TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

──────── \* ───────

ĐỒ ÁN

**TỐT NGHIỆP ĐẠI HỌC**

NGÀNH CÔNG NGHỆ THÔNG TIN

**XÂY DỰNG GIẢI PHÁP ICT ĐỂ THU THẬP VÀ PHÂN TÍCH DỮ LIỆU CHO HỆ THỐNG THỦY CANH**

Sinh viên thực hiện : **Lê Minh Đức**

Lớp AS – K59

Giáo viên hướng dẫn: PGS.TS **Ngô Quỳnh Thu**

HÀ NỘI 5-2019

# PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

1. Thông tin về sinh viên

Họ và tên sinh viên: Lê Minh Đức

Điện thoại liên lạc: 034 860 7613 Email: duclm96@gmail.com

Lớp: AS – K59 Hệ đào tạo: HEDSPI

Đồ án tốt nghiệp được thực hiện tại: Bộ môn Truyền thông và mạng máy tính, Viện CNTT&TT, Đại học Bách Khoa Hà Nội

Thời gian làm ĐATN: Từ ngày 15/01/2019 đến 26/05/2019

2. Mục đích nội dung của ĐATN

Xây dựng giải pháp ICT để thu thập dữ liệu, điều khiển bơm cho cây trồng trong hệ thống thủy canh.

3. Các nhiệm vụ cụ thể của ĐATN :

* Tìm hiểu tổng quan về Internet of Things.
* Tìm hiểu về giao thức MQTT trong Internet of Things.
* Tìm hiểu về kỹ thuật trồng rau thủy canh hồi lưu, đặc tính sinh trưởng và sự ảnh hưởng của nồng độ pH, EC đến sự phát triển của các cây rau ngắn hạn.
* Tìm hiểu về ESP-IDF và hệ điều hành FreeRTOS.
* Xây dựng hệ thống thu thập và điều khiển trên vi điều khiển ESP32.
* Đánh giá hệ thống thông qua thực nghiệm.

4. Lời cam đoan của sinh viên:

Tôi – *Lê Minh Đức* - cam kết ĐATN là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của PGS.TS Ngô Quỳnh Thu*.*

Các kết quả nêu trong ĐATN là trung thực, không phải là sao chép toàn văn của bất kỳ công trình nào khác.

|  |  |
| --- | --- |
|  | *Hà Nội, ngày tháng năm 2019*  Tác giả ĐATN  *Lê Minh Đức* |

5. Xác nhận của giáo viên hướng dẫn về mức độ hoàn thành của ĐATN và cho phép bảo vệ:

|  |  |
| --- | --- |
|  | *Hà Nội, ngày tháng năm 2019*  Giáo viên hướng dẫn  *PGS.TS Ngô Quỳnh Thu* |

# LỜI CẢM ƠN

5 năm đại học là một quãng thời gian đáng nhớ với nhiều những niềm vui và nỗi buồn khác nhau. Nhưng hành trình nào rồi cũng sẽ đến điểm kết thúc, đồ án tốt nghiệp này chính là điểm mốc cuối cho những năm tháng đại học của em. Để có được thành công trong đồ án này, ngoài sự cố gắng của bản thân, em còn được gia đình, thầy cô và bạn bè giúp đỡ rất nhiều.

Trước hết em xin gửi lời cảm ơn đến cô giáo – PGS.TS Ngô Quỳnh Thu và ThS.Phạm Ngọc Hưng, tuy rất bận rộn nhưng thầy cô đã không quản thời gian, nhiệt tình chỉ dẫn và giúp đỡ em hoàn thành đồ án tốt nghiệp.

Em cũng xin gửi lời cảm ơn đến tập thể thầy cô trong trường Đại Học Bách Khoa Hà Nội những người đã tận tình giảng dạy, truyền đạt kiến thức kinh nghiệm cho em trong suốt thời than theo học. Chúc thầy cô luôn mạnh khỏe và thành công trong cuộc sống.

Cuối cùng em xin gửi lời cảm ơn đến gia đình, bạn bè đặc biệt là các bạn trong lab NCT – 901 B1 đã đồng hành cùng em trong suốt thời gian làm đồ án tốt nghiệp. Nếu không có sự động viên giúp đỡ của mọi người trong những lúc khó khăn, em chắc sẽ không thể hoàn thành đồ án tốt thế này.

Mặc dù đã cố gắng hoàn thành và cải thiện đồ án tốt nghiệp tốt nhất có thể, nhưng vì hạn chế của bản thân về kiến thức và kinh nghiệm thực tiễn nên nội dung của báo cáo không thể tránh khỏi những thiếu xót. Em rất mong nhận được ý kiến đóng góp của thầy cô để đề tài của em ngày càng hoàn thiện hơn.

Em xin chân thành cảm ơn!

Love Bách Khoa three thousand!

Hà Nội, ngày tháng năm 2019

*Lê Minh Đức*

*Lớp Việt Nhật A K59*

*Viện CNTT-TT – ĐH Bách Khoa Hà Nội*

TÓM TẮT NỘI DUNG ĐỒ ÁN

Nông nghiệp luôn là ngành kinh tế mũi nhọn của Việt Nam, đóng góp rất nhiều cho sự phát triển của xã hội trong nhiều năm qua. Tuy nhiên, cùng với sự công nghiệp hóa-đô thị hóa nhanh chóng và biến đổi khí hậu, nhiều vấn đề nghiêm trọng đã phát sinh và ảnh hưởng xấu tới nền nông nghiệp của đất nước. Những phương pháp canh tác truyền thống đã bộc lộ nhiều yếu điểm khó có thể giải quyết trong tương lai gần.

Thủy canh là một trong những phương pháp trồng trọt tiến tiến nhất, trong đó dinh dưỡng sẽ được trộn với nước để cung cấp cho cây trong nhà kính. Với những ưu thế to lớn như không cần hoặc ít sử dụng đất, tận dụng tối đa được nguồn tài nguyên nước, giảm thiểu sâu bệnh, …Do vậy sản lượng có thể tăng cao hơn so với các phương pháp trồng trọt cũ.

Tuy nhiên, ở Việt Nam, thủy canh vẫn chưa được sử dụng rộng rãi do giá thành còn đắt, những nơi áp dụng vẫn chủ yếu thiên về giám sát và tự động hóa, chưa có khả năng phân tích và xử lý dữ liệu, vận hành cần có sự can thiệp của chuyên gia nông nghiệp.

Với sự phát triển của nông nghiệp thông minh, áp dụng các giải pháp ICT – thu thập, phân tích dữ liệu và đưa ra các hành động điều khiển ngược lại để đạt được năng suất cao, đang được phát triển cho các hệ thống thủy canh hứa hẹn sẽ tăng năng suất lên nhiều lần, mang lại lợi ích cho người nông dân, sẽ là giải pháp chiến lược cho sự phát triển của ngành nông nghiệp trong tương lai.

Với thực tế đó cùng với sự định hướng và chỉ dẫn của PSG.TS. Ngô Quỳnh Thu, em đã thực hiện đề tài: “Xây dựng giải pháp ICT để thu thập và phân tích dữ liệu cho hệ thống thủy canh” với các đặc tính: giá thành rẻ, thời gian thực, dễ cài đặt, dễ mở rộng, dễ quản lý, có khả năng xác thực và bảo mật,…

Hệ thống được chia làm 3 phân hệ.

* **Phân hệ hạ tầng IoT**
* **Phân hệ quản lý**
* **Phân hệ xử lý dữ liệu**

Trong khuôn khổ phạm vi đồ án, em tập trung vào phân hệ hạ tầng IoT. Để thu thập những thông số điều kiện môi trường, em sử dụng module ESP32 Dev-Kit kết nối với cảm biến đo nhiệt độ độ ẩm, đầu đo pH, đầu đo EC, cảm biến cường độ ánh sáng. Đồng thời kết nối các thành phần điều khiển như hệ thống đèn LED, bơm nhu động, bơm nước để kiểm soát và thay đổi môi trường trồng cây.

Cấu trúc đồ án gồm 2 phần với nội dung như sau:

* Phần 1: Đặt vấn đề, định hướng giải pháp và cơ sở lý thuyết
  + Chương 1: Đặt vấn đề
  + Chương 2: Định hướng giải pháp
  + Chương 3: Cơ sở lý thuyết
* Phần 2: Các kết quả đạt được
  + Chương 1: Phân tích và thiết kế
  + Chương 2: Triển khai hệ thống
* Phần 3: Kết luận và định hướng phát triển

MỤC LỤC

[PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP 2](#_Toc9665457)

[LỜI CẢM ƠN 3](#_Toc9665458)

[TÓM TẮT NỘI DUNG ĐỒ ÁN 4](#_Toc9665459)

[MỤC LỤC 6](#_Toc9665460)

[DANH MỤC HÌNH VẼ 8](#_Toc9665461)

[DANH MỤC CÁC TỪ VIẾT TẮT VÀ THUẬT NGỮ 10](#_Toc9665462)

[PHẦN I: ĐẶT VẤN ĐỀ VÀ ĐỊNH HƯỚNG GIẢI PHÁP 11](#_Toc9665463)

[Chương 1: Đặt vấn đề 11](#_Toc9665464)

[Chương 2: Các vấn đề cần giải quyết và định hướng giải pháp 13](#_Toc9665465)

[2.1 Các vấn đề cần giải quyết 13](#_Toc9665466)

[2.2 Định hướng giải pháp 14](#_Toc9665467)

[Chương 3: Cơ sở lý thuyết 15](#_Toc9665468)

[3.1 Kỹ thuật trồng rau thủy canh 15](#_Toc9665469)

[3.2 Dinh dưỡng và các chỉ số liên quan đến quá trình sinh trưởng của cây 18](#_Toc9665470)

[3.3 Tổng quan về Internet of Things 20](#_Toc9665471)

[3.4 Các công nghệ được sử dụng trong đồ án 20](#_Toc9665472)

[3.4.1 Framework ESP-IDF 20](#_Toc9665473)

[3.4.2 Hệ điều hành FreeRTOS 22](#_Toc9665474)

[3.4.3 Giao thức MQTT 22](#_Toc9665475)

[PHẦN II: CÁC KẾT QUẢ ĐẠT ĐƯỢC 25](#_Toc9665476)

[Chương 1: Phân tích và thiết kế 25](#_Toc9665477)

[1.1 Yêu cầu và nhiệm vụ 25](#_Toc9665478)

[1.2 Thiết kế hệ thống 26](#_Toc9665479)

[Chương 2: Triển khai hệ thống 39](#_Toc9665480)

[2.1 Phần cứng tại module thu thập và điều khiển 39](#_Toc9665481)

[2.2 Môi trường lập trình 45](#_Toc9665482)

[2.3 Triển khai hệ thống **47**](#_Toc9665483)

[Chương 3: Đánh giá kết quả đạt được 55](#_Toc9665484)

[KẾT LUẬN VÀ ĐỊNH HƯỚNG PHÁT TRIỂN 56](#_Toc9665485)

[TÀI LIỆU THAM KHẢO 57](#_Toc9665486)

DANH MỤC HÌNH VẼ

[Hình 1 Sơ đồ mô tả thủy canh 15](#_Toc9507645)

[Hình 2 Giàn trồng rau thủy canh trong các trang trại 16](#_Toc9507646)

[Hình 3 Sơ đồ mô tả thủy canh hồi lưu 17](#_Toc9507647)

[Hình 4 Sơ đồ biểu thị khả năng hấp thụ các chất theo độ pH 19](#_Toc9507648)

[Hình 5 ESP-IDF trong quá trình phát triển ứng dụng cho ESP32 21](#_Toc9507649)

[Hình 6 Kiến trúc và cơ chế hoạt động của MQTT 23](#_Toc9507650)

[Hình 7 Biểu đồ ca sử dụng phía người dùng 26](#_Toc9507651)

[Hình 8 Biểu đồ ca sử dụng phía thiết bị 28](#_Toc9507652)

[Hình 9 Sơ đồ tổng quan của hệ thống 29](#_Toc9507653)

[Hình 10 Giao diện chính setup ESP 30](#_Toc9507654)

[Hình 11 Giao diện truy cập Wifi 31](#_Toc9507655)

[Hình 12 Sơ đồ khối quá trình quản lý Wifi 32](#_Toc9507656)

[Hình 13 Sơ đồ khối quản lý MQTT 33](#_Toc9507657)

[Hình 14 Cấu trúc của command\_queue 36](#_Toc9507658)

[Hình 15 Sơ đồ khối quản lý cảm biến 37](#_Toc9507659)

[Hình 16 Sơ đồ khối quản lý thiết bị điều khiển 37](#_Toc9507660)

[Hình 17 Module Esp32 Dev-Kit 39](#_Toc9507661)

[Hình 18 Module cảm biến DHT22 40](#_Toc9507662)

[Hình 19 Module cảm biến ánh sáng BH1750 41](#_Toc9507663)

[Hình 20 Module cảm biến pH 42](#_Toc9507664)

[Hình 21 Module cảm biến EC 43](#_Toc9507665)

[Hình 22 Module cảm biến DS18B20 43](#_Toc9507666)

[Hình 23 Bơm nhu động 44](#_Toc9507667)

[Hình 24 Giàn đèn LED trong nông nghiệp 45](#_Toc9507668)

[Hình 25 Bơm nước thủy canh 45](#_Toc9507669)

[Hình 26 Hình minh họa Bash shell 46](#_Toc9507670)

[Hình 27 Hệ thống bắt đầu lấy các dữ liệu từ cảm biến 47](#_Toc9507671)

[Hình 28 Bật chế độ AP/STA và mở Http server 48](#_Toc9507672)

[Hình 29 Giao diện Setup ESP 49](#_Toc9507673)

