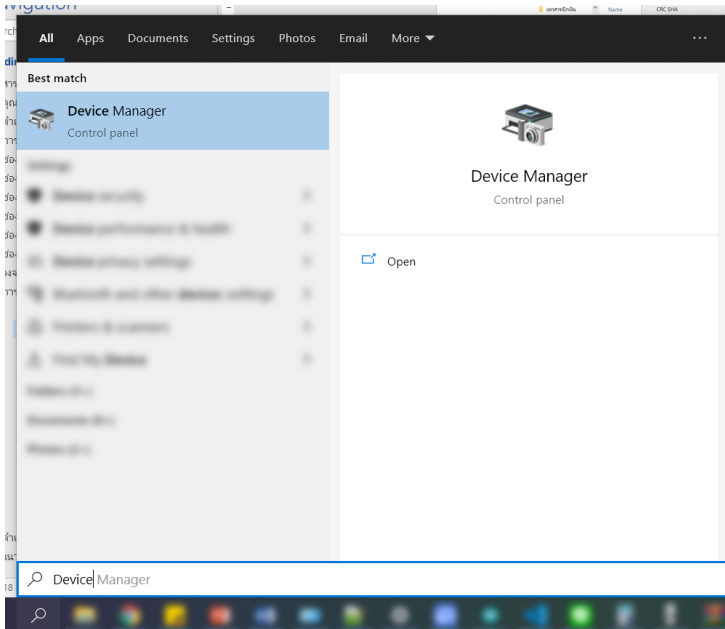
เข้าไปในโฟลเดอร์ที่แตกไฟล์มา แล้วคลิกขวาที่ไฟล์ ftdiport.inf กด Install



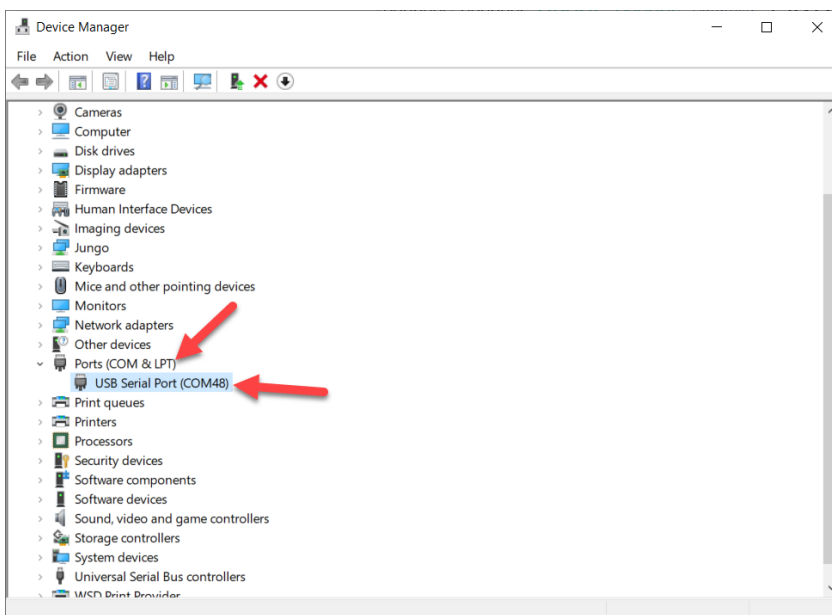
จากนั้นหน้าต่างให้ยืนยันจะแสดงขึ้นมา ให้กดปุ่ม Yes หากติดตั้งสำเร็จ จะมีข้อความแจ้ง (ดังรูป)



เปิด Device Manager ขึ้นมา โดยพิมพ์ในช่องค้นหา



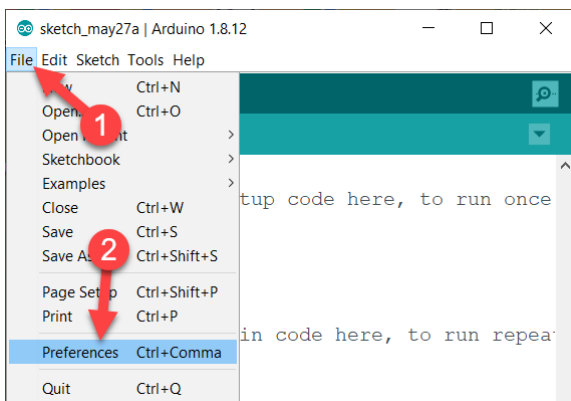
เสียบสาย USB ตามหัวข้อ [การเชื่อมต่อ Farm1 เข้ากับเครื่องคอมพิวเตอร์](#) จากนั้นสังเกตในหน้าต่าง Device Manager จะต้องมีส่วนหัวข้อ Ports และมี USB Serial Port ขึ้น เป็นอันเสร็จสิ้นการติดตั้งไดรเวอร์



การเพิ่มบอร์ด ESP32 ลงโปรแกรม Arduino IDE

โปรแกรม Arduino IDE เดิมรองรับการเขียนโปรแกรมสำหรับบอร์ดที่ผลิตโดย Arduino เท่านั้น หากต้องการใช้เขียนโปรแกรมให้ ESP32 จำเป็นต้องติดตั้งบอร์ด ESP32 เพิ่ม โดยติดตั้ง Arduino core for the ESP32

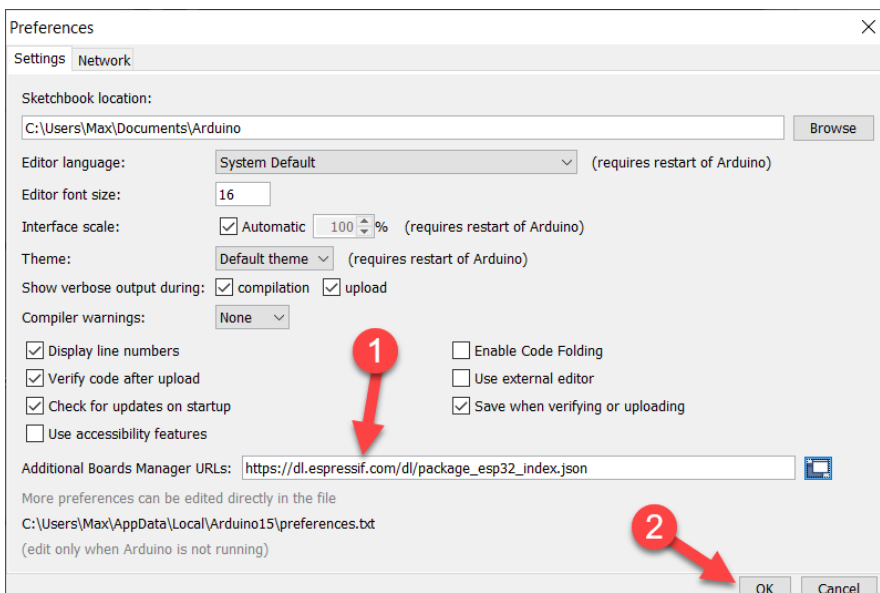
เปิดโปรแกรม Arduino IDE ขึ้นมา โดยดับเบิลคลิกที่ไอคอน Arduino บนหน้าเดสก์ท็อป กด File เลือก Preferences



