

BÁO CÁO NGHIÊN CỨU KHOA HỌC

TÊN ĐỀ TÀI: QUAN TRẮC LƯỢNG MƯA, LƯU LƯỢNG NƯỚC VÀ MỨC NƯỚC HỖ TRỢ GIÁM SÁT HỖ TRỢ HỒ THỦY ĐIỆN

Người thực hiện: Trần Đức Minh

ĐẶT VẤN ĐỀ

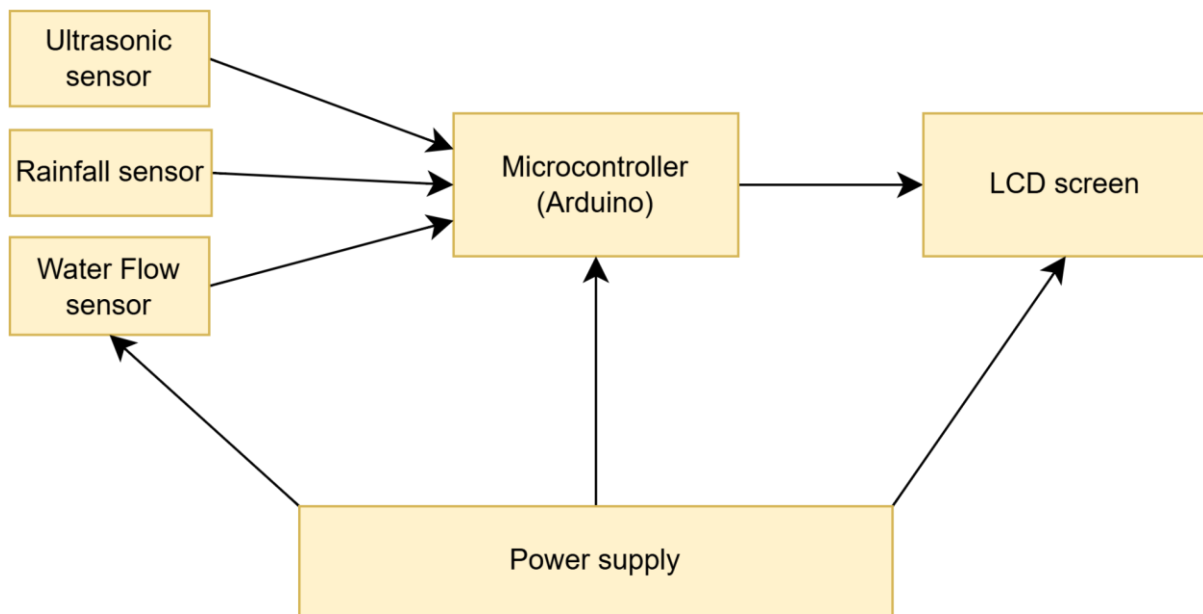
Nước là một trong những nhân tố quan trọng nhất cấu thành nên sự sống trên Trái Đất. Con người phụ thuộc vào nước cho việc ăn uống, giao thông, nông nghiệp, sản xuất điện và phát triển một số ngành công nghiệp khác. Ngoài ra, mực nước là một chỉ số quan trọng sử dụng trong dự báo lũ, và nguy cơ ảnh hưởng tới sự vận hành an toàn của các đập thủy điện. Khi nắm bắt được xu hướng tăng giảm của mực nước thì chúng ta có thể đưa ra những dự báo chính xác hơn về thời điểm tích nước và xả lũ để đảm bảo an toàn đập thủy điện, ổn định sản xuất và đời sống nhân dân. Vì vậy, việc giám sát, đo lường mực nước và lượng mưa hỗ trợ giám sát thủy điện là một trong những ưu tiên hàng đầu hiện nay của mọi tổ chức, chính phủ trên toàn cầu nói chung, và Việt Nam nói riêng. Hiện nay, có rất nhiều phương pháp đo mực nước như sử dụng cảm biến thủy tĩnh, cảm biến siêu âm, cảm biến điện dung, ... Tuy nhiên, các phương pháp này hiện vẫn chưa được sử dụng rộng rãi tại Việt Nam. Vì vậy mục tiêu của đề tài này, trước tiên, là đưa ra những thông số về lượng mưa, lưu lượng nước và mực nước để phục vụ cho quá trình giám sát an toàn hồ thủy điện. Trong tương lai, đề tài này sẽ hướng đến khả năng áp dụng trí tuệ nhân tạo AI để tăng năng lực phân tích và dự báo xu hướng về lượng mưa, lưu lượng dòng chảy và mực nước ở các hồ thủy điện của các đơn vị quan trắc tại Việt Nam.

CÁC NGHIÊN CỨU LIÊN QUAN

1. [\(PDF\) Nghiên cứu đánh giá sản phẩm mưa từ nhiệm vụ đo mưa toàn cầu \(GPM\) cho miền Bắc Việt Nam \(researchgate.net\)](#)
2. [18_nguyentuananh_1494.pdf \(tailieu.vn\)](#)
3. [NGHIÊN CỨU KHẢ NĂNG KẾT HỢP VÀ MỨC ĐỘ CHỐNG CHIÙ SÂU BỆNH \(vista.gov.vn\)](#)

GIẢI PHÁP THỰC HIỆN

Sơ đồ tổng quan hệ thống

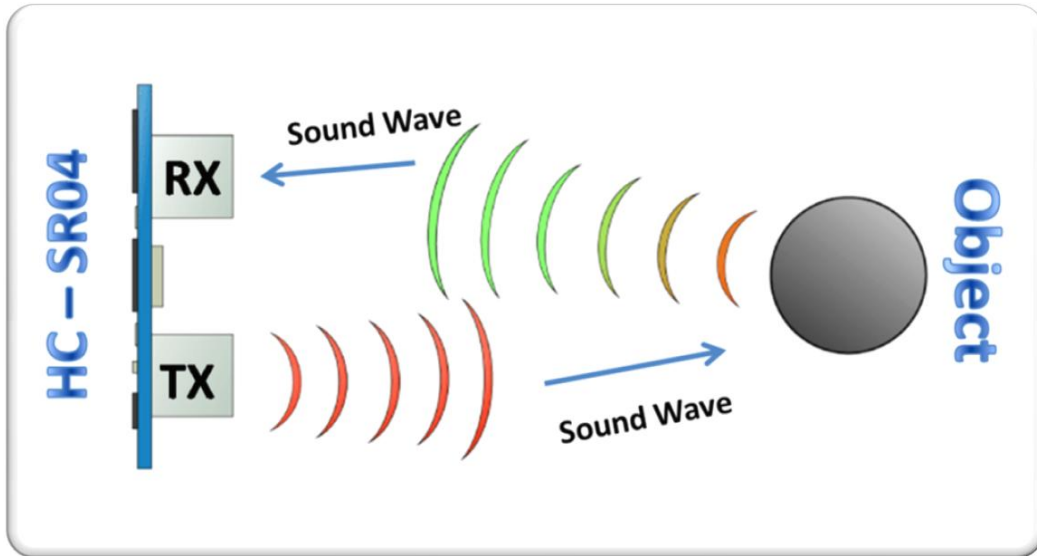


CƠ SỞ LÝ THUYẾT

Cảm biến siêu âm

Nguyên lý hoạt động của cảm biến đo khoảng cách bằng sóng siêu âm:

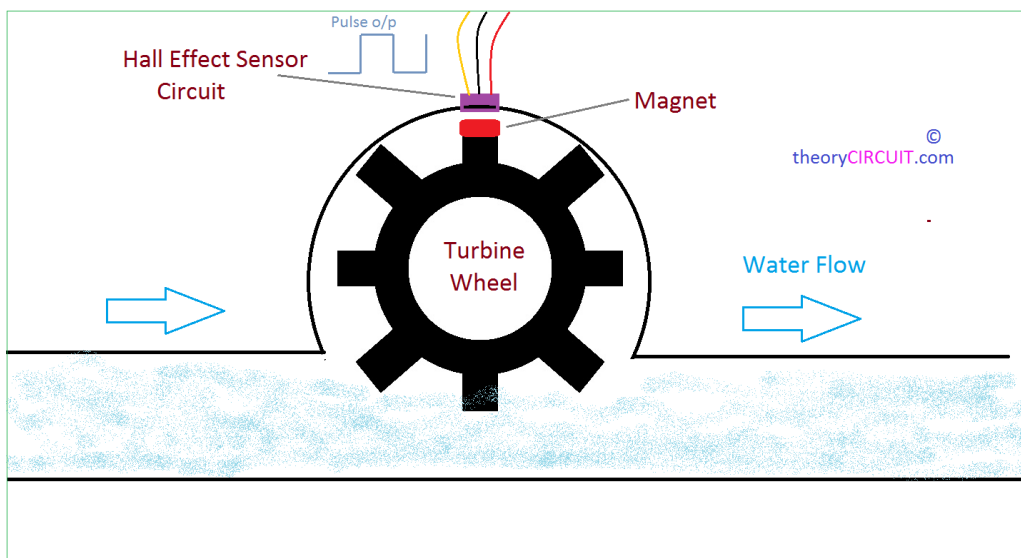
Để đo khoảng cách, ta sẽ phát 1 xung rất ngắn (5 microSeconds) từ chân Trig. Sau đó, cảm biến siêu âm sẽ tạo ra 1 xung HIGH ở chân Echo cho đến khi nhận lại được sóng phản xạ ở pin này. Chiều rộng của xung sẽ bằng với thời gian sóng siêu âm được phát từ cảm biến và quay trở lại.



