

Instrument virtual de monitorizare a unor parametri din mediul ambiant

Proiect IMPM

Curpina:

1.Introducere in sisteme IoT.....	2
2.Scurta descriere a componentelor utilizate si prezentarea diagramei hardware	3
3.Software pentru achizitie date rulat pe ESP32	3
4.Aplicatia de monitorizare realizata in Labview	15
5.Exemple de functionare	18
6.Anexe	21
7.Bibliografie.....	21

1.Introducere in sisteme IoT

În ultimele decenii, procesul tehnologic s-a dezvoltat masiv, transformând radical numeroase aspecte ale vieții cotidiene. Această evoluție rapidă a tehnologiei a deschis noi orizonturi pentru dezvoltarea unor soluții inovatoare, menite să îmbunătățească confortul, eficiența și securitatea în mediul casnic, industrial etc. Acest fapt este valabil și pentru sistemele de măsură, deoarece tehnologiile IoT facilitează achiziția datelor într-un mod convenabil cum ar fi stocarea acestora pe platforme cloud sau vizualizarea acestora pe instrumente virtuale fiind mai ușor de înțeles și de prelucrat.

Am creat un sistem de măsură a parametrilor de mediu, ce monitorizează temperatura, umiditatea și concentrațiile de gaze. Componentele utilizate pentru achiziția datelor sunt senzorii DHT11 și MQ2. Placă de achiziție date este un ESP 32. Pentru feedback vizual am adăugat și un ecran LCD și LED-uri de control și buzzer. Datele sunt transmise prin port-ul serial către instrumentul virtual și de asemenea sunt logate și în Google Spreadsheets prin WiFi-ul ESP-ului. În figura 1 este prezentată diagrama sistemului de măsură.

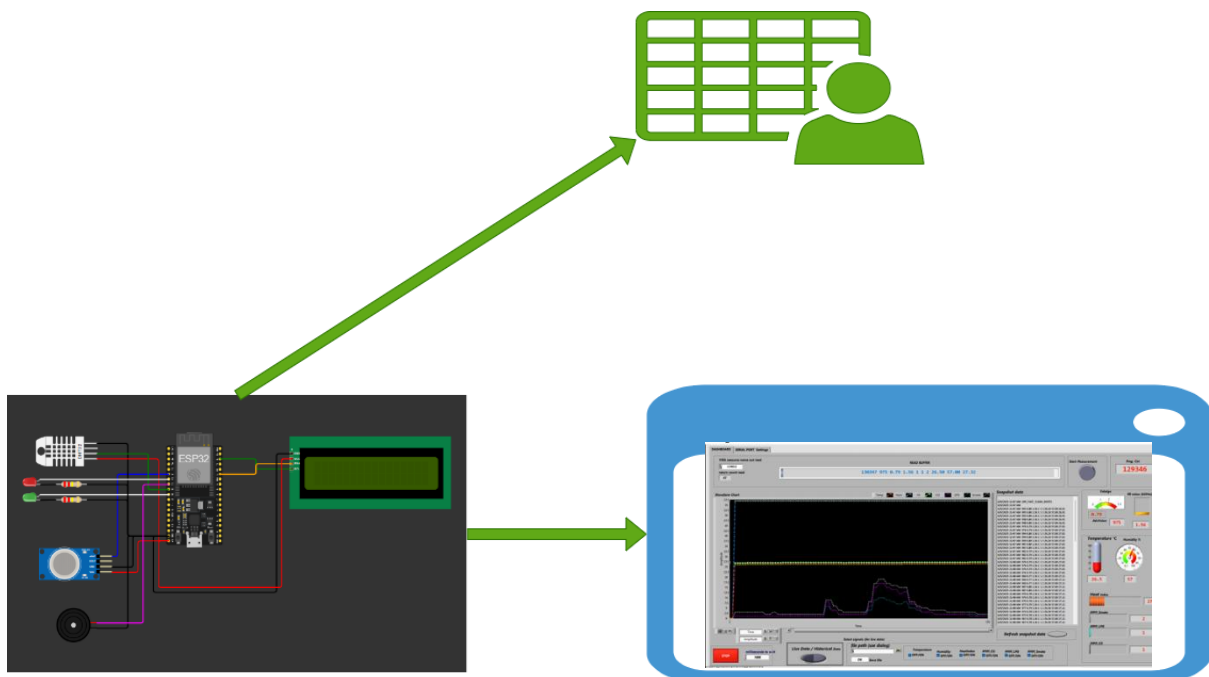


Fig.1 Schema sistemului de măsură

2.Scurta descriere a componentelor utilizate si prezentarea diagramei hardware

ESP32 este un microcontroller dual-core cu conectivitate Wi-Fi si Bluetooth integrata. Este ideal pentru proiecte IoT datorita consumului redus si performantei ridicate. Oferă numerosi pini GPIO, PWM, ADC, SPI, I2C și UART. Suportă FreeRTOS si are memorie flash integrata. Este compatibil cu Arduino IDE, PlatformIO și alte medii. ESP32 este potrivit pentru aplicatii de monitorizare, automatizare si control de la distanta. Dispune de functii avansate de criptare si securitate. ADC este pe 12 biti ceea ce il face util in achizitia de date analogice fata de Arduino unde dispunem de un ADC de 10 biti. In figurile 2, 3 si 4 sunt prezentate diagrama de pinout, diagrama bloc si un tabel cu specificatii pentru placuta, versiunea utilizata este [ESP32 DEVKIT V1 DOIT board](#).

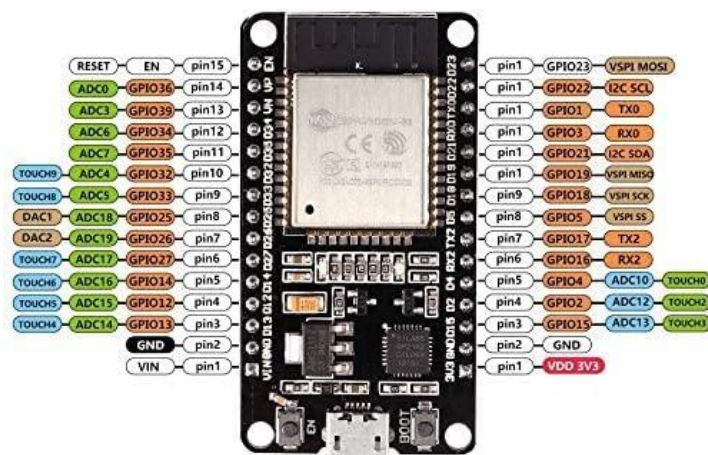


Fig.2 ESP 32 pinout

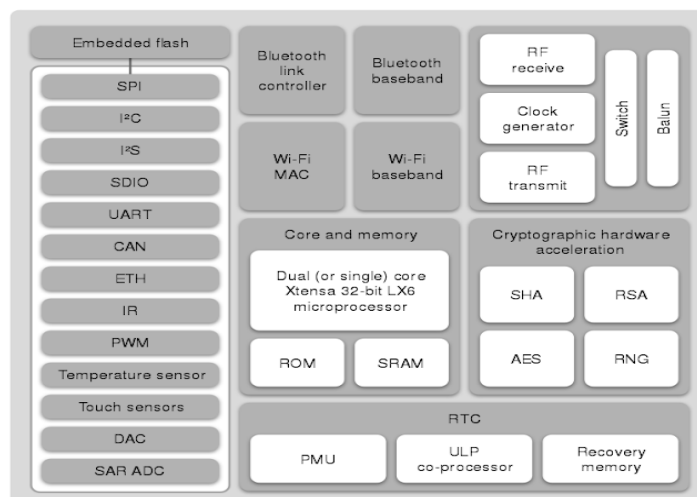


Fig.3 Diagrama bloc

Specifications	ESP8266	ESP32
MCU	Xtensa® Single-Core 32-bit L106	Xtensa® Dual-Core 32-bit LX6 600 DMIPS
802.11 b/g/n Wi-Fi	Yes, HT20	Yes, HT40
Bluetooth	None	Bluetooth 4.2 and below
Typical Frequency	80 MHz	160 MHz
SRAM	160 kBytes	512 kBytes
Flash	SPI Flash , up to 16 MBytes	SPI Flash , up to 16 MBytes
GPIO	17	36
Hardware / Software PWM	None / 8 Channels	1 / 16 Channels
SPI / I2C / I2S / UART	2/1/2/2	4/2/2/2
ADC	10-bit	12-bit
CAN	None	1
Ethernet MAC Interface	None	1
Touch Sensor	None	Yes
Temperature Sensor	None	Yes
Working Temperature	- 40°C – 125°C	- 40°C – 125°C

Fig. 4 Tabel cu specificatii

DHT11 este un senzor digital de temperatura și umiditate. Funcționează în intervalul 0–50°C și 20–90% umiditate relativă. Comunicarea cu microcontrolerul se face printr-un singur pin digital. Are o precizie moderată, suficientă pentru aplicații hobby și educaționale. Oferă valori actualizate la fiecare 1–2 secunde. Necesită o rezistență de pull-up pe linia de date. Este ușor de utilizat cu biblioteci pentru Arduino sau ESP32. Este ideal pentru monitorizarea mediului în proiecte simple. Documentația poate fi consultată la [DHT11 Humidity & Temperature Sensor](#). Figurile 5 și 6 prezintă un tabel cu informații generale despre senzorii DHT 11 folosit în proiectul fizic și DHT 22 folosit în simularea online ([Wokwi - Online ESP32, STM32, Arduino Simulator](#)) și structura internă a senzorilor.

