=

|  |
| --- |
| **TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**  KHOA ĐIỆN-ĐIỆN TỬ  BỘ MÔN KỸ THUẬT ĐIỆN TỬ <http://dee.utc.edu.vn/> |

A blue and yellow logo

Description automatically generated

**BÁO CÁO MÔN HỌC ĐỒ ÁN THIẾT KẾ**

**Thiết kế và thi công mạch IoT sử dụng ESP32,**

**định hướng ứng dụng trong nông nghiệp thông minh**

**Giảng viên hướng dẫn: PGS.TS. Đào Thanh Toản**

**Sinh viên thực hiện :** **Nguyễn Tiến Hải MSV:201403932**

**Lương Hoàng Quyền 201404073**

**Lớp : Kỹ thuật điện tử và Tin học công nghiệp 2**

**Khoá : K61**

HÀ NỘI, 2024

MỤC LỤC

[**DANH MỤC BẢNG BIỂU** 1](#_Toc167957342)

[**CHƯƠNG 1. TỔNG QUAN** 3](#_Toc167957343)

[**1.1** **Tổng quan** 3](#_Toc167957344)

[**1.2** **Các vấn đề** 5](#_Toc167957345)

[**1.3 Mục tiêu của đồ án thiết kế** 8](#_Toc167957346)

[**1.4** **Kết luận chương 1** 9](#_Toc167957347)

[**Chương 2. THIẾT KẾ THI CÔNG MẠCH** 10](#_Toc167957348)

[**2.1 Thiết kế tổng quan** 10](#_Toc167957349)

[**2.1.1 Phần mềm sử dụng** 10](#_Toc167957350)

[**2.1.2 Giới thiệu Blynk** 11](#_Toc167957351)

[**2.1.3 Tổng quan thiết kế** 13](#_Toc167957352)

[**2.2 Thiết kế chi tiết** 14](#_Toc167957353)

[**2.2.1 Lựa chọn linh kiện** 14](#_Toc167957354)

[**2.2.2 Tính toán** 20](#_Toc167957355)

[**2.2.3 Sơ đồ nguyên lý** 23](#_Toc167957356)

[**2.2.4 Thuật toán** 23](#_Toc167957357)

[**2.2.5 Chương trình code** 24](#_Toc167957358)

[**2.3 Thi công** 28](#_Toc167957359)

[**2.3.1 Đặt mạch in** 28](#_Toc167957360)

[**2.3.2 Test linh kiện** 30](#_Toc167957361)

[**2.3.3 Hàn mạch** 33](#_Toc167957362)

[**2.3.4 Thiết kế phần mền** 34](#_Toc167957363)

[**2.3.5 Thiết kế vỏ hộp** 35](#_Toc167957364)

[**2.3.6 Đóng gói sản phẩm** 36](#_Toc167957365)

[**2.4 Sản phẩm hoàn thiện** 36](#_Toc167957366)

[**2.5 Kết luận chương** 39](#_Toc167957367)

[**CHƯƠNG 3. THÍ NGHIỆM VÀ THẢO LUẬN** 40](#_Toc167957368)

[**3.1 Nội dung thử nghiệm** 40](#_Toc167957369)

[**3.2 Kết quả thử nghiệm và đánh giá** 41](#_Toc167957370)

[**3.3 Đánh giá** 42](#_Toc167957371)

# **DANH MỤC BẢNG BIỂU**

**Chương 1:**

[Hình 1.1: Iot 3](#_Toc168178095)

[Hình 1.2: Ứng dụng Iot 4](#_Toc168178096)

[Hình 1.3: Ảnh hệ thống tưới cây tự động 6](#_Toc168178097)

[Hình 1.4: Sơ đồ khối 7](#_Toc168178098)

**Chương** **2:**

[Hình 2.1: Arduino 10](#_Toc168178057)

[Hình 2.2: Giới thiệu Blynk 11](#_Toc168178058)

[Hình 2.4: Giao diện Blynk 12](#_Toc168178059)

[Hình 2.5: Sơ đồ khối 13](#_Toc168178060)

[Hình 2.6: ESP32 14](#_Toc168178061)

[Hình 2.7: Cảm biến độ ẩm đất 16](#_Toc168178062)

[Hình 2.8: Cảm biến DHT11 17](#_Toc168178063)

[Hình 2.9: Màn hình Oled 18](#_Toc168178064)

[Hình 2.10: Máy bơm 18](#_Toc168178065)

[Hình 2.11: Relay 19](#_Toc168178066)

[Hình 2.12: Mạch buck 21](#_Toc168178067)

[Hình 2.13: IC 7805 21](#_Toc168178068)

[Hình 2.14: Dữ liệu cảm biến DHT11 22](#_Toc168178069)

[Hình 2.15: Dữ liệu byte cảm biến DHT11 22](#_Toc168178070)

[Hình 2.16: Sơ đồ nguyên lý 23](#_Toc168178071)

[Hình 2.17: Sơ đồ thuật toán 23](#_Toc168178072)

[Hình 2. 18: Các bước thi công 28](#_Toc168178073)

[Hình 2.19: Mạch nguyên lý 28](#_Toc168178074)

[Hình 2.20: PCB đi dây 29](#_Toc168178075)

[Hình 2.21: Mạch PCB 3D 29](#_Toc168178076)

[Hình 2. 22: Mạch đặt công ty 29](#_Toc168178077)

[Hình 2.23: Sơ đồ phần cứng test oled 30](#_Toc168178078)

[Hình 2.24: Kết quả test oled 30](#_Toc168178079)

[Hình 2.25: Sơ đồ phần cứng test DHT11 31](#_Toc168178080)

[Hình 2.26: Sơ đồ thuật toán test DHT11 31](#_Toc168178081)

[Hình 2.27: Kết quả test DHT11 31](#_Toc168178082)

[Hình 2.28: Sơ đồ phần cứng test cảm biến độ ẩm đất 32](#_Toc168178083)

[Hình 2.29: Sơ đồ thuật toán test cảm biến độ ẩm đất 32](#_Toc168178084)

[Hình 2.30: Kết quả test cảm biến độ ẩm đất 33](#_Toc168178085)

[Hình 2.31: Mạch sau khi hàn 33](#_Toc168178086)

[Hình 2.32: Giao diện web blynk 34](#_Toc168178087)

[Hình 2. 33: Giao diện app blynk 34](#_Toc168178088)

[Hình 2.34: Ảnh mô phỏng nắp hộp 35](#_Toc168178089)

[Hình 2.35: Ảnh mô phỏng hộp 35](#_Toc168178090)

[Hình 2.36: Ảnh hộp 35](#_Toc168178091)

[Hình 2.37: Ảnh đóng gói test sản phẩm 36](#_Toc168178092)

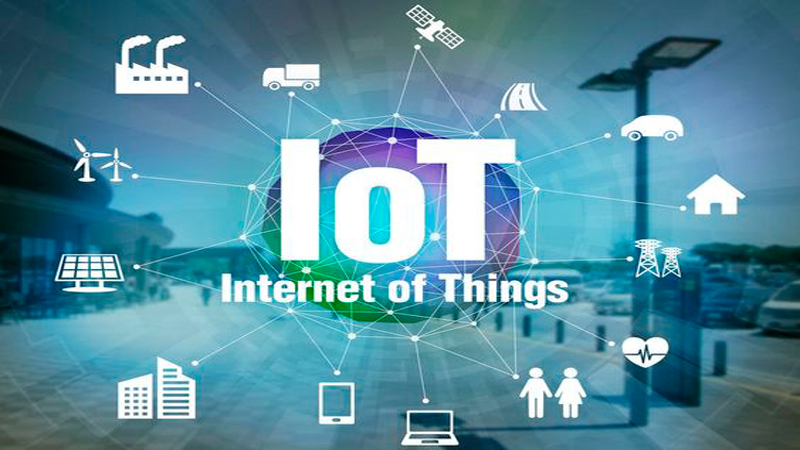
[Hình 2.38: Ảnh sản phẩm 37](#_Toc168178093)

[Hình 2.39: Ảnh ứng dụng phần mềm 38](#_Toc168178094)

# **CHƯƠNG 1. TỔNG QUAN**

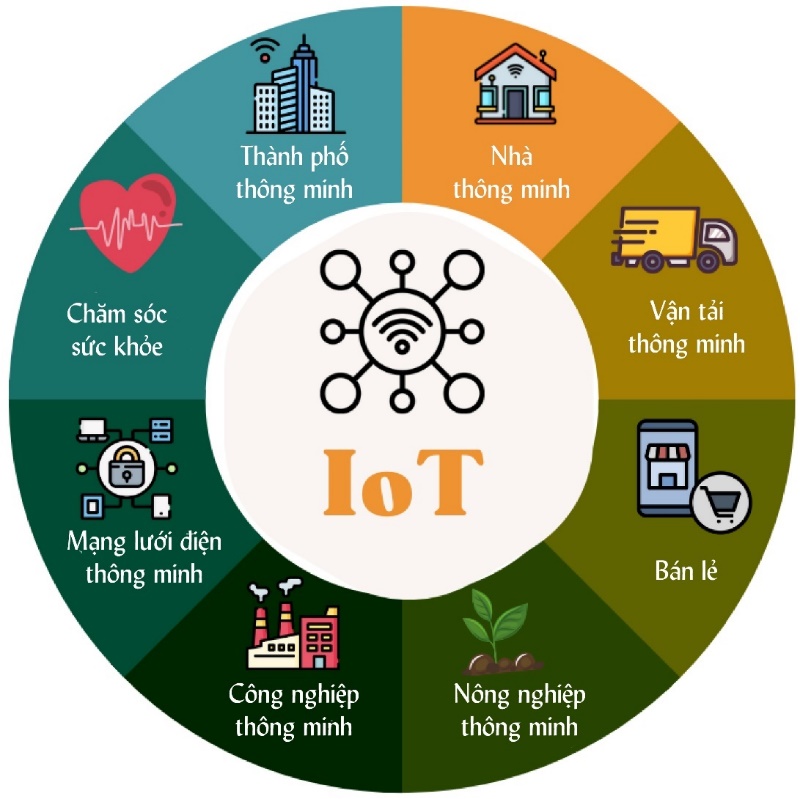
## **Tổng quan**

Thuật ngữ Internet of Things (viết tắt là IoT) được hiểu một cách đơn giản là một mạng lưới vạn vật kết nối với nhau thông qua Internet.