Cảm biến lưu lượng dòng chảy

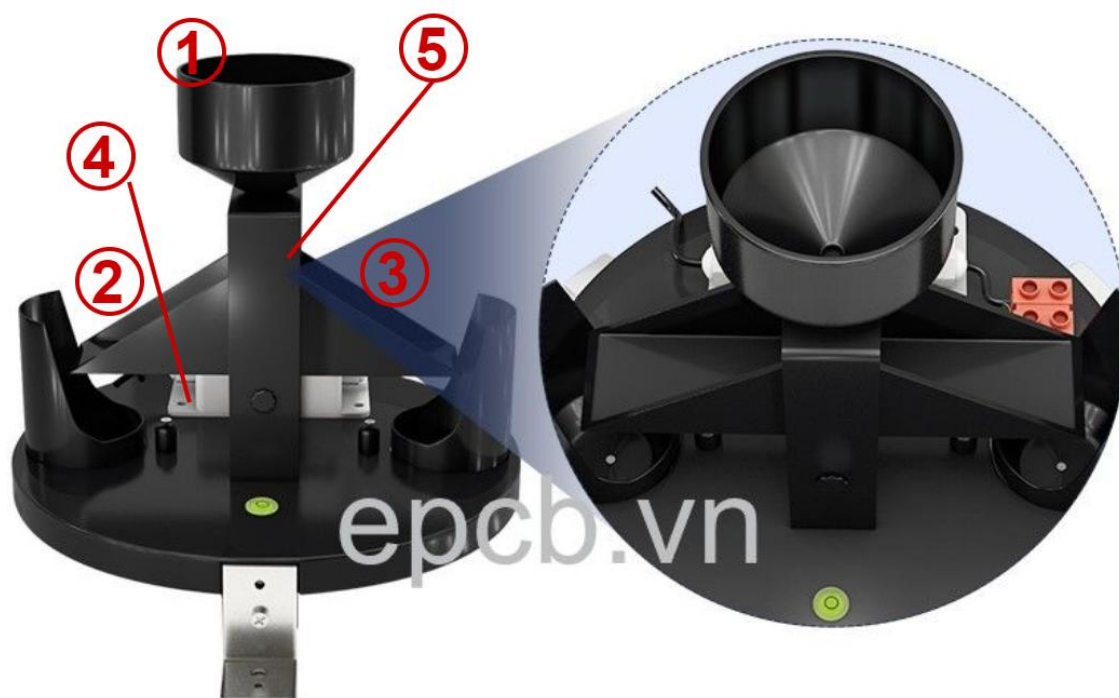
Nguyên lý hoạt động của cảm biến lưu lượng nước:

Cảm biến lưu lượng nước bao gồm một van nhựa mà nước có thể đi qua. Một roto nước cùng với một cảm biến hiệu ứng Hall chỉ định chiều hướng và đo lưu lượng nước. Khi chảy qua van, nước làm quay roto. Bằng cách này, chúng ta có thể quan sát sự thay đổi trong tốc độ của động cơ. Sự thay đổi này được tính là đầu ra dưới dạng tín hiệu xung bởi cảm biến hiệu ứng Hall, từ đó chúng ta có thể đo tốc độ dòng chảy.



Cảm biến lượng mưa

Để đo được lượng mưa ta sử dụng cảm biến lượng mưa Rainfall ES-RAIN-01. Cảm biến này cấu tạo chính gồm có 4 phần: 1) là gáo hứng nước mưa; 2) là máng chứa nước bên trái; 3) máng chứa nước bên phải; 4) mạch tính toán lượng mưa; 5) cảm biến từ trường.



Quá trình vận hành cơ học của thiết bị: khi lượng nước tích tụ đạt đến độ cao nhất định, gầu nghiêng sẽ mất sự cân bằng và đảo lộn nó. Lượng mưa liên tục sẽ làm cho xô nghiêng quay sang trái và phải.

Quá trình vận hành điện tử của thiết bị: khi lượng nước tích tụ đạt đến độ cao nhất định, gầu nghiêng sẽ mất sự cân bằng và đảo lộn nó. Lượng mưa liên tục sẽ làm cho xô nghiêng quay sang trái và phải và công tắc tiếp xúc sẽ biến thời gian quay của xô nghiêng thành tín hiệu điện, được gửi đến máy ghi. Trong quá trình này, số lần quay của các máng nước sẽ được cảm biến từ trường ghi lại và sau đó sẽ tính toán ra lượng mưa theo công thức quy đổi.

Cảm biến đo lượng mưa Rainfall ES-RAIN-01 sử dụng giao thức RS485 để giao tiếp với vi điều khiển.

Arduino Uno R3

Trong mạch, Arduino được sử dụng để tạo liên kết giữa thiết bị cảm biến và linh kiện điện tử khác nhau, đọc và xử lý dữ liệu analog từ các cảm biến, cung cấp nguồn và Ground cho các thiết bị hoạt động bình thường.

Dưới đây là một số thông tin về thông số kĩ thuật của Arduino:

Thông số	Arduino
Hình ảnh	
Vi điều khiển	ATmega328P
Điện áp hoạt động	5V
Điện áp vào khuyến dùng	7-12V
Điện áp vào giới hạn	6-20V
Digital I/O pin	14 (trong đó 6 pin có khả năng băm xung)
PWM Digital I/O Pins	6
Analog Input Pins	6
Cường độ dòng điện trên mỗi I/O pin	20 mA
Cường độ dòng điện trên mỗi 3.3V pin	50 mA

Flash Memory	32 KB (ATmega328P) 0.5 KB được sử dụng bởi bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Tốc độ	16 MHz
Chiều dài	68.6 mm
Chiều rộng	53.4 mm
Trọng lượng	25 g

Màn hình LCD

Trong mạch, LCD được sử dụng để thể hiện kết quả đo qua màn hình được dưới dạng chữ, số. Dưới đây là một số thông tin về thông số kỹ thuật của màn hình LCD:

Thông số	LCD 16x02
Hình ảnh	
Điện áp hoạt động	2.7 - 5.5 VDC
Dòng điện tiêu thụ (hoạt động)	350uA - 600uA
Điện áp ra mức cao	> 2.4 VDC

Điện áp ra mức thấp	< 0.4 VDC
Nhiệt độ hoạt động	-30 độ C - 75 độ C
Kích thước	80 x 36 x 12.5 mm

Chức năng của từng chân:

VSS (chân 1): Nối đất cho LCD.

VDD (chân 2): Cấp nguồn cho LCD (VCC=5V).

VE (chân 3): Điều chỉnh độ tương phản của LCD.

RS (chân 4): Chọn thanh ghi (ghi/đọc).

R/W (chân 5): Chế độ đọc/ghi.

E (chân 6): Cho phép (Enable).

D0 đến D7 (chân 7 đến 14): 8 đường bus dữ liệu.

A (chân 15): Nguồn dương cho đèn nền.

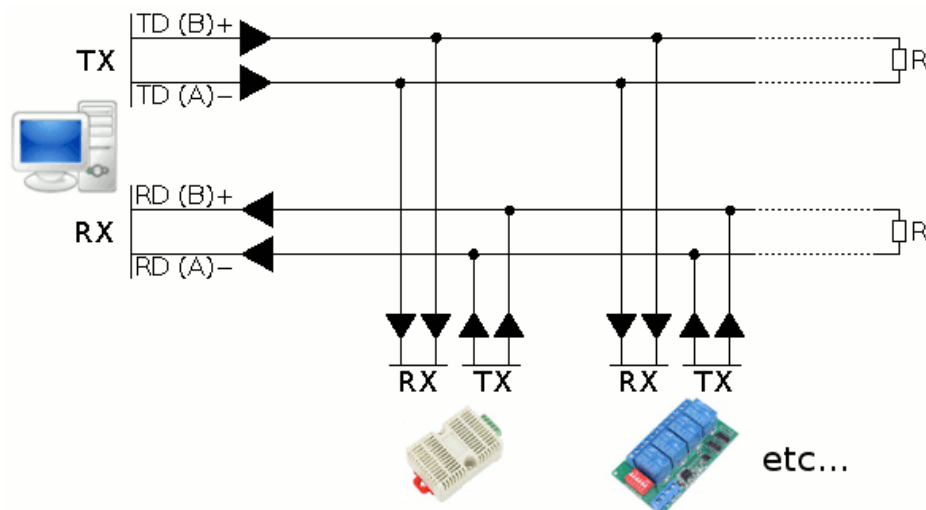
K (chân 16): Nguồn âm cho đèn nền 1.

