การอ่านค่าจากขาอนาล็อก

ในหัวข้อนี้ คุณจะได้เรียนรู้วิธีการอ่านค่าจากขาอนาล็อกบนบอร์ด ESP32 ซึ่งมีประโยชน์สำหรับการอ่านค่าจากอุปกรณ์อย่างตัวต้านทานปรับค่าได้ (Potentiometer) หรือเซนเซอร์อนาล็อกอื่น ๆ

การเรียนรู้เรื่องนี้จะช่วยให้คุณเข้าใจการใช้งานบอร์ด ESP32 มากยิ่งขึ้น และสามารถนำไปประยุกต์ใช้ในโครงการต่าง ๆ ได้ง่ายขึ้น

การอ่านค่าจากขาอนาล็อก

การอ่านค่าจากขาอนาล็อกบนบอร์ด ESP32 หมายถึงการวัดระดับแรงดันไฟฟ้าที่เปลี่ยนแปลงได้ในช่วง **0V** ถึง **3.3V** โดยแรงดันที่วัดได้จะถูกแปลงเป็นค่าตัวเลขระหว่าง **0** ถึง **4095**

- แรงดัน **0V** จะให้ค่าเป็น **0**
- แรงดัน **3.3V** จะให้ค่าเป็น **4095**
- หากแรงดันอยู่ระหว่าง **0V** และ **3.3V** ค่าเลขที่ได้ก็จะอยู่ในช่วงนั้นเช่นกัน

Deva DIY

ADC ของ ESP32 มีพฤติกรรมที่ไม่เป็นเชิงเส้น (Non-linear)

ในทางทฤษฎี เราคาดหวังว่าขา ADC (Analog-to-Digital Converter) ของ ESP32 จะทำงานอย่างเป็นเชิงเส้น ซึ่งหมายถึงค่าที่ได้จากการแปลงแรงดันไฟฟ้าควรเพิ่มขึ้นตามแรงดันไฟฟ้าที่เพิ่มขึ้นในอัตราส่วนคงที่

แต่ในความเป็นจริง ADC ของ ESP32 ไม่ได้ทำงานในลักษณะนั้น ค่าที่ได้จะมีพฤติกรรมที่ไม่เป็นเชิงเส้น ซึ่งหมายความว่า ค่าที่อ่านได้อาจไม่ตรงตามที่คาดไว้เมื่อแรงดันไฟฟ้าเพิ่มขึ้น

ฟังก์ชัน **analogRead()**

การอ่านค่าจากขาอนาล็อกบน ESP32 ด้วย Arduino IDE สามารถทำได้ง่าย ๆ โดยใช้ฟังก์ชัน **analogRead()** ฟังก์ชันนี้จะรับค่า **GPIO** ที่คุณต้องการอ่านเป็นอาร์กิวเมนต์ ดังตัวอย่างด้านล่าง:

```
analogRead(GPIO);
```

หมายเหตุ:
ขา **ADC2** บน ESP32 ไม่สามารถใช้งานได้ในขณะที่เปิดใช้งาน **Wi-Fi** หากคุณใช้ Wi-Fi และพบปัญหาในการอ่านค่าจากขา **GPIO** ที่เชื่อมต่อกับ **ADC2**

คำแนะนำ:
ลองเปลี่ยนไปใช้ขา **GPIO** ที่เชื่อมต่อกับ **ADC1** แทน วิธีนี้จะช่วยแก้ปัญหาได้

ตัวอย่างโครงการ

มาลองทำตัวอย่างง่าย ๆ เพื่ออ่านค่าจากตัวต้านทานปรับค่าได้ (Potentiometer) และดูวิธีการทำงานกันนะครับ

