

***TP. Hồ Chí Minh,15 tháng 6 năm 2024***

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**

**KHOA ĐIỆN-ĐIỆN TỬ**



**MÔN HỌC: ĐỒ ÁN 1**

**BÁO CÁO ĐỒ ÁN 1**

**GVHD: PGS.TS Phan Văn Ca**

**SINH VIÊN: Nguyễn Đình Nhật Minh**

**MSSV: 21119373**

**NGÀNH: Công Nghệ Kỹ Thuật Máy Tính**

**Ý KIẾN VÀ NHẬN XÉT**

................................................................................................................................................................................................................................................

................................................................................................................................................................................................................................................

................................................................................................................................................................................................................................................

................................................................................................................................................................................................................................................

................................................................................................................................................................................................................................................

................................................................................................................................................................................................................................................

................................................................................................................................................................................................................................................

................................................................................................................................................................................................................................................

................................................................................................................................................................................................................................................

................................................................................................................................................................................................................................................

................................................................................................................................................................................................................................................

................................................................................................................................................................................................................................................

................................................................................................................................................................................................................................................

................................................................................................................................................................................................................................................

Mục lục

[Lời mở đầu 5](#_Toc169249683)

[Chương 1: Tổng quan đề tài 7](#_Toc169249684)

[1.1/Lý do chọn đề tài 7](#_Toc169249685)

[1.2/Mục tiêu 7](#_Toc169249686)

[Chương 2: Cơ sở lý thuyết về các linh kiện 8](#_Toc169249687)

[2.1/ Cảm biến TDS 8](#_Toc169249688)

[2.1.1/ Giới thiệu 8](#_Toc169249689)

[2.1.2/ Thông số kĩ thuật 8](#_Toc169249690)

[2.1.3/ Ứng dụng 9](#_Toc169249691)

[2.2/ Cảm biến DS18B20 9](#_Toc169249692)

[2.2.1/ Giới thiệu 9](#_Toc169249693)

[2.2.2/ Thông số kĩ thuật 10](#_Toc169249694)

[2.2.3/ Ứng dụng 10](#_Toc169249695)

[2.3/ Cảm biến độ đục 10](#_Toc169249696)

[2.3.1/ Giới thiệu 10](#_Toc169249697)

[2.3.2/ Thông số kĩ thuật 11](#_Toc169249698)

[2.3.3/ Ứng dụng 11](#_Toc169249699)

[2.4/ Màn hình OLED 12](#_Toc169249700)

[2.4.1/ Giới thiệu 12](#_Toc169249701)

[2.4.2/ Thông số kỹ thuật: 12](#_Toc169249702)

[2.5/ Esp32 13](#_Toc169249703)

[2.5.1/ Giới thiệu 13](#_Toc169249704)

[2.5.2/ Sơ đồ chân 14](#_Toc169249705)

[2.5.3/ Thông số kỹ thuật: 18](#_Toc169249706)

[2.5.4/ Ứng dụng 18](#_Toc169249707)

[Chương 3: Phân tích và thiết kế hệ thống 20](#_Toc169249708)

[3.1/ Phân tích chức năng của hệ thống 20](#_Toc169249709)

[3.1.1/Chức năng phần cứng của hệ thống 20](#_Toc169249710)

[3.1.2/ Chức năng phần mềm của hệ thống 20](#_Toc169249711)

[3.2/ Sơ đồ khối và chức năng 20](#_Toc169249712)

[3.2.1/ Sơ đồ khối 20](#_Toc169249713)

[3.2.2/ Chức năng và nhiệm vụ từng khối 20](#_Toc169249714)

[3.3/ Sơ đồ nguyên lý 21](#_Toc169249715)

[Chương 4: Thi công và hướng phát triển của hệ thống 22](#_Toc169249716)

[4.1/ Lưu đồ giải thuật 22](#_Toc169249717)

[4.2/ Giới thiệu Arduino IDE 23](#_Toc169249718)

[4.3/ Lập trình code 24](#_Toc169249719)

[4.5/ Code web 29](#_Toc169249720)

[4.5.1/HTML 29](#_Toc169249721)

[4.5.2/CSS 31](#_Toc169249722)

[4.5.3/JavaScript 31](#_Toc169249723)

# Lời mở đầu

Suy giảm chất lượng nước đã trở thành vấn đề toàn cầu được quan tâm khi dân số tăng lên, các hoạt động công nghiệp và nông nghiệp mở rộng và biến đổi khí hậu có nguy cơ gây ra những thay đổi lớn đối với chu trình Thủy văn. Hàng năm, số người chết vì nước không an toàn nhiều hơn do mọi hình thức bạo lực, kể cả chiến tranh. Chất lượng nước là một yếu tố quan trọng trong các ngành công nghiệp và ứng dụng khác nhau.

Một trong những chỉ số quan trọng nhất của nước là TDS và độ đục trong nước

TDS (Total Dissolved Solids): là chỉ số thể hiện tổng chất rắn hòa tan (tổng số các ion mang điện tích, bao gồm muối, khoáng chất hoặc kim loại) tồn tại trong một thể tích nước nhất định. Những chất có trong nước chủ yếu là khoáng chất, muối, chất hữu cơ, các hợp chất vô cơ như kim loại nặng - chất rắn lơn lửng không lắng hoặc không hòa tan trong nước (canxi, magiê, natri, kali và các anion cacbonat, bicarbonate, clorua, sunfat). Không bao gồm các chất hữu cơ có tự nhiên trong nước và môi trường, một số hợp chất có thể cần thiết cho cơ thể, nhưng, có thể gây hại khi dùng nhiều hơn hàm lượng được khuyến nghị. TDS được biểu thị bằng đơn vị mg/L hoặc ppm (parts per million - phần triệu). Chất rắn hòa tan trong nước đến từ các nguồn nước tự nhiên, nước thải đô thị, nước thải công nghiệp và hóa chất được sử dụng trong quá trình xử lý nước hoặc rỉ sét từ đường ống được sử dụng để dẫn nước. Tổng chất rắn hòa tan có trong nước là một trong những nguyên nhân gây ra độ đục và trầm tích trong nước uống. Khi không được lọc, tổng chất rắn hòa tan có thể là nguyên nhân gây ra nhiều bệnh lý khác nhau. Vậy nên giám sát mức tổng chất rắn hòa tan (TDS) và nhiệt độ là điều cần thiết để đảm bảo an toàn cho nước và sự phù hợp của nước cho các mục đích khác nhau.

Độ đục chỉ là sự vẩn đục của nước. Sự vẩn đục xuất phát từ các hạt lơ lửng trong nước mà chúng ta có thể nhìn thấy chúng riêng lẻ. Những hạt này có thể là tảo, bụi bẩn, khoáng chất, protein, dầu hoặc thậm chí là vi khuẩn. Độ đục là một phép đo quang chỉ ra sự hiện diện của các hạt lơ lửng. Nó được đo bằng cách chiếu ánh sáng qua một mẫu và định lượng nồng độ hạt lơ lửng. Khi có càng nhiều hạt trong dung dịch, độ đục càng cao. Theo QCVN01:2009/BYT giới hạn tối đa cho phép độ đục của nước là 2NTU. Việc kiểm tra, kiểm soát chỉ số này trong nước ăn, nước sinh hoạt hàng ngày, nước bể bơi,…là điều cần thiết hơn bao giờ hết bởi nó có thể gây ra nhiều hậu quả. Trước hết, độ đục của nước cao sẽ ảnh hưởng đến tính thẩm mỹ của nước. Nó cũng là dấu hiệu cho thấy khả năng ảnh hưởng đến sức khỏe con người. Độ đục có thể là nguồn thức ăn và là nơi cư trú của nhiều mầm bệnh nguy hiểm. Nếu không kiểm tra và xử lý kịp thời, độ đục này là nguyên nhân thúc đẩy sự phát triển của mầm bệnh trong hệ thống phân phối, dẫn đến sự bùng phát dịch bệnh do sử dụng nguồn nước có độ đục cao. Mặc dù không phải là chỉ báo trực tiếp về nguy cơ ảnh hưởng đến sức khỏe nhưng các nghiên cứu đã cho thấy mối quan hệ giữa loại bỏ độ đục và loại bỏ động vật nguyên sinh. Các hạt của độ đục là chỗ dựa để vi khuẩn cư trú, từ đó làm giảm hiệu quả khử trùng nước.

