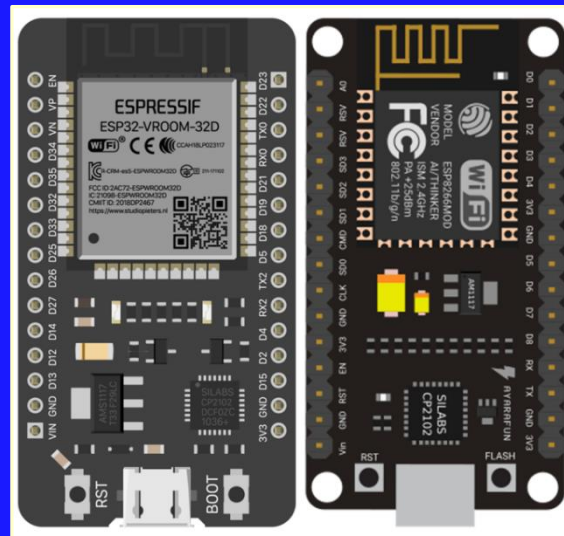




Aplicaciones de sistemas embebidos ESP32: Presentación de OPENPLC y Conectividad Utilizando ESP32



Parte II – Aplicaciones – Ing. Néstor Cortez - Ing. Rafael Oliva
Instituto de Tecnología Aplicada (ITA-UARG)

JORNADAS DE INFORMÁTICA 2025 – UNPA /UARG
3 de Noviembre de 2025



¿QUÉ ES UN P.L.C?



Es un sistema electrónico programable diseñado para controlar diversos tipos de máquinas o procesos en tiempo real en un entorno industrial.



¿Dónde se utiliza?



- Principalmente en industrias para automatización y control de procesos.

Tambien en..domótica





SISTEMAS
EMBEBIDOS
2025



Computadora Industrial
Abierta Argentina
Desarrollo colectivo



ESPRESSIF

OPENPLC

TO A MORE
OPEN FUTURE



EDITOR <> RUNTIME <>

CIAA-NXP



Fuente de alimentación

Fuente de 24VDC

Interfaces de comunicación

- Ethernet con soporte PoE (requiere módulo de alimentación PoE) (
- USB Device Auxiliar
- RS232
- RS485

Entradas/Salidas

- 8 entradas digitales optoacopladas
- 4 entradas analógicas configurables por jumper 0-10V o 0-20mA (
- 4 salidas open-drain de 24V, 1A
- 4 salidas a relé 24V, 2A (
- 1 salida analógica configurable por jumper 0-10V o 0-20mA (