[Hình 30 Giao diện đăng nhập Wifi 50](#_Toc9507674)

[Hình 31 Giao diện nhập User Token 51](#_Toc9507675)

[Hình 32 ESP gửi xác thực 52](#_Toc9507676)

[Hình 33 Kết quả xác thực 52](#_Toc9507677)

[Hình 34 Nhận được lệnh "start\_crop" 53](#_Toc9507678)

[Hình 35 Gửi dữ liệu và keep\_alive 53](#_Toc9507679)

[Hình 36 Các lệnh điều khiển từ server 54](#_Toc9507680)

DANH MỤC CÁC TỪ VIẾT TẮT VÀ THUẬT NGỮ

|  |  |  |
| --- | --- | --- |
| Chữ viết tắt | Viết đầy đủ | Thuật ngữ |
| IoT | Internet of Things |  |
| MQTT | Message Queing Telemetry Transport |  |
| EC | Electro Conductivity |  |
| QoS | Quality of Service |  |
| AP | Access Point |  |
| STA | Station |  |
| M2M | Machine to Machine |  |
| TCP | Transmission Control Protocol |  |
| SSL | Secure Sockets Layer |  |
| TLS | Transport Layer Security |  |
| JSON | JavaScript Object Notation |  |
| ESP-IDF | Espressif IoT Development Framework |  |
| Publish |  | Gửi thông điệp lên broker trong giao thức MQTT |
| Subscribe |  | Đăng ký một topic trong giao thức MQTT |

# 

# PHẦN I: ĐẶT VẤN ĐỀ VÀ ĐỊNH HƯỚNG GIẢI PHÁP

## Chương 1: Đặt vấn đề

Dân số thế giới đang tăng nhanh với tốc độ chóng mặt, nhân loại mất 200 nghìn năm lịch sử để dân số đạt mức 1 tỷ người nhưng chỉ mất 200 năm trở lại đây để đạt mức 8 tỷ người. Thêm vào đó là biến đổi khí hậu và ô nhiễm môi trường đang ảnh hưởng rất lớn tới vấn đề an ninh lương thực. Sản xuất nông nghiệp truyền thống đang đối mặt với nhiều thử thách như:

* Môi trường đất bị ô nhiễm nặng nề do các hoạt động sản xuất công nghiệp và nông nghiệp(do thuốc trừ sâu và các loại thuốc bảo vệ thực vật).
* Đô thị hóa khiến diện tích đất canh tác ngày càng thu hẹp.
* Biến đổi khí hậu khiến các diễn biến thời tiết ngày càng khó lường, các kiểu thời tiết xấu xuất hiện thường xuyên khiến mùa vụ bị phá hỏng.

Đã có nhiều giải pháp để khắc phục phần nào các vấn đề kể trên và một trong số đó là phương pháp nuôi trồng bằng “thủy canh”. Phương pháp này có nhiều ưu điểm vượt trội như khả năng nâng cao sản lượng gấp nhiều lần nông nghiệp truyền thống, không cần sử dụng đất và tránh được việc sử dụng thuốc bảo vệ thực vật bằng việc nuôi trồng cây trực tiếp trong nước đã pha sẵn dinh dưỡng. Tuy nhiên, sản xuất thủy canh ở Việt Nam vẫn còn rất nhiều hạn chế dù đã có mặt ở nước ta từ lâu. Gần đây, kể từ năm 2015, thủy canh bắt đầu được phát triển theo hướng “bình dân hóa”, dưới hình thức những giàn trồng rau nhỏ trên sân thượng các tòa nhà hay khu chung cư.

Những năm gần đây, nông nghiệp thông minh bằng cách đưa vào các giải pháp ICT để thu thập, phân tích và điều khiển để kiểm soát môi trường nuôi trồng đang là một trong những hướng phát triển trọng điểm nhằm giải quyết các thách thức của ngành nông nghiệp: giảm thiểu chi phí, tiết kiệm tài nguyên và gia tăng sản lượng.

Các giải pháp ICT cho thủy canh cũng đã được đưa vào áp dụng ở thị trường Việt Nam. Tiêu biểu có thể kể đến là các sản phẩm của Hachi và Lisado với những giải pháp công nghệ áp dụng như:

* Giám sát các thông số môi trường (nhiệt độ, độ ẩm, ánh sáng, pH, …)
* Đưa cảnh báo khi có thông số môi trường vượt ra khỏi ngưỡng an toàn.
* Cho phép điều khiển bơm, đèn từ xa.

Các giải pháp ứng dụng công nghệ này với các vườn rau mini đã phát huy phần nào việc giải phóng sức lao động, song vẫn có những nhược điểm như :

* Thiếu các quy trình phân tích dữ liệu quan trắc và điều khiển tự động
* Sản lượng/chi phí chưa tối ưu
* Mới chỉ áp dụng cho các khu vườn quy mô nhỏ
* Quan trọng nhất: chưa có khả năng điều khiển dinh dưỡng trong dung dịch nuôi trồng.

Do đó, để cải thiện hiệu quả của “thủy canh” tại Việt Nam, em xin đề xuất các giải pháp cải thiện trong chương sau.

## Chương 2: Các vấn đề cần giải quyết và định hướng giải pháp

**2.1 Các vấn đề cần giải quyết**

Nhằm mục đích cải thiện hiệu quả của các hệ thống thủy canh, các giải pháp phải thỏa mãn các yêu cầu:

* Chi phí thấp: bao gồm cả chi phí ban đầu và chi phí vận hành
* Có khả năng mở rộng: thích hợp với các mô hình từ cỡ nhở đến lớn.
* Ứng dụng công nghệ cao, cho phép người nông dân không có nhiều kinh nghiệm thủy canh vẫn có thể vận hành hệ thống thủy canh của họ một cách dễ dàng
* Sản lượng cao đi kèm với chất lượng cao trong khi giảm thiểu chi phí sản xuất.

Để đáp ứng các yêu cầu trên, trong đồ án này, em xin trình bày một hệ thống thủy canh gồm các phân hệ sau:

* Phân hệ hạ tầng IoT: là cơ sở hạ tầng kết nối đầu cuối, đảm bảo các yếu tố thời gian thực, an toàn, tin cậy. Phân hệ này bao gồm các thiết bị vi xử lý, các cảm biến, các thiết bị điều khiển. Phân hệ này sẽ thực hiện các nhiệm vụ gồm: thu thập dữ liệu từ cảm biến, nhận các lệnh điều khiển và thực hiện cơ chế điều khiển.
* Phân hệ quản lý đăng ký thiết bị, kết nối và lưu trữ dữ liệu: là thành phần quản lý kết nối, quản lý thiết bị và tương tác với người dùng. Nhận và lưu trữ thông tin thu thập từ cảm biến, đưa ra điều khiển đến các thiết bị trên giàn thủy canh.
* Phân hệ xử lý dữ liệu: là thành phần quản lý và phân tích các dữ liệu nhận được từ phân hệ hạ tầng IoT để đưa ra các hành động kiểm soát tối ưu sao cho năng suất là cao nhất và giảm thiểu chi phí.

**Trong đồ án này, em sẽ tập trung vào phân tích và thiết kế cho phân hệ “Hạ tầng IoT”.**

Trong khuôn khổ phân hệ “Hạ tầng IoT”, em xin nêu ra các vấn đề sau cần giải quyết:

* Thu thập được các thông số môi trường cho cây trồng như nhiệt độ, độ ẩm, cường độ ánh sáng, pH và EC của dung dich dinh dưỡng theo một chu kỳ hợp lý.
* Điều khiển thay đổi của các thông số môi trường thông qua hệ thống đèn LED, các bơm nhu động sao cho phù hợp với từng loại cây trồng để cây có thể sinh trưởng tốt nhất.
* Có các tính năng tiện ích hỗ trợ người dùng cuối như giao diện wifi-manager để người dùng có thể cài đặt wifi và hỗ trợ đăng ký thiết bị với tài khoản người dùng trên server.

**2.2 Định hướng giải pháp**

Từ các vấn đề nêu trên, em xin đưa các định hướng giải pháp như sau:

* Sử dụng vi xử lý ESP32, cụ thể là module ESP32-DEV-KIT để giao tiếp với các cảm biến và điều khiển các thiết bị điều khiển.
* Sử dụng cảm biến DHT22 để thu thập các dữ liệu về nhiệt độ và độ ẩm của môi trường trồng cây. Cảm biến cường độ ánh sáng BH1750 để thu thập dữ liệu về cường độ ánh sáng. Cảm biến PH\_meter\_SKU\_SEN0161 và Analog\_EC\_meter\_SKU\_DRF0300 để thu thập dữ liệu pH và EC của dung dịch dinh dưỡng nuôi cây. Cảm biến DS18B20 để thu thập dữ liệu về nhiệt độ nước.
* Sử dụng đèn LED nông nghiệp để cung cấp ánh sáng cho cây quang hợp.
* Bốn máy bơm nhu động để bơm các dung dịch điều chỉnh pH và EC, bốn máy bơm tương ứng lần lượt với các loại dung dịch Part A, Part B, pH up, pH down.
* Lập trình module ESP32-DEV-KIT sử dụng ngôn ngữ C dựa trên framework ESP-IDF, có sử dụng hệ điều hành FreeRTOS để kết nối tới server thông qua wifi và điều khiển các luồng quản lý tác vụ Wifi, MQTT, thu thập dữ liệu từ sensor, điều khiển thiết bị.

Sau khi xác định được định hướng giải pháp cho hệ thống, những cơ sở lý thuyết liên quan sau cần được tìm hiểu:

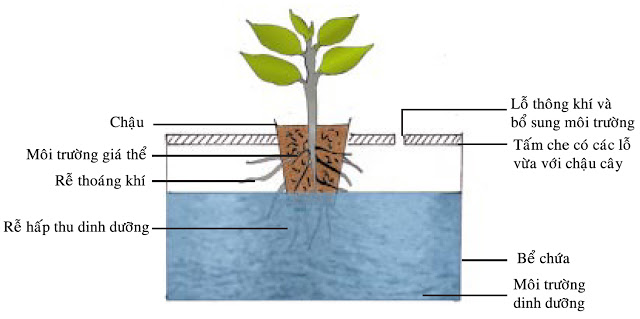
* Nông nghiệp thủy canh
* Dinh dưỡng và các chỉ số liên quan đến quá trình sinh trưởn của cây
* Tổng quan về Internet of Things
* Tổng quan về framework ESP-IDF và hệ điều hành FreeRTOS
* Tổng quan về giao thức MQTT

## Chương 3: Cơ sở lý thuyết

Trong chương này, em xin trình bày về các cơ sở lý thuyết có liên quan đến đồ án. Mục tiêu của đồ án này là xây dựng một hệ thống thủy canh có ứng dụng các giải pháp IoT, do đó, chúng ta sẽ tìm hiểu các kiến thức căn bản về trồng rau thủy canh cũng như các công nghệ đề xuất nhằm giải quyết các vấn đề đã được nêu

### Kỹ thuật trồng rau thủy canh

Điểm khác biệt lớn nhất của thủy canh khi so với các kỹ thuật nông nghiệp truyền thống là trồng cây mà không sử dụng đến đất mà trồng trực tiếp vào môi trường dinh dưỡng hoặc giá thể mà không phải là đất. Các giá thể này có thể là cát, trấu, xơ dừa nén, than bùn, … Kỹ thuật thủy canh là một trong những kỹ thuật nông nghiệp hiện đại nhất hiện nay.



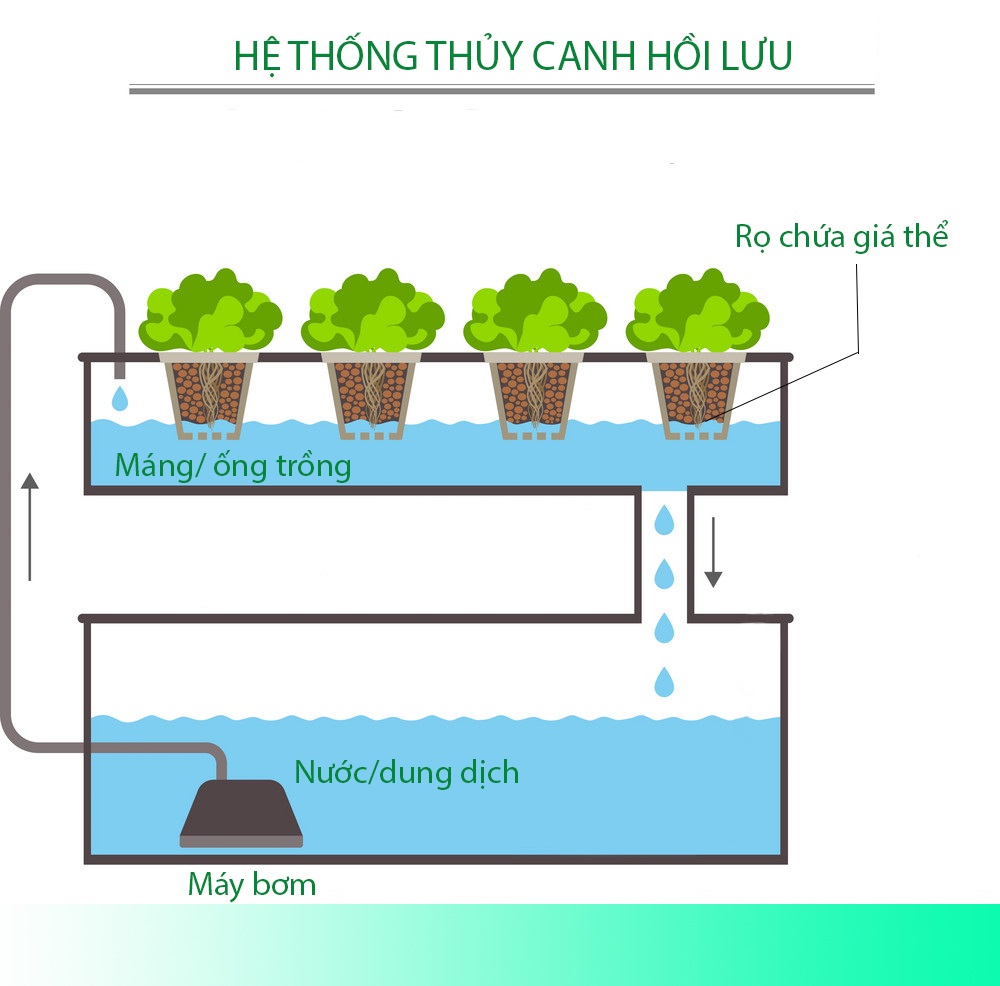
Hình 1 Sơ đồ mô tả thủy canh



Hình 2 Giàn trồng rau thủy canh trong các trang trại

Để cây trồng có thể phát triển theo đúng ý muốn, yêu cầu quan trọng nhất là cung cấp đầy đủ và kịp thời các chất dinh dưỡng cần thiết cho sự sinh trưởng của cây theo từng giai đoạn phát triển. Khi trồng cây bằng đất như truyền thống, một phần nước và chất dinh dưỡng sẽ ngấm sâu xuống đất, ra ngoài tầm với của rễ cây, gây lãng phí và khiến cho việc xác định bao nhiêu chất dinh dưỡng và nước là đủ cho cây rất khó. “Thủy canh” có thể khắc phục tốt vấn đề này vì nước và các chất dinh dưỡng được hòa tan trong nước luôn được giữ lại để tận dụng hiệu quả.

