# **LỜI NÓI ĐẦU**

Chúng ta thường dùng bể chứa nước và máy bơm để phục vụ nhu cầu hàng ngày. Tuy nhiên việc bật máy bơm để bơm nước lên bể chứa khi gần hết nước, rồi để ý thời gian và tắt máy bơm để tránh nước tràn ra ngoài bể là việc làm rất mất thời gian và thực sự phiền phức. Vì vậy, chiếc phao điện là một thiết bị cần thiết để có thể tự động hóa quá trình bật và tắt máy bơm cung cấp đầy đủ nước cho sinh hoạt.

Việc tắt và bật của máy bơm phải được tự động hoàn toàn và chúng ta không cần quan tâm đến việc máy bơm chạy hay ngắt nữa. Để cho máy bơm tự chạy và tự ngắt theo yêu cầu thì chúng ta cần lắp phao điện để cho máy bơm hoạt động tự động.

Trong quá trình học môn *Thiết kế hệ thống điện điện tử,* chúng em đã nhận được đề tài “Thiết kế hệ thống phao điện không dây sử dụng công nghệ RF”. Để hoàn thành đề tài này, ngoài sự nỗ lực của bản thân chúng em ,chúng em xin chân thành gửi lời cảm ơn sâu sắc tới các thầy cô Khoa Công Nghệ Tự Động Hóa trường Đại học Công nghệ thông tin và Truyền thông Thái Nguyên, đặc biệt là thầy **Th.S Đặng Văn Ngọc** đã tận tình hướng dẫn, giúp đỡ và tạo điều kiện tốt nhất cho chúng em kể từ khi nhận đề tài tới khi hoàn thành đề tài này.

Em xin chân thành cảm ơn!

# **MỤC LỤC**

[**LỜI NÓI ĐẦU** 1](#_Toc56668158)

[**MỤC LỤC** 2](#_Toc56668159)

[**DANH MỤC HÌNH ẢNH** 3](#_Toc56668160)

[**CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI** 4](#_Toc56668161)

[**1.1** **Giới thiệu chung** 4](#_Toc56668162)

[**1.2** **Tổng quan các công nghệ sử dụng** 4](#_Toc56668165)

[*1.2.1* *Sóng RF* 4](#_Toc56668166)

[*1.2.2* *Sóng siêu âm* 5](#_Toc56668167)

[**CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG** 8](#_Toc56668168)

[**2.1** **Phân tích và thiết kế** 8](#_Toc56668171)

[**2.2** **Đặc tả hệ thống** 8](#_Toc56668172)

[*2.2.1* *Sơ đồ khối của hệ thống và chức năng từng khối* 8](#_Toc56668173)

[*2.2.2* *Sơ đồ khối Slave* 9](#_Toc56668174)

[*2.2.3* *Lưu đồ thuật toán* 9](#_Toc56668175)

[*2.2.4* *Sơ đồ khối Master* 12](#_Toc56668176)

[**2.3** **Giới thiệu một số công cụ, thiết bị sử dụng** 13](#_Toc56668177)

[*2.3.1* *Atmega328p* 13](#_Toc56668178)

[*2.3.2* *Sơ đồ nguyên lý* 14](#_Toc56668179)

[*2.3.3* *Visual Studio Code và Platform IO* 16](#_Toc56668180)

[*2.3.4* *Altium Design* 18](#_Toc56668181)

[*2.3.5* *Module NRF24L01* 20](#_Toc56668182)

[*2.3.6* *LCD16x2* 21](#_Toc56668183)

[*2.3.7* *Module I2C* 23](#_Toc56668184)

[*2.3.8* *Mạch nạp code AVR* 23](#_Toc56668185)

[*2.3.9* *Cảm biến khoảng cách HC\_SR04* 25](#_Toc56668186)

[**CHƯƠNG 3: KẾT QUẢ ĐÁNH GIÁ** 27](#_Toc56668187)

[**3.1** **Các bước sử dụng hệ thống** 27](#_Toc56668189)

[**3.2** **Đánh giá hệ thống** 28](#_Toc56668190)

[*3.2.1* *Ưu điểm* 28](#_Toc56668191)

[*3.2.2* *Nhược điểm* 28](#_Toc56668192)

[**3.3** **Kết luận** 28](#_Toc56668193)

[**Tài liệu tham khảo** 29](#_Toc56668194)

[**Phụ lục** 30](#_Toc56668195)

# **DANH MỤC HÌNH ẢNH**

[Hình 1. 1 Ứng sụng sóng RF trong thẩm mỹ 6](#_Toc56668196)

[Hình 1. 2 Ứng dụng sóng RF trong vô tuyến 6](#_Toc56668197)

[Hình 1. 3 Máy siêu âm y khoa 8](#_Toc56668198)

[Hình 2. 1 Sơ đồ khối toàn hệ thống 10](#_Toc56668215)

[Hình 2. 2 Sơ đồ khối Slave 11](#_Toc56668216)

[Hình 2. 3 Sơ đồ thuật toán hệ thống 12](#_Toc56668217)

[Hình 2. 4 Lưu đồ thuật toán Master 13](#_Toc56668218)

[Hình 2. 5 Lưu đồ thuật toán Slave 14](#_Toc56668219)

[Hình 2. 6 Sơ đồ khối Master 15](#_Toc56668220)

[Hình 2. 9 Atmega328p 15](#_Toc56668221)

[Hình 2. 10 Sơ đồ chân Atmega328p 16](#_Toc56668222)

[Hình 2. 7 Sơ đồ nguyên lý mạch Master 17](#_Toc56668223)

[Hình 2. 8 Sơ đồ nguyên lý mạch Slave 17](#_Toc56668224)

[Hình 2. 11 Visual Studio Code 18](#_Toc56668225)

[Hình 2. 12 Platform IO 19](#_Toc56668226)

[Hình 2. 13 Tạo Project 19](#_Toc56668227)

[Hình 2. 14 Điền thông tin project 19](#_Toc56668228)

[Hình 2. 15 Cấu trúc project sau khi tạo xong 20](#_Toc56668229)

[Hình 2. 16 Altium Design 20](#_Toc56668230)

[Hình 2. 17 Module NRF24L01 22](#_Toc56668231)

[Hình 2. 18 Các chân giao tiếp module NRF2410L 22](#_Toc56668232)

[Hình 2. 19 Ghép nối với Arduino 23](#_Toc56668233)

[Hình 2. 20 LCD 16x2 24](#_Toc56668234)

[Hình 2. 21 Module I2C 25](#_Toc56668235)

[Hình 2. 22 Mạch nạp AVR 26](#_Toc56668236)

[Hình 2. 23 Sơ đồ chân mạch nạp AVR 27](#_Toc56668237)

[Hình 2. 24 Cảm biến khoảng cách HC\_SR04 28](#_Toc56668238)

# **CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI**

## **Giới thiệu chung**

Chúng ta thường dùng bể chứa nước và máy bơm để phục vụ nhu cầu hàng ngày. Tuy nhiên việc bật máy bơm để bơm nước lên bể chứa khi gần hết nước, rồi để ý thời gian và tắt máy bơm để tránh nước tràn ra ngoài bể là việc làm rất mất thời gian và thực sự phiền phức. Vì vậy, chiếc phao điện là một thiết bị cần thiết để có thể tự động hóa quá trình bật và tắt máy bơm cung cấp đầy đủ nước cho sinh hoạt.