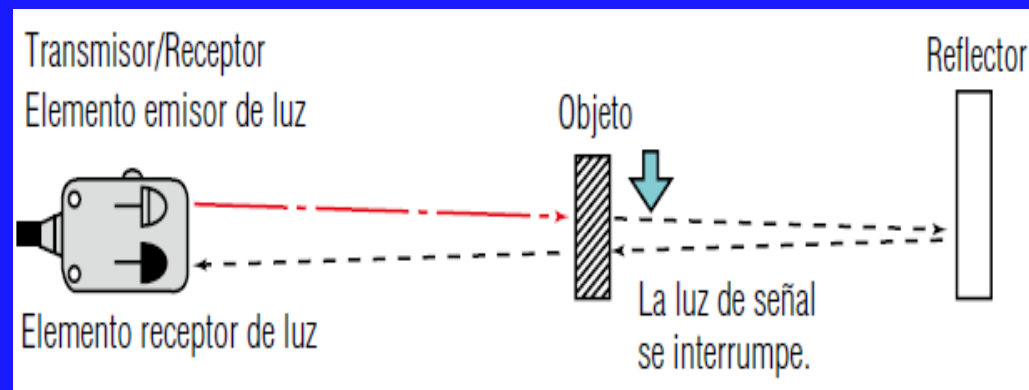


ENTRADAS - SALIDAS - DIGITALES

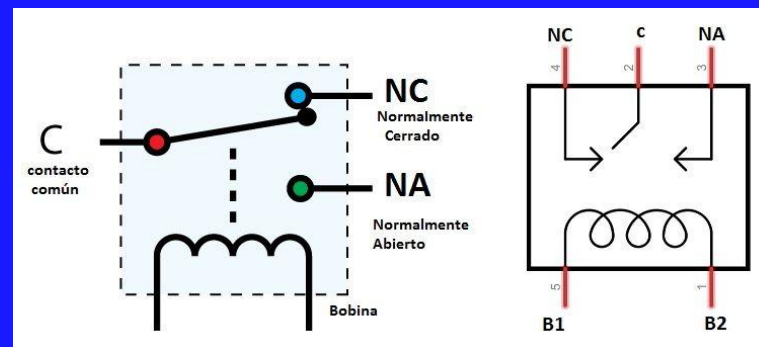
SENSORES CAPACITIVOS - INDUCTIVOS



SENSOR ÓPTICO RETROREFLECTIVO



SALIDA RELÉ



Lenguajes de programación

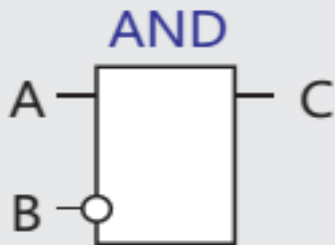
IL

```
LD    A
ANDN  B
ST    C
```

ST

```
C = A AND NOT B
```

FBD

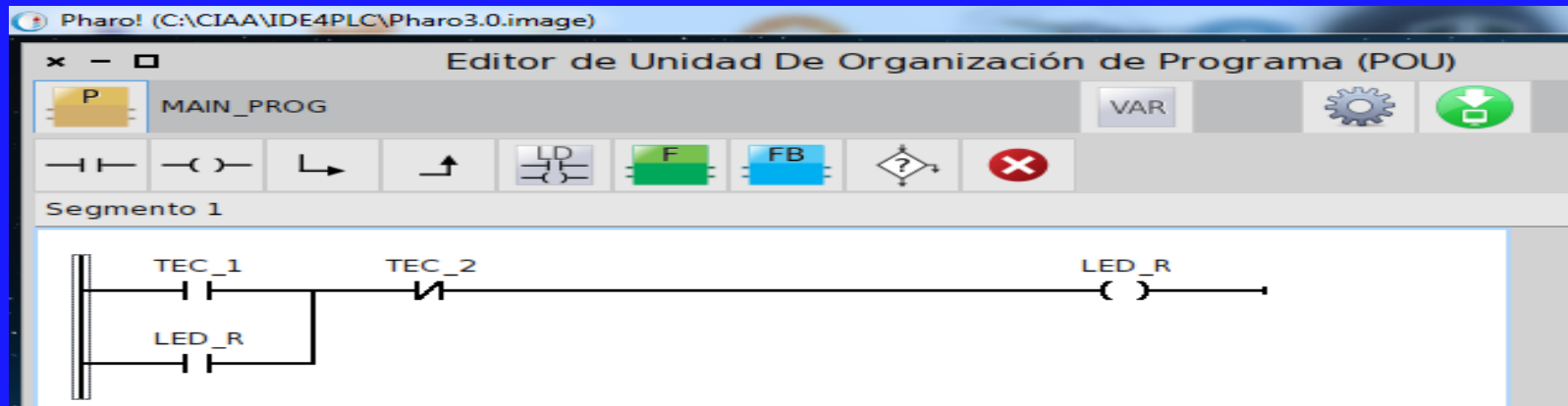


LD

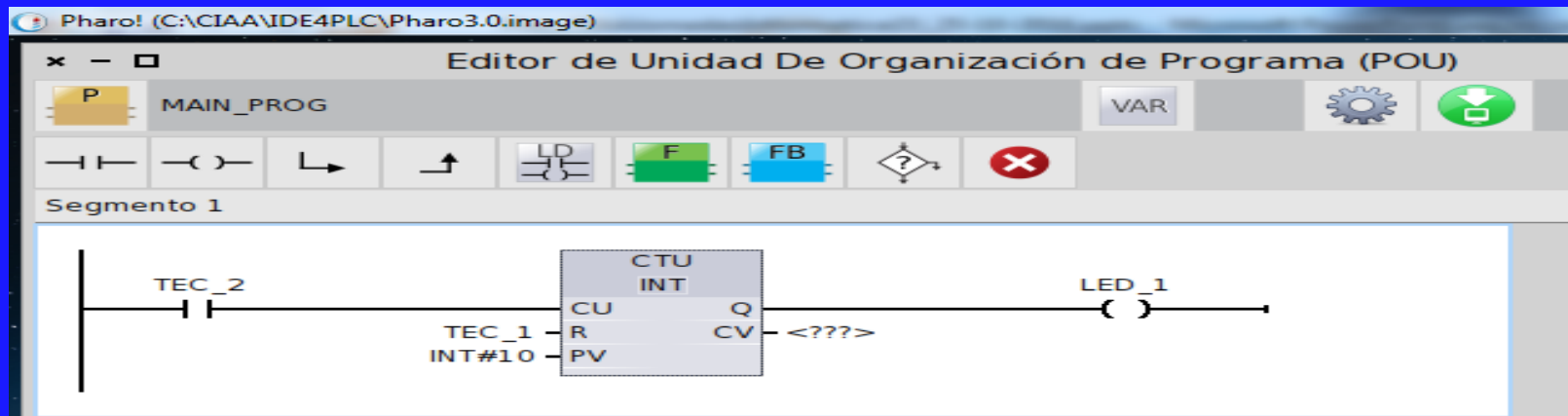




Ejemplos: Circuito con auto retención

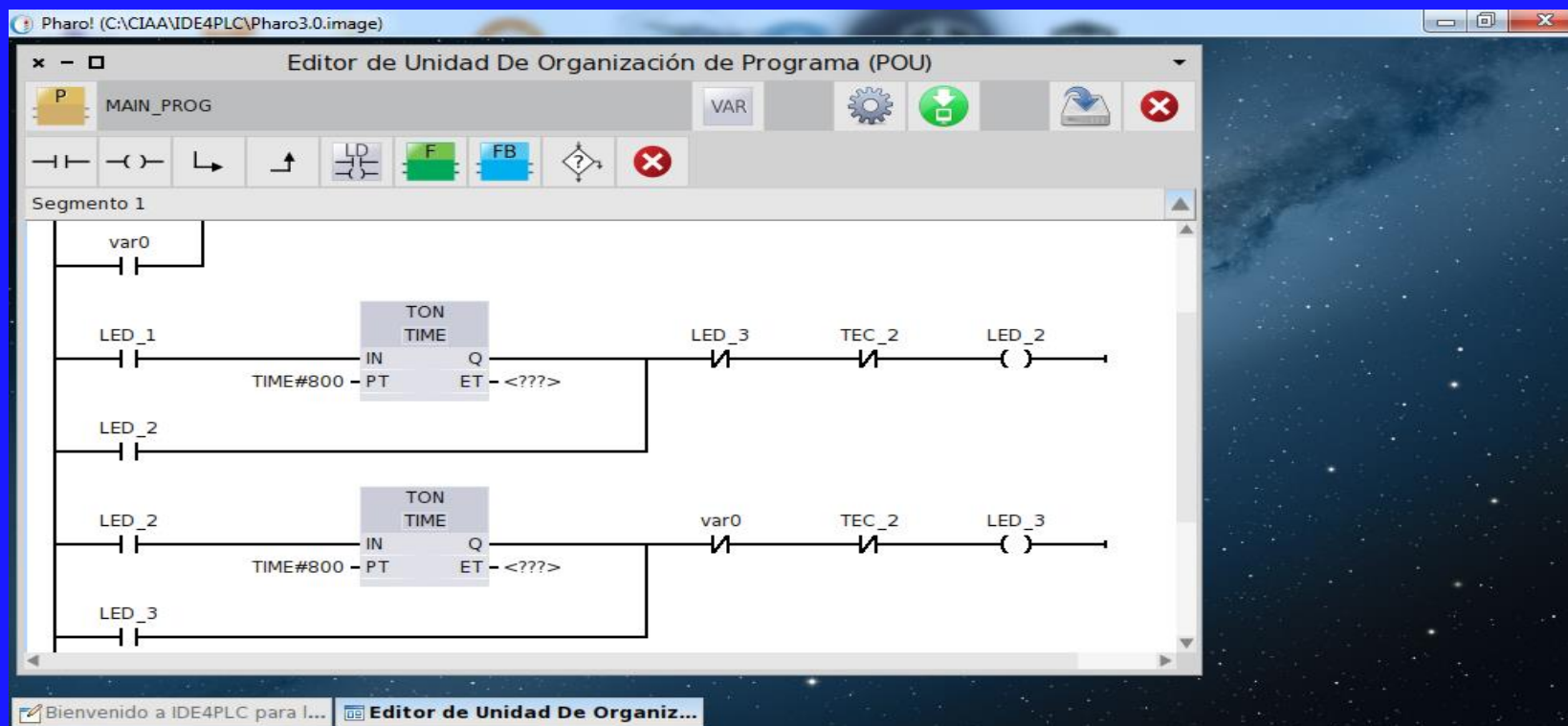


Ejemplos: Circuito con contador y reset





SOFTWARE DE SECUENCIA DE LUCES CONTENIDO EN EDU CIAA Y CIAA – NXP





OPENPLC <>

TO A MORE OPEN FUTURE

EDITOR <> RUNTIME <>

SUITE COMPLETO DE PLC

Cree sus proyectos en OpenPLC Editor, un completo IDE de programación de PLC. Ejecútelos en OpenPLC Runtime, un entorno de ejecución de PLC multiplataforma y portátil.



OpenPLC es un controlador lógico programable (PLC) de código abierto basado en un software fácil de usar. Es el primer PLC de código abierto estandarizado y totalmente funcional, tanto a nivel de software como de hardware. El proyecto OpenPLC se creó conforme a la norma IEC 61131-3, que define la arquitectura de software básica y los lenguajes de programación para PLC.

El editor OpenPLC es el software que se ejecuta en su ordenador y, como ya se ha mencionado, se utiliza para crear sus programas PLC. Es muy sencillo de usar y admite los cinco lenguajes definidos en la norma IEC 61131-3: Lógica de escalera (LD), Diagrama de bloques de función (FBD), Lista de instrucciones (IL), Texto estructurado (ST) y Diagrama de función secuencial (SFC).

- Arduino Uno/Nano/Leonardo/Micro
- Arduino Mega / Due
- Arduino Nano Every / IoT / BLE
- Conexión Arduino RB2040
- Arduino Mkr / Zero / WiFi
- Arduino Pro (Control de máquinas y EDGE)
- Controllino Maxi / Automatización / Mega / Mini
- Productividad Abierta P1AM
- ESP8266 (nodemcu)

- ESP32
- Raspberry Pi 2 / 3 / 4
- PiXtend
- Plataforma industrial UniPi
- Neurona PLC
- Enlace Zumlink de FreeWave
- FreeWave ZumIQ
- Windows (objetivo genérico como PLC virtual)
- Linux (objetivo genérico como PLC virtual)



ESPRESSIF

OPENPLC

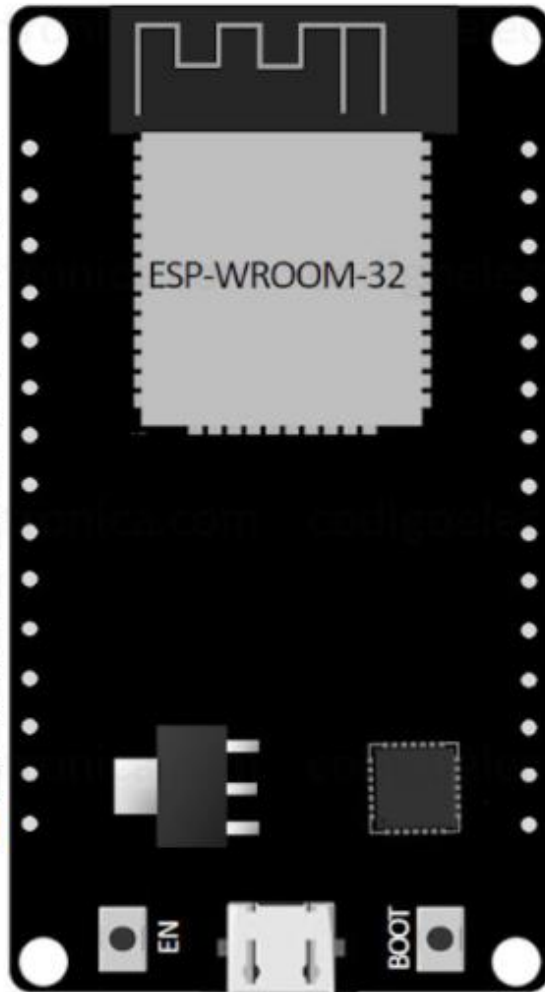
TO A MORE
OPEN FUTURE

EDITOR <> RUNTIME <>



ESP- WROOM-32

- Microprocesador de 32 – bits
- Interfaz serie-USB y regulador de voltaje
- Comunicación inalámbrica Wifi – Bluetooth
- Interfases periféricas: I2C, SPI, CAN, PWM, etc
- Entradas digitales y analógicas de 12 bit
- Salidas Analógicas
- Salidas PWM
- Controlador SD/MMC
- Controlador remoto infrarrojo
- Algunos modelos tienen integrados: Cámaras, pantallas Oled, Módulos LoRa, etc.





DIRECCIONAMIENTO FÍSICO DE ENTRADAS Y SALIDAS DE UN ESP32

esp32		
Digital en	17, 18, 19, 21, 22, 23, 27, 32 33	%IX0.0 - %IX0.7 %IX1.0 - %IX1.0
Salida digital	01, 02, 03, 04, 05, 12, 13, 14, 15, 16	%QX0.0 - %QX0.7 %QX1.0 - %QX1.1
Entrada analógica	34, 35, 36, 39	%IW0 - %IW3
Salida analógica	25, 26	%QW0 - %QW1



OpenPLC Editor - Contador

Archivo Editar Mostrar Ayuda

program0 x

Descripción: Filtro de clase: Todo

#	Nombre	Clase	Tipo	Ubicación	Valor Inicial	Opción	Documentación
1	ON1	Local	BOOL	%IX0.0			GPIO17
2	Reset	Local	BOOL	%IX0.1			GPIO18
3	K1	Local	BOOL	%QX0.0			GPIO0.2
4	CTU0	Local	CTU				
5	Contador1	Externa	INT				

Proyecto

- Sin nombre
 - program0
 - Res0
 - 0x: arduino_ext_0

Config0.Res0.instance0

- ON1 (BOOL)
- Reset (BOOL)
- K1 (BOOL)
- CTU0 (CTU)
- Contador1 (INT)

Librería Depurador

Buscar

- Bloques de función estándar
- Bloques adicionales
- Arduino
- Microver
- Communication
- P1AM Modules
- MQTT
- Sequent Microsystems Modules
- Jaguar
- SL-RP4
- Conversión de Tipos
- Numéricos
- Aritméticos
- Tiempo
- Intercambio de bits
- A nivel de bits
- Selección
- Comparación
- Cadena de caracteres

Buscar Consola Registro del PLC



Archivo Editar Mostrar Ayuda

program0 arduino_ext_0 ...s0.instance0 x

Proyecto

- Sin nombre
 - program0
 - Res0
 - 0.x: arduino_ext_0

Config0.Res0.instance0

- ON1 (BOOL)
- Reset (BOOL)
- K1 (BOOL)
- CTU0 (CTU)
- Contador1 (INT)

Descripción: Filtro de clase: Todo

#	Nombre	Clase	Tipo	Ubicación	Valor Inicial	Opción	Documentación
1	ON1	Local	BOOL	%IX0.0			GPIO17
2	Reset	Local	BOOL	%IX0.1			GPIO18
3	K1	Local	BOOL	%QX0.0			GPIO0.2
4	CTU0	Local	CTU				
5	Contador1	Externa	INT				