Trong dự án này, chúng ta sẽ phát triển một mô hình IoT được xây dựng từ vi điều khiển ESP32, chúng tôi có thể theo dõi mức TDS, Giá trị EC (Độ dẫn điện), độ đục của nước và nhiệt độ nước trong thời gian thực và hiển thị chúng lên OLED để người dùng có thể dễ dàng theo dõi trực tiếp. Ngoài ra các thông số trên còn được hiển thị lên WEB để người dùng có thể theo dõi từ xa

# Chương 1: Tổng quan đề tài

## 1.1/Lý do chọn đề tài

Chất lượng nước luôn là một vấn đề nhức nhối trong xã hội hiện nay. Đi cùng với sự phát triển của xã hội là sự gia tăng các nhà máy, phòng thí nghiệm, khu dân cư và dần dần hệ thống xử lý nước thải công nghiệp hay đôi thị trở nên quá tải rồi các chất thải, nước thải được xả trực tiếp vào mội trường.

## 1.2/Mục tiêu

Giám sát chất lượng nước để đảm bảo nước uống an toàn, duy trì điều kiện tối ưu cho sinh vật dưới nước, tối ưu hóa các biện pháp tưới tiêu, đánh giá tác động môi trường và đáp ứng các tiêu chuẩn công nghiệp.

* 1. Giới thiệu về đề tài

Yêu cầu chức năng hệ thống

* Theo dõi nhiệt độ
* Theo dõi độ dẫn điện trong nước
* Theo dõi nồng độ chất rắn hòa tan
* Theo dõi độ đục
* Hiển thị các giá trị đo được lên Oled
* Cập nhật lên web và firebase theo thời gian thực

Nhiệm vụ thiết kế

* Dùng cảm biến TDS để đo độ dẫn điện và nồng độ chất rắn hòa tan
* Dùng cảm biến DS18B20 để đo nhiệt độ của nước
* Dùng cảm biến Turbidity để đo độ đục của nước
* Sử dụng màn hình OLED để hiển thị các giá trị đo được

Linh kiện sử dụng

* OLED
* DS18B20
* Cảm biến TDS
* Cảm biến Turbidity
* Esp32

# Chương 2: Cơ sở lý thuyết về các linh kiện

## 2.1/ Cảm biến TDS

### 2.1.1/ Giới thiệu

Cảm biến TDS dung để đo giá trị TDS của nước. Nó có thể được áp dụng cho nước sinh hoạt, thủy canh và các lĩnh vực kiểm tra chất lượng nước khác. Sản phẩm này hỗ trợ đầu vào điện áp rộng 3,3 ~ 5,5V và đầu ra điện áp tương tự 0 ~ 2,3V , giúp sản phẩm tương thích với các hệ thống hoặc bảng điều khiển 5V hoặc 3,3V.



*Hình 1: Cảm biến TDS*

Nguồn kích thích là tín hiệu AC , có thể ngăn chặn sự phân cực của đầu dò một cách hiệu quả và kéo dài tuổi thọ của đầu dò, đồng thời có thể giúp tăng độ ổn định của tín hiệu đầu ra. Đầu **dò TDS** không thấm nước, có thể ngâm trong nước để đo trong thời gian dài.

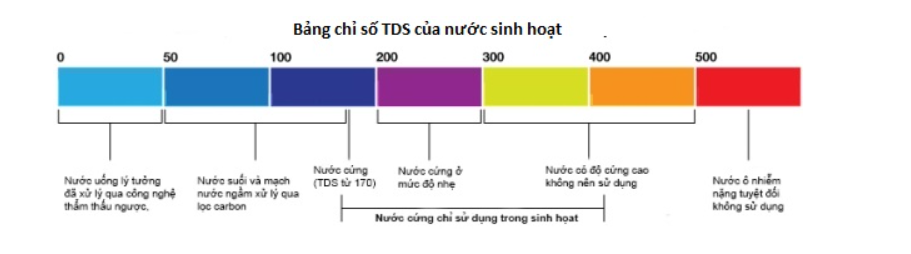
Cảm biến có Phạm vi đo TDS trong khoảng 0 ~ 1000ppm với độ chính xác ±5%F.S. (25oC). Không thể sử dụng đầu dò trong nước có nhiệt độ trên 55 độ C.

### 2.1.2/ Thông số kĩ thuật

|  |  |
| --- | --- |
|  | TDS |
| Điện áp cấp nguồn | 3.3-5V |
| Khoảng giá trị đo được | 0-1000ppm |
| Sai số | 5%F.S(25 độ C) |
| Điện áp ngõ ra tương ứng | 0-2.3VDC |
| Kích thước | 42 x 31.2mm |

### 2.1.3/ Ứng dụng

Kiểm tra chất lượng nước ở sông ngòi, nước sinh hoạt, phòng thí nghiệm, quá trình xử lý nước thải hay bất kì ứng dụng nào cần chỉ số TDS



*Hình 2: Bảng chỉ số TDS*

## 2.2/ Cảm biến DS18B20

### 2.2.1/ Giới thiệu

Đây là phiên bản có sẵn dây và chống thấm nước của cảm biến DS18B20 được sử dụng để đo vật ở xa hoặc kể cả ở trong nước. Cảm biến có thể đo nhiệt độ trong khoảng từ -55 đến 125°C (-67°F đến +257°F). Cáp được bọc bằng nhựa PVC. Những cảm biến nhiệt độ kỹ thuật số 1 dây này khá chính xác, tức là ±0,5°C trên hầu hết phạm vi. Chúng hoạt động tốt với bất kỳ bộ vi điều khiển nào sử dụng một chân kỹ thuật số duy nhất.

[](https://how2electronics.com/wp-content/uploads/2018/12/waterproof-temperature-sensor-ds18b20.jpg)

*Hình 3: Cảm biến nhiệt độ DS18B20*

Cảm biến yêu cầu hai thư viện như Thư viện cảm biến nhiệt độ Dallas & Thư viện một dây . Nó cũng yêu cầu điện trở 4,7k , được yêu cầu để kéo từ DATA sang đường VCC khi sử dụng cảm biến.

### 2.2.2/ Thông số kĩ thuật

|  |  |
| --- | --- |
|  | DS18B20 |
| Điện áp cấp nguồn | 3.0-5V |
| Phạm vi nhiệt độ | -55 đến 125 độ C |
| Sai số | +- 0.5 độ C |
| Chuẩn giao tiếp | 1 wire |

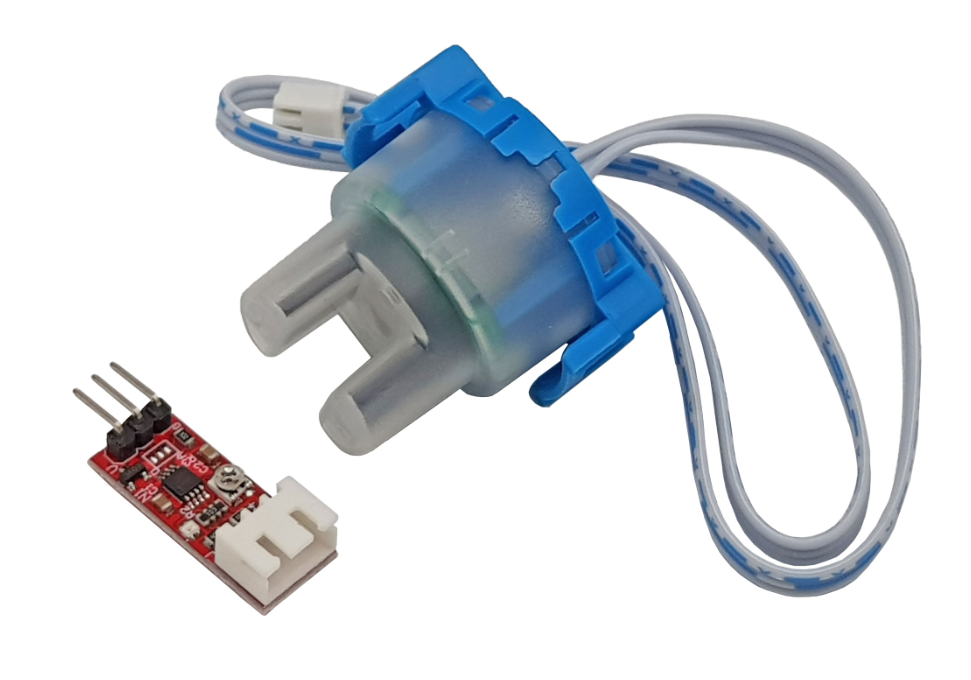
### 2.2.3/ Ứng dụng

Kiểm soát nhiệt môi trường, đo nhiệt độ bên trong các tòa nhà, thiết bị, máy móc, nhiệt độ chất lỏng và trong hệ thống giám sát.

