MỤC LỤC

[TÓM TẮT 3](#_Toc30074902)

[1. GIỚI THIỆU 5](#_Toc30074903)

[1.1. App Inventor 5](#_Toc30074904)

[1.2. Arduino UNO R3 6](#_Toc30074905)

[1.3. Module HC05 8](#_Toc30074906)

[1.5. Phần mềm thiết kế mạch và mô phỏng (Proteus version 7.10SP2) 10](#_Toc30074907)

[1.6. Rờ le (Relay) 11](#_Toc30074908)

[1.7. IC Lập trình Atmega16u2 12](#_Toc30074909)

[1.8. Phần mềm Sprint Layout Viewer 14](#_Toc30074910)

[2. Phương pháp nghiên cứu 15](#_Toc30074911)

[2.1. Thi công,thiết kế mạch in với Sprint Layout 15](#_Toc30074912)

[2.2. Mô phỏng phần mềm Proteus 16](#_Toc30074913)

[2.3. Thiết kế giao diện 17](#_Toc30074914)

[2.4. Lập trình chức năng 19](#_Toc30074915)

[2.4. Giao tiếp Arduino với HC05 20](#_Toc30074916)

[2.5. Điều khiển thiết bị với Arduino 20](#_Toc30074917)

[3. KẾT QUẢ 20](#_Toc30074918)

[3.1. Thi công mạch in 20](#_Toc30074919)

[3.2. Giao diện phần mềm điều khiển 20](#_Toc30074920)

[3.3. Điều khiển thiết bị 21](#_Toc30074921)

[4. KẾT LUẬN 22](#_Toc30074922)

[PHỤ LỤC 23](#_Toc30074923)

**DANH MỤC HÌNH**

[Hình 1.1. Giao diện app inventor 6](#_Toc30074747)

[Hình 1.2. Arduino UNO R3 6](#_Toc30074748)

[Hình 1.3. Chi tiết cấu tạo Arduino 7](#_Toc30074749)

[Hình 1.5. Sơ đồ chân của HC05 9](#_Toc30074750)

[Hình 1.6. Giao diện Arduino IDE 10](#_Toc30074751)

[Hình 1.7. Giao diện proteus 10](#_Toc30074752)

[Hình 1.8. Rờ le 5VDC 12](#_Toc30074753)

[Hình 1.9. Sơ đồ chân IC Atmega16u2 13](#_Toc30074754)

[Hình 1.10. Sơ đồ nguyên lý board mạch Arduino UNO R3 14](#_Toc30074755)

[Hình 1.11. Giao diện sprint layout 14](#_Toc30074756)

[Hình 2.1. Các bước thiết kế mạch in 15](#_Toc30074757)

[Hình 2.2. Sơ đồ nguyên lý hoạt động điều khiển thiết bị qua bluetooth trong proteus 16](#_Toc30074758)

[Hình 2.3. Sơ đồ mô phỏng khi tải kết nối với quạt không có dòng điện chạy qua 16](#_Toc30074759)

[Hình 2.4. Sơ đồ mô phỏng khi tải kết nối với quạt có dòng điện chạy qua 17](#_Toc30074760)

[Hình 2.5. giao diện thiết kế 17](#_Toc30074761)

[Hình 2.6. Giao diện chức năng và thiết kế 18](#_Toc30074762)

[Hình 2.7. Các block chức năng 19](#_Toc30074763)

[Hình 2.8. Các block chức năng 19](#_Toc30074764)

[Hình 3.1. Sản phẩm sau khi thiết kế 20](#_Toc30074765)

[Hình 3.2. Giao diện phần mềm 20](#_Toc30074766)

[Hình 3.3. Điều khiển thiết bị bằng điện thoại thông qua bluetooth 21](#_Toc30074767)

DANH MỤC BẢNG

[Bảng 1.1. Thông số kỹ thuật 8](#_Toc29993018)

TÓM TẮT

Hiện nay các ứng dụng thông minh trên điện thoại thông minh ngày càng phổ biến, hệ điều hành android ngày càng được sử dụng rộng rãi và phổ biến.Android được xây dựng và phát triển liên tục với các chia sẻ về mã nguồn mở, việc sử dụng thiết bị smartphone để điều khiển và giám sát các thiết bị hiện nay đã trở thành xu hướng. Vì thế em quyết định chọn đề tài: “Điều khiển thiết bị sử dụng điện thoại thông qua Bluetooth sử dụng Kit Arduino”

1. GIỚI THIỆU

1.1. App Inventor

Ngày 12/07/2010, Google chính thức giới thiệu công cụ lập trình trực quan App Inventor dùng để phát triền phần mềm ứng dụng trên hệ điều hành Android. App Inventor là công cụ lập trình dành cho mọi người. Với chương trình này, bạn không cần phải có bất cứ kiến thức nào về lập trình cũng có thể tạo các ra ứng dụng cho Android từ cơ bản đến nâng cao bằng những thao tác kéo-thả.

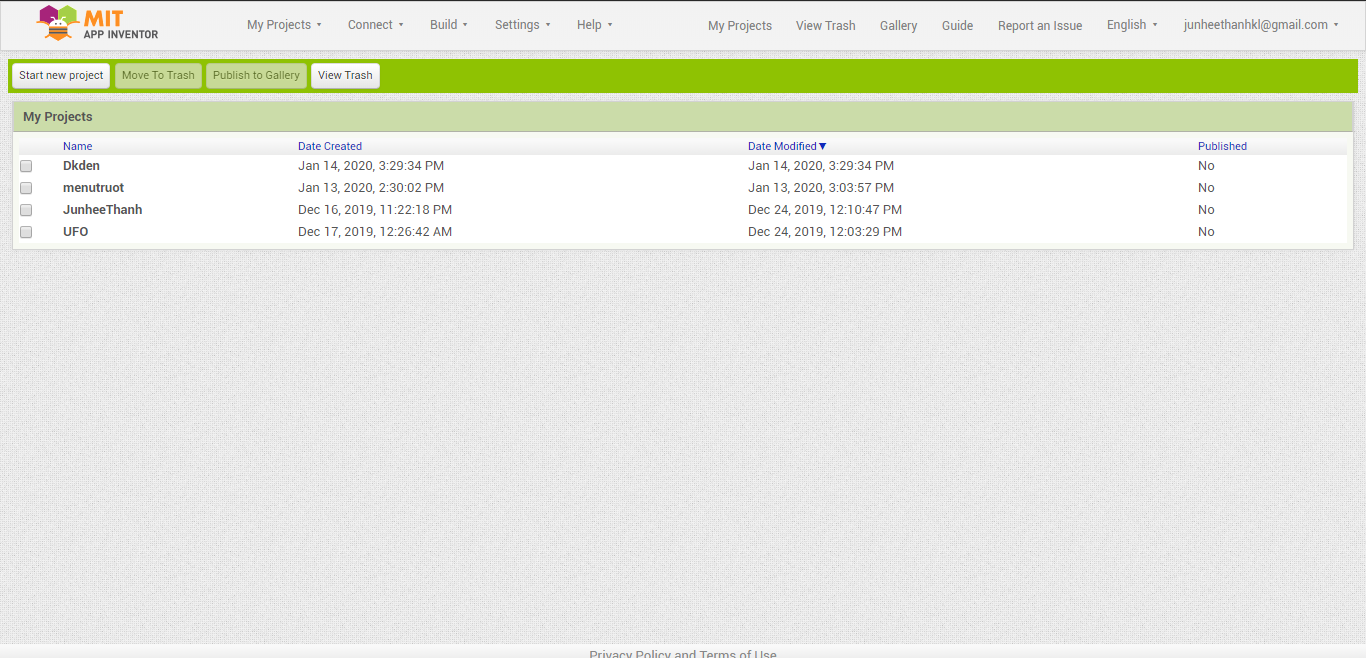
MIT App Inventor dành cho Android là một ứng dụng web nguồn mở ban đầu được cung cấp bởi Google và hiện tại được duy trì bởi Viện Công nghệ Massachusetts (MIT).

