

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



HCMUTE

ĐỒ ÁN 1

HỆ THỐNG BƠM NƯỚC TỰ ĐỘNG CHO HỒ CÁ

NHÓM SINH VIÊN THỰC HIỆN	MSSV
1. LÊ MINH HẢI ĐĂNG	20119217
2. PHẠM QUỐC VIỆT	20119307
NGÀNH:	KỸ THUẬT MÁY TÍNH
GVHD:	PGS.TS Trương Ngọc Sơn
HỌC KỲ:	2 (2022-2023)

TP. HCM 4/2023

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



HCMUTE

ĐỒ ÁN 1
HỆ THỐNG BƠM NƯỚC TỰ ĐỘNG CHO HỒ CÁ

NHÓM SINH VIÊN THỰC HIỆN	MSSV
1. LÊ MINH HẢI ĐĂNG	20119217
2. PHẠM QUỐC VIỆT	20119307
NGÀNH:	KỸ THUẬT MÁY TÍNH
GVHD:	PGS.TS Trương Ngọc Sơn
HỌC KỲ:	2 (2022-2023)

TP. HCM 3/2023

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Ký tên

PGS.TS Trương Ngọc Sơn

NHIỆM VỤ ĐỒ ÁN 1

Họ và tên sinh viên 1: Lê Minh Hải Đăng MSSV: 20119217

Họ và tên sinh viên 2: Phạm Quốc Việt MSSV: 20119307

Ngành: Công Nghệ Kỹ Thuật Máy Tính

Giảng viên hướng dẫn: TS. Trương Ngọc Sơn

Ngày chọn đề tài: 1/3/2023

Ngày nộp đề tài:

1. Tên đề tài: **HỆ THỐNG BƠM NƯỚC TỰ ĐỘNG CHO HỒ CÁ**

2. Mô tả hoạt động của hệ thống: Trung tâm của hệ thống là module Arduino sẽ thu thập và xử lý dữ liệu từ cảm biến siêu âm để biết được mực nước hiện tại so với mực nước tiêu chuẩn, nếu nước nhiều thì tiến hành xả nước hoặc nước ít hơn thì tiến hành bơm nước vào, có nút nhấn chuyển đổi giữa tự động và thủ công để dễ dàng bơm xả nước. Hiện thị trạng thái hoạt động của hệ thống (đang bơm, đang ngưng, v.v.) trên một màn hình LCD.

3. Yêu cầu đề tài:

Có thiết kế mạch điện tử, cho phép sử dụng module arduino, hệ thống phải có đầy đủ các thiết bị ngõ vào: nút nhấn, sensor, ngõ ra: hiển thị led 7 đoạn hoặc lcd, 1 hoặc nhiều relay, 1 hoặc nhiều động cơ. Từ các yêu cầu đó, sinh viên lựa chọn đề tài. Các nhóm vẽ mạch nguyên lý và pcb phải sử dụng phần mềm Altium, thực hiện gia công mạch in và lắp đặt linh kiện, mặt mạch in phải có tên của sinh viên trong nhóm.

4. Nội dung thực hiện đề tài:

THỨ TỰ	CÔNG VIỆC	Đăng	Việt
1			
2			
3			

5. Sản phẩm: **HỆ THỐNG BƠM NƯỚC TỰ ĐỘNG CHO HỒ CÁ**

LỜI CẢM ƠN

Người thực hiện đề tài
Phạm Quốc Việt
Lê Minh Hải Đăng

CHƯƠNG 1: TỔNG QUAN

1.1. Giới thiệu tình hình thực tế

Nuôi cá ở hồ ngoài trời là một sở thích được nhiều người Việt Nam yêu thích hiện nay. Đặc biệt, trong những năm gần đây, với sự phát triển của ngành kinh doanh cá cảnh và nhu cầu giải trí của người dân đang tăng cao, việc nuôi cá trong các hồ ngoài trời đã trở thành một xu hướng phổ biến. Các loại cá phổ biến để nuôi ở hồ ngoài trời ở Việt Nam bao gồm cá koi, cá vàng, cá chép, cá rồng và cá hoàng đế.

Tuy nhiên, để nuôi cá trong hồ ngoài trời thành công, cần phải đảm bảo môi trường sống cho cá được tốt nhất có thể. Trong đó, hệ thống bơm xả nước tự động là một trong những yếu tố quan trọng để giữ cho chất lượng nước trong hồ luôn được duy trì ổn định. Việc sử dụng hệ thống bơm xả nước tự động trong việc nuôi cá ở hồ ngoài trời không chỉ giúp cho việc chăm sóc cá trở nên dễ dàng hơn mà còn giảm thiểu tối đa những rủi ro trong việc nuôi cá. Điều này giúp cho việc nuôi cá trong hồ ngoài trời trở nên hiệu quả hơn và đem lại sự thỏa mãn và hạnh phúc cho người nuôi.

1.2. Tính cấp thiết của đề tài

Hệ thống bơm xả nước tự động cho hồ cá giúp loại bỏ các chất thải và tái tạo oxy trong nước. Hệ thống bơm sẽ kích hoạt để bơm nước từ hồ ra ngoài, đồng thời bơm nước mới vào hồ để thay thế. Việc thay nước thường xuyên và đúng cách giúp giảm thiểu nồng độ các chất độc hại trong nước, giúp cá có môi trường sống tốt hơn.

Ở nơi có thời tiết nắng nóng dai dẳng như ở Việt Nam, việc hao hụt nước nhanh chóng trong hồ cá ngoài trời là việc hiển nhiên, vì vậy, hệ thống bơm nước tự động sẽ bơm nước vào khi phát hiện sự hao hụt lượng nước so với mực nước chuẩn. Ngược lại, khi trời mưa dai dẳng, lượng nước trong hồ cũng sẽ dâng lên, gây ra tràn nước, trong hồ khiến cá bị tràn ra bên ngoài hồ tác động nguy hiểm đến cá, vì vậy, hệ thống có mặt sẽ giúp xả nước ra ngoài khi phát hiện lượng nước quá mức tiêu chuẩn. Điều này sẽ có ích cho việc ổn định lượng nước trong hồ cho cá và chống sự hao hụt nước do thời tiết.

Ngoài ra, bằng cách sử dụng hệ thống bơm xả nước tự động, bạn không cần phải kiểm soát mức nước trong hồ cá thủ công. Điều này giúp tiết kiệm thời gian và công sức cho người chăm sóc hồ cá. Do đó, việc trang bị hệ thống bơm xả nước tự động là việc quan trọng giúp hỗ trợ hiệu quả cho người nuôi cá. Từ những lý do trên nhóm em chọn đề tài

“hệ thống bơm xả nước tự động” với mục đích nghiên cứu và thiết kế hệ thống quan trọng này làm đề tài cho Đồ án 1 của mình.

1.3. Mục tiêu thiết kế

- Cảm biến siêu âm sẽ thu thập dữ liệu về khoảng cách đến mặt nước.
- Hệ thống hoạt động ngay lập tức khi nước không ở mức nước tiêu chuẩn
- Màn hình LCD hiển thị mực nước và trạng thái hoạt động của hệ thống (tự động và thủ công).
- Chuyển trạng thái từ thủ công sang tự động và ngược lại.
- Trạng thái tự động: vi điều khiển sẽ xử lý dữ liệu từ cảm biến siêu âm và điều khiển bơm nước.
- Trạng thái thủ công: trực tiếp bơm hoặc xả nước khi nhấn nút.
- Cảm biến siêu âm chống nước.
- Điều khiển hệ thống từ xa (phạm vi nhà) thông qua Bluetooth trên ứng dụng điện thoại thông minh và thấy được mực nước của hồ.

1.4. Đối tượng và phạm vi nghiên cứu

- Lập trình nhúng với Module Arduino Uno R3
- Giao tiếp cảm biến siêu âm
- Giao tiếp với module Bluetooth HC-06
- Lập trình và thiết kế giao diện app điện thoại bằng MIT app inventor
- Gia công mạch in

1.5. Phương pháp nghiên cứu

Để thực hiện ý tưởng đề tài đề ra ban đầu em đã sử dụng một số phương pháp sau:

- Phương pháp tổng hợp tài liệu lý thuyết, những kiến thức đã được học.
- Phương pháp thử nghiệm.
- Phương pháp tham khảo ý kiến người hướng dẫn.

1.6. Bố cục đề tài

Bố cục Đồ án được chia thành 6 chương như sau:

- Chương 1: Tổng quan.
- Chương 2: Cơ sở lý thuyết.
- Chương 3: Thiết kế kiến trúc hệ thống

- Chương 4: Kết quả và đánh giá.
- Chương 5: Kết luận và hướng phát triển.

Phụ lục

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Giới thiệu phần cứng

2.1.1. Module Arduino Uno R3

- Giới thiệu:

Arduino UNO R3 là kit Arduino UNO thế hệ thứ 3, với khả năng lập trình cho các ứng dụng điều khiển phức tạp do được trang bị cấu hình mạnh cho các loại bộ nhớ ROM, RAM và Flash, các ngõ vào ra digital I/O trong đó có nhiều ngõ có khả năng xuất tín hiệu PWM, các ngõ đọc tín hiệu analog và các chuẩn giao tiếp đa dạng như UART, SPI, TWI (I2C).



