ใบงานที่ 2

ขาอินพุต/เอาต์พุต GPIO

จุดประสงค์การเรียนรู้

1.ศึกษาการเปิด/ปิดหลอดไฟ LED

2.ศึกษาการเปิด/ปิดหลอดไฟ LED ด้วยปุ่มสวิตช์

เครื่องมือและอุปกรณ์การทดลอง

1.เครื่องไมโครคอมพิวเตอร์

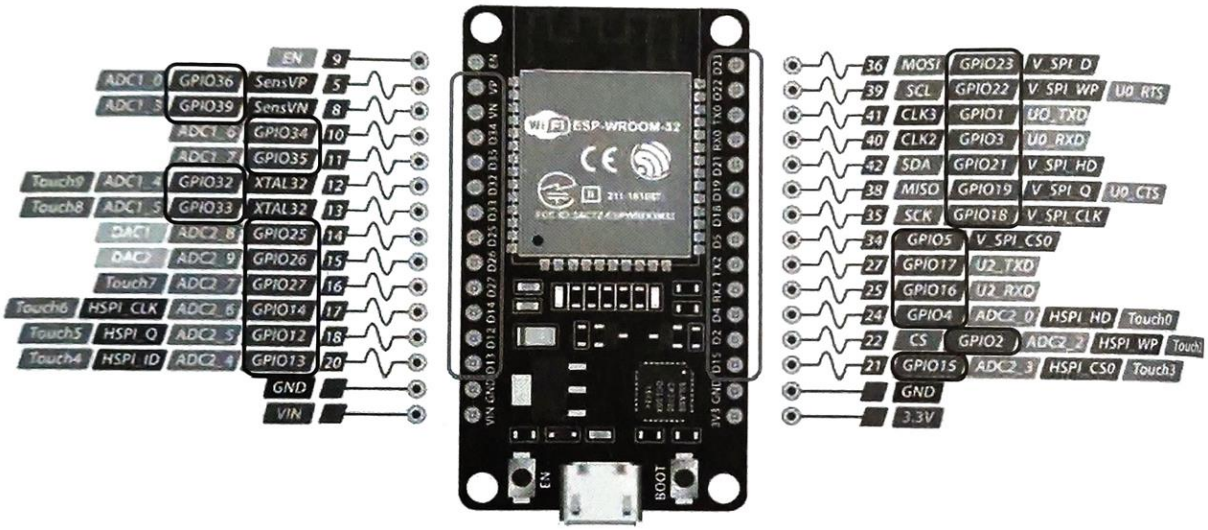
2.บอรด์ NodeMCU ESP32

3.โปรแกรมการทดลอง

4.อุปกรณ์อิเล็กทอรนิกส์สำหรับทดลอง

ขาอินพุต/เอาต์พุตอเนกประสงค์ GPIO

GPIO หรือ General Purpose Input Output เป็นขาที่เอาไว้เชื่อมต่อกับอุปกรณ์อินพุต/เอาต์พุต (I/O) ต่างๆ มีคุณสมบัติพิเศษคือ สามารถเขียนโค้ดคำสั่งเพื่อกำหนดให้ขาที่ต้องการ มีสถานะการทำงานเป็นอินพุต หรือเอาต์พุตก็ได้ เช่น เมื่อต่อเซ็นเซอร์เข้ากับขา 5 ก็แค่เขียนโค้ดสั่งให้ขา 5 มีสถานะเป็นอินพุตเพื่อรับค่าจาก เซ็นเซอร์เข้ามา หรือที่ตำแหน่งขาเดียวกันจะเปลี่ยนมาใช้หลอดไฟ LED เพื่อแสดงผล ก็สามารถเขียนโค้ดสั่งให้ขา มีสถานะเป็นเอาต์พุตได้ ด้วยคุณสมบัติพิเศษนี้จึงเป็นที่มาของชื่อเรียก ขาอินพุต/เอาต์พุตอเนกประสงค์ นั่นเอง