Hình 1.1: Iot

Chúng bao gồm các đồ vật, con người được cung cấp một định danh của riêng mình và tất cả có khả năng truyền tải hay trao đổi thông tin hay dữ liệu qua một mạng duy nhất mà không cần đến sự tương tác trực tiếp giữa người với người, hay người với máy tính. IoT đã phát triển từ sự hội tụ của công nghệ không dây, công nghệ vi cơ điện tử và quan trọng hơn là sự có mặt của Internet. Nói đơn giản IoT là một tập hợp các thiết bị có khả năng kết nối mọi thứ lại với nhau với Internet và với thế giới bên ngoài để thực hiện một công việc nào đó. Có thể nói IoT có mặt trong hầu hết các lĩnh vực đời sống của chúng ta hiện nay từ văn hóa, du lịch, giáo dục, y học, truyền thông, … Tất cả các lĩnh vực trên đều được ứng dụng từ các thành tựu của IoT cả. Như trước đây nếu chúng ta muốn thông tin liên lạc với nhau thì không còn cách nào khác đó là viết thư và truyền thư thông qua người giao nhận. Còn bây giờ thì thông qua các thiết bị thông minh, chúng được kết nối với nhau thông qua hệ thống wifi, 3G hay các ứng dụng riêng của từng hãng. Chưa đầy 10s chúng ta đã có thể gửi tin nhắn đến một người hay nhiều người khác.



Hình 1.2: Ứng dụng Iot

Bên cạnh đó thì còn có nhiều ứng dụng khác như:

* Quản lý môi trường: IoT giúp theo dõi và cải thiện chất lượng môi trường thông qua việc thu thập dữ liệu từ các cảm biến và thiết bị kết nối.
* Quản lý hệ thống máy móc: Công nghiệp 4.0 đã sử dụng IoT để tự động hóa và tối ưu hóa các quy trình sản xuất, từ giám sát hiệu suất máy móc đến dự báo và phòng ngừa sự cố.
* Hệ thống mua sắm trực tuyến: IoT đã biến việc mua sắm trở nên thuận tiện hơn bao giờ hết, từ việc đặt hàng thông qua loa thông minh đến việc tự động lên danh sách mua hàng khi các sản phẩm cạn kiệt.
* Hệ thống kiểm soát an ninh: Các hệ thống an ninh IoT bao gồm camera, cảm biến chuyển động và hệ thống báo động thông minh, giúp tăng cường an ninh cho cả gia đình và doanh nghiệp.
* Nhà thông minh: IoT đã biến các căn nhà trở nên thông minh hơn, từ việc điều chỉnh ánh sáng, nhiệt độ đến việc kiểm soát an ninh và giám sát tiết kiệm năng lượng.
* Giáo dục: Trong lĩnh vực giáo dục, IoT cung cấp cơ hội mới để tạo ra môi trường học tập tương tác và cá nhân hóa hơn bằng cách sử dụng các thiết bị thông minh và cảm biến để theo dõi và đánh giá tiến độ học tập của sinh viên.
* Y tế: Trong lĩnh vực y tế, IoT đóng vai trò quan trọng trong việc cải thiện chăm sóc sức khỏe bằng cách theo dõi sức khỏe cá nhân thông qua các thiết bị đeo thông minh và cảm biến y tế.
* Truyền thông: IoT mở ra cánh cửa cho truyền thông tương tác và cá nhân hóa hơn thông qua việc thu thập dữ liệu từ người tiêu dùng và tạo ra nội dung phù hợp với sở thích và hành vi của họ.
* Văn hóa và giải trí: IoT không chỉ làm thay đổi cách chúng ta sống và làm việc mà còn tác động đến cách chúng ta giải trí.

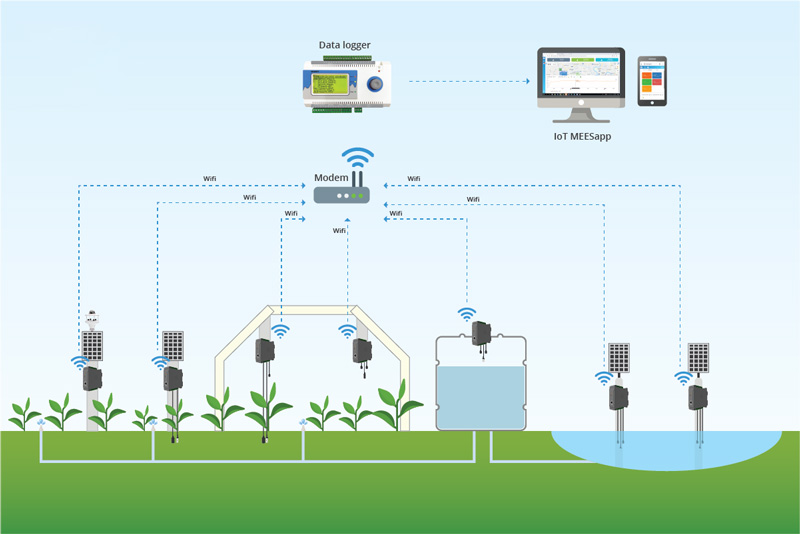
Hệ thống tưới nước tự động kết hợp theo dõi từ xa thông qua điện thoại là hệ thống thiết bị phụ trợ tốt nhất đáp ứng theo nhu cầu sinh trưởng của cây trồng, đang được ứng dụng rộng trên các nước đang phát triển. Hệ thống tưới nước tự động kết hợp theo dõi từ xa là một hình thức tưới nước hợp lý, tiết kiệm sức lao động và chi phí nhân công, vốn đã được phát triển ở nhiều nước trên thế giới. Tuy nhiên, ở Việt Nam chỉ vài ba năm trở lại đây, việc tận dụng hệ thống này mới trở thành xu hướng. Hệ thống tưới nước trên cũng trở nên phổ biến hơn với người nông dân ở nông thôn cùng với quá trình hiện đại hóa, công nghiệp hóa nông thôn nhưng chưa dám mạnh dạng đưa vào sử dụng vì chi phí cao. Nắm bắt được nhu cầu cấp thiết trên và mong muốn được góp chút công sức của mình làm giảm bớt gánh nặng cho người làm nông, nhóm quyết định chọn:

**“Thiết kế và thi công mạch IoT sử dụng ESP32,**

**định hướng ứng dụng trong nông nghiệp thông minh”**

## **Các vấn đề**

Hệ thống tưới nước thông minh không chỉ đơn thuần là một ứng dụng công nghệ, mà còn là biểu tượng của sự kết hợp giữa khoa học, công nghệ và sự bền vững. Cụ thể, việc áp dụng công nghệ IoT vào hệ thống tưới nước có những lợi ích đáng kể như tối ưu hóa sử dụng nước, tăng cường năng suất nông nghiệp, giảm thiểu tác động môi trường, đồng thời mang lại tiện lợi và tự động hóa trong quản lý và điều khiển hệ thống. Bên cạnh đó, khả năng tích hợp và mở rộng của hệ thống cũng mở ra nhiều cơ hội mới để tối ưu hóa quản lý nông nghiệp và thúc đẩy sự phát triển bền vững trong lĩnh vực này.



Hình 1.3: Ảnh hệ thống tưới cây tự động

Trong quá trình phát triển hệ thống tưới cây tự động, việc áp dụng kiến thức từ các môn học như kỹ thuật điện tử, vi xử lý và các lĩnh vực khác không chỉ đóng vai trò quan trọng mà còn là chìa khóa để hiểu và xây dựng các thiết bị và hệ thống hiệu quả. Đầu tiên, kiến thức về kỹ thuật điện tử là cơ sở để hiểu về nguyên lý hoạt động của các linh kiện điện tử như cảm biến, vi điều khiển, và mạch điện. Điều này giúp các nhà phát triển hiểu rõ cách các thành phần của hệ thống tương tác với nhau và hoạt động như thế nào. Kiến thức về vi xử lý là quan trọng để lập trình và điều khiển hệ thống. Vi xử lý đóng vai trò trung tâm trong việc xử lý và phân tích dữ liệu từ các cảm biến, đồng thời ra quyết định về các hành động cụ thể như việc bật/tắt bơm nước hoặc điều chỉnh lượng nước tưới. Hiểu biết sâu về vi xử lý cũng giúp tối ưu hóa hiệu suất và tiêu thụ năng lượng của hệ thống.

Ngoài ra, các lĩnh vực khác như truyền thông không dây, lập trình ứng dụng di động, và phát triển web cũng đóng vai trò quan trọng trong việc tạo ra các giao diện người dùng trực quan và dễ sử dụng cho hệ thống tưới cây tự động. Sự kết hợp của những kiến thức này giúp đảm bảo rằng hệ thống không chỉ hoạt động hiệu quả mà còn dễ dàng sử dụng và quản lý.

A diagram of a computer

Description automatically generated

Hình 1.4: Sơ đồ khối

Tuy nhiên, những người tự học thường phải đối mặt với nhiều thách thức trong việc áp dụng những kiến thức này vào thực tiễn. Một trong những thách thức đầu tiên đó là thiếu hụt kiến thức cơ bản. Điều này có thể là do thiếu hiểu biết về các khái niệm cơ bản của điện tử như điện áp, dòng điện, vàng mạch hoặc thiếu hiểu biết về lập trình vi xử lý và cách hoạt động của các vi điều khiển.

Thêm vào đó, việc thiếu kỹ năng lập trình và kỹ năng kỹ thuật cũng là một thách thức đáng kể. Lập trình các thiết bị nhúng để điều khiển hệ thống tưới cây đòi hỏi kỹ năng lập trình sâu và hiểu biết vững về ngôn ngữ lập trình như C/C++ hoặc Python. Việc sử dụng và giao tiếp với các loại cảm biến cũng đòi hỏi một kiến thức kỹ thuật đáng kể. Đối với những người tự học, việc tiếp cận và thực hành với các công cụ và thiết bị thực tế như cảm biến độ ẩm đất, cảm biến nhiệt độ và module kết nối không dây cũng là một thách thức.

Ngoài ra, hiểu biết về tiêu chuẩn và quy trình phát triển cũng là một vấn đề quan trọng. Thiết kế và phát triển hệ thống tưới cây tự động cần phải tuân thủ các tiêu chuẩn và quy trình phát triển chung để đảm bảo tính ổn định và an toàn cho hệ thống. Tuy nhiên, những người tự học thường thiếu hiểu biết về điều này và gặp khó khăn trong việc áp dụng vào thực tế.