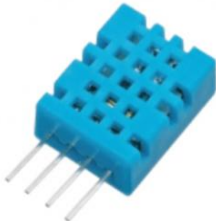


		
DHT11		DHT22
0 - 50°C / ± 2°C	Temperature Range	-40 - 125 °C / ± 0.5 °C
20 - 80% / ± 5%	Humidity Range	0 - 100 % / ± 2-5%
1Hz (one reading every second)	Sampling Rate	0.5 Hz (one reading every two seconds)
15.5mm x 12mm x 5.5mm	Body Size	15.1mm x 25mm x 7.7mm
3 - 5V	Operating Voltage	3 - 5V
2.5mA	Max Current During Measuring	2.5mA

Fig.5 Tabel cu informații generale despre senzorii DHT 11 folosit în proiectul fizic și DHT 22

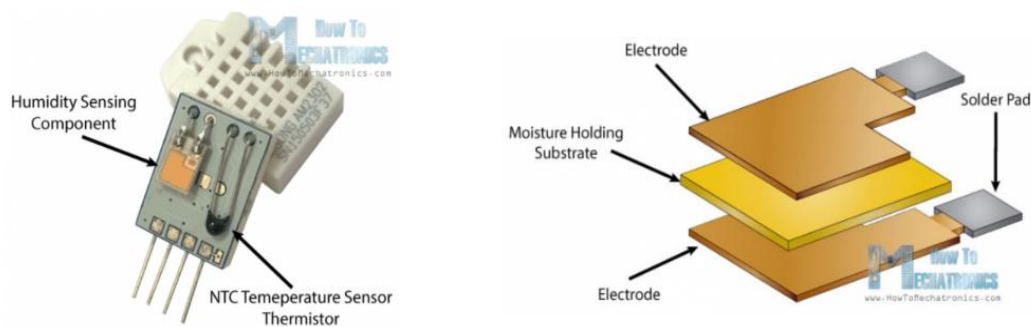


Fig.6 Structura interna a senzorilor

MQ-2 este un senzor de gaz rezistiv capabil să detecteze fum, LPG, metan, CO și alte gaze inflamabile. Produce o tensiune analogică proporțională cu concentrația gazului. Necesită o perioadă de preîncălzire (24–48h pentru R_0 , apoi 20s la pornire). Este sensibil și ușor de integrat cu ESP32 prin ADC. Se utilizează frecvent în sisteme de alarmă sau monitorizare a calității aerului. Oferă atât ieșire analogică, cât și digitală (cu comparator integrat). Este accesibil ca preț și potrivit pentru aplicații de siguranță. Valorile trebuie calibrate pentru estimarea corectă a PPM-urilor folosind curbele specificate în documentație: [MQ-2 \(Ver1.4\) - Manual.pdf](#). În figura 7 este prezentată curba de sensibilitate pentru anumite gaze (în proiectul actual se măsoară concentrația de CO, GPL și fum) și caracteristica temperatura/umiditate față de raportul R_s/R_{so} .

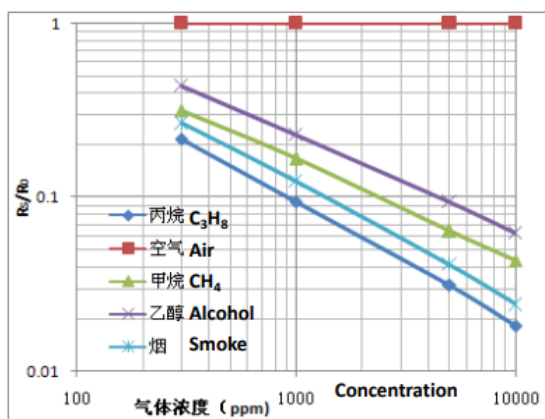


Fig3. Typical Sensitivity Curve

The ordinate is resistance ratio of the sensor (R_s/R_0), the abscissa is concentration of gases. R_s means resistance in target gas with different concentration, R_0 means resistance of sensor in clean air. All tests are finished under standard test conditions.

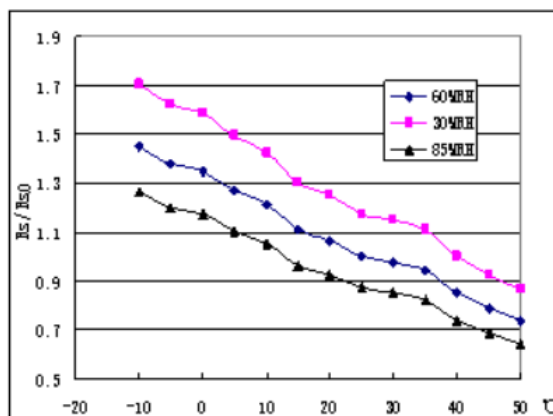


Fig4. Typical temperature/humidity characteristics

The ordinate is resistance ratio of the sensor (R_s/R_{so}). R_s means resistance of sensor in 2000ppm propane (C_3H_8) under different tem. and humidity. R_{so} means resistance of the sensor in 2000ppm propane under 20°C/55%RH.

Fig.7 Burba de sensibilitate pentru anumite gaze și caracteristica temperatura/umiditate față de raportul R_s/R_{so}

LCD-ul 16x2 afisează 2 linii a câte 16 caractere. Modulul I2C atasat permite controlul prin doar 2 pini (SDA si SCL). Adresa frecventa este 0x27, dar poate fi 0x3F în unele module. Este compatibil cu multe biblioteci Arduino și ESP32 (LiquidCrystal_I2C). Permite afisarea valorilor în timp real, meniuri sau mesaje. Consumă putina energie și are contrast reglabil printr-un potentiometru. Ideal pentru interfete simple de utilizator. Usor de implementat în proiecte embedded ([I2C_1602_LCD.pdf](#)).

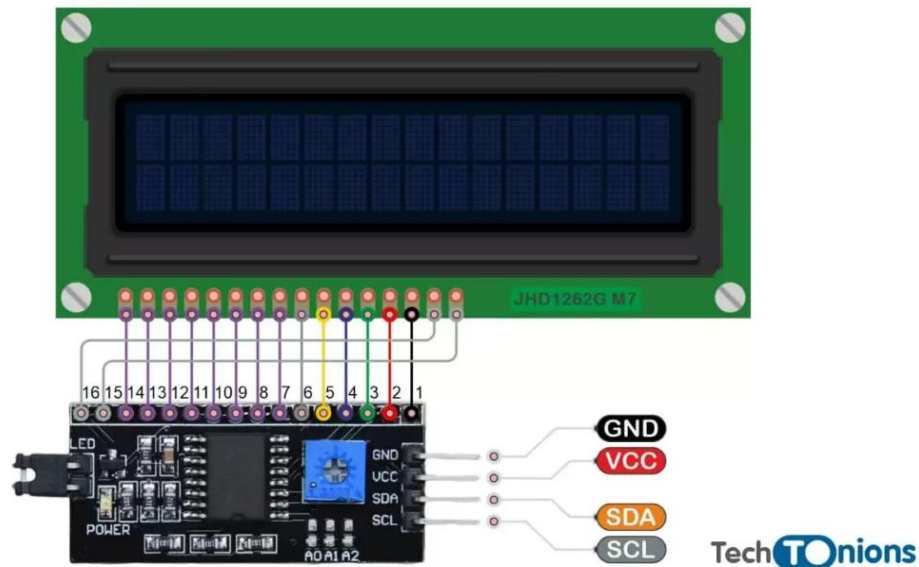


Fig. 9 Conexiunile LCD-ului

No.	PIN	Function
1	VSS	Ground
2	VCC	+5 Volt
3	VEE	Contrast control 0 Volt: High contrast.
4	RS	Register Select 0: Command Reg. 1: Data Reg.
5	RW	Read / write 0: Write 1: Read
6	E	Enable H-L pulse
7-14	D0 - D7	Data Pins D7: Busy Flag Pin
15	LED+	+5 Volt
16	LED-	Ground

16x2 LCD Pin Description

Fig.10 Pinii LCD-ului

In figura 11 este prezentata interfatarea senzorilor cu placuta ESP32 fiind specificate si protocoalele de comunicatie.

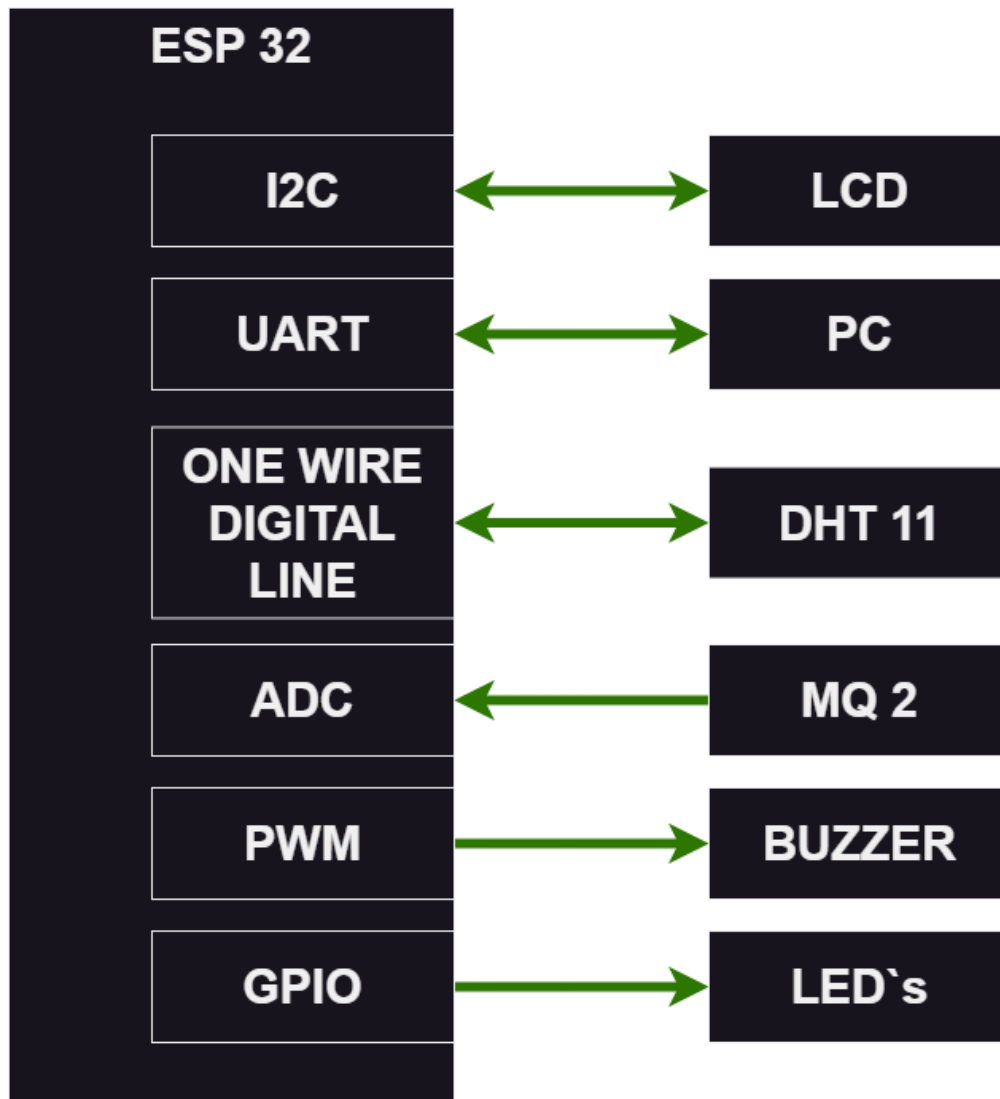


Fig.11 Interfatarea senzorilor si perfericelor cu placuta ESP 32

3. Software pentru achizite date rulat pe ESP32

Componenta software pentru acest sistem de monitorizare consta intr-un program ce achizitioneaza datele de la senzori si le transmite catre Google Spreadsheets printr-o cerere POST. Pentru achizitionarea datelor de la senzorul MQ2 se foloseste canalul de ADC si pentru DHT11 folosind protocolul specific de comunicare pe o singura linie digitala. In partea de setup se initializeaza comunicarea seriala si se initializeaza sistemul de operare. Procesorul fiind dual core s-a putut imparti achizitia de date de transmiterea acestora. In Figura 12 este prezentat tabelul cu impartirea functiilor, in figura 13 arhitectura software-ului, link-ul catre arhiva Github: [CodreanuDan/EnvDataMonitoring_ESP32_Cloud_LabView: Environment data monitoring using ESP32 Mq2 & Dht, Cloud and LabView Interface.](https://github.com/CodreanuDan/EnvDataMonitoring_ESP32_Cloud_LabView)