สำหรับ NodeMCU ESP32 หรือ ESP-WROOM-32 บนบอร์ดจะมีขา GPIO อยู่ทั้งสิ้น 25 ขา ที่ซึ่งในแต่ละขาจะถูกใช้เป็นช่องทางสื่อสารในรูปแบบต่างๆ ให้กับอุปกรณ์อื่นๆ ร่วมด้วย ดังรูป ในส่วนของการอ้างอิงตำแหน่งขาเวลาเขียนโค้ดโปรแกรม จะใช้วิธีระบุเป็นตัวเลขต่อท้ายของตำแหน่งขา GPIO โดยดูจากผังไดอะแกรม Pinout ของบอร์ด เช่น 0, 1, 2, 3,.. ยกตัวอย่าง pinMode(5, INPUT) ซึ่งจากรูปจะเห็นว่าที่ขา D5 ทั้ง 5 และ GPIO5 ก็คือตำแหน่งขาเดียวกันนั่นเอง (แตกต่างจากบอร์ด NodeMCU ESP8266 ที่สามารถจะระบุเป็น ตำแหน่งขา Digital ที่พิมพ์กำกับไว้ที่ขาบนบอร์ด เช่น D1, D2, D3, ... ได้ด้วย แต่ตัวเลขที่ระบุจะไม่ตรงกันกับ ตัวเลขต่อท้ายของตำแหน่งขา GPIO)

NOTE

ขา GPIO ของบอร์ด ESP-WROOM-32 จะมีจำนวนทั้งสิ้น 25 ขา แบ่งออกเป็น

-Input Only Pins หรือขาที่จะยอมให้ถูกกำหนดสถานะเป็นอินพุตได้เท่านั้นอยู่ 4 ขา คือ GPIO34,GPIO35, GPIO36 และ GPIO39

-Pins with internal Pull-up หรือขาที่มีการต่อ Pull-up Resistor ไว้ภายใน เวลาจะเรียกใช้ก็เพียงแค่กำหนดสถานะของขา INPUT ให้เป็น INPUT_PULLUP มีอยู่ 8 ขา คือ GPIO14, GPIO16, GPIO17, GPIO18, GPIO19, GPIO21, GPIO22 และ GPIO23

-Pins without internal Pull-up หรือขาที่ไม่ได้ต่อ Pull-up Resistor ไว้ภายใน เวลาจะทำ Pull-up ให้กับขา จึงต้องต่อ Pull-up Resistor ไว้ภายนอก ดังตัวอย่างในเนื้อหาของการต่อวงจร แบบ Pull-up ที่ได้อธิบายไว้ในบทที่ 4 มีอยู่ 6 ขา คือ GPIO13, GPIO25, GPIO26, GPIO27,GPIO32 และ GPIO33

การทดลองที่ 1 การเปิด/ปิดหลอดไฟ LED (Digital Output)

