# TRƯỜNG ĐẠI HỌC BÁCH KHOA TP. HỒ CHÍ MINH

**KHOA ĐIỆN – ĐIỆN TỬ**

**BỘ MÔN ĐIỀU KHIỂN TỰ ĐỘNG**

**NGUYỄN TRUNG TÍNH**

**HÀ TIẾN TÀI**

**NGUYỄN HÀ HIẾN**

**BÁO CÁO NGHIÊN CỨU KHOA HỌC**

**THIẾT KẾ THI CÔNG HỆ THỐNG BƠM SỬ DỤNG NĂNG LƯỢNG MẶT TRỜI VÀ ĐỘNG CƠ BLDC**

**GIẢNG VIÊN HƯỚNG DẪN**

**TS. NGUYỄN TRỌNG TÀI**

**TP. HỒ CHÍ MINH, 2020**

# TRƯỜNG ĐẠI HỌC BÁCH KHOA TP. HỒ CHÍ MINH

**KHOA ĐIỆN – ĐIỆN TỬ**

# NGUYỄN TRUNG TÍNH

# HÀ TIẾN TÀI

# NGUYẾN HÀ HIẾN

**BÁO CÁO NGHIÊN CỨU KHOA HỌC**

**THIẾT KẾ THI CÔNG HỆ THỐNG BƠM SỬ DỤNG NĂNG LƯỢNG MẶT TRỜI VÀ ĐỘNG CƠ BLDC**

**KỸ SƯ NGÀNH KỸ THUẬT ĐIỀU KHIỂN & TỰ ĐỘNG HÓA**

**GIẢNG VIÊN HƯỚNG DẪN**

**TS. NGUYỄN TRỌNG TÀI**

**TP. HỒ CHÍ MINH, 2020**

|  |
| --- |
|  |

|  |  |
| --- | --- |
|  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | TRƯỜNG ĐẠI HỌC BÁCH KHOA TP. HỒ CHÍ MINH  **KHOA ĐIỆN – ĐIỆN TỬ**  **BỘ MÔN: ĐIỀU KHIỂN TỰ ĐỘNG** | CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM  Độc lập - Tự do - Hạnh phúc | |  | *TP. HCM, ngày….tháng…..năm……..* | |  |

# ĐỀ CƯƠNG CHI TIẾT

|  |  |
| --- | --- |
| **TÊN ĐỀ TÀI:**THIẾT KẾ THI CÔNG HỆ THỐNG BƠM SỬ DỤNG NĂNG LƯỢNG MẶT TRỜI VÀ ĐỘNG CƠ BLDC | |
| **Cán bộ hướng dẫn: TS NGUYỄN TRỌNG TÀI** | |
| **Thời gian thực hiện:** Từ ngày………………..đến ngày…………… | |
| **Sinh viên thực hiện:**  **Nguyễn Trung Tính - 1713522**  **Hà Tiến Tài**  **Nguyễn Hà Hiến** | |
| **Nội dung đề tài:** *(Mô tả chi tiết mục tiêu, phạm vi, đối tượng, phương pháp thực hiện, kết quả mong đợi của đề tài)*  Nghiên cứu về bộ sạt mppt cho pin mặt trời ở mức cơ bản  Mô phỏng Điều khiển động cơ bằng pp định hướng từ thông + thiết kế mạch lái động cơ  Xây dựng giao diện điều khiển và sử dụng giao tiếp LoRa + Mobus RTU cơ bản | |
| **Kế hoạch thực hiện:** *(Mô tả kế hoạch làm việc và phân công công việc cho từng sinh viên tham gia)*   |  | | --- | | Giai đoạn 1 (6/2020->7/2020): Tìm hiểu các chi tiết của hệ thống năng lượng mặt trời, bộ điều khiển động cơ BLDC, xây dựng mô hình mô phỏng và nghiên cứu các giải pháp IoT phù hợp với đề tài đang nghiên cứu. | | Giai đoạn 2 (7/2020->9/2020): tập trung hoàn thành chi tiết phần cứng để xây dựng mô hình bơm nước bằng động cơ BLDC lấy năng lượng từ pin mặt trời, làm quen với các bộ điều khiển pin mặt trời, động cơ BLDC, IoT. | | Giai đoạn 3(9/2020->10/2020) : tập trung hoàn thành các bộ điều khiển, kết nối IoT giám sát hệ thống. | | Giai đoạn 4(10/2020->11/2020) : Vận hành thử nghiệm, khảo sát và chỉnh sửa các lỗi phát sinh không thể dự đoán trước. | | |
| **Xác nhận của Cán bộ hướng dẫn**  (Ký tên và ghi rõ họ tên) | TP. HCM, ngày….tháng …..năm…..  **Sinh viên**  (Ký tên và ghi rõ họ tên) |

MỤC LỤC

[Chương 1. TÊN CHƯƠNG 1 3](#_Toc367742496)

[1.1. Chủ đề cấp độ 2 3](#_Toc367742497)

[1.1.1. Chủ đề cấp độ 3 3](#_Toc367742498)

[1.1.2. Chủ đề cấp độ 3 3](#_Toc367742499)

[1.1.2.1. Chủ đề cấp độ 4 3](#_Toc367742500)

[Chương 2. TÊN CHƯƠNG 2 4](#_Toc367742501)

[2.1. Chủ đề cấp độ 2 4](#_Toc367742502)

[2.1.1. Chủ đề cấp độ 3 4](#_Toc367742503)

[2.1.1.1. Chủ đề cấp độ 4 4](#_Toc367742504)

[2.2. Chủ đề cấp độ 2 4](#_Toc367742505)

[2.2.1. Chủ đề cấp độ 3 4](#_Toc367742506)

[Chương 3. TÊN CHƯƠNG 3 5](#_Toc367742507)

[3.1. Chủ đề cấp độ 2 5](#_Toc367742508)

[3.1.1. Chủ đề cấp độ 3 5](#_Toc367742509)

[3.1.1.1. Chủ đề cấp độ 4 5](#_Toc367742510)

[3.2. Chủ đề cấp độ 2 5](#_Toc367742511)