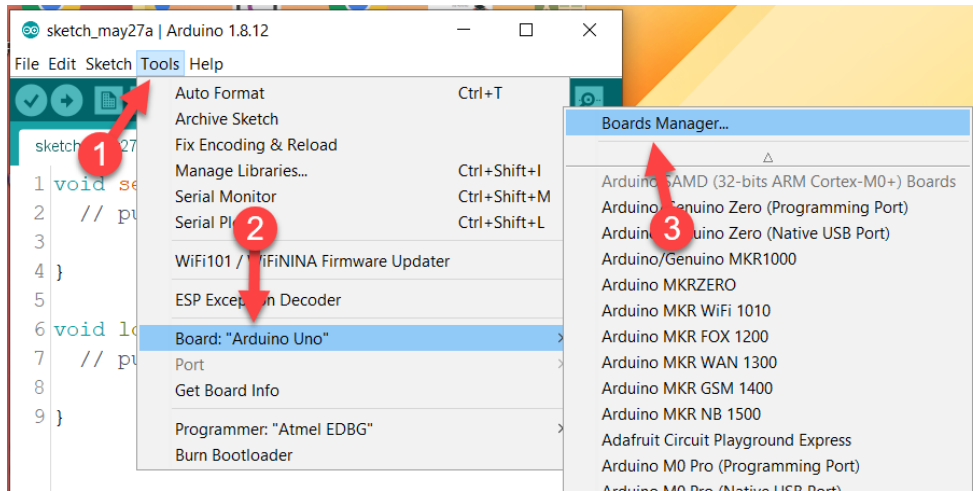
ในช่อง Additional Boards Manager URLs ใส่

https://raw.githubusercontent.com/espressif/arduino-esp32/gh-pages/package_esp32_index.json

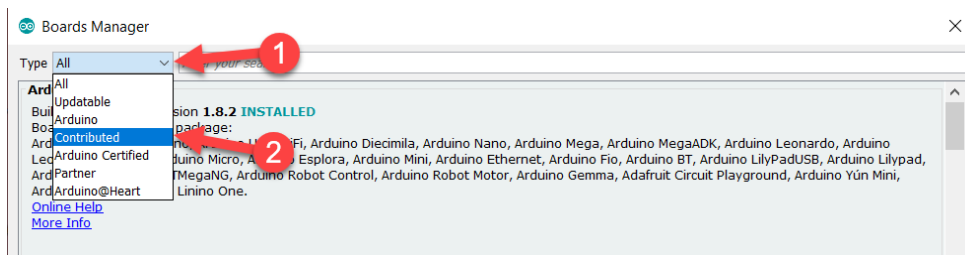
แล้วกดปุ่ม OK (ในรูปเป็นลิ่งดำ ให้ใส่ลิ่งใหม่ด้านบน)



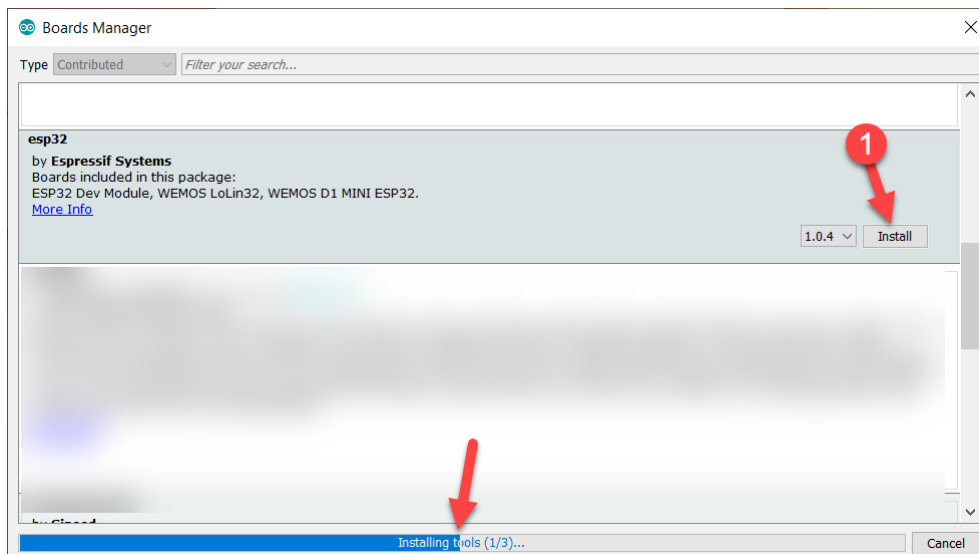
จากนั้นกด Tools > Board เลือก Boards Manager...



จากนั้นกด Type เลือก Contributed



เลือกหา esp32 แล้วกดปุ่ม Install

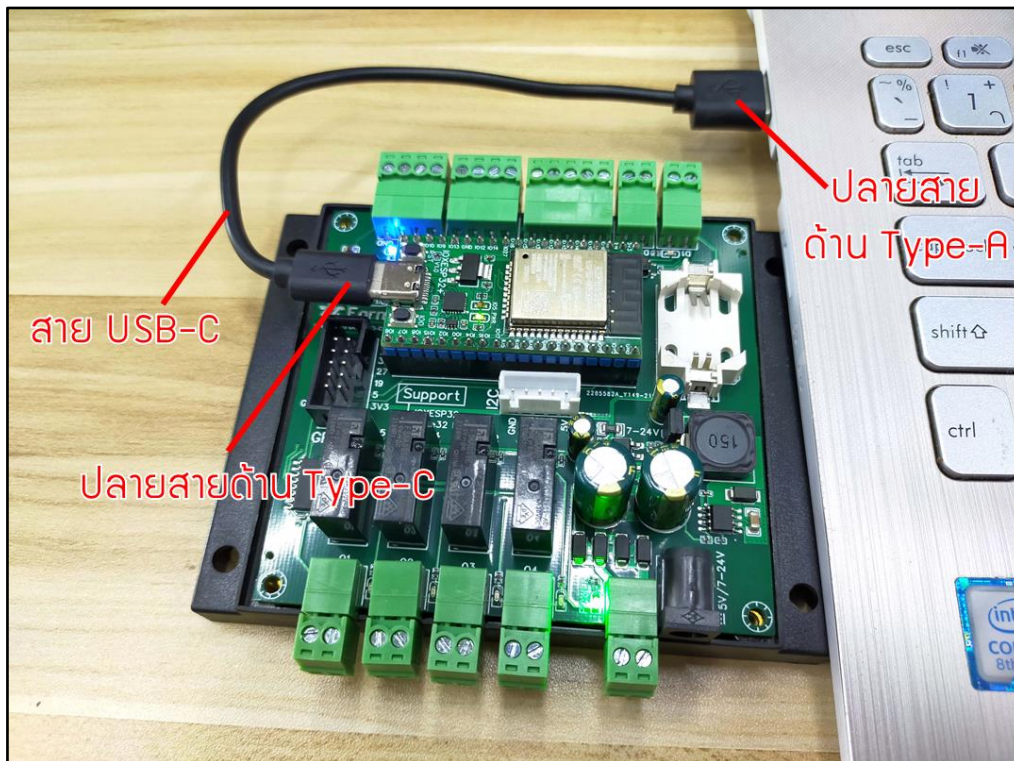


เมื่อติดตั้งเสร็จแล้ว จะแสดงคำว่า INSTALLED กดปุ่ม Close เพื่อปิดหน้าต่าง เป็นอันเสร็จสิ้นการเพิ่มบอร์ด ESP32 ลงโปรแกรม Arduino IDE

การเชื่อมต่อ Farm1 เข้ากับเครื่องคอมพิวเตอร์

ใช้สาย USB-C เสียบช่อง USB-C บนบอร์ด IOXESP32+ ปลายอีกด้านของสายเสียบเข้ากับเครื่องคอมพิวเตอร์

