

TRƯỜNG ĐẠI HỌC KHOA HỌC HUẾ

KHOA CÔNG NGHỆ THÔNG TIN



PHÁT TRIỂN ỨNG DỤNG IOT

ĐỀ TÀI:

HỆ THỐNG ĐO MỨC NƯỚC THÔNG MINH BẰNG ESP32

Tên lớp học phần: Phát triển ứng dụng IOT

Mã học phần: TIN4024.006

Giáo viên hướng dẫn: Võ Việt Dũng

Sinh viên thực hiện: Phan Ngọc Lân

Huế, tháng 4 năm 2025

MỤC LỤC

CHƯƠNG 1: MỞ ĐẦU	4
1. Giới thiệu	4
1.1. Tầm quan trọng của việc giám sát mực nước	4
1.2. Các vấn đề của phương pháp đo mực nước truyền thống	4
1.3. Lợi ích của hệ thống giám sát thông minh	4
2. Tổng quan về dự án	4
2.1. Mục tiêu dự án	4
2.2. Phạm vi nghiên cứu	4
2.2.1. Giới hạn về phần cứng	4
2.2.2. Giới hạn về phần mềm	5
2.2.3. Giới hạn về ứng dụng	5
3. Mô tả hệ thống	5
CHƯƠNG 2: NỘI DUNG	7
1. Cơ sở lý thuyết	7
1.1. Khái niệm IoT	7
1.2. Ứng dụng IoT trong giám sát môi trường:	8
1.3. Xu hướng phát triển:	8
1.4. Giới thiệu ESP32	8
1.5. Cảm biến siêu âm HC-SR04	9
1.6. Nền tảng giám sát Telegram	10
1.7. Nền tảng Wokwi	11
1.8. Các thiết bị khác (LDC)	11
2. Thiết kế hệ thống	12
2.1. Thành phần hệ thống	12
2.2. Mô hình hoạt động	12

2.2.1. Kiến trúc tổng thể	12
2.2.2. Luồng hoạt động hệ thống	13
2.2.3. Hoạt động liên tục và tương tác hai chiều	16
2.2.4. Ưu điểm của mô hình	16
2.3. Sơ đồ Wokwi	16
CHƯƠNG 3: KẾT LUẬN	17
1. Tóm tắt	17
2. Khó khăn và hạn chế	18
3. Hướng phát triển tương lai	18
TÀI LIỆU THAM KHẢO	20
PHỤ LỤC	21

CHƯƠNG 1: MỞ ĐẦU

1. Giới thiệu

1.1. Tầm quan trọng của việc giám sát mực nước

- Trong công nghiệp: Kiểm soát mực nước trong bồn chứa, hệ thống làm mát, xử lý nước thải.
- Trong nông nghiệp: Hệ thống tưới tiêu tự động, nuôi trồng thủy sản.
- Trong đời sống: Bể nước sinh hoạt, hồ bơi, hệ thống cảnh báo ngập lụt.

1.2. Các vấn đề của phương pháp đo mực nước truyền thống

- Phải kiểm tra thủ công, tốn nhân lực.
- Khó giám sát liên tục và từ xa.
- Phản ứng chậm với các tình huống khẩn cấp (tràn, cạn).
- Thiếu khả năng lưu trữ dữ liệu để phân tích xu hướng.

1.3. Lợi ích của hệ thống giám sát thông minh

- Giám sát từ xa, theo thời gian thực.
- Tự động hóa quy trình, giảm sai sót do con người.
- Cảnh báo kịp thời khi có bất thường.
- Thu thập dữ liệu để phân tích và tối ưu hóa sử dụng nước.

2. Tổng quan về dự án

2.1. Mục tiêu dự án

- Thiết kế và xây dựng hệ thống đo mức nước sử dụng vi điều khiển ESP32 và cảm biến siêu âm HC-SR04.
- Phát triển giải pháp truyền dữ liệu thu thập được từ cảm biến lên nền tảng đám mây IoT một cách ổn định và hiệu quả.
- Xây dựng hệ thống cảnh báo khi mức nước đạt đến hoặc vượt ngưỡng cài đặt sẵn.
- Phát triển giao diện người dùng (trên Web hoặc ứng dụng) cho phép theo dõi dữ liệu theo thời gian thực và điều khiển từ xa.
- Đánh giá hiệu suất hệ thống trong các điều kiện thực tế (độ chính xác, độ trễ, tính ổn định...).