DANH MỤC HÌNH VẼ

[Hình 1.1: Tên hình 1 3](#_Toc367742554)

DANH MỤC BẢNG

[Bảng 1.1: Tên bảng 1 3](#_Toc367742567)

[Bảng 2.1: Tên bảng 1 4](#_Toc367742568)

DANH MỤC TỪ VIẾT TẮT

TÓM TẮT LUẬN VĂN BẰNG TIẾNG VIỆT

ABSTRACT

MỞ ĐẦU

# GIỚI THIỆU ĐỀ TÀI

## Tính cấp thiết của nghiên cứu

Trong nước :

* Trong quá trình công nghiệp hoá, hiện đại hoá thì ngành năng lượng đống vai trò then chốt. Tại nước ta, ngành năng lượng đang đứng trước áp lực phải duy trì đà tăng trưởng, đồng thời phải giữ gìn môi trường và đảm bảo công bằng xã hội. Năng lượng ‘xanh’ đặt biệt là năng lượng mặt trời được xem là một giải pháp bổ xung hiệu quả cho nền công nghiệp năng lượng ở nước ta.
* Việt Nam là một nước có tiềm năng về năng lượng mặt trời với lợi thế là một nước nằm gần đường xích đạo, với số giờ nắng trung bình ở miền Trung và miền Nam khoản 2000-2600 giờ mỗi năm. Tính đến hiện này (2020), cả nước có tới 280 dự án điện mặt trời đang đăng kí đầu tư với tổng công suất lên tới 25000MW. Đồng thời các dự án điện mặt trời trong dân sinh cũng được phát triển mạnh và đã được thương mại hoá nhưng với chi phí khá cao và khó tiếp cận. Hiện nay đã có các nhóm nghiên cứu và ứng dụng thành công hệ thống năng lượng mặt trời vào những hệ thống cụ thể. Tiếp nối thành công đó, nhóm chúng em sẽ nghiên cứu hệ thống bơm phụ vụ tưới tiêu, bơm nước ao hồ sử dụng năng lượng mặt trời với chi phí tiết kiệm.
* Động cơ là thiết bị vô cùng phổ biến trong các hệ thống từ công nghiệp đến dân dụng. Rất nhiều loại động cơ điện, như động cơ DC có chỗi than, động cơ DC không chỗi than, động cơ đồng bộ và không đồng bộ ba pha ,… Tuỳ thuộc vào ứng dụng mà chúng có điểm mạnh và yếu khác nhau. Xét với ứng dụng sử dụng năng lượng từ pin mặt trời, các động cơ có nguồn cấp DC có ưu điểm nội bật giúp giảm bớt các bộ biến đổi công suất. Trong đó động cơ BLDC được sử dụng khá rộng rãi do vẫn giữ được các đặc tính tốt của động cơ DC có chỗi than đồng thời loại bỏ được nhược điểm phải bảo trì thường xuyên của động cơ DC truyền thống. Động cơ BLDC có nhiều ưu điểm nữa như công suất và tốc độ đa dạng, hiệu suất cao, gây tiếng ồn nhỏ và có độ bên cao. Cùng với sự phát triển mạnh mẽ của linh kiện bán dẫn công suất, các mạch điều khiển vận tốc hay mo-men của động cơ BLDC ngày càng được phát triển mạnh mẽ.
* Việc nghiên cứu và điều khiển động cơ BLDC đã được phát triển từ vài năm nay trong nước ta và phương pháp điều khiển hiệu quả chủ yếu là FOC(Field Oriented Control), sinwave control và square-wave control . Hiệu quả của phương pháp điều khiển FOC dựa vào hiệu quả của bộ điều khiển dòng và thường là bộ điều khiển PI kết hợp với PWM. Trong đề tài nghiên cứu này nhóm em sẽ thực hiện bộ điều khiển dòng dựa trên giải thuật Model Predictive Control. Đây là một giải thuật hiệu quả và khá đơn giản trong việc thực hiện.
* Thị trường Internet of Thing (IoT) đang tăng trưởng liên tục trong các năm gần đây ở nước ta. Việc áp dụng IoT và hệ thống giúp ta quản lý, giám sát, đồng thời phân tích dữ liệu thu thập được. Hiện nay các hướng nghiên cứu về ứng dụng công nghệ cao trong giám sát từ xa các thông số môi trường trong nông nghiệp thường được sử dụng các công nghệ truyền thống về truyền thông như Zigbee, wifi, GSM/GPRS, Bluetooth,… Trong các công nghệ kể trên có phạm vị hoạt động không cao từ 10 đến 100m. Với phạm vi hoạt động như vậy, các chuẩn truyền thông trên chỉ phù hợp cho việc giám sát trong một khu vực nhỏ. Khi khu vực giám sát lớn hoặc có nhiều khu vực, thì việc xây dựng mạng cảm biến trở lên phức tạp và tốn kém chi phí hơn. Nhằm mở rộng nhu cầu kết nối trên phạm vi rộng và tiêu thụ ít năng lượng, chuẩn truyền thông LORA đã ra đời với nhiều ưu điểm vượt trội so với các công nghệ sẵn có. Với mục đích chính là tiết kiệm năng lượng tiêu thụ và tăng khoảng cách truyền thông.

Ngoài nước:

* Phát triển năng lượng ‘xanh’ là xu thế, việc chế tạo và ứng dụng các hệ thống sử dụng năng lượng ‘xanh’ đang được nghiên cứu sôi nổi trên toàn thế giới. với những lợi ích kinh tế và môi trường mà nó mang lại, đây thật sự là một lĩnh vực đang được chú ý đến trên toàn cầu.
* Trong một thế giới đang chuyển qua năng lượng mặt trời, sử dụng năng lượng mặt trời cho hệ thống tưới có thể giúp ít rất nhiều và thúc đẩy nông nghiệp ở các nước nghèo. Khái niệm này được gọi là thuỷ lợi bằng năng lượng mặt trời và được sử dụng khá nhiều trên các quốc gia trên thế giới.
* Năng lượng mặt trời có thể xem là nguồn năng lượng dễ nhất để người nông dân tiếp cận, đặt biệt là các vùng có hạ tầng điện lưới chưa phát triển. Do đó bơm nước sử dụng năng lượng mặt trời đang ngày càng phổ biến trên thế giới.
* Việc nghiên cứu điều khiển động cơ BLDC trên thế giới phát triển từ lâu và đạt được những kết quả khả quan. Nhiều phương pháp được đề xuất, kết quả điều khiển ngày càng tốt và đã đưa vào các ứng dụng thật tế. Từ các ứng dụng công suất nhỏ như: trong các ổ đĩa quang, quạt làm mát, thiết bị văn phòng,… đến các ứng dụng công suất lớn hơn như xe điện, các hệ truyền động có công suất lên đến 100KW.
* Trên thế giới, giám sát và kiểm soát các hoạt động của cơ sở hạ tầng đô thị và nông thôn như cầu, đường ray tàu hoả, trang trại là một ứng dụng của IoT. Nó giúp cải thiện khả năng quản lý sự cố và phối hợp ứng phó khẩn cấp, giảm chi phí hoạt động trong tất cả các lĩnh vự cơ sở hạn tầng liên quan.

## Mục tiêu của nghiên cứu

Nghiên cứu về bộ sạt mppt cho pin mặt trời ở mức cơ bản

Mô phỏng Điều khiển động cơ bằng pp định hướng từ thông + thiết kế mạch lái động cơ

Xây dựng giao diện điều khiển và sử dụng giao tiếp LoRa + Mobus RTU cơ bản

## Đối tượng và phạm vi nghiên cứu

### Đối tượng của nghiên cứu

Động cơ BLDC

Giao tiếp Lora

Quy trình sạt acquy và pp điều khiển mppt

### Phạm vi của nghiên cứu

Mô phỏng pp điều khiển động cơ BLDC bằng pp vector FOC và thiết kế mạch lái

Giao tiếp Lora cơ bản

Phương pháp điều khiển bộ sạt acquy từ pin mặt trời

### Cấu trúc của báo cáo

# TỔNG QUAN VỀ HỆ THỐNG BƠM SỬ DỤNG NĂNG LƯƠNG MẶT TRỜI.

## Pin mặt trời

Nội dụng

## Bộ điều khiển sạt

Nội dung

## Động cơ bơm nước

Nội dung

## Giám sát hệ thống và khả năng tự động hoá

* Trước khi chọn thiết kế hệ thống giám sát, ta phải xét tới các đặc tính của đối tượng cần giám sát. Ở đây là hệ thống bơm nước dùng động cơ BLDC sử dụng pin mặt trời
* Với các đặc thù yêu cầu như việc lắp đặt hệ thống này ở nơi mà cơ sở hạ tầng chưa phát triển, vùng xa xôi gây khó khăn cho việc bảo trì, … Ta chọn LoRa làm giao thức truyền thông chính để có thể thu thập các giữ liệu cũng như điều khiển đối tượng.
* Thông tin sẽ được truyền về một node chính là nơi để upload lên database và ứng dụng giám sát. Mô hình kết nối được mô tả như sau

# XÂY DỰNG HỆ THỐNG BƠM NĂNG LƯỢNG MẶT TRỜI

Nội dung -> chưa biết viết gì ở đây

Có thể là nêu cách thiết kế 1 hệ thống năng lượng mặt trời -> có tài liệu

# THIẾT KẾ PHẦN CỨNG

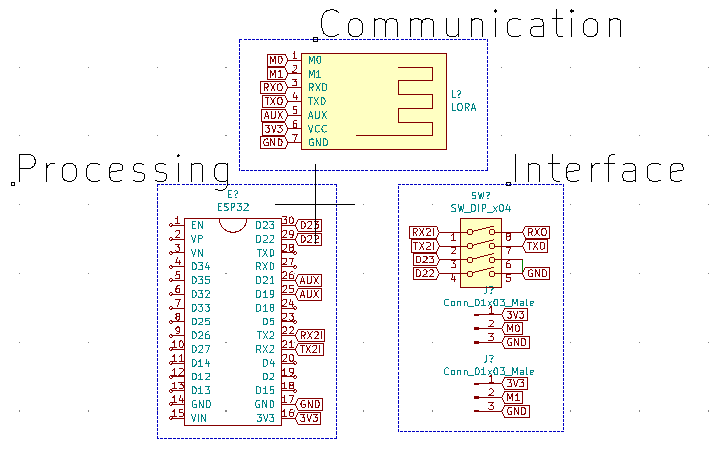
## Pin mặt trời và bộ sạt

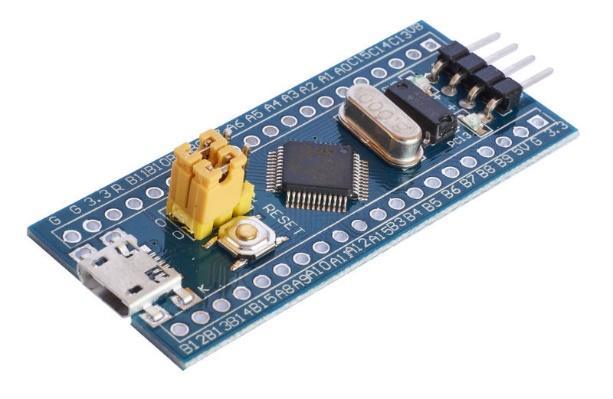
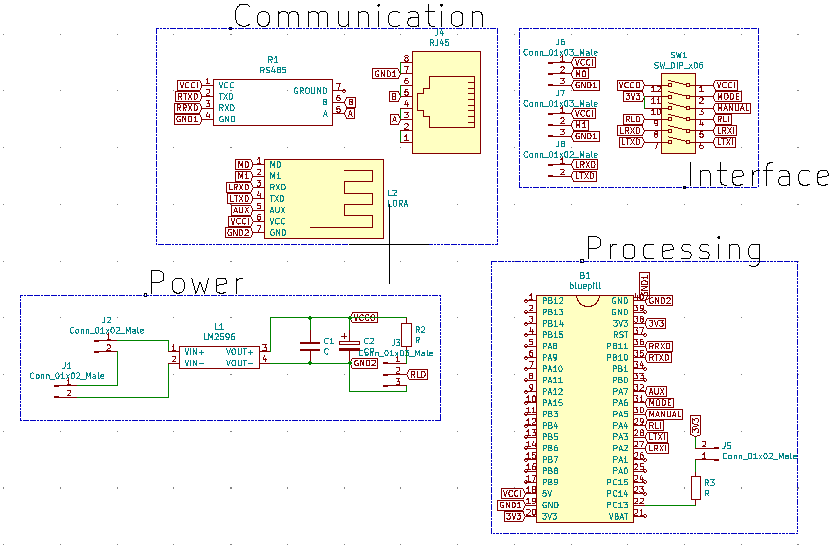
## Mạch lái động cơ