## 2.3/ Cảm biến độ đục

### 2.3.1/ Giới thiệu

Độ đục là một thuộc tính của các hạt nước. Cảm biến đo độ đục của nước đọc và xuất ra giá trị analog tương ứng với giá trị của độ đục. Độ đục này cần hiệu chỉnh về giá trị chuẩn thông qua biến trở để so sánh.



*Hình 4: Cảm biến độ đục*

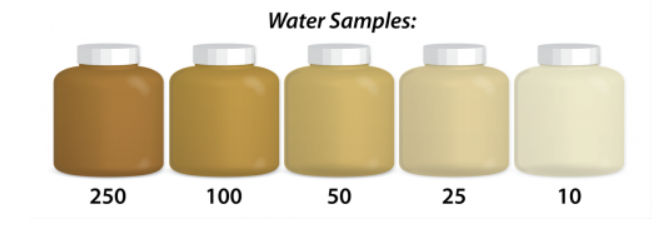
Nguyên lí của mạch là thay vì đo điện áp trực tiếp ở các độ đục khác nhau, ta có thể dung cảm biến đo độ đục này để xuất ra  các giá trị điện áp tương ứng từ 0-4.5V

### 2.3.2/ Thông số kĩ thuật

|  |  |
| --- | --- |
|  | DS18B20 |
| Điện áp cấp nguồn | 3.5 - 5V |
| Phạm vi | 0 - 1000NTU |
| Điện áp đầu ra | 0 - 4.5V |

### 2.3.3/ Ứng dụng

Kiểm soát nhiệt môi trường, đo nhiệt độ bên trong các tòa nhà, thiết bị, máy móc, nhiệt độ chất lỏng và trong hệ thống giám sát.



*Hình 4: Độ đục của nước*

## 2.4/ Màn hình OLED

### 2.4.1/ Giới thiệu



*Hình 5: Màn hình OLED*

Module OLED là một thiết bị điện tử nhỏ gọn có thể được sử dụng để hiển thị văn bản, hình ảnh và video. Module OLED được tạo thành từ một tấm nền OLED (diode phát quang hữu cơ) và một bộ điều khiển. Tấm nền OLED bao gồm một lớp vật liệu hữu cơ có thể phát sáng khi có dòng điện chạy qua. Bộ điều khiển chịu trách nhiệm điều khiển các điểm ảnh trên tấm nền OLED. Module OLED có một số ưu điểm so với các loại màn hình khác, bao gồm:

Màu sắc sống động: Module OLED có thể tạo ra màu sắc sống động và chân thực.

Độ tương phản cao: Module OLED có độ tương phản cao, cho phép hiển thị rõ ràng các hình ảnh và văn bản trong điều kiện ánh sáng yếu.

Góc nhìn rộng: Module OLED có góc nhìn rộng, cho phép người dùng xem hình ảnh và văn bản rõ ràng từ nhiều góc độ khác nhau.

### 2.4.2/ Thông số kỹ thuật:

|  |  |
| --- | --- |
|  | Màn hình OLED |
| Hoạt động ổn định | 3V3 - 5V |
| Độ phân giải | 128X64 |
| Số chân hoạt động | 4 chân |
| Chất lượng hình ảnh hiển thị | Dễ nhìn |
| Màu hiển thị | Xanh dương |
| Loại giao tiếp | I2C |

Module OLED được sử dụng trong nhiều ứng dụng khác nhau, bao gồm:

Điện thoại di động: Module OLED được sử dụng trong màn hình điện thoại di động, cho phép hiển thị hình ảnh và video chất lượng cao.

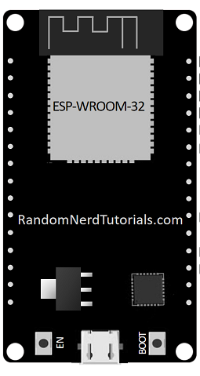
Đồng hồ thông minh: Module OLED được sử dụng trong màn hình đồng hồ thông minh, cho phép hiển thị thời gian và thông tin khác rõ ràng và trực quan.

Thiết bị đeo được: Module OLED được sử dụng trong màn hình của các thiết bị đeo được khác nhau, chẳng hạn như vòng đeo tay thông minh và kính thông minh.

Máy chơi game: Module OLED được sử dụng trong màn hình của máy chơi game, ví dụ như rắn săn mồi,…

## 2.5/ Esp32

### 2.5.1/ Giới thiệu

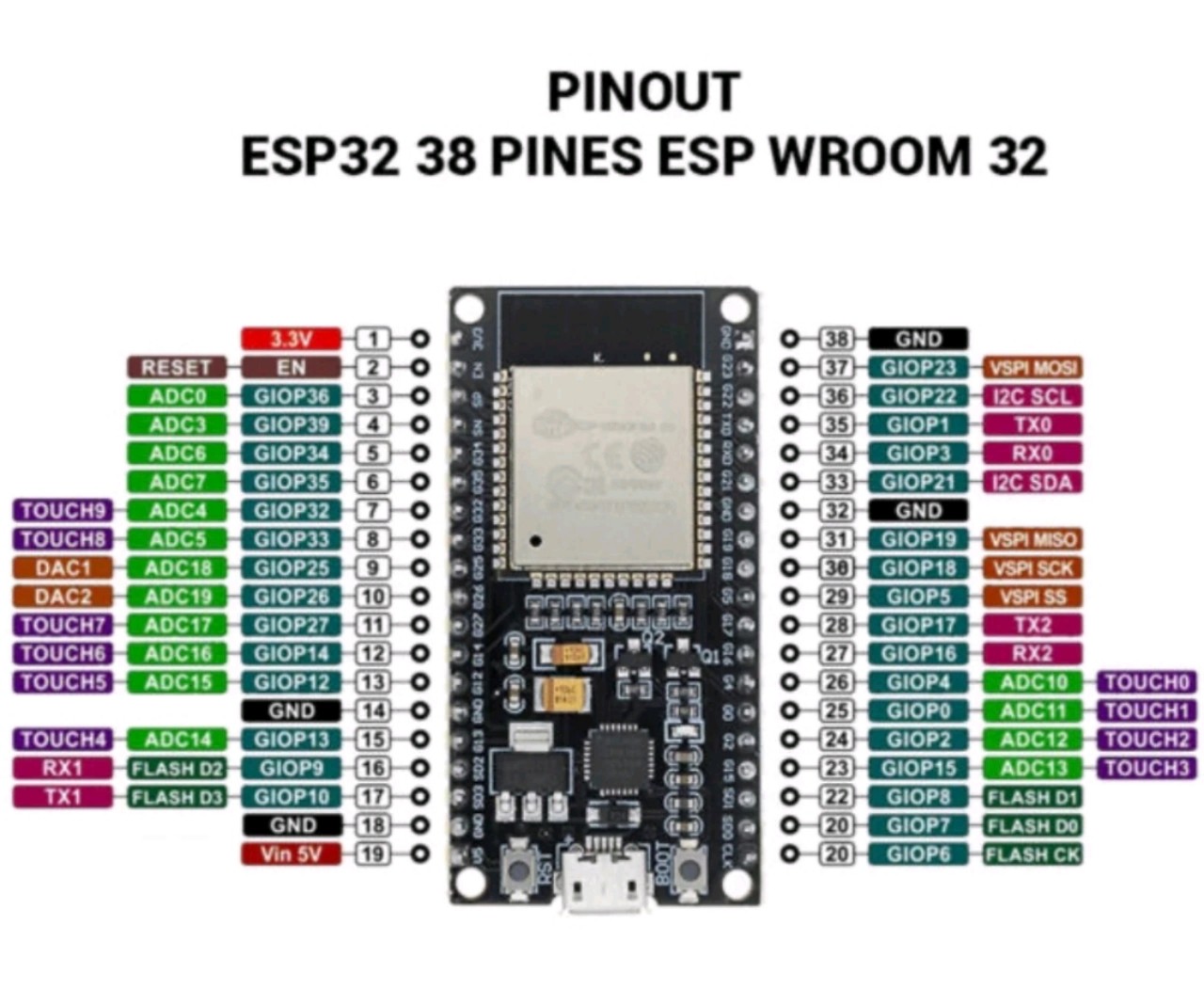


*Hình 6: Vi điều khiển ESP32*

ESP32 là một series các vi điều khiển trên một vi mạch (SoC) giá rẻ, năng lượng thấp có hỗ trợ WiFi và dual-mode Bluetooth. ESP32 được phát triển bởi Espressif Systems, một công ty công nghệ Trung Quốc. ESP32 có các tính năng sau:

* Hai lõi xử lý Xtensa LX6 với tốc độ lên đến 240 MHz
* 520 KiB SRAM
* 4 MB bộ nhớ flash
* Hỗ trợ Wi-Fi 802.11 b/g/n/e/i (802.11n @ 2.4 GHz lên đến 150 Mbit/s)
* Hỗ trợ Bluetooth v4.2 BR/EDR và Bluetooth Low Energy (BLE)
* Hỗ trợ nhiều giao thức truyền thông, bao gồm TCP/IP, UDP, HTTP, MQTT,...
* Hỗ trợ nhiều cảm biến, bao gồm cảm biến nhiệt độ, cảm biến ánh sáng, cảm biến gia tốc,..

