

|  |
| --- |
| **THIẾT KẾ MÔ HÌNH KIỂM SOÁT KHÔNG KHÍ TRONG NHÀ** |

Group 3

Bùi Hà Giang

Lê Quang Minh

Ngô Đức Mạnh

Nguyễn Hạnh Phúc

Nguyễn Lê Bảo Phúc

04/7/2025

**MỤC LỤC**

[DANH MỤC CÁC KÝ HIỆU, TỪ VIẾT TẮT 1](#_Toc202541588)

[DANH MỤC BẢNG BIỂU 2](#_Toc202541589)

[DANH MỤC HÌNH ẢNH 3](#_Toc202541590)

[MỞ ĐẦU 4](#_Toc202541591)

[Chương I. Tổng quan về hệ thống kiểm soát nhiệt độ và độ ẩm 8](#_Toc202541592)

[1.1 Mô hình hệ thống kiểm soát nhiệt độ và độ ẩm ứng dụng IoT 8](#_Toc202541593)

[1.1.1 Tổng quan 8](#_Toc202541594)

[1.1.2 Các phương pháp giám sát và điều khiển hệ thống 8](#_Toc202541595)

[1.2. ESP32 Dev Module [1] 9](#_Toc202541596)

[1.2.1. Đặc điểm 9](#_Toc202541597)

[1.2.2. Các chuẩn giao tiếp 10](#_Toc202541598)

[1.2.3. Cảm biến DHT11 [2] 14](#_Toc202541599)

[1.2.4. Cảm biến khí gas MQ2 [3],[4] 14](#_Toc202541600)

[Chương II. Thiết kế mô hình kiểm soát nhiệt độ,độ ẩm 16](#_Toc202541601)

[2.1. Phân tích yêu cầu bài toán 16](#_Toc202541602)

[2.1.1. Nhiệm vụ đề tài 16](#_Toc202541603)

[2.1.2. Mục tiêu thiết kế 16](#_Toc202541604)

[2.1.3. Điều kiện ràng buộc của thiết kế 16](#_Toc202541605)

[Về phần mềm, sử dụng phần mềm Arduino IDE để viết mã nguồn 16](#_Toc202541606)

[2.1.4. Thông số kỹ thuật 16](#_Toc202541607)

[2.1.5. Tiêu chí đánh giá sản phẩm 17](#_Toc202541608)

[2.2. THIẾT KẾ Ý TƯỞNG 17](#_Toc202541609)

[2.2.1. Vi điều khiển 17](#_Toc202541610)

[2.2.2 Cảm biến nhiệt độ và độ ẩm 18](#_Toc202541611)

[2.2.3 Cảm biến khí ga 19](#_Toc202541612)

[2.3. THIẾT KẾ CHI TIẾT 21](#_Toc202541613)

[2.3.1. Thiết kế phần cứng 21](#_Toc202541614)

[2.3.2. Code và lưu đồ thuật toán 22](#_Toc202541615)

[CHƯƠNG III: KẾT QUẢ THỬ NGHIỆM VÀ ĐÁNH GIÁ 24](#_Toc202541616)

[3.1.Mô hình hoàn thiện 24](#_Toc202541617)

[3.2.Kịch bản thử nghiệm 24](#_Toc202541618)

[3.3. Kết quả thử nghiệm 25](#_Toc202541619)

[KẾT LUẬN CHUNG 26](#_Toc202541620)

[TÀI LIỆU THAM KHẢO 30](#_Toc202541621)

PHỤ LỤC HƯỚNG DẪN SỬ DỤNG SẢN PHẨM 31

# DANH MỤC CÁC KÝ HIỆU, TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| Từ viết tắt | Tiếng Anh | Tiếng Việt |
| WiFi | Wireless Fidelity | Kết nối không dây |
| IoT | Internet of Things | Mạng lưới vạn vật kết nối Internet |
| I2C | Inter-Integrated Circuit | Giao tiếp liên tích hợp – giao tiếp nối tiếp hai dây |
| SPI | Serial Peripheral Interface | **Giao tiếp ngoại vi nối tiếp** |
| PWM | |  | | --- | | Pulse Width Modulation |  |  | | --- | |  | | Điều chế độ rộng xung – điều khiển tín hiệu bằng độ rộng xung |
| ADC | Analog to Digital Converter | Bộ chuyển đổi tín hiệu tương tự sang số |

# DANH MỤC BẢNG BIỂU

Bảng 1. Kết quả thực nghiệm

# DANH MỤC HÌNH ẢNH

[Hình 1.1 Hình ảnh ESP32 Dev Module 10](#_Toc202543531)

[Hình 1.2 Truyền thông giữa các vi mạch bằng giao thức I2C 11](#_Toc202543532)

[Hình 1.3 Truyền thông giữa các vi mạch bằng giao thức 1-Wire 13](#_Toc202543533)

[Hình 1.4 – Sơ đồ chân DHT11 dạng module và dạng linh kiện đơn 15](#_Toc202543534)

[Hình 1.5 – Sơ đồ chân của cảm biến MQ-2 dạng module 15](#_Toc202543535)

[Hình 2.1 Sơ đồ khối tổng quát 21](#_Toc202543536)

[Hình 2.2 Lưu đồ thuật toán 23](#_Toc202543537)

[Hình 3.1. Hình ảnh mô hình trên board test 24](#_Toc202543538)

[Hình 3.2 Hình ảnh nhiệt ẩm kế thương mại và thông số kỹ thuật 28](#_Toc202543539)

[Hình 0.1 Mô hình hoàn thiện 32](#_Toc202543540)

# MỞ ĐẦU

1. **Nội dung đề tài**

Trong thực tiễn hiện nay, khi thời tiết ngày càng trở nên cực đoan và thất thường – với những đợt nắng nóng kéo dài, độ ẩm giảm sâu hoặc nhiệt độ chênh lệch lớn giữa ngày và đêm – việc duy trì điều kiện nhiệt độ phù hợp trong không gian sống không còn đơn thuần là vấn đề tiện nghi mà đã trở thành một yêu cầu thiết yếu để bảo vệ sức khỏe và cải thiện chất lượng cuộc sống, đặc biệt tại các khu dân cư đông đúc hoặc các vùng khí hậu khắc nghiệt.

Mặt khác, việc sử dụng các thiết bị làm mát như quạt, máy lạnh hoặc máy tạo ẩm theo phương pháp thủ công, thiếu giám sát và không tự động điều chỉnh theo môi trường, dễ dẫn đến tình trạng lãng phí năng lượng hoặc không hiệu quả về mặt điều hòa nhiệt độ. Thực tế này đặt ra nhu cầu cấp thiết cho một giải pháp có khả năng thích nghi theo thời gian thực – nghĩa là hệ thống có thể cảm nhận được nhiệt độ, phân tích và đưa ra phản hồi phù hợp mà không cần người dùng can thiệp liên tục.