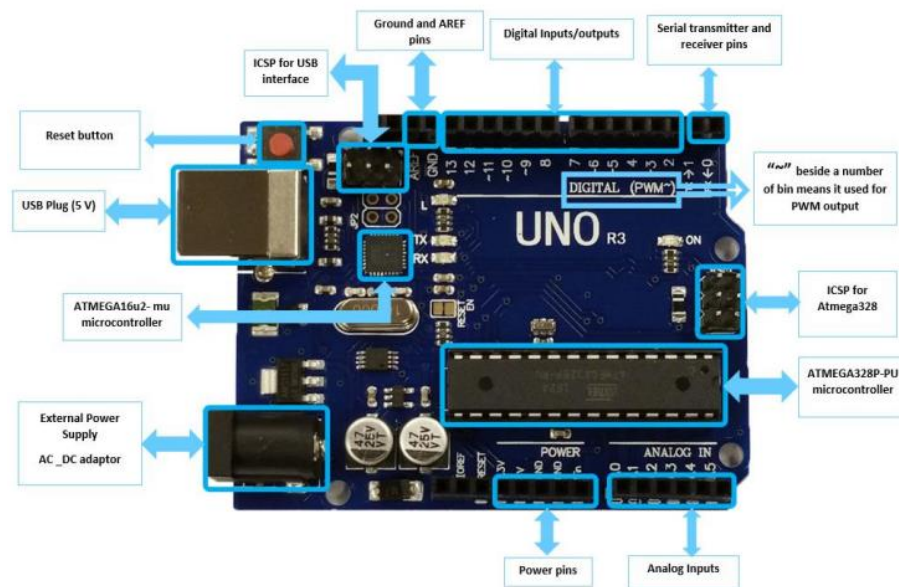
Hình 2.1. Kit Arduino Uno R3

Arduino Uno R3 được hỗ trợ bởi một cộng đồng lớn với nhiều nguồn tài liệu, ví dụ như các ví dụ code, thư viện và các hướng dẫn sử dụng. Điều này làm cho Arduino Uno R3 trở thành một lựa chọn phổ biến cho các dự án điện tử của người mới bắt đầu và các nhà phát triển kinh nghiệm.

Arduino Uno R3 có kích thước nhỏ gọn, chỉ khoảng 68,6 mm x 53,4 mm, điều này giúp nó trở thành một board dễ dàng để kết nối với các linh kiện điện tử khác. Board cũng có một cổng cắm ICSP (In-Circuit Serial Programming) để nạp chương trình trực tiếp vào vi điều khiển.

Ngoài những tính năng chính đã được đề cập ở trên, Arduino Uno R3 còn có nhiều tính năng và ưu điểm khác như:

- Board có kích thước nhỏ gọn, giá cả phải chăng và dễ dàng sử dụng, vì vậy nó rất phù hợp cho các dự án DIY hoặc các dự án có quy mô nhỏ.
- Arduino Uno R3 được lập trình bằng ngôn ngữ lập trình C/C++, là một ngôn ngữ lập trình phổ biến, dễ học và có nhiều tài liệu hỗ trợ.
- Board có thể hoạt động được trên nhiều hệ điều hành, bao gồm Windows, macOS và Linux.
- Arduino Uno R3 hỗ trợ rất nhiều thư viện phần mềm cho các chức năng khác nhau.



Hình 2.2. Các thành phần cơ bản của Arduino Uno R3

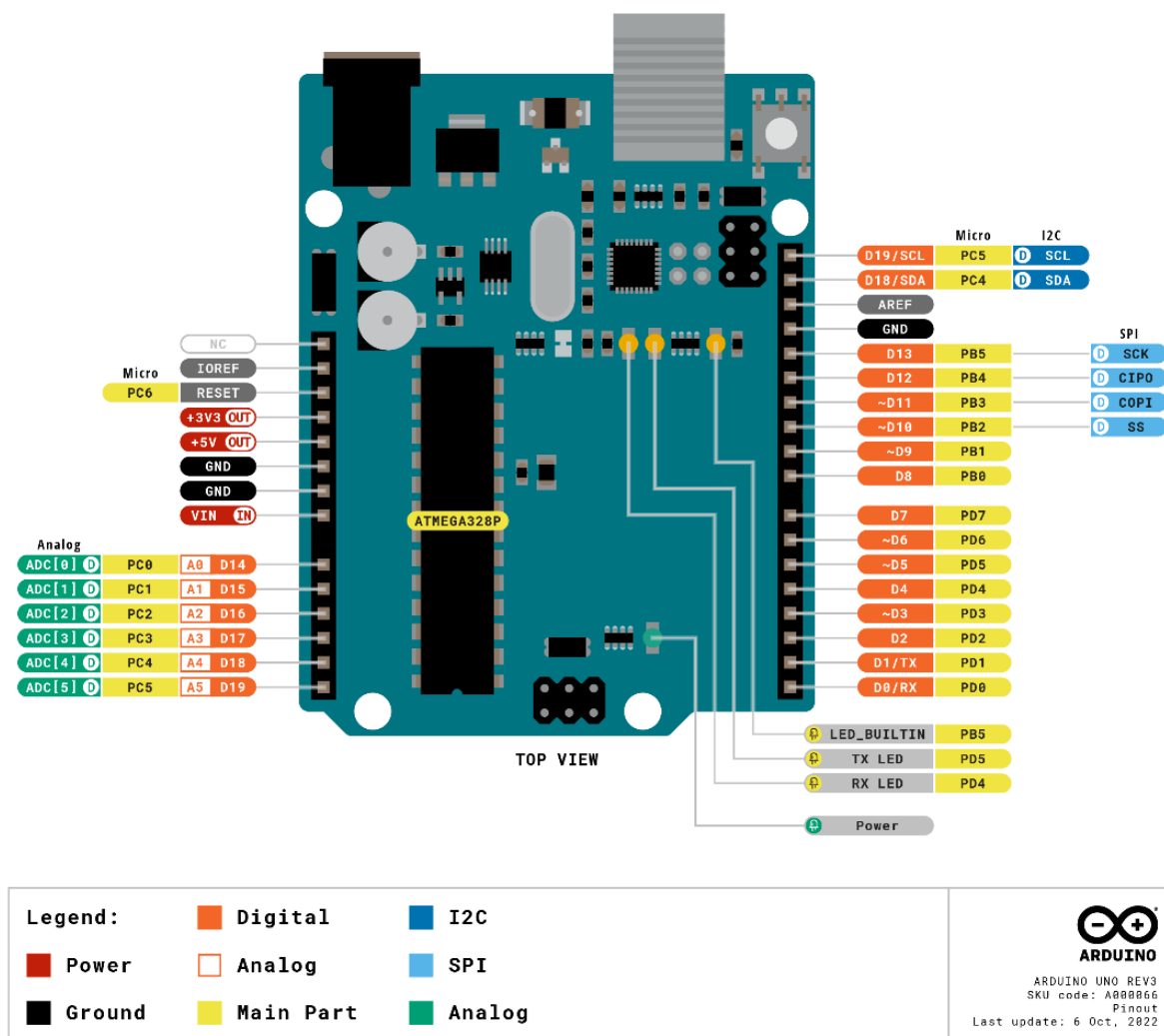
Arduino UNO có 14 chân digital dùng để đọc hoặc xuất tín hiệu. Chúng chỉ có 2 mức điện áp là 0V và 5V với dòng vào/ra tối đa trên mỗi chân là 40mA. Ở mỗi chân đều có các điện trở pull-up từ được cài đặt ngay trong vi điều khiển ATmega328 (mặc định thì các điện trở này không được kết nối).

Một số chân digital có các chức năng đặc biệt như sau:

- 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 → 255 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.

Arduino UNO có 6 chân analog (A0 → A5) cung cấp độ phân giải tín hiệu 10bit (0 → 1023) để đọc giá trị điện áp trong khoảng 0V → 5V. Với chân AREF trên board, bạn có thể để đưa vào điện áp tham chiếu khi sử dụng các chân analog. Tức là nếu bạn cấp điện áp 2.5V vào chân này thì bạn có thể dùng các chân analog để đo điện áp trong khoảng từ 0V → 2.5V với độ phân giải vẫn là 10bit.

Đặc biệt, Arduino UNO có 2 chân A4 (SDA) và A5 (SCL) hỗ trợ giao tiếp I2C/TWI với các thiết bị khác.



