2023/12/19 初版

2023/12/19 First edition

2023/12/19 作者李進衛

2023/12/19 Author Lee Chin wei

2023/12/19 使用Arduino UNO

2023/12/19 Using Arduino UNO

2022/12/20 新增加OLED 程式

2022/12/20 Newly added OLED program

2022/12/21 新增VR debounce 時間5ms 取樣一次改為20ms取樣一次

2022/12/21 Added VR debounce time 5ms sampling time changed to 20ms sampling time

每20ms讀取VR的電壓一次-->將讀取結果與上一次讀取結果比對如果一樣就不進行電壓換換與更新OLED的內容，如果不一樣

表示電壓值被改變就進行電壓和MIDI值換算，並且更新OLED的內容。

Read the VR voltage every 20ms --> Compare the reading result with the last reading result.

If they are the same, do not perform voltage replacement and update the OLED content.

If they are different,If the voltage value is changed,

the voltage and MIDI values ​​are converted, and the content of the OLED is updated.

2023/12/22 IQC 吳東陽要求電壓降為小數點兩位數，由於改為輸出CSV檔案格式因此保留小數點第四位

2023/12/22 IQC 要求數據丟到PC 讓他不用抄數據

2023/12/23 新增加按鍵，請邱盈繼幫忙焊接按鍵

2023/12/13 修正判斷式與運算(ADC值\*電壓)/(1023)

治具整理:鄭民杰

2023/12/22 IQC Wu Dongyang requires that the voltage drop be to two decimal places.

 Since the output is changed to CSV file format, the fourth decimal place is retained.

2023/12/22 IQC requires the data to be thrown to the PC so that he does not have to copy the data

2023/12/23 New buttons are added, please ask Qiu Yingji to help solder the buttons

2023/12/13 Modified judgment formula and operation (ADC value \* voltage)/(1023)

Fixture arrangement: Zheng Minjie

2023/12/24 增加abs絕對值判斷，以及好與壞的判斷。

使用Arduino 量測電壓

float voltage = 0.00;

const int MIN\_VALUE = 0;

const int MAX\_VALUE = 1023;

// 定義最大和最小的PWM值，對應於0%和100%的占空比

const int MIN\_PWM = 0;

const int MAX\_PWM = 127;

定義變數

    analogValue = analogRead(P\_A5);

    voltage = analogValue;

    voltage = (voltage \*3.30)/1024.00;

