TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN 1**

**ĐIỀU KHIỂN CÁNH TAY ROBOT THÔNG QUA ỨNG DỤNG TRÊN ĐIỆN THOẠI DI ĐỘNG**

*Người hướng dẫn*: ThS. Trần Trung Tín

*Người thực hiện*: **Đường Vinh Tài - 51303150**

**Dương Toàn - 51303180**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2017**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN 1**

**ĐIỀU KHIỂN CÁNH TAY ROBOT THÔNG QUA ỨNG DỤNG TRÊN ĐIỆN THOẠI DI ĐỘNG**

*Người hướng dẫn*: ThS. Trần Trung Tín

*Người thực hiện*: **Đường Vinh Tài - 51303150**

**Dương Toàn - 51303180**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2017**

# LỜI CẢM ƠN

Bài làm của chúng em đã hoàn thành, bài làm sẽ vẫn còn sai sót chưa được hoàn thiện như mong muốn của thầy. Nhưng chúng em cũng xin cảm ơn sự hướng dẫn, chỉ bảo của thầy ThS. Trần Trung Tín đã giúp để chúng em được hoàn thành bài báo cáo môn ĐỒ ÁN 1.

**BÀI LÀM ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm đồ án của chúng tôi và được sự hướng dẫn của ThS. Trần Trung Tín. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong bài thuyết trình còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào chúng tôi xin hoàn toàn chịu trách nhiệm về nội dung bài thuyết trình của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày tháng năm*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Đường Vinh Tài*

*Dương Toàn*

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

**Phần xác nhận của GV hướng dẫn**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

MỤC LỤC

[LỜI CẢM ƠN 2](#_Toc484433121)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN 4](#_Toc484433122)

[CHƯƠNG I: GIỚI THIỆU VỀ ĐỀ TÀI “Điều khiển cánh tay robot thông qua ứng dụng trên điện thoại di động”. 7](#_Toc484433123)

[1. Giới thiệu về đề tài nghiên cứu 7](#_Toc484433124)

[2. Mô hình hệ thống 8](#_Toc484433125)

[3. Cách sử dụng 9](#_Toc484433126)

[CHƯƠNG II: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG PHẦN MỀM. 15](#_Toc484433127)

[1. Ý tưởng thiết kế 15](#_Toc484433128)

[2. Lựa chọn linh kiện 16](#_Toc484433129)

[a. Bluetooth HC-05 16](#_Toc484433130)

[b. Arduino UNO R3 20](#_Toc484433131)

[c. Servo RC 24](#_Toc484433132)

[3. Yêu cầu hệ thống 26](#_Toc484433133)

[a. Chức năng đóng/mở Bluetooth 26](#_Toc484433134)

[b. Chức năng tìm Bluetooth 26](#_Toc484433135)

[c. Chức năng kết nối với Bluetooth 26](#_Toc484433136)

[d. Chức năng điều khiển thiết bị có Bluetooth 26](#_Toc484433137)

[4. Một số ứng dụng trong chương trình 27](#_Toc484433138)

[a. BluetoothDevice 27](#_Toc484433139)

[b. BluetoothAdapter 27](#_Toc484433140)

[c. BroadcastReceiver 27](#_Toc484433141)

[d. UUID 28](#_Toc484433142)

[e. BluetoothServerSocket và BluetoothSocket 28](#_Toc484433143)

[5. Thiết kế hệ thống 30](#_Toc484433144)

[a. Sơ đồ Use-Case 30](#_Toc484433145)

[b. Sơ đồ Class 31](#_Toc484433146)

[c. Sơ đồ Activity 32](#_Toc484433147)

[d. Giao diện hệ thống 33](#_Toc484433148)

[CHƯƠNG III: QUÁ TRÌNH THỰC HIỆN. 35](#_Toc484433149)

[1. Cài đặt các chế độ cho thiết bị Bluetooth bằng Arduino lên bo mạch Arduino R3 Uno: 35](#_Toc484433150)

[2. Sau khi cài đặt và cấu hình xong thiết bị thì nạp lại code cho bo mạch trên để kiểm tra. 37](#_Toc484433151)

[3. Sau khi demo thành công, nhóm bắt đầu tiến hành xây dựng hệ thống. 42](#_Toc484433152)

[CHƯƠNG IV: TỔNG KẾT. 43](#_Toc484433153)

[1. Đánh giá kết quả thực hiện đồ án 43](#_Toc484433154)

[2. Hướng phát triển 43](#_Toc484433155)

[Tài liệu tham khảo 45](#_Toc484433156)

**DANH MỤC CÁC BẢNG BIỂU, ĐỒ THỊ. HÌNH ẢNH**

[Hình 1: Mô hình hệ thống 8](#_Toc484433157)

[Hình 2: Icon của ứng dụng 10](#_Toc484433158)

[Hình 3: Màn hình khi chạy ứng dụng sẽ yêu cầu mở Bluetooth. 11](#_Toc484433159)

[Hình 4: Màn hình sau khi mở Bluetooth thành công và người dùng sẽ nhấn vào nút tìm để tìm kiếm thiết bị cần kết nối. 11](#_Toc484433160)

[Hình 5: Nhấn nút tìm và kết nối với Bluetooth 12](#_Toc484433161)

[Hình 6: Nhấn chọn và đợi ứng dụng tự động kết nối 12](#_Toc484433162)

[Hình 7: Nhập pin để kết nối (PIN: 1234) 13](#_Toc484433163)

[Hình 8: Sau khi kết nối thành công có thể thực hiện điều khiển 14](#_Toc484433164)

[Hình 9: Bluetooth HC-05 16](#_Toc484433165)

[Hình 10: Mô hình phần cứng của module 17](#_Toc484433166)

[Hình 11: Arduino UNO R3 20](#_Toc484433167)

[Hình 12: Thành phần của bo mạch Arduino UNO R3 21](#_Toc484433168)

[Hình 13: Động cơ Servo RC 24](#_Toc484433169)

[Hình 14: Sơ đồ minh họa động cơ Servo RC 25](#_Toc484433170)

[Hình 15: Sơ đồ Use-Case 30](#_Toc484433171)

[Hình 16: Sơ đồ Class 31](#_Toc484433172)

[Hình 17: Sơ đồ Activity 32](#_Toc484433173)

[Hình 18: Màn hình giao diện ứng dụng 33](#_Toc484433174)

[Hình 19: Màn hình giao diện yêu cầu mở Bluetooth 33](#_Toc484433175)

[Hình 20: Màn hình tìm và chọn thiết bị Bluetooth khác 34](#_Toc484433176)

[Hình 21: Màn hình nhập password để kết nối với thiết bị Bluetooth đã chọn 34](#_Toc484433177)

[Hình 22: Màn hình ứng dụng 34](#_Toc484433178)

[Hình 23: Sơ đồ nối dây của Bluetooth HC-05 với bo mạch Arduino 35](#_Toc484433179)

[Hình 24: Giao diện chương trình demo kết nối với Bluetooth HC-05 để kiểm tra 38](#_Toc484433180)

[Hình 25: Tìm kiếm và kết nối với thiết bị Bluetooth HC-05 39](#_Toc484433181)

[Hình 26: Giao tiếp từ ứng dụng 40](#_Toc484433182)

[Hình 27: Nhận tính hiệu từ ứng dụng demo gửi qua 41](#_Toc484433183)

[Hình 28: Gửi tính hiệu cho ứng dụng demo 41](#_Toc484433184)

[Hình 29: Cách nối dây với các động cơ Servo 42](#_Toc484433185)

1. GIỚI THIỆU VỀ ĐỀ TÀI “Điều khiển cánh tay robot thông qua ứng dụng trên điện thoại di động”.
   1. Giới thiệu về đề tài nghiên cứu

Đề tài “Điều khiển cánh tay robot thông qua ứng dụng trên điện thoại di động” là một đề tài giúp người sử dụng có thể điều khiển thiết bị bằng thiết bị điện thoại thông minh thông qua Bluetooth. Ngày nay điện thoại thông minh khá phổ biến và tiện lợi nên việc cài đặt ứng dụng nhanh chóng. Việc áp dụng điều khiển cánh tay robot này lên điện thoại cũng khá thuận tiện vì xây dựng ứng dụng trên hệ điều hành Android tương đối dễ dàng và quen thuộc. Ưu điểm là có thể kết nối Bluetooth giữa thiết bị cánh tay robot với điện thoại bằng một nút nhấn khá dễ dàng và ứng dụng có thể thực hiện việc điều khiển bất cứ khi nào mà không bị gò bó bởi sử dụng một bộ điều khiển riêng biệc.

Từ những thuận lợi trên, nhóm chúng em đã thực hiện việc tạo một ứng dụng trên điện thoại. Ứng dụng này có tên là “iRobot” có chức năng chính là truyền và nhận dữ liệu qua Bluetooth để điều khiển thiết bị cánh tay robot. Và nhóm chúng em áp dụng trên chiếc điện thoại SamSung.

*Về phía cánh tay robot:* nhóm đã sử dụng bo mạch Arduino UNO R3, Bluetooth HC05, và động cơ Servo gắn với cánh tay. Bluetooth có chức năng truyền nhận dữ liệu với điện thoại và việc truyền nhận dữ liệu này được điều khiển bằng bo mạch Arduino UNO R3. Bo mạch có chức năng nhận dữ liệu từ Bluetooth và xử lý dữ liệu để điều khiển các động cơ Servo của robot.

* 1. Mô hình hệ thống

|  |  |  |  |
| --- | --- | --- | --- |
| /Users/pc/Desktop/SAMSUNG.jpg | /Users/pc/Desktop/bluetooth.png | /Users/pc/Desktop/bluetooth.png |  |

Hình 1: Mô hình hệ thống

Việc thực hiện các điều khiển thông qua điện thoại để điều khiển 4 khớp cánh tay robot dựa trên phần cánh tay đã được dựng sẵn với 4 động cơ Servo, nhóm đã mô phỏng và thực hiện trên một động cơ Servo.

Cách lắp đặt dây dẫn để cung cấp nguồn và dẫn truyền dữ liệu trao đồi giữa Bluetooth và bo mạch, động cơ Servo và bo mạch.

Hệ thống trên cần phải có các thiết bị có Bluetooth để kết nối.