Depuración: Config0.Res0.instance0

Depuración: 1s

Tick: 3993 GO.RES0.INSTANCE t: 1m19s860ms

..ON1 True

Buscar Consola Registro del PLC

Todo Q Buscar

+1d +1h +1m +1s 30/10/25 12:42 -1s -1m -1h -1d



Project Explorer: Sin nombre, program0, Res0, 0x: arduino_ext_0

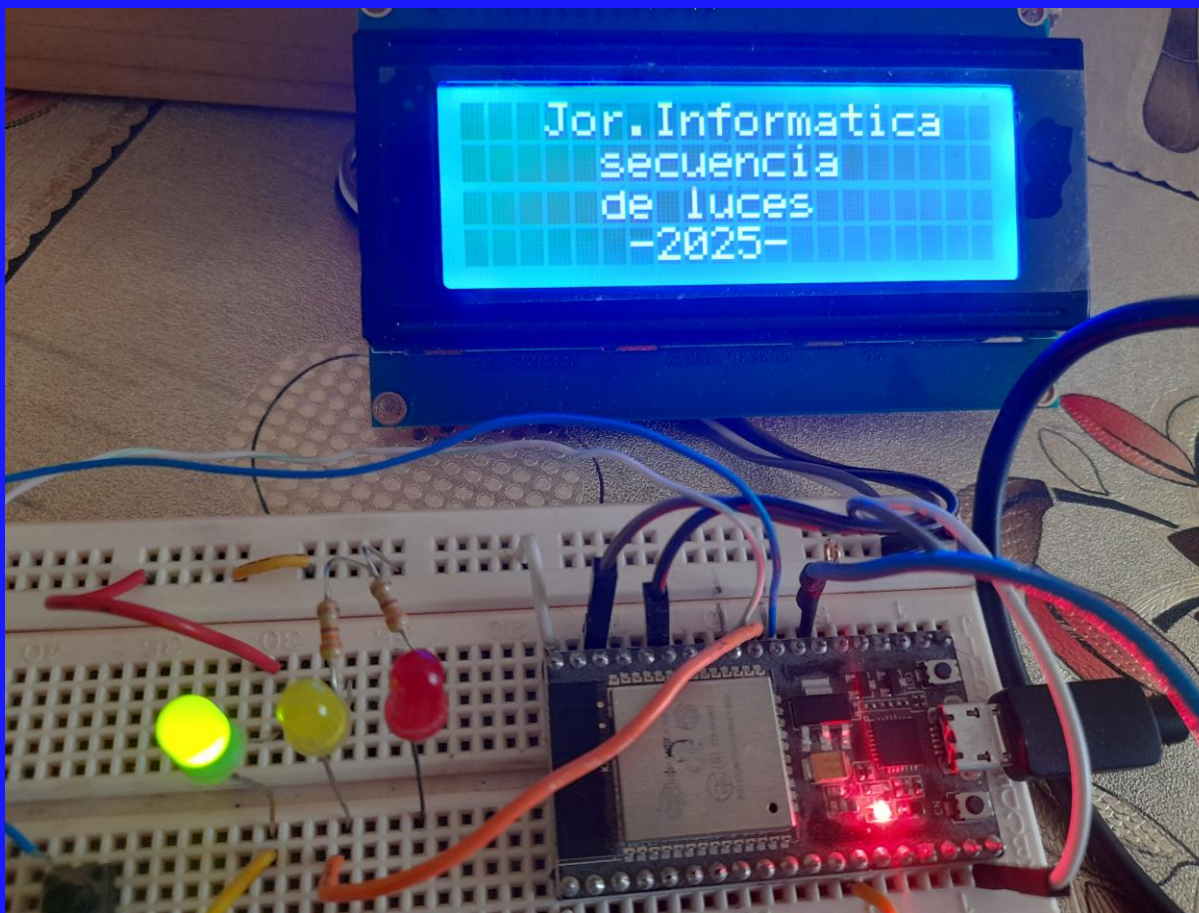
#	Nombre	Tipo	Inicial	Descripción	OnChange	Opc
1	Contador1	INT	0			

```
10 {
11 lcd.init ();
12 lcd.backlight ();
13 }
14
15 void sketch_loop()
16 {
17 lcd.setCursor (5,0);
18 lcd.print ("NJControl");
19 lcd.setCursor (0,1);
20 lcd.print ("contador=");
21 lcd.print (Contador1);
22 lcd.setCursor (0,2);
23 lcd.print ("temperatura:");
24 lcd.setCursor (0,3);
25 lcd.print ("humedad:");
26 lcd.print (" ");
```

Librería Depurador

- Buscar
- Bloques de función estandar
- Bloques adicionales
- Arduino
- Microver
- Communication
- P1AM Modules
- MQTT
- Sequent Microsystems Modules
- Jaguar
- SL-RP4
- Conversión de Tipos
- Numéricos
- Aritméticos
- Tiempo
- Intercambio de bits
- A nivel de bits
- Selección
- Comparación
- Cadena de caracteres

Buscar Consola Registro del PLC





OpenPLC Editor - semáforo

Archivo Editar Mostrar Ayuda

Proyecto

- Sin nombre
 - program0
 - Res0

Config0.Res0.instance0

- Rojo (BOOL)
- Amarillo (BOOL)
- Verde (BOOL)
- M1 (BOOL)
- M2 (BOOL)
- M3 (BOOL)
- TON0 (TON)

program0 x

Descripción: Filtro de clase: Todo

#	Nombre	Clase	Tipo	Ubicación	Valor Inicial	Opción	Documentación
1	Rojo	Local	BOOL	%QX0.0			
2	Amarillo	Local	BOOL	%QX0.1			
3	Verde	Local	BOOL	%QX0.2			
4	M1	Local	BOOL				
5	M2	Local	BOOL				

Diagrama de Ladder Logic:

```

  graph TD
    M1 --- L1(( ))
    M3 --- L1
    L1 --- TON0[TON0]
    TON0 --- M1
    TON0 --- M1
    TON0 --- M1
    M1 --- L2(( ))
    M2 --- L2
    L2 --- TON1[TON1]
    TON1 --- M2
    TON1 --- M2
    TON1 --- M2
    M2 --- L3(( ))
    M1 --- L3
    L3 --- Verde[Verde]
  
```

Librería Depurador

- Bloques de función estándar
- Bloques adicionales
- Arduino
- Microcover
- Communication
- PIAM Modules
- MQTT
- Sequent Microsystems Modules
- Jaguar
- SL-RP4
- Conversión de Tipos
- Núméricos
- Aritméticos
- Tiempo
- Intercambio de bits
- A nivel de bits
- Selección
- Comparación
- Cadena de caracteres
- Native POU
- POUs definidos por el usuario

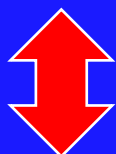
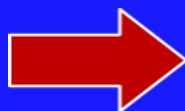
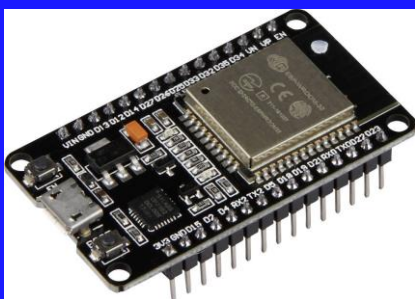
Buscar Consola Registro del PLC



CONECTIVIDAD -ESP32 MCU - DHT 22



CONECTIVIDAD -ESP32 MCU - DHT 22



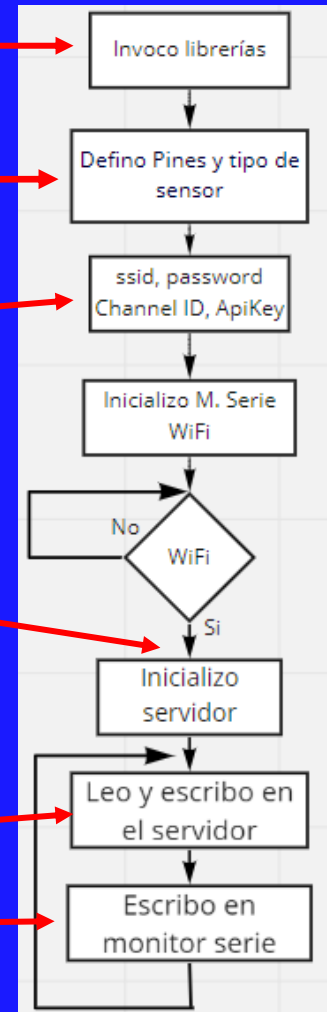
Sensor DHT22



```

1 #include "ThingSpeak.h" // agregado para comunicación thingspeak
2 #include "WiFi.h" // agregado para comunicación thingspeak
3 #include "DHT.h"
4
5
6 #define DHTPIN 4 // defino el pin donde estará conectado el
7 #define DHTTYPE DHT22 // defino el tipo de sensor
8
9
10 const char* ssid = "xxxxxxxxx"; // agregado para comunicac
11 const char* password = "xxxxxxxxx"; // agregado para comuni
12
13 unsigned long channelID = 144965; // agregado para comun
14 const char* WriteAPIKey = "1ZLADKSNDS30OU"; // agregado para con
15

```



```

.....WIFI CONECTADO!
Temperatura DHT22: 23.00 °C.
Humedad DHT22: 36.10 %.
-----
datos enviados a Write APIKey
Temperatura DHT22: 23.10 °C.
Humedad DHT22: 35.90 %.
-----
datos enviados a Write APIKey
Temperatura DHT22: 23.50 °C.
Humedad DHT22: 35.90 %.
-----
datos enviados a Write APIKey

```



ThingSpeak™ Canales Aplicaciones Apoyo ▾

Uso comercial Cómo comprar

ThingSpeak para proyectos de IoT

Recopilación de datos en la nube con análisis de datos avanzado usando MATLAB

Empiece gratis Aprende más

ThingSpeak para estudiantes y educadores

Implemente proyectos de investigación de IoT rápidamente con herramientas integradas de análisis de datos de MATLAB y recopilación de datos de sensores en tiempo real

Acerca de ThingSpeak

ThingSpeak es un servicio de plataforma de análisis de IoT que le permite agregar, visualizar y analizar flujos de datos en vivo en la nube. Puede enviar datos a ThingSpeak desde sus dispositivos, crear una visualización instantánea de datos en vivo y enviar alertas.

Recoger

Envíe los datos del sensor de forma privada a la nube.

Analizar

Analice y visualice sus datos con MATLAB.

actuar

Desencadena una reacción.



To use ThingSpeak, you must sign in with your existing MathWorks account or create a new one.

Non-commercial users may use ThingSpeak for free. Free accounts offer limits on certain functionality. Commercial users are eligible for a time-limited free evaluation. To get full access to the MATLAB analysis features on ThingSpeak, log in to ThingSpeak using the email address associated with your university or organization.

To send data faster to ThingSpeak or to send more data from more devices, consider the [paid license options](#) for commercial, academic, home and student usage.

