

Tp. HCM, ngày 15 tháng 7 năm 2018

## NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

|                   |                     |                |
|-------------------|---------------------|----------------|
| Họ tên sinh viên: | Nguyễn Văn Linh     | MSSV: 14141169 |
| Chuyên ngành:     | Điện tử công nghiệp | Mã ngành: 41   |
| Hệ đào tạo:       | Đại học chính quy   | Mã hệ: 1       |
| Khóa:             | 2014                | Lớp: 14141DT2C |

### I. TÊN ĐỀ TÀI: **ỨNG DỤNG CÔNG NGHỆ IOT ĐỂ THIẾT KẾ HỆ THỐNG GIÁM SÁT VÀ ĐIỀU KHIỂN THIẾT BỊ CÔNG NGHIỆP**

#### II. NHIỆM VỤ

##### 1. Các số liệu ban đầu:

- Module ESP8266 NODE MCU
- Arduino mega 2560
- PLC S7 200
- Module RS 485

##### 2. Nội dung thực hiện:

- Thiết kế giao diện điều khiển và giám sát trên Websever
- Thiết kế giao tiếp giữa Websever và Arduino, Arduino với PLC
- Thi công mạch và mô hình
- Viết báo cáo

III. NGÀY GIAO NHIỆM VỤ: 19/03/2018

IV. NGÀY HOÀN THÀNH NHIỆM VỤ: 04/07/2018

V. HỌ VÀ TÊN CÁN BỘ HƯỚNG DẪN: ThS.Nguyễn Thanh Giàu

CÁN BỘ HƯỚNG DẪN

BM. ĐIỆN TỬ CÔNG NGHIỆP – Y SINH

## LỊCH TRÌNH THỰC HIỆN ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên: NGUYỄN VĂN LINH      MSSV: 14141169

Lớp: 14141DT2C

Tên đề tài: *Ứng dụng công nghệ IoT để thiết kế hệ thống điều khiển và giám sát thiết bị công nghiệp*

| Tuần/ngày               | Nội dung   | Xác nhận GVHD |
|-------------------------|--|---------------|
| Tuần 1<br>(19/3- 25/3)  | - Gặp GVHD để nghe phỏ biến yêu cầu làm đồ án, tiến hành chọn đồ án.<br>- GVHD tiến hành xét duyệt đề tài.   |               |
| Tuần 2<br>(26/3 – 1/4)  | - Viết đề cương<br>- Viết lịch trình làm đề tài.   |               |
| Tuần 3<br>(2/4 – 8/4)   | -Tìm hiểu cơ sở lý thuyết liên quan với đề tài:<br>Arduino Mega 2560 R3, LCD 20x4, Module RS-485, Node MCU, PLC s7-200, các giao thức giao tiếp, cách thiết kế web |               |
| Tuần 4<br>(9/4 – 15/4)  | -Tìm hiểu các về chuẩn giao tiếp truyền thông Modbus RTU, UART của Arduino   |               |
| Tuần 5<br>(16/4 – 22/4) | - Thực hiện giao tiếp giữa các module lại với nhau, giữa Arduino và PLC<br>- Lập trình với một số chương trình đơn giản  |               |
| Tuần 6<br>(23/4 – 29/4) | - Thiết kế một trang Website đơn giản<br>- Thực hiện giao tiếp giữa truyền nhận dữ liệu giữa NodeMCU với Website   |               |
| Tuần 7<br>(30/4 – 6/5)  | -Thực hiện kết nối hai khối lại với nhau để có thể truyền nhận dữ liệu từ Website xuống PLC và ngược lại   |               |
| Tuần 8<br>(7/5 – 13/5)  | -Tiến hành lập trình cho toàn hệ thống, Code cho Arduino, code NodeMCU, code cho PLC hoạt động   |               |
| Tuần 9                  | -Thiết kế sơ đồ nguyên lý mạch điều khiển,   |               |

|                            |  |  |
|----------------------------|--|--|
| (14/5 – 20/5)              | mạch nguồn<br>- Vẽ và thi công mạch PCB  |  |
| Tuần 10,11<br>(21/5 – 3/6) | - Lắp ráp mạch, kiểm tra hoạt động của hệ thống<br>- Chính sửa code cho hệ thống hoạt động |  |
| Tuần 12,13<br>(4/6-17/6)   | - Viết báo cáo những nội dung đã thực hiện   |  |
| Tuần 14<br>(18/6-24/6)     | -Hoàn thiện báo cáo gửi cho GVHD nhận xét và chỉnh sửa                                     |  |
| Tuần 15<br>(25/6-1/7)      | Làm slide (6-10 slide), báo cáo với GVHD.  |  |

GV HƯỚNG DẪN  
(Ký và ghi rõ họ và tên)

## **LỜI CAM ĐOAN**

Đề tài này là do tôi tự thực hiện dựa vào một số tài liệu trước đó và không sao chép từ tài liệu hay công trình đã có trước đó.

Người thực hiện đề tài

**Nguyễn Văn Linh**

## **LỜI CẢM ƠN**

Em xin gửi lời cảm ơn sâu sắc đến Thầy Nguyễn Thanh Giàu - Giảng viên bộ môn Điện tử công nghiệp, đã trực tiếp hướng dẫn và tận tình giúp đỡ tạo điều kiện để hoàn thành tốt đề tài.

Em chân thành cảm ơn quý Thầy Cô, Giảng viên bộ môn Điện tử công nghiệp – y sinh đã góp ý và chia sẻ nhiều kinh nghiệm quý báu cho em thực hiện tốt đề tài.

Em xin gửi lời chân thành cảm ơn các thầy cô trong Khoa Điện-Điện Tử đã tạo những điều kiện tốt nhất cho em hoàn thành đề tài.

Em cũng gửi lời đồng cảm ơn đến các bạn lớp 14141DT2C đã chia sẻ trao đổi kiến thức cũng như những kinh nghiệm quý báu trong thời gian thực hiện đề tài.

Cảm ơn đến gia đình đã luôn bên cạnh và ủng hộ tinh thần.

Xin chân thành cảm ơn!

Người thực hiện đề tài

**Nguyễn Văn Linh**

# MỤC LỤC

| Nội dung  | Trang    |
|---|----------|
| Trang bìa.....  | i        |
| Nhiệm vụ đồ án .....  | ii       |
| Lịch trình .....  | iii      |
| Lời cam đoan .....  | v        |
| Lời Cảm ơn .....  | vi       |
| Mục lục .....   | vii      |
| Liệt kê hình vẽ .....   | ix       |
| Liệt kê bảng .....  | xii      |
| Tóm tắt .....   | xiii     |
| <b>Chương 1. TỔNG QUAN .....</b>  | <b>1</b> |
| 1.1 ĐẶT VẤN ĐỀ.....   | 1        |
| 1.2 MỤC TIÊU.....   | 2        |
| 1.3 NỘI DUNG NGHIÊN CỨU .....   | 2        |
| 1.4 GIỚI HẠN.....   | 2        |
| 1.5 BỐ CỤC .....  | 3        |
| <b>Chương 2. CƠ SỞ LÝ THUYẾT .....</b>                                    | <b>4</b> |
| 2.1 TỔNG QUAN VỀ IOT TRONG CÔNG NGHIỆP .....                              | 4        |
| 2.1.1 Khái niệm về IoT .....  | 4        |
| 2.1.2 IoT trong công nghiệp .....   | 4        |
| 2.1.3 Lý do sử dụng IoT .....   | 5        |
| 2.2 TỔNG QUAN WEBSEVER.....   | 6        |
| 2.2.1 Web Server.....   | 6        |
| 2.2.2 Ngôn ngữ lập trình PHP .....  | 6        |
| 2.2.3 Giới thiệu về MySQL.....  | 7        |
| 2.2.4 Giới thiệu về mạng không dây (Wifi) .....                           | 7        |
| 2.3 GIỚI THIỆU VỀ PHẦN CỨNG .....   | 8        |
| 2.3.1 Board xử lý trung tâm .....   | 8        |
| 2.3.2 Giới thiệu về thiết bị giao tiếp internet (ESP 8266 Node MCU) ..... | 10       |
| 2.3.3 Giới thiệu về thiết bị hiển thị (LCD).....                          | 11       |
| 2.3.4 Giới thiệu về thiết bị công nghiệp (PLC S7 200) .....               | 11       |
| 2.4 GIỚI THIỆU PHẦM MỀM VÀ GIAO THỨC .....                                | 15       |
| 2.4.1 Giao thức Modbus RTU.....   | 15       |
| 2.4.2 Giao thức UART .....  | 18       |

|   |           |
|---|-----------|
| <b>Chương 3. TÍNH TOÁN VÀ THIẾT KẾ.....</b>               | <b>19</b> |
| 3.1 GIỚI THIỆU .....                                      | 19        |
| 3.2 TÍNH TOÁN VÀ THIẾT KẾ HỆ THỐNG .....                  | 19        |
| 3.2.1 Thiết kế sơ đồ khái niệm .....                      | 19        |
| 3.2.2 Tính toán và thiết kế mạch .....                    | 21        |
| 3.2.3 Tính toán và thiết kế Web Server .....              | 31        |
| 3.2.4 Sơ đồ nguyên lý toàn mạch .....                     | 34        |
| <b>Chương 4. THI CÔNG HỆ THỐNG.....</b>                   | <b>35</b> |
| 4.1 GIỚI THIỆU .....                                      | 35        |
| 4.2 THI CÔNG HỆ THỐNG .....                               | 35        |
| 4.2.1 Thi công bo mạch .....                              | 35        |
| 4.2.2 Lắp ráp và kiểm tra .....                           | 39        |
| 4.3 ĐÓNG GÓI VÀ THI CÔNG MÔ HÌNH .....                    | 40        |
| 4.3.1 Đóng gói bộ điều khiển .....                        | 40        |
| 4.3.2 Thi công mô hình .....                              | 41        |
| 4.4 LẬP TRÌNH HỆ THỐNG .....                              | 43        |
| 4.4.1 Lưu đồ giải thuật .....                             | 43        |
| 4.4.3 Phần mềm lập trình cho Web Server .....             | 48        |
| 4.4.3 Phần mềm lập trình cho PLC (Step7- Micro/Win). .... | 50        |
| 4.5 LẬP TRÌNH MÔ PHỎNG .....                              | 51        |
| 4.6 VIẾT TÀI LIỆU HƯỚNG DẪN SỬ DỤNG, THAO TÁC .....       | 55        |
| 4.6.1 Viết tài liệu hướng dẫn sử dụng .....               | 55        |
| 4.6.2 Quy trình thao tác .....                            | 56        |
| <b>Chương 5. KẾT QUẢ - NHẬN XÉT - ĐÁNH GIÁ .....</b>      | <b>58</b> |
| 5.1 GIỚI THIỆU .....                                      | 58        |
| 5.2 KẾT QUẢ ĐẠT ĐƯỢC .....                                | 58        |
| 5.3 KẾT QUẢ THỰC NGHIỆM .....                             | 58        |
| 5.4 NHẬN XÉT – ĐÁNH GIÁ .....                             | 67        |
| 5.4.1 Nhận xét .....                                      | 67        |
| 5.4.2 Đánh giá .....                                      | 67        |
| <b>Chương 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....</b>        | <b>69</b> |
| 6.1 KẾT LUẬN .....  | 69        |
| 6.2 HƯỚNG PHÁT TRIỂN .....                                | 70        |
| <b>TAI LIỆU THAM KHẢO .....</b>                           | <b>71</b> |
| <b>TỪ VIẾT TẮT .....</b>                                  | <b>71</b> |
| <b>PHỤ LỤC .....</b>                                      | <b>73</b> |

## LIỆT KÊ HÌNH VẼ

|   |    |
|---|----|
| <b>Hình 2.1</b> Ứng dụng IoT kết nối các cảm biến nhiệt độ .....                    | 5  |
| <b>Hình 2.2</b> Các chuẩn Wifi thông dụng .....                                     | 8  |
| <b>Hình 2.3</b> Board Arduino Mega 2560 .....                                       | 9  |
| <b>Hình 2.4</b> Hình ảnh thực tế của ESP8266 node MCU .....                         | 10 |
| <b>Hình 2.5</b> Một PLC s7-200 của hãng siemens .....                               | 12 |
| <b>Hình 2.6</b> Cổng truyền thông RS-485 .....                                      | 13 |
| <b>Hình 2.7</b> Board mạch CPU 224XP .....  | 14 |
| <b>Hình 2.8</b> Phần mềm Step7- Micro/Win .....                                     | 14 |
| <b>Hình 2.9</b> Một mô hình sử dụng giao thức truyền theo chuẩn Modbus TCP IP ..... | 15 |
| <b>Hình 2.10</b> Cấu trúc khung dữ liệu Modbus RTU .....                            | 16 |
| <b>Hình 2.11</b> Quá trình yêu cầu và phản hồi giữa Master và Slave .....           | 17 |
| <b>Hình 2.12</b> Ví dụ một quá trình yêu cầu và phản hồi giữa Master và Slave ..... | 18 |
| <b>Hình 2.13</b> Truyền dữ liệu qua lại giữa vi điều khiển và node MCU .....        | 18 |
| <b>Hình 3.1</b> Sơ đồ khái niệm của hệ thống .....                                  | 20 |
| <b>Hình 3.2</b> Sơ đồ nguyên lý của khái niệm trung tâm .....                       | 22 |
| <b>Hình 3.3</b> Sơ đồ kết nối của Node MCU với Arduino .....                        | 23 |
| <b>Hình 3.4</b> Sơ đồ nguyên lý khái niệm thị .....                                 | 23 |
| <b>Hình 3.5</b> Mức điện áp của tín hiệu RS-485 và UART .....                       | 24 |
| <b>Hình 3.6</b> Sơ đồ nguyên lý mạch RS-485 .....                                   | 25 |
| <b>Hình 3.7</b> Sơ đồ mạch UART TTL-RS485 .....                                     | 25 |
| <b>Hình 3.8</b> Sơ đồ kết nối của module RS485 với PLC và Arduino .....             | 26 |
| <b>Hình 3.9</b> Sơ đồ kết nối PLC và các thiết bị .....                             | 27 |
| <b>Hình 3.10</b> Module cảm biến Encoder .....                                      | 27 |
| <b>Hình 3.11</b> Động cơ giảm tốc DC .....  | 28 |
| <b>Hình 3.12</b> Dòng chuyển tiếp của PC817 .....                                   | 28 |
| <b>Hình 3.13</b> Sơ đồ nguyên lý của mạch điều khiển cảm biến .....                 | 29 |
| <b>Hình 3.14</b> Bộ chuyển đổi PT100 4-20mA .....                                   | 30 |
| <b>Hình 3.15</b> Sơ đồ kết nối PT100 với bộ chuyển đổi .....                        | 30 |
| <b>Hình 3.16</b> Sơ đồ nguyên lý khái niệm nguồn .....                              | 31 |
| <b>Hình 3.17</b> Giao diện quản lý file lập trình .....                             | 33 |
| <b>Hình 3.18</b> Database của trang web .....                                       | 33 |

|   |    |
|---|----|
| <b>Hình 3.19 Giao diện web điều khiển .....</b>                                       | 34 |
| <b>Hình 4.1 Mặt dưới của mạch điều khiển chính .....</b>                              | 35 |
| <b>Hình 4.2 Sơ đồ bố trí linh kiện của mạch điều khiển chính .....</b>                | 36 |
| <b>Hình 4.3 Mặt dưới của mạch điều khiển cảm biến.....</b>                            | 36 |
| <b>Hình 4.4 Sơ đồ bố trí linh kiện của mạch chứa cảm biến .....</b>                   | 37 |
| <b>Hình 4.5 Sơ đồ bố trí linh kiện của mạch nguồn 5V .....</b>                        | 39 |
| <b>Hình 4.6 Thi công mạch điều khiển cảm biến .....</b>                               | 39 |
| <b>Hình 4.7 Sơ đồ toàn mạch điều khiển.....</b>                                       | 40 |
| <b>Hình 4.8 Bộ điều khiển được giữ cố định .....</b>                                  | 41 |
| <b>Hình 4.9 Hình nắp đậy cho hệ thống.....</b>  | 41 |
| <b>Hình 4.10 Cân chỉnh trước khi lắp vào .....</b>                                    | 42 |
| <b>Hình 4.11 Lắp thiết bị vào hệ thống .....</b>                                      | 42 |
| <b>Hình 4.12 Lắp ráp mô hình hoàn chỉnh .....</b>                                     | 43 |
| <b>Hình 4.13 Lưu đồ điều khiển cho Arduino .....</b>                                  | 44 |
| <b>Hình 4.14 Lưu đồ điều khiển cho Node MCU .....</b>                                 | 45 |
| <b>Hình 4.15 Bắt đầu cài đặt .....</b>  | 46 |
| <b>Hình 4.16 Kiểm tra lại driver .....</b>  | 47 |
| <b>Hình 4.17 Giao diện phần mềm Arduino IDE .....</b>                                 | 47 |
| <b>Hình 4.18 Trang chủ của web hosting .....</b>                                      | 48 |
| <b>Hình 4.19 Chọn dự án để sử dụng hành xử lý .....</b>                               | 49 |
| <b>Hình 4.20 Trang lập trình trực tiếp.....</b>                                       | 49 |
| <b>Hình 4.21 Có thể chỉnh sửa trực tiếp trên Website .....</b>                        | 50 |
| <b>Hình 4.22 Giao diện phần mềm Step7-Micro/Win .....</b>                             | 51 |
| <b>Hình 4.23 Khởi tạo Modbus cho PLC .....</b>  | 52 |
| <b>Hình 4.24 Thực hiện lệnh đọc dữ lệnh từ PLC.....</b>                               | 52 |
| <b>Hình 4.25 Các giá trị ô nhớ của thanh ghi trên PLC .....</b>                       | 53 |
| <b>Hình 4.26 Kết quả trả về của lệnh đọc .....</b>                                    | 53 |
| <b>Hình 4.27 Quá trình yêu cầu và phản hồi của Master và Slave với lệnh đọc .....</b> | 54 |
| <b>Hình 4.28 Thực hiện lệnh ghi dữ lệnh vào PLC.....</b>                              | 54 |
| <b>Hình 4.29 Quá trình yêu cầu và phản hồi của Master và Slave với lệnh ghi .....</b> | 55 |
| <b>Hình 4.30 Kết quả các dữ liệu được ghi vào PLC .....</b>                           | 55 |
| <b>Hình 4.31 Sơ đồ Quy trình thao tác .....</b>                                       | 57 |
| <b>Hình 5.1 Trang chủ của trang web.....</b>  | 59 |
| <b>Hình 5.2 Trang đăng ký tài khoản sử dụng cho người dùng .....</b>                  | 59 |

|   |    |
|---|----|
| <b>Hình 5.3</b> Thông tin đăng nhập và tên người dùng được lưu trữ .....      | 60 |
| <b>Hình 5.4</b> Giao diện đăng nhập của hệ thống .....                        | 61 |
| <b>Hình 5.5</b> Giao diện chính của trang điều khiển .....                    | 61 |
| <b>Hình 5.6</b> Nhập số sản phẩm xuống hệ thống bên dưới .....                | 62 |
| <b>Hình 5.7</b> Giá trị cài đặt hiển thị ở hệ thống trung tâm .....           | 62 |
| <b>Hình 5.8</b> Giá trị hiển thị ở LCD ở sản phẩm 1 .....                     | 63 |
| <b>Hình 5.9</b> Giá trị hiển thị ở web ở sản phẩm 1 .....                     | 63 |
| <b>Hình 5.10</b> Giá trị hiển thị ở LCD ở sản phẩm 3 .....                    | 64 |
| <b>Hình 5.11</b> Giao diện hiển thị giá trị ở trang giám sát sản phẩm 3 ..... | 64 |
| <b>Hình 5.12</b> Giá trị hiển thị ở LCD ở sản phẩm cuối cùng .....            | 65 |
| <b>Hình 5.13</b> Giá trị hiển thị ở web ở sản phẩm cuối cùng .....            | 65 |
| <b>Hình 5.14</b> Giá trị cập nhật trên web bị chậm .....                      | 66 |
| <b>Hình 5.15</b> Thông tin liên hệ của trang web .....                        | 66 |

## **LIỆT KÊ BẢNG**

|   |    |
|---|----|
| <b>Bảng 2.1</b> Thông số kỹ thuật board Arduino Mega 2560 ..... | 9  |
| <b>Bảng 2.2</b> Thông số kỹ thuật của ESP8266.....              | 10 |
| <b>Bảng 4.1</b> Danh sách các linh kiện. ....                   | 38 |
| <b>Bảng 5.1</b> Thời gian đáp ứng đọc từ PLC lên Arduino.....   | 67 |
| <b>Bảng 5.2</b> Giá trị nhận được bằng thực nghiệm .....        | 68 |

## TÓM TẮT

Ngày nay, với sự phát triển của khoa học kỹ thuật đã làm cho cuộc sống con người ngày càng tốt hơn, hiện đại hơn. Trong công nghiệp cũng vậy, với sự phát triển của khoa học các thiết bị trong nhà máy được hoạt động một cách tự động và việc giám sát, điều khiển dần dần được thực hiện từ xa.

Với sự xuất hiện của công nghệ 4.0 đi đầu công nghệ IoT được phát triển mạnh mẽ. Việc ứng dụng IoT vào hệ thống giám sát và điều khiển nhà máy giúp điều khiển nhanh chóng và dễ dàng, phát hiện sự cố nhanh, giảm nhân công từ đó mang lại hiệu quả kinh tế đáng kể. Các giải pháp ứng dụng dần được hình thành và phát triển. Cho nên nhóm đề xuất nghiên cứu đề tài: “**Ứng dụng công nghệ IoT để thiết kế hệ thống giám sát và điều khiển thiết bị công nghiệp**”.

Trong đề tài này, các thiết bị trong hệ thống sẽ được điều khiển và giám sát bằng Web Sever thông qua mạng Wifi. Sử dụng PLC giao tiếp với board Arduino theo chuẩn công nghiệp thông qua module RS485, Arduino sẽ kết nối với module Wifi ESP8266 để đưa dữ liệu lên Web Server. PLC sẽ nhận lệnh điều khiển từ Web Server thông qua kết nối với Arduino và điều khiển các thiết bị. Việc sử module ESP8266 giao tiếp với Arduino giúp người dùng có thể truy cập vào Web Server bằng mạng Internet, không gian lưu trữ dữ liệu trên Web Server lớn, có thể điều khiển thiết bị mọi lúc mọi nơi thay vì sử dụng SQL.

# **Chương 1. TỔNG QUAN**

## **1.1 ĐẶT VẤN ĐỀ**

Công nghệ ngày càng phát triển, đời sống con người ngày càng được nâng cao, xu hướng các dây chuyền tự động, hệ thống thông minh ngày càng được phát triển. Thực tế cho thấy, sự phát triển mạnh mẽ của các ứng dụng **Internet of Things** là nhờ cơ sở hạ tầng ngày càng phát triển, cùng với đó là chi phí sản xuất thấp. Chính vì vậy việc điều khiển và giám sát thiết bị công nghiệp từ xa thông qua Internet đang là xu thế phát triển mới trong ngành công nghiệp tự động hóa cũng như trong các nhà máy sản xuất.

Các hệ điều khiển và giám sát các thiết bị công nghiệp từ xa cũng được hình thành từ những năm 1990. Các thiết bị kết nối với nhau được điều khiển và giám sát bởi một hệ thống gọi là SCADA (Supervisory Control And Data Acquisition). Các hệ thống SCADA kết nối với nhau qua mạng LAN (Local Area Network) điều khiển và giám sát từ xa các nhà máy. Hiện nay thì các hệ thống này có hỗ trợ một số Web Server riêng do các hãng PLC (Programmable Logic Controllers) có thể kết nối điều khiển trên mạng Internet như hãng Siemens. Một số lợi ích của hệ thống đó là nâng cao năng suất, cải thiện chất lượng sản phẩm, giảm chi phí vận hành và bảo trì, giảm chi phí nhân lực...Nhưng bên cạnh đó thì hệ thống SCADA cũng có những hạn chế về cơ chế điều khiển và lập trình do giao diện mặc định của hãng.

Từ đó phát triển có một hệ thống mới có thể kế thừa những lợi ích và khắc phục những hạn chế của hệ thống SCADA. Do đó cần đó một thiết bị mà trên đó ta có thể lập trình mở và thiết kế giao diện theo ý muốn mà không bị hạn chế.

Từ thực tế trên cùng với lượng kiến thức quý báu được học tập ở trường và lòng đam mê, nhóm chúng em quyết định thực hiện nghiên cứu đồ án:

*“Ứng dụng công nghệ IoT để thiết kế hệ thống điều khiển và giám sát thiết bị công nghiệp”.*

## **CHƯƠNG 1. TỔNG QUAN**

---

### **1.2 MỤC TIÊU**

Nhóm thực hiện đồ án “*Ứng dụng công nghệ IoT để thiết kế hệ thống điều khiển và giám sát thiết bị công nghiệp*” với những mục tiêu cụ thể như sau:

- Thực hiện giao tiếp giữa Arduino và PLC.
- Thiết kế giao diện điều khiển và giám sát các qua Web Server.
- Đưa dữ liệu từ Web Server xuống điều khiển PLC và lấy dữ liệu từ PLC lên Web Server.
- Đánh giá những kết quả đạt được và những mặt hạn chế của đề tài.
- Đề xuất phương pháp giải quyết những vấn đề còn tồn tại và hướng phát triển đề tài.

### **1.3 NỘI DUNG NGHIÊN CỨU**

Với ý tưởng trên nhóm đã tiến hành đề tài nghiên cứu tập trung những vấn đề sau đây:

- Tìm hiểu về truyền thông Modbus RTU.
- Tìm hiểu cách trao đổi dữ liệu giữa Website và Arduino thông qua Wifi
- Truyền thông dữ liệu giữa PLC với Arduino.
- Thiết kế Web Server điều khiển và giám sát trạng thái thiết bị.
- Tìm hiểu tính thực thi của hệ thống.

### **1.4 GIỚI HẠN**

Vì một số yếu tố khách quan (điều kiện tài chính...) cũng như yếu tố chủ quan (hạn chế về kiến thức chuyên môn...) mà nội dung đề tài chỉ thực hiện trong phạm vi sau đây:

- Hệ thống chỉ sử dụng Wifi để điều khiển thay vào đó thì trên thực tế có rất nhiều loại sóng có thể điều khiển và giám sát như: sóng RF, Bluetooth...
- Sử dụng chuẩn modbus để truyền dữ liệu giữa PLC và Arduino thông qua module RS-485.
- Giao diện Web Server tự thiết kế, đơn giản dễ sử dụng.

### 1.5 BỘ CỤC

Nội dung chính của đề tài được trình bày như sau thành các chương như sau:

- **Chương 1. Tổng quan:** Trình bày tổng quan về tình hình nghiên cứu, mục tiêu, đối tượng và phạm vi nghiên cứu của đề tài.
- **Chương 2. Cơ sở lý thuyết:** Ở chương này nhóm trình bày cơ sở lý thuyết về các vấn đề liên quan của hệ thống. Giới thiệu các thiết bị phần cứng và giao thức truyền thông giữa các thiết bị trong hệ thống.
- **Chương 3. Tính toán và thiết kế:** Chương này sẽ thiết kế sơ đồ khái của hệ thống và chi tiết từng khái. Từ đó tính toán và lựa chọn linh kiện thích hợp để xây dựng sơ đồ nguyên lý của toàn hệ thống.
- **Chương 4. Thi công hệ thống:** Trình bày thiết kế phần cứng và vị trí sắp xếp các linh kiện của hệ thống. Đưa ra lưu đồ giải thuật, thiết kế giao diện giám sát trên website, quá trình điều khiển, giám sát và hoạt động của hệ thống.
- **Chương 5. Kết quả-nhận xét-dánh giá:** Những kết quả đạt được sau thời gian thực hiện, kết quả thực nghiệm, từ đó đưa ra đánh giá nhận xét.
- **Chương 6. Kết luận và hướng phát triển của đề tài:** Tóm tắt nội dung đề tài và kết luận những việc đã làm được, hạn chế. Từ đó rút ra những nhận xét về khả năng ứng dụng trong thực tế và hướng phát triển của đề tài. Đặt nền móng cho những đề tài nghiên cứu sau được hoàn thiện và tiến bộ hơn về mặt kết quả thực nghiệm.

## **Chương 2. CƠ SỞ LÝ THUYẾT**

### **2.1 TỔNG QUAN VỀ IOT TRONG CÔNG NGHIỆP**

#### **2.1.1 Khái niệm về IoT**

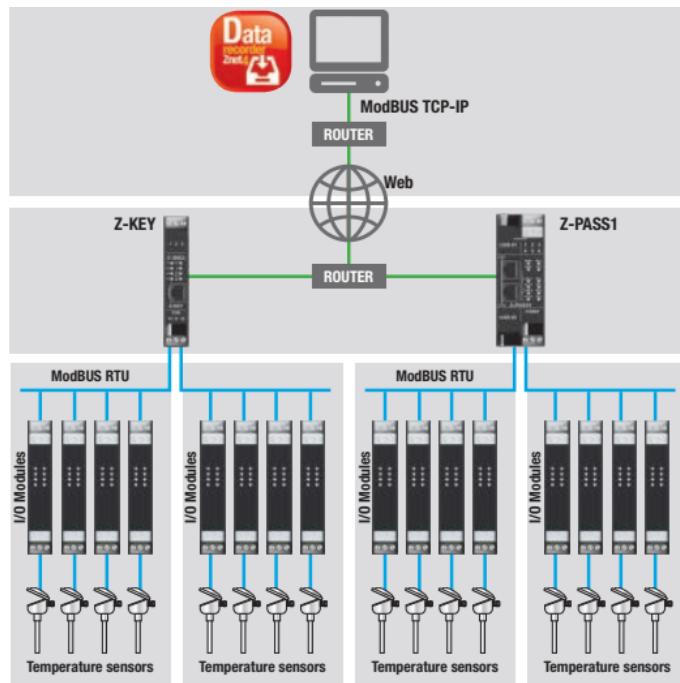
Internet of Things (IoT) mạng lưới vật kết nối Internet hoặc là mạng lưới thiết bị kết nối Internet, khi mà mỗi đồ vật, con người được cung cấp một định danh của riêng mình và tất cả có khả năng truyền tải, trao đổi thông tin, dữ liệu qua một mạng duy nhất mà không cần đến sự tương tác trực tiếp giữa người với người, hay người với máy tính. IoT đã phát triển từ sự hội tụ của công nghệ không dây, công nghệ vi cơ điện tử và Internet. Nói đơn giản là một tập hợp các thiết bị có khả năng kết nối với nhau với Internet và với thế giới bên ngoài để thực hiện một công việc nào đó.

#### **2.1.2 IoT trong công nghiệp**

IoT thực sự là cuộc cách mạng về công nghệ thông tin của thế giới hiện đại. Khi mỗi thiết bị, và con người được cấp một mã định danh riêng, tất cả có khả năng truyền tải, trao đổi thông tin, dữ liệu thông qua mạng Internet mà không cần sự tương tác giữa con người với con người, hoặc giữa người với máy tính.

Kết nối các thiết bị công nghiệp và điều khiển thông qua Internet là một vấn đề cực kỳ hấp dẫn với những người làm trong ngành kỹ thuật. Các nhà máy tại Việt Nam chúng ta hiện nay các thiết bị điều khiển hầu hết được kết nối với PLC, DCS hoặc SCADA điều khiển tự động hoạt bát tự động. Nhưng khi áp dụng IoT vào trong nhà máy việc quản lý các hệ thống này được thông qua Internet. Người quản lý không cần đến nhà máy cũng biết được các thông số của máy móc hoạt động ra sao. Và hơn hết chúng ta có thể điều khiển các thiết bị được kết nối ở bất kỳ nơi nào trên thế giới thông qua Internet.

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT



**Hình 2.1** *Ứng dụng IoT kết nối các cảm biến nhiệt độ*

Như vậy thông qua việc kết nối các thiết bị trong nhà máy, chúng ta sẽ biết được máy móc vận hành ra sao, điều khiển thiết bị từ xa, kiểm soát mức nhiên liệu có trong bồn chứa, các nguyên vật liệu trong từng silo...

Trong công nghiệp cực kỳ quan trọng nó không chỉ cắt giảm lao động mà nó có thể thay thế con người làm những ngành công nghiệp nguy hiểm thông qua việc giám sát và điều khiển trên Web Server.

### 2.1.3 Lý do sử dụng IoT

Ở đề tài chúng em sử dụng IoT vì nó hiện nay đang rất phổ biến và phù hợp với ứng dụng thực tế của chúng em, IoT có thể cho chúng em có thể kết nối các thiết bị với nhau với việc sử dụng Internet, vì có thể nói Internet là một trong những cái được sử dụng rộng rãi nhất hiện nay.

Ứng dụng công nghệ IoT vào đề tài này cũng làm góp phần xây dựng công nghệ 4.0 đang phát triển trên thế giới trở nên phong phú hơn.

Với những hiệu quả thông minh rất thiết thực mà IoT đem đến cho con người, IoT đã và đang được tích hợp trên khắp mọi thứ, mọi nơi xung quanh thế giới mà con người đang sống. Từ chiếc vòng đeo tay, những đồ gia dụng trong nhà, những mảnh vườn đang ươm hạt giống, cho đến những sinh vật sống như động vật hay con người...đều có sử dụng giải pháp IoT. Vì thế đó chính là lý do mà chúng em sử dụng IoT vào đề tài.

### 2.2 TỔNG QUAN WEBSEVER

#### 2.2.1 Web Server

Web Server là một hay nhiều máy tính mà tại đó chứa đựng mã nguồn của trang web, máy tính đó còn phải được cài các chương trình phục vụ web. Chính những chương trình này sẽ thiết lập các kết nối để người trình duyệt web có thể truy cập được vào trang web (ví dụ như IIS của Microsoft).

Trong đề tài chúng em sử dụng Web Server để đóng vai trò như một máy chủ để có thể lưu giá trị, nhập giá trị xuống cho hệ thống cũng như giám sát hệ thống bên dưới gửi lên.

Trong Web Server của đề tài thì chúng em sử dụng hosting để chuyển tải dữ liệu lên xuống cũng như lưu trữ, một Database để chứa tất cả nguồn của trang Web Server như là: tên đăng nhập, mật khẩu, giá trị dây chuyền nhập xuống....

Để làm được điều đó thì cũng cần một chương trình trung gian ở Server để nhận dữ liệu từ hệ thống cũng như trích xuất dữ liệu từ Database. Chương trình ở Server nhóm đã lựa chọn PHP, nhóm đã sử dụng ngôn ngữ PHP để lập trình để tiếp nhận dữ liệu từ hệ thống IoT và trích xuất các thông số thiết lập từ cơ sở dữ liệu truyền xuống thiết bị.

#### 2.2.2 Ngôn ngữ lập trình PHP

##### a. Giới thiệu

Cùng với Apache, PHP và MySQL đã trở thành chuẩn trên các máy chủ Web. Rất nhiều phần mềm Web mạnh sử dụng PHP và MySQL (PHP Nuke, Post Nuke, vBulletin...).

PHP là ngôn ngữ có cú pháp gần giống Perl nhưng tốc độ dịch của nó được các chuyên gia đánh giá là nhanh hơn ASP 5 lần, chạy trên nhiều hệ điều hành như Unix, Window, Linux nó hỗ trợ kết nối các hệ cơ sở dữ liệu lớn như MySQL, ngoài ra nó còn được Apache hỗ trợ như là một module cơ bản.

