

**TRƯỜNG ĐẠI HỌC KHOA HỌC
KHOA CÔNG NGHỆ THÔNG TIN**

--- --



**ĐỀ TÀI: “Phát triển hệ thống cảnh báo ngập nước với
ESP32”**

Tên lớp học phần: Phát triển ứng dụng IoT - Nhóm 4

Mã lớp học phần: 2024-2025.2.TIN4024.004

Giảng viên hướng dẫn: Võ Việt Dũng

Sinh viên thực hiện: Trần Quốc Hưng - 21T1020403

HUẾ, THÁNG 04 NĂM 2025

Mục lục

Chương 1: Giới thiệu.....	1
1.1. Đặt vấn đề.....	1
1.2. Mục tiêu của đề tài.....	1
Chương 2: Nội dung và phương pháp thực hiện.....	1
2.1. Các nội dung thực hiện.....	1
2.2. Thiết kế hệ thống IoT.....	1
2.2.1. Sơ đồ logic.....	1
2.2.2. Giải thích chức năng linh kiện.....	1
2.2.3. Nguyên lý hoạt động.....	1
2.2.4. Mạch vật lý.....	1
Chương 3: Cài đặt hệ thống.....	1
3.1. Công cụ sử dụng.....	1
3.2. Các bước cài đặt.....	1
Chương 4: Kết quả thực hiện.....	1
4.1. Hình ảnh và giao diện.....	1
4.2. Phân tích kết quả.....	1
Chương 5: Kết luận và hướng phát triển.....	1
5.1. Các công việc đã thực hiện được.....	1
5.2. Hạn chế.....	1
5.3. Hướng phát triển.....	1

Chương 1: Giới thiệu

1.1. Đặt vấn đề

Trong những năm gần đây, tình trạng ngập nước diễn ra ngày càng nghiêm trọng và thường xuyên tại nhiều khu vực đô thị ở Việt Nam như TP. Hồ Chí Minh, Hà Nội, Đà Nẵng,... Điều này xuất phát từ nhiều nguyên nhân như mưa lớn bất thường do biến đổi khí hậu, hệ thống thoát nước không đáp ứng đủ, quá trình đô thị hóa diễn ra nhanh chóng, cùng với triều cường và nước biển dâng. Tình trạng ngập úng không chỉ ảnh hưởng đến đời sống sinh hoạt của người dân mà còn gây thiệt hại về kinh tế, giao thông và tiềm ẩn nhiều nguy cơ về sức khỏe và an toàn.

Tuy nhiên, hiện nay các hệ thống cảnh báo ngập nước hiệu quả, kịp thời và chi phí thấp vẫn chưa được triển khai rộng rãi, đặc biệt tại các khu dân cư nhỏ, vùng trũng, hoặc vùng ven. Trước thực trạng đó, việc xây dựng một hệ thống cảnh báo ngập nước đơn giản, chi phí thấp và có thể giám sát từ xa là rất cần thiết.

1.2. Mục tiêu của đề tài

- Thiết kế và chế tạo mô hình hệ thống cảnh báo ngập: sử dụng cảm biến mực nước để đo độ cao mực nước thực tế, tích hợp với bộ xử lý ESP32 có khả năng kết nối Wi-fi và xử lý tín hiệu nhanh chóng.
- Xây dựng phần mềm điều khiển và giám sát: lập trình ESP32 để đọc dữ liệu từ cảm biến, xử lý ngưỡng cảnh báo và gửi thông báo đến người dùng thông qua giao diện web, email hoặc ứng dụng nhắn tin (ví dụ: Telegram, Zalo, v.v...).
- Đánh giá khả năng hoạt động thực tế: thử nghiệm mô hình trong các điều kiện giả lập và môi trường thực tế để kiểm tra độ chính xác, độ nhạy và tính ổn định của hệ thống.

- Đề xuất khả năng mở rộng và ứng dụng: hệ thống có thể triển khai tại các điểm đèn thường xuyên ngập, tích hợp vào các hệ thống cảnh báo thông minh của thành phố hoặc kết nối với hệ thống Iot để mở rộng phạm vi giám sát.

Chương 2: Nội dung và phương pháp thực hiện

2.1. Các nội dung thực hiện

- Tìm hiểu cơ sở lý thuyết và công nghệ sử dụng

Trước khi tiến hành thiết kế và lập trình hệ thống, việc tìm hiểu và nắm vững cơ sở lý thuyết là điều kiện tiên quyết nhằm đảm bảo tính hiệu quả và ứng dụng thực tiễn của đề tài. Trong phần này, nhóm tập trung nghiên cứu các nội dung sau:

- Vi điều khiển ESP32: Là một vi điều khiển tích hợp Wi-Fi và Bluetooth, có khả năng xử lý mạnh, được sử dụng rộng rãi trong các ứng dụng IoT. Việc sử dụng ESP32 giúp tiết kiệm chi phí và cho phép hệ thống cảnh báo từ xa qua mạng Internet.
- Cảm biến mực nước (HC-SR04): Đây là cảm biến siêu âm có khả năng đo khoảng cách dựa trên thời gian phản hồi của sóng siêu âm. Trong đề tài, cảm biến được dùng để đo độ cao mực nước nhằm phát hiện nguy cơ ngập.
- Giao tiếp với Telegram API: Nhằm gửi thông báo trực tiếp đến người dùng qua ứng dụng Telegram khi mực nước vượt ngưỡng cảnh báo. Đây là phương pháp cảnh báo hiệu quả, miễn phí, và có tính thời gian thực.
- Ngoài ra, các linh kiện phụ trợ như LCD I2C, đèn LED, còi buzzer cũng được tìm hiểu nhằm hỗ trợ hiển thị thông tin và phát cảnh báo tại chỗ.
- Thiết kế phần cứng của hệ thống

Dựa trên cơ sở lý thuyết, hệ thống phần cứng được thiết kế với các thành phần chính:

- ESP32 làm trung tâm xử lý dữ liệu và điều khiển.
- Cảm biến siêu âm HC-SR04 để đo mực nước.
- LCD I2C 16x2 để hiển thị giá trị mực nước và trạng thái hệ thống.
- Đèn LED dùng để cảnh báo trực quan khi nước vượt ngưỡng.
- Còi buzzer để cảnh báo âm thanh trong trường hợp nguy hiểm.
- Nguồn cấp 5V cho toàn bộ hệ thống.

Sơ đồ nguyên lý và sơ đồ lắp ráp được thiết kế mô phỏng trên nền tảng Wokwi, giúp kiểm tra tính đúng đắn trước khi triển khai thực tế.

- Xây dựng và lập trình phần mềm điều khiển

Phần mềm được viết bằng ngôn ngữ lập trình Arduino C++ và nạp lên vi điều khiển ESP32. Các chức năng chính bao gồm:

- Đọc giá trị khoảng cách từ cảm biến HC-SR04, từ đó tính toán mức nước hiện tại.
- So sánh mực nước với ngưỡng nguy hiểm đã định sẵn, từ đó kích hoạt cảnh báo nếu vượt ngưỡng.
- Hiển thị thông tin mực nước trên màn hình LCD, giúp người dùng theo dõi trực tiếp tại chỗ.
- Điều khiển LED và còi buzzer để cảnh báo ngay tại khu vực đặt hệ thống.

- Gửi tin nhắn cảnh báo qua Telegram đến người dùng từ xa nếu phát hiện nguy cơ ngập nước.

Toàn bộ chương trình được viết sao cho đảm bảo tính ổn định, dễ hiểu và dễ bảo trì, đồng thời có thể mở rộng trong tương lai.

- Tích hợp và kiểm thử hệ thống

Sau khi hoàn tất phần cứng và phần mềm, hệ thống được lắp ráp và tiến hành kiểm thử qua các bước:

- Kiểm tra đo mực nước tại các mức độ khác nhau để xác định độ chính xác và độ nhạy của cảm biến.
- Kiểm tra khả năng phản ứng của hệ thống cảnh báo (LED và buzzer) khi nước vượt ngưỡng.
- Đánh giá khả năng kết nối mạng và gửi thông báo qua Telegram trong điều kiện mạng thực tế.
- Kiểm tra hệ thống trong thời gian dài để đảm bảo độ ổn định và độ tin cậy.

Thông qua quá trình kiểm thử, nhóm tiến hành hiệu chỉnh các thông số và ngưỡng cảnh báo để đảm bảo hệ thống hoạt động hiệu quả trong điều kiện thực tế.

- Đánh giá kết quả và đề xuất cải tiến

Kết quả bước đầu cho thấy hệ thống có thể hoạt động ổn định, cảnh báo đúng thời điểm và gửi thông tin đến người dùng một cách chính xác. Đây là một giải

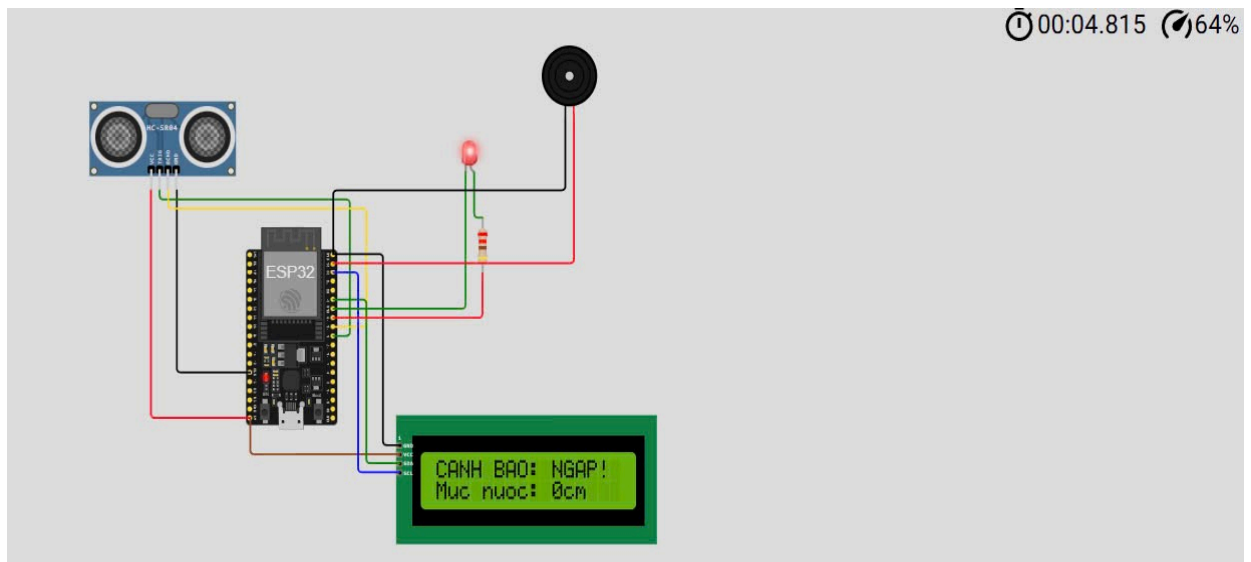
pháp khả thi và chi phí thấp cho các hộ dân sinh sống tại khu vực thường xuyên bị ngập.