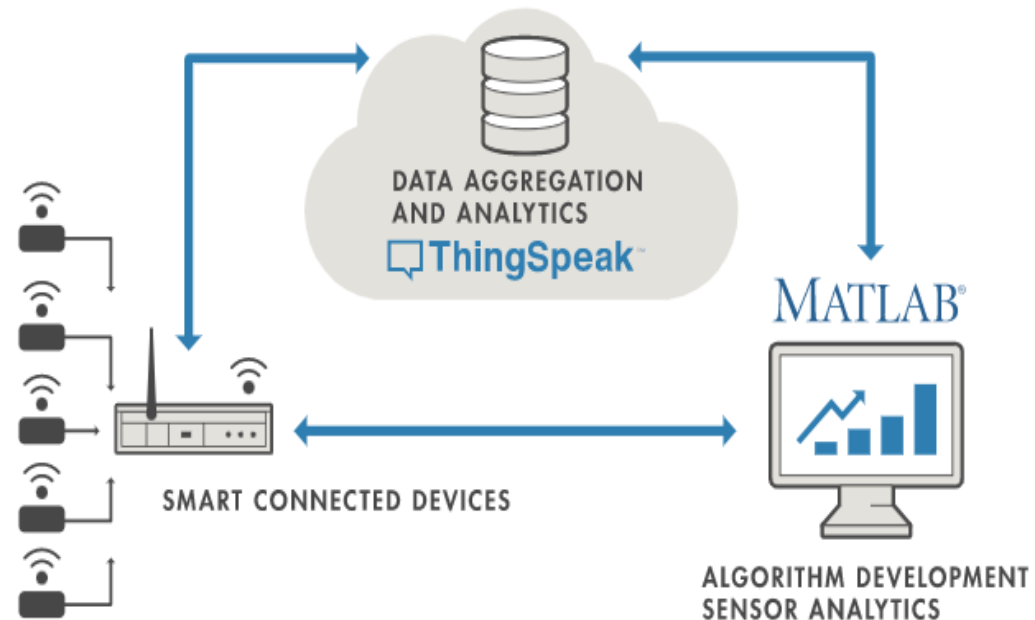


Email

No account? [Create one!](#)

By signing in you agree to our [privacy policy](#).

Next





Author: mwa0000023124537
Access: Private

Private View Public View Channel Settings Sharing API Keys Data Import / Export

+ Add Visualizations

+ Add Widgets

Export recent data

MATLAB Analysis

MATLAB Visualization

Channel Stats

Created: [3 months ago](#)
Last entry: [a day ago](#)

ThingSpeak™

Channels ▾

Apps ▾

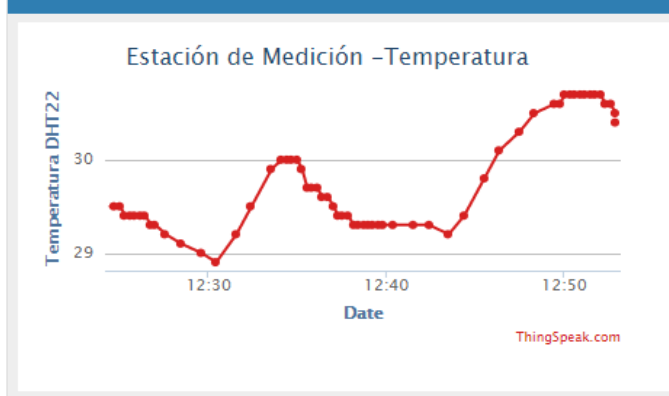
Devices ▾

Support ▾

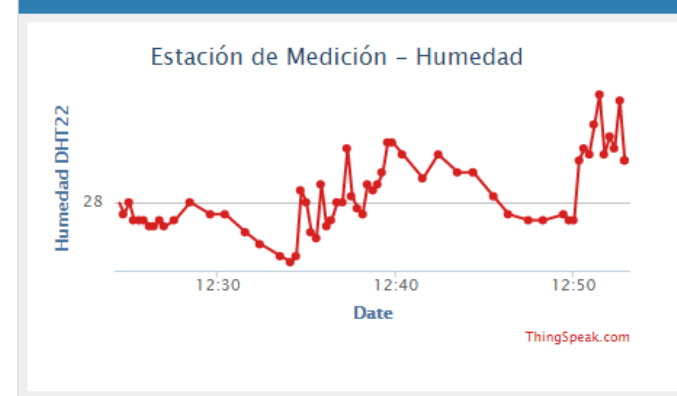
Commercial Use

How to Buy

NC



Temperatura DHT22

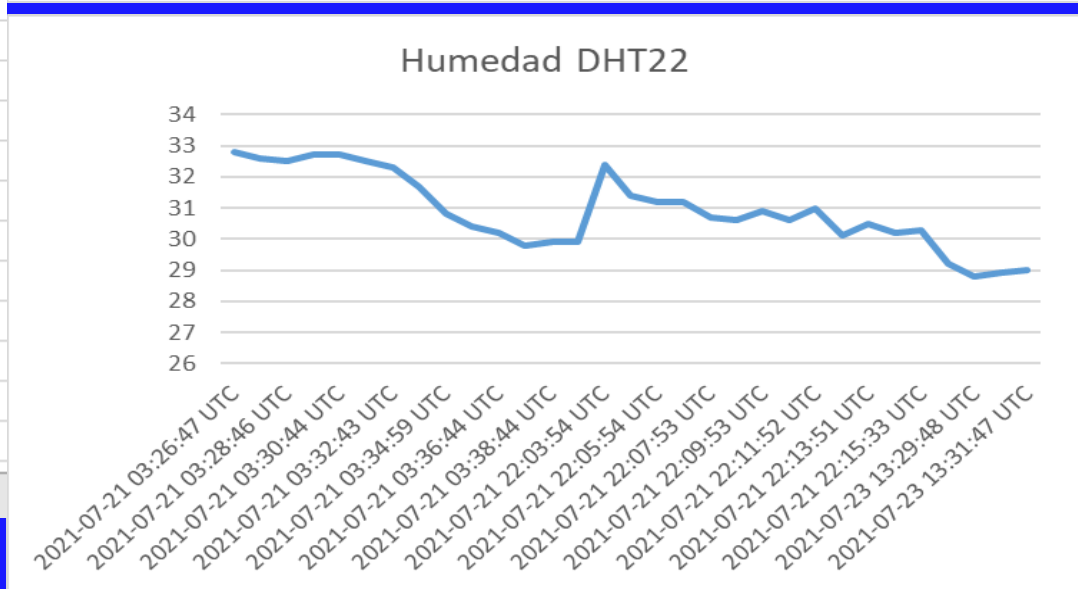
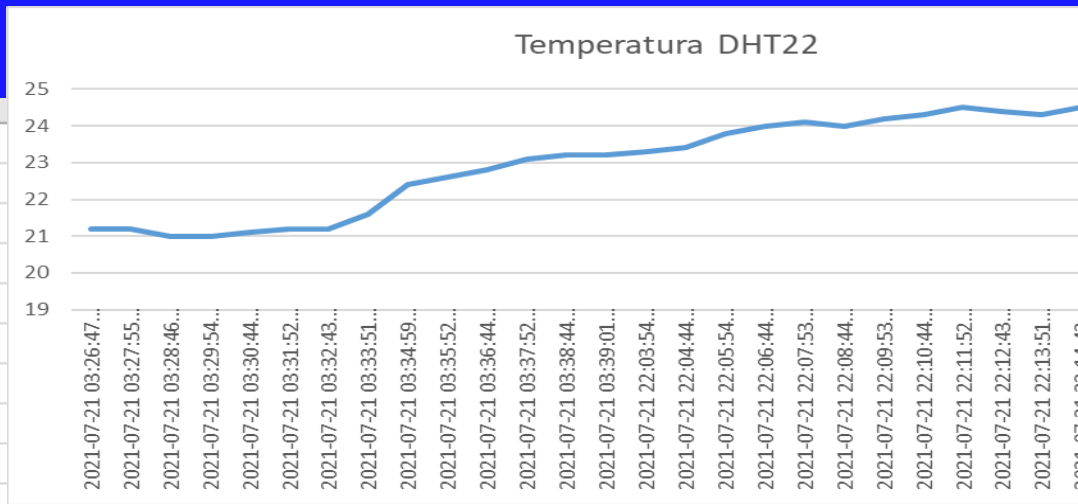