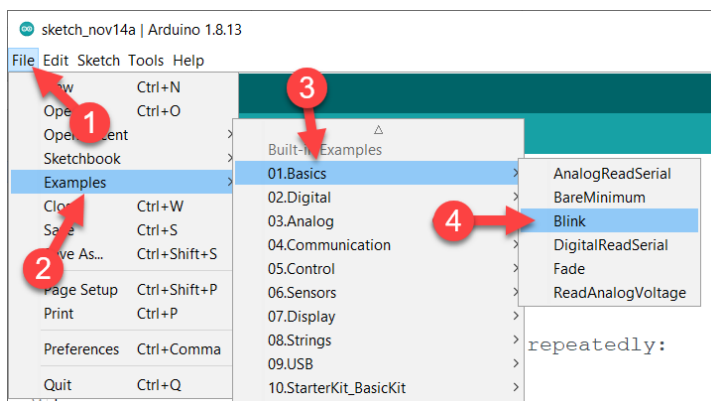
- สาย USB-C หาซื้อได้ทั่วไปตามร้านขายอุปกรณ์ IT, ร้านโทรศัพท์, ร้านสะดวกซื้อ, ห้างสรรพสินค้า



รูปที่ 24 การเชื่อมต่อ Farm1 เข้ากับเครื่องคอมพิวเตอร์ด้วยสาย USB-C

ทดสอบอัปโหลดโปรแกรมไฟกระพริบ

ที่โปรแกรม Arduino IDE กดไปที่ File > Examples > 01.Basics เลือก Blink



แก้ไข LED_BUILIN ให้เป็น 25 (ขา 01)

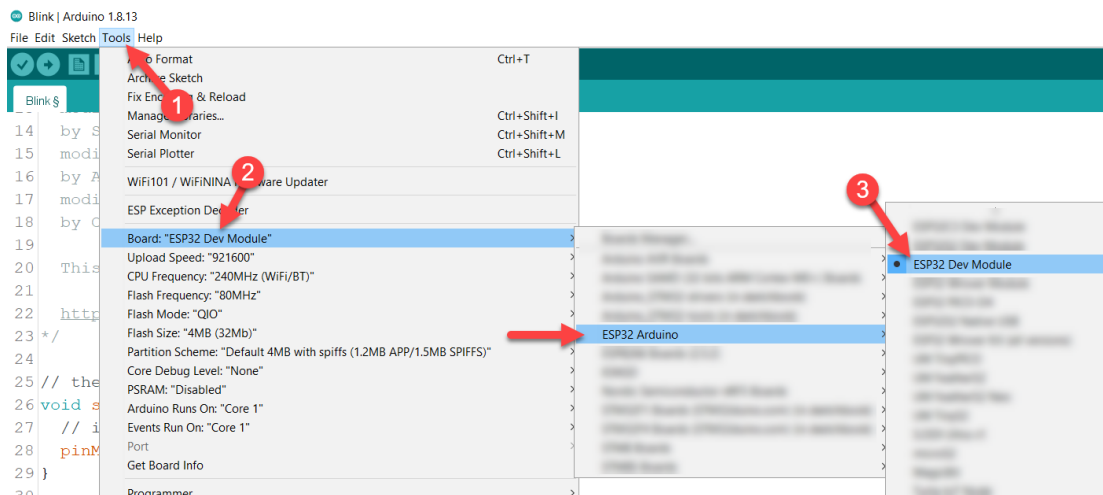
```

23 */
24
25 // the setup function runs once when you press
26 void setup() {
27   // initialize digital pin LED_BUILTIN as an
28   pinMode(LED_BUILTIN, OUTPUT);
29 }
30
31 // the loop function runs over and over again
32 void loop() {
33   digitalWrite(LED_BUILTIN, HIGH); // turn t
34   delay(1000); // wait f
35   digitalWrite(LED_BUILTIN, LOW); // turn t
36   delay(1000); // wait f
37 }
    
```

```

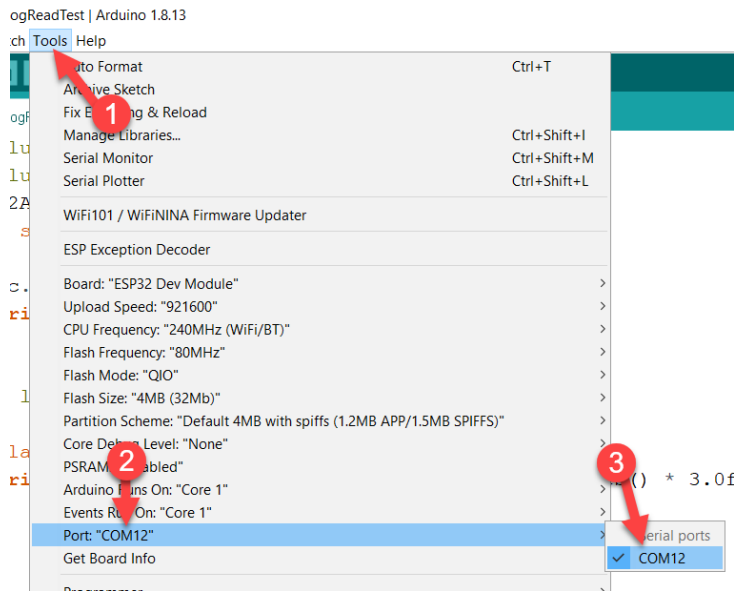
23 */
24
25 // the setup function runs once when you press
26 void setup() {
27   // initialize digital pin LED_BUILTIN as an
28   pinMode(25, OUTPUT);
29 }
30
31 // the loop function runs over and over again
32 void loop() {
33   digitalWrite(25, HIGH); // turn the LED on
34   delay(1000); // wait f
35   digitalWrite(25, LOW); // turn the LED of
36   delay(1000); // wait f
37 }
    
```


เลือกบอร์ดให้ถูกต้อง โดยกด Tool > Board > ESP32 Arduino เลือกบอร์ด ESP32 Dev Module

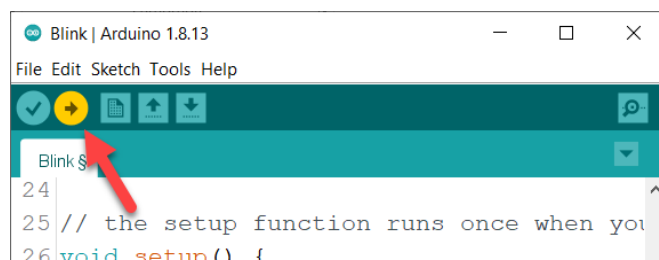


เลือกพอร์ตให้ถูกต้อง โดยกดไปที่ Tool > Port เลือก COM... ที่มีให้เลือก

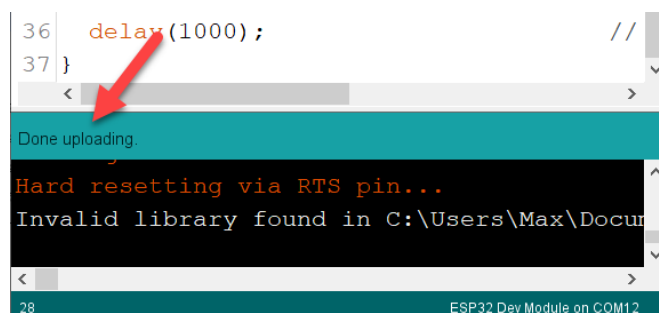
- หากไม่ขึ้นพอร์ตให้เลือก ให้ตรวจสอบสาย USB-C ว่าเสียบแน่นแล้วหรือยัง โดยสังเกตบอร์ด IOXESP32+ ไฟ PWR สีเขียวต้องติดสว่าง หากคาดว่าบอร์ดทำงานปกติ ให้ตรวจสอบการติดตั้งไดรเวอร์ FT231X อีกครั้ง