2.2. Phạm vi nghiên cứu

2.2.1. Giới hạn về phần cứng

- Sử dụng ESP32 làm vi điều khiển chính.
- Sử dụng cảm biến siêu âm HC-SR04 để đo mức nước trong khoảng từ 2cm đến 400cm.
- Sử dụng các linh kiện phụ trợ như màn hình LCD để hiển thị dữ liệu.

2.2.2. Giới hạn về phần mềm

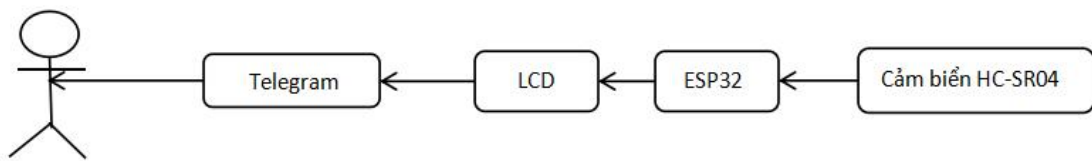
- Nền tảng đám mây IoT Telegram.
- Giao thức truyền dữ liệu (HTTP...).

2.2.3. Giới hạn về ứng dụng

- Hệ thống được thiết kế để hoạt động trong một khoảng đo cụ thể (ví dụ: từ 2cm đến 400cm).
- Hoạt động ổn định trong điều kiện môi trường tiêu chuẩn, nhiệt độ và độ ẩm nằm trong dải phù hợp với linh kiện điện tử.

3. Mô tả hệ thống

- Sơ đồ tổng quát:



- Quá trình hoạt động:
 - o Cảm biến siêu âm HC-SR04 phát tín hiệu sóng siêu âm và đo thời gian phản hồi để tính khoảng cách từ cảm biến đến mặt nước.
 - o ESP32 nhận dữ liệu từ cảm biến, tính toán khoảng cách, từ đó xác định mức nước hiện tại.
 - o Mức nước được hiển thị trên màn hình LCD.
 - o Dữ liệu mức nước được gửi lên nền tảng Telegram mỗi 5 phút thông qua HTTP request (hoặc Webhook).
 - o Nếu mức nước vượt ngưỡng cài đặt thì dữ liệu cảnh báo sẽ được gửi lên nền tảng Telegram ngay lập tức.
 - o Người dùng có thể nhận cảnh báo qua Telegram và theo dõi dữ liệu theo thời gian thực, tùy chỉnh ngưỡng cảnh báo.

- Có thể lưu trữ dữ liệu định kỳ để phục vụ phân tích xu hướng, hiệu suất hoạt động, và tối ưu hóa việc sử dụng nước.

CHƯƠNG 2: NỘI DUNG

1. Cơ sở lý thuyết

1.1. Khái niệm IoT

- Định nghĩa: Internet of Things (IoT) là một mạng lưới các thiết bị vật lý — như cảm biến, thiết bị gia dụng, xe cộ, máy móc công nghiệp — được kết nối với Internet, có khả năng thu thập, gửi và nhận dữ liệu mà không cần sự can thiệp trực tiếp của con người.
- Thành phần cơ bản:
 - Cảm biến (Sensors): Thu thập thông tin từ môi trường (nhiệt độ, độ ẩm, ánh sáng, chuyển động, v.v.).
 - Vi điều khiển (Microcontroller): Xử lý dữ liệu cảm biến, điều khiển thiết bị (ESP32, Arduino, Raspberry Pi...).
 - Kết nối mạng: Cho phép thiết bị giao tiếp với nhau và với máy chủ (Wi-Fi, Bluetooth, Zigbee, 4G...).
 - Nền tảng đám mây (Cloud): Lưu trữ, phân tích và xử lý dữ liệu từ thiết bị IoT.
 - Phần mềm ứng dụng (App/Web): Giao diện người dùng để giám sát, điều khiển, tương tác với hệ thống IoT.
- Kiến trúc IoT:
 - Tầng cảm biến (Perception Layer): Là tầng thấp nhất, gồm các cảm biến, bộ thu dữ liệu, thiết bị đo đạc, có nhiệm vụ thu thập dữ liệu từ môi trường thực.
 - Tầng mạng (Network Layer): Truyền dữ liệu từ tầng cảm biến đến tầng xử lý, sử dụng các công nghệ truyền thông như Wi-Fi, Ethernet, Zigbee, LoRa, 5G...
 - Tầng xử lý dữ liệu (Processing Layer / Middleware): Xử lý, lưu trữ và phân tích dữ liệu (thường sử dụng Cloud như AWS, Azure, Firebase, ThingSpeak...), có thể sử dụng trí tuệ nhân tạo, học máy để phân tích nâng cao.