Tuy nhiên, nhóm cũng nhận ra một số điểm cần cải thiện:

- Tăng khả năng chống nhiễu của cảm biến để đo chính xác hơn.
- Bổ sung chức năng ghi dữ liệu lên nền tảng đám mây như Firebase hoặc Google Sheets.
- Tích hợp thêm nhiều cảm biến tại các vị trí khác nhau để theo dõi toàn diện hơn.
- Thiết kế vỏ hộp bảo vệ chống nước, đảm bảo hệ thống hoạt động an toàn ngoài trời.

2.2. Thiết kế hệ thống IoT

2.2.1. Sơ đồ logic



2.2.2. Giải thích chức năng linh kiện

- ESP32 – Vi điều khiển trung tâm

ESP32 là một vi điều khiển 32-bit do Espressif Systems phát triển, nổi bật với khả năng tích hợp Wi-Fi và Bluetooth, cho phép xây dựng các ứng dụng IoT mạnh mẽ mà không cần module mạng rời. Trong hệ thống cảnh báo ngập nước, ESP32 đóng vai trò như “bộ não” trung tâm, thực hiện các nhiệm vụ sau:

- Thu thập dữ liệu từ cảm biến đo mực nước.
- Xử lý thông tin đo được và đánh giá điều kiện ngập lụt dựa trên ngưỡng cảnh báo.
- Điều khiển các thiết bị cảnh báo như đèn LED và còi buzzer.
- Hiển thị thông tin lên màn hình LCD để người dùng dễ dàng theo dõi.
- Gửi tin nhắn cảnh báo từ xa qua ứng dụng Telegram (nếu có kết nối Internet).

Nhờ tính linh hoạt và hiệu suất cao, ESP32 là lựa chọn phù hợp cho các hệ thống nhúng hiện đại, đặc biệt trong các ứng dụng giám sát môi trường theo thời gian thực.

- Cảm biến siêu âm HC-SR04 – Thiết bị đo mực nước

Cảm biến siêu âm HC-SR04 là loại cảm biến phổ biến dùng để đo khoảng cách không tiếp xúc, hoạt động dựa trên nguyên lý phản hồi của sóng siêu âm. Cảm biến bao gồm hai phần chính:

- Trig (Trigger): Gửi tín hiệu siêu âm với tần số 40kHz.
- Echo: Nhận sóng phản xạ từ vật cản (mặt nước).

Dựa vào thời gian từ khi phát đến khi nhận sóng phản hồi, khoảng cách được tính toán bằng công thức:

$$\text{Khoảng cách} = \frac{\text{Thời gian phản hồi} \times \text{Vận tốc âm thanh}}{2}$$

Khi đặt cảm biến hướng xuống mặt nước, khoảng cách từ cảm biến đến mặt nước có thể được chuyển đổi thành độ cao mực nước. Việc đo không tiếp xúc giúp tăng độ bền và hạn chế hư hỏng khi đặt trong môi trường ẩm ướt.

- Màn hình LCD 16x2 I2C – Giao diện hiển thị

Để người dùng dễ dàng theo dõi trạng thái mực nước và thông tin cảnh báo, đề tài sử dụng LCD 16x2 kết nối giao tiếp I2C. Màn hình này có khả năng hiển thị 2 dòng, mỗi dòng 16 ký tự, phù hợp để hiển thị thông tin dạng văn bản như:

- “CẢNH BÁO: NGẬP!”
- “Mực nước: 12cm”

Giao tiếp I2C chỉ sử dụng 2 dây (SDA và SCL), giúp tiết kiệm chân kết nối trên ESP32 và đơn giản hóa sơ đồ mạch.

- LED đỏ – Thiết bị cảnh báo trực quan

Đèn LED màu đỏ được sử dụng như một chỉ báo cảnh báo tại chỗ. Khi hệ thống phát hiện mực nước vượt ngưỡng nguy hiểm, ESP32 sẽ cấp tín hiệu để LED sáng lên, giúp người dùng dễ dàng nhận biết tình trạng ngập. Một điện trở hạn dòng (thường $220\Omega - 330\Omega$) được mắc nối tiếp với LED để bảo vệ không bị hư hỏng do dòng điện lớn.

