

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

\*\*\*\*\*



**ĐỀ TÀI**

**Hệ Thống Phát Hiện Rò Rỉ Khí Gas với ESP32**

**TÊN LỚP HỌC PHẦN – MÃ HỌC PHẦN**

**PHÁT TRIỂN ỨNG DỤNG IOT – NHÓM 4 – 2024-2025.2.TIN4024.004**

**Giáo viên hướng dẫn : ThS.Võ Việt Dũng**

**Sinh viên thực hiện :**

**Mã sinh viên :**

Huế, Tháng 04 Năm 2025

## DANH MỤC TỪ VIẾT TẮT

- ESP32: Vi điều khiển có Wi-Fi và Bluetooth
- IoT: Internet of Things (Mạng lưới vạn vật)
- ppm: Phần triệu (parts per million)
- ADC: Analog to Digital Converter (Bộ chuyển từ tín hiệu tương tự sang số)
- GPIO: General Purpose Input/Output (Chân vào ra tổng hợp)
- LCD: Liquid Crystal Display
- API: Application Programming Interface
- UPS: Uninterruptible Power Supply (Nguồn dự phòng)
- MQTT: Message Queuing Telemetry Transport

## MỤC LỤC

I. Giới thiệu:	2
II. Tổng quan về phần cứng:	2
1. ESP32:	2
2. Cảm biến MQ-6	3
3. Còi báo động	3
4. Relay (tùy chọn)	3
5. Linh kiện mở rộng	3
III. Nguyên lý hoạt động:	4
1. Hệ thống hoạt động theo nguyên lý sau:	4
2. Sơ đồ khối hệ thống:	4
IV. Mô phỏng trên Wokwi:	4
1. Các linh kiện sử dụng	5
2. Kết nối	5
V. Truyền thông tin qua IoT:	5
1. So sánh các nền tảng IoT:	5
2. ThingSpeak ( <a href="https://thingspeak.com">https://thingspeak.com</a> )	6
3. Blynk ( <a href="https://blynk.io">https://blynk.io</a> )	6
4. Telegram ( <a href="https://core.telegram.org/bots">https://core.telegram.org/bots</a> )	7
VI. Sơ đồ hệ thống:	7
VII. Mô hình hoạt động và kịch bản:	8
VIII. Kết luận, đánh giá, nhận xét và kiến nghị:	9
1. Kết luận:	9
2. Đánh giá:	9
3. Nhận xét:	10
4. Kiến nghị	10
5. Mở rộng hệ thống	10
Tài liệu tham khảo:	11

## Hệ Thống Phát Hiện Rò Rỉ Khí Gas với ESP32

### I. Giới thiệu:

Rò rỉ khí gas là một trong những nguyên nhân chính gây ra hỏa hoạn và ngộ độc trong môi trường gia đình và công nghiệp. Theo thống kê, hàng năm có hàng nghìn vụ cháy nổ do khí gas gây ra, đặc biệt là tại các hộ gia đình và cơ sở sản xuất nhỏ. Việc phát hiện rò rỉ khí gas sớm giúp giảm thiểu nguy cơ cháy nổ và đảm bảo an toàn cho con người.

Trong bối cảnh các thiết bị gia dụng sử dụng khí gas ngày càng phổ biến, nhu cầu phát hiện rò rỉ khí gas kịp thời để ngăn ngừa sự cố cháy nổ là rất quan trọng. Với sự phát triển của công nghệ IoT, việc xây dựng một hệ thống giám sát tự động là hoàn toàn khả thi, đặc biệt khi kết hợp cảm biến khí gas và vi điều khiển có khả năng kết nối mạng như ESP32.

Hiện nay, việc phát hiện rò rỉ khí gas chủ yếu dựa vào cảm quan (ngửi mùi gas) hoặc các thiết bị đo cầm tay. Tuy nhiên, các phương pháp này không hiệu quả trong trường hợp rò rỉ nhỏ và kéo dài. Vì vậy, hệ thống phát hiện rò rỉ khí gas tự động với ESP32, cảm biến MQ-6, còi báo động và gửi thông báo qua IoT sẽ giúp tăng cường độ an toàn và phản ứng nhanh chóng khi có sự cố xảy ra.

