BỘ CÔNG THƯƠNG

**TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP. HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO ĐỀ TÀI IOT**

**HỆ THỐNG CẢM BIẾN MỰC NƯỚC**

**GIẢNG VIÊN HƯỚNG DẪN:**

**Võ Hoàng Hải**

**SINH VIÊN THỰC HIỆN:**

1. 2001224631 - Lê Mạnh Tường
2. 2001223550 – Trần Tấn Nhựt
3. 2001221139 – Lâm Tiết Hải
4. 2044221994 – Lê Nguyên Khang

TP. HỒ CHÍ MINH, THÁNG 10 NĂM 2025

BỘ CÔNG THƯƠNG

**TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP. HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO ĐỀ TÀI IOT**

**HỆ THỐNG CẢM BIẾN MỰC NƯỚC**

**GIẢNG VIÊN HƯỚNG DẪN:**

**Võ Hoàng Hải**

**SINH VIÊN THỰC HIỆN:**

1. 2001224631 - Lê Mạnh Tường
2. 2001223550 – Trần Tấn Nhựt
3. 2001221139 – Lâm Tiết Hải
4. 2044221994 – Lê Nguyên Khang

TP. HỒ CHÍ MINH, THÁNG 10 NĂM 2025

**LỜI CAM ĐOAN**

Tôi xin cam đoan đây là công trình nghiên cứu của nhóm chúng tôi. Các số liệu, kết quả trong tiểu luận là trung thực, được nhóm tự thực hiện và chưa từng được công bố trong bất kỳ công trình nào khác.

Mọi sự hỗ trợ cho việc thực hiện đề tài này đã được ghi nhận và các nguồn tài liệu tham khảo đều được trích dẫn rõ ràng.

**Sinh viên thực hiện tiểu luận**

*(Ký và ghi rõ họ tên)*

|  |  |  |
| --- | --- | --- |
| Thành viên | Vai trò | Công việc chính |
| 2001223550 – Trần Tấn Nhựt | Quản lý, tổng hợp, thuyết trình | Xác định bài toán, phân chia công việc. Phụ trách mua đồ. Chuẩn bị **Arduino Uno R3**, cảm biến, mạch nối dây, test tín hiệu, lắp ráp mạch. Chỉnh sửa lại word và canva. |
| 2001224631 - Lê Mạnh Tường | Kỹ thuật phần cứng, thuyết trình. | Vẽ Wokwi, lắp ráp mạch, Lập trình trên Arduino IDE. |
| 2001221139 – Lâm Tiết Hải | Dashboard & Báo cáo, thuyết trình | Làm canva báo cáo |
| 2044221994 – Lê Nguyên Khang | Dashboard & Báo cáo | Làm word báo cáo |

**MỤC LỤC**

[MỞ ĐẦU 1](#_Toc216485142)

[Lý do chọn đề tài 1](#_Toc216485143)

[Mục đích nghiên cứu 1](#_Toc216485144)

[Đối tượng và phạm vi nghiên cứu 1](#_Toc216485145)

[Ý nghĩa khoa học và thực tiễn 2](#_Toc216485146)

[CHƯƠNG 1: GIỚI THIỆU 3](#_Toc216485147)

[1.1. Giới thiệu đề tài 3](#_Toc216485148)

[1.2. Mục tiêu của hệ thống 3](#_Toc216485149)

[1.3. Lợi ích mang lại 3](#_Toc216485150)

[CHƯƠNG 2: CÔNG NGHỆ SỬ DỤNG 5](#_Toc216485151)

[2.1. IoT Device (Thiết bị phần cứng) 5](#_Toc216485152)

[2.1.1. Arduino Uno R3 5](#_Toc216485153)

[2.1.2. Cảm biến mực nước (Water Level Sensor) 6](#_Toc216485154)

[2.1.3. Buzzer (Còi báo) 7](#_Toc216485155)

[2.1.4. Jumper Wires (Dây nối) 8](#_Toc216485156)

[2.1.5. Bảng kết nối mạch (Breadboard) 9](#_Toc216485157)

[2.2. Development Tool (Công cụ lập trình) 10](#_Toc216485158)

[2.2.1. Arduino IDE 10](#_Toc216485159)

[CHƯƠNG 3: KIẾN TRÚC HỆ THỐNG 12](#_Toc216485160)

[3.1. Mô hình kiến trúc và tương tác hệ thống 12](#_Toc216485161)

[3.1.1. Sơ đồ kiến trúc tổng thể 12](#_Toc216485162)

[3.2. Quy trình hoạt động của hệ thống 14](#_Toc216485163)

[3.3. Code chương trình điều khiển Arduino 16](#_Toc216485164)

[CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 17](#_Toc216485165)

[4.1. Tóm tắt kết quả đạt được 17](#_Toc216485166)

[4.2. Kiến nghị và hướng phát triển tiếp theo 17](#_Toc216485167)

[TÀI LIỆU THAM KHẢO 19](#_Toc216485168)

**Danh mục hình ảnh**

[Hình 1. Arduino Uno R3 – Vi điều khiển trung tâm của hệ thống 5](#_Toc216485169)

[Hình 2. Cảm biến mực nước 6](#_Toc216485170)

[Hình 3. Buzzer (Còi báo) 7](#_Toc216485171)

[Hình 4. Jumper Wires (Dây nối) 8](#_Toc216485172)

[Hình 5. Breadboard – Bảng kết nối mạch 9](#_Toc216485173)

[Hình 6. Arduino IDE (Integrated Development Environment) 10](#_Toc216485174)

[Hình 7. Sơ đồ đấu nối hệ thống cảm biến mực nước 11](#_Toc216485175)

