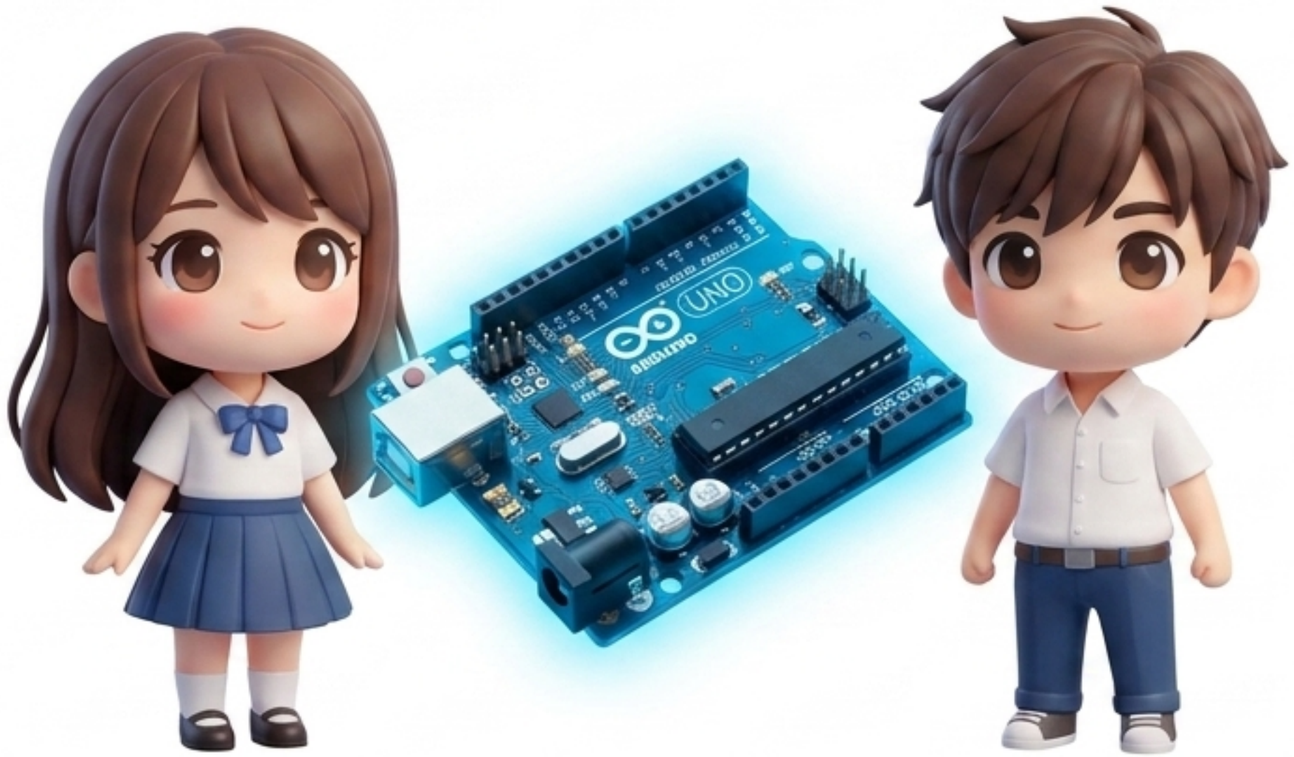
ก้าวแรกสู่นักประดิษฐ์ดิจิทัล: ควบคุมแสงสว่างด้วย Arduino



โครงการทดลองที่ 2: วิทยาลัย

ภารกิจวันนี้: ปลดล็อกทักษะ Maker (Maker Skill Tree)

