

**ĐỀ TÀI:**

**ĐỒ ÁN MÔN HỌC 1**

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

**GVHD: PGS.TS Phan Văn Ca**

**SVTH: Ngô Thái Bảo 18119058**

**Hấu Trung Đạt 18119064**

**Nguyễn Phúc Kha 18119085**

**NGÀNH CÔNG NGHỆ KỸ THUẬT MÁY TÍNH**

Hồ Chí Minh, tháng 7 năm 2021



**GVHD: PGS.TS Phan Văn Ca**

**SVTH: Ngô Thái Bảo 18119058**

**Hấu Trung Đạt 18119064**

**Nguyễn Phúc Kha 18119085**

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

**ĐỀ TÀI:**

**NGÀNH CÔNG NGHỆ KỸ THUẬT MÁY TÍNH**

**ĐỒ ÁN MÔN HỌC 1**

Hồ Chí Minh, tháng 7 năm 2021

**NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN**

*Ký tên*

PGS.TS Phan Văn Ca

**BẢNG PHÂN CÔNG CÔNG VIỆC**

|  |  |  |
| --- | --- | --- |
| **THỨ TỰ** | **CÔNG VIỆC** | **THỰC HIỆN** |
| 1 | Phụ trách luống ươm cây cà phê giống, điều chỉnh thời gian tưới nước  Giao tiếp bluetooth và thiết kế ứng dụng trên điện thoại | Ngô Thái Bảo |
| 2 | Phụ trách luống ươm cây cà phê con, tưới nước khi độ ẩm đất thấp dưới ngưỡng  Giao tiếp bluetooth và thiết kế ứng dụng trên điện thoại | Nguyễn Phúc Kha |
| 3 | Phụ trách luống ươm cây cà phê con, đo cường độ sáng và thu kéo màn che  Đo nhiệt độ trong không khí và phun sương | Hấu Trung Đạt |

**LỜI CÁM ƠN**

Nhóm em xin gửi lời cám ơn sâu sắc đến Thầy Phan Văn Ca – Giảng viên 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!

Người thực hiện đề tài

Ngô Thái Bảo

Hấu Trung Đạt

Nguyễn Phúc Kha

**MỤC LỤC**

[CHƯƠNG 1: GIỚI THIỆU 8](#_Toc76332366)

[1.1. Giới thiệu. 8](#_Toc76332367)

[1.2. Mục tiêu đề tài. 8](#_Toc76332368)

[1.3. Đặc tả kỹ thuật. 9](#_Toc76332369)

[1.4. Phương pháp nghiên cứu. 9](#_Toc76332370)

[1.5. Đối tượng và phạm vi nguyên cứu. 10](#_Toc76332371)

[1.6. Bố cục báo cáo. 10](#_Toc76332372)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 11](#_Toc76332373)

[2.1. Giới thiệu phần cứng. 11](#_Toc76332374)

[2.1.1. Board Arduino Uno R3 ATmega328 11](#_Toc76332375)

[2.1.2. Module cảm biến nhiệt độ, độ ẩm 11](#_Toc76332376)

[2.1.3. Module cảm biến độ ẩm đất 12](#_Toc76332377)

[2.1.4. Module cảm biến ánh sáng 13](#_Toc76332378)

[2.1.5. Module bluetooth 13](#_Toc76332379)

[2.1.6. Module thời gian thực 14](#_Toc76332380)

[2.1.7. Mạch điều khiển động cơ bước 15](#_Toc76332381)

[2.2. Chuẩn truyền dữ liệu và chuẩn kết nối. 15](#_Toc76332382)

[2.2.1. Chuẩn giao tiếp UART. 15](#_Toc76332383)

[2.2.2. Chuẩn giao tiếp I2C 17](#_Toc76332384)

[CHƯƠNG 3: THIẾT KẾ HỆ THỐNG 20](#_Toc76332385)

[3.1. Thiết kế phần cứng. 20](#_Toc76332386)

[3.1.1. Sơ đồ khối. 20](#_Toc76332387)

[3.1.2. Thiết kế từng khối. 21](#_Toc76332388)

[3.2. Thiết kế phần mềm. 25](#_Toc76332389)

[3.2.1. Phần mềm lập trình Arduino 25](#_Toc76332390)

[3.2.2. Phần mềm lập trình ứng dụng điện thoại 25](#_Toc76332391)

[3.2.3. Lưu đồ hoạt động. 27](#_Toc76332392)

[CHƯƠNG 4: KẾT QUẢ - ĐÁNH GIÁ 28](#_Toc76332393)

[4.1. Kết quả mô hình thi công. 28](#_Toc76332394)

[4.2. Đánh giá. 30](#_Toc76332395)

[CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 31](#_Toc76332396)

[5.1. Kết luận. 31](#_Toc76332397)

[5.2. Hướng phát triển. 31](#_Toc76332398)

[TÀI LIỆU THAM KHẢO 32](#_Toc76332399)

[PHỤ LỤC 33](#_Toc76332400)

# **CHƯƠNG 1: GIỚI THIỆU**

* 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 trông 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. **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 luống riêng biệt và có các chức năng:

* Luống ươm cây cà phê giống:

Tự động tưới nước ở 60ºC với chu kỳ 1 giờ/lần, mỗi lần tưới 10 phút liên tục trong 15 ngày.

* Luống ươm cây cà phê con:
* 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.
* Đo cường độ sáng và kéo màn che khi cường độ sáng dưới ngưỡng
  1. **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
* Che màn để giảm cường đồ ánh sáng trực tiếp vào cây
* Tưới nước ở 60ºC cho luống ươm hạt giống
* 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 35ºC 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 cảm biến ánh sáng để khi cường độ sáng dưới 20 lux thì tự động kéo màn che
* Sử dụng bộ điều khiển nhiệt độ để giữ nhiệt độ nước ở 60ºC
* Sử dụng mạch thời gian thực để cài đặt thời gian tưới nước
* Sử dụng Bluetooth để kết nối với điện thoại.
  1. **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. **Đối tượng và phạm vi nguyên cứu.**
* Đối tượng nguyên cứu
* Cây cà phê con
* Mầm ươm cây cà phê
* 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 Lâm Đồng)
  1. **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 nguyên 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).