- Còi buzzer – Thiết bị cảnh báo âm thanh

Còi buzzer là một thiết bị chuyển đổi tín hiệu điện thành âm thanh. Trong hệ thống này, buzzer được kích hoạt đồng thời với LED đỏ để cảnh báo âm thanh khi có ngập. Việc phát âm thanh giúp cảnh báo hiệu quả hơn trong trường hợp người dùng không trực tiếp quan sát màn hình hoặc LED.

- Nguồn điện

Toàn bộ hệ thống sử dụng nguồn 5V DC, có thể cấp thông qua cổng USB của ESP32 hoặc từ bộ nguồn rời. Đảm bảo nguồn cấp ổn định là điều kiện quan trọng để các thiết bị hoạt động chính xác.

2.2.3. Nguyên lý hoạt động

- Cảm biến siêu âm HC-SR04

Chức năng: Đo khoảng cách từ cảm biến đến mặt nước để xác định mực nước.

Phương pháp chuẩn cảm biến:

a) Chuẩn đơn giản

Trong trường hợp đại lượng đo chỉ có một đại lượng vật lý duy nhất tác động lên một đại lượng đo xác định và cảm biến sử dụng không nhạy với tác động của các đại lượng ảnh hưởng, người ta dùng phương pháp chuẩn đơn giản. Thực chất của chuẩn đơn giản là đo các giá trị của đại lượng đầu ra ứng với các giá xác định không đổi của đại lượng đo ở đầu vào. Việc chuẩn được tiến hành theo hai cách:

- Chuẩn trực tiếp: các giá trị khác nhau của đại lượng đo lấy từ các mẫu nước hoặc các phân tử so sánh có giá trị biết trước với độ chính xác cao.
- Chuẩn gián tiếp: kết hợp cảm biến cần chuẩn với một cảm biến so sánh đã có sẵn mức ngập chuẩn, cả hai được đặt trong cùng điều kiện làm việc. Khi tác động lên hai cảm biến với cùng một giá trị của đại lượng đo ta nhận được giá trị tương ứng của cảm biến so sánh và cảm biến cần chuẩn. Lặp lại tương tự với các giá trị khác của đại lượng đo cho phép ta xây dựng được đường cong chuẩn của cảm biến cần chuẩn.

b) Chuẩn nhiều lần

Khi cảm biến có phần tử bị trễ (trễ cơ hoặc trễ từ), giá trị đo được ở đầu ra phụ thuộc không những vào giá trị tức thời của đại lượng cần đo ở đầu cảm biến mà còn phụ thuộc vào giá trị trước đó của đại lượng này. Trong trường hợp như vậy, người ta áp dụng phương pháp chuẩn nhiều lần và tiến hành như sau:

- Đặt lại điểm 0 của cảm biến: đại lượng cần đo và đại lượng đầu ra có giá trị tương ứng với điểm mốc
- Đo giá trị đầu ra theo một loạt giá trị tăng dần đến giá trị cực đại của đại lượng đo ở đầu vào.
- Lặp lại quá trình đo với các giá trị giảm dần từ giá trị cực đại.

Khi chuẩn nhiều lần cho phép xác định mức nước đo chuẩn theo cả hai hướng đo tăng dần và đo giảm dần.

Nguyên lý hoạt động:

Cảm biến HC-SR04 hoạt động theo nguyên lý phát và nhận sóng siêu âm. Cụ thể, khi nhận được tín hiệu điều khiển từ ESP32, chân Trig sẽ phát ra một xung điện ngắn (10 micro giây) để kích hoạt cảm biến gửi sóng siêu âm có tần số 40kHz. Sóng này sẽ lan truyền trong không khí và phản xạ lại khi gặp vật cản (trong trường hợp này là mặt nước). Sóng phản xạ được nhận về bởi chân Echo của cảm biến.

Thời gian từ lúc phát đến lúc nhận lại sóng được đo và xử lý để tính khoảng cách theo công thức:

$$\text{Khoảng cách (cm)} = \frac{\text{Thời gian } (\mu\text{s}) \times 0.0343}{2}$$

Giá trị khoảng cách sẽ được dùng để suy ra chiều cao mực nước. Nếu mực nước vượt qua một ngưỡng nhất định (ví dụ 5cm, 10cm...), hệ thống sẽ chuyển sang trạng thái cảnh báo ngập.

- Vi điều khiển ESP32

Chức năng: Điều khiển toàn bộ hoạt động của hệ thống.

Nguyên lý hoạt động:

ESP32 là vi điều khiển tích hợp Wi-Fi, có khả năng xử lý nhanh và giao tiếp với nhiều thiết bị ngoại vi. Trong hệ thống cảnh báo ngập, chương trình được nạp vào ESP32 sẽ:

- Gửi tín hiệu điều khiển đến cảm biến HC-SR04 và đọc dữ liệu đo được.
- So sánh mực nước thực tế với giá trị ngưỡng đã lập trình sẵn.
- Điều khiển bật/tắt đèn LED, còi buzzer, và hiển thị cảnh báo lên LCD.
- Gửi thông báo từ xa đến người dùng qua Telegram Bot nếu kết nối Internet.