Humedad





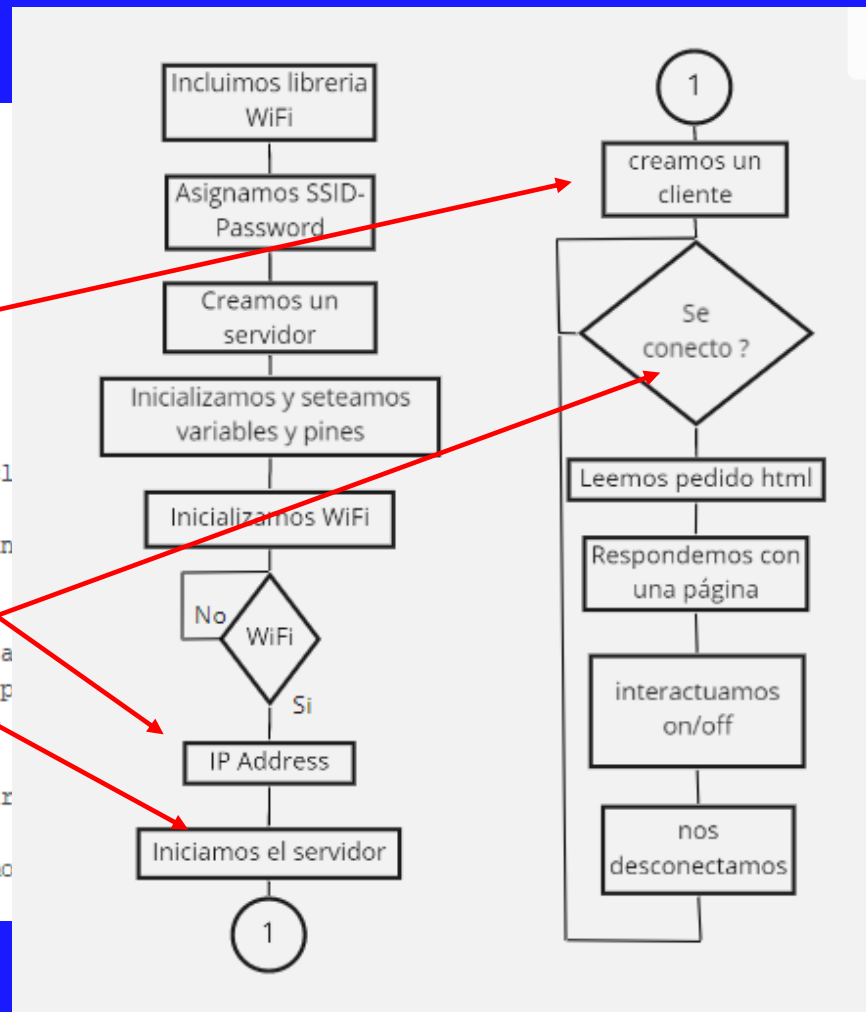
	A	B	C
1	created_at	field1	field2
2			
3	2021-07-21 03:26:47 UTC	21.2	32.8
4	2021-07-21 03:27:55 UTC	21.2	32.6
5	2021-07-21 03:28:46 UTC	21	32.5
6	2021-07-21 03:29:54 UTC	21	32.7
7	2021-07-21 03:30:44 UTC	21.1	32.7
8	2021-07-21 03:31:52 UTC	21.2	32.5
9	2021-07-21 03:32:43 UTC	21.2	32.3
10	2021-07-21 03:33:51 UTC	21.6	31.7
11	2021-07-21 03:34:59 UTC	22.4	30.8
12	2021-07-21 03:35:52 UTC	22.6	30.4
13	2021-07-21 03:36:44 UTC	22.8	30.2
14	2021-07-21 03:37:52 UTC	23.1	29.8
15	2021-07-21 03:38:44 UTC	23.2	29.9
16	2021-07-21 03:39:01 UTC	23.2	29.9
17	2021-07-21 22:03:54 UTC	23.3	32.4
18	2021-07-21 22:04:44 UTC	23.4	31.4
19	2021-07-21 22:05:54 UTC	23.8	31.2
20	2021-07-21 22:06:44 UTC	24	31.2
21	2021-07-21 22:07:53 UTC	24.1	30.7



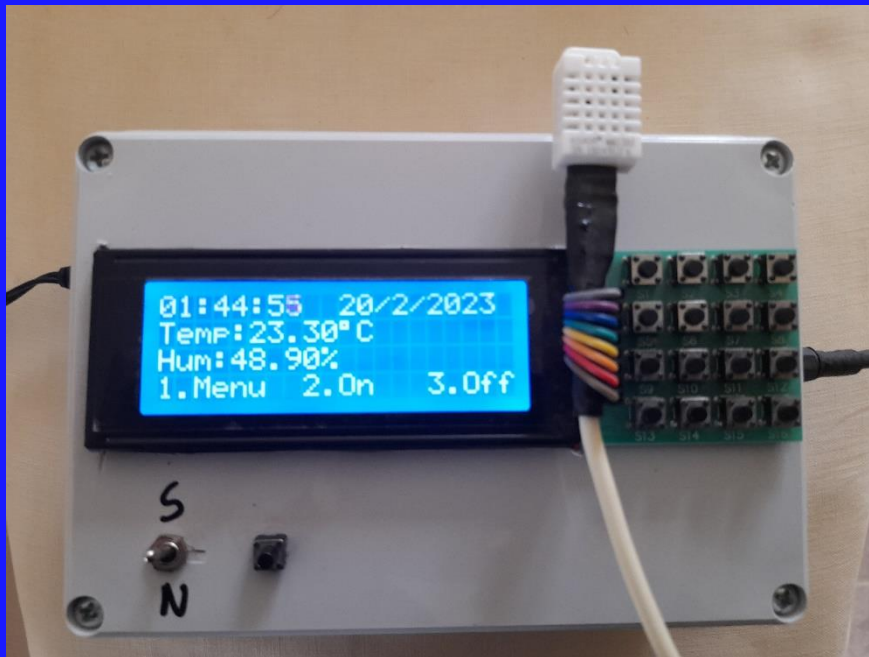
```

50 }
51 // Imprima la dirección IP y active el servidor |
52 Serial.println("");
53 Serial.println("WiFi connected.");
54 Serial.println("IP address: ");
55 Serial.println(WiFi.localIP());
56 server.begin();
57 }
58
59 void loop(){
60   WiFiClient client = server.available(); // Escucha a los cl
61
62   if (client) { // si se conecta un
63     currentTime = millis();
64     previousTime = currentTime;
65     Serial.println("New Client."); // escribe un mensa
66     String currentLine = ""; //Hace una cadena p
67     while (client.connected() && currentTime - previousTime <= 5000) {
68       currentTime = millis();
69       if (client.available()) { // Si hay datos par
70         char c = client.read(); // lea un byte
71         Serial.write(c); // imprima en el mo
72         header += c;

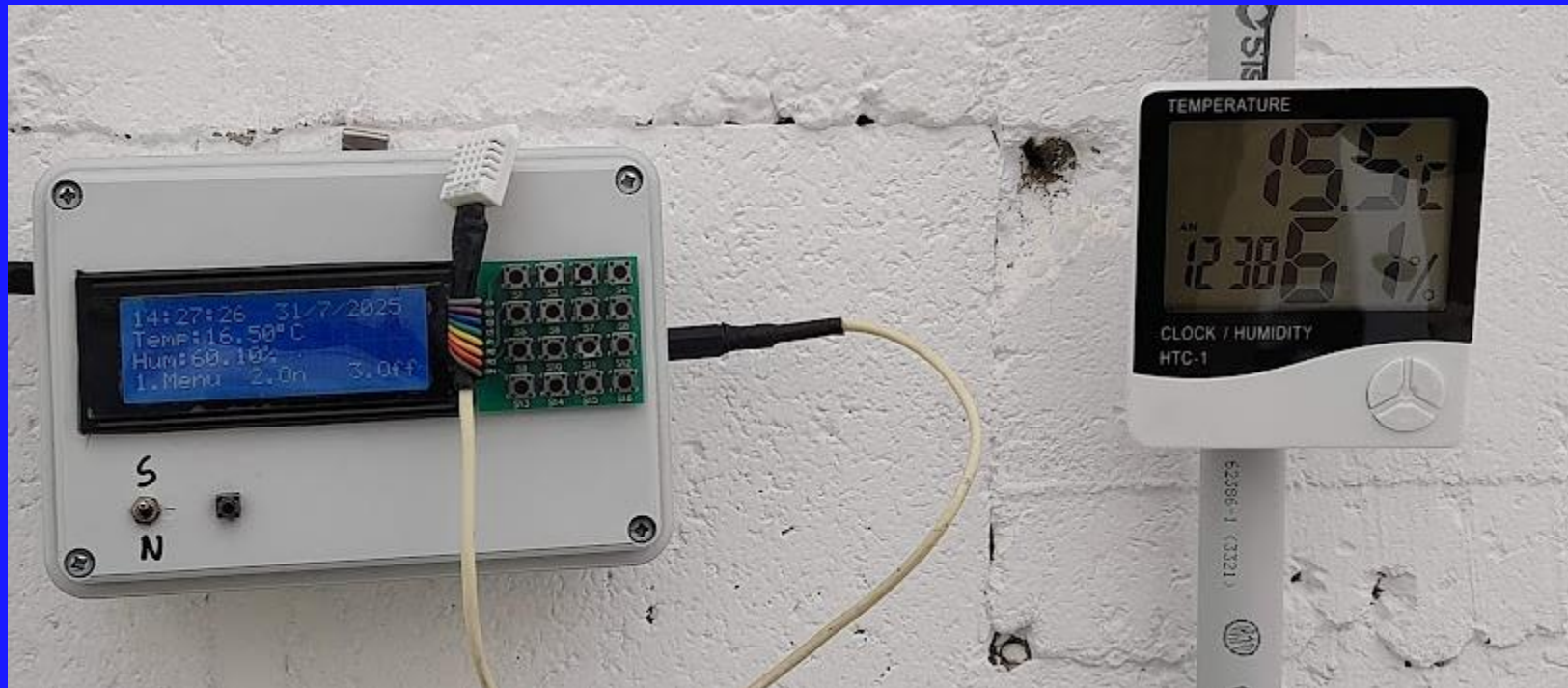
```



Módulo adquiredor de datos de temperatura y humedad ambiente









```
03-10-25: Bloc de notas
Archivo Edición Formato Ver Ayuda
Reading ID,Date,Time,t,h,
0,2025-09-16,11:22:01,31.50,24.30
1,2025-09-16,11:37:02,32.20,22.20
2,2025-09-16,11:52:02,32.90,21.70
3,2025-09-16,12:07:03,34.90,20.30
4,2025-09-16,12:22:03,34.10,19.60
5,2025-09-16,12:37:04,28.10,21.20
6,2025-09-16,12:52:04,29.20,20.60
7,2025-09-16,13:07:05,21.90,25.50
8,2025-09-16,13:22:06,25.40,24.40
9,2025-09-16,13:37:06,29.20,22.20
10,2025-09-16,13:52:07,22.20,27.10
11,2025-09-16,14:07:07,24.00,25.30
12,2025-09-16,14:22:08,23.80,24.60
13,2025-09-16,14:37:08,22.60,27.90
14,2025-09-16,14:52:09,18.70,32.90
15,2025-09-16,15:07:10,26.50,29.70
16,2025-09-16,15:22:10,31.70,26.90
17,2025-09-16,15:37:11,27.10,29.20
18,2025-09-16,15:52:11,32.10,27.00
19,2025-09-16,16:07:12,28.60,30.20
20,2025-09-16,16:22:12,24.60,33.40
21,2025-09-16,16:37:13,30.90,28.60
22,2025-09-16,16:52:14,30.10,27.20
23,2025-09-16,17:07:14,30.00,27.50
```

