แบบฝึกหัดภาคปฏิบัติ บทที่ 2

วิชาอินเทอร์เน็ตของทุกสิ่ง (Internet of Things)

โดย ผศ. ดร. นัฐพงศ์ ส่งเนียม http://www.siam2dev.com

สาขาวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มรภ. พระนคร

ความรู้พื้นฐานเกี่ยวกับอุปกรณ์ IoT และการใช้งาน Arduino IDE

คำชี้แจง: ให้นักศึกษาปฏิบัติงานโดยใช้บอร์ด Arduino UNO R3, Arduino IDE และอุปกรณ์พื้นฐานตามที่กำหนด พร้อมบันทึกผลการทดลอง โค้ดโปรแกรม และภาพการต่อวงจร

ข้อที่ 1 การติดตั้งและตั้งค่า Arduino IDE

วัตถุประสงค์:

เพื่อให้นักศึกษาสามารถติดตั้ง Arduino IDE และตั้งค่าบอร์ด Arduino UNO R3 ได้ถูกต้อง

อุปกรณ์ที่ใช้:

- คอมพิวเตอร์
- Arduino UNO R3
- สาย USB
- โปรแกรม Arduino IDE

ขั้นตอนปฏิบัติ:

1. ดาวน์โหลดและติดตั้ง Arduino IDE
2. เชื่อมต่อ Arduino UNO R3 กับคอมพิวเตอร์ด้วยสาย USB
3. เปิด Arduino IDE
4. เลือกบอร์ดเป็น Arduino Uno
5. เลือก Port ให้ตรงกับบอร์ดที่เชื่อมต่อ
6. ทดลองกด Verify และ Upload โปรแกรมตัวอย่าง

ผลลัพธ์ที่ต้องการ:

Arduino IDE สามารถเชื่อมต่อกับบอร์ดได้ และอัปโหลดโปรแกรมได้สำเร็จ

สิ่งที่ต้องส่ง:

ภาพหน้าจอ Arduino IDE ที่เลือก Board และ Port ถูกต้อง

ข้อที่ 2 ทดลองโปรแกรม Blink LED

วัตถุประสงค์:

เพื่อให้นักศึกษาสามารถเขียนโปรแกรมเปิด–ปิด LED เบื้องต้นได้

อุปกรณ์ที่ใช้:

- Arduino UNO R3
- LED 1 ดวง
- ตัวต้านทาน 220Ω
- Breadboard
- Jumper Wire

ขั้นตอนปฏิบัติ:

1. ต่อ LED เข้ากับขา Digital Pin 13
2. ต่อขา Cathode ของ LED ลง GND
3. เขียนโปรแกรมให้ LED ติด 1 วินาที และดับ 1 วินาที
4. อัปโหลดโปรแกรมลงบอร์ด
5. สังเกตผลการทำงาน

ผลลัพธ์ที่ต้องการ:

LED กระพริบติด–ดับทุก ๆ 1 วินาที

โจทย์เพิ่มเติม:

เปลี่ยนเวลา delay() เป็น 500 ms และ 2000 ms แล้วเปรียบเทียบผล

ข้อที่ 3 ใช้ตัวแปรกำหนดหมายเลขขา LED

วัตถุประสงค์:

เพื่อให้นักศึกษาเข้าใจการใช้ตัวแปรในการเก็บหมายเลขขา Arduino

อุปกรณ์ที่ใช้:

- Arduino UNO R3
- LED 1 ดวง
- ตัวต้านทาน 220Ω
- Breadboard

- Jumper Wire

เงื่อนไขของโปรแกรม:

1. ประกาศตัวแปร int ledPin = 13;
2. ใช้ตัวแปร ledPin แทนหมายเลขขาในคำสั่ง pinMode() และ digitalWrite()
3. ให้ LED ติด 0.5 วินาที และดับ 0.5 วินาที

ผลลัพธ์ที่ต้องการ:

LED กระพริบโดยใช้ตัวแปรควบคุมหมายเลขขา

คำถามท้ายกิจกรรม:

การใช้ตัวแปร ledPin มีข้อดีอย่างไรเมื่อเทียบกับการเขียนเลขขาโดยตรง?