ESP32 cũng xử lý tín hiệu thông qua giao tiếp I2C để truyền dữ liệu đến màn hình LCD.

- Màn hình LCD 16x2 I2C

Chức năng: Hiển thị dữ liệu mực nước và cảnh báo.

Nguyên lý hoạt động:

Màn hình LCD 16x2 có khả năng hiển thị tối đa 32 ký tự (chia làm 2 dòng, mỗi dòng 16 ký tự). Nhờ tích hợp mạch I2C, việc giao tiếp với ESP32 chỉ cần 2 dây tín hiệu:

- SDA (data): Truyền dữ liệu.
- SCL (clock): Tạo xung đồng hồ.

ESP32 sử dụng thư viện Wire và LiquidCrystal_I2C để điều khiển màn hình, hiển thị các thông tin như:

- “Mức nước: 8 cm”
- “Cảnh báo: NGẬP!”

Người dùng có thể quan sát mực nước thực tế mà không cần thiết bị hỗ trợ khác.

- LED đỏ

Chức năng: Cảnh báo trực quan bằng ánh sáng.

Nguyên lý hoạt động:

LED là một diode phát quang, chỉ sáng khi được phân cực thuận. Khi mực nước vượt ngưỡng, chân digital output của ESP32 sẽ được lập trình để đưa mức HIGH (3.3V) đến chân dương của LED. Một điện trở hạn dòng (thường là 220–330Ω) được mắc nối tiếp để bảo vệ LED khỏi hư hỏng.

LED sáng giúp cảnh báo ngay lập tức cho người ở gần khu vực có thiết bị.

- Buzzer

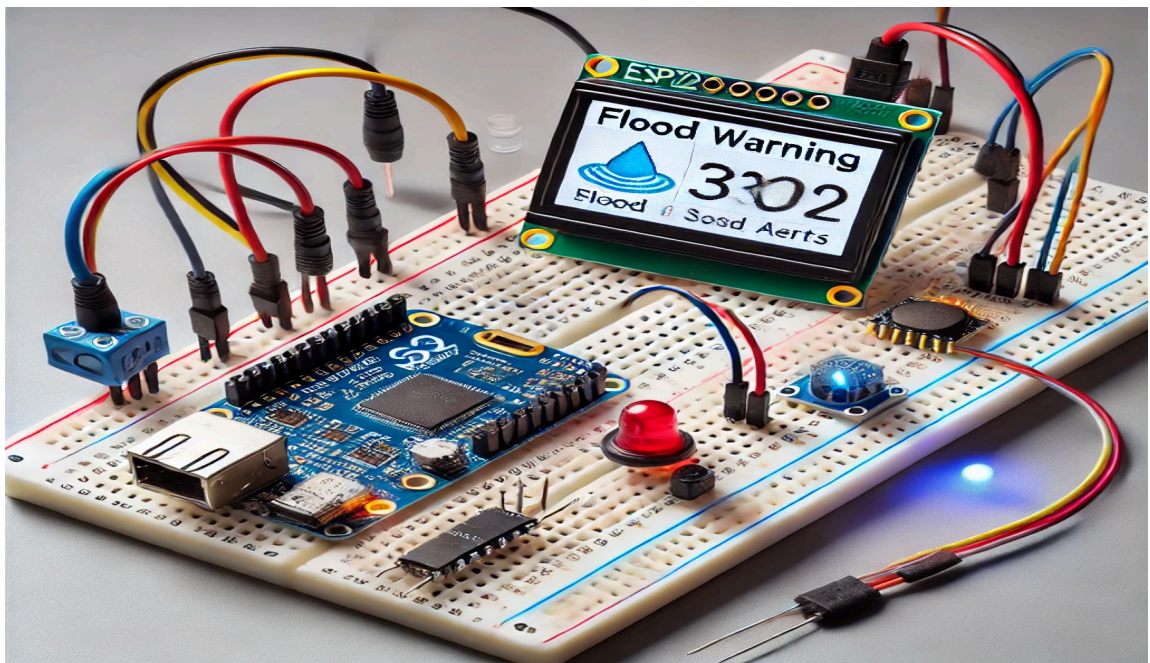
Chức năng: Cảnh báo bằng âm thanh.

Nguyên lý hoạt động:

Buzzer là thiết bị chuyển đổi tín hiệu điện thành âm thanh. Có hai loại buzzer: chủ động và thụ động. Trong hệ thống này, thường sử dụng buzzer chủ động – chỉ cần cấp nguồn điện (thường 5V hoặc 3.3V) là sẽ phát ra âm thanh.

ESP32 sẽ xuất mức HIGH hoặc xung PWM đến buzzer khi phát hiện mực nước cao, giúp người dùng nhận biết ngay cả khi không quan sát LED hoặc màn hình LCD.