กดปุ่ม  เพื่อคอมไพล์และอัปโหลดโปรแกรมลงบอร์ด (เวลาที่ใช้ในการคอมไพล์และอัปโหลดอาจถึง 2 นาที หากไม่มีข้อความแจ้ง Error หรือ Done uploading ให้รอก่อน)



หากคอมไพล์และอัปโหลดโปรแกรมสำเร็จ จะมีข้อความ Done uploading แสดงขึ้นมา (ดังรูป)



ผลลัพธ์ที่ได้ ไฟที่ช่อง 01 จะกระพริบทุก ๆ 1 วินาที และมีเสียงรีเลย์ตัด-ต่อ

เขียนโปรแกรมสั่งงานช่องเอาต์พุต

ใช้คำสั่ง `pinMode()` และ `digitalWrite()` ในการเขียนโปรแกรมสั่งงานช่องเอาต์พุต

กำหนดแชนที่ `O1_PIN`, ..., `O4_PIN` ด้วยหมายเลขขาที่ต่อกับ ESP32 ดังนี้

```
1. #define O1_PIN 25
2. #define O2_PIN 14
3. #define O3_PIN 12
4. #define O4_PIN 13
```

ใน `void setup()` ใช้สั่ง `pinMode()` กำหนดขา O1 ถึง O4 เป็นเอาต์พุต

```
1. void setup{
2.   pinModeO1_PIN, OUTPUT;
3.   pinModeO2_PIN, OUTPUT;
4.   pinModeO3_PIN, OUTPUT;
5.   pinModeO4_PIN, OUTPUT;
6. }
```

ใน `void loop()` ใช้คำสั่ง `digitalWrite()` สั่งให้รีเลย์ช่อง O1 ตัด-ต่อทุก ๆ 1 นาที

```
1. void loop{
2.   digitalWriteO1_PIN, HIGH;
3.   delay(1000);
4.   digitalWriteO1_PIN, LOW;
5.   delay(1000);
6. }
```

ได้โค้ดโปรแกรมที่สมบูรณ์ดังนี้

```
1. #define O1_PIN 25
2. #define O2_PIN 14
3. #define O3_PIN 12
4. #define O4_PIN 13
5.
6. void setup{
7.   pinModeO1_PIN, OUTPUT;
8.   pinModeO2_PIN, OUTPUT;
9.   pinModeO3_PIN, OUTPUT;
10.  pinModeO4_PIN, OUTPUT;
11. }
12.
13. void loop{
14.  digitalWriteO1_PIN, HIGH;
15.  delay(1000);
16.  digitalWriteO1_PIN, LOW;
17.  delay(1000);
18. }
```

อัปโหลดโปรแกรมลงบอร์ด จะได้ยินเสียงรีเลย์ติด-ต่อน และไฟช่องเอาต์พุตติด-ดับ ทุก ๆ 2 วินาที

เขียนโปรแกรมอ่านช่องอินพุตดิจิทัล

ใช้คำสั่ง `pinMode()` และ `digitalRead()` ในการอ่านค่าช่องอินพุตดิจิทัล

กำหนดแทนที่ `O1_PIN, ..., O4_PIN` และ `D1_PIN, D2_PIN` ด้วยหมายเลขขาที่ต่อกับ ESP32 ดังนี้

```
1. #define O1_PIN 25
2. #define O2_PIN 14
3. #define O3_PIN 12
4. #define O4_PIN 13
5.
6. #define D1_PIN 34
7. #define D2_PIN 34
```

ใน `void setup()` ใช้สั่ง `pinMode()` กำหนดขา O1 ถึง O4 เป็นเอาต์พุต และกำหนดขา D1 และ D2 เป็นอินพุตดิจิทัล

```
1. void setup{
2.   pinMode(O1_PIN, OUTPUT);
3.   pinMode(O2_PIN, OUTPUT);
4.   pinMode(O3_PIN, OUTPUT);
5.   pinMode(O4_PIN, OUTPUT);
6.
7.   pinMode(D1_PIN, INPUT);
8.   pinMode(D2_PIN, INPUT);
9. }
```

ใน `void loop()` ใช้คำสั่ง `digitalRead()` อ่านค่าเก็บลงตัวแปร `value` ชนิด `int`

```
1. int value = digitalRead(D1_PIN);
```

ใช้คำสั่ง `if` ตรวจสอบค่าในตัวแปร `value` หากได้ 0 ให้สั่งรีเลย์ช่อง O1 ทำงาน ใช้คำสั่ง `else` หากเงื่อนไขใน `if` ไม่เป็นจริง ให้รีเลย์ช่อง O1 หยุดทำงาน

```
2. if(value == 0){
3.   digitalWrite(O1_PIN, HIGH);
4. } else {
5.   digitalWrite(O1_PIN, LOW);
6. }
```


ได้โค้ดโปรแกรมที่สมบูรณ์ดังนี้ (ดาวน์โหลดโค้ดโปรแกรมได้ที่)

```
1. #define O1_PIN 25
2. #define O2_PIN 14
3. #define O3_PIN 12
4. #define O4_PIN 13
5.
6. #define D1_PIN 34
7. #define D2_PIN 34
8.
9. void setup(){
10.  pinMode(O1_PIN, OUTPUT);
11.  pinMode(O2_PIN, OUTPUT);
12.  pinMode(O3_PIN, OUTPUT);
13.  pinMode(O4_PIN, OUTPUT);
14.
15.  pinMode(D1_PIN, INPUT);
16.  pinMode(D2_PIN, INPUT);
17. }
18.
19. void loop(){
20.  int value=digitalRead(D1_PIN);
21.  if(value==0){
22.    digitalWrite(O1_PIN, HIGH);
23.  } else {
24.    digitalWrite(O1_PIN, LOW);
25.  }
26.  delay(10);
27. }
```

อัปโหลดโปรแกรมลงบอร์ด ที่ช่อง D1 ด้านนี้ต่อเข้ากับ GND อีกด้านใช้สายลอย (ดังรูปที่ ...) และกับ 5V หรือ VIN สวิตไฟสี่ขั้วของช่อง D1 จะติดเมื่อแตะโดน และจะดับเมื่อนำออก ส่วนรีเลย์ช่อง O1 จะทำงานเมื่อแตะสายไฟ และหยุดทำงานเมื่อนำสายไฟออก

เขียนโปรแกรมอ่านช่องอินพุตแอนะล็อก

ใช้คำสั่ง analogRead() อ่านค่าช่องอินพุตแอนะล็อก ได้ค่า 0 ถึง 4095 เมื่อ 4095 เทียบได้กับแรงดัน 9.9V (ตามวงจรในหัวข้อ [ช่องอินพุตแอนะล็อก](#))

กำหนดแชนที่ A1 และ A2 ด้วย 36 และ 39 ตามลำดับ

```
1. #define A1_PIN 36
2. #define A2_PIN 39
```

ใน void loop() ใช้คำสั่ง analogRead() อ่านค่าแอนะล็อกจาก A0 แล้วนำค่าที่ได้เทียบด้วยสูตรคูณ
เพื่อแปลงเป็นแรงดัน เก็บลงตัวแปร volt ชนิด float


```
1. float volt = analogRead(A1_PIN) / 4095.0 * 9.9;
```