##### b. Lý do chọn PHP

PHP là một ngôn ngữ mã nguồn mở có rất nhiều nhà phát triển tham gia vào và quan trọng hơn là dễ dàng để tìm hiểu và sử dụng.

## **CHƯƠNG 2. CƠ SỞ LÝ THUYẾT**

---

Nguồn tài liệu học PHP vô cùng phong phú và đa dạng. PHP rất dễ học và dễ tiếp cận cho những người mới học lập trình và không phải chuyên ngành chính của chúng em.

Ngôn ngữ này thì cũng không cần phải được biên dịch.

PHP là mã nguồn mở có thể thoái mái lập trình mà không tốn chi phí cho sinh viên có thể tiết kiệm chi phí trong việc thực hiện đề tài.

### **2.2.3 Giới thiệu về MySQL**

MySQL là một Database Server, là hệ thống quản lý cơ sở dữ liệu quan hệ. Trong việc lưu trữ, tìm kiếm, sắp xếp và truy vấn dữ liệu, nó tỏ ra rất nhanh và mạnh mẽ. MySQL Server điều khiển truy cập dữ liệu đa người dùng cùng một thời điểm, cung cấp khả năng truy cập dữ liệu nhanh, bảo đảm cho người sử dụng được cấp quyền truy cập dữ liệu của hệ thống. Do vậy, MySQL là đa người dùng, đa luồng. .

MySQL có thể quản lý tới hàng Terabyte dữ liệu, hàng triệu bản ghi, chạy trên nhiều môi trường khác nhau, có giao diện tương đối dễ sử dụng, có thể truy vấn dữ liệu thông qua câu lệnh SQL.

MySQL được sử dụng chung với PHP trong những trang web cần sử dụng đến cơ sở dữ liệu.

### **2.2.4 Giới thiệu về mạng không dây (Wifi)**

Hiện nay có rất nhiều mạng không dây: Wifi, bluetooth, RF, Zigbee....

Mạng không dây ở đây chúng em lựa chọn là mạng Wifi vì mạng này đang được sử dụng phổ biến hiện nay với mạng lưới được phủ sóng hầu như là tất cả các khu vực đều có.

Mạng Wifi này thì không cần có bản quyền sử dụng đối với thiết bị sử dụng nó và có tính bảo mật cao.

Mạng Wifi cho phép truyền dữ liệu liên tục không bị gián đoạn và chi phí đầu tư ban đầu của mạng không dây thường cao hơn mạng có dây, nhưng nếu tính tổng chi phí cùng tuổi thọ sử dụng thì sóng không dây đem lại hiệu quả kinh tế hơn nhiều.

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

| GIAO THỨC<br>802.11 | DÀI TẦN (GHZ)<br>FREQUENCY | TỐC ĐỘ DỮ LIỆU MỚI<br>LUÔNG (Mbs) | PHẠM VI TRONG NHÀ<br>(M) | PHẠM VI<br>NGOÀI TRỜI<br>(M) |
|---------------------|----------------------------|-----------------------------------|--------------------------|------------------------------|
| 802.11              | 2.4                        | 2                                 | 20                       | 100                          |
| 802.11a             | 5                          | 54                                | 35                       | 120                          |
| 802.11b             | 2.4                        | 11                                | 38                       | 140                          |
| 802.11g             | 2.4                        | 54                                | 38                       | 140                          |
| 802.11n             | 2.4/5                      | 72.2<br>150                       | 70<br>70                 | 250<br>250                   |
| 802.11ac            | 5                          | 866                               |                          |                              |

Hình 2.2 Các chuẩn WiFi thông dụng

### 2.3 GIỚI THIỆU VỀ PHẦN CỨNG

Để đạt được những mục tiêu đã đặt ra thì nhóm chúng em đã chọn những phần cứng để giúp chúng em làm được những nhiệm vụ đó: Vi điều khiển (atmega 2560), thiết bị công nghiệp (PLC S7 200), thiết bị giao tiếp Internet (ESP 8266 Node MCU), thiết bị giám sát tại vi điều khiển (LCD), thiết bị giao tiếp giữa vi điều khiển và PLC (RS 485).

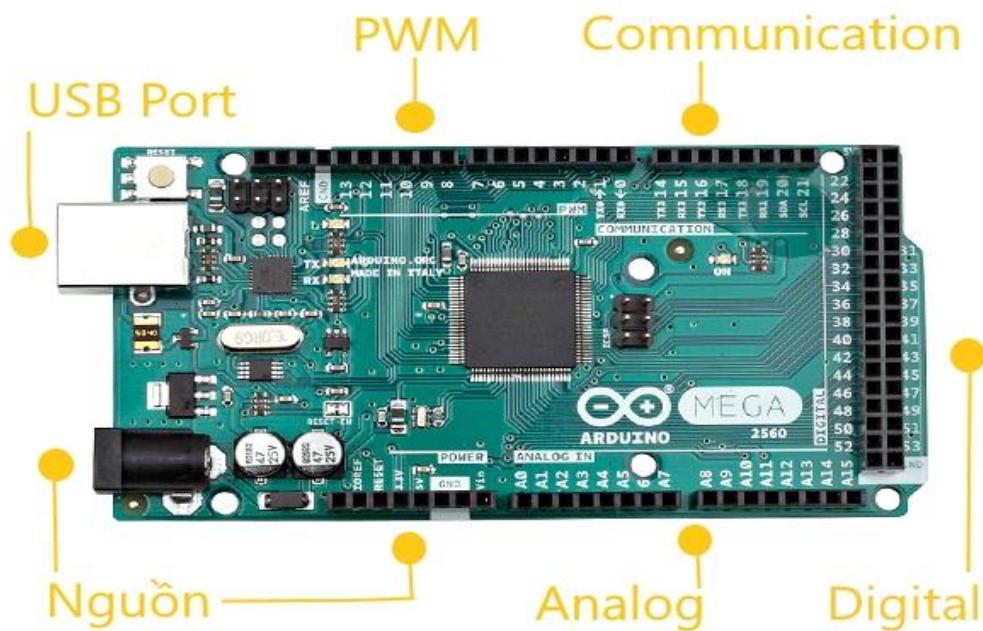
#### 2.3.1 Board xử lý trung tâm

Ở bo xử lý trung tâm chúng em chọn chip Atmega2560 có bộ nhớ flash memory 256 kB, 8kB cho bộ nhớ SRAM, 4kB cho bộ nhớ EEPROM.

Board xử lý này có nhiệm vụ quan trọng nhất của hệ thống, nó vừa nhận dữ liệu từ thiết bị giao tiếp Internet gửi về trên Web Server và phải chuyển dữ liệu nhận đó xuống thiết bị bên dưới, đồng thời cũng nhận dữ liệu liên tục từ bên dưới gửi lên và gửi ngược lại cho thiết bị giao tiếp Internet chuyển lên website hiển thị trên đó, quá trình này được thực hiện liên tục.

Ngoài việc đóng vai trò như một thiết bị trung gian thì board này còn phải xử lý và hiển thị giá trị đó ngay tại hệ thống nhờ thiết bị hiển thị (LCD).

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT



**Hình 2.3 Board Arduino Mega 2560**

Trong hình 2.3 là một Board Arduino Mega 2560 thực tế và vị trí các ngõ vào ra như hình, các thông số của Board được trình bày như bảng 2.1.

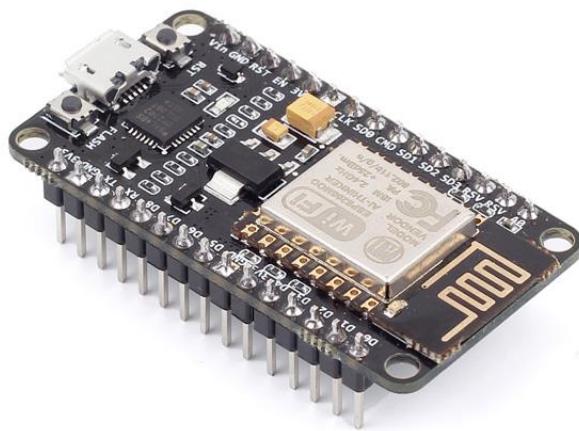
**Bảng 2.1 Thông số kỹ thuật board Arduino Mega 2560**

|                                      |            |
|--------------------------------------|------------|
| Chip xử lý                           | Atmega2560 |
| Điện áp hoạt động                    | 5V         |
| Điện áp vào (đèn nghỉ)               | 7V-15V     |
| Điện áp vào (giới hạn)               | 6V-20V     |
| Cường độ dòng điện trên mỗi 3,3V pin | 50 mA      |
| Cường độ dòng điện trên mỗi I/O pin  | 20 mA      |
| Flash Memory                         | 256 kB     |
| SRAM                                 | 8 kB       |
| EEPROM                               | 4 kB       |
| Clock speed                          | 16 MHz     |

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

### 2.3.2 Giới thiệu về thiết bị giao tiếp Internet (ESP 8266 Node MCU)

Module ESP8266 là module Wifi được đánh giá rất cao cho các ứng dụng liên quan đến Internet và Wifi cũng như các ứng dụng truyền nhận sử dụng thay thế cho các module RF khác với khoảng cách truyền lên tới 100 mét (Môi trường không có vật cản). Trên 400m với anten và router thích hợp.



**Hình 2.4** Hình ảnh thực tế của ESP8266 node MCU

**Bảng 2.2** Thông số kỹ thuật của ESP8266

|                               |   |
|-------------------------------|---|
| Chip xử lý                    | ESP8266EX   |
| Điện áp hoạt động             | 3.3V  |
| Điện áp làm việc              | 3.0V-3.6V   |
| Giao tiếp                     | SDIO 2.0, (H) SPI, UART, I2C, I2S, IR Remote Control, PWM, GPIO |
| Nhiệt độ hoạt động            | -40C ~ 125C   |
| Wi-Fi 2.4GHz, hỗ trợ WPA/WPA2 |   |
| SPI flash                     | 16 Mbs  |
| Clock speed                   | 26MHz đến 52MHz   |

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

---

### 2.3.3 Giới thiệu về thiết bị hiển thị (LCD)

Để hiển thị thông số ở ngay thiết bị trung tâm chúng em đã sử dụng LCD kích thước 20x4 để hiện thị các thông số giá trị cài đặt sản phẩm khi đưa từ trên webserver về và giá trị sản phẩm đang chạy được gửi từ phía PLC lên.

### 2.3.4 Giới thiệu về thiết bị công nghiệp (PLC S7 200)

Thiết bị công nghiệp ở đây mà nhóm chúng em lựa chọn sử dụng ở đây PLC, vì trong hầu hết thì các thiết bị công nghiệp thì PLC là cái được sử dụng nhiều nhất nó phổ biến nhất, nó đóng vai trò quan trọng trong những nhà máy, những dây chuyền sản xuất.

#### a. Giới thiệu về PLC

PLC (Programmable Logic Control) là bộ điều khiển khả trình cho phép thực hiện linh hoạt các thuật toán điều khiển logic thông qua một ngôn ngữ lập trình. Người sử dụng có thể lập trình để thực hiện một loạt trình tự các sự kiện. Các sự kiện này được kích hoạt bởi tác nhân kích thích (ngõ vào) tác động vào PLC hoặc qua các hoạt động có trễ như thời gian định thì hay các sự kiện được đếm. PLC dùng để thay thế các mạch relay (rơ le) trong thực tế. PLC hoạt động theo phương thức quét các trạng thái trên đầu ra và đầu vào. Khi có sự thay đổi ở đầu vào thì đầu ra sẽ thay đổi theo. Ngôn ngữ lập trình của PLC có thể là Ladder hay State Logic. Hiện nay có nhiều hãng sản xuất ra PLC như : Siemens, Allen-Bradley, Mitsubishi Electric, General Electric, Omron,...

Cấu trúc của một PLC gồm một đơn vị điều khiển trung tâm, bộ nhớ chương trình RAM có thể gắn thêm bộ nhớ ERROM, một bộ vi xử lí có cổng giao tiếp có thể kết nối các PLC lại với nhau và các module vào/ ra. Mỗi PLC sẽ có một phần mềm lập trình khác nhau được kết nối với PLC qua cổng RS-232, RS-485, Ethernet,...

#### b. Giới thiệu về PLC SIMATIC S7-200

Trong các hãng có rất nhiều loại PLC hỗ trợ giao tiếp chuẩn Modbus RTU. Diễn hình là hãng Siemens có các loại PLC SIMATIC như s7- 200, s7-300, s7-1200, s7-1500,... Nhóm sẽ giới thiệu PLC SIMATIC s7-200 của hãng Siemens là một PLC loại nhỏ với số lượng đầu vào/ra ít, bộ nhớ chương trình và dữ liệu nhỏ. Tuy nhiên nó lại tích hợp sẵn các tính năng phong phú.

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

PLC S7-200 có từ 6 đầu vào/4 đầu ra số (CPU 221) đến 24 đầu vào/16 đầu ra số (CPU 226). Có thể mở rộng đầu vào bằng các module mở rộng, ngoài còn có các đầu vào analog. Nguồn cấp cho S7-200 thường là 24VDC rất thích hợp cho việc kết nối các cảm biến khác và PLC cũng có đầu vào cấp nguồn 24 VDC.



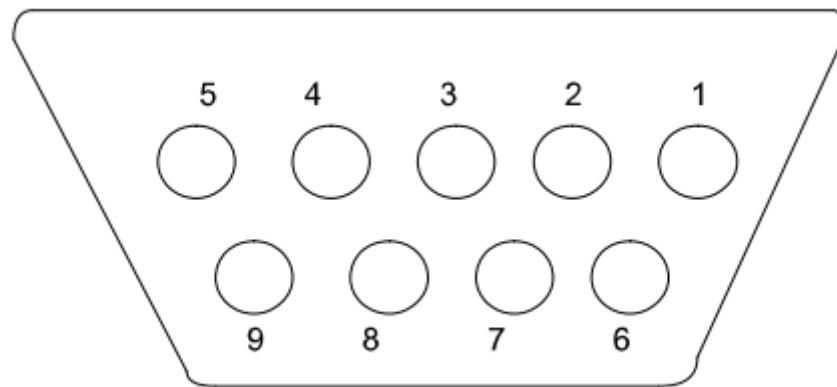
**Hình 2.5** Một PLC s7-200 của hãng siemens

Trong đề tài nhóm chọn một board mạch có chức năng tương tự với PLC S7-200 sử dụng CPU 224XP như hình 2.7 với các thông số sau:

- Nguồn cấp vào 24VDC.
- Có 8 ngõ vào và 6 ngõ ra số trên board, 2 ngõ vào analog.
- Bộ nhớ không mất dữ liệu trong khoảng thời gian 100 giờ khi PLC mất nguồn nuôi.
- Có 2 loại timer: TON và TONR.
- Có 256 bộ đếm Counter.
- Có 2 cổng truyền thông nối tiếp RS-485 với phích cắm 9 chân để thực hiện giao tiếp với các PLC khác.

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

---



**Hình 2.6** Cổng truyền thông RS-485

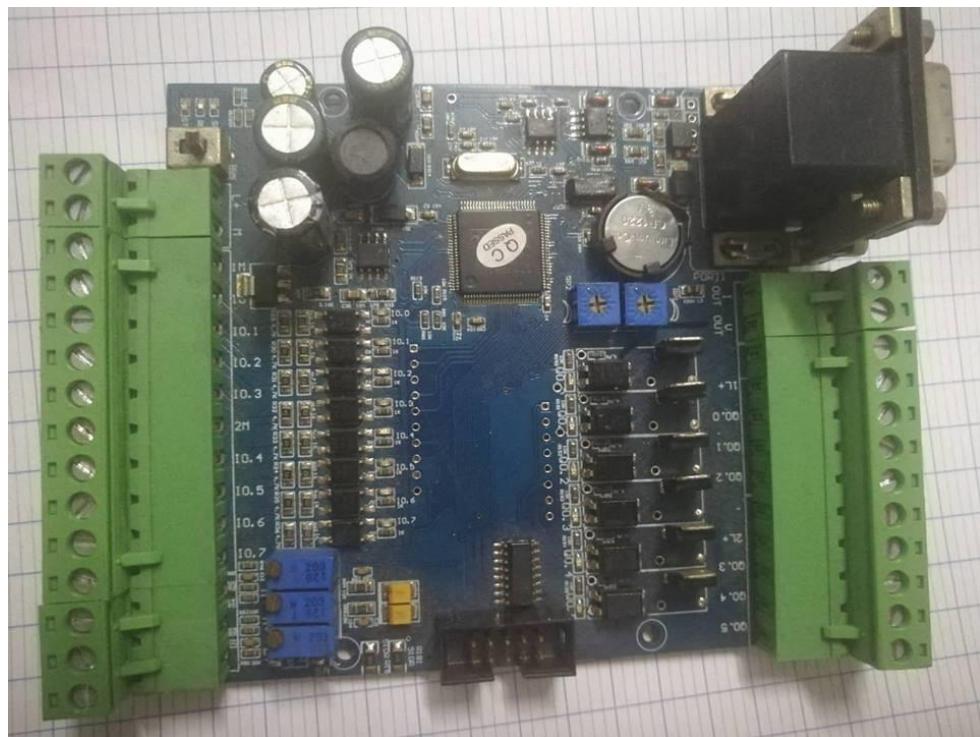
Theo hình 2.6 thì các chân có chức năng:

- Chân 1, 5: nối đất.
- Chân 2: nguồn 24 VDC hồi tiếp.
- Chân 3, 8: nhận truyền dữ liệu.
- Chân 4, 9: không sử dụng.
- Chân 6: nguồn 5VDC.
- Chân 7: nguồn 24VDC.

Bộ nhớ của board chia làm 4 vùng:

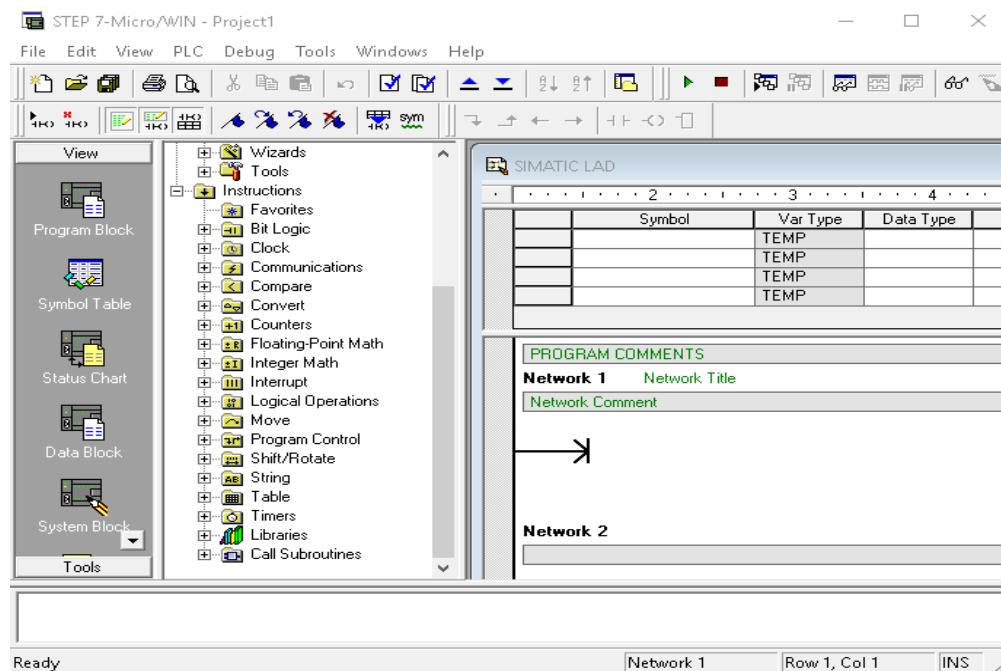
- Vùng nhớ chương trình: là miền nhớ được sử dụng để lưu trữ các lệnh chương trình.
- Vùng tham số: là miền lưu trữ các tham số như: từ khóa, địa chỉ trạm...
- Vùng dữ liệu: là miền nhớ động dùng để lưu giữ các dữ liệu của chương trình, gồm kết quả các phép tính, hằng số được định nghĩa...
- Vùng nhớ đối tượng: được sử dụng để lưu trữ cho các đối tượng lập trình như các giá trị tức thời, giá trị đặt trước của bộ đếm (Counter), bộ định thời (Timer).

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT



**Hình 2.7 Board mạch CPU 224XP**

Cũng giống như S7- 200 board sử dụng phần mềm Step7 – Micro/Win để lập trình và nạp chương trình qua cáp USB-PPI.



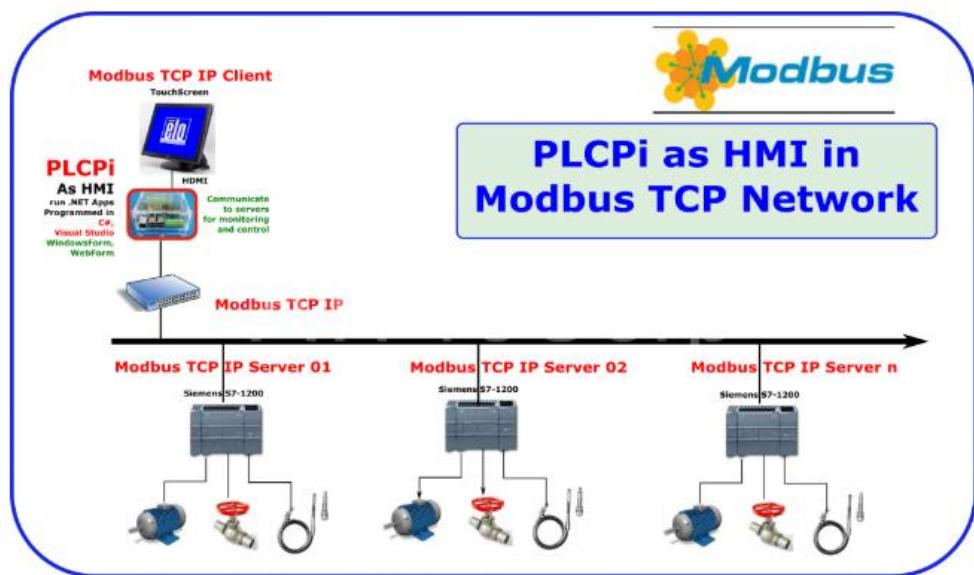
**Hình 2.8 Phần mềm Step7- Micro/Win**

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

### 2.4 GIỚI THIỆU PHẦM MỀM VÀ GIAO THỨC

#### 2.4.1 Giao thức Modbus RTU

Mạng truyền thông công nghiệp hay mạng công nghiệp (MCN) là một khái niệm chung chỉ các hệ thống mạng truyền thông số, truyền bit nối tiếp, được sử dụng để ghép nối các thiết bị công nghiệp. Trong mạng công nghiệp có thể sử dụng nhiều giao thức truyền thông khác nhau như: Modbus, Ethernet, Profibus, Profinet... Trong các giao thức lại chia thành những chuẩn khác nhau nữa. Ví dụ trong giao thức Modbus có các chuẩn giao thức như là Modbus RTU, Modbus ASCII, Modbus TCP.

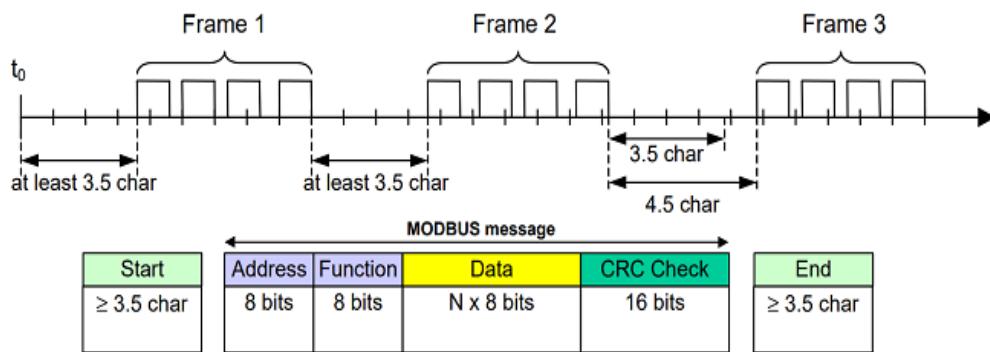


**Hình 2.9** Một mô hình sử dụng giao thức truyền theo chuẩn Modbus TCP IP

Hình 2.9 là một mô hình sử dụng giao thức Modbus TCP IP giữa một Master và n Slave để điều khiển các thiết bị cơ cấu chấp hành như động cơ, van, cảm biến... Tương tự trong đề tài nhóm sử dụng một giao thức là Modbus RTU (thiết bị đầu cuối từ xa) để truyền dữ liệu giữa PLC và Arduino. Giao thức Modbus RTU là một giao thức mở, sử dụng đường truyền vật lí là RS-232 hoặc RS-485 và sử dụng mô hình dạng Master-Slave (chủ - tớ). Trong giao thức Modbus thì chỉ có 1 Master và có thể có nhiều Slave khác nhau. Những ưu điểm của Modbus RTU là truyền ổn định, kết nối đơn giản, dễ sử dụng.

Khi truyền dữ liệu theo chuẩn modbus RTU dữ liệu sẽ gửi theo dạng một khung và có cấu trúc như sau:

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT



Hình 2.10 Cấu trúc khung dữ liệu Modbus RTU

Trong hình 2.10 miêu tả một bản tin được đặt vào khung dữ liệu có bit bắt đầu và bit kết thúc. Thời gian bắt đầu có một khoảng thời gian chờ phải lớn hơn hoặc bằng 3,5 thời gian truyền 1 kí tự, khoảng thời gian chờ giữa các khung ít nhất 3,5 thời gian truyền một kí tự. Trong một khung thì dữ liệu phải truyền với một chuỗi kí tự liên tục. Nếu khoảng thời gian giữa các kí tự lớn hơn 1,5 thời gian truyền một kí tự thì khung dữ liệu sẽ không được khai báo đầy đủ và bên nhận sẽ loại bỏ.

Một bản tin sẽ được bắt đầu bằng địa chỉ của Slave 8 bit, các thiết bị Slave riêng lẻ được gán địa chỉ trong phạm vi từ 1...247. Vị trí tiếp theo là mã lệnh hàm nó sẽ cho Slave biết địa chỉ nào thực hiện chức năng nào. Các mã hàm hỗ trợ trong Modbus có chức năng:

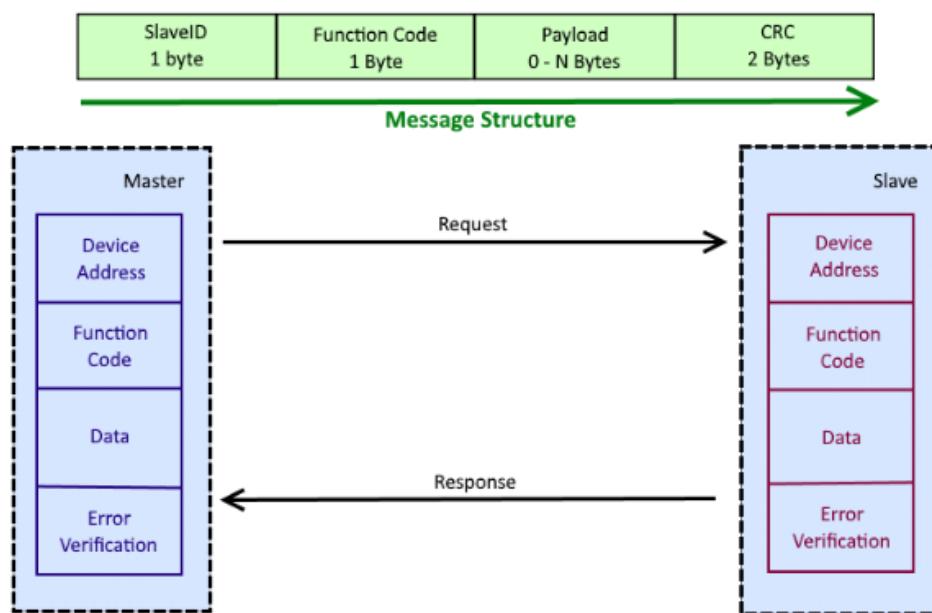
- 01 READ COIL STATUS: đọc giá trị cuộn dây trên thiết bị được điều khiển (Slave).
- 02 READ INPUT STATUS: đọc trạng thái tín hiệu ngõ vào rò rỉ rác của Slave.
- 03 READ HOLDING REGISTERS: đọc dữ liệu được lưu trữ trong thanh ghi của Slave.
- 04 READ INPUT REGISTERS: đọc dữ liệu đầu vào thanh ghi trong bộ nhớ Slave.
- 05 WRITE SINGLE COIL : có chức năng ghi dữ liệu đơn vào một cuộn dây
- 06 WRITE SINGLE REGISTERS: có chức năng ghi 1 giá trị vào 1 ô nhớ của một thanh ghi.
- 15 WRITE MULTIPLE COILS: có chức năng ghi một chuỗi dữ liệu vào các bộ nhớ.

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

- 16 WRITE MULTIPLE REGISTERS: có chức năng ghi một chuỗi dữ liệu vào các thanh ghi.

Ở vị trí tiếp theo là ô chứa dữ liệu yêu cầu hay dữ liệu cần gửi dạng N byte. Cuối bản tin là vị trí kiểm tra lỗi CRC 16 bit gồm 8 bit thấp nối trước và tiếp theo là 8 bit cao. Giá trị của CRC được tính trước khi gửi và nối vào khung, khi nhận sẽ tính toán lại và đem so sánh hai số nếu bằng nhau thì dữ liệu nhận là đúng còn nếu không bằng nhau thì dữ liệu là sai và yêu cầu gửi lại.

Quá trình nhận dữ liệu của Master và Slave được diễn ra theo quá trình như bảng sau:



**Hình 2.11** Quá trình yêu cầu và phản hồi giữa Master và Slave

Bên Master sẽ có một khung dữ liệu để gửi yêu cầu cho Slave có cấu trúc gồm địa chỉ Slave, mã hàm chức năng, dữ liệu cần yêu cầu và kiểm tra lỗi. Khi nhận được yêu cầu từ Master thì Slave sẽ phản hồi lại một khung dữ liệu có cấu trúc như bên Master và chứa dữ liệu yêu cầu.

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

Master Query, ASCII and RTU Framing

| Request             |     |                  |                 |
|---------------------|-----|------------------|-----------------|
| Field Name          | Hex | ASCII Characters | RTU 8-Bit Field |
| Header              |     | :                | (colon)         |
| Slave Address       | 06  | 0 6              | 0000 0110       |
| Function            | 03  | 0 3              | 0000 0011       |
| Starting Address Hi | 00  | 0 0              | 0000 0000       |
| Starting Address Lo | 6B  | 6 B              | 0110 1011       |
| No. of Registers Hi | 00  | 0 0              | 0000 0000       |
| No. of Registers Lo | 03  | 0 3              | 0000 0011       |
| Error Check         |     | LRC (2 chars.)   | CRC (16 bits)   |
| Trailer             |     | CR LF            | None            |
| Total Bytes:        | 17  |                  | 8               |

Slave Response, ASCII and RTU Framing

| RESPONSE      |     |                  |                 |
|---------------|-----|------------------|-----------------|
| Field Name    | Hex | ASCII Characters | RTU 8-Bit Field |
| Header        |     | :                | (colon)         |
| Slave Address | 06  | 0 6              | 0000 0110       |
| Function      | 03  | 0 3              | 0000 0011       |
| Byte Count    | 06  | 0 6              | 0000 0110       |
| Data Hi       | 02  | 0 2              | 0000 0010       |
| Data Lo       | 2B  | 2 B              | 0010 1011       |
| Data Hi       | 00  | 0 0              | 0000 0000       |
| Data Lo       | 00  | 0 0              | 0000 0000       |
| Data Hi       | 00  | 0 0              | 0000 0000       |
| Data Lo       | 63  | 6 3              | 0110 0011       |
| Error Check   |     | LRC (2 chars.)   | CRC (16 bits)   |
| Trailer       |     | CR LF            | None            |
| Total Bytes:  |     | 23               | 11              |

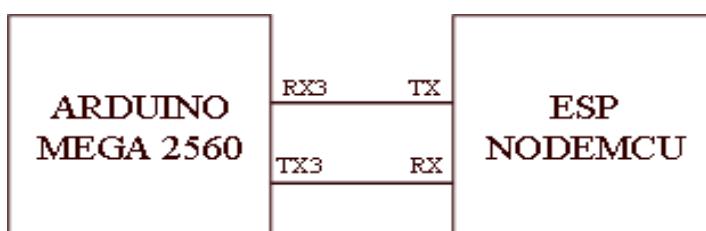
Hình 2.12 Ví dụ một quá trình yêu cầu và phản hồi giữa Master và Slave

Trong ví dụ hình 2.12 trên thì Master gửi một mã hàm là 03 để đọc giá trị thanh ghi với địa chỉ bắt đầu là 6B đọc 3 thanh ghi, sau cùng là mã kiểm tra lỗi CRC. Trong khung dữ liệu phản hồi từ Slave bắt đầu là địa chỉ của Slave và byte trả về là 6 byte, mỗi thanh ghi sẽ chứa 2 byte, một byte cao đứng trước và một byte thấp đứng sau. Cuối cùng là mã kiểm tra lỗi CRC 2 byte.

Trong đề tài nhóm đã chọn chuẩn giao thức Modbus RTU với kiểu kết nối vật lý là RS-485. Arduino đóng vai trò là Master trong hệ thống. Các thiết bị còn lại đóng vai trò là Slave (PLC, đồng hồ).

### 2.4.2 Giao thức UART

UART là viết tắt của Universal Asynchronous Receiver – Transmitter Là kiểu truyền thông tin nối tiếp không đồng bộ thường là một mạch tích hợp. Mục đích của UART là để truyền tín hiệu qua lại lẫn nhau (ví dụ truyền tín hiệu từ Laptop vào Modem hay ngược lại) hay truyền từ vi điều khiển tới vi điều khiển, từ laptop tới vi điều khiển.