- Tầng ứng dụng (Application Layer): Giao diện tương tác người dùng, cung cấp các dịch vụ theo mục đích cụ thể như: nhà thông minh, giám sát nông nghiệp, theo dõi sức khỏe, công nghiệp tự động hóa...

1.2. Ứng dụng IoT trong giám sát môi trường:

- Giám sát chất lượng nước.
- Hệ thống tưới tiêu thông minh.
- Cảnh báo ngập lụt.
- Quản lý tài nguyên nước...

1.3. Xu hướng phát triển:

- Tích hợp AI/ML để phân tích dữ liệu.
- IoT và Edge Computing.
- Bảo mật và quyền riêng tư trong IoT.
- Tiêu chuẩn hóa trong IoT.

1.4. Giới thiệu ESP32

- ESP32 là bộ vi điều khiển có tích hợp WiFi, mạnh hơn nhiều so với Arduino thông thường. Nó vừa có thể đọc dữ liệu từ cảm biến, vừa kết nối Internet và xử lý logic thông minh, như kiểm tra ngưỡng nước để gửi cảnh báo.
- So với Arduino, ESP8266, STM32, ESP32 có tốc độ nhanh hơn và hỗ trợ nhiều tính năng hơn, đặc biệt là khả năng xử lý song song nhờ hai nhân CPU

Tiêu chí	ESP32	ESP8266	Arduino Uno (ATmega328)	STM32F1 (Blue Pill)
CPU	Dual-core 240MHz	Single- core 80MHz	16 MHz	72 MHz
RAM	~520KB	~80KB	2KB	20KB
Wi-Fi	Có	Có	Không	Không (cần module)
Bluetooth	Có	Không	Không	Không
GPIO	>30 chân	~17 chân	14 chân	~37 chân

-
- Một số tính năng nổi bật của ESP32:
 - Xử lý song song với hai lõi.
 - Tích hợp WiFi và Bluetooth Không cần mua thêm module như với Arduino, giảm chi phí và tiết kiệm không gian trên mạch.
 - Hỗ trợ nhiều giao thức truyền thông: SPI, I2C, UART, ADC, DAC, PWM, giúp dễ dàng kết nối với cảm biến và các thiết bị ngoại vi.
 - Tiêu thụ điện năng thấp: Có chế độ tiết kiệm năng lượng Deep Sleep, phù hợp cho các ứng dụng chạy bằng pin.
 - Phạm vi nhiệt độ hoạt động rộng từ -40°C đến +125°C, phù hợp với môi trường công nghiệp.
 - Bảo mật (mã hóa, boot an toàn...).
- Với hai nhân xử lý và hỗ trợ đa tác vụ, ESP32 cực kỳ phù hợp cho các ứng dụng IoT chẳng hạn như triển khai hệ thống giám sát mực nước thông minh.

1.5. Cảm biến siêu âm HC-SR04

- Nguyên lý hoạt động:
 - Phát và thu sóng siêu âm.