* 1. Cách sử dụng

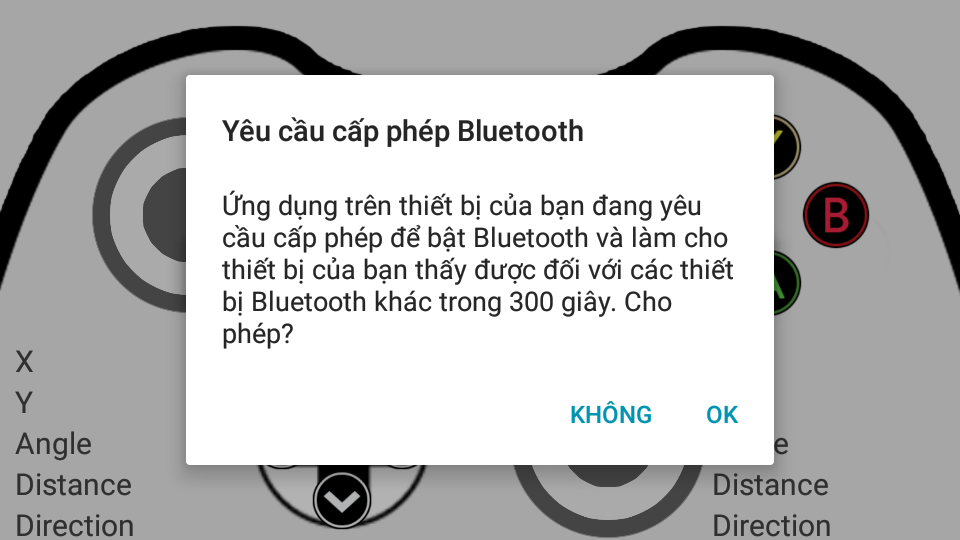
Ứng dụng sau khi đã được cài đặt trên điện thoại thành công chọn và chạy ứng dụng. Khi ứng dụng được chạy sẽ yêu cầu mở Bluetooth và sẽ xuất hiện màn hình tìm kiếm thiết bị. Chọn vào nút “Tìm” và đợi kết quả xuất hiện trên màn hình rồi chọn để kết nối với thiết bị cần kết nối. Sau khi đã kết nối thiết bị thành công chúng ta có thể điều khiển được thiết bị robot.

Nếu điện thoại không kết nối thành công sẽ không thể điều khiển được thiết bị robot. Nguyên nhân: có thể điện thoại không dùng đúng password để kết nối (password mặt định là: 1234), hay bị lỗi hệ thống của robot phải restart lại bằng cách tắt nguồn và khởi động lại.

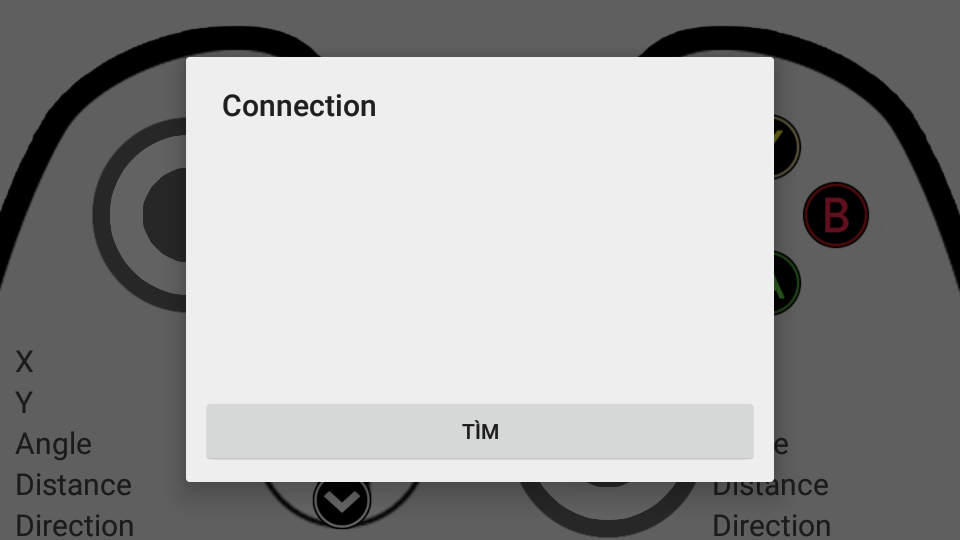
Sau khi kết nối 2 thiết bị thành công, ứng dụng có thể điều chỉnh lại Bluetooth của robot thông qua “command AT:” trên điện thoại. (file kèm theo để sử dụng “HC-0305\_serial\_module\_AT\_commamd\_set\_201104\_revised.pdf”).



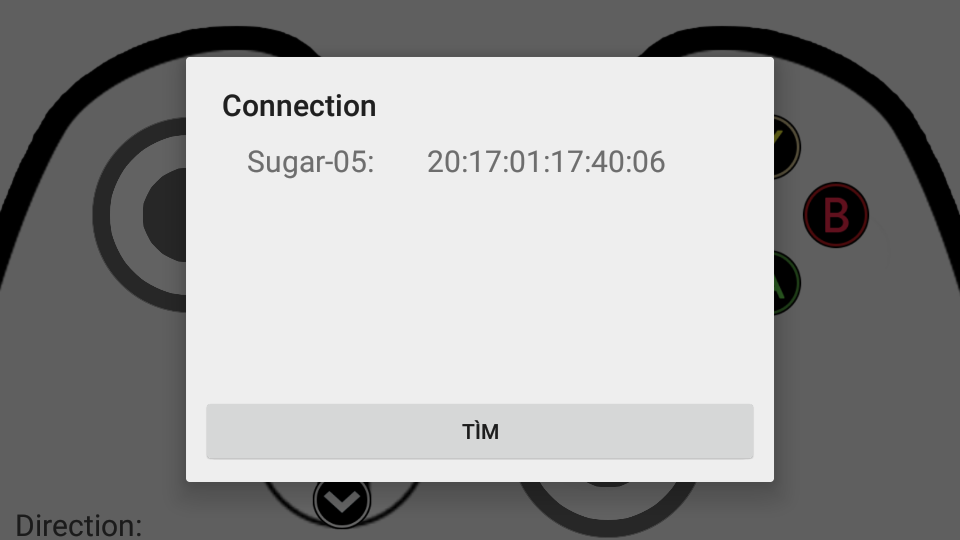
Hình 2: Icon của ứng dụng



Hình 3: Màn hình khi chạy ứng dụng sẽ yêu cầu mở Bluetooth.



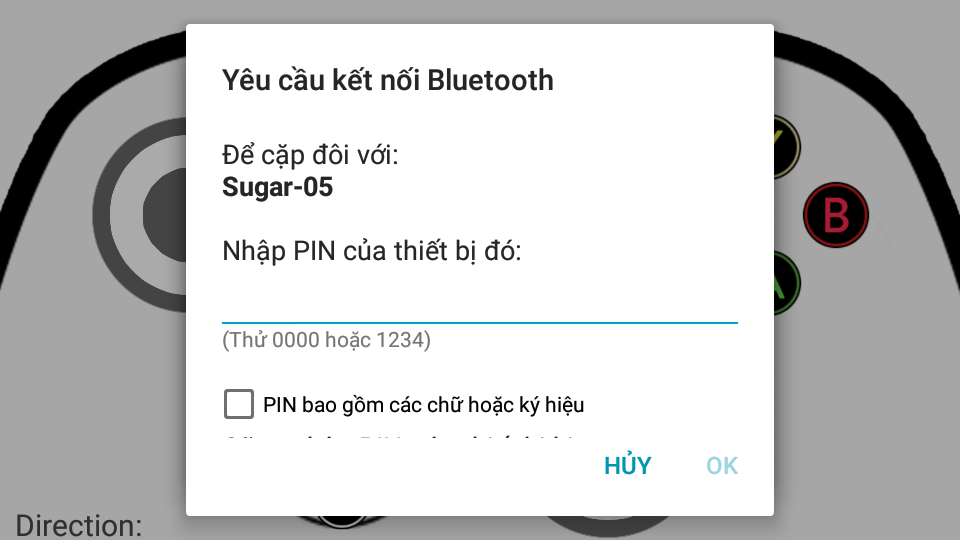
Hình 4: Màn hình sau khi mở Bluetooth thành công và người dùng sẽ nhấn vào nút tìm để tìm kiếm thiết bị cần kết nối.



Hình 5: Nhấn nút tìm và kết nối với Bluetooth



Hình 6: Nhấn chọn và đợi ứng dụng tự động kết nối



Hình 7: Nhập pin để kết nối (PIN: 1234)



Hình 8: Sau khi kết nối thành công có thể thực hiện điều khiển

1. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG PHẦN MỀM.
   1. Ý tưởng thiết kế

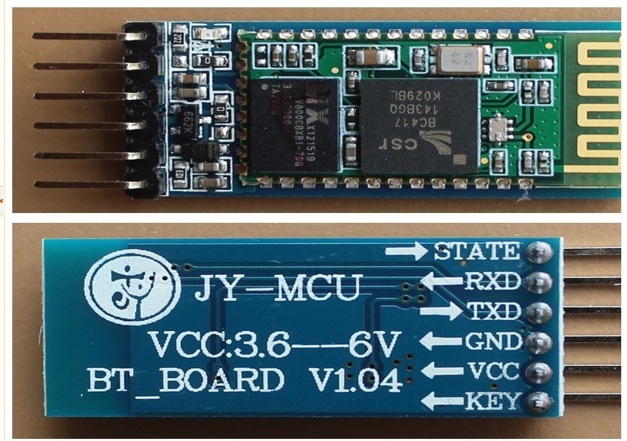
Nhận thấy việc áp dụng công nghệ vào đời sống giúp nâng cao năng suất, giảm thiểu rủi ro và đang là xu thế của thế giới. Đặc biệt sự ra đời và phát triển của công nghệ chế tạo robot nhằm tạo ra sự tự động hóa trong quá trình sản xuất giảm đi sức lao động. Đối với nước ngoài thì Robot đã được nghiên cứu và chế tạo để ứng dụng vào sản xuất đã có từ trước và đã đạt được những thành tựu hết sức to lớn, hỗ trợ đắc lực con người trong nhiều lĩnh vực như: sản xuất xe hơi, trang thiết bị y tế, môi trường vũ trụ, trang bị quân đội,...