Nền tảng cho phép nhà lập trình tạo ra các ứng dụng phần mềm cho hệ điều hành Android (OS). Bằng cách sử dụng giao diện đồ họa, nền tảng cho phép người dùng kéo và thả các khối mã (blocks) để tạo ra các ứng dụng có thể chạy trên thiết bị Android. Đến thời điểm hiện tại 07/2017, phiên bản iOS của nền tảng này đã bắt đầu được đưa vào thử nghiệm bởi Thunkable, là một trong các nhà cung cấp ứng dụng web cho ngôn ngữ này.

Nền tảng App Inventor được đưa ra thông qua yêu cầu vào ngày 12 tháng 7 năm 2010 và được phát hành công khai vào ngày 15 tháng 12 năm 2010. Nhóm App Inventor được dẫn dắt bởi Hal Abelson và Mark Friedman. Trong nửa sau của năm 2011, Google công bố mã nguồn, chấm dứt máy chủ và cung cấp tài trợ cho việc thành lập Trung tâm Nghiên cứu Điện thoại Di động MIT, do Hal Abelson và các giáo sư Eric Klopfer và Mitchel Resnick sáng lập để duy trì hoạt động của App Inventor. Phiên bản MIT được ra mắt vào tháng 3 năm 2012.

Vào ngày 6 tháng 12 năm 2013, MIT đã phát hành App Inventor 2, đổi tên từ tên gốc "App Inventor Classic". Tính đến tháng 5 năm 2014, nền tảng đã có 87 nghìn người dùng hoạt động hàng tuần và 1,9 triệu đăng ký từ 195 quốc gia với tổng số 4,7 triệu ứng dụng được xây dựng.Vào tháng 12 năm 2015, có 140 nghìn người sử dụng hoạt động hàng tuần và 4 triệu đăng ký tại 195 quốc gia với tổng số 12 triệu ứng dụng được xây dựng.

Hiện nay, mỗi tháng MIT App Inventor có hơn 400.000 người dùng đến từ 195 quốc gia và đã tạo ra gần 22 triệu ứng dụng, MIT App Inventor đang thay đổi cách thế giới tạo ra các ứng dụng và cách mà học sinh nhỏ tuổi bắt đầu học về máy tính.

**

Hình 1.1. Giao diện app inventor

1.2. Arduino UNO R3



Hình 1.2. Arduino UNO R3

*(Nguồn:* [***http://ictamky.com/product/arduino-uno-r3/***](http://ictamky.com/product/arduino-uno-r3/)*)*

Arduino là một nền tảng nguyên mẫu (mã nguồn mở) dựa trên nền phần mềm và phần cứng dễ sử dụng. Nó bao gồm một bo mạch - thứ mà có thể được lập trình (đang đề cập đến vi điều khiển) và một phần mềm hỗ trợ gọi là Arduino IDE (Môi trường phát triển tích hợp cho Arduino), được sử dụng để viết và nạp từ mã máy tính sang bo mạch vật lý.

**Những tính năng chính như:**

Các bo mạch Arduino có khả năng đọc các tín hiệu tương tự (analog) hoặc tín hiệu số (digital) làm đầu vào từ các cảm biến khác nhau và chuyển nó thành đầu ra như kích hoạt mô-tơ quay, bật / tắt đèn LED, kế nối mạng Internet hoặc nhiều hoạt động khác nữa.

Bạn có thể điều khiển các chức năng của bo mạch của mình bằng cách nạp các tập lệnh đến vi điều khiển trên bo mạch. Thông qua phần mềm hỗ trợ là Arduino IDE.

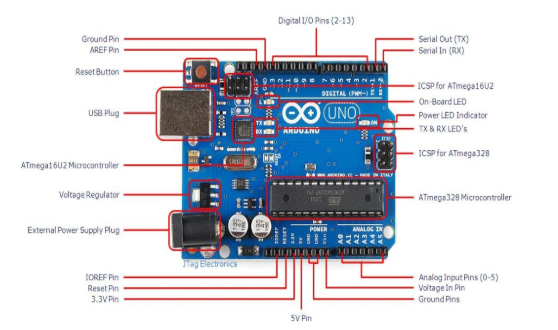
Không giống như bo mạch có khả năng lập trình trước kia, Arduino chỉ cần bạn sử dụng cáp USB để nạp mã vào trong bo mạch.

Hơn nữa, phần mềm Arduino IDE sử dụng phiên bản giản thể của C++, làm việc học lập trình nó trở nên dễ dàng hơn rất nhiều.

**Cấu tạo:**

Arduino Uno R3là một bo mạch vi điều khiển dựa trên chip Atmega328P. Uno có 14 chân I/O digital ( trong đó có 6 chân xuất xung PWM), 6 chân Input analog, 1 thạch anh 16MHz, 1 cổng USB, 1 jack nguồn DC, 1 nút reset.

Uno hỗ trợ đầy đủ những thứ cần thiết để chúng ta có thể bắt đầu làm việc.