Giao thức RS485

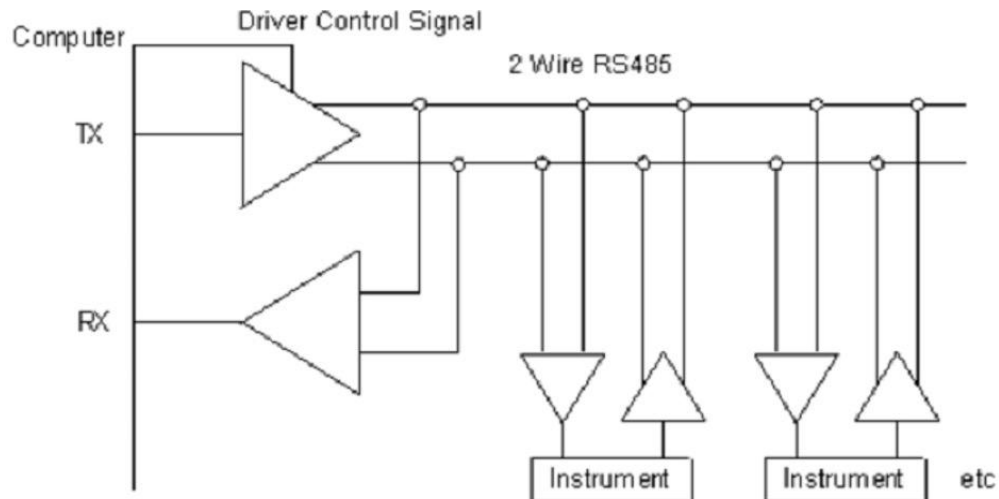
Cảm biến đo lượng mưa Rainfall ES-RAIN-01 sử dụng giao thức RS485 để giao tiếp với vi điều khiển.

Nguyên lý hoạt động của RS485 khá đơn giản, dữ liệu sẽ được truyền qua 2 dây khi xoắn lại với nhau, dây này được gọi là cáp xoắn. Khi dây được xoắn lại sẽ tạo cho RS485 khả năng chống nhiễu cao và khả năng truyền tín hiệu đường dài tốt hơn.

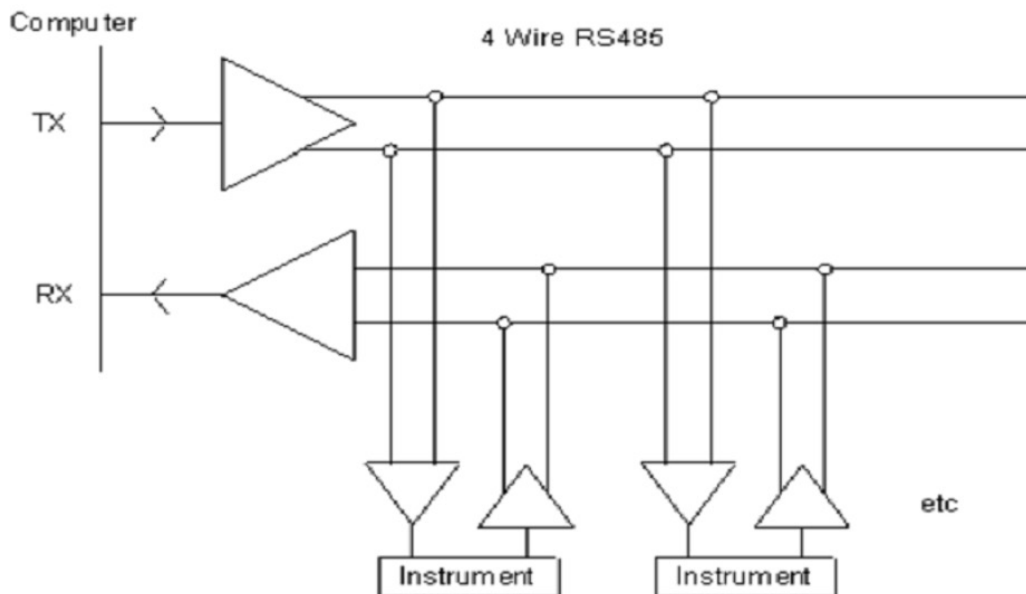
RS485 được chia làm 2 loại cấu hình, hiện đang được sử dụng nhiều nhất hiện nay là cấu hình 2 dây và cấu hình 4 dây.



Đối với cấu hình 2 dây, dữ liệu sẽ được truyền đi theo một hướng tại một thời điểm nhất định. Với kiểu thiết lập này, tín hiệu TX và RX sẽ cùng nhau dùng chung một cặp dây duy nhất giúp người sử dụng tiết kiệm được chi phí cài đặt.



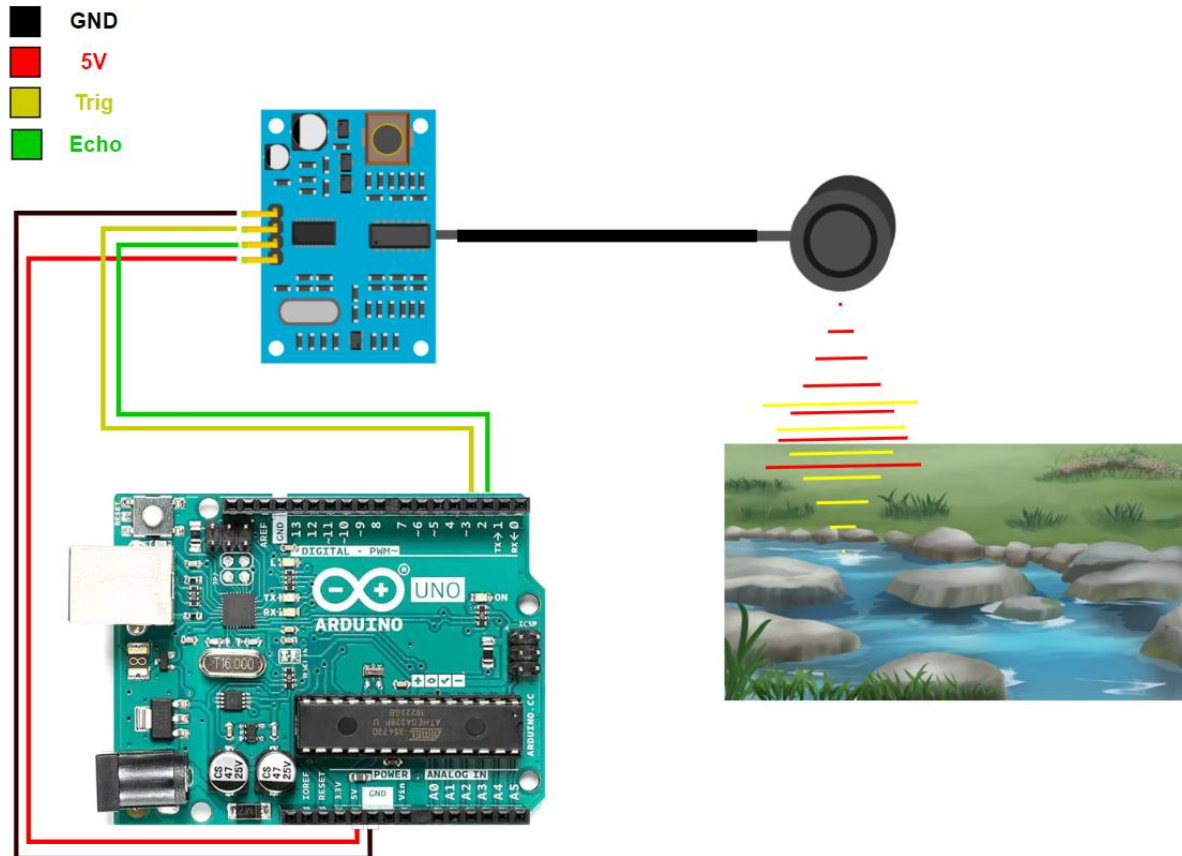
Nguyên lý hoạt động của cấu hình 4 dây tương đối khác, tại đây dữ liệu sẽ được truyền đi và đến đồng thời từ các nút. Tại đây, 2 dây sẽ đảm nhận nhiệm vụ truyền và 2 dây còn lại sẽ đảm nhận nhiệm vụ nhận.