Việc tắt và bật của máy bơm phải được tự động hoàn toàn và chúng ta không cần quan tâm đến việc máy bơm chạy hay ngắt nữa. Để cho máy bơm tự chạy và tự ngắt theo yêu cầu thì chúng ta cần lắp phao điện để cho máy bơm hoạt động tự động.



## **Tổng quan các công nghệ sử dụng**

### *Sóng RF*

*a, Khái niệm*

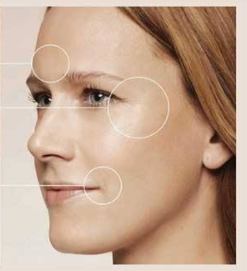
Sóng RF hay còn gọi là sóng siêu âm vô tuyến điện, viết tắt của từ Radio Frequency. Công nghệ RF trước hết được khai thác để sử dụng trong vô tuyến điện, truyền hình, phát thanh.

*b, Tính chất đặc biệt của dòng điện tần số vô tuyến*

Các [dòng điện](https://vi.wikipedia.org/wiki/D%C3%B2ng_%C4%91i%E1%BB%87n) dao động ở các tần số vô tuyến có các tính chất đặc biệt khác với [dòng một chiều](https://vi.wikipedia.org/wiki/M%E1%BB%99t_chi%E1%BB%81u_(%C4%91i%E1%BB%87n)) hay [dòng xoay chiều](https://vi.wikipedia.org/wiki/%C4%90i%E1%BB%87n_xoay_chi%E1%BB%81u) dao động ở tần số thấp. Năng lượng trong một dòng điện RF có thể lan truyền trong không gian như các [sóng điện từ](https://vi.wikipedia.org/wiki/B%E1%BB%A9c_x%E1%BA%A1_%C4%91i%E1%BB%87n_t%E1%BB%AB) ([sóng vô tuyến](https://vi.wikipedia.org/wiki/S%C3%B3ng_v%C3%B4_tuy%E1%BA%BFn)); đây là cơ sở của công nghệ vô tuyến. Dòng điện RF không chạy trong lòng dây dẫn mà phần lớn lại chạy trên bề mặt của dây dẫn; điều này được gọi là [hiệu ứng bề mặt](https://vi.wikipedia.org/wiki/Hi%E1%BB%87u_%E1%BB%A9ng_b%E1%BB%81_m%E1%BA%B7t). Vì lý do này, khi cơ thể con người tiếp xúc với các dòng điện RF công suất lớn có thể gây bỏng bề mặt da và còn được gọi là bỏng RF. Dòng điện RF có thể dễ dàng ion hóa không khí, tạo ra vùng dẫn điện qua nó. Đặc tính này được áp dụng cho các khối "cao tần" trong hàn hồ quang điện, cách hàn này sử dụng dòng điện ở tần số cao hơn so với phân bố công suất sử dụng. Đặc tính khác là khả năng xuất hiện dòng điện qua nơi chứa vật liệu cách điện, như chất li điện môi của một tụ điện. Khi dẫn điện bằng một dây cáp điện thông thường, dòng điện RF có xu hướng phản xạ không liên tục trong cáp chẳng hạn như trong các bộ đấu nối và phản xạ ngược trở lại nguồn, gây ra sóng đứng, do đó dòng điện RF phải được truyền trên một loại cáp đặc biệt gọi là đường dây truyền tải.

*c, Ứng dụng của sóng RF*

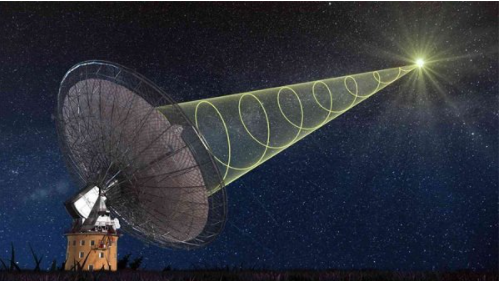
***Sóng RF trong thẩm mỹ:*** sóng RF được sử dụng nhiều trong thẩm mỹ, với những tác động trực tiếp lên làn da, vào trong lớp trung bì, hạ bì, kích thích tăng sinh collagen, giảm mỡ, tạo ra những thay đổi đặc biệt đối với bề mặt da và diện mạo cơ thể bên ngoài. Ứng dụng của sóng RF cực kỳ phổ biến trong thời gian gần đây.



Hình 1. 1 Ứng sụng sóng RF trong thẩm mỹ

*Sóng RF trong y khoa:*Trong nhiều xét nghiệm y khoa, người ta cũng đã biết tận dụng sóng RF trong việc xét nghiệm và tìm ra nguyên nhân bệnh lý viêm đa khớp dạng thấp. Những kết quả mà sóng RF mang lại thường chính xác tuyệt đối, giúp bác sĩ kịp thời có những can thiệp trong quá trình điều trị, giúp bệnh nhân nhanh chóng lấy lại sức khỏe.

***Sóng RF trong vô tuyến điện:*** Tìm hiểu sóng RF trong vô tuyến điện, chắc hẳn bạn đã hiểu cụm từ sóng RF là gì hay Radio Frequency là gì. Thực tế, Radio Frequency được ứng dụng nhiều trong vô tuyến điện, phát sóng hình ảnh, âm thanh. RF được ứng dụng nhiều trong radio, truyền thanh, truyền hình.



Hình 1. 2 Ứng dụng sóng RF trong vô tuyến

### *Sóng siêu âm*

*a, Khái niệm*

Sóng siêu âm là một loại sóng âm có tần số cao hơn giới hạn nghe thấy của con người. Siêu âm không khác với các loại âm thanh bình thường về tính chất, chỉ khác nhau ở chổ con người không thể nghe thấy được. Tân suất của nó tầm cở 20Hz cho tới vài Gigahertz. Trong áp suất khí quyển siêu âm có bước sóng dưới 1.9 cm.

*b, Nguyên lý hoạt động*

**Sóng siêu âm** di chuyển trong nhiều môi trường khác nhau như: không khí, chất lỏng, chất rắn, giống như âm thanh bình thường. Với tốc độ tương tương với tốc độ của âm thanh, nhưng do có tần số cao hơn nên bước sóng ngắn hơn bước sóng âm thanh.

*c, Ứng dụng sóng siêu âm*

Nhờ có bước sóng ngắn nên độ phân giải của ảnh chụp siêu âm đủ để phân biệt các vật thể rất nhỏ cỡ milimet. Chính vì vậy siêu âm được ứng dụng rất nhiều trong đời sống, như y học siêu âm chuẩn đoán hình ảnh, hoặc dò tìm mối hàn bị đứt, ứng dụng tẩy rửa bằng sóng siêu âm, hàn bằng sóng siêu âm....