### 2.5.2/ Sơ đồ chân



*Hình 7: Sơ đồ chân ESP32*

**Mô tả các chân ESP32**

**Chân Input Only**

GPIO từ 34 đến 39 là GPI – chân chỉ đầu vào. Các chân này không có điện trở kéo lên hoặc kéo xuống bên trong.

Chúng không thể được sử dụng làm đầu ra, vì vậy chỉ sử dụng các chân này làm đầu vào:

* GPIO 34
* GPIO 35
* GPIO 36
* GPIO 39

**Chân tích hợp Flash trên ESP32**

GPIO 6 đến GPIO 11 dùng để kết nối Flash SPI, không khuyến khích sử dụng trong các ứng dụng khác

* GPIO 6 (SCK/CLK)
* GPIO 7 (SDO/SD0)
* GPIO 8 (SDI/SD1)
* GPIO 9 (SHD/SD2)
* GPIO 10 (SWP/SD3)
* GPIO 11 (CSC/CMD)

**Chân cám biến điện dung**

Các chân ESP32 này có chức năng như 1 nút nhấn cảm ứng, có thể phát hiện sự thay đổi về điện áp cảm ứng trên chân.

Các cảm biến cảm ứng bên trong đó được kết nối với các GPIO sau:

* T0 (GPIO 4)
* T1 (GPIO 0)
* T2 (GPIO 2)
* T3 (GPIO 15)
* T4 (GPIO 13)
* T5 (GPIO 12)
* T6 (GPIO 14)
* T7 (GPIO 27)
* T8 (GPIO 33)
* T9 (GPIO 32)

Analog to Digital Converter (ADC)

ESP32 có các kênh đầu vào ADC 18 x 12 bit (trong khi ESP8266 chỉ có ADC 1x 10 bit).

Đây là các GPIO có thể được sử dụng làm ADC và các kênh tương ứng:

* ADC1\_CH0 (GPIO 36)
* ADC1\_CH1 (GPIO 37)
* ADC1\_CH2 (GPIO 38)
* ADC1\_CH3 (GPIO 39)
* ADC1\_CH4 (GPIO 32)
* ADC1\_CH5 (GPIO 33)
* ADC1\_CH6 (GPIO 34)
* ADC1\_CH7 (GPIO 35)
* ADC2\_CH0 (GPIO 4)
* ADC2\_CH1 (GPIO 0)
* ADC2\_CH2 (GPIO 2)
* ADC2\_CH3 (GPIO 15)
* ADC2\_CH4 (GPIO 13)
* ADC2\_CH5 (GPIO 12)
* ADC2\_CH6 (GPIO 14)
* ADC2\_CH7 (GPIO 27)
* ADC2\_CH8 (GPIO 25)
* ADC2\_CH9 (GPIO 26)

Các kênh đầu vào ADC có độ phân giải 12 bit. Điều này có nghĩa là bạn có thể nhận được các số đọc tương tự từ 0 đến 4095, trong đó 0 tương ứng với 0V và 4095 đến 3,3V.

Bạn cũng có thể lập trình độ phân giải của các kênh của mình trên code.

**Digital to Analog Converter (DAC)**

Có các kênh DAC 2 x 8 bit trên ESP32 để chuyển đổi tín hiệu kỹ thuật số thành đầu ra tín hiệu điện áp tương tự.

Các kênh này chỉ có độ phân giải 8 bit, nghĩa là có giá trị từ 0 – 255 tương ứng với 0 – 3.3V

Đây là các kênh DAC:

* DAC1 (GPIO25)
* DAC2 (GPIO26)

**Các chân thời gian thực RTC**

Các chân này có tác dụng đánh thức ESP32 khi trong chế độ Low Power Mode. Sử dụng như 1 chân ngắt ngoài.

Các chân RTC:

* RTC\_GPIO0 (GPIO36)
* RTC\_GPIO3 (GPIO39)
* RTC\_GPIO4 (GPIO34)
* RTC\_GPIO5 (GPIO35)
* RTC\_GPIO6 (GPIO25)
* RTC\_GPIO7 (GPIO26)
* RTC\_GPIO8 (GPIO33)
* RTC\_GPIO9 (GPIO32)
* RTC\_GPIO10 (GPIO4)
* RTC\_GPIO11 (GPIO0)
* RTC\_GPIO12 (GPIO2)
* RTC\_GPIO13 (GPIO15)
* RTC\_GPIO14 (GPIO13)
* RTC\_GPIO15 (GPIO12)
* RTC\_GPIO16 (GPIO14)
* RTC\_GPIO17 (GPIO27)

**Chân PWM**

ESP32 LED PWM có 16 kênh độc lập có thể được định cấu hình để tạo tín hiệu PWM với các thuộc tính khác nhau.

Tất cả các chân có thể hoạt động như đầu ra đều có thể được sử dụng làm chân PWM (GPIO từ 34 đến 39 không thể tạo PWM).

Để xuất PWM, bạn cần xác định các thông số này trong code:

* Frequency – tần số
* Duty cycle
* Kênh PWM
* Chân GPIO nơi bạn muốn xuất tín hiệu

**Chân I2C**

ESP32 có hai kênh I2C và bất kỳ chân nào cũng có thể được đặt làm SDA hoặc SCL. Khi sử dụng ESP32 với Arduino IDE, các chân I2C mặc định là:

* GPIO 21 (SDA)
* GPIO 22 (SCL)

Nếu các bạn muốn sử dụng chân khác cho việc điều khiển I2C có thể sử dụng code:

Wire.begin(SDA, SCL);

**Chân Ngắt Ngoài**

Tất cả các chân ESP32 đều có thể sử dụng ngắt ngoài

### 2.5.3/ Thông số kỹ thuật:

|  |  |
| --- | --- |
|  | ESP32 |
| Xung nhịp hoạt động | 240 MHz |
| Bộ nhớ flash | 4 MB |
| Bộ nhớ ROM | 448 KB |
| Bộ nhớ RAM | 520 KB |
| Kết nối | WiFi 2,4 GHz / Bluetooth / BLE |
| Số lõi | 2 lõi riêng |

### 2.5.4/ Ứng dụng

ESP32 được sử dụng trong nhiều ứng dụng khác nhau, bao gồm:

* Thiết bị IoT: ESP32 có thể được sử dụng để thu thập và truyền dữ liệu từ các thiết bị IoT. Ví dụ, ESP32 có thể được sử dụng để thu thập dữ liệu từ các cảm biến môi trường, dữ liệu từ các thiết bị sản xuất, ...
* Thiết bị nhà thông minh: ESP32 có thể được sử dụng để điều khiển các thiết bị nhà thông minh từ xa. Ví dụ, ESP32 có thể được sử dụng để bật tắt đèn, điều khiển điều hòa,...
* Thiết bị đeo: ESP32 có thể được sử dụng để tạo ra các thiết bị đeo, chẳng hạn như đồng hồ thông minh, vòng đeo tay theo dõi sức khỏe,...
* Thiết bị công nghiệp: ESP32 có thể được sử dụng để tạo ra các thiết bị công nghiệp, chẳng hạn như cảm biến, máy móc,...
* Thiết bị giải trí: ESP32 có thể được sử dụng để tạo ra các thiết bị giải trí, chẳng hạn như loa thông minh, máy chơi game,...