Hình 1.3. Chi tiết cấu tạo Arduino

*(Nguồn:* [*http://mlab.vn/10609-hoc-arduino-bai-2-nhung-dieu-co-ban-ve-arduino.html*](http://mlab.vn/10609-hoc-arduino-bai-2-nhung-dieu-co-ban-ve-arduino.html)***)***

Bảng 1.1. Thông số kỹ thuật

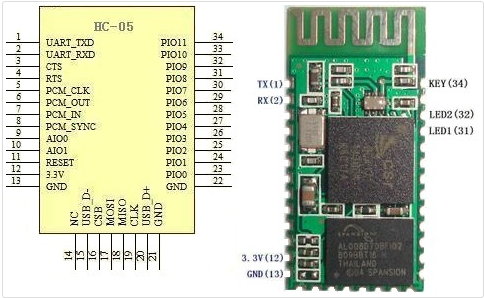
|  |  |
| --- | --- |
| Vi điều khiển | ATmega328 họ 8bit |
| Điện áp hoạt động | 5V DC (chỉ được cấp qua cổng USB) |
| Tần số hoạt động | 16 MHz |
| Dòng tiêu thụ | khoảng 30mA |
| Điện áp vào khuyên dùng | 7-12V DC |
| Điện áp vào giới hạn | 6-20V DC |
| Số chân Digital I/O | 14 (6 chân hardware PWM |
| Số chân Analog | 6 (độ phân giải 10bit) |
| Dòng tối đa trên mỗi chân I/O | 30 mA |
| Dòng ra tối đa (5V) | 500 mA |
| Dòng ra tối đa (3.3V) | 50 mA |
| Bộ nhớ flash | 32 KB (ATmega328) với 0.5KB dùng bởi bootloader |
| SRAM | 2 KB (ATmega328) |
| EEPROM | 1 KB (ATmega328) |

1.3. Module HC05

Module bluetooth HC05 master / slave dùng để thiết lập kết nối Serial giữa 2 thiết bị bằng sóng bluetooth. ... + Ở chế độ MASTER: module sẽ tự động dò tìm thiết bị bluetooth khác.

**Cấu tạo**

**Sơ đồ chân:**



Hình 1.5. Sơ đồ chân của HC05

*(Nguồn:*[*https://www.semiconvn.com/home/hoc-thiet-ke-vi-mach/bai-hc-vi-mch/9218-tim-hieu-va-su-dung-module-bluetooth-hc05.html*](https://www.semiconvn.com/home/hoc-thiet-ke-vi-mach/bai-hc-vi-mch/9218-tim-hieu-va-su-dung-module-bluetooth-hc05.html)***)***

Điện áp hoạt động: 3,3V, module có 2 chế độ làm việc(có thể lựa chọn chế độ làm việc bằng cách thay đổi trạng thái chân 34 KEY), tự động kết nối ,đáp ứng theo lệnh: khi làm việc ở chế độ này, các bạn có thể gửi các lệnh AT để giao tiếp với module.

Module HC05 có thể nhận 1 trong 3 chức năng: Master, Slave, Loopback(có thể lựa chọn các chức năng bằng lệnh AT).

Giao tiếp với module bằng giao tiếp nối tiếp không đồng bộ qua 2 đường RX và TX, vì vậy có thể sử dụng PC với chuẩn RS232 hoặc các dòng vi điều khiển để giao tiếp.

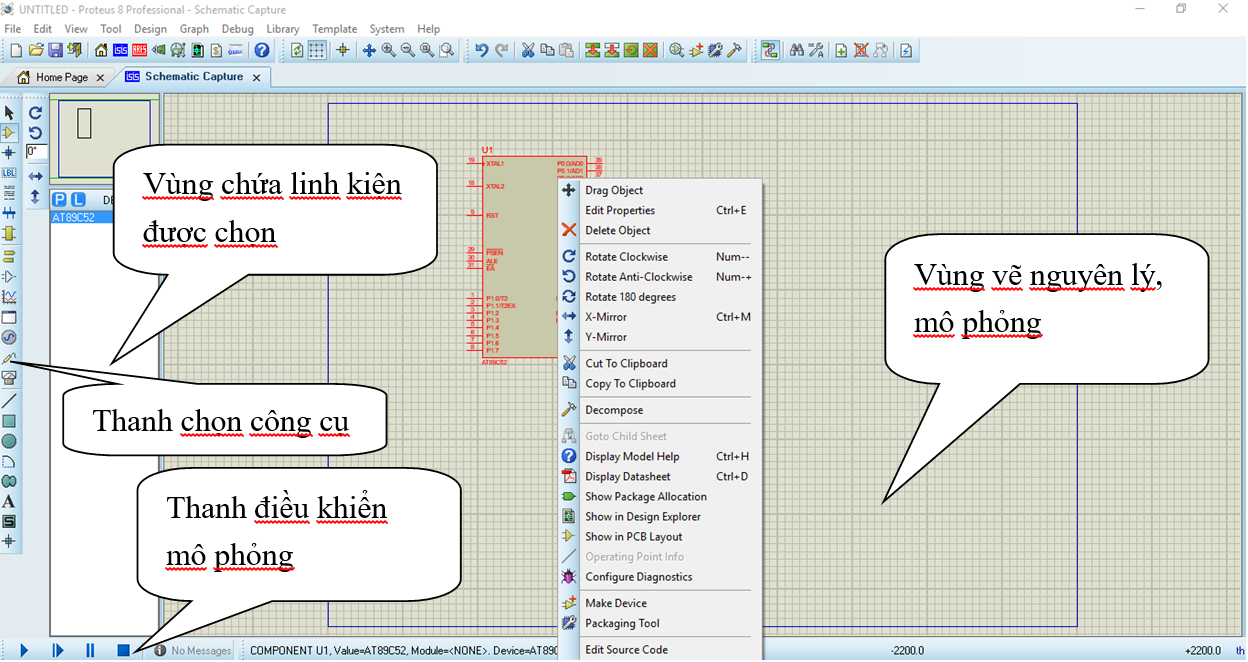
1.4. Arduino IDE.

