TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

**TRƯỜNG ĐIỆN – ĐIỆN TỬ**

----- □ & □ -----



**BÁO CÁO ĐỒ ÁN TỐT NGHIỆP**

**Đề tài: Thiết kế hệ thống GPS RTK**

**Hà Nội, tháng 12/2022**

# MỤC LỤC

[MỤC LỤC 2](#_Toc157360119)

[DANH MỤC HÌNH VẼ 4](#_Toc157360120)

[DANH MỤC BẢNG 5](#_Toc157360121)

[CHƯƠNG 1. CƠ SỞ LÝ THUYẾT 6](#_Toc157360122)

[1.1 Công nghệ truyền thông không dây LoRa 6](#_Toc157360123)

[1.1.1 Tổng quan về công nghệ truyền thông không dây 6](#_Toc157360124)

[1.1.2 Giới thiệu về LoRa 8](#_Toc157360125)

[1.1.3 Tỷ lệ mã hóa (Coding Rate) 9](#_Toc157360126)

[1.1.4 Hệ số trải phổ (Spread Factor) 9](#_Toc157360127)

[1.1.5 Băng thông 10](#_Toc157360128)

[1.2 Kỹ thuật đo RTK trong hệ thống GNSS 11](#_Toc157360129)

[1.2.1 Tổng quan về hệ thống GNSS 11](#_Toc157360130)

[CHƯƠNG 2. THIẾT KẾ HỆ THỐNG GPS RTK 13](#_Toc157360131)

[2.1 Sơ đồ kiến trúc hệ thống và chức năng 13](#_Toc157360132)

[2.2 Sơ đồ triển khai chi tiết từng khối 13](#_Toc157360133)

[2.2.1 Sơ đồ triển khai chi tiết thiết bị Rover 13](#_Toc157360134)

[2.2.1 Sơ đồ triển khai chi tiết thiết bị Base 18](#_Toc157360135)

[2.3 Công suất tiêu thụ của hệ thống 21](#_Toc157360136)

[2.4 Thiết kế phần cứng hệ thống 22](#_Toc157360137)

[CHƯƠNG 3. THIẾT KẾ GIAO TIẾP GIỮA THIẾT BỊ BASE – ROVER 23](#_Toc157360138)

[3.1 Giới thiệu về platform STM32CubeIDE và VSCode 23](#_Toc157360139)

[3.2 Cấu hình thiết bị Rover 23](#_Toc157360140)

[3.2.1 ADC 23](#_Toc157360141)

[3.2.2 LoRa 28](#_Toc157360142)

[3.2.3 Cấu hình GPS 30](#_Toc157360143)

[3.3 Cấu hình thiết bị Base 30](#_Toc157360144)

[3.3.1 Cấu hình GPS 30](#_Toc157360145)

[3.4 Thiết kế phần mềm giao tiếp 30](#_Toc157360146)

[CHƯƠNG 4. ĐÁNH GIÁ, KIỂM THỬ 32](#_Toc157360147)

# DANH MỤC HÌNH VẼ

[*Hình 4‑1 Mô hình truyền thông không dây 19*](#_Toc123077290)

[*Hình 4‑2 Các chuẩn tương ứng với các mạng truyền thông tương ứng 20*](#_Toc123077291)

[*Hình 4‑3 So sánh tốc độ truyền và khoảng cách truyền của Bluetooth, Zigbee và WiFi 22*](#_Toc123077292)

[*Hình 4‑4 Hai loại tín hiệu Up-chirp và Down-chirp trong kĩ thuật CSS 25*](#_Toc123077293)

[*Hình 4‑5 Các giá trị CR và dữ liệu tăng thêm tương ứng (Datasheet SX1278) 26*](#_Toc123077294)

[*Hình 4‑6 Mã hóa một Symbol bằng 2SF chip 26*](#_Toc123077295)

[*Hình 4‑7 Ba mức băng thông thường dùng trong mạng LoRa 27*](#_Toc123077296)

[*Hình 5‑1 Hình ảnh thiết bị giám sát nhiệt độ không dây 29*](#_Toc123077297)

[*Hình 5‑2 Hình ảnh ứng dụng 30*](#_Toc123077298)

[*Hình 6‑1 Kiến trúc chung của nút cảm biến không dây 32*](#_Toc123077299)

[*Hình 6‑2 Công suất tiêu thụ trong các chế độ hoạt động 34*](#_Toc123077300)

[*Hình 6‑3 Sơ đồ khối mô tả hệ thống 36*](#_Toc123077301)