[Hình 8. Sơ đồ kiến trúc tổng thể hệ thống cảm biến mực nước 12](#_Toc216485176)

[Hình 9. Lưu đồ hoạt động của hệ thống cảm biến mực nước 14](#_Toc216485177)

[Hình 10. Code chương trình 16](#_Toc216485178)

# MỞ ĐẦU

## Lý do chọn đề tài

Trong đời sống hiện nay, việc quản lý và giám sát nguồn nước đóng vai trò rất quan trọng, đặc biệt trong các hộ gia đình, khu dân cư, nông nghiệp và công nghiệp. Tình trạng tràn nước trong bể chứa thường xảy ra quên tắt hoặc mở máy bơm kịp thời, gây lãng phí tài nguyên và có thể làm hư hỏng thiết bị. Từ thực tế đó, nhóm lựa chọn đề tài **“Hệ thống cảm biến mực nước”** – một ứng dụng đơn giản nhưng hữu ích của IoT, giúp theo dõi mực nước trong bể chứa một cách tự động, góp phần tiết kiệm nước, điện năng và tăng tính an toàn trong sinh hoạt.

## Mục đích nghiên cứu

Xây dựng hệ thống cảm biến mực nước sử dụng **Arduino Uno R3 và cảm biến mực nước** kết hợp **Buzzer** để phát âm thanh khi mực nước đã tới ngưỡng từ đó góp phần tiết kiệm năng lượng và nâng cao tiện ích sinh hoạt.

## Đối tượng và phạm vi nghiên cứu

Đề tài tập trung nghiên cứu **hệ thống cảm biến mực nước sử dụng vi điều khiển Arduino Uno R3**, cảm biến **mực nước** và **mạch Buzzer** để tự động phát âm thanh khi mực nước đạt đến ngưỡng quy định.

Các thành phần chính gồm:

* **Cảm biến mực nước: d**ùng để phát hiện và đo mức nước trong bể chứa. Khi nước chạm tới các mức cảm biến xác định
* **Arduino Uno R3:** đóng vai trò xử lý trung tâm, đọc tín hiệu từ cảm biến, xử lý dữ liệu và đưa ra quyết định điều khiển.
* **Mạch Buzzer:** phát âm thanh cảnh báo khi mực nước vượt hoặc thấp hơn ngưỡng cho phép.

Phạm vi nghiên cứu:

* Nghiên cứu ở **mức độ mô hình thử nghiệm (prototype)**, không triển khai trong môi trường thực tế quy mô lớn.
* Tập trung vào **chức năng phát hiện mực nước và cảnh báo tự động**, chưa mở rộng sang điều khiển hoặc giám sát từ xa qua Internet.
* Hệ thống được **quan sát và kiểm thử thông qua Serial Monitor hoặc Dashboard Node-RED**, giúp người dùng theo dõi giá trị mực nước và trạng thái cảnh báo theo thời gian thực.
* Mục tiêu hướng đến là **mô phỏng nguyên lý hoạt động cơ bản của hệ thống giám sát mực nước thông minh**, làm nền tảng cho các nghiên cứu mở rộng trong lĩnh vực IoT (kết nối MQTT, Cloud hoặc điều khiển qua điện thoại thông minh).

## Ý nghĩa khoa học và thực tiễn

Về khoa học:

* Góp phần nghiên cứu và ứng dụng **công nghệ IoT trong lĩnh vực giám sát và điều khiển tự động mực nước**.
* Giúp sinh viên hiểu rõ **nguyên lý hoạt động của cảm biến mực nước**, quy trình **thu thập, xử lý dữ liệu cảm biến** và **điều khiển thiết bị điện** thông qua **vi điều khiển Arduino Uno R3**.
* Là bước khởi đầu để tìm hiểu sâu hơn về **các kỹ thuật điều khiển tự động**, **truyền dữ liệu cảm biến qua mạng (MQTT, HTTP)** và **xây dựng hệ thống IoT hoàn chỉnh** trong tương lai.

Về thực tiễn:

* Hệ thống cảm biến **mực** nước giúp **tự động cảnh báo** ngăn chặn tình trạng **tràn hoặc cạn bể chứa**, giảm thiểu sự phụ thuộc vào con người.
* Góp phần **tiết kiệm điện năng và nguồn nước**, phù hợp cho các ứng dụng trong **hộ gia đình, nông nghiệp, công nghiệp** hoặc **các khu vực cấp nước tập trung**.
* Hệ thống có **chi phí thấp, dễ lắp đặt và mở rộng**, có thể tích hợp thêm **module Relay, Wi-Fi (ESP8266/ESP32)** hoặc **ứng dụng giám sát qua điện thoại**, hướng tới mô hình **IoT giám sát nước thông minh (Smart Water System)**.
* Giúp người học **rèn luyện kỹ năng thực hành điện – điện tử, lập trình vi điều khiển và thiết kế hệ thống IoT thực tế**, từ đó nâng cao khả năng ứng dụng kiến thức vào đời sống và nghiên cứu khoa học.

# GIỚI THIỆU

## Giới thiệu đề tài

Trong bối cảnh **chuyển đổi số** và **phát triển mạnh mẽ của công nghệ Internet of Things (IoT)**, việc ứng dụng các hệ thống cảm biến nhằm **tự động hóa trong giám sát và điều khiển thiết bị** ngày càng trở nên phổ biến. Các hệ thống này không chỉ giúp nâng cao hiệu quả sử dụng tài nguyên mà còn góp phần giảm thiểu sai sót do con người, tối ưu hóa quy trình vận hành và tiết kiệm năng lượng.