1. อธิบายหลักการ:
เข้าใจทฤษฎีการเชื่อมต่อ LED แต่ละประเภท

2. ออกแบบ & จอดจำลอง:
สร้างวงจรเสมือนจริงด้วย Proteus 7

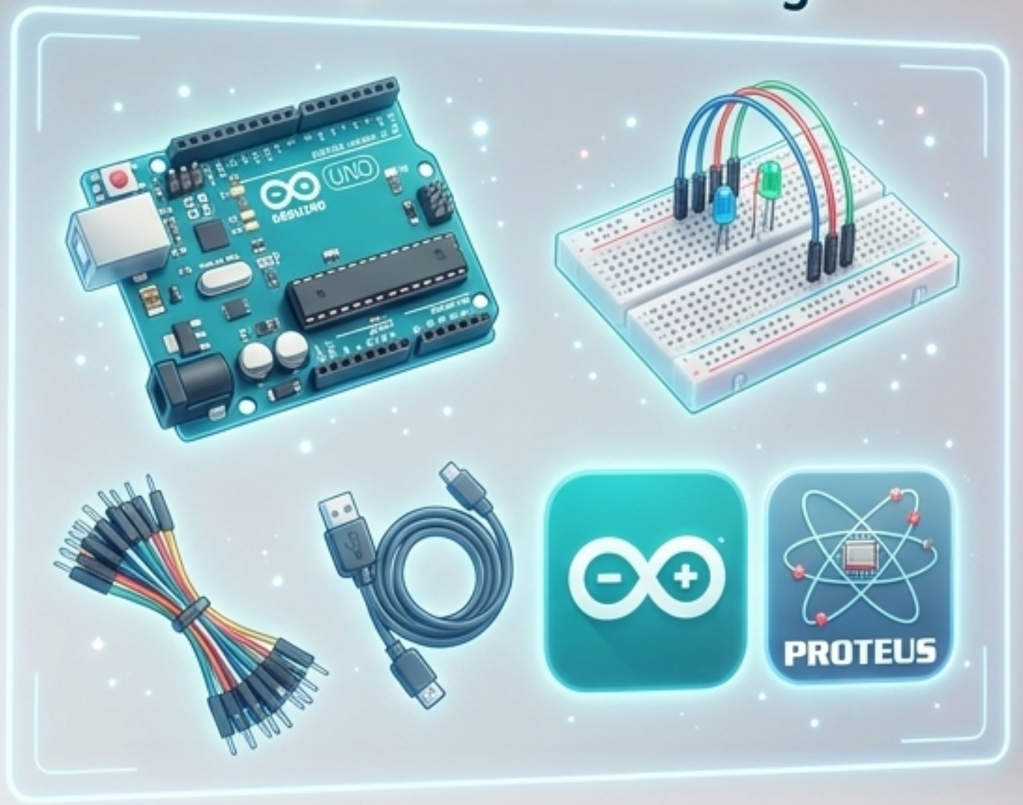
3. ประกอบวงจรจริง:
ต่อสายไฟบนแผงวงจร (Breadboard) ได้อย่างปลอดภัย