Hình 2.3. Sơ đồ chân của Arduino Uno R3

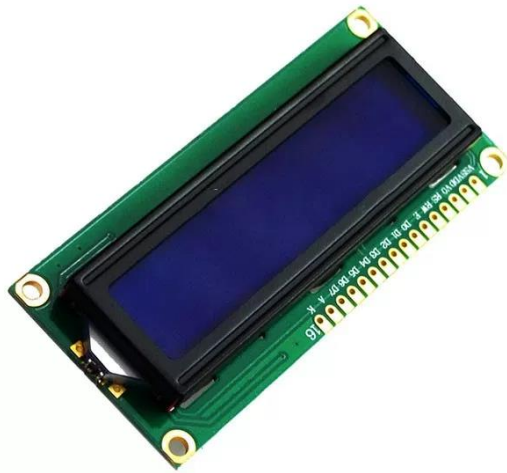
- Thông số kỹ thuật:
 - Vi điều khiển: ATmega328P(8bits)
 - Điện áp hoạt động: 5V
 - Tần số hoạt động: 16 MHz
 - Điện áp đầu vào khuyên dùng: 7VDC - 12VDC
 - Điện áp vào giới hạn: 6-20V DC
 - Số chân Digital I/O 14 (6 chân hardware PWM)
 - Số chân Analog 6 (độ phân giải 10bit)
 - Dòng tối đa trên mỗi 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)
- Clock Speed : 16 MHz

2.1.2. Màn hình LCD 1602

- Giới thiệu:

Ngày nay, thiết bị hiển thị LCD 1602 (Liquid Crystal Display) được sử dụng trong rất nhiều các ứng dụng của VĐK. LCD 1602 có rất nhiều ưu điểm so với các dạng hiển thị khác như: khả năng hiển thị kí tự đa dạng (chữ, số, kí tự đồ họa); dễ dàng đưa vào mạch ứng dụng theo nhiều giao thức giao tiếp khác nhau, tiêu tốn rất ít tài nguyên hệ thống, giá thành rẻ, ...



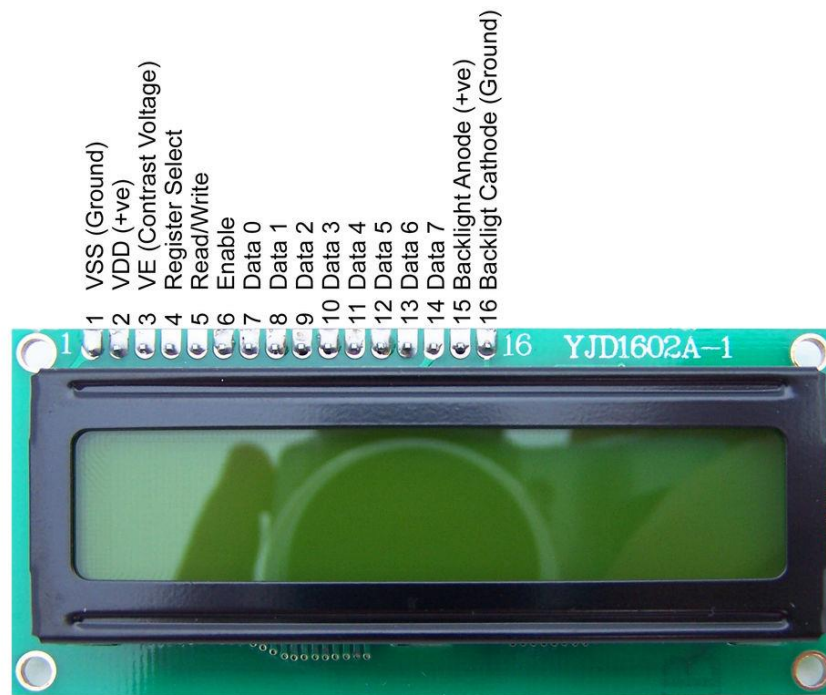
Hình 2.4. Màn hình LCD 1602

LCD 16×2 có 16 chân trong đó 8 chân dữ liệu (D0 – D7) và 3 chân điều khiển (RS, RW, EN).

5 chân còn lại dùng để cấp nguồn và đèn nền cho LCD 16×2.

Các chân điều khiển giúp ta dễ dàng cấu hình LCD ở chế độ lệnh hoặc chế độ dữ liệu.

Chúng còn giúp ta cấu hình ở chế độ đọc hoặc ghi.



Hình 2.5. Sơ đồ chân LCD 1602

- Thông số kỹ thuật:
 - Điện áp MAX: 7V
 - Điện áp MIN: - 0,3V
 - Hoạt động ổn định: 2.7-5.5V
 - Điện áp ra mức cao: > 2.4
 - Điện áp ra mức thấp: <0.4V
 - Dòng điện cấp nguồn: 350uA - 600uA
 - Nhiệt độ hoạt động: - 30 - 75 độ C

2.1.3. Module I2C Arduino giao tiếp LCD 1602

- Giới thiệu:



Hình 2.7. Module I2C Arduino giao tiếp LCD

Cảm biến siêu âm chống nước SR04 là một loại cảm biến siêu âm được thiết kế để đo khoảng cách và đo lường độ sâu của chất lỏng. Với thiết kế chống nước, cảm biến SR04 có thể hoạt động tốt trong môi trường ẩm ướt, làm cho nó trở thành một lựa chọn phổ biến cho các ứng dụng trong ngành công nghiệp và dân dụng.

Cảm biến SR04 sử dụng sóng siêu âm để đo khoảng cách, với tần số hoạt động lên đến 40 kHz. Khi sóng siêu âm được phát ra từ cảm biến, nó sẽ phản chiếu lại từ bề mặt của vật thể và trở lại cảm biến. Thời gian trễ giữa việc phát sóng và thu sóng sẽ được sử dụng để tính toán khoảng cách.

Lưu ý cảm biến siêu âm có sóng phát ra theo hình nón nên ở khoảng cách xa hoặc bề mặt vật thể không bằng phẳng sẽ dễ gây nhiễu cảm biến dẫn đến khoảng cách bị sai lệch.

- Tính năng:

- Kích thước nhỏ, dễ dàng sử dụng.
- Công suất thấp.
- Độ chính xác cao.
- Chống nhiễu tốt.
- Chống thấm.

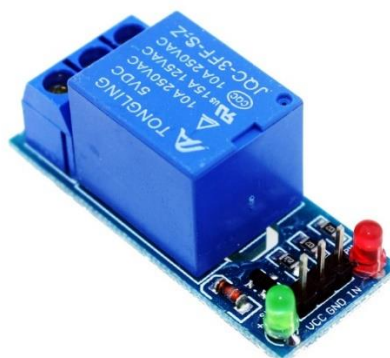
- Đặc điểm kỹ thuật:

- Điện áp hoạt động: 3-5.5VDC.
- Công suất: <40mW
- Tần số cảm biến: 40KHz
- Khoảng cách: 20cm-600cm.
- Độ chính xác: <2mm.
- Góc đo: 75 độ.

- Kích thước: 42x29x12mm (LWH).
- Nhiệt độ làm việc -20°C - 70°C.

2.1.5. Module Relay 5V 1 kênh (Low or High)

- Giới thiệu:



Hình 2.8. Module Relay 5V 1 kênh

- Module Relay 5V 1 Kênh được dùng như một công tắc điện , dùng để điều khiển các thiết bị công suất lớn (đèn, động cơ, ...)
- Module Relay 5V 1 Kênh gồm 1 rơ le hoạt động tại điện áp 5VDC, 12VDC chịu được hiệu điện thế lên đến 250VAC 10A. Module relay 1 kênh được thiết kế chắc chắn, khả năng cách điện tốt.
- Relay có 2 mức logic: kích mức thấp (L) và kích mức cao (H).

- Thông số kỹ thuật:

Thông số	Giá trị
Điện áp tải tối đa	AC 250V-10A / DC 30V-10A
Điện áp điều khiển	5 VDC
Dòng kích Relay	5mA
Trạng thái kích	Mức thấp (0V)
Đường kính Lỗ ốc	3.1 mm
Kích thước	50 * 26 * 18.5 mm

Hình 2.9. Thông số kỹ thuật Module Relay 5V 1 kênh

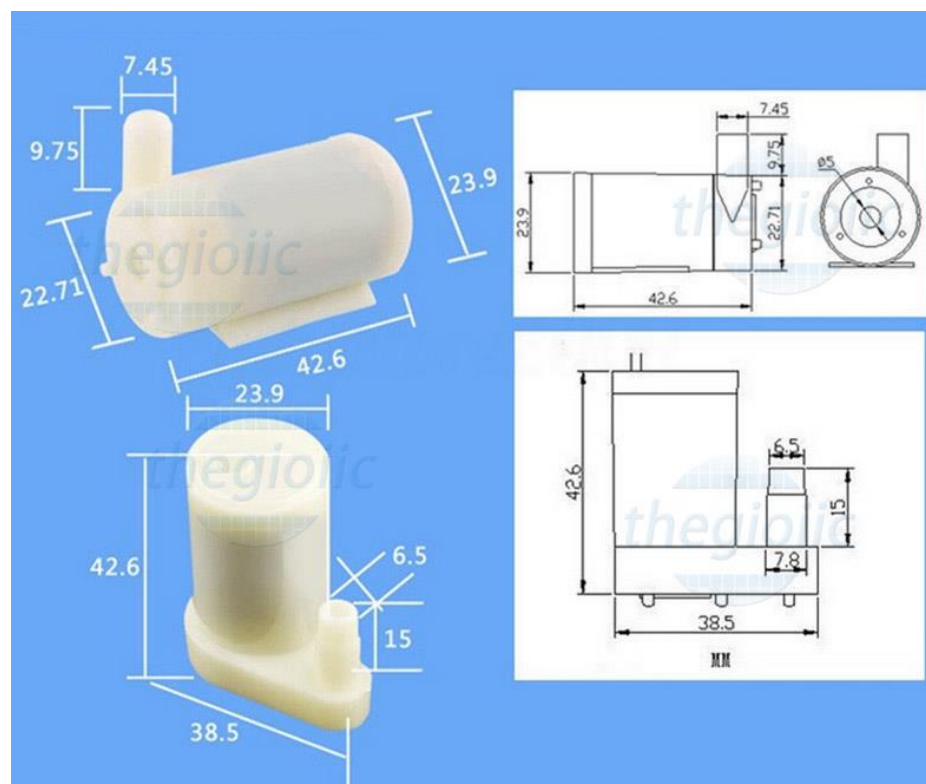
2.1.6. Máy bơm chìm mini 5V

Máy bơm chìm mini với điện áp sử dụng trong khoảng 3 ~ 5VDC, dùng bơm nước với độ xa lên đến 0,8 -> 1.5m.(ống càng nhỏ bơm càng xa). Nên cho bơm hoạt động ở điện áp 3 - 5 Vôn để bền động cơ.