## Mạch node chính và mạch node phụ

* Mạch node chính
* Mạch node chính sẽ làm nhiêm vụ nhận dữ liệu từ các node phụ, từ đó tải dữ liệu lên internet thông qua giao thức mqtt. Phần cứng sẽ bao gồm 2 module chủ yếu gồm ESP32 AI-Thinker và LoRa SX1278-UART từ EByte
* ESP32 AI-Thinker
* Module trung tâm: Wifi BLE SoC ESP32 ESP-WROOM-32
* Tích hợp mạch nạp và giao tiếp UART CP2102
* Ra chân đầy đủ module ESP32, chuẩn cắm 2.54mm
* Tích hợp Led Status, nút Boot và Enable
* Kích thước: 28.33x51.45mm

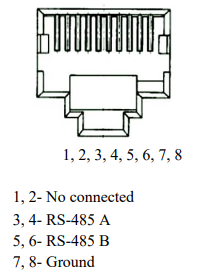
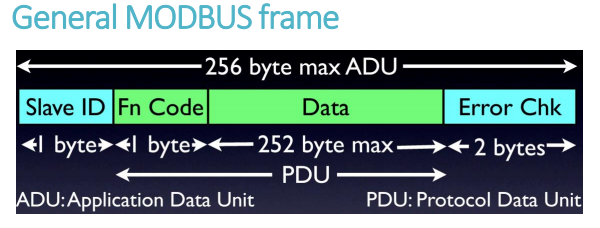
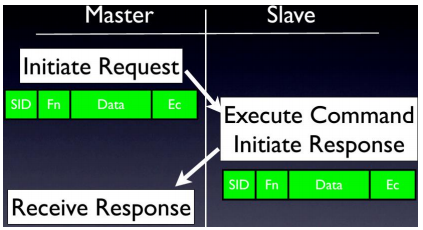
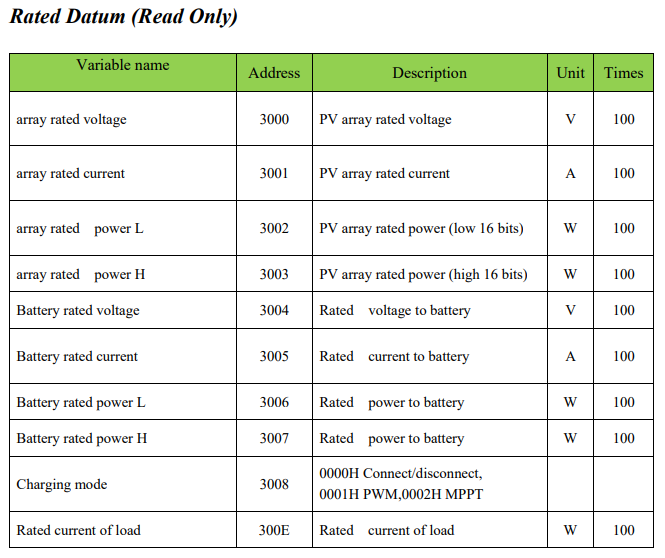
* LoRa SX1278-UART
* Model: E32-TTL-100 RF
* IC chính: SX1278 từ SEMTECH
* Điện áp hoạt động: 2.3 – 5.5 VDC
* Điện áp giao tiếp: TTL-3.3V
* Giao tiếp UART Data, 8N1, tốc độ từ 1200 – 115200 bps
* Tần số 410 – 441 Mhz
* Công suất: 20 dbm (100mW)
* Khoảng cách truyền tối đa trong điều kiện lý tưởng: 3000m
* Tốc độ truyền: 0.3 – 19.2 Kbps (mặc định 2.4 Kbps)
* 512 bytes bộ đệm, hỗ trợ 65536 địa chỉ cấu hình



* Mạch node phụ
* Mạch node phụ sẽ làm nhiệm vụ giao tiếp với MPTT thông qua Modbus và truyền các dữ liệu thu thập được qua LoRa về node chính. Phần cứng sẽ bao gồm 3 module chủ yếu gồm STM32F103c8t6-bluepill, LoRa SX1278-UART từ EByte và mạch chuyển giao tiếp RS485-UART dùng Max485
* Vi điều khiển: STM32F103C8T6.
* Điện áp cấp 5VDC qua cổng Micro USB sẽ được chuyển đổi thành 3.3VDC qua IC nguồn và cấp cho Vi điều khiển chính.
* Tích hợp sẵn thạch anh 8Mhz.
* Tích hợp sẵn thạnh anh 32Khz cho các ứng dụng RTC.
* Ra chân đầy đủ tất cả các GPIO và giao tiếp: CAN, I2C, SPI, UART, USB,...
* Tích hợp Led trạng thái nguồn, Led PC13, Nút Reset.
* LoRa SX1278-UART
* Model: E32-TTL-100 RF
* IC chính: SX1278 từ SEMTECH
* Điện áp hoạt động: 2.3 – 5.5 VDC
* Điện áp giao tiếp: TTL-3.3V
* Giao tiếp UART Data, 8N1, tốc độ từ 1200 – 115200 bps
* Tần số 410 – 441 Mhz
* Công suất: 20 dbm (100mW)
* Khoảng cách truyền tối đa trong điều kiện lý tưởng: 3000m
* Tốc độ truyền: 0.3 – 19.2 Kbps (mặc định 2.4 Kbps)
* 512 bytes bộ đệm, hỗ trợ 65536 địa chỉ cấu hình
* Điện áp hoạt động: 3 - 5VDC.
* Điện áp giao tiếp TTL: 3 - 5VDC.
* Khoảng cách truyền RS485 có thể lên đến 1km (khuyến nghị sử dụng dưới 800m và dây bus chuyên dụng cho RS485).
* Chuẩn chân cắm TTL 2.54mm.
* Có đèn led thông báo trạng thái truyền nhận RX và TX.
* 

