

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**
(★)



ĐỒ ÁN 1

NGÀNH CÔNG NGHỆ KỸ THUẬT MÁY TÍNH

ĐỀ TÀI:

**HỆ THỐNG GIÁM SÁT NÔNG NGHIỆP
DÙNG ĐỂ ƯƠM GIỐNG CÂY CÀ PHÊ**

GVHD: PGS.TS Trương Ngọc Sơn

SVTH: Trịnh Đình Phi 20119266

Nguyễn Tất Bình 20119201

Thành phố Hồ Chí Minh, tháng 6 năm 2023

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO
❧★❧

ĐỒ ÁN 1

NGÀNH CÔNG NGHỆ KỸ THUẬT MÁY TÍNH

ĐỀ TÀI:

**HỆ THỐNG GIÁM SÁT NÔNG NGHIỆP
DÙNG ĐỂ ƯƠM GIỐNG CÂY CÀ PHÊ**

GVHD: PGS.TS Trương Ngọc Sơn

SVTH: Trịnh Đình Phi 20119266

Nguyễn Tất Bình 20119201

Thành phố Hồ Chí Minh, tháng 6 năm 2023

This image shows a full page of primary-ruled paper. It features multiple horizontal rows, each defined by two parallel dotted lines. The rows are evenly spaced and extend across the entire width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the paper.

PGS.TS Trương Ngọc Sơn

LỜI CẢM ƠN

Nhóm em xin gửi lời cảm ơn sâu sắc đến Thầy Trương Ngọc Sơn – Trưởng ngành Công nghệ kỹ thuật Máy tính đã tận tình giúp đỡ chúng em trong lựa chọn đề tài cũng như trong quá trình thực hiện đề tài. Trong quá trình thực hiện đồ án cũng đã xảy ra nhiều khó khăn, thiếu sót nhưng được sự hỗ trợ và góp ý của Thầy nên nhóm đã hoàn thành được đồ án. Một lần nữa nhóm em xin chân thành cảm ơn Thầy!

Nhóm thực hiện đề tài

Trịnh Đình Phi

Nguyễn Tất Bình

MỤC LỤC

DANH MỤC HÌNH	vi
CHƯƠNG 1: GIỚI THIỆU	1
1.1. GIỚI THIỆU	1
1.2. MỤC TIÊU ĐỀ TÀI.....	1
1.3. ĐẶC TẢ KỸ THUẬT.....	2
1.4. PHƯƠNG PHÁP NGHIÊN CỨU.....	2
1.5. ĐỐI TƯỢNG VÀ PHẠM VI NGUYÊN CỨU	2
1.6. BỐ CỤC BÁO CÁO	3
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.....	4
2.1. GIỚI THIỆU PHẦN CỨNG.....	4
2.1.1. MODULE ARDUINO NANO ATMEGA328P.....	4
2.1.2. MODULE CẢM BIẾN NHIỆT ĐỘ, ĐỘ ẨM.....	4
2.1.3. MODULE CẢM BIẾN ĐỘ ẨM ĐẤT	5
2.1.4. MODULE BLUETOOTH HC - 05	6
2.2. CHUẨN TRUYỀN THÔNG UART	6
CHƯƠNG 3: THIẾT KẾ HỆ THỐNG	9
3.1. THIẾT KẾ PHẦN CỨNG	9
3.1.1. SƠ ĐỒ KHỐI	9
3.1.2. THIẾT KẾ TỪNG KHỐI	10
3.2. THIẾT KẾ PHẦN MỀM	13
3.2.1. PHẦN MỀM LẬP TRÌNH ARDUINO – ARDUINO IDE	13
3.2.2. PHẦN MỀM LẬP TRÌNH ỨNG DỤNG ĐIỆN THOẠI ANDROID	14
CHƯƠNG 4: KẾT QUẢ VÀ ĐÁNH GIÁ.....	16
4.1. KẾT QUẢ MÔ HÌNH THI CÔNG.....	16

4.2. ĐÁNH GIÁ	18
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	19
5.1. KẾT LUẬN.....	19
5.2. HƯỚNG PHÁT TRIỂN.....	19
PHỤ LỤC	20
TÀI LIỆU THAM KHẢO	23

DANH MỤC HÌNH

Hình 2.1. Module Arduino Nano.....	4
Hình 2.2. Module cảm biến nhiệt độ, độ ẩm DHT11	5
Hình 2.3. Module cảm biến độ ẩm đất	5
Hình 2.4. Module bluetooth HC – 05	6
Hình 2.5. Hệ thống truyền dữ liệu bất đồng bộ	7
Hình 2.6. Khung truyền dữ liệu trong chế độ bất đồng bộ	8
Hình 3.1. Sơ đồ khối hệ thống	9
Hình 3.2. Sơ đồ nguyên lý hệ thống	13
Hình 3.3. Giao diện Arduino IDE	14
Hình 3.4. Giao diện phần mềm lập trình ứng dụng Android	15
Hình 3.5. Lưu đồ hoạt động của hệ thống	15
Hình 4.1. Sơ đồ thiết kế mạch in PCB.....	16
Hình 4.2. Sản phẩm mạch in sau khi gia công	16
Hình 4.3. Giao diện ứng dụng trên thiết bị Android	17
Hình 4.4. Mô hình của hệ thống	17

CHƯƠNG 1

GIỚI THIỆU

1.1. GIỚI THIỆU

Trong thời buổi công nghiệp hoá – hiện đại hoá, việc sử dụng máy móc để thay thế sức lao động của con người là một nhu cầu rất lớn. Việc áp dụng khoa học – kỹ thuật vào các lĩnh vực như: công nghiệp, nông nghiệp, khai thác, sản xuất – chế biến,... vừa có thể giải quyết vấn đề năng suất lẫn chất lượng của sản phẩm.

Nông nghiệp là ngành kinh tế quan trọng của Việt Nam. Hiện nay, Việt Nam vẫn đang là một nước nông nghiệp. Với việc tự do hoá sản xuất nông nghiệp, Việt Nam đã trở thành nước đứng thứ hai trên thế giới về xuất khẩu gạo. Bên cạnh đó, Việt Nam còn có những nông sản quan trọng là: trà, cao su, sợi bông,... và đặc biệt là cây cà phê.

Cà phê là cây công nghiệp lâu năm được trồng chủ yếu ở Tây Nguyên – là nơi mà đất là đất đỏ bazan. Trồng và chăm sóc cây cà phê là một công việc rất cực khổ vì ở Tây Nguyên nơi đây chỉ có hai mùa mưa và nắng – hay có thể nói là mùa gieo trồng và mùa thu hoạch. Khí hậu và điều kiện tự nhiên nơi đây làm người nông dân chỉ có thể ươm mầm, gieo trồng, cấy ghép vào mùa mưa. Ông bà ta có câu: “Nhất nước, nhì phân, tam cần, tứ giống”. Ta có thể thấy để thu hoạch được sản phẩm nông nghiệp chất lượng và sản lượng tốt thì bên cạnh việc phải có nước, có phân bón, có sự cần cù chăm chỉ của người nông dân thì giống cũng là một yếu tố rất quan trọng và quyết định đến chất lượng và sản lượng của cây cà phê. Do đó, việc ươm mầm cây giống của cây cà phê rất được quan tâm. Bên cạnh đó việc ươm giống cây cà phê chỉ có thể thực hiện tốt nhất là vào mùa mưa để đảm bảo tỉ lệ cây giống sống sót là cao nhất. Vì vấn đề đó, nên nhóm chúng em quyết định thiết kế “hệ thống giám sát nông nghiệp dùng để ươm giống cây cà phê”.