Hình 2.13 Truyền dữ liệu qua lại giữa vi điều khiển và node MCU

## **Chương 3. TÍNH TOÁN VÀ THIẾT KẾ**

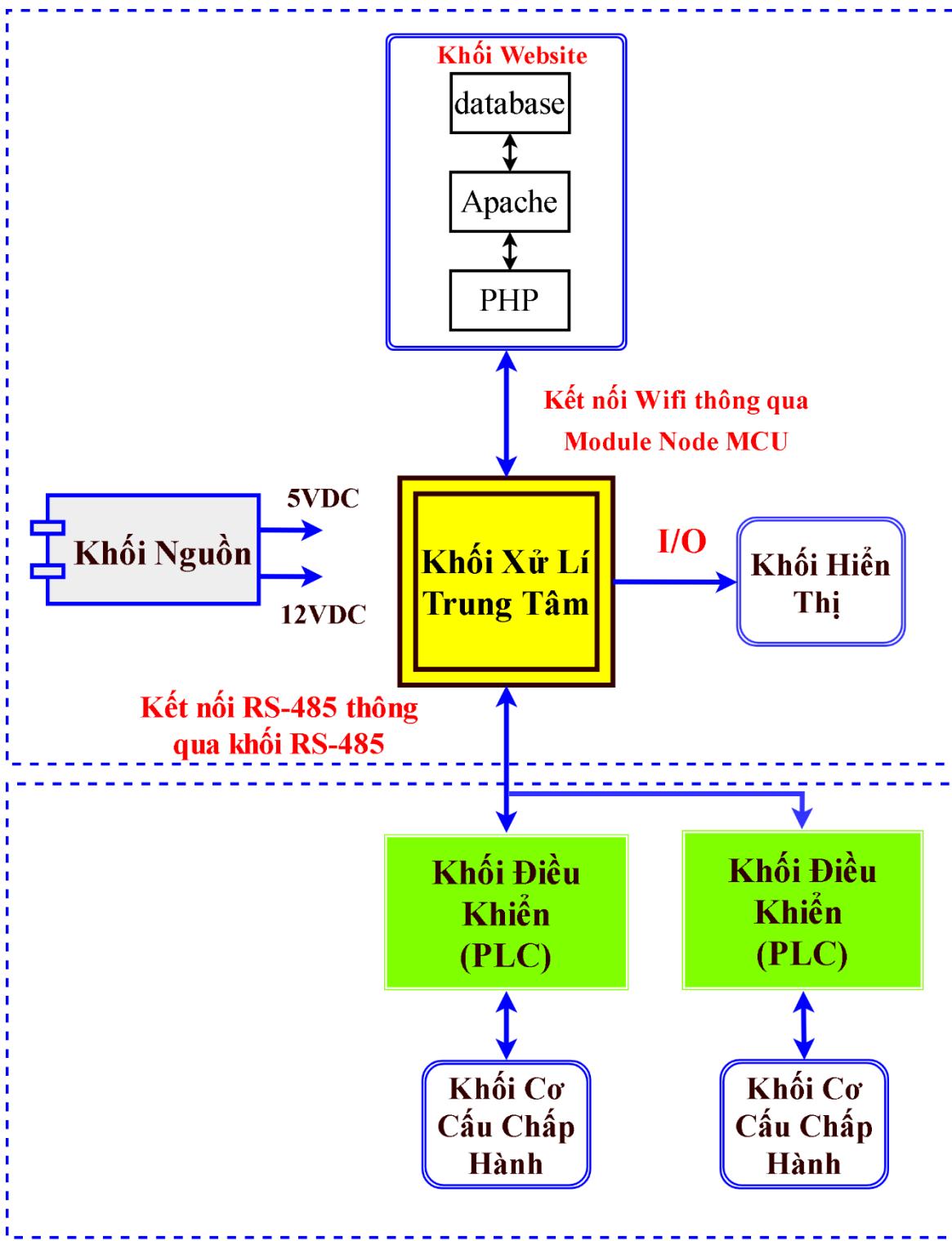
### **3.1 GIỚI THIỆU**

Theo mục tiêu đề ra của đề tài là thiết kế một hệ thống truyền dữ liệu từ Web Server xuống PLC để hỗ trợ người dùng cài đặt các giá trị cho hệ thống hoạt động và theo chiều ngược. Sẽ lấy các giá trị như counter, timer, giá trị cảm biến của PLC truyền lên Web Server để lưu trữ và giám sát hệ thống. Quá trình truyền nhận đó được thực hiện thông qua Arduino để xử lí.

### **3.2 TÍNH TOÁN VÀ THIẾT KẾ HỆ THỐNG**

#### **3.2.1 Thiết kế sơ đồ khối hệ thống**

Hệ thống được thiết kế với sơ đồ khối chia thành hai phần: một phần là đặt trong môi trường công nghiệp như PLC và các cơ cấu chấp hành. Phần còn lại là gồm có thể đặt trong môi trường khác mà bộ điều khiển có thể hoạt động được.



**Hình 3.1 Sơ đồ khái niệm của hệ thống**

**Chức năng của các khối:**

- **Khối Nguồn:** Trong đề tài của nhóm thì khối nguồn có chức năng cung cấp nguồn 5vdc cho khối xử lý trung tâm, khối hiển thị khối giao tiếp RS-485 và khối Module Node MCU, cấp nguồn 24vdc cho PLC hoạt động.

## CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

---

- *Khối cơ cấu chấp hành:* Đó là các cảm biến phát hiện và cảm biến nhiệt có chức năng đếm sản phẩm và đo nhiệt độ gửi dữ liệu về cho PLC.
- *Khối Điều Khiển (PLC):* Lập trình cho các thiết bị ở khối cơ cấu chấp hành hoạt động đúng yêu cầu và giao tiếp RS-485 để truyền nhận dữ liệu.
- *Khối Giao Tiếp RS-485:* Có chức năng giao là chuyển đổi giao tiếp RS-485 thành giao tiếp UART và tự động chọn chiều để khôi xử lý trung tâm truyền nhận dữ liệu với PLC.
- *Khối Xử Lý Trung Tâm:* Là khôi quan trọng của hệ thống chúa Arduino có chức năng giao tiếp UART với khôi Module NodeMCU truyền nhận dữ liệu và giao tiếp với khôi giao tiếp RS-485 để truyền nhận dữ liệu với PLC.
- *Khôi Module Node MCU:* Khôi này có chức năng kết nối Wifi và truy cập Web Server qua Wifi để lấy dữ liệu từ Web Server xuống cho khôi xử lý trung tâm qua giao thức UART và ngược lại lấy dữ liệu từ khôi xử lý trung tâm đưa lên Website để lưu trữ và hiển thị qua giao thức PHP.
- *Khôi Hiển Thị:* Khôi này kết nối trực tiếp với khôi xử lý trung tâm có chức năng hiển thị dữ liệu giao tiếp từ Web Server và PLC và báo hiệu các trạng thái của hệ thống.
- *Khôi Web Server:* Khôi này bao gồm 3 thành phần đó là Database, Apache và PHP. Database có chức năng chứa cơ sở dữ liệu của Web Server. Apache là một chương trình kết nối giữa PHP với Database. PHP là ngôn ngữ lập trình cho Website. Chức năng của toàn khôi là hiển thị giao diện cho người dùng điều khiển và giám sát dữ liệu từ các dây chuyền từ các thiết bị công nghiệp.

### 3.2.2 Tính toán và thiết kế mạch

#### a. Khối xử lý trung tâm

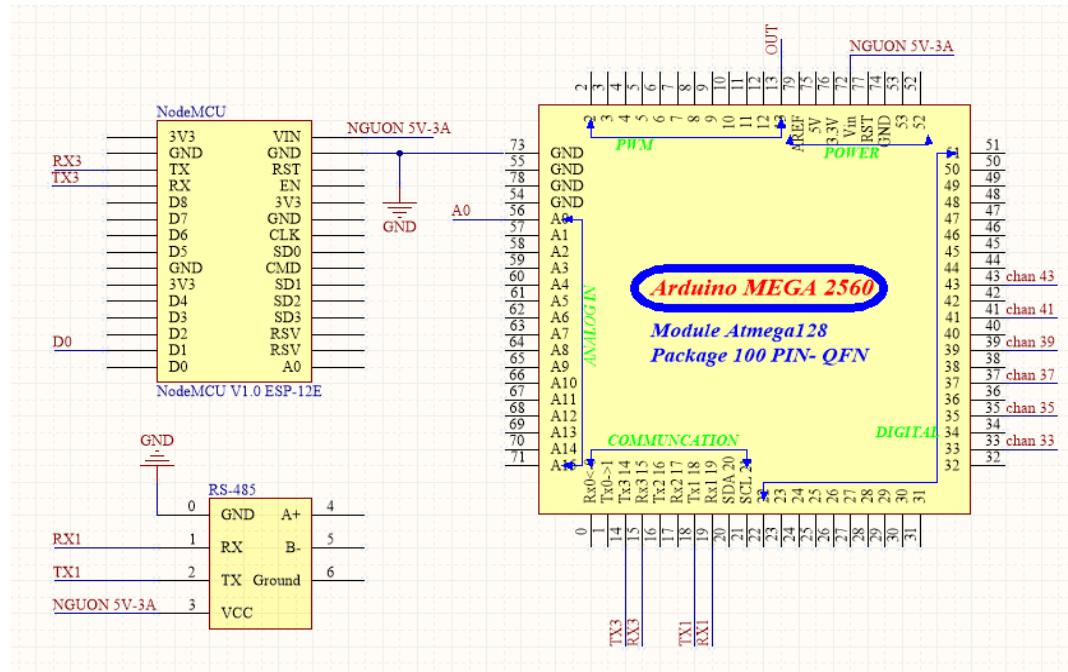
Để xử lý dữ liệu hai chiều phải cần một bộ xử lý dữ liệu và 3 cổng giao tiếp Serial một để giao tiếp với Node MCU, một giao tiếp với RS-485, và một dùng để nạp chương trình cho bộ xử lí. Cho nên nhóm chọn board Arduino Mega 2560 R3 sử dụng bộ xử lí Atmega2560.

Board Atmega 2560 này có nhiều chân I/O hỗ trợ cho chúng em giao tiếp, quan trọng hơn là nó có nhiều cổng UART(Tx, Rx) phù hợp với đề tài của chúng

## CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

em cùng với nó có những ưu điểm đó là: hỗ trợ thư viện cho giao tiếp Modbus, tương thích được với nhiều hệ điều hành, chương trình lập trình đơn giản, rõ ràng, dễ sử dụng, sử dụng mã nguồn mở và có thể kết hợp với nhiều module khác nhau.

Phần mềm để lập trình cho vi điều khiển là phần mềm IDE. Đây là phần mềm mã nguồn mở và có thể được coi là một phần mềm lập trình lý tưởng cho học sinh, sinh viên...



Hình 3.2 Sơ đồ nguyên lý của khối xử lý trung tâm

Trong sơ đồ thì khối xử lý trung tâm hình 3.2 thì Arduino Mega 2560 sẽ kết nối với module Node MCU qua cổng giao tiếp UART số 3 và kết nối với module RS-485 qua cổng UART số 1. Các chân 34 đến 43 sẽ kết nối với LCD 20x4.

### b. Module Node MCU

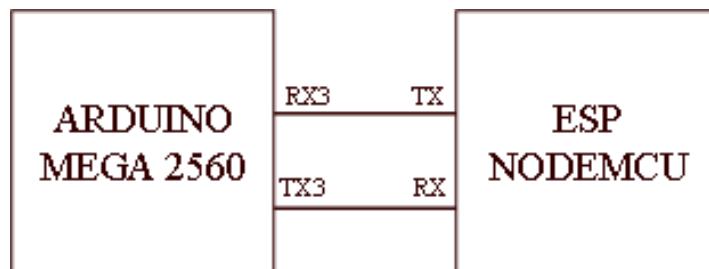
Để đưa dữ liệu lên cơ sở dữ liệu cần có một thiết bị kết nối với Wifi và có thể giao tiếp với Arduino đó chính là ESP8266. Và có thể lập trình trên ESP8266 và có thể nạp chương trình qua Internet nên nhóm chọn Module NodeMCU để đáp ứng những yêu cầu và chức năng của khối này.

So với ESP8266 V1 thì Module Node MCU có tích hợp mạch nạp trực tiếp trên nó và đặc biệt là Module Node MCU có hỗ trợ nạp thông qua Wifi, chúng ta không cần phải cắm dây hoặc đấu nối gì hết.

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

Module Node MCU có hỗ trợ nhiều chân UART hơn thuận tiện cho việc giao tiếp với ngoại vi hơn.

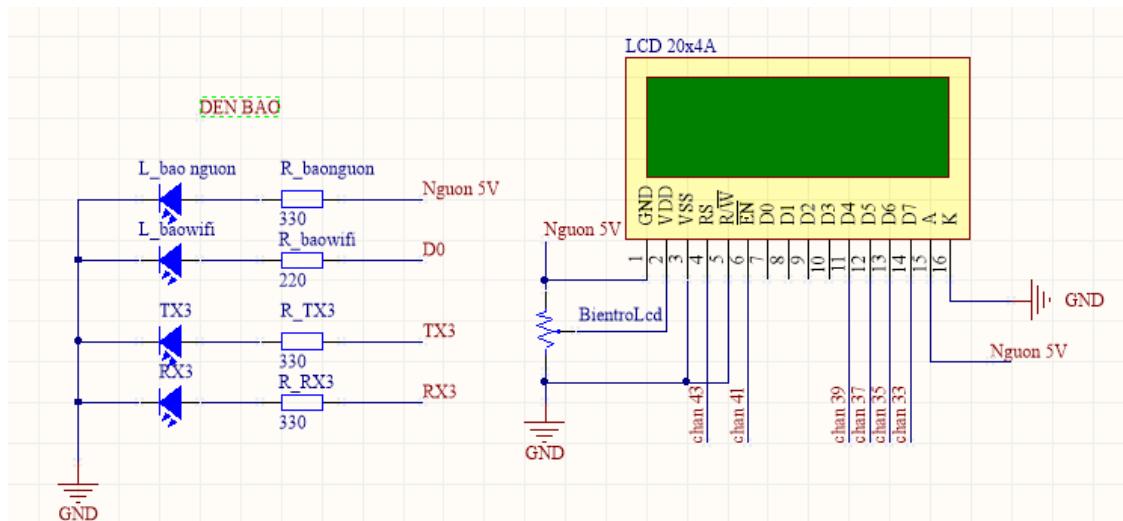
Trong đề tài thì nhóm sử dụng cổng giao tiếp UART của Module Node MCU giao tiếp với cổng UART số 3 của Arduino để truyền nhận dữ liệu được thể hiện qua hình 3.3.



Hình 3.3 Sơ đồ kết nối của Node MCU với Arduino

#### c. Khối hiển thị

Khối có chức năng hiển thị các dữ liệu giao tiếp giữa Web Server và PLC nên chọn LCD hiển thị dữ liệu, để kiểm tra dữ liệu có truyền đúng hay không. Do dữ liệu hiển thị nhiều nên nhóm chọn LCD 20x4. Ngoài ra còn các led để báo hiệu kết nối Wifi của NodeMCU, led báo nguồn 5V, led báo hiệu tín hiệu giao tiếp giữa Arduino và NodeMCU.



Hình 3.4 Sơ đồ nguyên lý khói hiển thị

Theo sơ đồ hình 3.4 các chân Enable, chân RS và chân dữ liệu D4-D7 kết nối với các chân của Arduino tương ứng. Biến trở để cân chỉnh độ tương phản của LCD. Các led có nhiệm vụ báo nguồn 5V, 2 led báo tín hiệu Tx3, Rx3 của Arduino và led còn lại nối vào chân D0 của Node MCU báo khi kết nối WiFi thành công.

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

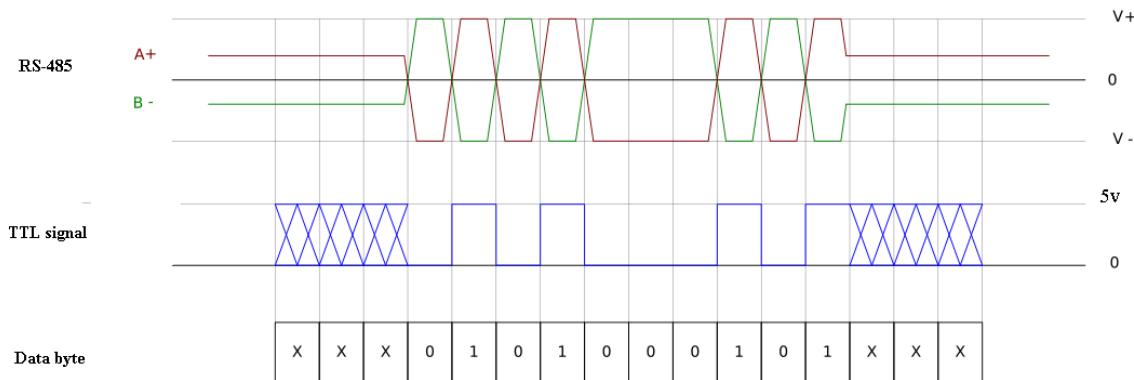
Điện trở của các led sẽ được tính theo công thức:

$$R = \frac{(V_s - V_{led})}{I_{led}} \quad (3.1)$$

Chọn  $I_{led}$  là dòng hoạt động bằng 10mA và  $V_{led}$  là điện áp hoạt động của led đỏ là 2,1V. Từ công thức (3.1) tính điện được  $R = \frac{(5 - 2,1)V}{10mA} = 290 \Omega$  cho nên chọn điện trở nguồn, TX, RX là  $330\Omega$ .

#### d. Khối giao tiếp RS-485

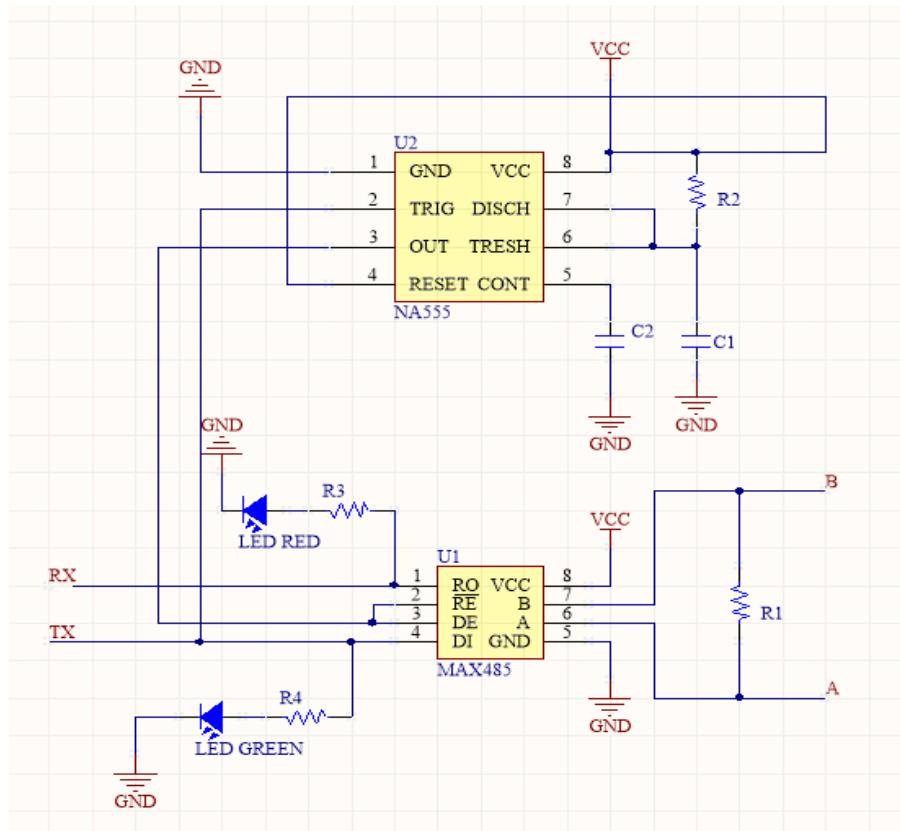
Do sự khác biệt về tín hiệu RS 485 và tín hiệu UART không có cùng một mức điện áp được thể hiện như hình:



**Hình 3.5 Mức điện áp của tín hiệu RS-485 và UART**

Do tín hiệu của RS-485 là sự chênh lệch điện áp giữa  $V+$  và  $V-$  và tín hiệu UART là chênh lệch điện áp 5V và 0V nên để có thể truyền dữ liệu từ PLC tới Arduino và ngược lại thì phải khối chuyển đổi cổng giao tiếp RS-485 của PLC thành UART được hỗ trợ bởi Arduino. Do chế độ truyền là đơn công nên phải tự động chọn chiều truyền nhận dữ liệu. Sử dụng một mạch chuyển đổi RS-485 sử dụng Max 485 và NA555 có sơ đồ như sau:

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

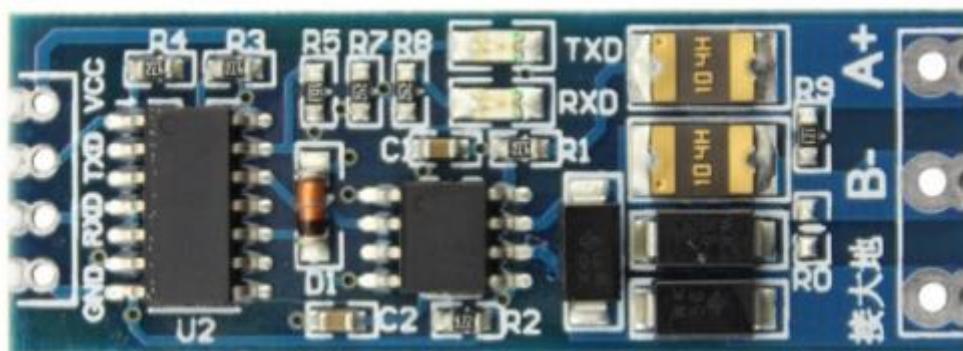


**Hình 3.6** Sơ đồ nguyên lý mạch RS-485

Trong sơ đồ ngõ ra sử dụng cạnh xuống của bit bắt đầu giao thức UART để thực hiện điều khiển hướng tự động. Cạnh xuống kích hoạt bộ đếm thời gian NA555 tạo tín hiệu cho phép chọn chiều truyền đi. Thời gian truyền của gói dữ liệu được tính theo công thức:

$$T = 1,1 \times R2 \times C1 \quad (3.2)$$

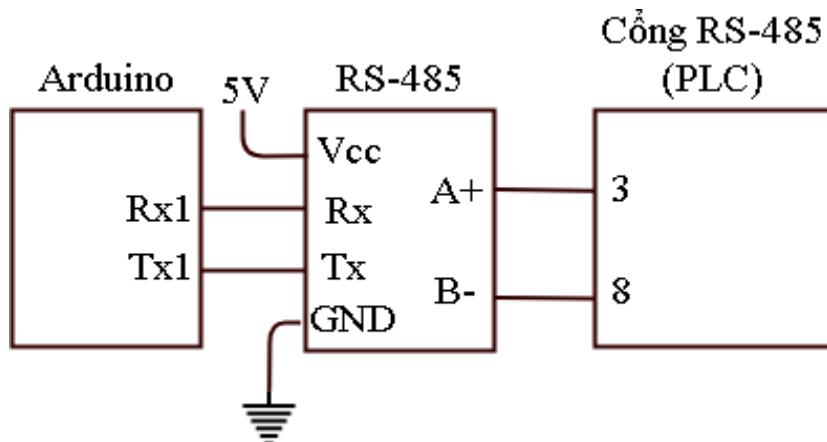
Trong mạch thực tế để mạch hoạt động thì phải có Diode chống nhiễu giúp hệ thống ổn định và cầu chì tự phục hồi (poly fuse) cho nên chọn một module có sẵn là mạch giao tiếp UART TTL – RS485 như hình:



**Hình 3.7** Sơ đồ mạch UART TTL-RS485

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

Trong hệ thống mạch giao tiếp này sẽ kết nối với PLC và Arduino qua sơ đồ bên dưới:



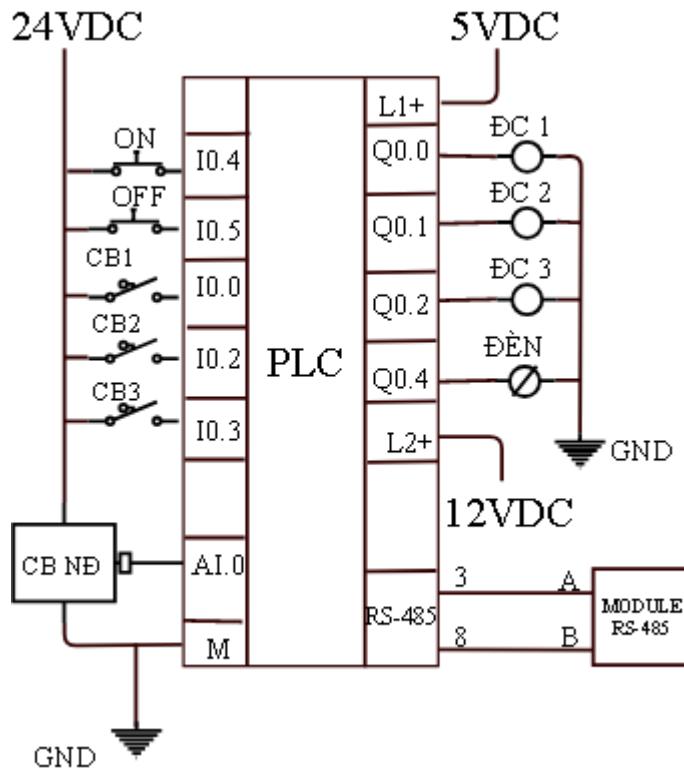
**Hình 3.8** Sơ đồ kết nối của module RS485 với PLC và Arduino.

Trong sơ đồ hình 3.8 thì module RS-485 sẽ kết nối với cổng giao tiếp UART số 1 của Arduino và chân A+,B- sẽ kết nối với chân 3, 8 của cổng giao tiếp truyền thông RS485 của PLC. Nguồn cấp cho module là 5V.

#### e. Khối điều khiển (PLC)

Do mô hình điều khiển nhỏ cho nên nhóm sử dụng board CPU 224x có chức năng tương tự như PLC s7-200 của hãng siemens. Ngõ vào của board kết nối với 3 cảm biến để giả lập đếm sản phẩm và kết nối với một cảm biến nhiệt độ. Ngõ ra được kết nối với 1 đèn báo hoạt động và 3 động cơ. Cổng truyền thông thứ nhất của board nối với Module RS485 để giao tiếp truyền dữ liệu cho Arduino. Sơ đồ kết nối như sau:

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

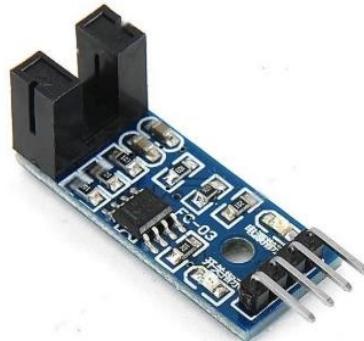


**Hình 3.9** Sơ đồ kết nối PLC và các thiết bị

Trong sơ đồ hình 3.9 thì hai nút nhấn cho hệ thống chạy và nút dừng hệ thống và các cảm biến đếm sản phẩm kết nối với PLC vào các ngõ vào digital (số), cảm biến nhiệt độ nối ngõ vào analog của PLC. Ngõ ra của PLC kết nối với các động cơ và đèn báo hoạt động để báo hiệu. Cổng giao tiếp RS-485 sẽ kết nối với module RS-485 chân 3, 8 là hai chân truyền nhận dữ liệu của PLC sẽ kết nối với chân A, B của module RS-485.

#### f. Thiết kế khối cơ cấu chấp hành

Trong khối này nhóm sử dụng module cảm biến Encoder với động cơ giảm tốc để mô phỏng lại cảm biến đếm sản phẩm.



**Hình 3.10** Module cảm biến Encoder

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

Module cảm biến hoạt động với điện áp vào là 5VDC, hai bên khe màu đen có một led thu và một led phát hồng ngoại. Khi khe trống thì ánh sáng từ led phát truyền đến led thu và mức điện áp ra ở chân D0 là 5V còn ngược lại khi bị che thì mức điện áp là 0V.



**Hình 3.11** Động cơ giảm tốc DC

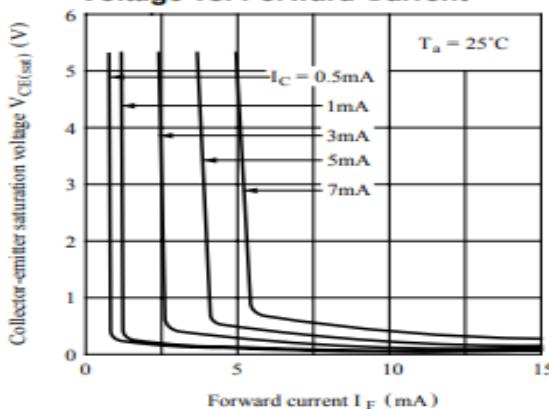
Động cơ giảm tốc sẽ kết hợp với một bánh xe có một gạch để khi động cơ quay thì gạch quay theo và khi đến khe của cảm biến sẽ che đi ánh sáng từ led phát đến led thu tạo ra một tín hiệu 0V. Tín hiệu đó sẽ được vào của PLC để giả lập một cảm biến đếm sản phẩm.

Trong sơ đồ nguyên lý hình 3.13 để kích ngõ vào mà PLC có thể nhận biết được thì dòng khoảng 5mA. Từ đó tính được điện trở ở nguồn 24V:

$$R = \frac{V}{I} == 4,8 \text{ } (k\Omega) \quad (3.3)$$

Cho nên chọn điện trở 4,7 ( $k\Omega$ ) theo công thức 3.3 thì suy ra được dòng 5,106 (mA) thì có thể kích được ngõ vào cho PLC.

**Fig.12 Collector-emitter Saturation Voltage vs. Forward Current**



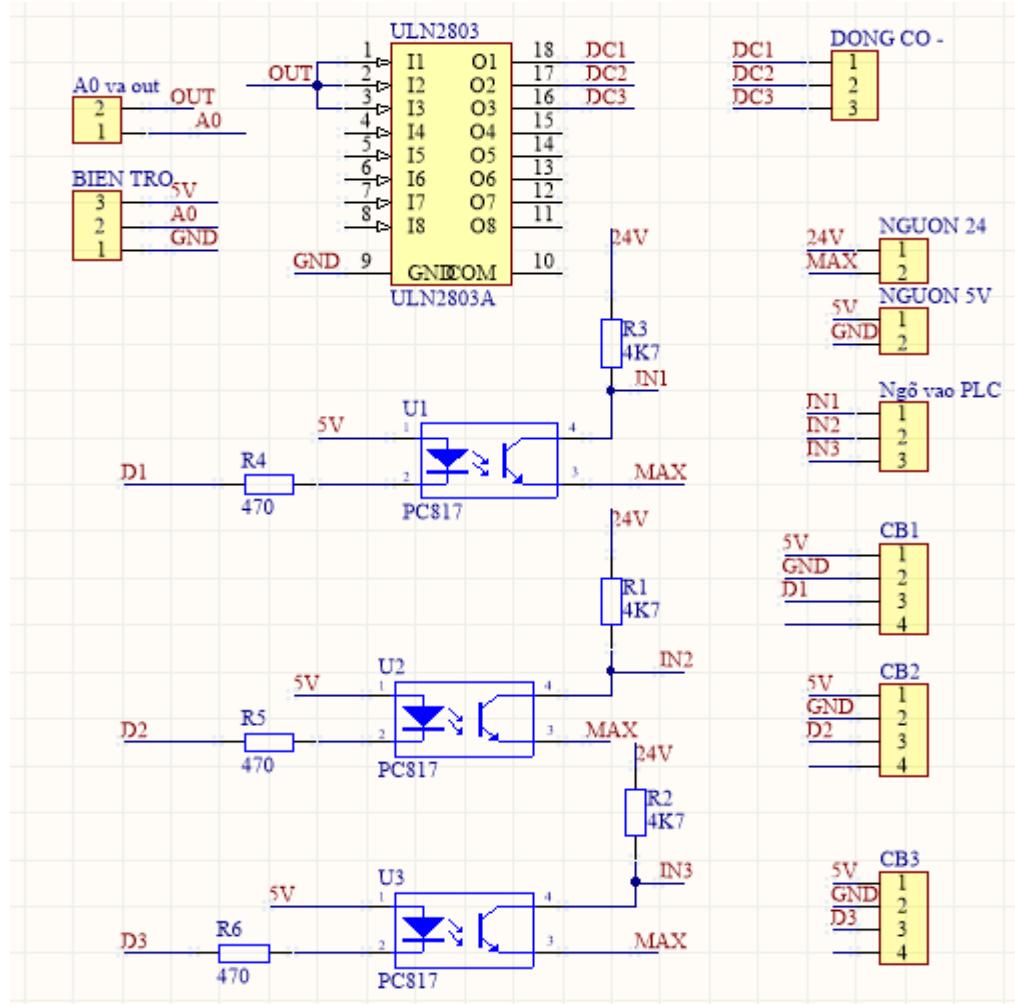
**Hình 3.12** Dòng chuyển tiếp của PC817

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

Theo hình 3.12 dòng điện để kích cho PC817 dẫn khi ở nguồn 5V khoảng  $10mA \leq I_F \leq 15mA$ . Điện trở ngõ vào cảm biến sẽ được tính bằng công thức:

$$R = \frac{5V}{I_F} = \frac{5V}{10mA} = 500 (\Omega) \quad (3.4)$$

Cho nên chọn điện trở 470 ( $\Omega$ ) từ đó theo công thức (3.4) tính lại  $I_F = 10,6mA$  thỏa mãn điều kiện để khi ngõ ra cảm biến xuống mức thấp thì ngõ vào của PLC.



Hình 3.13 Sơ đồ nguyên lý của mạch điều khiển cảm biến

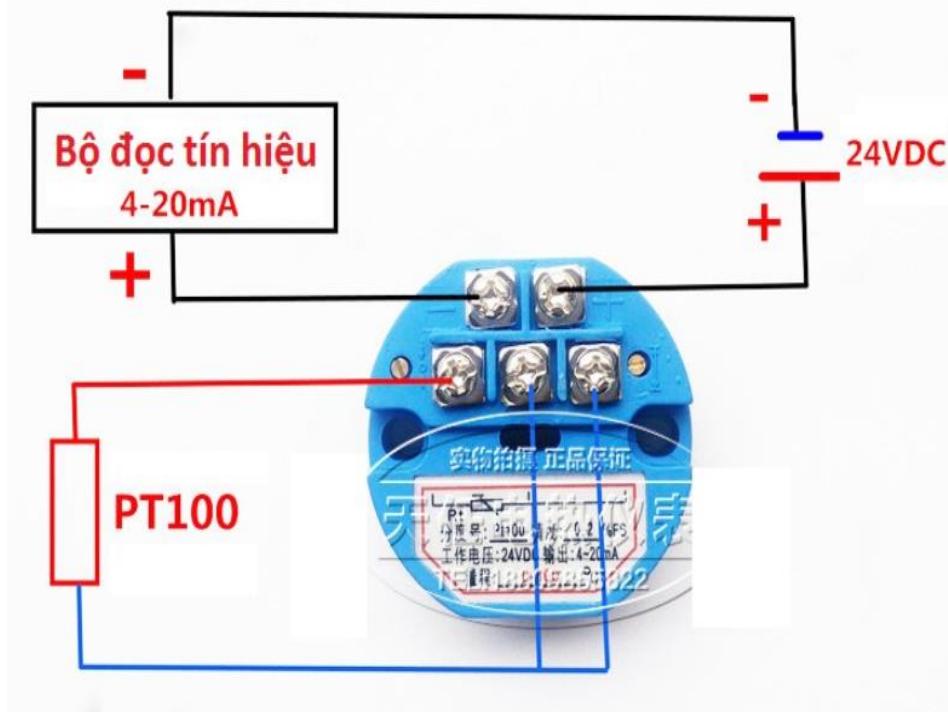
Theo hình 3.13 khối này thì sẽ lấy xung PWM ở chân 13 của Arduino qua ULN2803 để khuếch đại dòng cho động cơ chạy và điều chỉnh tốc độ qua biến trở. Và các ngõ ra số của module cảm biến Encoder đưa vào mạch Opto để kích ngõ vào 24V cho PLC. Khi ngõ ra của cảm biến 0V thì đồng thời ngõ vào của PLC là 24V, tương ứng với một sản phẩm đi qua.

### CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

Trong khối này nhóm sử dụng cảm biến nhiệt độ PT100 loại 2 dây kết hợp với bộ chuyển đổi nhiệt độ từ cảm biến PT100 sang tín hiệu 4-20mA cấp cho ngõ vào analog của PLC.