4. เขียนโปรแกรม:
ควบคุมด้วย Arduino IDE

5. ทดสอบ & บำรุงรักษา:
แก้ไขปัญหาและดูแลอุปกรณ์

เตรียมคลังอุปกรณ์ และกฎความปลอดภัยในห้องแล็บ

คลังอุปกรณ์ - Inventory



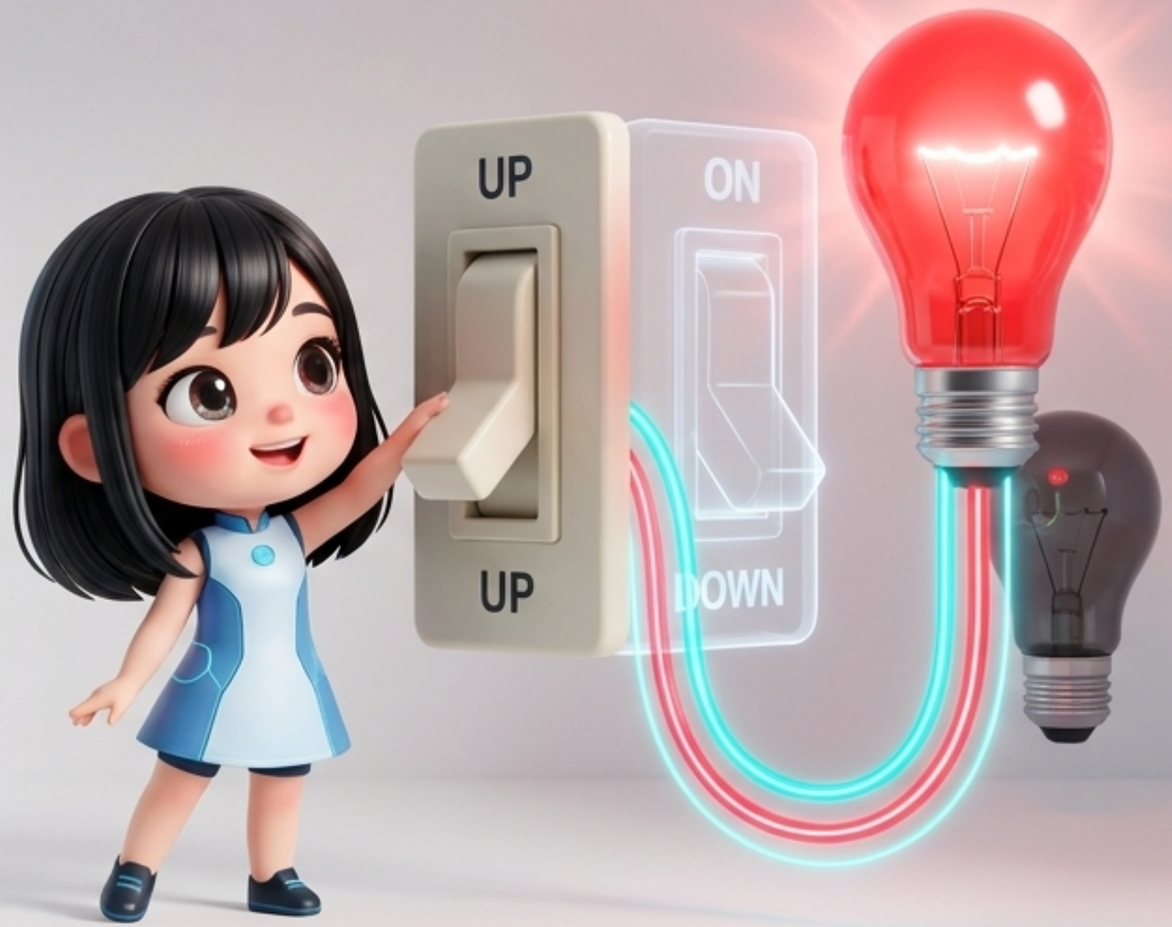
กฎเหล็ก 5 ข้อ - Safety Protocols

1. ห้ามเล่นหรือหยอกล้อกันขณะปฏิบัติงาน
2. ระวังการวางบอร์ดบนโต๊ะโลหะ (ป้องกันไฟฟ้าลัดวงจร)
3. ควรถอดสายวงจรออกให้หมดเมื่อเลิกใช้งาน
4. ห้ามถอด/เสียบสาย USB ตลอดเวลาโดยไม่จำเป็น
5. ปฏิบัติตามขั้นตอนอย่างเคร่งครัดเพื่อป้องกันอุปกรณ์เสียหาย

Level 1: การควบคุมลอจิกพื้นฐาน (Discrete Control)

Concept:
ควบคุม LED
4 ดวงแบบอิสระ
ด้วยคำสั่ง Digital

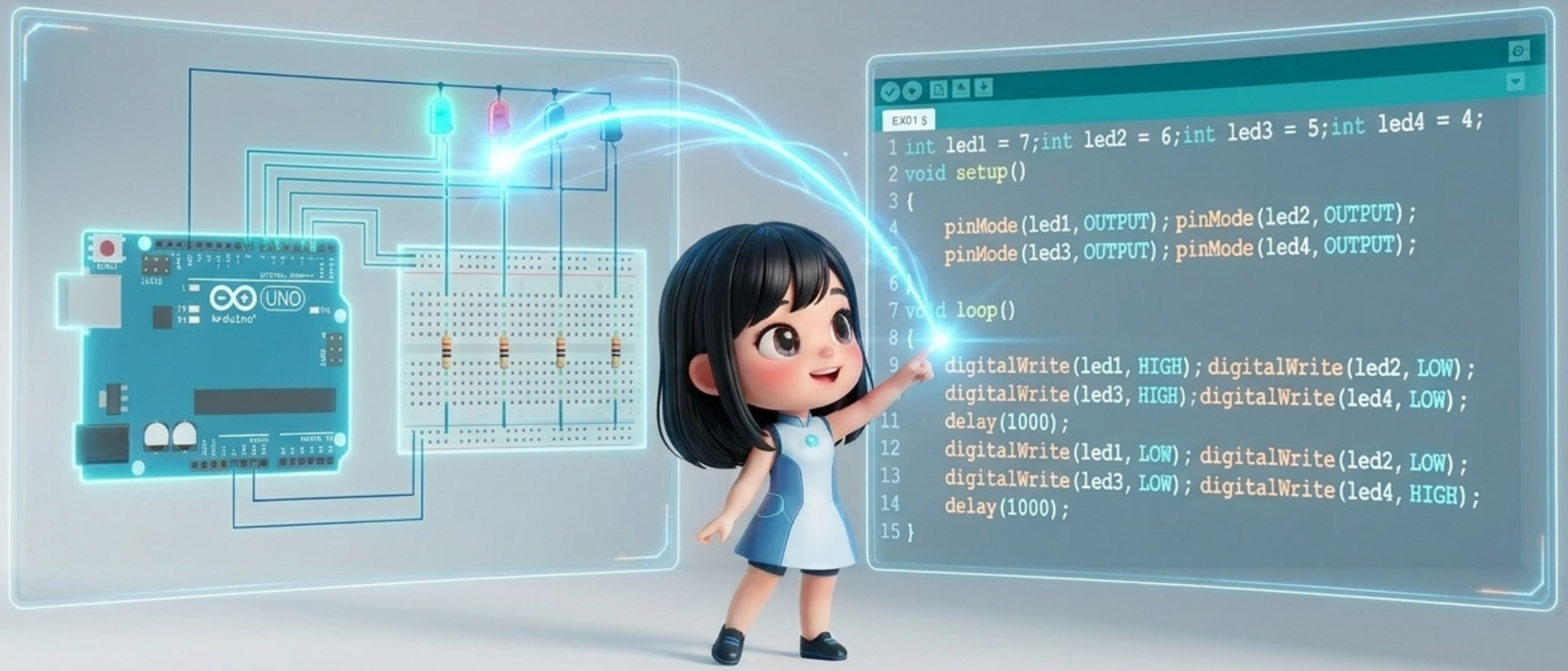
State 0:
LOW = จ่ายไฟ 0V
(ปิดไฟ)