Task Timing Configuration									
Item nr.	Item name	Type	Core	Priority	Cycle	Stack Size	Debug Counters	Comments	
#	Task_C0_500ms	Basic		0	20	500ms	16384	cnt_500ms_C0	
#	LimitAlertCheck	Runnable		0					
#	Task_C0_2s	Basic		0	10	2s	16384	cnt_2s_C0	
#	MQGetGasPercentage (LPG)	Runnable		0					
#	MQGetGasPercentage (CO)	Runnable		0					
#	MQGetGasPercentage (Smoke)	Runnable		0					
#	MQGetVoltage	Runnable		0					
#	DHT_Measurement	Runnable		0					
#	LcdDisplayData	Runnable		0					
#	Task_C0_Init	Basic		20		/	16384	init_flag_C0	
1	Port_Init	Runnable		0					
2	Adc_Init	Runnable		0					
3	I2C_Init	Runnable		0					
4	LCD_Init	Runnable		0					
5	MQCalibration	Runnable		0					
6	Dht_Init	Runnable		0					
#	Task_C1_Init	Basic		1	20		16384	init_flag_C1	
#	Wifi_Init	Runnable		1					
#	Task_C1_60s	Basic		1	15	60s	16384	cnt_60s_C1	
#	Wifi_SendDataToGoogleSpreadsheets	Runnable		1					
#	Task_AsyncEvents_C1	Basic		1	20	Event driven	16384	cnt_AsyncEvents_C1	
#	Wifi_SendDataToGoogleSpreadsheets	Runnable		1					
#	loop	Basic		0	?	/	?	/	
#	DebugPrint	Runnable		0					
#	setup	Basic		0	?	2s	?	/	
#	SerialBegin	Runnable		0					
#	PSRAM_Check	Runnable		0					
#	CheckObjectPointers	Runnable		0					
#	SPIFFS_Init	Runnable		0					
#	OS_InitOS	Runnable		0					
#	OS_CreateTaskSemphForCores	Runnable		0					
#	OS_CreateQueues	Runnable		0					
#	OS_CreateTasks	Runnable		0					

Fig.12 Organizarea functiilor pe task-uri si pe core-uri

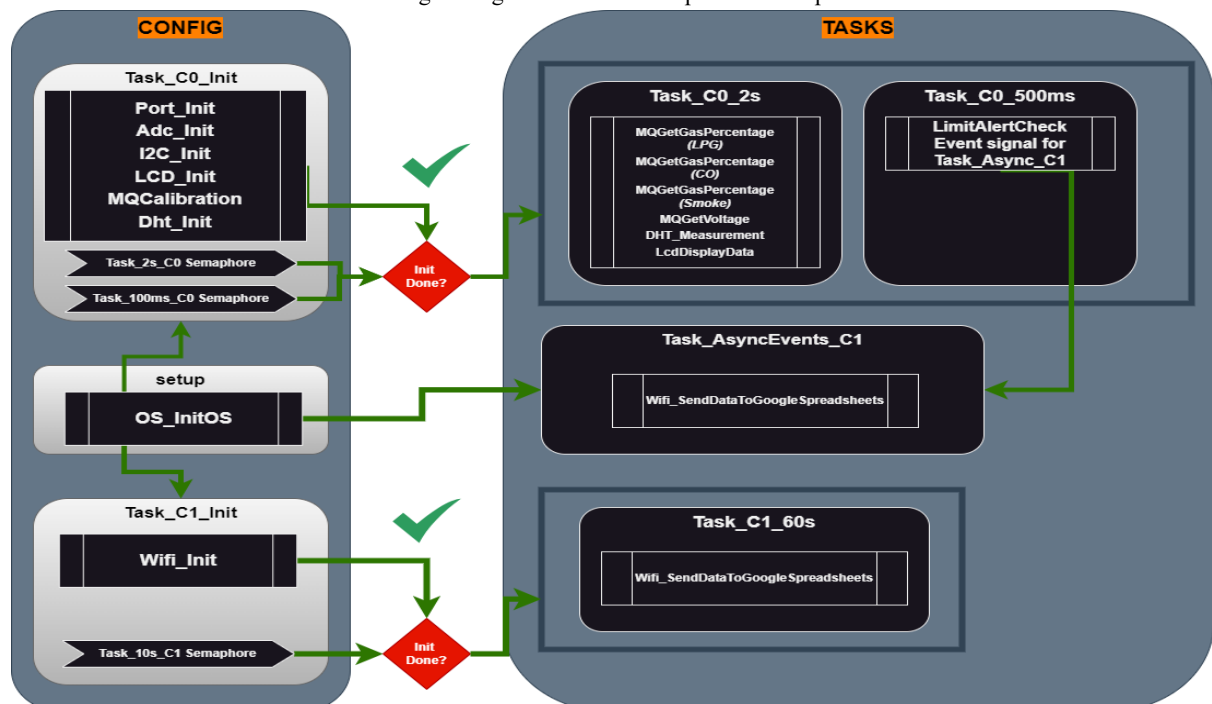


Fig. 13 Arhitectura software utilizata

Fig. 14 Setarile de flash

Dezvoltarea initiala a aplicatiei s-a facut prima data simulat online in Wokwi.com ([Wokwi - Online ESP32, STM32, Arduino Simulator](#)) simularea continand toate functionalitatile de baza ale aplicatiei reale: Achizitie date, afisare pe ecran, LED-uri de notificare si buzzer si transmitere date catre resursa cloud. In figura 15 este prezentata diagram de conexiuni care serveste si ca diagrama hardware generala a proiectului fix cu mentiunea ca senzorul simulat este DHT 22 si cel fizic DHT 11, de asemenea Buzzer-ul a fost eliminat pentru a nu produce sunete deranjante cand se dapasesc pragurile de conc. la gaze.

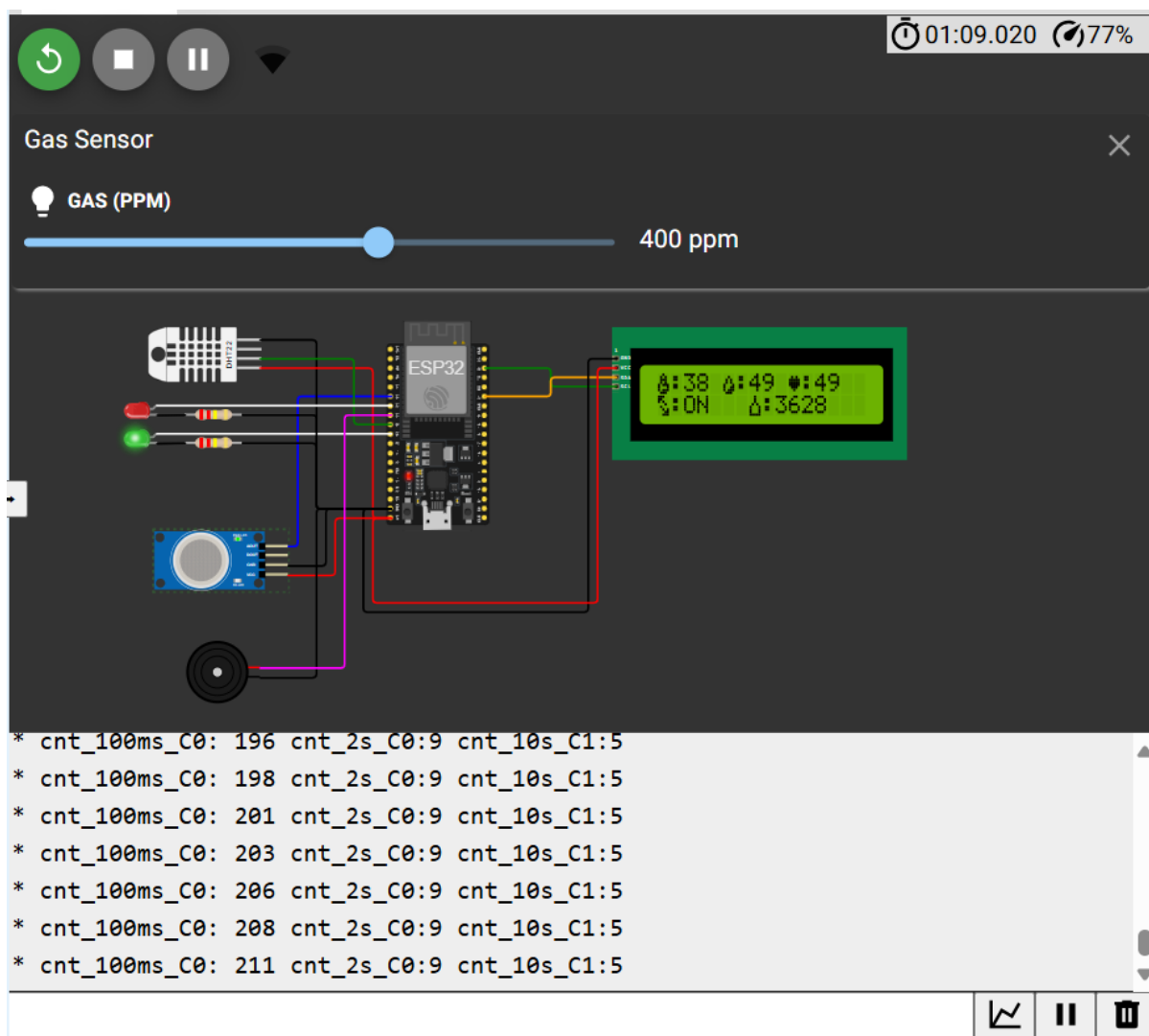


Fig. 15 Simularea sistemului in Wokwi

Componentele principale ale software-ului sunt:

1. Grupul de functii pentru achizitie de date de la senzori
2. Functiile de comunicare prin Wifi
3. Functiile de afisare pe ecranul LCD
4. Functiile de verificat parametrii si alerta

1. Grupul de functii pentru achizitie de date de la senzori

Citirea de la senzorul DHT 11 se face interogand senzorul. Acesta livreaza o secventa de 5 bytes dintre care primii doi returneaza valoarea umiditatii, urmasorii doi returneaza valoarea temperaturii byte-ul 5 returneaza CRC-ul. Validarea datelor sa face prin compararea CRC-ului cu suma celoralti bytes **AND** cu **0xFF** ca mai jos:

```
if (data[4] == ((data[0] + data[1] + data[2] + data[3]) & 0xFF))
```

Datele sunt obtinute prin sumarea $\text{data}[0] + \text{data}[1] / 10$ pentru *temperatura* si $\text{data}[2] + \text{data}[3] / 10$ pentru *umiditate*. De asemenea se obtine si indicile de temperatura resimtita (*Heat Index*) dupa formula:

```
hi = 0.5 * (temp + 61.0 + ((temp - 68.0) * 1.2) + (hum * 0.094));
```

Diagrama interfetei cu senzorul DHT 11 este prezentata in figura 16:

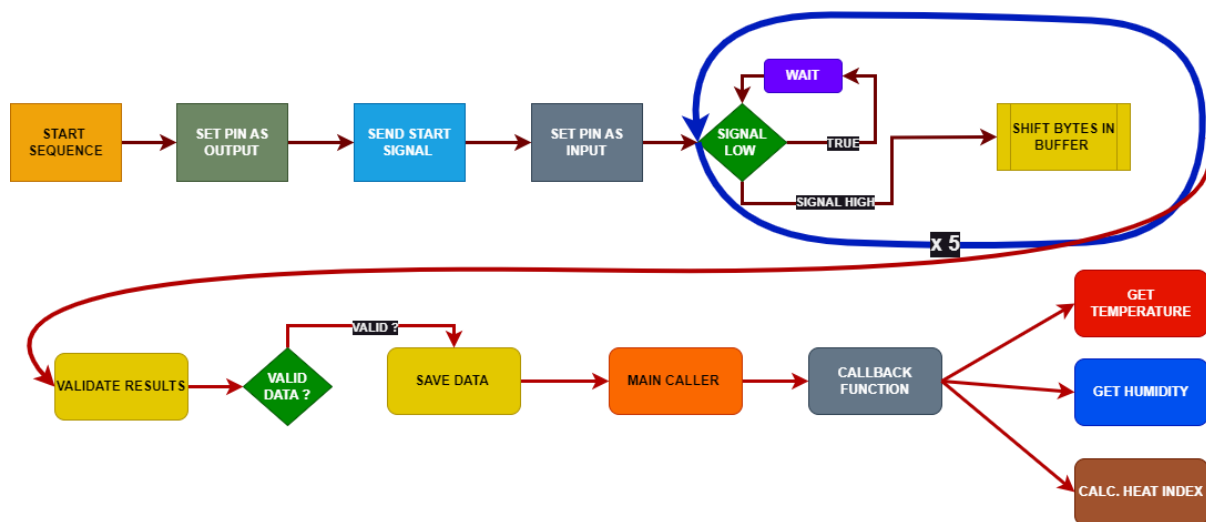


Fig. 16 Diagrama interfetei cu senzorul DHT 11

Pentru achizitia datelor de la senzorul MQ2 se vor citi datele brute preluate de la ADC, se si vor converti in tensiune cu formula:

```
float voltage = (adc_raw_value / ADC_MAX_RESOLUTION) * 3.3;
```

unde rezolutia maxima a ADC-ului este 12 biti deci 4095.

Calibrarea se face o singura data, la pornirea sistemului, afland valoarea de referinta **Ro** in aer curat (curba de comparatie pentru alte masuratori). Se executa:

```
val += MQResistanceCalculation(analogRead(mq_pin));
```

Se repeta de CALIBARAION_SAMPLE_TIMES ori (ex: 500). Se calculeaza:

```
val = val / CALIBARAION_SAMPLE_TIMES; // Medie Rs
```

```
Ro = val / RO_CLEAN_AIR_FACTOR; // Ro = Rs / factor (9.83 default)
```

Citirea periodica se executa in bucla pentru a obtine valoarea curenta a gazului in urmatorul mod `rs += MQResistanceCalculation(analogRead(mq_pin));` si poi se face media utlizand formula `rs = rs / READ_SAMPLE_TIMES;` cu `READ_SAMPLE_TIMES = 10.`

Calculul rezistentei senzorului (Rs) se face astfel ca in ecuatia 1:

```
return ( ((float)RL_VALUE*(ADC_MAX_RESOLUTION - raw_adc)) / raw_adc );
```

$$R_s = R_L \times \left(\frac{V_{max} - V_{out}}{V_{out}} \right) \quad (1)$$

Calcul concentratiei de gaz:

```
return (pow(10, (((log(rs_ro_ratio) - pcurve[1]) / pcurve[2]) + pcurve[0])));
```

unde:

- pcurve[0] – x1
- pcurve[1] – y1
- pcurve[2] – panta dreptei

dupa ecuatia 2:

$$ppm = 10^{\left(\frac{\log_{10}(R_s \backslash R_0) - y_1}{slope} + x_1 \right)} \quad (2)$$

unde R0 este valoarea de referinta si este calculat dupa ecuatia 3:

$$R_0 = \frac{R_{Smediu}}{RO_CLEAN_AIR_FACTOR} \quad (3)$$

unde RO_CLEAN_AIR_FACTOR=9.83 (Rezistenta in aer curat / R0)si RL_VALUE=5.

Si pantele punctele si pantele definite si in software si rezistenta R0:

```
/*
```

```
 * Two points are taken from the curve. With these two points, a line is
 * formed which is "approximately equivalent" to the original curve.
```

```
 * data format:{ x, y, slope}; point1: (lg200, 0.21), point2: (lg10000, -
 * 0.59)
```

```
*/
```

```
float LPGCurve[3] = {2.3,0.21,-1.04};
```

```
float COCurve[3] = {2.3,0.72,-0.78};
```

```
float SmokeCurve[3] = {2.3,0.53,-0.98};
```

```
float Ro = 10;
```

Diagrama pentru calcularea concentratiei de gaze este prezentata in figura 17:

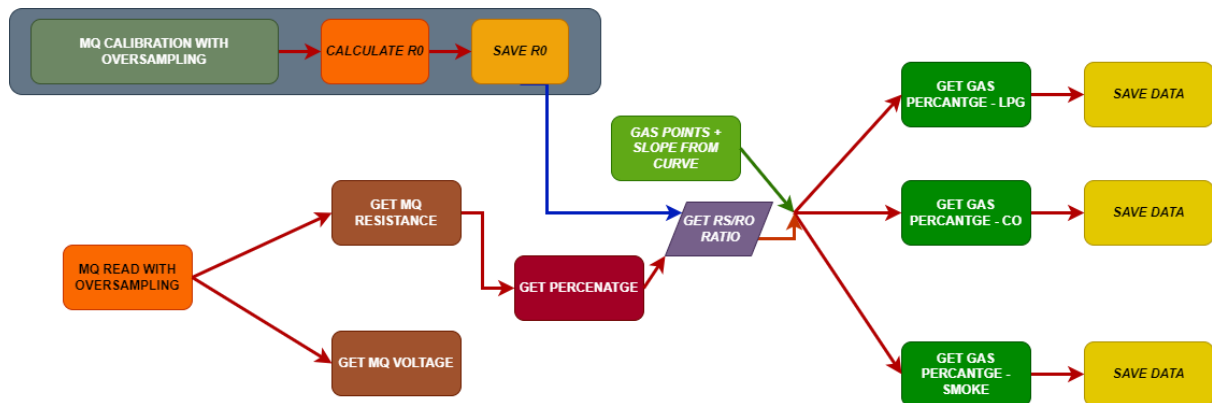


Fig. 17 Diagrama pentru calcularea concentratiei de gaze

2.Funcțiile de comunicare prin Wifi

Transmisia datelor se face utilizand modulul WiFi 2.4 GHz al ESP 32. Conexiunea la o retea de internet se face prin incercarea de conctare a perechilor SSID + PAROLA din fisierul `WIFI_CONFIG_FILE "/wifi_networks.txt"` din memoria SPIFFS a placutei ESP 32.