- Công thức tính khoảng cách: Khoảng cách = (Thời gian × Vận tốc âm thanh) / 2.
- Cấu tạo: phần phát (transmitter) và phần thu (receiver).
- Thông số kỹ thuật
 - Điện áp hoạt động: 5V DC.
 - Dòng điện tiêu thụ: <15mA.
 - Tần số siêu âm: 40kHz.
 - Góc đo: <15°.
 - Phạm vi đo: 2cm - 400cm.
 - Độ chính xác: ~3mm.
- Ưu điểm:
 - Giá thành thấp.
 - Dễ sử dụng và tích hợp.
 - Không cần tiếp xúc với chất lỏng.
 - Phù hợp với nhiều loại chất lỏng khác nhau.
- Nhược điểm:
 - Bị ảnh hưởng bởi nhiệt độ và độ ẩm môi trường.
 - Hạn chế về góc đo.
 - Có thể bị nhiễu do bề mặt phản xạ không đều.
 - Không phù hợp với môi trường có nhiều bọt khí.
- Nhược điểm:
 - Bù nhiệt độ cho vận tốc âm thanh.
 - Kỹ thuật lọc nhiễu và trung bình hóa kết quả.
 - Hiệu chỉnh góc đo

1.6. Nền tảng giám sát Telegram

- Telegram là một nền tảng nhắn tin phổ biến, hỗ trợ bot để gửi và nhận dữ liệu tự động, rất phù hợp cho các ứng dụng IoT. Trong dự án này, Telegram được sử dụng để gửi cảnh báo và thông báo tình trạng mực nước từ ESP32 đến người dùng, giúp người dùng giám sát hệ thống từ xa qua điện thoại.
- Lợi ích khi sử dụng Telegram::

- Dễ triển khai: Không cần giao diện phức tạp, chỉ cần tạo bot và lấy mã token.
- Nhận thông báo tức thì: Người dùng có thể nhận tin nhắn cảnh báo (ví dụ: mực nước quá thấp hoặc tràn bể) gần như ngay lập tức.
- Đa nền tảng: Hoạt động trên điện thoại, máy tính bảng và cả trình duyệt web.
- Miễn phí và bảo mật: Telegram cung cấp API mạnh mẽ mà không tính phí, lại có tính bảo mật cao.

1.7. Nền tảng Wokwi

- Wokwi là nền tảng mô phỏng mạnh mẽ, cho phép người dùng kiểm tra hệ thống IoT với ESP32 mà không cần phần cứng.
- Ưu điểm của Wokwi:
 - Hỗ trợ ESP32: Mô phỏng đầy đủ cả chức năng broker và client của ESP32.
 - Tiết kiệm chi phí: Thay vì mua thiết bị thực, Wokwi cung cấp mô hình ảo để thử nghiệm.
 - Giao diện thân thiện: Cung cấp môi trường mã hóa trực quan để kiểm tra hoạt động của hệ thống.

1.8. Các thiết bị khác (LDC)

- Thiết bị LCD:
 - Hiển thị thông tin mực nước theo thời gian thực, giúp người dùng quan sát trực tiếp trên thiết bị mà không cần mở ứng dụng.
 - Loại màn hình: LCD 16 ký tự, 2 dòng.
 - Điện áp hoạt động: 5V DC.
 - Giao tiếp: Có thể dùng song song hoặc I2C (nếu có module chuyển đổi).
 - Độ tương phản có thể điều chỉnh bằng biến trở.
 - Giá rẻ, dễ sử dụng
 - Hiển thị thông tin trực quan
 - Có thể kết hợp với nhiều loại vi điều khiển như Arduino, ESP32
 - Hiển thị đơn giản, chỉ phù hợp cho văn bản cơ bản