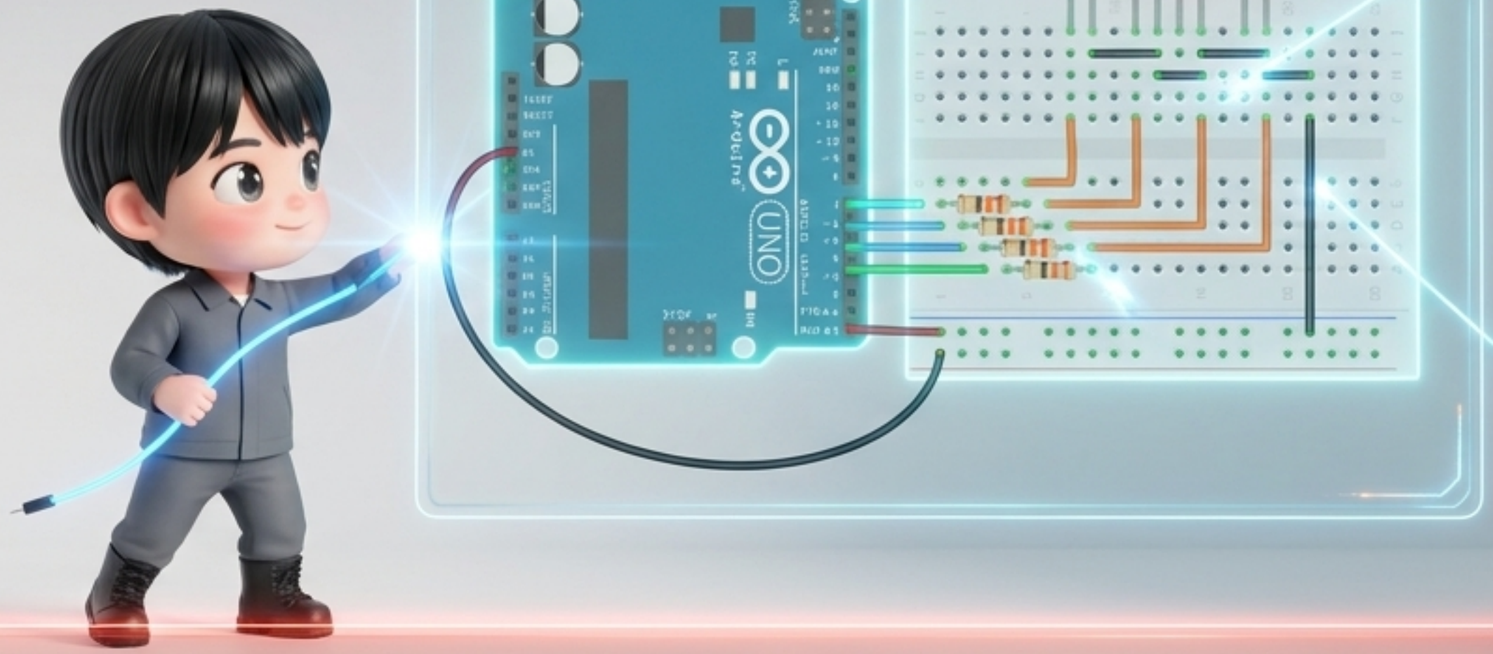
State 1:
HIGH = จ่ายไฟ 5V
(เปิดไฟ)