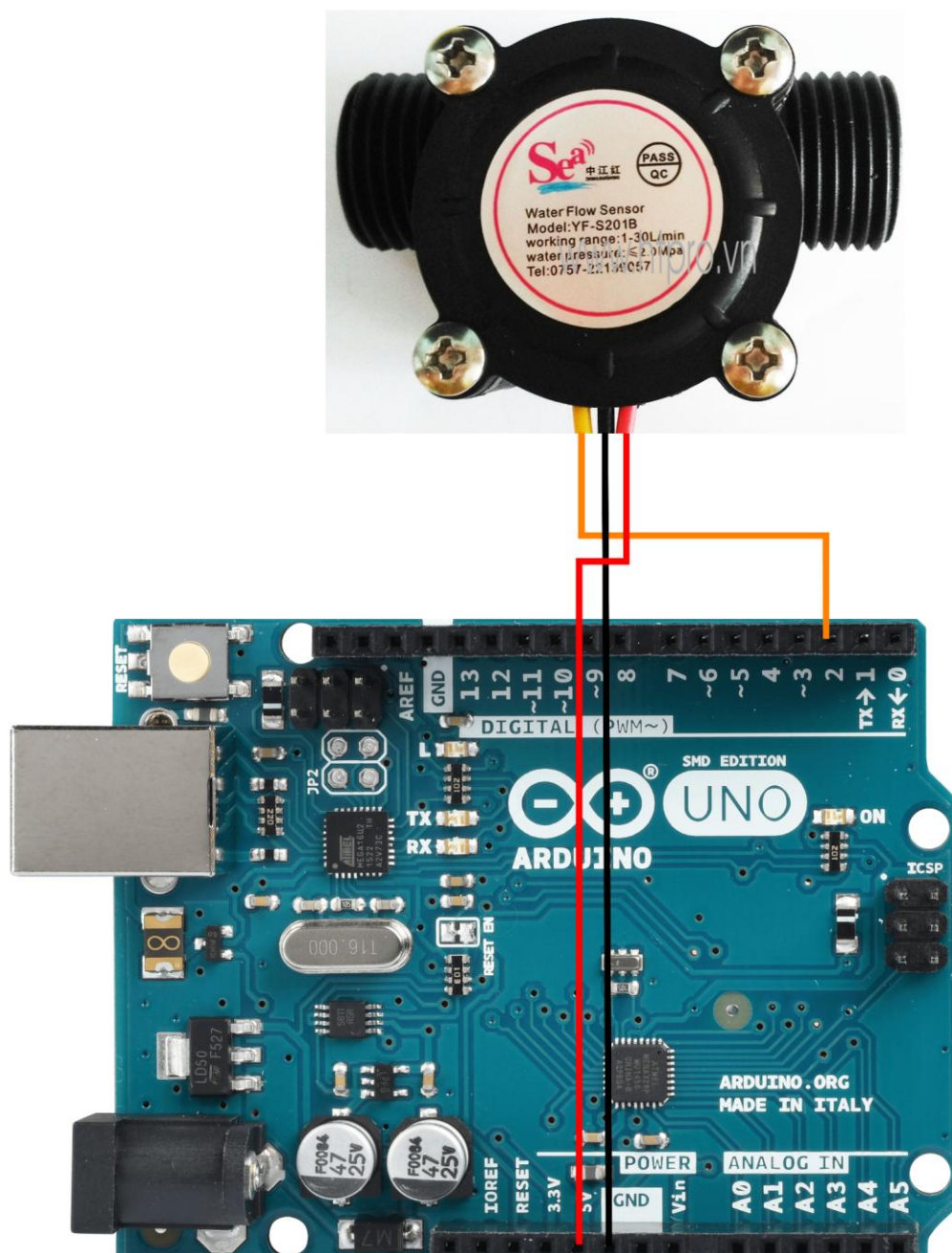
THIẾT KẾ MẠCH

Thiết kế sơ đồ mạch

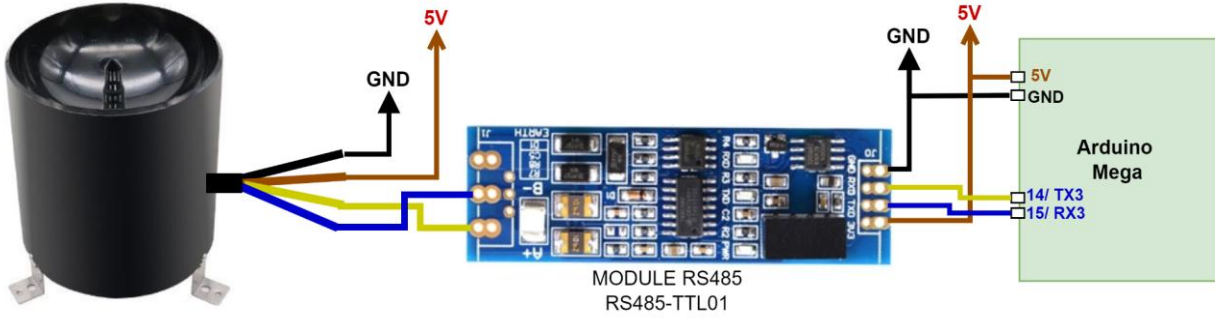
Mạch đo mực nước



Mạch đo lưu lượng nước

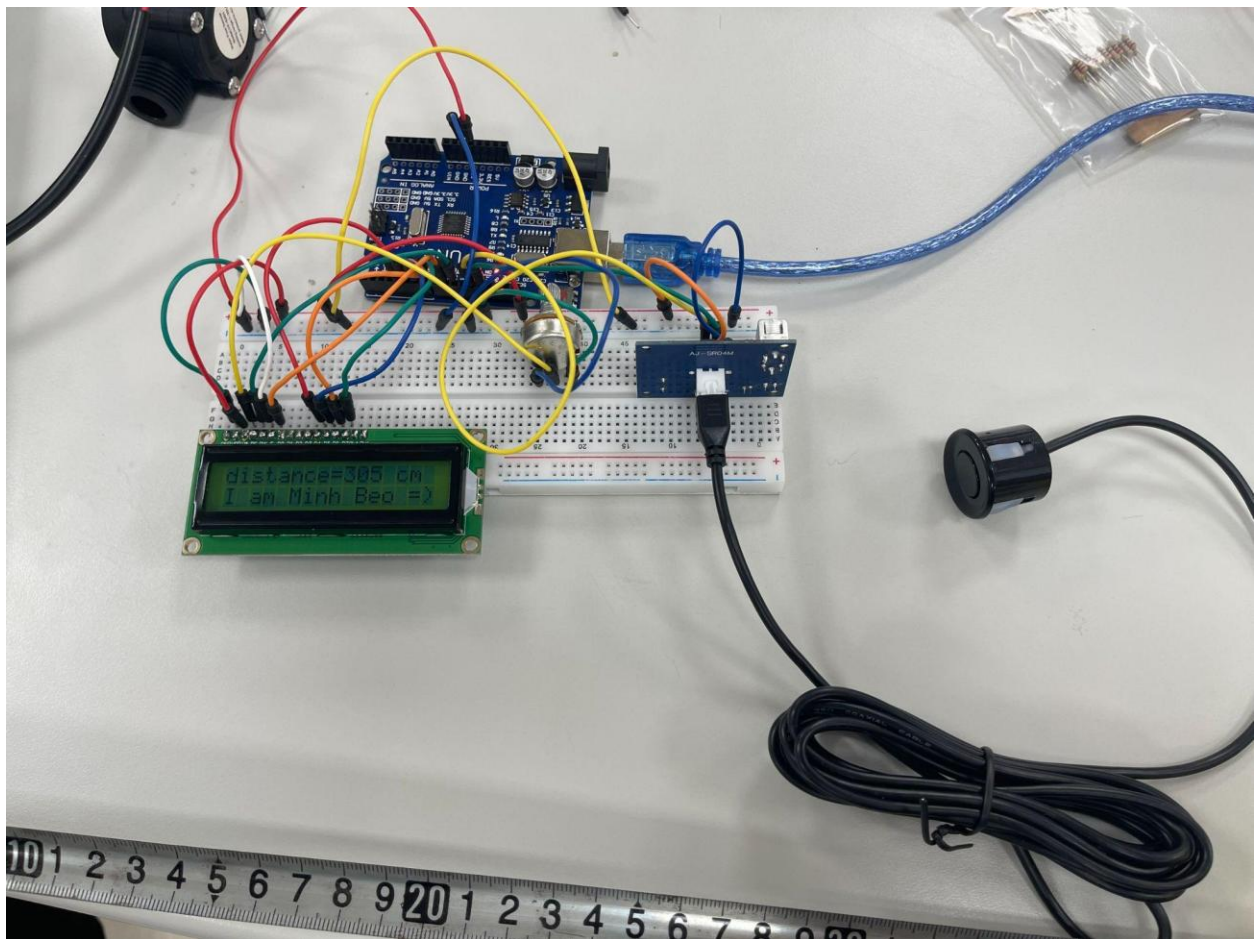


Mạch đo lượng mưa

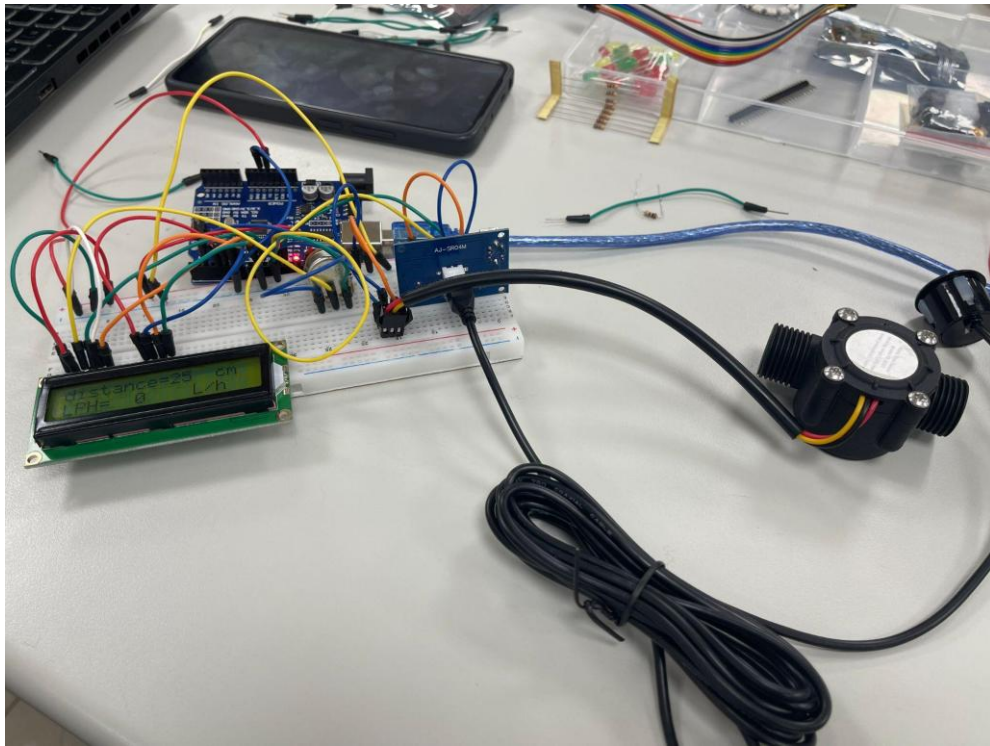


Lắp mạch

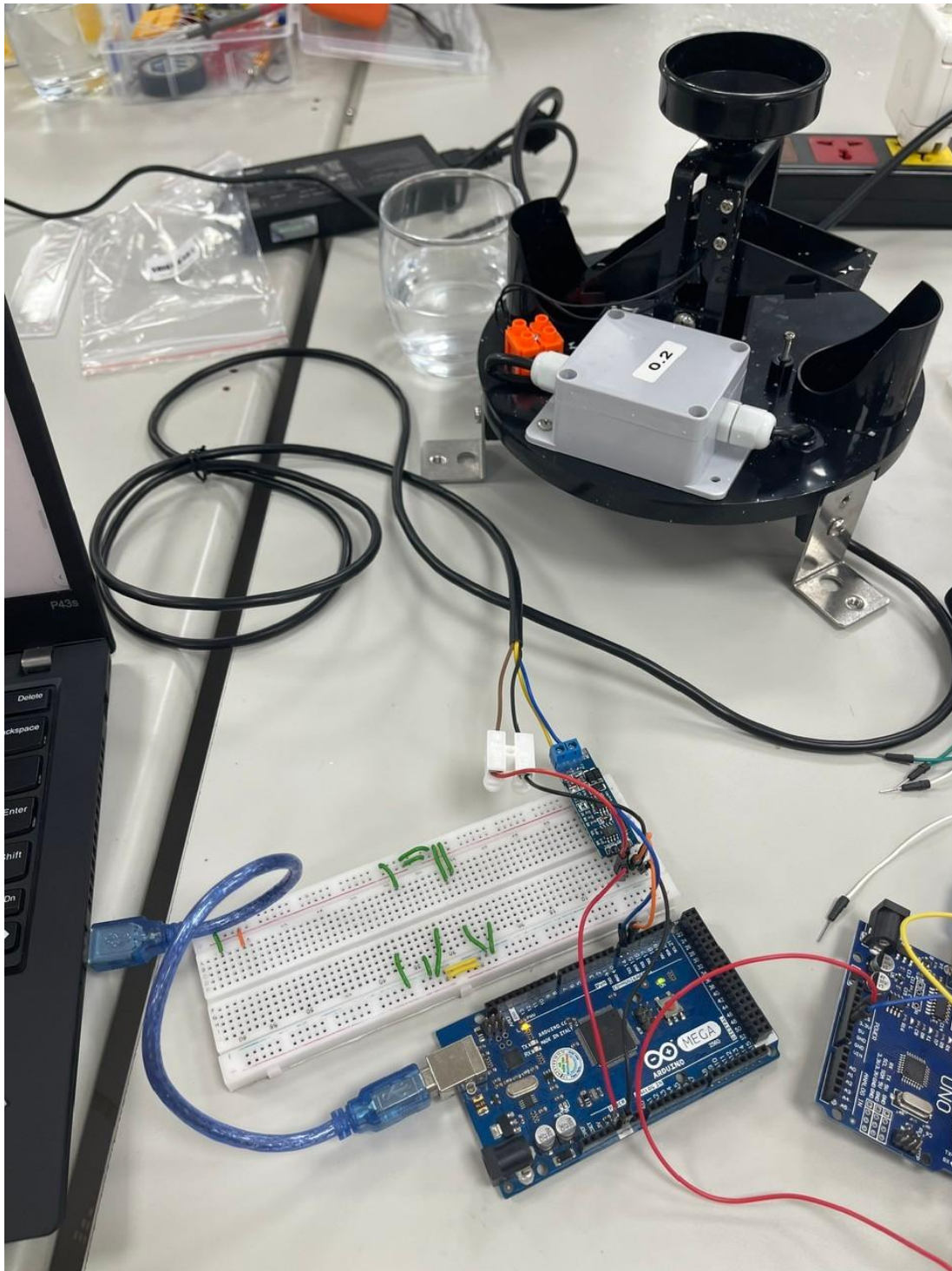
Mạch đo mực nước



Mạch đo lưu lượng nước



Mạch đo lượng mưa



LẬP TRÌNH

Code mạch đo mực nước

```
const int trig = 8;      // chân trig của HC-SR04
const int echo = 7;      // chân echo của HC-SR04

void setup()
{
    Serial.begin(9600);    // giao tiếp Serial với baudrate 9600
    pinMode(trig,OUTPUT);  // chân trig sẽ phát tín hiệu
    pinMode(echo,INPUT);   // chân echo sẽ nhận tín hiệu
}

void loop()
{
    unsigned long duration; // biến đo thời gian
    int distance;           // biến lưu khoảng cách

    /* Phát xung từ chân trig */
    digitalWrite(trig,0);   // tắt chân trig
    delayMicroseconds(2);
    digitalWrite(trig,1);   // phát xung từ chân trig
    delayMicroseconds(5);   // xung có độ dài 5 microseconds
    digitalWrite(trig,0);   // tắt chân trig

    /* Tính toán thời gian */
    // Đo độ rộng xung HIGH ở chân echo.
    duration = pulseIn(echo,HIGH);
    // Tính khoảng cách đến vật.
    distance = int(duration/2/29.412);