Sự phát triển nhanh chóng của công nghệ Internet of Things (IoT) mở ra một hướng đi khả thi. Các cảm biến, bộ điều khiển và nền tảng giám sát từ xa cho phép thiết bị “giao tiếp” với nhau thông qua mạng internet, tạo thành một hệ thống thông minh – có khả năng học hỏi, ghi nhớ trạng thái môi trường và đưa ra quyết định dựa trên dữ liệu thu thập được.

Chính từ bối cảnh trên, đề tài “Phát triển hệ thống kiểm soát nhiệt độ trong nhà bằng IoT” được lựa chọn nhằm tiếp cận bài toán một cách toàn diện: không chỉ giải quyết vấn đề điều hòa nhiệt độ mà còn ứng dụng được kiến thức về hệ thống nhúng, lập trình điều khiển, mạng không dây và xử lý tín hiệu môi trường. Hơn thế nữa, giải pháp này có tính ứng dụng cao, phù hợp để nhân rộng trong các mô hình nhà ở thông minh, nhà kính, văn phòng, hoặc không gian công cộng cần giám sát điều kiện môi trường ổn định và tiết kiệm năng lượng.

**2. Mục tiêu nghiên cứu**

Thiết kế một hệ thống có thể duy trì nhiệt độ trong không gian kín. Có thể thiết lập nhiệt độ mà người sử dụng mong muốn, dựa theo yêu cầu đó hệ thống sẽ điều khiển các thiết bị để điều chỉnh nhiệt độ phòng đáp ứng mong muốn của người sử dụng.

Qua đề tài này, các thành viên trong nhóm có thể hiểu hơn về cách thức hoạt động của vi điều khiển ESP32 trong thực tế, đồng thời tìm hiểu cách sử dụng 1 số giao thức để giao tiếp giữa vi điều khiển với cảm biến và một số thiết bị ngoại vi như:I2C,PWM,HTTP.

3. Đối tượng nghiên cứu

Đối tượng nghiên cứu trong đồ án này bao gồm:

- Thiết kế mạch vi điều khiển ESP32, tìm hiểu các chức năng tương ứng của vi điều khiển ESP32 để ứng dụng cho việc điều khiển và giám sát.

- Tìm hiểu, sử dụng Platform IO viết code điều khiển cho mô hình kiểm soát nhiệt độ,độ ẩm

- Cảm biến nhiệt độ được sử dụng: DHT11

- Các giao thức được sử dụng trong đề tài I2C, PWM,HTTP.

1. **Mục đích nghiên cứu**

Mục đích nhóm chúng em đưa ra đề tài này nhằm ứng dụng công nghệ IOT, giúp con người có thể điều khiển và giám sát nhiệt độ,độ ẩm bất cứ nơi Do đó, nhóm chúng em đưa ra mục tiêu phát triển của mô hình như sau:

- Giám sát nhiệt độ và độ ẩm theo thời gian thực.

- Cảnh báo khi nhiệt độ hoặc độ ẩm vượt ngưỡng cho phép.

- Cho phép người dùng theo dõi và điều khiển hệ thống từ xa thông qua điện thoại thông minh.

1. **Nhu cầu thực tế cần giải quyết**

Hệ thống kiểm soát nhiệt độ và độ ẩm ứng dụng công nghệ IoT không chỉ là một thiết bị đo đạc thông thường, mà còn là một giải pháp giám sát môi trường toàn diện, kết hợp giữa cảm biến, kết nối mạng và phần mềm điều khiển nhằm mang lại sự an tâm và tiện lợi cho người sử dụng trong các tình huống sau:

- Người dùng cần một hệ thống giám sát nhiệt độ, độ ẩm chính xác, tự động và dễ quản lý hơn so với phương pháp thủ công.

- Giảm thiểu rủi ro hư hỏng hàng hóa, thiết bị hoặc ảnh hưởng đến môi trường làm việc do điều kiện nhiệt độ hoặc độ ẩm vượt ngưỡng.

- Đáp ứng xu hướng phát triển của **IoT (Internet of Things) và smart city , đặc biệt trong các lĩnh vực như nông nghiệp, kho bãi, y tế và nhà thông minh**.

1. **Phương pháp nghiên cứu**

Để hoàn thiện học phần Đồ án chuyên ngành, nhóm em sử dụng hai

phương pháp nghiên cứu chính:

- Phương pháp thu thập thông tin: nghiên cứu tài liệu về ESP32, module DHT11 và các giao thức được sử dụng trong bài báo cáo thông qua Internet.

- Thực nghiệm: Chạy mạch thực nghiệm sản phẩm dựa trên kịch bản thực nghiệm đã thiết kế và ghi chép lại kết quả.

- Phân tích: Phân tích kết quả thu được, từ đó so sánh sản phẩm với các sản phẩm có tính năng tương tự trên thị trường.

1. **Ý nghĩa thực tiễn của đề tài**

**- Hệ thống giám sát nhiệt độ và độ ẩm tích hợp ESP32 và các cảm biến môi trường được thiết kế để thu thập dữ liệu chính xác theo thời gian thực và cảnh báo kịp thời khi các thông số vượt ngưỡng cho phép, góp phần đảm bảo điều kiện môi trường ổn định và hỗ trợ quản lý từ xa một cách hiệu quả, mang lại sự an tâm và tiện lợi cho người sử dụng.**

**- Mô hình có thể được mở rộng để tích hợp thêm các tính năng thông minh, ví dụ như điều khiển thiết bị làm mát/làm ẩm tự động, lưu trữ dữ liệu lên nền tảng đám mây để theo dõi xu hướng dài hạn, hoặc áp dụng trí tuệ nhân tạo (AI) để dự đoán và tối ưu điều kiện môi trường theo nhu cầu sử dụng, từ đó nâng cao hiệu quả vận hành và tự động hóa toàn hệ thống.**

1. **Cấu trúc của báo cáo**

Đề tài: “Thiết kế hệ thống kiểm soát nhiệt độ và độ ẩm ứng dụng IoT”ngoài phần mở đầu và kết luận, danh mục từ viết tắt, danh mục bảng và biểu đồ, danh mục tài liệu tham khảo và phụ lục, báo cáo được kết cấu thành 3 chương như sau:

**Mở đầu:** *Giới thiệu về lý do chọn đề tài, lý do ra đời khóa thông minh, đối tượng nghiên cứu, mục đích nghiên cứu, nhu cầu thực tế cần giải quyết, phương pháp nghiên cứu, ý nghĩa thực tiễn và cấu trúc của báo cáo.*