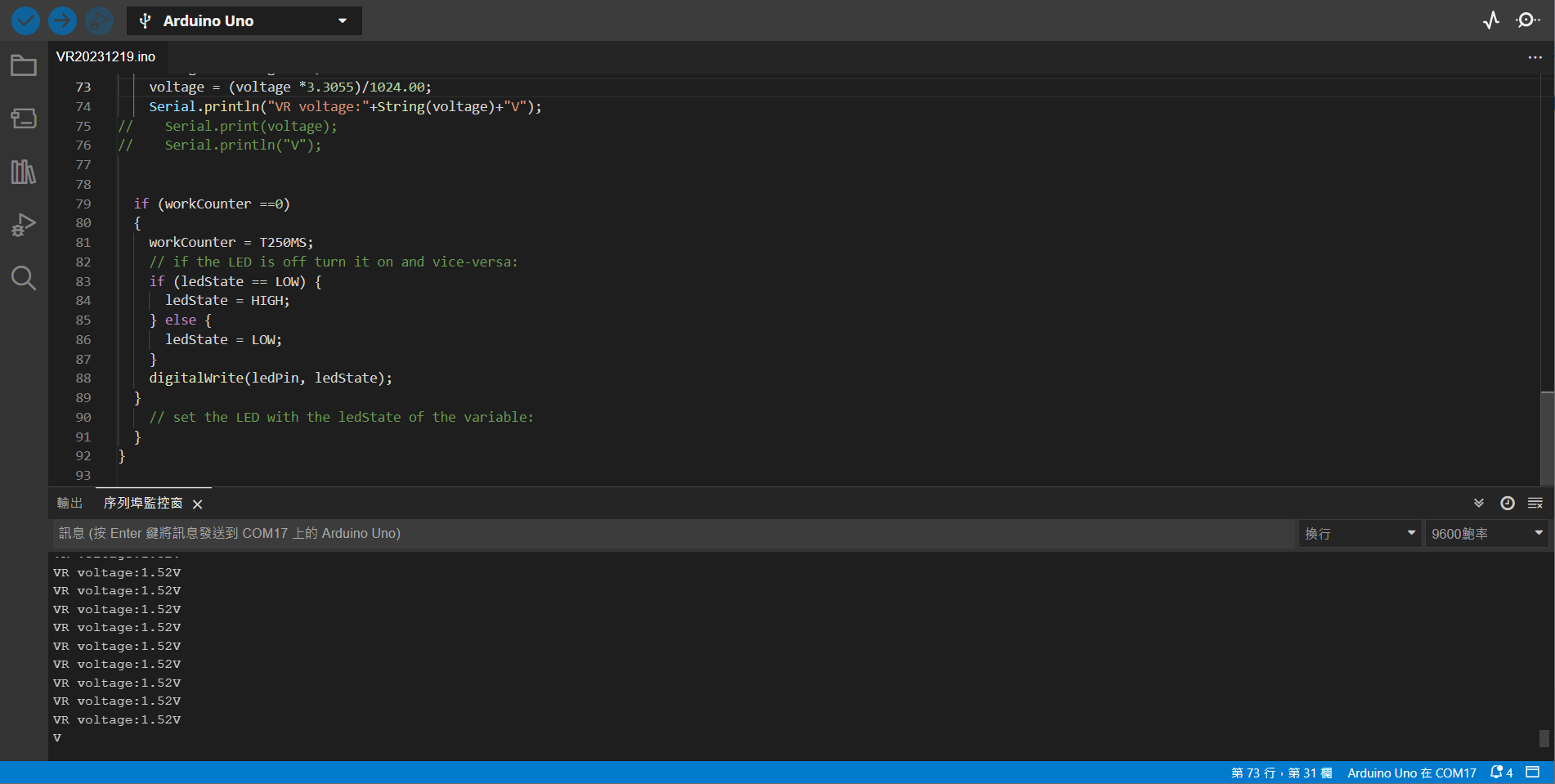
    Serial.print("A5 voltage:");

    Serial.print(voltage);

    Serial.println("V");

    VR\_NEW[i + (8 \* 5)] = map(analogValue, MIN\_VALUE, MAX\_VALUE, MIN\_PWM, MAX\_PWM); //analogValue   //(P\_A5);

簡易程式碼



VR 測試程式碼

/\*

  Blink without Delay

  Turns on and off a light emitting diode (LED) connected to a digital pin,

  without using the delay() function. This means that other code can run at the

  same time without being interrupted by the LED code.

  The circuit:

  - Use the onboard LED.

  - Note: Most Arduinos have an on-board LED you can control. On the UNO, MEGA

    and ZERO it is attached to digital pin 13, on MKR1000 on pin 6. LED\_BUILTIN

    is set to the correct LED pin independent of which board is used.

    If you want to know what pin the on-board LED is connected to on your

    Arduino model, check the Technical Specs of your board at:

    https://www.arduino.cc/en/Main/Products

  created 2005

  by David A. Mellis

  modified 8 Feb 2010

  by Paul Stoffregen

  modified 11 Nov 2013

  by Scott Fitzgerald

  modified 9 Jan 2017

  by Arturo Guadalupi

  This example code is in the public domain.

  https://www.arduino.cc/en/Tutorial/BuiltInExamples/BlinkWithoutDelay

\*/

// constants won't change. Used here to set a pin number:

const int ledPin = LED\_BUILTIN;  // the number of the LED pin

// Variables will change:

int ledState = LOW;  // ledState used to set the LED

#define P\_VR A0

#define TBASE 5

float voltage = 0.000;

#define T250MS 250/TBASE

int workCounter = T250MS;

int analogValue = 0;

// Generally, you should use "unsigned long" for variables that hold time

// The value will quickly become too large for an int to store

unsigned long previousMillis = 0;  // will store last time LED was updated

// constants won't change:

const long interval = 5;  // interval at which to blink (milliseconds)

void setup() {

  Serial.begin(9600);

  Serial.println("VR20231219.ino");

  // set the digital pin as output:

  pinMode(ledPin, OUTPUT);

  pinMode(P\_VR, INPUT);

}

void loop() {

  // here is where you'd put code that needs to be running all the time.