เป้าหมาย:
สั่งให้ LED 4 ดวงติด
และดับสลับกันตาม
จังหวะเวลา (Delay)

แปลงพิมพ์เขียวเป็นโค้ด: การเขียนโปรแกรมสั่งงาน



ลงมือประกอบวงจรจริง (The Physical Build)



การต่อตัวต้านทาน 220 โอห์ม (220R) เพื่อป้องกัน LED ขาด

การใช้ขาสัญญาณ D4 ถึง D7

Checkpoint: ตรวจสอบการต่อสายให้ตรงกับ Proteus ก่อนอัปโหลดไฟล์ LAB2_1.ino

Level 2: ผสมสีสันด้วยสัญญาณแอนะล็อก (The Color Spectrum)

Concept: LED RGB 1 ตัว สามารถสร้างสีได้นับล้านสี

Technology: ใช้เทคนิค PWM (Pulse Width Modulation) ขาที่มีสัญลักษณ์ ~ (เช่น ~9, ~10, ~11)

Function: คำสั่ง analogWrite(pin, value);

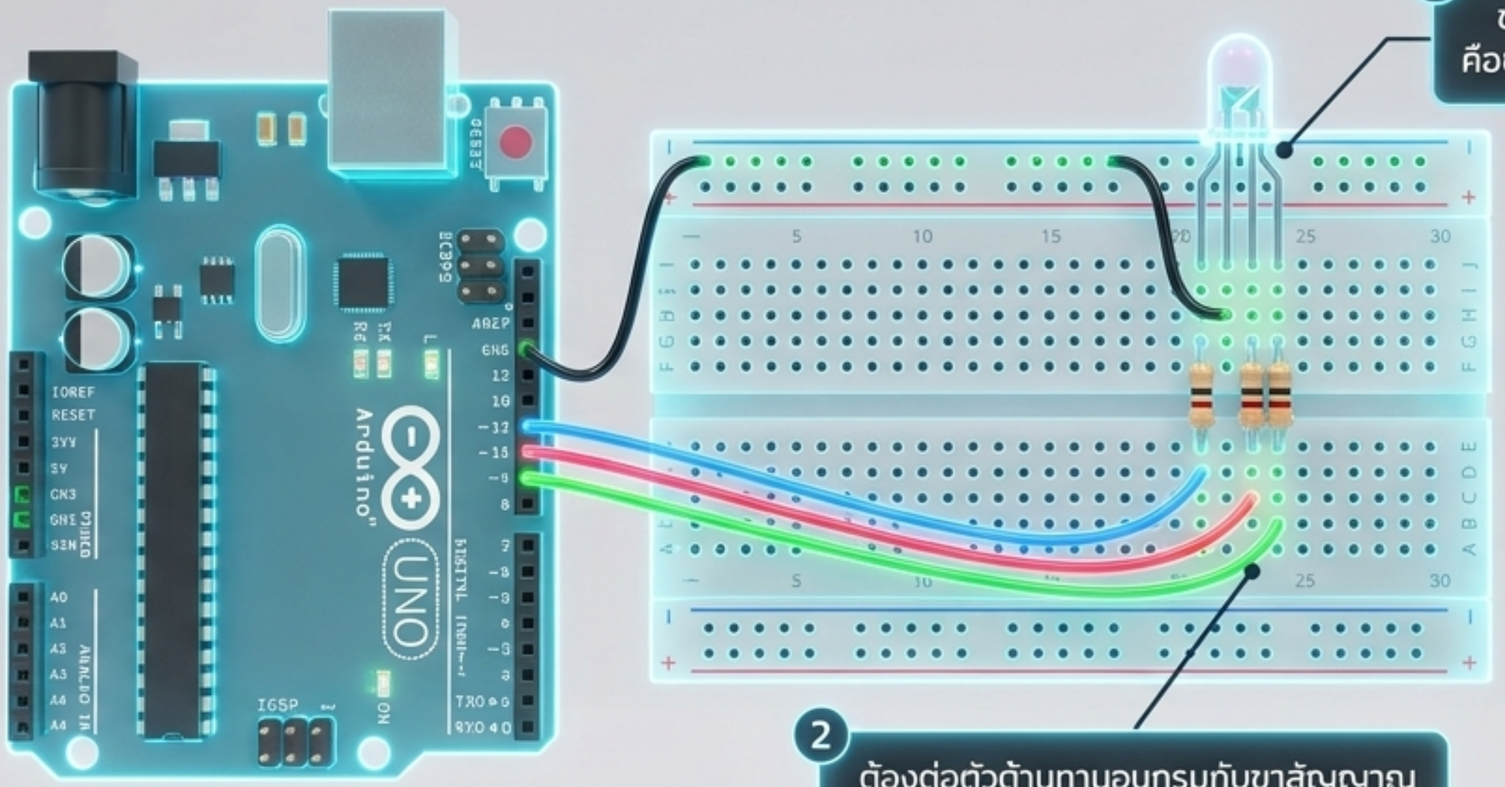
Value Range: สามารถปรับค่าความสว่างได้ตั้งแต่ 0 (ดับสนิท) ถึง 255 (สว่างสุด) ของแต่ละแม่สี (Red, Green, Blue)