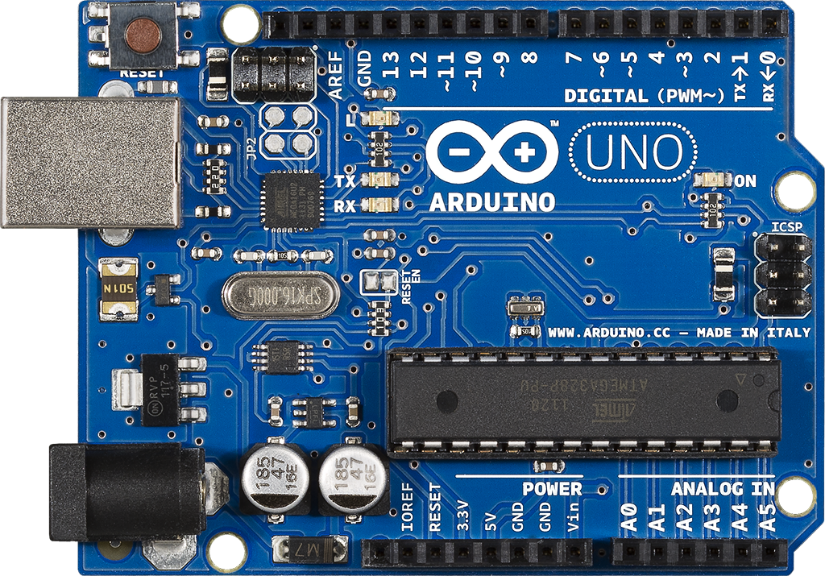
# **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

1. **Giới thiệu phần cứng.**

### **2.1.1. Board Arduino Uno R3 ATmega328**

Arduino Uno R3 - ATmega328 là dòng Arduino thế hệ thứ 3, cũng giống như các phiên bản trước đây giúp người dùng dễ dàng tiếp cận với lập trình để tạo ra phần cứng có những tính năng mong muốn một cách nhanh chóng với chi phí hợp lý.

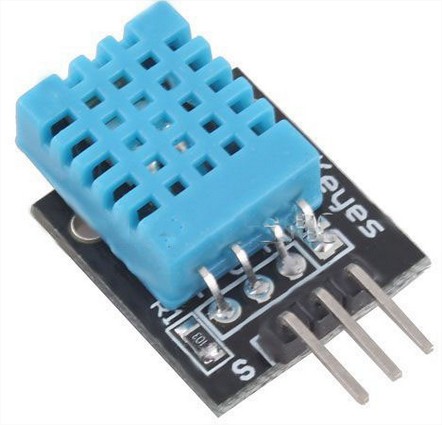
Nó là một bo mạch vi điều khiển dựa trên ATmega328, có 14 chân đầu vào / đầu ra kỹ thuật số (trong đó 6 chân có thể được sử dụng làm đầu ra PWM), 6 đầu vào analog, bộ cộng hưởng 16 MHz, kết nối USB, giắc cắm nguồn, đầu cắm ICSP và nút reset. Mạch chứa mọi thứ cần thiết để hỗ trợ vi điều khiển, chỉ cần kết nối thiết bị với máy tính bằng cáp USB hoặc cấp nguồn từ 7 đến 12VDC để bắt đầu.



**Hình 2.1** Board Arduino Uno R3 – Atmega323

### **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.

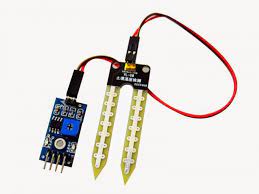


**Hình 2.2** Module cảm biến nhiệt độ, độ ẩm DHT11

### **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.

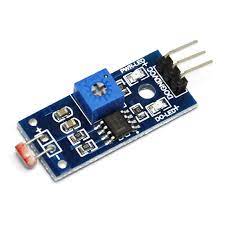
Cảm biến bao gồm một biến trở để đặt ngưỡng độ ẩm mong muốn. Đầu ra kỹ thuật số có thể được kết nối với một bộ điều khiển vi mô để cảm nhận mức độ ẩm. Cảm biến cũng xuất ra một đầu ra tương tự có thể được kết nối với ADC của bộ điều khiển vi mô để có được mức độ ẩm chính xác trong đất, phù hợp để thực 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 cảm biến ánh sáng**

Module cảm biến ánh sáng quang trở 3 chân sử dụng bộ so sánh điện áp LM393 chất lượng cao. Dễ dàng lắp đặt bằng cách sử dụng cảm biến điện trở cảm quang loại nhạy, tín hiệu đầu ra của bộ so sánh cho dạng sóng tốt và ổn định. Có thể điều chỉnh độ nhạy thông qua chiết áp, được ứng dụng trong nhiều lĩnh vực khác nhau như: điều khiển bật tắt thiết bị theo ánh sáng, hệ thống cảnh báo chống trộm, đèn led chiếu sáng tự động…

****

**Hình 2.4** Module cảm biến ánh sáng

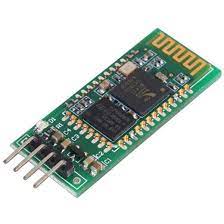
### **2.1.5. Module bluetooth**

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 chê 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 2.5** Module bluetooth HC – 05

### **2.1.6. Module thời gian thực**

Mạch đồng hồ thời gian thực DS3231 là module đồng hồ thời gian thực giao tiếp qua chuẩn I2C và có ưu điểm là cực kỳ chính xác với bộ dao động thạch anh nội có tính năng bù nhiệt. So sánh với DS1302 thì DS3231 có nhiều ưu điểm hơn hẳn, với khả năng hoạt động điện áp từ 2.3V đến 5.5V và sử dụng pin ngoài. Không giống như DS1307, DS3231 dùng thạch anh tích hợp bên trong giúp cho thời gian chạy chính xác hơn, có cảm biến nhiệt độ.