### II. Tổng quan về phần cứng:

#### 1. ESP32:



ESP32 là một vi điều khiển phổ biến trong các dự án IoT nhờ có Wi-Fi tích hợp, khả năng xử lý mạnh và tiêu thụ điện năng thấp. Các đặc điểm kỹ thuật:

- CPU: Dual-core Tensilica LX6.
- Tốc độ xử lý: Lên tới 240MHz.
- RAM: 520KB SRAM.
- Tích hợp Wi-Fi chuẩn 802.11 b/g/n và Bluetooth 4.2.
- Hỗ trợ nhiều cổng giao tiếp: ADC, DAC, SPI, I2C, UART, PWM...
- Nhiệt độ hoạt động: -40°C đến +125°C.

ESP32 có thể lập trình bằng Arduino IDE hoặc PlatformIO. Với việc tích hợp Wi-Fi, nó có thể dễ dàng kết nối Internet để gửi dữ liệu lên các nền tảng IoT như ThingSpeak hoặc Blynk.

## 2. Cảm biến MQ-6



Cảm biến MQ-6 là loại cảm biến bán dẫn có khả năng phát hiện khí gas LPG, iso-butane, propane, methane, alcohol và hydrogen. Thông số kỹ thuật:

- Điện áp hoạt động: 5V DC.
- Dòng tiêu thụ: 150mA.
- Thời gian đáp ứng: 10 giây.
- Tín hiệu đầu ra: Analog (có thể chuyển sang digital với bộ so sánh).
- Dải đo: 200 ppm - 10000 ppm.

Cảm biến cần thời gian "làm nóng" (preheat) khoảng 20 giây trước khi hoạt động ổn định. Đầu ra analog được kết nối tới chân ADC của ESP32 để đo mức nồng độ khí.

## 3. Còi báo động

Còi báo động sử dụng điện áp 5V để phát ra âm thanh cảnh báo khi có sự cố. Trong hệ thống này, còi được kích hoạt bằng cách điều khiển một chân digital của ESP32 qua một transistor NPN hoặc relay.

- Loại còi: Active Buzzer (tự phát âm khi cấp điện).
- Dòng tiêu thụ: Khoảng 30mA.
- Kết nối: Cực dương nối với nguồn 5V, cực âm qua transistor về GND.

## 4. Relay (tùy chọn)

Relay được dùng để điều khiển còi hoặc thiết bị ngoại vi có điện áp cao hơn. Có thể thay thế bằng transistor nếu chỉ điều khiển còi 5V.

## 5. Linh kiện mở rộng

- LCD/OLED hiển thị giá trị ppm khí.

- Cảm biến nhiệt độ (DHT11/DHT22) để kiểm tra ảnh hưởng môi trường.
- Cảm biến chuyển động (PIR) để phát hiện người xung quanh.

### III. Nguyên lý hoạt động:

#### 1. Hệ thống hoạt động theo nguyên lý sau:

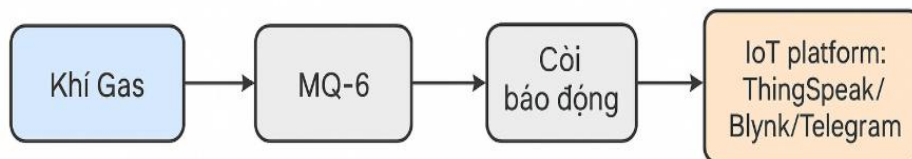
- MQ-6 liên tục đo nồng độ khí trong không khí.
- Khi nồng độ vượt ngưỡng định sẵn (ví dụ 600 ppm), tín hiệu analog tăng.
- ESP32 đọc giá trị này, nếu vượt ngưỡng sẽ thực hiện:
  - Kích hoạt còi báo động.
  - Gửi dữ liệu và cảnh báo lên nền tảng IoT.
  - Hiện thị thông tin trên màn hình (nếu có).

Ngưỡng giá trị có thể thiết lập trong phần mềm tùy theo môi trường sử dụng để đảm bảo phát hiện sớm nhưng không gây báo giả.