อุปกรณ์ในการทดลอง

1.บอรด์ NodeMCU ESP32

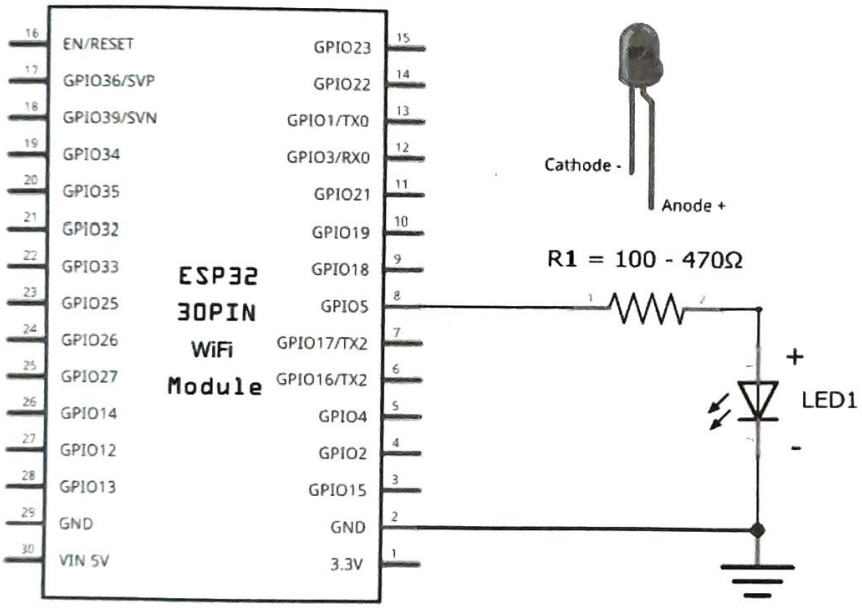
2.หลอดไฟ LED

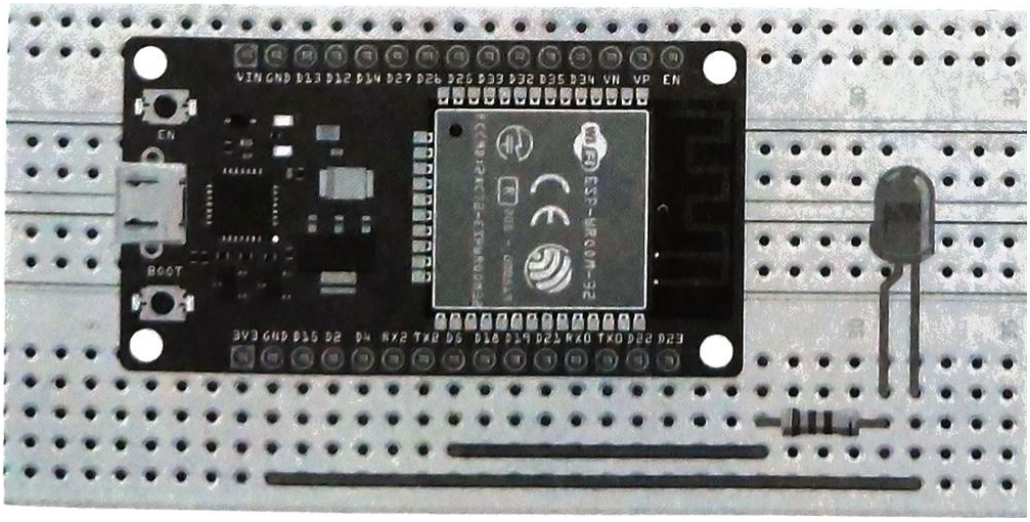
3.ตัวต้านทาน (Resistor)

4.แผงต่อวงจร

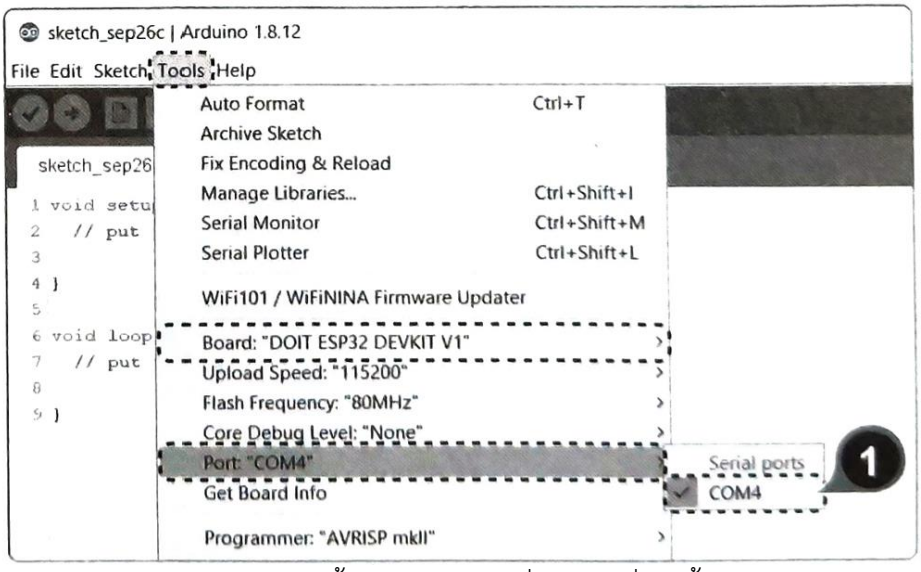
5.สายไฟต่อวงจร

ประกอบวงจรตามรูป





1. หลังจากเชื่อมต่อบอร์ดเข้ากับ คอมพิวเตอร์ เปิด Arduino IDE แล้วไปที่เมนู Tools เลือก บอร์ด (Board) และ พอร์ต (Port) ให้ตรงกับที่ใช้งาน



2. จากนั้นเปิดตัวอย่างโค้ดโปรแกรม Blink ขึ้นมา แล้วแก้ไขเพิ่มเติมคำสั่งดังนี้ (การอ้างอิงตำแหน่งขาจะใช้วิธีกำหนดตัวเลขขา GPIO)

