

```

/* =====
=           Arduino UNO + TEMP_Shield           =
=
=           温度 シールド V1.0 検査スケッチ     =
=
=                                           By M.Obinata =
=                                           2014/9/12    =
=                                           Ver 1.0      =
=====
*/

#include <SoftwareSerial.h>
// #include <avr/pgmspace.h>
#include <Wire.h>
#include <SPI.h>

// I2C_LCD set
#define sdaPin A4      // Arduino I2C SDA
#define sclPin A5      // Arduino I2C SCL

// アナログピン
// 未使用

// デジタルピン

#define SPI_CS1 10     // SPI CS Reserve (Not USE)
#define SPI_CS2 9      // SPI CS
#define SEL_TMP 8      // SEL TEMP Senser

#define EXT_CON 7      // EXT CONTROL
#define OUT_PWM 6      // OUT Poer (PWM)
#define INPUT_SEN 5    // INPUT Port
#define SW 4          // プッシュ SW

// =====

void setup() {

  Serial.begin(9600); // PC 設定

  pinMode(SPI_CS2, OUTPUT);
  pinMode(SEL_TMP, OUTPUT);
  pinMode(EXT_CON, OUTPUT);
  pinMode(OUT_PWM, OUTPUT);
  pinMode(INPUT_SEN, INPUT);
  pinMode(SW, INPUT);

  lcd_init();// I2C LCD 初期化

// SPI 初期化
SPI.begin();
SPI.setBitOrder(MSBFIRST);
SPI.setClockDivider(SPI_CLOCK_DIV4);
SPI.setDataMode(SPI_MODE0);

// 初期メッセージ
Serial.println("Start TEMP_Shield TEST");

  ini_Title(); // TEMP Get Push SW1

// 初期化
digitalWrite(SPI_CS2, HIGH);
digitalWrite(SEL_TMP, HIGH);
digitalWrite(EXT_CON, LOW);
digitalWrite(OUT_PWM, HIGH);

// TEST開始 メッセージ
for (int i=0; i <= 50; i++){
  if(digitalRead(SW)==LOW){
  }
  else{
    i=0;
  }
}
  Serial.println("TEMP_Shield TEST Start!!!");

  TEMP_Title(); // TEMP Get Start!!

  delay(2000);

```

```

}

char pr[8]="";

// ===== LOOP =====

void loop() {

//=====
unsigned int thermocouple; // 14-Bit Thermocouple Temperature Data + 2-Bit
unsigned int internal; // 12-Bit Internal Temperature Data + 4-Bit
float disp; // display value

delay(500);
digitalWrite(SPI_CS2, LOW); // Enable the chip
thermocouple = (unsigned int)SPI.transfer(0x00) << 8; // Read high byte thermocouple
thermocouple |= (unsigned int)SPI.transfer(0x00); // Read low byte thermocouple
internal = (unsigned int)SPI.transfer(0x00) << 8; // Read high byte internal
internal |= (unsigned int)SPI.transfer(0x00); // Read low byte internal
digitalWrite(SPI_CS2, HIGH); // Disable the chip

if((thermocouple & 0x0001) != 0) {
  Serial.print("ERROR: ");
  if (digitalRead(SW)==0) { //Push SW
    sprintf(pr, "%8s", "CN6 ERR. ");
  }
  else {
    sprintf(pr, "%8s", "CN7 ERR. ");
  }
  lcd.setCursor(0,0);
  lcd_printStr(pr);

  if ((internal & 0x0004) !=0) {
    Serial.print("Short to Vcc, ");
    TEMP_ERR1(); //ERROR!! ShortVCC
  }
  if ((internal & 0x0002) !=0) {
    Serial.print("Short to GND, ");
    TEMP_ERR2(); //ERROR!! ShortGND
  }
  if ((internal & 0x0001) !=0) {
    Serial.print("Open Circuit, ");
    TEMP_ERR3(); //ERROR!! Open(NC)
  }
  Serial.println();
}
else {
  if((thermocouple & 0x8000) == 0) { // 0°C以上 above 0 Degrees Celsius
    disp = (thermocouple >> 2) * 0.25;
  } else { // 0°C未満 below zero
    disp = (0x3fff - (thermocouple >> 2) + 1) * -0.25;
  }
  Serial.print(thermocouple, HEX);
  Serial.print(" : ");
  Serial.print(disp);

  // Disp LCD
  sprintf(pr, "%+3d.%02d c", (int)disp, (int)(abs(disp)*100.0)%100);

  if (digitalRead(SW)==0) { //Push SW
    TEMP_ext1(); // EXT1 TEMP
  }
  else {
    TEMP_ext2(); // EXT2 TEMP
  }

  lcd.setCursor(0,1);
  lcd_printStr(pr);
  delay(2000);

  Serial.print(" // ");

  if((internal & 0x8000) == 0) { // 0°C以上 above 0 Degrees Celsius
    disp = (internal >> 4) * 0.0625;
  } else { // 0°C未満 below zero
    disp = (((0xffff - internal) >> 4) + 1) * -0.0625;
  }
  Serial.print(internal, HEX);
  Serial.print(" : ");
  Serial.print(disp);

  Serial.println();
}
}