- Khó đọc nếu ánh sáng yếu hoặc góc nhìn không tốt

2. Thiết kế hệ thống

2.1. Thành phần hệ thống

- Hệ thống được xây dựng dựa trên các thành phần chính sau:
- Phần cứng:
 - ESP32:
 - Vi điều khiển trung tâm, có khả năng kết nối Wi-Fi, xử lý dữ liệu từ cảm biến và gửi thông tin đến người dùng thông qua Telegram..
 - Ưu điểm: Hiệu năng cao, tích hợp Wi-Fi, hỗ trợ nhiều giao thức truyền thông.
 - Cảm biến siêu âm HC-SR04:
 - Đo khoảng cách từ cảm biến đến bề mặt nước để tính toán mực nước.
 - Kết nối trực tiếp với ESP32 thông qua các chân digital.
 - Màn hình LCD:
 - Hiển thị thông tin mực nước theo thời gian thực.
 - Giao tiếp qua I2C giúp tiết kiệm chân kết nối.
 - Nguồn cung cấp điện: Đảm bảo cung cấp năng lượng cho ESP32 và các thiết bị ngoại vi.
- Phần mềm:
 - Telegram Bot và Internet Wi-Fi
 - Gửi cảnh báo và cập nhật mực nước qua ứng dụng Telegram đến người dùng.
 - Giúp người dùng nhận thông báo từ xa nhanh chóng và tiện lợi.
 - Giúp người dùng có thể tùy chỉnh ngưỡng cảnh báo.
- Wokwi Platform: Dùng để mô phỏng hệ thống trước khi triển khai thực tế...

2.2. Mô hình hoạt động

2.2.1. Kiến trúc tổng thể

- Kiến trúc hệ thống đo mực nước thông minh sử dụng ESP32 được thiết kế theo mô hình IoT kết nối trực tiếp đến người dùng thông qua Telegram. Hệ thống bao gồm 3 lớp chính:
 - Lớp cảm biến (Sensor Layer)
 - Cảm biến siêu âm HC-SR04 liên tục đo khoảng cách từ cảm biến đến bề mặt nước.
 - Dữ liệu khoảng cách này được xử lý để tính ra mực nước thực tế trong bể
 - Lớp xử lý và điều khiển (Control Layer)
 - ESP32 đóng vai trò trung tâm, thu thập dữ liệu từ cảm biến.
 - Dựa vào mực nước đo được, ESP32 sẽ:
 - Hiển thị lên LCD.
 - Phân tích và gửi dữ liệu lên nền tảng Telegram
 - Lớp người dùng (User Layer)
 - Người dùng sử dụng Telegram trên điện thoại để:
 - Nhận cảnh báo khi mực nước vượt ngưỡng cài đặt.
 - Theo dõi mực nước theo thời gian thực.
 - Tùy chỉnh ngưỡng cảnh báo.

2.2.2. Luồng hoạt động hệ thống

- Hệ thống vận hành theo một chuỗi các bước cụ thể dưới đây:

Bước 1: Kết nối ban đầu

- ESP32 khởi động và kết nối Wi-Fi:
 - ESP32 khởi tạo các thành phần: cảm biến HC-SR04, màn hình LCD, đèn LED và kết nối Wi-Fi.
- Kết nối ứng dụng Telegram với ESP32:
 - Ứng dụng Telegram được cấu hình để giao tiếp với ESP32 thông qua Telegram Bot. Bot này được tạo bằng BotFather, có Bot Token riêng dùng để xác thực và kết nối.

- ESP32 sử dụng thư viện như UniversalTelegramBot để xử lý lệnh và gửi dữ liệu đến Telegram. Dữ liệu được gửi đến Group ID (hoặc Chat ID) xác định người nhận.

Bước 2: Đọc dữ liệu cảm biến

- Thu thập dữ liệu từ cảm biến HC-SR04:
 - ESP32 định kỳ kích hoạt cảm biến HC-SR04 để đo khoảng cách từ cảm biến đến mặt nước
 - Khoảng cách = (Thời gian × Vận tốc âm thanh) / 2.
 - Mức nước = Chiều cao cảm biến HC-SR04 – Khoảng cách đo được.
- Hiển thị dữ liệu lên màn hình LCD:
 - ESP32 hiển thị kết quả đo được ngay trên màn hình LCD
- Gửi dữ liệu lên Telegram:
 - Gửi dữ liệu định kỳ mỗi 5 phút một lần lên telegram, nếu khi mực nước thấp hơn hoặc cao hơn mức tối thiểu đã đặt, hệ thống sẽ tự động cảnh báo,