# THIẾT KẾ PHẦN MỀM

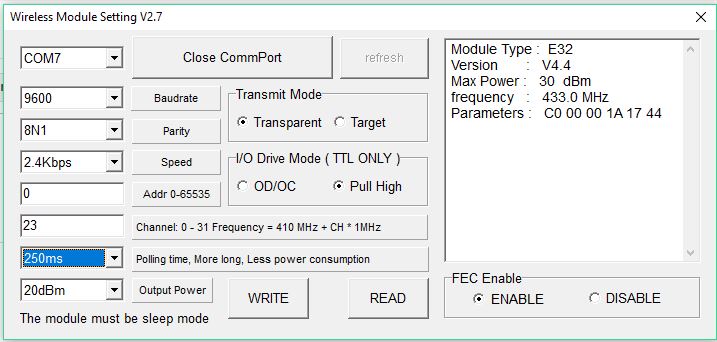
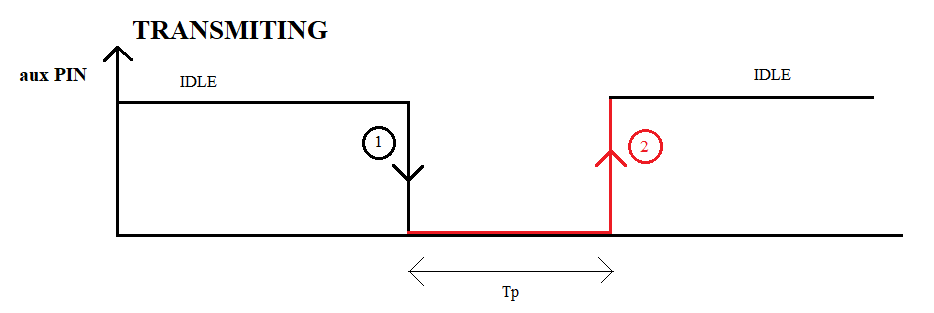
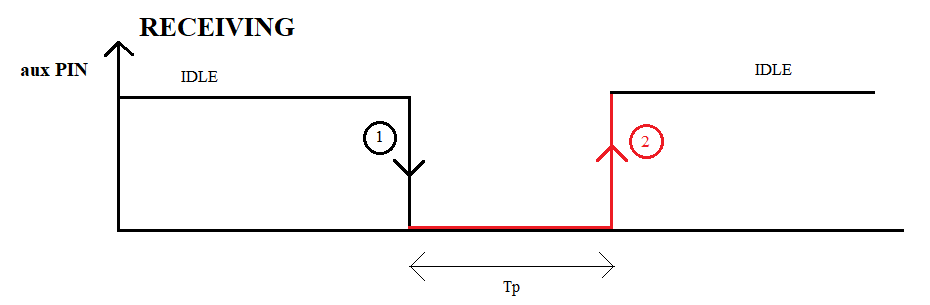
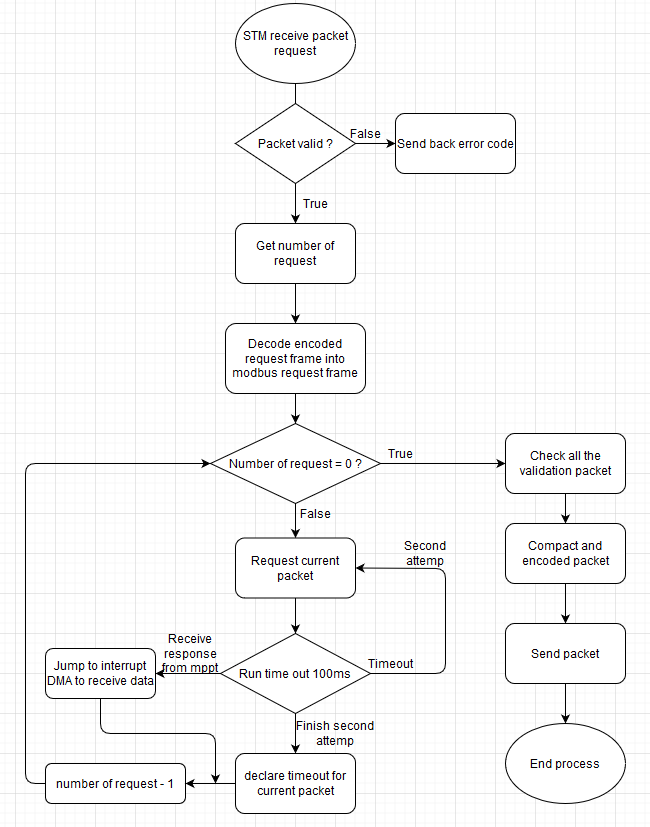
## Giao tiếp trên bộ mppt