Datele sunt tranmise sub forma de URL la un Google Spreadsheet formatat astfel:

```

String url = googleScriptURL +
    "?temperature=" + String(data->temperature, 2) +
    "&humidity=" + String(data->humidity, 2) +
    "&heatIndex=" + String(data->heatIndex, 2) +
    "&adcMq2Voltage=" + String(data->adcMq2Voltage, 2) +
    "&Ro=" + String(data->Ro, 2) +
    "&iPPM_LPG=" + String(data->iPPM_LPG) +
    "&iPPM_CO=" + String(data->iPPM_CO) +
    "&iPPM_Smoke=" + String(data->iPPM_Smoke) +
    "&adcRawValue=" + String(data->adcRawValue);
  
```

Datele sunt transmise cu functia `Wifi_SendDataToGoogleSheets` prin intermediul task-ului de 60s `Task_C1_60s` ce ruleaza pe CORE 1 sau prin task-ul asincron `Task_AsyncEvents_C1` ce ruleaza atunci cand o alarma de depasire a concentratiei de gaze este trimisa in coada de care asculta acest task. In figura 18 este repsrezentata foaia de calcul din Google Drive:

Data									
Timestamp	Temperature	Humidity	HeatIndex	adcMq2Voltage	Ro	IPPM_LPG	IPPM_CO	IPPM_Smoke	adcRawValue
Avg.value	31.67617729	44.79362881	33.45069252	1.378836565	1.46398892	60.58725762	505948.4432	789.2409972	1709.740997
Nr.Samples	722	722	722	722	722	722	722	722	722
6/3/2025 18:34:21	33.8	34	33.75	1.91	2.31	27	3603	358	2365
6/3/2025 18:33:54	33.8	34	33.75	1.79	2.31	12	1188	146	2227
6/3/2025 18:36:25	33.7	34	33.61	1.76	2.31	6	459	69	2179
6/3/2025 18:36:12	33.7	34	33.61	1.79	2.31	9	717	93	2217
6/3/2025 18:35:58	33.7	34	33.61	1.8	2.31	10	659	102	2237
6/3/2025 18:35:30	33.7	34	33.61	1.87	2.31	14	2158	245	2315
6/3/2025 18:35:16	33.7	34	33.61	1.89	2.31	25	3476	337	2351
6/3/2025 18:35:03	33.7	34	33.61	1.94	2.31	39	6541	566	2404
6/3/2025 18:34:49	33.7	34	33.61	1.94	2.31	55	10985	809	2404
6/3/2025 18:34:35	33.7	34	33.61	2.01	2.31	81	17150	1228	2494
6/3/2025 18:34:07	33.7	34	33.61	1.87	2.31	20	2398	256	2320
6/5/2025 15:19:38	33.3	40	34.26	0.66	2.09	2	2	3	815
6/3/2025 20:13:16	33.2	40	34.1	1.53	1.91	12	53	32	1902
6/3/2025 20:13:13	33.2	40	34.1	1.53	1.91	12	53	32	1902
6/3/2025 20:13:09	33.2	40	34.1	1.53	1.91	12	53	32	1902
6/3/2025 20:13:05	33.2	40	34.1	1.53	1.91	12	53	32	1902
6/3/2025 20:13:01	33.2	40	34.1	1.53	1.91	12	53	32	1902
6/3/2025 20:12:58	33.2	40	34.1	1.53	1.91	12	53	32	1902
6/3/2025 20:12:54	33.2	40	34.1	1.53	1.91	12	53	32	1902
6/3/2025 20:12:51	33.2	40	34.1	1.53	1.91	12	53	29	1900
6/3/2025 20:12:31	33.2	40	34.1	0.85	1.91	0	1	1	1056
6/5/2025 15:10:31	33.2	40	34.1	0.65	2.09	2	1	3	806
6/5/2025 15:11:32	33.2	40	34.1	0.65	2.09	2	1	3	810
6/5/2025 15:13:33	33.2	40	34.1	0.65	2.09	2	2	3	811
6/5/2025 15:14:34	33.2	40	34.1	0.65	2.09	2	2	3	811

Fig.18 Foaia de calcul din Google Drive

3. Functiile de afisare pe ecranul LCD

Afisarea pe ecranul lcd se face cu ajutorul unei masini de stari care modifica informatiile afisate in functie de valoarea counter-ului task-ului de 2s `Task_C0_2s:cnt_2s_C0`. Daca valoarea este *para* se vor afisa Temperatura, umiditatea, heat index-ul si valoarea citita de ADC si status-ul conexiunii la WiFi , iar daca valoarea este *impara* concentratiile pentru cele 3 tipuri de gaze, folisndu-se inconite custom generate cu ajutorul: [LCD Custom Character Generator](#). In faza de initializare, apare o animate care indica etapa de setup pentru MQ2, dupa ce se observa in fig.19:

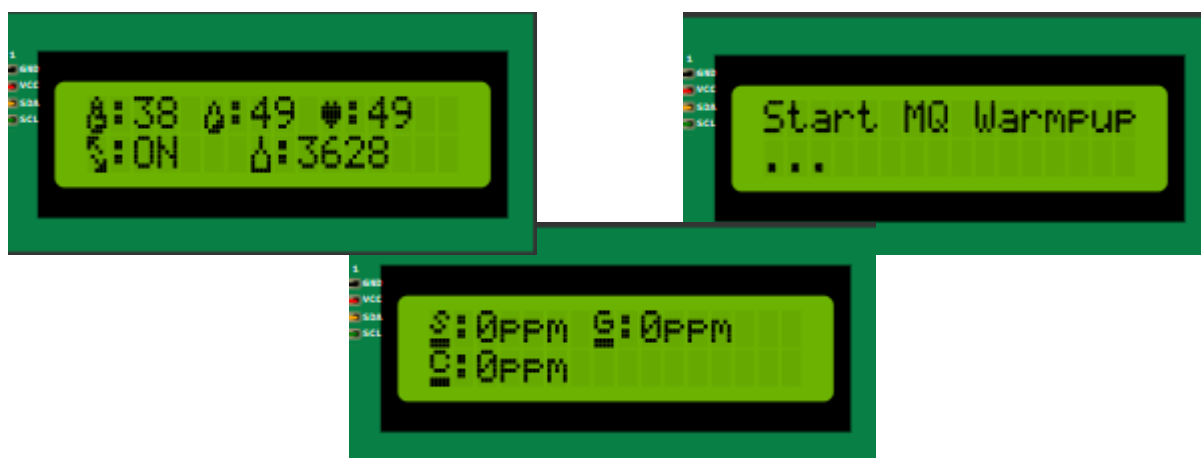


Fig.19 Afisarea pe ecran

4. Functiile de verificat parametrii si alerta

Acest modul verifica daca concentratiile de gaze depasesc limitele stabilite si notifica cazurile urmatoare: cazul normal, nivelurile nu sunt depasite, led verde aprins, daca sunt depasite led rosu aprins, buzzer on si in coada de notificare pentru **Task_AsyncEvents_C1** sunt incarcate datele pentru ca **Wifi_SendDataToGoogleSpreadsheets** sa le trimita in Google Drive, dupa cum este prezentat in figura 20:

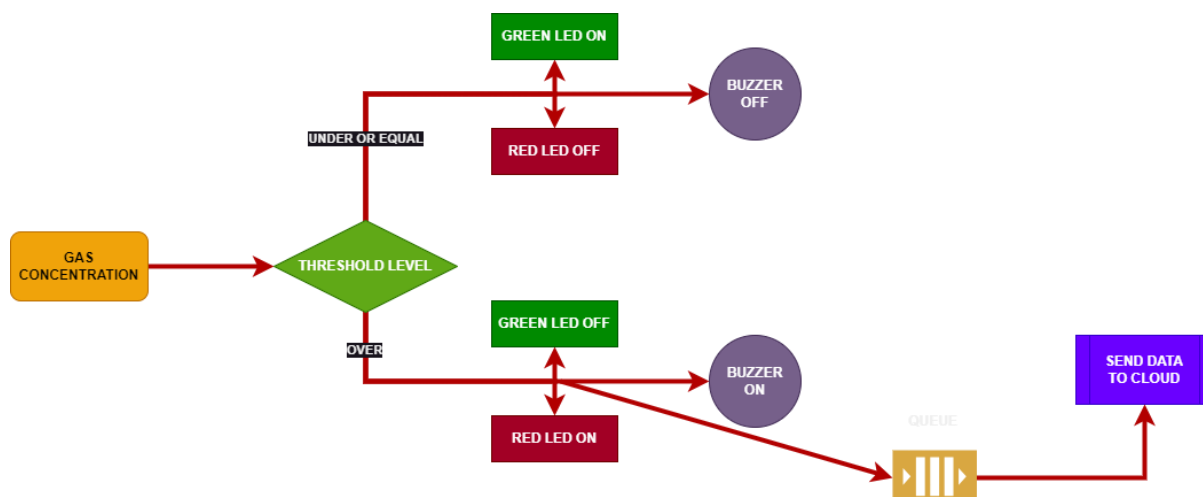


Fig. 20 Sistemul de alerta

In figura 21 sunt afisate setarile de flash pentru Software:

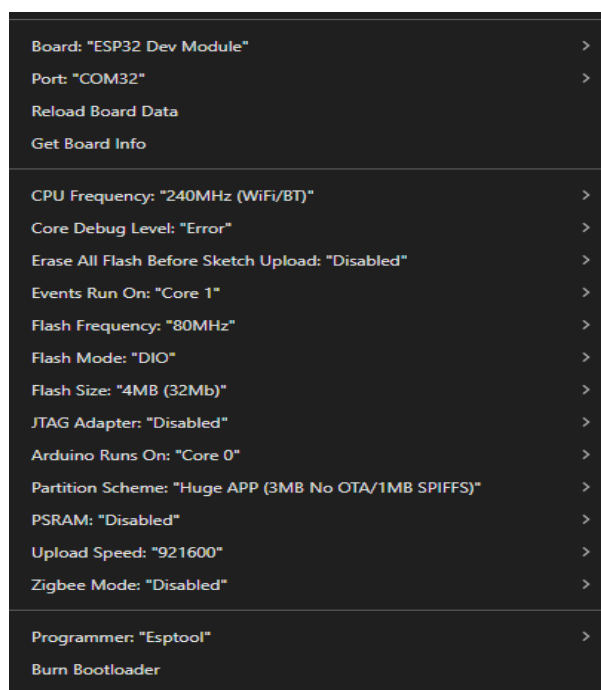


Fig. 21 Setarile de flash pentru software

!!! Probleme de rezolvat pe viitor: De rezolvat crash-urile cauzate de pornirea citiri cu VISA din Labview, crash-uri legate de nescrinoizare pe partea de OS si cele legate de timeout-uri la trimiterea datelor catre Google Drive.