Bước 3: Người dùng gửi lệnh từ ứng dụng Telegram

- Tương tác trên ứng dụng:
 - Giao diện Telegram trên điện thoại hoặc máy tính cho phép người dùng gửi các lệnh điều khiển và truy vấn thông tin hệ thống, ví dụ:
 - Gửi lệnh điều khiển như: /Status (hiển thị mực nước hiện tại), /setmin 30 (đặt ngưỡng tối thiểu cảnh báo)...
 - Hiển thị tin nhắn cảnh báo từ hệ thống khi mực nước vượt ngưỡng.
 - Theo dõi dữ liệu cảm biến thời gian thực dưới dạng tin nhắn văn bản đơn giản hoặc sử dụng bot để trả lời tự động...
- Gửi lệnh đến ESP32 thông qua Bot Token và Chat ID:
 - Lệnh được gửi từ người dùng đến Telegram Bot (đã cấu hình với Bot Token).

- ESP32 định kỳ kiểm tra các tin nhắn mới thông qua Telegram API (sử dụng thư viện UniversalTelegramBot).
- Khi phát hiện có lệnh mới, ESP32 phân tích nội dung và thực hiện tương ứng

Bước 4: Xử lý lệnh tại ESP32

- Tiếp nhận lệnh từ Telegram:
 - ESP32 truy cập vào Telegram API để lấy danh sách tin nhắn mới.
 - Phân tích cú pháp lệnh từ người dùng để xác định hành động cần thực hiện.
 - Nếu là lệnh bật/tắt, thì bật/tắt GPIO tương ứng với LED.
 - Nếu là lệnh kiểm tra trạng thái (/status), hệ thống sẽ đo mực nước và trả về kết quả.
 - Nếu là lệnh cài ngưỡng (/setmin), giá trị cài đặt sẽ được lưu lại để so sánh và cảnh báo...

Bước 5: Thực thi lệnh tại ESP32

- Xử lý lệnh điều khiển:
 - Ví dụ:
 - Khi nhận lệnh /status ESP32 kích hoạt cảm biến siêu âm HC-SR04 để đo khoảng cách từ cảm biến đến mặt nước.
 - Khoảng cách = (Thời gian × Vận tốc âm thanh) / 2
 - Tính toán mực nước thực tế bằng công thức:
 - Mực nước = Chiều cao cảm biến HC-SR04 – Khoảng cách đo được
 - Khi nhận lệnh /setmin [giá trị] ESP32 cập nhật ngưỡng cảnh báo mực nước tối thiểu.
- Gửi phản hồi về Telegram:
 - Lệnh /status.
 - "Mực nước hiện tại: [giá trị]"
 - Lệnh /setmin [giá trị].

- "Ngưỡng cảnh báo tối thiểu đã được đặt là [giá trị]."