1.2. MỤC TIÊU ĐỀ TÀI

Hệ thống giám sát nông nghiệp dùng để ươm giống cây cà phê được thiết kế để tạo ra môi trường thuận lợi thích hợp cho việc ươm mầm cây cà phê và chăm sóc cây cà phê con với điều kiện giống với môi trường tự nhiên vào mùa ươm trồng cây giống, cụ thể là vào mùa mưa.

Hệ thống gồm 2 chế độ riêng biệt và có các chức năng:

- Chế độ tự động:
 - Đo nhiệt độ trong không khí và phun sương khi nhiệt độ thấp dưới ngưỡng.
 - Đo độ ẩm đất và tưới nước khi độ ẩm đất thấp dưới ngưỡng.
- Chế độ thủ công:
 - Điều khiển máy bơm và máy phun sương bằng nút nhấn trên ứng dụng trên điện thoại Android thông qua kết nối Bluetooth.

1.3. ĐẶC TẢ KỸ THUẬT

- Chức năng hệ thống:
 - Thu thập dữ liệu từ các cảm biến
 - Xử lý dữ liệu, tính toán đưa ra quyết định phù hợp
 - Phun sương để làm mát cây
 - Tưới nước để tăng độ ẩm đất
 - Theo dõi trạng thái hoạt động và thông số của cảm biến đến điện thoại.
- Đặc tính kỹ thuật:

Dùng vi điều khiển để giao tiếp với các cảm biến để lấy dữ liệu.

- Sử dụng cảm biến nhiệt độ để khi nhiệt độ môi trường trên 35°C thì tự động phun sương
- Sử dụng cảm biến độ ẩm đất để khi độ ẩm đất dưới ngưỡng 55% thì tự động tưới nước
- Sử dụng Bluetooth để kết nối với điện thoại.

1.4. PHƯƠNG PHÁP NGHIÊN CỨU

Để hoàn thành đề tài trên, nhóm đã sử dụng phương pháp khảo sát hệ thống thực tế. Nhóm đã nghiên cứu và theo sát cách thức ươm giống và chăm sóc cây con của bà con nông dân để có thể đề ra một hệ thống giám sát có hiệu quả và tối ưu nhất cho việc tạo ra một môi trường nhân tạo thuận lợi nhất cho cây trồng và tiết kiệm được sức lao động của bà con nông dân.

1.5. ĐỐI TƯỢNG VÀ PHẠM VI NGUYÊN CỨU

- Đối tượng nguyên cứu
 - Cây cà phê con
- Phạm vi nguyên cứu
 - Nghiên cứu trên cơ sở lý thuyết về cây cà phê
 - Khảo sát thực tế về kinh nghiệm của người nông dân trồng cây cà phê

- Tham khảo về điều kiện khí hậu ở địa phương phù hợp với việc ươm trồng cây cà phê (cụ thể ở tỉnh Đắk Lắk)

1.6. BỐ CỤC BÁO CÁO

Báo cáo đề tài gồm 5 chương:

- Chương 1: Giới thiệu sơ lược về đề tài (lý do chọn đề tài, mục tiêu đề ra, giới hạn của đề tài, phương pháp và phạm vi nghiên cứu).
- Chương 2: Cơ sở lý thuyết (giới thiệu về lý thuyết liên quan tới các vấn đề cần giải quyết trong đề tài).
- Chương 3: Thiết kế hệ thống (giới thiệu về ý tưởng thiết kế, thiết kế phần cứng và phần mềm).
- Chương 4: Kết quả (các kết quả đạt được của đề tài).
- Chương 5: Kết luận và hướng phát triển (kết luận tổng kết lại các vấn đề đã giải quyết được của đề tài và hướng phát triển xa hơn cho đề tài).

CHƯƠNG 2

CƠ SỞ LÝ THUYẾT

2.1. GIỚI THIỆU PHẦN CỨNG

2.1.1. MODULE ARDUINO NANO ATMEGA328P

Arduino Nano là một board phát triển nhỏ gọn và mạnh mẽ trong hệ thống Arduino. Với kích thước nhỏ chỉ bằng một thẻ tín dụng, Arduino Nano cung cấp một nền tảng linh hoạt để thực hiện các dự án điện tử và IoT. Với vi xử lý ATmega328P và bộ nhớ flash 32KB, Arduino Nano có khả năng xử lý mạnh mẽ và đa dạng cho các ứng dụng nhỏ và trung bình.

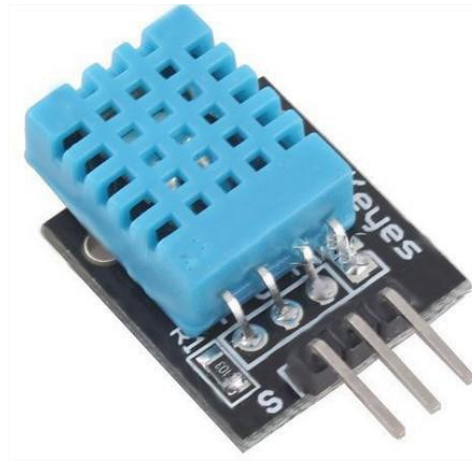
Arduino Nano hỗ trợ một loạt các chân IO số và chân analog để kết nối với các linh kiện điện tử khác nhau như cảm biến, động cơ và các module mở rộng khác. Ngoài ra, nó cũng có tích hợp một cổng USB cho phép bạn kết nối và lập trình board dễ dàng.



Hình 2.1. Module Arduino Nano

2.1.2. MODULE CẢM BIẾN NHIỆT ĐỘ, ĐỘ ẨM

Module cảm biến nhiệt độ, độ ẩm DHT11 là module cảm biến dùng để đo nhiệt độ và độ ẩm ngoài không khí rất thông dụng hiện nay vì chi phí rẻ, rất dễ lấy dữ liệu thông qua giao tiếp 1 – wire (giao tiếp digital 1 – wire truyền dữ liệu duy nhất). Cảm biến được tích hợp bộ tiền xử lý tín hiệu giúp dữ liệu nhận về được chính xác mà không cần phải qua bất kỳ tính toán nào.