[*Hình 6‑4 Sơ đồ khối sạc pin 37*](#_Toc123077302)

[*Hình 6‑5 Sơ đồ nguyên lý khối nguồn LDO RT9013 38*](#_Toc123077303)

[*Hình 6‑6 Sơ đồ nguyên lý tụ bypass cho STM32 38*](#_Toc123077304)

[*Hình 6‑7 Sơ đồ nguyên lý vi điều khiển STM32 39*](#_Toc123077305)

[*Hình 6‑8 Sơ đồ nguyên lý khối thạch anh ngoài cho vi điều khiển STM32 39*](#_Toc123077306)

[*Hình 6‑9 Khối giám sát điện áp pin LiPo 40*](#_Toc123077307)

[*Hình 6‑10 Sơ đồ nguyên lý module Ra-02 LoRa 40*](#_Toc123077308)

[*Hình 6‑11 Sơ đồ nguyên lý khối debug 41*](#_Toc123077309)

[*Hình 6‑12 Sơ đồ nguyên lý các nút nhấn và LED 41*](#_Toc123077310)

[*Hình 6‑13 Sơ đồ nguyên lý khối hiển thị LED 7 thanh 42*](#_Toc123077311)

[*Hình 6‑14 Sơ đồ nguyên lý khối nguồn LDO 43*](#_Toc123077312)

[*Hình 6‑15 Sơ đồ nguyên lý vi điều khiển ESP32 43*](#_Toc123077313)

[*Hình 6‑16 Sơ đồ nguyên lý các nút nhấn và LED 44*](#_Toc123077314)

[*Hình 6‑17 Sơ đồ nguyên lý khối debug và nạp code 44*](#_Toc123077315)

[*Hình 6‑18 Sơ đồ nguyên lý khối hỗ trợ nạp 45*](#_Toc123077316)

[*Hình 6‑19 Sơ đồ nguyên lý module Ra-02 LoRa 45*](#_Toc123077317)

# DANH MỤC BẢNG

[*Bảng 1‑1 Yêu cầu đặt ra 7*](#_Toc123077275)

[*Bảng 1‑2 Thời gian chuyển đổi ứng với độ phân giải của cảm biến DS18B20 8*](#_Toc123077276)

[*Bảng 1‑3 Thông số của module LoRa sử dụng trong Gateway và Node 9*](#_Toc123077277)

[*Bảng 1‑4 Bảng giá trị SNR 10*](#_Toc123077278)

[*Bảng 1‑5 Phân chia vùng nhớ OTA theo dữ liệu lưu trữ và địa chỉ 13*](#_Toc123077279)

[*Bảng 2‑1 Phân chia công việc chung 14*](#_Toc123077280)

[*Bảng 3‑1 Phân chia công việc của Trần Quang Minh 16*](#_Toc123077281)

[*Bảng 3‑2 Phân chia công việc của Nguyễn Văn Dũng 17*](#_Toc123077282)

[*Bảng 3‑3 Phân chia công việc của Nguyễn Trọng Phong 18*](#_Toc123077283)

[*Bảng 4‑1 So sánh các công nghệ truyền thông không dây 24*](#_Toc123077284)

[*Bảng 6‑1 Các chế độ của thiết bị 33*](#_Toc123077285)

[*Bảng 6‑2 Công suất tiêu thụ của từng phần tử 35*](#_Toc123077286)

# CƠ SỞ LÝ THUYẾT

## Công nghệ truyền thông không dây LoRa

### Tổng quan về công nghệ truyền thông không dây

**Khái niệm:**

Wireless: vô tuyến, không dùng dây dẫn

Truyền thông không dây là việc truyền tải thông tin qua một khoảng cách mà không cần dây dẫn làm môi trường truyền.

**Mô hình truyền thông không dây:**

A diagram of a radio signal

Description automatically generated

Hình 4‑1 Mô hình truyền thông không dây

**Phân loại các công nghệ truyền thông không dây**

Mạng truyền thông được phân loại như sau:

* Dựa vào dạng tín hiệu thông tin (Information Signals): Digital, Analog
* Dựa vào phạm vi (Scale): LAN, WAN, MAN, Internet
* Dựa vào công nghệ truyền dẫn (Transmission Technology): Broadcast, Point – to – Point
* Dựa vào phương tiện truyền dẫn (Transmission Medium): Wired, Wireless

…

Mạng không dây được phân loại dựa theo:

