

```

////////// variables Co2:
int co2 = 9999;
int cycleTotal = 0;
int timeSpan = 0;
int timePeriod = 0;
int analogPin = A0;
// Calibracion:
float v400ppm = 3.399;
float v40000ppm = 2.230;
float deltavs = v400ppm - v40000ppm;
float A = deltavs/(log10(400) - log10(40000));
float B = log10(400);
////////// variables PAR:
float Intercept = 0;
float Slope = 430;
int ReadingNumber=0;
float Time;
////////// variables Temperatura/Humedad:
int SHT_clockPin = 3;
int SHT_dataPin = 2;

void setup(){
  Serial.begin(9600);
  Serial1.begin(9600);
  pinMode(analogPin, INPUT);
}

void loop(){
  //////////// Co2:
  int data = analogRead(analogPin);
  float voltage = data/204.6;
  float power = ((voltage - v400ppm)/A) + B;
  float co2ppm = pow(10,power);
  co2 = co2ppm;
  //////////// PAR:
  float Count = analogRead(A1);
  float Voltage = Count / 1023 * 5.0;
  float Sensor= Intercept + Voltage * Slope;
  float Par = Sensor;
  //////////// Temp./Humedad:
  float temperatura = getTemperature();
  float humedad = getHumidity();

  ////////////Lectura de datos:
  //Puerto serial 1 = Dioxido de carbono:
  Serial.println(co2);
  delay(1000);
  //Puerto serial 2 = Display:
  clearLCD();
  backlightOn();
  selectLineOne();
  Serial1.print("Temperatura");
  selectLineTwo();
  Serial1.print(temperatura);
  Serial1.print(" C");
  delay(5000);
  clearLCD();
  backlightOn();
  selectLineOne();
  Serial1.print("Humedad");
  selectLineTwo();
  Serial1.print(humedad);
  Serial1.print(" %");
  delay(5000);
  clearLCD();
}

```

```

        backlightOn();
        selectLineOne();
        Serial1.print("Rad.fotosintesis");
        selectLineTwo();
        Serial1.print(Par);
        Serial1.print(" umol m-2s-1");
        delay(5000);
    }

    //////////////////////////////////////////////////Funciones Temperatura/Humedad:
    float getTemperature(){
        SHT_sendCommand(B00000011, SHT_dataPin, SHT_clockPin);
        SHT_waitForResult(SHT_dataPin);
        int val = SHT_getData(SHT_dataPin, SHT_clockPin);
        SHT_skipCrc(SHT_dataPin, SHT_clockPin);
        return (float)val * 0.01 - 40;
    }
    float getHumidity(){
        SHT_sendCommand(B00000101, SHT_dataPin, SHT_clockPin);
        SHT_waitForResult(SHT_dataPin);
        int val = SHT_getData(SHT_dataPin, SHT_clockPin);
        SHT_skipCrc(SHT_dataPin, SHT_clockPin);
        return -4.0 + 0.0405 * val + -0.0000028 * val * val;
    }
    void SHT_sendCommand(int command, int dataPin, int clockPin){
        pinMode(dataPin, OUTPUT);
        pinMode(clockPin, OUTPUT);
        digitalWrite(dataPin, HIGH);
        digitalWrite(clockPin, HIGH);
        digitalWrite(dataPin, LOW);
        digitalWrite(clockPin, LOW);
        digitalWrite(clockPin, HIGH);
        digitalWrite(dataPin, HIGH);
        digitalWrite(clockPin, LOW);
        shiftOut(dataPin, clockPin, MSBFIRST, command);
        digitalWrite(clockPin, HIGH);
        pinMode(dataPin, INPUT);
        if (digitalRead(dataPin)) Serial1.println("ACK error 0");
        digitalWrite(clockPin, LOW);
        if (!digitalRead(dataPin)) Serial1.println("ACK error 1");
    }
    void SHT_waitForResult(int dataPin){
        pinMode(dataPin, INPUT);
        int ack;
        for (int i = 0; i < 1000; ++i){
            delay(2);
            ack = digitalRead(dataPin);
            if (ack == LOW) break;
        }
        if (ack == HIGH) Serial1.println("ACK error 2");
    }
    int SHT_getData(int dataPin, int clockPin){
        pinMode(dataPin, INPUT);
        pinMode(clockPin, OUTPUT);
        byte MSB = shiftIn(dataPin, clockPin, MSBFIRST);
        pinMode
        (dataPin, OUTPUT);
        digitalWrite(dataPin, HIGH);
        digitalWrite(dataPin, LOW);
        digitalWrite(clockPin, HIGH);
        digitalWrite(clockPin, LOW);
        pinMode(dataPin, INPUT);
        byte LSB = shiftIn(dataPin, clockPin, MSBFIRST);
        return ((MSB << 8) | LSB);
    }

```

```

}
void SHT_skipCrc(int dataPin, int clockPin){
    pinMode(dataPin, OUTPUT);
    pinMode(clockPin, OUTPUT);
    digitalWrite(dataPin, HIGH);
    digitalWrite(clockPin, HIGH);
    digitalWrite(clockPin, LOW);
}

//////////funciones del Display:
void selectLineOne(){
    Serial1.write(0xFE);
    Serial1.write(128);
    delay(10);
}
void selectLineTwo(){
    Serial1.write(0xFE);
    Serial1.write(192);
    delay(10);
}
void clearLCD(){
    Serial1.write(0xFE);
    Serial1.write(0x01);
    delay(5);
}
void backlightOn(){
    Serial1.write(0x7C);
    Serial1.write(30);
    delay(5);
}

```