Hình 3.14 Bộ chuyển đổi PT100 4-20mA



Hình 3.15 Sơ đồ kết nối PT100 với bộ chuyển đổi

Theo hình 3.14 thì cảm biến PT100 sẽ được kết nối vào 3 đầu bên dưới của module chuyển đổi. Bộ đọc tín hiệu 4-20mA chính là ngõ vào Analog của PLC, đầu còn lại được kết nối với nguồn 24VDC.

#### g. Thiết kế khối nguồn

Tính toán dòng tiêu thụ:

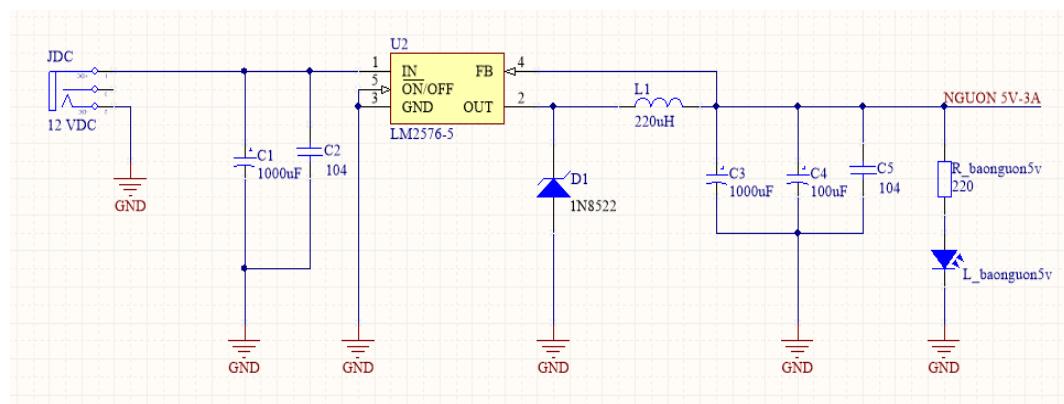
- Board Arduino Mega2560 tiêu thụ dòng tối đa khoảng 200mA.

## CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

- Node MCU dùng áp 5 V và tiêu thụ dòng <= 170mA.
- LCD 20x4 dùng áp 5V tiêu thụ dòng khoảng 30mA tính cả đèn nền.
- Module RS-485 dùng áp 5V tiêu thụ dòng khoảng 200mA.
- 4 led tiêu thụ dòng 60mA.
- Các cảm biến encoder tiêu tốn khoảng 60 mA.
- Các động cơ tiêu thụ dòng khoảng 360mA.

Tổng dòng tiêu thụ của các linh kiện trong mạch khoảng 1080mA, cho nên nhóm chọn thiết kế mạch nguồn 5V dùng LM2576-5V. Mạch nguồn dùng LM2576 tạo ra nguồn 5V ổn định có dòng lên tới 3A để cho các linh kiện trong mạch hoạt động một cách tốt nhất.

Sơ đồ nguyên lý mạch nguồn:



Hình 3.16 Sơ đồ nguyên lý khói nguồn

Trong mạch hình 3.13 nguồn vào được cấp bởi Adapter 12V- 3A qua tụ C1, C2, để lọc các tần số cao và tần số thấp vào LM2576. Các tụ C3, C4, C5 tương tự cũng để lọc các tần số cao, tần số thấp ở ngõ ra. Cuộn dây có chức năng chấn các tần số cao đi qua. Còn Diode D1 là Schottky được mắc ngược để bảo vệ điện áp đầu ra. Chân số Feedback của LM2576 là chân phản hồi điện áp về để điều chỉnh điện áp ngõ ra. Các thông số của linh kiện được chọn dựa trên Datasheet của LM2576.

### 3.2.3 Tính toán và thiết kế webserver

Nhóm chúng em đã chọn sử dụng trang <https://vn.000webhost.com/> để viết web đây là web hosting miễn phí. Web được không gian lưu trữ và băng thông gần như không giới hạn cùng với các kết nối không giới hạn tới các máy chủ. Web

## **CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ**

---

hosting miễn phí duy nhất cam kết đảm bảo uptime lên tới 99%, với hầu hết các servers đạt tỉ lệ uptime tới 99.9%.

Do web sử dụng với mục đích chính là điều khiển và giám sát hệ thống nên giao diện thiết kế không quá cầu kỳ đơn giản dễ sử dụng.

Web cũng có tất cả các thông tin giới thiệu liên hệ về người phụ trách về hệ thống. Web được thiết kế dựa trên nền tảng là ngôn ngữ PHP với mã nguồn mở nên có thể dễ dàng tìm hiểu và sử dụng.

Hệ thống thiết kế ước lượng dữ liệu đưa lên cơ sở dữ liệu khoảng 15 biển kiểu thập phân và dữ liệu được lưu trữ mỗi 3 giây. Trung bình một ngày vận hành có khoảng 864kb dữ liệu được đưa từ cảm biến. Các giá trị cài đặt, thiết lập không đáng kể và cố định. Với lượng dữ liệu này thì có thể đáp ứng trên các hệ thống Server/Hosting miễn phí. Do đã được định hình từ trước nên web được thiết kế theo cái mục đích sử dụng bên dưới, sẽ gồm có các nút nhấn start, stop cùng với những textbox có thể nhập giá trị xuống bên dưới, có các thanh trạng thái chuyển đổi từ việt điều khiển sang giám sát, bên giám sát thì có một bảng giám sát được đưa từ hệ thống bên dưới lên.

Trong thiết kế web có thể viết trên phần mềm dreamweaver hoặc có thể chỉnh sửa trực tiếp trên web, trên web có một nơi chứa tất cả các file lập trình về giao diện cũng như nội dung và các file kết nối xuống thiết bị. Việc lập trình hoặc chỉnh sửa cũng trở nên dễ dàng hơn khi chúng ta chỉ việc thao tác trực tiếp trên đó chỉ cần chúng ta có sử dụng 1 trình duyệt web bất kỳ nào như: Cốc Cốc, Chrome, Explore..... Chúng ta cũng không cần phải cài bất kỳ một phần mềm nào mà vẫn có thể viết tiếp hoặc chỉnh sửa.

## CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ

The screenshot shows a file management interface for a website. The left sidebar lists directories: /, public\_html (selected), and tmp. The main area displays a table of files in the public\_html directory:

|  | Name          | Size   | Date                | Permissions |
|--|---------------|--------|---------------------|-------------|
|  | htaccess      | 0.2 kB | 2018-05-06 04:05:00 | -rW-f-r-    |
|  | about_us.php  | 0.5 kB | 2018-05-14 04:21:00 | -rW-f-r-    |
|  | add_data.php  | 0.4 kB | 2018-06-10 08:29:00 | -rW-f-r-    |
|  | cap3.php      | 2.4 kB | 2018-05-31 01:23:00 | -rW-f-r-    |
|  | cap7.php      | 2.4 kB | 2018-05-31 01:22:00 | -rW-f-r-    |
|  | chart.php     | 4.1 kB | 2018-05-31 00:43:00 | -rW-f-r-    |
|  | control.php   | 5.5 kB | 2018-06-11 08:30:00 | -rW-f-r-    |
|  | dangky.php    | 1.6 kB | 2018-06-08 02:46:00 | -rW-f-r-    |
|  | dangnhap.php  | 2.5 kB | 2018-05-06 07:47:00 | -rW-f-r-    |
|  | dbconnect.php | 0.5 kB | 2018-05-14 03:57:00 | -rW-f-r-    |
|  | dc1.json      | 0.1 kB | 2018-06-08 02:31:00 | -rW-f-r-    |
|  | footer.php    | 0.1 kB | 2018-05-14 04:23:00 | -rW-f-r-    |
|  | getdata.php   | 0.4 kB | 2018-06-11 06:41:00 | -rW-f-r-    |
|  | getdata1.php  | 0.4 kB | 2018-06-06 08:08:00 | -rW-f-r-    |
|  | giamsat.php   | 3.3 kB | 2018-06-10 08:45:00 | -rW-f-r-    |

Hình 3.17 Giao diện quản lý file lập trình

Web cũng hỗ trợ một Database để có thể lưu trữ thông tin về người đăng nhập, giá trị đo dặt từ bên dưới cũng như giá trị nhập từ web vào.

The screenshot shows the phpMyAdmin interface connected to a MySQL database named 'id5643192\_database'. The 'myproject' table is selected, displaying 24758 rows. The table structure includes columns: id, temp, hum, sp3, may2, and datetimet. The data shows various sensor readings over time.

|    | id   | temp | hum | sp3 | may2 | datetimet           |
|----|------|------|-----|-----|------|---------------------|
| 1  | 2575 | 32   | 76  | 0   | 0    | 2018-05-21 11:15:53 |
| 2  | 2576 | 32   | 77  | 0   | 0    | 2018-05-21 11:15:56 |
| 3  | 2577 | 32   | 77  | 0   | 0    | 2018-05-21 11:15:59 |
| 4  | 2578 | 32   | 77  | 0   | 0    | 2018-05-21 11:16:02 |
| 5  | 2579 | 32   | 76  | 0   | 0    | 2018-05-21 11:16:04 |
| 6  | 2580 | 32   | 76  | 0   | 0    | 2018-05-21 11:16:07 |
| 7  | 2581 | 32   | 77  | 0   | 0    | 2018-05-21 11:16:10 |
| 8  | 2582 | 32   | 78  | 0   | 0    | 2018-05-21 11:16:13 |
| 9  | 2583 | 32   | 78  | 0   | 0    | 2018-05-21 11:16:15 |
| 10 | 2584 | 32   | 79  | 0   | 0    | 2018-05-21 11:16:18 |
| 11 | 2585 | 32   | 79  | 0   | 0    | 2018-05-21 11:16:21 |
| 12 | 2586 | 32   | 80  | 0   | 0    | 2018-05-21 11:16:24 |
| 13 | 2587 | 32   | 80  | 0   | 0    | 2018-05-21 11:16:26 |
| 14 | 2588 | 32   | 81  | 0   | 0    | 2018-05-21 11:16:29 |
| 15 | 2589 | 32   | 82  | 0   | 0    | 2018-05-21 11:16:32 |
| 16 | 2590 | 32   | 81  | 0   | 0    | 2018-05-21 11:16:34 |

Hình 3.18 Database của trang web

Sau khi thiết kế tất cả được nằm trên web mà nhóm đặt tên là [websystemspkt.000webhostapp.com](http://websystemspkt.000webhostapp.com)

Giao diện trang điều khiển giống như hình 3.19 với 3 ô để nhập các giá trị của các sản phẩm khác nhau. Còn ô ở dưới để nhập số dòng cần kiểm tra để giám sát.

## CHƯƠNG 3. TÍNH TOÁN VÀ THIẾT KẾ



TRƯỜNG ĐẠI HỌC  
SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH  
**BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP - Y SINH**  
Khoa Điện - Điện Tử

HCMUTE

ĐĂNG XUẤT    GIÁM SÁT    VIEW    CHART

Nhập sản phẩm  
Sản phẩm 1 :   
Sản Phẩm 2 :   
Sản Phẩm 3 :   
gửi

Nhập số lượng hàng cần kiểm tra  
dây chuyền 1 :  Xác nhận

**Hình 3.19** Giao diện web điều khiển

### 3.2.4 Sơ đồ nguyên lý toàn mạch

Sơ đồ nguyên lý mạch điện tử được thiết kế có các cổng RS-485 để kết nối với PLC như hình bên:

- Khối xử lí trung tâm: là khu vực chứa Arduino và module Node MCU kết nối với nhau qua cổng UART số 3.
- Khối cảm biến: chứa mạch nguyên lý dùng để khuếch đại dòng động cơ và mạch cảm biến Encoder để điều khiển ngõ vào PLC.
- Khối nguồn 5V: chứa mạch nguyên lý sử dụng LM2576 để tạo nguồn 5V-3A từ nguồn 12VDC cấp cho các khối còn lại.
- Khối giao tiếp: mạch nguyên lý kết nối LCD và các led hiển thị.
- Khối giao tiếp RS-485: chứa module RS-485 TTL và các kết nối DB9 để giao tiếp với PLC.

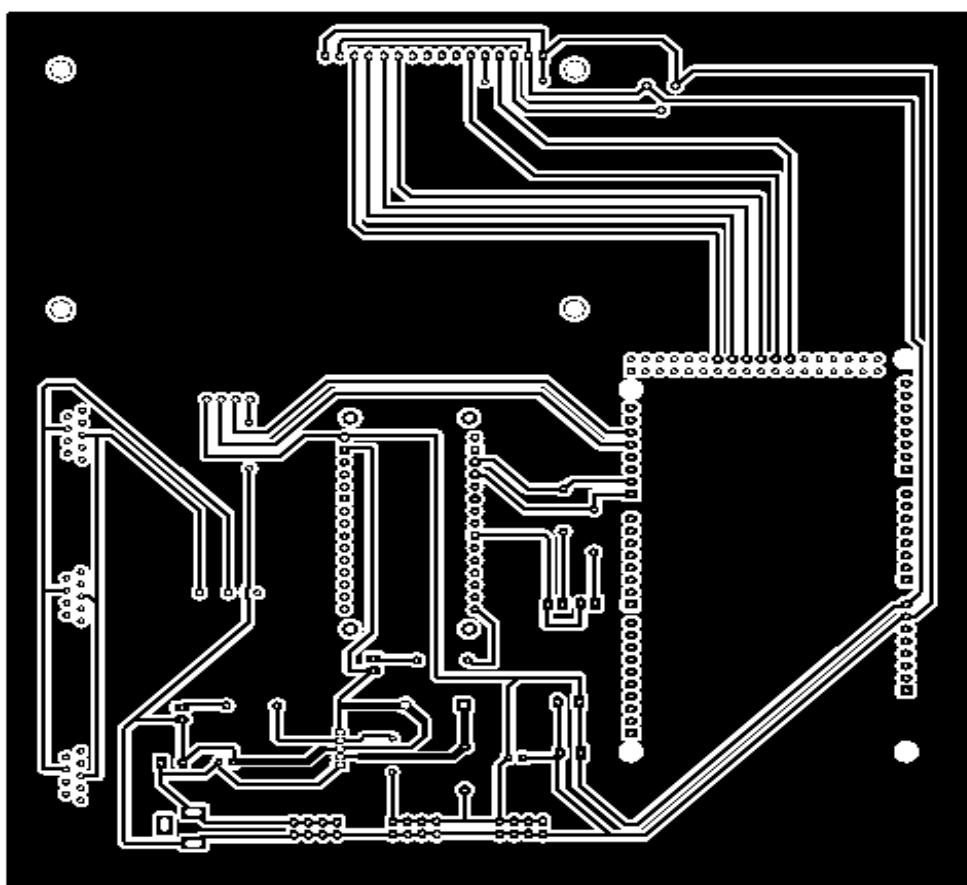
## **Chương 4. THI CÔNG HỆ THỐNG**

### **4.1 GIỚI THIỆU**

Trong chương này là quá trình thi công PCB, lập trình, lắp ráp phần cứng và test mạch. Bên cạnh đó là hình vẽ được chụp từ mô hình thực của hệ thống bên ngoài, hình chụp các kết quả chạy mà hệ thống tính đến thời điểm hiện tại.

### **4.2 THI CÔNG HỆ THỐNG**

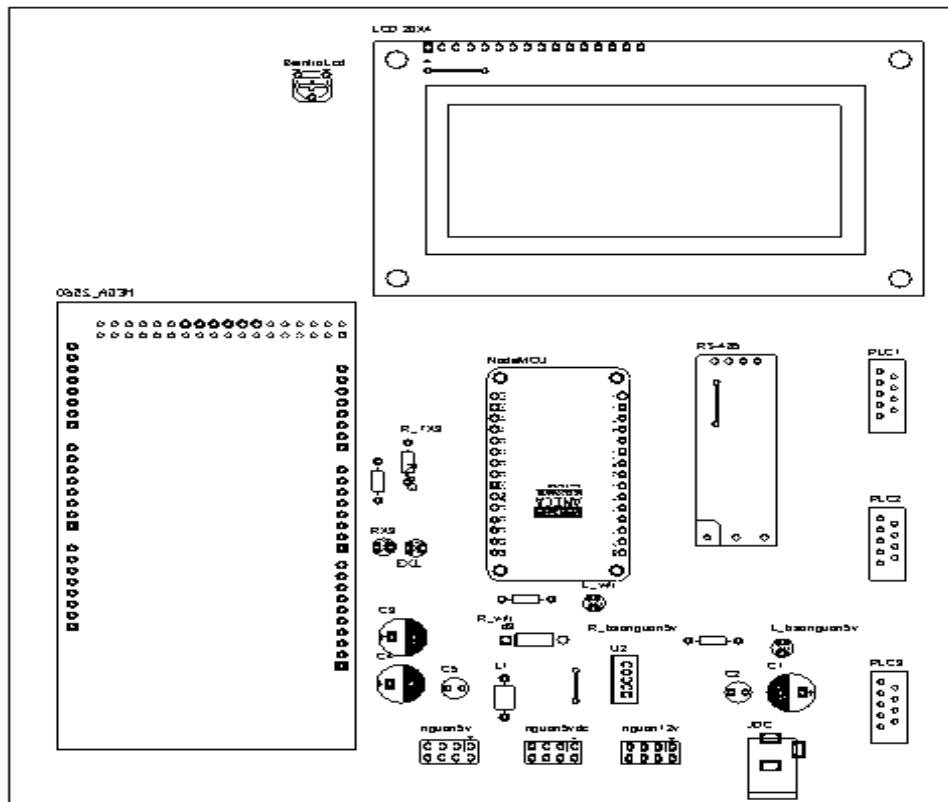
#### **4.2.1 Thi công bo mạch**



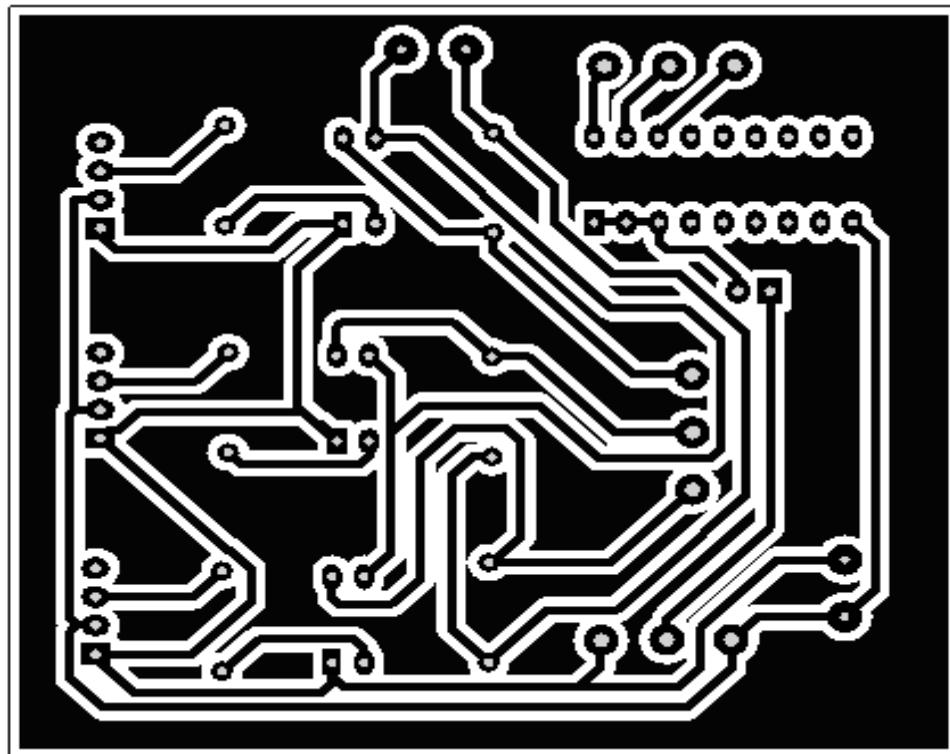
**Hình 4.1** Mặt dưới của mạch điều khiển chính

Trong hình 4.1 là sơ đồ đi dây mặt dưới của mạch điều khiển chính được dựa trên sơ đồ nguyên lý và được vẽ bằng phần mềm Altium Designer. Các linh kiện và vị trí của chúng được sắp xếp và bố trí như hình 4.2.

## CHƯƠNG 4. THI CÔNG HỆ THỐNG



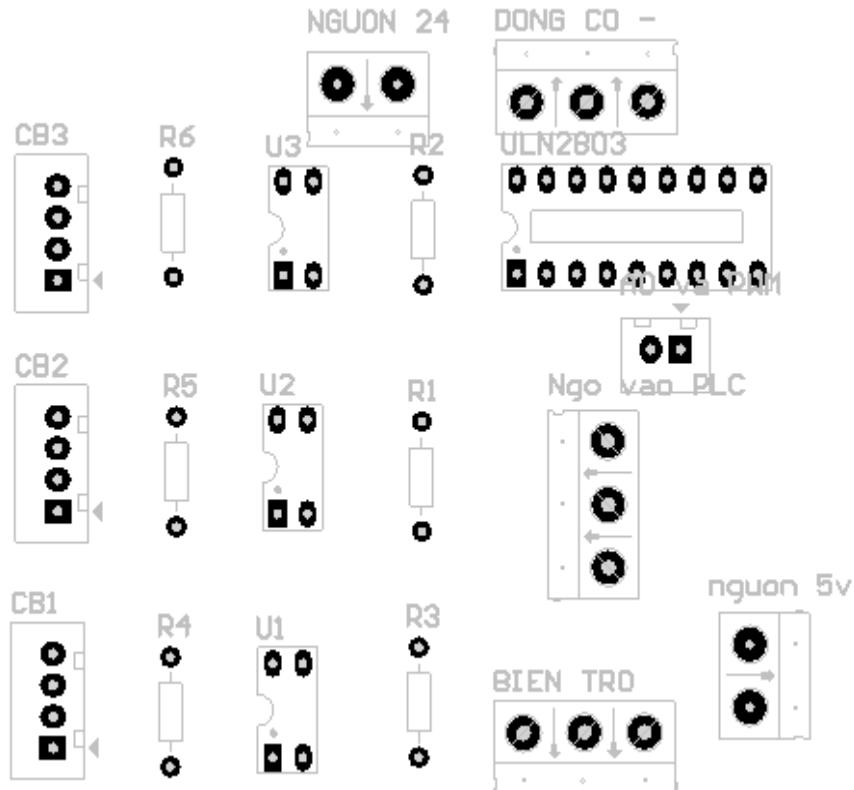
Hình 4.2 Sơ đồ bố trí linh kiện của mạch điều khiển chính



Hình 4.3 Mặt dưới của mạch điều khiển cảm biến

## CHƯƠNG 4. THI CÔNG HỆ THỐNG

Tương tự như mạch điều khiển chính mạch hình 4.3 cũng được thiết kế dựa trên sơ đồ nguyên lý đã trình bày ở chương 3 và cũng các linh kiện được bố trí như hình 4.4.



**Hình 4.4** Sơ đồ bố trí linh kiện của mạch chia cảm biến

Các linh kiện được sử dụng để thi công hệ thống được phân loại, thống kê số lượng, giá trị và các chú thích cho người dùng dễ dàng tìm và mua một cách chi tiết như bảng sau:

## CHƯƠNG 4. THI CÔNG HỆ THỐNG

Bảng 4.1 Danh sách các linh kiện.

| STT | Tên linh kiện           | Giá trị                   | Số lượng | Chú thích            |
|-----|-------------------------|---------------------------|----------|----------------------|
| 1   | IC LM2576               | 5V                        | 1        | Có tản nhiệt         |
| 2   | Board Arduino Mega 2560 |                           | 1        |                      |
| 3   | Module NodeMCU          |                           | 1        |                      |
| 4   | Module RS-485 TLL       | 5V                        | 1        |                      |
| 5   | LCD                     | 20 X 4                    | 1        |                      |
| 6   | Jack DB9                |                           | 3        | Loại male            |
| 7   | Led đơn                 | Đỏ, xanh                  | 4        |                      |
| 8   | Biến trở                | 5kΩ, 1kΩ                  | 2        |                      |
| 9   | Tụ điện                 | 1000uF, 100uF (25V), 10nF | 3        | Tụ hóa và tụ gốm     |
| 10  | Cuộn cảm                | 220uH                     | 1        |                      |
| 11  | Diode schotkky          | 3A                        | 1        |                      |
| 12  | Điện trở                | 220Ω, 330 Ω, 470 Ω, 4k7 Ω | 10       |                      |
| 13  | IC ULN2803              |                           | 1        | Ic nâng dòng         |
| 14  | PC817                   |                           | 3        |                      |
| 15  | Module cảm biến Encoder | 5V                        | 3        | Sử dụng led thu phát |
| 16  | Động cơ giảm tốc        | 3-5V                      | 3        |                      |

### 4.2.2 Lắp ráp và kiểm tra

Phần này chúng em trình bày quá trình lắp ráp từng phần:

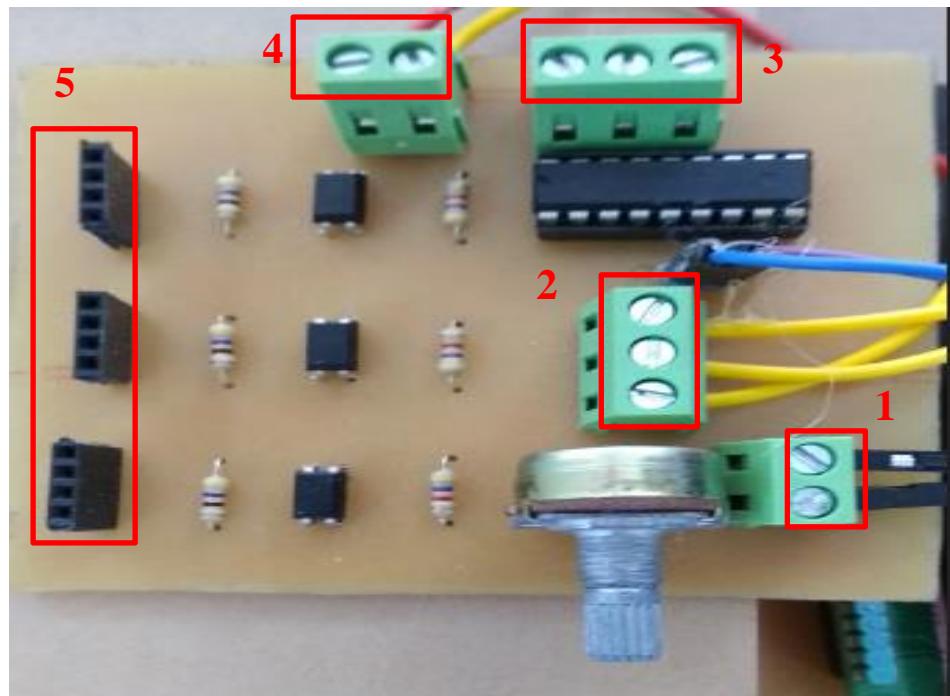
#### a. Module nguồn



**Hình 4.5** Sơ đồ bố trí linh kiện của mạch nguồn 5V

Tiến hành lắp ráp và hàn các linh kiện của module nguồn. Tiến hành đo và kiểm tra nguồn vào và ra xem đúng sai, cân chỉnh cho đúng.

#### b. Mạch cảm biến



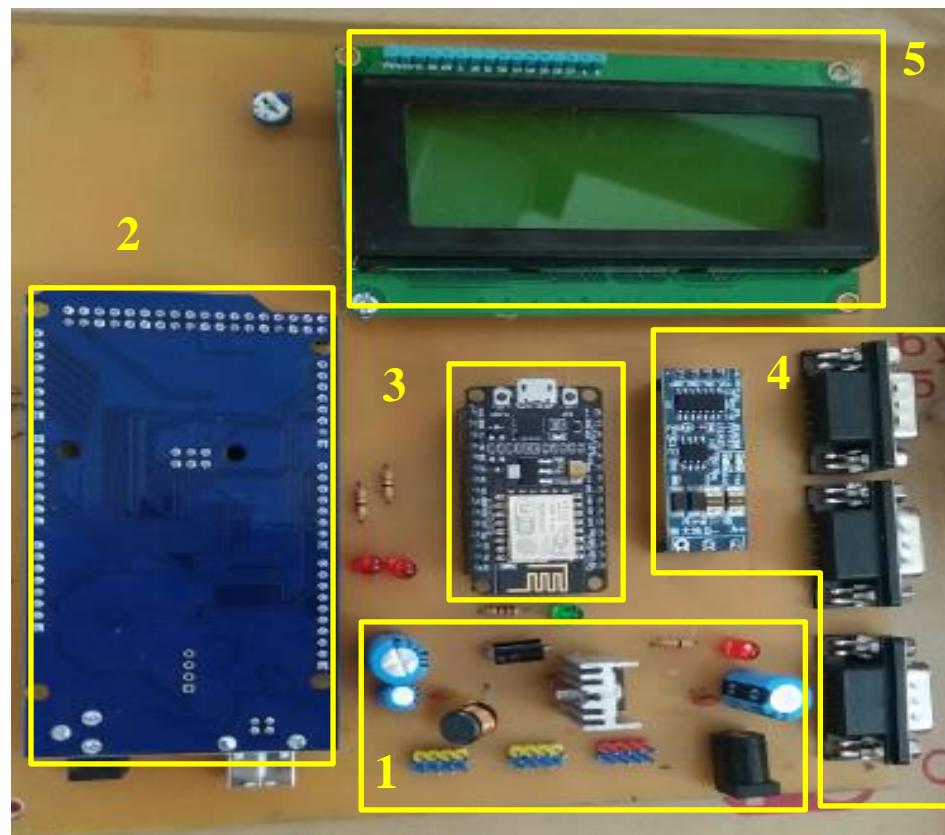
**Hình 4.6** Thi công mạch điều khiển cảm biến

Trong hình 4.6 là vị trí các linh kiện trong mạch điều khiển cảm biến được đánh số như sau: vị trí số 1 là nguồn 5VDC, vị trí 2 là ngõ vào của PLC, vị trí 3 là

## CHƯƠNG 4. THI CÔNG HỆ THỐNG

ngõ ra khuếch đại dòng cho động cơ, vị trí số 4 là nguồn 24VDC, cuối cùng vị trí số 5 là vị trí của các module cảm biến Encoder.

Tiến hành lắp ráp tất cả các module thì ta có sơ đồ toàn mạch.



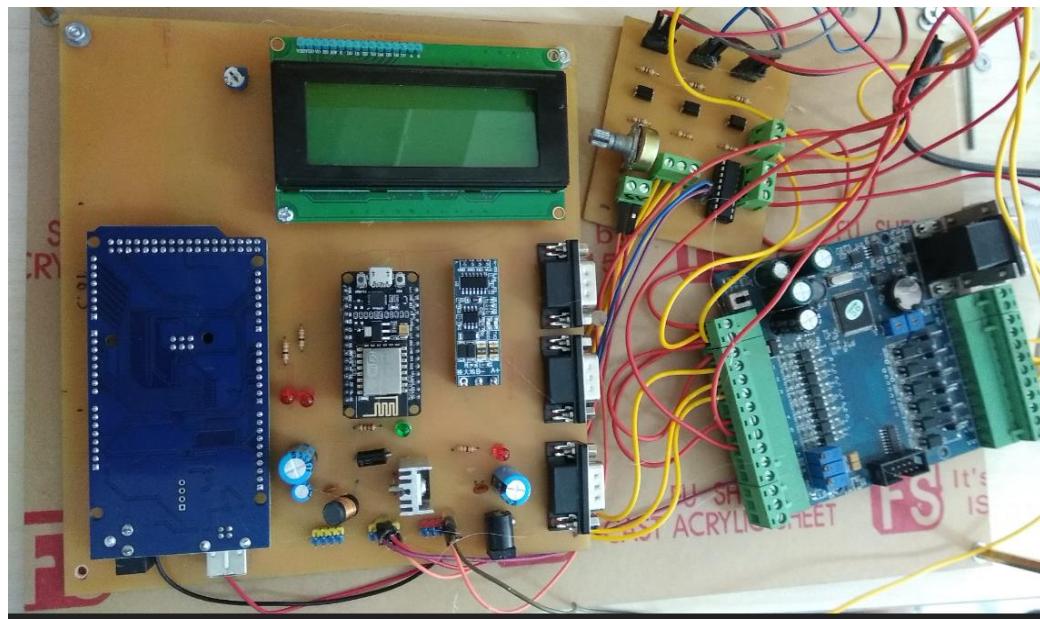
**Hình 4.7 Sơ đồ toàn mạch điều khiển**

Theo hình 4.7 sơ đồ nguyên lý mạch điều khiển chính chứa các khối của hệ thống được đánh số thứ tự như sau: khu vực số 1 là vị trí của khối nguồn 5V, khu vực 2 là vị trí của board Arduino Mega 2560, số 3 là vị trí module Node MCU và bên cạnh là 2 led báo truyền tín hiệu UART cổng số 3. Ở khu vực số 4 là module RS-485 và các ngõ ra DB9 để kết nối với PLC, cuối cùng là khu vực số 5 là khối hiển thị có LCD 20x4.

### 4.3 ĐÓNG GÓI VÀ THI CÔNG MÔ HÌNH

#### 4.3.1 Đóng gói bộ điều khiển

Sau khi thi công, kiểm tra và rắp ráp xong các bo mạch hệ thống, nhóm tiến hành đóng gói bộ điều khiển.

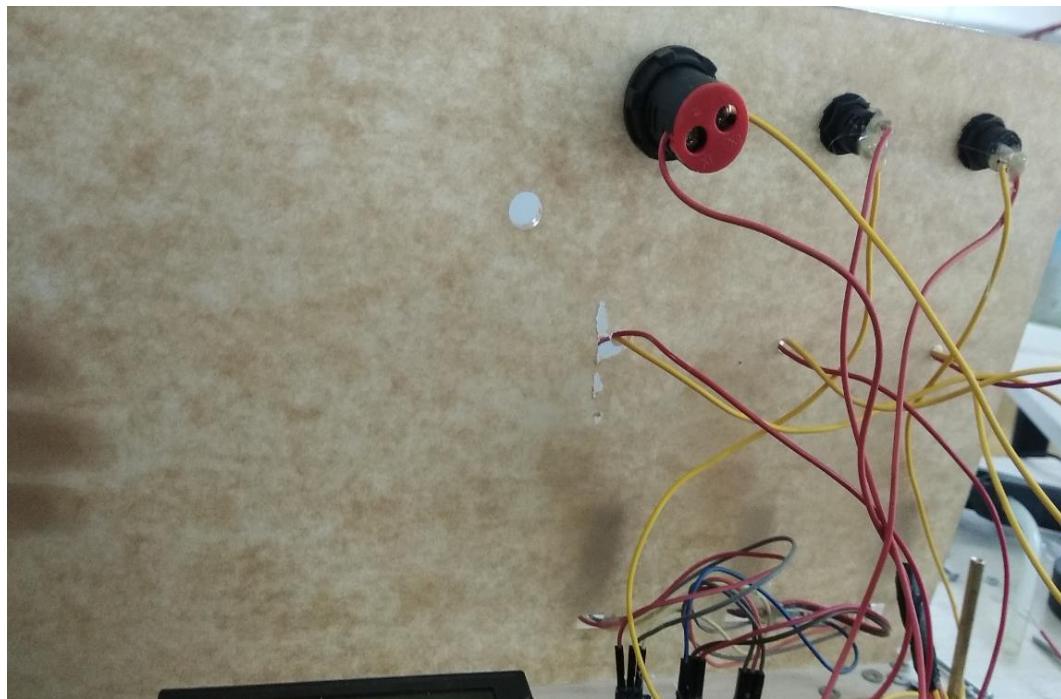


**Hình 4.8** Bộ điều khiển được giữ cố định

Cùng với bộ điều khiển thì bo mạch cảm biến cũng như PLC cũng được cố định trên tấm mica.