2.2.4. Mạch vật lý



Chương 3: Cài đặt hệ thống

3.1. Công cụ và thiết bị sử dụng

- Thiết bị phần cứng

Hệ thống được xây dựng từ các linh kiện điện tử phổ biến và dễ tiếp cận, cụ thể như sau:

- ESP32 DevKit: Là bộ vi điều khiển lõi kép, hỗ trợ kết nối WiFi và Bluetooth, có khả năng xử lý mạnh mẽ, là trung tâm điều khiển toàn bộ hệ thống.
- Cảm biến siêu âm HC-SR04: Có chức năng đo khoảng cách từ cảm biến đến mặt nước, từ đó tính toán ra độ cao của mực nước.
- Màn hình LCD 16x2 giao tiếp I2C: Dùng để hiển thị thông tin mực nước hiện tại và cảnh báo nếu vượt ngưỡng.
- LED đỏ kèm điện trở 330Ω: Cảnh báo trực quan khi phát hiện mực nước vượt ngưỡng cho phép.
- Còi Buzzer chủ động: Phát âm thanh báo động khi có ngập.
- Breadboard và dây nối: Hỗ trợ kết nối các linh kiện lại với nhau dễ dàng và không cần hàn.
- Công cụ phần mềm

Để lập trình và mô phỏng hệ thống, nhóm sử dụng các phần mềm sau:

- Arduino IDE: Là môi trường phát triển tích hợp (IDE) được dùng để viết, biên dịch và nạp chương trình vào vi điều khiển ESP32.
- Thư viện Arduino: Bao gồm các thư viện như [Wire.h](#), [LiquidCrystal_I2C.h](#), [WiFi.h](#), [HTTPClient.h](#) để giao tiếp với LCD, cảm biến và kết nối Internet.

- Wokwi : Dùng để mô phỏng mạch trước khi triển khai thực tế, giúp kiểm tra tính đúng đắn của mạch và chương trình.
- Telegram: Ứng dụng nhắn tin hỗ trợ tạo Bot và API để nhận cảnh báo từ hệ thống.

3.2. Quy trình cài đặt hệ thống

Bước 1: Kết nối phần cứng

- ESP32 được gắn lên breadboard làm trung tâm điều khiển.
- Cảm biến siêu âm HC-SR04 được kết nối với các chân số của ESP32 (ví dụ: Trig – D5, Echo – D18).
- Màn hình LCD kết nối qua giao tiếp I2C đến chân SDA và SCL trên ESP32.
- Đèn LED và điện trở nối với một chân digital (ví dụ: D2), đồng thời nối mass.
- Còi buzzer được nối với một chân digital (ví dụ: D4), nối mass và cấp nguồn.
- Kiểm tra lại tất cả các kết nối để đảm bảo chắc chắn.

Bước 2: Lập trình hệ thống

- Cài đặt Arduino IDE, thêm board ESP32 thông qua URL trong phần Preferences.
- Cài đặt thư viện hỗ trợ LCD, cảm biến và kết nối WiFi.

- Viết chương trình có các chức năng sau:
- Đọc khoảng cách từ cảm biến → chuyển đổi thành mực nước.
- So sánh với mực nước giới hạn để xác định trạng thái cảnh báo.
- Hiển thị mực nước và cảnh báo trên LCD.
- Khi ngập, bật LED, còi và gửi cảnh báo đến Telegram thông qua API Bot.

Bước 3: Kết nối Telegram

- Truy cập Telegram và sử dụng BotFather để tạo Bot mới, lấy bot token.
- Tìm chat_id bằng cách nhắn tin cho Bot và dùng một API để lấy thông tin.
- Trong code Arduino, chèn bot token và chat_id vào lệnh HTTP để gửi cảnh báo.

Bước 4: Nạp và chạy chương trình

- Kết nối ESP32 với máy tính qua cổng USB.
- Chọn board ESP32 và cổng COM tương ứng trong Arduino IDE.
- Nạp chương trình vào vi điều khiển.
- Mở Serial Monitor để theo dõi giá trị đo và trạng thái.
- Kiểm tra hoạt động của LCD, LED, còi buzzer và tin nhắn Telegram.

Chương 4: Kết quả thực hiện

4.1. Hình ảnh và giao diện



4.2. Phân tích kết quả

Hình ảnh minh họa là giao diện ứng dụng Telegram trên một chiếc điện thoại thông minh, thể hiện rõ ràng tin nhắn cảnh báo ngập nước đã được gửi thành công từ hệ thống.

Nội dung chính trong hình ảnh:

- Tiêu đề tin nhắn: “⚠ Cảnh báo ngập!” — thể hiện rõ ràng tính chất khẩn cấp của tình trạng mực nước đang cao bất thường.
- Số liệu đo: “8 cm” — là mực nước hiện tại mà cảm biến ghi nhận được, báo hiệu rằng nước đã chạm ngưỡng nguy hiểm (có thể do thiết lập hệ thống).

Thông qua hình ảnh, có thể khẳng định rằng hệ thống đã:

- Đo chính xác mực nước bằng cảm biến.
- Phân tích dữ liệu và xác định trạng thái nguy hiểm.
- Gửi cảnh báo thời gian thực đến người dùng thông qua nền tảng Telegram.