**Chương 1: Tổng quan về hệ thống kiểm soát nhiệt độ và độ ẩm**

**Chương 2: Thiết kế mô hình hệ thống kiểm soát nhiệt độ và độ ẩm**

**Chương 3: Kết quả thực nghiệm và đánh giá.**

**Kết luận**

# Chương I. Tổng quan về hệ thống kiểm soát nhiệt độ và độ ẩm

## 1.1 Mô hình hệ thống kiểm soát nhiệt độ và độ ẩm ứng dụng IoT

### 1.1.1 Tổng quan

Hệ thống kiểm soát nhiệt độ và độ ẩm ứng dụng IoT là mô hình sử dụng các cảm biến môi trường (như DHT11, DHT22, SHT21...) kết hợp với vi điều khiển (ví dụ ESP32) và nền tảng IoT để giám sát các thông số môi trường theo thời gian thực. Dữ liệu thu thập được có thể được lưu trữ, phân tích và hiển thị trên điện thoại hoặc máy tính từ xa thông qua mạng Internet. Ngoài chức năng giám sát, hệ thống còn có thể điều khiển thiết bị làm mát, làm ấm, hoặc tạo ẩm một cách tự động khi nhiệt độ/độ ẩm vượt ngưỡng cho phép.

### 1.1.2 Các phương pháp giám sát và điều khiển hệ thống

Có nhiều kỹ thuật và giao thức có thể được áp dụng để xây dựng và điều hành hệ thống kiểm soát nhiệt độ – độ ẩm:

* Kết nối qua mạng Internet:
* Sử dụng Wi-Fi, Ethernet hoặc mạng di động (4G/5G) để truyền dữ liệu đến server hoặc nền tảng đám mây (Firebase, Thingspeak, Blynk...).
* Cho phép theo dõi và điều khiển từ xa qua ứng dụng hoặc website.
* Kết nối nội bộ không dây:
* Sử dụng các giao thức không dây tầm ngắn như Bluetooth, Zigbee hoặc LoRa để truyền dữ liệu trong mạng cục bộ.
* Phù hợp với những mô hình không yêu cầu truy cập Internet nhưng cần giám sát tập trung.
* Điều khiển qua thiết bị di động:
* Ứng dụng điều khiển cài đặt trên điện thoại có thể nhận dữ liệu từ hệ thống và gửi lệnh điều khiển.
* Kết hợp với thông báo đẩy (push notification) để cảnh báo khi vượt ngưỡng.
* Tự động hóa và thu thập dữ liệu:
* Sử dụng cảm biến để đo liên tục các giá trị nhiệt độ và độ ẩm.
* Dữ liệu được ghi lại và phân tích để điều khiển thiết bị (quạt, máy lạnh, máy tạo ẩm...) theo kịch bản định sẵn.

## 1.2. ESP32 Dev Module [1]

### 1.2.1. Đặc điểm

ESP32 là một hệ thống vi điều khiển trên chip (SoC) giá rẻ của Espressif Systems, nhà phát triển của ESP8266 SoC. Nó là sự kế thừa của SoC ESP8266 và có cả hai biến thể lõi đơn và lõi kép của bộ vi xử lý 32-bit Xtensa LX6 của Tensilica có thể chạy ở tốc độ lên đến 240 MHz, ESP32 đáp ứng tốt các ứng dụng cần xử lý nhanh và đa nhiệm với Wi-Fi và Bluetooth tích hợp.

Ngoài ra, ESP32 còn được trang bị nhiều chân GPIO và các giao thức giao tiếp ngoại vi như UART, SPI, I2C, PWM, ADC, DAC…, cho phép điều khiển linh hoạt các thiết bị ngoại vi. Nó cũng hỗ trợ các tính năng cảm ứng chạm, chế độ tiết kiệm năng lượng (deep sleep), và cập nhật chương trình từ xa (OTA). Với bộ nhớ trong gồm 520 KB SRAM và bộ nhớ flash từ 4 MB trở lên (tùy module), cùng khả năng lập trình bằng VS code, Arduino IDE, ESP-IDF, MicroPython hoặc PlatformIO, ESP32 trở thành một nền tảng mạnh mẽ, linh hoạt và được sử dụng rộng rãi trong các dự án nhúng và điều khiển từ xa.



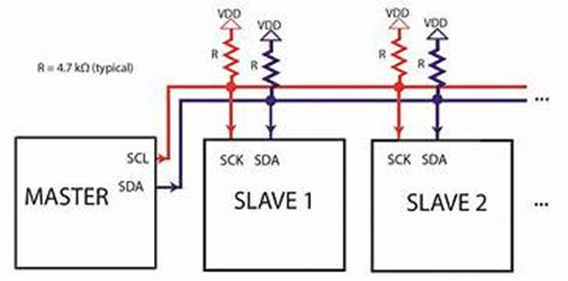
Hình 1.1 Hình ảnh ESP32 Dev Module

### 1.2.2. Các chuẩn giao tiếp

Trong sản phẩm của nhóm sử dụng 3 chuẩn giao tiếp là I2C,PWM và HTTP để phục vụ giao tiếp giữa vi điều khiển, thiết bị ngoại vi và app điều khiển trên điện thoại.

a) Chuẩn giao tiếp **I2C**

**- Mô tả:** ESP32 hỗ trợ tối đa 2 cổng I2C phần cứng. I2C là chuẩn truyền thông nối tiếp hai dây phổ biến (SDA – Dữ liệu, SCL – Xung nhịp), cho phép giao tiếp với nhiều thiết bị (tối đa 127 thiết bị) chỉ qua 2 chân tín hiệu. Tốc độ truyền có thể đạt đến 400 kHz (Fast mode), hoặc 1 MHz (Fast mode plus).



Hình 1.2 Truyền thông giữa các vi mạch bằng giao thức I2C

**- Tính năng:**

* + Giao tiếp đa thiết bị, sử dụng địa chỉ thiết bị (7-bit hoặc 10-bit).
  + Hỗ trợ truyền dữ liệu 8-bit, có cơ chế ACK/NACK kiểm tra lỗi truyền.
  + Giao tiếp đồng bộ, dễ mở rộng hệ thống cảm biến.
  + Có thể cấu hình chân SDA/SCL tùy ý nhờ GPIO Matrix của ESP32.