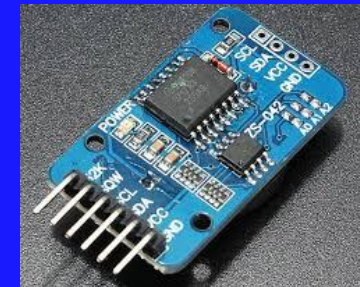
Reading ID	Date	Time	t	h
0	16/09/2025	11:22:01	31.50	24.30
1	16/09/2025	11:37:02	32.20	22.20
2	16/09/2025	11:52:02	32.90	21.70
3	16/09/2025	12:07:03	34.90	20.30
4	16/09/2025	12:22:03	34.10	19.60
5	16/09/2025	12:37:04	28.10	21.20
6	16/09/2025	12:52:04	29.20	20.60
7	16/09/2025	13:07:05	21.90	25.50
8	16/09/2025	13:22:06	25.40	24.40
9	16/09/2025	13:37:06	29.20	22.20
10	16/09/2025	13:52:07	22.20	27.10
11	16/09/2025	14:07:07	24.00	25.30
12	16/09/2025	14:22:08	23.80	24.60
13	16/09/2025	14:37:08	22.60	27.90
14	16/09/2025	14:52:09	18.70	32.90
15	16/09/2025	15:07:10	26.50	29.70
16	16/09/2025	15:22:10	31.70	26.90
17	16/09/2025	15:37:11	27.10	29.20
18	16/09/2025	15:52:11	32.10	27.00
19	16/09/2025	16:07:12	28.60	30.20
20	16/09/2025	16:22:12	24.60	33.40
21	16/09/2025	16:37:13	30.90	28.60

Prototipo adquirente de datos de Temperatura y Humedad de tierra rtc

Sensor de temperatura



ESP WROOM 32



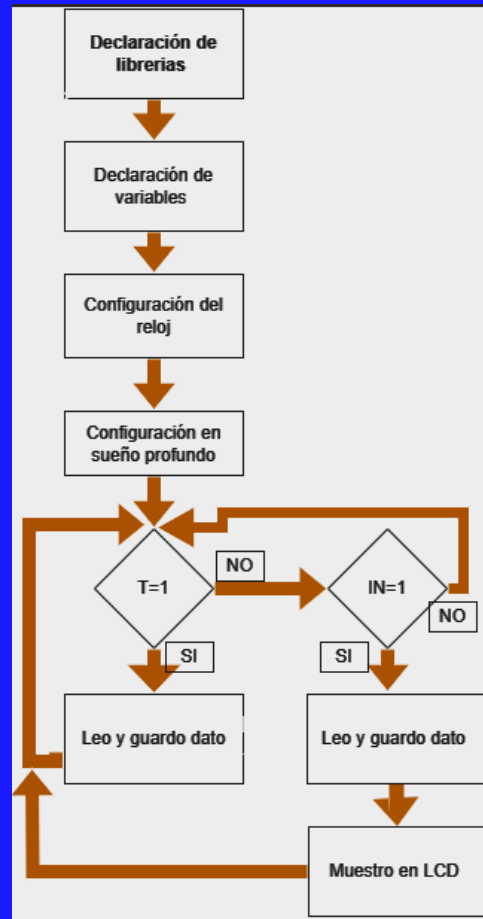
Memoria SD

Sensor de humedad





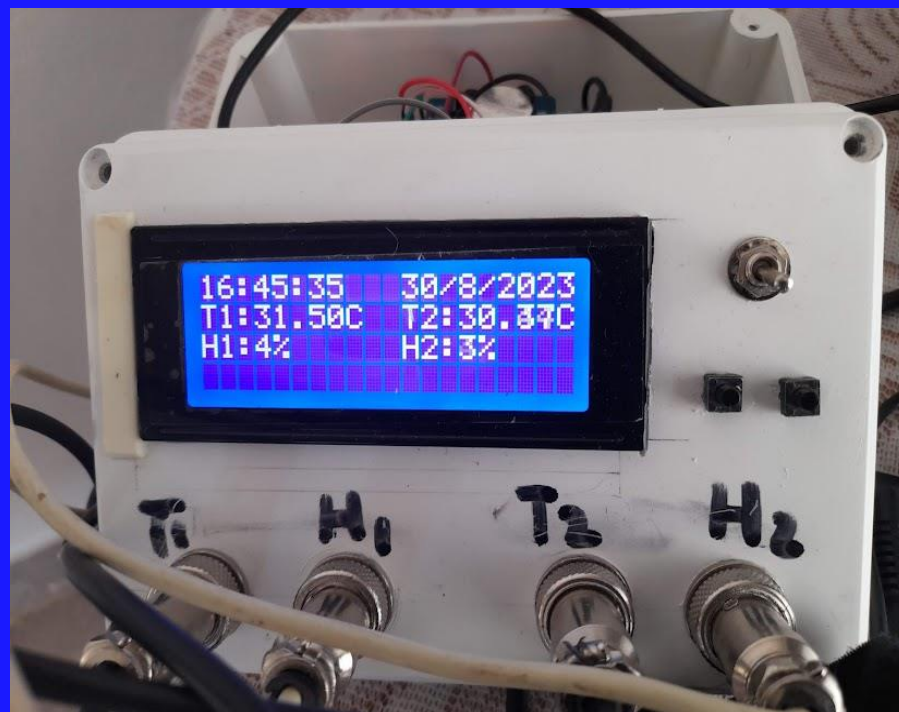
Prototipo adquirente de datos de Temperatura y Humedad de tierra - Diagrama de Flujo



Parte del programa residente en el esp32

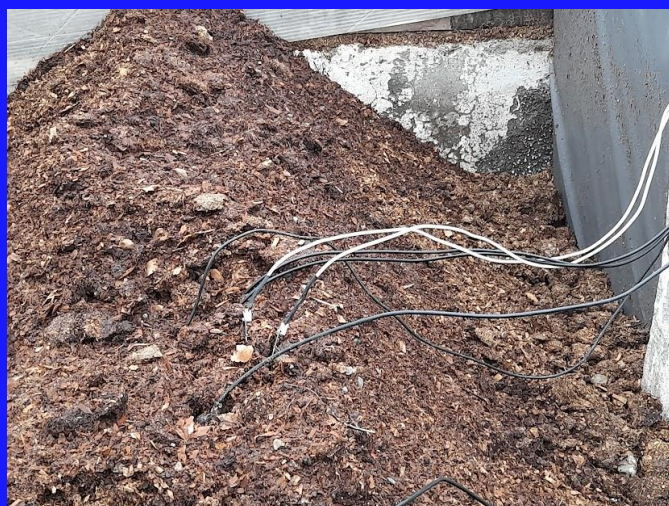
```
95 lcd.setCursor(10,0); //adopto la posi
96 lcd.print(now.day());
97 lcd.print("/");
98 lcd.print(now.month());
99 lcd.print("/");
00 lcd.print(now.year());
01
02 //-----Temperaturas
03 sensors1.requestTemperatures(); //Se
04 temp1= sensors1.getTempCByIndex(0);
05 lcd.setCursor(0,1); //primera columna-
06 lcd.print("T1:");
07 lcd.print(temp1);
08 lcd.print("C");
09
10 sensors2.requestTemperatures(); //Se
11 temp2= sensors2.getTempCByIndex(0);
12 lcd.setCursor(11,1); //12 columna - seg
13 lcd.print("T2:");
14 lcd.print(temp2);
15 lcd.print("C");
16
```

Prototipo adquirente de datos de Temperatura y Humedad de tierra - Calibración





Adquisidor de datos de Temperatura y Humedad de tierra Mediciones en campo





Adquisidor de datos de Temperatura y Humedad de tierra

Datos adquiridos en la memoria SD

```

Reading ID,Date,Time,Temp1,Temp2,Hum1,Hum2,
0,2023-09-16,12:51:54,19.12,19.87,1,0
1,2023-09-16,12:52:00,19.00,19.81,0,0
2,2023-09-16,13:07:24,15.50,16.12,1,0
3,2023-09-16,13:22:22,15.31,16.12,1,0
4,2023-09-16,13:37:21,15.50,16.12,1,0
5,2023-09-16,13:52:19,15.69,16.31,1,0
6,2023-09-16,14:07:18,15.50,16.19,1,0
7,2023-09-16,14:22:16,15.63,16.25,1,0
8,2023-09-16,14:37:15,15.81,16.44,1,0
9,2023-09-16,14:52:14,15.75,16.50,1,0
10,2023-09-16,15:07:14,15.81,16.50,1,0
11,2023-09-16,15:22:13,15.94,16.56,1,0
12,2023-09-16,15:22:39,15.88,16.62,1,0
13,2023-09-16,15:38:02,17.19,18.12,0,0
14,2023-09-16,15:53:01,17.69,18.75,0,0
15,2023-09-16,16:08:00,17.94,19.06,0,0
16,2023-09-16,16:22:59,18.00,19.31,0,0
17,2023-09-16,16:37:59,18.12,19.37,0,0
18,2023-09-16,16:52:58,18.37,19.62,0,0
19,2023-09-16,17:07:57,18.00,19.56,0,0
20,2023-09-16,17:22:55,18.00,19.62,0,0
21,2023-09-16,17:37:54,18.06,19.69,0,0
22,2023-09-16,17:52:52,17.94,19.62,0,0
23,2023-09-16,18:07:51,17.87,19.56,0,0

```

Datos adquiridos y procesados en Excel

3	27/9/2023	10:45:28	13.75	11.81	39	34
4	27/9/2023	11:00:50	14.25	12.13	38	33
5	27/9/2023	11:15:48	14.44	12.31	37	30
6	27/9/2023	11:30:47	14.50	12.56	39	31
7	27/9/2023	11:45:46	14.69	12.69	36	29
8	27/9/2023	12:00:46	14.81	12.94	37	31
9	27/9/2023	12:15:45	15.06	13.13	36	30
10	27/9/2023	12:30:44	15.19	13.31	36	30
11	27/9/2023	12:45:43	15.31	13.56	38	32
12	27/9/2023	13:00:42	15.50	13.69	37	32
13	27/9/2023	13:15:41	15.75	13.94	37	32
14	27/9/2023	13:30:40	15.88	14.06	40	35
15	27/9/2023	13:45:39	16.12	14.31	40	33
16	27/9/2023	14:00:38	16.19	14.44	42	36
17	27/9/2023	14:15:37	16.31	14.50	39	32
18	27/9/2023	14:30:36	16.37	14.56	39	32
19	27/9/2023	14:45:34	16.44	14.56	38	32
20	27/9/2023	15:00:33	16.56	14.63	42	36
21	27/9/2023	15:15:33	16.62	14.69	38	30
22	27/9/2023	15:30:31	16.75	14.75	37	30
23	27/9/2023	15:45:30	16.81	14.81	35	29
24	27/9/2023	16:00:29	16.94	14.75	37	29



DATOS ALOJADOS DIRECTAMENTE EN EL EXCEL DEL GOOGLE DRIVE

Estos son algunos de los **beneficios de usar Hojas de cálculo de Google para aplicaciones de IoT** :

- El registro de datos es bastante simple y robusto, y no necesita servicios de terceros.
- Fácil manipulación y análisis de los datos recopilados mediante funciones.
- Admite acceso tanto desde ordenador como desde dispositivos móviles.
- Funciones de hojas de cálculo personalizadas fáciles de usar e integración con Google Apps mediante Google Scripts.
- El formato condicional facilitará enormemente la monitorización y el análisis de datos.