ข้อที่ 4 แสดงข้อความผ่าน Serial Monitor

วัตถุประสงค์:

เพื่อให้นักศึกษาสามารถใช้งาน Serial Monitor เบื้องต้นได้

อุปกรณ์ที่ใช้:

- Arduino UNO R3
- สาย USB
- Arduino IDE

ขั้นตอนปฏิบัติ:

1. เปิด Serial Monitor
2. เขียนโปรแกรมให้แสดงข้อความ "Hello Arduino" ทุก ๆ 1 วินาที
3. ตั้งค่า Baud rate ให้ตรงกับ Serial.begin(9600);
4. สังเกตผลลัพธ์บน Serial Monitor

ผลลัพธ์ที่ต้องการ:

Serial Monitor แสดงข้อความ "Hello Arduino" ซ้ำทุก ๆ 1 วินาที

โจทย์เพิ่มเติม:

ให้เพิ่มการแสดงชื่อนักศึกษาและรหัสนักศึกษาใน Serial Monitor

ข้อที่ 5 ใช้ if...else ตรวจสอบค่าตัวแปร

วัตถุประสงค์:

เพื่อให้นักศึกษาเข้าใจการใช้คำสั่งควบคุมเงื่อนไข if...else

เงื่อนไขของโปรแกรม:

1. ประกาศตัวแปร int score = 75;
2. ถ้า score >= 50 ให้แสดงข้อความ "Pass"
3. ถ้าไม่ใช่ ให้แสดงข้อความ "Fail"
4. แสดงผลผ่าน Serial Monitor ทุก ๆ 2 วินาที

ผลลัพธ์ที่ต้องการ:

ถ้าคะแนนมากกว่าหรือเท่ากับ 50 จะแสดง "Pass" ถ้าน้อยกว่า 50 จะแสดง "Fail"

โจทย์เพิ่มเติม:

เปลี่ยนค่า score เป็น 45, 60 และ 90 แล้วบันทึกผลลัพธ์

ข้อที่ 6 ควบคุม LED ด้วยปุ่มกดและ if...else

วัตถุประสงค์:

เพื่อให้นักศึกษาสามารถอ่านค่าจากปุ่มกดและใช้เงื่อนไขควบคุม LED ได้

อุปกรณ์ที่ใช้:

- Arduino UNO R3
- Push Button 1 ตัว
- LED 1 ดวง
- ตัวต้านทาน 10kΩ สำหรับ Pull-down
- ตัวต้านทาน 220Ω
- Breadboard
- Jumper Wire

การต่อวงจร:

- ปุ่มกดต่อเข้ากับ Pin 2
- ใช้ตัวต้านทาน 10kΩ ต่อจาก Pin 2 ลง GND แบบ Pull-down
- LED ต่อเข้ากับ Pin 13 ผ่านตัวต้านทาน 220Ω

เงื่อนไขของโปรแกรม:

1. ถ้ากดปุ่ม ให้ LED ติด
2. ถ้าไม่กดปุ่ม ให้ LED ดับ
3. ใช้คำสั่ง digitalRead() อ่านค่าจากปุ่ม

4. ใช้คำสั่ง if...else ตรวจสอบสถานะปุ่ม

ผลลัพธ์ที่ต้องการ:

เมื่อกดปุ่ม LED ติด และเมื่อปล่อยปุ่ม LED ดับ

ข้อที่ 7 ใช้ for loop แสดงข้อความซ้ำ

วัตถุประสงค์:

เพื่อให้นักศึกษาเข้าใจการใช้คำสั่ง for ในการทำงานซ้ำแบบรู้จำนวนรอบ

เงื่อนไขของโปรแกรม:

1. ใช้ for แสดงข้อความ "Internet of Things" ผ่าน Serial Monitor จำนวน 10 ครั้ง
2. แสดงลำดับรอบ เช่น Round 1, Round 2, ... Round 10
3. หน่วงเวลาแต่ละรอบ 500 ms

ผลลัพธ์ที่ต้องการ:

Serial Monitor แสดงข้อความจำนวน 10 รอบ พร้อมเลขลำดับรอบ

โจทย์เพิ่มเติม:

แก้โปรแกรมให้แสดงข้อความจำนวน 20 รอบ

ข้อที่ 8 ควบคุม LED หลายดวงด้วย for loop

วัตถุประสงค์:

เพื่อให้นักศึกษาสามารถใช้ array และ for loop ควบคุม LED หลายดวงได้

อุปกรณ์ที่ใช้:

- Arduino UNO R3
- LED 5 ดวง
- ตัวต้านทาน 220Ω จำนวน 5 ตัว
- Breadboard
- Jumper Wire

การต่อวงจร:

- LED1 ต่อกับ Pin 2
- LED2 ต่อกับ Pin 3
- LED3 ต่อกับ Pin 4

- LED4 ต่อกับ Pin 5
- LED5 ต่อกับ Pin 6
- LED ทุกดวงต่อผ่านตัวต้านทาน 220Ω และลง GND

เงื่อนไขของโปรแกรม:

1. ใช้ int ledPins[] = {2, 3, 4, 5, 6};
2. ใช้ for ใน setup() เพื่อกำหนดทุกขาเป็น OUTPUT
3. ใช้ for ใน loop() เพื่อเปิด–ปิด LED ทีละดวง
4. หน่วงเวลาแต่ละดวง 200 ms

ผลลัพธ์ที่ต้องการ:

LED ติดไล่จาก Pin 2 ไป Pin 6 คล้ายไฟวิ่ง

โจทย์เพิ่มเติม:

ปรับให้ LED วิ่งกลับจาก Pin 6 ไป Pin 2

ข้อที่ 9 ใช้ while ตรวจสอบค่าจาก Sensor จำลอง

วัตถุประสงค์:

เพื่อให้นักศึกษาเข้าใจการใช้ while ตรวจสอบเงื่อนไขซ้ำ

อุปกรณ์ที่ใช้:

- Arduino UNO R3
- Potentiometer หรือ Sensor แบบ Analog
- Breadboard
- Jumper Wire

การต่อวงจร:

- ขากลางของ Potentiometer ต่อกับ A0
- ขาอีกด้านต่อ 5V
- ขาอีกด้านต่อ GND

เงื่อนไขของโปรแกรม:

1. อ่านค่าจาก A0 ด้วย analogRead(A0)
2. ถ้าค่าน้อยกว่า 500 ให้แสดงข้อความ "Value is low" ผ่าน Serial Monitor
3. ถ้าค่ามากกว่าหรือเท่ากับ 500 ให้แสดงข้อความ "Value is normal"