**Arduino là môi trường phát triển tích hợp mã nguồn mở, cho phép người dùng dễ dàng viết code và tải nó lên bo mạch. Môi trường phát triển được viết bằng Java dựa trên ngôn ngữ lập trình xử lý và phần mềm mã nguồn mở khác. Phần mềm này có thể được sử dụng với bất kỳ bo mạch Arduino nào.**



Hình 1.6. Giao diện Arduino IDE

1.5. Phần mềm thiết kế mạch và mô phỏng (Proteus version 7.10SP2)



Hình 1.7. Giao diện proteus

*(Nguồn:*[*http://svcokhi.blogspot.com/2016/06/phan-mem-mo-phong-mach-ien-tu-proteus.html#.Xh\_Jo8gzbDc*](http://svcokhi.blogspot.com/2016/06/phan-mem-mo-phong-mach-ien-tu-proteus.html#.Xh_Jo8gzbDc)***)***

Proteus là phần mềm cho phép mô phỏng hoạt động của mạch điện tử bao gồm phần thiết kế mạch và viết chương trình điều khiển cho các họ vi điều khiển như MCS-51, PIC, AVR,…

Proteus là phần mềm mô phỏng mạch điện tử của Lancenter Electronics, mô phỏng cho hầu hết các linh kiện điện tử thông dụng, đặc biệt hỗ trợ cho cả các MCU như PIC, 8051, AVR, Motorola.

Phần mềm bao gồm 2 chương trình: ISIS cho phép mô phỏng mạch và ARES dùng để vẽ mạch in. Proteus là công cụ mô phỏng cho các loại Vi Điều Khiển khá tốt, nó hỗ trợ các dòng VĐK PIC, 8051, PIC, dsPIC, AVR, HC11, MSP430, ARM7/LPC2000 ... các giao tiếp I2C, SPI, CAN, USB, Ethenet,... ngoài ra còn mô phỏng các mạch số, mạch tương tự một cách hiệu quả. Proteus là bộ công cụ chuyên về mô phỏng mạch điện tử.

ISIS đã được nghiên cứu và phát triển trong hơn 12 năm và có hơn 12000 người dùng trên khắp thế giới. Sức mạnh của nó là có thể mô phỏng hoạt động của các hệ vi điều khiển mà không cần thêm phần mềm phụ trợ nào. Sau đó, phần mềm ISIS có thể xuất file sang ARES hoặc các phần mềm vẽ mạch in khác.

Trong lĩnh vực giáo dục, ISIS có ưu điểm là hình ảnh mạch điện đẹp, cho phép ta tùy chọn đường nét, màu sắc mạch điện, cũng như thiết kế theo các mạch mẫu (templates)

Những khả năng khác của ISIS là:

Tự động sắp xếp đường mạch và vẽ điểm giao đường mạch.

Chọn đối tượng và thiết lập thông số cho đối tượng dễ dàng

Xuất file thống kê linh kiện cho mạch

Xuất ra file Netlist tương thích với các chương trình làm mạch in thông dụng.

Đối với người thiết kế mạch chuyên nghiệp, ISIS tích hợp nhiều công cụ giúp cho việc quản lý mạch điện lớn, mạch điện có thể lên đến hàng ngàn linh kiện.

Thiết kế theo cấu trúc (hierachical design)

Khả năng tự động đánh số linh kiện

1.6. Rờ le (Relay)

Rơ le( Relay) là 1 công tắc chuyển đổi hoạt động bằng điện. Nói là một công tắc vì rơ le có 2 trạng thái là ON và OFF. Rơ le ở trạng thái ON hay OFF phụ thuộc vào có dòng điện chạy qua rơ le hay không.



Hình 1.8. Rờ le 5VDC

*(Nguồn:* [*https://sptshop.vn/san-pham/relay-12v-5-chan-dc-ro-le-5-chan-gan-den-coi/*](https://sptshop.vn/san-pham/relay-12v-5-chan-dc-ro-le-5-chan-gan-den-coi/) *)*

**Thông số kỹ thuật:**

Điện áp điều khiển: 5V.

Dòng điện cực đại: 10A.

Thời gian tác động: 10ms.

Thời gian nhả hãm: 5ms.

Nhiệt độ hoạt động: -45oC ~ 75oC.

Relay 5 chân HLS8L-5VDC là loại linh kiện đóng ngắt điện cơ đơn giản. Nó gồm 2 phần chính là cuộn hút và các tiếp điểm. Cấu tạo của relay được mô tả trong hình.

**Ứng dụng của Rờ le**

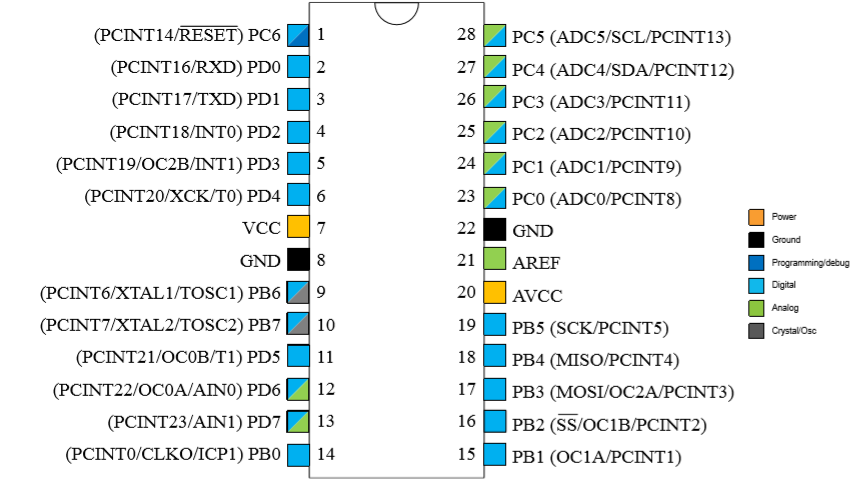
Nhìn chung, công dụng của rờ-le là "dùng một năng lượng nhỏ để đóng cắt nguồn năng lượng lớn hơn".

Rờ-le được dùng khá thông dụng trong các ứng dụng điều khiển động cơ và chiếu sáng.

Khi cần đóng cắt nguồn năng lượng lớn, rờ-le thường được ghép nối tiếp. Nghĩa là một rờ-le nhỏ điều khiển một rờ-le lớn hơn, và rơ-le lớn sẽ điều khiển nguồn công suất.

1.7. IC Lập trình Atmega16u2

ATMEGA16U2 sản xuất theo kiến trúc RISC tiên tiến bao gồm: 125 tập lệnh đơn lệnh chu kỳ; 32 thanh ghi 8bit dùng chung; Tối đa 16 tập lệnh MÍP với tần số cho phép là 16MHz. Một bộ đếm 8bit và 1 bộ đếm 16 bit dùng cho điều biến độ rộng xung PWM. Giao tiếp USART với chế độ SPI master và chế độ kiểm soát phần cứng. Bộ chuyển đổi tín hiệu ADC công cụ thời gian thực và khả năng hoạt động ở mức điện áp thấp khi ở chế độ tiết kiệm năng lượng.