**- Ứng dụng trong hệ thống giám sát nhiệt độ – độ ẩm:** Chuẩn I2C thường được sử dụng để kết nối ESP32 với các cảm biến môi trường như SHT20, BH1750 hoặc BME280 – những cảm biến đo đa thông số (nhiệt độ, độ ẩm, áp suất). Việc sử dụng I2C giúp đơn giản hóa kết nối phần cứng, giảm số lượng dây nối, tiết kiệm chân GPIO và thuận tiện mở rộng hệ thống với nhiều cảm biến khác nhau.

b) Giao tiếp PWM (Pulse Width Modulation – Điều chế độ rộng xung)

-Mô tả: ESP32 hỗ trợ phát tín hiệu PWM trên hầu hết các chân GPIO nhờ bộ điều khiển LEDC (LED Control). PWM là kỹ thuật điều khiển tín hiệu đầu ra dạng xung, dùng phổ biến để điều chỉnh độ sáng LED, tốc độ motor, hoặc điều khiển các thiết bị điện tử.

- Tính năng:

* + Tần số PWM có thể cấu hình từ vài Hz đến 40 MHz.
  + Độ phân giải lên đến 16 bit (tùy cấu hình).
  + Có thể chạy độc lập trên nhiều kênh cùng lúc.
  + Hỗ trợ thay đổi duty cycle theo thời gian thực.

- Ứng dụng trong hệ thống giám sát nhiệt độ – độ ẩm:  
 PWM có thể được sử dụng để điều khiển tốc độ quạt thông gió, độ sáng đèn cảnh báo, hoặc kích hoạt các thiết bị điều hòa nhiệt độ như máy phun sương tạo ẩm. Nhờ khả năng điều chỉnh chính xác, PWM cho phép hệ thống phản ứng linh hoạt với sự thay đổi của nhiệt độ và độ ẩm.

c) Giao tiếp TCP

- Mô tả: TCP/IP hoặc Transmission Control Protocol/Internet Protocol (Giao thức điều khiển truyền vận/giao thức mạng) là một bộ các giao thức trao đổi thông tin được sử dụng để kết nối các thiết bị mạng trên Internet. TCP/IP có thể được sử dụng như là một giao thức trao đổi thông tin trong một mạng riêng ([intranet](https://quantrimang.com/cong-nghe/ly-thuyet-intranet-la-gi-117238" \o "Lý thuyết - Intranet là gì?) hoặc extranet)

- Tính năng:

TCP/IP chỉ định cách dữ liệu được trao đổi qua Internet bằng cách cung cấp thông tin trao đổi đầu cuối nhằm mục đích xác định cách thức nó được chia thành các gói, được gắn địa chỉ, vận chuyển, định tuyến và nhận ở điểm đến. TCP/IP không yêu cầu quản lý nhiều và nó được thiết kế để khiến mạng đáng tin cậy hơn với khả năng phục hồi tự động..

- Ứng dụng trong hệ thống giám sát nhiệt độ – độ ẩm:  
 TCP cho phép ESP32 truyền dữ liệu đo được từ cảm biến lên nền tảng đám mây để lưu trữ và hiển thị. Ngoài ra, người dùng có thể điều khiển từ xa thông qua giao diện web hoặc app bằng cách gửi yêu cầu TCP/IP đến ESP32. Điều này giúp nâng cao tính tương tác và khả năng quản lý thiết **bị từ xa qua Internet.**

d) Chuẩn giao tiếp **1-Wire**

**- Mô tả**:  
 1-Wire là một chuẩn giao tiếp nối tiếp do Dallas Semiconductor (nay là Maxim Integrated) phát triển, cho phép truyền dữ liệu và cấp nguồn chỉ thông qua **một dây tín hiệu duy nhất** (cộng với dây GND).

Ảnh có chứa hàng, biểu đồ, văn bản, Song song

Nội dung do AI tạo ra có thể không chính xác.

Hình 1.3 Truyền thông giữa các vi mạch bằng giao thức 1-Wire

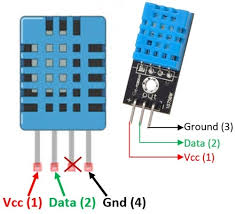
**- Tính năng**:

* + Chỉ cần **1 dây tín hiệu + GND** cho cả truyền dữ liệu và cấp nguồn (trong chế độ parasitic power).
  + Mỗi thiết bị trên bus 1-Wire có **mã địa chỉ 64-bit duy nhất**, cho phép kết nối nhiều cảm biến cùng lúc.
  + Tốc độ truyền dữ liệu thấp (~16.3 kbps), nhưng **ổn định và đơn giản**.
  + Giao tiếp đồng bộ, điều khiển bởi Master (ESP32 đóng vai trò Master).

**- Ứng dụng trong hệ thống giám sát nhiệt độ – độ ẩm**: Chuẩn 1-Wire thường được sử dụng để kết nối **cảm biến nhiệt độ DHT11**với ESP32. Loại cảm biến này có độ chính xác cao (±0.5°C), hoạt động tốt trong môi trường từ -55°C đến +125°C và đặc biệt hữu ích khi cần đo **nhiệt độ tại nhiều điểm** bằng cách mắc nối tiếp nhiều cảm biến trên cùng một dây. Ứng dụng này rất phù hợp cho hệ thống giám sát môi trường trong nhà kính, kho lạnh, chuồng trại chăn nuôi, hoặc các khu vực yêu cầu đo nhiệt độ phân tán.

### 1.2.3. Cảm biến DHT11 [2]

Cảm biến DHT11 là một loại cảm biến nhiệt độ và độ ẩm kỹ thuật số phổ biến, sử dụng giao tiếp 1-wire đơn giản để truyền dữ liệu đến vi điều khiển. Trong đề tài này, DHT11 được kết nối với vi điều khiển ESP32 thông qua giao tiếp 1-wire, cho phép sử dụng linh hoạt bất kỳ chân GPIO nào, từ đó tiết kiệm số chân I/O và đơn giản hóa quá trình thiết kế mạch.