Hình 2.10. Máy bơm mini 3-5V

Máy bơm mini là máy bơm được sử dụng phổ biến trong các ứng dụng để bơm nước, dung dịch trong các thiết kế nhỏ, mô hình tưới cây, hồ cá,...



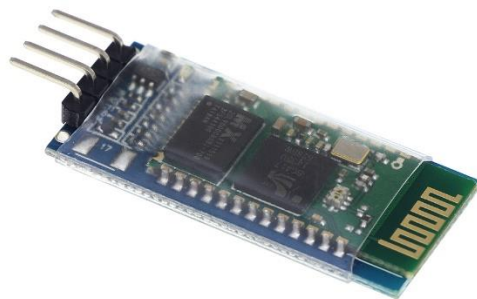
Hình 2.11. Thông số kích thước máy bơm mini 3-5V

- Thông số kỹ thuật:

- Nguồn: 3-5VDC
- Dòng điện: 100-200mA
- Lưu lượng bơm: 1.2-1.6L/phút,
- Kích thước: 42.6 x 23.9mm

2.1.7. Module Bluetooth HC-06

Với thiết kế nhỏ gọn, tiện lợi, giao tiếp với vi điều khiển chỉ bằng 2 chân (Tx và Rx), module bluetooth HC06 sẽ giúp bạn thực hiện các dự án truyền dẫn và điều khiển từ xa một cách dễ dàng.



Hình 2.12. Module Bluetooth HC-06

Điểm khác biệt so với HC05 đó là HC06 chỉ có thể chạy được 1 chế độ Slave (khác với HC05 có thể hoạt động với chế độ Master hoặc Slave). Điều này có nghĩa là bạn không thể chủ động kết nối từ vi điều khiển đến các thiết bị ngoại vi. Mà cách kết nối là: bạn phải sử dụng thiết bị ngoại vi (điện thoại thông minh, máy tính laptop) để dò tín hiệu kết nối Bluetooth mà HC06 phát ra. Sau khi pair thành công bạn có thể gửi tín hiệu từ vi điều khiển đến các thiết bị ngoại vi này, và ngược lại.

- Thông số kỹ thuật:

- Điện áp hoạt động: 3V3-5V DC
- Dòng điện tiêu thụ: 20-30mA
- Nhiệt độ hoạt động: -20~75°C
- Sử dụng chip: CSR Bluetooth V2.0
- Cấu hình Slave mặc định, không thay đổi được.
- Hỗ trợ tốc độ baud: 200, 2400, 4800, 9600, 19200, 38400, 57600, 115200
- Kích thước: 28x15x2,35mm

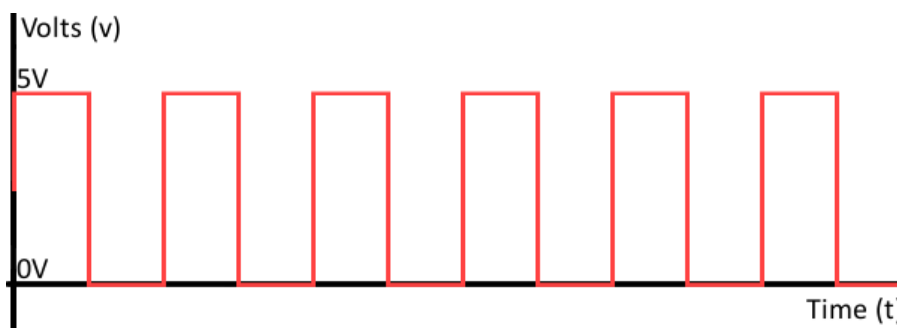
- Giao tiếp: UART (TX,RX)
- Tốc độ:
- Bất đồng bộ: 2.1Mbps(Max)/160kbps
- Đồng bộ: 1Mbps/1Mbps
- Bảo mật: mã hóa và chứng thực
- Cấu hình mặc định:
- Tốc độ baud 9600, N, 8, 1
- Mật khẩu: 1234

2.2. Chuẩn truyền dữ liệu và chuẩn kết nối.

2.2.1. Chuẩn truyền dữ liệu Digital

Chuẩn truyền dữ liệu Digital (còn được gọi là chuẩn giao tiếp số hoặc chuẩn digital communication) là phương thức truyền dữ liệu theo hình thức tín hiệu số (0 hoặc 1), thay vì tín hiệu tương tự (analog). Các chuẩn truyền dữ liệu Digital phổ biến nhất là UART, I2C, SPI, OneWire, CAN, Ethernet, USB,...

Các chuẩn truyền dữ liệu Digital được sử dụng phổ biến trong các ứng dụng IoT (Internet of Things) và các hệ thống nhúng (embedded systems), trong đó Arduino Uno R3 là một ví dụ điển hình.



Hình 2.13. Tín hiệu Digital

Trong mô hình hệ thống bơm xả nước tự động cho hồ cá của nhóm, chuẩn truyền dữ liệu Digital được sử dụng để truyền tín hiệu số giữa các linh kiện trong hệ thống, bao gồm Arduino Uno R3, cảm biến siêu âm SR04, máy bơm mini 5V và relay 5V. Cụ thể, Arduino Uno R3 sử dụng các chân Digital để kết nối với cảm biến siêu âm SR04 và relay 5V, trong đó 1 chân Digital được sử dụng để đọc tín hiệu đo từ cảm biến siêu âm SR04, và chân Digital khác được sử dụng để điều khiển relay 5V và bật/tắt máy bơm mini 5V.

Chuẩn truyền dữ liệu Digital có nhiều ưu điểm như tốc độ truyền dữ liệu nhanh, khả năng xử lý lỗi tốt, độ tin cậy cao, chi phí thấp và dễ sử dụng. Tuy nhiên, nó cũng có một

số hạn chế như cần nhiều dây kết nối hơn so với truyền dữ liệu tương tự (analog), khó thực hiện truyền dữ liệu ở khoảng cách xa và không phù hợp cho các ứng dụng yêu cầu độ chính xác và độ phân giải cao.

2.2.2. Chuẩn kết nối I2C

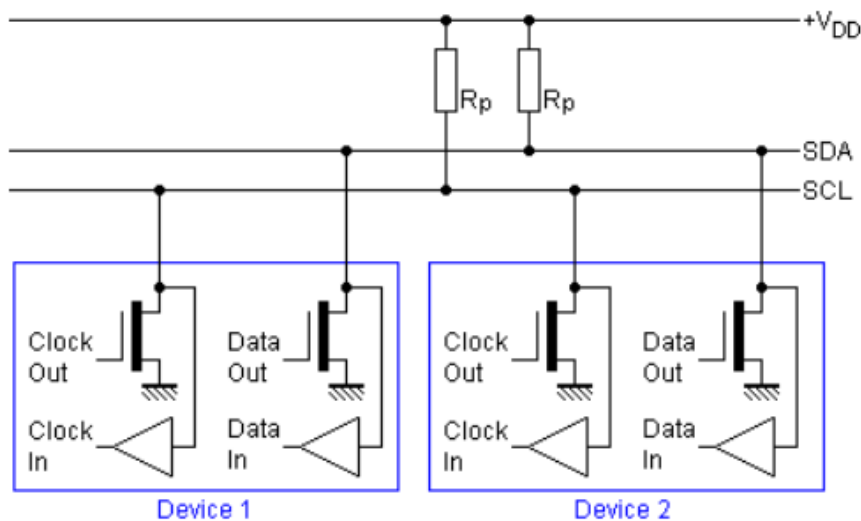
I2C, viết tắt của Inter-Integrated Circuit, là một loại bus nối tiếp được phát triển bởi hãng sản xuất linh kiện điện tử Philips. Ban đầu, loại bus này chỉ được dùng trong các linh kiện điện tử của Philips. Sau đó, do tính ưu việt và đơn giản của nó, I2C đã được chuẩn hóa và được dùng rộng rãi trong 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 2.14. 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 bởi 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

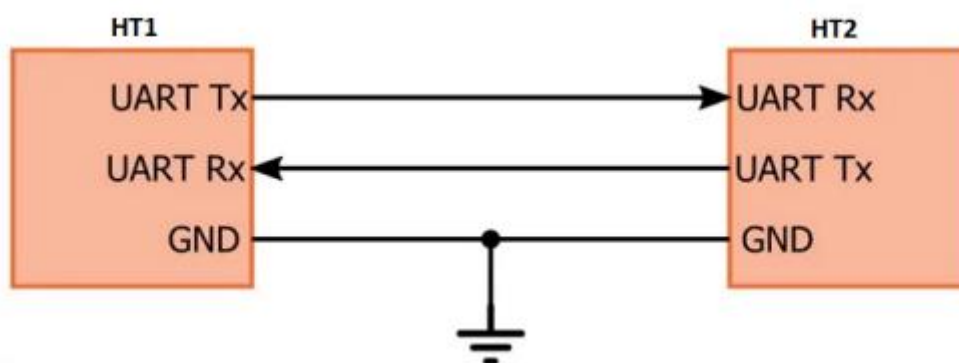
động như là thiết bị nhận hoặc truyền dữ liệu hay có thể vừa truyền vừa nhận. Hoạt động truyền hay nhận còn tùy thuộc vào việc thiết bị đó là chủ (master) hay tớ (slave).