ใช้คำสั่ง Serial.println() ให้แสดงค่าในตัวแปร volt ออกมาทาง Serial Monitor

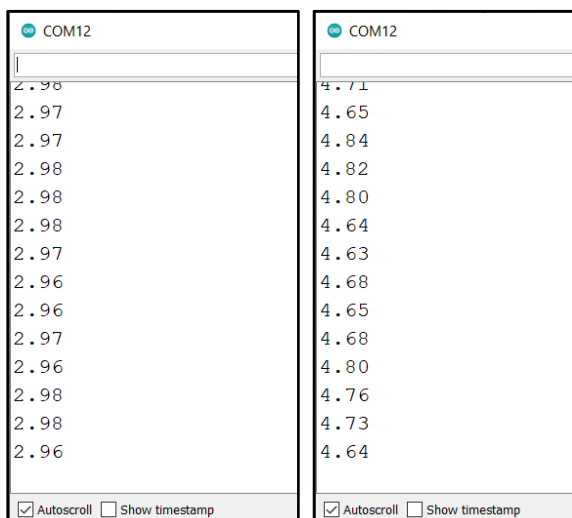
```
2. Serial.println(volt);
```

ได้โค้ดโปรแกรมที่สมบูรณ์ดังนี้

```
3. #define A1_PIN 36
4. #define A2_PIN 39
5.
6. void setup() {
7.   Serial.begin(115200);
8. }
9.
10. void loop() {
11.   float volt = analogRead(A1_PIN) / 4095.0 * 9.9;
12.   Serial.println(volt);
13.   delay(10);
14. }
```

อัปโหลดโค้ดโปรแกรมลงบอร์ด เปิด Serial Monitor โดยกด  ตั้ง band เป็น 115200 แล้วกดลอง
จัมระหว่าง A1 กับ 5V จากนั้นทดสอบจัมระหว่าง A1 กับ 3.3V สังเกตผลที่ได้ใน Serial Monitor จะ
ต่างกัน

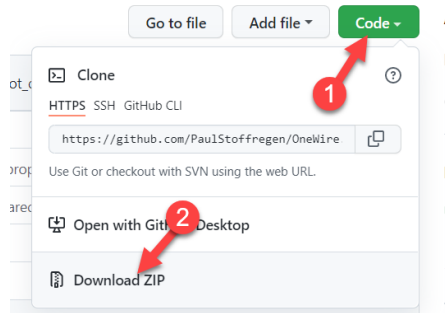
■ **วงจรส่วนแอนะล็อกใช้ไฟจาก VIN จึงจำเป็นต้องต่อไฟเข้า VIN เพื่อให้วงจรแอนะล็อกทำงานได้ถูกต้อง**



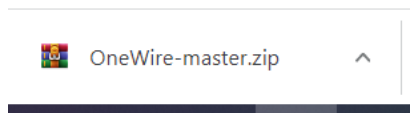
เขียนโปรแกรมช่อง I/O อ่านค่าอุณหภูมิจาก DS18B20

ต่อวงจรตามหัวข้อ [ช่อง I/O](#) ติดตั้งไลบรารี OneWire ตามขั้นตอนดังนี้

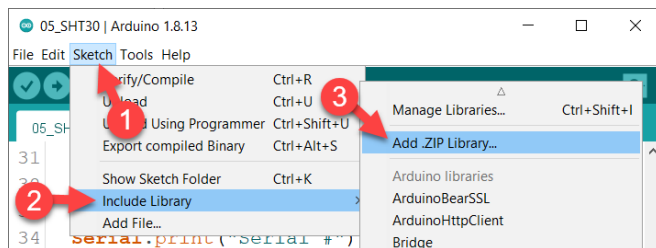
เข้าไปที่ <https://github.com/PaulStoffregen/OneWire> กด Code แล้วกด Download ZIP



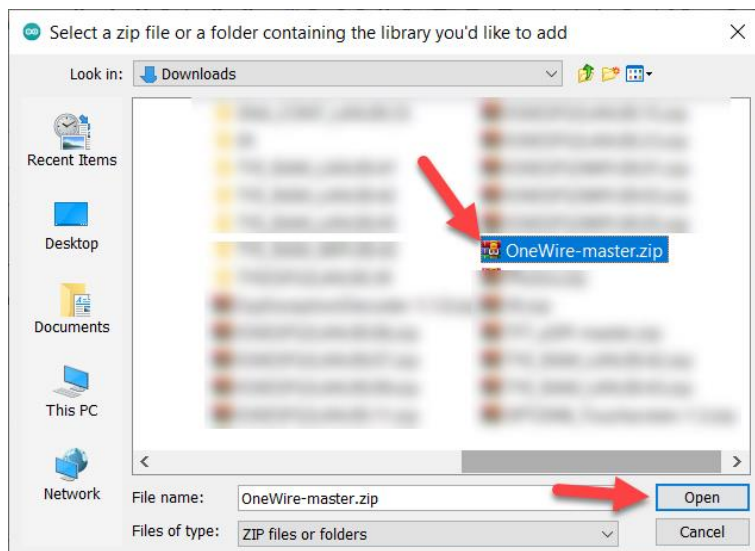
จะได้ไฟล์ OneWire-master.zip มา



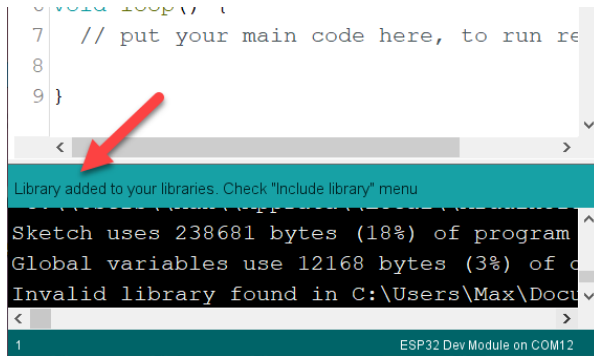
ที่โปรแกรม Arduino IDE กด Sketch > Include Library เลือก Add .ZIP Library



เลือกไฟล์ OneWire-master.zip ที่ดาวน์โหลดมา กดปุ่ม Open

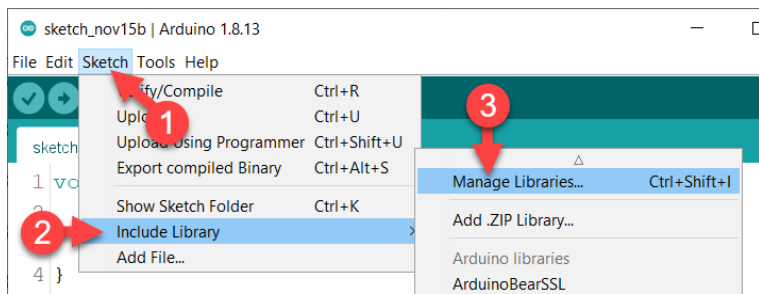


หากติดตั้งสำเร็จ จะมีข้อความ Library added ขึ้น (ดังรูป)