**Lưu ý về thiết lập ngưỡng:** Giá trị ngưỡng (ví dụ: 600 ppm) cần được hiệu chỉnh theo từng môi trường cụ thể. Trong phòng kín, nồng độ gas tăng rất nhanh nên cần đặt ngưỡng thấp hơn, trong khi ở nơi thông thoáng, ngưỡng có thể cao hơn để tránh báo giả. Ngưỡng này có thể cài đặt linh hoạt trong phần mềm hoặc qua ứng dụng Blynk nếu muốn điều chỉnh từ xa.

#### 2. Sơ đồ khối hệ thống:

Khí Gas → MQ-6 → ESP32 → (Còi báo động, IoT platform: ThingSpeak/Blynk/Telegram)



### IV. Mô phỏng trên Wokwi:

Wokwi là nền tảng mô phỏng phần cứng trực tuyến cho Arduino và ESP32. Để mô phỏng hệ thống:

**Xem mô phỏng tại:**

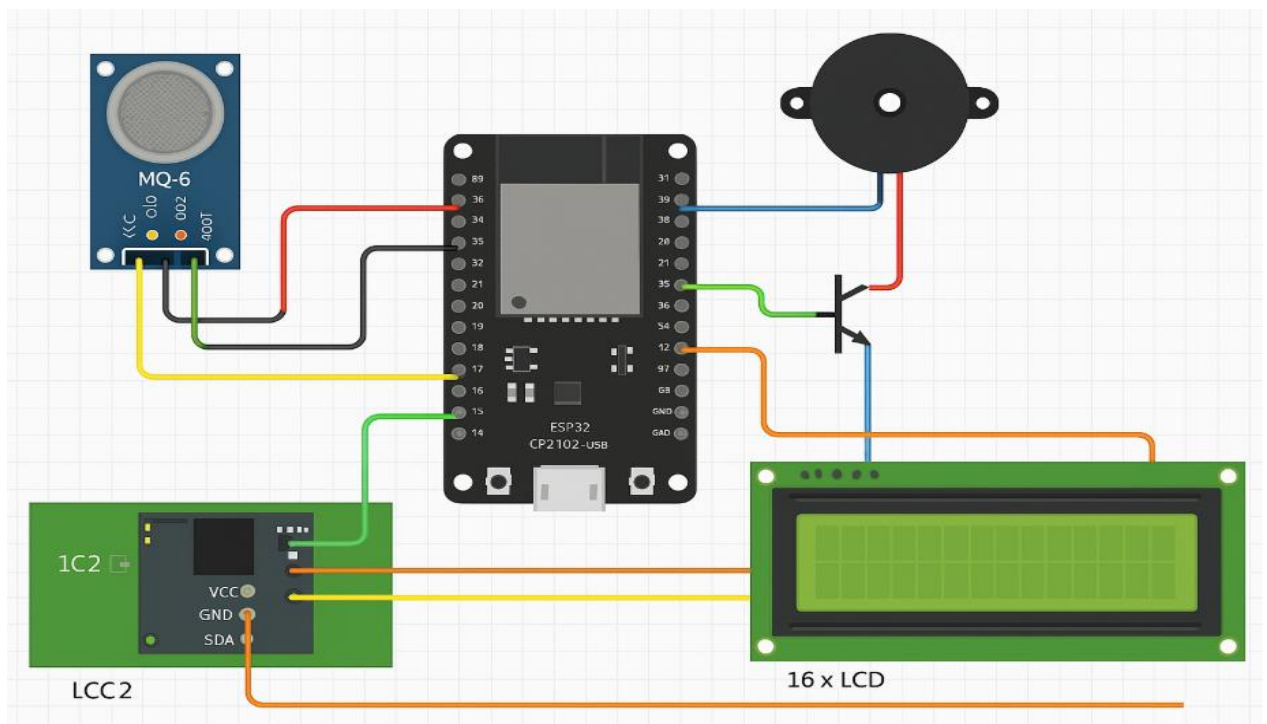
<https://wokwi.com/projects/305569599398609473>

## 1. Các linh kiện sử dụng

- ESP32 Dev Module
- Cảm biến MQ-6
- Còi báo động (Active Buzzer)
- Transistor NPN (ví dụ 2N2222 hoặc S8050)
- Điện trở 220Ω (cho chân base transistor)
- LCD 16x2 I2C (tùy chọn hiển thị)
- Dây nối và breadboard ảo

## 2. Kết nối

- MQ-6 VCC → 5V, GND → GND, AOUT → GPIO36 (ADC) của ESP32.
- Còi: Dây + nối 5V, dây – qua transistor tới GPIO32 của ESP32.
- LCD I2C: SDA → GPIO21, SCL → GPIO22, VCC → 5V, GND → GND.



