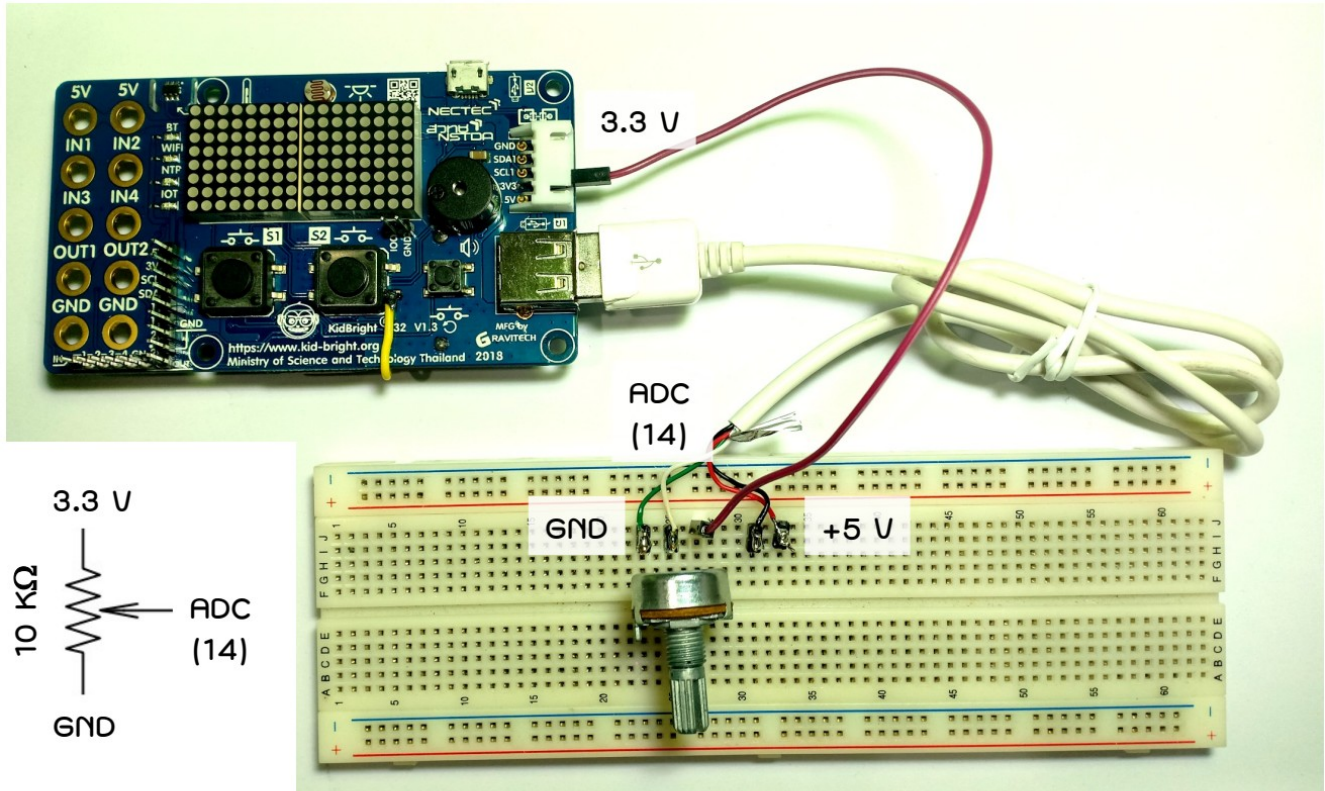


จากนั้นจึงหาสาย USB ทั่วไป ที่ไม่ใช้แล้ว มาตัดแล้วทำขั้วต่อ โดย สายสีแดงคือไฟ 5 โวลต์ สายสีขาวคือ D+ และสายสีเขียวคือ D- ซึ่งสาย D+ และ D- ต่อกับ ADC2_CH8 และกราวด์ (GND) อยู่ ตามลำดับ และเตรียมการทดสอบโดยใช้ตัวต้านทานปรับค่าได้ (10 KOhm) ดังรูป