Bên cạnh đó, module còn có tích hợp EEPROM AT24C32 trên board, hỗ trợ đến 32KB EEPROM cho việc lưu trữ dữ liệu, module DS3231 là sự lựa chọn tốt cho việc ghi nhớ thời gian.



**Hình 2.6** Module thời gian thực RTC DS3231

### **2.1.7. Mạch điều khiển động cơ bước**

Mạch điều khiển động cơ bước ULN2003 có bo mạch sử dụng IC chính ULN2003 để điều khiển động cơ bước 4 pha 5 dây (5v-12v) có nhiệm vụ thu kéo màn che.

Mạch sử dụng chất liệu và linh kiện điện tử chất lượng cao, đảm bảo độ bền khi sử dụng, có thể lập trình dễ dàng điều khiển tốc độ / hướng quay và chế độ làm việc của động cơ bước.

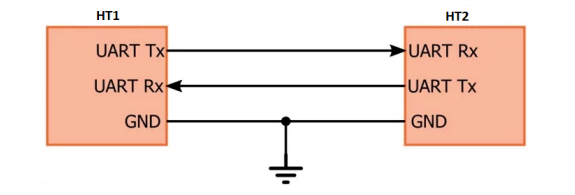


**Hình 2.7** Mạch điều khiển động cơ bước ULN2003

1. **Chuẩn truyền dữ liệu và chuẩn kết nối.**

### **2.2.1. Chuẩn giao tiếp 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 ...



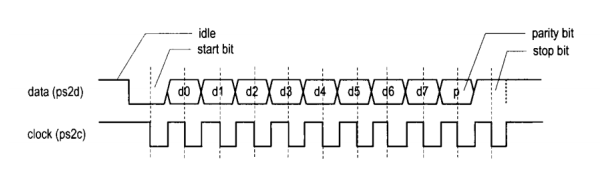
**Hình 2.8** Hệ thống truyền dữ liệu bất đồng bộ

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 1/9600~104.167us.

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 2.9** 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).

### **2.2.2. Chuẩn giao tiếp I2C**

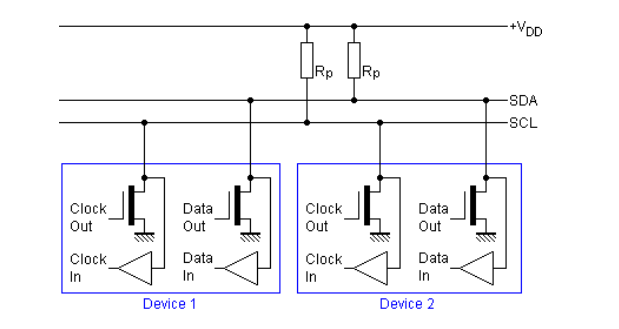
I2C, viết tắt của Inter-Integrated Circuit, là một loại bus nối tiếp được phát triển bởi hãng sản xuất linh kiện điện tử Philips. Ban đầu, loại bus này chỉ được dùng trong các linh kiện điện tử của Philips. Sau đó, do tính ưu việt và đơn giản của nó, I2C đã được chuẩn hóa và được dùng rộng rãi trong các mô đun truyền thông nối tiếp của vi mạch tích hợp ngày nay.

Đây là một loại giao thức giao tiếp nối tiếp đồng bộ. Nó có nghĩa là các bit dữ liệu được truyền từng bit một theo các khoảng thời gian đều đặn được thiết lập bởi một tín hiệu đồng hồ tham chiếu.

*Cấu tạo và nguyên lý hoạt động*

I2C sử dụng hai đường truyền tín hiệu:

* Một đường xung nhịp đồng hồ (SCL) chỉ do Master phát đi (thông thường ở 100kHz và 400kHz. Mức cao nhất là 1Mhz và 3.4MHz).
* Một đường dữ liệu (SDA) theo 2 hướng.



**Hình 2.10** Sơ đồ kết nối chuẩn I2C

Có rất nhiều thiết bị có thể cùng được kết nối vào một bus I2C, tuy nhiên sẽ không xảy ra chuyện nhầm lẫn giữa các thiết bị, bởi mỗi thiết bị sẽ được nhận ra bởỉ một địa chỉ duy nhất với một quan hệ chủ/tớ tồn tại trong suốt thời gian kết nối. Mỗi thiết bị có thể hoạt động như là thiết bị nhận hoặc truyền dữ liệu hay có thể vừa truyền vừa nhận. Hoạt động truyền hay nhận còn tùy thuộc vào việc thiết bị đó là chủ (master) hay tớ (slave).

Một thiết bị hay một IC khi kết nối với bus I2C, ngoài một địa chỉ (duy nhất) để phân biệt, nó còn được cấu hình là thiết bị chủ hay tớ vì trên một bus I2C thì quyền điều khiển thuộc về thiết bị chủ. Thiết bị chủ nắm vai trò tạo xung đồng hồ cho toàn hệ thống, khi giữa hai thiết bị chủ - tớ giao tiếp thì thiết bị chủ có nhiệm vụ tạo xung đồng hồ và quản lý địa chỉ của thiết bị tớ trong suốt quá trình giao tiếp. Thiết bị chủ giữ vai trò chủ động, còn thiết bị tớ giữ vai trò bị động trong việc giao tiếp.