  // check to see if it's time to blink the LED; that is, if the difference

  // between the current time and last time you blinked the LED is bigger than

  // the interval at which you want to blink the LED.

  unsigned long currentMillis = millis();

  if (currentMillis - previousMillis >= interval) {

    // save the last time you blinked the LED

    previousMillis = currentMillis;

  analogValue = analogRead(P\_VR);

    voltage = analogValue;

    voltage = (voltage \*3.3055)/1024.00;

    Serial.println("VR voltage:"+String(voltage)+"V");

//    Serial.print(voltage);

//    Serial.println("V");

  workCounter--;

  if (workCounter ==0)

  {

    workCounter = T250MS;

    // if the LED is off turn it on and vice-versa:

    if (ledState == LOW) {

      ledState = HIGH;

    } else {

      ledState = LOW;

    }

    digitalWrite(ledPin, ledState);

  }

    // set the LED with the ledState of the variable:

  }

}

如何在輸出顯示小數點第四位?

    voltage = (voltage \*3.3030)/1024.00;

    Serial.println("VR voltage:"+String(voltage,4)+"V");

在String(voltage,4)加上,4即可。

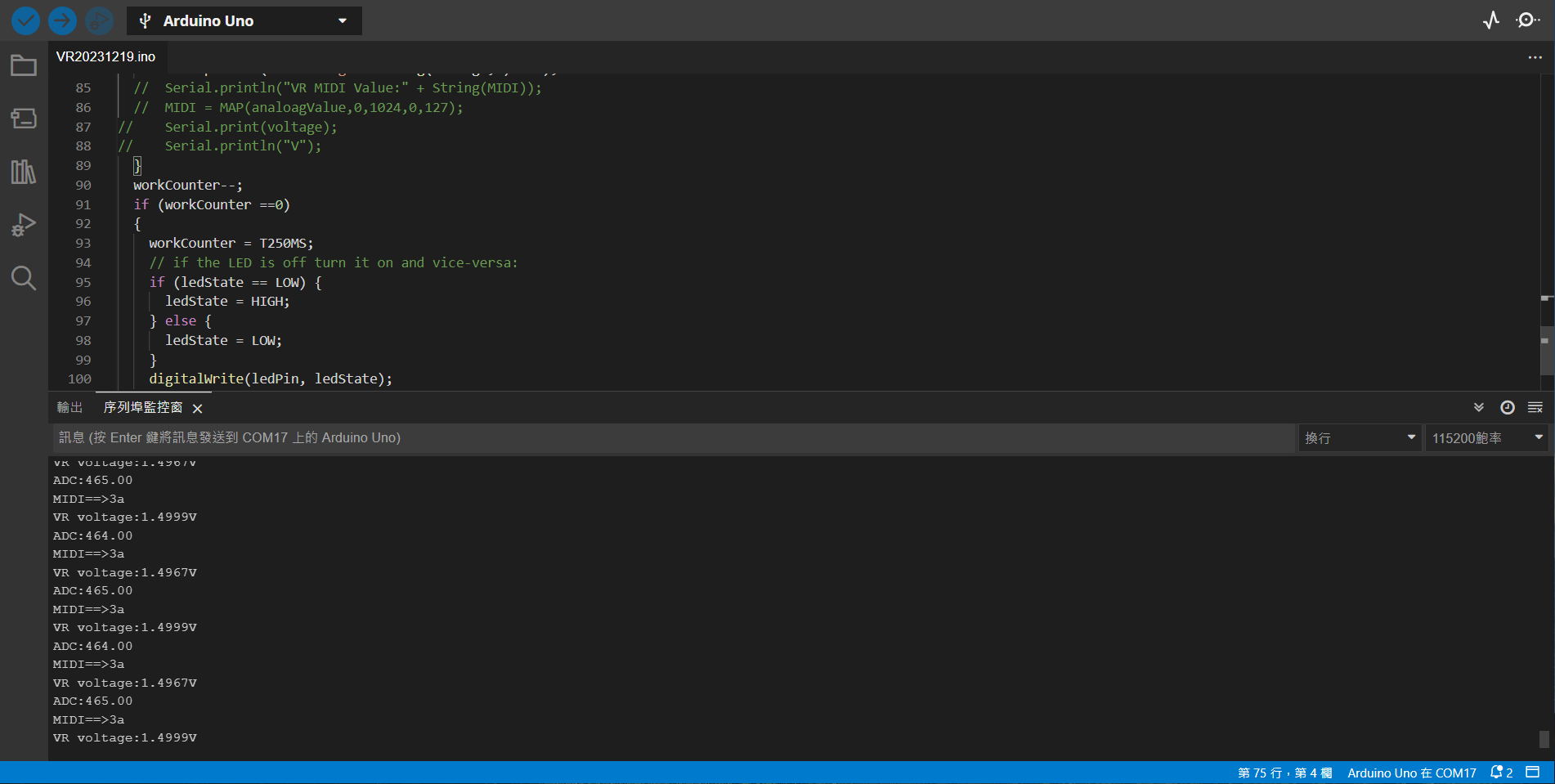
同理，在後面加上,HEX可以轉成16進制

    Serial.println("MIDI==>" + String(shiftvalue,HEX));

2023/12/20 新增加判斷式當ADC數值有變化才丟資料到電腦

  if (analogValue != old\_analogValue)

  {



2023/12/20 新增加OLED

#include <Arduino.h>

#include <U8g2lib.h>

#include <Wire.h>

新增加的函數庫

U8G2\_SSD1306\_128X64\_NONAME\_F\_HW\_I2C u8g2(U8G2\_R0, /\* reset=\*/ U8X8\_PIN\_NONE);

使用硬體設定