Chương 5: Kết luận và hướng phát triển

5.1. Các công việc đã thực hiện được

Trong quá trình thực hiện đề tài “Phát triển hệ thống cảnh báo ngập nước với ESP32”, tôi đã tiến hành nghiên cứu, thiết kế và xây dựng một hệ thống cảnh báo mực nước hiệu quả, kết hợp giữa phần cứng điện tử và công nghệ IoT. Cụ thể, các công việc đã được hoàn thành bao gồm:

- Tìm hiểu lý thuyết và cơ sở khoa học của đề tài:
Nhóm đã nghiên cứu các giải pháp cảnh báo ngập hiện nay, đồng thời xác định tính cấp thiết và tính ứng dụng thực tiễn của hệ thống cảnh báo ngập nước.
- Lựa chọn và phân tích linh kiện phần cứng:
Bao gồm ESP32 DevKit, cảm biến siêu âm HC-SR04, màn hình LCD I2C, LED cảnh báo, còi buzzer và các thiết bị phụ trợ như breadboard, dây nối. Mỗi linh

kiện được phân tích rõ về chức năng, sơ đồ chân và cách kết nối.

- Thiết kế sơ đồ mạch nguyên lý và mô phỏng hệ thống trên Wokwi:

Việc sử dụng nền tảng mô phỏng giúp kiểm tra và điều chỉnh sơ đồ trước khi lắp ráp thực tế, giảm thiểu rủi ro và tiết kiệm thời gian.

- Lập trình điều khiển hệ thống trên Arduino IDE:

Bao gồm viết mã điều khiển cảm biến đo mực nước, xử lý ngưỡng cảnh báo, điều khiển hiển thị LCD, điều khiển LED và buzzer, và gửi cảnh báo từ xa qua Telegram. Chương trình được tối ưu để hoạt động ổn định, phản hồi nhanh.

- Tạo và cấu hình Telegram Bot:

Đã tạo thành công một bot Telegram để hệ thống gửi cảnh báo đến người dùng khi phát hiện tình trạng mực nước vượt ngưỡng. Người dùng có thể nhận thông báo cảnh báo từ xa, góp phần nâng cao tính tiện lợi và an toàn.

- Thử nghiệm và hiệu chỉnh hệ thống trong thực tế:

Mạch được lắp ráp hoàn chỉnh trên breadboard, chạy ổn định, các chức năng hoạt động như mong đợi. Đặc biệt, tính năng gửi cảnh báo qua Telegram đã hoạt động chính xác, góp phần tăng tính thực tiễn của hệ thống.

5.2. Hạn chế

Mặc dù hệ thống cảnh báo ngập nước sử dụng ESP32 đã đạt được những mục tiêu đề ra, tuy nhiên trong quá trình thiết kế và triển khai, đề tài vẫn còn tồn tại một số hạn chế nhất định, cụ thể như sau:

1. Độ chính xác phụ thuộc vào môi trường hoạt động

Hệ thống sử dụng cảm biến siêu âm HC-SR04 để đo mực nước, tuy nhiên cảm biến này hoạt động dựa trên sóng âm nên có thể bị ảnh hưởng bởi:

- Nhiệt độ môi trường,
- Hơi nước hoặc bụi trong không khí,
- Bề mặt nước gợn sóng hoặc có vật cản.

Những yếu tố trên có thể gây sai lệch trong phép đo, ảnh hưởng đến độ tin cậy của hệ thống.

2. Phạm vi đo hạn chế

Cảm biến HC-SR04 có phạm vi đo tối đa khoảng 4 mét. Do đó, hệ thống chỉ phù hợp với các ứng dụng cảnh báo ngập cục bộ như hầm xe, sàn nhà, rãnh thoát nước,... chứ chưa đủ khả năng ứng dụng trong môi trường rộng như hệ thống cảnh báo lũ ở quy mô lớn.

3. Giao diện người dùng còn đơn giản

Việc hiển thị thông tin trên màn hình LCD 16x2 khá hạn chế, chỉ thể hiện được một số dòng thông tin cơ bản. Bên cạnh đó, giao diện Telegram tuy hiệu quả nhưng chưa hỗ trợ phản hồi tương tác từ người dùng, như kiểm tra lại trạng thái hệ thống hoặc điều khiển từ xa.

4. Hệ thống chưa có nguồn cấp điện dự phòng

Hiện tại, hệ thống hoạt động chủ yếu dựa vào nguồn điện USB hoặc adapter. Trong trường hợp mất điện đột ngột (thường xảy ra khi có mưa bão, ngập lụt), toàn bộ hệ thống sẽ ngừng hoạt động. Điều này làm giảm độ tin cậy và khả năng phản ứng trong các tình huống khẩn cấp.

5. Chưa có cơ chế lưu trữ dữ liệu lịch sử

Hệ thống hiện tại chỉ hoạt động theo thời gian thực mà chưa lưu trữ các dữ liệu đo được như mực nước theo thời gian, số lần cảnh báo,... Điều này gây khó khăn trong việc phân tích, đánh giá xu hướng ngập nước trong khu vực theo thời gian dài.