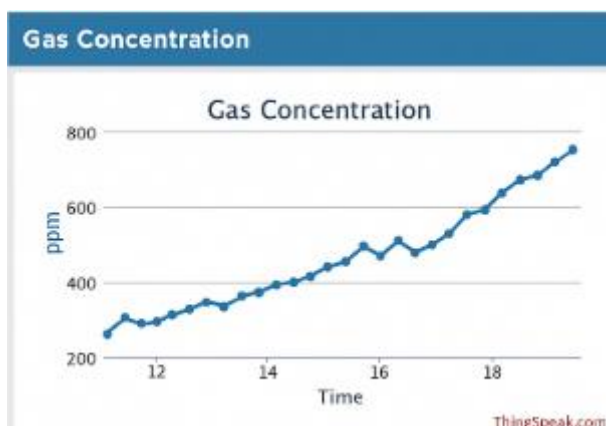
## V. Truyền thông tin qua IoT:

### 1. So sánh các nền tảng IoT:

Tiêu chí	ThingSpeak	Blynk	Telegram
Giao diện	Biểu đồ, đơn giản	Đẹp, thân thiện	Chat, dòng lệnh

Tiêu chí	ThingSpeak	Blynk	Telegram
Cảnh báo	Email, Trigger	Notification	Tin nhắn trực tiếp
Lập trình	Dễ (HTTP GET)	Trung bình (Auth + widget)	Trung bình (API URL)
Yêu cầu app	Không	Có	Có (Telegram)
Khả năng mở rộng	Cao (MATLAB, IoT Hub)	Cao (Nhiều widget)	Trung bình

## 2. ThingSpeak (<https://thingspeak.com>)



ThingSpeak là một nền tảng mã nguồn mở cho phép thu thập, phân tích và hiển thị dữ liệu từ các thiết bị IoT. Nó hỗ trợ nhiều loại giao thức, bao gồm HTTP và MQTT, và có thể sử dụng trực tiếp với MATLAB để phân tích dữ liệu. ThingSpeak cung cấp khả năng gửi thông báo cảnh báo qua email khi giá trị khí gas vượt ngưỡng.

### Ưu điểm:

- Miễn phí, dễ tích hợp với MATLAB.
- Có khả năng chia sẻ dữ liệu công khai.
- Tự động gửi cảnh báo qua email hoặc trigger khi vượt ngưỡng.

### Nhược điểm:

- Giao diện không quá trực quan.
- Có một số giới hạn về số lần cập nhật dữ liệu.

## 3. Blynk (<https://blynk.io>)

Blynk là một nền tảng IoT mạnh mẽ cho phép xây dựng ứng dụng điều khiển thiết bị từ xa qua điện thoại. Blynk cung cấp các widget đa dạng cho phép hiển thị dữ liệu, gửi

thông báo và điều khiển thiết bị. Hệ thống có thể gửi cảnh báo khi khí gas vượt ngưỡng được cài đặt sẵn.

**Ưu điểm:**

- Giao diện dễ sử dụng, trực quan.
- Hỗ trợ nhiều thiết bị và widget.
- Cảnh báo qua push notification, dễ dàng tùy chỉnh.

**Nhược điểm:**

- Cần một chút thời gian làm quen với cách cài đặt Auth Token và cấu hình widget.
- Yêu cầu sử dụng app Blynk trên điện thoại.

**4. Telegram (<https://core.telegram.org/bots>)**

Telegram Bot API cho phép tạo bot tự động gửi tin nhắn cảnh báo về sự cố khí gas trực tiếp trên ứng dụng Telegram. Hệ thống sẽ gửi tin nhắn tự động khi phát hiện rò rỉ khí gas, giúp người dùng phản ứng kịp thời.

**Ưu điểm:**

- Không cần cài thêm ứng dụng, chỉ cần Telegram.
- Hỗ trợ gửi thông điệp, hình ảnh và vị trí.
- Tính bảo mật cao và dễ dàng tích hợp với các hệ thống khác.