***Máy tẩy rửa sóng siêu âm:*** Dòng máy tẩy rửa sóng siêu âm được cấu tạo chủ yếu bộ rung sóng siêu âm (có hộp điều khiển), bộ đầu dò, bể tẩy rửa và bộ phận gia nhiệt. Trong đó, máy rung sóng siêu âm sẽ sản sinh ra tín hiệu dao động của điện từ cũng như cung cấp nguồn năng lượng làm việc cho máy. [Thiết bị sóng siêu âm](https://vnhwd.com/vi/tin-tuc/thiet-bi-song-sieu-am-va-nhung-dieu-ban-can-biet) có thể sản sinh ra tín hiệu sóng siêu âm cao khoảng 20KHZ, cung cấp năng lượng cho đầu dò. Đầu dò sẽ khiến máy phát sóng siêu âm sản sinh tín hiệu rung điện từ, và chuyển hóa thành một loại chấn động rung siêu âm cho máy. Quá trình này cũng chính là quá trình chuyển đổi từ năng lượng này sang năng lượng khác. Đồng thời, sẽ sản sinh ra vô số bọt bong bóng (hiện tượng không hóa), khi đó trên bề mặt vật thể cần tẩy rửa sẽ bùng nổ, phát ra luồng xung kích rất mạnh len lõi quét qua toàn bộ bề mặt vật thể, nó vận hành như chiếc chổi vô hình từ đó giúp linh kiện được tẩy sạch; gia nhiệt nhằm mục đích nâng cao hiệu quả cũng như tốc độ của việc tẩy rửa.

*Máy hàn bằng sóng siêu âm:* Do mạch rung của máy phát sóng siêu âm phát rung cùng công suất khuếch đại đầu ra của tín hiệu tần số sóng siêu âm. Hiệu ứng chuyển đổi từ điện áp ngược của đầu dò thành hiệu ứng rung động rung với tần số cơ học. Sau đó từ khuôn hàn chuyển đến đến linh kiện nhựa cần hàn. Thông qua sự ma sát giữa các phân tử bên trong và trên bề mặt linh kiện, giúp cho nhiệt độ tại các mối hàn nhanh chống tăng cao khi nhiệt độ tại các mối hàn đạt đến mức làm các phôi hàn nóng chảy. Sẽ giúp cho các mối hàn kết dính lại với nhau, từ đó nhanh chóng lắp đầy các chỗ hở tại mối hàn, Và khi các chấn động dừng lại thì cũng là lúc các mối hàn trên linh kiện cũng nguội đi, và định hình cho các mối hàn thật hoàn mĩ.

*Y học*: [Trong ngành y học](https://www.fda.gov/radiation-emitting-products/medical-imaging/ultrasound-imaging) chắc hẳn bạn đã nghe nhiều về "siêu âm". Siêu âm bào thai, ổ bụng... Đó là phương pháp sử dụng sóng siêu âm có tần số cao tạo ra hình ảnh về cấu trúc bên trong cơ thể con người. Nhờ đó bác sĩ có thể chuẩn đoán được các bất thường bên trong cơ thể và đưa ra phác đồ điều trị hợp lý.

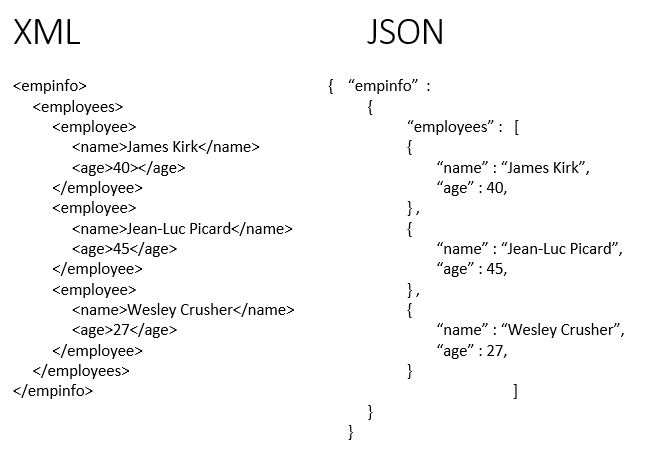


Hình 1. 3 Máy siêu âm y khoa

### ***J****ava****S****cript****O****bject****N****otation*

*a, Json là gì?*

**JSON** là viết tắt của **J**ava**S**cript **O**bject **N**otation, là một kiểu định dạng dữ liệu tuân theo một quy luật nhất định mà hầu hết các ngôn ngữ lập trình hiện nay đều có thể đọc được. **JSON**là một tiêu chuẩn mở để trao đổi dữ liệu trên web.



Hình 1. 4 Cấu trúc Json

*b, Định nghĩa*

Định dạng JSON sử dụng các cặp *key* – *value* để dữ liệu sử dụng. Nó hỗ trợ các cấu trúc dữ liệu như đối tượng và mảng.

Ta có thể thấy cú pháp của JSON có 2 phần đó là key và value

* Chuỗi JSON được bao lại bởi dấu ngoặc nhọn {}
* Các key, valuecủa JSON bắt buộc phải đặt trong dấu nháy kép {“}, nếu bạn đặt nó trong dấu nháy đơn thì đây không phải là một chuỗi JSON đúng chuẩn. Nếu trường hợp trong value của bạn có chứa dấu nháy kép " thì hãy dùng dấu () để đặt trước nó, ví dụ  "json là gì".
* Nếu có nhiều dữ liệu thì dùng dấu phẩy , để ngăn cách.
* Các key của JSON bạn nên đặt chữ cái không dấu hoặc số, dấu \_ và không có khoảng trắng., ký tự đầu tiên không nên đặt là số.

*c, Cấu trúc chuỗi Json*

**Object** trong Json được thể hiện bằng dấu ngoặc nhọn {}. Khái niệm Object trong Json cũng khá tương đồng với Object trong Javascript. Tuy nhiên, Object trong Json vẫn có những giới hạn như:

* **Key**: phải luôn nằm trong dấu ngoặc kép, không được phép là biến số.
* **Value**: Chỉ cho phép các kiểu dữ liệu cơ bản: numbers, String, Booleans, arrays, objects, null. Không cho phép function, date, undefined.
* Không cho phép dấy phẩy cuối cùng như Object trong Javascript.

# **CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG**



## **Phân tích và thiết kế**

### *Phân tích hệ thống*

Hệ thống gồm hai thiết bị phát và thu.

*a, Thiết bị phát*

* Gồm một vi xử lý trung tâm Atmega328p
* Một cảm biến siêu âm HC-SR04
* Một module thu phát RF Nrf24l01

*b, Thiết bị thu*

* Gồm một vi xử lý trung tâm Atmega328p
* Một module relay 5v 10A
* Một module thu phát RF Nrf24l01
* Một LCD 16x2 để hiển thị dữ liệu

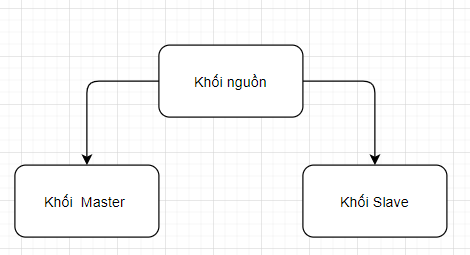
## **Đặc tả hệ thống**

Hệ thống sau khi xây dựng xong phải đảm bảo các yêu cầu thiết kế sau:

* Hệ thống có khả năng đóng mở nguồn điện xoay chiều hoặc 1 chiều
* Hệ thống có khả năng truyền tín hiệu qua sóng RF
* Có thể đo được mực nước hiện tại trong bể chứa
* Thay thế được phao điện truyền thống
* Tiết kiệm năng lượng cho người dùng
* Đảm bảo an toàn cho người sử dụng
* Tự động bật máy bơm khi hết nước và ngược lại
* Tiết kiệm chi phí, thời gian

