

NGHIÊN CỨU ỨNG DỤNG TRÍ THÔNG MINH NHÂN TẠO VÀ ARDUINO ĐIỀU KHIỂN CÁC THIẾT BỊ ĐIỆN TRONG NGÔI NHÀ THÔNG MINH

RESEARCH ON ELECTRIC DEVICES CONTROLLING IN THE SMARHOUSE USING ARTIFICIAL INTELLIGENCE AND ARDUINO

NGUYỄN VĂN TIẾN*, HOÀNG XUÂN BÌNH

Khoa Điện-Điện tử, trường Đại học Hàng Hải Việt Nam

*Email liên hệ: nguyenvantien@vimaru.edu.vn

Tóm tắt

Hệ thống ngôi nhà thông minh (Smart home) ngày càng trở lên phổ biến do tính tiện ích mà nó mang lại. Trong một ngôi nhà thông minh tất cả các thiết bị điện sẽ được kết nối với nhau để tạo thành một hệ thống kín nhờ bộ điều khiển trung tâm. Hiện nay với sự phát triển của công nghệ cho phép tích hợp trí tuệ nhân tạo (AI - Artificial intelligence) trong bộ điều khiển trung tâm làm cho chúng thông minh hơn, tuy nhiên giá thành của các hệ thống đó thường rất cao. Để có thể khai thác được thế mạnh của AI sử dụng trong ngôi nhà thông với giá thành hợp lý cần phải được nghiên cứu để tìm ra giải pháp. Bài báo trình nghiên cứu cách thức sử dụng trí thông minh nhân tạo kết hợp với phần cứng Arduino để xây dựng bộ điều khiển trung tâm. Bộ điều khiển trung tâm đã được thử nghiệm làm việc ổn định và tin cậy.

Từ khóa: Ngôi nhà thông minh, trí thông minh nhân tạo, Arduino Nano, NodeMCU.

Abstract

Nowaday, Smart houses are becoming more popular because of the user-friendly for human life. All device are connected together to make a close-system by using a centre controller. Moreover, the growth of AI is a condition to design more intelligent controller, but the controlller always has a high price. Therefore, we need to research to find a cheaper solution. The paper research a method to make the controller with Arduino hardware. The controller successfully has been expermeting in house.

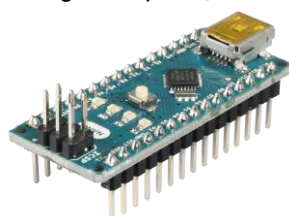
Keywords: Smart home, Artificial intelligence, Arduino Nano, NodeMcu.

1. Đặt vấn đề

Ngôi nhà thông minh và trí thông minh nhân tạo là hai lĩnh vực được nghiên cứu nhiều trong thời gian vài năm trở lại đây và đã đạt được những bước phát triển mạnh mẽ [1, 2].

Ngôi nhà thông minh được hiểu là một ngôi nhà/căn hộ được ứng dụng công nghệ theo một phương thức nào đó nhằm làm thỏa mãn và gia tăng giá trị cuộc sống của con người. Để làm được điều đó, ngôi nhà thông minh được trang bị các thiết bị điều khiển tự động như điều khiển và giám sát ánh sáng, nhiệt độ, các vấn đề về an ninh, cảnh báo sớm hỏa hoạn,... toàn bộ các thiết bị được kết nối tới bộ điều khiển trung tâm và được quản lý từ xa.