**Nhược điểm:**

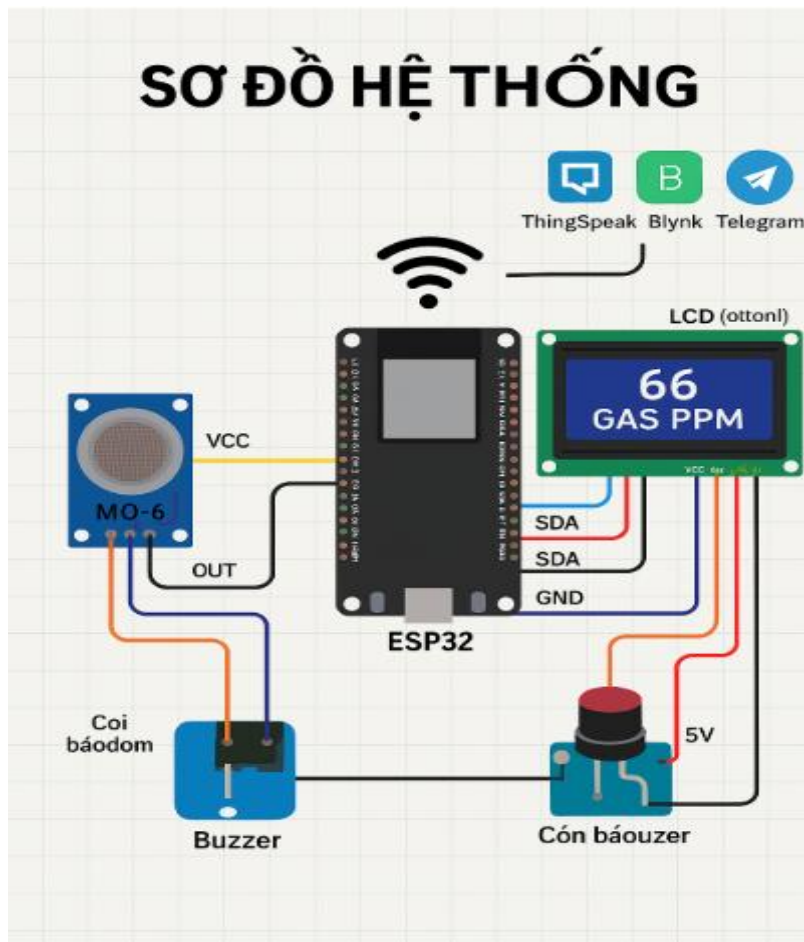
- Giao diện chỉ hiển thị thông tin đơn giản, không có biểu đồ hoặc phân tích.
- Phải cấu hình bot và lấy API Token từ BotFather.

**VI. Sơ đồ hệ thống:**

Sơ đồ hệ thống kết nối các thành phần trong dự án sẽ được mô phỏng như sau:

- **MQ-6** (Cảm biến khí gas) kết nối với **ESP32** qua chân ADC để đo nồng độ khí.
- **Còi báo động** được điều khiển qua transistor, kết nối với chân GPIO của **ESP32**.
- **ESP32** sẽ kết nối Wi-Fi để gửi dữ liệu lên các nền tảng IoT như ThingSpeak, Blynk, hoặc Telegram.
- **LCD (tùy chọn)** có thể được sử dụng để hiển thị giá trị nồng độ khí gas đo được.