    /* In kết quả ra Serial Monitor */
    Serial.print(distance);
    Serial.println("cm");
    delay(200);
}
```

Code mạch đo lưu lượng nước

```
#define CONSTANT_FACTOR 7.5 //covert hall sensor signal to lit / minute

volatile int flowFrequency;
unsigned int lPerHour;
unsigned char flowSensorPin = 2;
unsigned long currentTime;
unsigned long cloopTime;
void flowCount() // Interrupt function
{
    flowFrequency++;
}
void setup()
{
    pinMode(flowSensorPin, INPUT_PULLUP);
    Serial.begin(9600);
    attachInterrupt(0, flowCount, RISING);
    sei(); // Enable interrupts
    currentTime = millis();
    cloopTime = currentTime;
}
int Waterflow()
{
    currentTime = millis();
    //after 1s read the sensor signal count and reset it
    if (currentTime >= (cloopTime + 1000))
    {
        cloopTime = currentTime; // update referent time
        lPerHour = (flowFrequency * 60 / CONSTANT_FACTOR);
        flowFrequency = 0;
        return lPerHour;
    }
    else
        return 0;
}

void loop()
```

```

{

    lcd.setCursor(0,1);
    lcd.print("LPH=");
    lcd.setCursor(11,1);
    lcd.print("L/h");
    lcd.setCursor(6,1);
    int flow = Waterflow();
    lcd.setCursor(6,1);
    lcd.print("      ");
    lcd.setCursor(6,1);
    lcd.print(flow);
    delay(1000);

}