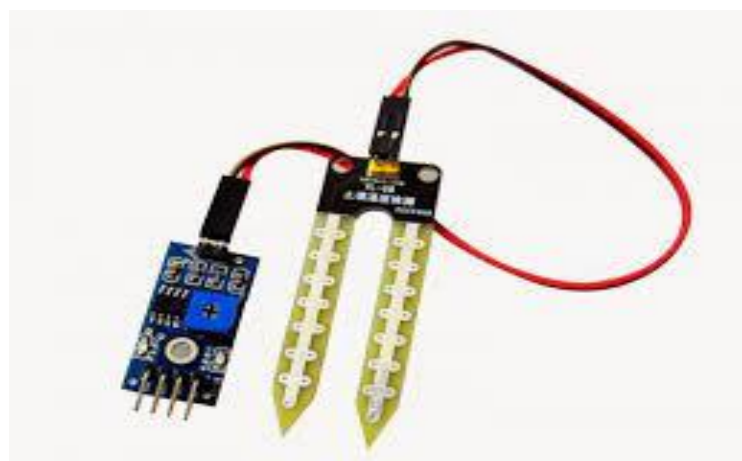


Hình 2.2. Module cảm biến nhiệt độ, độ ẩm DHT11

2.1.3. MODULE CẢM BIẾN ĐỘ ẨM ĐẤT

Module cảm biến độ ẩm đất sử dụng IC chính LM393, là một cảm biến độ ẩm đất kỹ thuật số dễ sử dụng. Chỉ cần lắp cảm biến vào đất và nó có thể đo hàm lượng độ ẩm hoặc mực nước trong đó. Nó cho đầu ra kỹ thuật số 5V khi độ ẩm cao và 0V khi độ ẩm trong đất thấp.

Cảm biến bao gồm một biến trở để đặt ngưỡng độ ẩm mong muốn. Đầu ra kỹ thuật số có thể được kết nối với một bộ điều khiển vi mô để cảm nhận mức độ ẩm. Cảm biến cũng xuất ra một đầu ra tương tự có thể được kết nối với ADC của bộ điều khiển vi mô để có được mức độ ẩm chính xác trong đất, phù hợp để thực hiện các dự án mô hình về nông nghiệp, nhà thông minh...



Hình 2.3. Module cảm biến độ ẩm đất

2.1.4. MODULE BLUETOOTH HC - 05

Module bluetooth HC – 05 dùng để thiết lập kết nối Serial giữa 2 thiết bị bằng sóng bluetooth. Điểm đặc biệt của module bluetooth HC – 05 là module có thể hoạt động được ở 2 chế độ: Master hoặc Slave.

- Ở chế độ Master: module sẽ tự động dò tìm thiết bị bluetooth khác (1 module bluetooth khác, USB bluetooth, bluetooth của laptop, smartphone...) và tiến hành pair chủ động mà không cần thiết lập gì từ máy tính hoặc smartphone.
- Ở chế độ Slave: cần thiết lập kết nối từ smartphone, laptop, USB bluetooth để dò tìm module sau đó pair với mã PIN. Sau khi pair thành công, ta sẽ có 1 cổng serial từ xa hoạt động ở baud rate 9600.

Module bluetooth HC05 được điều khiển bằng tập lệnh AT để thực hiện các tác vụ mong muốn. Để bluetooth module chuyển từ chế độ thông thường qua điều khiển bằng AT, có 2 cách như sau:

- Cấp nguồn cho module bluetooth (VCC và GND) đồng thời cấp mức điện áp cao (=VCC) cho chân KEY của module bluetooth. Khi đó giao tiếp bằng tập lệnh AT với module bằng cổng Serial (TX và RX) với baud rate là 38400.
- Cấp nguồn cho module bluetooth trước, sau đó cấp mức điện áp cao cho chân KEY của module bluetooth. Lúc này ta có thể giao tiếp với module bằng tập lệnh AT với baud rate là 9600.

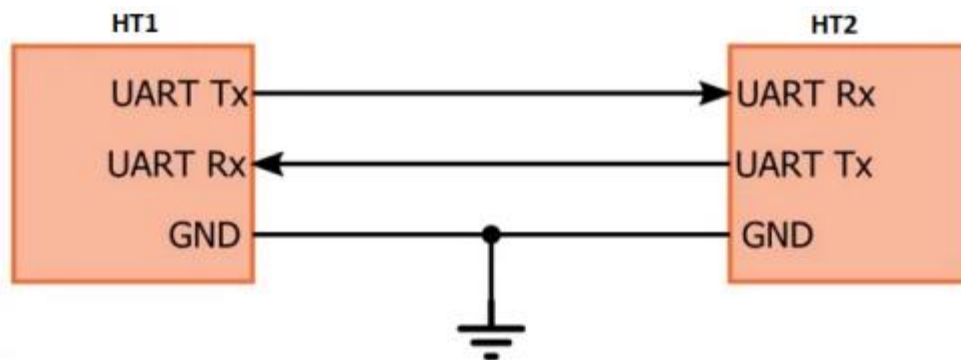


Hình 2.4. Module bluetooth HC – 05

2.2. CHUẨN TRUYỀN THÔNG UART

UART là viết tắt của Universal Asynchronous Receiver – Transmitter có nghĩa là truyền dữ liệu nối tiếp bất đồng bộ. Truyền dữ liệu nối tiếp bất đồng bộ có một đường

phát dữ liệu và một đường nhận dữ liệu, không có tín hiệu xung clock nên gọi là bất đồng bộ. Để truyền được dữ liệu thì cả bên phát và bên nhận phải tự tạo xung clock có cùng tần số và thường được gọi là tốc độ baud, ví dụ như 2400 baud, 4800 baud, 9600 baud...



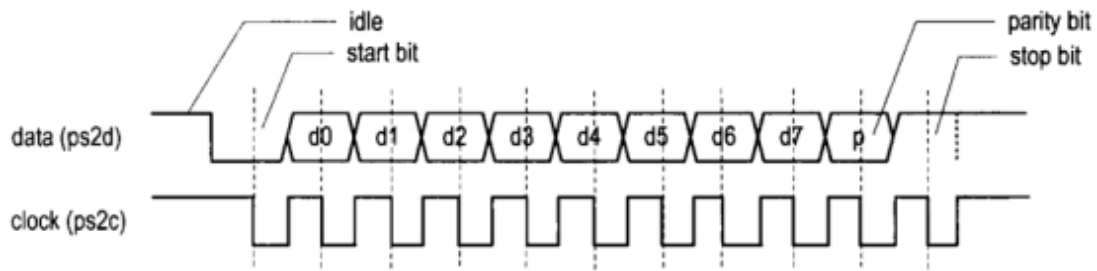
Hình 2.5. Hệ thống truyền dữ liệu bất đồng bộ

Giao tiếp UART chế độ bất đồng bộ sử dụng một dây kết nối cho mỗi chiều truyền dữ liệu do đó để quá trình truyền nhận dữ liệu thành công thì việc tuân thủ các tiêu chuẩn truyền là hết sức quan trọng. Sau đây là các khái niệm quan trọng trong chế độ truyền thông này.

Baud rate (tốc độ Baud): Để việc truyền và nhận bất đồng bộ xảy ra thành công thì các thiết bị tham gia phải thống nhất với nhau về khoảng thời gian dành cho 1 bit truyền, hay nói cách khác tốc độ truyền phải được cài đặt như nhau trước khi truyền nhận, tốc độ này gọi là tốc độ Baud. Tốc độ Baud là số bit truyền trong một giây. Ví dụ, nếu tốc độ Baud được đặt là 9600 bit/giây thì thời gian dành cho một bit truyền là $1/9600 \sim 104.167\mu s$.