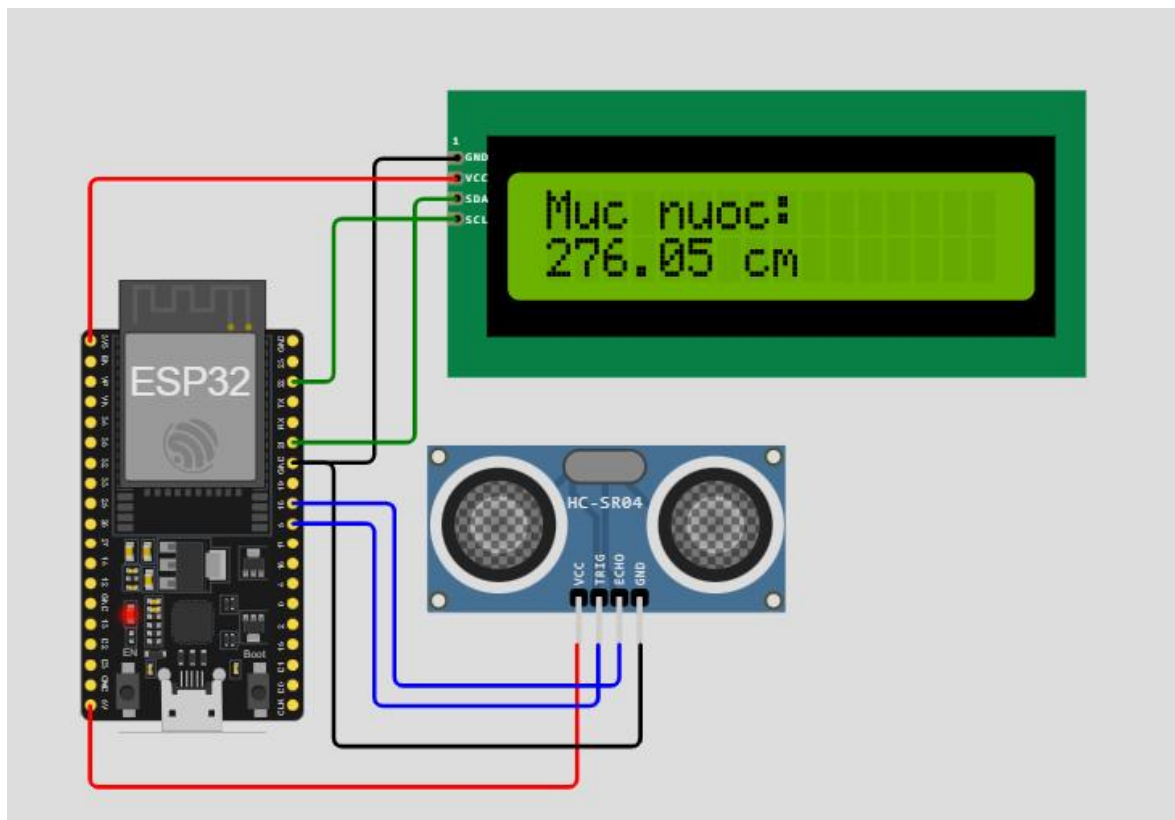
2.2.3. Hoạt động liên tục và tương tác hai chiều

- Hệ thống hoạt động liên tục, định kỳ đo mực nước bằng cảm biến HC-SR04 và cập nhật thông tin lên màn hình LCD và ứng dụng telegram.
- Khi có thay đổi bất thường (ví dụ: mực nước xuống dưới ngưỡng tối thiểu), hệ thống.
 - Tự động cảnh báo người dùng qua Telegram.
- Người dùng có thể tương tác hai chiều với hệ thống qua Telegram.
 - Gửi lệnh điều khiển từ xa: /status, /setmin [giá trị], v.v.
 - Nhận phản hồi theo thời gian thực từ ESP32 về trạng thái thiết bị hoặc mực nước hiện tại.
- Các thao tác được xử lý nhanh chóng nhờ kết nối Wi-Fi và Telegram Bot Token & GroupID riêng.

2.2.4. Ưu điểm của mô hình

- Giám sát từ xa:
 - Người dùng có thể theo dõi mực nước ở bất kỳ đâu chỉ với ứng dụng Telegram.
- Cảnh báo tự động:
 - Hệ thống gửi cảnh báo ngay khi mực nước thay đổi vượt ngưỡng cài đặt, giảm nguy cơ cạn nước hoặc hỏng bơm, hoặc gửi cảnh báo ngay khi mực nước lên cao, giảm nguy cơ đi vào vùng lũ...
- Chi phí thấp:
 - Sử dụng ESP32 và cảm biến HC-SR04 – các linh kiện rẻ, dễ mua, dễ lắp đặt.
- Tương tác linh hoạt:
 - Điều khiển, đặt ngưỡng cảnh báo, xem trạng thái... chỉ cần nhắn tin Telegram
- Điều khiển, đặt ngưỡng cảnh báo, xem trạng thái... chỉ cần nhắn tin Telegram.

2.3. Sơ đồ Wokwi



Hình 1: Sơ đồ Wokwi



Hình 2: Ví dụ nhận thông báo và gửi lệnh ở telegram

CHƯƠNG 3: KẾT LUẬN

1. Tóm tắt

- Đề tài “Hệ thống đo mức nước thông minh bằng ESP32” đã được triển khai thành công với những kết quả nổi bật.
 - Thiết kế hệ thống sử dụng vi điều khiển ESP32 để đo mức nước bằng cảm biến siêu âm HC-SR04, hiển thị thông tin lên màn hình LCD.
 - Kết nối hệ thống với ứng dụng Telegram, cho phép người dùng gửi lệnh điều khiển từ xa (bật/tắt LED, xem mức nước, thiết lập ngưỡng), nhận thông báo trạng thái mức nước theo thời gian thực.
 - Giao diện điều khiển đơn giản, tiện lợi và có khả năng tương tác hai chiều giữa người dùng và thiết bị
- Hệ thống đã góp phần giải quyết bài toán giám sát mức nước từ xa với chi phí thấp, dễ triển khai và hiệu quả cao trong thực tế.