### 4.3.2 Thi công mô hình

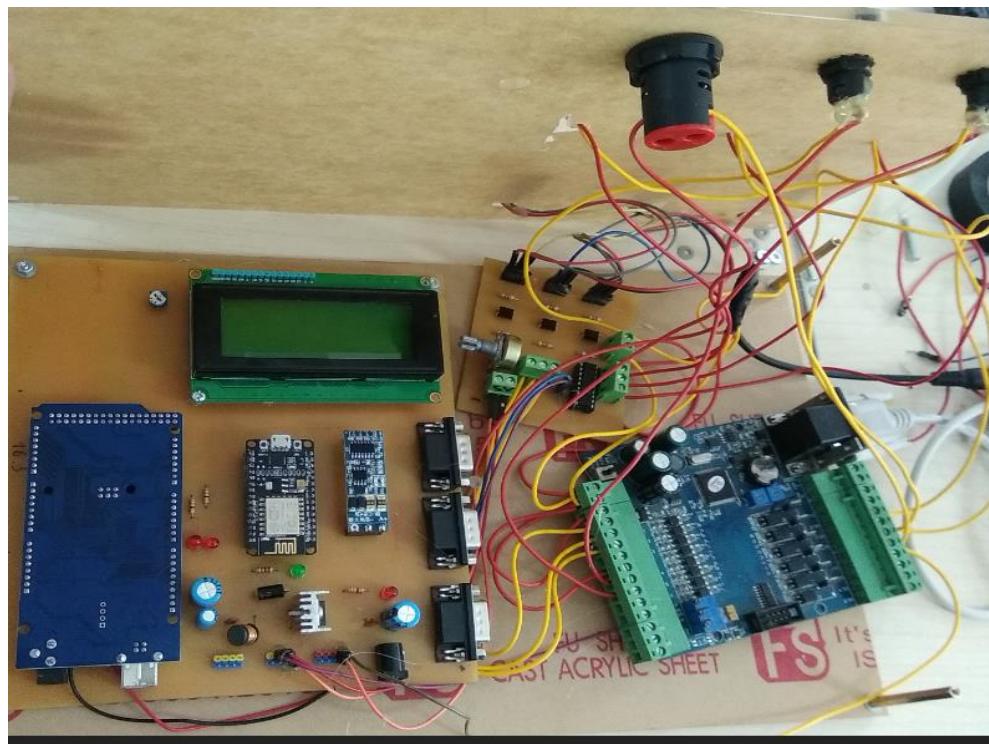
Thi công nắp dậy cho mô hình.



**Hình 4.9** Hình nắp dậy cho hệ thống

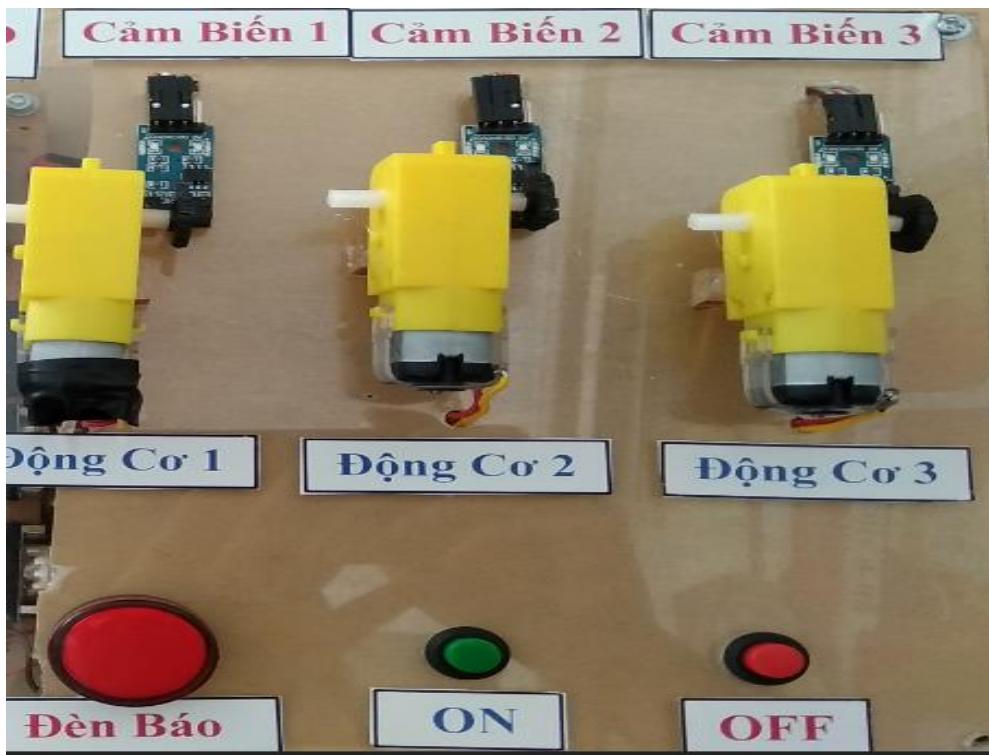
## CHƯƠNG 4. THI CÔNG HỆ THỐNG

---



Hình 4.10 Cân chỉnh trước khi lắp vào

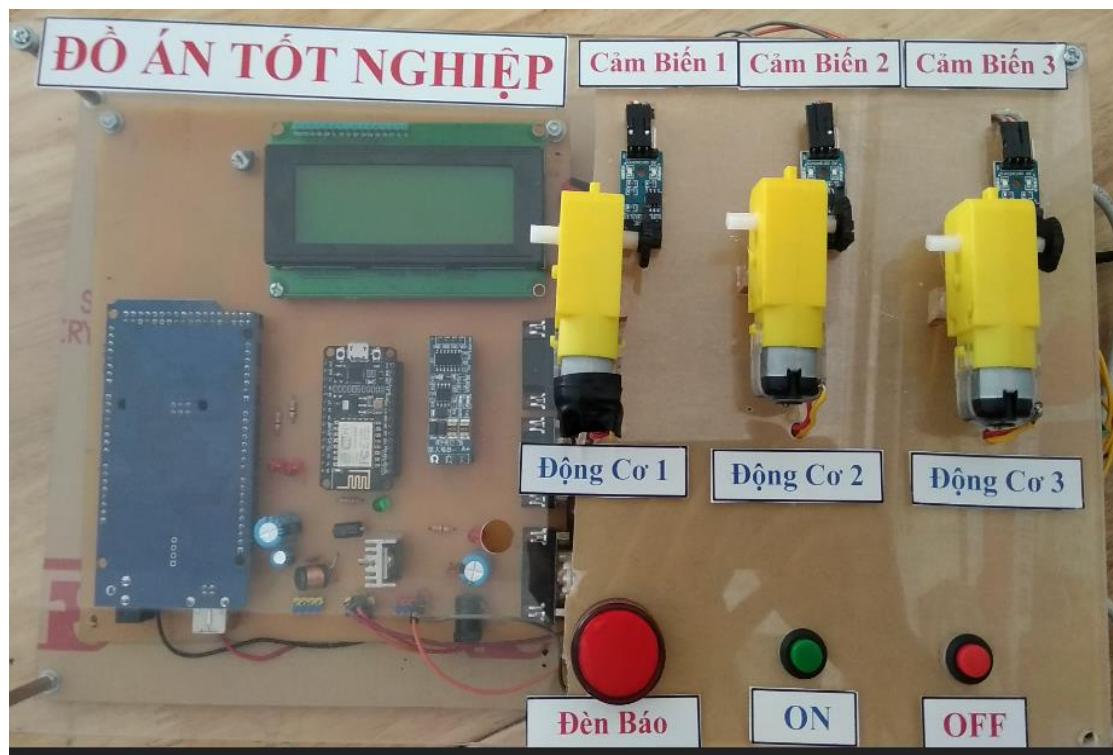
Lắp nút nhấn, đèn, motor và cảm biến vào mặt trước của hệ thống như hình 4.11



Hình 4.11 Lắp thiết bị vào hệ thống

Sau đó rắp hệ thống và cân chỉnh sẽ có mô hình hoàn chỉnh như hình 4.12

## CHƯƠNG 4. THI CÔNG HỆ THỐNG



Hình 4.12 Lắp ráp mô hình hoàn chỉnh

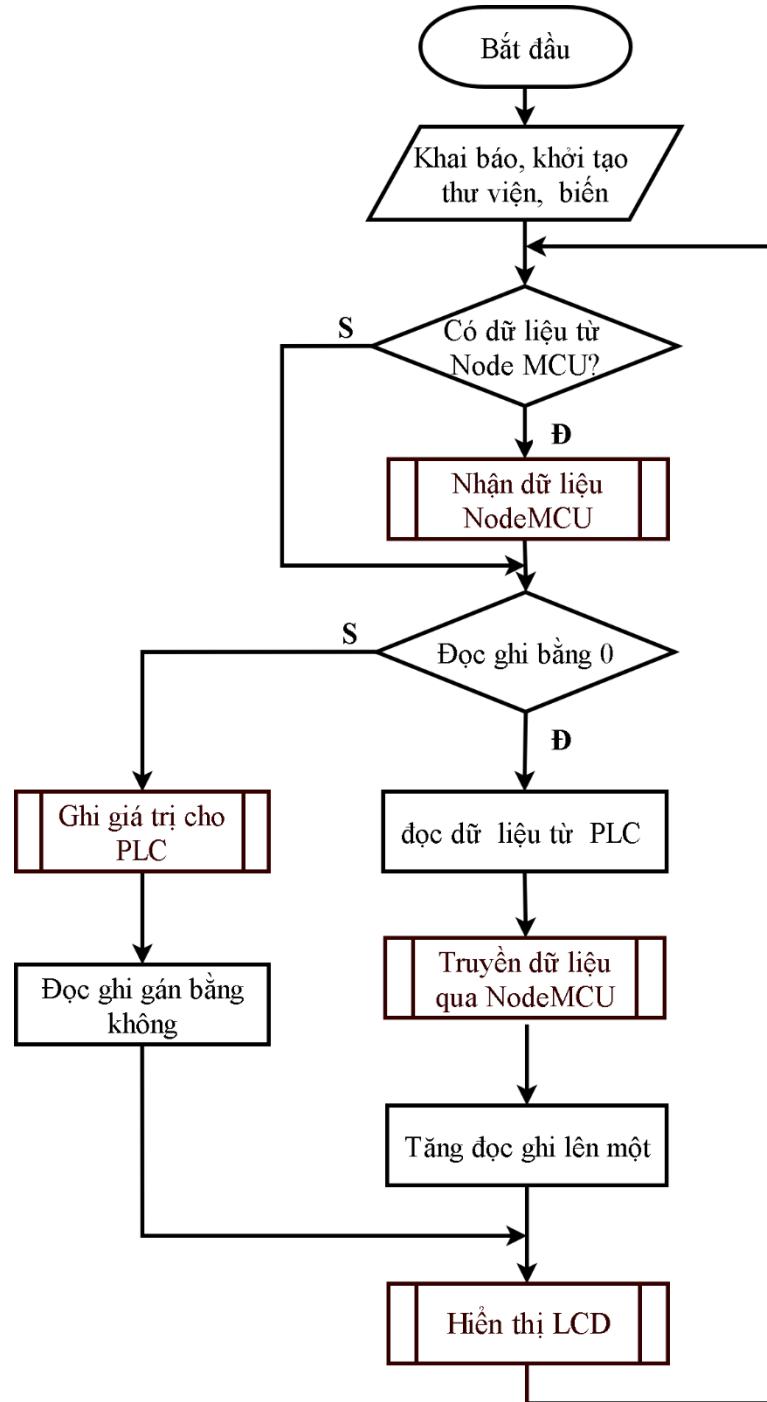
### 4.4 LẬP TRÌNH HỆ THỐNG

#### 4.4.1 Lưu đồ giải thuật

##### a. Lưu đồ cho Arduino

Giải thích lưu đồ hình 4.13: Đầu tiên sẽ khai báo thư viện và chọn cổng giao tiếp modbus, khai báo các biến của chương trình, các chân kết nối LCD. Thiết lập tốc độ truyền, thiết lập các thông số của chuẩn giao tiếp modbus. Kiểm tra trên cổng giao tiếp Serial số 3 có chuỗi dữ liệu truyền từ Node MCU không, nếu có thì thực hiện nhận dữ liệu bằng cách cắt chuỗi và lấy dữ liệu cần thiết. Kiểm tra biến ghi đọc nếu bằng 0 thì thực hiện đọc dữ liệu từ thanh ghi của PLC qua cổng Serial số 1 và gửi dữ liệu cho NodeMCU thông qua cổng Serial số 3 bằng cách gửi một chuỗi dữ liệu có kèm các kí tự đặc biệt để phân biệt khi nhận, sau đó biến ghi đọc tăng lên một. Khi đó ghi đọc bằng 1 thì thực hiện chế độ ghi dữ liệu mà Arduino nhận từ NodeMCU ghi vào PLC qua cổng Serial số 1, sau đó gán biến ghi đọc bằng 0. Tất cả các dữ liệu truyền từ PLC và Node MCU đều được hiển thị trên LCD 20x4. Quá trình được lặp lại liên tục.

## CHƯƠNG 4. THI CÔNG HỆ THỐNG



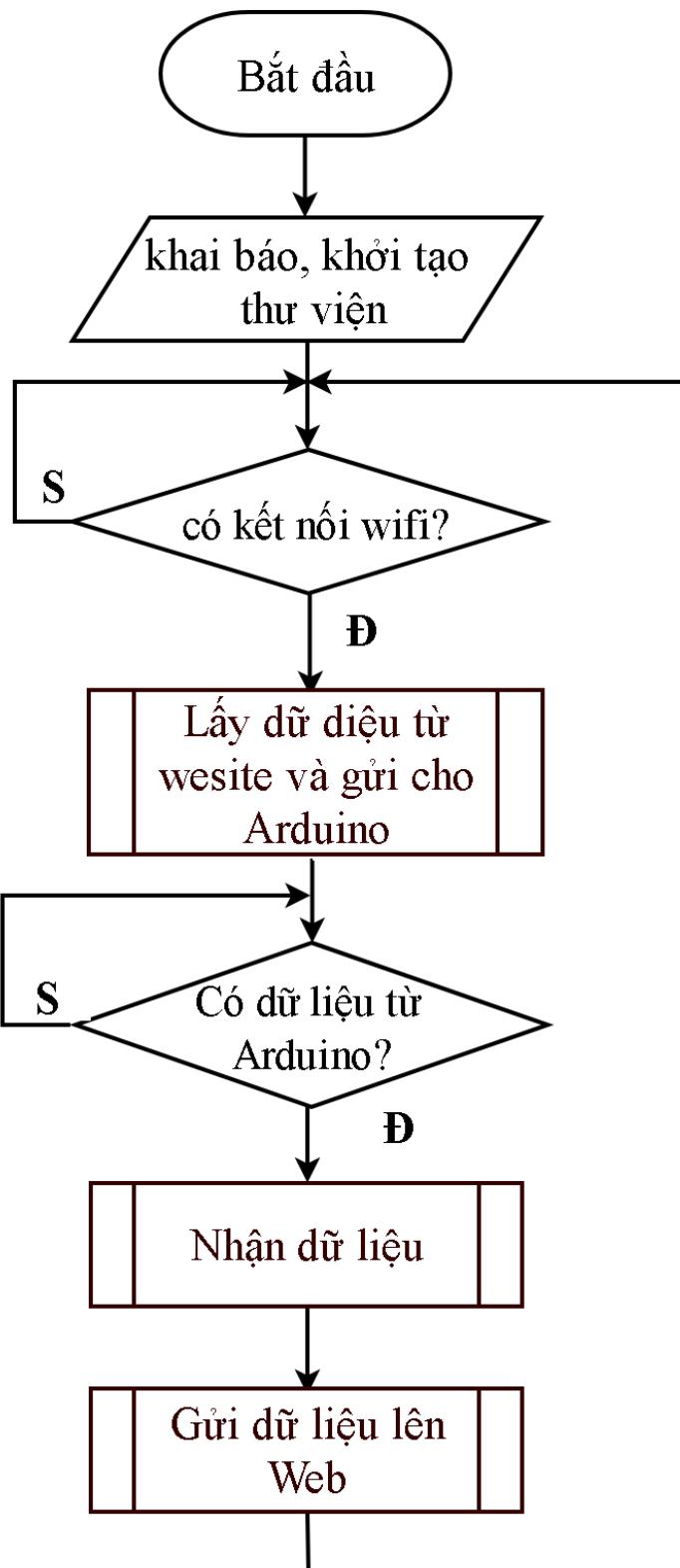
Hình 4.13 Lưu đồ điều khiển cho Arduino

### b .Lưu đồ cho Arduino

Giải thích lưu đồ hình 4.14: Đầu tiên khai báo thư viện cho module Wifi, sau đó kiểm tra có kết nối Wifi. Nếu kết nối Wifi thành công thì led sẽ sáng báo hiệu kết nối thành công. Truy cập vào database của trang web để lấy dữ liệu và truyền cho Arduino bằng cách gửi qua cổng giao tiếp Serial một chuỗi kí tự có cả kí tự đặc biệt. Tiếp theo là kiểm tra trên cổng giao tiếp UART của Node MCU có dữ liệu thì thực hiện nhận chuỗi dữ liệu và cắt chuỗi lấy dữ liệu cần thiết rồi gửi lên

## CHƯƠNG 4. THI CÔNG HỆ THỐNG

Web Server và nó sẽ lưu vào Database của trang Web Server. Quá trình được lặp lại liên tục.



Hình 4.14 Lưu đồ điều khiển cho Node MCU

## CHƯƠNG 4. THI CÔNG HỆ THỐNG

### 4.4.2 Phần mềm lập trình cho vi điều khiển

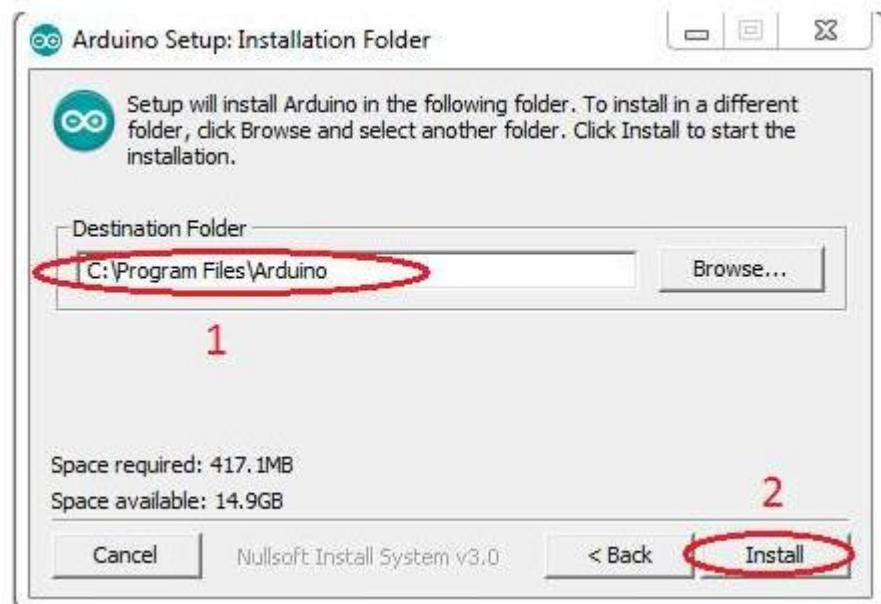
#### a. Giới thiệu phần mềm lập trình

Bước 1: Download Arduino IDE

Bước 2: Cài đặt Arduino IDE

Click chuột phải vào file Arduino-1.8.3-windows vừa tải về và chọn Run as administrator.

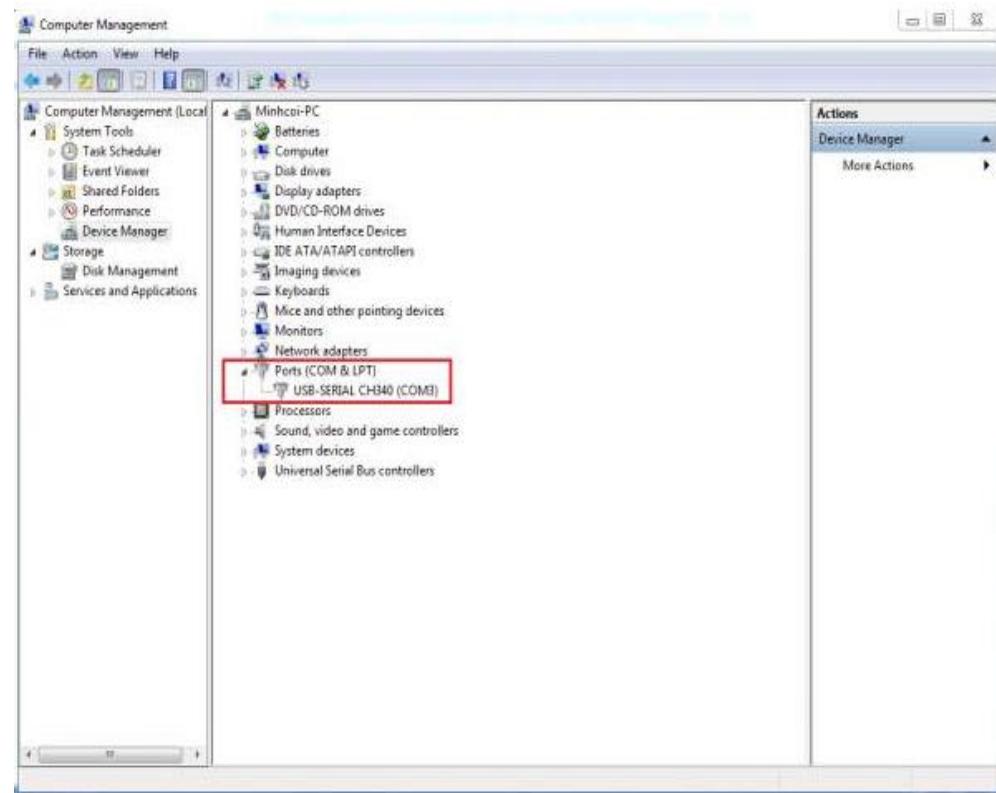
Cửa sổ Arduino Setup: License Agreement xuất hiện, các bạn chọn I Agree để tiếp tục, ngay sau đó cửa sổ Arduino Setup: Installation Options xuất hiện. Tại đây các bạn chú ý đánh dấu tích vào ô Install Usb Driver để cài đặt cả Usb driver cho Arduino và chọn Next để tiếp tục. Cửa sổ Arduino Setup: Installation Folder xuất hiện, các bạn chọn nơi lưu file cài đặt (mặc định là C:\Program File\Arduino) và chọn Next để bắt đầu quá trình cài đặt.



**Hình 4.15** Bắt đầu cài đặt

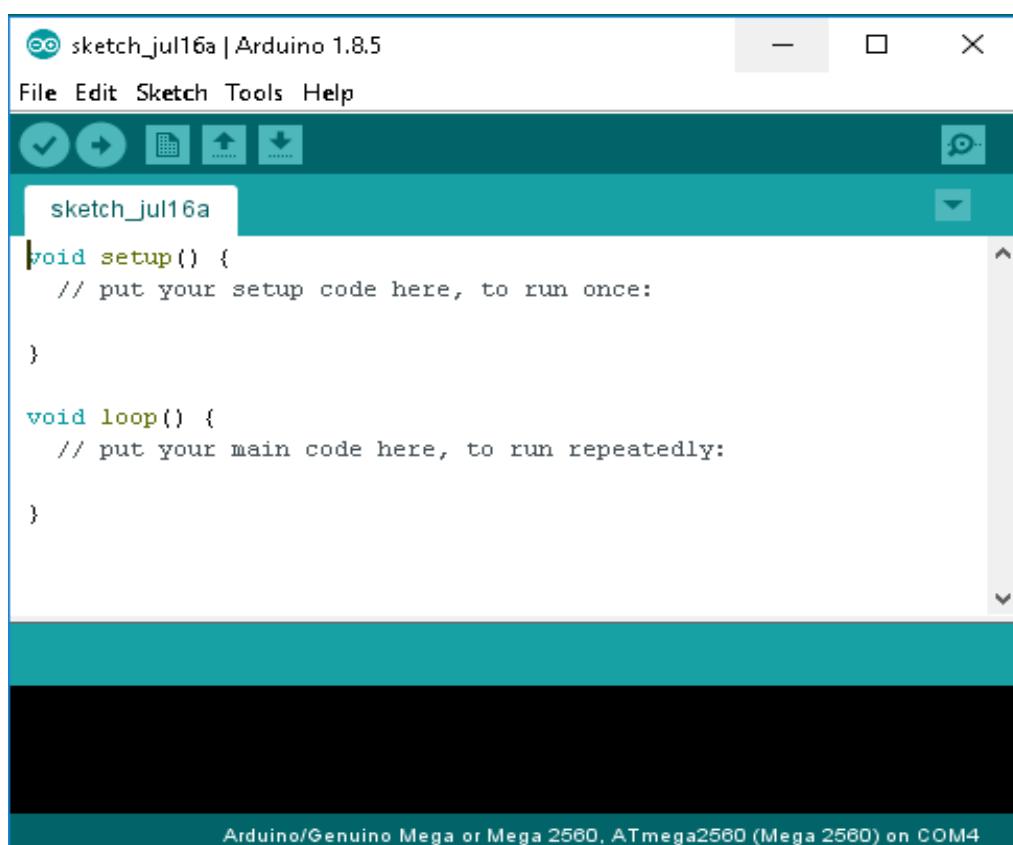
Sau khi cài đặt xong chương trình thì trong phần Manager xuất hiện công giao tiếp như hình sau:

## CHƯƠNG 4. THI CÔNG HỆ THỐNG



Hình 4.16 Kiểm tra lại driver

Sau khi cài đặt xong phần mềm ta có thể lập trình trên đó và tiến hành nạp chương trình điều khiển cho board Arduino thông qua cáp USB.



Hình 4.17 Giao diện phần mềm Arduino IDE

## CHƯƠNG 4. THI CÔNG HỆ THỐNG

### 4.4.3 Phần mềm lập trình cho Web Server.

Đề tài của nhóm chúng em sử dụng web miễn phí và công cụ để lập trình là phần mềm dreamweaver và để chỉnh sửa chúng ta có thể sử dụng phần mềm hoặc ngoài ra chúng ta có thể chỉnh sửa trực tiếp trên web. Sau đây em xin hướng dẫn cách sử dụng phần mềm trên trình duyệt Cốc cốc.

Bước 1: truy cập vào trang <https://vn.000webhost.com>.



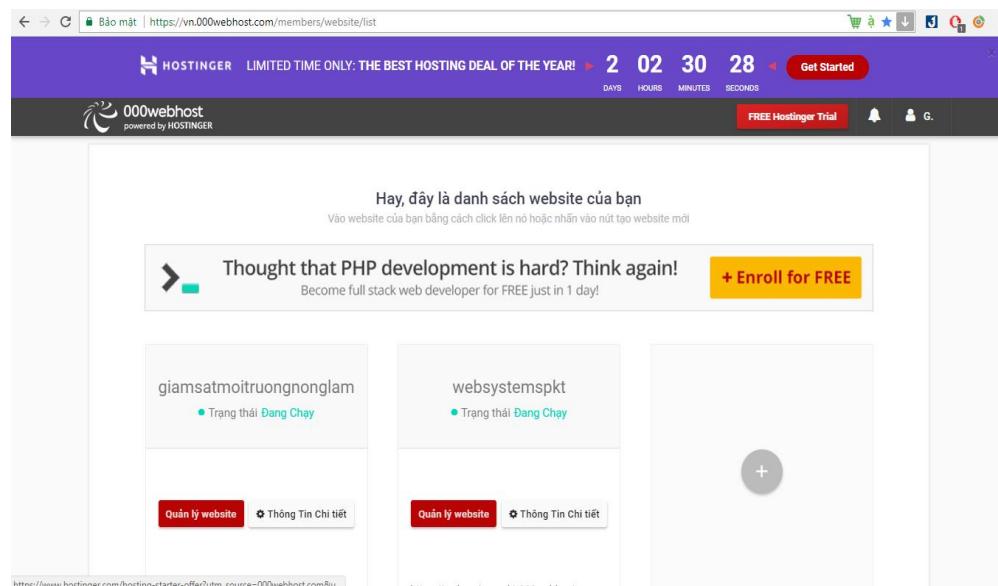
Hình 4.18 Trang chủ của web hosting

Bước 2: Tiến hành đăng nhập vào hệ thống.

Do chúng em đã viết từ trước nên chỉ cần đăng nhập còn nếu chưa có tài khoản thì tiến hành đăng ký trước khi đăng nhập.

Bước 3: Sau khi đăng nhập chúng ta sẽ thấy được những cái website của chúng ta.

## CHƯƠNG 4. THI CÔNG HỆ THỐNG



Hình 4.19 Chọn dự án để sử tiến hành xử lý

Click chuột vào quản lý website của dự án chúng ta (ở đây dự án của chúng em là cái thứ 2, websystemspkt).

Bước 4: Sau khi click vào của số sẽ mở ra như hình bên dưới. Để chỉnh sửa trang web thì chúng ta click vào quản lý file vào được nơi chỉnh sửa chúng ta lại tiếp tục click vào tải file lên bây giờ.

| Name          | Size   | Date                | Permissions |
|---------------|--------|---------------------|-------------|
| .htaccess     | 0.2 kB | 2018-05-06 04:05:00 | -rW-f-r-    |
| about_us.php  | 0.5 kB | 2018-05-14 04:21:00 | -rW-f-r-    |
| add_data.php  | 0.4 kB | 2018-06-10 08:29:00 | -rW-f-r-    |
| cap3.php      | 2.4 kB | 2018-05-31 01:23:00 | -rW-f-r-    |
| cap7.php      | 2.4 kB | 2018-05-31 01:22:00 | -rW-f-r-    |
| chart.php     | 4.1 kB | 2018-05-31 00:43:00 | -rW-f-r-    |
| control.php   | 6.1 kB | 2018-06-26 06:42:00 | -rW-f-r-    |
| dangky.php    | 1.6 kB | 2018-06-08 02:46:00 | -rW-f-r-    |
| dangnhap.php  | 2.5 kB | 2018-05-06 07:47:00 | -rW-f-r-    |
| dbconnect.php | 0.5 kB | 2018-05-14 03:57:00 | -rW-f-r-    |
| dc1.json      | 0.1 kB | 2018-06-08 02:31:00 | -rW-f-r-    |
| footer.php    | 0.1 kB | 2018-05-14 04:23:00 | -rW-f-r-    |
| getdata.php   | 0.4 kB | 2018-06-11 06:41:00 | -rW-f-r-    |
| getdata1.php  | 0.4 kB | 2018-06-06 08:08:00 | -rW-f-r-    |
| giamsat.php   | 3.3 kB | 2018-06-10 08:45:00 | -rW-f-r-    |
| header.php    | 1.3 kB | 2018-05-21 07:11:00 | -rW-f-r-    |

Hình 4.20 Trang lập trình trực tiếp

Tất cả những file lập trình đều nằm trong đây, bây giờ muốn chỉnh sửa chúng ta chỉ cần nhấp đôi vào thư mục cần chỉnh sửa.

## CHƯƠNG 4. THI CÔNG HỆ THỐNG

The screenshot shows a red header bar with the text "Edit file" and a close button "X". Below the header is a code editor window containing PHP code. The code is as follows:

```
/public_html/add_data.php

1 <?php
2
3     include("dbconnect.php");
4
5     $date= date("Y-m-d H:i:s");
6     // Prepare the SQL statement
7
8     $SQL = "INSERT INTO myproject(temp,hum,sp3,may2,datetimetet) VALUES ('".$_GET["temp"]."',
9                                         '".$_GET["hum"]."','".$_GET["sp3"]."','"
10                                        ."$_GET["may2"]."','".$date"')";
11
12     $result = mysqli_query($conn, $SQL);
13     echo "ok!";
14 ?>
```

At the bottom right of the code editor are two buttons: "SAVE & CLOSE" and a red "SAVE" button.

**Hình 4.21** Có thẻ chỉnh sửa trực tiếp trên Website

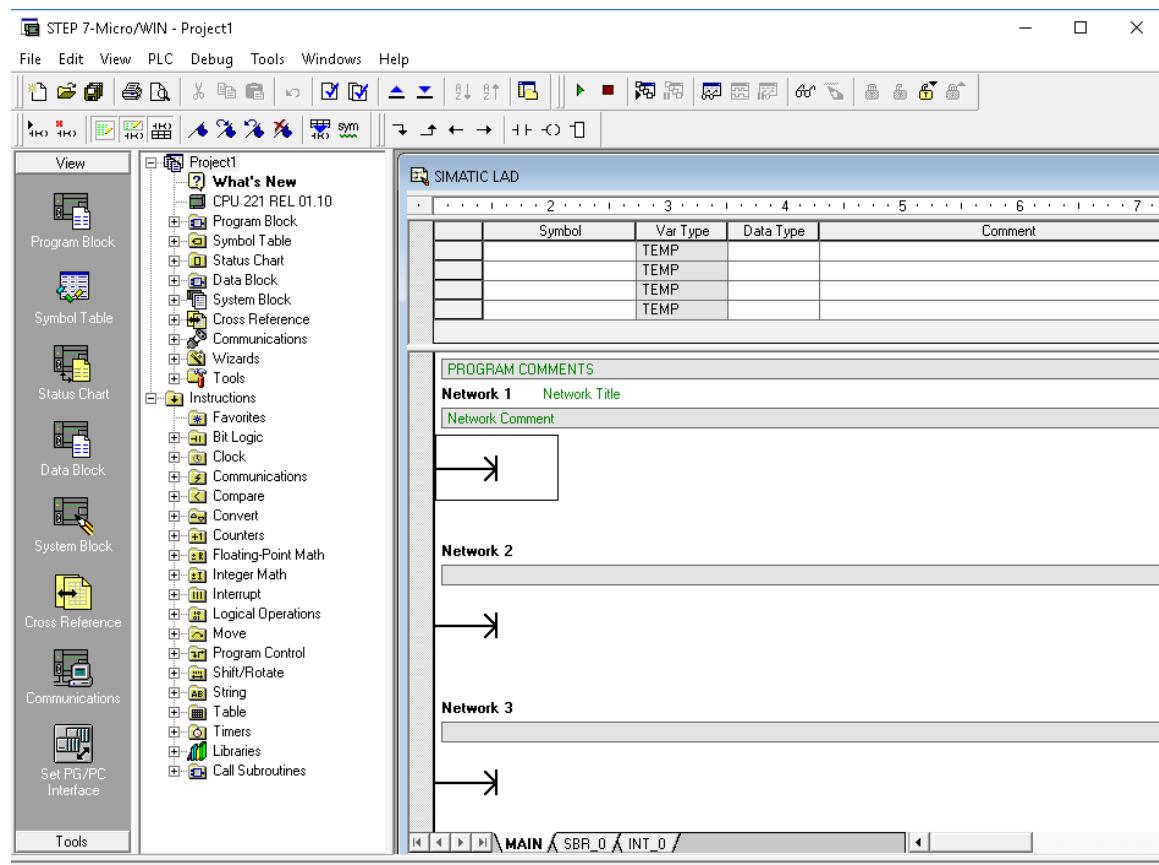
Khi chỉnh sửa xong hoặc tạo thêm chúng ta chỉ cần click vào save thì trang web của chúng ta sẽ tự động cập nhật.