## **1.3 Mục tiêu của đồ án thiết kế**

Mục tiêu nghiên cứu của thiết kế và thi công mạch IoT sử dụng ESP32, định hướng ứng dụng trong nông nghiệp thông minh có thể bao gồm nhiều khía cạnh khác nhau, tùy thuộc vào phạm vi cụ thể của dự án và các vấn đề mà bạn muốn giải quyết. Dưới đây là một số mục tiêu nghiên cứu phổ biến:

* Tối ưu hóa lịch trình tưới nước: nghiên cứu cách tối ưu hóa lịch trình tưới nước dựa trên dữ liệu từ cảm biến độ ẩm đất, cảm biến thời tiết và thông tin về loại cây cảnh cụ thể. Mục tiêu là tối ưu hóa việc sử dụng nước và đảm bảo rằng cây cối nhận đủ nước.
* Phát triển mô hình dự đoán: xây dựng mô hình dự đoán để dự báo nhu cầu nước tưới trong tương lai dựa trên dữ liệu lịch sử và dữ liệu thời tiết. Mục tiêu là cung cấp thông tin dự đoán chính xác để điều chỉnh lịch trình tưới nước.
* Tích hợp các công nghệ IoT mới: nghiên cứu và tích hợp các công nghệ IoT mới như edge computing, machine learning, và blockchain để cải thiện khả năng hiệu quả và an toàn của hệ thống.
* Tương tác người dùng: nghiên cứu cách tương tác và giao tiếp với người dùng một cách hiệu quả thông qua ứng dụng di động, giao diện web hoặc thiết bị khác. Mục tiêu là cung cấp trải nghiệm người dùng tốt nhất và tính linh hoạt cao.
* Nghiên cứu về tiết kiệm năng lượng: tìm kiếm cách tiết kiệm năng lượng trong hệ thống, chẳng hạn như sử dụng nguồn năng lượng tái tạo hoặc quản lý mô-đun tiêu thụ năng lượng.
* An toàn và bảo mật dữ liệu: nghiên cứu và triển khai các biện pháp bảo mật để đảm bảo an toàn của dữ liệu từ cảm biến và thông tin người dùng. Mục tiêu là bảo vệ hệ thống khỏi các mối đe dọa mạng và tấn công.
* Điều khiển từ xa: phát triển tính năng điều khiển từ xa để người dùng có thể quản lý hệ thống từ bất kỳ đâu, bất kỳ khi nào. Mục tiêu là tạo ra tính linh hoạt và tiện lợi cho người sử dụng.
* Kiểm thử thực tế: thực hiện các cuộc thử nghiệm thực tế để đánh giá hiệu suất của hệ thống trong điều kiện thực tế và thu thập dữ liệu thực tế cho việc phát triển và cải tiến.
* Tích hợp với hệ thống khác: nghiên cứu cách tích hợp hệ thống tưới nước thông minh với các hệ thống khác trong ngôi nhà thông minh, chẳng hạn như hệ thống giám sát năng lượng hoặc an ninh.
* Quản lý vận hành và bảo dưỡng: xây dựng tính năng quản lý vận hành và bảo dưỡng để theo dõi trạng thái của các thành phần trong hệ thống và đưa ra thông báo khi cần thực hiện bảo trì.

Bằng cách đặt ra những mục tiêu này và nghiên cứu chúng một cách chi tiết, bạn có thể phát triển một thiết kế và thi công mạch IoT sử dụng ESP32, định hướng ứng dụng trong nông nghiệp thông minh mạnh mẽ và hiệu quả.

## **Kết luận chương 1**

Trong thời đại số hóa ngày nay, Internet of Things (IoT) đang trở thành một trong những xu hướng công nghệ quan trọng nhất, tạo ra sự kết nối thông minh giữa các thiết bị và hệ thống. IoT không chỉ là một công nghệ, mà còn là một triển vọng mới trong việc cải thiện cuộc sống hàng ngày và tạo ra những giải pháp đổi mới trong nhiều lĩnh vực, bao gồm nông nghiệp.

Trong lĩnh vực nông nghiệp, việc sử dụng công nghệ IoT để xây dựng hệ thống tưới cây tự động đã thu hút sự quan tâm của nhiều nhà nghiên cứu và nhà phát triển. Mục tiêu là tận dụng các thiết bị cảm biến và kết nối Internet để tự động hóa quá trình tưới nước, giúp tiết kiệm tài nguyên và tối ưu hóa sản xuất nông nghiệp.

Tuy nhiên, việc xây dựng hệ thống tưới cây tự động không đơn giản và đặt ra nhiều thách thức. Điều này đòi hỏi sự hiểu biết sâu rộng về các nguyên lý của điện tử, vi xử lý, truyền thông không dây và quản lý dữ liệu. Ngoài ra, cần phải xem xét các yếu tố như tiết kiệm năng lượng, bảo mật thông tin và tích hợp với các hệ thống khác.

# **Chương 2. THIẾT KẾ THI CÔNG MẠCH**

## **2.1 Thiết kế tổng quan**

### **2.1.1 Phần mềm sử dụng**

**Arduino IDE** (Integrated Development Environment) - thực hiện việc viết chương trình - là một phần mềm mã nguồn mở được sử dụng để lập trình và nạp code cho các bo mạch Arduino. Nó cung cấp một môi trường phát triển đơn giản và dễ sử dụng cho việc viết và tải lên code vào bo mạch Arduino.



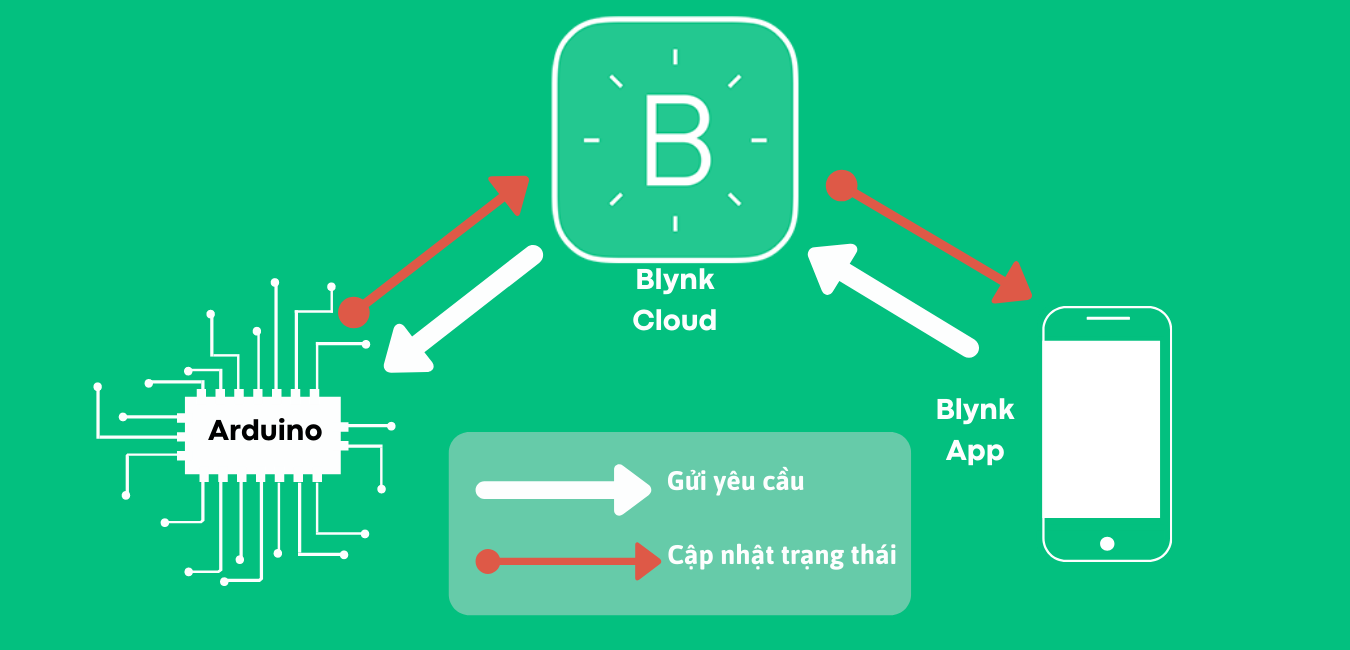
Hình 2.1: Arduino

Dưới đây là một số điểm nổi bật về Arduino IDE:

* Giao diện đồ họa thân thiện: Arduino IDE có giao diện đồ họa đơn giản và thân thiện, giúp người dùng dễ dàng thao tác và lập trình cho bo mạch Arduino.
* Editor code: Arduino IDE đi kèm với một trình soạn thảo code tích hợp, cho phép người dùng viết mã Arduino dễ dàng. Nó hỗ trợ các tính năng như kiểm tra cú pháp, tô màu cú pháp, và gợi ý từ khóa.
* Thư viện và ví dụ: Arduino IDE cung cấp một bộ thư viện phong phú cho các chức năng và module phổ biến, như điều khiển động cơ, giao tiếp không dây, đọc cảm biến, và nhiều hơn nữa. Ngoài ra, nó cũng cung cấp các ví dụ code để người dùng có thể học và áp dụng vào dự án của mình.
* Nạp code dễ dàng: Arduino IDE cho phép người dùng nạp code từ máy tính vào bo mạch Arduino thông qua cổng kết nối USB. Người dùng có thể chọn bo mạch Arduino mục tiêu và cổng COM tương ứng để nạp code.
* Hỗ trợ đa nền tảng: Arduino IDE có sẵn cho nhiều hệ điều hành như Windows, macOS và Linux, giúp người dùng có thể sử dụng trên nhiều nền tảng khác nhau.

### **2.1.2 Giới thiệu Blynk**

Blynk là một nền tảng IoT (Internet of Things) mạnh mẽ được thiết kế để giúp người phát triển tạo ra ứng dụng điều khiển từ xa cho các thiết bị thông minh và các dự án IoT. Blynk mang lại sự linh hoạt và dễ sử dụng cho việc kết nối và điều khiển các thiết bị từ xa thông qua Internet.