* Dựa vào kiến trúc (Infrastructure): Mạng có kiến trúc
* Sử dụng các node và gateway có dây và không dây
* Mỗi node chính được định nghĩa trước vai trò trong mạng
* Mạng có kiến trúc: Mạng tế bào, WLAN dùng access point, …
* Mạng không có kiến trúc (Ad – hoc network)
* Không có sự sắp xếp trước
* Tập hợp ngẫu nhiên các node lại với nhau tạo thành một mạng
* Dựa vào chuẩn (Standardizations)
* 3GPP: WCDMA, HSDPA, 4G LTE
* 3GPP2: CDMA2000, EV – DO, UMB
* IEEE: IEEE802.11, IEEE802.16

A diagram of different types of internet systems

Description automatically generated

Hình 4‑2 Các chuẩn tương ứng với các mạng truyền thông tương ứng

* Dựa vào phạm vi (Scale)
* Body Area Networks: như cảm biến được gắn vào người
* Personal Area Networks: Home networking, Bluetooth và ZigBee
* Local Area Networks: Wireless LAN, WiFi
* Metropolitan Area Networks (Mạng đô thị): WiMAX
* Wide Area Networks: Cellula và Satellite networks
* Dựa vào sự di động (Mobility)
* Mạng không dây cố định: người dùng đã kết nối được xem như ở yên một chỗ
* Mạng không dây di động: một phần trong các thiết bị không dây là di dộng: Việc thay đổi điểm kết nối vào mạng có thể được thực hiện khi thiết bị vẫn online.

VD: Cellula network, mobile ad hoc network (MANET)

* Dựa vào dạng tín hiệu (Signal)
* Dựa vào băng thông (Bandwidth)
* Narrowband Wireless Network: Mạng băng hẹp.
* Broadband Wireless Network: Mạng băng rộng (UWB, WiMax, LTE, ...)

### Giới thiệu về LoRa

LoRa (Long Range Radio) là một nền tảng công nghệ không dây có công  
suất thấp và phạm vi xa, sử dụng dải tần miễn phí. Mục đích của tạo ra công nghệ  
LoRa là nhằm loại bỏ repeater, giảm giá thành thiết bị. Nó là một lớp vật lí, được  
sử dụng cho giao tiếp với khoảng cách lớn. LoRa là một kĩ thuật điều chế dựa trên  
kĩ thuật trải phổ Spread-Spectrum và biến thể của Chirp Spread Spectrum (CSS),  
được nghiên cứu và phát triển bởi Cycleo và được mua lại bởi công ty Semtech  
vào năm 2012. Thay vì sử dụng công suất lớn và băng thông hẹp thì tín hiệu được  
trải ra băng thông rộng hơn với mức công suất nhỏ hơn (tổng mức công suất không  
đổi) mà bên thu có bộ giải mã thích hợp vẫn có thể nhận được chính xác dữ liệu.  
Nhờ áp dụng kĩ thuật trải phổ mà tín hiệu truyền đi ít bị suy hao và chống nhiễu  
tốt hơn, do đó có thể truyền tín hiệu đi xa trong khi truyền với mức công suất thấp.

Dữ liệu truyền đi được mã hóa thành các “Chirp” là tín hiệu hình sin có tần  
số thay đổi tuyến tính theo thời gian, có hai loại tín hiệu Chirp: Up – Chirp (Tần số tín hiệu tăng dần theo thời gian) và Down – Chirp (Tần số tín hiệu giảm dần theo thời gian).

A diagram of a graph

Description automatically generated with medium confidence

Hình 4‑4 Hai loại tín hiệu Up-chirp và Down-chirp trong kĩ thuật CSS

Trên một dải băng thông cố định cho phép truyền nhận dữ liệu nhờ sử dụng  
hệ số trải phổ trực giao khác nhau. Điều này làm tăng tính linh hoạt trong thiết kế  
hệ thống khi các thiết bị có thể tối ưu về khoảng cách, tốc độ đường truyền hay là  
về tiết kiệm năng lượng. Để tăng độ nhạy tín hiệu cho thiết bị, mỗi gói tin LoRa  
truyền đi đều có phần mở đầu (Preamble). Phần mở đầu là một chuỗi các chu kì  
phát tín hiệu. Trong mỗi chu kỳ, tín hiệu được truyền đi có tần số tăng tuyến tính  
bắt đầu từ tần số thấp nhất đến tần số cao nhất. Hết một chu kì tương ứng với việc  
tạo ra một tín hiệu “chirp”. Kết thúc của phần mở đầu (Preamble) được phát hiện  
khi bộ lọc nhận được một tín hiệu “Down chirp”, tức là tần số tín hiệu giảm dần.  
Hơn nữa khả năng chống nhiễu và bảo mật thông tin cũng được cải thiện vì một  
gói tin chỉ được xử lý khi đã xác nhận đúng phần mở đầu.