Còn trí thông minh nhân tạo là một phần mềm máy tính, tuy nhiên điểm khác biệt so với các phần mềm lập trình logic thông thường đó là việc ứng dụng các hệ thống học máy để mô phỏng quá trình xử lý của con người. Cụ thể hơn thì trí thông minh nhân tạo biết cách suy nghĩ và lập luận, biết giao tiếp, hiểu tiếng nói, biết tự học và thích nghi. Chính bởi khả năng tự học hỏi và suy luận mà trí thông minh nhân tạo được ứng dụng rộng rãi trong nhiều lĩnh vực [1, 2]. Mô hình trong bài báo sử dụng trí thông minh nhân tạo của Amazon được sử dụng như một trợ lý ảo có tên là Alexa, đây là một trong những trí tuệ nhân tạo mạnh mẽ và phổ biến nhất trên thế giới. Khả năng nhận dạng giọng nói tốt và giao tiếp được với nhiều loại phần cứng là điểm mạnh của Alexa..



Phần cứng Arduino Nano



Phần cứng NodeMCU



Amazon Echo Dot (Alexa)

Hình 1. Phần cứng sử dụng trong mô hình

Mô hình trong bài báo sử dụng loa Amazon Echo Dot có tích hợp Alexa để nhận dạng các câu lệnh điều khiển thiết bị từ người sử dụng kết hợp với phần cứng NodeMCU và Arduino để điều khiển các thiết bị điện (Hình 1). Sở dĩ bài báo lựa chọn Arduino Nano vì đây là phần cứng rất linh hoạt, có thể mở rộng đồng thời các công cụ phần mềm cũng được hỗ trợ rất tốt [3].

Điểm mới trong cách xây dựng mô hình đó lựa chọn các phần cứng với giá thành rẻ và phổ biến (Arduino) đồng thời khai thác các dịch vụ đám mây miễn phí (Sinric) để cho phép điều khiển các thiết bị từ bất kì đâu thông qua kết nối internet một cách dễ dàng. Ngoài ra cấu trúc trên còn cho phép mở rộng các đầu ra điều khiển một cách linh hoạt do Arduino Nano có rất nhiều các đầu ra.

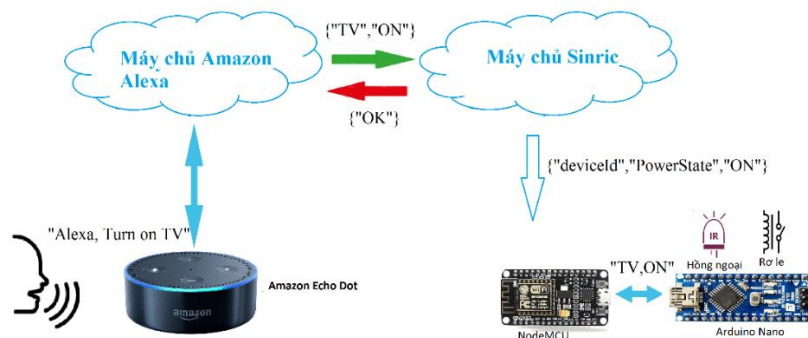
2. Cấu trúc bộ điều khiển

Trợ lý ảo Alexa có khả năng chơi các bản nhạc, cung cấp các thông tin về thời tiết, giao thông. Ngoài ra Alexa cũng có khả năng điều khiển các thiết bị điện thông của các hãng như Philips Hue, Belkin Wemo,... Tuy nhiên trong bài báo không sử dụng các thiết bị điện của các hãng nói trên mà tự tạo ra bộ điều khiển riêng phù hợp với điều kiện thực tế ở Việt Nam là có nhiều các thiết bị không có tính năng thông minh. Chính vì thế có thể sử dụng ngay cả trong những ngôi nhà bình thường chưa có các thiết bị điện thông minh giúp giảm giá thành và thời gian lắp đặt của hệ thống.

Nguyên tắc điều làm việc của bộ điều khiển trung tâm được mô tả như trong Hình 2 và yêu cầu kết nối với mạng internet để hoạt động. Bộ điều khiển bao gồm loa Amazon Echo dot tích hợp Alexa, NodeMCU và Arduino. Với loa Amazon Echo dot khi làm việc sẽ luôn trong trạng thái chờ câu lệnh tương tác của người sử dụng. Khi Amazon Echo dot phát hiện thấy từ khóa "Alexa" thì lập tức sẽ khởi động quy trình xử lý ngôn ngữ và gửi về máy chủ Amazon để phân tích [4].