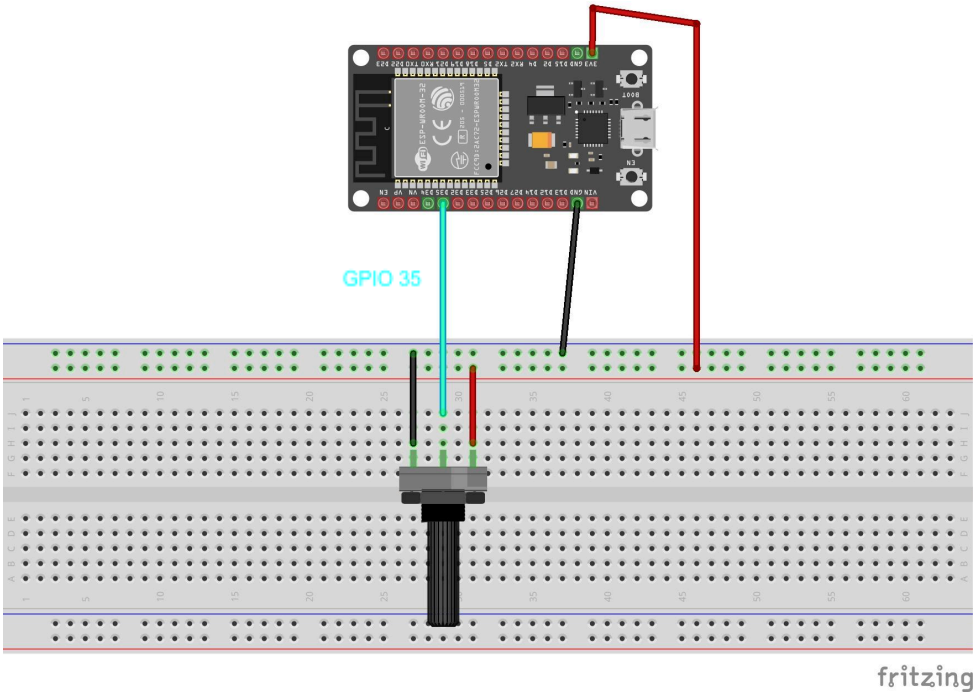
การต่อวงจร
รายการอุปกรณ์ที่คุณต้องใช้ในการต่อวงจรมีดังนี้:

- บอร์ด ESP32 DOIT DEVKIT V1
- ตัวต้านทานปรับค่า (Potentiometer)
- แผ่นทดลอง (Breadboard)
- สายจั้มเปอร์ (Jumper wires)

Deva DIY

เชื่อมต่อขาต่าง ๆ ของตัวต้านทานปรับค่าไปยังบอร์ด ESP32 ตามแผนผังที่แสดงด้านล่าง โดยขากลางของตัวต้านทานปรับค่าจะเชื่อมต่อกับ **GPIO 35**

การเลือกใช้ GPIO 35 จะทำให้คุณหลีกเลี่ยงปัญหาหากมีการใช้งาน Wi-Fi บนขา ADC2 (GPIO 34) ของ ESP32 และสามารถใช้งานได้ปกติ 🙂



โค้ดตัวอย่าง (ไฟล์ **Analog_Read**)

Deva DIY

```
// ตัวต้านทานปรับค่า (Potentiometer) เชื่อมต่อกับ GPIO 35 (Analog ADC1_CH6)
const int potPin = 35;  // กำหนดหมายเลขขา GPIO ที่เชื่อมต่อกับตัวต้านทานปรับค่า

// ตัวแปรสำหรับเก็บค่าที่อ่านจากตัวต้านทานปรับค่า
int potValue = 0;

void setup() {
  Serial.begin(115200);  // เริ่มต้นการสื่อสารผ่านพอร์ตอนุกรมที่ความเร็ว 115200 บิตต่อวินาที
  delay(1000);  // รอ 1 วินาที (1000 มิลลิวินาที) ก่อนเริ่มต้นการทำงาน
}

void loop() {
  // อ่านค่าจากตัวต้านทานปรับค่า
```

```
  potValue = analogRead(potPin);  // อ่านค่าอนาล็อกจากขา GPIO 35 (ADC1_CH6)

  // แสดงค่าที่อ่านได้ออกทางพอร์ตอนุกรม
  Serial.println(potValue);  // พิมพ์ค่าของตัวแปร potValue ลงใน Serial Monitor

  delay(500);  // รอ 500 มิลลิวินาทีก่อนทำงานรอบถัดไป
}
```

อธิบายการทำงานของโค้ด:

1. กำหนดขา **GPIO**
   โค้ดเริ่มต้นโดยการกำหนดให้ตัวต้านทานปรับค่า (Potentiometer) เชื่อมต่อกับ **GPIO 35** ของ ESP32 ซึ่งรองรับการอ่านค่าอนาล็อก (ADC1_CH6)

   ```
   const int potPin = 35;
   ```

2. การตั้งค่าในฟังก์ชัน **setup()**
   ในฟังก์ชัน `setup()` จะเริ่มต้นการสื่อสารผ่านพอร์ตอนุกรมด้วยความเร็ว 115200 บิตต่อวินาที (`Serial.begin(115200)`) และหน่วงเวลา 1 วินาที (`delay(1000)`) เพื่อให้ตัวระบบเริ่มทำงานก่อน

   ```
   Serial.begin(115200);
   ```

3. การอ่านค่าจากตัวต้านทานในฟังก์ชัน **loop()**
   ในฟังก์ชัน `loop()` โค้ดจะอ่านค่าอนาล็อกจาก **GPIO 35** (ที่เชื่อมต่อตัวต้านทานปรับค่า) ผ่านฟังก์ชัน `analogRead(potPin)` และเก็บค่าไว้ในตัวแปร `potValue`