Frame (khung truyền): Do truyền thông nối tiếp mà nhất là nối tiếp bất đồng bộ rất dễ mất hoặc sai lệch dữ liệu, quá trình truyền thông theo kiểu này phải tuân theo một số quy cách nhất định. Bên cạnh tốc độ Baud, khung truyền là một yếu tố quan trọng tạo nên sự thành công khi truyền và nhận. Khung truyền bao gồm các quy định về số bit trong mỗi lần truyền, các bit báo hiệu như bit Start và bit Stop, các bit kiểm tra như Parity, ngoài ra số lượng các bit dữ liệu trong mỗi lần truyền cũng được quy định bởi khung truyền.

Để bắt đầu cho việc truyền dữ liệu bằng UART, một START bit được gửi đi, sau đó là các bit dữ liệu và kết thúc quá trình truyền là STOP bit.



Hình 2.6. Khung truyền dữ liệu trong chế độ bất đồng bộ

Start bit: Là bit đầu tiên được truyền trong một khung truyền, bit này có chức năng báo cho thiết bị nhận biết rằng có một gói dữ liệu sắp được truyền tới. Start bit là bit bắt buộc phải có trong khung truyền.

Data: Data hay dữ liệu cần truyền là thông tin chính mà chúng ta cần gửi và nhận. Dữ liệu cần truyền không nhất thiết phải là gói 8 bit, có thể quy định số lượng bit của dữ liệu là 5, 6, 7, 8 hoặc 9. Trong truyền thông nối tiếp USART, bit có ảnh hưởng nhỏ nhất của dữ liệu sẽ được truyền trước và cuối cùng là bit có ảnh hưởng lớn nhất.

Parity bit: Là bit dùng kiểm tra dữ liệu truyền đúng không (một cách tương đối). Có 2 loại parity là parity chẵn và parity lẻ. Parity chẵn nghĩa là số lượng bit 1 trong dữ liệu bao gồm bit parity luôn là số chẵn. Ngược lại, tổng số lượng các bit 1 trong parity lẻ luôn là lẻ. Ví dụ, nếu dữ liệu của bạn là 10111011 nhị phân, có tất cả 6 bit 1 trong dữ liệu này, nếu parity chẵn được dùng, bit parity sẽ mang giá trị 0 để đảm bảo tổng các bit 1 là số chẵn (6 bit 1). Nếu parity lẻ được yêu cầu thì giá trị của parity bit là 1. Parity bit không phải là bit bắt buộc và vì thế chúng ta có thể loại bit này khỏi khung truyền.

Stop bits: Là một hoặc các bit báo cho thiết bị nhận rằng một gói dữ liệu đã được gửi xong. Sau khi nhận được stop bits, thiết bị nhận sẽ tiến hành kiểm tra khung truyền để đảm bảo tính chính xác của dữ liệu. Stop bits là các bit bắt buộc xuất hiện trong khung truyền.

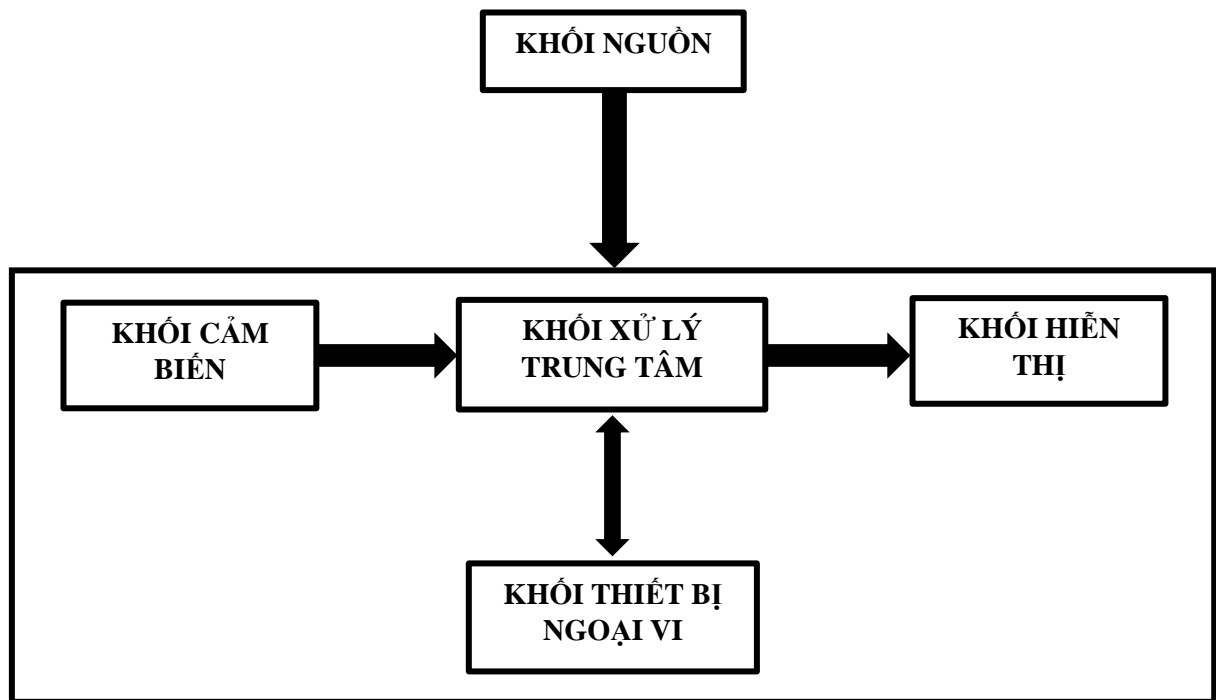
Khung truyền phổ biến nhất là (Start bit + 8 bit dữ liệu + stop bit).

CHƯƠNG 3

THIẾT KẾ HỆ THỐNG

3.1. THIẾT KẾ PHẦN CỨNG

3.1.1. SƠ ĐỒ KHỐI



Hình 3.1. Sơ đồ khối hệ thống

Khối xử lý trung tâm: Xử lý dữ liệu từ khối cảm biến gửi về, đồng thời kiểm tra hoạt động của khối ngoại vi sau đó sẽ gửi dữ liệu lên điện thoại liên tục thông qua Bluetooth.

Khối cảm biến: Bao gồm cảm biến nhiệt độ, cảm biến độ ẩm đất dùng để thu thập dữ liệu về nhiệt độ, độ ẩm đối tượng canh tác, từ đó đưa tín hiệu về khối xử lý trung tâm, so sánh với giá trị đặt trước, rồi sau đó khối xử lý trung tâm sẽ xử lý để phù hợp với yêu cầu của giá trị đặt trước.

Khối thiết bị ngoại vi: Nhận tín hiệu từ khối xử lý trung tâm để hoạt động.

Khối hiển thị: Điện thoại để hiển thị số liệu đọc được từ khối cảm biến và trạng thái của ngoại vi.

Khối nguồn: Khối cung cấp điện cho toàn hệ thống.

3.1.2. THIẾT KẾ TỪNG KHỐI

- **Khối điều khiển trung tâm**

Hiện nay trên thị trường có rất nhiều dòng vi điều khiển khác nhau như PIC, AVR, 8051, Raspberry, Arduino...Tất cả đều có thể đáp ứng được yêu cầu đặt ra nhưng nhóm chọn Arduino Nano ATmega328P vì nó có những ưu điểm sau:

- Giá thành rẻ, dễ sử dụng, là module hoàn chỉnh sử dụng vi điều khiển AVR.
- Kích thước nhỏ gọn.
- Là dòng vi điều khiển mã nguồn mở, có nhiều thư viện hỗ trợ cho các module chức năng khác nhau, trình biên dịch đơn giản, dễ sử dụng.