### *Sơ đồ khối của hệ thống và chức năng từng khối*

*a, Sơ đồ khối của hệ thống*

**

Hình 2. 1 Sơ đồ khối toàn hệ thống

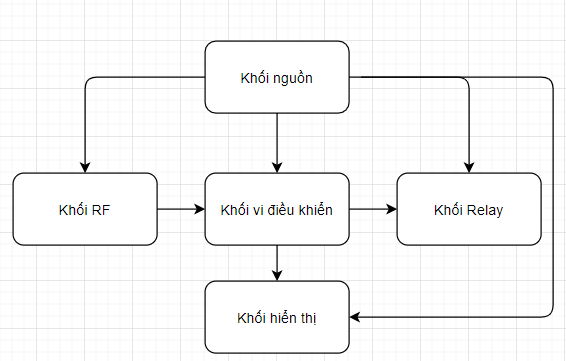
*b, Chức năng chi tiết từng khối*

*Khối nguồn:* Cung cấp điện áp ổn định tới các khối trong hệ thống, khối nguồn được sử dụng trong hệ thống có điện áp 5 VDC và 3.3 VDC.

*Khối Master:* Đây là khối gắn ở bể chứa có tác dụng đo và gửi tín hiệu về cho khối Slave

*Khối Slave:* Khối này có chức năng thực thi các lệnh được lập trình sẵn khi có tín hiệu từ Mater gửi đến.

### *Sơ đồ khối Slave*



Hình 2. 2 Sơ đồ khối Slave

*Khối nguồn:* Cung cấp điện áp ổn định tới các khối trong hệ thống, khối nguồn được sử dụng trong hệ thống có điện áp 5 VDC và 3.3 VDC.

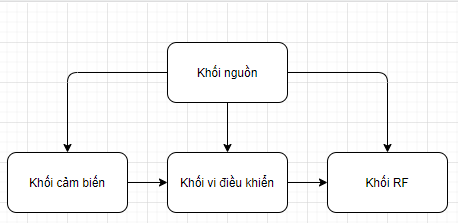
*Khối hiển thị:* Hiển thị các thông số nhận được.

*Khối RF*: Nhận dữ liệu từ master rồi gửi cho vi điều khiển xử lý

*Khối Relay:* Chấp hành lệnh và vi điều khiển xử lý được.

### *Sơ đồ khối Master*

*a, Sơ đồ khối*



Hình 2. 6 Sơ đồ khối Master

*b, Chức năng các khối*

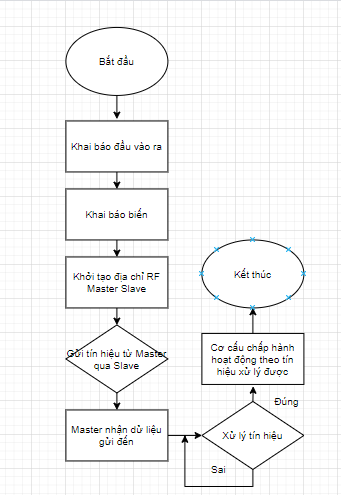
*Khối nguồn:* Cung cấp điện áp ổn định tới các khối trong hệ thống, khối nguồn được sử dụng trong hệ thống có điện áp 5 VDC và 3.3 VDC.

*Khối cảm biến:* Đo đạc các thông số yêu cầu và truyền về cho khối vi điều khiển tính toán và xử lý

Khối RF: Có chức năng gửi dữ liệu mà vi điều khiển khiển xử lý được khi nhận từ cảm biến cho Slave.

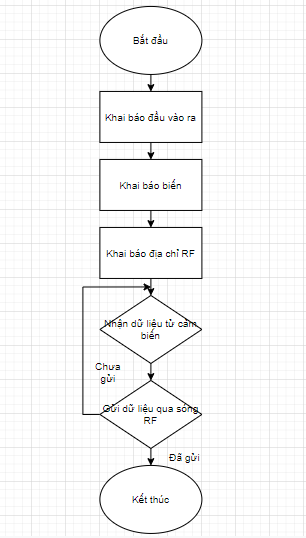
### *Lưu đồ thuật toán*

*a, Lưu đồ toàn hệ thống*



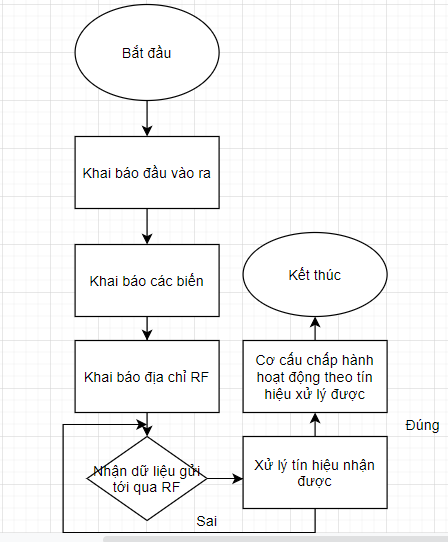
Hình 2. 3 Sơ đồ thuật toán hệ thống

*b, Lưu đồ thuật toán Master*



Hình 2. 4 Lưu đồ thuật toán Master

*c, Lưu đồ thuật toán Slave*



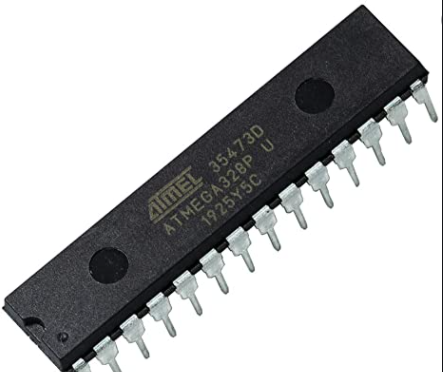
Hình 2. 5 Lưu đồ thuật toán Slave

## **Giới thiệu một số công cụ, thiết bị sử dụng**

### *Atmega328p*

*a, Giới thiệu*

Atmega328 là một chíp vi điều khiển được sản xuất bời hãng [Atmel](http://www.atmel.com/) thuộc họ MegaAVR có sức mạnh hơn hẳn [Atmega8](https://linhkienbandan.com/shop/atmega8-16pu-vi-dieu-khien-8-bit/). Atmega 328 là một bộ vi điều khiển 8 bít dựa trên kiến trúc RISC bộ nhớ chương trình 32KB ISP flash có thể ghi xóa hàng nghìn lần, 1KB EEPROM, một bộ nhớ RAM vô cùng lớn trong thế giới vi xử lý 8 bít (2KB SRAM)