Arduino là thiết bị thực hiện điều khiển các rơ le hoặc phát lệnh điều khiển thông qua sóng hồng ngoại tuy nhiên Arduino không có khả năng kết nối internet do vậy cần thêm phần cứng NodeMCU. NodeMCU có trang bị kết nối không dây (Wi-Fi) do vậy có thể kết nối dễ dàng vào máy chủ trung gian Sinric để nhận lệnh điều khiển rồi gửi tới Arduino.

Giả thiết chúng ta ra lệnh là "Alexa, Turn on TV". Câu lệnh sẽ được thiết bị Amazon Echo dot thu ghi lại và gửi đến trí thông minh nhân tạo cài đặt trong máy chủ của Amazon là Amazon Alexa để nhận dạng ngôn ngữ và phân tích ngữ pháp câu lệnh. Sau khi đã hiểu điều người sử dụng mong muốn thì Alexa sẽ gửi lệnh bao gồm thông tin về thiết bị (TV) và lệnh bật (ON) bằng cách gửi chuỗi {"TV, ON"} đến một máy chủ trung gian đã được người sử dụng quy định sẵn. Amazon Alexa không hỗ trợ kết nối trực tiếp tới phần cứng để giảm tải cho máy chủ vì thế chúng ta cần sử dụng một máy chủ trung gian để kết nối tới phần cứng, trong trường hợp này đó là Sinric. Sở dĩ lựa chọn Sinric bởi đây là một máy chủ miễn phí, thời gian đáp ứng rất nhanh và cài đặt dễ dàng trong phần mềm Alexa.



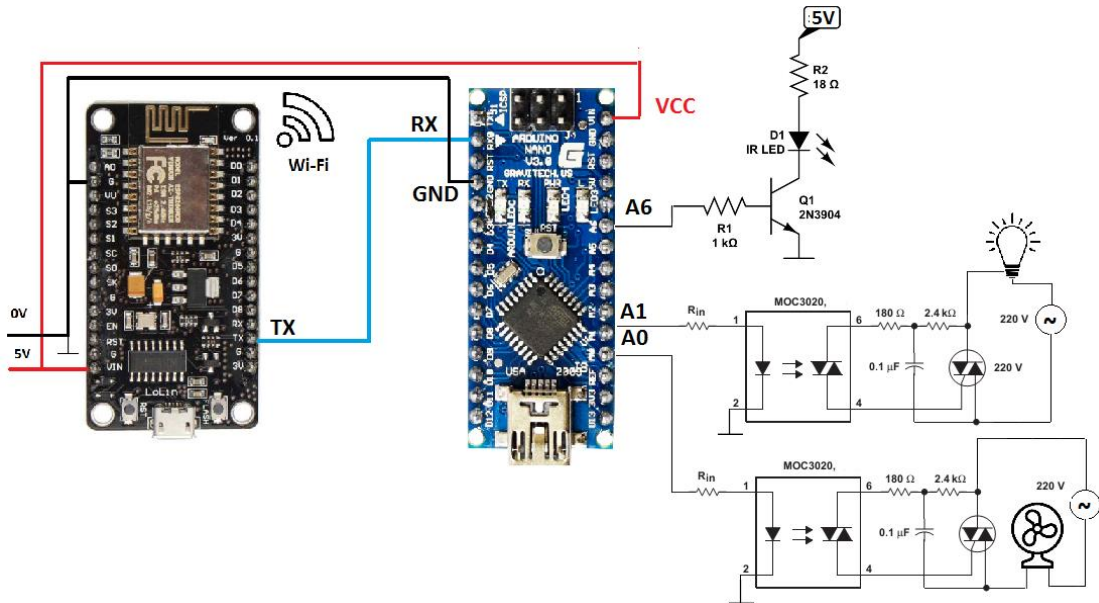
Hình 2. Cấu trúc của bộ điều khiển trung tâm smarthouse