Qua tìm hiểu trên mạng thì tỷ lệ người sử dụng điện thoại thông minh ngày càng cao và ứng dụng chia sẻ dữ liệu qua Bluetooth khá dễ dàng và tiện dụng. Việc thiết kế tạo ứng dụng trên Android SDK đã quen thuộc và dễ dàng. Hầu hết các điện thoại thông minh bây giờ đều có thiết bị Bluetooth, đều được hướng dẫn và cài đặt sử dụng rất nhiều trên mạng.

Việc tìm hiểu về Arduino cũng như công nghệ Bluetooth, hay về động cơ Servo RC thông qua các diễn đàn.

Dựa vào thông tin trên nhóm đã quyết định làm hệ thống “Điều khiển cánh tay robot thông qua ứng dụng điện thoại di động”.

* 1. Lựa chọn linh kiện
     1. Bluetooth HC-05



Hình 9: Bluetooth HC-05

Module HC-05 là một cách dễ dàng để có thể sử dụng giao tiếp bluetooth qua Serial Port, được thiết kế để truyền dữ liệu nối tiếp qua wireless.

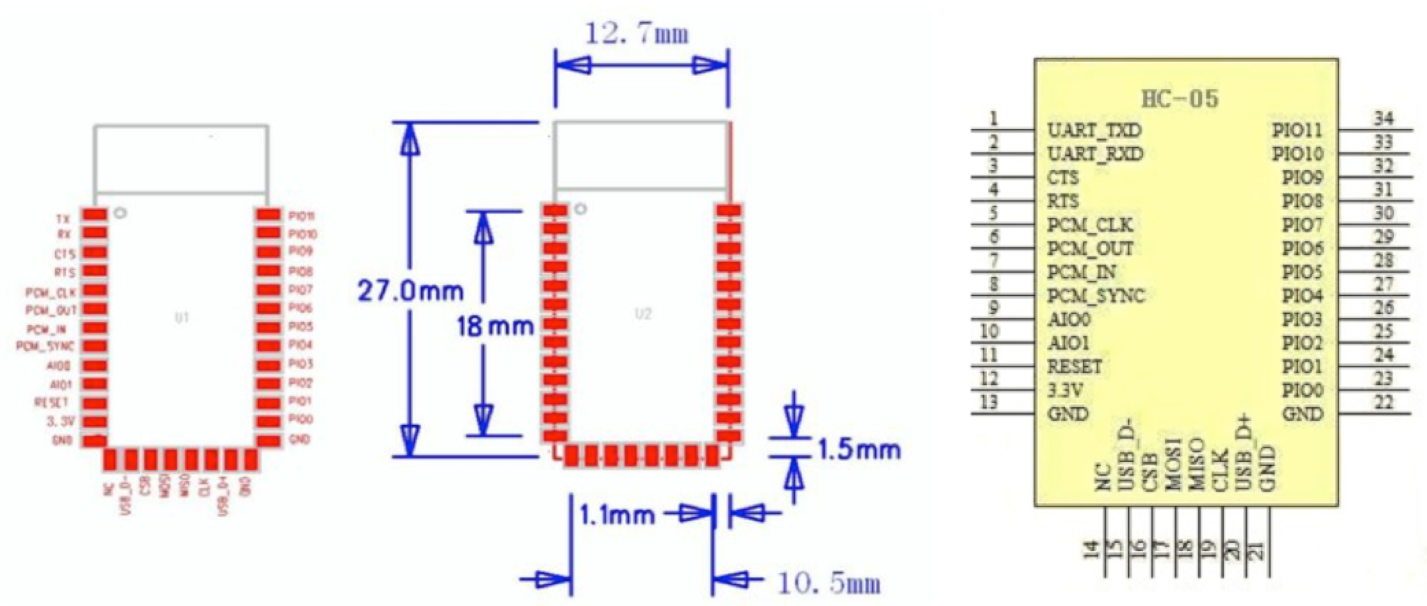
Sử dụng chuẩn bluetooth 2.0 + EDR(Enhanced Data Rate) 3Mbps

Điều chế thu phát radio ở tần số 2,4Ghz, sử dụng chip bluecore bluetooth 04 - chip duy nhất với hệ thống Cmos và AFH thích ứng với tính năng nhảy tần. Kích thước 12,7mm x 27mm.

**Đặc điểm kỹ thuật:**

* Tính năng phần cứng
* Độ nhạy -80dBm
* Công suất truyền lên đến +4dBm
* Công suất thấp 3.6V, hoạt động từ 3.6V đến 6V
* Điều khiển PIO
* Giao tiếp UART với tốc baud lập trình được
* Tích hợp anten
  + Kết nối ở biên mạch.
* Tính năng phần mềm
* Mặc định tốc độ baud là 9600,databits : 8, Stopbit : 1,Parity : No.Hỗ trợ tốc độ baud : 9600,19200,38400,57600,115200,230400,460800.
* Khi có 1 xung ở PIO0, thiết bị sẽ bị ngắt kế nối.
* Trạng thái chỉ thị port PIO1: low- ngắt kết nối, high- đã kết nối.
* PIO10 và PIO11 có thể được lết nối với led đỏ và led xanh riêng. Khi master và slave được kết nối với nhau, led đỏ và led xanh sẽ nháy 1 lần 2s, khi ngắt kết nối chỉ led xanh nháy 2 lần/s.
* Tự động kết nối với thiết bị cuối cùng khi nguồn được cấp.
* Cho phép kết nối thiết bị mặc định.
* Tự động kết nối với pincode mặc định: "0000"
* Tự động reconnect trong 30 phút nếu bị đứt kết nối như vượt ra ngoài phạm vi kết nối.

**Chi tiết các chân của module**

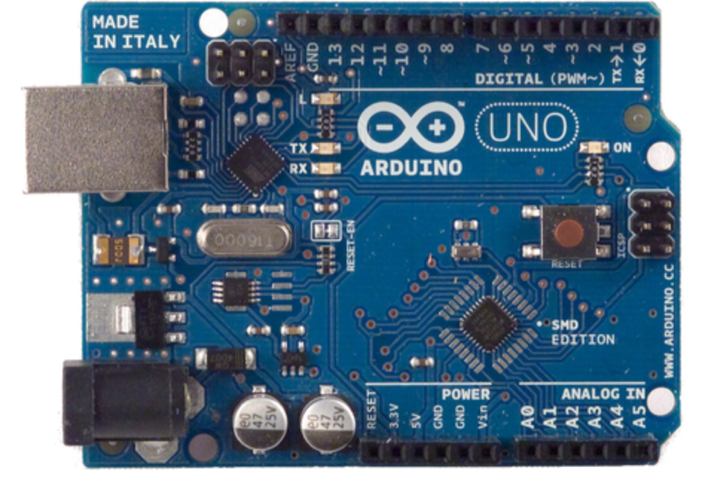
****

Hình 10: Mô hình phần cứng của module

Bảng mô tả chức năng các chân của module:

|  |  |  |  |
| --- | --- | --- | --- |
| Tên chân | STT chân | Loại chân | Mô tả chân |
| GND | 13,21,22 | VSS | Chân đất |
| 3,3v VCC | 12 | Chân 3,3V | Chân nguồn |
| AIO0 | 9 | 2 Hướng | Lập trình I/O |
| AIO1 | 10 | 2 Hướng | Lập trình I/O |
| PIO0 | 23 | 2 Hướng RX EN | Lập trình I/O, điều khiển output cho LNA nếu được trang bị. |
| PIO1 | 24 | 2 Hướng TX EN | Lập trình I/O, điều khiển LA nếu được trang bị. |
| PIO2 | 25 | 2 Hướng | Lập trình I/O |
| PIO3 | 26 | 2 Hướng | Lập trình I/O |
| PIO4 | 27 | 2 Hướng | Lập trình I/O |
| PIO5 | 28 | 2 Hướng | Lập trình I/O |
| PIO6 | 29 | 2 Hướng | Lập trình I/O |
| PIO7 | 30 | 2 Hướng | Lập trình I/O |
| PIO8 | 31 | 2 Hướng | Lập trình I/O |
| PIO9 | 32 | 2 Hướng | Lập trình I/O |
| PIO10 | 33 | 2 Hướng | Lập trình I/O |
| PIO11 | 34 | 2 Hướng | Lập trình I/O |
| ResetB | 11 | Cmos input with weak internal pull-up | Reset nếu LOW |
| UART\_RST | 4 | Cmos output, tri-stable with weak internal pull-up. | UART yêu cầu khi gửi, hoạt động LOW. |
| UART\_CTS | 3 | Cmos input with weak internal pull-down. | UART xóa khi gửi, hoạt động LOW. |
| UART\_RX | 2 | Cmos input with weak internal pull-down | Dữ liệu vào UART |
| UART\_TX | 1 | Cmos output, tri-stable with weak internal pull-up. | Dữ liệu ra UART |
| SPI\_MOSI | 17 | Cmos input with weak internal pull-down. | Serial peripheral interface data input |
| SPI\_CSB | 16 | Cmos input with weak internal pull-up. | Chip select for serial peripheral interface, active low. |
| SPI\_CLK | 19 | Cmos input with weak internal pull-down. | Serial peripheral interface clock. |
| SPI\_MISO | 18 | Cmos input with weak internal pull-down. | Serial peripheral interface data output |
| USB- | 15 | 2 hướng |  |
| USB+ | 20 | 2 hướng |  |
| NC | 14 |  |  |
| PCM\_CLK | 5 | 2 hướng | Synchronous PCM data clock |
| PCM\_OUT | 6 | Cmos output | Synchronous PCM data output |
| PCM\_IN | 7 | Cmos input | Synchronous PCM data input |
| PCM\_SYNC | 8 | 2 hướng | Synchronous PCM data strobe |