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.

2.2.3. 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 2.15. Hệ thống truyền dữ liệu bất đồng bộ

Với Bluetooth HC-06, chuẩn truyền dữ liệu UART được sử dụng để truyền dữ liệu giữa module Bluetooth HC-06 và các thiết bị khác như vi điều khiển, máy tính, hoặc các module Bluetooth khác.

Bluetooth HC-06 hỗ trợ chuẩn truyền UART với tốc độ truyền dữ liệu tối đa là 9600 baud, tương đương với 9600 bit truyền dữ liệu mỗi giây. Tốc độ truyền dữ liệu có thể được

cấu hình bằng cách sử dụng các lệnh AT (Attention) được gửi đến module Bluetooth HC-06 thông qua giao diện UART.

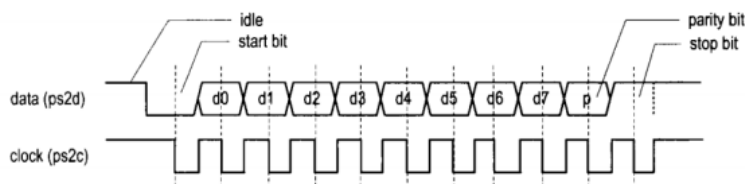
Chuẩn truyền UART sử dụng các tín hiệu logic 0 và 1 để truyền dữ liệu. Các tín hiệu này được gửi qua hai chân TX (Transmit) và RX (Receive) của module Bluetooth HC-06. Chân TX được sử dụng để truyền dữ liệu từ module Bluetooth HC-06 đến thiết bị khác, trong khi chân RX được sử dụng để nhận dữ liệu từ thiết bị khác đến module Bluetooth HC-06.

Để truyền và nhận dữ liệu qua giao diện UART, người dùng có thể sử dụng các lệnh AT để cấu hình các thông số như tốc độ truyền dữ liệu, chế độ kết nối, mã hóa dữ liệu, vv. Sau khi cấu hình, người dùng có thể sử dụng các hàm trong các ngôn ngữ lập trình như C/C++, Python, Java, vv. để truyền và nhận dữ liệu giữa module Bluetooth HC-06 và các thiết bị khác.

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 2.16. 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.

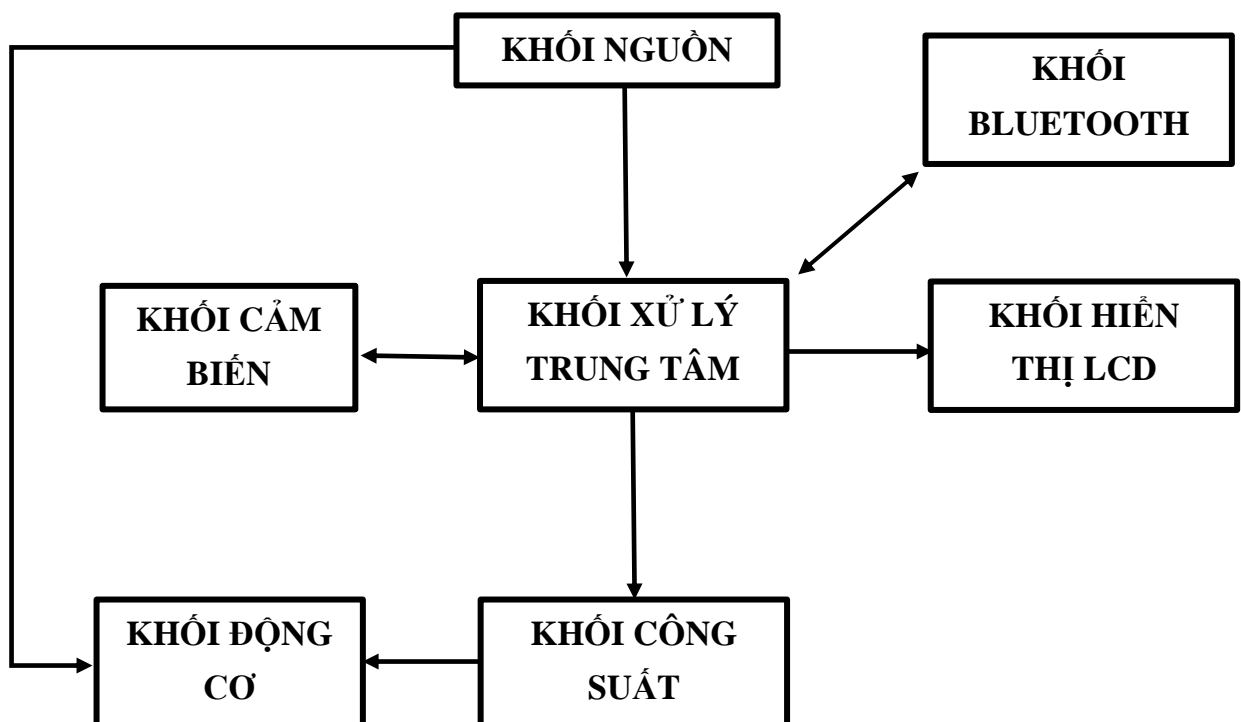
Stop bits: Là một hoặc các bit báo cho thiết bị nhận rằng một gói dữ liệu đã được gửi xong. Sau khi nhận được stop bits, thiết bị nhận sẽ tiến hành kiểm tra khung truyền để đảm bảo tính chính xác của dữ liệu. Stop bits là các bit bắt buộc xuất hiện trong khung truyền.

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

CHƯƠNG 3: THIẾT KẾ KIẾN TRÚC HỆ THỐNG

3.1. Sơ đồ khối và chức năng mỗi khối

Hệ thống bao gồm 6 khối chính sau đây:



Hình 3.1. Sơ đồ khối hệ thống

Khối xử lý trung tâm: Cụ thể trong hệ thống như mô hình hệ thống bơm xả nước tự động cho hồ cá, CPU (ở đây là Arduino Uno R3) sẽ được sử dụng để xử lý các tín hiệu đầu vào từ cảm biến siêu âm SR04, điều khiển máy bơm mini thông qua relay, và hiển thị kết quả đo lên màn hình LCD. CPU sẽ nhận các tín hiệu đầu vào, xử lý và tính toán để đưa ra quyết định điều khiển máy bơm, và sau đó hiển thị kết quả lên màn hình để người dùng có thể quan sát.

Khối cảm biến: Bao gồm cảm biến siêu âm có tính chính xác cao, dùng để thu thập dữ liệu từ đó đưa tín hiệu về khối xử lý trung tâm, rồi sau đó khối xử lý trung tâm sẽ xử lý. Khối cảm biến trong hệ thống bơm xả nước tự động cho hồ cá có nhiệm vụ đo lường và cung cấp thông tin về mức độ nước trong hồ cá cho hệ thống điều khiển.

Khối công suất: kết hợp với relay được sử dụng trong các ứng dụng điện tử để chuyển đổi tín hiệu điều khiển từ các bộ vi xử lý hoặc các thiết bị đầu vào khác sang các tín hiệu điện áp cao hơn để điều khiển các thiết bị công suất lớn hơn, khối công suất kết hợp với relay được sử dụng để điều khiển máy bơm mini.

Máy bơm mini là một thiết bị công suất, sử dụng điện áp cao hơn để hoạt động, vì vậy không thể điều khiển trực tiếp từ các thiết bị đầu vào như Arduino Uno R3. Thay vào đó, khối công suất kết hợp với relay được sử dụng để tách galvanic giữa nguồn điện của hệ thống và nguồn điện của máy bơm, giúp bảo vệ các thiết bị nhạy cảm và đảm bảo an toàn cho người sử dụng. Khi CPU (Arduino Uno R3) cảm nhận được các điều kiện để bơm nước, tín hiệu điều khiển sẽ được truyền đến relay, và relay sẽ chuyển đổi tín hiệu điều khiển sang điện áp cao hơn để bật máy bơm mini.