Hình 1.9. Sơ đồ chân IC Atmega16u2

*Nguồn(*[*http://www.iotsky.vn/san-pham/arduino-uno-r3-chip-cam-dip28.html*](http://www.iotsky.vn/san-pham/arduino-uno-r3-chip-cam-dip28.html)*)*

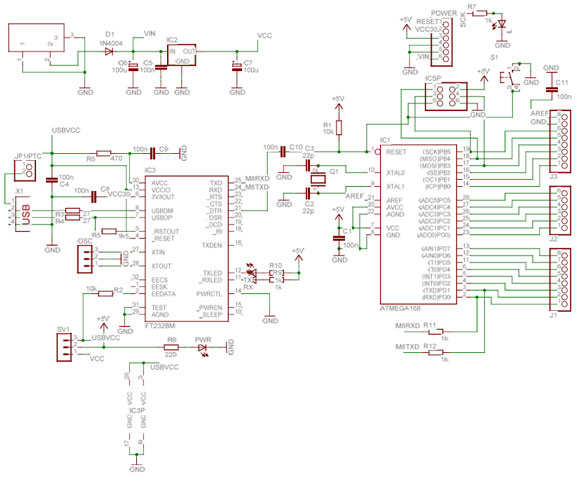
**Cấu tạo:**

22 chân đa dụng

2 chân nối với bộ dao động thạch anh có tần số 16MHz

3 chân nguồn VCC và GND cấp nguồn 5V và Mass.

6 chân ADC (Analog Digital Convert): bộ chuyển đổi ADC 10 bitphoois hợp với chân AREF hiệu chỉnh giá trị điện áp đầu vào cho ADC.

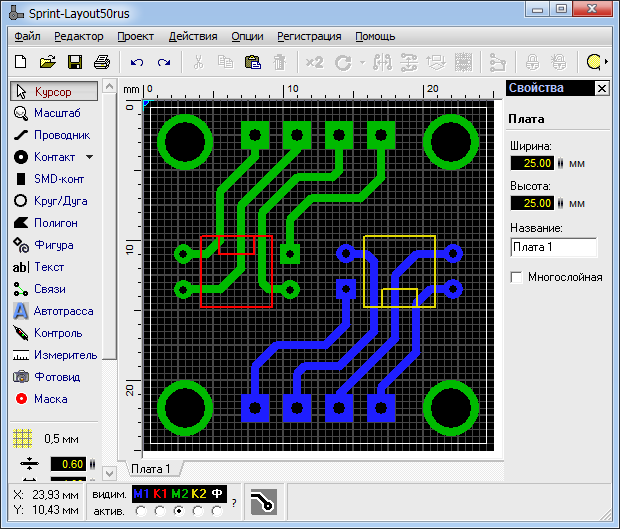


Hình 1.10. Sơ đồ nguyên lý board mạch Arduino UNO R3

*(Nguồn:* [*https://techmaster.vn/posts/33768/arduino-la-gi-arduino-de-lam-gi*](https://techmaster.vn/posts/33768/arduino-la-gi-arduino-de-lam-gi)*)*

1.8. Phần mềm Sprint Layout Viewer

Sprint layout là một trong những phần mềm thiết kế và sao chép mạch điện tốt nhất



Hình 1.11. Giao diện sprint layout

(Nguồn: <https://teylopotap.tk/d/bYII29ad5e048dbf120581ecc5efed4e97d0d8l1yx3y/>)

Ưu điểm: Phần mềm dễ sử dụng, dụng lượng chỉ 7MB.

Dễ dàng sao chép mạch điện

Tập chung vẽ thủ công không cần vẽ sơ đồ nguyên lý (schematic).

Khả năng đi dây cực nhanh và linh hoạt.

Thư viện phong phú, đa dạng.

Giao diện đơn giản, dễ sử dụng.

Yêu cầu hệ thống.

Yêu cầu hệ thống.

Phần mềm dễ sử dụng, dụng lượng chỉ 7MB.

Dễ dàng sao chép mạch điện

Tập chung vẽ thủ công không cần vẽ sơ đồ nguyên lý (schematic).

Khả năng đi dây cực nhanh và linh hoạt.

Thư viện phong phú, đa dạng.

Giao diện đơn giản, dễ sử dụng.

2. Phương pháp nghiên cứu

2.1. Thi công,thiết kế mạch in với Sprint Layout

Quá trình thiết kế sản phẩm và gia công theo các bước:

**(1)** In mạch in mạch ra giấy Decal.

**(2)** Ủi mạch phủ giấy Decal lên mạch đồng và ủi trong thời gian nhất định.

**(3)** Ăn mòn hòa tan dung dịch Fe2O3 với nước, gia nhiệt cũng như tốc độ chuyển động của bo mạch trong dung dịch này.

**(4)** Khoan mạch Dùng máy khoan khoan theo đường vẽ đã thiết kế.

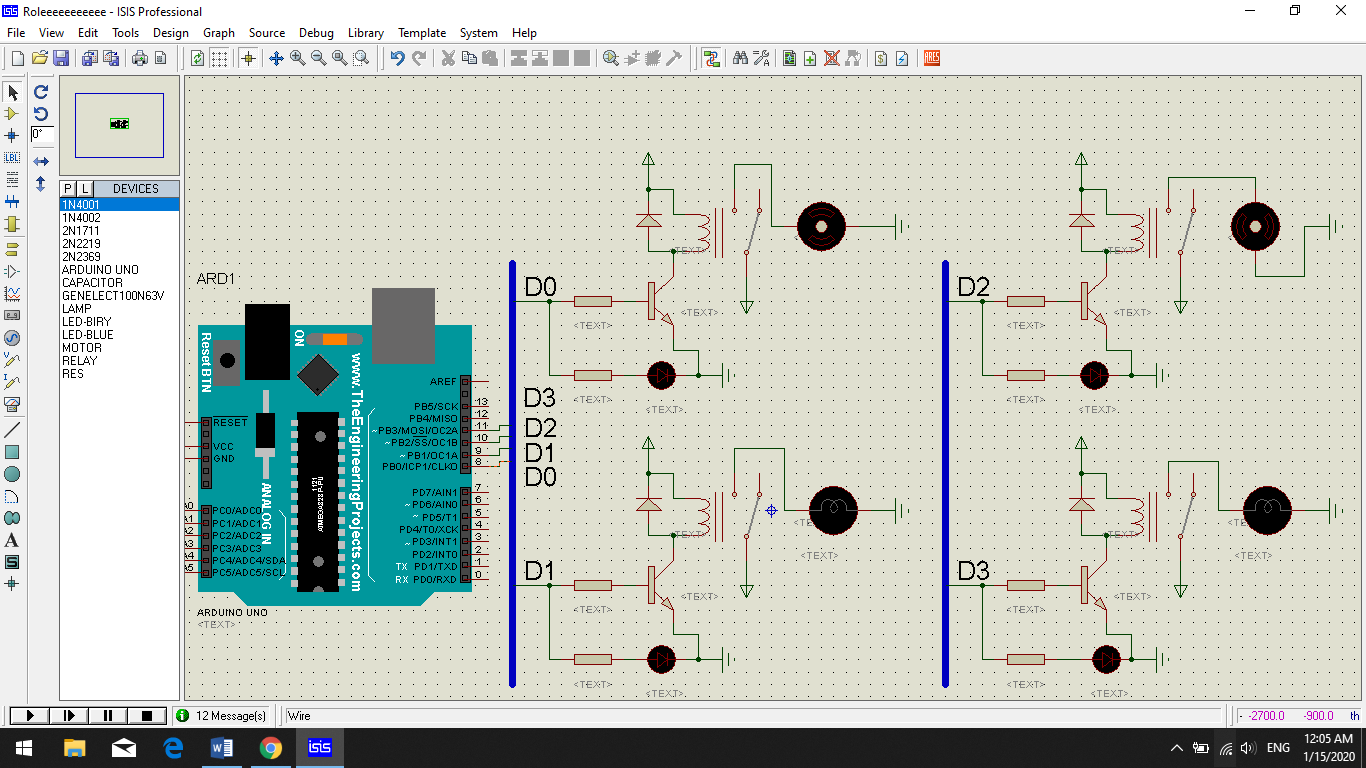
**(5)**  Gắn linh kiện Gắn linh kiện lên các vị trí thích hợp đã được thiết kế và tiến hành lắp dây.



Hình 2.1. Các bước thiết kế mạch in

2.2. Mô phỏng phần mềm Proteus

Nguyên lý hoạt động:



Arduino

Tải kết nối đèn

Tải kết nối quạt

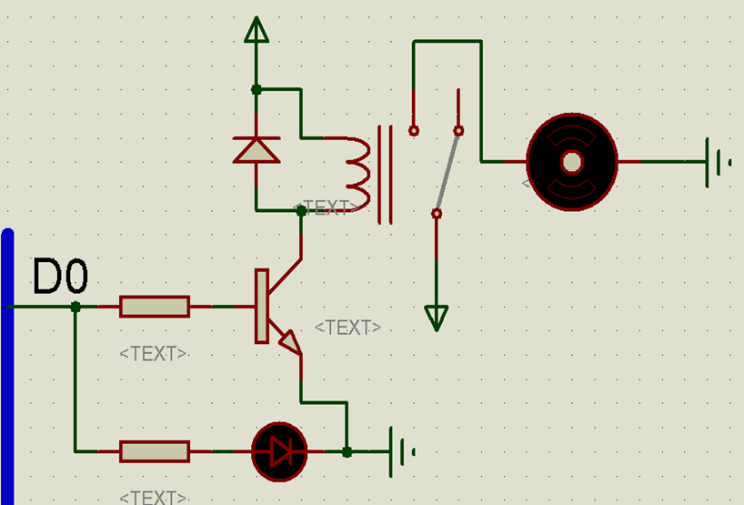
Hình 2.2. Sơ đồ nguyên lý hoạt động điều khiển thiết bị qua bluetooth trong proteus

Khái quát sơ đồ nguyên lý:

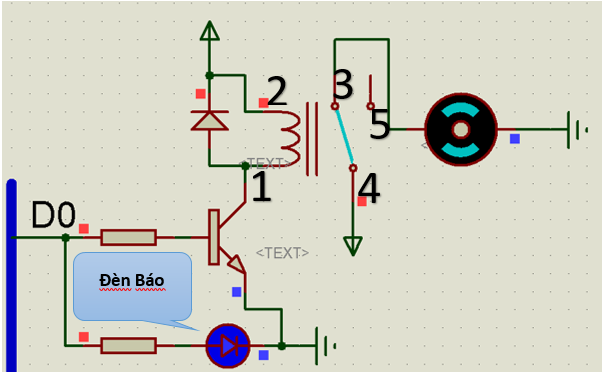
Để điều khiển các tải kết nối với quạt ta dùng các bit D0, D2.

Để điều khiển các tải kết nối với đèn ta dùng các bit D1, D3.

Trong đó các chân D0, D1, D2, D3 được nối với các chân digital 8, 9, 10, 11 của Arduino.



Hình 2.3. Sơ đồ mô phỏng khi tải kết nối với quạt không có dòng điện chạy qua



Hình 2.4. Sơ đồ mô phỏng khi tải kết nối với quạt có dòng điện chạy qua

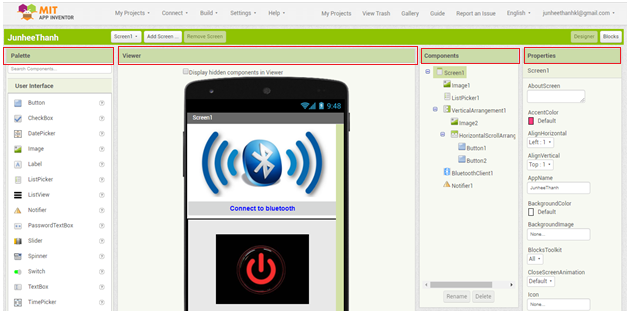
Khi ở trạng thái bình thường các chân digital của Arduino có giá trị là 0 thì các thiết bị điện tử sẽ không hoạt động.

Khi chân digital của Arduino 8 có giá trị là 1 thì D0 sẽ mang giá trị là 1, làm cho đén báo sáng lên báo hiệu có dòng điện chạy qua. Transistor được mắc song song với đèn báo sẽ dẫn, lúc đó sẽ có một dòng khép kín đi từ chân số 1 sang chân số 2 của Role (Relay). Bây giờ cuộn dây sẽ tạo thành một lực từ, lực từ này hút chân số 4 từ chân số 5 về chân số 3. Tạo ra vòng điện khép kín từ chân số 3 nối với chân số 4 và nối đất. Lúc này quạt sẽ quay.