* + 1. Arduino UNO R3

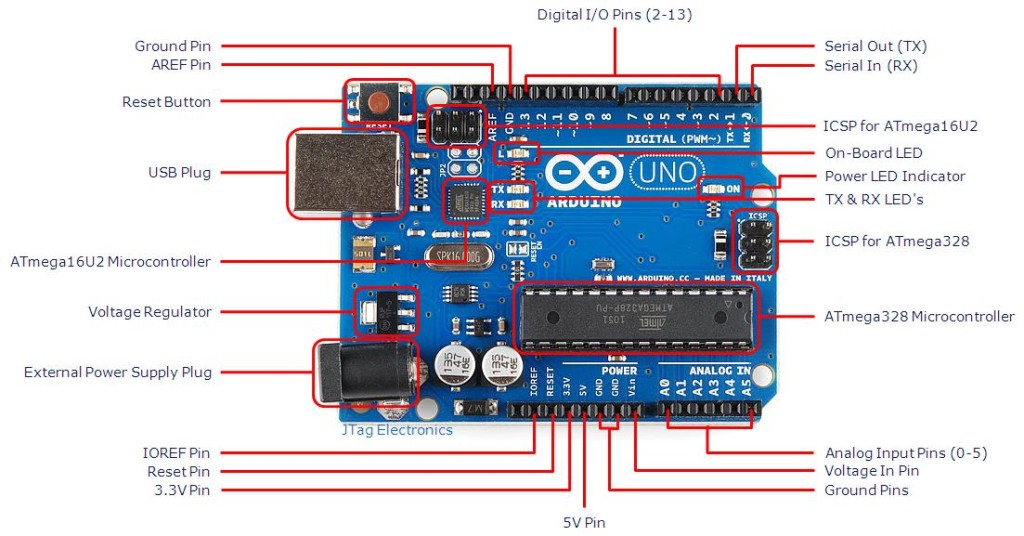


Hình 11: Arduino UNO R3

Arduino UNO có thể sử dụng 3 vi điều khiển họ 8bit AVR là ATmega8, ATmega168, ATmega328. Bộ não này có thể xử lí những tác vụ đơn giản như điều khiển đèn LED nhấp nháy, xử lí tín hiệu cho xe điều khiển từ xa, làm một trạm đo nhiệt độ - độ ẩm và hiển thị lên màn hình LCD....Với chi phí vừa túi tiền đáp ứng theo mỗi nhu cầu người sử dụng, Arduino UNO R3 (thế hệ 3) đang được sử dụng phổ biến hiện nay.

Tiêu hao năng lượng: Arduino UNO có thể được cấp nguồn 5V thông qua cổng USB hoặc cấp nguồn ngoài với điện áp khuyên dùng là 7-12V DC và giới hạn là 6-20V. Thường thì cấp nguồn bằng pin vuông 9V là hợp lí nhất nếu bạn không có sẵn nguồn từ cổng USB. Nếu cấp nguồn vượt quá ngưỡng giới hạn trên, bạn sẽ làm hỏng Arduino UNO. Để biết thêm thông tin chi tiết: <Arduino.vn> .

**Chi tiết các chân của Arduino UNO R3**

****

Hình 12: Thành phần của bo mạch Arduino UNO R3

Ở đây, em sử dụng vi điều khiển ATmega328(họ 8bit):

* + Điện áp: 5V DC(cổng USB).
  + Tần số: 16 Mhz.
* Tiêu thụ: ~30mA.
* Điện áp khuyên dùng: 7-12V DC.
* Điện áp tối đa: ~20V DC (6-20V).
* Digital I/O: 14 (6 chân hardware PWM).
* Analog: 6 (độ phân giải 10bit).
* Dòng tối đa mỗi chân I/O: 30mA.
* Dòng ra tối đa(5V): 500mA.
* Dòng ra tối đa(3.3V): 50mA.
* Bộ nhớ: 32 KB(ATmega328) với 0.5KB dung bởi bootloader.
* SRAM: 2KB(ATmega328).
* EEPROM(ATmega328).

**Các chân năng lượng:**

* **GND (Ground)**: cực âm của nguồn điện cấp cho Arduino UNO. Khi dùng các thiết bị sử dụng những nguồn điện riêng biệt thì những chân này phải được nối với nhau.
* **5V**: cấp điện áp 5V đầu ra. Dòng tối đa cho phép ở chân này là 500mA.
* **3.3V**: cấp điện áp 3.3V đầu ra. Dòng tối đa cho phép ở chân này là 50mA.
* **Vin (Voltage Input)**: để cấp nguồn ngoài cho Arduino UNO, bạn nối cực dương của nguồn với chân này và cực âm của nguồn với chân GND.
* **IOREF**: điện áp hoạt động của vi điều khiển trên Arduino UNO có thể được đo ở chân này. Và dĩ nhiên nó luôn là 5V. Mặc dù vậy không được lấy nguồn 5V từ chân này để sử dụng bởi chức năng của nó không phải là cấp nguồn.
* **RESET**: việc nhấn nút Reset trên board để reset vi điều khiển tương đương với việc chân RESET được nối với GND qua 1 điện trở 10KΩ.

**Bộ nhớ**:

* **32KB bộ nhớ Flash**: những đoạn lệnh bạn lập trình sẽ được lưu trữ trong bộ nhớ Flash của vi điều khiển. Thường thì sẽ có khoảng vài KB trong số này sẽ được dùng cho bootloader nhưng đừng lo, bạn hiếm khi nào cần quá 20KB bộ nhớ này đâu.
* **2KB cho SRAM** (**S**tatic **R**andom **A**ccess **M**emory): giá trị các biến bạn khai báo khi lập trình sẽ lưu ở đây. Bạn khai báo càng nhiều biến thì càng cần nhiều bộ nhớ RAM. Tuy vậy, thực sự thì cũng hiếm khi nào bộ nhớ RAM lại trở thành thứ mà bạn phải bận tâm. Khi mất điện, dữ liệu trên SRAM sẽ bị mất.
* **1KB cho EEPROM** (**E**lectrically **E**raseble **P**rogrammable **R**ead **O**nly **M**emory): đây giống như một chiếc ổ cứng mini – nơi bạn có thể đọc và ghi dữ liệu của mình vào đây mà không phải lo bị mất khi cúp điện giống như dữ liệu trên SRAM.

**Chân digital có chức năng đặc biệt:**

* **2 chân Serial**: 0 (RX) và 1 (TX): dùng để gửi (transmit – TX) và nhận (receive – RX) dữ liệu TTL Serial. Arduino Uno có thể giao tiếp với thiết bị khác thông qua 2 chân này. Kết nối bluetooth thường thấy nói nôm na chính là kết nối Serial không dây. Nếu không cần giao tiếp Serial, bạn không nên sử dụng 2 chân này nếu không cần thiết.
* **Chân PWM (~): 3, 5, 6, 9, 10, và 11**: cho phép bạn xuất ra xung PWM với độ phân giải 8bit (giá trị từ 0 → 28-1 tương ứng với 0V → 5V) bằng hàm analogWrite(). Nói một cách đơn giản, bạn có thể điều chỉnh được điện áp ra ở chân này từ mức 0V đến 5V thay vì chỉ cố định ở mức 0V và 5V như những chân khác.
* **Chân giao tiếp SPI:** 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).  Ngoài các chức năng thông thường, 4 chân này còn dùng để truyền phát dữ liệu bằng giao thức SPI với các thiết bị khác.
* **LED 13**: trên Arduino UNO có 1 đèn led màu cam (kí hiệu chữ L). Khi bấm nút Reset, bạn sẽ thấy đèn này nhấp nháy để báo hiệu. Nó được nối với chân số 13. Khi chân này được người dùng sử dụng, LED sẽ sáng.
  + 1. Servo RC

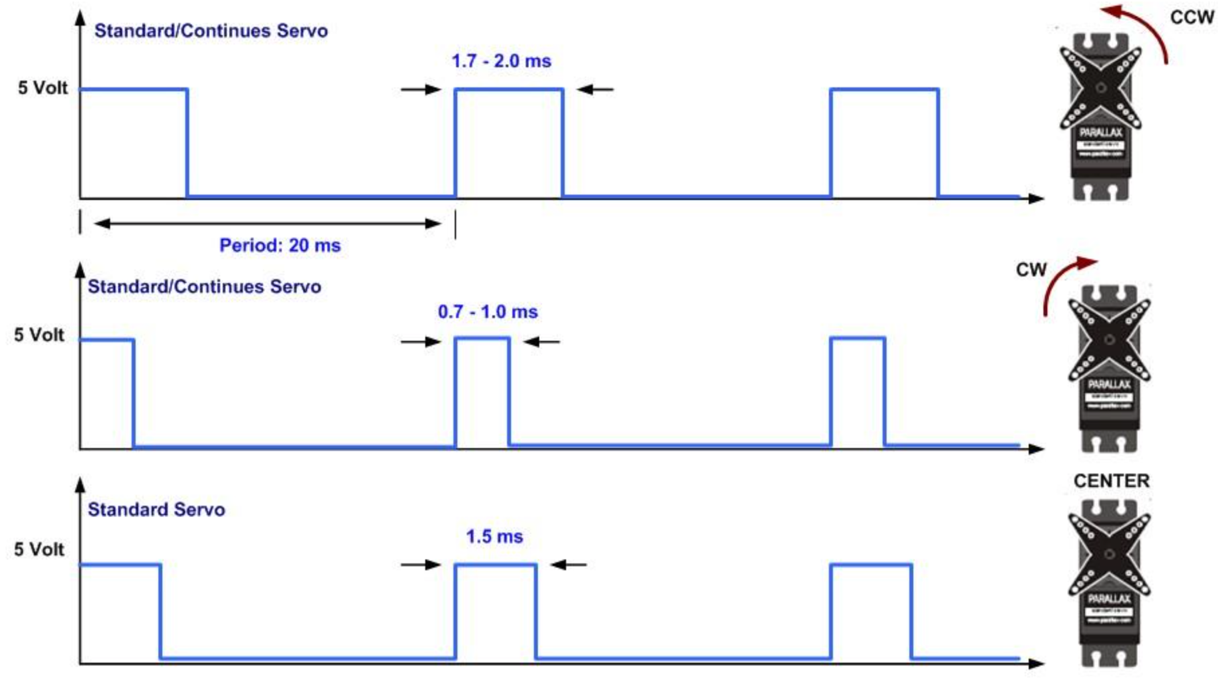


Hình 13: Động cơ Servo RC

Trên thị trường hiện nay, có rất nhiều loại Servo của nhiều hãng khác nhau đáp ứng mọi nhu cầu của người dùng.