หมายเหตุ ห้ามจ่าย +5 V ให้กับชิพ ESP32 เพราะบอร์ดทนแรงดันสูงสุดไม่เกิน 3.6 V

ในส่วนของซอฟต์แวร์ เพื่อควบคุมชิพ ESP-WROOM-32

เลือกใช้ Arduino IDE และทำการเพิ่มบอร์ด ESP32 (โดยอาจเลือกบอร์ด Node32S แทนสำหรับบอร์ดคิตบอร์ด) ทำการติดตั้งไลบรารีของ HT16K33 16*8 LED Controller (จาก <https://www.adafruit.com/product/1427>) และ Adafruit-GFX-library (หากยังไม่ได้ติดตั้งมาก่อน) เพื่อการควบคุมการแสดงผลผ่านจอแอลอีดีอาเรียบบนบอร์ด รายละเอียดของการเพิ่มบอร์ดอาจอ่านได้จาก <https://randomnerdtutorials.com/installing-the-esp32-board-in-arduino-ide-windows-instructions/> และ ข้อมูลเกี่ยวกับ HT16K33 16*8 LED Controller อยู่ใน <https://www.adafruit.com/product/1427>

เปิดตัวอย่างโค้ด AnalogReadSerial จากนั้นจึงเปลี่ยนขา A0 เป็น 14 และเพิ่มการหน่วงเวลา (delay) เป็น 100 มิลลิวินาที แล้วจึงบันทึกและอัปโหลดโปรแกรม

```

void loop() {
  // read the input on analog pin 0:
  int sensorValue = analogRead(14);
  // print out the value you read:
  Serial.println(sensorValue);
  delay(100);          // delay in between reads for stability
}

```

สังเกตผลบน Serial Monitor จะเห็นว่า ค่าที่แสดง (0-4095) เปลี่ยนตามการหมุนของตัวต้านทานปรับค่าได้

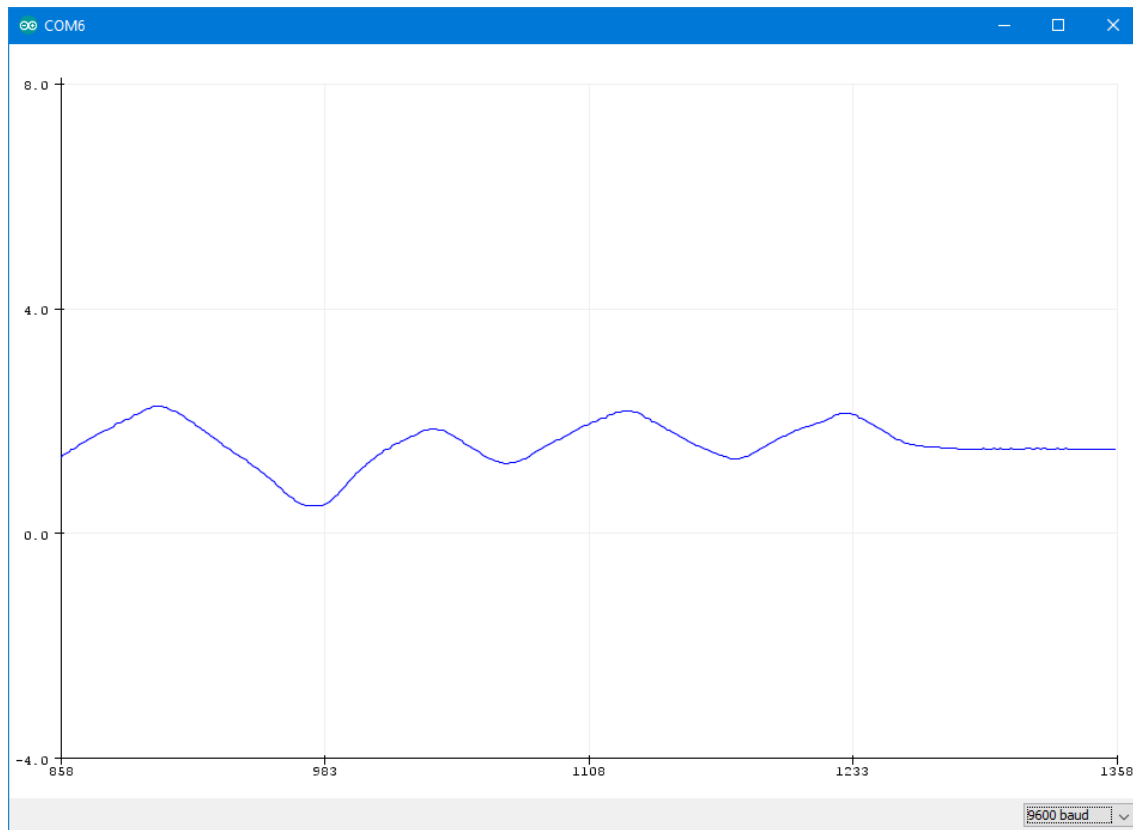
ทดสอบการใช้แอลอีดีอาเรย์ โดยศึกษาจาก ตัวอย่างในไลบรารีที่ดาวน์โหลดมา (เช่น matrix16x8.ino แต่ไม่ขอกล่าวรายละเอียดในที่นี้)