Tiếp theo đó máy chủ Sinric sẽ gửi câu lệnh {"PowerState, ON"} kèm theo tên của thiết bị tới phần cứng NodeMCU thông qua kết nối internet. Cuối cùng câu lệnh được gửi tới Arduino để phát lệnh bật TV thông qua sóng hồng ngoại. Với các thiết bị không có bộ điều khiển từ xa bằng hồng ngoại thì thiết bị sẽ được bộ điều khiển cấp nguồn qua các bộ rơ le do vậy có thể áp dụng để điều khiển cho cả các thiết bị không thông minh trong một ngôi nhà.

3. Xây dựng mô hình

3.1. Kết nối phần cứng

Phần cứng ở đây bao gồm NodeMCU và Arduino Nano. Kết nối các thiết bị này với nhau được thực hiện thông qua cổng truyền thông nối tiếp (Serial port) của hai thiết bị (Hình 3). Việc sử dụng cổng truyền thông nối tiếp trong cấu trúc kết nối cho phép nhiều lệnh hơn được gửi từ Alexa tới Arduino trong khi đó số lượng đường dây tín hiệu chỉ là một sợi tiết kiệm so với dùng cấu trúc truyền dạng song song.

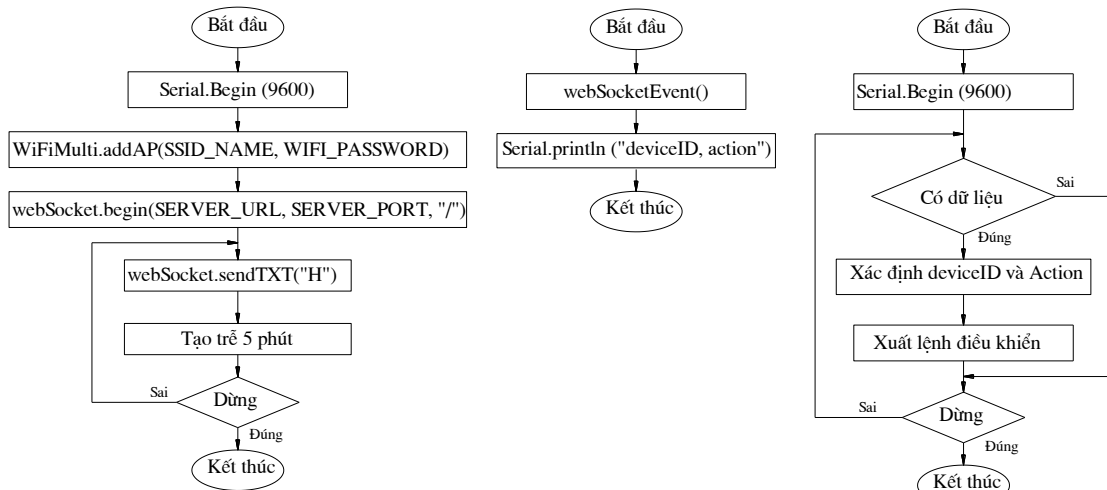


Hình 3. Kết nối phần cứng bộ điều khiển

Chân TX (chân truyền dữ liệu cổng nối tiếp) của NodeMCU được kết nối tới chân Rx (chân nhận) của Arduino Nano, tốc độ truyền được khởi tạo là 9600 bit/s. Đầu A6 của Arduino Nano được khuếch đại qua bằng transistor và dùng để điều khiển một LED hồng ngoại (Infrared-Light Emitting Diode) phát ra các lệnh điều khiển tới các thiết bị điện có bộ thu hồng ngoại (Tivi, điều hòa, đầu kỹ thuật số,...). Hai đầu A0 và A1 dùng để điều khiển Bật-Tắt cho các thiết bị chưa có bộ thu hồng ngoại như đèn, quạt,... Mạch động lực sử dụng MOC3020 và Triac thay cho các rơ le để giảm kích thước và tiết kiệm năng lượng.