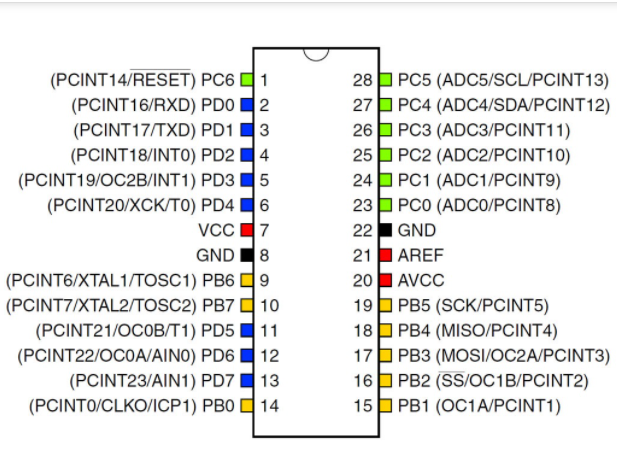
Với 23 chân có thể sử dụng cho các kết nối vào hoặc ra i/O, 32 thanh ghi, 3 bộ timer/counter có thể lập trình, có các gắt nội và ngoại (2 lệnh trên một vector ngắt), giao thức truyền thông nối tiếp USART, SPI, I2C. Ngoài ra có thể sử dụng bộ biến đổi số tương tự 10 bít (ADC/DAC) mở rộng tới 8 kênh, khả năng lập trình được watchdog timer, hoạt động với 5 chế độ nguồn, có thể sử dụng tới 6 kênh điều chế độ rộng xung (PWM), hỗ trợ bootloader.  


Hình 2. 9 Atmega328p

*b, Thông số kỹ thuật*

* Điện áp sử dụng: 5 VDC
* Điện áp giao tiếp: 5 VDC
* Dòng tiêu thụ: Max 0,2mA (nên sử dụng module cấp nguồn riêng cho mạch).
* Kiến trúc: AVR 8bit
* Xung nhịp lớn nhất: 20Mhz
* Bộ nhớ chương trình (FLASH): 32KB
* Bộ nhớ EEPROM: 1KB
* Bộ nhớ RAM: 2KB
* Số timer: 3 timer gồm 2 timer 8-bit và 1 timer 16-bit
* Số kênh xung PWM: 6 kênh (1timer 2 kênh)

*c, Chức năng các chân*



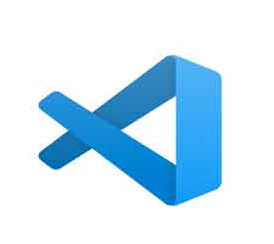
Hình 2. 10 Sơ đồ chân Atmega328p

### *Visual Studio Code và Platform IO*

*a, Visual Studio Code*

Là một trình biên tập lập trình code miễn phí dành cho Windows, Linux và macOS, Visual Studio Code được phát triển bởi Microsoft. Nó được xem là một sự kết hợp hoàn hảo giữa IDE và Code Editor.

Visual Studio Code hỗ trợ chức năng debug, đi kèm với Git, có syntax highlighting, tự hoàn thành mã thông minh, snippets, và cải tiến mã nguồn. Nhờ tính năng tùy chỉnh, Visual Studio Code cũng cho phép người dùng thay đổi theme, phím tắt, và các tùy chọn khác.



Hình 2. 11 Visual Studio Code

Một số tính năng của Visual Studio Code:

* Hỗ trợ nhiều ngôn ngữ lập trình
* Hỗ trợ đa nền tảng
* Cung cấp kho tiện ích mở rộng
* Kho lưu trữ an toàn
* Hỗ trợ Web, lưu trữ dữ liệu đa dạng phân cấp

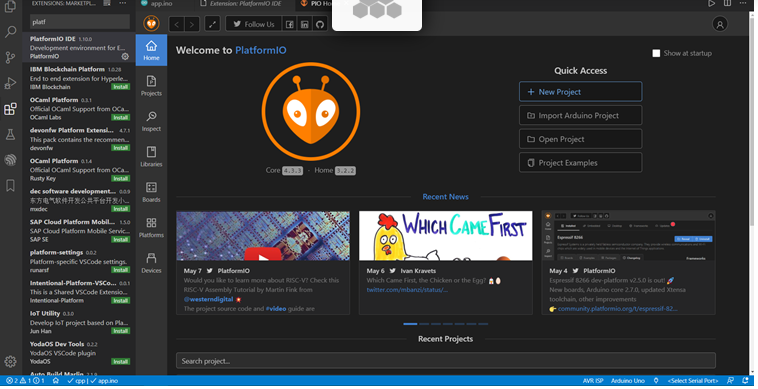
*b, Platform IO*

[PlatformIO](http://platformio.org/) là mội plugin có khả năng lập trình Arduino/ ARM mbed nó hỗ trợ tới hơn 250 board khác nhau.



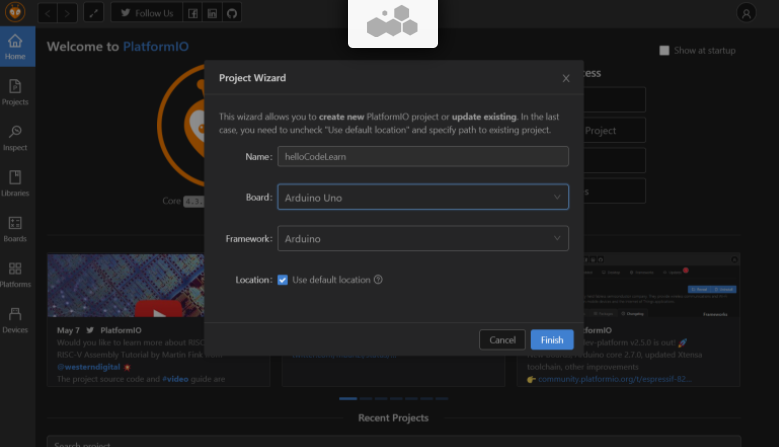
Hình 2. 12 Platform IO

Để bắt đầu với project mới, các bạn chọn New Project:



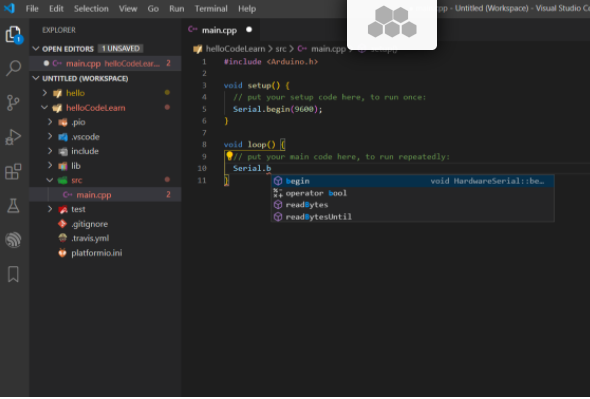
Hình 2. 13 Tạo Project

Điền thông tin vào, nhớ rằng platformio sẽ cấu hình luôn board mạch bạn sử dụng ngay khi khởi tạo project:



Hình 2. 14 Điền thông tin project

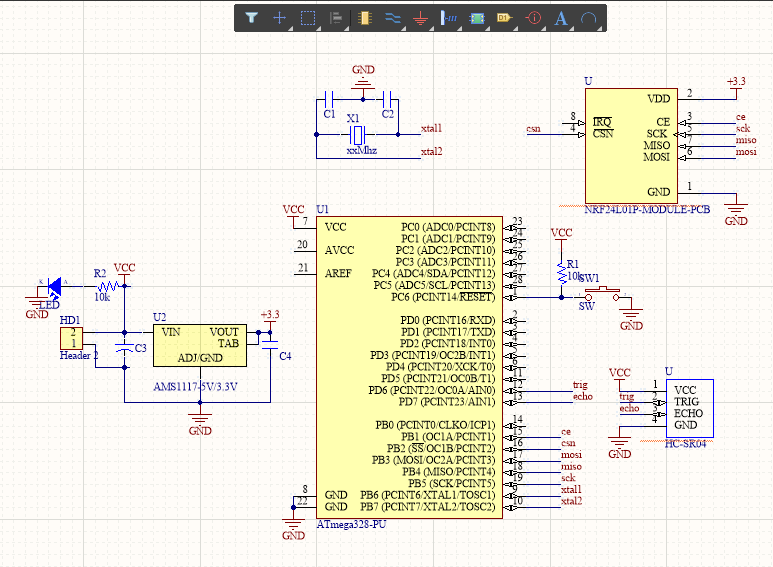
Kết quả sau khi tạo project:



Hình 2. 15 Cấu trúc project sau khi tạo xong

### *Sơ đồ nguyên lý*

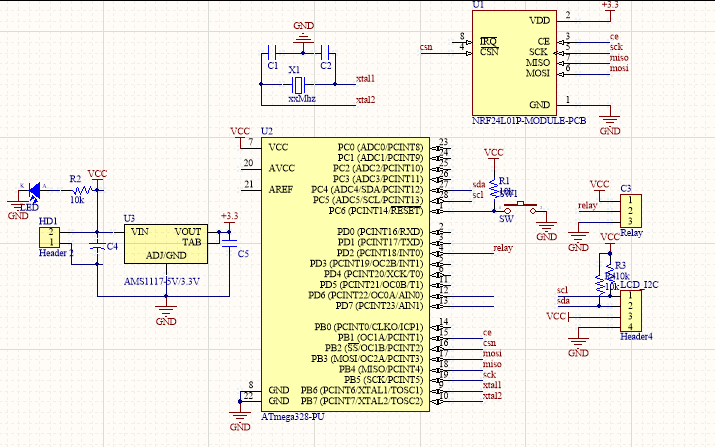
*a, Sơ đồ nguyên lý mạch Master*



Hình 2. 7 Sơ đồ nguyên lý mạch Master

Sơ đồ gồm: Khối nguồn, khối tạo dao động, khối RF, khối cảm biến khoảng cách.

*b, Sơ đồ nguyên lý mạch Slave*

**

Hình 2. 8 Sơ đồ nguyên lý mạch Slave

### *Altium Design*

*a, Giới thiệu*

Altium Designer trước kia có tên gọi quen thuộc là Protel DXP, là một trong những công cụ vẽ mạch điện tử mạnh nhất hiện nay. Được phát triển bởi hãng Altium Limited. Altium designer là một phần mềm chuyên nghành được sử dụng trong thiết kế mạch điện tử. Nó là một phần mềm mạnh với nhiều tính năng thú vị, tuy nhiên phần mềm này còn được ít người biết đến so với các phần mềm thiết kế mạch khác như orcad hay proteus.



Hình 2. 16 Altium Design

*b, Một số đặc trưng*

Giao diện thiết kế, quản lý và chỉnh sửa thân thiện, dễ dàng biên dịch, quản lý file, quản lý phiên bản cho các tài liệu thiết kế.

Hỗ trợ mạnh mẽ cho việc thiết kế tự động, đi dây tự động theo thuật toán tối ưu, phân tích lắp ráp linh kiện. Hỗ trợ việc tìm các giải pháp thiết kế hoặc chỉnh sửa mạch, linh kiện, netlist có sẵn từ trước theo các tham số mới.

Mở, xem và in các file thiết kế mạch dễ dàng với đầy đủ các thông tin linh kiện, netlist, dữ liệu bản vẽ, kích thước, số lượng…

Hệ thống các thư viện linh kiện phong phú, chi tiết và hoàn chỉnh bao gồm tất cả các linh kiện nhúng, số, tương tự…

 Đặt và sửa đối tượng trên các lớp cơ khí, định nghĩa các luật thiết kế, tùy chỉnh các lớp mạch in, chuyển từ schematic sang PCB, đặt vị trí linh kiện trên PCB.

Mô phỏng mạch PCB 3D, đem lại hình ảnh mạch điện trung thực trong không gian 3 chiều, hỗ trợ MCAD-ECAD, liên kết trực tiếp với mô hình STEP, kiểm tra khoảng cách cách điện, cấu hình cho cả 2D và 3D

Hỗ trợ thiết kế PCB sang FPGA và ngược lại.

*c, Tóm tắt các bước thiết kế mạch trên Altium*

- Đặt ra các yêu cầu bài toán.

-  Lựa chọn linh kiện.

- Thiết kế mạch nguyên lý.

- Lựa chọn các chân linh kiện để chuyển sang mạch in Update mạch nguyên lý sang mạch in.

-  Lựa chọn kích thước mạch in Sắp sếp các vị trí các loại linh kiện  như điện trở , tụ điện, IC...

-  Đặt kích thước các loại dây nối.

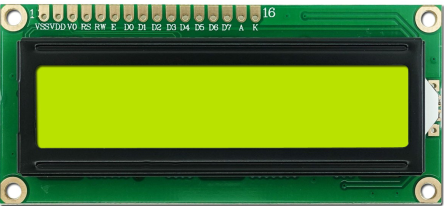
- Đi dây trên mạch.

- Kiểm tra toàn mạch.

### *LCD16x2*

*a, Giới thiệu*

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



Hình 2. 20 LCD 16x2

*b, Sơ đồ chân và chức năng*

|  |  |  |
| --- | --- | --- |
| Chân | Ký hiệu | Mô tả |
| 1 | Vss | Chân nối đất cho LCD, khi thiết kế mạch ta nối chân này với GND của mạch điều khiển |
| 2 | VDD | Chân cấp nguồn cho LCD, khi thiết kế mạch ta nối chân này với VCC=5V của mạch điều khiển |
| 3 | VEE | Điều chỉnh độ tương phản của LCD. |
| 4 | RS | Chân chọn thanh ghi (Register select). Nối chân RS với logic “0” (GND) hoặc logic “1” (VCC) để chọn thanh ghi.  + Logic “0”: Bus DB0-DB7 sẽ nối với thanh ghi lệnh IR của LCD (ở chế độ “ghi” - write) hoặc nối với bộ đếm địa chỉ của LCD (ở chế độ “đọc” - read)  + Logic “1”: Bus DB0-DB7 sẽ nối với thanh ghi dữ liệu DR bên trong LCD. |
| 5 | R/W | Chân chọn chế độ đọc/ghi (Read/Write). Nối chân R/W với logic “0” để LCD hoạt động ở chế độ ghi, hoặc nối với logic “1” để LCD ở chế độ đọc. |
| 6 | E | Chân cho phép (Enable). Sau khi các tín hiệu được đặt lên bus DB0-DB7, các lệnh chỉ được chấp nhận khi có 1 xung cho phép của chân E.  + Ở chế độ ghi: Dữ liệu ở bus sẽ được LCD chuyển vào(chấp nhận) thanh ghi bên trong nó khi phát hiện một xung (high-to-low transition) của tín hiệu chân E.  + Ở chế độ đọc: Dữ liệu sẽ được LCD xuất ra DB0-DB7 khi phát hiện cạnh lên (low-to-high transition) ở chân E và được LCD giữ ở bus đến khi nào chân E xuống mức thấp. |
| 7 - 14 | DB0 - DB7 | Tám đường của bus dữ liệu dùng để trao đổi thông tin với MPU. Có 2 chế độ sử dụng 8 đường bus này :  + Chế độ 8 bit : Dữ liệu được truyền trên cả 8 đường, với bit MSB là bit DB7.  + Chế độ 4 bit : Dữ liệu được truyền trên 4 đường từ DB4 tới DB7, bit MSB là DB7 |
| 15 | - | Nguồn dương cho đèn nền |
| 16 | - | GND cho đèn nền |