Hình 2.2: Giới thiệu Blynk

Chức năng và Đặc điểm Chính của Blynk:

* Giao diện Người dùng Thân thiện: Blynk cung cấp một giao diện người dùng thân thiện và dễ sử dụng, cho phép người dùng tạo ra các bảng điều khiển đa dạng mà không cần kiến thức lập trình sâu.
* Hỗ trợ Đa Nền Tảng: Blynk hỗ trợ nhiều nền tảng, bao gồm iOS, Android và cả dạng web, giúp người dùng có thể kiểm soát thiết bị từ bất kỳ nơi nào và qua bất kỳ thiết bị nào.
* Thư Viện Phong Phú: Blynk cung cấp thư viện mã nguồn mở cho nhiều loại vi điều khiển và các ngôn ngữ lập trình khác nhau, giúp người phát triển tích hợp dễ dàng vào dự án của họ.
* Kết Nối Đám Mây: Blynk sử dụng đám mây để kết nối thiết bị với ứng dụng di động, giúp quản lý và theo dõi các dự án IoT từ xa.
* Phương Tiện Điều Khiển: Blynk cho phép điều khiển các thiết bị IoT bằng cách sử dụng nút, thanh trượt, cảm biến, và nhiều phương tiện điều khiển khác nhau.
* Dự Án và API: Người dùng có thể tạo và quản lý nhiều dự án khác nhau trên Blynk, với khả năng sử dụng API để tích hợp vào các dự án IoT tự chế tạo của họ.
* Blynk.Cloud và Blynk.Server: Blynk cung cấp Blynk.Cloud, một dịch vụ đám mây miễn phí để lưu trữ và quản lý dự án IoT. Ngoài ra, người dùng cũng có thể triển khai Blynk.Server trên các máy chủ riêng để có sự kiểm soát lớn hơn.
* Danh Sách Hỗ Trợ Thiết Bị Rộng Lớn: Blynk hỗ trợ nhiều loại thiết bị và giao tiếp với chúng thông qua các giao thức như Wi-Fi, Ethernet, Bluetooth, và cả 2G/3G.

Blynk là một công cụ mạnh mẽ cho việc phát triển ứng dụng IoT mà không đòi hỏi sự chuyên sâu về lập trình. Điều này làm cho nó trở thành lựa chọn phổ biến đối với cả người mới bắt đầu và những người có kinh nghiệm trong lĩnh vực IoT.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Hình 2.3: Giao diện Blynk

### **2.1.3 Tổng quan thiết kế**

A diagram of a computer

Description automatically generated

Hình 2.4: Sơ đồ khối

* Khối nguồn: Nhiệm vụ chính của khối này là cung cấp nguồn điện cho các thành phần khác trong hệ thống. Nó có thể bao gồm việc chuyển đổi nguồn điện từ nguồn cung cấp ngoại vi thành các mức điện áp hoặc dòng điện cần thiết cho các phần khác của hệ thống.
* Khối cảm biến: Khối này thường chứa các cảm biến đo lường môi trường như nhiệt độ, độ ẩm, ánh sáng, độ rung, áp suất, vv. Nhiệm vụ của khối này là thu thập dữ liệu từ môi trường xung quanh và chuyển đổi nó thành dữ liệu số có thể được xử lý bởi khối vi xử lý.
* Khối hiển thị: Nhiệm vụ của khối này là hiển thị thông tin cho người dùng hoặc cho các thiết bị khác trong hệ thống. Nó có thể bao gồm màn hình LCD, LED hoặc bất kỳ loại hiển thị nào khác phù hợp với ứng dụng cụ thể.
* Khối chấp hành: Khối này thường bao gồm các thiết bị chấp hành như động cơ, van, bóng đèn, vv. Nhiệm vụ của nó là nhận dữ liệu từ khối vi xử lý và thực hiện các hành động cụ thể dựa trên dữ liệu đó. Ví dụ, điều chỉnh van để điều chỉnh lưu lượng nước, hoặc bật/tắt đèn dựa trên điều kiện môi trường.
* Khối vi xử lý ESP32: Đây là "bộ não" của hệ thống, nơi mà toàn bộ logic và quyết định được thực hiện. ESP32 là một vi xử lý mạnh mẽ với khả năng kết nối WiFi và Bluetooth tích hợp, vì vậy nó có thể xử lý dữ liệu từ các cảm biến, quản lý giao tiếp mạng, điều khiển các chấp hành và hiển thị dữ liệu lên các thiết bị ngoại vi. Nó là trung tâm của hệ thống IoT (Internet of Things) và thường chịu trách nhiệm quản lý và điều khiển các khối chức năng khác trong hệ thống.

## **2.2 Thiết kế chi tiết**

### **2.2.1 Lựa chọn linh kiện**

* ESP32:

Là một series các vi điều khiển trên một vi mạch giá rẻ, năng lượng thấp có hỗ trợ WiFi và dual-mode Bluetooth. Dòng ESP32 sử dụng bộ vi xử lý Tensilica Xtensa LX6 ở cả 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. ESP32 được chế tạo và phát triển bởi Espressif Systems, một công ty Trung Quốc có trụ sở tại Thượng Hải, và được sản xuất bởi TSMC bằng cách sử dụng công nghệ 40 nm. ESP32 là sản phẩm kế thừa từ vi điều khiển ESP8266.



Hình 2.5: ESP32

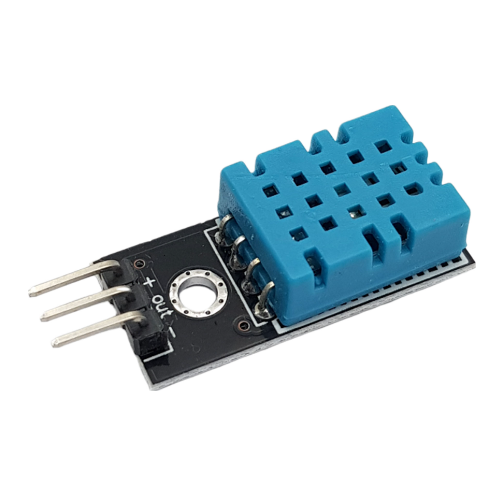
* Bộ xử lý:
* CPU: Bộ vi xử lý Xtensa lõi kép (hoặc lõi đơn) 32-bit LX6, hoạt động ở tần số 240 MHz (160 MHz cho ESP32-S0WD và ESP32-U4WDH) và hoạt động ở tối đa 600 MIPS (200 MIPS với ESP32-S0WD/ESP32-U4WDH)
* Bộ đồng xử lý (co-processor) công suất cực thấp (Ultra low power, viết tắt: ULP) hỗ trợ việc đọc ADC và các ngoại vi khi bộ xử lý chính (main processor) vào chế độ deep sleep.
* Hệ thống xung nhịp: CPU Clock, RTC Clock và Audio PLL Clock.
* Bộ nhớ nội:
* 448 KB bộ nhớ ROM cho việc booting và các tính năng lõi.
* 520 KB bộ nhớ SRAM trên chip cho dữ liệu và tập lệnh.
* Kết nối không dây:
* Wi-Fi: 802.11 b/g/n.
* Bluetooth: v4.2 BR/EDR và BLE (chia sẻ sóng vô tuyến với Wi-Fi)
* 34 GPIO pad vật lý với các ngoại vi:
* SAR 12 bit, 18 kênh.
* DAC 2 × 8-bit.
* 10 cảm biến cảm ứng (touch sensor) (GPIO cảm ứng điện dung).
* 3 SPI (SPI, HSPI và VSPI) hoạt động ở cả 2 chế độ master/slave. Module ESP32 hỗ trợ 4 ngoại vi SPI với SPI0 và SPI1 kết nối đến bộ nhớ flash của ESP32 còn SPI2 và SPI3 tương ứng với HSPI và VSPI.[10] Các GPIO đều có thể được dùng để triển khai HSPI và VSPI.
* 2 I²C, hoạt động được ở cả chế độ master và slave, với chế độ Standard mode (100 Kbit/s) và Fast mode (400 Kbit/s). Hỗ trợ 2 chế độ định địa chỉ là 7-bit và 10-bit. Các GPIO đều có thể được dùng để triển khai I²C.
* 3 UART (UART0, UART1, UART2) với tốc độ lên đến 5 Mbps
* Ethernet MAC interface cho DMA và IEEE 1588 Precision Time Protocol (tạm dịch: Giao thức thời gian chính xác IEEE 1588)
* Bộ điều khiển hồng ngoại từ xa (TX/RX, lên đến 8 kênh)
* PWM cho điều khiển động cơ
* LED PWM (lên đến 16 kênh)
* Cảm biến hiệu ứng Hall
* Bộ tiền khuếch đại analog công suất cực thấp (Ultra low power analog pre-amplifier)
* Bảo mật:
* Hỗ trợ tất cả các tính năng bảo mật chuẩn IEEE 802.11, bao gồm WFA, WPA/WPA2 và WAPI.
* Secure boot (tạm dịch: khởi động an toàn)
* Mã hóa flash 1024-bit OTP, lên đến 768-bit cho khách hàng
* Tăng tốc mã hóa phần cứng: AES, SHA-2, RSA, elliptic curve cryptography (ECC, tạm dịch: mật mã đường cong ellip), random number generator (viết tắt: RNG, tạm dịch: trình tạo số ngẫu nhiên)
* Quản lý năng lượng:
* Hỗ trợ 5 chế độ hoạt động với mức tiêu thụ năng lượng khác nhau: Active, Modem¬-sleep, Light-¬sleep, Deep-¬sleep và Hibernation
* Bộ ổn áp nội với điện áp rơi thấp (internal low-dropout regulator)Individual power domain (tạm dịch: Miền nguồn riêng) cho RTC. Trở lại hoạt động từ ngắt GPIO, timer, đo ADC, ngắt với cảm ứng điện dung
* **Cảm biến độ ẩm đất:**

**A close-up of a device

Description automatically generated**

Hình 2.6: Cảm biến độ ẩm đất

* Điện áp hoạt động: 3.3~5VDC
* Tín hiệu đầu ra:
  + - * Analog: theo điện áp cấp nguồn tương ứng.
      * Digital: High hoặc Low, có thể điều chỉnh độ ẩm mong muốn bằng biến trở thông qua mạch so sánh LM393 tích hợp.
* Kích thước: 3 x 1.6cm.
* **Cảm biến DHT11:**

****

Hình 2.7: Cảm biến DHT11

* Điện áp hoạt động: 5VDC
* Chuẩn giao tiếp: TTL, 1 wire.
* Khoảng đo độ ẩm: 20%-80%RH sai số ± 5%RH
* Khoảng đo nhiệt độ: 0-50°C sai số ± 2°C
* Tần số lấy mẫu tối đa 1Hz (1 giây / lần)
* Kích thước: 28mm x 12mm x10m
* **Màn hình OLED:**

Là một loại màn hình hiển thị điện tử sử dụng các hạt tổ chức hữu cơ để phát sáng khi được kích thích điện. Cụ thể, OLED bao gồm các lớp màng hữu cơ giữa hai điện cực, và khi dòng điện đi qua chúng, các hạt tổ chức phát sáng, tạo nên hình ảnh.