# Chương 3: Phân tích và thiết kế hệ thống

## 3.1/ Phân tích chức năng của hệ thống

### 3.1.1/Chức năng phần cứng của hệ thống

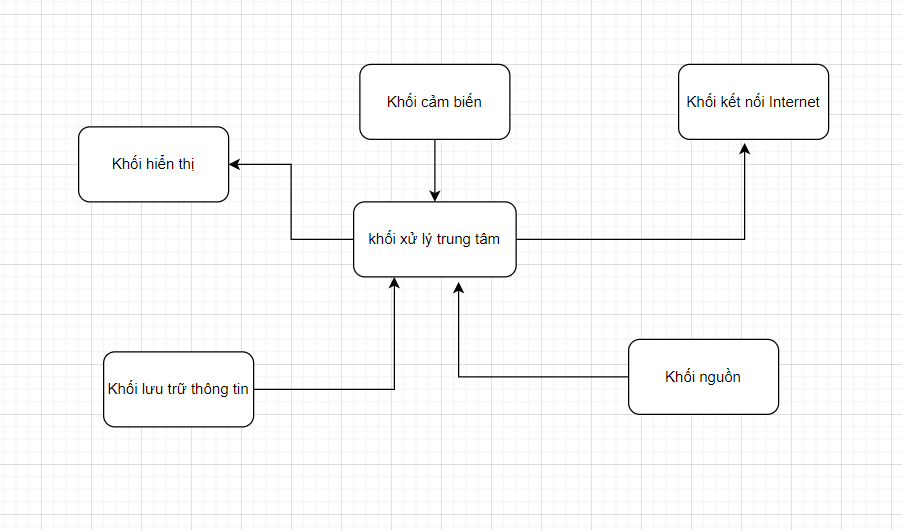
* Thu thập dữ liệu nhiệt độ, nồng độ chất rắn hòa tan, độ dẫn điện và độ đục của nước
* Hiển thị các dữ liệu đo được lên OLED

### 3.1.2/ Chức năng phần mềm của hệ thống

* Gửi các dữ liệu đo được lên firebase
* Hiển thị dữ liệu đo được lên web và vẽ đồ thị theo thời gian thực

## 3.2/ Sơ đồ khối và chức năng

### 3.2.1/ Sơ đồ khối

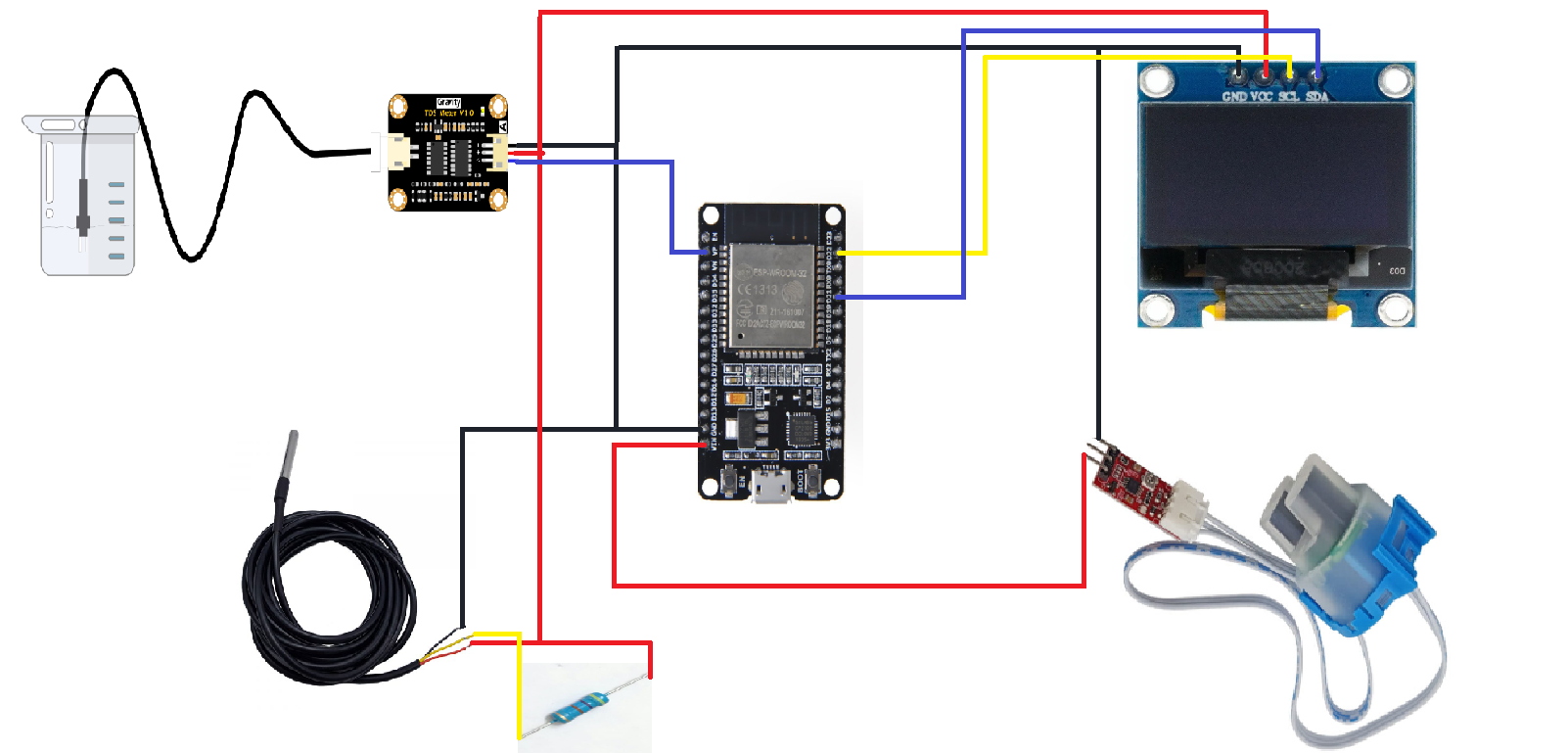


*Hình 8: Sơ đồ khối*

### 3.2.2/ Chức năng và nhiệm vụ từng khối

* Khối nguồn: Tạo ra nguồn cung cấp và dòng điện ổn định cung cấp cho toàn mạch hoạt động.
* Khối xử lý trung tâm: Chịu trách nhiệm xử lý các dữ liệu và đưa ra quyết định điều khiển hoạt động của hệ thống.
* Khối hiển thị: Dùng để hiện thông tin từ khối nhận dữ liệu
* Khối cảm biến: Thu thập thông tin người dùng nhập vào để từ đó so sánh và đưa ra quyết định đóng, mở cửa..
* Khối kết nối Internet: Nhận dữ liệu điều khiển từ xa