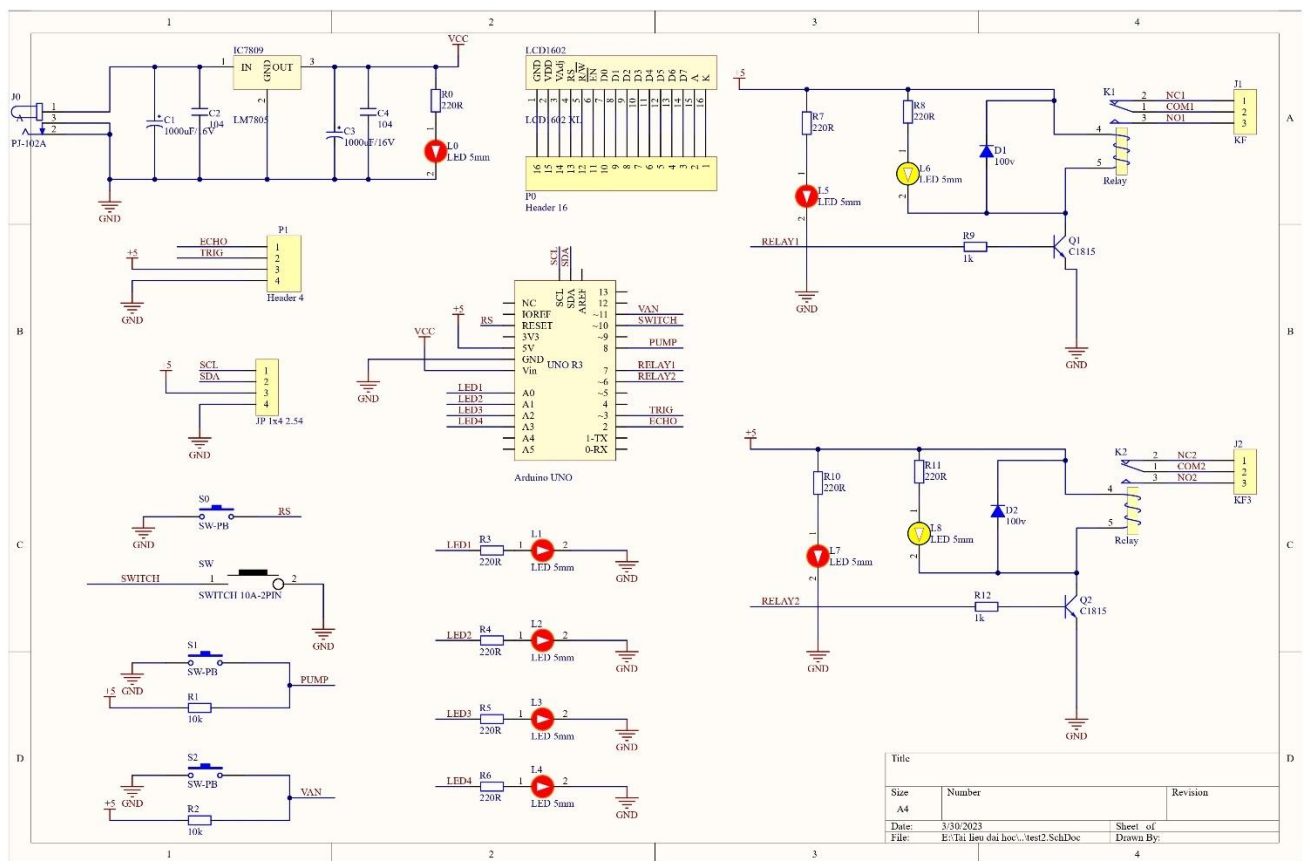
Khối hiển thị LCD: trong mô hình hệ thống bơm xả nước tự động cho hồ cá được sử dụng để hiển thị các thông tin về trạng thái của hệ thống và các thông số liên quan đến việc điều khiển bơm nước.

Khối động cơ: máy bơm trong mô hình này là máy bơm mini 5V, sử dụng để bơm nước trong hồ cá. Khi tín hiệu điều khiển được truyền từ khối xử lý trung tâm (Arduino Uno R3) đến khối công suất kết hợp với relay, relay sẽ kích hoạt khối động cơ máy bơm để bơm nước.

Khối nguồn: Khối nguồn trong mô hình hệ thống bơm xả nước tự động cho hồ cá có chức năng cung cấp điện cho toàn bộ hệ thống, bao gồm các linh kiện và khối động cơ máy bơm.

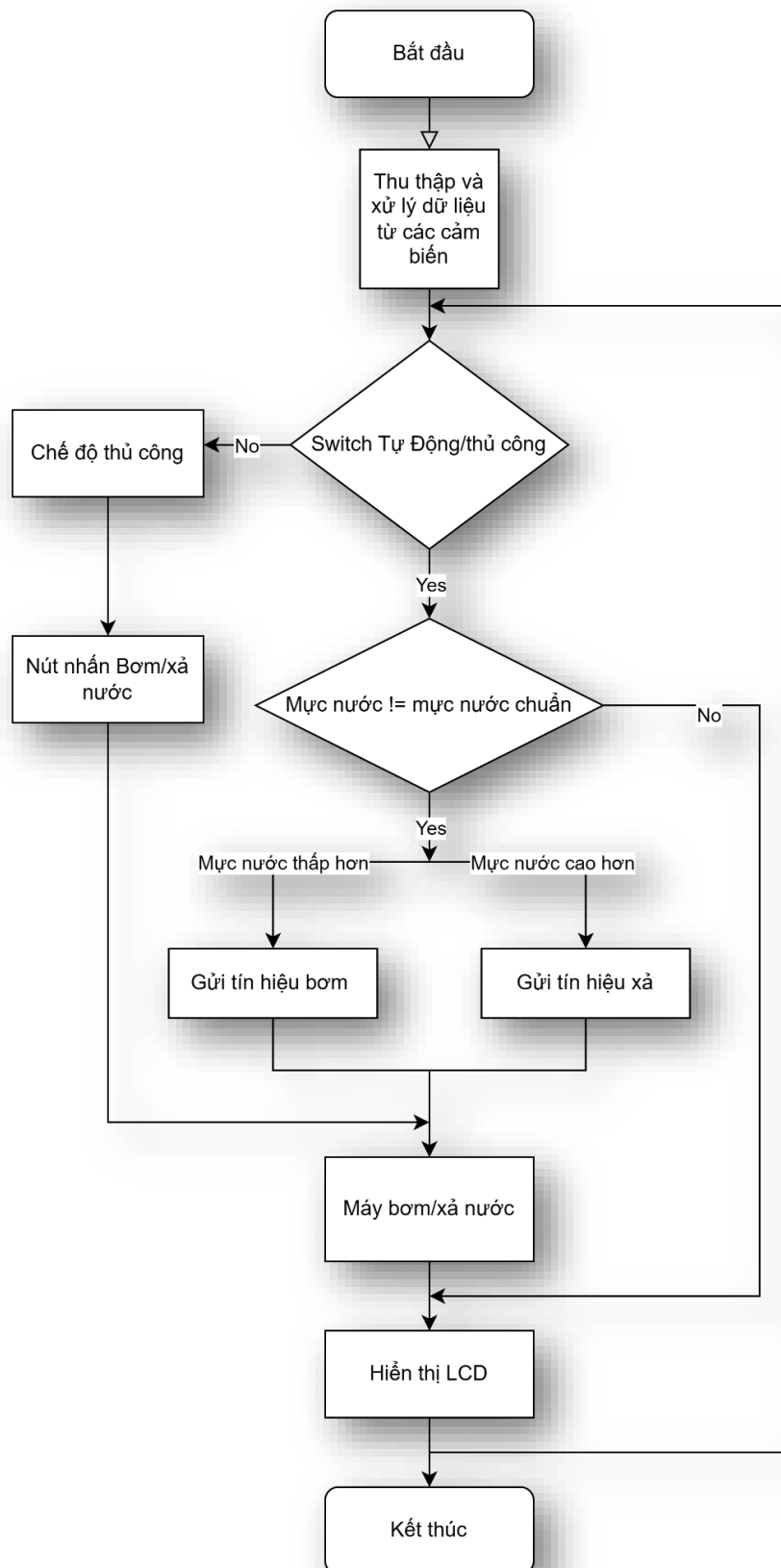
Khối Bluetooth: Khối Bluetooth HC-06 được sử dụng để thiết lập kết nối không dây giữa Arduino Uno R3 và ứng dụng trên điện thoại thông qua Bluetooth. Chức năng chính của khối Bluetooth là cung cấp kết nối không dây giữa hai thiết bị và truyền dữ liệu giữa chúng. Với chức năng này, khối Bluetooth HC-06 sẽ giúp cho hệ thống bơm xả nước tự động cho hồ cá trở nên linh hoạt hơn và dễ dàng trong việc điều khiển và theo dõi qua ứng dụng trên điện thoại.