3.2. Lập trình cho NodeMcu và Arduino

Cả NodeMCU và Arduino đều sử dụng chung phần mềm tên là Arduino IDE . Thuật toán điều khiển như trên Hình 4.



a. Chương trình chính NodeMCU

b. Ngắt nhận NodeMCU

c. Chương trình Arduino Nano

Hình 4. Lưu đồ thuật toán chương trình cho NodeMCU và Arduino Nano

Chương trình bao gồm 3 thuật toán đó là thuật toán chương trình chính của NodeMCU, ngắt nhận NodeMCU và chương trình của Arduino Nano.

Chương trình chính của NodeMCU bao gồm bước khởi tạo cổng truyền thông với Arduino dùng hàm Serial.Begin (). Tốc độ truyền là 9600 bit/giây. Sau đó là kết nối tới mạng Wi-Fi với thông số nhập vào SSID (tên mạng) và PASSWORD của mạng. Sau đó hàm websocket.Begin() được gọi để kết nối tới địa chỉ URL (Uniform Resource Locator) của máy chủ (iot.sinric.com) qua cổng đã mở là 80. Quá trình kết nối máy chủ thành công thì NodeMCU đã sẵn sàng để nhận dữ liệu từ máy chủ, tuy nhiên sau 5 phút mà không có sự trao đổi dữ liệu giữa máy chủ Sinric và NodeMCU thì máy chủ

sẽ đóng lại vì thế cần liên tục gửi dữ liệu 5 phút một lần để duy trì kết nối này sử dụng hàm `WebSocket.send()`.

Khi có lệnh điều khiển của người sử dụng, Sinric sẽ gửi dữ liệu đến NodeMCU với các thông số là tên thiết bị (deviceID) và hành động cần thực hiện tương ứng (action) và tạo ra một sự kiện ngắt với NodeMCU. NodeMCU xử lý và sẽ gửi các thông số đó tới Arduino Nano để thực hiện. Tất cả các công việc này được thực hiện bằng chương trình con ngắt của NodeMCU.

Arduino Nano sau khi nhận được dữ liệu sẽ căn cứ vào deviceID và action để điều khiển các thiết bị phù hợp.

Toàn bộ chương trình được viết và biên dịch trên phần mềm Arduino IDE như Hình 5 cho cả NodeMCU và Arduino.

Sau khi đã lập trình xong, tiến hành kết nối và hoàn thiện mô hình như trong Hình 6.

Mô hình bao gồm loa Amazon Echo dot, NodeMCU kết nối tới Arduino Nano. LED phát hồng ngoại được kết nối tới Arduino. Mạch động lực sử dụng triac cũng được kết nối tới Arduino. Trong mô hình có hai triac, đầu ra của triac được nối tới hai ổ cắm để điều khiển hai thiết bị điện không có công tắc ngoài như quạt, đèn,... Các thiết bị khác như tivi, điều hòa, đầu kỹ thuật số,... được điều khiển từ thông qua tín hiệu hồng ngoại.

Mô hình đã được thử nghiệm trong thực tế cho kết quả tốt, thời gian đáp ứng từ khi ra lệnh cho tới khi đáp ứng bởi phần cứng dưới 1 giây, độ trễ thời gian này hoàn toàn có thể chấp nhận được đối với điều khiển thiết bị. Kết nối giữa phần cứng và máy chủ được duy trì liên tục, không bị mất kết nối khi hoạt động trong thời gian dài. Các câu lệnh được Alexa nhận dạng với độ chính xác cao, và có điểm thú vị là càng sử dụng nhiều thì Alexa càng nhận dạng các câu lệnh chính xác hơn do đây là trí thông minh nhân tạo lên có khả năng tự học để thích nghi với ngôn ngữ của người sử dụng.