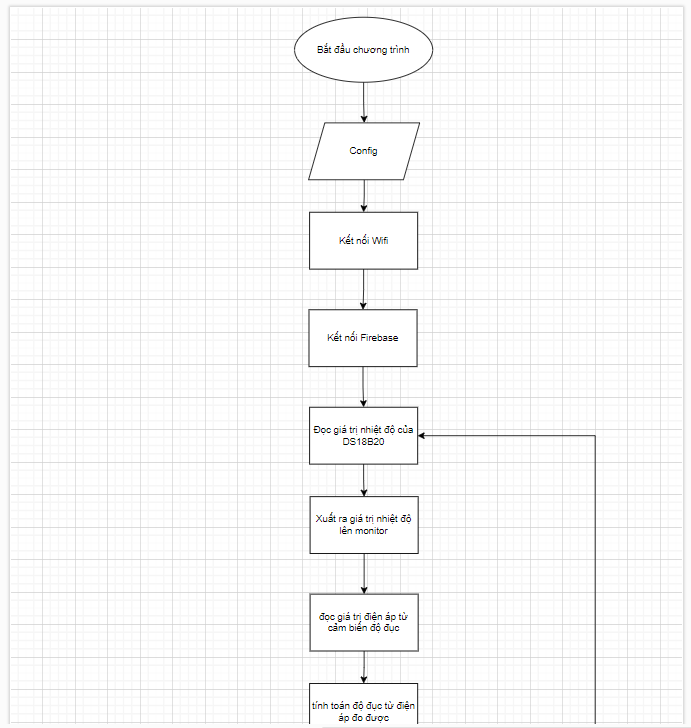
## 3.3/ Sơ đồ nguyên lý

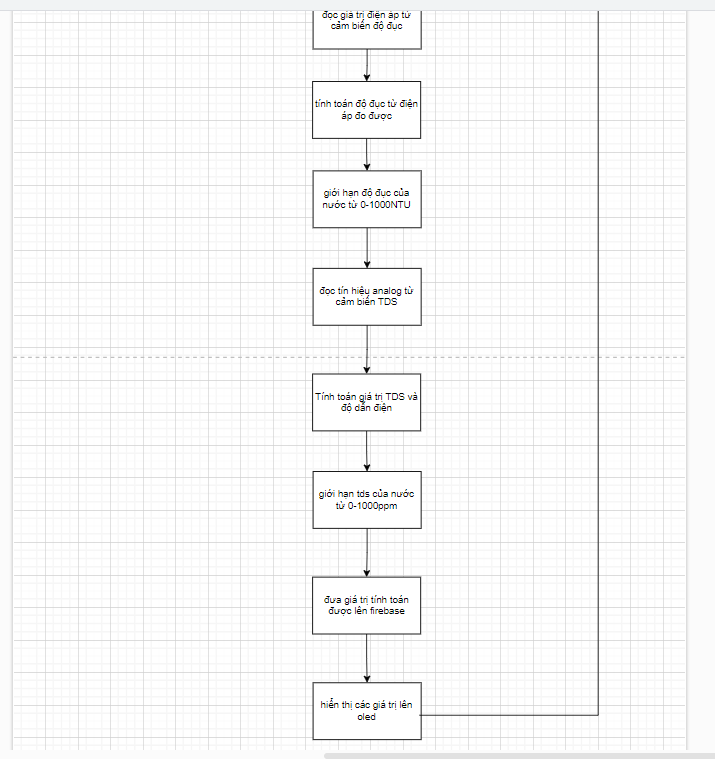


*Hình 9: Sơ đồ nguyên lý*

# Chương 4: Thi công và hướng phát triển của hệ thống

## 4.1/ Lưu đồ giải thuật



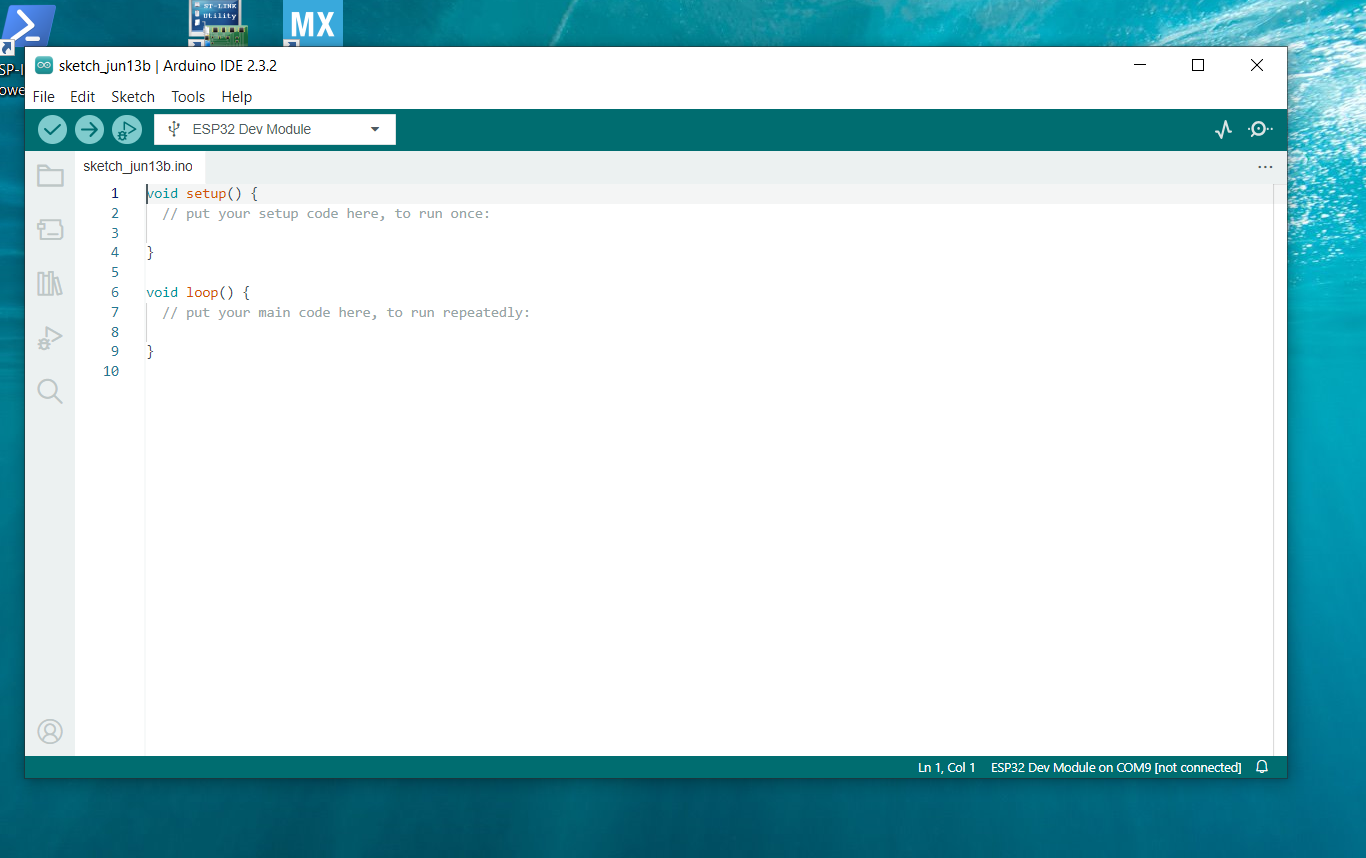


*Hình 10: Lưu đồ thuật toán*

## 4.2/ Giới thiệu Arduino IDE

Arduino IDE là một môi trường phát triển tích hợp (IDE) mã nguồn mở được sử dụng để lập trình các thiết bị Arduino. Nó được phát triển bởi Arduino.cc và được sử dụng bởi hàng triệu người trên toàn thế Giới. Arduino IDE cung cấp các công cụ cần thiết để tạo, biên dịch và tải mã lên các thiết bị Arduino. Nó bao gồm một trình soạn thảo văn bản, trình biên dịch, trình gỡ lỗi và một số công cụ khác. Trình soạn thảo văn bản của Arduino IDE cho phép bạn tạo mã Arduino. Nó hỗ trợ các tính năng như tô sáng cú pháp, hoàn thành tự động và kiểm tra lỗi. Trình biên dịch của Arduino IDE biến mã Arduino thành mã máy có thể được thực thi bởi các thiết bị Arduino. Trình gỡ lỗi của Arduino IDE cho phép bạn tìm và sửa lỗi trong mã Arduino của mình. Arduino IDE là một công cụ mạnh mẽ và dễ sử dụng giúp bạn bắt đầu với Arduino. Nó là một lựa chọn tuyệt vời cho các nhà phát triển mới bắt đầu và chuyên nghiệp. Dưới đây là một số tính năng chính của Arduino IDE:

* Trình soạn thảo văn bản: Trình soạn thảo văn bản của Arduino IDE hỗ trợ các tính năng như tô sáng cú pháp, hoàn thành tự động và kiểm tra lỗi.
* Trình biên dịch: Trình biên dịch của Arduino IDE biến mã Arduino thành mã máy có thể được thực thi bởi các thiết bị Arduino.
* Trình gỡ lỗi: Trình gỡ lỗi của Arduino IDE cho phép bạn tìm và sửa lỗi trong mã Arduino của mình.
* Trình quản lý thư viện: Trình quản lý thư viện của Arduino IDE cung cấp cho bạn quyền truy cập vào một thư viện lớn các thư viện Arduino.



*Hình 11: Arduino IDE*

## 4.3/ Lập trình code

#include <Firebase\_ESP\_Client.h>

#include <SPI.h>

#include <Wire.h>

#include <Adafruit\_GFX.h>

#include <Adafruit\_SH110X.h>

#include <OneWire.h>

#include <DallasTemperature.h>

#define i2c\_Address 0x3c //initialize with the I2C addr 0x3C Typically eBay OLED's

//#define i2c\_Address 0x3d //initialize with the I2C addr 0x3D Typically Adafruit OLED's

#define SCREEN\_WIDTH 128 // OLED display width, in pixels

#define SCREEN\_HEIGHT 64 // OLED display height, in pixels

#define OLED\_RESET -1    // QT-PY / XIAO

Adafruit\_SH1106G display = Adafruit\_SH1106G(SCREEN\_WIDTH, SCREEN\_HEIGHT, &Wire, OLED\_RESET);

//Provide the token generation process info.

#include "addons/TokenHelper.h"

//Provide the RTDB payload printing info and other helper functions.