```

Code mạch đo lượng mưa

```

#include <SoftwareSerial.h>

#define ADDRESS 0x01
#define FUNCTION_CODE 0x03
#define INITIAL_ADDRESS_HI 0x00
#define INITIAL_ADDRESS_LO 0x00
#define DATA_LENGTH_HI 0x00
#define DATA_LENGTH_LO 0x01
#define CHECK_CODE_LO 0x84
#define CHECK_CODE_HI 0x0A
#define NUMBER_BYTES_RESPONES 7
// Request frame for the soil sensor
const byte request[] = {ADDRESS, FUNCTION_CODE, INITIAL_ADDRESS_HI,
INITIAL_ADDRESS_LO, DATA_LENGTH_HI, DATA_LENGTH_LO, CHECK_CODE_LO,
CHECK_CODE_HI};
byte sensorResponse[NUMBER_BYTES_RESPONES];

//SoftwareSerial mod(2, 3); // Software serial for RS485 communication

```



```

unsigned long startTime;
void setup()
{
    Serial.begin(9600); // Initialize serial communication for debugging
    //mod.begin(4800);    // Initialize software serial communication at
4800 baud rate
    Serial3.begin(4800);
    delay(500);
    startTime = millis();
}

void loop()
{
    // Wait for the response from the sensor or timeout after 1 second
    if((millis() > startTime+2000)){
        startTime = millis();
        // Send the request frame to the soil sensor
        Serial3.write(request, sizeof(request));
        Serial.println("Sent request data");
    }
    if (Serial3.available() >= NUMBER_BYTES_RESPONES) // If valid response
received
    {
        // Read the response from the sensor
        byte index = 0;
        while (Serial3.available() && index < NUMBER_BYTES_RESPONES)
        {
            sensorResponse[index] = Serial3.read();
            Serial.print(sensorResponse[index], HEX); // Print the received byte
in HEX format
            Serial.print(" ");
            index++;
        }
        Serial.println(" $End of Rx data");

        // Parse and calculate the Rainfall value
        int Sensor_Int = int(sensorResponse[3] << 8 | sensorResponse[4]);
        float Senor_Float = Sensor_Int / 10.0;
    }
}

```

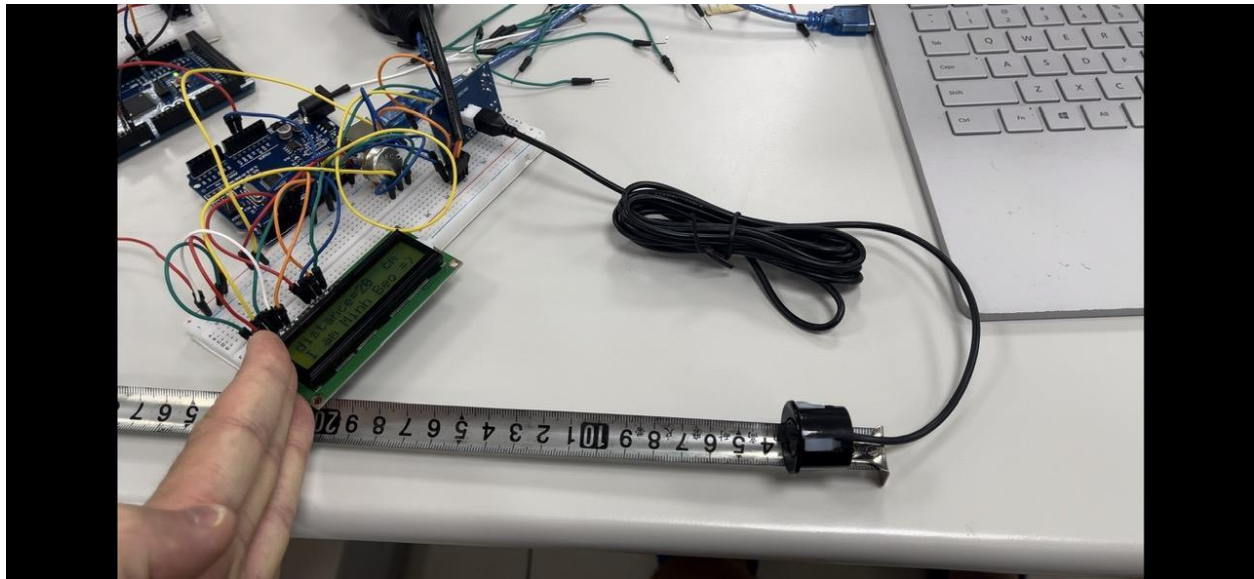
```
Serial.print("Float result: ");  
Serial.println(Senor_Float);  
}  
}
```

KẾT QUẢ, ĐÁNH GIÁ

Cảm biến siêu âm

Tiến hành đo ở khoảng cách 20 cm

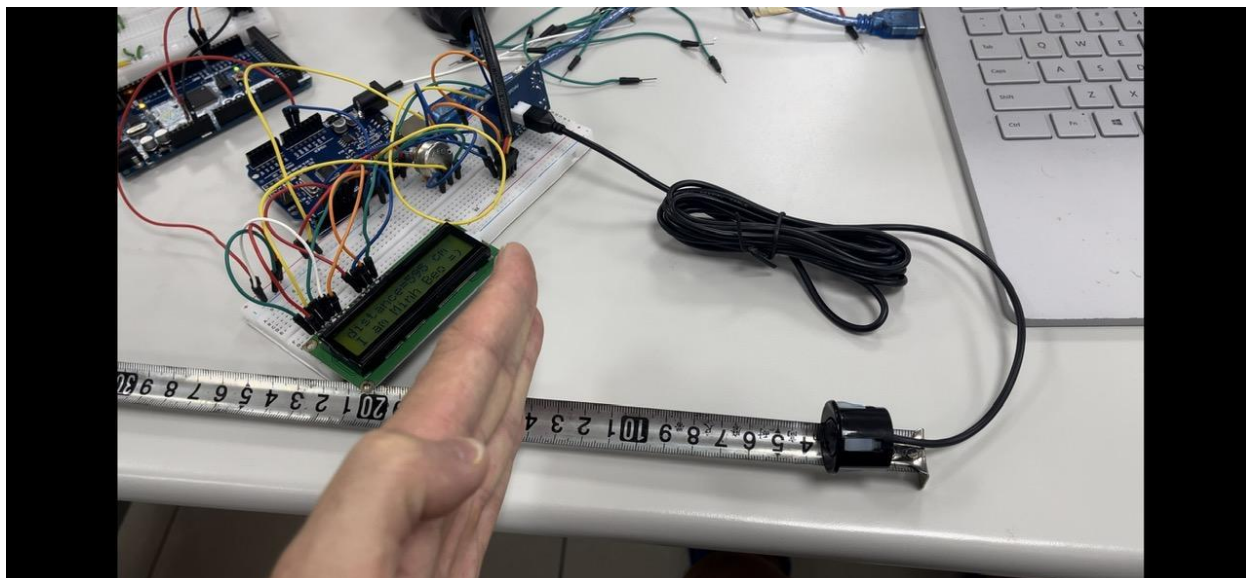
Kết quả: LCD hiển thị khoảng cách đo được là 20 cm



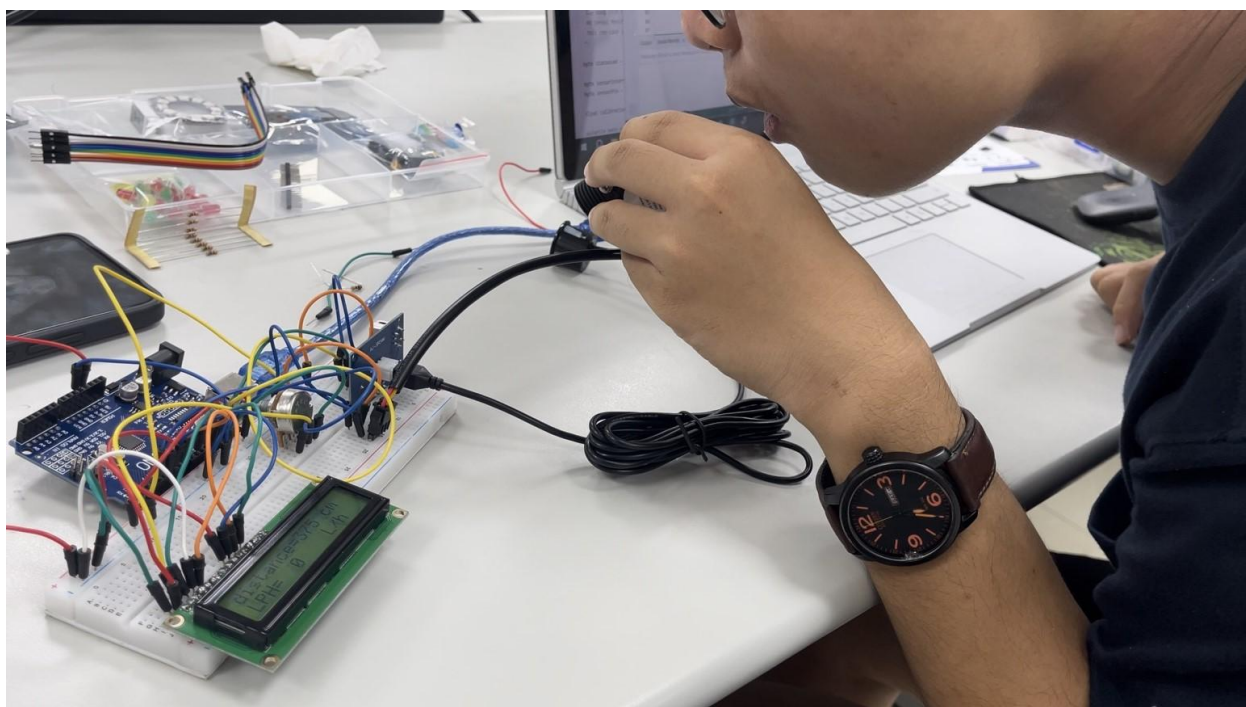
Tiến hành đo ở khoảng cách 14 cm (nhỏ hơn 20 cm)