Điểm mạnh của I2C chính là hiệu suất và sự đơn giản của nó: một khối điều khiển trung tâm có thể điều khiển cả một mạng thiết bị mà chỉ cần hai lối ra điều khiển.

# **CHƯƠNG 3: THIẾT KẾ HỆ THỐNG**

## **3.1. Thiết kế phần cứng.**

### **3.1.1. Sơ đồ khối.**

**KHỐI NGUỒN**

**KHỐI HIỂN THỊ**

**KHỐI CẢM BIẾN**

**KHỐI XỬ LÝ TRUNG TÂM**

**KHỐI THIẾT BỊ NGOẠI VI**

**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, độ ẩm không khí, cảm biến độ ẩm đất, cảm biến ánh sáng có tính chính xác cao, dùng để thu thập dữ liệu về nhiệt độ, đổ ẩm, độ sáng xung quanh 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 Uno R3 ATmega328 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 hổ trợ cho các module chức năng khác nhau, trình biên dịch đơn giản, dễ sử dụng.

Thông số kỹ thuật:

* Vi điều khiển chính: ATmega328
* Điện áp hoạt động:5VDC
* Điện áp vào: 7~12VDC
* Điện áp vào giới hạn: 6~20VDC
* Số chân Digital: 14 (6 chân PWM)
* Số chân vào Analog: 6
* Dòng DC trên mỗi chân: 40mA
* Dòng DC trên chân 3.3V: 50mA
* Bộ nhớ Flash: 32 KB (0.5KB dùng cho bootloader)
* SRAM: 2 KB - EEPROM: 1KB
* Tần số xung clock: 16 MHz
* **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 dung 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)

+ Dãi độ ẩm hoạt động: 20% – 90% RH, sai số ±5%RH

+ Dãi nhiệt độ hoạt động: 0°C ~ 50°C, sai số ±2°C

+ Khoảng cách truyển tối đa: 20m

* Module cảm biến ánh sáng

Thông số kỹ thuật:

+ Dùng IC so sánh LM393 cho dòng ra lớn đến 15mA.

+ Nguồn cấp 3.3 ~ 5 VDC

+ Đầu ra digital DO (0 và 1)

+ Đầu ra điện áp analog AO

* Module thời gian thực RTC DS3231

Thông số kỹ thuật:

+ Điện áp hoạt động: 3.3~5.5VDC

+ Chip đồng hồ: IC DS3231

+ Độ chính xác đồng hồ: trong phạm vi 0~40 độ C, độ chính xác là 2ppm

+ Đồng hồ thời gian thực, phút, giờ, ngày, tháng, năm và có giá trị cho đến 2100, có tính năm nhuận

+ Cảm biến nhiệt độ với độ chính xác ± 3℃

+ Chip nhớ: AT24C32 (dung lượng lưu trữ 32K)

+ Với pin sạc LIR2032, để đảm bảo hệ thống vẫn lưu các thông số sau khi mất điện

* **Khối thiết bị ngoại vi**
* Module Relay 5V kích mức cao

Module Relay 5VDC có kích thước nhỏ gọn, giá thành rẻ được sử dụng để đóng ngắt thiết bị AC hoặc DC, module sử dụng điện áp 5VDC với chỉ 3 chân kết nối dễ dàng sử dụng.Có sẵn header rất tiện dụng khi kết nối với vi điều khiển.

Thông số kỹ thuật:

+ Điện áp hoạt động: 5VDC

+ Relay tiêu thụ dòng khoảng 70mA.

+ Điện thế đóng ngắt tối đa: AC250V ~ 10A hoặc DC30V ~ 10A.

+ Tích hợp Diod chống nhiễu và đèn báo tín hiệu kích.

* Mạch điều khiển động cơ bước ULN2003

Thông số kỹ thuật:

+ Điện áp: 5 ~ 12 VDC

+ Tín hiệu ngõ vào: 4 chân In 1; In 2; In 3 : In 4

+ Tín hiệu ngõ ra: Jack cắm động cơ bước 28BYJ-48

+ 4 Led hiện thị trạng thái hoạt động của động cơ

* 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℃

+ 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

* **Khối 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

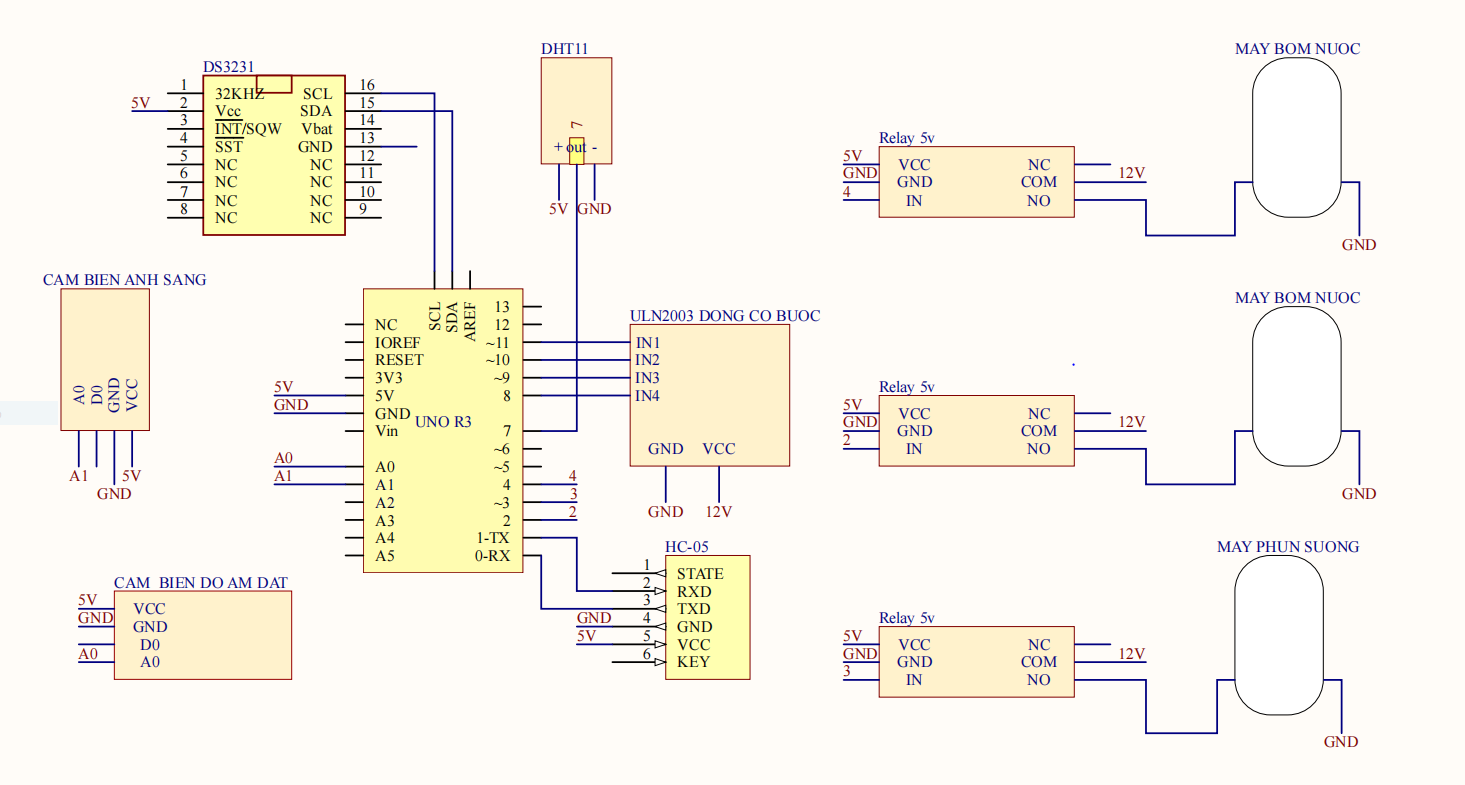
+ Nguồn: 3.3 ~ 6V

* **Khối nguồn**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên linh kiện | Số lượng | Điện áp (V) | Dòng điện (A) | Công suất (W) |
| 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 |
| Module cảm biến ánh sáng | 1 | 5 | 0.04 | 0.2 |
| Module thời gian thực | 1 | 5 | 0.04 | 0.2 |
| Bluetooth HC-05 | 1 | 5 | 0.04 | 0.2 |
| Module Relay | 3 | 5 | 0.04 | 0.6 |
| Mạch ULN2003 | 1 | 5 | 0.16 | 0.8 |
| Máy bơm nước | 2 | 12 | 0.5 | 12 |
| Máy phun sương | 1 | 12 | 2 | 24 |
| Động cơ bước | 1 | 12 | 1 | 12 |
| Tổng cộng: 50.4W | | | | |

Bảng tổng hợp công suất của các linh kiện trong hệ thống

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 50.4W.