#include "addons/RTDBHelper.h"

// Insert your network credentials

#define WIFI\_SSID "Redmi Note 10S"

#define WIFI\_PASSWORD "123456789"

#define DATABASE\_URL "https://iot2-37175-default-rtdb.firebaseio.com/"

#define API\_KEY "AIzaSyBRcKbTuTnJ0ljvkK4OQQFpiI\_jXY8XOZI"

//Define Firebase Data object

FirebaseData fbdo;

FirebaseAuth auth;

FirebaseConfig config;

// GPIO where the DS18B20 is connected to

const int oneWireBus = 4;

int TDS\_Sensor = A0;

int sensorPin = 36;

float Aref = 3.3;

float ec\_Val = 0;

unsigned int tds\_value = 0;

float ecCal = 1;

double wTemp;

bool signupOK = false;

float volt;

float ntu;

long ntu\_standardized;

// Setup a oneWire instance to communicate with any OneWire devices

OneWire oneWire(oneWireBus);

// Pass our oneWire reference to Dallas Temperature sensor

DallasTemperature sensors(&oneWire);

void setup() {

  // Start the Serial Monitor

  Serial.begin(115200);

  delay(250); // wait for the OLED to power up

  display.begin(i2c\_Address, true); // Address 0x3C default

  display.display();

  delay(2000);

  display.clearDisplay();

  // Start the DS18B20 sensor

  sensors.begin();

  WiFi.begin(WIFI\_SSID, WIFI\_PASSWORD);

  Serial.print("Connecting to Wi-Fi");

  while (WiFi.status() != WL\_CONNECTED){

    Serial.print(".");

    delay(100);

  }

  Serial.println();

  Serial.print("Connected with IP: ");

  Serial.println(WiFi.localIP());

  Serial.println();

  /\* Assign the api key (required) \*/

  config.api\_key = API\_KEY;

  /\* Assign the RTDB URL (required) \*/

  config.database\_url = DATABASE\_URL;

  /\* Sign up \*/

  if (Firebase.signUp(&config, &auth, "", "")){

    Serial.println("ok");

    signupOK = true;

  }

  else{

    Serial.printf("%s\n", config.signer.signupError.message.c\_str());

  }

  /\* Assign the callback function for the long running token generation task \*/

  config.token\_status\_callback = tokenStatusCallback;

  Firebase.begin(&config, &auth);

  Firebase.reconnectWiFi(true);

}

void loop() {

  sensors.requestTemperatures();

  float temperatureC = sensors.getTempCByIndex(0);

  float temperatureF = sensors.getTempFByIndex(0);

  Serial.print(temperatureC);

  Serial.println("ºC");

  Serial.print(temperatureF);

  Serial.println("ºF");

  volt = 0;

  wTemp = temperatureC;

  float V\_level= Aref / 1024.0;

  float rawEc = analogRead(TDS\_Sensor) \* V\_level;  // Raw  data of EC

  float T\_Cof = 1.0 + 0.02 \* (wTemp - 25.0);  // Temperature Coefficient

  for(int i = 0; i < 800; i++)

  {

      volt += ((float)analogRead(sensorPin) / 1023) \* 5;

  }

  volt = volt / 800 / 667 \* 140;

  volt = round\_to\_dp(volt, 2);

  // if(volt < 2.5){

  //   ntu = 3000;

  // }else{

    ntu = -1120.4 \* volt \* volt + 5742.3 \* volt - 4353.8;

  // }

  // Quy chuẩn NTU từ 0 tới 1000

  ntu\_standardized = map(ntu, 0, 3000, 0, 1000);

  ntu\_standardized = constrain(ntu\_standardized, 0, 1000); // Đảm bảo giá trị nằm trong khoảng từ 0 tới 1000

  delay(200);

  Serial.print(volt);

  Serial.println(" V");

  // Serial.print(ntu);

  // Serial.println(" NTU");

  Serial.print(ntu\_standardized);

  Serial.println(" NTU ");

  ec\_Val = (rawEc / T\_Cof) \* ecCal;// temperature and calibration compensation

  tds\_value = (133.42 \* pow(ec\_Val, 3) - 255.86 \* ec\_Val \* ec\_Val + 857.39 \* ec\_Val) \* 0.5;

  tds\_value = map(tds\_value, 0, 50000, 0, 1000);

  tds\_value = constrain(tds\_value, 0, 1000); // Đảm bảo giá trị nằm trong khoảng từ 0 tới 1000

  double Far= (((wTemp \*9)/5) + 32); // Temp in Far\*

  Serial.print("TDS:"); Serial.println(tds\_value);

  Serial.print("EC:"); Serial.println(ec\_Val, 2);

  Firebase.RTDB.setInt(&fbdo, "water/water\_temp(C)", temperatureC);

  Firebase.RTDB.setInt(&fbdo, "water/turbidity(NTU)", ntu\_standardized);

  Firebase.RTDB.setInt(&fbdo, "water/TDS", tds\_value);

  Firebase.RTDB.setInt(&fbdo, "water/EC", ec\_Val);

  display.clearDisplay();

  display.setTextSize(1); // Normal 1:1 pixel scale

  display.setTextColor(SH110X\_WHITE); // Draw white text

  display.setCursor(0, 0); // Start at top-left corner

  display.print("water\_temp:");

  display.print(temperatureC);

  display.println(" C");

  display.println("");

  display.print("turbidity:");

  display.print(ntu\_standardized);

  display.println(" NTU");

  display.println("");

  display.print("TDS: ");

  display.print(tds\_value);

  display.println(" ppm");

  display.println("");

  display.print("EC:");

  display.println(ec\_Val, 2);

  // display.println(" %");

  display.display();

  delay(3000);

}

float round\_to\_dp(float in\_value, int decimal\_place)

{

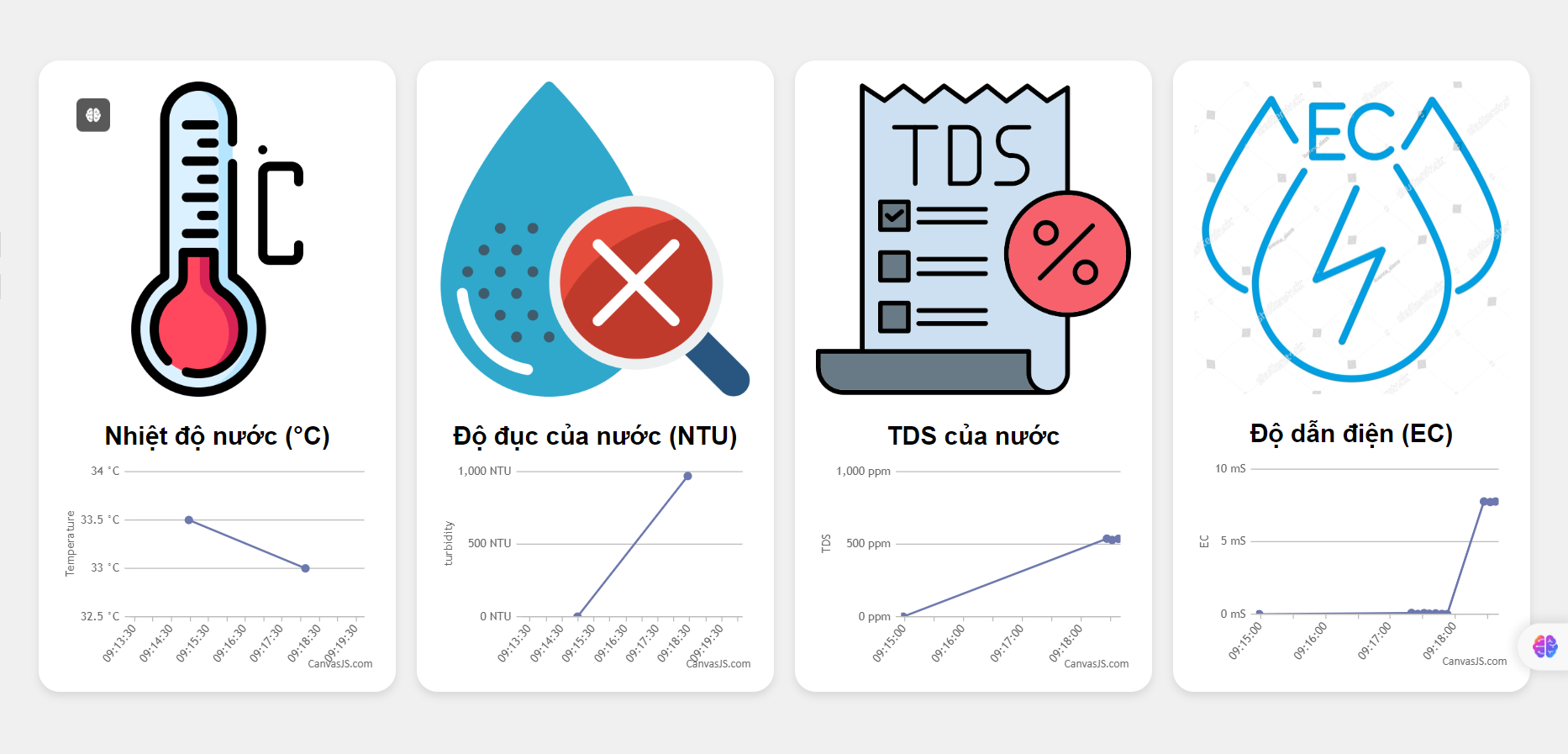
  float multiplier = powf(10.0f, decimal\_place);

  in\_value = roundf(in\_value \* multiplier) / multiplier;

  return in\_value;