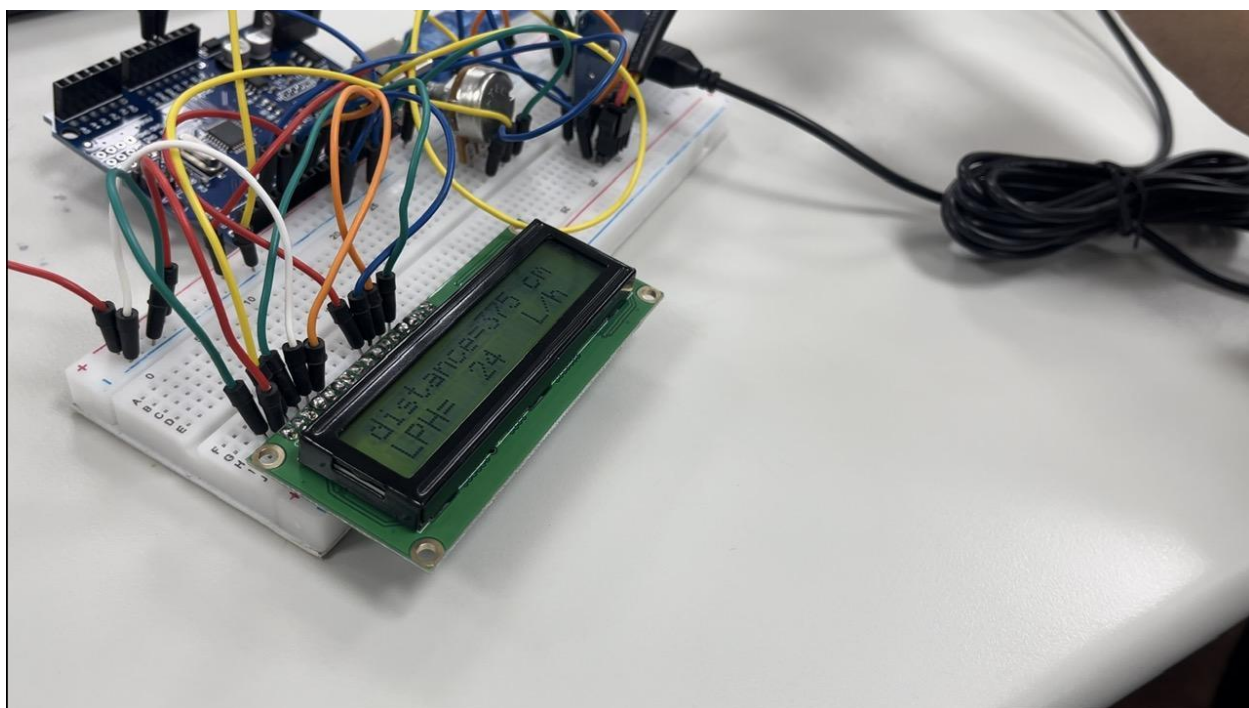
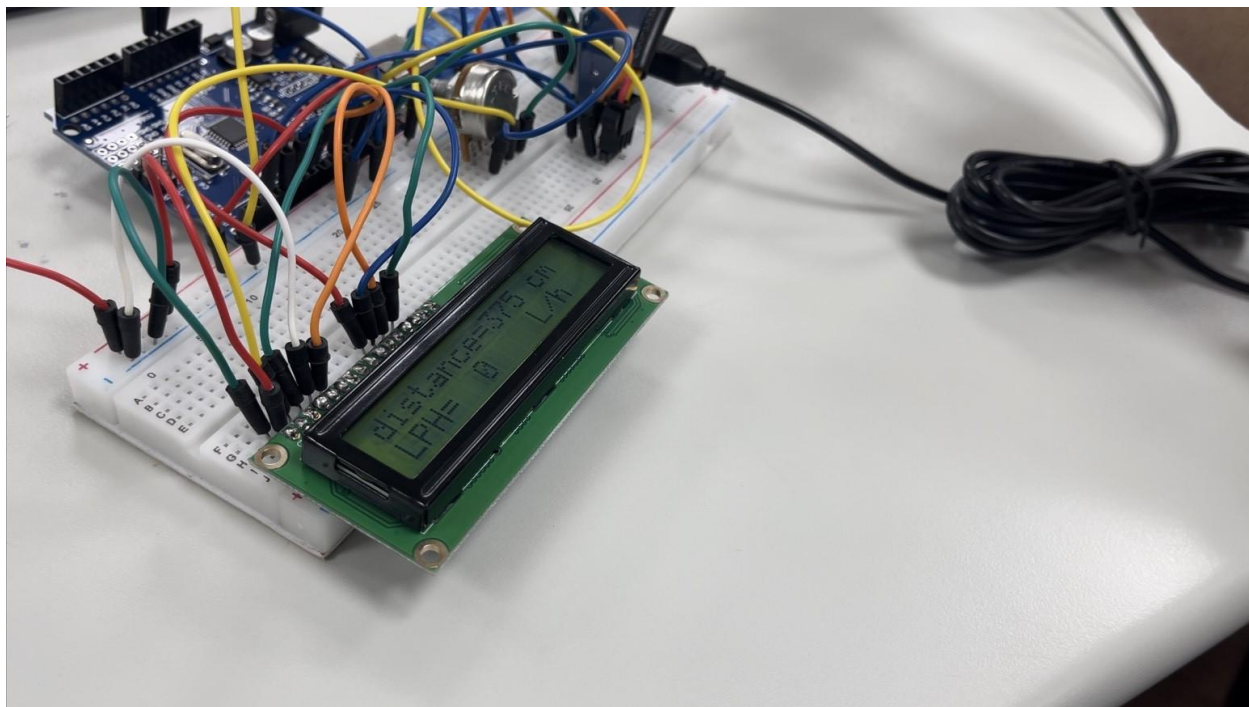
Kết quả: LCD không hiển thị khoảng cách đo được là 14 cm