4. Aplicatia de monitorizare realizata in Labview

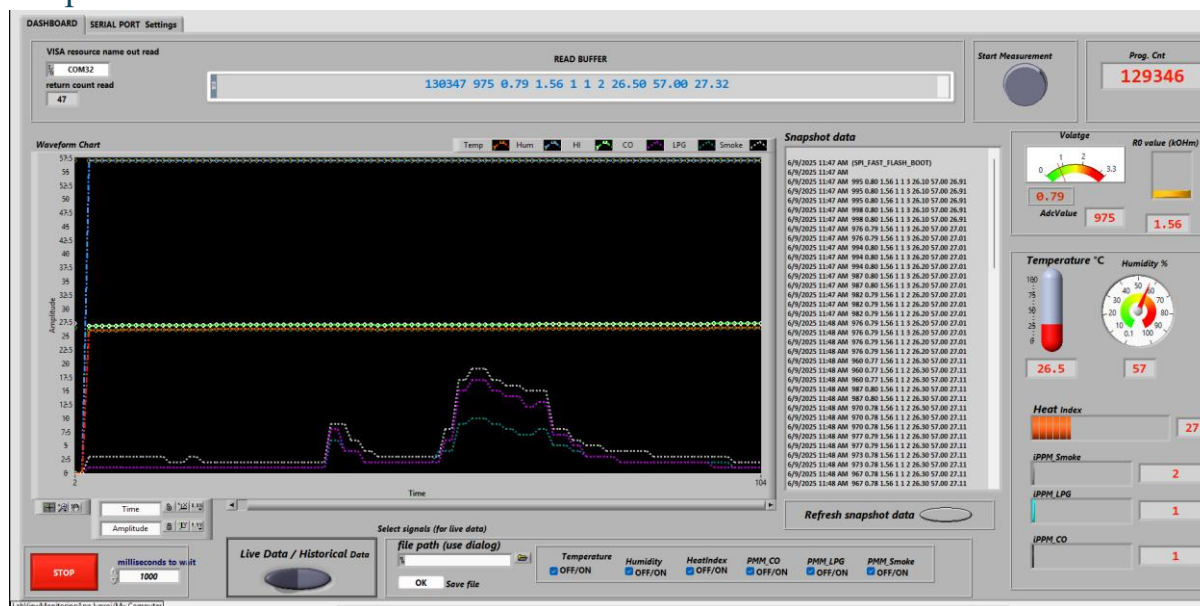


Fig. 22 Panoul de control in Labview

In figura 22 este prezentat panoul de control realizat in Labview unde datele pot fi urmarite sub forma grafica sau ca si indicator. Acestea sunt transmise de ESP32 prin portul serial si receptionate de aplicatia Labview utilizand VISA.

Datele sunt salvate si intr-un fisier text cu format data-ora dupa cum se observa si in figura 23 si se pot afisa pe grafic.

File	Edit	View	
6/9/2025 11:48 AM	967	0.78	1.56 1 1 2 26.30 57.00 27.11
6/9/2025 11:48 AM	994	0.80	1.56 1 1 2 26.30 57.00 27.11
6/9/2025 11:48 AM	994	0.80	1.56 1 1 2 26.30 57.00 27.11
6/9/2025 11:48 AM	1396	1.12	1.56 6 8 9 26.30 57.00 27.11
6/9/2025 11:48 AM	1396	1.12	1.56 4 6 8 26.30 57.00 27.11
6/9/2025 11:48 AM	1247	1.00	1.56 4 6 26.30 57.00 27.11
6/9/2025 11:48 AM	1247	1.00	1.56 4 6 26.30 57.00 27.11
6/9/2025 11:48 AM	1090	0.88	1.56 2 2 4 26.30 57.00 27.11
6/9/2025 11:48 AM	1090	0.88	1.56 2 2 4 26.30 57.00 27.11
6/9/2025 11:48 AM	1082	0.87	1.56 2 2 3 26.20 57.00 27.01
6/9/2025 11:48 AM	1051	0.85	1.56 2 2 3 26.30 57.00 27.11
6/9/2025 11:48 AM	1051	0.85	1.56 2 2 3 26.30 57.00 27.11
6/9/2025 11:48 AM	1051	0.85	1.56 2 2 3 26.30 57.00 27.11
6/9/2025 11:48 AM	1039	0.84	1.56 2 2 3 26.30 57.00 27.11
6/9/2025 11:48 AM	1040	0.84	1.56 2 2 3 26.30 57.00 27.11
6/9/2025 11:48 AM	1040	0.84	1.56 2 2 3 26.30 57.00 27.11
6/9/2025 11:48 AM	1040	0.84	1.56 4 2 3 26.30 57.00 27.11
6/9/2025 11:48 AM	1373	1.11	1.56 4 6 8 26.30 57.00 27.11
6/9/2025 11:48 AM	1373	1.11	1.56 4 6 8 26.30 57.00 27.11
6/9/2025 11:48 AM	1648	1.33	1.56 9 15 17 26.30 57.00 27.11
6/9/2025 11:48 AM	1648	1.33	1.56 9 15 17 26.30 57.00 27.11
6/9/2025 11:48 AM	1683	1.36	1.56 10 17 19 26.30 57.00 27.11
6/9/2025 11:48 AM	1683	1.36	1.56 10 17 19 26.30 57.00 27.11
6/9/2025 11:48 AM	1683	1.36	1.56 10 17 19 26.30 57.00 27.11
6/9/2025 11:48 AM	1623	1.31	1.56 9 15 17 26.30 57.00 27.11
6/9/2025 11:48 AM	1629	1.31	1.56 8 14 16 26.30 57.00 27.11
6/9/2025 11:48 AM	1629	1.31	1.56 7 14 16 26.30 57.00 27.11
6/9/2025 11:48 AM	1585	1.28	1.56 7 12 15 26.30 57.00 27.11
6/9/2025 11:48 AM	1584	1.28	1.56 8 13 15 26.40 57.00 27.21
6/9/2025 11:48 AM	1584	1.28	1.56 8 13 15 26.40 57.00 27.21

Fig. 23 Fisierul ce stocheaza datele achizitionate

In schema bloc, partea de citire a datelor este prezentata in figura 24, unde datele sunt salvate sub forma unui spreadsheet string si de asemenea formate pentru scrierea in fisier folosind un dialog pentru selectarea fisierului si buton de confirmare (fig.22 mijloc-jos).

In figura 25 este prezentat mecanismul de extragere a datelor din buffer-ul de la portul serial plus adaugarea datelor in draft-ul fisierului de date si in figurile 26 si 27 mecanismul ce comuta intre datele citite din fisier si cele in timp real ce for fi afisate pe grafic.

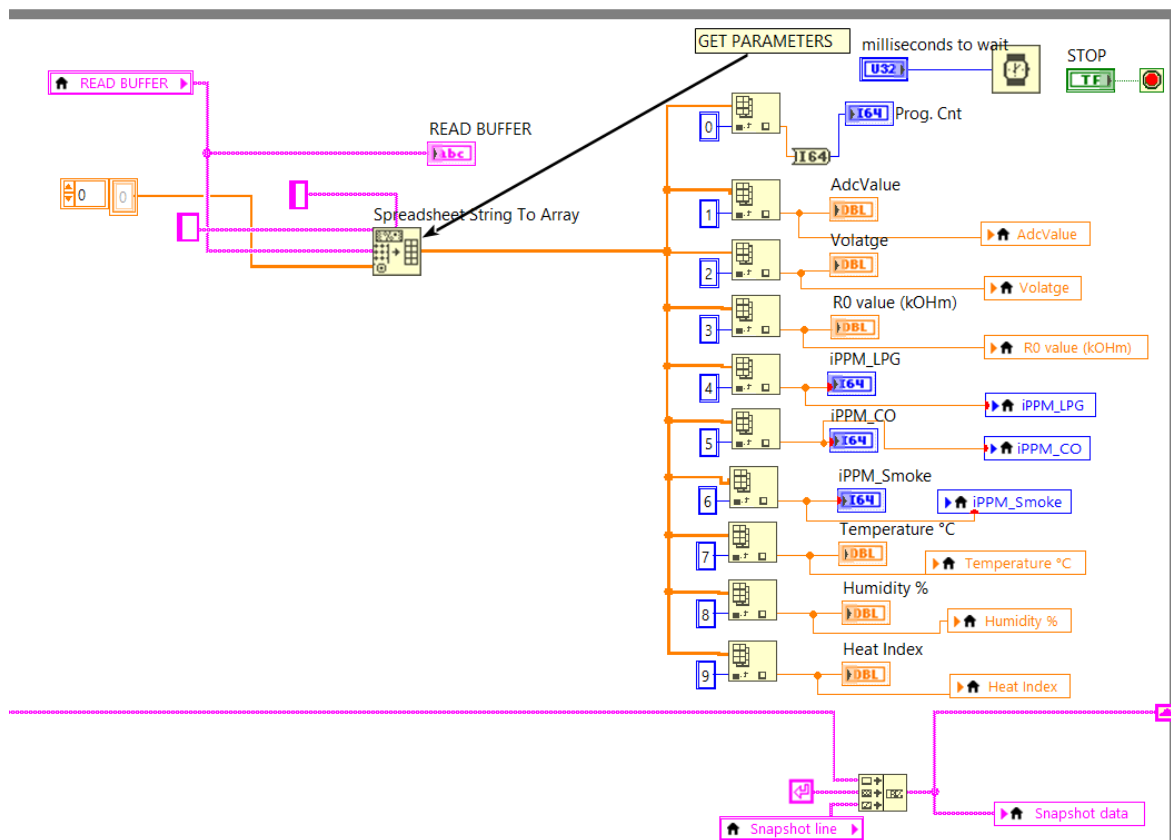


Fig. 25 Mecanismul de extragere a datelor din buffer-ul de la portul serial plus adaugarea datelor in draft-ul fisierului de date

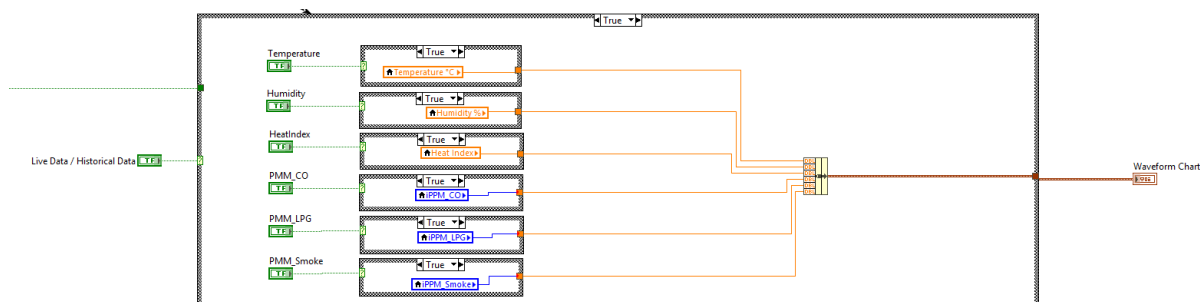


Fig.26 Afisarea datelor pe chart in timp real cu optiunea de a selecta parametrii doriti