การเขียนโค้ดผสมสี (Analog Blending Code)

```
1  int redPin = 11;
2  int greenPin = 10;
3  int bluePin = 9;
4  void setup()
5  {
6      pinMode(redPin, OUTPUT);
7      pinMode(greenPin, OUTPUT);
8      pinMode(bluePin, OUTPUT);
9  }
```

```
10 void loop()
11 {
12     analogWrite(redPin, 255);
13     analogWrite(greenPin, 0);
14     analogWrite(bluePin, 0);
15     delay(1000);
16     analogWrite(redPin, 0);
17     analogWrite(greenPin, 255);
18     analogWrite(bluePin, 0);
19     delay(1000);
20     analogWrite(redPin, 0);
       analogWrite(greenPin, 0);
```

การเชื่อมต่อวงจร LED RGB

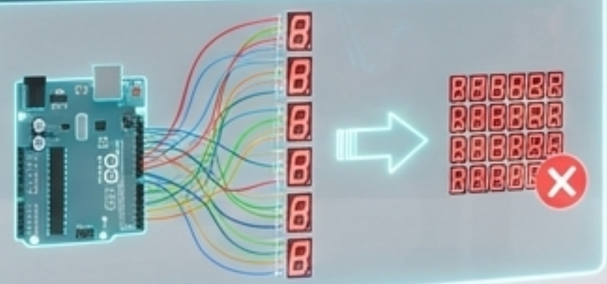


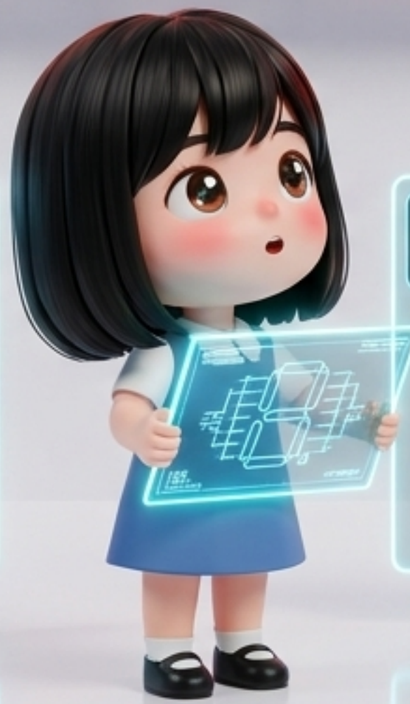
1 ขาที่ยาวที่สุดของ LED RGB
คือขา Common Ground (GND)

2 ต้องต่อตัวต้านทานอนุกรมกับขาสัญญาณ
R (Pin 11), G (Pin 10), B (Pin 9) เสมอ

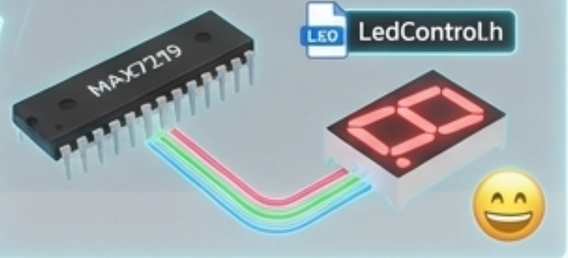
Level 3: การสื่อสารข้อมูลเพื่อแสดงตัวเลข (7-Segment)

The Challenge: การแสดงตัวเลขต้องใช้การควบคุม LED หลายดวงพร้อมกัน (Multiplexing)

