/\*\*\*\*\*\*\*\*\* Code complet de la sonde mesurant le rayonnement global \*\*\*\*\*\*\*\*\*/

// Ajout de librairies pour prendre en compte les différents constituants du capteur (affichage LCD, carte SD, horloge, communication)

#include <LiquidCrystal.h>

#include <SD.h>

#include <RTClib.h>

#include <SPI.h>

#include <Wire.h>

int analogPin = 2; // mesure sur la broche 2

int num = 100; // initialisation tampon

int n = 1;

int year;

int month;

int day;

int hour;

int minute;

int second;

int yeardeb;

int monthdeb;

int daydeb;

int hourdeb;

int minutedeb;

int seconddeb;

int temps;

Sd2Card card;

const int chipSelect = 10;

char heure;

float temp = 0.0;

int numanc = 0;

float voltage = 0.0;

// Partie vérification des broches de l’afficheur

LiquidCrystal lcd(8, 9, 4, 5, 6, 7); // broches de connexion afficheur utilisé

// LiquidCrystal lcd(7, 6, 5, 4, 3, 2); // connexions autre type d’afficheur

RTC\_DS1307 RTC; // fait appel à la Classe RTC\_DS1307(puce d'horloge)

File logfile;

void setup()

{

  Serial.begin(9600);

  pinMode (analogPin, INPUT); // entrée du capteur

  SD.begin(10); // initialise la carte SD pour prendre les mesures

  analogReference(INTERNAL); // val ref = 1.1V

  // Entraîne une résolution de 1mV sur la tension

  lcd.begin(16, 2); // utilisation d'un écran 16 colonnes et 2 lignes

  RTC.begin(); // démarrage de la librairie RTClib.h

  // Si RTC ne fonctionne pas

  if (! RTC.isrunning())

  {

    Serial.println("RTC ne fonctionne pas !");

    // Synchronisation de l’heure de la puce RTC sur celle de l’ordinateur

    RTC.adjust(DateTime(\_\_DATE\_\_, \_\_TIME\_\_));

    // Cela fonctionne également avec :

// RTC.adjust(DateTime("Dec 5 2012","12:00:00"));

// RTC.adjust(DateTime(2014, 1, 21, 3, 0, 0));

  }

  RTC.adjust(DateTime(\_\_DATE\_\_, \_\_TIME\_\_));

  lcd.clear();

  lcd.setCursor(0, 0); // curseur dans le coin supérieur droit

  if (!card.init(SPI\_HALF\_SPEED, chipSelect))

  {

    lcd.print("ERREUR SD");

    delay(20000); // attente 20 secondes

    return;

  }

  else

  {

    lcd.print("SD OK");

  }

// Mise à jour de l’horloge lorsqu’il y’a connexion à l’ordinateur (téléversage)

  DateTime now = RTC.now();

  yeardeb = now.year();

  monthdeb = now.month();

  daydeb= now.day();

  hourdeb = now.hour();

  minutedeb = now.minute();

  seconddeb = now.second();

  logfile = SD.open("data2.txt", FILE\_WRITE);

  // Enregistrement du 1er champ sur carte SD

  logfile.print("numero");

  logfile.print(",");

  logfile.print("heure");

  logfile.print(",");

  logfile.print("secondes");

  logfile.print(",");

  logfile.print("rayonnement (W/m2)");

  logfile.print(",");

  logfile.print("jour:");

  logfile.print(daydeb);

  logfile.print(",");

  logfile.print("mois:");

  logfile.print(monthdeb);

  logfile.println();

  logfile.close();

  // Affichage de l’heure sur écran LCD

  lcd.clear();

  lcd.setCursor(0, 0); // curseur dans le coin supérieur droit

  lcd.print(hourdeb);

  lcd.print("h");

  lcd.setCursor(5, 0);

  lcd.print(minutedeb);

}

void loop()

{

  numanc = 0.;

  delay(10000); // attente 10s

  // récupération de l'heure RTC

  DateTime now = RTC.now();

  year = now.year();

  month = now.month();

  day = now.day();

  hour = now.hour();

  minute = now.minute();

  second = now.second();

  lcd.clear();

  lcd.setCursor(0, 0); // curseur dans le coin supérieur droit

  lcd.print("At.raynt");

  lcd.setCursor(10,0);

  lcd.print("="); // on écrit le message "At. raynt ="

  numanc = num;

  num = analogRead(analogPin); // on récupère la tension sur la broche 2

  voltage = num \* (1.10 / 1023);

  temp = voltage\*2000 ; // affichage en W/m2

  char message[16] = "";

  lcd.setCursor(11, 0);

  lcd.print(temp,1); // afficher la tension sur LCD

  lcd.home();

  lcd.clear();

  lcd.setCursor(0, 0); // curseur dans le coin supérieur droit

  lcd.print("Rayt OK"); // on écrit le message  "Rayt OK"

  lcd.setCursor(11, 0);

  lcd.print(temp,1); // afficher la tension sur LCD

  temps=abs((hour-hourdeb)\*3600+(minute-minutedeb)\*60+(second-seconddeb));

  // enregistrement du champ de données sur SD

  logfile = SD.open("data2.txt", FILE\_WRITE);

  logfile.print(n);

  logfile.print(",");

  logfile.print(hour, DEC);

  logfile.print(':');

  logfile.print(minute, DEC);

  logfile.print(':');

  logfile.print(second, DEC);

  logfile.print(",");

  logfile.print(temps);

  logfile.print(",");

  logfile.print(temp);

  logfile.println();

  logfile.close();

  n = n + 1;

  numanc=0;

  num=100;

  delay (110000); // attente de 2 minutes (prise en compte 10 secondes)

  lcd.clear();

  lcd.setCursor(0, 0); // curseur dans le coin supérieur droit

}