RC servo là thiết bị có gắn động cơ mini và các hệ bánh răng giảm tốc, mục đích giúp trục quay đến góc mong muốn mà vẫn giữ được góc chính xác và không bị trôi lệch, ngoài ra RC servo còn có khả năng chịu tải lớn hơn rất nhiều lần so với trọng lượng của nó. RC servo thường được ứng dụng trong robotic, đồ chơi mô hình RC.

Hoạt động của RC servo dựa trên nguyên lý nhận xung PWM và cho ra góc quay. RC servo chỉ có thể xoay ở góc cố định có nghĩa là không thể xoay quanh trục như những loại động cơ bình thường. Tùy loại RC servo mà góc quay hoạt động được 90o hay 180o, đa phần thì 90o.



Hình 14: Sơ đồ minh họa động cơ Servo RC

Ta thấy được chu kỳ của PWM là 20ms (50 Hz), thời gian mức cao sẽ quyết định góc quay của RC servo dao động từ 1ms -> 2ms <=> 0 -> 90o.

* 1. Yêu cầu hệ thống
     1. Chức năng đóng/mở Bluetooth

Chức năng giúp người dùng có thể đóng/mở Bluetooth điện thoại bằng nút nhấn.

* + 1. Chức năng tìm Bluetooth

Chức năng giúp người dùng tìm đến thiết bị xung quanh có Bluetooth để kết nối với chương trình ứng dụng của điện thoại bằng cách nhấn nút “Tìm”.

* + 1. Chức năng kết nối với Bluetooth

Chức năng này bằng cách nhấn vào “Tên” và “Địa chỉ” của thiết bị Bluetooth đã tìm được để tiến hành kết nối. Yêu cầu của kết nối là phải nhập password đúng của thiết bị đã tìm được.

* + 1. Chức năng điều khiển thiết bị có Bluetooth

Chức năng sẽ hoạt động được sau khi đã kết nối thành công với thiết bị khác và người dùng sẽ thực hiện việc điều khiển thông qua các nút nhấn để gửi tính hiệu thông qua Bluetooth.

* 1. Một số ứng dụng trong chương trình
     1. BluetoothDevice

BluetoothDevice là lớp đại diện cho các thiết bị Bluetooth bên ngoài. Nó có thể tạo kết nối với thiết bị điện thoại, hay truy vấn thông tin về Tên, Địa chỉ, hay trạng thái đang hoạt động của Bluetooth đó.

Bằng cách gọi BluetoothDevice device;

device.getName();//Lấy tên của thiết bị Bluetooth

device.getAddress();//Lấy địa chỉ của thiết bị Bluetooth

device.getBondState();//Lấy trạng thái liên kết của thiết bị Bluetooth

Link tham khảo: https://developer.android.com/reference/android/bluetooth/BluetoothDevice.html

* + 1. BluetoothAdapter

BluetoothAdapter là lớp đại diện cho thiết bị Bluetooth máy điện thoại. Đây là lớp có thể giúp các lập trình viên thực hiện các nhiệm vụ của Bluetooth trên máy điện thoại như là mở tắt Bluetooth, hay tìm kiếm các thiết bị khác có thể kết nối Bluetooth. Lập các kết nối với thiết bị khác thông qua lớp BluetoothServerSocket để lắng nghe tất cả các yêu cầu kết nối hiện tại.

Để thực hiện các thao tác trên Bluetooth ta phải khai báo hàm getDefaultAdapter() để lấy quyền truy cập Bluetooth. Mà thực hiện việc trên phải khai báo quyền truy cập BLUETOOTH\_ADMIN permission trong file AndroidManifest.xml, các phiển bản yêu cầu JELLY\_BEAN\_MR1 và các bản về trước để sử dụng.

Link tham khảo: https://developer.android.com/reference/android/bluetooth/BluetoothAdapter.html

* + 1. BroadcastReceiver

BroadcastReceiver là một trong 4 thành phần lớn trong Android, với mục đích là lắng nghe các sự kiện, trạng thái của hệ thống phát ra thông qua Intent nhờ đó mà các lập trình viên có thể xử lý được các kiện hệ thống ở bên trong ứng dụng của mình. Nó có thể hoạt động được cả khi ứng dụng bị tắt đi và có thể dùng chung với service. Có 2 cách để đăng ký Android broadcast receiver:

* Một là phương thức tĩnh khi đăng ký broadcast receiver trong ứng dụng thông qua file AndroidManifest.xml.
* Hai là phương thức động để đăng ký broadcast receiver thông qua việc sử dụng method Context.registerReceiver(). Các broadcast receiver được đăng ký động có thể được gỡ bởi method Context.unregisterReceiver().

Link tham khảo:

https://developer.android.com/reference/android/content/BroadcastReceiver.html

* + 1. UUID

UUID là viết tắt của Universally Unique Identifier, nó là một chuỗi 128-bit dùng để xác định thông tin trong hệ thống thiết bị đang sử dụng.

Một chuỗi UUID cơ bản có dạng như sau: ***6ba7b810-9dad-11d1-80b4-00c04fd430c8*** gồm 36 kí tự, 32 ký tự thỏa mãn [a-z 0-9] và chia thành 4 nhóm bởi 4 dấu “-”. Ví dụ sau đây sẽ giải thích về UUID:

Bạn đang làm một ứng dụng chat mà sử dụng Bluetooth để chat với một người ở gần đó. Để tìm người trò chuyện, bạn sẽ muốn tìm kiếm các thiết bị khác có cài ứng dụng Chat của bạn. Để có thể tìm kiếm được các thiết bị đó, chúng ta sẽ tìm kiếm các UUID trong danh sách dịch vụ của các thiết bị ở gần đó. Sử dụng một UUID để lắng nghe và chấp nhận các kết nối Bluetooth, sau đó sẽ tự động thêm UUID vào danh sách các dịch vụ của điện thoại hoặc dịch vụ phát hiện giao thức.

Link tham khảo:

https://en.wikipedia.org/wiki/Universally\_unique\_identifier

* + 1. BluetoothServerSocket và BluetoothSocket

Kết nối Bluetooth hoạt động cũng giống với các kết nối khác (Socket, Http...). Phải có một máy chủ (server) và một máy khách (client) và giao tiếp thông qua RFCOMM sockets. Trên Android, RFCOMM sockets được đại diện bởi object BluetoothSocket.

Là một interface cho Socket Bluetooth tương tự như TCP socket gồm: Socket và ServerSocket. Về phía máy chủ, sử dụng một BluetoothServerSocket để tạo ra một máy chủ lắng nghe. Khi một kết nối được chấp nhận bởi BluetoothServerSocket, nó sẽ trả về một BluetoothSocket để quản lý kết nối. Về phía máy khách, sử dụng một BluetoothSocket duy nhất cho cả hai bắt đầu một kết nối mới và quản lý kết nối. Android cũng sử dụng cơ chế client – server, khi máy chủ và máy khách có chung một kết nối BluetoothSocket trên cùng một kênh RFCOMM thì 2 thiết bị được kết nối với nhau. Lúc này mỗi thiết bị có thể chuyển giao dữ liệu cho nhau. Mở một BluetoothServerSocket bởi lời gọi hàm listenUsingRfcommWithServiceRecord(String, UUID):

- String là tên bất kì có thể là tên thiết bị hay ứng dụng.

- UUID là một mã 128 bit định dạng là một chuỗi ID để sử dụng xác định thông tin hay xác định ứng dụng bluetooth.

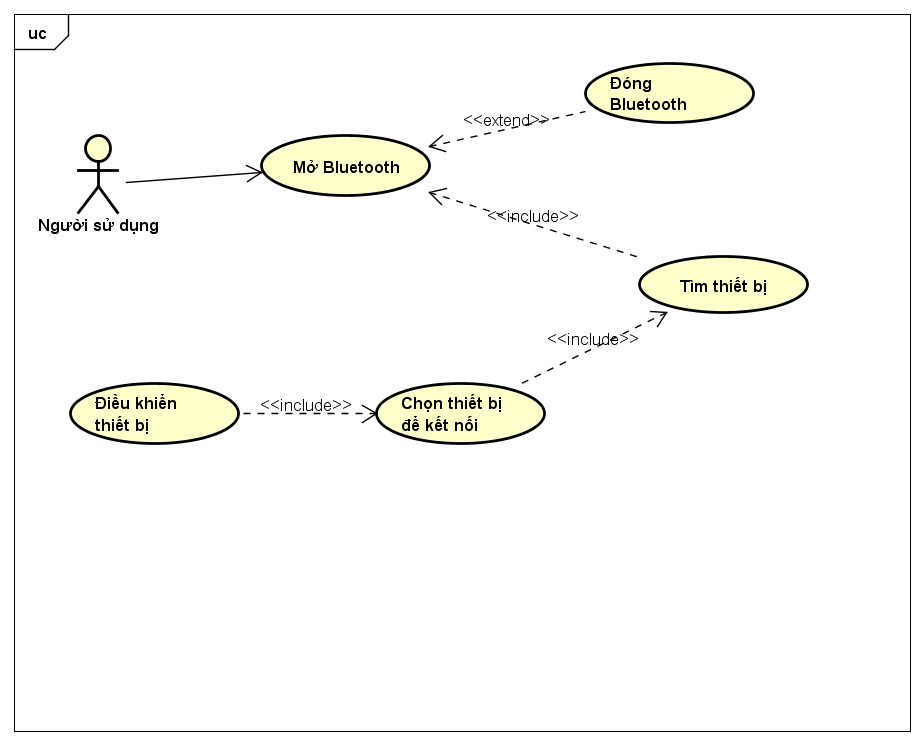
Bắt đầu lắng nghe và chấp nhận kết nối bằng lời gọi accept().

Đóng kết nối bằng lời gọi close().