Hiện nay tại Việt Nam có 4 mô hình-biến thể thủy canh chính, đó là: khí canh, tưới nhỏ giọt trên nền giá thể, thủy canh tĩnh và thủy canh hồi lưu. Trong đồ án này, em xin tập trung vào xây dựng các giải pháp IoT cho mô hình thủy canh hồi lưu.



Hình 3 Sơ đồ mô tả thủy canh hồi lưu

Thủy canh hồi lưu có các thành phần chính là:

* Máng/ống trồng dùng để dẫn nước và đựng các giá thể chứa cây trồng, thường được để nghiêng để dung dịch dinh dưỡng có thể chảy về bể dinh dưỡng.
* Thùng chứa nước, dinh dưỡng được pha sẵn trong thùng này, sẵn sàng được bơm lên ống trồng, lượng nước thừa sẽ chạy lại về thùng, tiết kiệm và tận dụng tối đa nước.
* Bơm dung dịch để đưa nước từ thùng lên ống trồng.
* Tùy theo quy mô và thể tích thùng chứa, sẽ có thêm máy sục không khí để đảm bảo lượng oxy trong nước.

Ưu điểm của mô hình này là:

* Nhờ vào sự luân chuyển của dòng dinh dưỡng nên giúp cho cây có khả năng hấp thụ chất dinh dưỡng tốt hơn và trao đổi chất nhanh hơn, từ đó cây có thể phát triển nhanh hơn.
* Không phải làm đất
* Không có cỏ dại
* Trồng được nhiều vụ, có thể trái vụ
* Không cần tưới, tiết kiệm nước.
* Không cần sử dụng thuốc trừ sâu bệnh.
* Tiết kiệm không gian.

Tuy nhiên nó cũng có các nhược điểm:

* Mô hình thủy canh hồi lưu có chi phí đầu tư ban đầu khá lớn.
* Không thích hợp để trồng các loại cây như cây ăn quả lâu năm hay cây ăn củ.

### 3.2 Dinh dưỡng và các chỉ số liên quan đến quá trình sinh trưởng của cây

Dinh dưỡng dùng trong thủy canh bao gồm các vi chất cần thiết cho cây. Dinh dưỡng cho cây bao gồm đủ 14 vi chất: Nitrogen (N), Potassium (K), Phosphorus (P), Calcium (Ca), Magnesium (Mg), Sulphur (S), Iron (Fe), Manganese (Mn), Copper (Cu), Zinc(Zn), Molybdate (Mo), Boron (B), Chlorine (Cl).

Tùy theo nhu cầu về dinh dưỡng của từng loại cây trồng, người trồng có thể pha dinh dưỡng từ bột hoặc có thể mua dung dịch dinh dưỡng được pha sẵn trên thị trường.

Hệ thống thủy canh đang nghiên cứu sử dụng dung dịch dinh dưỡng được pha sẵn trên thị trường. Dung dịch dinh dưỡng này bao gồm hai phần: phần A và phần B. Phần A bao gồm chủ yếu vi chất Nitrogen (N) và Iron (Fe), phần B bao gồm các vi chất còn lại. Vì phần A và phần B thường được pha rất đậm đặc nên để tránh các chất tạo kết tủa nhà sản xuất đã cố tình chia ra làm hai phần như vậy.

**Ngoài thành phần và tỷ lệ các vi chất, 2 chỉ số vô cùng quan trọng tới khả năng hấp thụ của cây trồng là pH và EC.**

**Chỉ số EC** (viết tắt của Electro-Conductivity) là chỉ số diễn tả nồng độ ion hòa tan trong dung dịch. EC có thể được thể hiện bằng một số đơn vị khác nhau nhưng đơn vị tiêu chuẩn được dùng để đo lường EC là millisiemens trên centimet (mS / cm). Chỉ số EC tuy có thể diễn tả được tổng lượng chất có trong dung dịch tuy nhiên lại không thể diễn tả nồng độ của từng chất trong dung dịch đồng thời cũng không thể hiện mức độ cân bằng của các chất dinh dưỡng trong dung dịch.

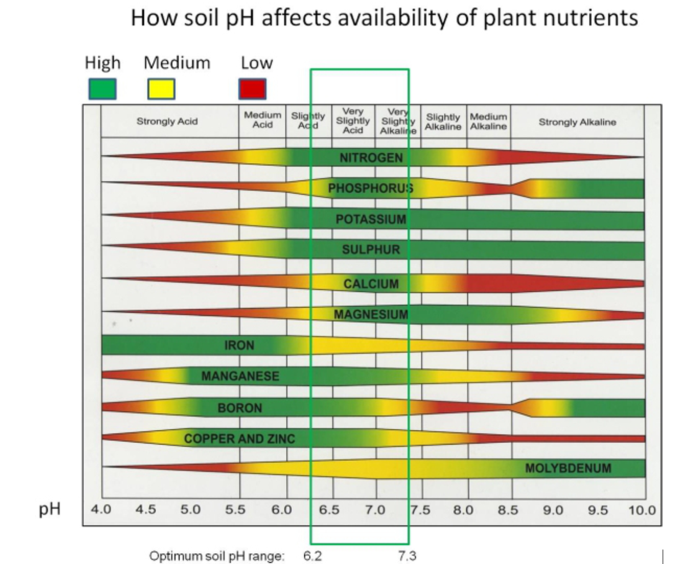
Trong suốt quá trình tăng trưởng, cây hấp thu khoáng chất mà chúng cần. Do vậy duy trì EC ở một mức ổn định là rất quan trọng.

* Nếu dung dịch có chỉ số EC cao thì sự hấp thu nước của cây diễn ra nhanh hơn sự hấp thu khoáng chất. Điều này làm nồng độ dung dịch tăng cao và gây ngộ độc cho cây. Khi đó ta phải bổ sung thêm nước vào môi trường.
* Ngược lại, nếu EC thấp, cây sẽ hấp thu khoáng chất nhanh hơn hấp thu nước. Khi đó, nồng độ dung dịch giảm mạnh, cây sẽ không được cung cấp đầy đủ khoáng chất, chậm lớn và phát triển kém.

**Chỉ số pH** biểu thị nồng độ H+ có trong dung dịch. Chỉ số pH có thang đo từ 0 đến 14, với dung dịch có độ pH dưới 7 là dung dịch mang tính axit, bằng 7 là trung tính và lớn hơn 7 là mang tính bazơ. Chỉ số pH cũng có vai trò rất lớn trong việc quyết định năng suất của cây.

Các chất dinh dưỡng không thể tự đi vào các rễ cây, bởi vì các tế bào thực vật được bảo vệ bởi các màng làm cho các ion hòa tan trong nước khó xâm nhập.

Để vượt qua rào cản này, các chất dinh dưỡng được các chất vận chuyển đặc biệt mang vào bên trong cây. Các chất vận chuyển này là các phân tử protein lớn trôi trong các màng tế bào. Chúng nhận diện các ion của chất dinh dưỡng và cho phép các ion này vào trong các tế bào thực vật. Vì các protein chức năng phụ thuộc vào độ pH. Vì vậy, mỗi protein vận chuyển có phạm vi pH tối ưu để nó hoạt động tốt nhất.



Hình 4 Sơ đồ biểu thị khả năng hấp thụ các chất theo độ pH

Duy trì độ pH càng gần với độ pH “điểm ngọt” của các loài thực vật càng tốt, rất quan trọng để có được một vụ thu hoạch bội thu. Điểm ngọt là phạm vi pH tối ưu, tất cả các chất dinh dưỡng cần thiết cho cây sẵn có để cây hấp thụ. Đối với các cây trồng, điểm ngọt là pH 6.0 đến 7.0. Đây cũng là phạm vi pH thường được đề xuất cho hầu hết các loại cây trồng thủy canh. Việc trồng cây với đất có thể yêu cầu mức pH cao hơn một chút.

### 3.3 Tổng quan về Internet of Things

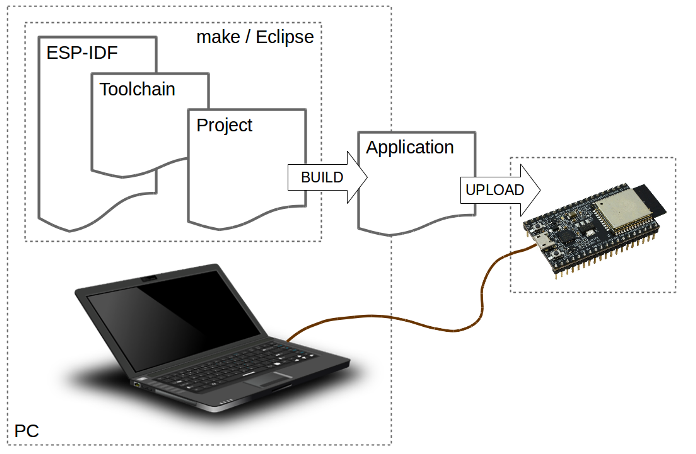
Internet of Things (IoT) là một khái niệm xuất hiện từ thời sơ khai của Internet, khi các nhà phát minh mong muốn kết nối tất cả mọi thứ (Things) thông qua mạng lưới Internet đồng nhất và có thể điều khiển chúng phục vụ mục đích của con người. Trong kịch bản của IoT, tất cả mọi đồ vật đều được cung cấp định danh riêng và có khả năng truyền tải và trao đổi thông tin thông qua mạng lưới duy nhất mà không cần sự tác động của con người hay hệ thống máy tính khác. IoT nói cách khác là tập hợp các thiết bị kết nối với nhau và kết nối với Internet. Việc kết nối có thể sử dụng nhiều công nghệ kết nối khác nhau : băng thông rộng (3G, 4G), bluetooth, ZigBee, hồng ngoại… Các thiết bị có thể là máy tính thông minh, máy café, thiết bị đeo tay thông minh, thiết bị giám sát giao thông … Các thiết bị - Things được cung cấp định danh riêng để xác định từng thiết bị, thường sử dụng địa chỉ IP. Sự xuất hiện của Ipv6 mang lại không gian địa chỉ vô cùng rộng lớn, giúp cho việc định danh dễ dàng hơn và việc kết nối đến mạng Internet cũng dễ dàng hơn. IoT là một cách thực hiện của Ubiquitous Computing (tạm dịch là Tính toán khắp nơi) trong đó máy tính thay đổi để hòa nhập với môi trường sống tự nhiên của con người thay vì con người phải nhập vào thế giới máy móc - một sự phát triển của tương tác giữa người và máy tính sau thời kỳ sử dụng máy tính để bàn.

Các lĩnh vực của IOT vô cùng rộng lớn, có thể kể đến như: Quản lý hạ tầng, y tế thông minh, giao thông thông minh, nông nghiệp thông minh, xây dựng và theo dõi môi trường…

### 3.4 Các công nghệ được sử dụng trong đồ án

#### 3.4.1 Framework ESP-IDF

ESP-IDF, viết tắt của Espressif IoT Development Framework, là một framework hỗ trợ phát triển cho các vi điều khiển họ ESP32. Framework này tập trung vào việc hỗ trợ phát triển cho các ứng dụng IoT, với Wifi, Bluetooth, quản lý năng lượng và nhiều tính năng hệ thống khác.



Hình 5 ESP-IDF trong quá trình phát triển ứng dụng cho ESP32

Một dự án ESP-IDF có thể hỗn hợp nhiều “components”. Ví dụ: với 1 Web Server hiển thị độ ẩm hiện tại, chúng ta cần:

* Thư viện cơ bản cho ESP32 (libc, rom bindings v.v..)
* WiFi drivers
* TCP/IP stack
* Hệ điều hành FreeRTOS
* Một webserver
* Một driver cho cảm biến độ ẩm

ESP-IDF làm cho các “component” rõ ràng và dễ dàng cấu hình. Để làm được điều đó, khi một dự án được biên soạn, môi trường biên dịch sẽ tìm tất cả các thành phần trong thư mục ESP-IDF, các thư mục dự án và thông tin cấu hình của mỗi component. Nó cho phép người sử dụng có thể cấu hình mỗi lần biên dịch sử dụng hệ thống menu dựa trên văn bản thân thiện để tùy chỉnh ESP-IDF cũng như các thành phần khác được yêu cầu bởi dự án. Sau khi các component được tùy biến, tiến trình biên dịch sẽ biên dịch tất cả mọi thứ vào một tập tin cuối cùng, để có thể được nạp vào bảng mạch ESP32.