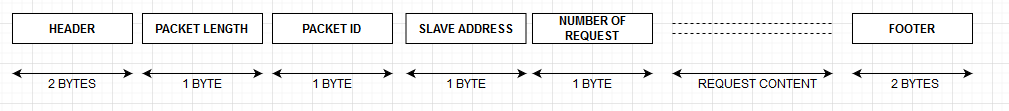
* Bộ MPPT có chuẩn giao tiếp vật lý RS485, hoạt động truyền nhận theo giao thức Modbus RTU
* Port của cổng được đấu nối theo jack RJ45 kiểu B gồm 8 dây, chia ra làm 4 cặp mỗi cặp hai dây bao gồm: Vcc, A, B, GND
* 
* Do nguồn vcc của port không được ổn định nên hãng khuyên không nên kết nối chân 1,2 làm nguồn cho board giao tiếp ngoài
* Port hoạt động ở baudrate 115200 bps, 8N1
* Nhắc đến giao thức Modbus RTU, đây là một giao thức request và response khá linh hoạt được dùng nhiều trong các hệ thống công nghiệp
* Một format của frame truyền Modbus RTU được quy định như sau
* 
* Trong đó slave ID là địa chỉ của MPPT, ở đây hãng quy định cho bộ này là 0x01
* Fn Code chính là function code để ta quy định loại hình data chúng ta muốn request, frame data sẽ khác và giống so với từng frame Modbus phụ thuộc vào function code nào
* Và cuối cùng là 2 byte check-sum của nguyên chuỗi data
* 
* Hình ảnh mô tả quy trình truyền và nhận của master và slave
* Về function code, mppt hỗ trợ 5 function code bao gồm 0x01, 0x03, 0x10, 0x05, 0x04
* Ví dụ như với các thanh ghi sau trong datasheet của hãng là các thanh ghi input register
* 
* Ta sẽ request với Modbus frame như sau: 0x01 0x04 0x30 0x00 0x00 0x0A

## Giải thuật điều khiển động cơ trên MCU

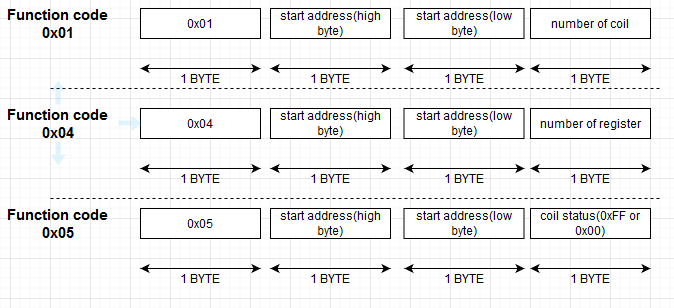
## Firmware giao tiếp và ứng dụng giám sát

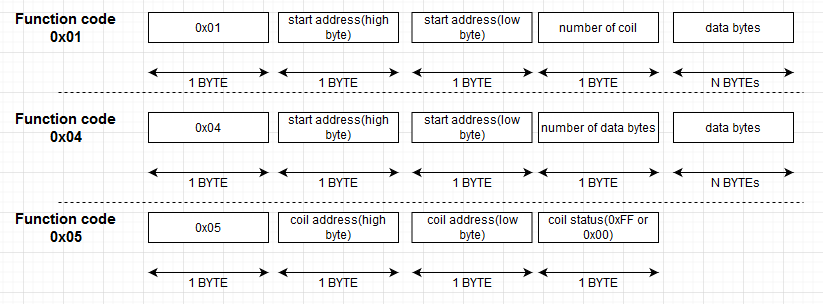
**Phần nhúng**

* 1. **Giao tiếp với module LoRa**
* **LoRa** là viết tắt của Long Range Radio được nghiên cứu và phát triển bởi Cycleo và sau này được mua lại bởi công ty Semtech năm 2012. Với công nghệ này, chúng ta có thể truyền dữ liệu với khoảng cách lên hàng km mà không cần các mạch khuếch đại công suất; từ đó giúp tiết kiệm năng lượng tiêu thụ khi truyền/nhận dữ liệu. Do đó, LoRa có thể được áp dụng rộng rãi trong các ứng dụng thu thập dữ liệu như sensor network trong đó các sensor node có thể gửi giá trị đo đạc về trung tâm cách xa hàng km và có thể hoạt động với battery trong thời gian dài trước khi cần thay pin.
* Module LoRa SX1278 UART từ Ebyte có các thông số về giao tiếp LoRa có thể tuỳ chỉnh qua phần mềm của hãng như tốc độ baudrate của cổng UART, tốc độ bit on air, địa chỉ của module và kênh của phát sóng của module, ...
* 
* Phần mềm cấu hình của hãng
* Để hai module có thể gửi và nhận của nhau, chúng phải có chung địa chỉ và chung kênh truyền
* Module có một chân AUX dùng để báo hiệu các trạng thái truyền và nhận để từ đó ta có thể đọc được data ra các chân UART. Biểu đồ thời gian khi truyền và nhận của chân AUX được diện tả như sau
* 
* Khi ta gửi các byte dữ liệu qua cổng UART vào module, module sẽ đợi thời gian khoảng 3 byte dữ liệu để chốt một gói data. Nếu trong khoảng thời gian nay không có động tĩnh trên UART, module sẽ chốt và tự hiểu là người dùng đã gửi xong một packet data. Tại giai đoạn này chân AUX sẽ xuất cạnh xuống báo hiệu việc bắt đầu truyền data on air
* Khi Aux bắt đầu ở giai đoạn (1), data UART đưa vào sẽ không được nhận xử lý
* Khoảng thời gian Tp là khoảng thời gian mà module đang xử lý truyền tín hiếu không dây. Tuỳ vào độ dài của dữ liệu mà thời gian Tp này sẽ khác nhau. Nếu chọn chế độ FEC (forward error checking) thì khoảng thời gian này sẽ tăng lên nhiều lần, ngược lại độ tin cậy của gói dữ liệu nhận được sẽ cao hơn
* Sau khi các byte dữ liệu cuối cùng được gửi ra khỏi module lúc này chân Aux sẽ xuất xung cạnh lên trở về IDLE. Đây là lúc báo hiều người dùng có thể bắt đầu đưa chuỗi stream data mới vào. Theo datasheet quy định, chiều dài một gói packet một lần đưa vào có thể lên tối đa tới 512 bytes
* 
* Khi module nhận được data từ sóng LoRa và đã decode thành công, Aux pin xuất xung cạnh xuống báo hiệu wireless data đã được nhận và dữ liệu được toàn vẹn
* Khoảng thời gian Tp là khoảng thời gian module xuất data qua UART
* Khi tất cả data đã được truyền qua UART Aux pin xuất xung cạnh lên và trở về mức IDLE. Đây là lúc ta đọc dữ liệu data từ UART
  1. **Tạo thư viện request Modbus một cách linh động**
* Ngôn ngữ lập trình chủ yếu ở phần nhúng ta dùng C++. Với đặc trung thế mạnh OOP của C++ ta có thể giải quyết vấn đề trên một cách có hệ thống hơn
* Ta viết nên các thư viện để handle những chuỗi request và response từ Modbus cũng như encode và decode các chuỗi byte theo định nghĩa của người lập trình. Từ đó rút ngắn chiều dài packet phải gửi không dây, tăng độ tin cậy của đường truyền hơn
* Sau đây là một sơ đồ giải thuật khi gửi một hệ các request xuống node phụ
* 
* Với ý tưởng thực hiện model trên được diễn tả như sau. Các data trên các thanh địa chỉ của Modbus thường nằm liên tục, tuy nhiên khi giám sát các thông số, yêu cầu chỉ cần lấy những thông số cần thiết để giám sát. Mà các địa chỉ thanh ghi các thông số này không liên tục nhau
* Từ đây ta có 2 hướng giải quyết
* Đọc hết tất cả những thanh ghi không cần thiết để có tính liên tục rồi gửi tất cả chúng về node chính để lọc => **lượng data dư nhiều, gây hao phí đường truyền**
* Đọc những thanh ghi cần thiết bằng cách request nhiều lần với mỗi lần ở những địa chỉ khác nhau => **có thể thực hiện nhiều function code trong một lần gửi, chiều dài data gửi về node chính giảm đi**
* Chính vì lý do này ta viết một thư viện để có thể xử lí tự động nhiều request. Đó cũng chính là nội dung của flowchart ở trên
* Để có thể tạo ra một frame truyền nhiều request gửi xuống node phụ, ta không thể ghép các frame Modbus của mỗi request vào với nhau rồi gửi. Như vậy node phụ cũng khó có thể tách các request đó ra và chiều dài chuỗi byte truyền trở nên bị dư. Chính vì thế ta phải encode nhiều frame request theo định dạng sau