Link tham khảo:

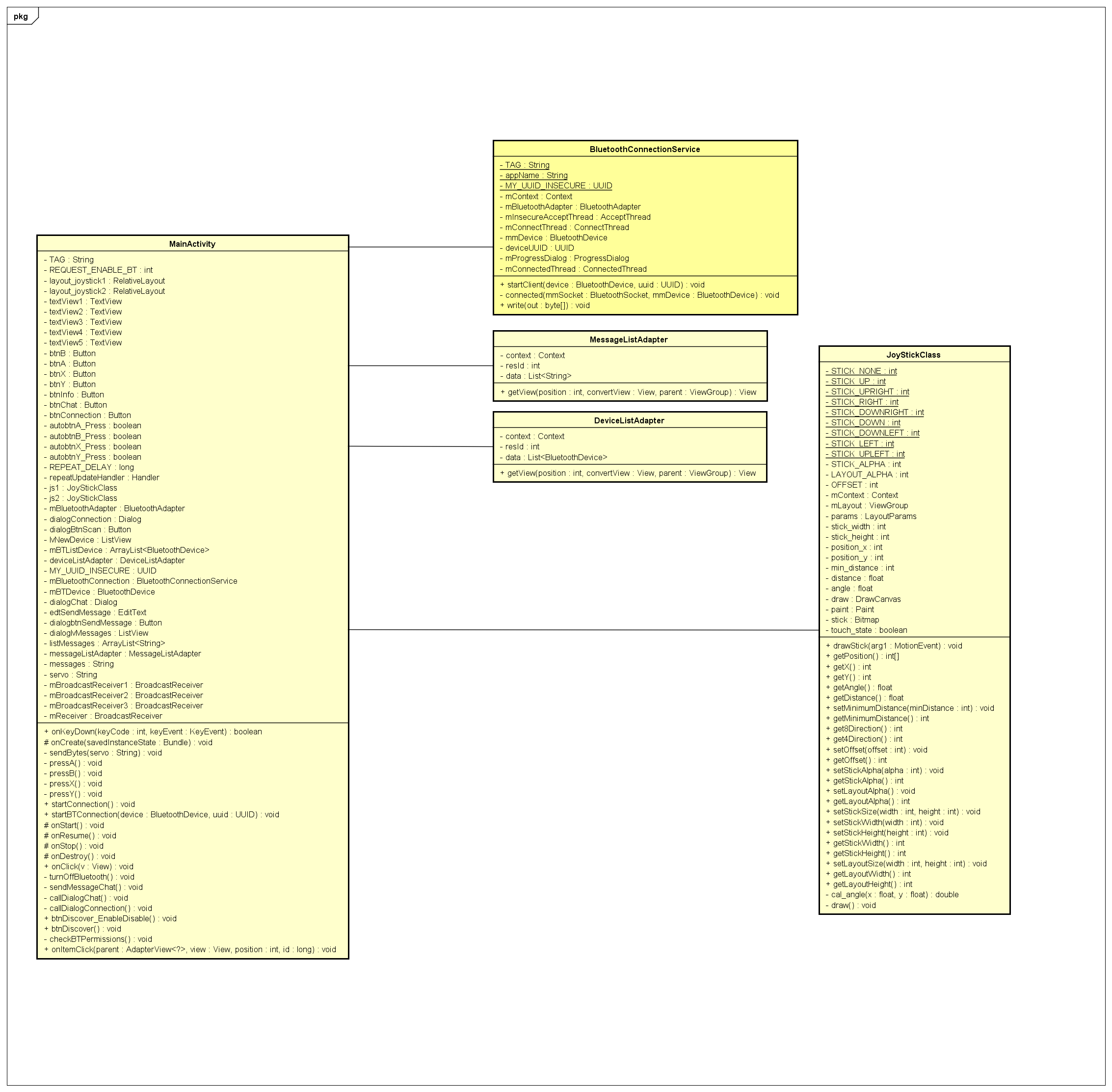
http://developer.android.com/guide/topics/wireless/bluetooth.html