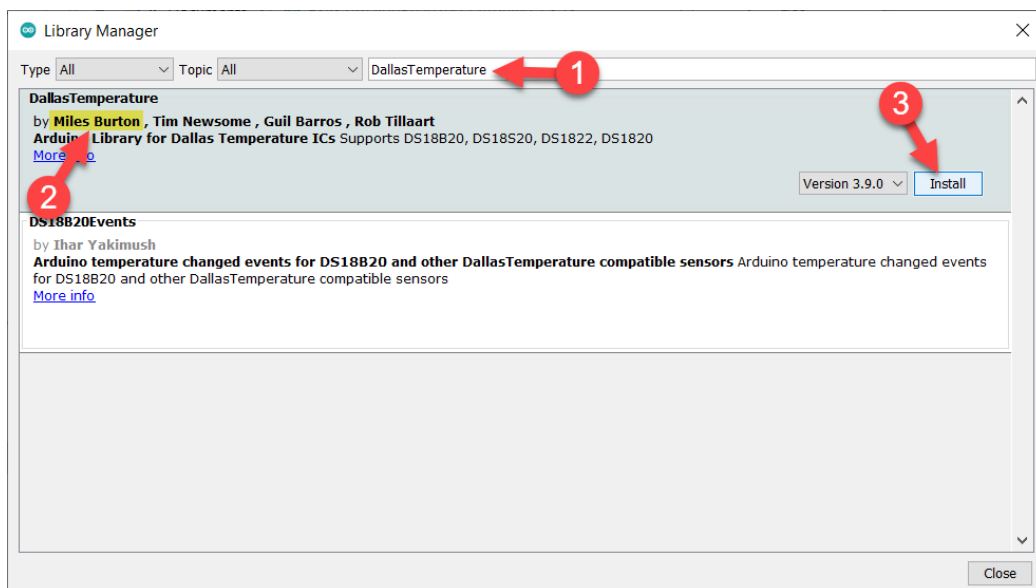


ติดตั้งไลบรารี DallasTemperature ด้วย Library Manager ตามขั้นตอนดังนี้

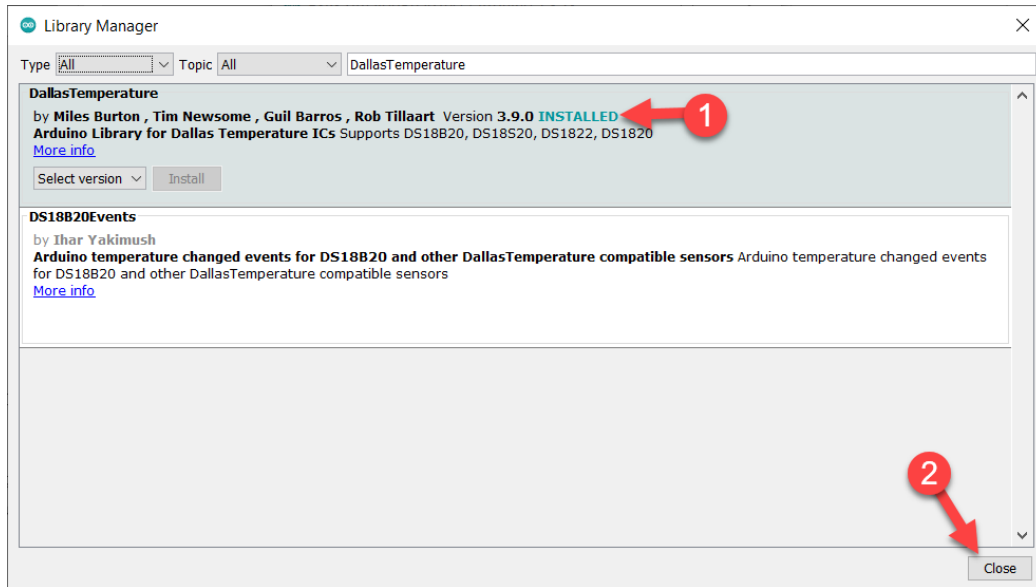
กด Sketch > Include Library เลือก Manage Libraries



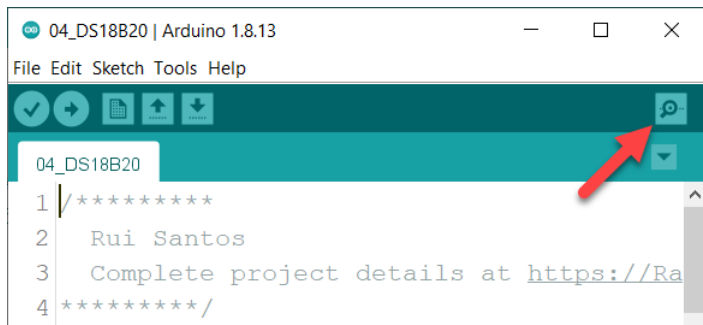
หน้าต่าง Library Manager จะแสดงขึ้นมา ให้ค้นหา DallasTemperature แล้วเลือกกดปุ่ม Install ไลบรารี by Miles Burton



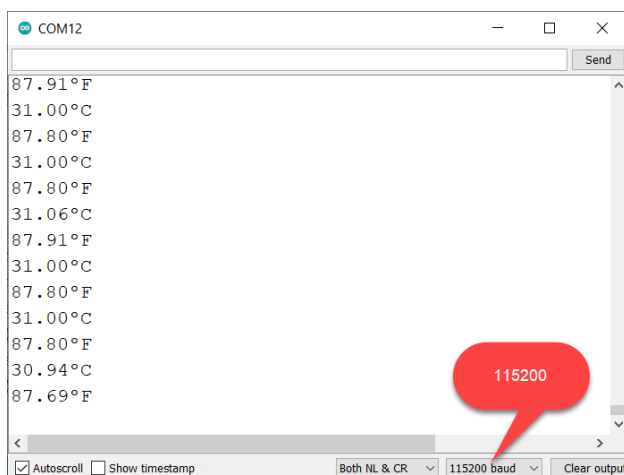
เมื่อติดตั้งเสร็จแล้ว จะมีข้อความ INSTALLED แสดง (ดังรูป) กดปุ่ม Close เพื่อปิดได้เลย



ดาวน์โหลดโปรแกรมตัวอย่างจาก <https://bit.ly/3HkpJx5> อัปโหลดโปรแกรมลงบอร์ด แล้วเปิด Serial Monitor ขึ้นมา โดยกด 

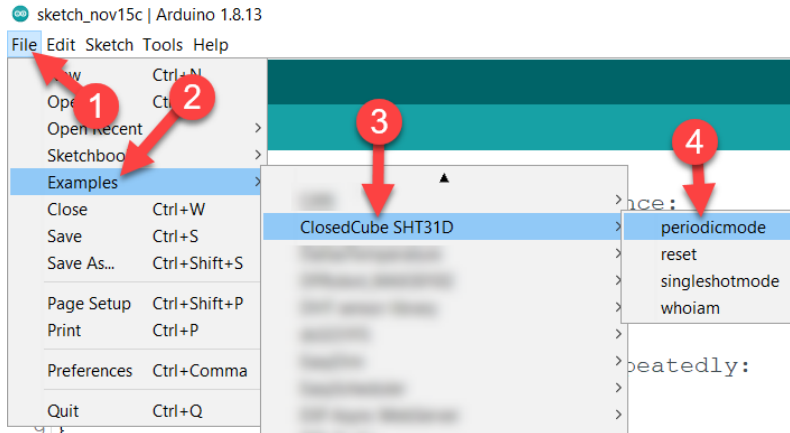


เลือก 115200 baud หากต่อวงจรถูกต้อง เช่นเซอร์กัมวาบได้ จะมีอุณหภูมิในหน่วยองศาเซลเซียส และองศาฟาเรนไฮต์แสดงขึ้นมา



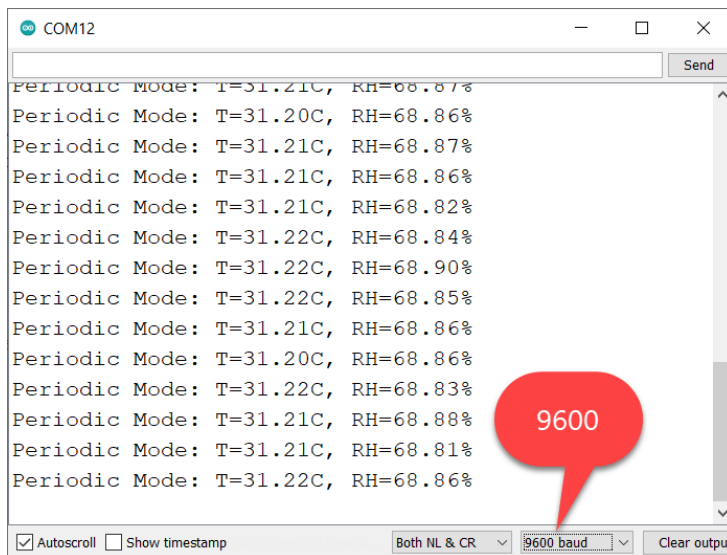
เขียนโปรแกรมช่อง I²C อ่านค่าอุณหภูมิและความชื้นจาก SHT30