Một trong những ưu điểm tuyệt vời của ESP-IDF là tích hợp sẵn hệ điều hành FreeRTOS – hệ điều hành chuyên dụng cho các thiết nhúng.

#### 3.4.2 Hệ điều hành FreeRTOS

FreeRTOS là một hệ điều hành nhúng thời gian thực (Real Time Operating System) mã nguồn mở được phát triển bởi Real Time Engineers Ltd, sáng lập và sở hữu bởi Richard Barry. FreeRTOS được thiết kế phù hợp cho nhiều hệ nhúng nhỏ gọn vì nó chỉ triển khai rất ít các chức năng như:

* Cơ chế quản lý bộ nhớ và tác vụ cơ bản
* Các hàm API quan trọng cho cơ chế đồng bộ.

Nó không cung cấp sẵn các giao tiếp mạng, drivers, hay hệ thống quản lý tệp (file system) như những hệ điều hành nhúng cao cấp khác. Tuy vậy, FreeRTOS có nhiều ưu điểm:

* Hỗ trợ nhiều kiến trúc vi điều khiển khác nhau
* Kích thước nhỏ gọn (4.3 Kbytes sau khi biên dịch trên Arduino)
* Được viết bằng ngôn ngữ C và có thể sử dụng, phát triển với nhiều trình biên dịch C khác nhau (GCC, OpenWatcom, Keil, IAR, Eclipse, …)
* Cho phép không giới hạn các tác vụ chạy đồng thời, không hạn chế quyền ưu tiên thực thi, khả năng khai thác phần cứng.

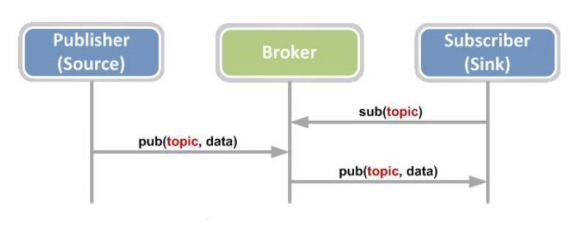
Ngoài ra, nó cũng cho phép triển khai các cơ chế điều độ giữa các tiến trình như: queues, counting semaphore, mutexes.

#### 3.4.3 Giao thức MQTT

MQTT (Message Queing Telemetry Transport) được thiết kế bởi hai kĩ sư Andy Stanford-Clark và Arlen Nipper vào năm 1999. Nó là một giao thức truyền thông điệp theo mô hình publish/subscribe, chuyên được sử dụng cho các thiết bị IoT với băng thông thấp, độ tin cậy cao, và có khả năng hoạt động trong mạng lưới thiếu ổn định. Bởi vì giao thức này có thể sử dụng được băng thông thấp trong môi trường có độ trễ cao nên nó vô cùng lý tưởng cho các ứng dụng M2M và IoT.

MQTT hoạt động theo mô hình client/server, mỗi một nút là một client, server là thành phần trung gian broker, thông qua giao thức TCP/IP.

**Kiến trúc và cơ chế hoạt động của giao thức MQTT:**

**

Hình 6 Kiến trúc và cơ chế hoạt động của MQTT

* Trong hệ thống sử dụng giao thức MQTT, các client kết nối đến một MQTT server (còn được gọi là **broker**). Mỗi client sẽ đăng ký một vài kênh (hay còn gọi là **topic**), ví dụ như “/client1/channel1”, “/client2/channel2”, để lắng nghe. Quá trình này đăng ký này được gọi là **subscribe**.
* Các client đã đăng ký vào một kênh nào đó sẽ nhận được gói tin dữ liệu khi có bất kỳ client nào khác gửi dữ liệu vào đúng kênh đó, quá trình gửi dữ liệu còn được gọi là **publish**.
* Các client cũng có thể hủy đăng ký với kênh đã đăng ký trước đó, quá trình này còn được gọi là **unsubscribe**
* Gói tin: Các dữ liệu được vận chuyển bên trong giao thức MQTT trên mạng cho các ứng dụng. Mỗi một gói tin sẽ có tên **topic** và **QoS**.
* Server/Broker: Là một thành phần trung gian đứng giữa các client để nhận các subscribe từ client và đồng thời publish các gói tin đến các client.

**Quality of service – QoS:**

Quality of Service là một thỏa thuận giữa bên publish và bên subscribe trong việc đảm bảo phân phối gói tin. Có 3 cấp độ QoS trong MQTT:

* At most once (QoS 0): Broker/client sẽ gởi dữ liệu đúng 1 lần, quá trình gởi được xác nhận bởi chỉ giao thức TCP/IP.
* At least once (QoS 1): Broker/client sẽ gởi dữ liệu với ít nhất 1 lần xác nhận từ đầu kia, nghĩa là có thể có nhiều hơn 1 lần xác nhận đã nhận được dữ liệu.
* Exactly once (QoS 2): Broker/client đảm bảm khi gởi dữ liệu thì phía nhận chỉ nhận được đúng 1 lần, quá trình này phải trải qua 4 bước bắt tay.

Một gói tin có thể được gởi ở bất kỳ QoS nào, và các client cũng có thể subscribe với bất kỳ yêu cầu QoS nào. Có nghĩa là client sẽ lựa chọn QoS tối đa mà nó có để nhận tin. Ví dụ, nếu 1 gói dữ liệu được publish với QoS2, và client subscribe với QoS0, thì gói dữ liệu được nhận về client này sẽ được broker gởi với QoS0, và 1 client khác đăng ký cùng kênh này với QoS 2, thì nó sẽ được Broker gởi dữ liệu với QoS2.

Một ví dụ khác, nếu 1 client subscribe với QoS2 và gói dữ liệu gởi vào kênh đó publish với QoS0 thì client đó sẽ được Broker gởi dữ liệu với QoS0. QoS càng cao thì càng đáng tin cậy, đồng thời độ trễ và băng thông đòi hỏi cũng cao hơn.

**Bảo mật trong MQTT**

Bảo mật trong MQTT được chia ra làm nhiều layer, mỗi một layer có một kiểu tấn công khác nhau. MQTT là giao thức gọn nhẹ nên không có nhiều cơ chế bảo mật, trong thực tế thường sử dụng mã hóa SSL/TLS.

* Ở tầng mạng sử dụng một mạng bảo mật vật lý hay VPN như nền tảng cho kết nối giữa Client và Broker là một cách để cung cấp bảo mật và kết nối đáng tin cậy.
* Ở tầng ứng dụng, MQTT cung cấp một xác thực user/password khi một client kết nối đến broker. Một cách khác là sử dụng mã hóa tải trên tầng ứng dụng để thông tin được truyền đi an toàn khi không có mã hóa giao vận đầy đủ.

# PHẦN II: CÁC KẾT QUẢ ĐẠT ĐƯỢC

## Chương 1: Phân tích và thiết kế

### 1.1 Yêu cầu và nhiệm vụ

Yêu cầu của hệ thống:

* Đăng ký thiết bị với tài khoản của người dùng để quản lý và xác thực thiết bị để đảm bảo rằng thông tin gửi về server là chính xác từ thiết bị.
* Hệ thống bao gồm các mạch điều khiển có khả năng điều khiển thiết bị thông qua các chân điều khiển và có khả năng kết nối wifi, gửi nhận dữ liệu giữa bản thân và server
* Hệ thống bao gồm các cảm biến đo những thông số môi trường có khả năng ảnh hưởng đến sự phát triển của cây trồng và gửi dữ liệu về máy chủ xử lý
* Hệ thống có thể đảm bảo khả năng mở rộng ra các mô hình lớn hơn.
* Hệ thống đảm bảo tính thời gian thực.
* Hệ thống có khả năng làm việc tự động, đồng thời cũng có thể điều khiển bằng tay do người dùng phụ trách.

Nhiệm vụ cần thực hiện:

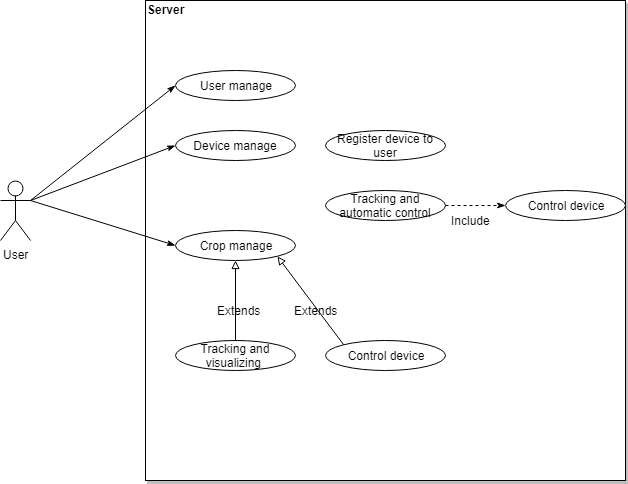
* Xây dựng luồng/tác vụ quản lý kết nối Wifi, thực hiện chuyển đổi giữ các chế độ “**station”** (STA) và “**access point/station”** (AP/STA).
* Xây dựng một Http server trên ESP32 để phục vụ việc đăng ký thiết bị và set-up Wifi.
* Xây dựng luồng/tác vụ thực thi công việc thu thập dữ liệu từ các cảm biến theo chu kỳ được cấp bởi server.
* Xây dựng luồng/tác vụ quản lý kết nối tới MQTT broker, tác vụ này cũng phụ trách việc publish dữ liệu từ các cảm biến lên broker và lắng nghe các lệnh từ server gửi lên broker, định kỳ gửi gói tin keep\_alive để thông báo hoạt động của module ESP32.
* Xây dựng luồng/tác vụ publish gói tin đăng ký thiết bị với user.
* Xây dựng luồng/tác vụ quản lý và giám sát việc thực thi các lệnh nhận được do server gửi.
* Lưu trữ các token phục vụ cho việc quản lý và xác thực.

### 1.2 Thiết kế hệ thống

Hệ thống được thiết kế với hai phần chính:

* Server sử dụng để lưu trữ, hiển thị và phân tích dữ liệu, từ đó đưa ra các hành động cho ESP32.
* Module thu thập và điều khiển với bo mạch ESP32 là trung tâm

Biểu đồ ca sử dụng phía người dùng:

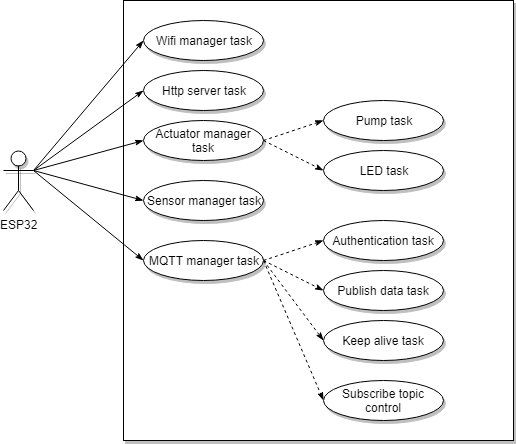


Hình 7 Biểu đồ ca sử dụng phía người dùng

Mô tả :

* Tên ca sử dụng: Hệ thống thủy canh phía server
* Điều kiện đầu vào: Người dùng đã đăng ký tài khoản để sử dụng hệ thống và đăng nhập vào hệ thống bằng tài khoản đăng ký.
* Điều kiện kích hoạt: Người dùng truy cập vào website ứng dụng
* Kịch bản chính:
  1. Người dùng đăng nhập vào hệ thống thông qua tài khoản đã đăng ký.
  2. Người dùng đăng nhập hệ thống có thể quản lý các đối tượng trong hệ thống bao gồm “người dùng”, “thiết bị đầu cuối” – module ESP32, các “mùa vụ”.
  3. Các “thiết bị đầu cuối” có thể được đăng ký thêm hoặc xóa bỏ khỏi tài khoản của người dùng.
  4. Các “mùa vụ” ở đây tương ứng với một giàn hoặc nhiều giàn trồng thủy canh đang được giám sát bởi một “thiết bị đầu cuối”. Người dùng có thể quản lý và xem các dữ liệu của giàn trồng tương ứng với “mùa vụ” và có thể gửi các lệnh điều khiển. Người dùng có thể bắt đầu một mùa vụ mới nếu còn “thiết bị đầu cuối”, đã đăng ký với tài khoản, rảnh rỗi.
  5. Người dùng sẽ quyết định khi nào thì sẽ xóa một “mùa vụ”.
  6. Hệ thống có một luồng nhận bao gồm khả năng phân tích dữ liệu và từ đó tự động điều khiển các thiết bị đựa trên việc so sánh các thông tin có sẵn với thông tin thu thập được.
  7. Mọi dữ liệu thu thập, điều khiển đều được ghi vào CSDL.
* Điều kiện kết thúc: Người dùng đăng xuất khỏi hệ thống.
* Trạng thái sau khi kết thúc: Người dùng trở lại trạng thái trước khi bắt đầu truy cập vào hệ thống.