Thông số kỹ thuật:

- Vi xử lý: ATmega328P
- Tốc độ xung nhịp: 16MHz
- Bộ nhớ:
 - + Flash: 32KB (trong đó 2KB được sử dụng cho bootloader)
 - + SRAM: 2KB
 - + EEPROM: 1KB
- Điện áp hoạt động:
 - + Điện áp hoạt động: 5V
 - + Điện áp đầu vào (đề nghị): 7-12V
 - + Điện áp đầu vào (tối đa): 6-20V
- Chân I/O:
 - + Chân I/O số: 14 (bao gồm 6 chân có khả năng PWM)
 - + Chân đầu vào Analog: 8
 - + Dòng điện tối đa trên chân I/O: 40Ma
- Giao tiếp:
 - + UART: 1
 - + SPI: 1
 - + I2C: 1
- Kích thước:
 - + Kích thước: 18mm x 45mm
 - + Trọng lượng: 5g
- Các tính năng khác:

- + USB: 1 cổng Micro-USB
- + LED báo trạng thái: 1 (chân D13)
- + Hỗ trợ nạp chương trình thông qua bootloader Arduino

- **Khối cảm biến**

- Module cảm biến độ ẩm đất

Thông số kỹ thuật:

- + Điện áp làm việc 3.3V ~ 5V
- + Có lỗ cố định để lắp đặt thuận tiện
- + Sử dụng chip LM393 để so sánh, ổn định làm việc

- Module cảm biến nhiệt độ, độ ẩm DHT11

Thông số kỹ thuật:

- + Điện áp hoạt động: 3V – 5V (DC)
- + Dải độ ẩm hoạt động: 20% – 90% RH, sai số $\pm 5\%RH$
- + Dải nhiệt độ hoạt động: 0°C ~ 50°C, sai số $\pm 2^\circ C$
- + Khoảng cách truyền tối đa: 20m

- **Khối thiết bị ngoại vi**

- Máy bơm nước

Thông số kỹ thuật:

- + Điện áp hoạt động: 12 VDC
- + Dòng làm việc: 0.5 - 0.7A
- + Lưu lượng: 1.5 - 2L / phút (trái và phải), hút tối đa: 2m
- + Đầu: đẩy 3m
- + Tuổi thọ: 2500h
- + Nhiệt độ nước: dưới 80°C
- + Nếu nguồn cấp là 6 VDC thì dòng cấp tầm 1A

- Máy bơm áp suất phun sương

Thông số kỹ thuật:

- + Điện áp: 12VDC
- + Dòng điện tối đa: 2A
- + Áp lực tối đa: 0.48MPa
- + Đẩy: 4-5m
- + Khả năng chịu nhiệt cao nhất: 55 °C

+ Lưu lượng chảy tối đa: 3.5L/phút

- **Khởi hiển thị**

Module bluetooth HC – 05

Thông số kỹ thuật:

+ Thích hợp với bất kỳ USB Bluetooth adapters

+ Khoảng cách truyền 10m

+ Bluetooth version: V2.0+EDR

+ Điện áp hoạt động: 3.3 VDC

+ LED hiển thị trạng thái

+ Chip ổn áp 150mA 3.3 VD

+ Chân VCC, GND, TXD, RXD for the Bluetooth

+ Nút "Re-seach" (chân ON/OFF/WAKE, MCU ngoài mức "High level" có thể điều khiển to re-seach)

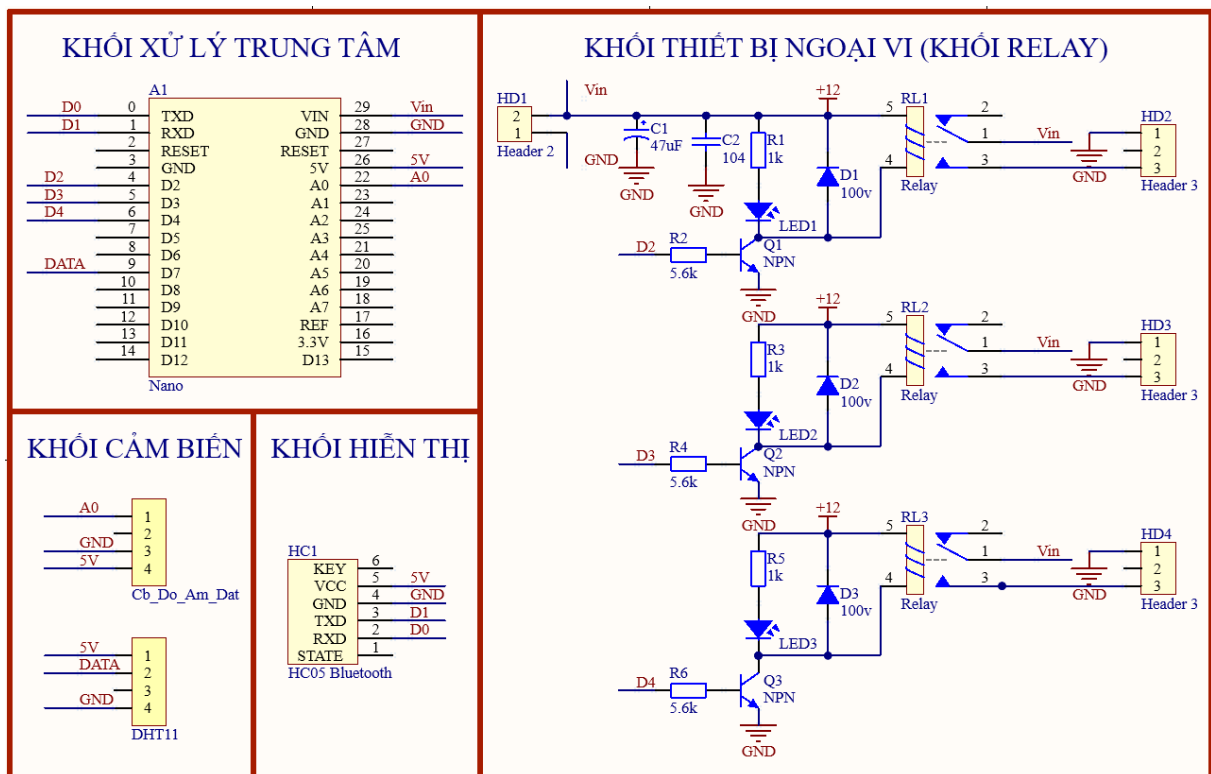
+ Tương thích với bluetooth master module "slave module" or master-slave(whole) module

+ Nguồn: 3.3 ~ 6V

- **Khởi nguồn**

Tên linh kiện	Số lượng	Điện áp (V)	Dòng điện (A)	Công suất (W)
Adruino Nano	1	12	0.019	0.228
Module cảm biến độ ẩm đất	1	5	0.04	0.2
Module cảm biến nhiệt độ, độ ẩm	1	5	0.04	0.2
Bluetooth HC-05	1	5	0.04	0.2
Máy bơm nước	2	12	0.5	12
Máy phun sương	1	12	2	24
Tổng cộng: 36.8W				

Vì điện áp vào của vi điều khiển nằm trong khoảng 7-12V nên ta sẽ chọn mạch nguồn có điện áp 12V với công suất khoảng 40W.