A small blue electronic device

Description automatically generated

Hình 2.8: Màn hình Oled

Màn hình Oled 0.96inch giao tiếp I2C cho khả năng hiển thị đẹp, sang trọng, rõ nét vào ban ngày và khả năng tiết kiệm năng lượng tối đa với mức chi phí phù hợp, màn hình sử dụng giao tiếp I2C cho chất lượng đường truyền ổn định và rất dễ giao tiếp chỉ với 2 chân GPIO.

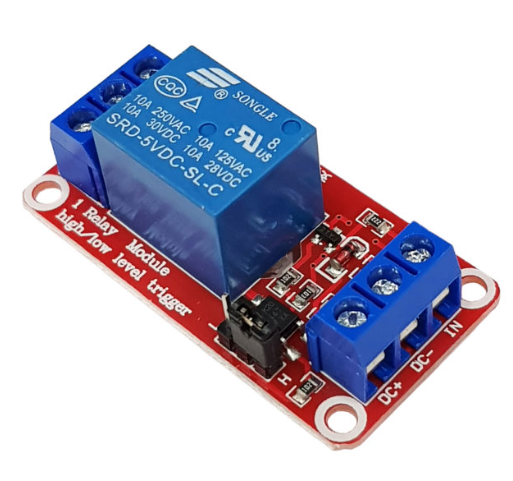
* + Điện áp sử dụng: 2.2-5.5VDC.
  + Công suất tiêu thụ: 0.04w.
  + Góc hiển thị: lớn hơn 160 độ.
  + Số điểm hiển thị: 128x64 điểm.
  + Độ rộng màn hình: 0.96 inch.
  + Màu hiển thị: Xanh Dương.
  + Giao tiếp: I2C.
  + Driver: SSD1306.
  + VCC 2.2-5.5VDC.
  + GND 0VDC.
  + SCL xung Clock.
  + SDA dữ liệu vào Data in.
* **Máy bơm:**

**A small white and black device

Description automatically generated**

Hình 2.9: Máy bơm

* Điện áp: 9V ~ 12V DC
* Công xuất: 12 W
* Áp lực: < 0.1 Mpa
* lượng: 2.0 L/phút
* Ống nước vào ra: 10 mm.
* Tự hút nước cao 1 mét.
* Đẩy cao trên 2 mét
* Đầu bơm công nghệ mới chất lượng.
* Motor lõi đồng tuổi thọ cao.
* Motor 385 12V chất lượng.
* Chất liệu: Đồng - Gang - Nhựa ABS.
* Kích thước: 90 x 40 x 35 mm.
* Trọng lượng:  110 gam.
* Sử dụng nguồn điện ắc quy 12V 1.5Ah trở lên.
* Sử dụng biến thế adapter 12V 2A trở lên.
* **Module relay:**



Hình 2.10: Relay

[Module 1 Relay](https://nshopvn.com/product/module-1-relay-kich-muc-thap-5vdc/) gồm 1 rơ le hoạt động tại điện áp 5VDC, chịu được hiệu điện thế lên đến 250VAC 10A. [Module 1 relay kích mức thấp](https://nshopvn.com/product/module-1-relay-kich-muc-thap-5vdc/) được thiết kế chắc chắn, khả năng cách điện tốt. Trên module đã có sẵn mạch kích relay sử dụng transistor và IC cách ly quang giúp cách ly hoàn toàn mạch điều khiển (vi điều khiển) với rơ le bảo đảm vi điều khiển hoạt động ổn định.

### **2.2.2 Tính toán**

#### **2.2.2.1 Giới thiệu mạch Buck**

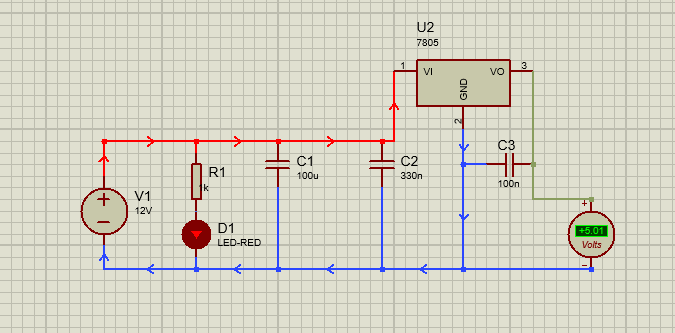
Mạch Buck, hay còn gọi là mạch giảm áp (step-down converter), là một loại mạch điện tử được sử dụng để giảm điện áp từ nguồn cung cấp cao hơn xuống mức độ thấp hơn. Mạch này thường được sử dụng trong các ứng dụng điện tử như nguồn cung cấp điện cho linh kiện điện tử, vi điều khiển, hoặc các thiết bị di động.

Dưới đây là một số đặc điểm chính của mạch Buck:

* Nguyên lý hoạt động:
* Mạch Buck hoạt động dựa trên nguyên lý của cuộn cảm và tụ điện để giảm áp.
* Khi công tắc (transistor) mở, dòng điện chảy qua cuộn cảm, làm tăng năng lượng trong cuộn cảm.
* Khi công tắc đóng, dòng điện không chảy qua cuộn cảm, và năng lượng được truyền tới tụ điện và tải.
* Ưu điểm:
* Hiệu suất cao: Mạch Buck thường có hiệu suất cao hơn so với các mạch giảm áp khác.
* Dải điều chỉnh rộng: Có khả năng điều chỉnh đầu ra để đáp ứng các yêu cầu khác nhau.
* Cấu trúc cơ bản: Mạch Buck bao gồm một cuộn cảm, một transistor công tắc (thường là MOSFET), một tụ điện và một bộ điều khiển để điều chỉnh mức đầu ra.
* Ứng dụng: Mạch Buck thường được sử dụng trong các ứng dụng cần cung cấp nguồn ổn định với áp đầu ra thấp hơn so với nguồn đầu vào, chẳng hạn như trong điện tử tiêu thụ năng lượng như điện thoại di động, laptop, hoặc các thiết bị di động khác.
* Điều khiển và Điều chỉnh: Mạch Buck có thể được kiểm soát thông qua các phương pháp điều khiển dòng, điều khiển áp, hoặc các phương pháp điều khiển khác tùy thuộc vào yêu cầu cụ thể của ứng dụng.
* Hiệu ứng hồi quy đảo (Buck-Boost): Một số mạch Buck có thể được cấu hình để hoạt động như mạch Buck-Boost, có khả năng điều chỉnh áp đầu ra ở cả chế độ giảm áp và tăng áp.

Mạch Buck là một phần quan trọng trong hệ thống điện tử, giúp cung cấp nguồn điện ổn định và hiệu suất cao cho nhiều ứng dụng khác nhau.

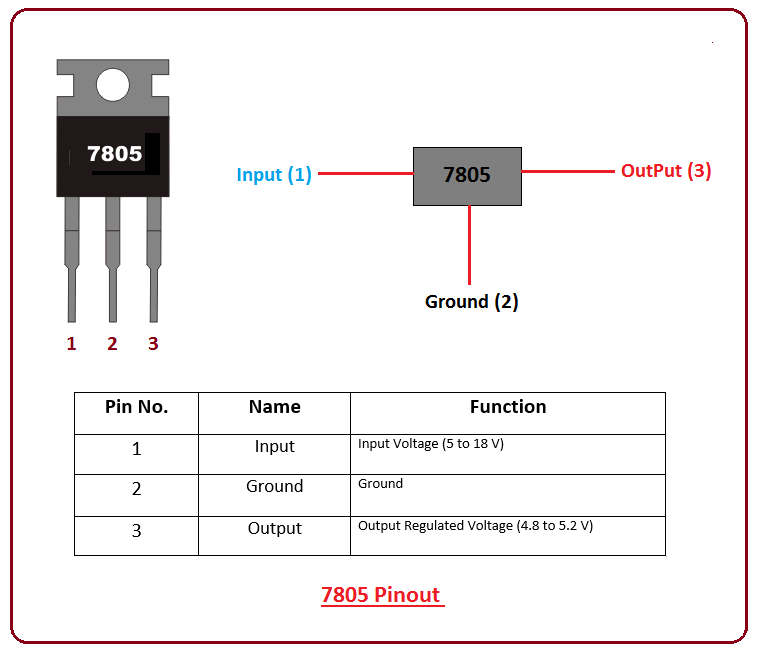
* Mô phỏng mạch buck 12V-5V:



Hình 2.11: Mạch buck

* Nguyên lý hoạt động:

Sử dụng ic LM7805 để điều chỉnh điện áp từ 12V xuống 5V



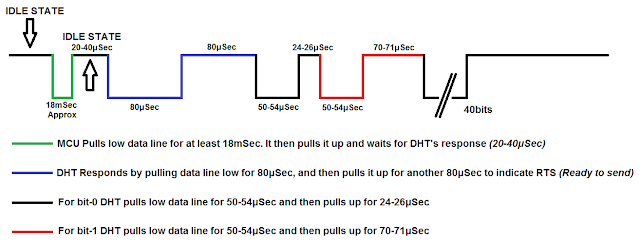
Hình 2.12: IC 7805

*Hình 11: IC LM7805*

Tụ C1 và C2 để lọc điện áp cấp cho tải tiêu thụ lấy từ chân Vo của IC 7805, tụ C2 có các dụng cung cấp điện áp tạm thời cho tải khi điện áp tải đột ngột bị sụt áp, tụ C3 có tác dụng lọc nhiễu điện áp đầu ra (nhiễu là các điện áp không mong muốn làm cho dạng sóng điện áp ngõ ra có hình răng cưa).

#### **2.2.2.2 Cách đọc dữ liệu từ cảm biến DHT11**

Để đọc giá trị nhiệt độ, độ ẩm từ cảm biến DHT11, chúng ta sẽ làm theo sequense sau:

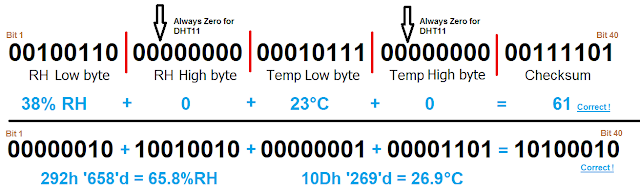


Hình 2.13: Dữ liệu cảm biến DHT11

Sau khi kéo đường truyền (chân Data) xuống mức thấp 1 khoảng thời gian (18ms), đưa đường truyền trở lại mức cao, thì Cảm biến sẽ bắt đầu truyền dữ liệu nhiệt độ và độ ẩm về. Tổng data mà Cảm biến trả về bao gồm 40 bits (5 bytes), trong đó:

* bytes đầu là dữ liệu về nhiệt độ, bao gồm phần nguyên và phần thập phân.
* bytes tiếp theo là dữ liệu độ ẩm, bao gồm phần nguyên và phần thập phân.
* Byte cuối cùng là byte checksum.