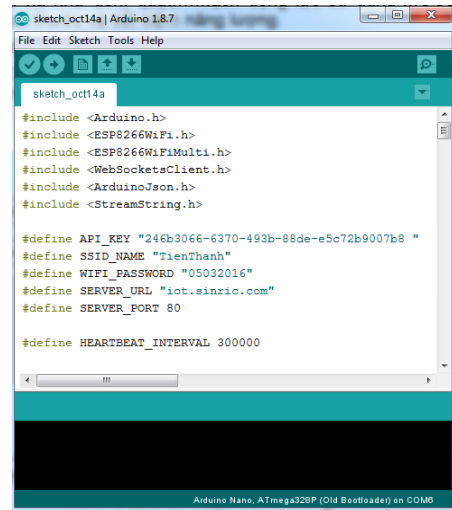
4. Kết luận

Bài báo đã giới thiệu giải pháp kết hợp trí thông minh nhân tạo với phần cứng Arduino để xây dựng mô hình bộ điều khiển trung tâm cho ngôi nhà thông minh. Với các lợi thế của trí thông minh nhân tạo là khả năng nhận biết ngôn ngữ, học hỏi và tư duy kết hợp với phần cứng Arduino linh hoạt có ứng dụng cao trong thực tế mang lại hiệu quả kinh tế. Ngoài ra bài báo cũng mở ra các hướng nghiên cứu mới cho giảng viên và sinh viên để nâng cao chất lượng giảng dạy, xa hơn nữa có thể xây dựng các mô hình tích hợp cao hơn để triển khai trong thương mại.

TÀI LIỆU THAM KHẢO

- [1] David Bregman, *Smart Home Intelligence -The eHome that Learns*, International Journal of Smart Home Smart Home Smart Home Smart Home Vol.4, No.4, October 2016.
- [2] Prof. Garima Tripathi and Melnita Dabre, *Home Automation System using Artificial Intelligence*, International Journal for Research in Applied Science & Engineering Vol 5, 2017.
- [3] Lê Mỹ Hà, *Lập trình IoT với Arduino*, Nhà xuất bản Thanh Niên, 2016.
- [4] Irene Lopatovska, *Talk to me: Exploring user interactions with the Amazon Alexa*, Journal of Librarianship and Information Science, 2018.

Ngày nhận bài: 10/3/2019
Ngày nhận bản sửa lần 01: 05/4/2019
Ngày nhận bản sửa lần 02: 16/4/2019
Ngày duyệt đăng: 28/4/2019

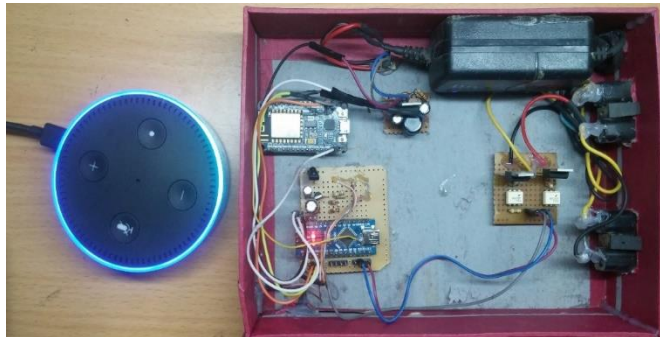


```
sketch_oct14a
#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <ESP8266WiFiMulti.h>
#include <WebSocketsClient.h>
#include <ArduinoJson.h>
#include <StreamString.h>

#define API_KEY "246b3066-6370-493b-88de-e5c72b9007b8 "
#define SSID_NAME "TienThanh"
#define WIFI_PASSWORD "05032016"
#define SERVER_URL "iot.sinric.com"
#define SERVER_PORT 80

#define HEARTBEAT_INTERVAL 300000
```

Hình 5. Chương trình viết trên Arduino IDE



Hình 6. Hình ảnh mô hình hoàn thiện