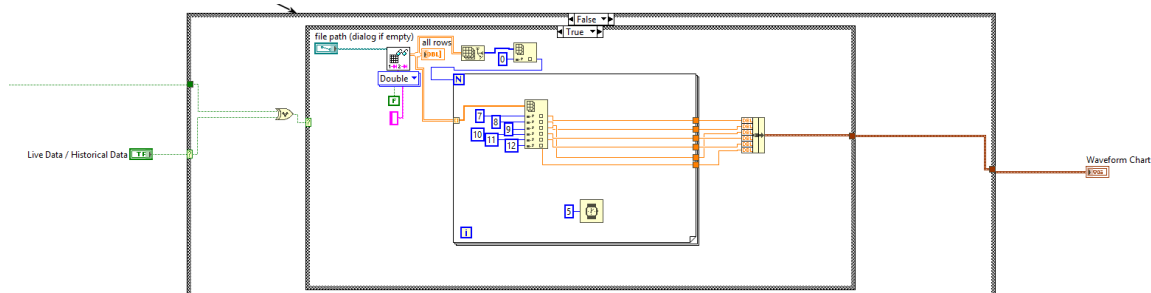


Fig.27 Afisarea datelor din fisier

Indicele de confort termic (sau indicele de disconfort termic) este un indicator care reflecta senzatia termică resimțită de corpul uman în functie de temperatura și umiditate. Exista mai multe formule utilizate international, dar in Romania si în Europa Centrala se foloseste frecvent **indicele de disconfort termic ITU**, definit de ANM. Formula utilizată de ANM in ecuatia 4 in °C si 5 in °F:

$$ITU = T - (0.55 - 0.0055 \cdot RH) \cdot (T - 14.5) \quad (4)$$

$$ITU = (T * 1.8 + 32) - (0.55 - 0.0055 * RH) * ((T * 1.8 + 32) - 58) \quad (5)$$

unde:

- T = temperatura aerului ($^{\circ}\text{C}$),
- RH = umiditatea relativă (%),
- ITU = indicele de temperatură-umezeală (indice de confort termic).

In figura 28 este afiasata implementarea in Labview si in figura 29 afisarea pe interfata principala:

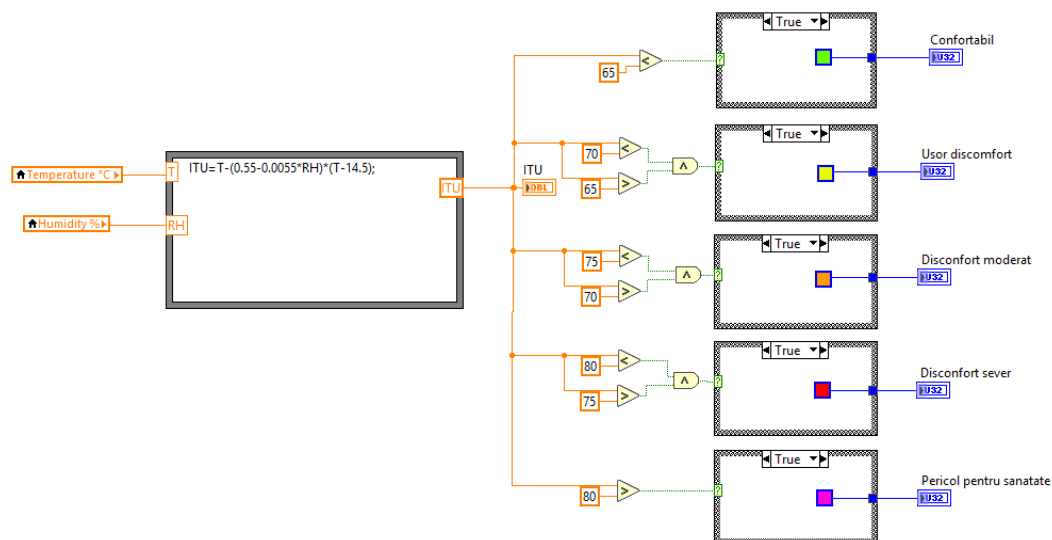


Fig.28 Implementarea in Labview a calculului ITU

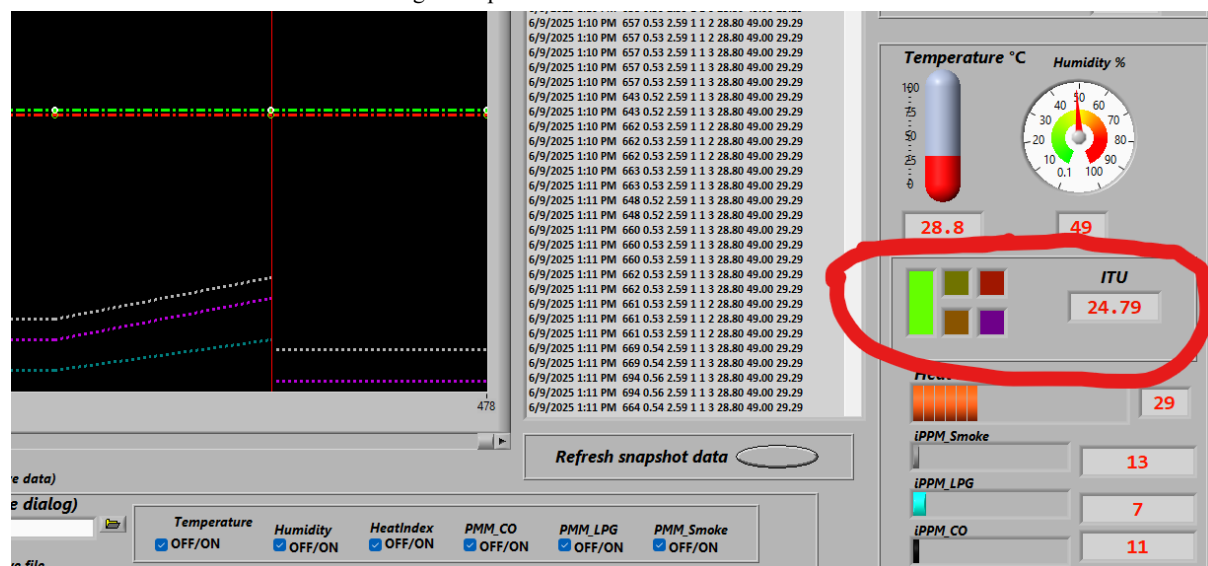
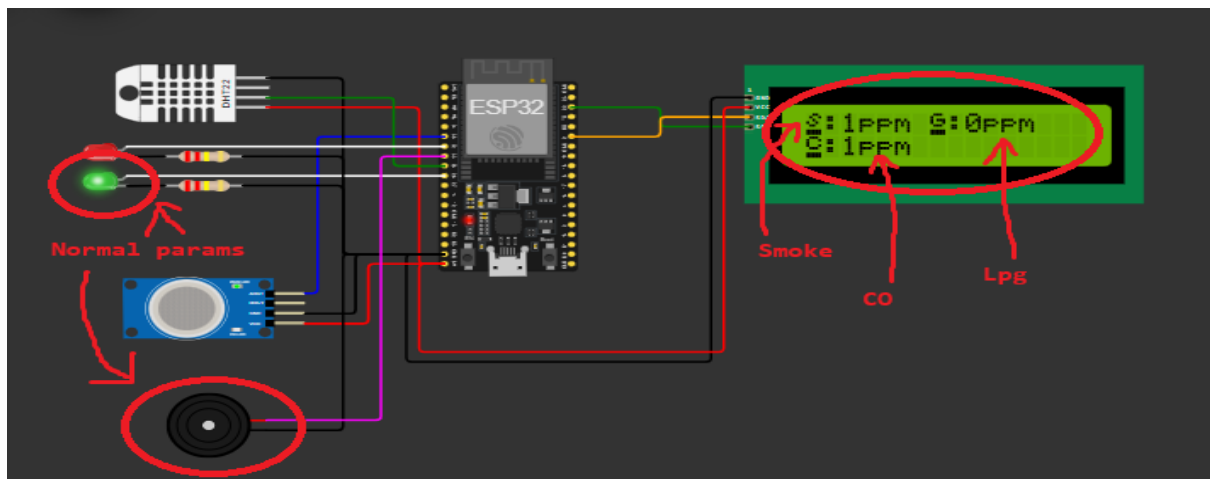
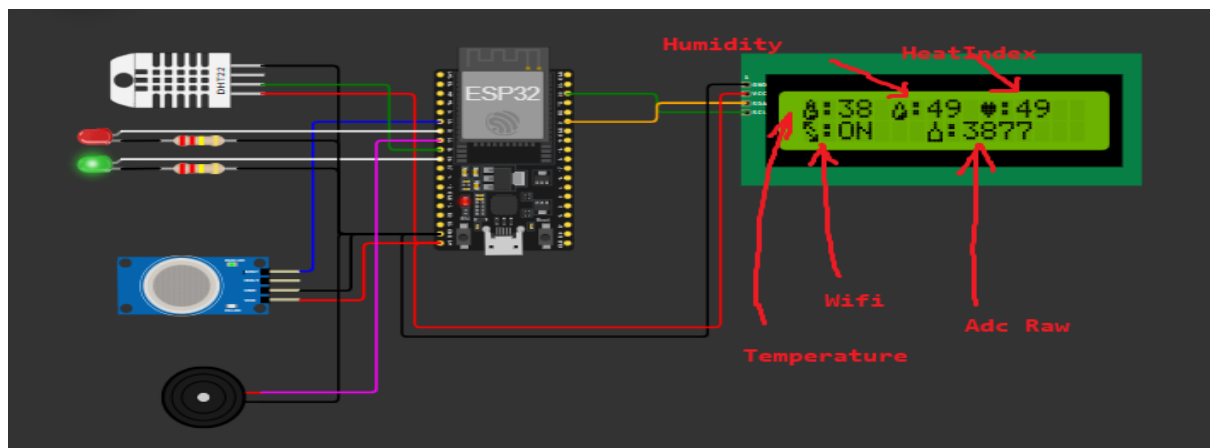