   ```
   potValue = analogRead(potPin);
   ```

4. แสดงผลค่าที่อ่านได้
   ค่า `potValue` ที่อ่านจากตัวต้านทานจะถูกพิมพ์ออกทาง **Serial Monitor** โดย

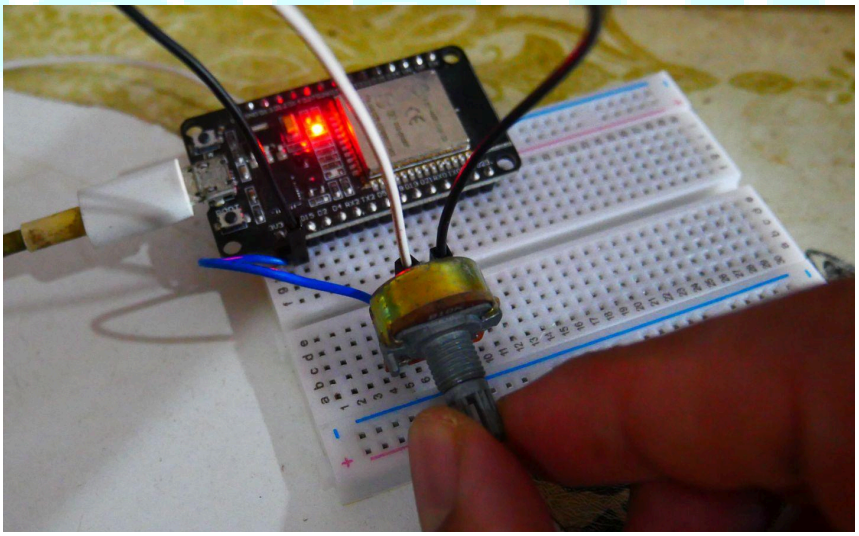
ใช้ `Serial.println(potValue)` ซึ่งจะแสดงค่าที่เก็บใน `potValue` ทุก ๆ 500 มิลลิวินาที

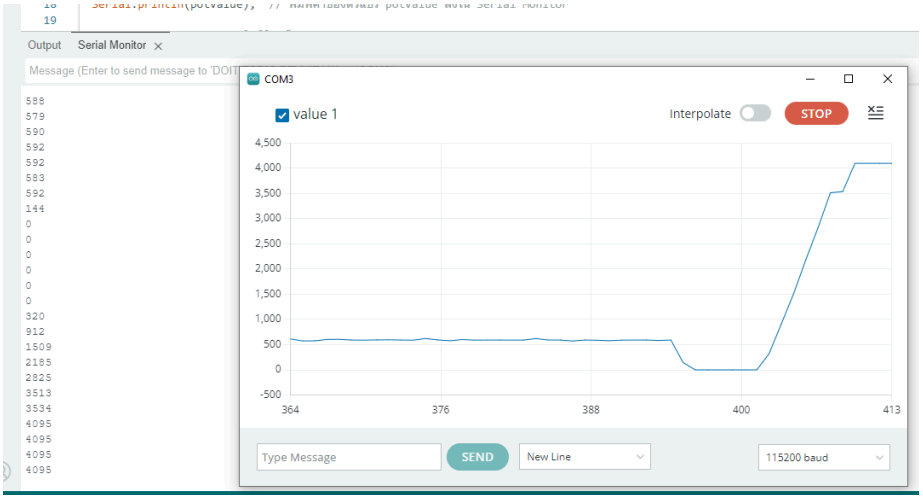
```
Serial.println(potValue);
delay(500);
```

ทดสอบตัวอย่าง

1. อัปโหลดโค้ดที่ให้ไปยังบอร์ด ESP32 ของคุณ
2. เปิด **Serial Monitor** ใน Arduino IDE โดยตั้งค่าความเร็วในการสื่อสารที่ **115200** บิตต่อวินาที
3. หมุนตัวต้านทานปรับค่า (Potentiometer) และสังเกตค่าที่แสดงใน **Serial Monitor** ว่าจะเปลี่ยนแปลงไปตามการหมุนของตัวต้านทาน

คุณจะเห็นค่าที่แสดงใน **Serial Monitor** เปลี่ยนไปในช่วง **0** ถึง **4095** ตามการหมุนของตัวต้านทาน ปรับค่าให้ค่าเพิ่มหรือลดตามระดับแรงดันไฟฟ้าที่วัดได้จากตัวต้านทานนั้น ๆ




Output
Serial Monitor
COM3
value 1
Interpolate
STOP
4,500
4,000
3,500
3,000
2,500
2,000
1,500
1,000
500
0
-500
364
376
388
400
413
Type Message
SEND
New Line
115200 baud

Deva DIY