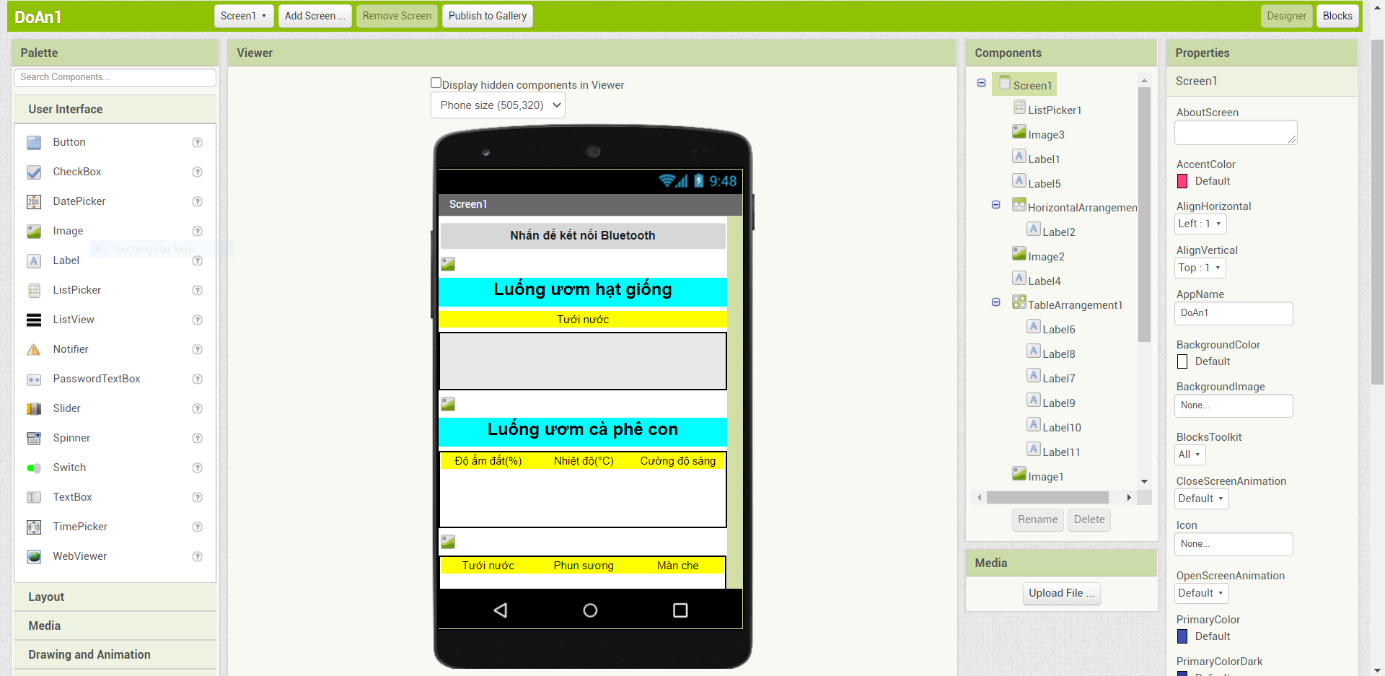
**Hình 3.2** Sơ đồ nguyên lý hệ thống

## **3.2. Thiết kế phần mềm.**

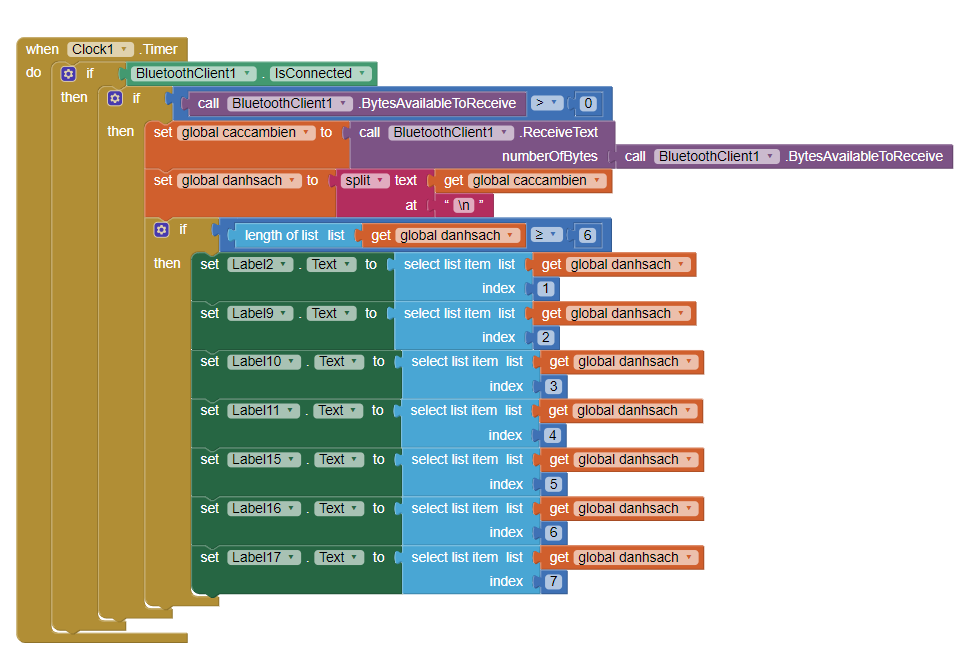
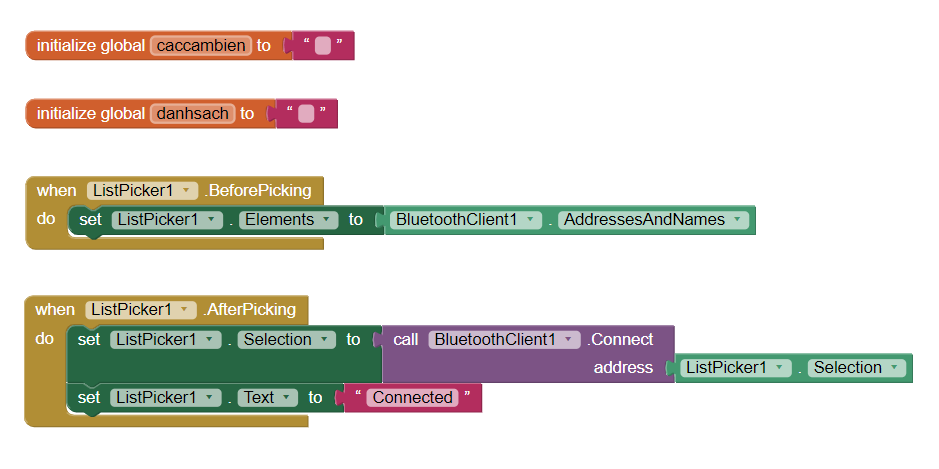
### **3.2.1. Phần mềm lập trình Arduino**

(Code chương trình ở phần phụ lục)