Đối với DHT11 thì 2 byte thứ 2 và 4 (Chứa phần thập phân của nhiệt độ/độ ẩm) luôn có giá trị là 0x00, vì vậy chúng ta có thể bỏ qua mà không xử lý.



Hình 2.14: Dữ liệu byte cảm biến DHT11

#### **2.2.2.3 Cách đọc dữ liệu từ cảm biến đất**

Cảm biến nhiệt độ độ ẩm sẽ hoạt động dựa theo nguyên lý hấp thụ hơi nước để biến đổi tính chất, thành phần cảm nhận trong cảm biến từ đó làm cho thiết bị điện trở thay đổi giá trị, xuất hiện sự biến đổi của dòng điện nhờ vậy sẽ xác định, đo lường được độ ẩm thay đổi.

Với thiết bị cảm biến nhiệt độ độ ẩm điện dung thì khi không khí đi qua 2 tấm kim loại thì khi có sự thay đổi độ ẩm không khí sẽ tạo được sự biến đổi điện dung giữa các bản.

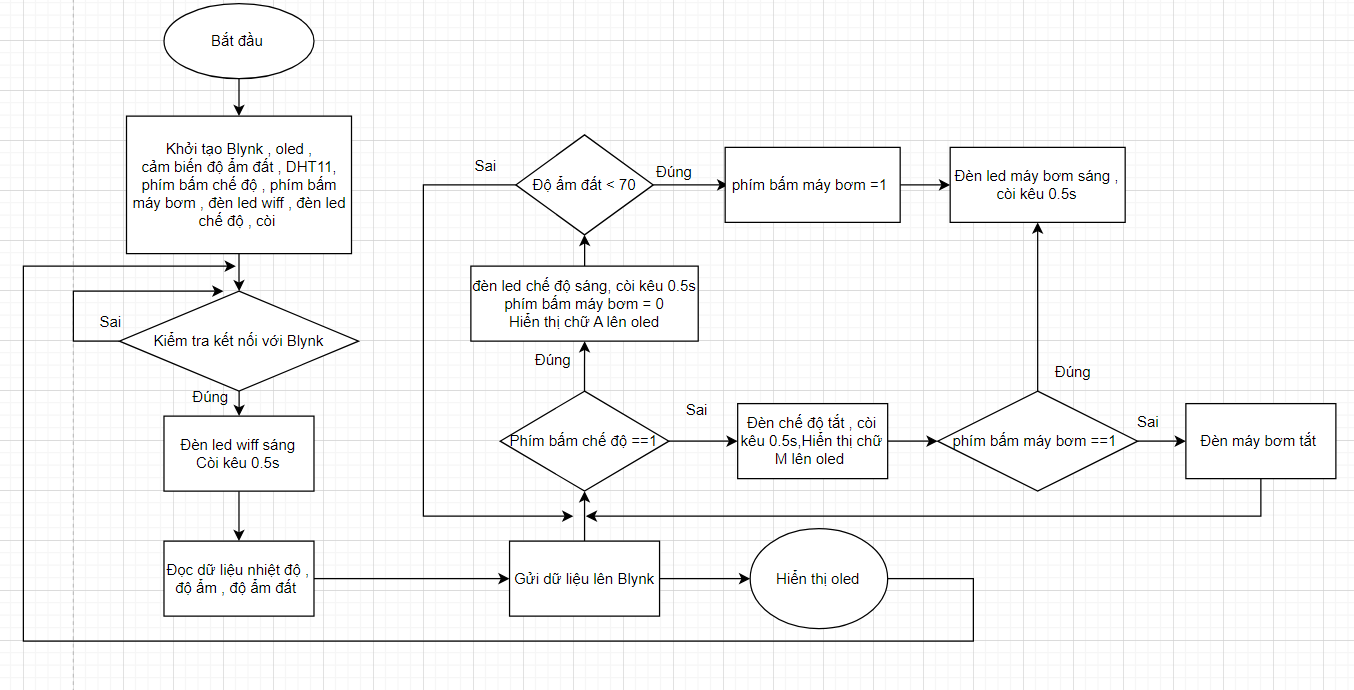
### **2.2.3 Sơ đồ nguyên lý**

A diagram of a circuit

Description automatically generated

Hình 2.15: Sơ đồ nguyên lý

### **2.2.4 Thuật toán**



Hình 2.16: Sơ đồ thuật toán

### **2.2.5 Chương trình code**

|  |  |
| --- | --- |
| 1. #define BLYNK\_TEMPLATE\_ID "TMPL6ujbdk8j1" 2. #define BLYNK\_TEMPLATE\_NAME "doamdat" 3. #define BLYNK\_AUTH\_TOKEN   "nxmFlzQRwtEtsTJ9IOTLceyFFLsGLW7k"   1. char ssid[] = "nguyenhai"; 2. char pass[] = "23456788"; 3. int moistPerLow = 20 ; 4. int moistPerHigh = 80 ; 5. #include <Adafruit\_SSD1306.h> 6. #include <WiFi.h> 7. #include <WiFiClient.h> 8. #include <BlynkSimpleEsp32.h> 9. #include <DHT.h> 10. #include <AceButton.h> 11. using namespace ace\_button; 12. // Define connections to sensor 13. #define SensorPin 34 14. #define DHTPin 14 //D14 15. #define RelayPin 25 //D25 16. #define wifiLed 2 //D2 17. #define RelayButtonPin 32 //D32 18. #define ModeSwitchPin 33 //D33 19. #define BuzzerPin 26 //D26 20. #define ModeLed 15 //D15 21. #define DHTTYPE DHT11 22. #define VPIN\_MoistPer V1 23. #define VPIN\_TEMPERATURE V2 24. #define VPIN\_HUMIDITY V3 25. #define VPIN\_MODE\_SWITCH V4 26. #define VPIN\_RELAY V5 27. #define SCREEN\_WIDTH 128 28. #define SCREEN\_HEIGHT 32 29. #define OLED\_RESET -1 30. Adafruit\_SSD1306display(SCREEN\_WIDTH, SCREEN\_HEIGHT,&Wire,OLED\_RESET); 31. int sensorVal; 32. int moisturePercentage; 33. bool toggleRelay = LOW; 34. bool prevMode = true; 35. int temperature1 = 0; 36. int humidity1 = 0; 37. String currMode = "A"; 38. char auth[] = BLYNK\_AUTH\_TOKEN; 39. ButtonConfig config1; 40. AceButton button1(&config1); 41. ButtonConfig config2; 42. AceButton button2(&config2); 43. void handleEvent1(AceButton\*, uint8\_t, uint8\_t); 44. void handleEvent2(AceButton\*, uint8\_t, uint8\_t); 45. BlynkTimer timer; 46. DHT dht(DHTPin, DHTTYPE); 47. //Kiểm tra kết nối với Blynk 48. void checkBlynkStatus() { // called every 3 seconds by SimpleTimer 49. bool isconnected = Blynk.connected(); 50. if (isconnected == false) { 51. Serial.print("Blynk Not Connected "); 52. digitalWrite(wifiLed, LOW); } 53. if (isconnected == true) { 54. digitalWrite(wifiLed, HIGH); 55. //Serial.println("Blynk Connected"); }} 56. //Đồng bộ hóa dữ liệu với Blynk 57. BLYNK\_CONNECTED() { 58. Blynk.syncVirtual(VPIN\_MoistPer); 59. Blynk.syncVirtual(VPIN\_RELAY); 60. Blynk.syncVirtual(VPIN\_TEMPERATURE); 61. Blynk.syncVirtual(VPIN\_HUMIDITY); 62. //Blynk.syncVirtual(VPIN\_MODE\_SWITCH) 63. Blynk.virtualWrite(VPIN\_MODE\_SWITCH, prevMode); } 64. BLYNK\_WRITE(VPIN\_RELAY) { 65. if(!prevMode){ 66. toggleRelay = param.asInt(); 67. digitalWrite(RelayPin, toggleRelay); } 68. else{ 69. Blynk.virtualWrite(VPIN\_RELAY, toggleRelay); }} 70. BLYNK\_WRITE(VPIN\_MODE\_SWITCH) { 71. if(prevMode != param.asInt()){ 72. prevMode = param.asInt(); 73. currMode = prevMode ? "A" : "M"; 74. digitalWrite(ModeLed, prevMode); 75. controlBuzzer(500); 76. if(!prevMode && toggleRelay == HIGH){ 77. digitalWrite(RelayPin, LOW); 78. toggleRelay = LOW; 79. Blynk.virtualWrite(VPIN\_RELAY, toggleRelay); }}} 80. void controlBuzzer(int duration){ 81. digitalWrite(BuzzerPin, HIGH); 82. delay(duration); 83. digitalWrite(BuzzerPin, LOW); } 84. void displayData(String line1 , String line2){ 85. display.clearDisplay(); 86. display.setTextSize(2); 87. display.setCursor(30,2); 88. display.print(line1); 89. display.setTextSize(1); 90. display.setCursor(1,25); 91. display.print(line2); 92. display.display(); } 93. void getMoisture(){ 94. sensorVal = analogRead(SensorPin); 95. sensorVal = map(sensorVal, 0, 4095, 0, 100); 96. moisturePercentage = 100 - sensorVal; 97. Serial.print("Moisture Percentage: "); 98. Serial.print(moisturePercentage); 99. Serial.println(" %"); 100. delay(100);} 101. void getWeather(){ 102. float h = dht.readHumidity(); 103. float t =dht.readTemperature(); 104. if (isnan(h) || isnan(t)) { 105. Serial.println("Failed to read from DHT sensor!"); 106. return; } | 1. else { 2. humidity1 = int(h); 3. temperature1 = int(t); }} 4. void sendSensor() { 5. getMoisture(); 6. getWeather(); 7. displayData(String(moisturePercentage) + " %", "T:" +String(temperature1) + " C, H:" + String(humidity1) + " % " +currMode); 8. Blynk.virtualWrite(VPIN\_MoistPer, moisturePercentage); 9. Blynk.virtualWrite(VPIN\_TEMPERATURE, temperature1); 10. Blynk.virtualWrite(VPIN\_HUMIDITY, humidity1); } 11. void controlMoist(){ 12. if(prevMode){ 13. if (moisturePercentage < (moistPerLow)){ 14. if (toggleRelay == LOW){ 15. controlBuzzer(500); 16. digitalWrite(RelayPin, HIGH); 17. toggleRelay = HIGH; 18. Blynk.virtualWrite(VPIN\_RELAY, toggleRelay); 19. delay(1000); }} 20. if (moisturePercentage > (moistPerHigh)){ 21. if (toggleRelay == HIGH){ 22. controlBuzzer(500); 23. digitalWrite(RelayPin, LOW); 24. toggleRelay = LOW; 25. Blynk.virtualWrite(VPIN\_RELAY, toggleRelay); 26. delay(1000); }}} 27. else{ 28. button1.check(); }} 29. void setup() { 30. Serial.begin(115200); 31. // Set pinmodes for GPIOs 32. pinMode(RelayPin, OUTPUT); 33. pinMode(wifiLed, OUTPUT); 34. pinMode(ModeLed, OUTPUT); 35. pinMode(BuzzerPin, OUTPUT); 36. pinMode(RelayButtonPin, INPUT\_PULLUP); 37. pinMode(ModeSwitchPin, INPUT\_PULLUP); 38. digitalWrite(wifiLed, LOW); 39. digitalWrite(ModeLed, LOW); 40. digitalWrite(BuzzerPin, LOW); 41. dht.begin(); 42. config1.setEventHandler(button1Handler); 43. config2.setEventHandler(button2Handler); 44. button1.init(RelayButtonPin); 45. button2.init(ModeSwitchPin); 46. if(!display.begin(SSD1306\_SWITCHCAPVCC, 0x3C)) { 47. Serial.println(F("SSD1306 allocation failed")); 48. for(;;); } 49. delay(1000); 50. display.setTextSize(1); 51. display.setTextColor(WHITE); 52. display.clearDisplay(); 53. WiFi.begin(ssid, pass); 54. timer.setInterval(2000L, checkBlynkStatus); 55. timer.setInterval(3000L, sendSensor); 56. Blynk.begin(auth, ssid, pass, "blynk.cloud", 80); 57. //delay(1000); 58. controlBuzzer(1000); 59. digitalWrite(MdeLed,prevMode);} 60. void loop() { 61. Blynk.run(); 62. timer.run(); // Initiates SimpleTimer 63. button2.check(); 64. controlMoist(); } 65. void button1Handler(AceButton\* button, uint8\_t eventType, uint8\_t buttonState) { 66. Serial.println("EVENT1"); 67. switch (eventType) { 68. case AceButton::kEventReleased: 69. //Serial.println("kEventReleased"); 70. digitalWrite(RelayPin, !digitalRead(RelayPin)); 71. toggleRelay = digitalRead(RelayPin); 72. Blynk.virtualWrite(VPIN\_RELAY, toggleRelay); 73. break; }} 74. void button2Handler(AceButton\* button, uint8\_t eventType, uint8\_t buttonState) { 75. Serial.println("EVENT2"); 76. switch (eventType) { 77. case AceButton::kEventReleased: 78. if(prevMode && toggleRelay == HIGH){ 79. digitalWrite(RelayPin, LOW); 80. toggleRelay = LOW; 81. Blynk.virtualWrite(VPIN\_RELAY, toggleRelay); } 82. prevMode = !prevMode; 83. currMode = prevMode ? "A" : "M"; 84. digitalWrite(ModeLed, prevMode); 85. Blynk.virtualWrite(VPIN\_MODE\_SWITCH, prevMode); 86. controlBuzzer(500); 87. break; }} |