Fig. 29 Afisarea ITU pe interfata principala

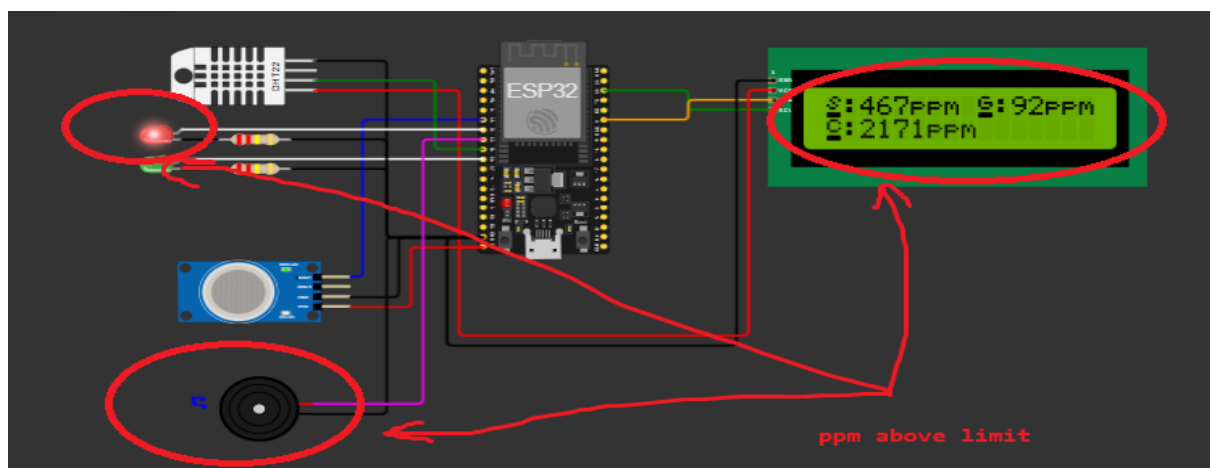
5.Example de functionare



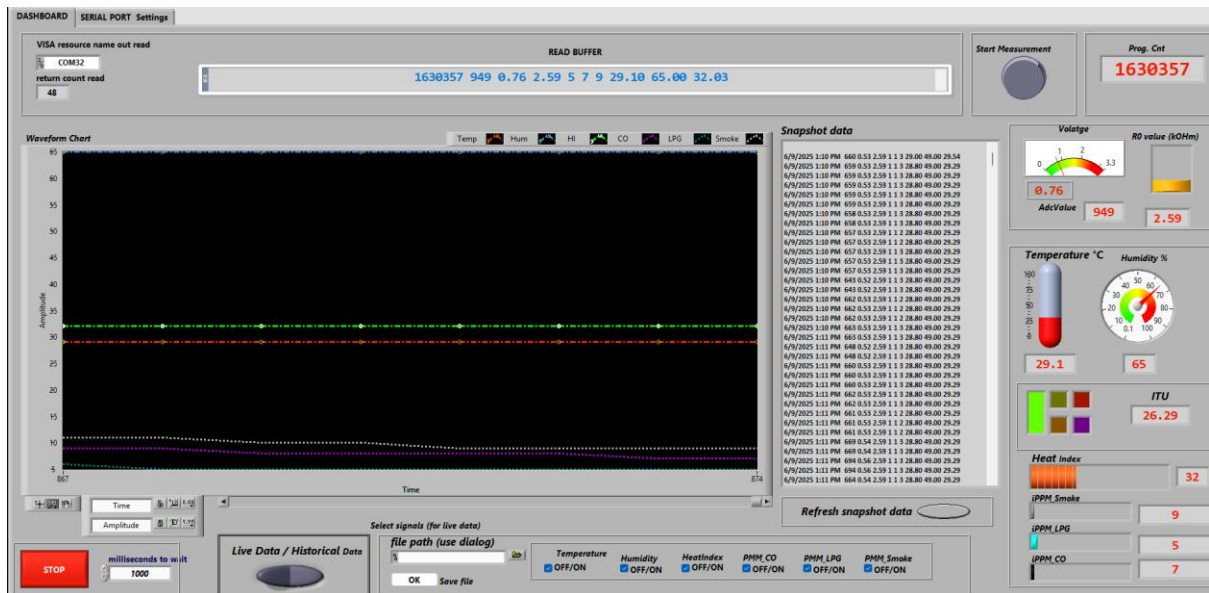
Exemplu 1: Simulare Wokwi parametri gaz normali



Exemplu 2: Simulare Wokwi inainte de declansare alarma



Exemplu 3: Simulare Wokwi alarma declansata



Exemplu 4: Functionare in timp real pe panoul din Labview

```
17:03:46.254 -> [WiFi] Connected to TP-Link_47D4 IP: 192.168.1.208
17:03:46.254 -> [WiFi] Test connection to google.com...
17:03:46.286 -> [WiFi] Connection to google.com succsesful !
```

Exemplu 5: Conectare la Wifi folosind fisierul cu date de conectare



Exemplu 6: Dispozitiv fizic in parametri normali



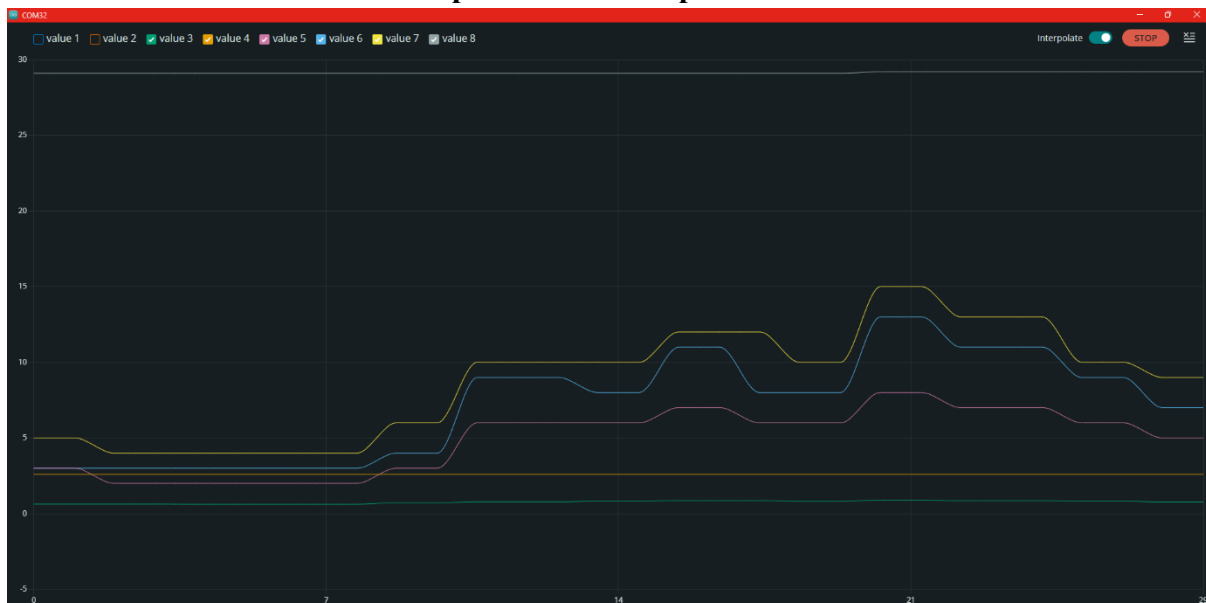
Exemplu 6: Dispozitiv fizic cand limitele concentratiei de gaz sunt depasite

```

13:32:54.374 -> 2024042 699 0.56 2.59 2 2 3 29.20 57.00 30.90
13:32:55.350 -> 2025042 692 0.56 2.59 2 2 3 29.20 57.00 30.90
13:32:56.386 -> 2026042 692 0.56 2.59 2 2 3 29.20 57.00 30.90
13:32:57.350 -> 2027042 692 0.56 2.59 2 2 3 29.20 57.00 30.90
13:32:58.378 -> 2028042 638 0.51 2.59 2 2 3 29.20 57.00 30.90
13:32:59.375 -> 2029042 638 0.51 2.59 2 2 3 29.20 57.00 30.90
13:33:00.350 -> 2030042 690 0.56 2.59 2 2 3 29.20 57.00 30.90
13:33:01.367 -> 2031042 690 0.56 2.59 2 2 3 29.20 57.00 30.90
13:33:02.375 -> 2032042 982 0.79 2.59 5 7 9 29.20 57.00 30.90
13:33:03.383 -> 2033043 982 0.79 2.59 5 7 9 29.20 57.00 30.90
13:33:04.383 -> 2034043 982 0.79 2.59 5 7 9 29.20 57.00 30.90
13:33:05.351 -> 2035043 1003 0.81 2.59 6 8 10 29.20 57.00 30.90
13:33:06.351 -> 2036043 1003 0.81 2.59 6 8 10 29.20 57.00 30.90
13:33:07.381 -> 2037043 1051 0.85 2.59 8 11 12 29.20 57.00 30.90
13:33:08.351 -> 2038043 1051 0.85 2.59 8 11 12 29.20 57.00 30.90
13:33:09.379 -> 2039043 1051 0.85 2.59 8 11 12 29.20 57.00 30.90
13:33:10.351 -> 2040043 1109 0.89 2.59 8 12 14 29.20 57.00 30.90
13:33:11.353 -> 2041043 1109 0.89 2.59 8 12 14 29.20 57.00 30.90

```

Exemplu7: Datele din portul serial



Exemplu 8: Afisarea datelor in serial plotter

6.Anexe

Cod sursa: [CodreanuDan/EnvDataMonitoring_ESP32_Cloud_LabView: Environment data monitoring using ESP32 Mq2 & Dht, Cloud and LabView Interface.](#)

7.Bibliografie

Dataheet senzori si componente:

1. DHT11: [DHT11 Humidity & Temperature Sensor](#)
2. MQ2: [MQ-2 \(Ver1.4\) - Manual.pdf](#)
3. LCD: [I2C_1602_LCD.pdf](#)
4. ESP32: [DOIT ESP32 DevKit V1 Wi-Fi Development Board - Pinout Diagram & Arduino Reference - CIRCUITSTATE Electronics](#)

Interfatare MQ2:

- [Explain MQ2, MQ3 and MQ135 Gas Sensor With Arduino.](#)
- [MQ-2 Smoke/Gas, MQ-3 Alcohol, & MQ-7 Carbon Monoxide sensor modules](#)

Labview load data from csv: [Load and visualize .csv data in LabVIEW](#)

ITU:

- [Administratia Nationala de Meteorologie-Harta vreme extrema](#)
- [Ce este indicele de confort termic \(Heat Index\) și cum se calculează? | Starea Vremii](#)
- [Calculare indice de confort termic cu senzor DHT · One Transistor \[RO\]](#)