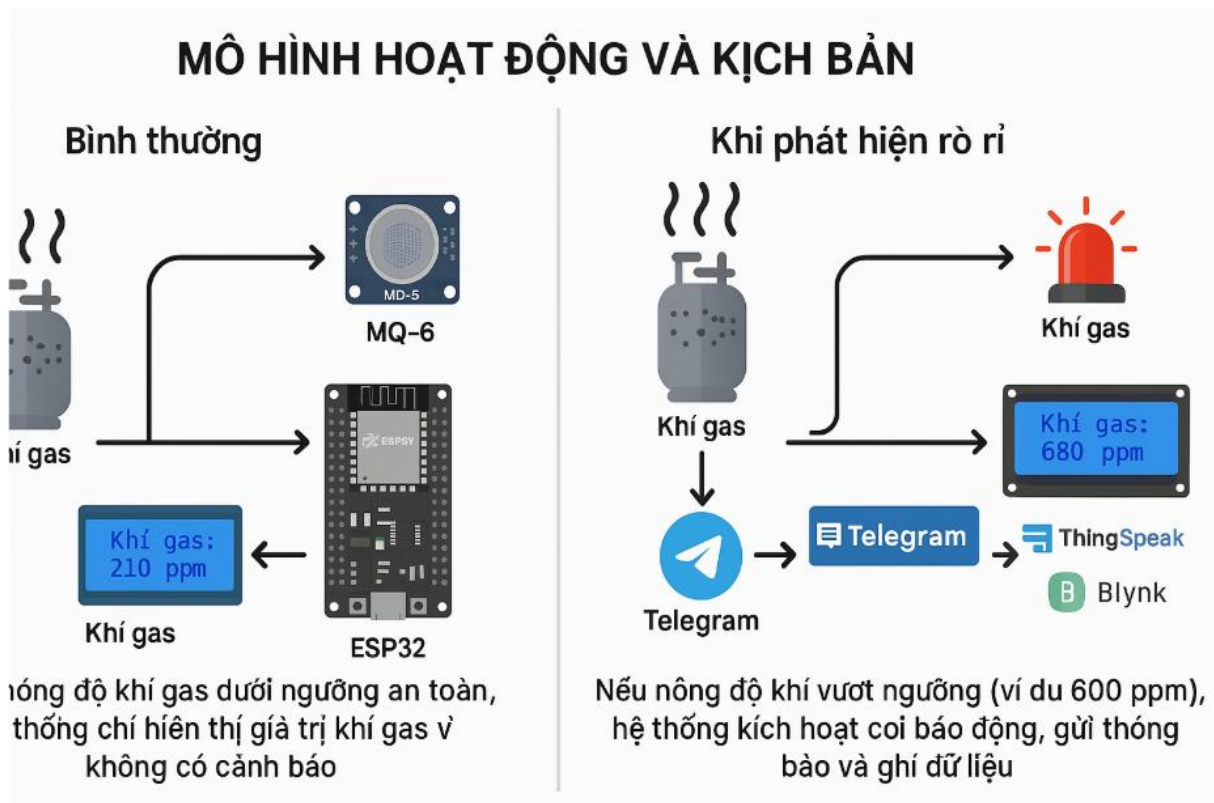




## VII. Mô hình hoạt động và kịch bản:

Hệ thống hoạt động theo các kịch bản sau:

- **Bình thường:** Khi nồng độ khí gas dưới ngưỡng an toàn, hệ thống chỉ hiển thị giá trị khí gas và không có cảnh báo.
- **Khi phát hiện rò rỉ:** Nếu nồng độ khí vượt ngưỡng (ví dụ 600 ppm), hệ thống sẽ kích hoạt còi báo động, gửi thông báo qua Telegram và ghi lại dữ liệu trên ThingSpeak hoặc Blynk.



## VIII. Kết luận, đánh giá, nhận xét và kiến nghị:

### 1. Kết luận:

Hệ thống phát hiện rò rỉ khí gas sử dụng ESP32 là một giải pháp hiệu quả, chi phí thấp, dễ triển khai và phù hợp với các hộ gia đình, nhà trọ, bếp công nghiệp. Việc sử dụng cảm biến MQ-6 giúp phát hiện sớm nguy cơ rò rỉ khí gas, từ đó hạn chế nguy cơ cháy nổ, bảo vệ tài sản và tính mạng con người. Hệ thống có thể hoạt động độc lập hoặc tích hợp với nền tảng IoT để giám sát và cảnh báo từ xa.

### 2. Đánh giá:

- **Tính hiệu quả:** Phát hiện nhanh chóng, cảnh báo kịp thời.
- **Tính linh hoạt:** Có thể điều chỉnh ngưỡng, tích hợp nhiều loại cảm biến khác.
- **Tính mở rộng:** Dễ nâng cấp với các nền tảng IoT hiện đại như Blynk, Telegram, Firebase,...
- **Tính ứng dụng thực tế:** Cao, đặc biệt trong bối cảnh đô thị hóa và nguy cơ cháy nổ gia tăng.