## **2.3 Thi công**

A diagram of a diagram

Description automatically generated

Hình 2. 17: Các bước thi công

### **2.3.1 Đặt mạch in**

* Vẽ mạch nguyên lý

A diagram of a circuit

Description automatically generated

Hình 2.18: Mạch nguyên lý

* Thiết kế PCB và đi dây

|  |  |
| --- | --- |
| * Lớp Top Layer   A red circuit board with many different colored lines  Description automatically generated | * Lớp Bottom Layer |

Hình 2.19: PCB đi dây

* Tạo hình 3D

|  |  |
| --- | --- |
|  | A green circuit board with many small circles and dots  Description automatically generated |

Hình 2.20: Mạch PCB 3D

* Mạch đặt công ty

|  |  |
| --- | --- |
| A red circuit board with many small holes  Description automatically generated | A red circuit board with white lines  Description automatically generated |

Hình 2. 21: Mạch đặt công ty

### **2.3.2 Test linh kiện**

* Màn hình oled

|  |  |
| --- | --- |
| * Sơ đồ phần cứng   A circuit board with wires  Description automatically generated  Hình 2.22: Sơ đồ phần cứng test oled | * Code chương trình   A screenshot of a computer program  Description automatically generated |

* Kết quả

A circuit board with wires

Description automatically generated

Hình 2.23: Kết quả test oled

* Cảm biến DHT11

|  |  |
| --- | --- |
| * Sơ đồ phần cứng   A diagram of a circuit board  Description automatically generated  Hình 2.24: Sơ đồ phần cứng test DHT11 | 1. Sơ đồ thuật toán   A diagram of a flowchart  Description automatically generated  Hình 2.25: Sơ đồ thuật toán test DHT11 |

1. Kết quả

A diagram of a circuit board

Description automatically generated

Hình 2.26: Kết quả test DHT11

* Cảm biến độ ẩm đất

1. Sơ đồ phần cứng

A circuit board with wires

Description automatically generated

Hình 2.27: Sơ đồ phần cứng test cảm biến độ ẩm đất

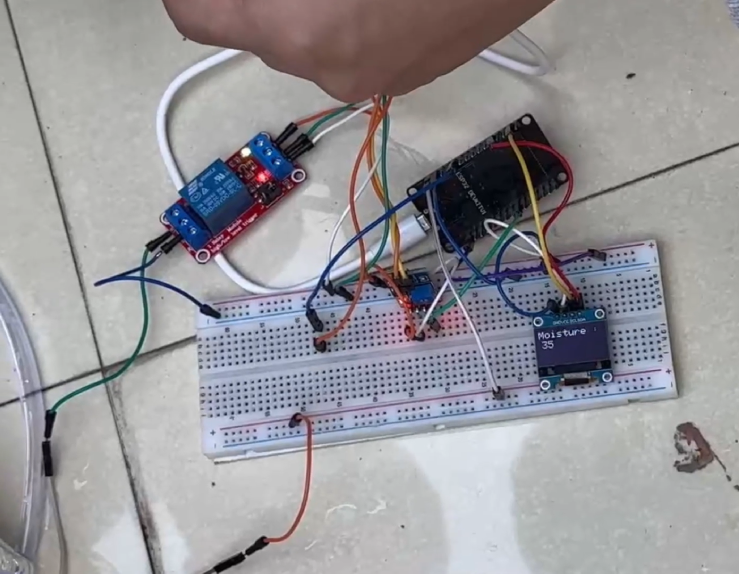
1. Sơ đồ thuật toán

A diagram of a flowchart

Description automatically generated

Hình 2.28: Sơ đồ thuật toán test cảm biến độ ẩm đất

1. Kết quả



Hình 2.29: Kết quả test cảm biến độ ẩm đất

### **2.3.3 Hàn mạch**

|  |  |
| --- | --- |
| * Mặt sau   A red circuit board with screws  Description automatically generated | * Mặt trước   A red circuit board with a chip  Description automatically generated |

Hình 2.30: Mạch sau khi hàn

### **2.3.4 Thiết kế phần mền**

* Giao diện trên web

A screenshot of a computer

Description automatically generated

Hình 2.31: Giao diện web blynk

* Giao diện trên app

A screenshot of a device

Description automatically generated

Hình 2. 32: Giao diện app blynk

### **2.3.5 Thiết kế vỏ hộp**

Hộp có kích thức là 158cm x 90cm x 60cm

|  |  |
| --- | --- |
| Hình ảnh nắp trên khi mô phỏng  A drawing of a rectangular object  Description automatically generated | Hình ảnh nắp dưới khi mô phỏng  A drawing of a rectangular object  Description automatically generated |

Hình 2.33: Ảnh mô phỏng nắp hộp

1. Hình ảnh hộp

|  |  |
| --- | --- |
| A drawing of a rectangular object with screws  Description automatically generated | A drawing of a rectangular object  Description automatically generated |

Hình 2.34: Ảnh mô phỏng hộp

* Hình ảnh thực tế

|  |  |
| --- | --- |
| A rectangular plastic box with clear plastic lid  Description automatically generated | A plastic box with a clear lid  Description automatically generated |

Hình 2.35: Ảnh hộp

### **2.3.6 Đóng gói sản phẩm**

|  |  |
| --- | --- |
| Mở ảnh | Mở ảnh |

Hình 2.36: Ảnh đóng gói test sản phẩm

## **2.4 Sản phẩm hoàn thiện**

* Phần cứng

|  |  |
| --- | --- |
| A transparent box with a circuit board  Description automatically generated |  |
|  |  |

Hình 2.37: Ảnh sản phẩm

* Tham số kĩ thuật sản phẩm
* Chất liệu: Nhựa ABS
* Màu sắc: Trắng Đục, nắp trong
* Chịu nhiệt độ ngưỡng: -20℃ đến 90 ℃
* Có khả năng chống nước theo chuẩn IP65, chống bụi.
* Điệp áp làm việc: 12V
* Trọng lượng sản phẩm: 200g
* Jack tròn DC 5.5-2.1mm
* Công suất: 1A
* Phần mền

Blynk là một nền tảng với các ứng dụng iOS và Android để điều khiển Arduino, Raspberry Pi và các ứng dụng tương tự qua Internet.

Nó là một bảng điều khiển kỹ thuật số nhờ đó bạn có thể xây dựng giao diện đồ họa cho dự án của mình bằng cách kéo và thả các widget.

A screenshot of a cell phone

Description automatically generated

Hình 2.38: Ảnh ứng dụng phần mềm

* Đặc tính
* API và giao diện người dùng tương tự cho tất cả phần cứng và thiết bị được hỗ trợ
* Kết nối với đám mây bằng cách sử dụng:
* Wifi
* Bluetooth và BLE
* Ethernet
* USB (Nối tiếp)
* GSM
* Bộ Widget dễ sử dụng
* Thao tác ghim trực tiếp mà không cần viết mã
* Dễ dàng tích hợp và thêm chức năng mới bằng cách sử dụng ghim ảo
* Theo dõi dữ liệu lịch sử qua tiện ích SuperChart
* Giao tiếp giữa thiết bị với thiết bị sử dụng Bridge Widget
* Gửi email, tweet, push notification...