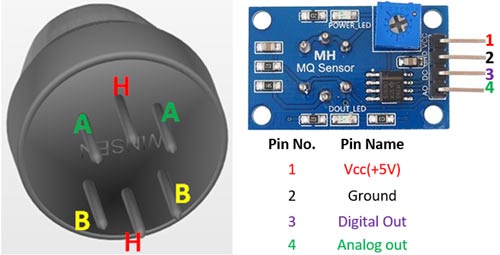


**Hình 1.4** – Sơ đồ chân DHT11 dạng module và dạng linh kiện đơn

Cảm biến DHT11 tích hợp cả hai chức năng đo nhiệt độ và độ ẩm trong cùng một thiết bị. Dữ liệu được truyền dưới dạng số (digital) thông qua một chân tín hiệu duy nhất. Giao tiếp 1-wire của DHT11 sử dụng một giao thức riêng (không phải chuẩn 1-Wire của Maxim), nhưng vẫn dựa trên nguyên tắc truyền – nhận nối tiếp và có thể dễ dàng xử lý bằng phần mềm

### 1.2.4. Cảm biến khí gas MQ2 [3],[4]

Cảm biến MQ-2 là một cảm biến khí gas bán dẫn, chuyên dùng để phát hiện các loại khí dễ cháy như LPG, butan, methane, cồn, hydrogen, và khói. Trong đề tài này, MQ-2 được kết nối với vi điều khiển ESP32 thông qua chân ADC, vì đầu ra của cảm biến là tín hiệu analog, có điện áp thay đổi theo nồng độ khí đo được. ESP32 được tích hợp bộ chuyển đổi ADC 12-bit, cho phép đọc tín hiệu điện áp từ MQ-2 một cách chính xác, phục vụ cho việc đánh giá mức độ ô nhiễm không khí.



**Hình 1.5** – Sơ đồ chân của cảm biến MQ-2 dạng module

Trong hệ thống này, ESP32 đọc giá trị điện áp analog từ chân A0 của MQ-2, sau đó thực hiện chuyển đổi ADC để xác định mức nồng độ khí trong không khí. Khi giá trị vượt ngưỡng, hệ thống có thể:

* Gửi cảnh báo qua ứng dụng di động hoặc nền tảng IoT (như Firebase, Blynk...).
* Tự động kích hoạt quạt hút khí hoặc còi báo độ

# Chương II. Thiết kế mô hình kiểm soát nhiệt độ,độ ẩm

## 2.1. Phân tích yêu cầu bài toán

### 2.1.1. Nhiệm vụ đề tài

Thiết kế một mô hình kiểm soát nhiệt độ trong nhà tùy theo ý muốn của người sử dụng.

### 2.1.2. Mục tiêu thiết kế

Thiết kế hệ thống kiểm soát nhiệt độ có các chức năng:

- Đo nhiệt độ,độ ẩm phòng

- Hệ thống tự động kiểm soát nhiệt độ ở khoảng mong muốn của người dùng thông qua điều khiển quạt.

- Đo nồng độ khí ga trong phòng nếu nồng độ khí ga vượt ngưỡng an toàn thì điều khiển quạt thông gió.

- Có khả năng báo động khi nhiệt độ phòng và nồng độ khí ga bất thường

### 2.1.3. Điều kiện ràng buộc của thiết kế

Về phần cứng mạch cần được thiết kế phù hợp với các linh kiện, mạch phải hoạt động được đúng với chức năng. Thành phần nguồn cung cấp duy trì cho mạch phải ổn định ở mức 5V và đảm bảo an toàn điện. Các thành phần mạch được bố trí hợp lí để mạch đáp ứng được kích thước nhỏ gọn và điều khiển dễ dàng nhất.

### Về phần mềm, sử dụng phần mềm Arduino IDE để viết mã nguồn

### 2.1.4. Thông số kỹ thuật

- Mạch sử dụng LM2596 cung cấp nguồn 5 VDC hạ áp từ apdapter 18v.

- Oled 0.92 inch, động cơ servo S90, còi 5v.

- Hệ thống được lắp ráp trên board test, không thiết kế mạch in, bố trí linh kiện linh hoạt phục vụ thử nghiệm

### 2.1.5. Tiêu chí đánh giá sản phẩm

Thông qua nhiệm vụ đề tài và mục tiêu thiết kế. Nhóm đã tổng hợp được tiêu chí đánh giá sản phẩm như sau.

- Mach chạy đúng theo yêu cầu.

- Nhiệt độ, độ ẩm đọc được từ cảm biến DHT11 chênh lệch dưới 5 độ C (5%) so với nhiệt độ, độ ẩm được thông báo từ đài khí tượng thủy văn quốc gia.

- Cảm biến khí ga hoạt động hiệu quả, trả lại giá trị analog với độ phân giải 12 bits.

- Gửi dữ liệu lên blynk mỗi 1s.

- Tự động điều khiển còi, led đơn, động cơ servo khi nhiệt độ hoặc khí ga vượt ngưỡng mong muốn.

## 2.2. THIẾT KẾ Ý TƯỞNG

### 2.2.1. Vi điều khiển

*- Đề xuất 1: Sử dụng PIC 18f2580:*

Là một vi điều khiển 8 bit hiệu suất cao, PIC18F2580 được sử dụng rộng rãi trong các ứng dụng nhúng và dự án điện tử. Vi điều khiển này thường được lập trình bằng ngôn ngữ C và C++. Với 28 chân I/O linh hoạt, 31 thanh ghi, và 4 bộ timer/counter có thể lập trình, PIC18F2580 hỗ trợ đa dạng các chức năng. Nó tích hợp các ngắt nội và ngoại (2 mức ưu tiên ngắt), cùng với các giao thức truyền thông nối tiếp như CAN, USART, SPI, và I2C.

Ngoài ra, vi điều khiển này còn trang bị bộ chuyển đổi tín hiệu tương tự sang số (ADC) 10 bit, mở rộng lên đến 10 kênh, cùng với khả năng lập trình Watchdog Timer. PIC18F2580 hoạt động với 4 chế độ nguồn khác nhau và hỗ trợ 1 kênh điều chế độ rộng xung (PWM) được tích hợp trên chân RC2, mang lại sự linh hoạt và hiệu quả trong thiết kế hệ thống.

*-Đề xuất 2: ESP32*

ESP32 là một series các vi điều khiển trên một vi mạch giá rẻ, năng lượng thấp có tích hợp WiFi và dual-mode Bluetooth (tạm dịch: Bluetooth chế độ kép). Dòng ESP32 sử dụng bộ vi xử lý Tensilica Xtensa LX6 có hai biến thể lõi kép và lõi đơn, và bao gồm các công tắc antenna tích hợp, RF balun, bộ khuếch đại công suất, bộ khuếch đại thu nhiễu thấp, bộ lọc và module quản lý năng lượng.

***Kết luận:*** *Với giá thành rẻ, hiệu năng mạnh mẽ và kết hợp wifi. ESP32 là sự lựa chọn tốt hơn cho các ứng dụng IOT vừa và nhỏ khi so sánh với PIC18F2580.*

### 2.2.2 Cảm biến nhiệt độ và độ ẩm

*- Đề xuất 1: Sử dụng DHT11*

Cảm biến độ ẩm và nhiệt độ DHT11 là cảm biến rất thông dụng hiện nay vì chi phí rẻ và rất dễ lấy dữ liệu thông qua chuẩn giao tiếp 1 wire.

Chuẩn giao tiếp 1 wire là dùng 1 chân Digital để truyền dữ liệu.

