**Lectura\_sensor\_IR\_2\_sensores\_LAPSUS\_RESET\_GRAVEDAD\_REVISADO\_LCD**

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

//Crear el objeto lcd dirección 0x27 y 16 columnas x 2 filas

LiquidCrystal\_I2C lcd(0x27,16,2); //

long tiempo = 0; // variable general de tiempo

int i=0; // contador general

int corte[] = {0.0,0.0}; // valores de corte Vin en los sensores.

long lapsus[]={0,0}; // tiempo en cada sensor

long medida = 0; // valor de la medida temporal

double umbral = 1.3; // valor del umbral de corte

double deltat = 0; // medida temporal en segundos

double gravedad = 0.0; // valor de la gravedad

double error = 0.0; // error respecto a 9.8065m/s2

// the setup routine runs once when you press reset:

void setup() {

**for (i=0;i<=100;i++) {**

**corte[0] = corte[0]+analogRead(A0); // valor inicial sensor A0**

**corte[1] = corte[1]+analogRead(A1); // valor inicial sensor A1**

**}**

**corte[0]=corte[0]/100;**

**corte[1]=corte[1]/100;**

// Pantalla LCD

// Inicializar el LCD

lcd.init();

//Encender la luz de fondo.

lcd.backlight();

// Escribimos el Mensaje en el LCD.

lcd.setCursor(0, 0); // Fila 0

lcd.print("V.ini. A0:" + (String) corte[0]);

lcd.setCursor(0, 1); // Fila 1

lcd.print("V.ini. A1:" + (String) corte[1]);

// initialize serial communication at 9600 bits per second:

Serial.begin(9600);

Serial.println("Preparado para lanzar el móvil");

Serial.println("Valor inicial sensor A0: " + (String) corte[0]);

Serial.println("Valor inicial sensor A1: " + (String) corte[1]);

delay(3000); // Espera 3 segundos

lcd.clear();

lcd.setCursor(0,0);

lcd.print("Esperando tirada...");

}

void(\* resetFunc) (void) = 0; //declare reset function at address 0

void resetear() {

// Si se presiona PIN 7, hay un RESET

if (digitalRead(7)==HIGH) {

Serial.println("Otra vez!!!");

delay(1000);

resetFunc(); //call reset

}

}

// the loop routine runs over and over again forever:

void loop() {

tiempo = micros(); //contando el tiempo desde que arranca ARDUINO

// Serial.println("sensor A0: " + (String) analogRead(A0));

// Serial.println("sensor A1: " + (String) analogRead(A1));

// Sensor A

// Si la lectura supera en un 100% a la de corte y aún no se ha definido lapsus[0]

**if (analogRead(A0)>corte[0]\*umbral && lapsus[0]==0) {**

**lapsus[0]=tiempo;**

**Serial.println("Valor del tiempo sensor A0: " + (String) lapsus[0] + " // Valor sensor A0: " + analogRead(A0)); // toma el valor de tiempo lapsus[0]**

**}**

// Sensor B

// Si la lectura supera en un 100% a la de corte del segundo sensor y aún no se ha definido medida

**if ((analogRead(A1)>corte[1]\*umbral) && medida==0 && lapsus[0]>0) {**

**lapsus[1]=tiempo; // captura en lapsus[1] la segunda medida**

**medida = lapsus[1]-lapsus[0]; // calcula la medida temporal**

**Serial.println("Valor del tiempo sensor A1: " + (String) lapsus[1]+ " // Valor sensor A1: " + analogRead(A1)); // toma el valor de tiempo lapsus[1]**

**Serial.println("Diferencia de tiempo en us: " + (String) medida); // diferencia temporal**

**Serial.println("Medida en milisegundos: " + (String) ((float) (medida/1000.0)) +" ms.");**

**deltat = (double) (medida/1000000.0); //**

**Serial.println("Medida en segundos: " + (String) (deltat) +" s.");**

**gravedad = sqrt(2\*0.24) - sqrt(2\*0.03); // fórmula.**

**gravedad = gravedad / deltat;**

**gravedad = pow(gravedad,2);**

**error = 100\* (abs(gravedad - 9.8065) / 9.8065);**

**// gravedad = g**

**Serial.println("RESULTADO: " + (String) gravedad +" m/s2.");**

**Serial.println("Error respecto valor teórico: "+ (String) error + "%");**

**// En pantalla**

**lcd.clear();**

**lcd.setCursor(0,0);**

**lcd.print("t= "+ (String) deltat+"s ");**

**lcd.setCursor(0,1);**

**lcd.print("g = "+ (String) gravedad + " m/s2");**

**i = 0;**

**while (1==1) {**

**i = (i+1)\*(i<32);**

**if ((i%2)==0) {**

**lcd.setCursor(0,0);**

**lcd.print("error= " + (String) error +"% ");**

**} else {**

**lcd.setCursor(0,0);**

**lcd.print("t= "+ (String) deltat+" s ");**

**}**

**delay(1500);**

**resetear();**

**}**

}

// Si se presiona PIN 6, hay un RESET

resetear();

}