ต่อวงจรตามหัวข้อ [ช่อง I2C](#) ติดตั้งไลบรารี ClosedCube SHT31D ด้วย Library Manager จากนั้นเปิดโค้ดโปรแกรมตัวอย่าง periodicmode ขึ้นมา



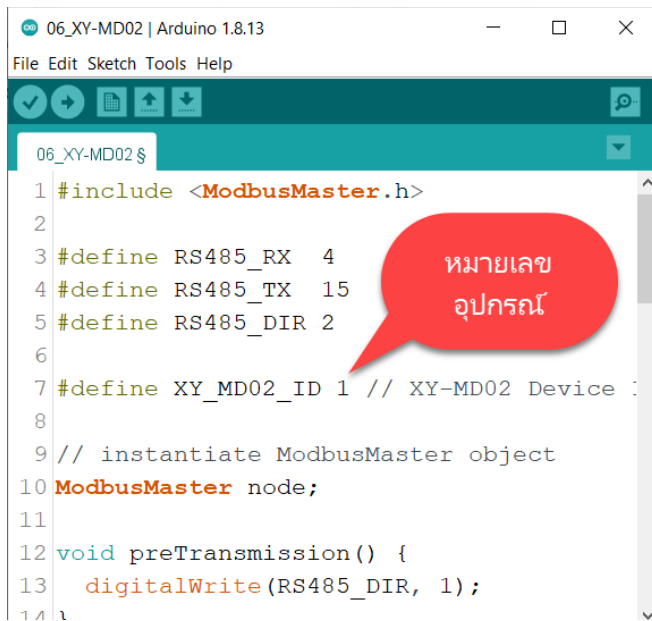
อัปโหลดโปรแกรมลงบอร์ด เปิด Serial Monitor ขึ้นมา ปรับ band เป็น 9600 ค่าอุณหภูมิและความชื้นที่ SHT30 วัดได้จะแสดงขึ้นมา

■ หากมีข้อความ Periodic Mode: [ERROR] ให้แก้ไขหมายเลขของอุปกรณ์บอร์ดที่ 32 จาก 0x44 เป็น 0x45 อัปโหลดโปรแกรมแล้วดูผลอีกครั้ง



เขียนโปรแกรมช่อง RS485 อ่านค่าอุณหภูมิและความชื้นจาก XY-MD02

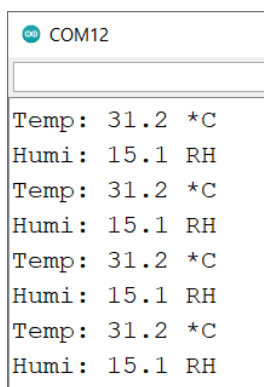
ต่อวงจรตามหัวข้อ [ช่อง RS485](#) ติดตั้งไลบรารี ModbusMaster ด้วย Library Manager จากนั้นดาวน์โหลดโค้ดตัวอย่างจาก <https://bit.ly/3ccUxRV> แก้ไขหมายเลขอุปกรณ์ (Device ID) ของ XY-MD02 ให้ถูกต้อง (ค่าจากโรงงาน เป็น ID: 1) อัปโหลดโปรแกรมลงบอร์ด



```
06_XY-MD02 | Arduino 1.8.13
File Edit Sketch Tools Help
06_XY-MD02 §
1 #include <ModbusMaster.h>
2
3 #define RS485_RX 4
4 #define RS485_TX 15
5 #define RS485_DIR 2
6
7 #define XY_MD02_ID 1 // XY-MD02 Device ID
8
9 // instantiate ModbusMaster object
10 ModbusMaster node;
11
12 void preTransmission() {
13   digitalWrite(RS485_DIR, 1);
14 }
```

หมายเลขอุปกรณ์

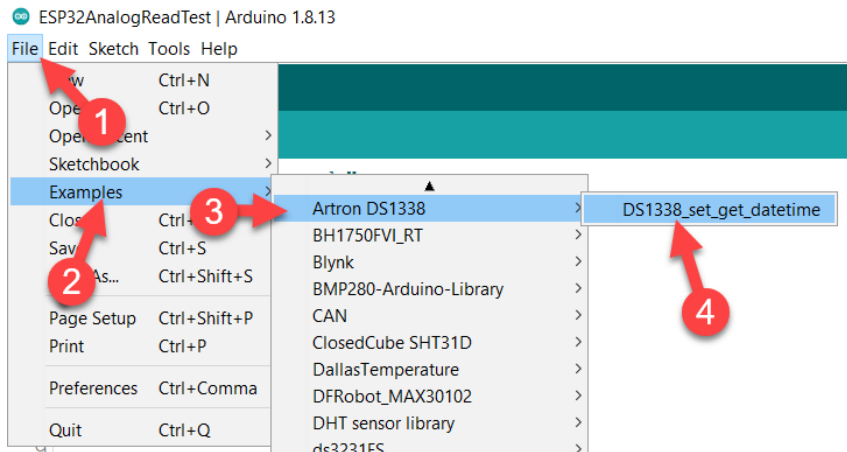
เปิด Serial Monitor ขึ้นมา ปรับ Band เป็น 115200 หากต่อวงจรถูกต้อง เซ็นเซอร์ทำงานได้ ค่าอุณหภูมิและความชื้นที่วัดได้จะแสดงขึ้นมา



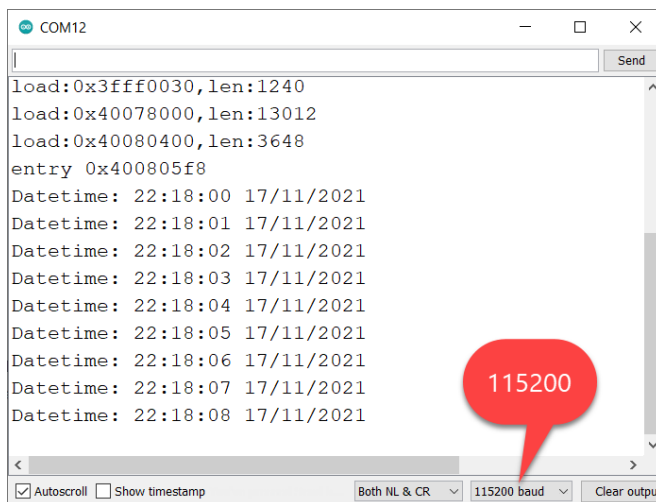
```
COM12
Temp: 31.2 *C
Humi: 15.1 RH
Temp: 31.2 *C
Humi: 15.1 RH
Temp: 31.2 *C
Humi: 15.1 RH
Temp: 31.2 *C
Humi: 15.1 RH
```

เขียนโปรแกรมตั้งเวลาและอ่านค่าเวลาจากวงจรนับเวลา (RTC)

ติดตั้งไลบรารี Artron DS1338 ด้วย Library Manager เปิดตัวอย่างโค้ดจากเมนู File > Examples > Artron DS1338 > DS1338_set_get_datetime



อัปโหลดโปรแกรมตัวอย่างลงบอร์ด เปิด Serial Monitor ปรับ 115200 baud ผลที่ได้คือ เวลาใน RTC จะถูกตั้ง และแสดงเวลาที่นับขึ้นเรื่อย ๆ บน Serial Monitor