Bộ tiền xử lý tín hiệu được tích hợp trong cảm biến giúp bạn có thể đọc dữ liệu chính xác mà không phải qua bất kỳ tính toán nào.

Thông số:

* Điện áp hoạt động: 3V – 5V (DC)
* Dãi nhiệt độ hoạt động: 0°C ~ 50°C, sai số ±2°C

*- Đề xuất 2:Sử dụng DS18B20*

Cảm biến nhiệt độ DS18B20 là cảm biến ( loại digital ) đo nhiệt độ mới của hãng MAXIM với độ phân giải cao ( 12bit ). Sử dụng giao tiếp 1-Wire

Ds18b20 có thể hoạt động ở 125 độ. Đây là cảm biến kỹ thuật số, nên không bị suy hao tín hiệu đường dây dài

Thông số:

* Nguồn: 3 – 5.5V
* Dải đo nhiệt độ: -55 đến 125 độ C ( -67 đến 257 độ F)
* Sai số: ± 0.5 độ C khi đo ở dải -10 – 85 độ C

***Kết luận:*** *DHT11 có giá thành rẻ và đồng thời cung cấp khả năng đo cả giá trị nhiệt độ và độ ẩm đem đã thể hiện khả năng vượt trội hơn trong đề tài này.*

### 2.2.3 Cảm biến khí ga

*- Đề xuất 1: Sử dụng MQ-2*

Các cảm biến MQ-2 có thể phát hiện và đo khí LPG, Rượu, Propane, Hydrogen, CO, và thậm chí là metan. Phiên bản module của cảm biến này có chân digital giúp cảm biến hoạt động ngay cả khi không có vi điều khiển và rất hữu ích khi chỉ cần phát hiện một loại khí cụ thể.

Với các đặc điểm nổi bật:

* Điện áp hoạt động + 5V
* Sử dụng để đo hoặc phát hiện khí LPG, Alcohol, Propane, Hydrogen, CO và thậm chí cả methane
* Điện áp đầu ra analog: từ 0V đến 5V
* Điện áp đầu ra digital: 0V hoặc 5V (chuẩn logic TTL)
* Thời gian làm nóng trước khi sử dụng 20 giây
* Có thể sử dụng như một cảm biến digital hoặc analog
* Độ nhạy của chân DO (digital output) có thể được thay đổi bằng cách chiết áp

*- Đề xuất 2: Sử dụng MQ-3*

Cảm biến MQ3 nồng độ cồn sử dụng để phát hiện nồng độ cồn trong môi trường, hơi thở. Cảm biến có độ nhạy cao khả năng phản hồi nhanh, độ nhạy có thể điều chỉnh được bằng biến trở. Cảm biến cung cấp một đầu ra tương tự dựa trên nồng độ cồn.

***Kết luận :*** *Với đề tài đo nồng độ khí ga trong nhà và cảnh báo khi có trường hợp bất thường thì cảm biến MQ-2 có chức năng phù hợp hơn so với MQ-3*

***2.2.4 Nền tảng IoT hỗ trợ phân tích dữ liệu***

-Đề xuất 1: Sử dụng ThingSpeak

ThingSpeak là một nền tảng IoT mã nguồn mở hỗ trợ thu thập, lưu trữ, phân tích và trực quan hóa dữ liệu cảm biến theo thời gian thực. Nền tảng này cho phép người dùng gửi dữ liệu lên cloud thông qua HTTP hoặc MQTT, sau đó hiển thị bằng các biểu đồ dạng đường (line chart) đơn giản và dễ quan sát. Một điểm mạnh nổi bật của ThingSpeak là khả năng tích hợp MATLAB để thực hiện các phân tích dữ liệu nâng cao như lọc nhiễu, tính toán trung bình, hoặc phát hiện xu hướng. Ngoài ra, ThingSpeak cũng hỗ trợ các phản hồi tự động thông qua tính năng React, giúp người dùng xây dựng hệ thống giám sát thông minh mà không cần máy chủ riêng.

-Đề xuất 2: Sử dụng Adafruit IO

Adafruit IO là một nền tảng IoT trực tuyến được thiết kế để dễ dàng kết nối và hiển thị dữ liệu từ các thiết bị như ESP32 hoặc Arduino. Nền tảng này cung cấp các dashboard trực quan, hỗ trợ nhiều dạng hiển thị như biểu đồ đường, đồng hồ đo (gauge), công tắc điều khiển, bản đồ GPS và hơn thế nữa. Adafruit IO hỗ trợ tốt các giao thức MQTT và HTTP, đồng thời cung cấp ứng dụng di động giúp theo dõi và điều khiển thiết bị từ xa. Tuy nhiên, Adafruit IO không có công cụ phân tích dữ liệu chuyên sâu và giới hạn lưu trữ miễn phí theo thời gian.

**Kết luận:** *Nếu mục tiêu là không chỉ hiển thị dữ liệu mà còn thực hiện các phân tích chuyên sâu phục vụ học thuật, nghiên cứu hoặc các dự án kỹ thuật, thì ThingSpeak là lựa chọn phù hợp hơn. Với khả năng tích hợp MATLAB mạnh mẽ, hỗ trợ API đầy đủ, và tính năng phản hồi tự động, ThingSpeak mang lại một giải pháp toàn diện hơn trong việc xây dựng hệ thống IoT theo dõi và xử lý dữ liệu thời gian thực.*

## 2.3. THIẾT KẾ CHI TIẾT

### 2.3.1. Thiết kế phần cứng

a) Sơ đồ tổng quát

**Ảnh có chứa văn bản, biểu đồ, Kế hoạch, hàng

Nội dung do AI tạo ra có thể không chính xác.**

Hình 2.1 Sơ đồ khối tổng quát

Hình 2.1 biểu diễn quá trình hoạt động của mạch thông qua 7 khối chính: khối nguồn, khối hiển thị, khối xử lý, khối cảm biến, khối kiểm soát nhiệt độ, khối thông báo và khối giao diện người dùng lần lượt:

- Khối nguồn: bao gồm một adapter chuyển đổi điện áp 18 V thành 5 V thông qua LM2576.

- Khối hiển thị: gồm màn hình oled 128x64.

-Khối cảm biến: Gồm 2 cảm biến DHT11 và MQ-2 để đo nhiệt độ, độ ẩm môi trường và nồng độ khí ga.

- Khối xử lý: Khối điều khiển gồm 1 ESP32 để điều khiển thiết bị ngoại vi và các linh kiện trong mạch.

- Khối kiểm soát nhiệt độ: Gồm một động cơ servo S90 quay 380. Mô phỏng quạt thông khí trong thực tế.