## **2.5 Kết luận chương**

Chương thiết kế và thi công bao gồm các quá trình sau: quá trình thiết kế tổng quát, thiết kế và thi công mạch IoT sử dụng ESP32, định hướng ứng dụng trong nông nghiệp thông minh. Thiết kế tổng quát tập trung vào việc xác định các yêu cầu chức năng của hệ thống, cùng với việc lập kế hoạch tổ chức và giao tiếp giữa các thành phần. Qua đó, đã xây dựng một cấu trúc tổng thể cho hệ thống.

Đối với thiết kế chi tiết, đã đề xuất các phương pháp và công nghệ cụ thể để triển khai các chức năng đã xác định trong thiết kế tổng quát. Điều này bao gồm việc lựa chọn các cảm biến, thiết bị điều khiển và giao thức kết nối phù hợp để đảm bảo tính linh hoạt và hiệu suất của hệ thống.

Cuối cùng, trong quá trình thi công, thực hiện triển khai thực tế của hệ thống, bao gồm lắp đặt cảm biến, thiết bị điều khiển và hạ tầng mạng cần thiết. Trong quá trình cũng đã thực hiện các bước kiểm tra và điều chỉnh để đảm bảo rằng hệ thống hoạt động ổn định và hiệu quả.

Như vậy, thông qua quá trình thiết kế tổng quát, thiết kế chi tiết và thi công, đã xây dựng thành công một thiết kế và thi công mạch IoT sử dụng ESP32, định hướng ứng dụng trong nông nghiệp thông minh đáng tin cậy và hiệu quả, đáp ứng được các yêu cầu và mong muốn của người dùng.

# **CHƯƠNG 3. THÍ NGHIỆM VÀ THẢO LUẬN**

## **3.1 Nội dung thử nghiệm**

* Hướng dẫn sử dụng
* Cấp nguồn 12V cho sản phẩm
* Thực hiện cắm cảm biến đất xuống vùng cần đo
* Cấp nguồn nước cho máy bơm và đi dây nước tới nơi cần tưới
* Khi đèn led số 1 sản phẩm đã có điện, còi kêu
* Thực hiện kế nối wiff và truy cập vào web hoặc app
* Đèn led số 2 sáng báo hiệu sản phẩm đã có wiff , còi kêu
* Trên màn hình oled sẽ hiển thị thông tin của độ ẩm đất và nhiệt độ như độ ẩm, bên góc dưới bên phải màn hình có chữ A là sản phẩm đang ở chế độ auto còn chữ M là đang ở chế độ bình thường
* Chế độ A (auto) là chế độ tự động khi cảm biến độ ẩm đất dưới 20% thì máy bơm sẽ được kích hoạt và trên 70% máy bớm sẽ tự động ngắt
* Chế độ M là chế độ điều khiển máy bớm thông qua nút bấm
* Nút bấm 1 tương đương nút bấm 1 trên app hoặc web để chuyển chế độ
* Nút bấm 2 thực hiện điều khiển máy bơm khi ở chế độ M
* Một số trường hợp gặp lỗi ở sản phẩm
* Mất kết nối mạng: đôi khi, hệ thống có thể mất kết nối với mạng Wi-Fi, dẫn đến khả năng không thể điều khiển từ xa hoặc không nhận được thông tin từ các cảm biến.
* Cảm biến gặp sự cố: cảm biến độ ẩm đất hoặc cảm biến nước có thể gặp sự cố kỹ thuật, dẫn đến việc không ghi nhận được thông tin chính xác về tình trạng đất hoặc mức nước.
* Thiết bị điều khiển lỗi: thiết bị điều khiển tưới nước có thể gặp sự cố kỹ thuật hoặc phần cứng, dẫn đến việc không thể điều khiển tưới nước theo mong muốn.
* Lỗi phần mềm: có thể xảy ra lỗi phần mềm trên ứng dụng điều khiển, dẫn đến khả năng không thể truy cập hoặc thực hiện các chức năng mong muốn.
* Nguồn nước bị chặn đứng: trong một số trường hợp, nguồn nước có thể bị chặn đứng do nút van bị kẹt hoặc đường ống bị nghẹt, làm gián đoạn quá trình tưới nước.

## **3.2 Kết quả thử nghiệm và đánh giá**

* Kết quả đạt được:

Sau một thời gian tìm hiểu, tham khảo tài liệu từ nhiều nguồn khác nhau, cũng như được sự giúp đỡ của giảng viên bộ môn và bạn bè, nhóm em đã hoàn thành đề tài: “Thiết kế và thi công mạch IoT sử dụng ESP32, định hướng ứng dụng trong nông nghiệp thông minh”

Nhìn chung đề tài có ưu điểm:

* Mạch có cấu tạo và hoạt động đơn giản không phức tạp nhưng tính chính xác và khả năng áp dụng thực tế cao, giảm bớt công sức của con người trong việc chăm sóc vườn.
* Mạch được thiết kế với các module có biến trở, ta có thể dễ dàng điều chỉnh tùy thuộc vào yêu cầu và trường hợp cụ thể.
* Các thiết bị dễ mua được ngoài thị trường, giá thành rẻ.
* Dễ lắp ráp đối với người mới tập làm mạch điện tử.
* Tính linh hoạt và mở rộng: thiết kế và thi công mạch IoT sử dụng ESP32, định hướng ứng dụng trong nông nghiệp thông minh không chỉ giới hạn trong việc quản lý tưới nước cho vườn hoa hay cây trồng nhỏ, mà còn có thể mở rộng để áp dụng cho các hệ thống tưới lớn hơn như vườn rau, trang trại hoặc khu vườn công cộng.
* Tiết kiệm nước và năng lượng: bằng cách sử dụng công nghệ IoT để tự động điều chỉnh việc tưới nước dựa trên điều kiện thực tế của môi trường, hệ thống giúp tiết kiệm lượng nước và năng lượng đáng kể so với phương pháp tưới nước truyền thống, đồng thời giảm thiểu tác động tiêu cực lên môi trường.
* Tăng cường giám sát và điều khiển từ xa: khả năng giám sát và điều khiển hệ thống từ xa thông qua ứng dụng điện thoại thông minh giúp người dùng dễ dàng theo dõi và quản lý việc tưới nước ngay cả khi không có mặt tại hiện trường, tạo ra sự thuận tiện và linh hoạt cao.
* Nâng cao hiệu quả và sản lượng: sử dụng thiết kế và thi công mạch IoT sử dụng ESP32, định hướng ứng dụng trong nông nghiệp thông minh có thể cải thiện hiệu suất sử dụng nước và dinh dưỡng, từ đó tăng cường sản lượng và chất lượng của cây trồng, đồng thời giảm thiểu rủi ro của việc thiếu nước hoặc quá nhiều nước.
* Khả năng tương tác và học hỏi: hệ thống có thể được phát triển thêm tính năng tương tác với người dùng thông qua trí tuệ nhân tạo (AI), cho phép nó học hỏi từ các dữ liệu môi trường và thói quen sử dụng của người dùng để cung cấp các đề xuất và tối ưu hóa hơn về việc quản lý tưới nước.

Đề tài trình bày được những nguyên lý cơ bản nhất của một hệ thống tưới nước tự động và theo dõi từ xa, từ đây có thể phát triển đề tài thành hệ thống vườn thông minh phù hợp với nhu cầu của xã hội.

* Đánh giá, đề xuất hướng phát triển:

Với đề tài này nếu có đủ thời gian nghiên cứu thì có thể mở rộng ra ngoài việc sử dụng cảm biến độ ẩm còn có thể sử dụng thêm các loại cảm biến khác như: cảm biến độ ẩm, cảm biến nhiệt độ, cảm biến tốc độ động cơ để điều chỉnh lượng nước tưới tiết kiệm hiệu quả, cảm biến đo độ pH để đo độ pH trong nước tưới, từ đó kịp thời xử lý nếu nước bị nhiễm phèn.

## **3.3 Đánh giá**

* Đánh giá kết quả:
* Hiệu quả và tính chính xác: thiết kế và thi công mạch IoT sử dụng ESP32, định hướng ứng dụng trong nông nghiệp thông minh đã chứng minh được tính chính xác và hiệu quả trong việc điều khiển tưới nước dựa trên các thông số môi trường như độ ẩm đất, nhiệt độ và mức nước. Sự linh hoạt và mở rộng của hệ thống cũng là một ưu điểm lớn, cho phép nó được áp dụng trong nhiều bối cảnh khác nhau.
* Tính tiện ích và sử dụng: sự dễ dàng lắp đặt, điều chỉnh và sử dụng của hệ thống là một điểm sáng, đặc biệt là với những người mới tiếp xúc với mạch điện tử và công nghệ IoT. Khả năng tương tác và điều khiển từ xa qua ứng dụng cũng tăng cường tính tiện ích và sự linh hoạt cho người dùng.
* Tiết kiệm và bền vững: hệ thống không chỉ giúp tiết kiệm nước và năng lượng mà còn đóng góp tích cực vào việc bảo vệ môi trường và tài nguyên tự nhiên. Điều này phản ánh tính bền vững và tính đúng đắn của dự án trong việc giải quyết các vấn đề liên quan đến nước và môi trường.
* Đánh giá đề tài:
* Ưu điểm: đề tài đã thành công trong việc trình bày và thực hiện một thiết kế và thi công mạch IoT sử dụng ESP32, định hướng ứng dụng trong nông nghiệp thông minh với các ưu điểm như tính linh hoạt, tính chính xác, và tính tiện ích cao. Các nguyên tắc cơ bản của hệ thống IoT và các phương pháp thiết kế mạch điện tử đã được trình bày một cách rõ ràng và dễ hiểu.
* Hạn chế và khuyết điểm: mặc dù có những ưu điểm rõ ràng, đề tài cũng có thể cải thiện bằng cách mở rộng phạm vi nghiên cứu và ứng dụng để đáp ứng các yêu cầu và thách thức phức tạp hơn trong lĩnh vực tưới nước và quản lý môi trường.
* Khả năng phát triển: đề tài cung cấp nền tảng vững chắc cho các nghiên cứu và ứng dụng tiếp theo trong lĩnh vực IoT và nông nghiệp thông minh. Khả năng mở rộng và phát triển của hệ thống cũng mở ra nhiều cơ hội cho việc tạo ra các sản phẩm và dịch vụ mới trong tương lai.

Tóm lại, đề tài đã đạt được những kết quả tích cực và mang lại giá trị trong việc áp dụng công nghệ IoT vào lĩnh vực quản lý tưới nước, đồng thời cung cấp một cơ sở cho sự phát triển và nâng cao hiệu quả của các hệ thống tương tự trong tương lai.