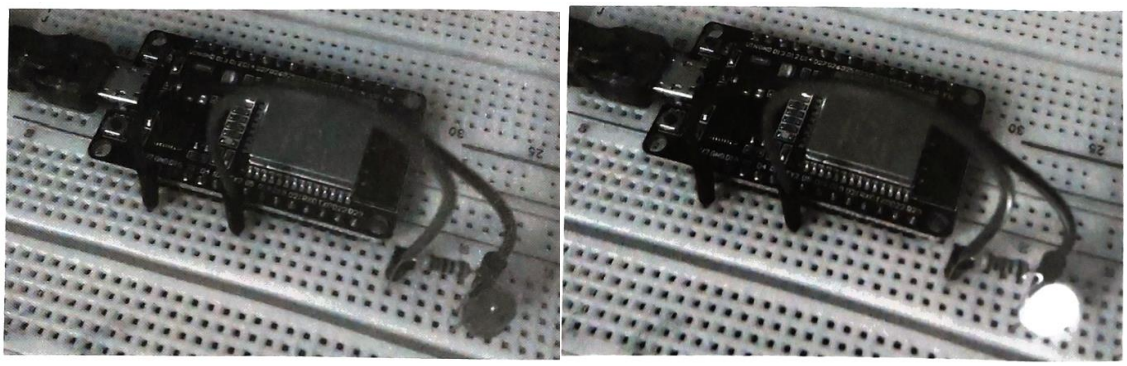
```
void setup() {
  pinMode(5, OUTPUT);          //กำหนดให้ขา D5/GPIO5 เป็น Output
}

void loop() {
  digitalWrite(5, HIGH);       //กำหนดให้ Output ที่ขา D5/GPIO5 เป็น HIGH หลอดไฟติด
  delay(1000);                 //หน่วงรอเป็นเวลา 1 วินาที
  digitalWrite(5, LOW);        //กำหนดให้ Output ที่ขา D5/GPIO5 เป็น LOW หลอดไฟดับ
  delay(1000);                 //หน่วงรอเป็นเวลา 1 วินาที
}
```

2

3. คลิกปุ่ม โปรแกรม Upload เพื่ออัพโหลด

4. ผลลัพธ์ที่ได้คือ จะเห็นหลอดไฟ LED ติด..ดับ...ติด..ดับ สลับกันไปเรื่อยๆ จนกว่าจะหยุดจ่ายไฟ



บันทึกผลการทดลอง

.....

.....

.....

.....

การทดลองที่ 2 การเปิด/ปิดหลอดไฟ LED ด้วยปุ่มสวิตช์ (Digital Input)

อุปกรณ์ในการทดลอง

1.บอรด์ NodeMCU ESP32

2.หลอดไฟ LED

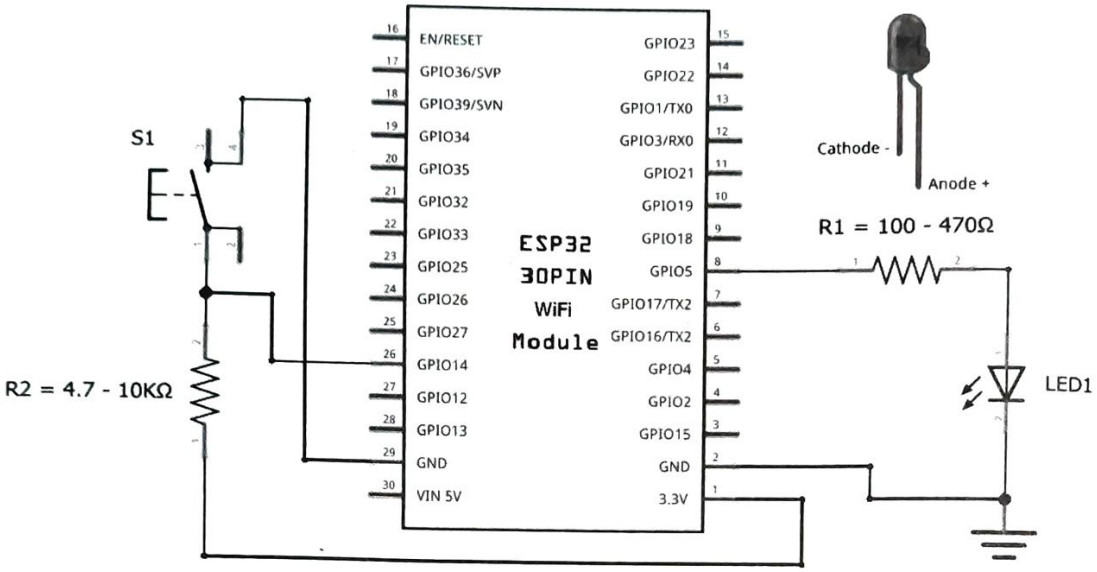
3.ตัวต้านทาน (Resistor)