Đó là giới thiệu của chúng em về cách sử dụng trang web, nếu có gì thắc mắc thì liên hệ trực tiếp chúng em, chúng em sẽ hỗ trợ hết mình.

### 4.4.3 Phần mềm lập trình cho PLC (Step7- Micro/Win).

Trong đê tài có sử dụng phần mềm Step7-Micro/Win để lập trình cho PLC S7-200. Tương tự như trên thì chúng ta truy cập vào trang chủ và tiến hành cài đặt phần mềm. Sau khi cài đặt thì phần mềm có giao diện như hình 4.22

## CHƯƠNG 4. THI CÔNG HỆ THỐNG



Hình 4.22 Giao diện phần mềm Step7-Micro/Win

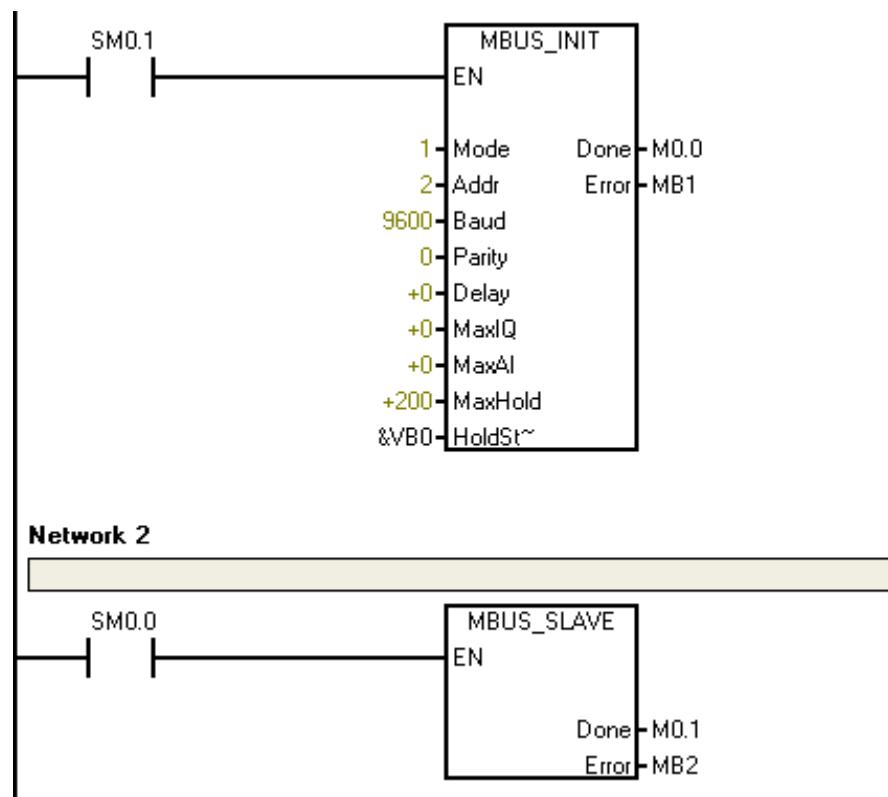
Chúng ta viết chương trình cho PLC theo ngôn ngữ Labview là ngôn ngữ dùng các khối kết nối với nhau để lập trình thường dùng để lập trình cho PLC.

### 4.5 LẬP TRÌNH MÔ PHỎNG

Trong phần này nhóm thực hiện mô phỏng chuẩn giao tiếp modbus qua phần mềm Modbus Poll để thực hiện lệnh ghi và đọc dữ từ PLC.

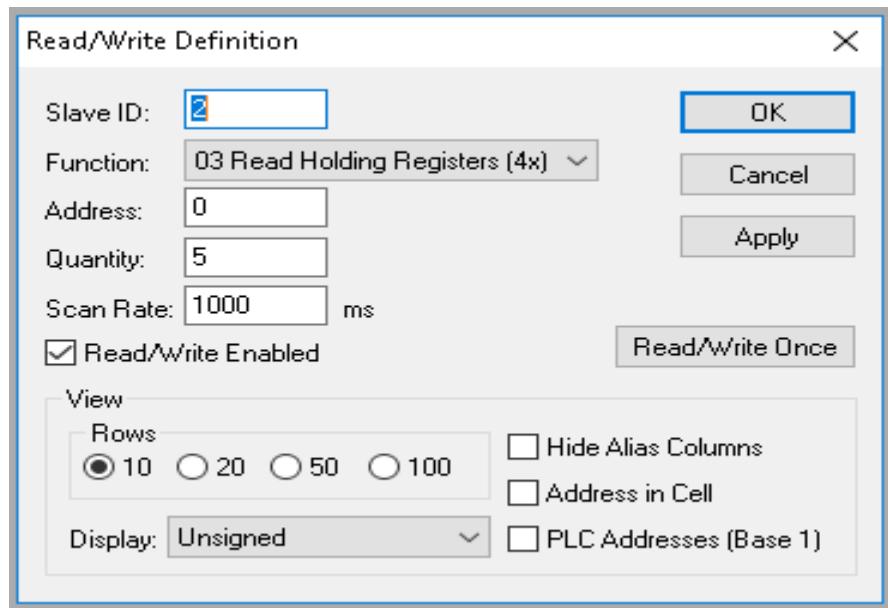
Trước tiên chúng ta khởi tạo modbus cho PLC đóng vai trò là Slave với địa chỉ là 2, tốc độ là 9600 baud, dữ liệu trả về dạng byte.

## CHƯƠNG 4. THI CÔNG HỆ THỐNG



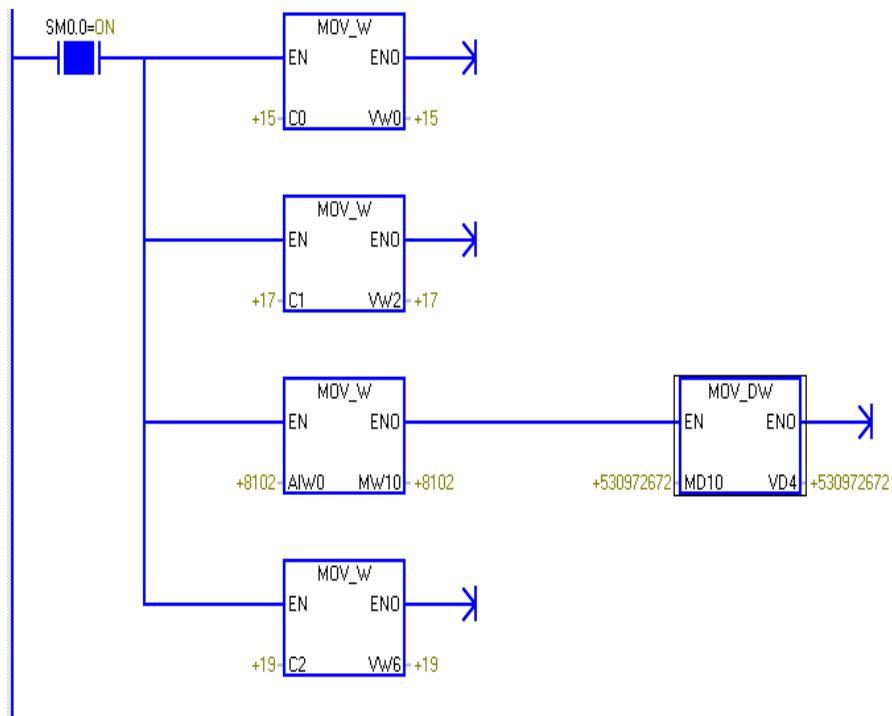
Hình 4.23 Khởi tạo Modbus cho PLC

Tiếp theo sử dụng phần mềm Modbus Poll thực hiện lệnh đọc từ thanh ghi bắt đầu từ địa chỉ 0 và đọc 5 byte như hình 4.24



Hình 4.24 Thực hiện lệnh đọc dữ liệu từ PLC

## CHƯƠNG 4. THI CÔNG HỆ THỐNG



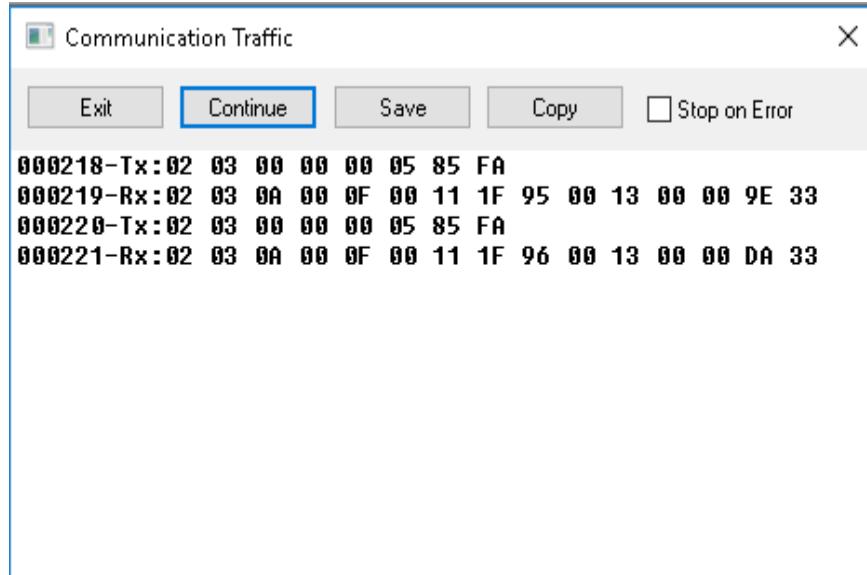
Hình 4.25 Các giá trị ô nhớ của thanh ghi trên PLC

Kết quả trả về từ lệnh đọc các giá trị từ được hiển thị như hình 4.26

|   | Alias | 00000 |
|---|-------|-------|
| 0 |       | 15    |
| 1 |       | 17    |
| 2 |       | 8086  |
| 3 |       | 19    |
| 4 |       | 0     |
| 5 |       |       |
| 6 |       |       |
| 7 |       |       |
| 8 |       |       |
| 9 |       |       |

Hình 4.26 Kết quả trả về của lệnh đọc

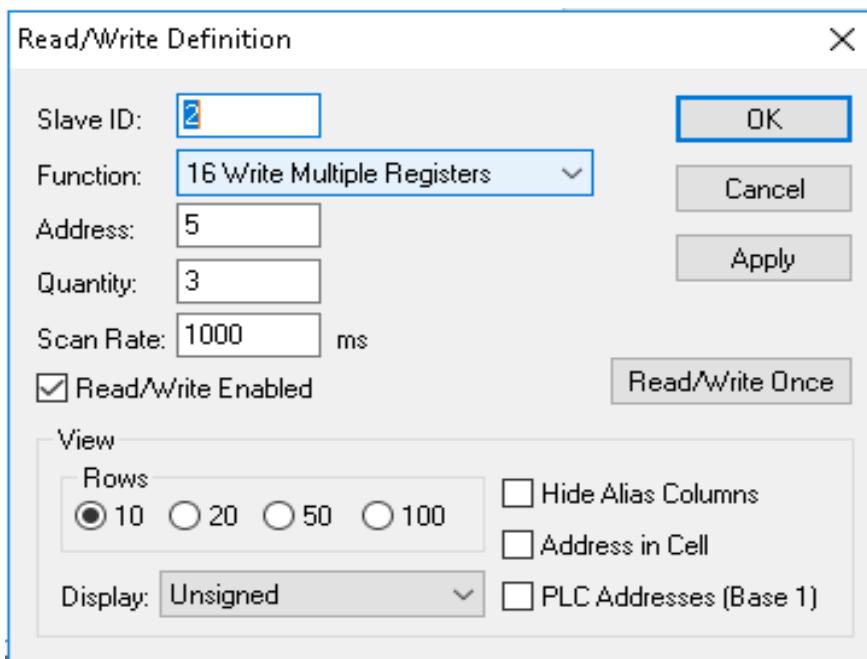
## CHƯƠNG 4. THI CÔNG HỆ THỐNG



**Hình 4.27** Quá trình yêu cầu và phản hồi của Master và Slave với lệnh đọc

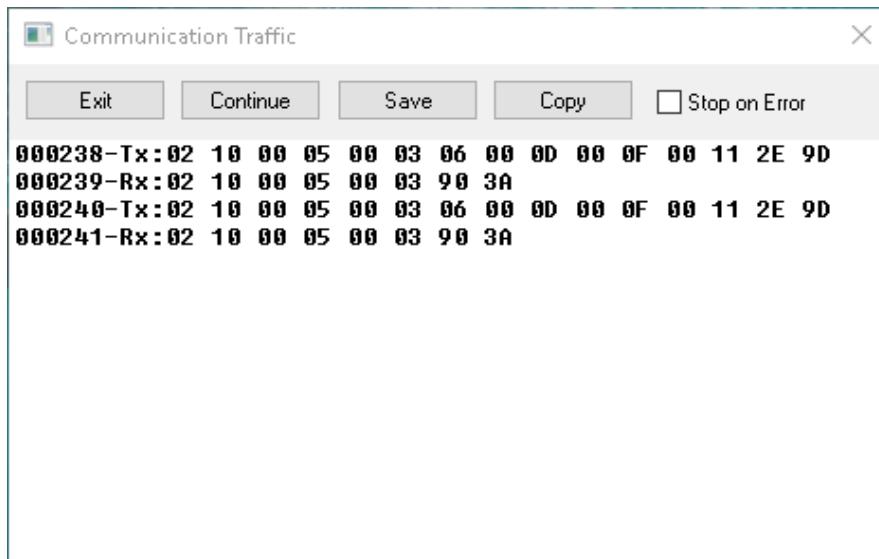
Theo như hình 4.27 thì Master sẽ gửi yêu cầu là lệnh đọc dữ liệu thanh ghi (03) từ địa chỉ Slave số 2. Với địa chỉ bắt đầu là 00 và đọc đi 5 byte, cuối cùng kiểm tra lỗi CRC. Phản hồi mà Slave trả lại địa chỉ Slave là 02, mã hàm 03, tổng số byte trả về là 10 byte (0A) và tiếp theo là các dữ liệu trả về của các ô nhớ theo thứ tự là byte cao trước và byte thấp sau và cuối cùng là mã kiểm tra lỗi CRC.

Tương tự chúng ta cũng thực hiện lệnh ghi dữ liệu vào thanh ghi của PLC. Thực hiện ghi với địa chỉ bắt đầu là 05 và ghi 3 byte liên tiếp.

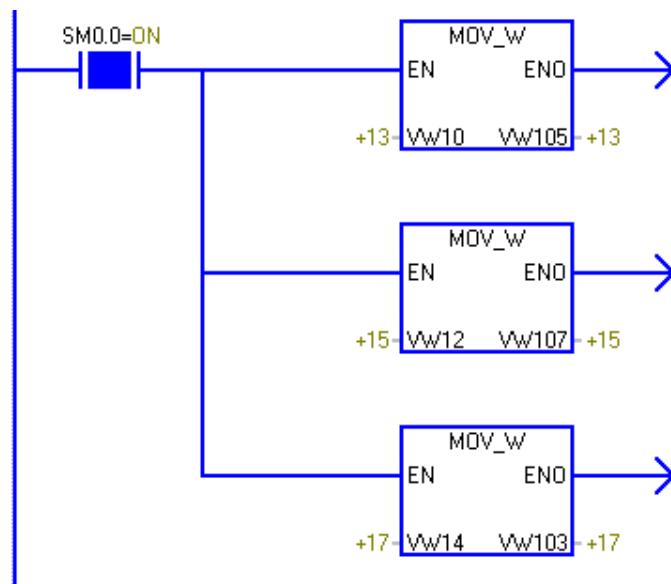


**Hình 4.28** Thực hiện lệnh ghi dữ lệnh vào PLC

## CHƯƠNG 4. THI CÔNG HỆ THỐNG



Hình 4.29 Quá trình yêu cầu và phản hồi của Master và Slave với lệnh ghi



Hình 4.30 Kết quả các dữ liệu được ghi vào PLC

## 4.6 VIẾT TÀI LIỆU HƯỚNG DẪN SỬ DỤNG, THAO TÁC

### 4.6.1 Viết tài liệu hướng dẫn sử dụng

Sau đây chúng em sẽ hướng dẫn cách sử dụng mô hình.

**Bước 1:** Cấp nguồn cho hệ thống, hệ thống sử dụng nguồn là 220V AC, khi cấp nguồn và mở công tắc nguồn thì đèn báo hiệu có điện sáng lên, và đợi trong giây lát để hệ thống tự động kết nối với Wifi.

## **CHƯƠNG 4. THI CÔNG HỆ THỐNG**

---

**Bước 2:** Mở trang web lên và đăng nhập vào hệ thống (có thể sử dụng laptop, smartphone... có kết nối Internet).

**Bước 3:** Nhập số liệu đã tính toán từ trước trên website (lưu ý phải nhập đầy đủ các loại sản phẩm, trường hợp không có sản phẩm thì nhập số 0, nếu bỏ trống hệ thống sẽ báo là chúng ta phải nhập lại). Sau khi đã nhập xong thì chúng ta tiến hành gửi dữ liệu bằng nút gửi.

**Bước 4:** Kiểm tra LCD trên bộ xử lý đã hiển thị giá trị đã nhập chưa, nếu có thì kiểm tra xem có đúng với giá trị vừa nhập chưa (trường hợp mà chúng ta ở xa bộ điều khiển trung tâm thì chúng ta có thể vào Database của hệ thống để xem dữ liệu đã nhập vào chưa).

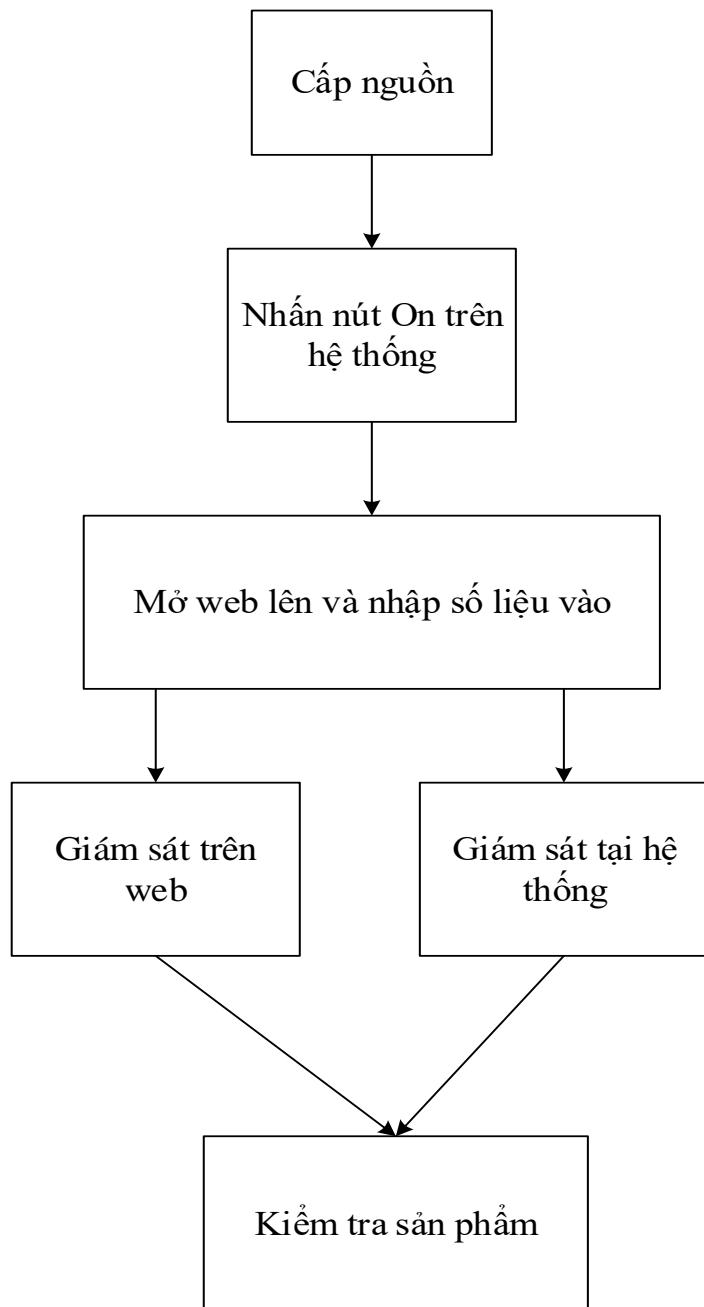
**Bước 5:** Nếu tất cả đã xong thì chúng ta sẽ đi qua thư mục View để giám sát xem giá trị gửi lên từ bên dưới có hợp lý không (trường hợp nếu sai quá nhiều so với mức cho phép chúng ta có thể cho dừng hệ thống và tiến hành kiểm tra).

### **4.6.2 Quy trình thao tác**

Giải thích: khi cấp nguồn hệ thống sẽ kết nối với Wifi đã kết nối trước, sau đó ta đăng nhập vào trang web và nhập vào số sản phẩm vào hệ thống. Sau khi nhập vào thì giá trị đó được gửi vào hệ thống hiển thị trên LCD, sau đó hệ thống đã sẵn sàng để hoạt động, trong quá trình hoạt động thì chúng ta có thể giám sát trực tiếp trên hệ thống hoặc thông qua Web Server.

## CHƯƠNG 4. THI CÔNG HỆ THỐNG

---



Hình 4.31 Sơ đồ Quy trình thao tác

## **Chương 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ**

### **5.1 GIỚI THIỆU**

Trong chương này nhóm chúng em sẽ trình bày kết quả của cả quá trình nghiên cứu làm đề tài trong khoảng thời gian 15 tuần. Bên cạnh đó là nhận xét cùng với những đánh giá và đề xuất hướng phát triển của sản phẩm mô hình để hoàn thiện và có thể áp dụng vào thực tế.

### **5.2 KẾT QUẢ ĐẠT ĐƯỢC**

Qua quá trình nghiên cứu và thực hiện đề tài, nhóm chúng em nhận thấy được đề tài có khả năng ứng dụng vào thực tiễn cao, có thể giám sát trực tiếp trên Web Sever và có thể nhập hệ thống trực tiếp trên web mà không cần phải đến nhà máy trực tiếp, đáp ứng được nhu cầu xã hội hiện đại ngày nay và đi đúng với xu hướng của thời đại. Đồng thời đề tài này cũng là một nguồn tài liệu có giá trị cho các bạn sinh viên những khóa tiếp theo có thể tham khảo khi nghiên cứu những đề tài có liên quan cũng như là phát triển thêm đề tài từ nền tảng có sẵn mà nhóm đã nghiên cứu. Bên cạnh đó, nhóm chúng em cũng bổ sung cho mình những kiến thức hay và bổ ích.

### **5.3 KẾT QUẢ THỰC NGHIỆM**

Cáp nguồn cho hệ thống, và chờ đợi trong giây lát cho thiết bị kết nối với Wifi. Mở trình duyệt web lên và nhập link:

<http://websystemspkt.000webhostapp.com/>. Giao diện mở ra bên dưới

## CHƯƠNG 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

The screenshot shows the homepage of the website. At the top left is the URL: websystemspkt.000webhostapp.com. At the top right are icons for a shopping cart, user profile, and other site functions. The header features the logo of HCMUTE (Hochiminh City University of Technology) with the text "TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH" and "BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP - Y SINH". Below the logo, it says "Khoa Điện - Điện Tử". A horizontal menu bar at the bottom includes "TRANG CHỦ" (selected), "ĐĂNG KÝ", "ĐĂNG NHẬP", and "LIÊN HỆ".

**GIỚI THIỆU**

Ngày nay, với sự phát triển của khoa học kỹ thuật đã làm cho cuộc sống con người ngày càng tốt hơn, hiện đại hơn. Trong công nghiệp cũng vậy, với sự phát triển của khoa học các thiết bị trong nhà máy được hoạt động một cách tự động và việc giám sát, điều khiển đều thực hiện từ xa.

Với sự xuất hiện của công nghệ 4.0 đi đầu công nghệ IoT được phát triển mạnh mẽ. Việc ứng dụng IoT vào hệ thống giám sát và điều khiển nhà máy giúp điều khiển nhanh chóng và dễ dàng, phát hiện sự cố nhanh, giảm nhân công từ đó mang lại hiệu quả kinh tế đáng kể . Các giải pháp ứng dụng dần được hình thành và phát triển. Cho nên nhóm đề xuất nghiên cứu đề tài : "Ứng dụng công nghệ IOT để thiết kế hệ thống giám sát và điều khiển thiết bị công nghiệp".

Trong đề tài này, các thiết bị trong hệ thống sẽ được điều khiển và giám sát bằng webserver hoặc app điện thoại thông qua mạng wifi. Sử dụng PLC giao tiếp với board Arduino theo chuẩn công nghiệp thông qua mô đun RS485, Arduino sẽ kết nối với mô đun wifi ESP8266 để đưa dữ liệu lên website. PLC sẽ nhận lệnh điều khiển từ website thông qua kết nối với Arduino và điều khiển các thiết bị. Việc sử mô đun ESP8266 giao tiếp với Arduino giúp người dùng có thể truy cập vào website bằng mạng Internet, không gian lưu trữ dữ liệu trên website lớn, có thể điều khiển thiết bị mọi lúc mọi nơi thay vì sử dụng SQL.

**Hình 5.1 Trang chủ của trang web**

Sau khi truy cập vào trang web nếu người sử dụng chưa có tài khoản thì phải đăng ký và chờ xét duyệt.

The screenshot shows the registration page. At the top left is the URL: websystemspkt.000webhostapp.com/dangky.php. At the top right are icons for a shopping cart, user profile, and other site functions. The header features the logo of HCMUTE and the text "TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH" and "BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP - Y SINH". Below the logo, it says "Khoa Điện - Điện Tử". A horizontal menu bar at the bottom includes "TRANG CHỦ" (selected), "ĐĂNG KÝ", "ĐĂNG NHẬP", and "LIÊN HỆ".

**Form đăng ký**

Username :

Password :

Nhập lại password :

Họ tên :

**Hình 5.2 Trang đăng ký tài khoản sử dụng cho người dùng**

Ở đây người dùng bắt buộc phải điền đầy đủ thông tin của người dùng, tất cả thông tin đăng ký được đưa vào trong Database ở đây quản lý có thể xem cũng như có cho phép người đó tiếp tục đăng nhập vào hệ thống hay không.

## CHƯƠNG 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

The screenshot shows the phpMyAdmin interface for a database named 'id5643192\_database'. The left sidebar lists databases like 'Mới', 'logging', 'myproject', 'servertoPLC', 'servertoPLC1', and 'signin'. The 'signin' database is selected. The main area displays the 'signin' table with two rows of data:

|                          | Tùy chọn   | user_id | username | password   | password2  | fullname    | createdate          |
|--------------------------|--|---------|----------|------------|------------|-------------|---------------------|
| <input type="checkbox"/> | <input type="checkbox"/> Sửa <input type="checkbox"/> Chép <input type="checkbox"/> Xóa bỏ | 8       | linh     | Linh123123 | Linh123123 | vân linh    | 2018-05-06 18:34:12 |
| <input type="checkbox"/> | <input type="checkbox"/> Sửa <input type="checkbox"/> Chép <input type="checkbox"/> Xóa bỏ | 9       | conguy   | uyanh      | uyanh      | huynhconguy | 2018-05-07 21:28:46 |

**Hình 5.3 Thông tin đăng nhập và tên người dùng được lưu trữ**

Sau khi đăng ký xong thì chúng ta tiếp tục đăng nhập vào hệ thống.

Trong cơ sở dữ liệu trên thì có 4 bảng để quản lý:

- Logging: lưu trữ thông tin đăng nhập và mật khẩu của người tham gia điều khiển giám sát.
- Myproject: chứa thông tin cảm biến nằm được hệ thống bên dưới gửi lên.
- ServertoPLC: chứa các thông số mà trên web gửi xuống.
- ServertoPLC1: chứa các thông số các hàng cần hiển thị bên khu vực giám sát.

## CHƯƠNG 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

websystemspt.000webhostapp.com/dangnhap.php

TRƯỜNG ĐẠI HỌC  
SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH  
**BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP - Y SINH**  
Khoa Điện - Điện Tử

HCMUTE

Đăng nhập

|  |       |
|--|-------|
| Username                                 | linh  |
| Password                                 | ***** |
| <input type="button" value="Đăng nhập"/> |       |

**Hình 5.4 Giao diện đăng nhập của hệ thống**

Sau khi đăng nhập đúng tài khoản và mật khẩu của mình thì sẽ được chuyển đến trang điều khiển chính.

TRƯỜNG ĐẠI HỌC  
SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH  
**BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP - Y SINH**  
Khoa Điện - Điện Tử

HCMUTE

|                                    |
|------------------------------------|
| Nhập sản phẩm                      |
| Sản phẩm 1 :                       |
| Sản Phẩm 2 :                       |
| Sản Phẩm 3 :                       |
| <input type="button" value="gửi"/> |

|   |
|---|
| Nhập số lượng hàng cần kiểm tra         |
| dây chuyền 1 :                          |
| <input type="button" value="Xác nhận"/> |

**Hình 5.5 Giao diện chính của trang điều khiển**

Ở trang này chúng ta có thể nhập các giá trị của sản phẩm đã được tính toán từ trước, ở đây chúng em làm mô phỏng 3 cảm biến nên có 3 loại sản phẩm, cái này có thể được thay đổi khi áp dụng vào mô hình thật. Sau khi nhập xong thì chúng ta nhấn nút gửi (hoặc enter trên bàn phím). Lưu ý là chúng ta phải nhập đầy đủ thông tin của cả 3 sản phẩm nếu không có thì nhập số 0 và không để trống

## CHƯƠNG 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

hệ thống sẽ báo là chúng ta nhập lại, nếu nhập đầy đủ và nhấn gửi rồi thì hệ thống sẽ gửi lại lại dòng chữ “chúc mừng bạn đã nhập sản phẩm thành công”.

TRƯỜNG ĐẠI HỌC  
SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH  
BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP - Y SINH  
Khoa Điện - Điện Tử

HCMUTE

ĐĂNG XUẤT      GIÁM SÁT      VIEW      CHART

Nhập sản phẩm

|              |    |
|--------------|----|
| Sản Phẩm 1 : | 25 |
| Sản Phẩm 2 : | 20 |
| Sản Phẩm 3 : | 15 |

Nhập số lượng hàng cần kiểm tra

dây chuyền 1 :

**Hình 5.6 Nhập số sản phẩm xuống hệ thống bên dưới**

Chúng ta cũng có LCD để giám sát khi hệ thống chuyển số xuống chúng ta sẽ nhận được giá trị trên LCD. Các giá trị nhập trên giao diện Website thông qua ESP8266 đọc dữ liệu về Arduino sau đó thông qua module RS485-TTL truyền xuống PLC thông qua giao thức MODBUS-RTU.

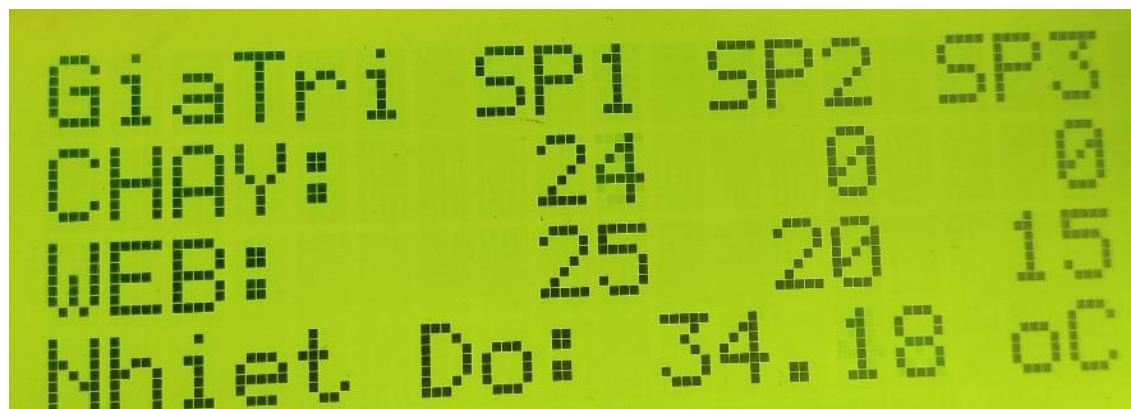
Ở đây có dòng số hàng kiểm tra là phần giám sát sẽ xuất hiện bao nhiêu hàng tương đương với số hàng nhập với bên trang này. Nó sẽ tiện cho việc kiểm tra và tránh quá tải cho trang web.



**Hình 5.7 Giá trị cài đặt hiển thị ở hệ thống trung tâm**

Sau đó ta nhấn nút Start ở hệ thống, hệ thống sẽ bắt đầu chạy.