Hình 3.2. Sơ đồ nguyên lý hệ thống

3.2. THIẾT KẾ PHẦN MỀM

3.2.1. PHẦN MỀM LẬP TRÌNH ARDUINO – ARDUINO IDE

Arduino IDE (Integrated Development Environment) là một phần mềm mã nguồn mở được sử dụng để lập trình và nạp chương trình cho các bo mạch Arduino. Nó cung cấp một giao diện đồ họa dễ sử dụng và tích hợp các công cụ cần thiết để phát triển ứng dụng cho Arduino.

Arduino IDE hỗ trợ ngôn ngữ lập trình C/C++ và cung cấp một loạt các chức năng để viết, biên dịch và nạp chương trình vào bo mạch Arduino. Giao diện người dùng đơn giản của Arduino IDE cho phép người dùng viết mã, kiểm tra lỗi, biên dịch và tải chương trình lên bo mạch Arduino chỉ bằng vài cú nhấp chuột.

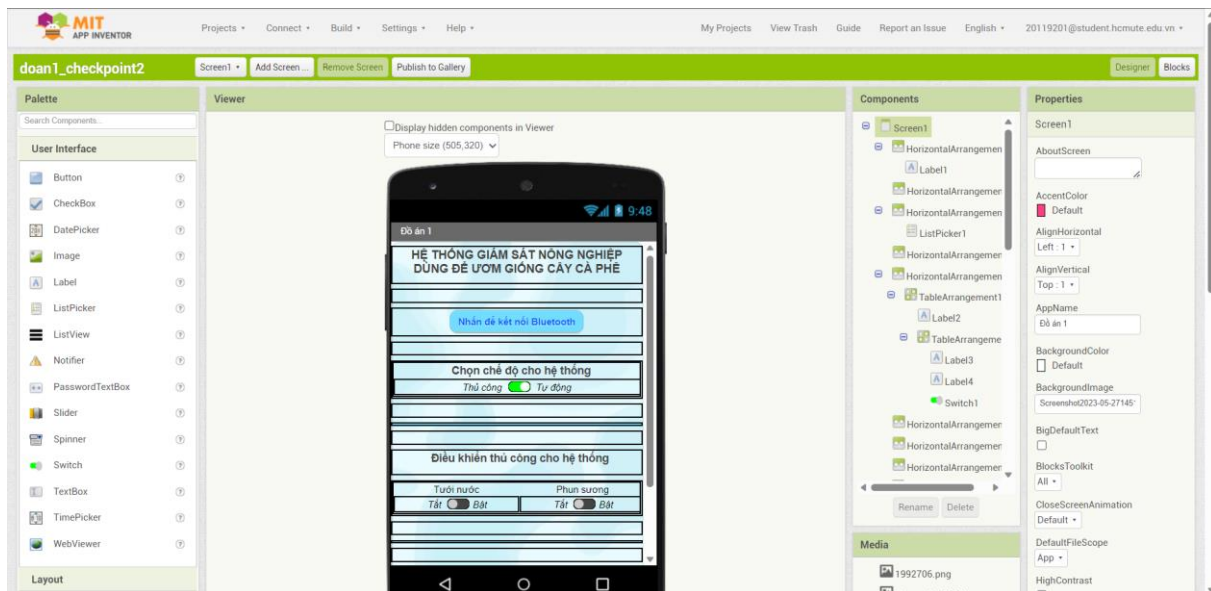


Mã nguồn chương trình Arduino ở phần phụ lục.

3.2.2. PHẦN MỀM LẬP TRÌNH ỨNG DỤNG ĐIỆN THOẠI ANDROID

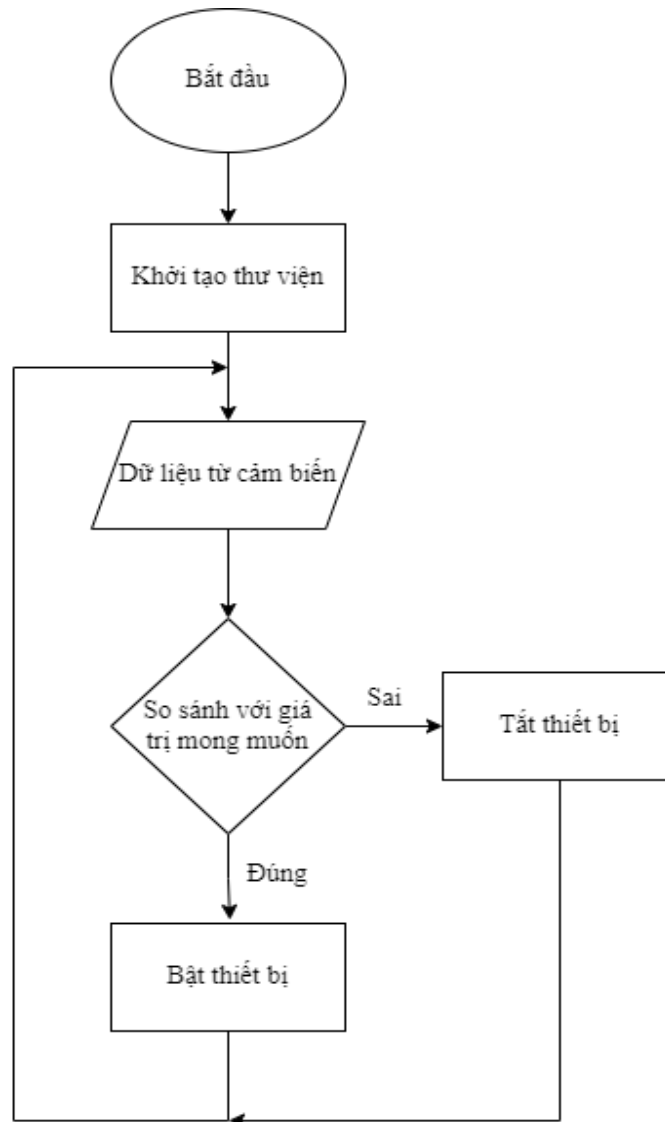
MIT App Inventor là một công cụ phát triển ứng dụng di động dựa trên web, cho phép người dùng tạo ra các ứng dụng Android mà không cần có kiến thức lập trình sâu. Được phát triển bởi Viện Công nghệ Massachusetts (Massachusetts Institute of Technology - MIT), MIT App Inventor có mục tiêu giúp mọi người, đặc biệt là người mới học lập trình, có thể dễ dàng tạo ra các ứng dụng di động.

Với MIT App Inventor, người dùng có thể tạo ra các ứng dụng di động bằng cách sử dụng giao diện kéo và thả (drag-and-drop interface) để xây dựng giao diện người dùng và logic ứng dụng. Thay vì viết mã từ đầu, người dùng chỉ cần chọn và kết hợp các khối logic có sẵn trong MIT App Inventor để tạo ra các chức năng và tương tác cho ứng dụng.