### **3.2.2. Phần mềm lập trình ứng dụng điện thoại**



**Hình 3.3** Thiết kế giao diện ứng dụng trên điện thoại



**Hình 3.4** Các block của ứng dụng

### **3.2.3. Lưu đồ hoạt động.**

Bắt đầu

Khởi tạo thư viện

Dữ liệu từ cảm biến

So sánh với giá trị mong muốn

Sai

Đúng

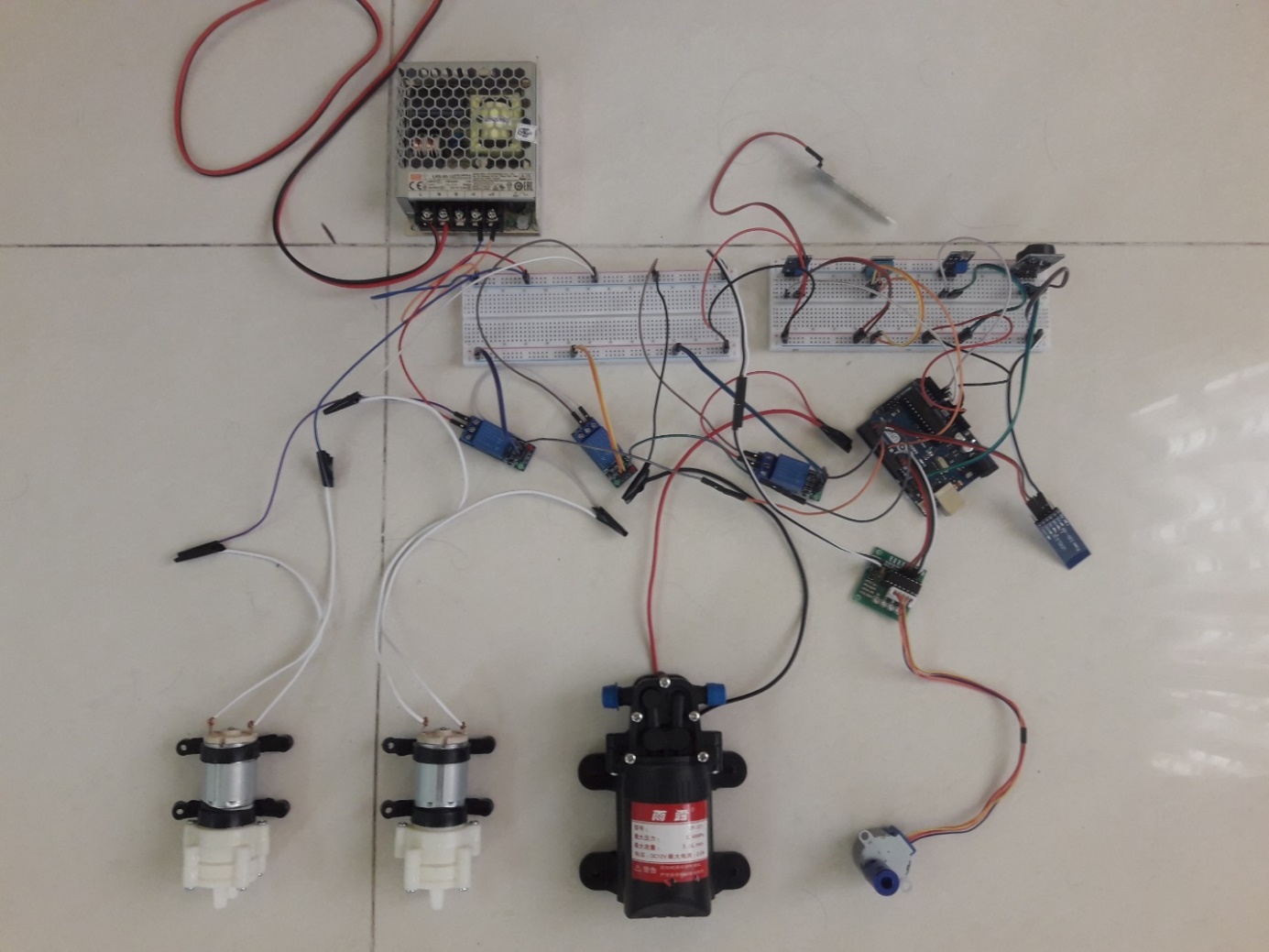
Tắt thiết bị

Bật thiết bị

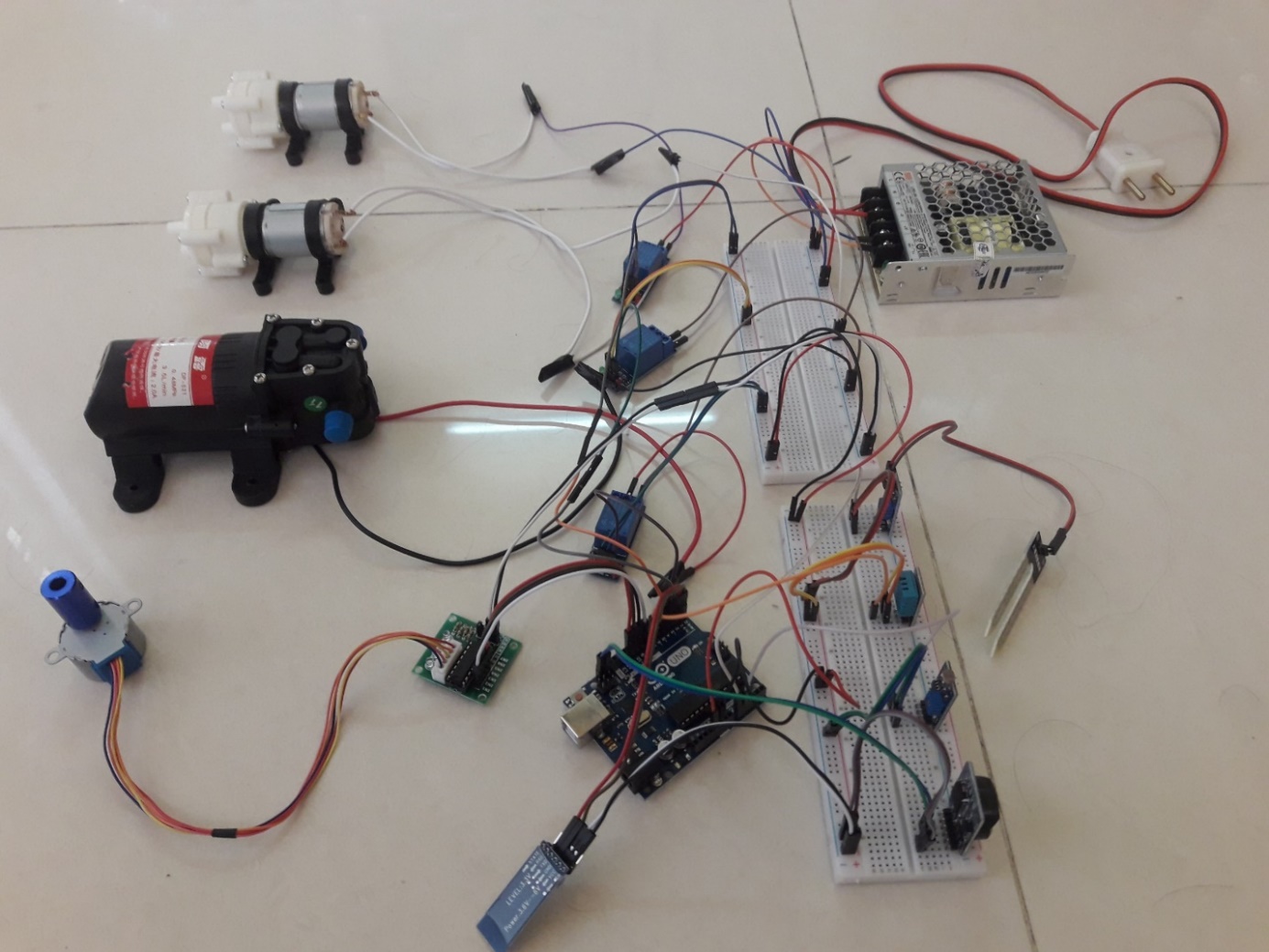
**Hình 3.5** Lưu đồ hoạt động

# **CHƯƠNG 4: KẾT QUẢ - ĐÁNH GIÁ**

## **4.1. Kết quả mô hình thi công.**



**Hình 4.1** Hình ảnh thực tế phần cứng sản phẩm



**Hình 4.2** Hình ảnh thực tế phần cứng sản phầm

**Hình 4.3** Giao diện ứng dụng trên Android

## **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 phải mất vài giây dữ liệu mới được 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 qua 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, cảm biến ánh sáng.

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

# **TÀI LIỆU THAM KHẢO**

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

[2] *http://arduino.vn/bai-viet/1053-giao-tiep-i2c-voi-nhieu-module*

[3]*https://www.makerguides.com/28byj-48-stepper-motor-arduino-tutorial/?fbclid=IwAR06M35Pfkk\_CGJXTlNE-DvkpWBe9BWVNBXCGqfGomLMR9D97DXqfVAss7A*

[4]*https://appinventor.mit.edu/explore/ai2/tutorials?fbclid=IwAR3TCxKp0\_Kt5pLzFIcHbGd6yGRVpF0DqMSkqpythKF3wunZ6DyTP4XzmeE*

[5]*https://github.com/hieucao/tuoicay-tudong-arduino?fbclid=IwAR1neHPYCUKI\_uUkcUkppWMUw8XotOo7SLvR9qVI\_BjAsycJUw00koRPiuE*

# **PHỤ LỤC**

Code Arduino:

#include "DHT.h"

#include <AccelStepper.h>

#include <Wire.h>

#include <ds3231.h>

#define motorPin1 8

#define motorPin2 9

#define motorPin3 10

#define motorPin4 11

const int DHTPIN = 7;

const int DHTTYPE = DHT11;

int relaydat = 2;

int cbdat = A0;

int doc\_cb, TBcb;

int relaykk = 3;

DHT dht(DHTPIN, DHTTYPE);

#define MotorInterfaceType 8

int sang = 0;