6. Bảo mật và kết nối Internet

Việc gửi dữ liệu cảnh báo qua mạng Wi-Fi có thể gặp rủi ro về mất kết nối hoặc bảo mật thông tin nếu không cấu hình đúng. Ngoài ra, nếu mất Internet, hệ thống sẽ không thể gửi cảnh báo qua Telegram.

Trong một thế giới mở, IOT sẽ mang tính chất phức tạp bởi nó bao gồm một lượng lớn đường liên kết giữa các thiết bị, máy móc, dịch vụ với nhau, ngoài ra còn bởi khả năng thêm vào các nhân tố mới.

5.3. Hướng phát triển

Mặc dù hệ thống cảnh báo ngập nước hiện tại đã cơ bản đáp ứng được yêu cầu đo mực nước và gửi cảnh báo thời gian thực, tuy nhiên để tăng tính ổn định, mở rộng phạm vi ứng dụng và nâng cao hiệu quả sử dụng trong thực tế, đề tài có thể được phát triển theo một số hướng sau:

1. Tích hợp nhiều cảm biến để tăng độ chính xác

Thay vì chỉ sử dụng cảm biến siêu âm, hệ thống có thể kết hợp thêm các loại cảm biến khác như:

- Cảm biến mực nước áp suất (Water Level Pressure Sensor) để đo mực nước một cách ổn định, ít bị nhiễu.
- Cảm biến mưa, để đánh giá khả năng xảy ra ngập từ trước.

- Cảm biến nhiệt độ và độ ẩm để phân tích ảnh hưởng của môi trường đến hệ thống.

Sự kết hợp nhiều loại cảm biến sẽ giúp hệ thống đưa ra cảnh báo chính xác hơn, đa chiều hơn, và linh hoạt hơn.

2. Phát triển giao diện người dùng (UI) nâng cao

Thay vì chỉ gửi cảnh báo qua Telegram, hệ thống có thể được nâng cấp với:

- Web dashboard hiển thị thông tin mực nước theo thời gian thực, biểu đồ lịch sử dữ liệu, tình trạng hệ thống.
- Ứng dụng di động cho phép người dùng kiểm tra, nhận cảnh báo, điều khiển từ xa và phản hồi ngay lập tức.
- Phản hồi tương tác với bot Telegram như: yêu cầu trạng thái hiện tại, bật/tắt báo động từ xa,...

3. Bổ sung chức năng lưu trữ dữ liệu và phân tích

Hệ thống có thể tích hợp:

- Thẻ nhớ microSD hoặc cơ sở dữ liệu đám mây (Firebase, Thingspeak...) để lưu trữ lịch sử mực nước.
- Từ đó, người dùng có thể theo dõi mức độ thường xuyên và xu hướng ngập theo từng khu vực, từng thời điểm.

Các dữ liệu này có thể hỗ trợ cho công tác quy hoạch hạ tầng thoát nước đô thị, cảnh báo sớm, và ứng dụng vào các hệ thống thành phố thông minh (Smart City).

4. Nâng cao độ tin cậy và khả năng hoạt động độc lập

Hệ thống có thể được cải tiến với:

- Nguồn pin dự phòng hoặc tích hợp năng lượng mặt trời, đảm bảo hoạt động ngay cả khi mất điện.
 - Kết nối GSM (SIM800/900) để gửi SMS cảnh báo trong trường hợp mất Wi-Fi.
 - Tự khởi động lại khi mất kết nối, đảm bảo không bị treo hệ thống.
5. Mở rộng phạm vi triển khai thực tế

Hệ thống có thể được nhân rộng và áp dụng cho:

- Hầm để xe, nhà máy, tầng hầm tòa nhà,...
- Kênh rạch, điểm giao thông dễ ngập trong đô thị.
- Vùng dân cư trũng thấp, ven sông hoặc thường xuyên xảy ra lũ.

Khi kết nối các hệ thống này thành mạng lưới cảm biến, có thể tạo thành một hệ thống cảnh báo ngập quy mô lớn, phục vụ công tác quản lý đô thị và phòng chống thiên tai.

Tài liệu tham khảo

Tiếng Việt:

[1]ThS. Lê Văn Doanh. Kỹ thuật cảm biến. đoạn trích ở trang 13 của tài liệu số 1: [1, 13].

<https://fr.slideshare.net/slideshow/gio-trnh-k-thut-cm-binpdf/257494265>

[2]ThS. Nguyễn Hữu Tuấn. Internet vạn vật (IoT) – Ứng dụng và triển khai thực tế.

[3]TS. Nguyễn Trọng Hòa. Điện tử cơ bản và thực hành.

[4] TS. Nguyễn Hoàng Hải. Arduino – Những dự án và ứng dụng thực tiễn.

Tiếng Anh:

[5] Gómez, J., Huete, J. F., Hoyos, O., Perez, L., & Grigori, D. (2013). Interaction system based on internet of things as support for education. Procedia Computer Science, [5,132-139]