* 1. Thiết kế hệ thống
     1. Sơ đồ Use-Case



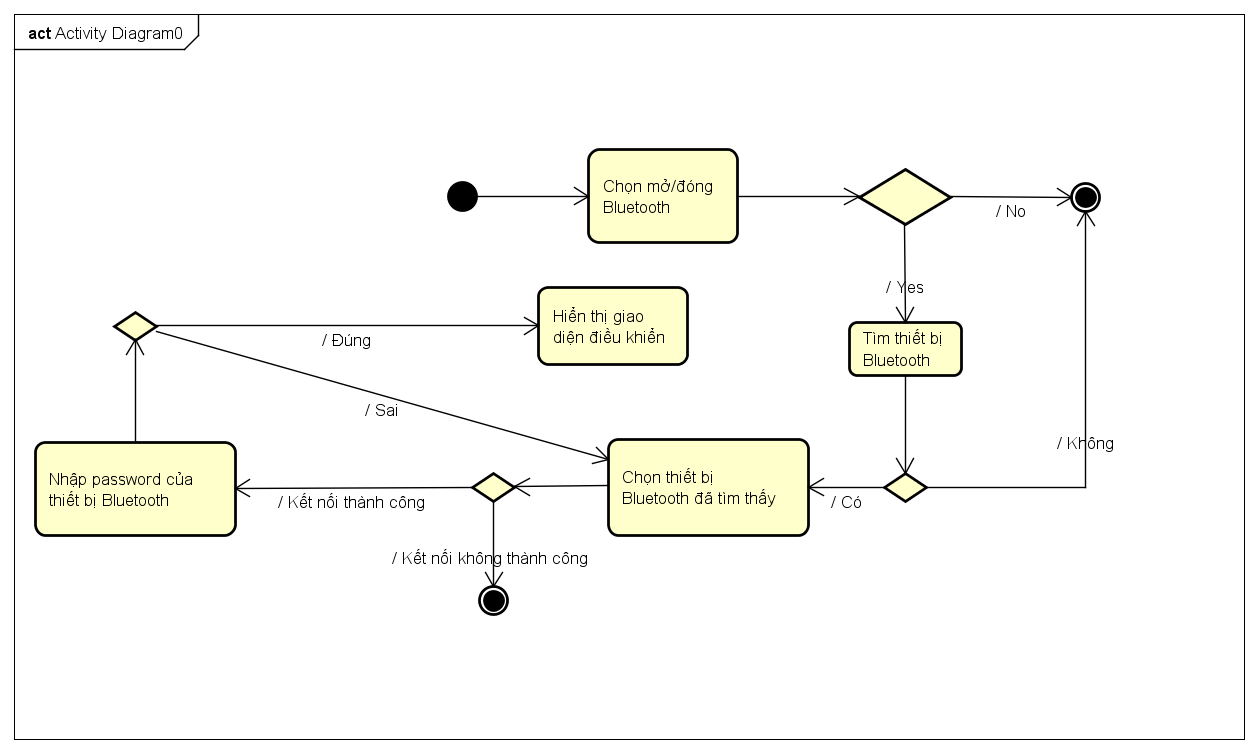
Hình 15: Sơ đồ Use-Case

* + 1. Sơ đồ Class



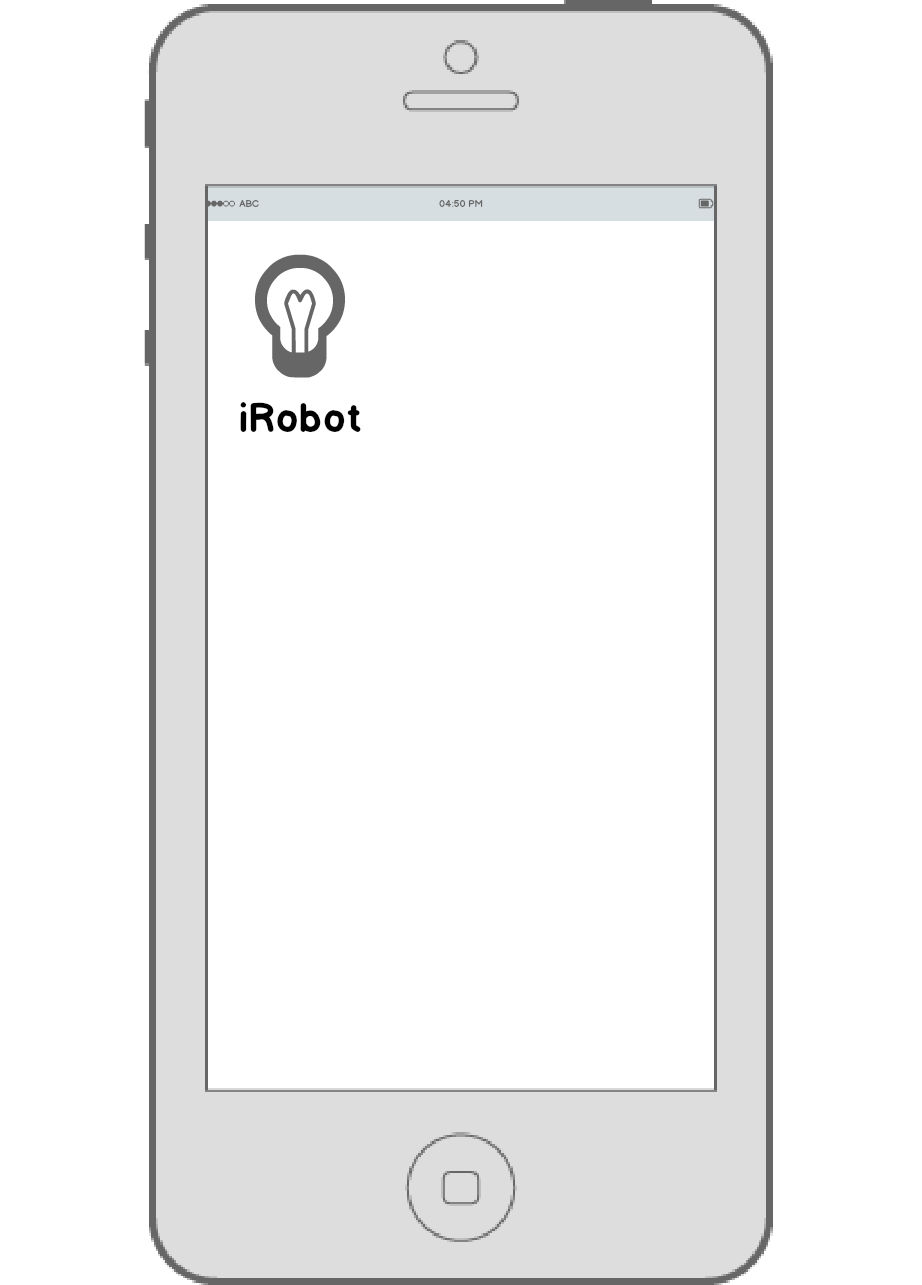
Hình 16: Sơ đồ Class

* + 1. Sơ đồ Activity



Hình 17: Sơ đồ Activity

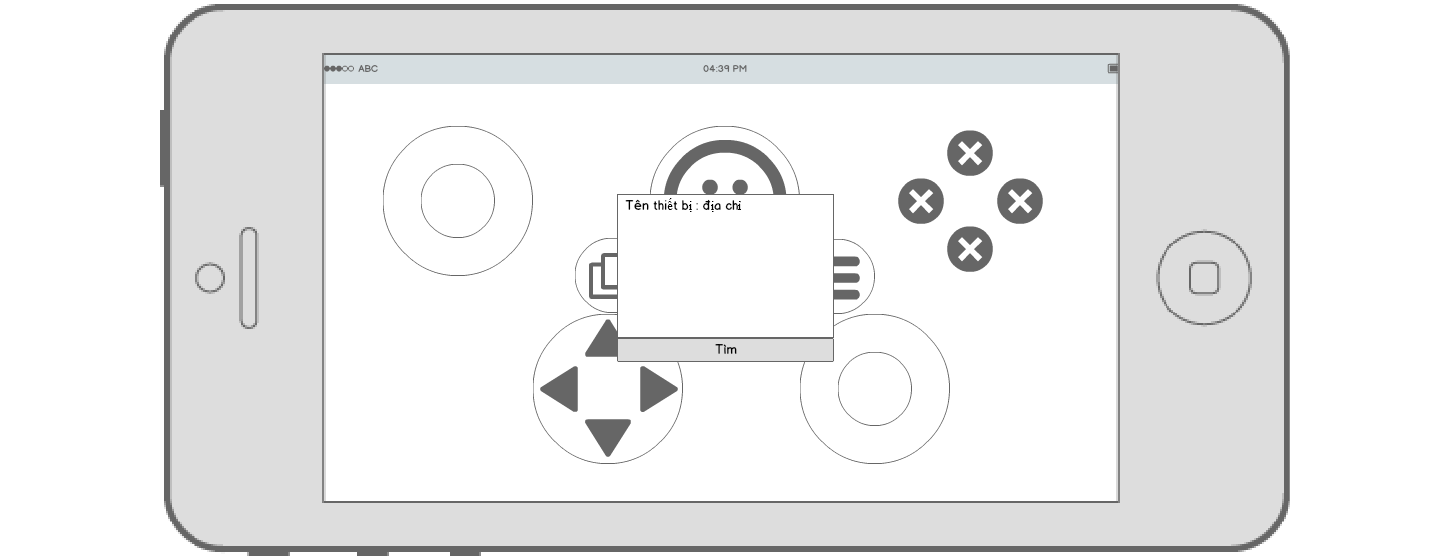
* + 1. Giao diện hệ thống



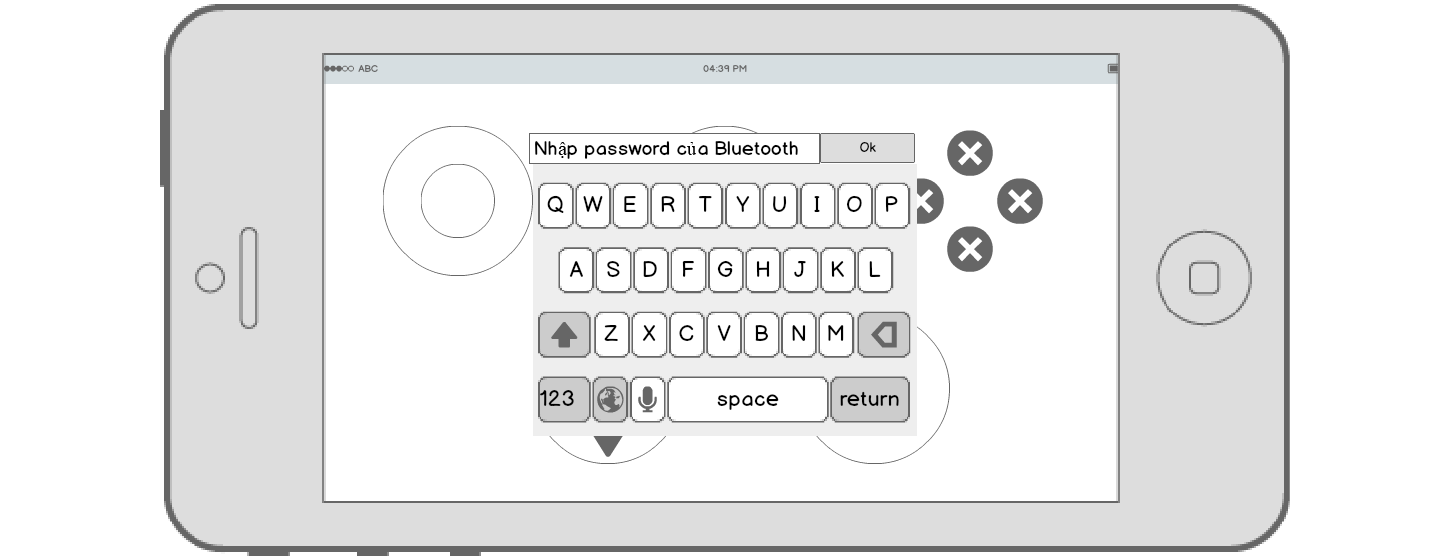
Hình 18: Màn hình giao diện ứng dụng



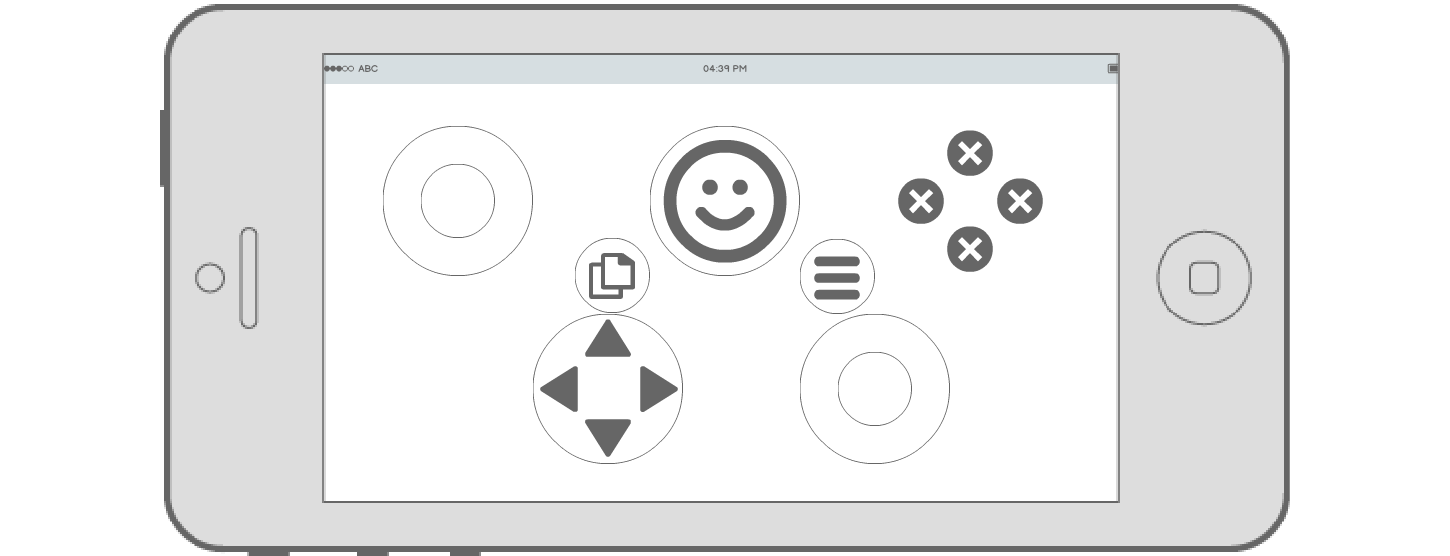
Hình 19: Màn hình giao diện yêu cầu mở Bluetooth



Hình 20: Màn hình tìm và chọn thiết bị Bluetooth khác



Hình 21: Màn hình nhập password để kết nối với thiết bị Bluetooth đã chọn



Hình 22: Màn hình ứng dụng

1. QUÁ TRÌNH THỰC HIỆN.

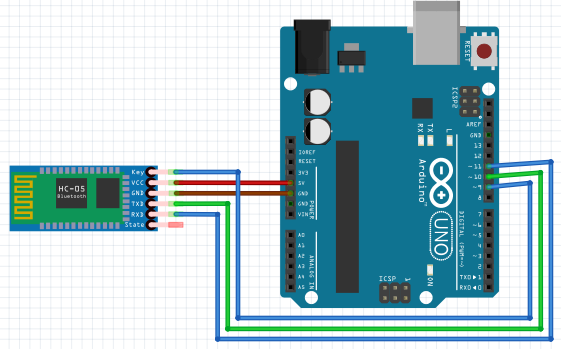
Việc cài đặt các chương trình cần thiết là:

* Android SDK
* Arduino 1.8.1
* sketch Fritzing (để vẽ hình ảnh)

1. **Cài đặt các chế độ cho thiết bị Bluetooth bằng Arduino lên bo mạch Arduino R3 Uno:**

Mở file kèm theo để tùy chỉnh thiết bị:

“HC-0305\_serial\_module\_AT\_commamd\_set\_201104\_revised.pdf”



Hình 23: Sơ đồ nối dây của Bluetooth HC-05 với bo mạch Arduino

Tùy chỉnh cấu hình và thông tin về Bluetooth HC-05:

* STATE: Kiểm tra có kết nối hay không
* Vcc: +5V
* RXD: Truyền dữ liệu tới cổng Serial của HC-05
* TXD: Truyền dữ liệu qua cổng Serial từ HC-05
* KEY: Cấu hình HC-05 ở chế độ Command Mode

AT (lệnh TEST, nếu kết quả trả về OK kết nối thành công với HC-05 qua cổng Serial)

AT+VERSION? (trả về phiên bản phần mềm của HC-05)

AT+ORGL: Reset lại cài đặt mặc định  
AT+RMAAD: xóa tất cả những thiết bị đã paired với HC-05  
AT+ROLE=0: Thiết lập Slave  
AT+ADDR: Hiển thị địa chỉ của SLAVE

AT+UART=9600,0,0 (Thiết lập baud rate to 9600, 1 stop bit, no parity)

AT+NAME=Sugar-05: Đặt tên thiết bị là Sugar-05

AT+PSWD=1234: Thiết lập password là 1234

Mở file “cauhinhBluetoothHC05.txt” để nạp code vào bo mạch.