2.3. Thiết kế giao diện

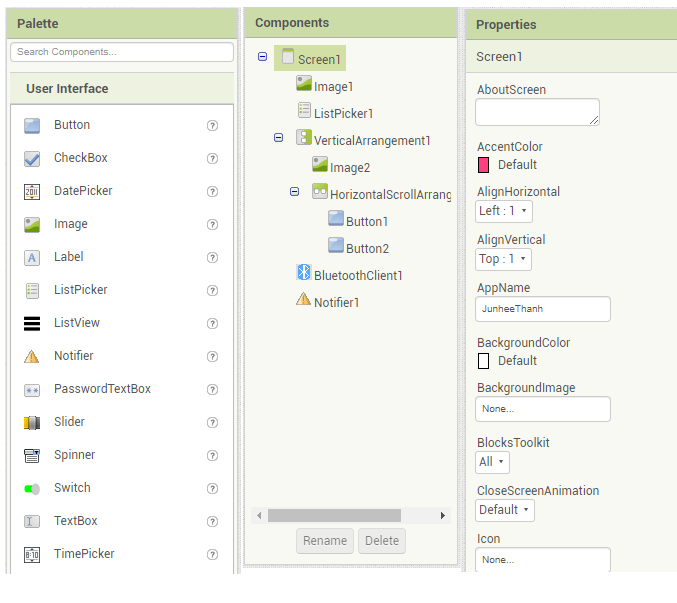
Đầu tiên bạn cần đăng nhập vào đường link: [*ai2.appinventor.mit.edu*](http://ai2.appinventor.mit.edu/)*.* Sau đó tiến hành đăng nhập bằng tài khoản Google của bạn để mở trang quản lí project.

Giao diện thiết kế (Design)



Hình 2.5. giao diện thiết kế

Cửa sổ thiết kế gồm 4 khung chức năng chính như hình dưới đây:



Hình 2.6. Giao diện chức năng và thiết kế

Đầu tiên, một ứng dụng có thể có nhiều cửa sổ giao diện, trong MIT AI2 gọi là các Screen.

Palette: Chứa các thành phần có thể đặt lên trên Screen như: Button, Label, Image, Listview, Video player, …. Đến các thanh phần chức năng không nhìn thấy trên Screen như: BLE extension, Notifier, các sensors,…

Viewer: Hiển thị giao diện screen. Kéo thả các thành phần từ khung Palette sang đây để thiết kế giao diện cho phần mềm của bạn.

Components: Sơ đồ cây thể hiện cấu trúc các thành phần đã được bố trí trên Screen.

Properties: Hiển thị thuộc tính của component tương ứng được chọn.

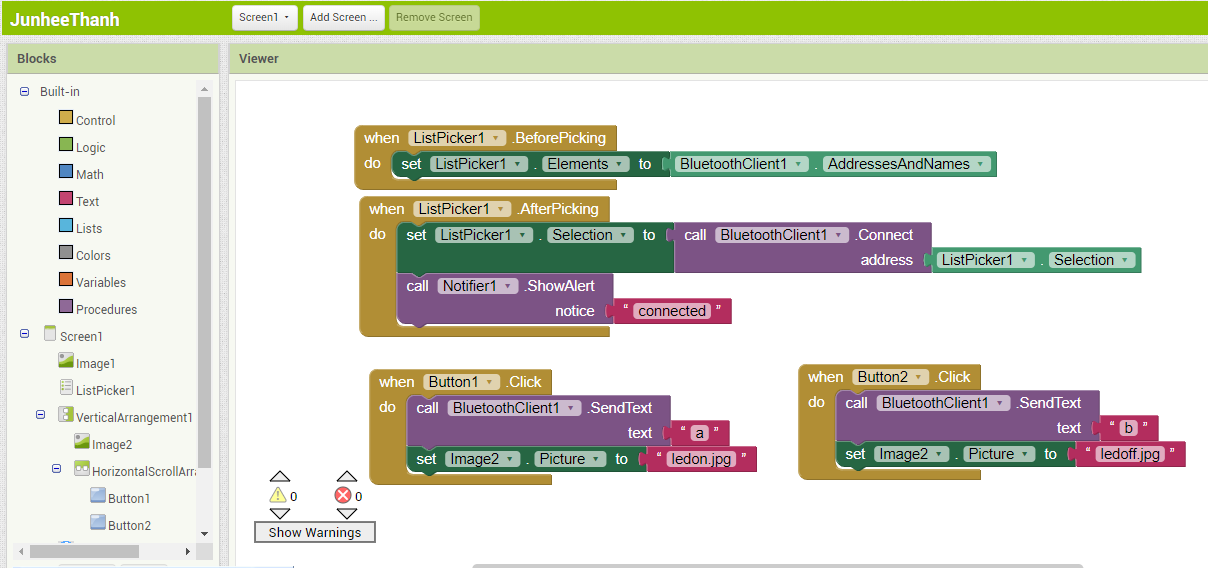
Ngoài ra còn khung Media chứa các file media bạn tải lên để sử dụng trong chương trình như: Ảnh icon, ảnh nền, …Bên trái là các phần để bạn kéo thả vào khung màn hình điện thoại. Bên phải, bảng Component là thứ tự của các cục giao diện mà bạn thả vào, bấm vào từng cái sẽ xuất hiện bên Properties để chỉnh sửa nhanh các thuộc tính giao diện như độ dài, rộng,… Nếu để mặc định chưa bấm vào gì thì Properties đó là của khung lớn.

2.4. Lập trình chức năng

Trong MIT AI2, code chính là các Blocks, việc của chúng ta là kéo thả các blocks này sang khung Viewer và kết nối chúng theo chức năng mong muốn.

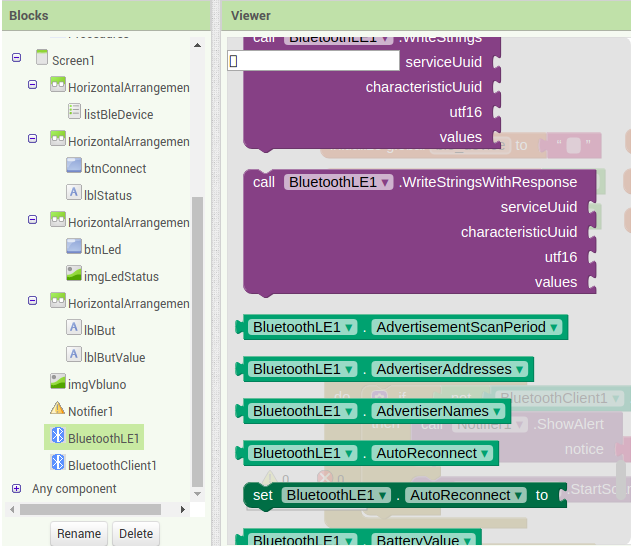
Blocks gồm 2 nhóm chính:

Các block chức năng cơ bản của một chương trình như: điều khiển luồng, logic, toán học, ký tự, biến,…



Hình 2.7. Các block chức năng

Các block chức năng theo từng component trong ứng dụng: Mỗi component của ứng dụng đều có các block chức năng tương ứng.