Một trong những ứng dụng tiêu biểu của công nghệ IoT trong đời sống là **hệ thống cảm biến mực nước**. Hệ thống này cho phép **giám sát mức nước trong bể chứa theo thời gian thực**, đồng thời **tự động cảnh báo hoặc điều khiển máy bơm** khi mực nước đạt đến ngưỡng giới hạn (cao hoặc thấp). Nhờ đó, người dùng có thể **chủ động kiểm soát nguồn nước**, tránh được các tình huống như **tràn bể, cạn bể**, gây **lãng phí nước** hoặc **ảnh hưởng đến thiết bị điện**.  
 Đề tài **“Hệ thống cảm biến mực nước”** hướng đến việc xây dựng một mô hình IoT đơn giản và hiệu quả, sử dụng vi điều khiển **Arduino Uno R3**, **cảm biến mực nước** và **mạch Buzzer** để **cảnh báo hoặc điều khiển máy bơm tự động**. Hệ thống giúp **tiết kiệm thời gian, công sức, năng lượng** và **nâng cao tính an toàn, tiện nghi trong sinh hoạt.**

## Mục tiêu của hệ thống

* Thiết kế và triển khai **hệ thống cảm biến mực nước tự động** sử dụng **Arduino Uno R3** và **cảm biến mực nước**.
* Xây dựng chương trình để **phát cảnh báo âm thanh qua Buzzer** khi mực nước đạt đến ngưỡng giới hạn (cao/thấp).
* Thu thập và hiển thị dữ liệu mực nước trên **Serial Monitor** hoặc **Dashboard**
* Ứng dụng kiến thức **IoT và lập trình nhúng** để **mô phỏng hệ thống giám sát – điều khiển mực nước thông minh**.
* Giúp sinh viên **làm quen với việc kết nối cảm biến, xử lý tín hiệu và điều khiển thiết bị điện tử trong thực tế.**

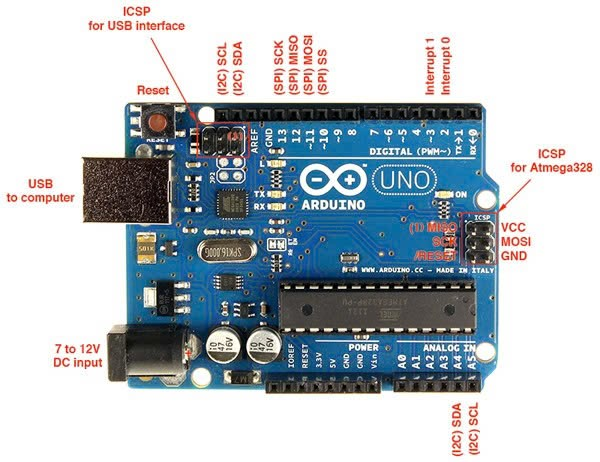
## Lợi ích mang lại

* **Đối với người dùng:** hệ thống giúp **tự động cảnh báo** khi mực nước đạt ngưỡng, **giảm nguy cơ tràn hoặc cạn bể**, tiết kiệm thời gian và công sức.
* **Đối với cộng đồng:** Góp phần **tiết kiệm điện năng và nguồn nước**, **bảo vệ môi trường**, phù hợp với các ứng dụng trong **hộ gia đình, nông nghiệp và công nghiệp**.
* **Đối với người học:** giúp sinh viên **hiểu rõ nguyên lý hoạt động của cảm biến mực nước, lập trình điều khiển vi điều khiển Arduino**, và **rèn luyện kỹ năng thực hành hệ thống IoT thực tế.**
* **Về lâu dài:** Hệ thống có thể **mở rộng thành mô hình giám sát nước thông minh (Smart Water System), tích hợp Wi-Fi (ESP8266/ESP32), điều khiển qua điện thoại,** hoặc **kết nối với nền tảng Cloud IoT** để quản lý tập trung.

# CÔNG NGHỆ SỬ DỤNG

## IoT Device (Thiết bị phần cứng)

### Arduino Uno R3



Hình 1. Arduino Uno R3 – Vi điều khiển trung tâm của hệ thống

Arduino **Uno R3** là một **bo mạch vi điều khiển 8-bit sử dụng chip ATmega328P**, được ứng dụng rộng rãi trong các dự án **IoT và tự động hóa** nhờ **tính dễ lập trình, giá thành thấp và độ ổn định cao**. Bo mạch này có khả năng đọc dữ liệu từ nhiều loại cảm biến khác nhau và điều khiển các thiết bị điện thông qua các chân digital output, rất phù hợp cho các hệ thống điều khiển thông minh quy mô nhỏ.

Trong **hệ thống cảm biến mực nước**, Arduino Uno R3 đóng vai trò **bộ xử lý trung tâm**, thực hiện các nhiệm vụ chính sau:

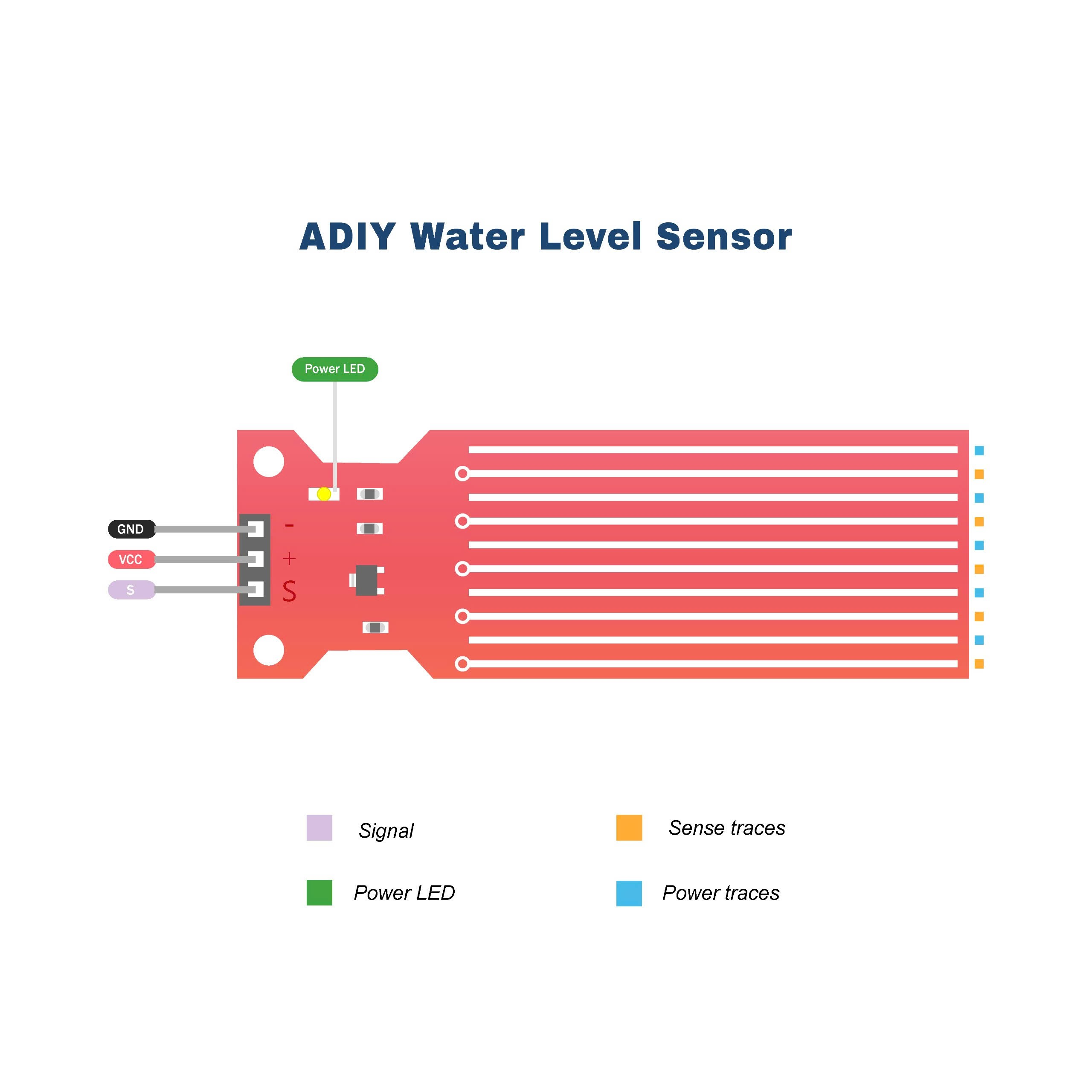
* **Đọc tín hiệu từ cảm biến mực nước** (Water Level Sensor) tại các mức khác nhau.
* **So sánh giá trị thu được với ngưỡng giới hạn** (cao hoặc thấp) được thiết lập sẵn trong chương trình
* K**ích hoạt mạch Buzzer** để **phát cảnh báo âm thanh** hoặc khi mực nước đạt đến giới hạn tương ứng.

Arduino Uno R3 được lựa chọn vì có **đầy đủ các chân analog và digital, dễ dàng kết nối** với cảm biến, relay, buzzer và các thiết bị ngoại vi khác, đồng thời có **thư viện lập trình phong phú**, hỗ trợ tốt cho việc phát triển mô hình IoT.

**Ví dụ hoạt động:**

* Khi mực nước đạt đến ngưỡng cao, cảm biến gửi tín hiệu điện áp cao.  
  → Arduino gửi **tín hiệu LOW** để **kích hoạt Buzzer** phát cảnh báo ngắn.

### Cảm biến ****mực**** nước (Water Level Sensor)



Hình 2. Cảm biến **mực** nước

Cảm biến **mực** nước là **thiết bị dùng để đo mức nước trong bể chứa**, hoạt động dựa trên **nguyên lý dẫn điện của nước**. Khi mực nước thay đổi, lượng điện áp hoặc tín hiệu analog trả về cảm biến cũng thay đổi tương ứng, giúp **vi điều khiển Arduino** xác định được mực nước hiện tại (thấp, trung bình, cao).

**Thông số cơ bản:**

* Điện áp hoạt động: 3.3V – 5V
* Tín hiệu xuất ra Analog (thay đổi theo mức nước)
* **Dòng tiêu thụ:** < 20mA
* Độ **nhạy:** Cao, phản ứng nhanh với sự thay đổi mực nước.
* **Kết nối:** 3 chân (VCC, GND, AO – tín hiệu analog)

**Nguyên lý hoạt động:**

* Khi **mực nước thấp** → phần đầu của cảm biến không tiếp xúc nước → điện trở cao → **giá trị analog trả về nhỏ.**
* Khi **mực nước trung bình** → cảm biến tiếp xúc một phần nước → điện trở giảm → **giá trị analog tăng dần.**
* Khi **mực nước cao** → toàn bộ cảm biến ngập nước → điện trở rất thấp → **giá trị analog cao nhất.**

**Ví dụ:**

**Mực nước > 700** → Arduino hiểu là **mức nước cao** → **kích hoạt Buzzer** để **phát tín hiệu cảnh báo**.

### Buzzer (Còi báo)



Hình 3. Buzzer (Còi báo)

Buzzer là thiết bị phát ra âm thanh khi được cấp điện, thường dùng trong các hệ thống cảnh báo. Trong **hệ thống cảm biến mực nước**, buzzer đóng vai trò **phát tín hiệu âm thanh khi mực nước vượt ngưỡng cho phép** – ví dụ như nước đầy bồn hoặc cạn bình chứa.