- Crear y configurar una hoja de cálculo en Google drive
- Crear un script de Google
- Extensiones
- Nueva implementación



Parte del programa residente en el microcontrolador

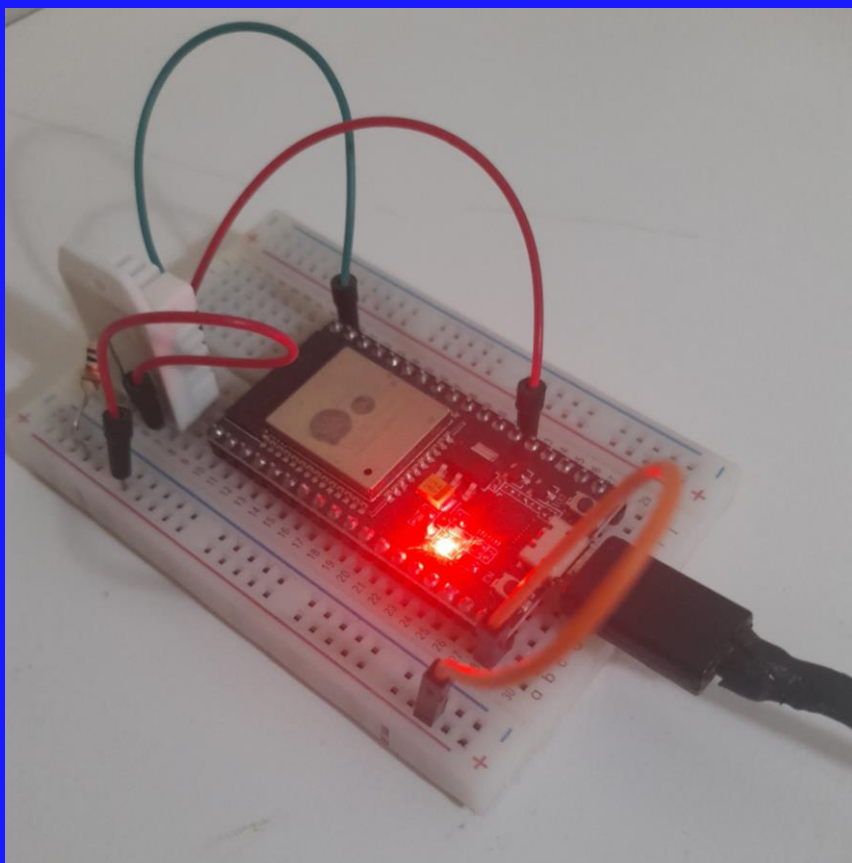
Parte del script alojado en la hoja de cálculo

```
var sheet_name = "NOMBRE DE SU HOJA";  
función doGet(e){  
var ss = SpreadsheetApp.openById(sheet_id);  
var hoja = ss.getSheetByName(nombre_hoja);  
sensor var = Número (e.parámetro.sensor);  
var fecha = Number(e.parameter.date);  
hoja.appendRow([sensor, fecha]);  
}
```

```
#include <HTTPClient.h>  
#incluir "time.h"  
const char* ntpServer = "pool.ntp.org";  
constante larga gmtoffset_sec = 19800;  
constante int daylightOffset_sec = 0;  
// Credenciales de WiFi  
const char* ssid = "SKYNET 4G"; // cambiar SSID  
const char* contraseña = "jobitjos"; // cambiar contraseña  
// ID de script de Google y credenciales necesarias  
String GOOGLE_SCRIPT_ID = "AKfycby-snbh-5j0jsiqBwfc-XB1FWy38lks4VHcxLBIGNadeCVcSzUoozHz";  
contador entero = 0;  
void setup() {  
  retraso(1000);  
  Serial.begin(115200);  
  retraso(1000);  
  // Conéctate a la red WiFi  
  Serial.println();  
}
```



DATOS ALOJADOS DIRECTAMENTE EN EL EXCEL DEL GOOGLE DRIVE



	A	B	C
1		TEMP °C	HUM %
1095	31/10/2025 4:59:05	27.10	31.00
1096	31/10/2025 5:14:10	26.80	30.50
1097	31/10/2025 5:29:14	26.90	30.60
1098	31/10/2025 5:44:19	27.00	30.60
1099	31/10/2025 5:59:24	26.90	30.50
1100	31/10/2025 6:14:29	26.80	30.40
1101	31/10/2025 6:29:34	26.90	30.30
1102	31/10/2025 6:44:38	26.80	30.30
1103	31/10/2025 6:59:43	26.90	30.30
1104	31/10/2025 7:14:47	27.00	30.00
1105	31/10/2025 7:29:52	27.00	30.10
1106	31/10/2025 7:44:56	27.20	30.10
1107	31/10/2025 8:00:01	27.10	30.00
1108	31/10/2025 8:40:06	27.10	30.10
1109	31/10/2025 8:55:11	26.80	30.50
1110	31/10/2025 9:10:16	26.80	30.40
1111	31/10/2025 9:25:21	26.80	30.10
1112	31/10/2025 9:40:26	27.00	30.00



APLICACIONES EN LA DOCENCIA

Aplicaciones en Docencia

LEY FUNDAMENTAL DE LA HIDROSTÁTICA

Presión absoluta
 $P = P_0 + \rho * g * h$

Presión manométrica
 $P - P_0 = \rho * g * h$

ECUACIÓN DE CONTINUIDAD

$$Q = A_1 * V_1 = A_2 * V_2$$

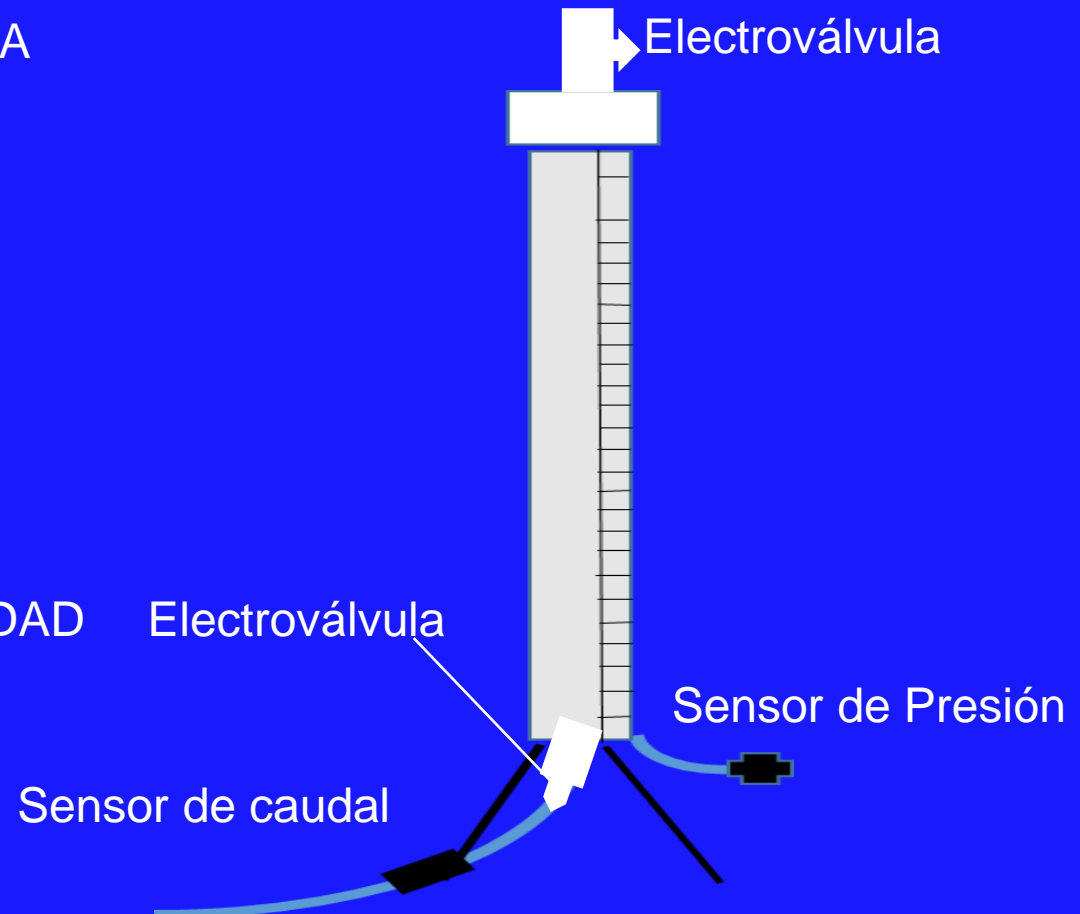
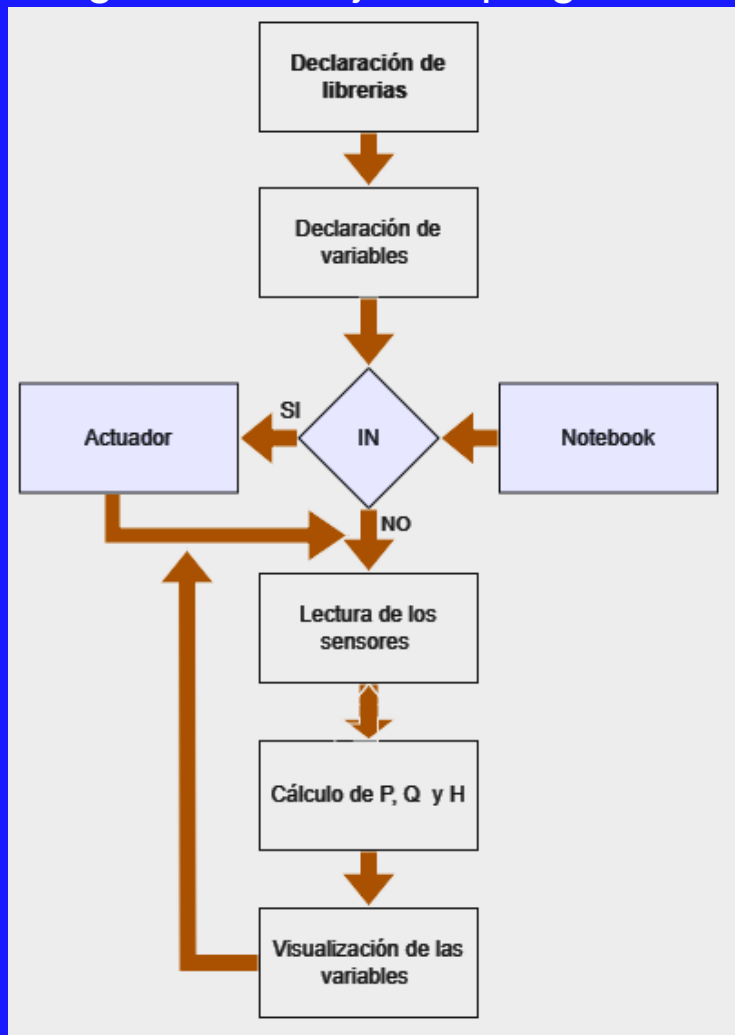