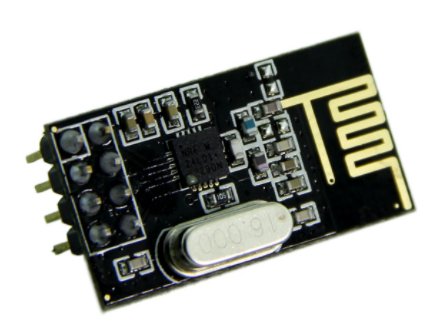
Bảng 2. 1 Chức năng các chân LCD

### *Module NRF24L01*

*a, Giới thiệu*

Module thu phát không dây 2.4GHz Nrf24L01+ sử dụng chip thu phát không dây Nrf24L01+ của Nordic.

Module thu phát không dây 2.4GHz cho bạn giải pháp kết nối không dây nhanh và chi phí thấp. Chỉ cần 2 module, bạn có thể lập trình để chia xẻ dữ liệu giữa 2 vi điều khiển hoặc 2 board Arduino với nhau.



Hình 2. 17 Module NRF24L01

*b, Các tính năng cơ bản*

Điện áp làm việc: 1.9V - 3.6V.

Đầu vào dữ liệu tương thích với điện áp 5V (5V tolerance).

Có thể lựa chọn 3 mức tốc độ kết nối là 2Mbit/s, 1Mbit/s và 250Kbit/s. Tốc độ càng thấp, thì khả năng thu phát càng xa, bù lại dữ liệu truyền / giây bị giảm xuống

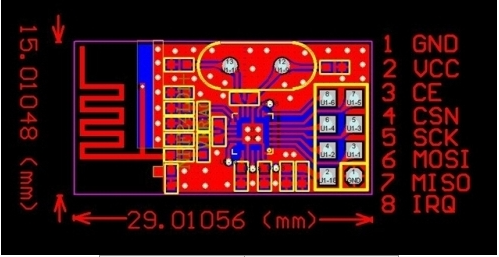
Số kênh kết nối: 6 kênh

  Phương thức giao tiếp : SPI

Có sẵn anten trên module, bạn chỉ cần kết nối với vi điều khiển và chạy ứng dụng.

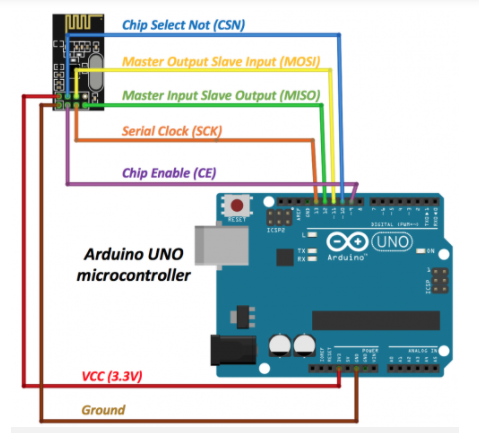
Tiêu thụ dòng thấp. Tại tốc độ truyền 2Mbit/s, dòng tối đa là 12mA. Chế độ nghỉ, module 2.4Ghz tiêu thụ 32uA.

*c, Các chân giao tiếp*



Hình 2. 18 Các chân giao tiếp module NRF2410L

Ghép nối với Arduino

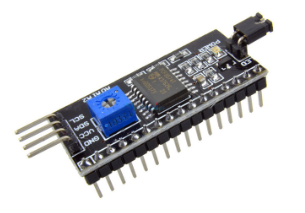


Hình 2. 19 Ghép nối với Arduino

### *Module I2C*

*a, Giới thiệu*

LCD có quá nhiều nhiều chân gây khó khăn trong quá trình đấu nối và chiếm dụng nhiều chân trên vi điều khiển. **Module I2C LCD** ra đời và giải quyết vấn để này cho bạn.



Hình 2. 21 Module I2C

*b, Thông số kỹ thuật*

* Điện áp hoạt động: 2.5-6V DC.
* Hỗ trợ màn hình: LCD1602,1604,2004 (driver HD44780).
* Giao tiếp: I2C.
* Địa chỉ mặc định: 0X27 (có thể điều chỉnh bằng ngắn mạch chân A0/A1/A2).
* Tích hợp Jump chốt để cung cấp đèn cho LCD hoặc ngắt.
* Tích hợp biến trở xoay điều chỉnh độ tương phản cho LCD.

### *Cảm biến khoảng cách HC\_SR04*

*a, Giới thiệu*

Cảm biến khoảng cách HC-SR04là cảm biến dùng để xác định khoảng cách trong phạm vi nhỏ bằng cách phát sóng siêu âm. Cảm biến với độ chính xác chính xác khá cao (với khoảng cách nhận biết nhỏ nhất 3mm) và độ ổn định cao trong quá trình sử dụng, đồng thời dễ dàng kết nối với các MCU (Arduino, DSP, AVR, PIC, ARM…)

Các ứng dụng thường thấy với cảm biến như: xe tránh vật cản, xác định khoảng cách của vật thể đến cảm biến, hay mô phỏng làm radar…

*b, Thông số kỹ thuật*

* Điện áp làm việc: 5V (DC)
* Góc quét tốt nhất: 30°
* Khoảng cách hoạt động: 2cm ~ 450cm
* Khoảng cách nhận biết nhỏ nhất: 3mm
* Pinout: VCC, GND, TRIG, ECHO



Hình 2. 24 Cảm biến khoảng cách HC\_SR04

### *Mạch nạp code AVR*

*a, Thông số kỹ thuật*

* Mạch nạp usb asp chuyên nạp cho dòng Vi điều khiển AVR.
* Giao tiếp và cấp nguồn qua cổng USB 5VDC.
* Cổng nạp chuẩn ISP 10 pins.
* Hỗ trợ set Fuse bit.
* Onboard ATMega8(L).
* Onboard over-current protection.
* 3.3V and 5V logic level Selection jumper (JP1)
* Self program (firmware update) jumper (JP2)
* Clock Speed control Jumper (JP3)
* USB Type A connector for directly pluging in to PC USB port
* On board power and programmer status LEDs
* Supported OS: Windows 8, Windows 7, Works under multiple platforms. Linux, Mac\_OS\_X and Windows are tested.



Hình 2. 22 Mạch nạp AVR

*b, Sơ đồ chân*

* Nạp cho epprom 24Cxx
* (RST)MCLR-Not Connect
* VDD-Vcc
* GND-Vss
* (MISO)PGD – Not Connect
* (SCK)PGC -SCL
* (MOSI)AUX-SDA
* Chú ý phải có trở từ 2-10k kéo lên dương nguồn ở các chân SCL và SDA.