**Nguyên lý hoạt động:**

* Khi cảm biến phát hiện **mực nước đạt đến mức tối đa hoặc tối thiểu**, Arduino sẽ gửi tín hiệu **mức HIGH** đến chân điều khiển của buzzer.
* Buzzer kêu “bíp” để **cảnh báo người dùng** kiểm tra hoặc ngắt máy bơm.

**Thông số cơ bản:**

* Điện áp hoạt động: 3.3V – 5V
* Dòng tiêu thụ thấp, dễ điều khiển bằng tín hiệu digital từ Arduino
* Có hai loại: **buzzer chủ động (có mạch dao động sẵn)** và **buzzer thụ động (cần phát tín hiệu PWM từ Arduino)**

### Jumper Wires (Dây nối)

A group of colorful wires

AI-generated content may be incorrect.

Hình 4. Jumper Wires (Dây nối)

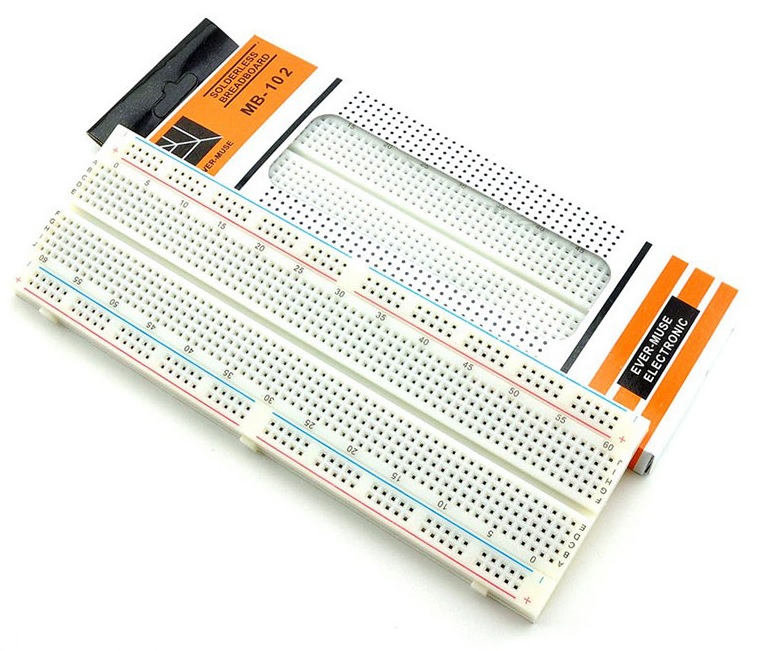
Dây jumper là các dây dẫn điện nhỏ dùng để kết nối giữa **Arduino, cảm biến mực nước và buzzer** trên breadboard.

Có ba loại dây jumper phổ biến: **đực–đực, cái–cái, và đực–cái**, phù hợp với nhiều kiểu đầu nối khác nhau.

**Ví dụ:**

* Một đầu dây jumper cắm vào chân A0 của Arduino để đọc tín hiệu cảm biến **mực** nước, đầu còn lại cắm vào chân tín hiệu của cảm biến trên Breadboard (Bảng mạch thử).
* Các dây còn lại kết nối với chân VCC (5V) và GND để cung cấp nguồn cho cảm biến.

### Bảng kết nối mạch (Breadboard)



Hình 5. Breadboard – Bảng kết nối mạch

Breadboard là công cụ giúp lắp ráp và kiểm tra **mạch điện tạm thời** mà không cần hàn. Trong hệ thống **cảm biến mực nước**, breadboard được sử dụng để **kết nối Arduino, cảm biến mực nước, Buzzer và dây jumper**, giúp dễ dàng thay đổi cấu trúc mạch hoặc khắc phục lỗi trong quá trình thử nghiệm.

**Lợi ích:**

* Dễ tháo lắp, thử nghiệm nhanh.
* Tiết kiệm thời gian thiết kế mạch.
* Giúp sinh viên dễ dàng học và thực hành.

## Development Tool (Công cụ lập trình)

### Arduino IDE



Hình 6. Arduino IDE (Integrated Development Environment)

**Arduino IDE (Integrated Development Environment)** là công cụ chính để lập trình cho các bo mạch như Arduino Uno R3, ESP8266, ESP32,.  
IDE cung cấp giao diện trực quan giúp người dùng:

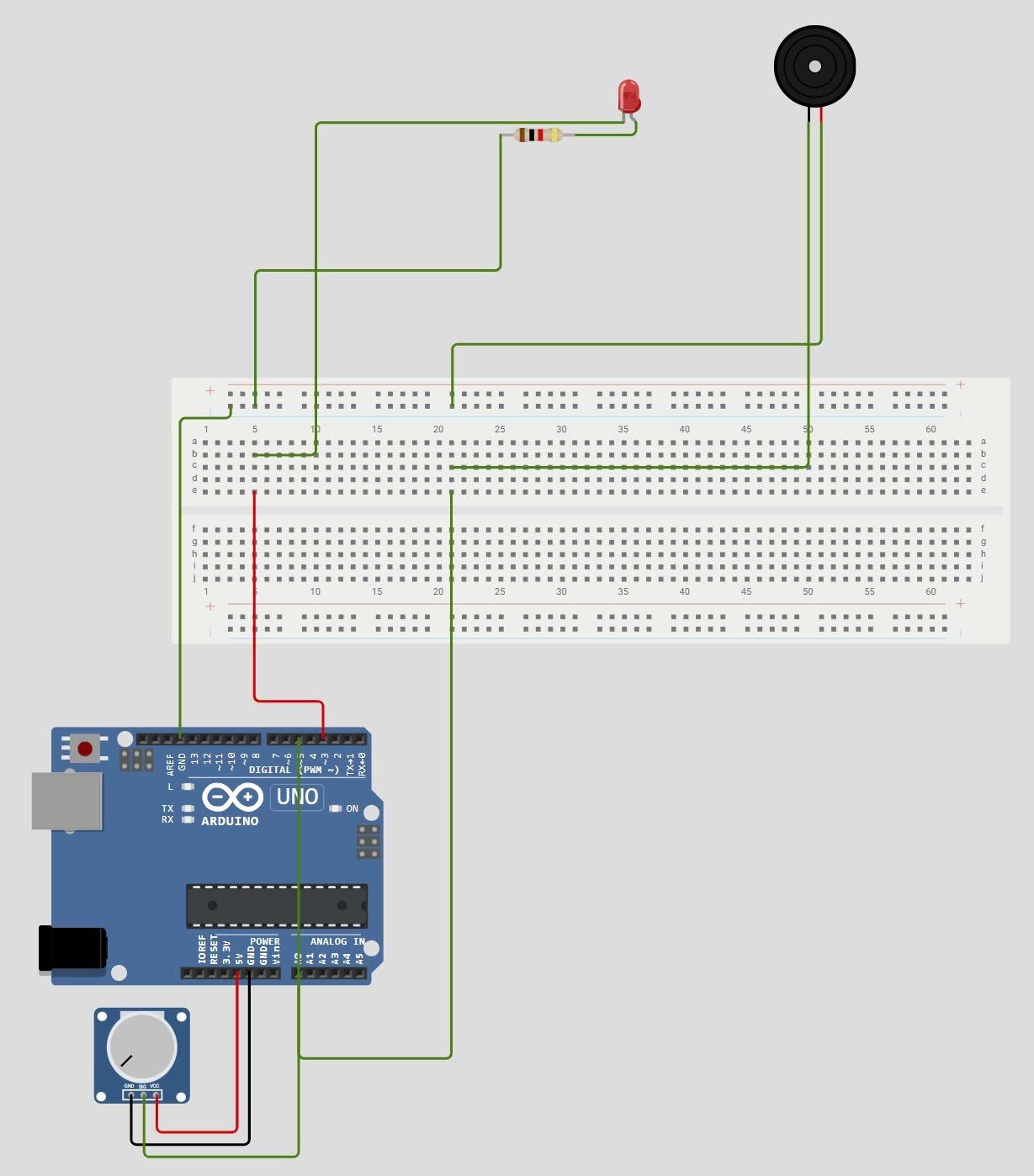
* Viết mã nguồn C/C++ cho Arduino.
* Biên dịch, nạp chương trình trực tiếp qua cáp USB.
* Theo dõi kết quả từ cảm biến qua **Serial Monitor**.

**Ưu điểm:**

* Dễ sử dụng cho người mới bắt đầu.
* Miễn phí, hỗ trợ nhiều thư viện sẵn có.
* Tương thích với các module và cảm biến phổ biến trong IoT.

**Ví dụ:** Nhóm sử dụng Arduino IDE để viết chương trình đọc giá trị mực nước, so sánh với ngưỡng định trước và phát cảnh báo tương ứng.