### 3. Nhận xét:

- Hệ thống tuy đơn giản nhưng mang lại giá trị cao nếu triển khai đúng cách.
- Việc kết hợp giữa phần cứng (ESP32, cảm biến, còi báo) và phần mềm (ứng dụng IoT, nền tảng giám sát) giúp tăng tính thông minh cho thiết bị.

### 4. Kiến nghị

- **Nên triển khai rộng rãi** trong các khu dân cư, trường học, nhà máy.
- **Đào tạo người sử dụng** về cách nhận biết cảnh báo và xử lý khi có rò rỉ.
- **Nâng cấp hệ thống định kỳ** để đảm bảo độ chính xác và tránh báo giả.
- **Chính quyền địa phương** có thể hỗ trợ triển khai hệ thống này theo từng cụm dân cư.

### 5. Mở rộng hệ thống

Hệ thống có thể được mở rộng và nâng cấp theo nhiều hướng:

- Cảm biến nhiệt độ/độ ẩm để theo dõi môi trường.
- Cảm biến chuyển động để tự động tắt van gas khi không có người.
- Giao tiếp MQTT để tích hợp với hệ thống nhà thông minh.
- Gửi SMS/cuộc gọi tự động bằng mô-đun GSM.
- Kết nối Google Sheets hoặc Firebase để lưu log dữ liệu dài hạn.
- Tích hợp AI để phân tích dữ liệu bất thường và phát hiện nguy cơ tiềm ẩn.
- Sử dụng pin dự phòng (UPS) để đảm bảo hoạt động liên tục khi mất điện.
- Tích hợp thêm camera AI để nhận diện sự cố cháy hoặc con người.
- Kết nối với Google Home hoặc Amazon Alexa để điều khiển thông minh.
- Gửi cảnh báo đến đơn vị cứu hỏa địa phương trong trường hợp khẩn cấp.
- Thêm chức năng tự động ngắt nguồn điện, khóa van gas khi phát hiện nguy hiểm.
- Hiển thị thông báo trên đồng hồ thông minh hoặc thiết bị đeo tay của người dùng.

## Tài liệu tham khảo:

**ESP32 Documentation - Espressif Systems.** Tài liệu chính thức về ESP32, tính năng và cách lập trình:

- <https://docs.espressif.com/>
- <https://www.espressif.com/en/products/socs/esp32>

**MQ-6 Gas Sensor Datasheet** – Hanwei Electronics, SparkFun. Thông tin kỹ thuật cảm biến khí gas MQ-6:

- <https://www.winsen-sensor.com>
- <https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-6.pdf>

**ThingSpeak API Documentation** – MathWorks. Hướng dẫn sử dụng ThingSpeak để truyền và hiển thị dữ liệu IoT:

- <https://thingspeak.com/docs>
- <https://thingspeak.com/pages/tutorials>

**Blynk Documentation** – Blynk Inc. Hướng dẫn sử dụng Blynk để điều khiển và theo dõi từ xa:

- <https://blynk.io/docs>
- <https://docs.blynk.io/>

**Telegram Bot API** – Telegram. Hướng dẫn tạo và sử dụng bot Telegram để gửi cảnh báo:

- <https://core.telegram.org/bots/api>

**PlatformIO Documentation** – PlatformIO. Hướng dẫn lập trình ESP32 và mô phỏng dự án:

- <https://docs.platformio.org>

**Wokwi Simulator** – Công cụ mô phỏng mạch ESP32 trực tuyến:

- <https://wokwi.com/>

## Tài liệu học IoT & hệ thống nhúng:

- *IoT and Smart Systems: Using Sensors, Cloud Computing, and Big Data Analytics* – John Wiley & Sons, 2023.
- *Internet of Things: Architecture and Applications* – Raj Kamal, McGraw-Hill Education, 2023.
- *Arduino: A Technical Reference* – Jürgen Müller, 2022.

- *Designing Embedded Systems with PIC Microcontrollers* – Tim Wilmshurst, 2024.
- *Random Nerd Tutorials* – Học IoT căn bản: <https://randomnerdtutorials.com/>
- *Arduino Project Hub*: <https://create.arduino.cc/projecthub>

**Phụ lục:**

Phụ lục A – Mã nguồn Arduino

Phụ lục B – File diagram.json của Wokwi

Phụ lục C – Hình ảnh mô phỏng