Diagrama de flujo del programa



Sensor de Presión



Display 20x4



Sensor de Caudal





Parte del programa que reside en el ESP32

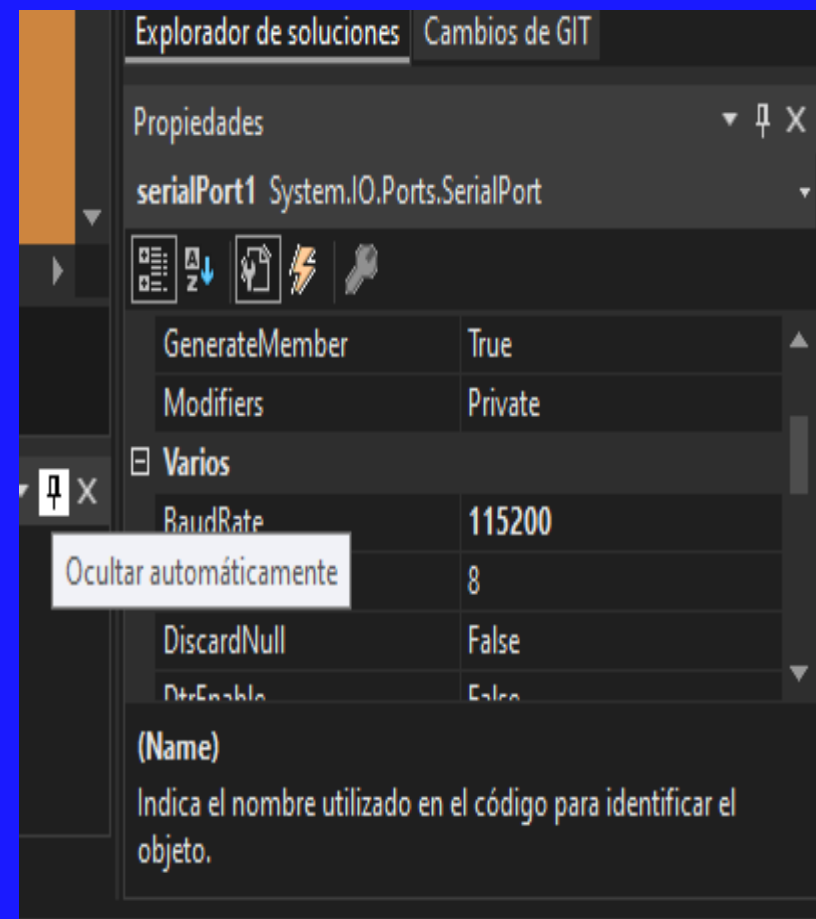
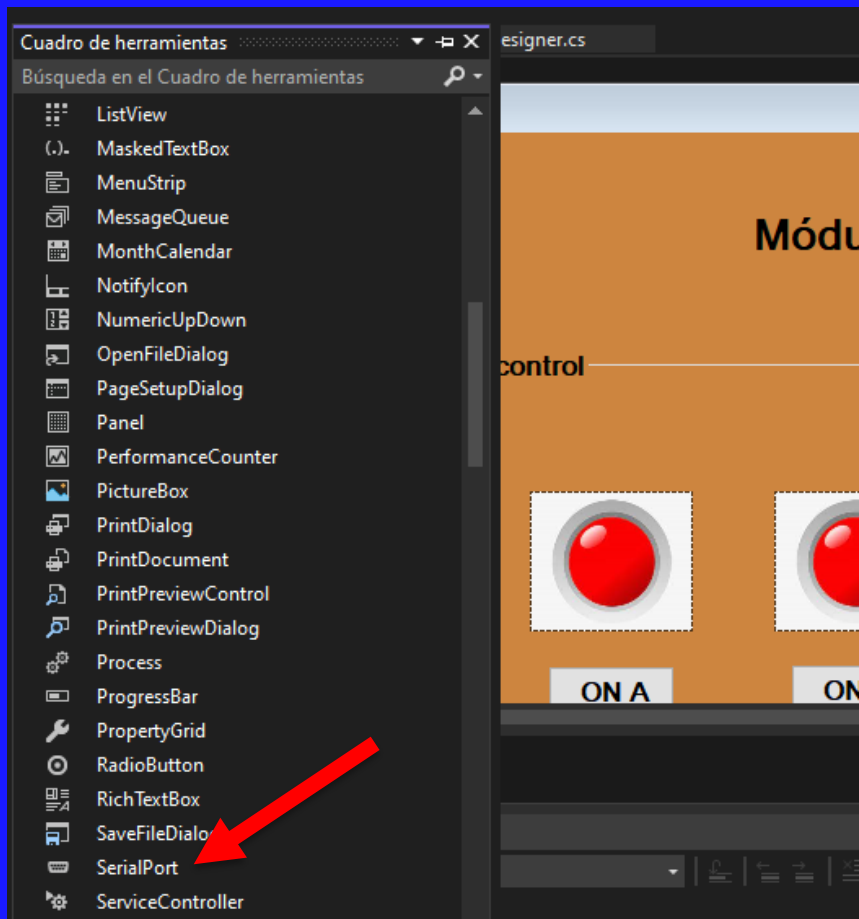
```
51 Serial.print("AD = ");
52 Serial.println(promedio); // promedio de todas la
53
54
55
56 lcd.setCursor(0,0); //
57 lcd.print("Presion Manometrica ");
58
59 lcd.setCursor(0,1); //
60 lcd.print (" P= ");
61 lcd.print(manoMetrica);
62 lcd.print(" Kpa ");
63
64 lcd.setCursor(0,2); //
65 lcd.print ("Altura H=");
66 lcd.print (Altura);
67 lcd.print (" m ");
68
69 lcd.setCursor (0,3);
70 lcd.print ("V=");
71 lcd.print( promedio );
72 lcd.print(" Volts ");
73
```

Parte del programa en VSC

```
Form1.cs [Diseño]
Form1.Designer.cs
HMTresleds
HMTresleds.Controldeleds
pictureBoxLedA.Image = Properties.Resources.led3;
serialPort1.WriteLine("$Aon");
catch (Exception error)
{
    MessageBox.Show(error.Message);
}
private void buttonOffA_Click(object sender, EventArgs e)
{
    if (serialPort1.IsOpen)
    {
        try
        {
            pictureBoxLedA.Image = Properties.Resources.led2;
            serialPort1.WriteLine("$Aoff");
        }
        catch (Exception error)
        {
            MessageBox.Show(error.Message);
        }
    }
}
```



CUADRO DE HERRAMIENTAS Y CONFIGURACIÓN

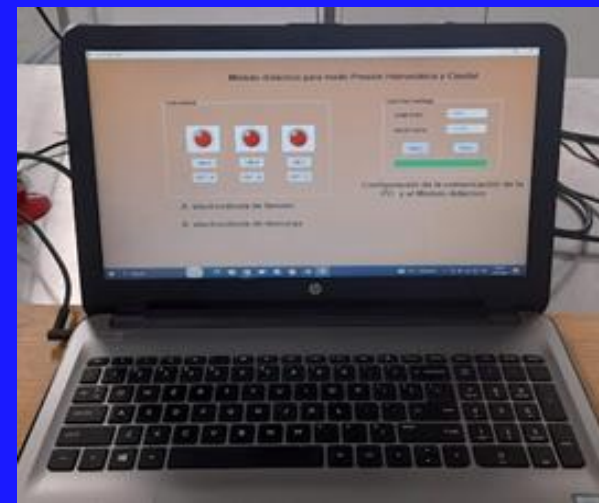
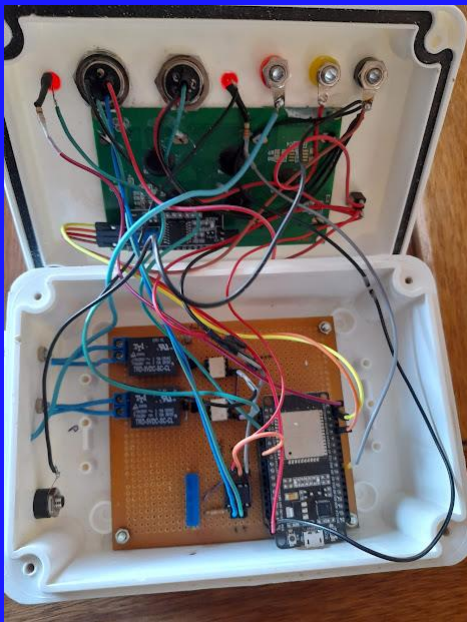




Entorno de desarrollo del VSC



MÓDULO DIDÁCTICO PARA MEDIR PRESIÓN, CAUDAL DE SALIDA Y ALTURA DE UNA COLUMNA DE AGUA



Fe de erratas: kPa y l/min



MÓDULO DIDÁCTICO PARA MEDIR PRESIÓN, CAUDAL DE SALIDA Y ALTURA DE UNA COLUMNA DE AGUA





Alumnos Industrial N°4 - 2025



Alumnos UNPA UARG – Física I - 2025





MUCHAS GRACIAS
POR SU ATENCIÓN