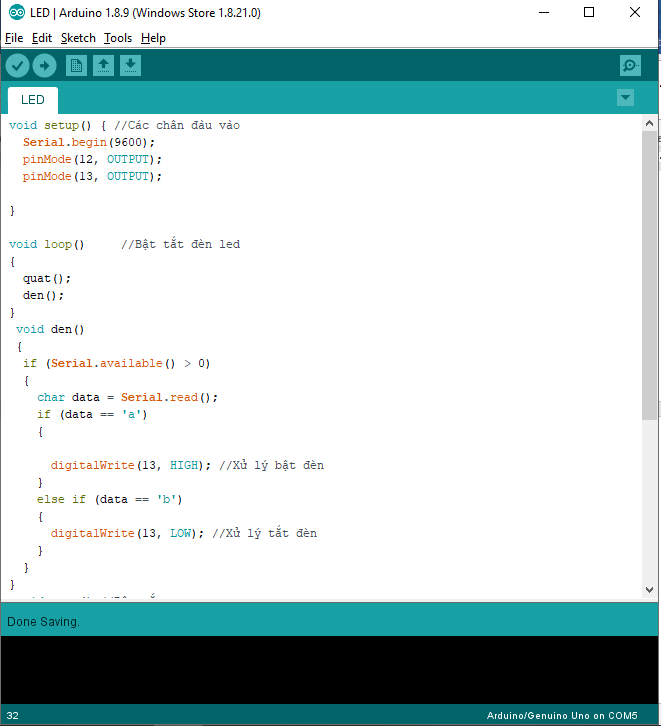


Hình 2.8. Các block chức năng

2.4. Giao tiếp Arduino với HC05

Nối chân TX của Bluetooth HC05 vào chân RX của Arduino(chân này có nhiệm vụ phát tín hiệu khi bluetooth được kết nối thành công). Khi đó chân TX sẽ phát tín hiệu truyền đến chân RX của Arduino, Arduino sẽ nhận tín hiệu và xử lý.

2.5. Điều khiển thiết bị với Arduino

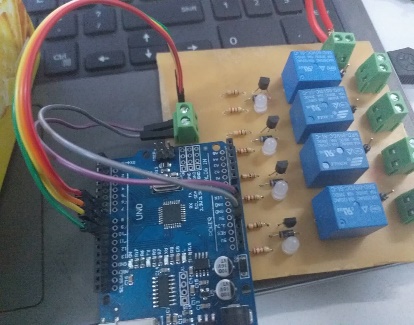
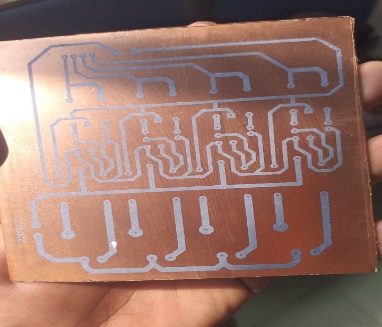




3. KẾT QUẢ

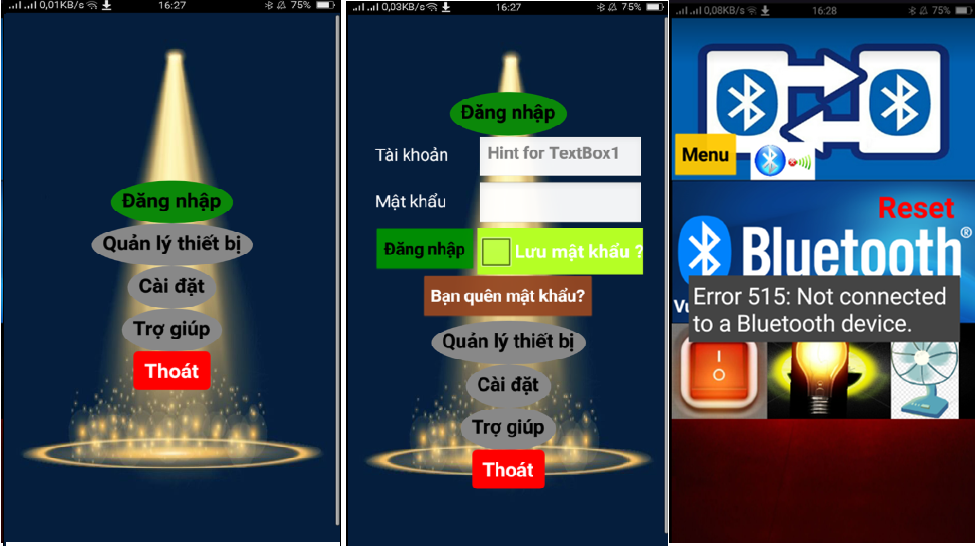
3.1. Thi công mạch in

Sau khi thi công mạch in bao gồm 1 bảng mạch, 4 rờ le, 1Arrduino, 4 đèn led, dây và cáp, cọc xanh, 8 Registor.



Hình 3.1. Sản phẩm sau khi thiết kế

3.2. Giao diện phần mềm điều khiển



1. (2) (3)

Hình 3.2. Giao diện phần mềm

Đầu tiên sẽ hiển thị một màn hình yêu cầu đăng nhập (1) Nhập thông tin tài khoản (2) Giao diện chính để điều khiển thiết bị (3).

3.3. Điều khiển thiết bị

**

Hình 3.3. Điều khiển thiết bị bằng điện thoại thông qua bluetooth

Kết nối thiết bị với mô- đun điều khiển được điều khiển bởi ứng dụng đã được thiết kế bằng app inventor. Mở ứng dụng trên điện thoại, tiến hành đăng nhập và chọn thiết bị(quạt, đèn,...), kết nối bluetooth và điều khiển thiết bị bằng bluetooth.

4. KẾT LUẬN

Sản phẩm sau khi hoàn thành đã thể hiện được các tiêu chí của đề tài thực tập yêu cầu, có thể điều khiển được 2 thiết bị bằng bluetooth thông qua điện thoại là quạt và đèn. Trong quá trình điều khiển có thể bị trễ do nhiễu sóng hoặc do giao tiếp từ module HC-05 chưa được tốt. Còn bị giới hạn khi kết nối bluetooth với thiết bị.

PHỤ LỤC