```

```

// Disp LCD
sprintf(pr, "%+3d.%02d c", (int)disp, (int) (abs(disp)*100.0)%100);
TEMP_internal(); // IC TEMP
lcd_setCursor(0, 1);
lcd_printStr(pr);
delay(2000);
}
if (digitalRead(SW)==0) {
digitalWrite(SEL_TMP, LOW);
digitalWrite(EXT_CON, HIGH);
digitalWrite(OUT_PWM, LOW);
}
else
{
digitalWrite(SEL_TMP, HIGH);
digitalWrite(EXT_CON, LOW);
digitalWrite(OUT_PWM, HIGH);
}
}

//=====
//
// sprintf(pr, " %3d.%d C", val, abs(val%10));
// lcd_setCursor(0, 1);
// lcd_printStr(pr);
//-----

// 関数 =====

void ini_Title() {
char title1[8];
char ttl1[]={0x54, 0x45, 0x4D, 0x50, 0x20, 0x47, 0x45, 0x54}; //TEMP GET
strcpy(title1, ttl1);

char title2[8];
char ttl2[]={0x50, 0x75, 0x73, 0x68, 0x20, 0x53, 0x57, 0x31}; //Push SW1
strcpy(title2, ttl2);

lcd_setCursor(0, 0);
lcd_printStr(title1);
lcd_setCursor(0, 1);
lcd_printStr(title2);
}

void TEMP_Title() {
// char title1[8];
// char ttl1[]={0x54, 0x45, 0x4D, 0x50, 0x20, 0x47, 0x45, 0x54}; //TEMP GET
// strcpy(title1, ttl1);

char title2[8];
char ttl2[]={0x20, 0x53, 0x74, 0x61, 0x72, 0x74, 0x21, 0x20}; // Start!
strcpy(title2, ttl2);

// lcd_setCursor(0, 0);
// lcd_printStr(title1);
lcd_setCursor(0, 1);
lcd_printStr(title2);
}

void TEMP_ext1() {
char title1[8];
char ttl1[]={0x43, 0x4E, 0x36, 0x20, 0x54, 0x45, 0x4D, 0x50}; //CN6 TEMP
strcpy(title1, ttl1);

lcd_setCursor(0, 0);
lcd_printStr(title1);
}

void TEMP_ext2() {
char title1[8];
char ttl1[]={0x43, 0x4E, 0x37, 0x20, 0x54, 0x45, 0x4D, 0x50}; //CN7 TEMP
strcpy(title1, ttl1);

lcd_setCursor(0, 0);
lcd_printStr(title1);
}

void TEMP_internal() {
char title1[8];

```

```

char tt11[]={0x20,0x49,0x43,0x20,0x54,0x45,0x4D,0x50}; //IC TEMP
strcpy(title1,tt11);

lcd_setCursor(0,0);
lcd_printStr(title1);
}

void TEMP_ERR1() {           //ERROR!! ShortVCC
// char title1[8];
// char tt11[]={0x20,0x45,0x52,0x52,0x4F,0x52,0x21,0x21}; // ERROR!!
// strcpy(title1,tt11);

char title2[8];
char tt12[]={0x53,0xC8,0xCF,0x72,0x74,0x56,0x43,0x43}; //ShortVCC
strcpy(title2,tt12);

// lcd_setCursor(0,0);
// lcd_printStr(title1);
lcd_setCursor(0,1);
lcd_printStr(title2);
}

void TEMP_ERR2() {           //ERROR!! ShortGND
// char title1[8];
// char tt11[]={0x20,0x45,0x52,0x52,0x4F,0x52,0x21,0x21}; // ERROR!!
// strcpy(title1,tt11);

char title2[8];
char tt12[]={0x53,0xC8,0xCF,0x72,0x74,0x47,0x4E,0x44}; //ShortGND
strcpy(title2,tt12);

// lcd_setCursor(0,0);
// lcd_printStr(title1);
lcd_setCursor(0,1);
lcd_printStr(title2);
}

void TEMP_ERR3() {           //ERROR!! Open (NC)
// char title1[8];
// char tt11[]={0x20,0x45,0x52,0x52,0x4F,0x52,0x21,0x21}; // ERROR!!
// strcpy(title1,tt11);

char title2[8];
char tt12[]={0x4F,0x70,0x65,0x6E,0x28,0x4E,0x43,0x29}; //Open (NC)
strcpy(title2,tt12);

// lcd_setCursor(0,0);
// lcd_printStr(title1);
lcd_setCursor(0,1);
lcd_printStr(title2);
}

```