4. ทดลองหมุน Potentiometer แล้วสังเกตผล

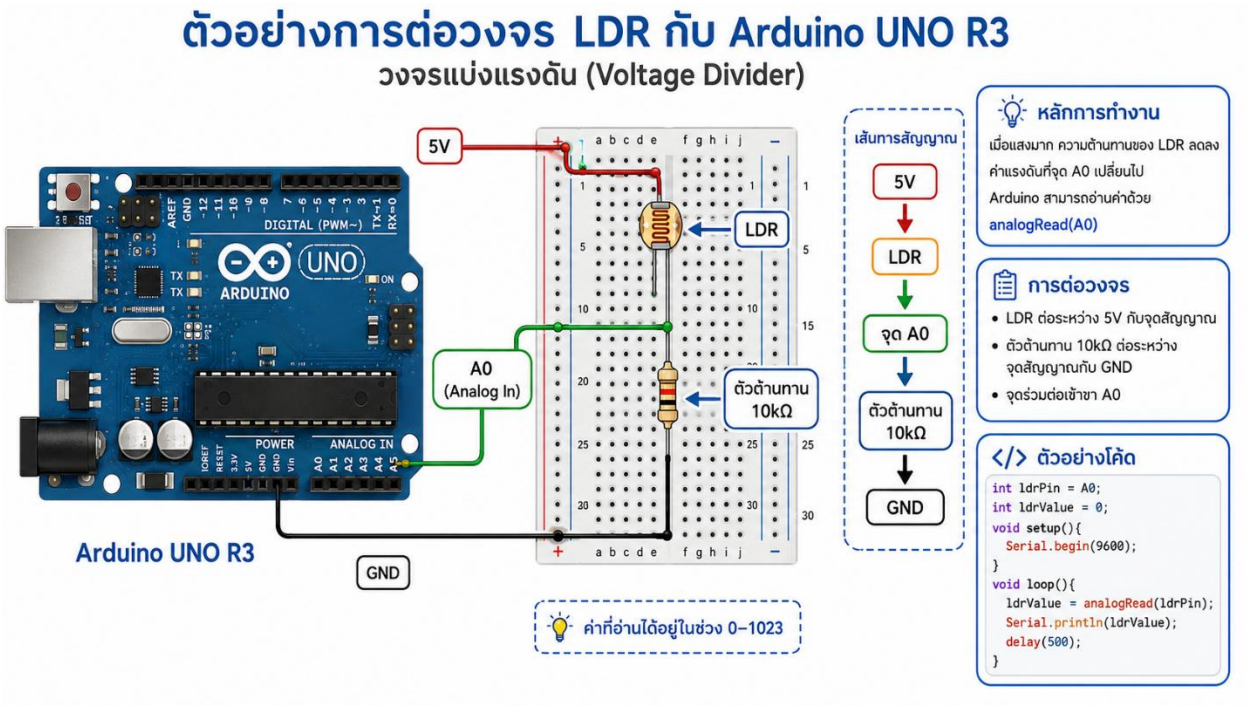
ผลลัพธ์ที่ต้องการ:

Serial Monitor แสดงข้อความตามค่าที่อ่านได้จาก A0

ข้อที่ 10 Mini Project: ระบบควบคุมไฟอัตโนมัติจำลอง

วัตถุประสงค์:

เพื่อให้นักศึกษาประยุกต์ใช้ความรู้เรื่องตัวแปร, if...else, loop, Sensor และ LED ในงาน IoT เบื้องต้น



อุปกรณ์ที่ใช้:

- Arduino UNO R3
- LDR หรือ Potentiometer 1 ตัว
- LED 1 ดวง
- ตัวต้านทาน 220Ω
- ตัวต้านทาน 10kΩ
- Breadboard
- Jumper Wire

เงื่อนไขของระบบ:

1. อ่านค่าความสว่างจากขา Analog A0
2. ถ้าค่าที่อ่านได้น้อยกว่าค่าที่กำหนด ให้ LED ติด
3. ถ้าค่าที่อ่านได้มากกว่าหรือเท่ากับค่าที่กำหนด ให้ LED ดับ
4. แสดงค่าที่อ่านได้ผ่าน Serial Monitor
5. ใช้ตัวแปรกำหนดค่า Threshold เช่น int threshold = 500;

ผลลัพธ์ที่ต้องการ:

เมื่อสภาพแสงน้อย LED จะติด และเมื่อสภาพแสงมาก LED จะดับ

สิ่งที่ต้องส่ง:

1. ภาพการต่อวงจร
2. โค้ดโปรแกรม
3. ภาพผลลัพธ์จาก Serial Monitor
4. คำอธิบายการทำงานของระบบ
5. ปัญหาที่พบและแนวทางแก้ไข

เกณฑ์การประเมินแบบฝึกหัดภาคปฏิบัติ

รายการประเมิน	คะแนน
ต่อวงจรถูกต้องและปลอดภัย	20
เขียนโค้ดถูกต้องตามโจทย์	25
โปรแกรมทำงานได้จริง	25
อธิบายหลักการทำงานได้	15
ส่งงานครบถ้วน มีภาพวงจร โค้ด และผลลัพธ์	15
รวม	**100 คะแนน**

สามารถใช้ชุดนี้เป็น **ใบงาน Lab บทที่ 2** ได้เลย โดยข้อ 1–4 เป็นพื้นฐาน, ข้อ 5–8 เป็นการเขียนโปรแกรมควบคุม และข้อ 9–10 เป็นการประยุกต์กับงาน IoT เบื้องต้นครับ.