}

## 4.4/ Web hiển thị



Hình 12 Hiển thị trên web

Khi các giá trị cảm biến thay đổi, ESP32 sẽ cập nhật lên firebase và từ firebase sẽ cập nhật lên web



Hình 13: Hiển thị trên Firebase

## 4.5/ Code web

### 4.5.1/HTML

<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Water Monitoring</title>

  <link rel="stylesheet" href="styles.css">

  <link href="https://fonts.googleapis.com/icon?family=Material+Icons+Outlined" rel="stylesheet">

  <link rel="shortcut icon" href="#">

  <script src="https://www.gstatic.com/firebasejs/8.1.1/firebase-app.js"></script>

  <script src="https://www.gstatic.com/firebasejs/3.1.0/firebase.js"></script>

  <script src="https://code.jquery.com/jquery-3.2.1.js"></script>

  <link href="https://cdn.jsdelivr.net/npm/round-slider@1.6.1/dist/roundslider.min.css" rel="stylesheet" />

  <script src="https://cdn.jsdelivr.net/npm/round-slider@1.6.1/dist/roundslider.min.js"></script>

  <script src="https://canvasjs.com/assets/script/canvasjs.min.js"></script>

</head>

<body>

  <div class="container">

    <div class="card" id="TemperatureC">

      <img id="d01\_img" src="./img/temperature.png" alt="img" style="width:100%">

      <h2>Nhiệt độ nước (°C)</h2></h2>

      <!-- <p id="tempC-value">Đang tải...</p> -->

      <div id="chartContainer" style="height: 200px; width: 100%;"></div>

    </div>

    <div class="card" id="TemperatureF">

      <img id="d01\_img" src="./img/water.png" alt="img" style="width:100%">

      <h2>Độ đục của nước (NTU)</h2>

      <!-- <p id="tempF-value">Đang tải...</p> -->

      <div id="chartContainer\_F" style="height: 200px; width: 100%;"></div>

    </div>

    <div class="card" id="TDS">

      <img id="d01\_img" src="./img/bill.png" alt="img" style="width:100%">

      <h2>TDS của nước</h2>

      <!-- <p id="tds-value">Đang tải...</p> -->

      <div id="chartContainer\_TDS" style="height: 200px; width: 100%;"></div>

    </div>

    <div class="card" id="EC">

      <img id="d01\_img" src="./img/ec.jpg" alt="img" style="width:100%">

      <h2>Độ dẫn điện (EC)</h2>

      <!-- <p id="ec-value">Đang tải...</p> -->

      <div id="chartContainer\_EC" style="height: 200px; width: 100%;"></div>

    </div>

  </div>

  <script src="app.js"></script>

  <script src="https://www.gstatic.com/firebasejs/8.2.9/firebase-app.js"></script>

  <script src="https://www.gstatic.com/firebasejs/8.2.9/firebase-database.js"></script>

</body>

</html>

### 4.5.2/CSS

body {

    font-family: Arial, sans-serif;

    display: flex;

    justify-content: center;

    align-items: center;

    height: 100vh;

    background-color: #f0f0f0;

    margin: 0;

  }

  .container {

    display: grid;

    grid-template-columns: repeat(4, 1fr);

    gap: 20px;

  }

  .card {

    background: #fff;

    border-radius: 20px;

    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);

    padding: 20px;

    text-align: center;

    width: 300px;

  }

  .card h2 {

    margin-bottom: 10px;

  }

  .card p {

    font-size: 24px;

    color: #333;

  }

### 4.5.3/JavaScript

// Cấu hình Firebase

const firebaseConfig = {

  apiKey: "AIzaSyBRcKbTuTnJ0ljvkK4OQQFpiI\_jXY8XOZI",

  authDomain: "iot2-37175.firebaseapp.com",

  databaseURL: "https://iot2-37175-default-rtdb.firebaseio.com",

  projectId: "iot2-37175",

  storageBucket: "iot2-37175.appspot.com",

  messagingSenderId: "748064719360",

  appId: "1:748064719360:web:e56159539854e15dc0cee5",

};

// Khởi tạo Firebase

firebase.initializeApp(firebaseConfig);

const database = firebase.database();

// Hàm khởi tạo biểu đồ chung

function createChart(containerId, title, suffix) {

  return new CanvasJS.Chart(containerId, {

    animationEnabled: true,

    theme: "light2",

    axisX: {

      valueFormatString: "HH:mm:ss",

      labelAngle: -50

    },

    axisY: {

      title: title,

      suffix: suffix

    },

    data: [{

      type: "line",

      xValueType: "dateTime",

      dataPoints: []

    }]

  });

}

// Khởi tạo các biểu đồ

const chart\_C = createChart("chartContainer", "Temperature ", " °C");

const chart\_F = createChart("chartContainer\_F", "turbidity ", " NTU");

const chart\_TDS = createChart("chartContainer\_TDS", "TDS", " ppm");

const chart\_EC = createChart("chartContainer\_EC", "EC", " mS");

// Hàm cập nhật biểu đồ

function updateChart(chart, dataPoints, newValue) {

  const currentTime = new Date();

  dataPoints.push({ x: currentTime, y: newValue });

  if (dataPoints.length > 100) {

    dataPoints.shift(); // Giới hạn số lượng điểm dữ liệu hiển thị

  }

  chart.options.data[0].dataPoints = dataPoints;

  chart.render();

}

// Dữ liệu cho các biểu đồ

let dataPoints\_C = [];

let dataPoints\_F = [];

let dataPoints\_TDS = [];

let dataPoints\_EC = [];

// Lắng nghe dữ liệu từ Firebase và cập nhật biểu đồ

firebase.database().ref("/water/water\_temp(C)").on("value", function(snapshot) {

  if (snapshot.exists()) {

    const tempC = snapshot.val();

    updateChart(chart\_C, dataPoints\_C, tempC);

    console.log(tempC);

  } else {

    console.log("No data for temperature(C)");

  }

});

firebase.database().ref("/water/turbidity(NTU)").on("value", function(snapshot) {

  if (snapshot.exists()) {

    const tempF = snapshot.val();

    updateChart(chart\_F, dataPoints\_F, tempF);

    console.log(tempF);

  } else {

    console.log("No data for turbidity(NTU)");

  }

});

firebase.database().ref("/water/TDS").on("value", function(snapshot) {

  if (snapshot.exists()) {

    const tempTDS = snapshot.val();

    updateChart(chart\_TDS, dataPoints\_TDS, tempTDS);

    console.log(tempTDS);

  } else {

    console.log("No data for TDS");

  }

});

firebase.database().ref("/water/EC").on("value", function(snapshot) {

  if (snapshot.exists()) {

    const tempEC = snapshot.val();

    updateChart(chart\_EC, dataPoints\_EC, tempEC);

    console.log(tempEC);

  } else {

    console.log("No data for EC");

  }

});

Tài liệu tham khảo

<https://mlab.vn/2458624-cam-bien-tds-do-tong-chat-ran-hoa-tan.html>

<https://www.geyser.com.vn/tin-tuc/chi-so-tds-la-gi/>

<https://maydochuyendung.com/tin-tuc/chi-tiet/may-do-do-duc-cua-nuoc>

<https://how2electronics.com/aquarium-water-quality-monitor-with-tds-sensor-esp32/>

<https://nshopvn.com/product/cam-bien-nhiet-do-ds18b20-day/>

<https://www.geyser.com.vn/tin-tuc/chi-so-tds-la-gi/>