AccelStepper stepper = AccelStepper(MotorInterfaceType, motorPin1, motorPin3, motorPin2, motorPin4);

int relayds3231 = 4;

struct ts t;

void setup()

{

Serial. begin (9600);

pinMode (cbdat, INPUT);

pinMode (relaydat, OUTPUT);

digitalWrite (relaydat, LOW);

dht.begin(); pinMode(A1, INPUT);

stepper.setMaxSpeed(1000);

pinMode (relayds3231, OUTPUT);

Wire.begin();

DS3231\_init(DS3231\_CONTROL\_INTCN);

t.hour=12;

t.min=30;

t.sec=0;

t.mday=25;

t.mon=12;

t.year=2019;

DS3231\_set(t);

}

void loop()

{

for(int i=0;i<=9;i++)

{

doc\_cb += analogRead(cbdat);

}

TBcb=doc\_cb/10;

int phantramao = map(TBcb, 0, 1023, 0, 100);

int phantramthuc = 100 - phantramao;

doc\_cb=0;

if ( phantramthuc >= 30 )

{

digitalWrite (relaydat, LOW);

}

else

{

digitalWrite (relaydat, HIGH);

}

int h = dht.readHumidity();

int t1 = dht.readTemperature();

if ( t1 < 40 )

{

digitalWrite (relaykk, LOW);

}

else

{

digitalWrite (relaykk, HIGH);

}

unsigned int value = analogRead(A1);

if (sang == 0)

{

if ( value < 15 )

{

sang = 1;

Serial.println(sang);

stepper.setCurrentPosition(0);

while (stepper.currentPosition() != 4096) {

stepper.setSpeed(1000);

stepper.runSpeed(); }

}

}

else if (sang==1){

if (value > 50){

sang = 0;

Serial.println(sang);

stepper.setCurrentPosition(0);

while (stepper.currentPosition() != -4096) {

stepper.setSpeed(-1000);

stepper.runSpeed(); }

}}

DS3231\_get(&t);

if ( (t.min % 1 ==0) && (t.sec < 15))

{ digitalWrite (relayds3231, HIGH);

Serial.println("ON");

}

else

{

digitalWrite (relayds3231, LOW);

Serial.println("OFF");

}

Serial.println(phantramthuc);

Serial.println(t1);

Serial.println(value);

if ( phantramthuc >= 30 )

{

Serial.println("OFF");

}

else

{

Serial.println("ON");

}

if ( t1 < 40 )

{

Serial.println("OFF");

}

else

{

Serial.println("ON");

}

if (sang == 0)

{

Serial.println("OFF");

}

else if (sang==1)

{

Serial.println("ON");

}

delay(500);

}