## CHƯƠNG 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ



Hình 5.8 Giá trị hiển thị ở LCD ở sản phẩm 1

[websystemspkt.000webhostapp.com/view2.php](http://websystemspkt.000webhostapp.com/view2.php)

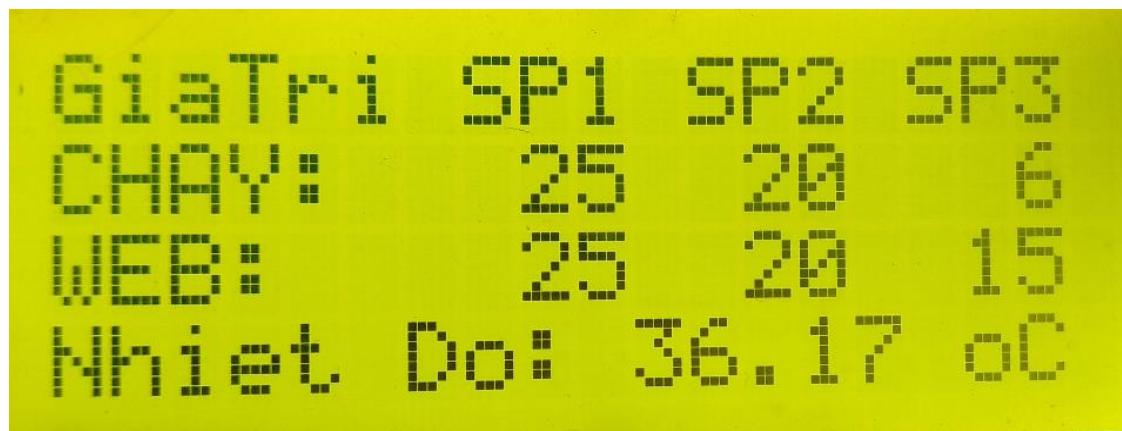
The screenshot shows a website for the Faculty of Electrical Engineering - Industrial Electronics at HCMUTE. The logo features two hands holding a gear and a flame. The main text reads: TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH, BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP - Y SINH, Khoa Điện - Điện Tử. Below the logo are three buttons: ĐĂNG XUẤT, ĐIỀU KHIỂN, and GIÁM SÁT. A table titled 'Bảng giám sát dây chuyền 1' displays data from July 13, 2018, with columns for Time, Product Size, and Temperature.

| Thời gian           | Sản phẩm nhỏ | Sản phẩm vừa | Sản phẩm lớn | Nhiệt độ |
|---------------------|--------------|--------------|--------------|----------|
| 2018-07-13 16:19:27 | 24           | 0            | 0            | 32.22    |
| 2018-07-13 16:19:24 | 20           | 0            | 0            | 32.31    |
| 2018-07-13 16:19:20 | 16           | 0            | 0            | 32.17    |
| 2018-07-13 16:19:17 | 12           | 0            | 0            | 35.57    |
| 2018-07-13 16:19:14 | 9            | 0            | 0            | 35.43    |
| 2018-07-13 16:19:10 | 5            | 0            | 0            | 35.42    |
| 2018-07-13 16:19:07 | 1            | 0            | 0            | 35.46    |
| 2018-07-13 16:19:04 | 10           | 20           | 30           | 34.29    |
| 2018-07-13 16:19:01 | 10           | 20           | 30           | 33.77    |
| 2018-07-13 16:18:57 | 10           | 20           | 30           | 33.54    |
| 2018-07-13 16:18:54 | 10           | 20           | 30           | 32.09    |
| 2018-07-13 16:18:50 | 20           | 30           | 31.91        |          |

Hình 5.9 Giá trị hiển thị ở web ở sản phẩm 1

Sau khi sản phẩm 1 đã đủ số lượng thì dừng lại và sản phẩm 2 bắt đầu chạy. Và tương tự cho sản phẩm 3 chạy dữ liệu sẽ được hiển thị trên LCD, giám sát trên Website được thể hiện như hai hình sau:

## CHƯƠNG 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ



Hình 5.10 Giá trị hiển thị ở LCD ở sản phẩm 3

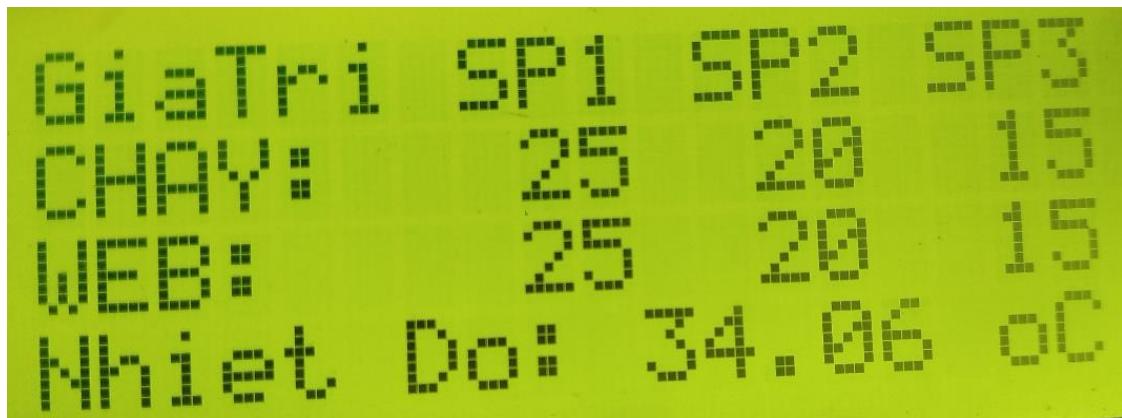


Hình 5.11 Giao diện hiển thị giá trị ở trang giám sát sản phẩm 3

Ngoài giám sát theo bảng thì cũng có thẻ giám sát số sản phẩm hiện tại qua trang giám sát như hình 5.11.

Cho đến khi hệ thống chạy đến sản phẩm cuối cùng thì 3 động cơ sẽ dừng và giá trị đang chạy sẽ bằng giá trị cài đặt ban đầu từ Website. Khi nhập số sản phẩm khác thì hệ thống thiết lập lại các thông số cài đặt và tự động chạy.

## CHƯƠNG 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ



Hình 5.12 Giá trị hiển thị ở LCD ở sản phẩm cuối cùng



Bảng giám sát dây chuyền 1

| Thời gian           | Sản phẩm nhỏ | sản phẩm vừa | sản phẩm lớn | Nhiệt độ |
|---------------------|--------------|--------------|--------------|----------|
| 2018-07-13 16:09:10 | 25           | 20           | 15           | 33.62    |
| 2018-07-13 16:09:06 | 25           | 20           | 15           | 33.64    |
| 2018-07-13 16:09:02 | 25           | 20           | 15           | 34.22    |
| 2018-07-13 16:08:59 | 25           | 20           | 15           | 37.19    |
| 2018-07-13 16:08:55 | 25           | 20           | 15           | 34.13    |
| 2018-07-13 16:08:53 | 25           | 20           | 15           | 34.05    |
| 2018-07-13 16:08:49 | 25           | 20           | 15           | 34.21    |
| 2018-07-13 16:08:46 | 25           | 20           | 15           | 34.37    |
| 2018-07-13 16:08:42 | 25           | 20           | 15           | 34.43    |
| 2018-07-13 16:08:39 | 25           | 20           | 15           | 34       |
| 2018-07-13 16:08:35 | 25           | 20           | 15           | 34       |
| 2018-07-13 16:08:31 | 25           | 20           | 15           | 33.98    |

Hình 5.13 Giá trị hiển thị ở web ở sản phẩm cuối cùng

Khi nhấn nút OFF thì hệ thống sẽ dừng các động cơ ngừng chạy, các giá trị đang đếm sẽ bằng 0. Khi nhấn ON thì hệ thống sẽ hoạt động với các giá cài đặt trước khi nhấn nút OFF.

Bên cạnh đó do sự ổn định của đường truyền (mạng Internet) cũng có sự chậm trễ trong việc truyền dữ liệu từ trên Web Sever xuống và từ phía dưới lên, cụ thể là theo như việc lập trình là thời gian cập nhật trên web là 3 giây 1 lần nhưng do tốc độ mạng và linh kiện chưa được tốt trong khoảng vài chu kỳ thì có lúc 4 giây 5 giây mới cập nhật 1 lần.

## CHƯƠNG 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

### Bảng giám sát dây chuyền 1

| Thời gian           | Sản phẩm nhỏ | sản phẩm vừa | sản phẩm lớn | Nhiệt độ |
|---------------------|--------------|--------------|--------------|----------|
| 2018-07-14 21:54:59 | 25           | 20           | 15           | 32.43    |
| 2018-07-14 21:54:56 | 25           | 20           | 15           | 32.35    |
| 2018-07-14 21:54:53 | 25           | 20           | 15           | 32.3     |
| 2018-07-14 21:54:50 | 25           | 20           | 15           | 32.83    |
| 2018-07-14 21:54:46 | 25           | 20           | 15           | 32.46    |
| 2018-07-14 21:54:43 | 25           | 20           | 15           | 32.56    |
| 2018-07-14 21:54:34 | 25           | 20           | 15           | 32.81    |
| 2018-07-14 21:54:31 | 25           | 20           | 11           | 32.74    |
| 2018-07-14 21:54:29 | 25           | 20           | 7            | 32.63    |
| 2018-07-14 21:54:26 | 25           | 20           | 3            | 32.68    |
| 2018-07-14 21:54:23 | 25           | 13           | 0            | 32.59    |
| 2018-07-14 21:54:20 | 25           | 13           | 0            | 32.59    |
| 2018-07-14 21:54:17 | 25           | 8            | 0            | 32.5     |
| 2018-07-14 21:54:13 | 25           | 5            | 0            | 32.58    |
| 2018-07-14 21:54:10 | 25           | 1            | 0            | 32.58    |
| 2018-07-14 21:54:07 | 19           | 0            | 0            | 32.65    |
| 2018-07-14 21:54:04 | 17           | 0            | 0            | 32.64    |
| 2018-07-14 21:54:01 | 13           | 0            | 0            | 32.7     |
| 2018-07-14 21:53:57 | 11           | 0            | 0            | 32.83    |
| 2018-07-14 21:53:54 | 2            | 0            | 0            | 32.63    |
| 2018-07-14 21:53:51 | 2            | 0            | 0            | 32.63    |
| 2018-07-14 21:53:48 | 0            | 0            | 0            | 32.64    |

Hình 5.14 Giá trị cập nhật trên web bị chậm

Kết luận: do tính chất của việc điều khiển là thiết bị công nghiệp nên sản phẩm làm ra thời gian có thể lâu và số lượng lớn nên việc trễ thời gian một vài giây không làm ảnh hưởng nhiều đến hệ thống.

Phần cuối cùng của trang web mà chúng em muốn giới thiệu là về thông tin và liên hệ của trang web.

The screenshot shows a website for the Faculty of Electrical Engineering and Technology (HCMUTE). The header includes the URL 'websystemspkt.000webhostapp.com/about\_us.php'. The main content features the university's logo (a blue circle with two red hands holding a gear and a flame), the text 'TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH', 'BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP - Y SINH', 'Khoa Điện - Điện Tử', and navigation links for 'TRANG CHỦ', 'ĐĂNG KÝ', 'ĐĂNG NHẬP', and 'LIÊN HỆ'.

#### VỀ CHÚNG TÔI

Giảng viên hướng dẫn: Nguyễn Thanh Giàu

Sinh viên thực hiện: Huỳnh Công Ủy - Nguyễn Văn Linh

Mọi thắc mắc xin vui lòng liên hệ số điện thoại: 0976 184 113 hoặc gửi mail vào địa chỉ này:

nguyenlinh051996@gmail.com

HCMUTE - ©Fri/Jun/2018

P

Hình 5.15 Thông tin liên hệ của trang web

### 5.4 NHẬN XÉT – ĐÁNH GIÁ

#### 5.4.1 Nhận xét

Sau thời gian 15 tuần nghiên cứu và thực hiện đề tài, hệ thống đã đáp ứng được yêu cầu đặt ra ban đầu. Dưới đây là một số nhận xét:

- a. Những vấn đề đã hoàn thành
  - Hệ thống có thể được điều khiển bất kì nơi đâu có Internet hoặc 3G.
  - Có lưu trữ lại dữ liệu khi cần thiết để kiểm tra và theo dõi
  - Giao diện thiết kế đơn giản dễ sử dụng.
- b. Hạn chế
  - Hệ thống còn phụ thuộc vào tốc độ của Wifi hoặc 3G.
  - Server được tạo ra là Server miễn phí nên thời gian đáp ứng còn chậm.

#### 5.4.2 Đánh giá

Sau quá trình vận hành thử hệ thống, nhóm đã tiến hành thực nghiệm và chọn ra thời gian đáp ứng tốt nhất cho hệ thống là 0.5 giây.

**Bảng 5.1 Thời gian đáp ứng đọc từ PLC lên Arduino**

| Thời gian(ms) | Tỉ lệ nhận (%) |
|---------------|----------------|
| 0             | 0              |
| 100           | 68             |
| 200           | 75             |
| 300           | 85             |
| 400           | 95             |
| 500           | 100            |
| 1000          | 100            |

Sau khi tiến hành kiểm tra thực nghiệm trên 100 lần gửi về trên bảng giám sát nhóm đã thống kê và chọn thời gian cập nhật trên Web Sever số lần gửi về với thời gian 3 giây vì giá trị vừa đảm bảo 100% với giá trị cập nhật nhỏ nhất và hệ thống có độ trễ nhỏ nhất.

## **CHƯƠNG 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ**

---

**Bảng 5.2 Giá trị nhận được bằng thực nghiệm**

| Thời gian (s) | Tỉ lệ nhận (số lần) | Tỉ lệ phần trăm (%) |
|---------------|---------------------|---------------------|
| 1             | 20                  | 20                  |
| 1.5           | 39                  | 39                  |
| 2             | 57                  | 57                  |
| 2.5           | 86                  | 86                  |
| 3             | 100                 | 100                 |
| 3.5           | 100                 | 100                 |
| 4             | 100                 | 100                 |

Qua những số liệu ở bảng trên, nhóm đánh giá hệ thống đạt yêu cầu với những mục tiêu đã đề ra. Mô hình có tính thẩm mỹ, an toàn và dễ sử dụng. Sau thời gian test thử, mạch cho thấy sự ổn định. Tuy nhiên vẫn còn một số hạn chế cần khắc phục nếu như muốn đưa hệ thống vào thực tế như: tốc độ điều khiển và phản hồi còn chậm, hệ thống web thì bảo mật chưa cao, hiện tượng nhiễu do sai sót trong quá trình thi công mạch in và hàn linh kiện.

# **Chương 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

## **6.1 KẾT LUẬN**

Sau quá trình nghiên cứu và tìm hiểu chúng em đã hoàn thành đề tài đáp ứng được những yêu cầu đặt ra ban đầu như: thiết kế một hệ thống có thể truyền dữ liệu từ Web Server để điều khiển PLC và giám sát hệ thống qua Web Server. Sử dụng được giao thức Modbus RTU để giao tiếp giữa Arduino và PLC. Thiết kế giao diện Website đơn giản để giám sát và điều khiển. Các thông số trong của hệ thống sẽ phụ thuộc vào tốc độ truyền, thời gian truyền dữ liệu giữa các khối và độ chính xác của việc truyền nhận dữ liệu. Vì vậy khi cài đặt và cập nhật các thông số cho hệ thống thì hệ thống sẽ có độ trễ, cụ thể là thời gian cài đặt từ Web Server mà đến khi PLC nhận được dữ liệu là khoảng gần 2 giây, ngược lại khi cập nhật dữ liệu lên Web Server thì độ trễ khoảng gần 5 giây. Nguyên nhân gây ra độ trễ cho hệ thống có thể gồm những yếu sau:

- Bộ nhớ chương trình, bộ xử lí của Web Server đang sử dụng cũng có thể gây ra độ trễ cho hệ thống.
- Tốc độ mạng Wifi đang sử dụng .
- Tốc độ giao tiếp UART giữa Arduino và Module NodeMCU, tốc độ giao tiếp của Modbus RTU qua module RS-485.
- Ngoài ra thì độ trễ cũng có thể sinh ra từ các khối Counter, Timer của PLC.

Với một hệ thống công nghiệp với qui mô lớn và quá trình cài đặt hệ thống chỉ diễn ra 1,2 trong một ngày nên với độ trễ vài giây thì hệ thống hoàn toàn có thể đáp ứng được. Còn đối với độ chính xác của dữ liệu thì được đáp ứng tương đối khi sử dụng giao thức Modbus RTU là một chuẩn giao thức công nghiệp đã được sử dụng rộng rãi với độ chính xác cao. Ngoài ra giao tiếp UART sử dụng phương pháp thêm vào kí tự khi gửi và lọc dữ liệu khi nhận thì độ chính xác rất cao. Cho độ chính xác của dữ liệu điều khiển là có thể đáp ứng được.

## **CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

---

Một hệ thống đáp ứng được những yêu cầu trên thì hoàn toàn có thể áp dụng trong thực tế. Cụ thể có thể áp dụng trong các nhà máy xí nghiệp, những nơi có mạng Wifi hoặc Internet để cho bộ điều khiển hoạt động. Còn đối với các thiết bị công nghiệp thì đặt bất kì trong nhà máy vì phạm vi giao tiếp của RS-485 có thể lên đến 1km.

### **6.2 HƯỚNG PHÁT TRIỂN**

Mở rộng số thiết bị cần điều khiển với nhiều dây chuyền. Một Web Server có thể điều khiển và giám sát nhiều nhà máy đặt ở nhiều nơi.

Với những gì đã làm được trong tương lai nhóm sẽ mở rộng hơn về giao thức M2M (machine to machine) tức là các máy sẽ tự nói chuyện với nhau mà không cần có sự can thiệp của con người, cụ thể là nếu như hệ thống cần 1000 sản phẩm mà nhà máy có 4 dây chuyền, thì hệ thống sẽ tự động chia ra mỗi dây chuyền sẽ sản xuất 250 sản phẩm, trường hợp may lõi như có 1 hoặc 2 sản dây chuyền gấp sự cố thì các dây chuyền còn lại sẽ tự phân chia ra làm nốt số sản phẩm của dây chuyền gấp sự cố, đảm bảo đúng số lượng sản phẩm cần.

Để phát triển hệ thống lớn hơn, nhiều chức năng hơn thì cần phát triển thêm đó là sử dụng một MCU có tốc độ nhanh hơn, mạnh hơn. Thay thế sóng Wifi bằng sóng Lora để truyền đi xa hơn hoặc thay đổi mạng Wifi bằng mạng cáp quang mạnh hơn. Có thể nâng cấp Database tốc độ cao hơn, Server mạnh hơn để có thể lưu trữ nhiều dữ liệu và điều khiển nhanh hơn.

## **TÀI LIỆU THAM KHẢO**

### **1. Sách tham khảo**

- [1] Tran Thu Ha, *Giáo trình Điện tử cơ bản*, Đại học SPKT Tp.HCM 2013
- [2] Nguyen Dinh Phu, Nguyen Truong Duy, *Giáo Trình: Kỹ Thuật Số*, Xuất bản ĐH Quốc Gia, Tp.HCM, 2013.
- [3] Nguyen Viet Hung, Nguyen Ngo Lam, Nguyen Van Phuc, *Giáo Trình: Kỹ Thuật truyền Số Liệu*, xuất bản 8/2016
- [4] Alasdair Gilchrist, *Industry 4.0: The Industrial Internet of Things*, Springer Science + Business Media New York 2016.

### **2. Trang web tham khảo**

- [5] Cộng đồng Arduino Việt Nam, “Giới thiệu Arduino Mega2560”, 28/11/2015, <http://arduino.vn/bai-viet/542-gioi-thieu-arduino-mega2560>
- [6] Diễn Đàn Vi Điều Khiển, “Bài 8: Giao Tiếp UART”, 17/09/2012, <http://mcu.banlinhkien.vn/threads/bai-8-giao-tiep-uart.28/>
- [7] Samuel, “Modbus-Master-Slave-for-Arduino”, 30/07/2016, <https://github.com/smarmengol/Modbus-Master-Slave-for-Arduino>
- [8] Modbus tool, “Protocol Description”, <http://www.modbustools.com/modbus.h>

## **DANH SÁCH TỪ VIẾT TẮT**

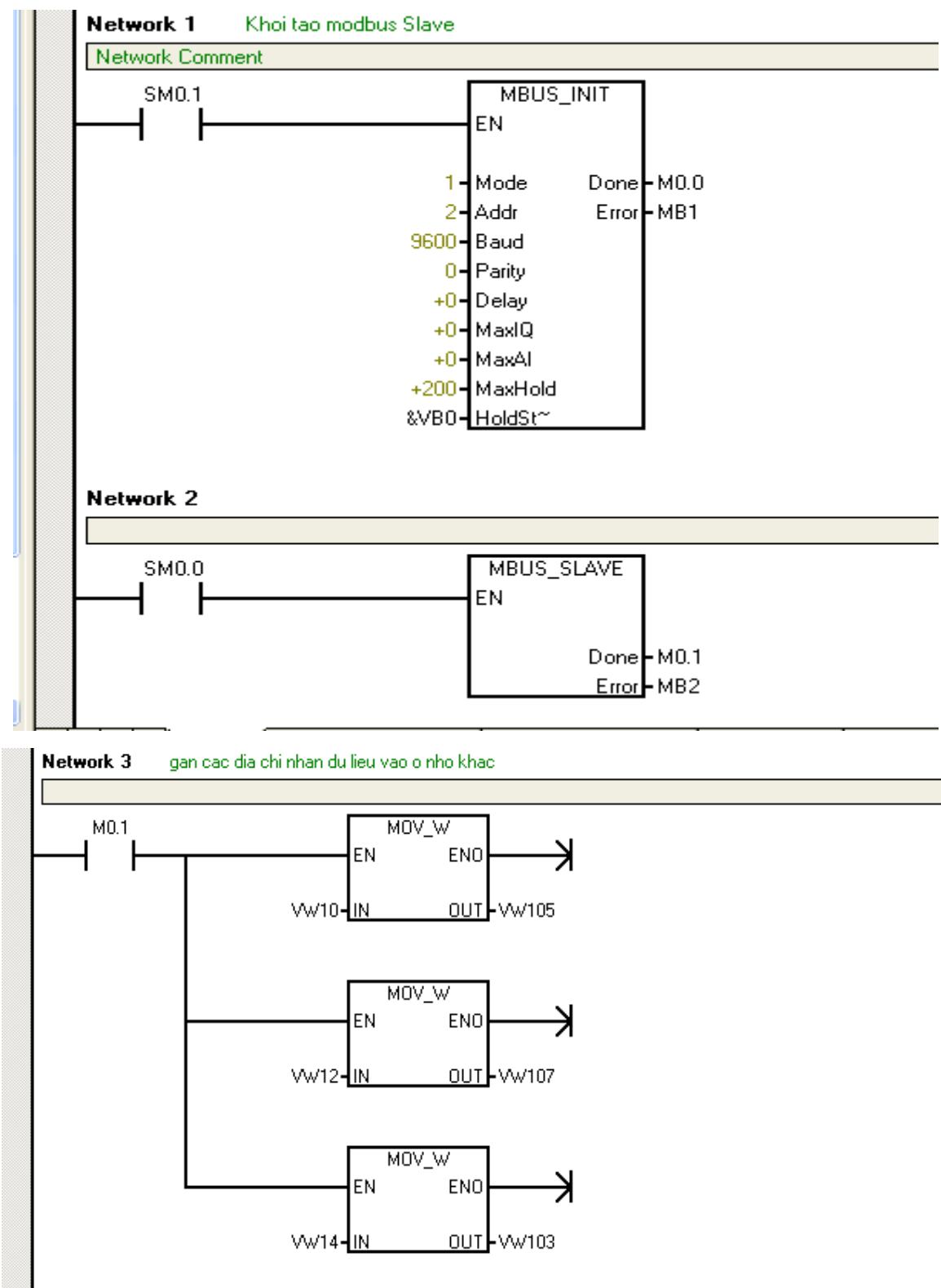
---

### **DANH SÁCH TỪ VIẾT TẮT**

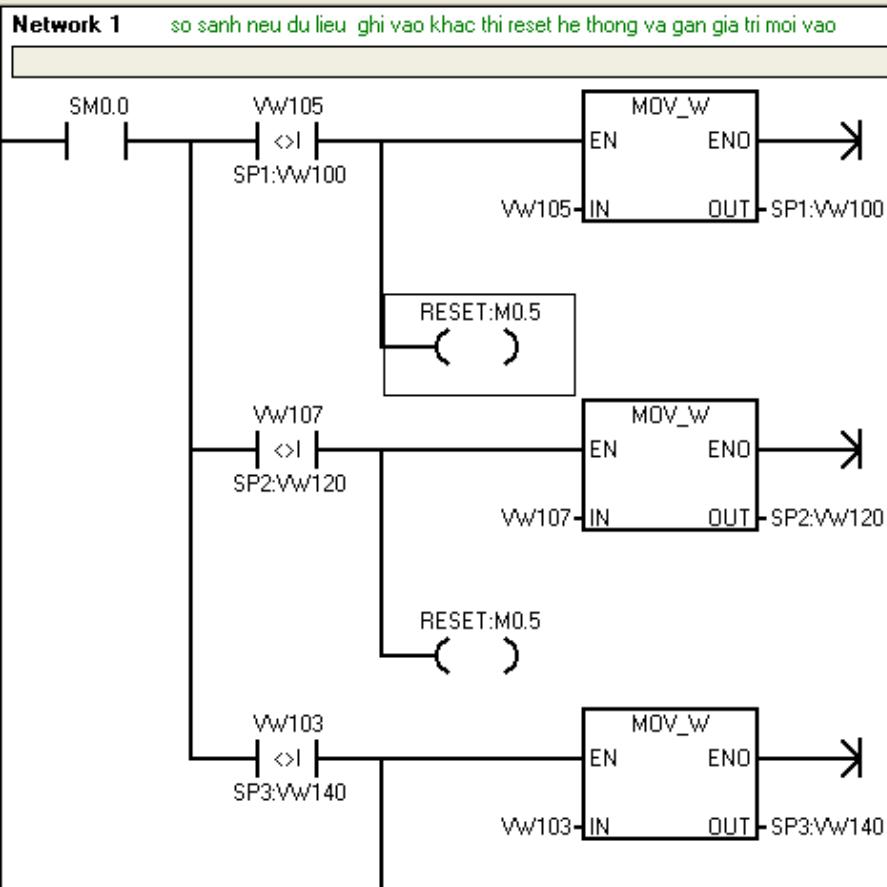
|       |  |
|-------|--|
| SCADA | Supervisory control and data acquisition           |
| IoT   | Internet of things                                 |
| LAN   | Local area network                                 |
| DCS   | Distributed control system                         |
| PHP   | Hypertext preprocessor                             |
| LCD   | Liquid crystal display                             |
| SRAM  | Static random-access memory                        |
| PWM   | Pulse-width modulation                             |
| ASCII | American Standard Code for Information Interchange |
| UART  | Universal asynchronous receiver – transmitter      |
| IDE   | Integrated development environment                 |
| PCB   | Printed circuit board                              |
| M2M   | Machine to machine                                 |
| PLC   | Programmable Logic Control                         |

# **PHỤ LỤC**

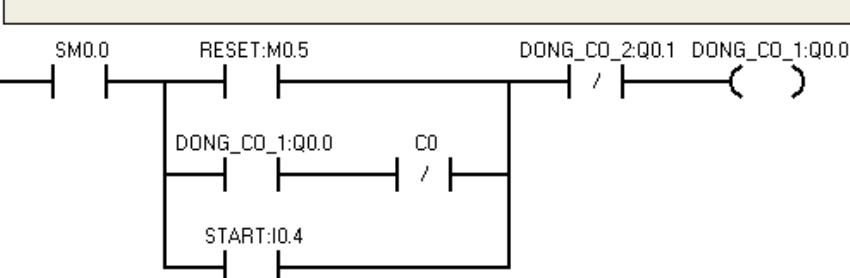
## **1. Code chương trình cho PLC**



## PHỤ LỤC

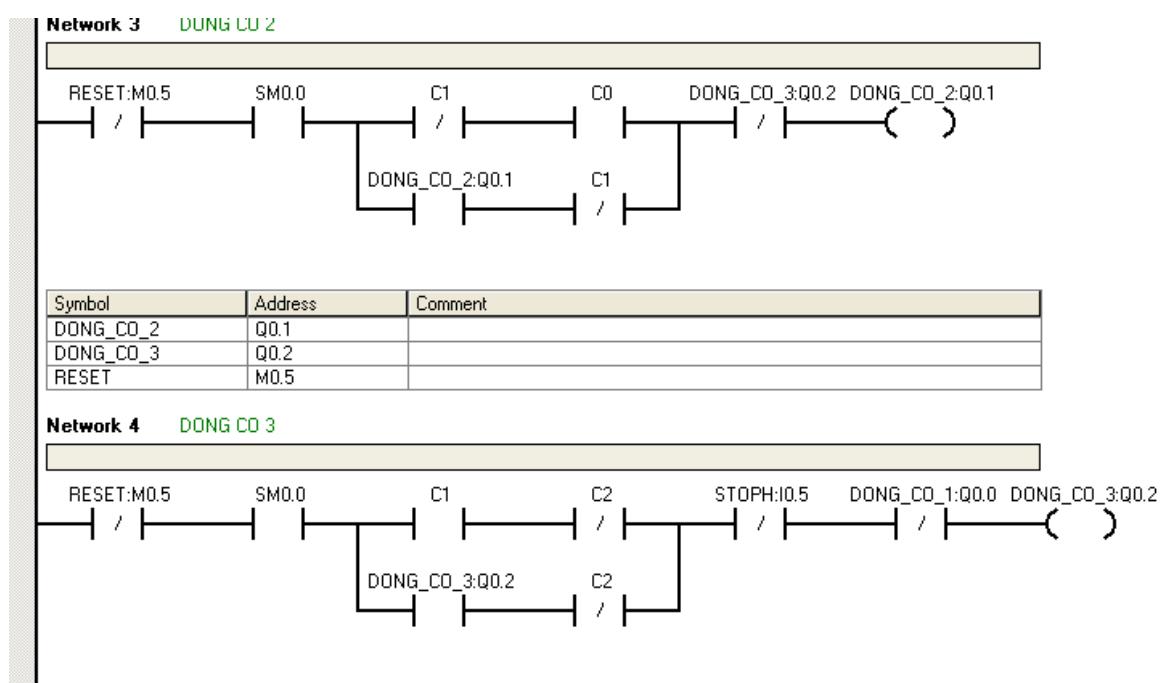


**Network 2** DÒNG CO 1

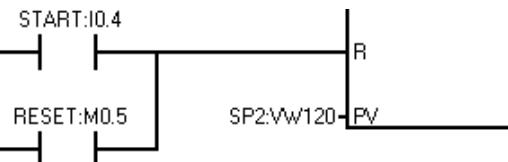


| Symbol    | Address | Comment |
|-----------|---------|---------|
| DONG_CO_1 | Q0.0    |         |
| DONG_CO_2 | Q0.1    |         |
| RESET     | M0.5    |         |
| START     | I0.4    |         |

## PHỤ LỤC

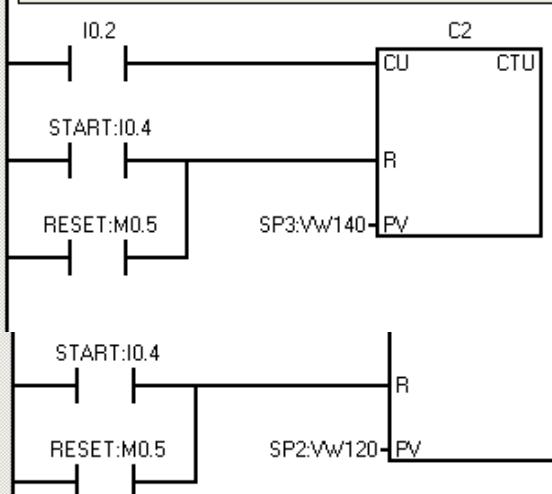


## PHỤ LỤC



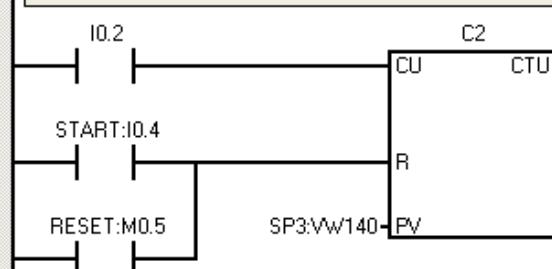
| Symbol | Address | Comment |
|--------|---------|---------|
| RESET  | M0.5    |         |
| SP2    | Vw120   |         |
| START  | I0.4    |         |

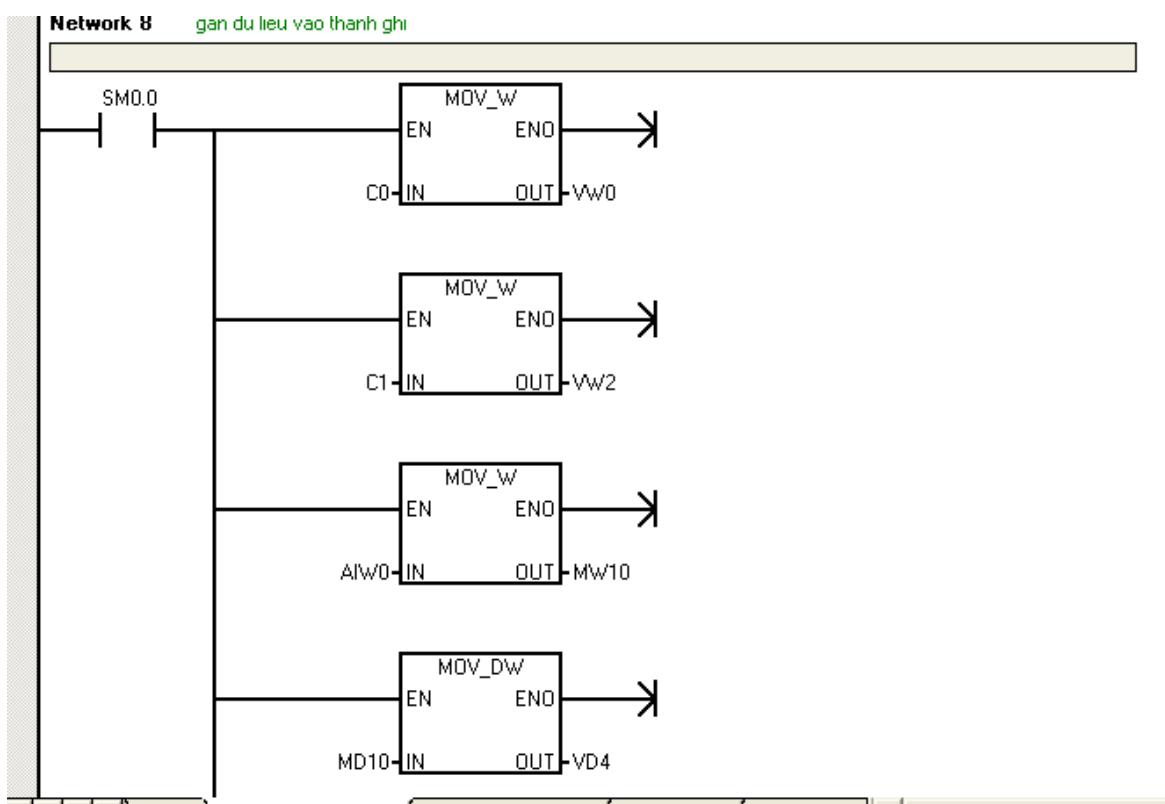
**Network 7** dem san pham 3



| Symbol | Address | Comment |
|--------|---------|---------|
| RESET  | M0.5    |         |
| SP2    | Vw120   |         |
| START  | I0.4    |         |

**Network 7** dem san pham 3





## 2. Chương trình điều khiển Node MCU

```

#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>

const char* ssid = "No Comment";
const char* password = "mangyeu6789";
String inputString = "", data;
boolean stringComplete = false;
String dem1, dem2, dem3, dem4;

void setup() {
  pinMode(16, OUTPUT);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
  }
  digitalWrite(16, HIGH);
  Serial.begin(9600);
  inputString.reserve(200);
}

void loop() {
  
```

## PHỤ LỤC

```
if (WiFi.status() == WL_CONNECTED) {  
  
    HTTPClient http;  
  
    http.begin("http://websystemspt.000webhostapp.com/getdata.php?getvar=1&  
SN=DSNW");  
    http.addHeader("Content-Type", "text/plain");  
    int httpCode = http.POST("");  
    String sp1 = http.getString();  
    Serial.print("A");  
    Serial.print(sp1);  
    Serial.print("@");  
    http.end();  
  
  
    http.begin("http://websystemspt.000webhostapp.com/getdata.php?getvar=2&  
SN=SP2");  
    http.addHeader("Content-Type", "text/plain");  
    int httpCode1 = http.POST("");  
    String sp2 = http.getString();  
    Serial.print("B");  
    Serial.print(sp2);  
    Serial.print("#");  
    http.end();  
  
  
    http.begin("http://websystemspt.000webhostapp.com/getdata.php?getvar=3&  
SN=SP3");  
    http.addHeader("Content-Type", "text/plain");  
    int httpCode2 = http.POST("");  
    String sp3 = http.getString();  
    Serial.print("C");  
    Serial.print(sp3);  
    Serial.print("$");  
    http.end();  
}  
if (WiFi.status() == WL_CONNECTED) {  
  
    HTTPClient http;  
    data = "http://websystemspt.000webhostapp.com//add_data.php?temp="  
+ String(dem1)
```