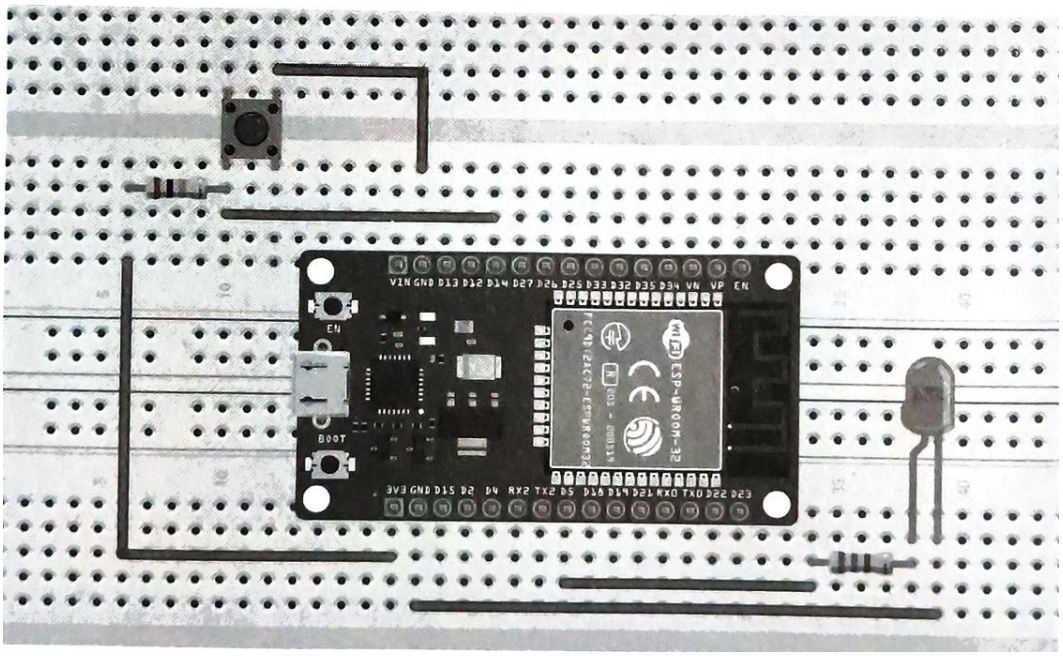
4.สวิตช์กดติดปล่อยดับ

5.แผงต่อวงจร

6.สายไฟต่อวงจร

ประกอบวงจรตามรูป





1. ให้นำโค้ดโปรแกรมจาก การทดลองที่ 1 มาแก้ไขเพิ่มเติมคำสั่งดังนี้

2. คลิกปุ่ม Upload เพื่ออัพโหลดโปรแกรม

```
void setup() {
  pinMode(14, INPUT);              //กำหนดให้ขา GPIO14 หรือ D14 เป็น Input
  pinMode(5, OUTPUT);              //กำหนดให้ขา GPIO5 หรือ D5 เป็น Output
}

void loop() {
  int buttonStatus = digitalRead(14);  //อ่านค่าสถานะสวิทช์ที่เป็นดิจิตอล 0 หรือ 1 จากขา Input
  if (buttonStatus == 0) {         //ตรวจสอบสถานะสวิทช์ ถ้าเป็น 0 หรือ LOW (สวิทช์ถูกกด)
    digitalWrite(5, HIGH);         //กำหนดให้ Output ที่ขา GPIO5 เป็น HIGH หลอดไฟติด
  } else {                         //แต่ถ้าไม่ หรือนอกเหนือจากเงื่อนไขที่กล่าวมา
  digitalWrite(5, LOW);            //กำหนดให้ Output ที่ขา GPIO5 เป็น LOW หลอดไฟดับ
  }
  delay(100);                      //หน่วงรอเป็นเวลา 0.1 วินาที ในแต่ละรอบคำสั่ง
                                   เพื่อให้ตอบสนองต่อการกดได้รวดเร็ว
}
```

3. ผลลัพธ์ที่ได้คือ เมื่อกดปุ่มสวิทช์ หลอดไฟ LED จะติด และถ้าปล่อย หลอดไฟก็จะดับ

บันทึกผลการทดลอง

.....

.....

.....

.....

สรุปผลการทดลอง

.....

.....

.....

.....