ทำการอัปเดตโค้ดด้านล่างนี้ ก็จะได้ โวลต์มิเตอร์ที่สร้างจากบอร์ดคิโดไบร์ท

```

1 #include <Wire.h>
2 #include <Adafruit_GFX.h>
3 #include "Adafruit_LEDBackpack.h"
4
5 Adafruit_8x16matrix matrix = Adafruit_8x16matrix();
6
7 void setup() {
8   Serial.begin(9600);
9   Serial.println("DVM on KB");
10  matrix.begin(0x70); // pass in the address
11  matrix.setRotation(2); // for KidBright
12 }
13
14 void loop() {
15   int readVal = analogRead(14); // 12-bit data (0 - 4095)
16   float voltVal = readVal*3.3/4095.0; // Convert to Volt
17   // e.g. 3.21: '3' is the first digit and '2' is the first decimal digit
18   int firstVal = floor(voltVal);
19   int secondVal = round((voltVal - (float) firstVal)*10.0);
20   // Serial.print(readVal);
21   // Serial.print("\t");
22   Serial.println(voltVal);
23   matrix.clear(); // clear display
24   matrix.setTextSize(1);
25   matrix.setTextWrap(false);
26   matrix.setTextColor(LED_ON);
27   matrix.setCursor(2,8); // First Number
28   matrix.print(firstVal);
29   matrix.drawPixel(0,6, LED_ON); // decimal point
30   matrix.setCursor(2,0); // Second Number
31   matrix.print(secondVal);
32   matrix.writeDisplay();
33   delay(500);
34 }

```



แนวทางการพัฒนาต่อยอด

ด้านซอฟต์แวร์

1. ปรับปรุงการแสดงผลตัวเลข ให้ได้ค่าที่ละเอียดขึ้น (ใช้อักษรวิ่ง)
2. ใช้ปุ่ม SW1 ในการเปลี่ยนโหมดการแสดงผล จากดิจิทัลเป็นแอนะล็อก
3. แสดงผลที่อ่านได้ผ่านทาง WiFi
4. สร้างพล็อตเตอร์ (คือแสดงผลเป็นกราฟที่เลื่อนไปเรื่อย ๆ)
5. สร้างออสซิลโลสโคป (มีการวัดความถี่และทริกเกอร์สัญญาณ)
6. สร้างเป็น datalogger

ด้านฮาร์ดแวร์

1. นำเซนเซอร์ที่ให้เอาต์พุตเป็นสัญญาณแอนะล็อก เช่น เซนเซอร์วัดแก๊ส เซนเซอร์วัดความชื้นดิน เซนเซอร์น้ำฝน มาต่อ
2. สร้างโพรบที่สามารถเพิ่มระดับในการวัด (เช่น 0 – 5 V หรือ 0 - 30 V) ด้วยการใช้วงจรแบ่งแรงดัน