3.2. Sơ đồ nguyên lý của hệ thống



Hình 3.2. Sơ đồ nguyên lý hệ thống

3.3. Lưu đồ thuật toán của chương trình chính



Hình 3.3. Lưu đồ thuật toán hệ thống

3.4. Thiết kế phần mềm

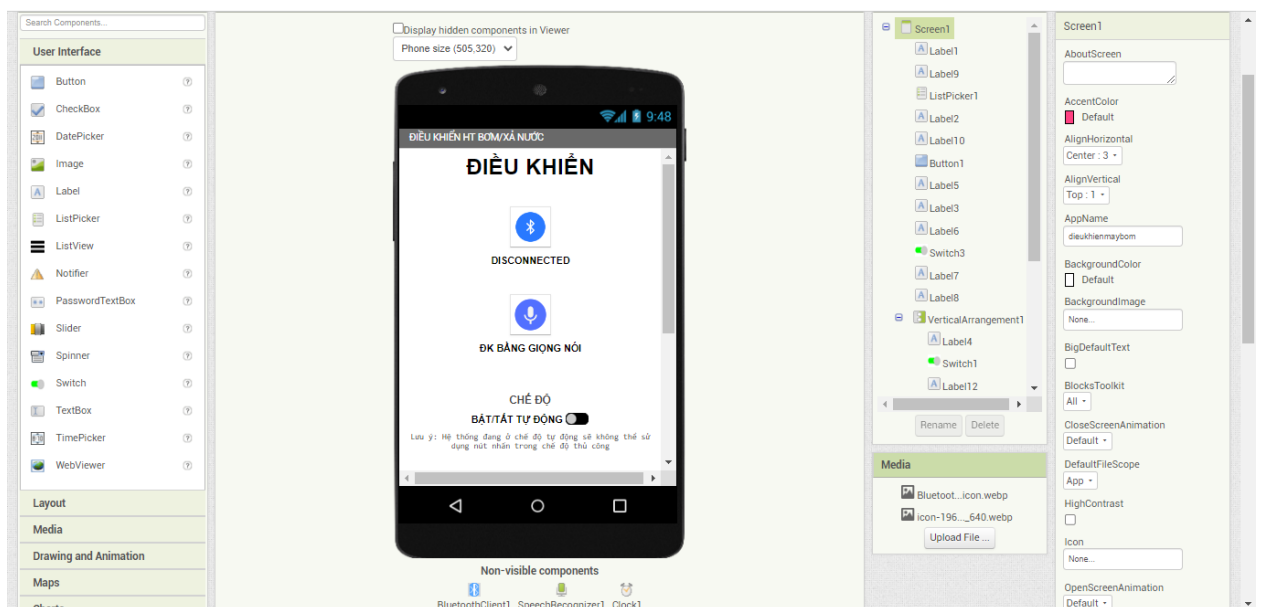
3.4.1. Phần mềm lập trình Arduino

(Code chương trình ở phần phụ lục)

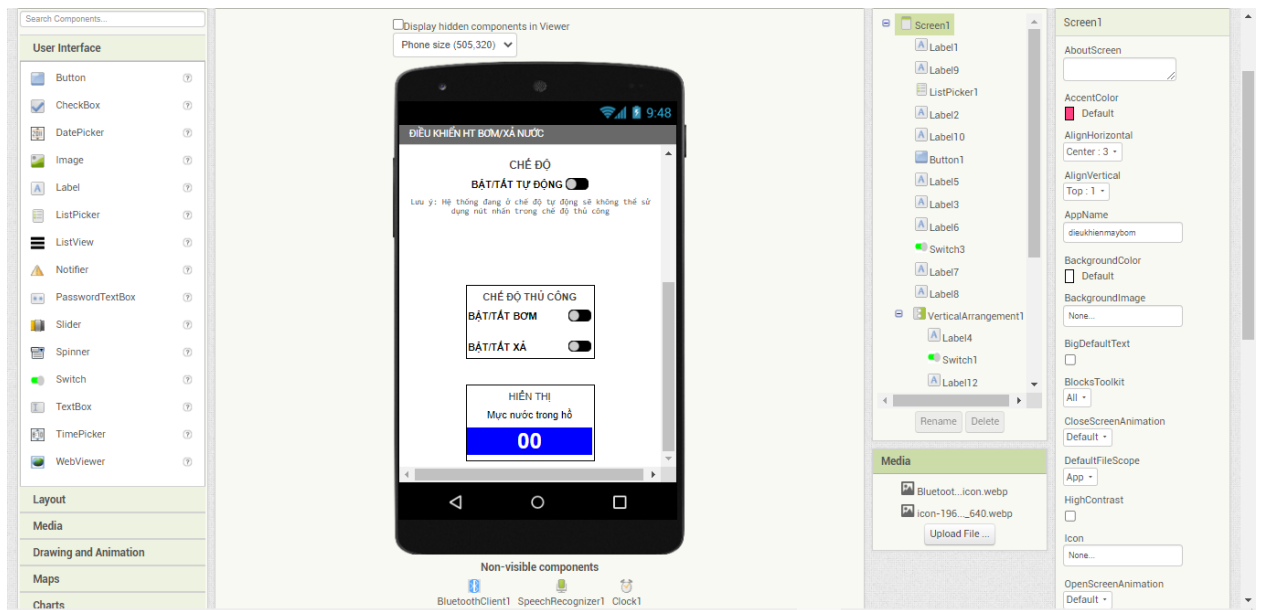
3.4.2. Phần mềm lập trình ứng dụng điện thoại

Thiết kế ứng dụng điện thoại điều khiển hệ thống qua Bluetooth gồm các chức năng:

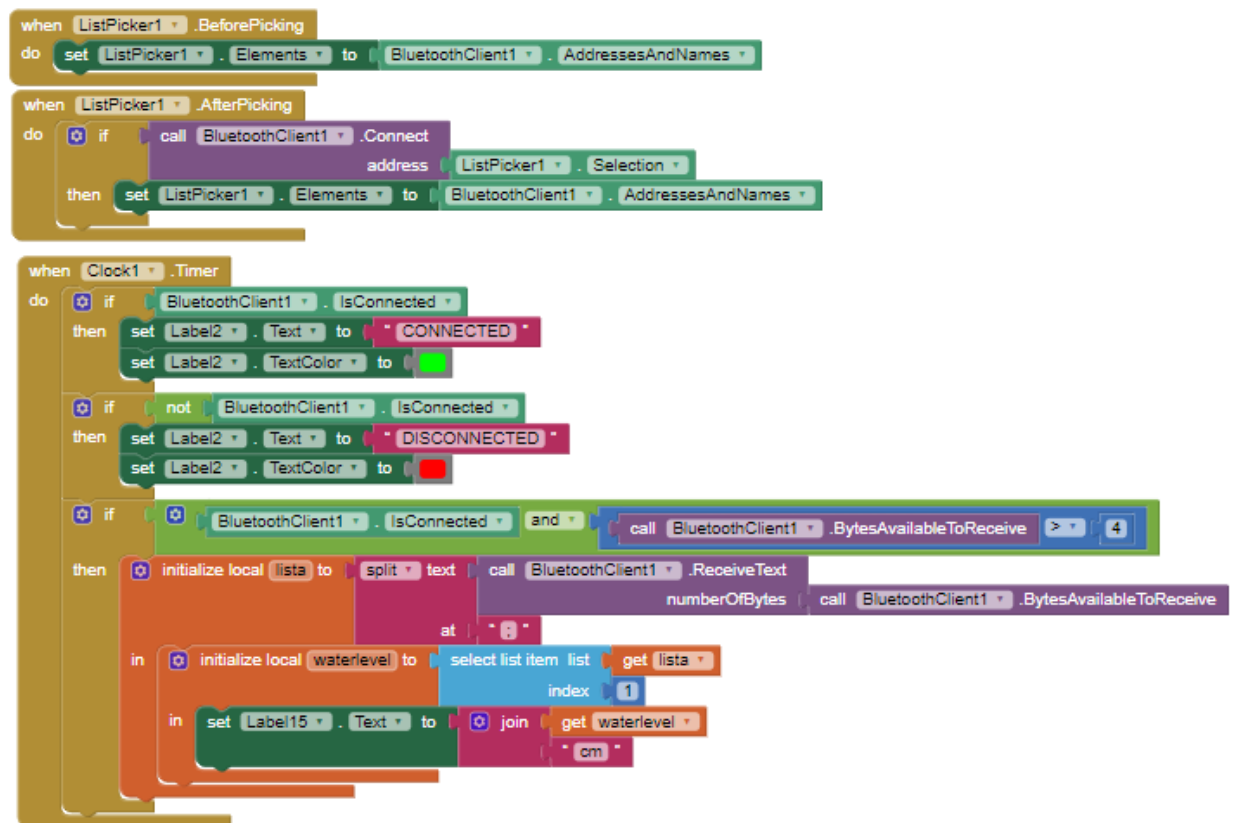
- Kết nối Bluetooth, hiển thị trạng thái khi đã connect và chưa connect (disconnect)
- Điều khiển bằng giọng nói
- Chuyển đổi chế độ: tự động và thủ công
- Chế độ thủ công, bơm và xả nước
- Hiển thị mực nước trong hồ



Hình 3.4. Thiết kế giao diện ứng dụng điện thoại trên MIT app inventor (1)