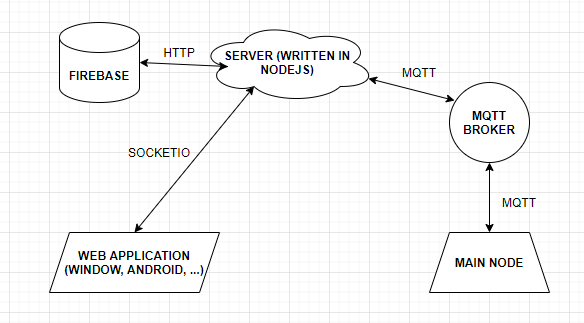
* **Header, Footer** và **packet length** sẽ là những yếu tố chung để lọc ra một frame data hoàn chỉnh
* **Packet ID** là con số sẽ xác định chức năng của packet gửi tới. Ở đây chức năng của node phụ sẽ chỉ giao tiếp với MPPT nên sẽ có riêng một packet ID là 38. Ngoài ra còn có các packet ID cho các Error handler. Trong tương lai nếu mở rộng sẽ có các packetID cho các chức năng bật tắt IO riêng biệt. Với mỗi packet ID các data theo sau nó sẽ có các format khác nhau
* **Slave address** chính là địa chỉ của slave Modbus mà node phụ giao tiếp tới
* **Number of request** là số request tổng cộng
* **Request content** bao gồm nhiều frame request được encode với format được nêu như sau

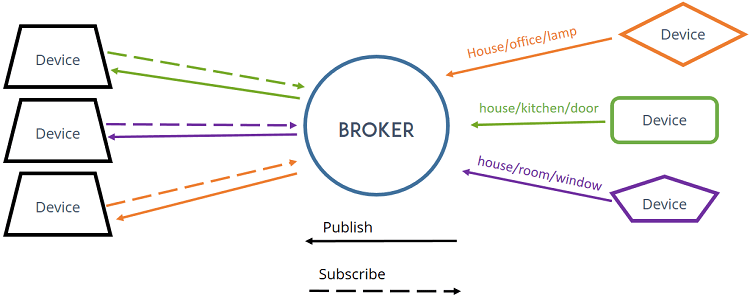
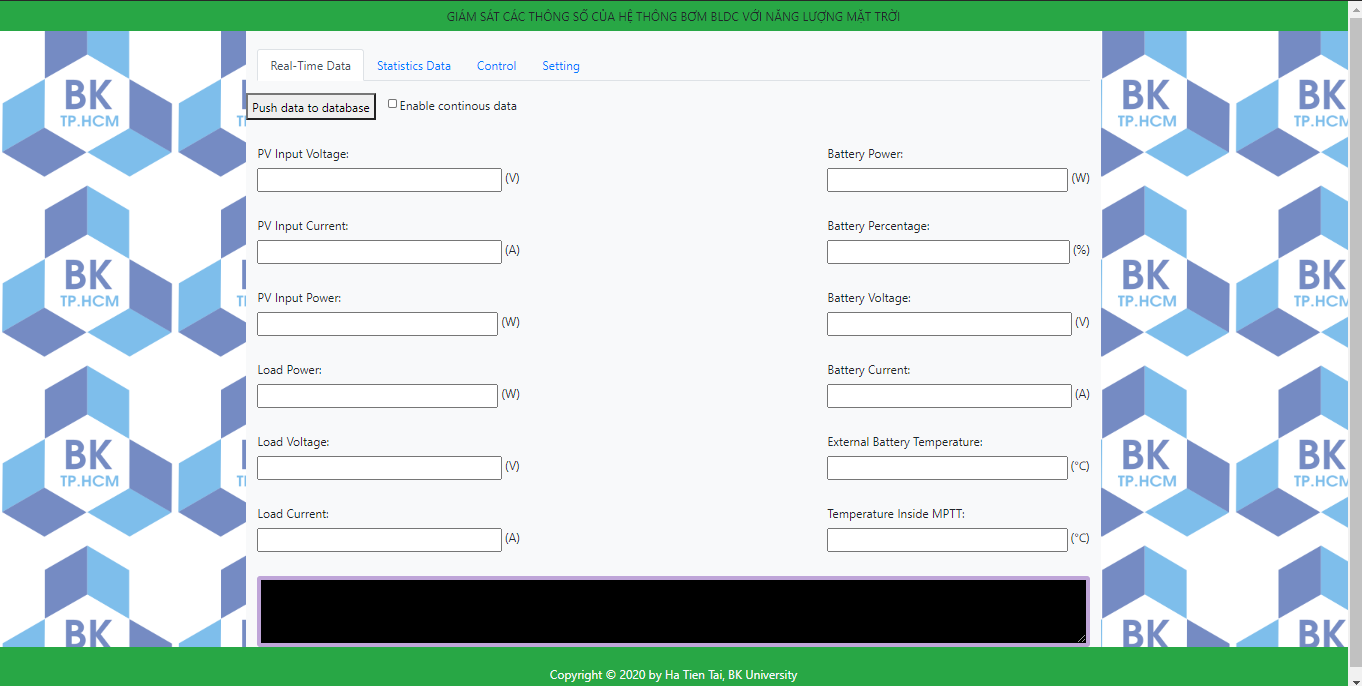