1. **Sau khi cài đặt và cấu hình xong thiết bị thì nạp lại code cho bo mạch trên để kiểm tra.**

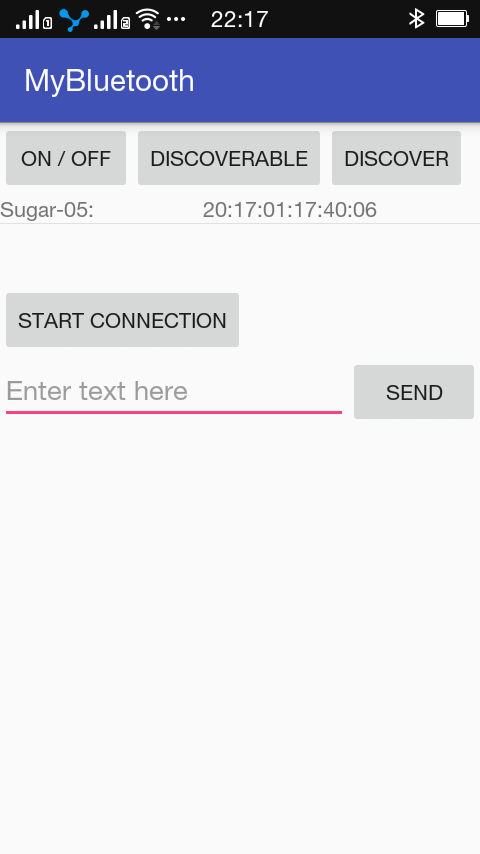
Đầu tiên nạp lại code cho mạch để kiểm tra, mở file “naplaicodekiemtra.txt” để nạp code vào bo mạch.

Sau khi nạp code thành công thì tạo chương trình trên điện thoại để tiến hành demo thử. Chương trình được tạo bằng Android SDK và có giao diện như sau:

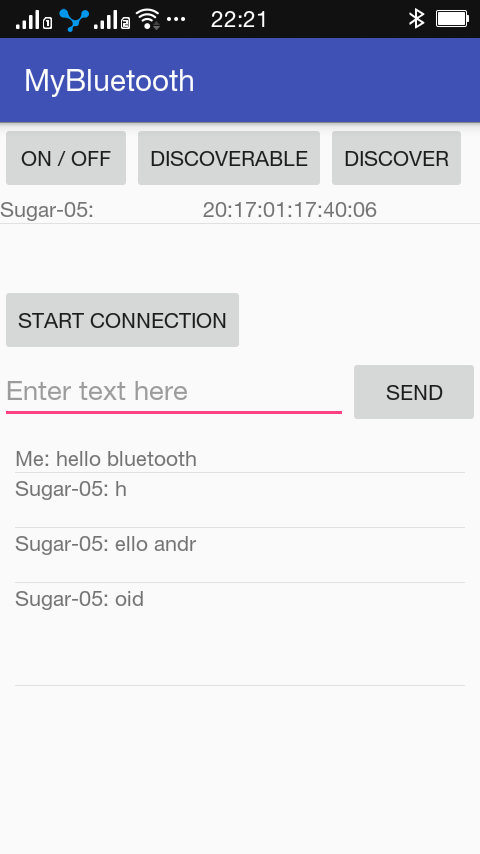


Hình 24: Giao diện chương trình demo kết nối với Bluetooth HC-05 để kiểm tra

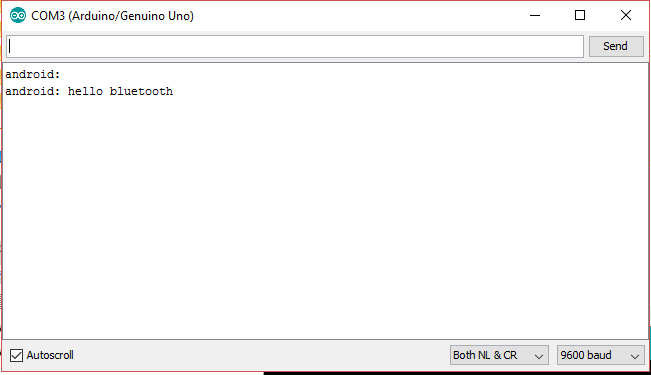
***Quá trình kiểm tra:***



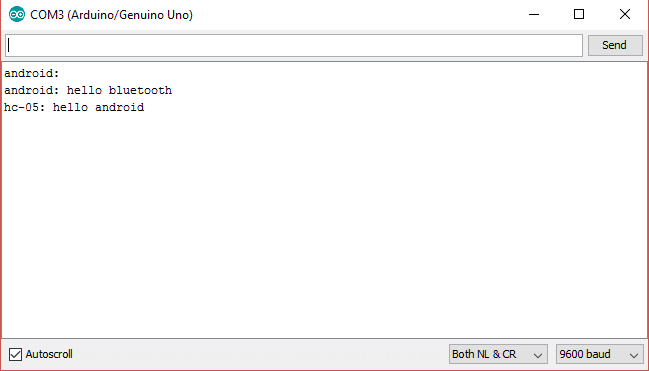
Hình 25: Tìm kiếm và kết nối với thiết bị Bluetooth HC-05



Hình 26: Giao tiếp từ ứng dụng



Hình 27: Nhận tính hiệu từ ứng dụng demo gửi qua



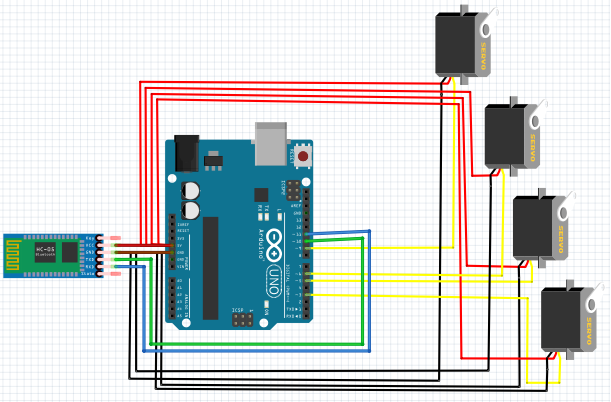
Hình 28: Gửi tính hiệu cho ứng dụng demo

Đây là link ứng dụng demo:

https://drive.google.com/drive/folders/0B9ENsfY5A63TdGQwVXBFRzI4dDA?usp=sharing

1. **Sau khi demo thành công, nhóm bắt đầu tiến hành xây dựng hệ thống.**

Đầu tiên là nối dây và nạp code vào bo mạch để điều khiển động cơ servo, mở file “dieukhienServo.txt” rồi nạp code vào bo mạch:



Hình 29: Cách nối dây với các động cơ Servo

Sau khi nạp code vào bo mạch thành công, nhóm bắt đầu thực hiện xây dựng ứng. Và đây là link của ứng dụng:

https://drive.google.com/drive/folders/0B9ENsfY5A63TWjhnUlZfODdhYVU?usp=sharing

1. TỔNG KẾT.
   1. Đánh giá kết quả thực hiện đồ án

Đồ án **“ĐIỀU KHIỂN CÁNH TAY ROBOT THÔNG QUA ỨNG DỤNG TRÊN ĐIỆN THOẠI DI ĐỘNG”,** như đã trình bày là một đề tài có tính ứng dụng cao, có thể sử dụng cho nhiều bài toán thực tế như điều khiển các thiết bị trong nhà bằng Mobile Android, giám sát và điều khiển các thiết bị báo cháy, báo ga, ổn định nhiệt độ…. Việc xây dựng thiết bị liên quan đến nhiều mảng kiến thức, từ những kiến thức của mảng điện tử, hệ thống nhúng cho tới những kiến thức về Android.

Vận dụng những kiến thức đã học cùng sự cố gắng của em, em đã thực hiện được những phần sau đây:

* Nắm rõ được giao tiếp Bluetooth.
* Thực hiện viết ứng dụng trên Mobile Android.
* Thực hiện kết nối Bluetooth giữa Mobile Android và module Bluetooth.
* Thực hiện kết nối giữa module Bluetooth và vi điều khiển Arduino UNO R3.
* Thực hiện điều khiển vi điều khiển Arduino UNO R3 bằng Mobile Android thông qua module Bluetooth HC-05.

Và còn một số nhược điểm sau:

* Giao diện ứng dụng không thỏa được UI/UX.
* Do hoạt động bằng Bluetooth nên smart phone nhanh hết pin và khoảng cách không được xa (tầm 10m).
  1. Hướng phát triển

Đồ án đã đạt được một số kết quả nhất định, song để đáp ứng cho một bài toán thực tế còn chưa thật rối ưu. Sau đây là một số hướng phát triển mà em dự định sẽ làm:

* Tiếp tục nghiên cứu những ứng dụng của Hệ điều hành thời gian thực trong các hệ thống phức tạp và đặc biệt là trong lĩnh vực hệ thống nhúng, một lĩnh vực rất được quan tâm hiện nay.
* Thiết kế phần cơ khí của ROBOT cho phù hợp với các bài toán thực tế.
* Tích hợp thêm việc truyền hình ảnh từ ROBOT lên điện thoại.
* Điều khiển qua Wifi & truyền video.
* Phát triển thêm các tính năng mới cho ứng dụng.
* Phát triển thêm các chức năng cho robot như đo khoảng cách, đo độ nghiêng…

Tài liệu tham khảo

1. http://*youtube.com*
2. http://*developer.android.com*
3. <http://icviet.vn>
4. <http://arduino.vn>
5. [http://*stackoverflow.com*](http://stackoverflow.com)