Hình 2. 23 Sơ đồ chân mạch nạp AVR

# **CHƯƠNG 3: KẾT QUẢ ĐÁNH GIÁ**



## **Các bước sử dụng hệ thống**

*Bước 1:* Đặt thiết bị master cố định(tránh nước vào thiết bị) vào bể chứa sao cho cảm biến vuông góc với mặt phẳng nước.

*Bước 2:* Sau khi cố định thiết bị ta gắn thiết bị slave sao cho phù hợp với khoảng cách thu phát của master – slave.

*Bước 3:* Cấp nguồn cho cả hai thiết bị

*Bước 4*: Chờ đợi kết quả thiết bị hoạt động

## **Đánh giá hệ thống**

### *Ưu điểm*

Hệ thống đáp ứng được đúng những yêu cầu đã được được đưa ra.

Giúp người sử dụng có thể điều khiển tự động được máy bơm nước

Giúp người dùng tiết kiệm điện năng

Thay thế được phao điện truyền thống

Chống giật điện cho người sử dụng

### *Nhược điểm*

Khoảng cách điều khiển hạn chế (3m - 5m).

Chỉ sử dụng để điều khiển được một thiết bị.

Thiết bị cần được đảm bảo chống ẩm để hoạt động tốt nhất có thể.

## **Kết luận**

Trải qua thời gian nghiên cứu và thực hiện đề tài đã giúp em hiểu ra hơn về vai trò, các ứng dụng của vi điều khiển trong thực tế, cũng như cách lập trình cho vi điều khiển Atmega328p, cách thiết kế mạch trên phần mềm Altium, cách truyền thông tin bằng sóng RF

Ngoài ra qua đề tài này em cũng hiểu rõ hơn về các cách truyền dữ liệu qua sóng RF và các ứng dụng của lĩnh vực IOT trong đời sống từ đó có những hướng nghiên cứu và phát triển những thiết bị thông minh.

# **Tài liệu tham khảo**

[1] http://arduino.vn/bai-viet/562-su-dung-module-nrf24l01

[2] https://yadi.sk/d/CXjuENmaEEwun

[3] https://arduinokit.vn/do-khoang-cach-bang-cam-bien-sieu-am-hc-srf04

# **Phụ lục**

a, Code Master

#include <Arduino.h>

#include <SPI.h>

#include "nRF24L01.h"

#include "RF24.h"

#define TRIG\_PIN 6

#define ECHO\_PIN 7

#define TIME\_OUT 5000

float GetDistance();

// const char on[] = "bat";

// const char off[] = "tat";

int on = 0;

int off = 1;

char msg[100] = "";

#define JSON\_KEY\_STATUS "status"

#define JSON\_KEY\_DISTANCE "khoangcach"

RF24 radio(9, 10);                // Chọn chân kết nối CE, CSN

const byte address[7] = "000001"; // Địa chỉ cần truyền

void setup()

{

  Serial.begin(9600);

  radio.begin();

  radio.openWritingPipe(address); // Set địa chỉ truyền

  radio.setPALevel(RF24\_PA\_MIN);  // Set chế độ hoạt động

  radio.stopListening();          // Đặt module NRF24L01 ở chế độ phát

  pinMode(TRIG\_PIN, OUTPUT);

  pinMode(ECHO\_PIN, INPUT);

}

void loop()

{

  int dai = GetDistance();

  Serial.print("do dai: ");

  Serial.println(dai);

  if (dai > 3 && dai < 15)

  {

    sprintf(msg, "{\"%s\":%d, \"%s\":%d}", JSON\_KEY\_STATUS, off, JSON\_KEY\_DISTANCE, dai);

    radio.write(msg, sizeof(msg));

    Serial.println("send ok");

  }

  if (dai > 15)

  {

    sprintf(msg, "{\"%s\":%d, \"%s\":%d}", JSON\_KEY\_STATUS, on, JSON\_KEY\_DISTANCE, dai);

    radio.write(msg, sizeof(msg));

    Serial.println("send ok");

  }

  // radio.write(text, sizeof(text)); //Gửi dữ liệu có trong mảng text

  delay(1500);

}

float GetDistance()

{

  long duration, distanceCm;

  digitalWrite(TRIG\_PIN, LOW);

  delayMicroseconds(2);

  digitalWrite(TRIG\_PIN, HIGH);

  delayMicroseconds(10);

  digitalWrite(TRIG\_PIN, LOW);

  duration = pulseIn(ECHO\_PIN, HIGH, TIME\_OUT);

  // convert to distance

  distanceCm = duration / 29.412 / 2;

  return distanceCm;

}

b, Code Slave

#include <Arduino.h>

#include <SPI.h>

#include "nRF24L01.h"

#include "RF24.h"

#include "LiquidCrystal\_I2C.h"

#include "ArduinoJson.h"

#define JSON\_KEY\_STATUS "status"

#define JSON\_KEY\_DISTANCE "khoangcach"

char\* stt = "Status:";

char\* distance = "Distance:";

char msg[200] = "";

char msg1[200] = "";

char\* on = "ON";

char\* off = "OFF";

LiquidCrystal\_I2C lcd(0x27,16,2);

StaticJsonBuffer<1024> JSONBuffer;

RF24 radio(9, 10); // Chá»n chÃ¢n káº¿t ná»‘i CE, CSN

const byte address[7] = "000001"; // Äá»‹a chá»‰ nháº­n

char text[32] = "";

void setup() {

Serial.begin(9600);

radio.begin();

radio.openReadingPipe(0, address); // Set Ä‘á»‹a chá»‰ nháº­n

radio.setPALevel(RF24\_PA\_MIN); // Set cháº¿ Ä‘á»™ hoáº¡t Ä‘á»™ng

radio.startListening(); // Set module NRF24L01 á»Ÿ cháº¿ Ä‘á»™ thupinMo

pinMode(2,OUTPUT);

digitalWrite(2,LOW);

Serial.println("doi du lieu");

lcd.init();

lcd.backlight();

lcd.setCursor(0,0);

lcd.print(stt);

lcd.setCursor(0,1);

lcd.print(distance);

}

void loop() {

if (radio.available()) { // Kiá»ƒm tra cÃ³ dá»¯ liá»‡u thu Ä‘Æ°á»£c hay khÃ´ng?

radio.read(text, sizeof(text)); // Äá»c dá»¯ liá»‡u thu Ä‘Æ°á»£c

// Chuyá»ƒn dá»¯ liá»‡u lÃªn cá»•ng Serial

//JsonObject &root = JSONBuffer.parseObject((char\*)text);

Serial.println(text);

JsonObject &root = JSONBuffer.parseObject((char\*)text);

int kc = root[JSON\_KEY\_DISTANCE];

int status = root[JSON\_KEY\_STATUS];

switch(status)

{

case 1:

digitalWrite(2,HIGH);

lcd.setCursor(7,0);

lcd.print(on);

lcd.print(" ");

lcd.setCursor(9,1);

lcd.print(kc);

lcd.print(" ");

break;

case 0:

digitalWrite(2,LOW);

lcd.setCursor(7,0);

lcd.print(off);

lcd.print(" ");

lcd.setCursor(9,1);

lcd.print(kc);

lcd.print(" ");

break;

}

}

}