Biểu đồ ca sử dụng phía “thiết bị”:

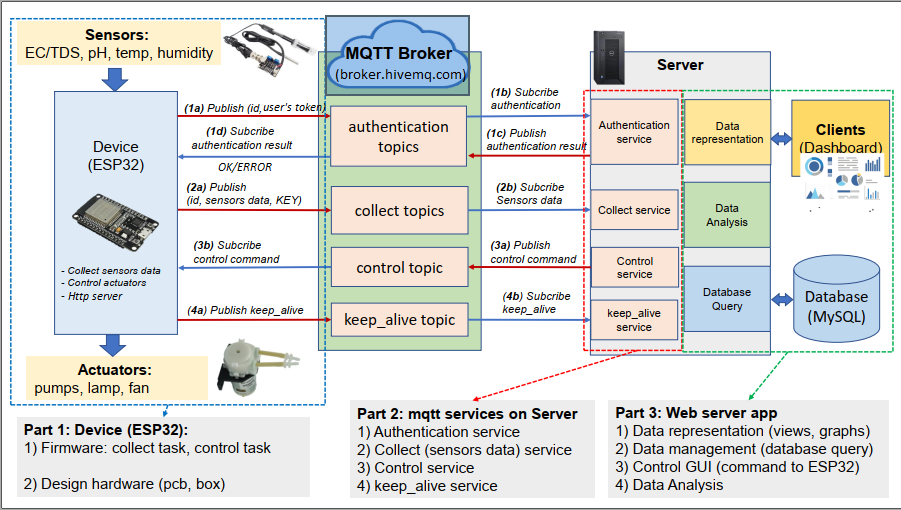


Hình 8 Biểu đồ ca sử dụng phía thiết bị

Mô tả :

* Tên ca sử dụng: Hệ thống thủy canh phía hạ tầng thiết bị IoT
* Điều kiện đầu vào: thiết bị được cấp nguồn và nằm trong khu vực sẵn có mạng wifi.
* Kịch bản chính:
  1. Nếu thiết bị được khỏi động lần đầu, các thông tin của Wifi là SSID và Password đều chưa được setup, người dùng phải bấm nút bật chế độ AP/STA trên mạch để khởi động Http server cho ESP32, thông qua Http server, người dùng có thể setup wifi. Sau khi setup wifi thành công, thiết bị sẽ tự động kết nối wifi trong các lần tiếp theo.
  2. Thiết bị sẽ khởi tạo tác vụ thu thập dữ liệu từ cảm biến.
  3. Thiết bị sẽ khỏi tạo tác vụ thực thi lệnh điều khiển từ server. Khi server gửi đến một lệnh điều khiển, lệnh này sẽ được lưu vào hàng đợi và sẽ được lần lượt lấy ra để thực thi, điều khiển các thiết bị tương ứng như bơm, đèn, ….
  4. Thiết bị kết nối đến MQTT broker. Từ đó các thiết bị sẽ publish và subscribe các topic trên MQTT broker được quy định sẵn.
  5. Thiết bị sau khi nhận được lệnh bắt đầu “mùa vụ” của server sẽ nhận được chu kì theo dõi, từ đó sẽ tiến hành gửi dữ liệu cảm biến theo chu kỳ theo dõi
  6. Ngoài ra, nếu đã bắt đầu “mùa vụ”, thiết bị cứ mỗi phút sẽ gửi cho server một bản tin keep alive để xác nhận với server mình vẫn đang hoạt động.
* Điều kiện kết thúc: thiết bị hỏng, nguồn cấp bị ngắt.
* Trạng thái sau khi kết thúc: Nếu chưa được xác thực, thiết bị sẽ liên tục gửi gói tin xác thực nếu wifi sẵn sàng. Nếu đã được xác thực, nhưng chưa bắt đầu “mùa vụ” nào, thiết bị sẽ vào trạng thái chờ lệnh. Nếu đã bắt đầu mùa vụ, thiết bị sẽ tiếp tục gửi dữ liệu cảm biến lên broker.

Từ hai biểu đồ ca sử dụng trên, ta có sơ đồ tổng quan của hệ thống như sau:



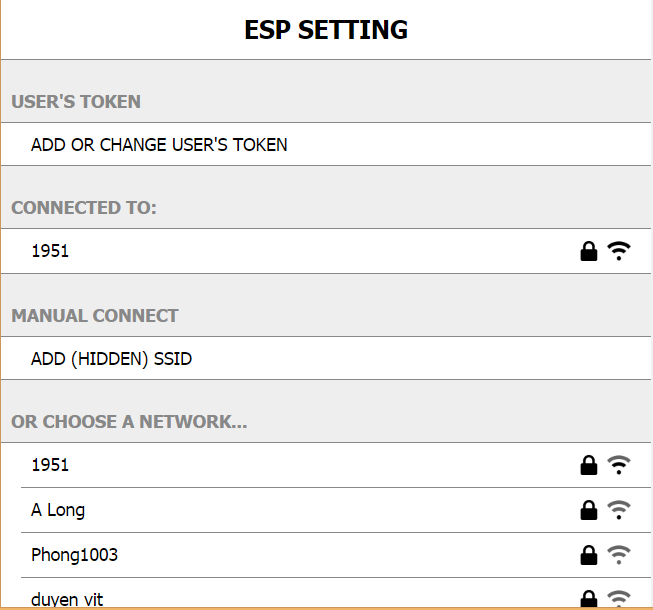
Hình 9 Sơ đồ tổng quan của hệ thống

Chức năng của các khối trong sơ đồ hệ thống:

* Thiết bị (ESP32): là nơi kết nối với các cảm biến, có nhiệm vụ thu thập nhiệt độ, độ ẩm, nồng độ pH của dung dịch dinh dưỡng sau đó gửi về MQTT broker. Đồng thời cũng kết nối với hệ thống đèn LED, máy bơm nhu động, máy bơm nước, nhận lệnh điều khiển từ server để bật tắt các thiết bị này.
* Server: Nhận dữ liệu thu thập từ các thiết bị, lưu trữ xuống cơ sở dữ liệu. Phân tích các dữ liệu này và đưa ra điều khiển đến các thiết bị.
* MQTT Broker: Là máy chủ MQTT, thành phần trung gian nhận subscribe, publish message tới ESP32 từ server và ngược lại.
* Database: Nơi lưu trữ những dữ liệu thu thập được từ các sensor, bản ghi tập lệnh thực hiện trên các thiết bị. Thông tin tài khoản người dùng. Những đặc tính sinh trưởng của từng loại cây cho việc phân tích dữ liệu.

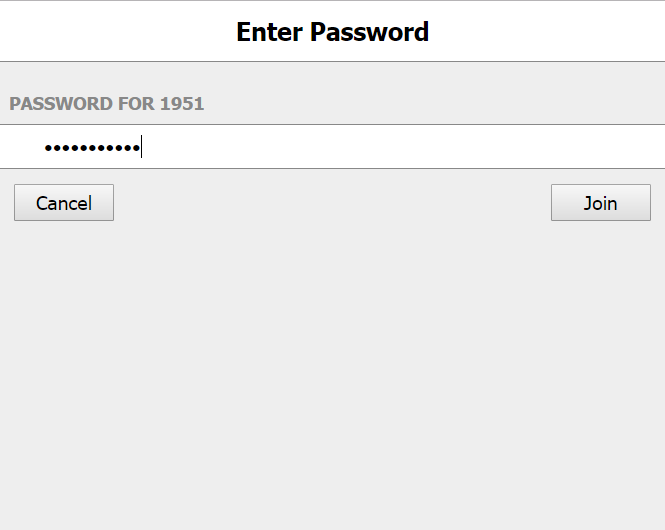
Quy trình hoạt động của hệ thống gồm các bước sau:

* Người dùng đăng ký tài khoản trên server, lúc này người dùng sẽ được cấp một chuỗi bảo mật, lưu giữ thông tin của người dùng, gọi là “User’s Token”.
* Người dùng cấp nguồn cho ESP32, nhấn nút chuyển đổi chế độ STA sang chế độ AP/STA, sử dụng điện thoại hoặc máy tính, kết nối vào mạng Wifi có SSID là “nct\_esp” với Password là “esp32pwd”. Truy cập vào đường link: <http://192.168.1.1>
* Lúc này ta có giao diện như sau:



Hình 10 Giao diện chính setup ESP

* Người dùng có thể nhập “User’s Token” đã được cấp và lựa chọn Wifi để ESP32 có thể truy cập.



Hình 11 Giao diện truy cập Wifi

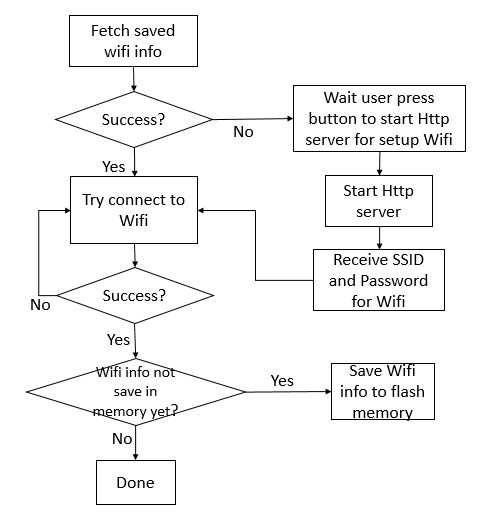
* Thoát chế độ AP/STA bằng cách nhấn nút thêm lần nữa. Lúc này ESP32 sẽ tự động kết nối vào Wifi đã chỉ định và gửi gói tin xác thực có chứa “User’s Token” và “ID” của thiết bị. Server khi nhận được gói tin này sẽ đăng ký thiết bị dựa trên “ID” vào tài khoản của người dùng có “User’s Token” tương ứng.
* Lúc này, trên giao diện web của server, người dùng có thể khởi tạo một “mùa vụ” và gán thiết bị vào “mùa vụ” đó để có thể giám sát và điều khiển.
* Khi “mùa vụ” được khởi tạo, thiết bị sẽ nhận được gói tin điều khiển từ server, qua đó thiết bị sẽ biết “mùa vụ” đã bắt đầu, gói tin này có chứa “chu kỳ gửi”, “Device’s Token”. Lúc này thiết bị sẽ liên tục gửi dữ liệu từ cảm biến theo chu kỳ đã được cho và gói tin “keep\_alive” lên server, các gói tin này đều có chứa “Device’s Token” phục vụ cho việc bảo mật. Đồng thời lắng nghe các gói tin điều khiển từ server.
* Người dùng sẽ quyết định khi nào thì mùa vụ kết thúc bằng cách hủy mùa vụ thông qua giao diện của server. Lúc này thiết bị sẽ ngừng gửi dữ liệu từ cảm biến và vào chế độ chờ.

Như vậy, dựa trên quy trình vận hành và chức năng đã nêu, ta có thể chia chương trình của thiết bị thành các khối sau:

* Khối quản lý kết nối Wifi
* Khối quản lý MQTT
* Khối quản lý cảm biến
* Khối quản lý thiết bị điều khiển (bơm, đèn, …)

**Các khối được chạy ở các luồng riêng, song song và độc lập với nhau, giao tiếp với nhau thông qua các biến trạng thái và bộ nhớ flash.**

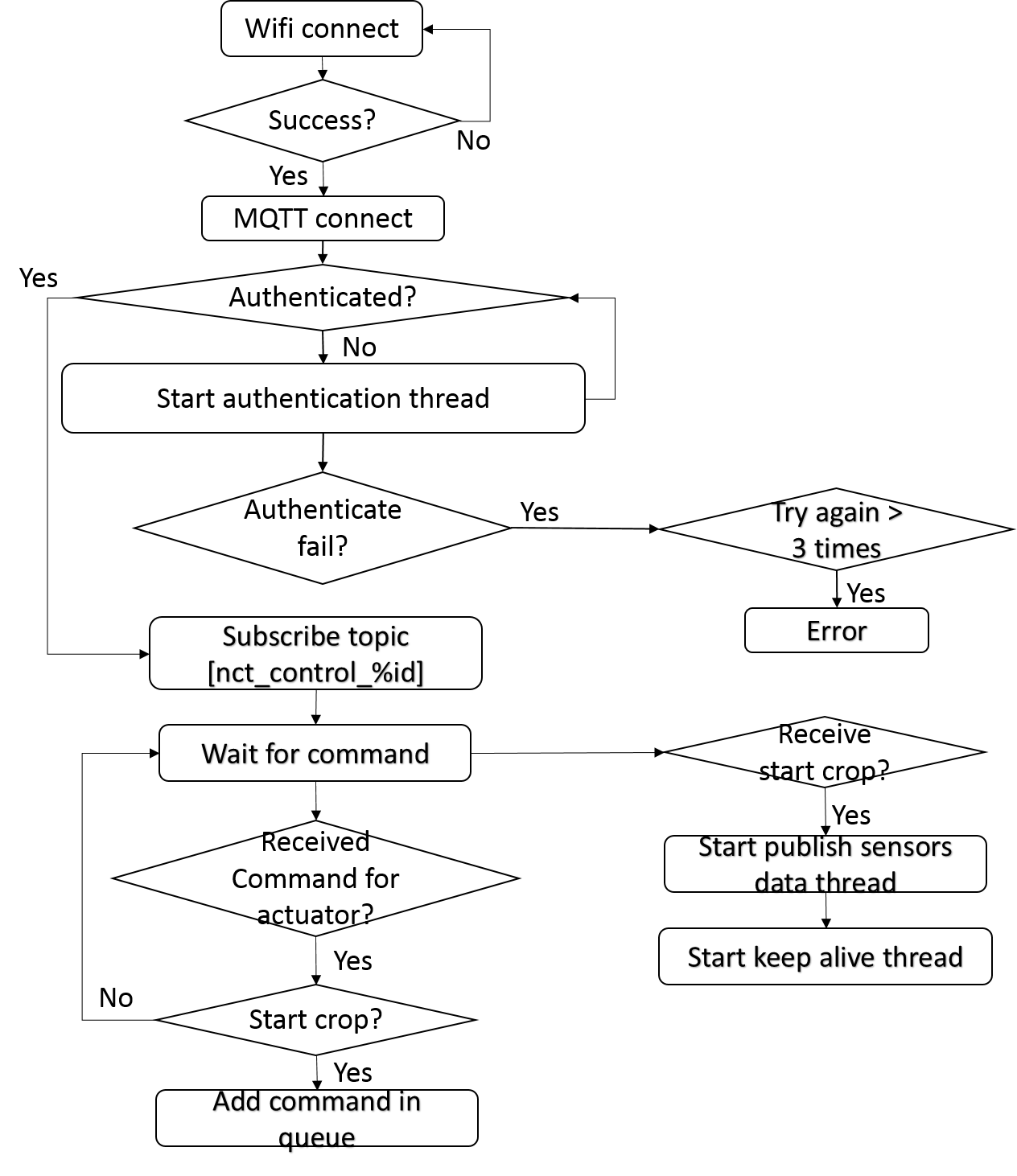
* **Khối quản lý kết nối Wifi:**



Hình 12 Sơ đồ khối quá trình quản lý Wifi

Mô tả quá trình:

* Bước 1: Khởi động, ESP32 sẽ lấy các thông tin về Wifi (SSID, Password) trong bộ nhớ flash.
* Bước 2: Nếu có các thông tin về Wifi, ESP sẽ kết nối tới Wifi nếu Wifi này sẵn sàng.
* Bước 2.1: Nếu các thông tin về Wifi không có trong bộ nhớ flash, ESP32 sẽ vào chế độ chờ, đợi người dùng bật Http server để setup Wifi cho ESP32. Sau khi nhận được SSID và Password của Wifi từ người dùng, ESP sẽ kết nối tới Wifi được chỉ định.
* Bước 3: Nếu thành công, thông tin về Wifi sẽ được lưu trong bộ nhớ nếu nó chưa được lưu, kết thúc quá trình kết nối Wifi.
* Bước 3.1: Nếu kết nối thành công, ESP32 sẽ cố gắng kết nối lại cho đến khi thành công hoặc khi người dùng đổi cài đặt sang Wifi khác.
* **Khối quản lý MQTT:**



Hình 13 Sơ đồ khối quản lý MQTT

Mô tả quá trình:

* Bước 1: Khởi động, nếu đã kết nối Wifi thành công, thực hiện kết nối với MQTT broker, ở đồ án này, broker được sử dụng là “**hivemq.mqtt.com:1883**”.
* Bước 1.1: Nếu không có kết nối Wifi, không thực hiện kết nối tới MQTT broker.
* Bước 2: Sau khi kết nối broker thành công, nếu thiết bị chưa được xác thực, “tác vụ xác thực” sẽ được khởi tạo, tác vụ này sẽ subscribe topic “**nct\_authentication\_result\_{id}”** với “id” là id của thiết bị, đồng thời định kỳ publish gói tin xác thực lên topic “**nct\_authentication**” cho đến khi nhận được gói tin xác nhận việc xác thực thành công từ server.

Gói tin xác thực có dạng JSON như sau:

{"id" : [id],

"user\_token" : "string",

"device\_type" : "ESP32"}

Trong đó:

* Trường “id” là id của thiết bị ESP, có nhiệm vụ định danh thiết bị trong hệ thống, có kiểu chuỗi ký tự.
* Trường “user\_token” là chuỗi ký tự được mã hóa, có chứa thông tin của người dùng để xác nhận và đăng ký thiết bị vào đúng người, có kiểu chuỗi ký tự.
* Trường “device\_type” là phân loại cho thiết bị, phục vụ việc mở rộng trong tương lại khi có nhiều loại thiết bị khác nhau, có kiểu chuỗi ký tự.

Về phía server, gói tin xác nhận được gửi lại thiết bị có dạng JSON như sau:

{"status" : "ERROR/OK",

"message" : "string"}

Trong đó:

* Trường “status” thể hiện trạng thái xác thực thành công hay lỗi.
* Trường “message” là thông điệp thông báo lỗi đã gặp phải.
* Bước 2.1: Nếu server từ chối xác thực do thông tin xác thực bị sai, thiết bị sẽ cố gắng gửi xác thực lại 3 lần, nếu vẫn không được, dừng “tác vụ xác thực”. Thiết bị quay trở lại chế độ chờ.
* Bước 3: Sau khi thiết bị đã được xác thực, thiết bị sẽ subscribe topic “**nct\_control\_{id}**” và lắng nghe các mệnh lệnh từ server.
* Bước 4: Nếu nhận được mệnh lệnh từ server thông qua topic “**nct\_control\_{id}**”, nếu là lệnh “bắt đầu mùa vụ”, thiết bị sẽ thực hiện khởi tạo các tác vụ gửi dữ liệu của cảm biến tới topic “**nct\_collect**” theo chu kỳ nhận được trong trong gói tin của lệnh “bắt đầu mùa vụ”. Đồng thời khởi tạo tác vụ keep alive, tác vụ này sẽ định kỳ gửi gửi một gói tin lên topic “**nct\_keep\_alive**”, có nhiệm vụ thông báo cho server biết rằng ESP vẫn đang chạy ổn định. Server sẽ dựa vào những gói tin này để nhận biết có bao nhiêu thiết bị vẫn đang chạy trong hệ thống.

Gói tin “bắt đầu mùa vụ” có dạng JSON như sau:

{"action" : "start\_crop/stop\_crop",

"time\_period" : number,

" device\_token" : string}

Trong đó:

* Trường “action” là hành động cần thực hiện, ở đây là hành động “start\_crop” – bắt đầu mùa vụ, còn hành động nữa là “stop\_crop” – kết thúc mùa vụ.
* Trường “time\_period” là chu kỳ server yêu cầu ESP gửi dữ liệu cảm biến cho server.
* Trường “device\_token” do server cung cấp cho ESP nhằm mục đích xác thực các gói tin sau này của ESP gửi đi. Đây là một chuỗi ký tự mã hóa các thông tin về “mùa vụ”.

Gói tin dữ liệu của cảm biến có dạng JSON như sau:

{

"id" : [id], // id của ESP32

"device\_token" : string, // từ mã để xác thực gói tin

"timestamp" : string, // thời gian gửi

"packet\_no" : int, // số thứ tự gói tin

"temperature" : float, // dữ liệu nhiệt độ

"humidity" : float, // dữ liệu độ ẩm

"EC" : float, // dữ liệu EC

"pH" : float, // dữ liệu pH

"light\_intensity" : float // dữ liệu về cường độ ánh sáng

}

Gói tin keep alive có dạng JSON như sau:

{

“id" : [id],

"device\_token" : string,

"timestamp" : string,

"active" : "true"

}

* Bước 4.1: Nếu nhận được gói tin “kết thúc mùa vụ”, thiết bị sẽ kết thúc các tác vụ gửi dữ liệu cảm biến và keep alive.
* Bước 5: Nếu đã bắt đầu mùa vụ và nhận được các gói tin điều khiển các thiết bị điều khiển như đèn, bơm, … thông qua topic “**nct\_control\_{id}**”. Các gói tin có định dạng JSON sẽ được thông dịch và lưu trữ tại “**command\_queue**” rồi sau đó sẽ được lần lượt lấy ra để thực thi.

Định dạng gói tin như sau:

{

"dev\_id" : [id],

"command\_no" :number,

"actuator\_id" :number,

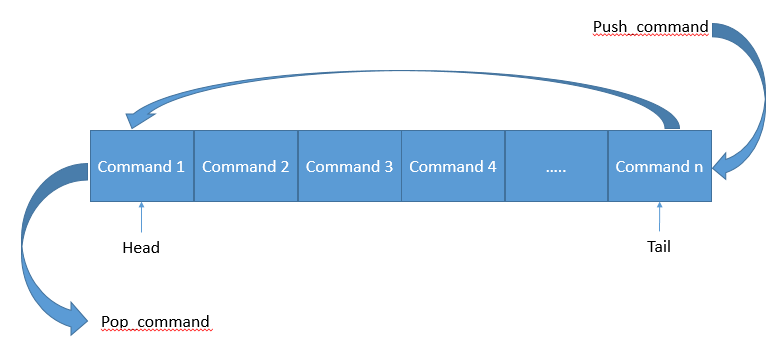
"actuator\_name" :string,

"action" :string,

"param" :number

}

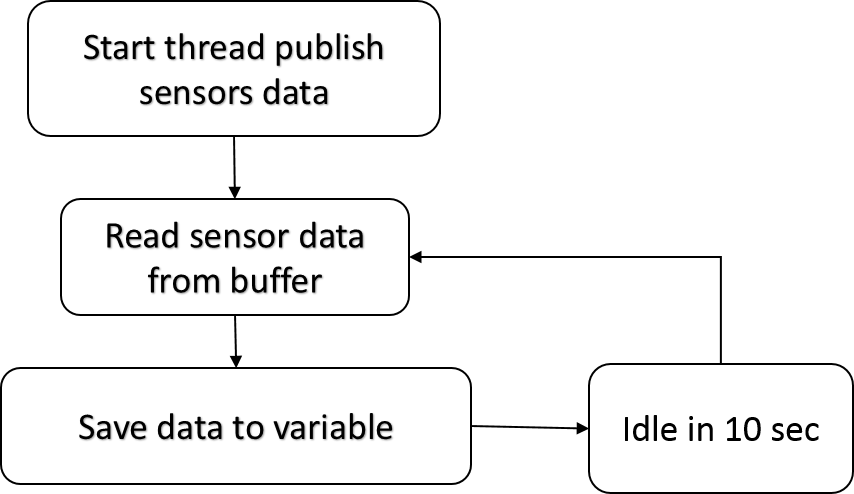
Cấu trúc của “command\_queue” như sau:



Hình 14 Cấu trúc của command\_queue

Cách thức thực hiện các lệnh sẽ được nói rõ hơn ở khối quản lý thiết bị điều khiển

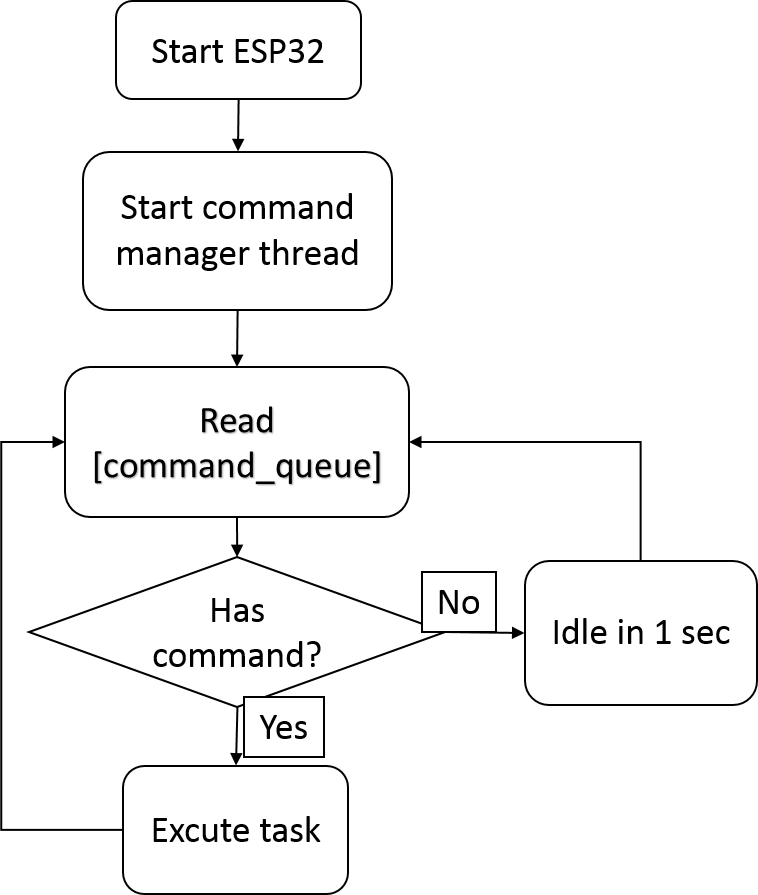
* **Khối quản lý cảm biến:**

****

Hình 15 Sơ đồ khối quản lý cảm biến

Mô tả: khối này phụ trách việc đo đạc và lưu trữ các giá trị môi trường do cảm biến đo đạc được. Các giá trị được lưu trữ là các giá trị tại thời điểm đo đạc gần nhất, các giá trị mới sẽ ghi đè lên các giá trị cũ. Các giá trị này sẽ được tác vụ “gửi dữ liệu” trong khối quản lý MQTT lấy ra và gửi về server.

* **Khối quản lý thiết bị điều khiển (bơm, đèn, …):**

****

Hình 16 Sơ đồ khối quản lý thiết bị điều khiển

Sau khi khởi động, ESP sẽ tạo một luồng song song để thực hiện việc quản lý và thực thi lệnh điều khiển. Luồng này liên tục kiểm tra “command\_queue” – hàng đợi dùng để lưu các lệnh điều khiển nhận được từ server, và thực thi nếu có.

Khi thực thi một lệnh, ESP sẽ tạo ra một luồng riêng khác để chạy thiết bị ngoại vi thông qua các GPIO tương ứng. Luồng này sẽ được gán một handler nhằm mục đích quản lý luồng.

Luồng chạy thiết bị ngoại vi sẽ chạy cho đến khi đạt điều kiện quy đinh (về thời gian hoặc cho đến khi các thông số môi trường thay đổi đến mức thích hợp: ví dụ quạt sẽ chạy cho đến khi độ ẩm giảm xuống) thì sẽ tự động ngắt thiết bị ngoại vi và xóa luồng để tiết kiệm bộ nhớ.

Nếu lệnh đến sau điều khiển cùng 1 thiết bị ngoại vi của lệnh đến trước thì lệnh đến sau xóa luồng thực hiện của lệnh đến trước và tạo một luồng thực thi mới.