- Khối thông báo: Gồm 2 led đơn xanh đỏ và còi 5VDC. Tại điều kiện bình thường led xanh sáng, các thiết bị ngoại vi khác không hoạt động, khi mà nhiệt độ phòng hoặc khí ga vượt ngưỡng cho phép thì còi kêu và led đỏ sáng.

### 2.3.2. Code và lưu đồ thuật toán

a) Lưu đồ thuật toán

**Ảnh có chứa văn bản, biểu đồ, ảnh chụp màn hình, Song song

Nội dung do AI tạo ra có thể không chính xác.**

Hình 2.2 Lưu đồ thuật toán

b) Code

Phần code của mạch đã được tổng hợp ở link dưới đây: https://github.com/Minhhauier/IOT\_SIC

# CHƯƠNG III: KẾT QUẢ THỬ NGHIỆM VÀ ĐÁNH GIÁ

## 3.1.Mô hình hoàn thiện

Ảnh có chứa Kỹ thuật điện, đồ điện tử, Dây điện, dây cáp

Nội dung do AI tạo ra có thể không chính xác.

Hình 3.1. Hình ảnh mô hình trên board test

## 3.2.Kịch bản thử nghiệm

- TH1: *Điều chỉnh nhiệt độ thấp hơn nhiệt độ mong muốn và không có khí gas*

Yêu cầu: Đèn và còi, servo sẽ không hoạt động ,gửi kết quả lên blynk

- TH2:Điều *nhiệt độ bằng với nhiệt độ mong muốn và không có khí gas*

Yêu cầu: : Đèn và còi, servo sẽ không hoạt động ,gửi kết quả lên blynk

- TH3: *Điều chỉnh nhiệt độ cao hơn nhiệt độ mong muốn và không có khí gas*

Yêu cầu: Đèn và còi,servo sẽ hoạt động ,gửi kết quả lên blynk

- TH4: *Điều chỉnh nhiệt độ cao hơn nhiệt độ mong muốn và có khí gas*

Yêu cầu: Đèn và còi, servo sẽ hoạt động ,gửi kết quả lên blynk

- TH5: *Điều chỉnh nhiệt độ thấp hơn nhiệt độ mong muốn và không có khí gas*

Yêu cầu: Đèn và còi, servo sẽ hoạt động ,gửi kết quả lên blynk

- TH6: *Điều chỉnh nhiệt độ thấp hơn nhiệt độ mong muốn và có khí gas*

Yêu cầu: Đèn và còi, servo sẽ hoạt động ,gửi kết quả lên blynk

## 3.3. Kết quả thử nghiệm

Bảng 1. Kết quả thực nghiệm

|  |  |  |
| --- | --- | --- |
| TRƯỜNG HỢP | KẾT QUẢ | GHI CHÚ |
| 1 | Đạt | Chạy đúng yêu cầu |
| 2 | Đạt | Chạy đúng yêu cầu |
| 3 | Đạt | Chạy đúng yêu cầu |
| 4 | Đạt | Chạy đúng yêu cầu |
| 5 | Đạt | Chạy đúng yêu cầu |
| 6 | Đạt | Chạy đúng yêu cầu |

**3.4.Kết luận**

Mạch chạy đúng tất cả yêu cầu của nội dung thiết kế : đọc dữ liệu về nhiệt độ, khí gas .gửi kết quả lên blynk điềi khiển thiết bị ngoại vi theo yêu cầu.

## 3.4 Phân tích dữ liệu trên ThingSpeak

Ảnh có chứa văn bản, biểu đồ, ảnh chụp màn hình, Sơ đồ

Nội dung do AI tạo ra có thể không chính xác.

Hình 3.2 Dữ liệu dưới dạng biểu đồ trên ThingSpeak

Tại những địa điểm dữ liệu thay đổi mạnh là do tác động của nhóm vào các cảm biến để thể hiển rõ dàng hơn sự thay đổi giá trị được đẩy lên thingspeak. Khi không có sự tác động của con người, nhiệt độ và nồng độ khí ga không thay đổi hoặc thay đổi rất nhỏ trong khi đó dữ liệu từ cảm biến độ ẩm có sự thay đổi nhẹ mỗi lần đo kể cả khi không có sự tác động của con người.

# KẾT LUẬN CHUNG

**1.KIẾN THỨC LĨNH HỘI**

Trong quá trình thực hiện đề tài “Mô hình kiểm soát không khí trong nhà”, nhóm em đã tích lũy được nhiều kiến thức và kinh nghiệm thực tiễn, đặc biệt thông qua việc sử dụng vi điều khiển ESP32 kết hợp với cảm biến nhiệt độ DHT11 và cảm biến khí gas MQ-2.

Nhóm đã được thực hành kết nối các thiết bị phần cứng, cấu hình truyền dữ liệu lên nền tảng Blynk IoT và ThingSpeak, cũng như xử lý dữ liệu cảm biến để hiển thị và cảnh báo theo thời gian thực. Bên cạnh đó, việc tích hợp ESP32 – một vi điều khiển mạnh mẽ hỗ trợ Wi-Fi và Bluetooth – giúp nhóm tiếp cận được với công nghệ kết nối không dây và ứng dụng IoT hiện đại.

Ngoài ra, nhóm cũng vận dụng được các kiến thức từ các môn học như: Vi xử lý, Hệ thống nhúng, Giao tiếp truyền thông và Thực hành điện tử cơ bản. Từ đó, kỹ năng làm mạch, lập trình và xử lý tín hiệu cảm biến của nhóm được củng cố rõ rệt.

Không chỉ về mặt kỹ thuật, việc triển khai dự án theo nhóm còn giúp từng thành viên rèn luyện kỹ năng giao tiếp, phối hợp công việc, quản lý tiến độ và giải quyết vấn đề phát sinh trong quá trình thực hiện.

**2.TỔNG KẾT QUÁ TRÌNH THỬ NGHIỆM VÀ SO SÁNH VỚI SẲN PHẨM THƯƠNG MẠI**

Quá trình thử nghiệm hệ thống:

**Ưu điểm:**

- Hệ thống vận hành ổn định, đáp ứng đầy đủ các yêu cầu kỹ thuật và phù hợp với phạm vi nghiên cứu của đề tài.

- Các thành phần linh kiện hoạt động trơn tru, không gặp tình trạng quá tải hoặc nhiễu trong quá trình sử dụng kéo dài.

- Kết quả thử nghiệm cho thấy hệ thống cung cấp dữ liệu chính xác, tự động kiểm soát nhiệt độ thông qua thiết bị ngoại vi một cách hiệu quả.