Setup()的設定

  u8g2.begin();

  u8g2.enableUTF8Print();   // enable UTF8 support for the Arduino print() function

  u8g2.clearBuffer();         // clear the internal memory

  u8g2.setFont(u8g2\_font\_cu12\_tr);  //u8g2\_font\_unifont\_t\_chinese2); //u8g2\_font\_ncenB08\_tr); // choose a suitable font

  u8g2.setCursor(0, 15);

  u8g2.print(S\_MIDI); // write something to the internal memory

  u8g2.setCursor(0, 40);

  u8g2.print(S\_VOLTAGE);  // write something to the internal memory

  u8g2.sendBuffer();          // transfer internal memory to the display

OLED 顯示程式

CSV 檔案格式輸出

不知道什麼原因，Arduino似乎無法支援長字串的輸出，將長字串拆成短字串輸出

      Serial.print(voltage,4);  //送出量測電壓值

      Serial.print(",");

      Serial.print(SS\_MIDI); //送出16進制MIDI

      Serial.print(",");  //SS\_VOLTAGE + "," + SS\_MIDI );

      Serial.print(DEC\_MIDI + "," ); //送出10進制MIDI

      Serial.print(absvalu);  //送出與中心電壓MIDI值的差距絕對值

      Serial.println( "," + Result); // 輸出CSV檔案格式好方便將log file匯入excel分析

OLED顯示增加好壞判斷字串

      u8g2.clearBuffer();              // clear the internal memory

      u8g2.setFont(u8g2\_font\_cu12\_tr); // u8g2\_font\_unifont\_t\_chinese2); //u8g2\_font\_ncenB08\_tr); // choose a suitable font

      u8g2.setCursor(0, 15);

      u8g2.print(S\_MIDI); // 顯示MIDI值

      u8g2.setCursor(0, 30);

      u8g2.print(voltage,4); // 顯示電壓值

      u8g2.print("V"); // write something to the internal memory

      u8g2.setCursor(0, 45);

      u8g2.print(Result); // 2023/12/25增加顯示測試結果