## Chương 2: Triển khai hệ thống

Trong chương này, em sẽ trình bày các phần cứng được sử dụng tại module thu thập và điều khiển, môi trường lập trình cho module ESP32 Dev-Kit, xây dựng chương trình để thu thập, điều khiển và gửi nhận dữ liệu cho ESP32 Dev-Kit.

### Phần cứng tại module thu thập và điều khiển

Tại module thu thập và điều khiển có nhiệm vụ thu thập dữ liệu từ các cảm biến rồi gửi dữ liệu này lên server thông qua giao thức MQTT. Đồng thời cũng nhận lệnh điều khiển từ server, cũng thông qua MQTT, và thực hiện các lệnh này bằng các thiết bị điều khiển, như bơm, quạt, đèn, … nhằm thay đổi các thông số môi trường.

Thành phần phần cứng bao gồm:

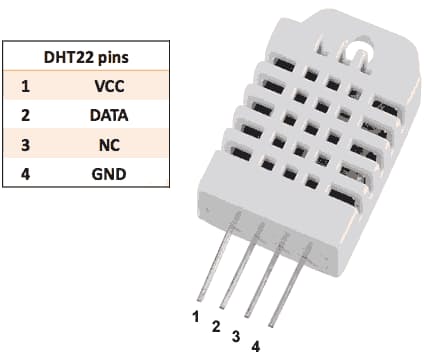
* Module ESP32 Dev-Kit
* Cảm biến nhiệt độ, độ ẩm DHT22
* Cảm biến cường độ ánh sáng BH1750
* Cảm biến PH\_meter\_SKU\_SEN0161
* Cảm biến EC Analog\_EC\_meter\_SKU\_DRF0300
* Cảm biến DS18B20
* Máy bơm nhu động 12V
* Đèn LED
* Bơm nước 220V
* Module ESP32 Dev-Kit: Được dùng để kết nối và điều khiển các thiết bị ngoại vi khác như cảm biến, bơm, đèn. Kết nối tới Internet qua Wifi để gửi nhận dữ liệu.



Hình 17 Module Esp32 Dev-Kit

Thông số kỹ thuật:

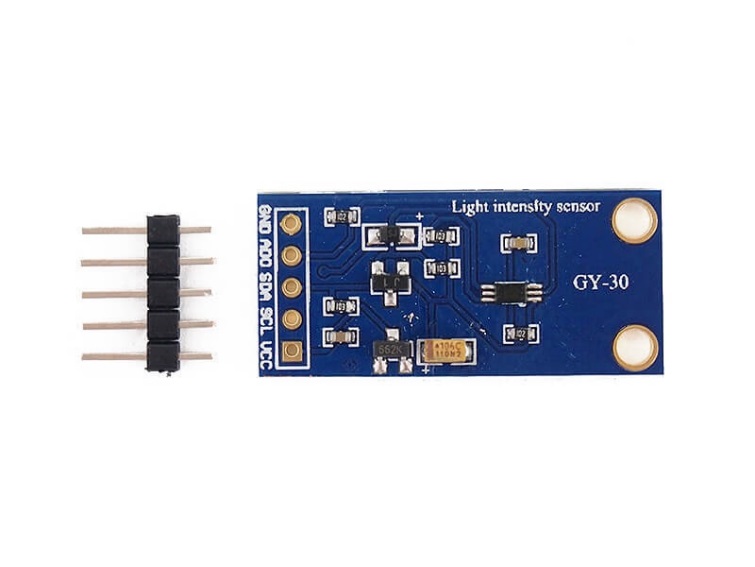
* CPU
  + CPU Xtensa Dual-Core LX6 microprocessor
  + Tốc độ xử lý 160MHz – 240MHz
  + RAM: 520 Kbyte SRAM
* Giao tiếp không dây
  + Wifi: 802.11 b/g/n/e/i
  + Bluetooth: v4.2 BR/EDR và BLE
* Các loại giao tiếp khác
  + 8 bit DAC (Digital to Analog) 2 cổng
  + Analog 12 bit 6 cổng
  + UART 3 cổng
  + Thẻ nhớ SD
  + SPI 3 cổng
  + CAN bus 2.0
* Các cảm biến tích hợp
  + 1 cảm biến Hall
  + 1 cảm biến đo nhiệt độ
  + Cảm biến chạm với 10 đầu vào khác nhau
* Khả năng bảo mật
  + Mã hóa flash.
  + Mã hóa phần cứng: AES, SHA-2, RSA, sinh số ngẫu nhiên.
* Cảm biến nhiệt độ, độ ẩm DHT22



Hình 18 Module cảm biến DHT22

Thông số kỹ thuật:

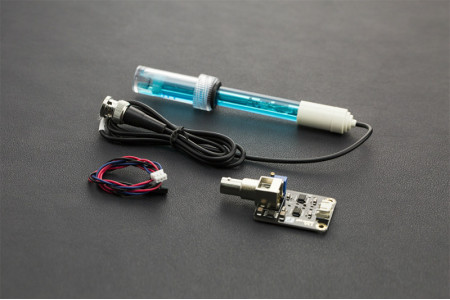
* Điện áp hoạt động: 3.3 - 5V
* Chuẩn giao tiếp: 1 wireSinh viên thực hiện: Đỗ Đình Thái – 20133570 – K58 – Lớp CNTT1.01 33
* Dải đo độ ẩm: 0 – 99.9%
* Dải đo nhiệt độ: -40 – 80oC
* Sai số độ ẩm: ± 2%
* Sai số nhiệt độ: ± 0.5oC
* Cảm biến cường độ ánh sáng BH1750



Hình 19 Module cảm biến ánh sáng BH1750

Thông số kỹ thuật:

* Nguồn: 3 -> 5VDC
* Giao tiếp: I2C
* Khoảng đo: 1 -> 65535 lux
* Kích cỡ: 21\*16\*3.3mm
* Độ phân giải: 16bit
* Cảm biến PH\_meter\_SKU\_SEN0161



Hình 20 Module cảm biến pH

Thông số kỹ thuật:

* Điện áp hoạt động: 5.00V
* Thang đo: 0-14PH
* Khoảng nhiệt độ đo :0-60 ℃
* Độ chính xác: ± 0.1pH (25 ℃)
* Thời gian phản ứng: ≤ 1 phút
* Cảm biến EC Analog\_EC\_meter\_SKU\_DRF0300



Hình 21 Module cảm biến EC

Thông số kỹ thuật:

* Điện áp hoạt động: +5.00 V
* Khoảng giá trị đo được: 1ms/cm~20ms/cm
* Khoảng nhiệt độ hoạt động: 5-40 ℃
* Độ chính xác: <±10% F.S
* Cảm biến đo nhiệt độ nước DS18B20



Hình 22 Module cảm biến DS18B20

Thông số kỹ thuật:

* Điện áp hoạt động: 3.0 - 5.5V DC
* Dòng hoạt động: 4.0 mA max, 750 nA standby
* Độ phân giải: 9 đến 12 bit (có thể lựa chọn)
* Khoảng nhiệt độ hoạt động:
  + -10 to +85 độ C với độ chính xác ~0.5 độ C
  + -55 to 125 độ C với độ chính xác ~2.0 độ C
* Máy bơm nhu động 12V



Hình 23 Bơm nhu động

* Đèn LED



Hình 24 Giàn đèn LED trong nông nghiệp

* Bơm nước 220V



Hình 25 Bơm nước thủy canh

### 2.2 Môi trường lập trình

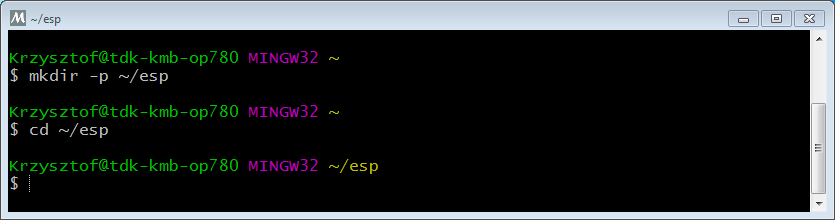
Chúng ta có thể sử dụng bộ công cụ của ESP-IDF hoặc Arduino IDE để lập trình cho module ESP32 Dev-Kit. Trong đồ án này, em sử dụng bộ công cụ của ESP-IDF để làm môi trường lập trình trên hệ điều hành Windows.

Để xây dựng và lập trình một dự án bằng ESP-IDF, ta cần thực hiện các bước sau:

* Cài đặt toolchain:
  + Trong Windows chưa sẵn có một môi trường nào để có thể cài đặt toolchain. Vì vậy trước khi cài đặt toolchain thì cần phải tạo ra môi trường phù hợp, cụ thể hơn ở đây sẽ sử dụng MSYS2. Để cài đặt tất cả mọi thứ từ MSYS2 và toolchain chúng ta chỉ cần tải một tệp nén từ dl.espressif.com:

<https://dl.espressif.com/dl/esp32_win32_msys2_environment_and_toolchain-20181001.zip>

* + Giải nén tập tin trên vào thư mục tùy ý, ở đây em sẽ chọn “D:\”, sau khi giải nén ta sẽ được thư mục “msys32”.
  + Truy cập vào thư mục “msys32” và tìm đến tập tin “mingw32.exe”. Sẽ mở ra một cửa sổ, tại đây sẽ cung cấp một môi trường gọi là “bash shell”.
  + Tạo một thư mục “esp” để có thể cài đặt ESP-IDF, để là vậy, ta chi cần gõ “**mkdir -p ~/esp**” vào bash shell, gõ “**cd esp**” để truy cập thư mục vừa tạo



Hình 26 Hình minh họa Bash shell

* + Để tải ESP-IDF, vẫn giữ bash shell trên, ta nhập vào dòng lệnh sau:

git clone --recursive <https://github.com/espressif/esp-idf.git>

* + ESP-IDF sẽ được download vào thư mục “**~/esp/esp-idf**”
* Cài đặt biến môi trường:
  + Tạo một file script trong đường dẫn “**~/msys32/etc/profile.d/**”. Đặt tên là **export\_idf\_path.sh**.
  + Xác định đường dẫn cho ESP-IDF. Nó tùy thuộc theo cài đặt hệ thống, ví dụ như trong trường hợp này là “**D:\msys32\home\user-name\esp\esp-idf**”
  + Thêm “**export**” command vào script file, e.g.:

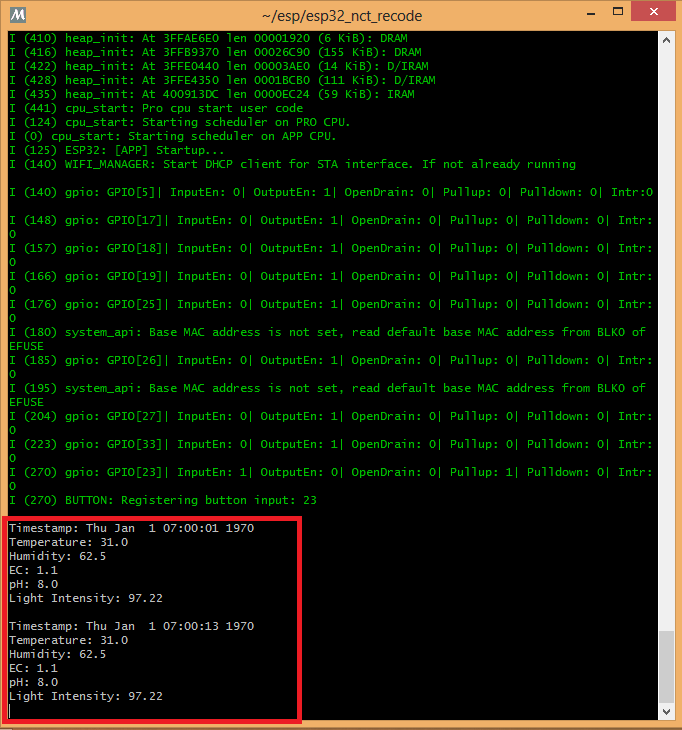
export IDF\_PATH="C:/msys32/home/user-name/esp/esp-idf"

* + Lưu script file.
* Cài đặt các gói Python:
  + Chạy dòng lệnh sau trong bash shell:

python -m pip install --user -r $IDF\_PATH/requirements.txt

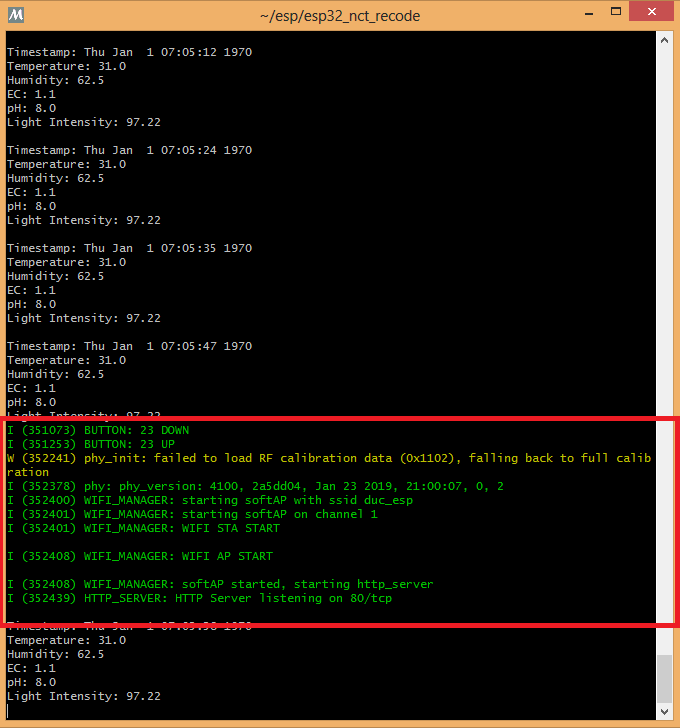
* Hoàn thành.
  1. **Triển khai hệ thống**