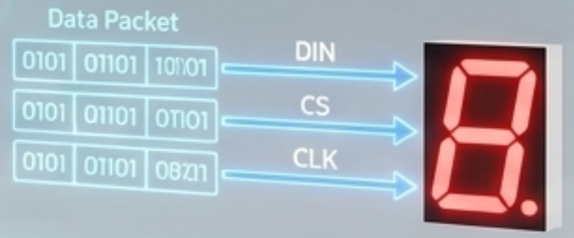




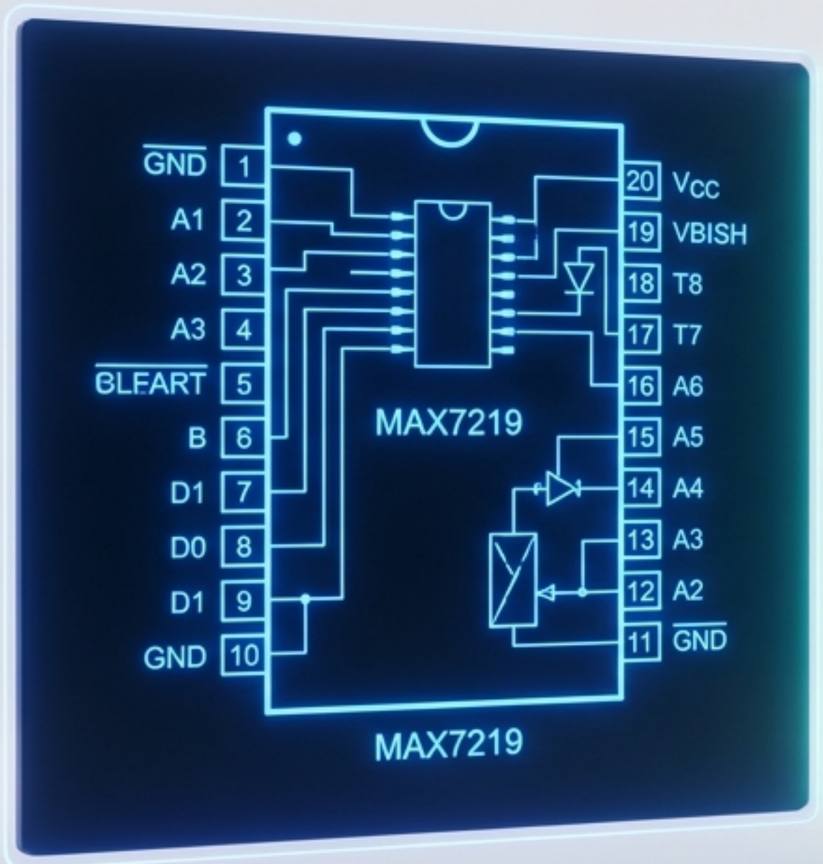
The Solution: แทนที่จะเขียนโค้ดเปิด/ปิดทีละเส้น เราจะใช้ชิป MAX7219 และ Library LedControl.h



Concept: เปลี่ยนจากการสั่งงาน ฮาร์ดแวร์โดยตรง เป็นการส่งชุดข้อมูล (Data Packet) ผ่านสายเพียง 3 เส้น (DIN, CS, CLK)



สั่งงานด้วย Library LedControl.h




```
1 #include "LedControl.h"
2 LedControl lc=LedControl(5,7,6,1);
3 int num = 0,seg1,seg2;
4 void setup()
5 {
6   lc.shutdown(0,false);
7   lc.setIntensity(0,5);
8   lc.clearDisplay(0);
9 }
10 void loop()
11 {
12   segi=num%10;
13   seg2=num/10;
14   lc.setDigit(0, 0, segi, false);
15   lc.setDigit(0, 1, seg2, false);
16     delay(1000);
17    num=num+1;
18     if (num > 99)
19     {
20       num = 0;
21     }
22 }
```

lc.shutdown(0,false);