2. Khó khăn và hạn chế

- Độ chính xác của cảm biến HC-SR04 có thể bị ảnh hưởng bởi điều kiện môi trường như nhiệt độ, độ ẩm, bọt khí, bề mặt chất lỏng không bằng phẳng.
- Hệ thống chưa có cơ chế tự động hiệu chỉnh (calibration) theo điều kiện thực tế tại mỗi vị trí lắp đặt.
- Ứng dụng Telegram còn hạn chế về giao diện trực quan so với các nền tảng chuyên biệt cho IoT như Blynk, Firebase,...
- Chưa có tính năng lưu trữ dữ liệu để theo dõi mức nước theo thời gian hoặc thống kê.

3. Hướng phát triển tương lai

- Tích hợp thêm các loại cảm biến khác: như cảm biến nhiệt độ (DHT11/22), cảm biến pH, cảm biến mức nước dạng điện cực để đánh giá chất lượng và trạng thái tổng thể của nước.
- Nâng cao độ chính xác và phạm vi đo: sử dụng cảm biến đo mức nước công nghiệp có độ chính xác cao hơn, áp dụng kỹ thuật lọc nhiễu và hiệu chỉnh nhiệt độ cho HC-SR04.
- Tích hợp trí tuệ nhân tạo (AI): để phân tích dữ liệu mức nước theo thời gian, từ đó dự đoán xu hướng (ví dụ: khi nào sẽ cạn nước, khi nào tràn), phát hiện bất thường như rò rỉ nước hoặc thiết bị hỏng.

- Xây dựng hệ thống quản lý nước toàn diện: theo dõi mực nước ở nhiều bể chứa khác nhau, tự động điều phối bơm/xả nước, kết hợp giám sát chất lượng nước, cảnh báo nguy cơ khan hiếm hoặc lãng phí nước.
- Triển khai lưu trữ dữ liệu đám mây: như Google Firebase hoặc Thingspeak để người dùng có thể xem lại lịch sử và biểu đồ mực nước.

TÀI LIỆU THAM KHẢO

1. **Telegram APIs:** Tài liệu về API telegram .
 - Website: <https://core.telegram.org/api>
2. **Espressif ESP32 Datasheet:** Tài liệu kỹ thuật chi tiết về ESP32, bao gồm đặc điểm, thông số kỹ thuật và cách sử dụng.
 - Website: <https://www.espressif.com/en/products/socs/esp32/resources>
3. **Wokwi Simulator:** Nền tảng mô phỏng các dự án IoT với ESP32, hỗ trợ thử nghiệm mã nguồn và cấu hình phần cứng.
 - Website: <https://wokwi.com>
4. **ESP32 telegram bot:** cách điều khiển đầu ra của ESP32 thông qua ứng dụng Telegram.
 - Website: <https://www.iotzone.vn/esp32/cach-dung-esp32-telegram-dieu-khien-den-led-voi-arduino-ide/>
5. **Wokwi-hc-sr04 Reference:**
 - Website: <https://docs.wokwi.com/parts/wokwi-hc-sr04>
6. **ChatGPT**

PHỤ LỤC

1. Bảng nối chân:

Thiết bị	Chân thiết bị	Kết nối đến ESP32	Màu dây
HC-SR04	VCC	5V	Đỏ
	GND	GND	Đen
	TRIG	ESP:5	Xanh dương
	ECHO	ESP:18	Xanh dương
LCD I2C	VCC	ESP:3V3	Đỏ
	GND	GND	Đen
	SDA	ESP:21	Xanh lá
	SCL	ESP:25	Xanh lá