**Sơ đồ kết nối phần cứng:**

****

Hình 7. Sơ đồ đấu nối hệ thống cảm biến **mực** nước

# **KIẾN TRÚC HỆ THỐNG**

## Mô hình kiến trúc và tương tác hệ thống

### Sơ đồ kiến trúc tổng thể

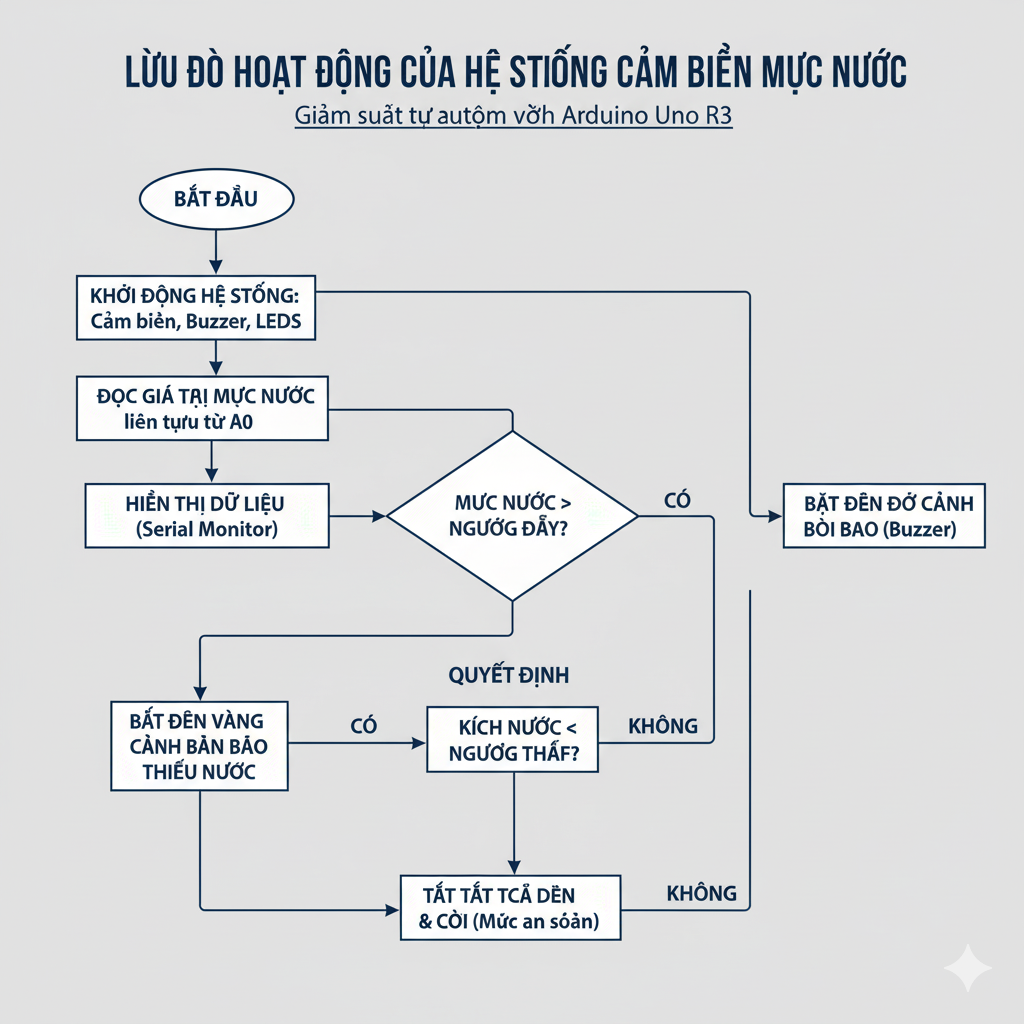
  
Hình 8. Sơ đồ kiến trúc tổng thể hệ thống cảm biến **mực** nước

Hệ thống cảm biến **mực** nước được thiết kế dựa trên mô hình **IoT cơ bản**, gồm các thành phần chính sau:

1. **Cảm biến mực nước (Water Level Sensor):** Đo mức nước trong bể chứa và gửi tín hiệu tương tự (analog) về vi điều khiển Arduino.
2. **Vi điều khiển Arduino Uno R3:** Đọc dữ liệu từ cảm biến, so sánh với ngưỡng cài đặt trước, **buzzer** để phát cảnh báo.
3. **Dashboard hoặc Serial Monitor:** Hiển thị giá trị mực nước và trạng thái hoạt động của hệ thống theo thời gian thưc, hỗ trợ giám sát và kiểm tra quá trình vận hành.

Toàn bộ quá trình hoạt động của hệ thống diễn ra **tự động** mà **không cần sự can thiệp thủ công**. Arduino Uno R3 đóng vai trò trung tâm, xử lý dữ liệu cảm biến và điều phối tín hiệu điều khiển đến các thiết bị đầu ra, đảm bảo hệ thống vận hành ổn định, an toàn và hiệu quả.

## Quy trình hoạt động của hệ thống



Hình 9. Lưu đồ hoạt động của hệ thống cảm biến **mực** nước

Hệ thống cảm biến **mực** nước hoạt động theo chu trình khép kín, gồm các bước như sau:

**Bước 1: Khởi động hệ thống**

* Khi cấp nguồn cho Arduino Uno R3, vi điều khiển khởi tạo cảm biến **mực** nước, buzzer.
* Arduino bắt đầu đọc giá trị mực nước liên tục thông qua **chân analog (A0).**
* Hệ thống sẵn sàng hoạt động ở chế độ **tự động giám sát và điều khiển mực nước**.

**Bước 2: Đọc dữ liệu cảm biến mực nước**

* **Cảm biến mực nước** đo mức nước trong bể (thông qua điện trở thay đổi theo độ cao nước) và gửi tín hiệu điện áp tương tự (analog) về Arduino.
* A**rduino** đọc giá trị này (từ 0 đến 1023) bằng hàm “analogRead(A0)”.
* Giá trị càng lớn → mực nước càng cao; giá trị càng nhỏ → mực nước càng thấp.

**Bước 3: Xử lý và so sánh dữ liệu**

* Arduino so sánh giá trị mực nước với một ngưỡng cài đặt trước (ví dụ: 500).
* Nếu giá trị nhỏ hơn ngưỡng thấp → xác định bể đang thiếu nước.
* Nếu giá trị lớn hơn ngưỡng cao → xác định bể đã đầy.

**Bước 4: Điều khiển buzzer**

* Khi **nước đầy** → Arduino xuất tín hiệu **LOW** → **Buzzer phát âm thanh cảnh báo**.

**Bước 5: Hiển thị và giám sát dữ liệu**

* Giá trị mực nước và trạng thái (cảnh báo hoạt động) được hiển thị trên **Serial Monitor** hoặc **Dashboard**
* Người dùng có thể **quan sát mức nước theo thời gian thực** và **điều chỉnh ngưỡng cài đặt** để phù hợp với nhu cầu thực tế.

## Code chương trình điều khiển Arduino



Hình 10. Code chương trình

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Tóm tắt kết quả đạt được

Sau quá trình thiết kế, lắp ráp và thử nghiệm, **hệ thống cảm biến mực nước tự động cảnh báo** đã đạt được nhiều kết quả tích cực cả về **phần cứng** lẫn **phần mềm.**

**Về phần cứng:** Nhóm đã xây dựng thành công **mạch điện gồm Arduino Uno R3, cảm biến mực nước, Buzzer**. Mạch hoạt động ổn định, có thể **phát hiện chính xác mức nước trong bể chứa** và **phát âm thanh cảnh báo** khi nước đầy hoặc cạn. Hệ thống phản hồi nhanh, chính xác nhờ sử dụng giữa mạch điều khiển và mạch tải điện, đảm bảo tính ổn định trong suốt quá trình thử nghiệm.