## PHỤ LỤC

```
+ "&hum=" + String(dem2) + "&sp3=" + String(dem3) + "&may2=" +
String(dem4);

    http.begin(data);
    int httpCode3 = http.GET();
    if (httpCode3 > 0)
    {
        String payload = http.getString();
    }
    http.end();
}

if (stringComplete) {
    dem1 = inputString.substring(inputString.indexOf('D') + 1,
inputString.indexOf('!'));
    dem2 = inputString.substring(inputString.indexOf('E') + 1,
inputString.indexOf('%'));
    dem3 = inputString.substring(inputString.indexOf('F') + 1,
inputString.indexOf('&'));
    dem4 = inputString.substring(inputString.indexOf('G') + 1,
inputString.indexOf('*'));
    inputString = "";
    stringComplete = false;
}
serialEvent();
}

void serialEvent() {
    while (Serial.available()) {
        char inChar = (char)Serial.read();
        inputString += inChar;
        if (inChar == '%') {
            stringComplete = true;
        }
    }
}
```

### 3. Chương trình điều khiển cho Arduino

```
#include <ModbusRtu.h>
#include <LiquidCrystal.h>

uint16_t au16data[16];
```

## PHỤ LỤC

```
uint16_t au16data2[16];
uint8_t u8state;
uint8_t u8query;

Modbus master(0, 1, 0);

const int rs = 43, en = 41, d4 = 39, d5 = 37, d6 = 35, d7 = 33;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
String inputString = "", inputString2 = "";
boolean stringComplete = false, stringComplete2 = false;
modbus_t telegram[2];
float nhietdo ;
int x, a, c, d, writerin;
unsigned long u32wait;
unsigned long timer;
int a_tam , d_tam, c_tam;
int ledPin = 13;
int val = 0;

void setup() {
    pinMode(ledPin, OUTPUT);
    Serial3.begin(9600);
    lcd.begin(20, 4);
    lcd.print("Khoi Dong.....");
    inputString2.reserve(200);
    telegram[0].u8id = 2;
    telegram[0].u8fct = 3;
    telegram[0].u16RegAdd = 0;
    telegram[0].u16CoilsNo = 5;
    telegram[0].au16reg = au16data;

    telegram[1].u8id = 2;
    telegram[1].u8fct = 16;
    telegram[1].u16RegAdd = 5;
    telegram[1].u16CoilsNo = 10;
    telegram[1].au16reg = au16data2 ;

    master.begin( 9600 );
    master.setTimeOut( 5000 );
    u32wait = millis() + 1000;
    u8state = u8query = 0;
}
```

```
void gan_du_lieu_plc() {  
  
    if ((a_tam != 0) && (d_tam != 0) && (c_tam != 0))  
    {  
        au16data2[0] = a_tam;  
        au16data2[2] = d_tam;  
        au16data2[1] = c_tam ;  
    }  
    x = au16data[2];  
    nhietdo = (float)x / 320.00;  
}  
  
void hien_thi_lcd() {  
    lcd.setCursor(0, 0);  
    lcd.print("GiaTri SP1 SP2 SP23 ");  
    char buff[20];  
    sprintf(buff, "CHAY: %3d %3d %3d", au16data[0], au16data[1],  
au16data[3]);  
    String a = (char*) buff;  
    lcd.setCursor(0, 1 );  
    lcd.print(a);  
    char buff2[20];  
    sprintf(buff2, "WEB: %3d %3d %3d", au16data2[0], au16data2[1],  
au16data2[2]);  
    String b = (char*) buff2;  
    lcd.setCursor(0, 2 );  
    lcd.print(b);  
    lcd.setCursor(0, 3);  
    lcd.print("Nhiet Do: "); lcd.print(nhietdo);  
    lcd.print(" oC");  
}  
void gui_du_lieu ()  
{  
    Serial3.print('D');  
    Serial3.print(au16data[0]);  
    Serial3.print('!');  
    Serial3.print('E');  
    Serial3.print(au16data[1]);  
    Serial3.print('%');  
    Serial3.print('F');  
    Serial3.print(au16data[3]);  
}
```

## PHỤ LỤC

```
Serial3.print('&');
Serial3.print('G');
Serial3.print(nhietdo);
Serial3.print('*');
}

void loop() {
    val = analogRead(A0);
    analogWrite(ledPin, val / 4);
    if (stringComplete2) {
        String giatriA = inputString2.substring(inputString2.indexOf("A") +
1, inputString2.indexOf("@"));
        a = giatriA.toInt();
        String giatriB = inputString2.substring(inputString2.indexOf("B") +
1, inputString2.indexOf("#"));
        c = giatriB.toInt();
        String giatriC = inputString2.substring(inputString2.indexOf("C") +
1, inputString2.indexOf("$"));
        d = giatriC.toInt();
        inputString2 = "";
        stringComplete2 = false;
    }
    if ((a_tam != a) || (d_tam != d) || (c_tam != c))
    {
        writerin = 2;
        a_tam = a; d_tam = d; c_tam = c;
    }
    else
    {
        writerin = 1;
    }

    if (millis() > timer) {
        gui_du_lieu ();
        timer = millis() + 500;
    }
    else {
        switch ( u8state ) {
            case 0: //trong kenh
                if (millis() > u32wait) u8state++;
                break;
            case 1:
```

```
    master.query( telegram[u8query] );
    u8state++;    u8query++;
    if (u8query > writerin) u8query = 0;
    break;
case 2:
    master.poll();
    if (master.getState() == COM_IDLE) {
        u8state = 0;
        u32wait = millis() + 50;
    }
    break;
}
gan_du_lieu_plc();
hien_thi_lcd();
}
}

void serialEvent3() {
    while (Serial3.available()) {
        char inChar2 = (char)Serial3.read();
        inputString2 += inChar2;
        if (inChar2 == '$') {
            stringComplete2 = true;
        }
    }
}
```

#### 4. Datasheet của các linh kiện

##### a. LM 2576-5V

**LM2576****3.0 A, 15 V, Step-Down  
Switching Regulator**

The LM2576 series of regulators are monolithic integrated circuits ideally suited for easy and convenient design of a step-down switching regulator (buck converter). All circuits of this series are capable of driving a 3.0 A load with excellent line and load regulation. These devices are available in fixed output voltages of 3.3 V, 5.0 V, 12 V, 15 V, and an adjustable output version.

These regulators were designed to minimize the number of external components to simplify the power supply design. Standard series of inductors optimized for use with the LM2576 are offered by several different inductor manufacturers.

Since the LM2576 converter is a switch-mode power supply, its efficiency is significantly higher in comparison with popular three-terminal linear regulators, especially with higher input voltages. In many cases, the power dissipated is so low that no heatsink is required or its size could be reduced dramatically.

A standard series of inductors optimized for use with the LM2576 are available from several different manufacturers. This feature greatly simplifies the design of switch-mode power supplies.

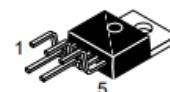
The LM2576 features include a guaranteed  $\pm 4\%$  tolerance on output voltage within specified input voltages and output load conditions, and  $\pm 10\%$  on the oscillator frequency ( $\pm 2\%$  over  $0^\circ\text{C}$  to  $125^\circ\text{C}$ ). External shutdown is included, featuring 80  $\mu\text{A}$  (typical) standby current. The output switch includes cycle-by-cycle current limiting, as well as thermal shutdown for full protection under fault conditions.

**Features**

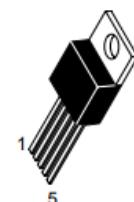
- 3.3 V, 5.0 V, 12 V, 15 V, and Adjustable Output Versions
- Adjustable Version Output Voltage Range, 1.23 to 37 V  $\pm 4\%$



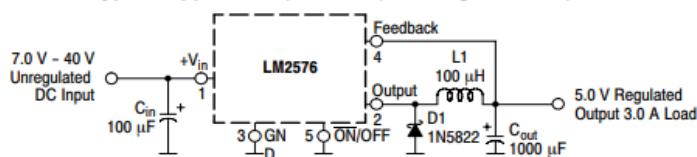
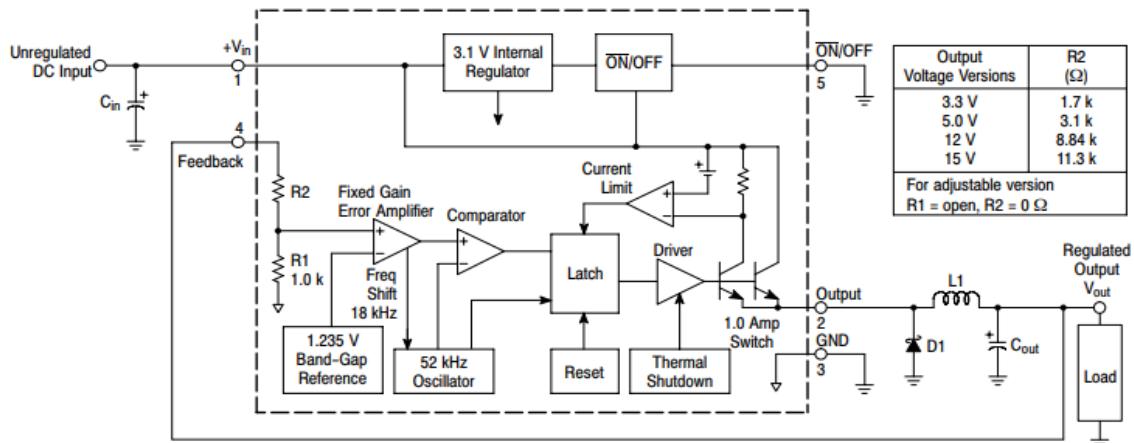
ON Semiconductor®

<http://onsemi.com>TO-220  
TV SUFFIX  
CASE 314B

Heatsink surface connected to Pin 3

TO-220  
T SUFFIX  
CASE 314D

|     |    |                 |
|-----|----|-----------------|
| Pin | 1. | V <sub>in</sub> |
|     | 2. | Output          |
|     | 3. | Ground          |
|     | 4. | Feedback        |
|     | 5. | ON/OFF          |

**LM2576****Typical Application (Fixed Output Voltage Versions)****Representative Block Diagram and Typical Application**

This device contains 162 active transistors.

Figure 1. Block Diagram and Typical Application

## PHỤ LỤC

---

### MAXIMUM RATINGS

| Rating   | Symbol   | Value  | Unit                                   |
|--|--|--|--|
| Maximum Supply Voltage   | $V_{in}$   | 45   | V                                      |
| ON/OFF Pin Input Voltage   | -  | $-0.3 \text{ V} \leq V \leq +V_{in}$                               | V                                      |
| Output Voltage to Ground (Steady-State)  | -  | -1.0   | V                                      |
| Power Dissipation<br>Case 314B and 314D (TO-220, 5-Lead)<br>Thermal Resistance, Junction-to-Ambient<br>Thermal Resistance, Junction-to-Case<br>Case 936A (D <sup>2</sup> PAK)<br>Thermal Resistance, Junction-to-Ambient<br>Thermal Resistance, Junction-to-Case | $P_D$<br>$R_{\theta JA}$<br>$R_{\theta JC}$<br>$P_D$<br>$R_{\theta JA}$<br>$R_{\theta JC}$ | Internally Limited<br>65<br>5.0<br>Internally Limited<br>70<br>5.0 | W<br>°C/W<br>°C/W<br>W<br>°C/W<br>°C/W |
| Storage Temperature Range  | $T_{stg}$  | -65 to +150  | °C                                     |
| Minimum ESD Rating (Human Body Model: C = 100 pF, R = 1.5 kΩ)  | -  | 2.0  | kV                                     |
| Lead Temperature (Soldering, 10 seconds)   | -  | 260  | °C                                     |
| Maximum Junction Temperature   | $T_J$  | 150  | °C                                     |

Maximum ratings are those values beyond which device damage can occur. Maximum ratings applied to the device are individual stress limit values (not normal operating conditions) and are not valid simultaneously. If these limits are exceeded, device functional operation is not implied, damage may occur and reliability may be affected.

### LM2576-5 (Note 1 Test Circuit Figure 15)

|   |           |             |          |             |   |
|---|-----------|-------------|----------|-------------|---|
| Output Voltage ( $V_{in} = 12 \text{ V}$ , $I_{Load} = 0.5 \text{ A}$ , $T_J = 25^\circ\text{C}$ )  | $V_{out}$ | 4.9         | 5.0      | 5.1         | V |
| Output Voltage ( $8.0 \text{ V} \leq V_{in} \leq 40 \text{ V}$ , $0.5 \text{ A} \leq I_{Load} \leq 3.0 \text{ A}$ )<br>$T_J = 25^\circ\text{C}$<br>$T_J = -40 \text{ to } +125^\circ\text{C}$ | $V_{out}$ | 4.8<br>4.75 | 5.0<br>- | 5.2<br>5.25 | V |
| Efficiency ( $V_{in} = 12 \text{ V}$ , $I_{Load} = 3.0 \text{ A}$ )   | $\eta$    | -           | 77       | -           | % |

### b. ULN 2803

## PHỤ LỤC

**MAXIMUM RATINGS** ( $T_A = 25^\circ\text{C}$  and rating apply to any one device in the package, unless otherwise noted.)

| Rating                              | Symbol    | Value       | Unit |
|-------------------------------------|-----------|-------------|------|
| Output Voltage                      | $V_O$     | 50          | V    |
| Input Voltage (Except ULN2801)      | $V_I$     | 30          | V    |
| Collector Current – Continuous      | $I_C$     | 500         | mA   |
| Base Current – Continuous           | $I_B$     | 25          | mA   |
| Operating Ambient Temperature Range | $T_A$     | 0 to +70    | °C   |
| Storage Temperature Range           | $T_{stg}$ | -55 to +150 | °C   |
| Junction Temperature                | $T_J$     | 125         | °C   |

$R_{0JA} = 55^\circ\text{C}/\text{W}$

Do not exceed maximum current limit per driver.

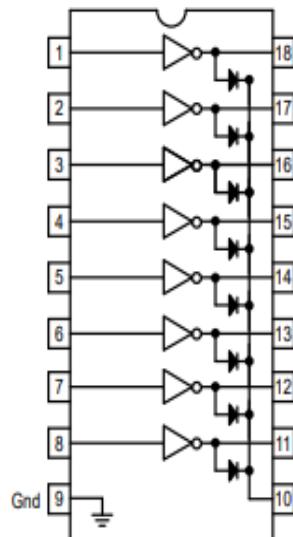
### ORDERING INFORMATION

| Device               | Characteristics                         |                                      |   |
|----------------------|---|--------------------------------------|---|
|                      | Input Compatibility                     | $V_{CE}(\text{Max})/I_C(\text{Max})$ | Operating Temperature Range             |
| ULN2803A<br>ULN2804A | TTL, 5.0 V CMOS<br>6 to 15 V CMOS, PMOS | 50 V/500 mA                          | $T_A = 0 \text{ to } +70^\circ\text{C}$ |



A SUFFIX  
PLASTIC PACKAGE  
CASE 707

### PIN CONNECTIONS



**ULN2803 ULN2804**

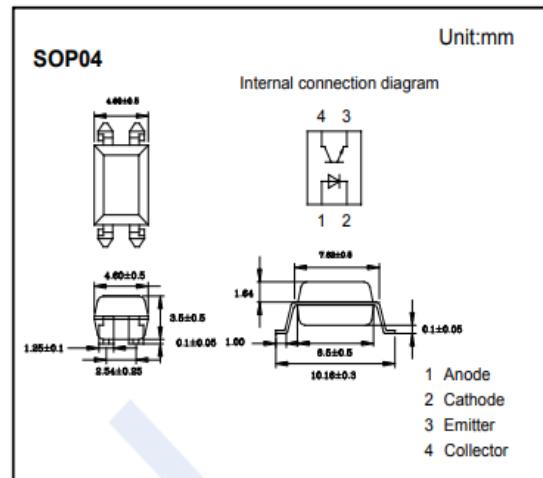
ELECTRICAL CHARACTERISTICS ( $T_A = 25^\circ\text{C}$ , unless otherwise noted)

| Characteristic  | Symbol               | Min  | Typ                         | Max   | Unit          |
|---|----------------------|------|-----------------------------|---|---------------|
| Output Leakage Current (Figure 1)<br>( $V_O = 50 \text{ V}, T_A = +70^\circ\text{C}$ )<br>( $V_O = 50 \text{ V}, T_A = +25^\circ\text{C}$ )<br>( $V_O = 50 \text{ V}, T_A = +70^\circ\text{C}, V_I = 6.0 \text{ V}$ )<br>( $V_O = 50 \text{ V}, T_A = +70^\circ\text{C}, V_I = 1.0 \text{ V}$ )   | $I_{CEX}$            | —    | —                           | 100   | $\mu\text{A}$ |
| All Types<br>All Types<br>ULN2802<br>ULN2804  |                      | —    | —                           | 50  |               |
| Collector-Emitter Saturation Voltage (Figure 2)<br>( $I_C = 350 \text{ mA}, I_B = 500 \mu\text{A}$ )<br>( $I_C = 200 \text{ mA}, I_B = 350 \mu\text{A}$ )<br>( $I_C = 100 \text{ mA}, I_B = 250 \mu\text{A}$ )  | $V_{CE(\text{sat})}$ | —    | 1.1<br>0.95<br>0.85         | 1.6<br>1.3<br>1.1                                   | $\text{V}$    |
| All Types<br>All Types<br>All Types   |                      | —    |                             |   |               |
| Input Current – On Condition (Figure 4)<br>( $V_I = 17 \text{ V}$ )<br>( $V_I = 3.85 \text{ V}$ )<br>( $V_I = 5.0 \text{ V}$ )<br>( $V_I = 12 \text{ V}$ )  | $I_{I(\text{on})}$   | —    | 0.82<br>0.93<br>0.35<br>1.0 | 1.25<br>1.35<br>0.5<br>1.45                         | $\text{mA}$   |
| ULN2802<br>ULN2803<br>ULN2804<br>ULN2804  |                      | —    |                             |   |               |
| Input Voltage – On Condition (Figure 5)<br>( $V_{CE} = 2.0 \text{ V}, I_C = 300 \text{ mA}$ )<br>( $V_{CE} = 2.0 \text{ V}, I_C = 200 \text{ mA}$ )<br>( $V_{CE} = 2.0 \text{ V}, I_C = 250 \text{ mA}$ )<br>( $V_{CE} = 2.0 \text{ V}, I_C = 300 \text{ mA}$ )<br>( $V_{CE} = 2.0 \text{ V}, I_C = 125 \text{ mA}$ )<br>( $V_{CE} = 2.0 \text{ V}, I_C = 200 \text{ mA}$ )<br>( $V_{CE} = 2.0 \text{ V}, I_C = 275 \text{ mA}$ )<br>( $V_{CE} = 2.0 \text{ V}, I_C = 350 \text{ mA}$ ) | $V_{I(\text{on})}$   | —    | —                           | 13<br>2.4<br>2.7<br>3.0<br>5.0<br>6.0<br>7.0<br>8.0 | $\text{V}$    |
| ULN2802<br>ULN2803<br>ULN2803<br>ULN2803<br>ULN2804<br>ULN2804<br>ULN2804<br>ULN2804  |                      | —    |                             |   |               |
| Input Current – Off Condition (Figure 3)<br>( $I_C = 500 \mu\text{A}, T_A = +70^\circ\text{C}$ )  | $I_{I(\text{off})}$  | 50   | 100                         | —   | $\mu\text{A}$ |
| All Types   |                      | —    |                             |   |               |
| DC Current Gain (Figure 2)<br>( $V_{CE} = 2.0 \text{ V}, I_C = 350 \text{ mA}$ )  | $h_{FE}$             | 1000 | —                           | —   | —             |
| ULN2801   |                      | —    |                             |   |               |
| Input Capacitance   | $C_I$                | —    | 15                          | 25  | $\text{pF}$   |
| Turn-On Delay Time<br>(50% $E_I$ to 50% $E_O$ )   | $t_{on}$             | —    | 0.25                        | 1.0   | $\mu\text{s}$ |
| Turn-Off Delay Time<br>(50% $E_I$ to 50% $E_O$ )  | $t_{off}$            | —    | 0.25                        | 1.0   | $\mu\text{s}$ |
| Clamp Diode Leakage Current (Figure 6)<br>( $V_R = 50 \text{ V}$ )<br>$T_A = +25^\circ\text{C}$<br>$T_A = +70^\circ\text{C}$  | $I_R$                | —    | —                           | 50<br>100   | $\mu\text{A}$ |
|   |                      | —    |                             |   |               |
| Clamp Diode Forward Voltage (Figure 7)<br>( $I_F = 350 \text{ mA}$ )  | $V_F$                | —    | 1.5                         | 2.0   | $\text{V}$    |
|   |                      | —    |                             |   |               |

**c. C817**

**■ Features**

- Current transfer ratio  
(CTR: MIN. 50% at  $I_F = 5\text{mA}$ ,  $V_{CE} = 5\text{V}$ )
- High isolation voltage between input and output



**■ Absolute Maximum Ratings  $T_a = 25^\circ\text{C}$**

| Parameter                           | Symbol    | Rating     | Unit                    |
|-------------------------------------|-----------|------------|-------------------------|
| Input Reverse voltage               | $V_R$     | 6          | V                       |
| Collector - Emitter Output Voltage  | $V_{CEO}$ | 35         |                         |
| Emitter-Collector Output Voltage    | $V_{ECO}$ | 6          |                         |
| Isolation Voltage                   | $V_{ISO}$ | 5000       | $\text{V}_{\text{rms}}$ |
| Input Forward Current               | $I_F$     | 50         | $\text{mA}$             |
| Input Peak Forward Current (Note.1) | $I_{FM}$  | 1          | A                       |
| Collector Current - Continuous      | $I_C$     | 50         | $\text{mA}$             |
| Input Power Dissipation             | P         | 70         | $\text{mW}$             |
| Collector Output Power dissipation  | $P_C$     | 150        |                         |
| Total Power Dissipation             | $P_{tot}$ | 200        |                         |
| Junction Temperature                | $T_J$     | 125        |                         |
| Soldering temperature               | $T_{sol}$ | 260        | $^\circ\text{C}$        |
| Operating Temperature               | $T_{opr}$ | -30 to 100 |                         |
| Storage Temperature Range           | $T_{stg}$ | -55 to 125 |                         |

Note.1:Pulse width  $\leq 100\text{ms}$ , Duty ratio : 0.001

## PHỤ LỤC

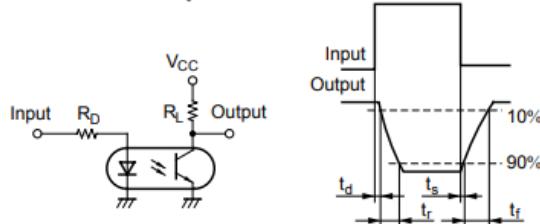
### ■ Electrical Characteristics $T_a = 25^\circ C$

| Parameter                            | Symbol               | Test Conditions   | Min                | Typ       | Max | Unit          |
|--------------------------------------|----------------------|---|--------------------|-----------|-----|---------------|
| Input Forward Voltage                | $V_F$                | $I_F = 20 \text{ mA}$                                       |                    |           | 1.4 | V             |
| Input Peak Forward Voltage           | $V_{FM}$             | $I_{FM} = 500 \text{ mA}$                                   |                    |           | 3   |               |
| Input Reverse Current                | $I_R$                | $V_R = 4 \text{ V}$   |                    |           | 10  | $\mu\text{A}$ |
| Collector-emitter cut-off current    | $I_{CEO}$            | $V_{CE} = 20 \text{ V}, I_E = 0$                            |                    |           | 10  |               |
| Collector-emitter saturation voltage | $V_{CE(\text{sat})}$ | $I_F = 20 \text{ mA}, I_C = 1 \text{ mA}$                   |                    | 0.1       | 0.2 | V             |
| Isolation resistance                 | $R_{iso}$            | DC 500V, 40 to 60% RH                                       | $5 \times 10^{10}$ | $10^{11}$ |     | $\Omega$      |
| Current Transfer Ratio               | $CTR$                | $V_{CE} = 5 \text{ V}, I_F = 5 \text{ mA}$                  | 50                 |           | 600 | %             |
| Rise time                            | $t_r$                | $V_{CE} = 2 \text{ V}, I_C = 2 \text{ mA}, R_L = 100\Omega$ |                    | 4         | 18  | $\mu\text{s}$ |
| Fall time                            | $t_f$                |   |                    | 3         | 18  |               |
| Input Terminal Capacitance           | $C_t$                | $V = 0 \text{ V}, f = 1 \text{ KHz}$                        |                    | 30        | 250 | $\text{pF}$   |
| Floating Capacitance                 | $C_f$                |   |                    | 0.6       | 1   |               |
| Cut-off frequency                    | $f_c$                | $V_{CE} = 5 \text{ V}, I_C = 2 \text{ mA}, R_L = 100\Omega$ |                    | 80        |     | KHz           |

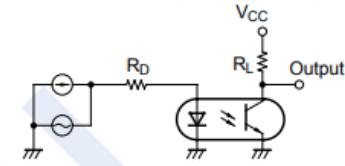
### ■ Classification of CTR(%)

| Type  | PC817A | PC817B  | PC817C  | PC817D  | PC817  |
|-------|--------|---------|---------|---------|--------|
| Range | 80-160 | 130-260 | 200-400 | 300-600 | 50-600 |

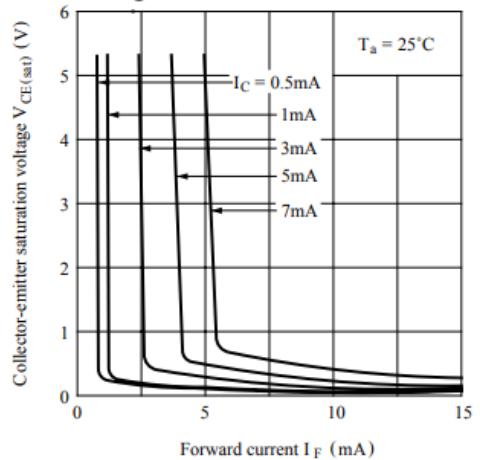
**Test Circuit for Response Time**



**Test Circuit for Frequency Response**

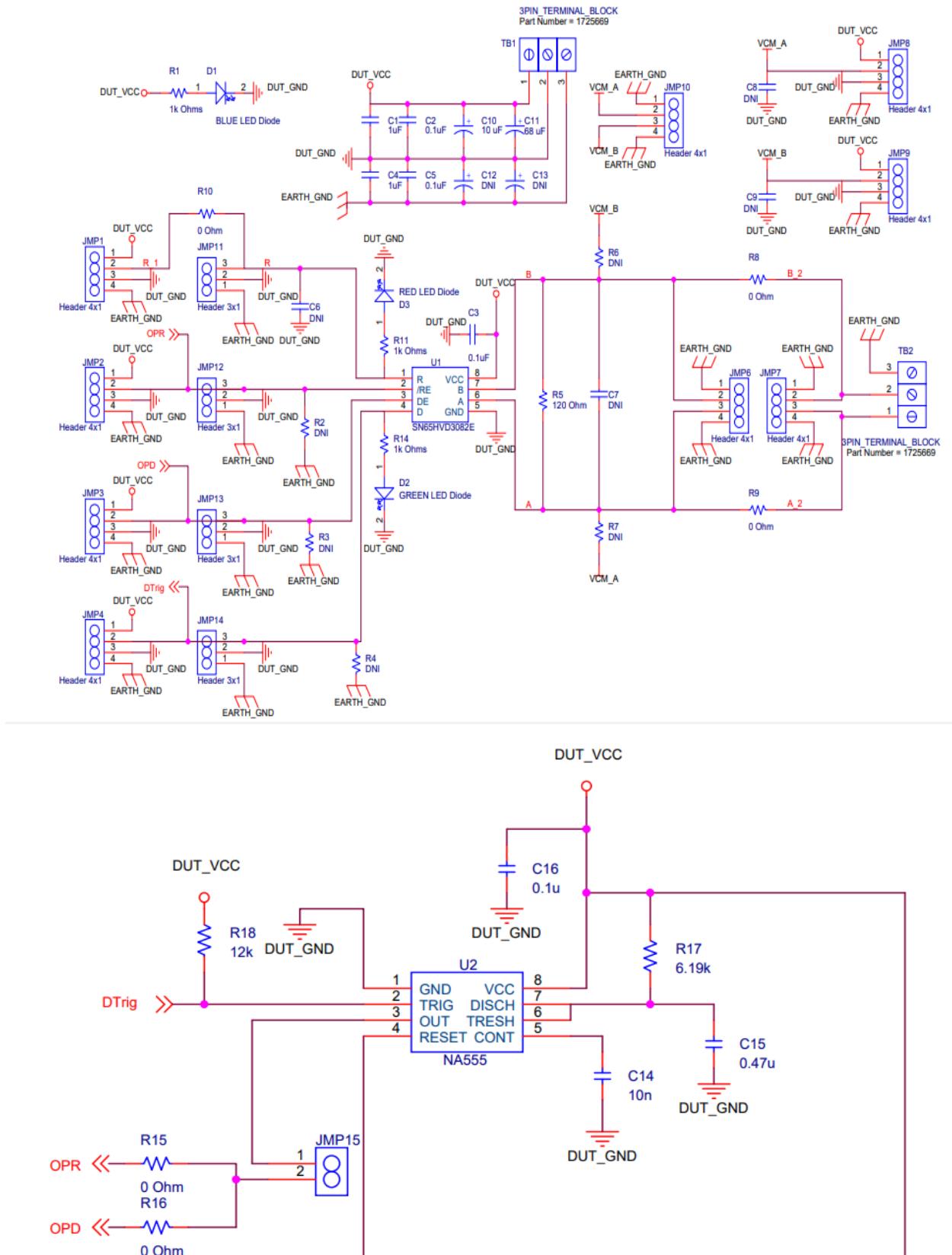


**Fig.12 Collector-emitter Saturation Voltage vs. Forward Current**



## 5. Mạch nguyên lý của RS-485

Đây là mạch nguyên lý thiết kế RS-485 truyền tự động chọn chiều sử dụng NA555 có thể tham khảo



## 6. Code PHP Web Server

Code web ở trang điều khiển chính.

## PHỤ LỤC

```
<?php $page = 'control'; ?>
<?php
// Start the session
session_start();
?>
<?php require_once("dbconnect.php"); ?>
<?php
@$light = $_GET['light'];
@$motor = $_GET['motor'];
@$dc1 = $_GET['dc1'];
@$dc2 = $_GET['dc2'];
@$dc3 = $_GET['dc3'];
@$send = $_GET['send'];
@$l = $_GET['l'];
    if (isset($_POST["btn_submit"])) {
        //lấy thông tin từ các form bằng phương thức POST
        $dc1 = $_POST["dc1"];
        $dc2 = $_POST["dc2"];
        $dc3 = $_POST["dc3"];
    if ($dc1 == "" || $dc2 == "" || $dc3 == "") {
        echo "Bạn vui lòng nhập đầy đủ thông tin";
    } else{
        //sql = "INSERT INTO servertoPLC(SN,varname,value)
VALUES ( 'DSNW','1','$dc1')";
        //sql = "INSERT INTO servertoPLC1(SN,varname,value)
VALUES ( 'DSNW1','1','$dc2')";

        $sql = "UPDATE servertoPLC SET value='$dc1' WHERE SN = 'DSNW' and
varname = '1' ";
        $result = mysqli_query($conn, $sql);
        mysqli_query($conn,$sql);
        $sq2 = "UPDATE servertoPLC SET value='$dc2' WHERE SN = 'SP2' and
varname = '2' ";
        $result = mysqli_query($conn, $sq2);
        // thực thi câu $sql với biến conn lấy từ file
connection.php
        mysqli_query($conn,$sq2);
        $sq3 = "UPDATE servertoPLC SET value='$dc3' WHERE SN = 'SP3' and
varname = '3' ";
        $result = mysqli_query($conn, $sq3);
        mysqli_query($conn,$sq3);
        echo "Chúc mừng bạn đã nhập số sản phẩm thành công";
    }
    //

if($light == "on")
{
    $file = fopen("light.json", "w") or die("can't open file");
    fwrite($file, 'light:on');
    fclose($file);
}
else if ($light == "off")
{
    $file = fopen("light.json", "w") or die("can't open file");
    fwrite($file, 'light:off');
    fclose($file);
}

if ($motor == "on")
{
    $file = fopen("motor.json", "w") or die("can't open file");
    fwrite($file, 'motor:on');
```

```
fclose($file);
}
else if ($motor == "off")
{
    $file = fopen("motor.json", "w") or die("can't open file");
    fwrite($file, 'motor:off');
    fclose($file);
}
?>

<html>
<head>
    <meta name="google-site-verification"
content="wgrr7kGz9Znj5VzmjolgQA8LQSlicULPIhtuF_lneAw" />
    <title>Điều Khiển</title>
    <meta name=viewport content="width=device-width, initial-scale=1">
    <link rel = "stylesheet" type="text/css" href="mystyle.css">

        <script src="https://code.jquery.com/jquery-
2.1.4.min.js"></script>
        <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js"
"></script>
        <link
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.
css" rel="stylesheet">
        <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-
awesome/4.3.0/css/font-awesome.min.css">

</head>
<body>
    <div id = "wrapper">
        <a href="http://hcmute.edu.vn/"></a>
        <br />
        <header class="site-header">
            <nav class="site-nav">
                <ul>
                    <li><a href="index.php" <?php if ($page ==
'home') { ?> class='active'<?php ; }?>>ĐĂNG XUẤT</a></li>
                    <li><a href="giamsat.php" <?php if ($page ==
'giamsat') { ?> class='active'<?php ; }?>>GIÁM SÁT</a></li>
                    <li><a href="view.php" <?php if ($page == 'view') { ?>
class='active'<?php ; }?>>VIEW</a></li>
                        <?php if ($page=='') {} ;?>
                    <li><a href="chart.php" <?php if ($page ==
'chart') { ?> class='active'<?php ; }?>>CHART</a></li>
                        <?php if ($page=='') {} ;?>
                    <li><a href="getdata.php" <?php if ($page ==
'getdata') { ?> class='active'<?php ; }?>>DAY CHUYEN 1</a></li>
                        <?php if ($page=='') {} ;?>
                </ul>
            </nav>
        </header>

        <form action="control.php" method="post">
            <table>
                <tr>
                    <td colspan="2">Nhập sản phẩm </td>
                </tr>
                <tr>
                    <td>dây chuyền 1 :</td>
```

## PHỤ LỤC

```
</td>
</tr>
<tr>
    <td>dây chuyền 2 :</td>
    <td><input type="dc2" id="pass" name="dc2"></td>
</tr>

<tr>
    <td>dây chuyền 3 :</td>
    <td><input type="dc3" id="dc3" name="dc3"></td>
</tr>
<tr>
    <td colspan="2" align="center"><input
type="submit" name="btn_submit" value="gửi"></td>
</tr>
</table>

</form>
</form>
<div class="row" style="margin-top: 20px;">
<div class="col-md-8 col-md-offset-2">
    <a href="?light=on" class="btn btn-default btn-block btn-
lg">START</a>
    <br />
    <a href="?light=off" class="btn btn-default btn-block btn-
lg">STOP</a>
    <br />
    <a href="?motor=on" class="btn btn-default btn-block btn-
lg">DAY CHUYEN 1</a>
    <br />
    <a href="?motor=off" class="btn btn-default btn-block btn-
lg">DAY CHUYEN 2</a>
    <br />
    <div class="light-status well" style="margin-top: 5px; text-
align:center">
        </div>
    </div>
</div>

<? include('footer.php') ; ?>
```