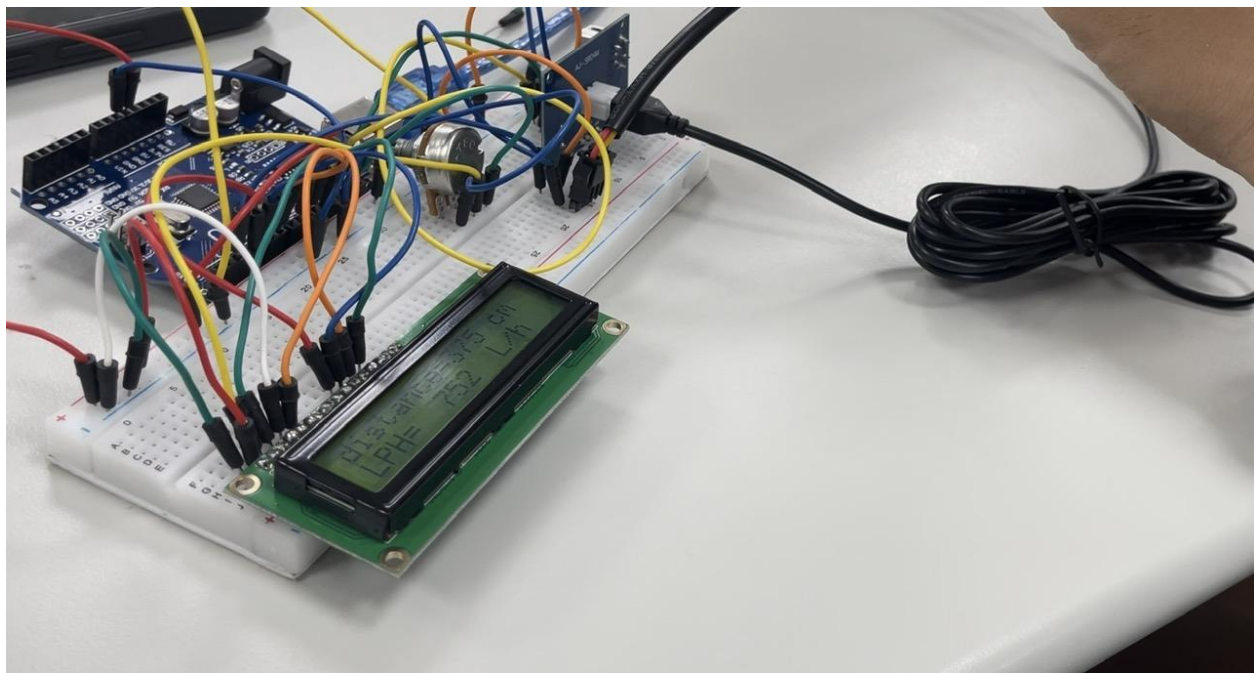
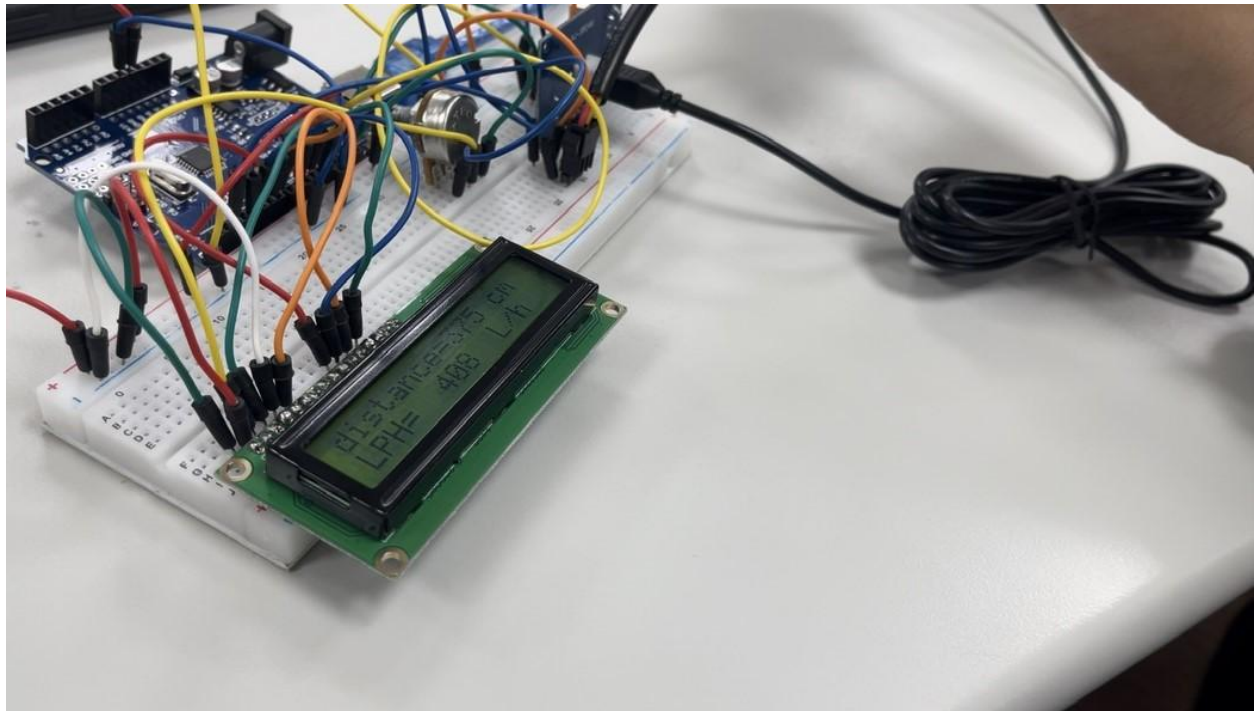
Nguyên nhân: khoảng cách này đã vượt qua giới hạn đo của cảm biến siêu âm nên trả ra kết quả sai.



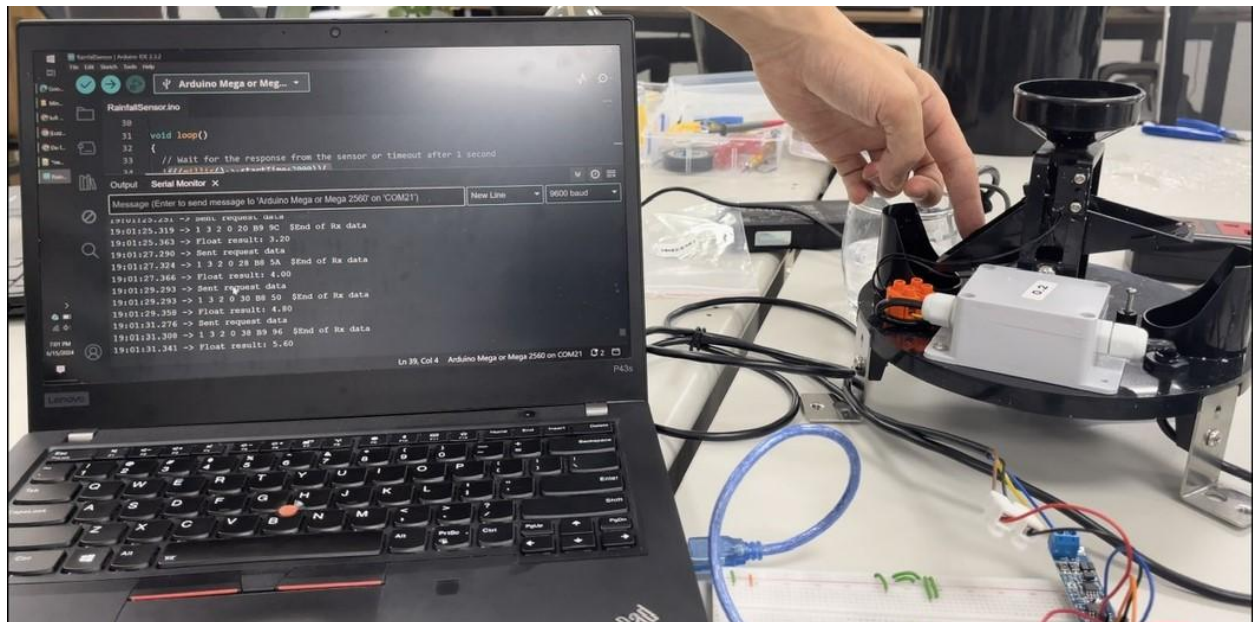
Cảm biến lưu lượng dòng chảy







Cảm biến lượng mưa



Thời gian lấy mẫu	VRAIN	Our system
20:19 12/08/2024	0.0	0.0
17:09 24/08/2024	28.2	27.0
17:10 24/08/2024	28.2	27.6

PHỤ LỤC

Cảm biến siêu âm

Link tinkercad của tác giả: [Circuit design Ultrasonic sensor - LCD session - Tinkercad](#)

TÀI LIỆU THAM KHẢO

Cảm biến siêu âm

1. https://blog.aem.eco/fundamentals_of_water_level_measurement
2. <https://hshop.vn/products/cam-bien-sieu-am-chong-nuoc-jsn-sr04t-v3-0-waterproof-ultrasonic-sensor>
3. <https://www.arduino.cc/reference/en/language/functions/math/map/>
4. <https://circuitdiagrams.in/water-tank-level-controller-using-arduino/>
5. <https://circuitdigest.com/microcontroller-projects/interfacing-ultrasonic-sensor-module-with-arduino>
6. <https://www.makerguides.com/jsn-sr04t-arduino-tutorial/>
7. <https://www.makerguides.com/wp-content/uploads/2019/02/JSN-SR04T-Datasheet.pdf>
8. <https://projecthub.arduino.cc/SurtrTech/measure-distance-using-jsn-sr-04t-lcd-b936b8>
9. [Ultrasonic sensor with Arduino » PIJA Education](#)

Cảm biến lưu lượng dòng chảy

1. https://blog.aem.eco/fundamentals_of_water_level_measurement
2. <https://nshopvn.com/product/cam-bien-luu-luong-nuoc/>
3. <https://www.electroschematics.com/working-with-water-flow-sensors-arduino/>
4. [Water Flow Sensor YF-S201 Arduino Interface \(theorycircuit.com\)](#)

Cảm biến lượng mưa

1. <https://epcb.vn/products/cam-bien-do-luu-luong-mua-rainfall-es-rainf-01-rs485-modbus-rtu>
2. <https://drive.google.com/file/d/1aXrREB5fpxaq2GmCJEFaWXzMa-epJfJv/view>
3. <https://cafelinhkien.com/san-pham/module-rs485-rs485-ttl01/>
4. [How is rainfall measured? | What does 1 mm rainfall mean? \(geographyhost.com\)](#)
5. [RS485 Là Gì : Cấu Tạo - Nguyên Lý - Cách Sử Dụng Hiệu Quả \(thietbikythuat.com.vn\)](#)
6. [Module RS485 RS485-TTL01 \(banlinhkien.com\)](#)