**Nhược điểm:**

- Dải đo nhiệt độ còn hạn chế, không phù hợp với môi trường có nhiệt độ quá cao hoặc quá thấp.

- Hệ thống chỉ thích hợp cho các không gian có quy mô vừa và nhỏ như hộ gia đình, văn phòng, phòng học,...

- Độ bền của hệ thống chưa cao, chủ yếu do một số hạn chế về kỹ thuật và kinh nghiệm chế tạo.

Với những kết quả đạt được, nhóm nghiên cứu tự tin rằng đề tài “Thiết kế mô hình kiểm soát nhiệt độ trong nhà” đã phần nào đáp ứng được nhu cầu thực tế của người dùng trong việc theo dõi và kiểm soát nhiệt độ một cách hiệu quả.

**So sánh với sản phẩm thương mại**

Ảnh có chứa văn bản, đồng hồ, số, Phông chữ

Nội dung do AI tạo ra có thể không chính xác.

Hình 3.2 Hình ảnh nhiệt ẩm kế thương mại và thông số kỹ thuật

Khi so sánh đến độ chính xác và dải nhiệt độ thì có vẻ sản phẩm của nhóm nhỉnh hơn nhưng điều đó là không đáng kể và không đem lại quá nhiều giá trị trong môi trường thực tế. Trái lại sự hoàn thiện và kích thước sản phẩm của nhóm chưa thể so sánh được với sản phẩm thương mại. Và điểm quan trọng nhất là giá thành của sản phẩm thực tế trên chỉ khoảng 73 nghìn đồng, rẻ hơn rất nhiều so với chi phí để hoàn thành sản phẩm của nhóm.

**3. HƯỚNG PHÁT TRIỂN**

Dựa vào kết quả hiện tại, nhóm đề xuất một số định hướng cải tiến như sau:

* **Sử dụng cảm biến có độ chính xác cao hơn** như DHT22 hoặc cảm biến môi trường đa năng (BME280).
* **Tích hợp chức năng cảnh báo tự động**: nếu nhiệt độ vượt ngưỡng hoặc phát hiện khí gas, hệ thống sẽ tự động bật còi báo hoặc gửi thông báo qua điện thoại.
* **Chuyển sang nền tảng IoT mở rộng** như Firebase, Node-RED, hoặc MQTT broker để dễ dàng tùy chỉnh hệ thống lớn hơn.
* **Thêm tính năng điều khiển từ xa:** như bật quạt hoặc đèn khi nhiệt độ vượt ngưỡng bằng công tắc ảo trên ứng dụng.
* **Ứng dụng trong thực tế**: mở rộng để giám sát tại phòng học, nhà kho, phòng máy,... nơi cần theo dõi nhiệt độ và khí gas liên tục.

**4. BÀI HỌC KIỂM NGHIỆM**

Qua quá trình thực hiện đồ án, nhóm đã rút ra được một số bài học kinh nghiệm quý báu:

* **Phân công rõ ràng công việc** giúp tiết kiệm thời gian và tránh trùng lặp thao tác.
* **Không đặt cảm biến gần nguồn nhiệt hoặc động cơ**, tránh gây sai lệch số liệu do nhiễu từ môi trường.
* **Không dùng chân GPIO không phù hợp trên ESP32,** vì một số chân không hỗ trợ đầy đủ chức năng hoặc có điện áp đặc biệt.

Những điều nêu trên là tổng hợp từ những vấn đề gặp phải trong suốt quá trình làm đồ án, đó là những kinh nghiệm quý giá đối với từng thành viên nhóm chúng em. Những kinh nghiệm này góp phần giúp chúng em hoàn thiện hơn trong những dự án sau này.

# TÀI LIỆU THAM KHẢO

[1] Điện Tử Việt, *Giới thiệu ESP32*, [Online]. Available at: [https://dientuviet.com/gioi-thieu-esp32/](https://dientuviet.com/gioi-thieu-esp32/" \t "_new) [Accessed 3 Jul. 2025].

[2] Aosong, *DHT11 Temperature & Humidity Sensor – Technical Data Sheet (Translated Version)*, [Online PDF]. Available at: [https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf](https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf" \t "_new) [Accessed 3 Jul. 2025].

[3] Mecsu, *Cảm biến khí MQ2 – nguyên lý và ứng dụng*, [Online]. Available at: [https://mecsu.vn/ho-tro-ky-thuat/cam-bien-khi-mq2.61z](https://mecsu.vn/ho-tro-ky-thuat/cam-bien-khi-mq2.61z" \t "_new) [Accessed 3 Jul. 2025].

[4] Hanwei Electronics, *MQ-2 Gas Sensor Datasheet*, [Online PDF]. Available at: [https://www.pololu.com/file/0j309/mq2.pdf](https://www.pololu.com/file/0j309/mq2.pdf" \t "_new) [Accessed 3 Jul. 2025].

[5] Phong, N.X., (2020). *Hệ thống nhúng và ứng dụng*. Hà Nội: Nhà Xuất Bản Khoa Học và Kỹ Thuật.

[6] ArduinoKit.vn, *Hướng dẫn đọc cảm biến DHT11 với Arduino*, [Online]. Available at: [https://arduinokit.vn/doc-cam-bien-nhiet-do-do-am-dht11-arduino/](https://arduinokit.vn/doc-cam-bien-nhiet-do-do-am-dht11-arduino/" \t "_new) [Accessed 3 Jul. 2025]

PHỤ LỤC HƯỚNG DẪN SỬ DỤNG SẢN PHẨM

Bước 1: Cấp nguồn cho mô hình. Có thể cấp nguồn qua 2 đường màu đỏ (+) và màu xanh(-) trên board test hoặc cấp nguồn qua cổng type C trên ESP32.

Ảnh có chứa Kỹ thuật điện, đồ điện tử, Dây điện, dây cáp

Nội dung do AI tạo ra có thể không chính xác.

Hình 0.1 Mô hình hoàn thiện

Sau khi cấp nguồn màn hình oled sẽ hiển thị thông số về nhiệt độ, độ ẩm và nồng độ khí ga đo được từ các cảm biến

Bước 2. Dùng bật lửa thổi khí ga vào cảm biến MQ-2. Khi này nồng độ khí ga hiển thị trên màn hình oled sẽ tăng lên đến khi vượt ngưỡng 1000 thì động cơ servo quay, led sáng màu đỏ và kết hợp với bật còi báo hiệu.

Bước 3. Tương tự bước 2, có thể dùng máy sấy để tăng nhiệt độ đọc được từ cảm biến DHT11. Khi vượt ngưỡng 40 độ C thì động cơ servo quay, led sáng màu đỏ và kết hợp với bật còi báo hiệu.