Khi thiết bị được cấp nguồn, nó sẽ khởi tạo các khối quản lý cảm biến và bắt đầu thu thập dữ liệu:



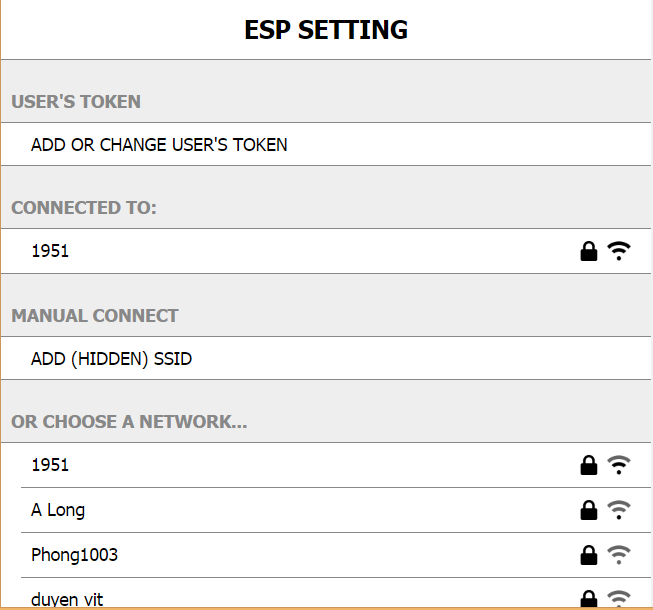
Hình 27 Hệ thống bắt đầu lấy các dữ liệu từ cảm biến

Lúc này Wifi chưa được setup nên MQTT vẫn chưa được kết nối. Nếu người dùng nhấn nút chuyển sang chế độ AP/STA, ESP32 sẽ phát Wifi và bật Http server để người dùng có thể setup Wifi:



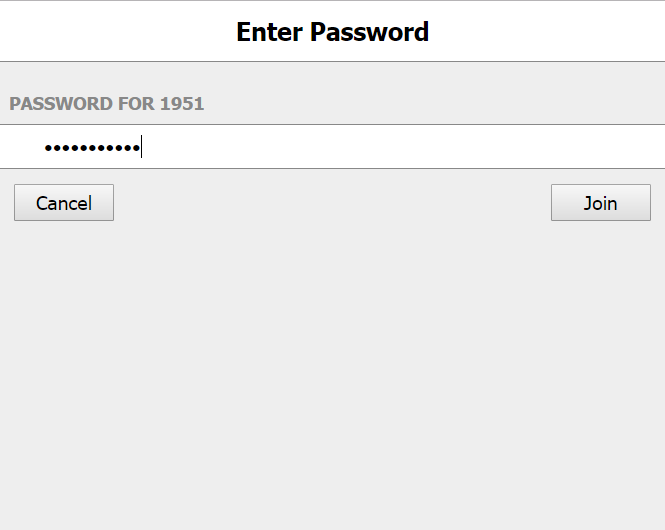
Hình 28 Bật chế độ AP/STA và mở Http server

Bây giờ, khi kết nối vào Wifi “**nct\_esp**” và truy cập vào đường link “**192.168.1.1**”, giao diện Setup ESP sẽ hiện ra:

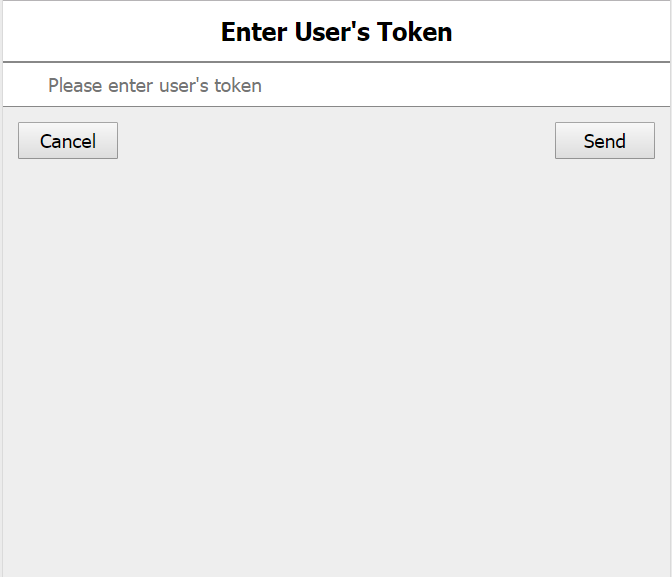


Hình 29 Giao diện Setup ESP

Tại đây, người dùng có thể lựa chọn Wifi mà người dùng muốn kết nối và nhập User Token mà người dùng được cung cập khi đăng nhập Server.

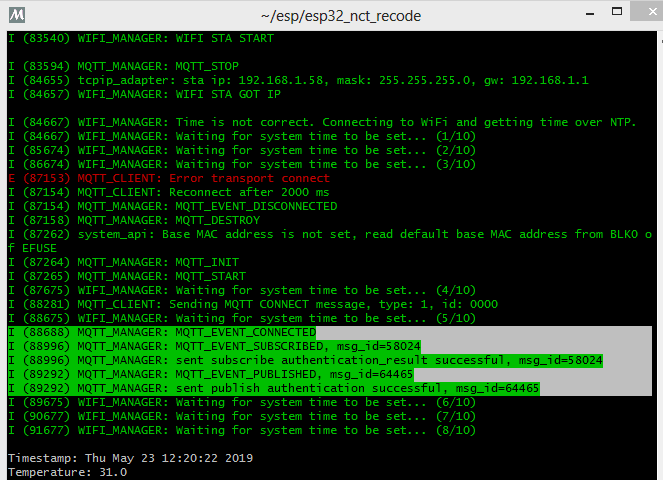


Hình 30 Giao diện đăng nhập Wifi

****

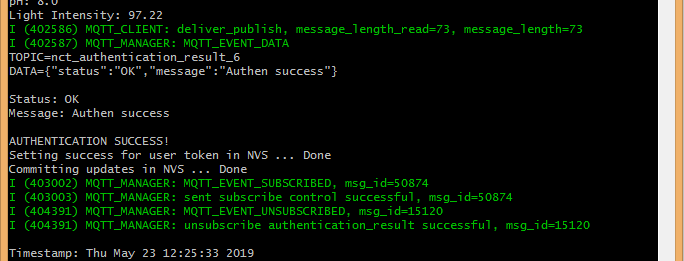
Hình 31 Giao diện nhập User Token

Sau khi có User Token và kết nối Wifi thành công, ESP32 sẽ gửi gói tin xác thực lên server thông qua MQTT.



Hình 32 ESP gửi xác thực

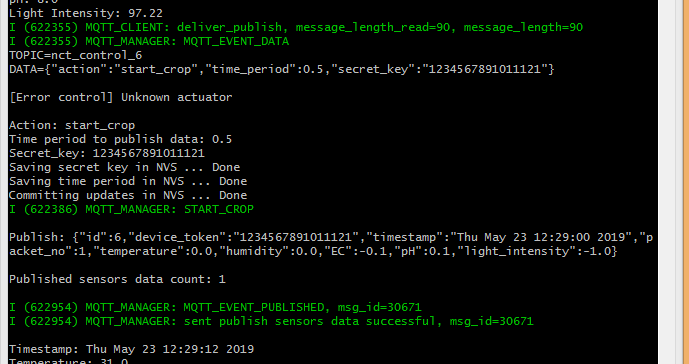
Server sẽ xác thực gói tin và gửi về kết quả.



Hình 33 Kết quả xác thực

Lúc này ESP32 sẽ subscribe topic “**nct\_control\_{id}**” để nhận các lệnh điều khiển tiếp theo.

Nếu lệnh điều khiển là “**start\_crop**”, ESP32 sẽ bắt đầu gửi các dữ liệu từ cảm biến, đồng thời gửi gói tin keep\_alive.

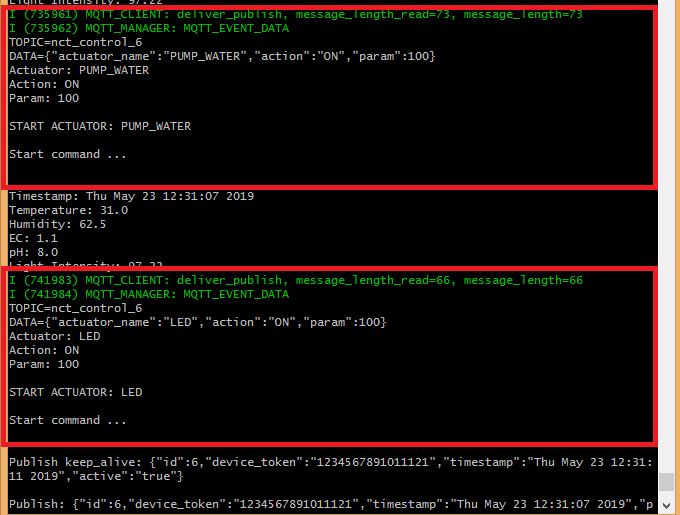


Hình 34 Nhận được lệnh "start\_crop"



Hình 35 Gửi dữ liệu và keep\_alive

ESP32 cũng sẽ nhận được các lệnh điều khiển các thiết bị ngoại vi khác từ server.



Hình 36 Các lệnh điều khiển từ server

## Chương 3: Đánh giá kết quả đạt được

Sau khi phân tích thiết kế và triển khai hệ thống, dựa theo các kết quả đạt được, em thấy.

* Đã đáp ứng:
  + Đã có đăng ký thiết bị với tài khoản để quản lý và xác thực.
  + Hệ thống đã đảm bảo tính thời gian thực, gửi nhận dữ liệu và mệnh lệnh thông qua giao thức MQTT đảm bảo các thông tin đến đích với độ trễ nhỏ.
  + Hệ thống đã có thể đo đạc chính xác các thông số môi trường và thực thi các hành động điều khiển.
  + Hệ thống đã có khả năng chạy tự động, server sẽ ra các hành động điều khiển dựa trên các dữ liệu có sẵn.
* Ưu điểm đạt được
  + Thu thập dữ liệu môi trường chính xác và gửi về server và thực hiện hành động đối phó một một cách nhanh chóng với những chuyển biến của môi trường.
  + Có khả năng mở rộng cao.
  + Hệ thống có khả năng bảo mật chống lại các tấn công xấu.
  + Các thiết bị sử dụng có chí phí thấp, dễ dàng mua và thay thế, tổng sản phẩm chỉ rơi vào khoảng dưới 10 triệu đồng.
  + Được thiết kế khá hoàn chỉnh từ thiết bị đầu cuối cho đến server.
* Những nhược điểm còn tồn tại
  + Chưa có ứng dụng đa nền tảng.
  + Hệ thống vẫn còn nhiều ngoại lệ và lỗi chưa được tính toán hết.
  + Do điều kiện trang bị nên chất lượng của hệ thống còn thấp, dễ hỏng hóc sau thời gian dài.

# KẾT LUẬN VÀ ĐỊNH HƯỚNG PHÁT TRIỂN

Trên đây là toàn bộ đồ án của em. Tuy đã cố gắng hoàn thiện với khả năng cao nhất nhưng vẫn không thể tránh khỏi thiết sót do còn thiếu kinh nghiệm trong lĩnh vực này cũng như cơ sở vật chất còn thiếu. Sau đây là những điểm đã làm được và chưa làm được của em khi thực hiện đồ án:

* **Kết quả đạt được**
* Tìm hiểu được về phương pháp trồng rau thủy canh và đặc tính sinh trưởng của các loại cây.
* Tìm hiểu đặc tính kĩ thuật và áp dụng các cảm biến vào hệ thống.
* Tìm hiểu, thiết kế và xây dựng chương trình cho module ESP32.
* Tìm hiểu và ứng dụng được giao thức MQTT.
* Phân tích, thiết kế và xây dựng hệ thống hoàn chỉnh.
* Xây dựng được tính năng cấu hình cơ bản cho ESP32.
* **Những điểm hạn chế**
* Chưa đánh giá được mức tiêu thụ năng lượng của các thiết bị.
* Chưa xây dựng được tính năng OTA, cập nhật mã nguồn từ xa cho module ESP32.
* **Định hướng phát triển**
* Mở rộng tính năng cấu hình cho ESP32, có khả năng hoạt động mà không cần kết nối server, đề phòng cho các trường hợp mất kết nối.
* Mở rộng các loại cảm biến khác, phục vụ cho nhiều mô hình nông nghiệp khác nhau.
* Xây dựng ứng dựng đa nền tảng trên các thiết bị.

# TÀI LIỆU THAM KHẢO

1. Dựa trên các thông tin về dân số thế giới tại Wikipedia, URL: <https://vi.wikipedia.org/wiki/D%C3%A2n_s%E1%BB%91_th%E1%BA%BF_gi%E1%BB%9Bi>
2. Thông tin về hệ điều hành FreeRTOS, URL: <https://vi.wikipedia.org/wiki/FreeRTOS>
3. Khái niệm và kiến thức chung về MQTT, URL: <https://kipalog.com/posts/Tim-hieu-ve-giao-thuc-MQTT---IoT-protocol>
4. MQTT Essentials: Part 1 – Introducing MQTT, URL:  
   <https://www.hivemq.com/blog/mqtt-essentials-part-1-introducing-mqtt>
5. MQTT Essentials Part 6: Quality of Service 0, 1 & 2, URL:  
   <https://www.hivemq.com/blog/mqtt-essentials-part-6-mqtt-quality-of-service-levels>
6. Tổng quan về ESP-IDF, URL: <https://docs.espressif.com/projects/esp-idf/en/latest/index.html>