* Hình trên là ví dụ cho ba function code 0x01, 0x04, 0x05
* Khi node phụ nhận được giả sử một chuỗi byte gồm ba request này, với từng frame request trên node phụ sẽ construct lên một object class. Từng object class tương ứng với một request sẽ tự tạo một frame request Modbus đầy đủ với CRC và sẽ hoạt động như flow chart ở mục trên
* Ví dụ như một frame: 0x04 0x31 0x00 0x0A thì node phụ sẽ decode thành chuỗi request đầy đủ 0x01 0x04 0x31 0x00 0x00 0x0A 0x7E 0xF1 và gửi xuống cho MPPT
* Như thấy ở trên thay vì gửi một chuỗi đầy đủ với chuỗi encoded ta đã tiết kiệm đi được 4 byte
* Và tương tự như vậy với chuỗi data response đóng gói để gửi về node chính ta cũng sẽ có kiểu encode như sau
* 
* Ví dụ cho ba function code 0x01, 0x04, 0x05
* Với kiểu trên, cho từng request ta sẽ tiết kiệm được 1 byte

**Application for web server**

* Phía application giám sát ta sẽ viết một giao diện trên nên tảng web và server
* Ngôn ngữ chính ta viết cho web server ở đây là Nodejs, giao diện hiển thị ta dùng EJS engine đi kèm với bootstrap css
* Mô hình giao tiếp giữa web server và node chính được mô tả tổng quát ở hình sau

****

* 1. **MQTT ( Message Queue Telemetry Transport)**
* Đây là một giao thức truyền thông điệp (message) theo mô hình publish/subscribe (xuất bản – theo dõi), sử dụng băng thông thấp, độ tin cậy cao và có khả năng hoạt động trong điều kiện đường truyền không ổn định.
* Kiến truc của MQTT gồm 2 phần chính là Broker và Clients
* Trong đó, broker được coi như trung tâm, nó là điểm giao của tất cả các kết nối đến từ client. Nhiệm vụ chính của broker là nhận mesage từ publisher, xếp các message theo hàng đợi rồi chuyển chúng tới một địa chỉ cụ thể. Nhiệm vụ phụ của broker là nó có thể đảm nhận thêm một vài tính năng liên quan tới quá trình truyền thông như: bảo mật message, lưu trữ message, logs,…
* Client thì được chia thành 2 nhóm là publisher và subscriber. Client là các software components hoạt động tại phần nhúng nên chúng được thiết kế để có thể hoạt động một cách linh hoạt (lightweight). Client chỉ làm ít nhất một trong 2 việc là publish các message lên một topic cụ thể hoặc subscribe một topic nào đó để nhận message từ topic này.
* 
* Các ưu điểm của giao thức mqtt
  + Tăng khả năng mở rộng
  + Giảm đáng kể tiêu thụ băng thông mạng
  + Giảm tốc độ cập nhật xuống giây
  + Rất phù hợp cho điều khiển và do thám
  + Tối đa hóa băng thông có sẵn
  + Chi phí cực nhẹ
  + Rất an toàn với bảo mật dựa trên sự cho phép
* Ở ứng dụng ở đây khi frame dữ liệu đã encode đến từ node chính, node chính sẽ publish dữ liệu này để từ đó server là người subscribe vào topic này thấy được dữ liệu gửi lên
* Packet nhận được sẽ được decode phía server và chuyển thành những object json để có thể gửi cho các web application bằng giao thức socketio
  1. **SocketIO**
* **Socket.io** là một module trong Node.js được phát triển vào năm 2010. Nó được phát triển để sử dụng các kết nối mở để tạo điều kiện giao tiếp thời gian thực, trả về giá trị thực ở tại thời điểm đó. **Socket.io** cho phép giao tiếp hai chiều giữa máy khách và máy chủ. Giao tiếp hai chiều được bật khi máy khách có **Socket.io** trong trình duyệt và máy chủ cũng đã tích hợp gói **Socket.io**
* Nó được sử dụng trong việc xây dựng các ứng dụng web real-time cần tốc độ phản hồi ngay lập tức như: chat, trực tiếp bóng đá,.... **Socket.io**xây dựng dựa trên Engine.IO, đầu tiên nó sẽ thiết lập một kết nối long-polling, sau đó cố gắng nâng cấp lên các kết nối khác tốt hơn giống như WebSocket.
* Chính vì đặc thù trên của SocketIO là tính realtime giữa server và client, nên ta đã chọn SocketIO để gửi dữ liệu nhận được hiển thị lên web app
* Giao diện mô hình web app chạy trên trình duyệt
* 

# KẾT QUẢ THỰC NGHIỆM, ĐÁNH GIÁ VÀ HƯỚNG PHÁT TRIỂN

## Kết quả thực nghiệm bộ sạt

Nội dung

## Kết quả điều khiển động cơ BLDC

Nội dung

## Kết quả thực nghiệm bộ giám sát

* Mạch node chính đã được đóng hộp



* Mạch node phụ đã được đóng hộp
* 
* Lắp đặt node phụ hệ thống bơm
* 

## Đánh giá hệ thống

Nội dung

## Hướng phát triển

* Mở rộng hệ thống ra sao ?

TÀI LIỆU THAM KHẢO

Theo chuẩn IEEE