คำแนะนำการติดตั้ง

อุปกรณ์รองรับการติดตั้งบนราวปีกนก ไม่กันน้ำ การใช้งานจริงจึงควรติดตั้งบนราวปีกนกในกล่องกันน้ำ รู้อย่างไฟที่เข้ากล่องควรใช้คอนดนเคเตอร์แบบกันน้ำ หรือใช้เคเบิลเกลนต์ เพื่อป้องกันน้ำเข้าภายในกล่อง ควรรักษาอุณหภูมิให้ไม่เกิน 80 องศาเซลเซียส ซึ่งอาจทำให้อุปกรณ์อิเล็กทรอนิกส์เสียหาย การเก็บรักษาควรเก็บในที่แห้ง ไม่โดนแดดโดยตรง

พาวเวอร์ซัพพลายควรเลือกใช้รุ่นที่สามารถทนไฟกระชากได้ มีระบบตัดไฟเมื่อกระแสเกิน มีอุปกรณ์ป้องกันฟ้าผ่า โดยอาจสังเกตุและเลือกใช้พาวเวอร์ซัพพลายที่มีสัญลักษณ์ CE หรือ มอก. พร้อมติดตั้งเบรกเกอร์เพื่อใช้ปิดอุปกรณ์เมื่อต้องบำรุงรักษา

แนวทางการแก้ไขปัญหา

ปัญหา	แนวทางการแก้ไข
ปัญหาเอาต์พุต	
ใช้คำสั่ง <code>digitalWrite()</code> แต่รีเลย์ไม่ทำงาน ไฟเอาต์พุตไม่ติด	<p>เป็นได้ทั้งปัญหาซอฟต์แวร์และปัญหาด้านฮาร์ดแวร์</p> <ul style="list-style-type: none"> • อาจลืมใช้คำสั่ง <code>pinMode()</code> หรือใช้ไม่ถูกต้อง ก่อนเรียกใช้ <code>digitalWrite()</code> • ใช้คำสั่ง <code>digitalWrite()</code> ผิดขา • ESP32 รีเซ็ตก่อนทำงานถึงคำสั่ง <code>digitalWrite()</code> (ให้เปิด Serial Monitor ขึ้นมาดู) • บอร์ด IOXESP32+ หลวม ทำให้ไฟฟฟ้าไม่สามารถไหลได้ • กระแสไฟฟ้าจากแหล่งจ่ายไม่เพียงพอให้อุปกรณ์ทำงานได้ปกติ
ไฟติด-ดับ แต่ไม่มีเสียงรีเลย์ติด-ต่อ	<ul style="list-style-type: none"> • รีเลย์เสีย
ปัญหาอินพุตดิจิทัล	
ใช้คำสั่ง <code>digitalRead()</code> ได้ค่าเป็น 1 หรือเป็น 0 ตลอดเวลา	<ul style="list-style-type: none"> • ให้สังเกตไฟตรจูดเอาต์พุต หากต่อวงจรให้ต้องเป็น 1/0 แต่ไฟไม่ติด-ไม่ดับ เป็นปัญหาด้านฮาร์ดแวร์หรือระบบไฟฟ้า ให้ใช้มัลติมิเตอร์ตรวจสอบแรงดันแต่ละจุด • หากไฟติด-ดับปกติ ให้ตรวจสอบโค้ดโปรแกรมว่าอ่านออกมาถูกต้อง • ตรวจสอบว่าใช้คำสั่ง <code>pinMode()</code> ก่อนใช้คำสั่ง <code>digitalRead()</code> แล้ว

ปัญหาอินพุตแอนะล็อก	
วัดแรงดันได้สูงสุด 2V กว่า ๆ แม้จะป้อนแรงดันมากกว่านี้	เกิดจากไม่ได้จ่ายไฟเข้า VIN 7V ถึง 12V วงจรอินพุตแอนะล็อกใช้ไฟจาก VIN ในการทำงาน หากไม่จ่ายไฟ ค่าที่วัดได้จะไม่ถูกต้อง
วัดแรงดันได้ไม่ตรง	เป็นปัญหาในระดับชิป ให้ใช้ไลบรารี ESP32AnalogRead อ่านแรงดันแทน จะให้ค่าที่ตรงความเป็นจริงมากขึ้น
ช่อง I ² C	
ติดต่อกับอุปกรณ์ I ² C ไม่ได้	สังเกตไฟช่อง I ² C ว่ากระพริบหรือไม่ หากกระพริบแสดงว่ามีการส่งข้อมูลออกจากบอร์ดแล้ว แต่อุปกรณ์ปลายทางอาจไม่ได้รับ หรือไม่ทำงาน หากไฟไม่กระพริบ แสดงว่าไม่มีข้อมูลส่งออกจากบอร์ด ให้ตรวจสอบโค้ดโปรแกรม
ช่อง RS485	
ติดต่อกับอุปกรณ์ RS485 ไม่ได้	สังเกตไฟช่อง RS485 ตรง TX ว่ากระพริบหรือไม่ หากกระพริบแสดงว่ามีการส่งข้อมูลออกจากบอร์ดแล้ว จากนั้นให้สังเกตไฟ RX ว่าทำงานอย่างไร <ul style="list-style-type: none"> ไฟ RX กระพริบ แสดงว่าอุปกรณ์ปลายทางส่งข้อมูลเข้ามาแล้ว แต่ฝั่ง Farm1 รับข้อมูลไม่ได้ ให้ตรวจสอบโค้ดโปรแกรม ไฟ RX ไม่กระพริบ แสดงว่าอุปกรณ์ปลายทางไม่ตอบสนอง อาจเป็นไปได้ว่าหมายเลข ID ของอุปกรณ์ที่ร้องขอข้อมูลไม่มีอยู่ หรือ ID ของอุปกรณ์ผิด ให้ใช้โปรแกรม CAS Modbus Scanner ลองค้นหา ID ดูก่อน รวมทั้งอาจตรวจสอบการตั้งค่า Band Rate และ

	<p>Prarity ด้วย ว่าตั้งค่าไว้ตรงกับที่ อุปกรณ์ต้องการหรือไม่</p> <p>หากไฟ TX ไม่กระพริบ</p> <ul style="list-style-type: none"> • ยืมส่งข้อมูลออกมาไม่ได้ ให้ตรวจสอบโค้ด โปรแกรม
<p>ปัญหาจนวนับเวลา</p>	
<p>เวลาไม่ถูกต้องหลังตัดไฟ</p>	<p>ไม่ได้ใส่แบตเตอรี่ หรือแบตเตอรี่หมด</p>

บริษัท อาร์ทรอน ซอป จำกัด รับพัฒนาสินค้าอิเล็กทรอนิกส์ บอร์ดอิเล็กทรอนิกส์ ด้านระบบ IoT พัฒนาเว็บไซต์ระบบ IoT ด้วย ReactJS / Next.js รับทำระบบหลังบ้านให้อุปกรณ์ IoT รับเขียนเฟิร์มแวร์อุปกรณ์ IoT ด้วย Arduino IDE / PlatformIO / Atmel Studio / MPLAB IDE / STM32CubeIDE / IAR / ESP-IDF / Arm Keil รับจัดหาชิ้นส่วนอิเล็กทรอนิกส์ ออกใบเสนอราคา และใบกำกับภาษีได้

ติดต่อสอบถามข้อมูลเพิ่มเติม

- เว็บไซต์ www.artronshop.co.th
- โทร: 02 003 3688 (สำนักงาน)
- อีเมล: contact@artronshop.co.th
- LINE: @artronshop
- Facebook: [ArtronShop](https://www.facebook.com/ArtronShop)
- Youtube: [ArtronShop](https://www.youtube.com/ArtronShop)