→ ปลุกจอแสดงผลให้ทำงาน

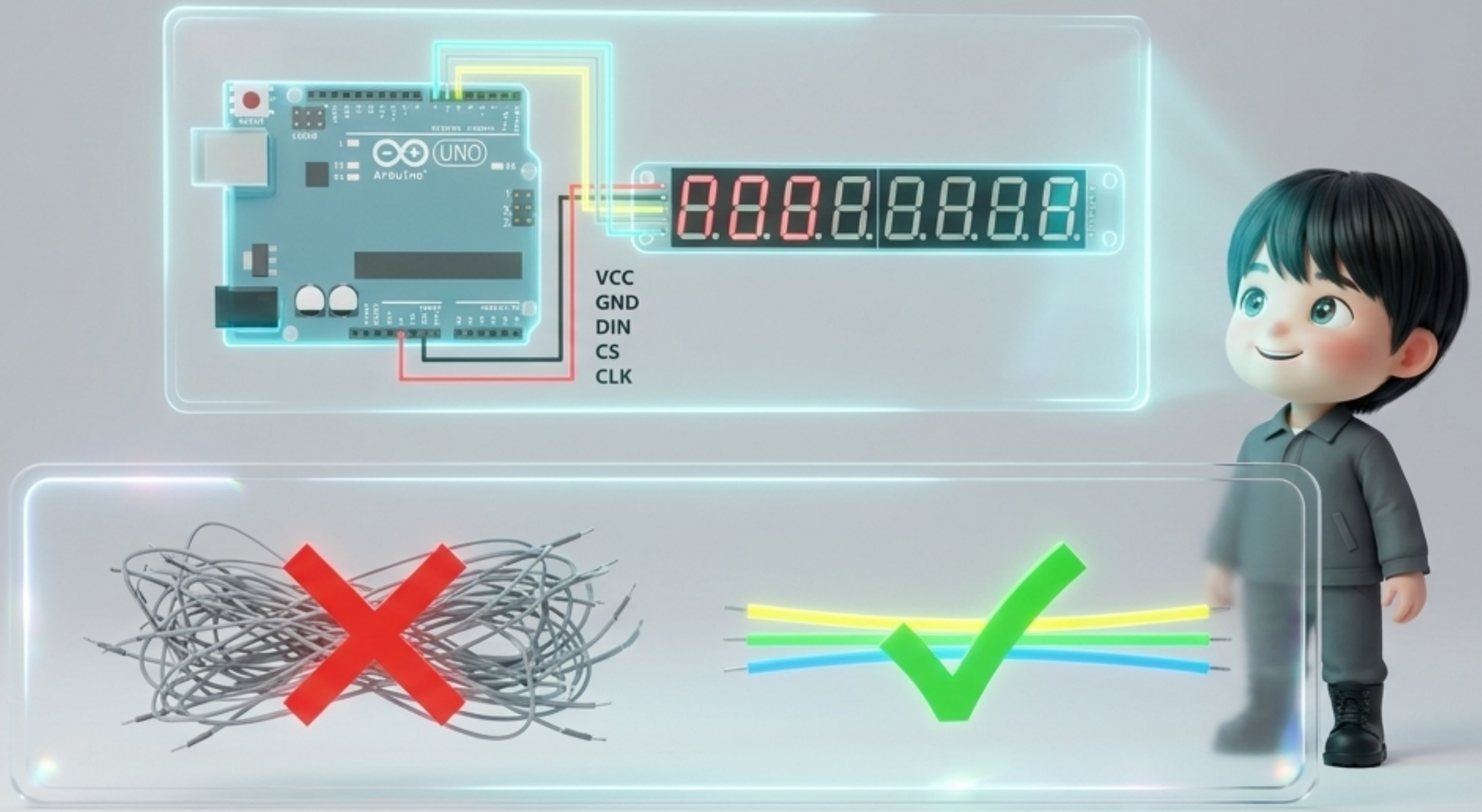
lc.setIntensity(0,5);

→ ตั้งค่าความสว่าง (0-15)

lc.setDigit(0, 0, seg1, false);

→ ส่งตัวเลข seg1 ไปแสดงที่หลักที่ 0

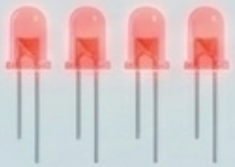
วงจรขั้นสูง: การเชื่อมต่อแบบ Serial



เปรียบเทียบความซับซ้อน: หากไม่ใช้โมดูล MAX7219 จะต้องใช้สายไฟนับสิบเส้น แต่ด้วยการสื่อสารแบบ Serial เราใช้สายสัญญาณเพียง 3 เส้น (เชื่อมต่อกับ Pin 11, 12, 13)

บทสรุป: วิวัฒนาการของการควบคุมเอาท์พุต (Output Evolution)

4 LEDs



เทคนิค: Discrete
คำสั่งหลัก: digitalWrite()
ผลลัพธ์: ติด/ดับ (1 บิต)

RGB LED

เทคนิค: Analog/PWM
คำสั่งหลัก: analogWrite()
ผลลัพธ์: ผสมสี (8 บิต)

7-Segment

เทคนิค: Serial Data
คำสั่งหลัก: LedControl
ผลลัพธ์: ข้อมูลตัวเลข

Takeaway: การเข้าใจทั้ง 3 เทคนิคนี้ คือรากฐานของการสร้างระบบดิจิทัลทั้งหมด

ภารกิจสำเร็จ! คุณคือ Digital Maker

- ตรวจสอบความเรียบร้อย ถอดสายวงจร และจัดเก็บอุปกรณ์
- สรุปผลการทดลองในใบงาน และเตรียมพร้อมสำหรับการสร้างสรรค์สิ่งประดิษฐ์ในระดับต่อไป!