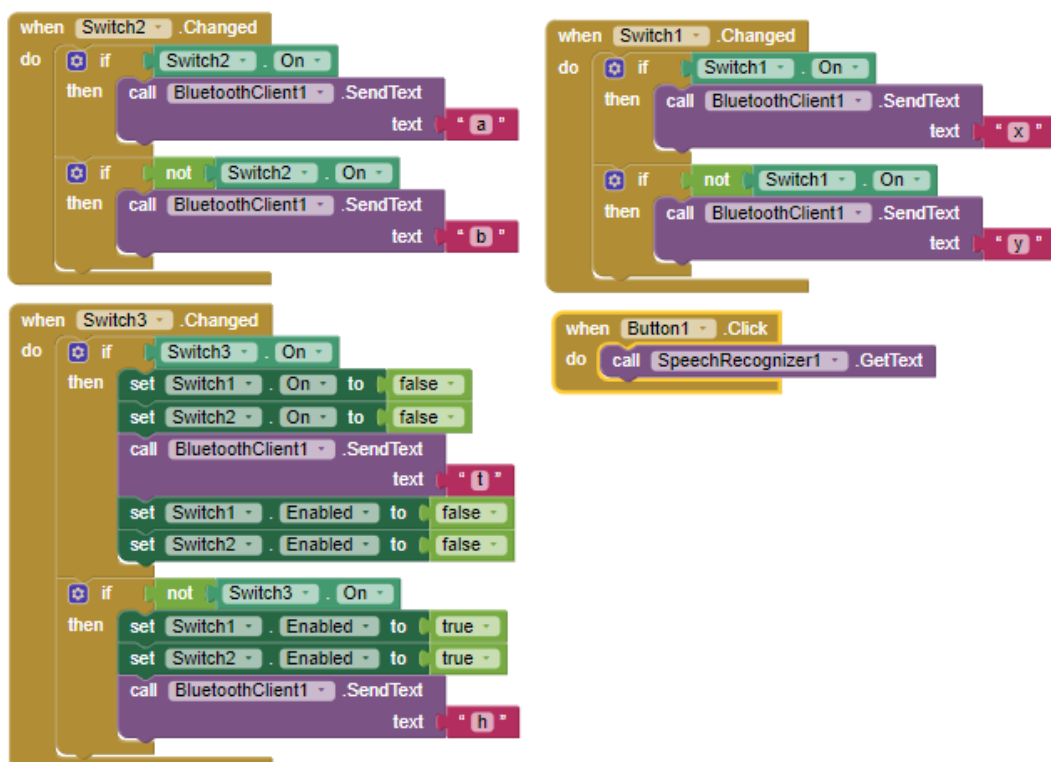
Hình 3.5. Thiết kế giao diện ứng dụng điện thoại trên MIT app inventor (2)



Hình 3.6. Các khối code của ứng dụng (1)



Hình 3.7. Các khối code của ứng dụng (2)



Hình 3.8. Các khối code của ứng dụng (3)

CHƯƠNG 4: KẾT QUẢ VÀ ĐÁNH GIÁ

4.1. Kết quả mô hình thi công



4.2. Đánh giá

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1. Kết luận.

5.2. Hướng phát triển.

TÀI LIỆU THAM KHẢO

[Arduino UNO R3 \(iotmaker.vn\)](https://iotmaker.vn)

[Arduino UNO R3 là gì? | Cộng đồng Arduino Việt Nam](#)

[Tìm hiểu thông số kỹ thuật của LCD 1602 \(suachualaptop24h.com\)](https://suachualaptop24h.com)

PHỤ LỤC

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <SoftwareSerial.h>

//khai báo chân
SoftwareSerial bluetooth(4, 5); //Tx hc06 nối rx-4; Rx hc06 nối tx-5
#define ECHO_PIN 2
#define TRIGGER_PIN 3
#define MOTOR_PIN1 6 //relay
#define MANUAL_MODE_PIN 10
#define MOTOR_PIN2 7 //relay
int BUTTON_PIN = 8;
int BUTTON_PIN1 = 11;
bool sta = 0;

//khai báo biến

char kytu;
String chuoi;
int n=0;

int buttonStatepump = 0;
int buttonState = 0; // initial state of the button

int temp = 0;
LiquidCrystal_I2C lcd(0x27, 20, 4);

//thiết lập ngõ vào và ngõ ra
void setup() {
  pinMode(ECHO_PIN, INPUT);
  pinMode(TRIGGER_PIN, OUTPUT);
  pinMode(MANUAL_MODE_PIN, INPUT_PULLUP);
  pinMode(BUTTON_PIN, INPUT);
  pinMode(BUTTON_PIN1, INPUT);
  // pinMode(PUMP_MODE_PIN, INPUT);
  pinMode(MOTOR_PIN1, OUTPUT);
  pinMode(MOTOR_PIN2, OUTPUT);

  Serial.begin(9600);
  bluetooth.begin(9600);
  lcd.init();
  lcd.backlight();
  lcd.setCursor(0, 0);
  lcd.print("Level:");
  //
  digitalWrite(TRIGGER_PIN, LOW);
```

```

    delayMicroseconds(2);
    digitalWrite(TRIGGER_PIN, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIGGER_PIN, LOW);

    attachInterrupt(digitalPinToInterrupt(MANUAL_MODE_PIN), nutbom, RISING);
}

// đo mực nước
int getWaterLevel() {
    long duration;
    digitalWrite(TRIGGER_PIN, LOW);
    delayMicroseconds(2);
    digitalWrite(TRIGGER_PIN, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIGGER_PIN, LOW);
    duration = pulseIn(ECHO_PIN, HIGH);
    return (duration / 2) / 29.1;
}

void buttonstatepump()
{

}

// ham nut bom
void nutbom() {
    //bool sta = 0;
    if(digitalRead(BUTTON_PIN)==0)
    {
        delay(20);
        if(digitalRead(BUTTON_PIN)==0)
        {
            sta =!sta;
            digitalWrite(MOTOR_PIN1, sta);
        }

        while(digitalRead(BUTTON_PIN)==0);
    }
}

// ham nut bom
void nutxa() {
    //bool sta = 0;
    if(digitalRead(BUTTON_PIN1)==0)
    {
        delay(20);
        if(digitalRead(BUTTON_PIN1)==0)
        {
            temp =!temp;
            digitalWrite(MOTOR_PIN2, temp);
        }

        while(digitalRead(BUTTON_PIN1)==0);
    }
}

```



```

    }
}
//hàm auto
void automode(){
    int waterLevel = getWaterLevel();
    if (waterLevel < 30)
    {
        digitalWrite(MOTOR_PIN1, HIGH);

        digitalWrite(MOTOR_PIN2, LOW);

    }

    else if(waterLevel > 30 & waterLevel < 80 )
    {
        digitalWrite(MOTOR_PIN1, LOW);

        digitalWrite(MOTOR_PIN2, LOW);

    }
    else if(waterLevel > 80 )
    {

        digitalWrite(MOTOR_PIN2, HIGH);

        digitalWrite(MOTOR_PIN1, LOW);

    }

}

//hàm thủ công
void manualmode() {
    nutbom();
    nutxa();
}

//ham chính
void loop() {
    buttonState = digitalRead(MANUAL_MODE_PIN); // read the button state
    bool sta = 0;

    if (buttonState == 0) { // if the button is pressed
        //int waterLevel = getWaterLevel();
        automode();
    }
    else if (buttonState == 1) {

        //digitalWrite(MOTOR_PIN1, LOW);
        //digitalWrite(MOTOR_PIN2, LOW);
        manualmode();
    }
}

```

```

int waterLevel = getWaterLevel();
bluetooth.print(waterLevel);
bluetooth.print(";");
// Serial.println(waterLevel);

//nhận tín hiệu của bluetooth
if (bluetooth.available()) {
    kytu=bluetooth.read();
    chuoi=chuoi+kytu;
    Serial.write(kytu);

    if(chuoi.indexOf("t")>=0) {
        while(chuoi.indexOf("h")<0) {
            automode();
            kytu=bluetooth.read();
            chuoi=chuoi+kytu;
        }
        digitalWrite(MOTOR_PIN1,LOW);
        digitalWrite(MOTOR_PIN2,LOW);
        chuoi="";
    }
    if(chuoi.indexOf("x")>=0){
        digitalWrite(MOTOR_PIN1,HIGH);
        chuoi="";
    }
    if(chuoi.indexOf("y")>=0){
        digitalWrite(MOTOR_PIN1,LOW);
        chuoi="";
    }
    if(chuoi.indexOf("a")>=0){
        digitalWrite(MOTOR_PIN2,HIGH);
        chuoi="";
    }
    if(chuoi.indexOf("b")>=0){
        digitalWrite(MOTOR_PIN2,LOW);
        chuoi="";
    }
}

updateDisplay();

}

//CẬP NHẬT LCD1602
void updateDisplay() {

    lcd.setCursor(7,0); //Set cursor to first column of second row
    lcd.print("          "); //Print blanks to clear the row
    lcd.setCursor(7, 0);
    lcd.print(getWaterLevel());
    lcd.print("cm");
    delay(500);
    lcd.setCursor(0, 1);

```

```
if (buttonState == LOW ) {  
    lcd.print("          "); //Print blanks to clear the row  
    lcd.setCursor(0, 1);  
    lcd.print("Mode: Auto");  
  
}  
else if (buttonState == HIGH){  
    lcd.print("          "); //Print blanks to clear the row  
    lcd.setCursor(0, 1);  
    lcd.print("Mode: Manual");  
  
}  
}
```