**Về phần mềm:** Chương trình được lập trình bằng **Arduino IDE**, thực hiện tốt nhiệm vụ **đọc dữ liệu cảm biến**, **so sánh giá trị với ngưỡng định sẵn**, và **điều khiển buzzer** tương ứng. Dữ liệu mực nước được hiển thị và cập nhật liên tục trên **Serial Monitor** hoặc **Dashboard**, giúp người dùng dễ dàng **quan sát trạng thái mực nước theo thời gian thực**.

**Kết quả thu được:** chứng minh rằng **mô hình cảm biến mực nước** có thể ứng dụng hiệu quả trong các hệ thống **công nghiệp quy mô nhỏ**, giúp **tiết kiệm thời gian, điện năng và nguồn nước,** đồng thời **nâng cao tính an toàn và tiện nghi sinh hoạt.**

Hệ thống có **giá thành thấp, dễ lắp đặt, dễ mở rộng**, rất phù hợp cho mục đích **học tập, nghiên cứu hoặc ứng dụng thực tế trong IoT.**

## Kiến nghị và hướng phát triển tiếp theo

Trong tương lai, nhóm đề xuất một số hướng phát triển nhằm **nâng cao tính ứng dụng và mở rộng phạm vi hoạt động** của hệ thống:

* **Tích hợp module Wi-Fi (ESP8266 hoặc ESP32):** giúp gửi dữ liệu mực nước lên **Cloud hoặc Dashboard web**, phục vụ giám sát và điều khiển **qua Internet**.
* **Điều khiển máy bơm qua điện thoại di động:** cho phép người dùng **bật/tắt thủ công** ngoài chế độ tự động, hướng đến mô hình **Smart Water Control System**.
* **Bổ sung cảm biến siêu âm (Ultrasonic Sensor):** để **đo mực nước chính xác hơn** thay vì cảm biến điện trở thông thường.
* **Ứng dụng trong các môi trường thực tế:** như **bể nước hộ gia đình, nông trại tưới tiêu, hệ thống cấp nước tòa nhà hoặc nhà máy.**
* **Phân tích dữ liệu IoT:** thu thập và lưu trữ dữ liệu mực nước theo thời gian để **dự đoán xu hướng sử dụng** và **tối ưu chu kỳ bơm nước.**

Nhìn chung, đề tài **“Hệ thống cảm biến mực nước”** không chỉ giúp sinh viên hiểu rõ quy trình thiết kế – lập trình – kiểm thử – vận hành một mô hình IoT (Internet of Things) cơ bản, mà còn là **bước đệm quan trọng** để tiếp cận và phát triển các hệ thống **giám sát – điều khiển tự động hóa thông minh** trong thực tế.

Thông qua quá trình nghiên cứu và thực hành, nhóm đã nắm vững được nguyên lý hoạt động của cảm biến **mực** nước, cơ chế xử lý tín hiệu của vi điều khiển Arduino, cũng như quy trình kết nối và điều khiển thiết bị đầu ra còi cảnh báo, đèn báo. Hệ thống đã chứng minh khả năng hoạt động **ổn định, chính xác và kịp thời** trong việc phát hiện thay đổi mực nước, giúp tự động cảnh báo khi mực nước vượt ngưỡng an toàn.

Bên cạnh đó, mô hình còn thể hiện rõ tiềm năng ứng dụng trong thực tế, đặc biệt trong các lĩnh vực như **giám sát bể chứa nước gia đình, hệ thống tưới tiêu tự động, quản lý hồ chứa hoặc phòng chống ngập lụt đô thị**. Nhờ tính linh hoạt, chi phí thấp và dễ mở rộng, hệ thống cảm biến **mực** nước có thể trở thành một phần quan trọng trong các giải pháp IoT thông minh phục vụ đời sống và sản xuất.

Từ những kết quả đạt được, có thể khẳng định rằng đề tài không chỉ giúp sinh viên rèn luyện tư duy logic, kỹ năng lập trình và xử lý dữ liệu cảm biến, mà còn hình thành nền tảng kiến thức vững chắc để tiếp cận các xu hướng công nghệ hiện đại trong kỷ nguyên Công nghiệp 4.0 – nơi mà **mọi thiết bị đều có thể “giao tiếp” và vận hành thông minh** nhằm phục vụ con người một cách hiệu quả và bền vững hơn.

# TÀI LIỆU THAM KHẢO

1. **ArduinoKit Việt Nam. Cảm biến mực nước sử dụng Arduino.**

<https://arduinokit.vn/cach-thuc-hoat-dong-cua-cam-bien-muc-nuoc-water-sensor-voi-arduino/>

1. **Arduino Official Documentation.** Arduino Uno Board Overview.

<https://www.arduino.cc/en/Guide/ArduinoUno>

1. **Random Nerd Tutorials.** Guide for Relay Module with Arduino.

<https://randomnerdtutorials.com/guide-for-relay-module-with-arduino/>

1. **Maker Advisor.** Water Level Sensor with Arduino – Detecting Water Presence and Depth.

[https://makeradvisor.com/water-sensor-arduino/](file:///C:\Users\ASUS\AppData\Local\Temp\Zalo%20Temp\TempDownloads\Nhom14_DeTai_01.docx)

1. **Node-RED Documentation.** Introduction to Node-RED for IoT Applications.

<https://nodered.org/docs/>