Hình 3.4. Giao diện phần mềm lập trình ứng dụng Android

3.2.3. Lưu đồ hoạt động

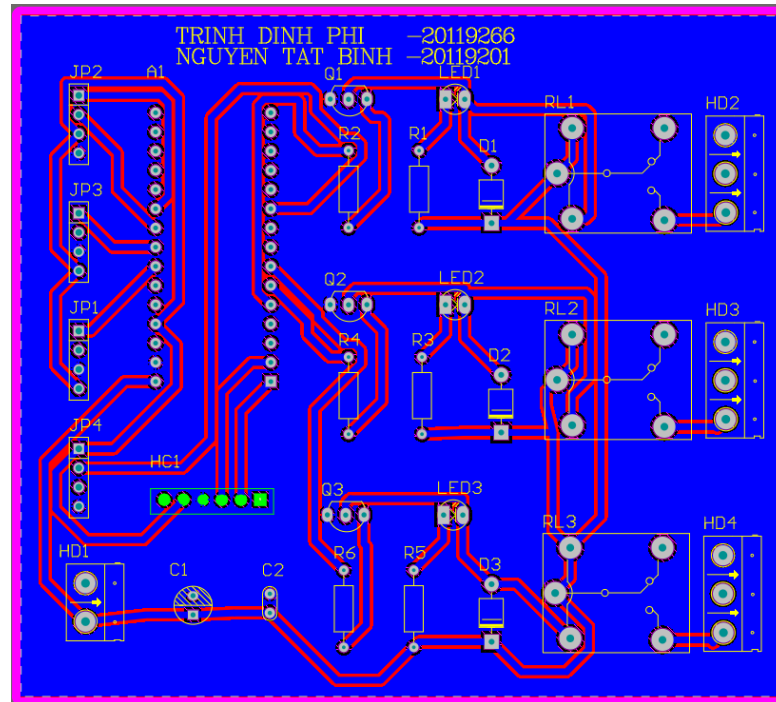


Hình 3.5. Lưu đồ hoạt động của hệ thống

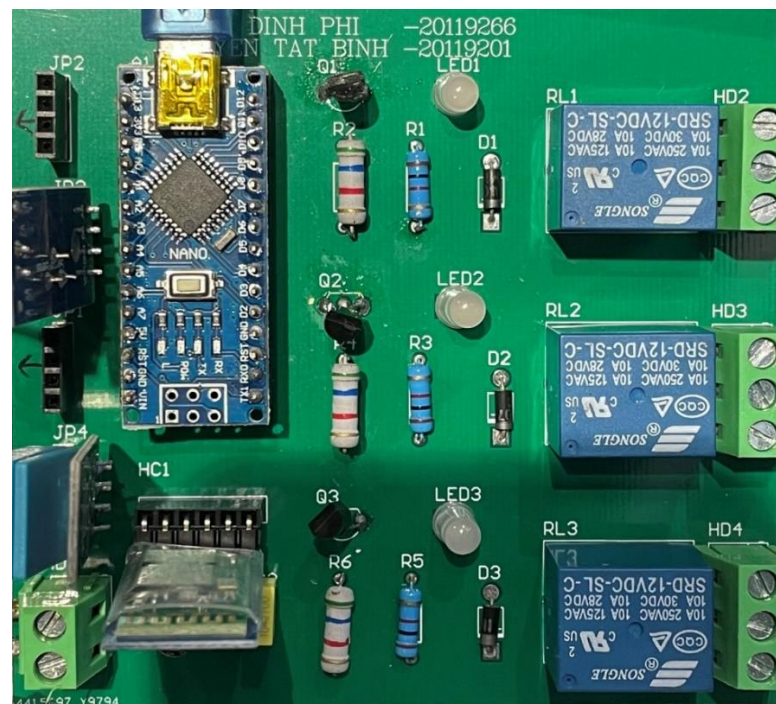
CHƯƠNG 4

KẾT QUẢ VÀ ĐÁNH GIÁ

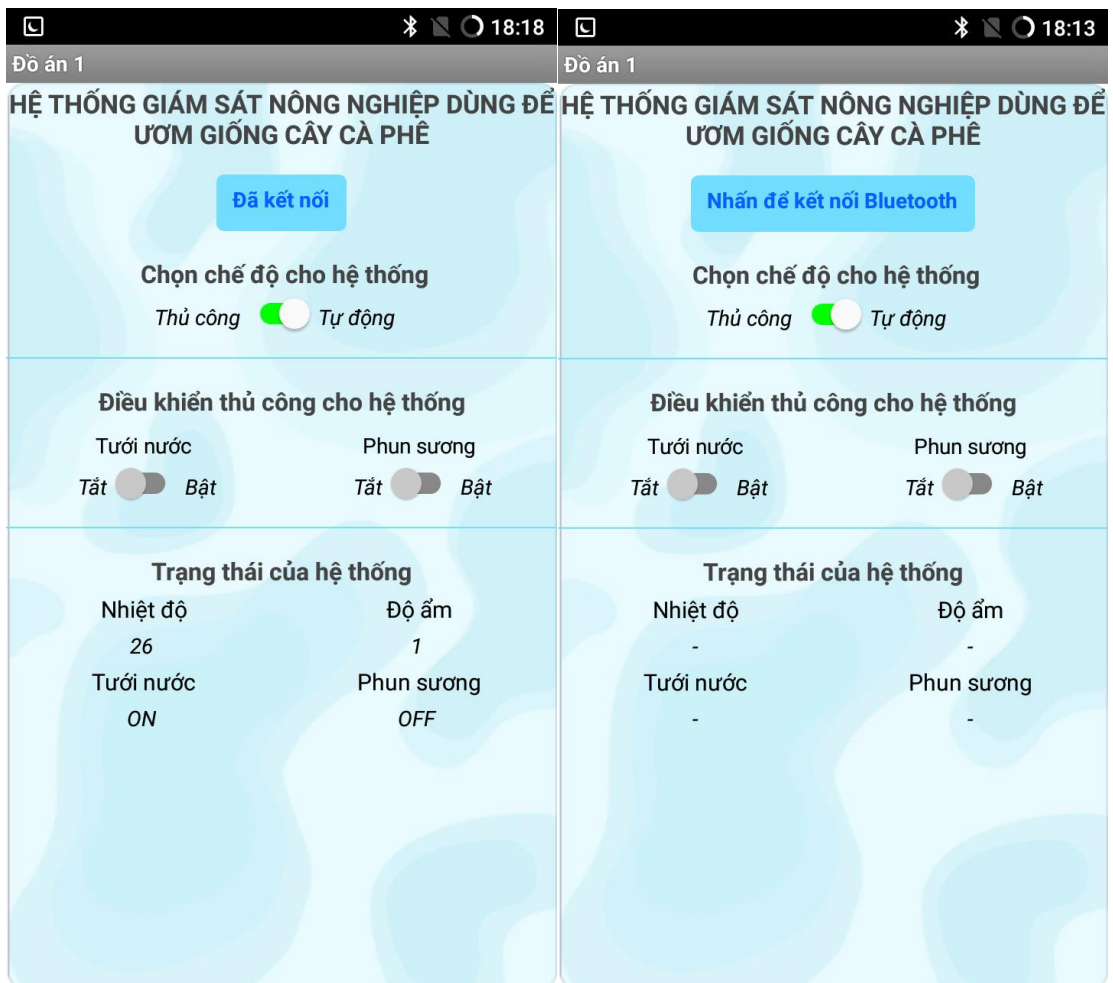
4.1. KẾT QUẢ MÔ HÌNH THI CÔNG



Hình 4.1. Sơ đồ thiết kế mạch in PCB



Hình 4.2. Sản phẩm mạch in sau khi gia công



Hình 4.3. Giao diện ứng dụng trên thiết bị Android



Hình 4.4. Mô hình của hệ thống

4.2. ĐÁNH GIÁ

- Hệ thống hoạt động đúng với yêu cầu đề ra, dữ liệu được gửi đến ứng dụng trên điện thoại.
- Các cảm biến hoạt động tốt, thu thập các dữ liệu một cách chính xác tuy nhiên vẫn có độ trễ khi dữ liệu cập nhật lên ứng dụng trên điện thoại.

CHƯƠNG 5

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1. KẾT LUẬN

Sau một thời gian nghiên cứu và hoàn thành đề tài, nhóm đã nhận thấy mô hình đã hiệu quả được 85% - 90%. Trong thời gian nghiên cứu và thực hiện đề tài, nhóm đã học hỏi và tìm hiểu thêm được nhiều kiến thức mới cũng như củng cố lại kiến thức đã học giúp hoàn thành đề tài này. Vì đây là đề tài hướng đến việc giúp cho những người nông dân giảm bớt thời gian và sức lao động, trong quá trình canh tác nông nghiệp nên phải chú trọng độ ổn định và chính xác dẫn đến nhiều khó khăn trong quá trình lập trình. Nhưng nhờ sự hướng dẫn của giảng viên hướng dẫn và các tài liệu tham khảo thì nhóm đã giải quyết được tương đối yêu cầu của đề tài.

Tuy rằng sản phẩm đã được hoàn thành nhưng nhóm vẫn nhận thấy sản phẩm còn nhiều thiếu sót, cần được chỉnh sửa và cải tiến hơn.

5.2. HƯỚNG PHÁT TRIỂN

Đề tài cơ bản đáp ứng được những yêu cầu đặt ra tuy nhiên để sản phẩm hoàn thiện được hơn nữa thì đòi hỏi cần được cải tiến và nghiên cứu thêm.

Về chức năng, thiết bị được nghiên cứu chỉ dừng lại ở các chức năng cơ bản: đọc nhiệt độ độ ẩm đất, độ ẩm không khí.

Một số chức năng nên bổ sung thêm là:

- Độ pH: Xử lý pH trong đất.
- Phát triển mô hình với quy mô lớn hơn, nhiều loại cây trồng hơn.
- Hoàn thiện mô hình và tối ưu một cách kinh tế hơn.
- Có thể dùng năng lượng từ pin mặt trời để tạo ra nguồn điện.

PHỤ LỤC

Mã nguồn chương trình Arduino:

```
1. #include "DHT.h"
2. #include <SoftwareSerial.h>
3.
4. const int DHTPIN = 7;
5. const int DHTTYPE = DHT11;
6. int relaydat = 2;
7. int cbdat = A4;
8. int doc_cb, TBcb;
9. int relaykk = 3;
10. DHT dht(DHTPIN, DHTTYPE);
11.
12. // Bluetooth control variables
13. char mode = 't';
14. char relaydatState = 'b';
15. char relaykkState = 'y';
16.
17. void setup()
18. {
19.     Serial.begin(9600);
20.     pinMode(cbdat, INPUT);
21.     pinMode(relaydat, OUTPUT);
22.     pinMode(relaykk, OUTPUT);
23.     digitalWrite(relaydat, LOW);
24.     digitalWrite(relaykk, LOW);
25.     dht.begin();
26. }
27.
28. void loop()
29. {
30.     // Bluetooth control
31.     if (Serial.available() > 0)
32.     {
33.         char command = Serial.read();
34.         handleBluetoothCommand(command);
35.     }
36.
37.     for (int i = 0; i <= 9; i++)
38.     {
39.         doc_cb += analogRead(cbdat);
40.     }
41.     TBcb = doc_cb / 10;
42.     int phantramao = map(TBcb, 0, 1023, 0, 100);
43.     int phantramthuc = 100 - phantramao;
```

```

44.     doc_cb = 0;
45.
46.     int t1 = dht.readTemperature();
47.
48.     // Automatic mode
49.     if (mode == 't')
50.     {
51.         Serial.println(t1);
52.         Serial.println(phantramthuc);
53.         if (phantramthuc >= 55)
54.         {
55.             digitalWrite(relaydat, LOW);
56.             Serial.println("OFF");
57.         }
58.         else
59.         {
60.             digitalWrite(relaydat, HIGH);
61.             Serial.println("ON");
62.         }
63.
64.         if (t1 < 35)
65.         {
66.             digitalWrite(relaykk, LOW);
67.             Serial.println("OFF");
68.         }
69.         else
70.         {
71.             digitalWrite(relaykk, HIGH);
72.             Serial.println("ON");
73.         }
74.     }
75.
76.     // Manual mode
77.     if (mode == 'h')
78.     {
79.         Serial.println(t1);
80.         Serial.println(phantramthuc);
81.         if (relaydatState == 'a')
82.         {
83.             digitalWrite(relaydat, HIGH);
84.             Serial.println("ON");
85.         }
86.         else if (relaydatState == 'b')
87.         {
88.             digitalWrite(relaydat, LOW);
89.             Serial.println("OFF");
90.         }

```

```

91.
92.     if (relaykkState == 'x')
93.     {
94.         digitalWrite(relaykk, HIGH);
95.         Serial.println("ON");
96.     }
97.     else if (relaykkState == 'y')
98.     {
99.         digitalWrite(relaykk, LOW);
100.        Serial.println("OFF");
101.    }
102. }
103. delay(1000);
104. }
105.
106. void handleBluetoothCommand(char command)
107. {
108.     switch (command)
109.     {
110.         case 't': // Automatic mode
111.             mode = 't';
112.             break;
113.         case 'h': // Manual mode
114.             mode = 'h';
115.             break;
116.         case 'a': // Turn on relaydat
117.             relaydatState = 'a';
118.             break;
119.         case 'b': // Turn off relaydat
120.             relaydatState = 'b';
121.             break;
122.         case 'x': // Turn on relaykk
123.             relaykkState = 'x';
124.             break;
125.         case 'y': // Turn off relaykk
126.             relaykkState = 'y';
127.             break;
128.         default:
129.             break;
130.     }
131. }

```

TÀI LIỆU THAM KHẢO

[1]: <https://dientutuonglai.com/giao-tiep-uart-la-gi.html>

[2]: https://www.makerguides.com/28byj-48-stepper-motor-arduino-tutorial/?fbclid=IwAR06M35Pfk_CGJXTlNE-DvkpWBe9BWVNBXCGqfGomLMR9D97DXqfVAss7A

[3]: https://github.com/hieuca/tuoicay-tudong-arduino?fbclid=IwAR1neHPYCUKI_uUkcUkppWMUw8XotOo7SLvR9qVI_BjAsyncJUw00koRPiuE