Các tham số đặc trưng cho điều chế LoRa:

* Tỷ lệ mã hóa
* Hệ số trải phổ
* Băng thông

### Tỷ lệ mã hóa (Coding Rate)

Kỹ thuật FEC (Forward error correction) được sử dụng trong LoRa để tăng  
khả năng xử lý dữ liệu bên phía thiết bị nhận. FEC là một kỹ thuật mã hóa kênh  
dùng để phát hiện và sửa lỗi gói tin truyền đi bằng cách thêm vào sau gói tin một  
đoạn mã được tính toán từ dữ liệu bên trong rồi được gửi đi cùng với kiểu FEC.

Vì vậy độ lớn của gói tin phụ thuộc vào hệ số CR (coding rate). Nhà sản xuất LoRa  
đã đặt giá trị mã CR trong khoảng từ 0 đến 4, ở đây CR = 0 nghĩa là không có FEC.  
LoRa sử dụng tỉ lệ mã code là 4/5, 4/6, 4/7, 4/8. Hình dưới tương ứng với tỉ số  
giữa độ lớn mã FEC so với gói tin.

Table

Description automatically generated

Hình 4‑5 Các giá trị CR và dữ liệu tăng thêm tương ứng (Datasheet SX1278)

Nếu CR = 4/8 thì cứ mỗi 4bit data nó sẽ được mã hóa bởi 8bit, tức là chipset  
LoRa phải gửi gấp đôi dữ liệu cần truyền. Do đó chúng ta có thể sử  
dụng CR thấp để tăng throughput nhưng độ nhạy sẽ kém đi do khả năng tự phục  
hồi dữ liệu của chipset LoRa sẽ thấp hơn.

### Hệ số trải phổ (Spread Factor)

SF xác định số lượng chip khi mã hóa tín hiệu đã được điều chế tần số, biểu diễn số lượng bit thông tin tối đa mà một symbol có thể mang theo, có giá trị từ 7 – 12.

A screenshot of a computer

Description automatically generated with low confidence

Hình 4‑6 Mã hóa một Symbol bằng 2SF chip

Mỗi *Symbol* mang thông tin trong LoRa sẽ được mã hóa thành 2SF “chip”.  
Việc tăng hệ số trải phổ lên một sẽ làm tăng số lượng “chip” mã hóa một *symbol*làm cho tốc độ truyền nhận dữ liệu giảm đi.

LoRa thường sử dụng các hệ số trải phổ trực giao từ 7 đến 12 (riêng với chip  
“SX1277” là từ 6 đến 9), cho phép các thiết bị có thể sử dụng nhiều hệ số khác  
nhau trong một hệ thống mà không gây nhiễu sang nhau. Hệ số trải phổ ảnh hưởng  
trực tiếp đến tốc độ truyền dữ liệu cũng như khoảng cách truyền. Ngoài ra khả  
năng kháng nhiễu cũng được tăng lên khi sử dụng hệ số trải phổ cao hơn.

### Băng thông

Băng thông là thông số quan trọng nhất của điều chế LoRa. Một *Symbol* bao  
gồm 2SF chip được truyền đi trong toàn dải băng thông. Loại tín hiệu sử dụng là  
“Up – chirp”. Tần số tín hiệu tăng dần đến khi đạt đến tần số tối đa của dải băng  
thông thì sẽ quay ngược lại tần số nhỏ nhất và lại tiếp tục tăng, cứ như vậy trong  
suốt khoảng thời gian truyền một Symbol. Trong kỹ thuật LoRa thì tốc độ chirp  
phụ thuộc vào độ rộng băng thông, dải băng thông sử dụng càng rộng thì thời gian  
một chirp được truyền đi trong không khí càng lớn hay tốc độ đường truyền càng  
giảm. LoRa thường sử dụng 3 mức băng thông phổ biến là: 125kHz, 250kHz và 500kHz.

Chart

Description automatically generated with low confidence

*Hình 4‑7 Ba mức băng thông thường dùng trong mạng LoRa*

Có thể nói SF, BW và CR là 3 thông số cơ bản và quan trọng của chipset  
LoRa. Trong đó, SF và BW sẽ ảnh hưởng thời gian và khoảng cách truyền dữ liệu;  
CR thì chỉ ảnh hưởng thời gian truyền dữ liệu. Tùy yêu cầu của ứng dụng cụ thể  
về khoảng cách, tốc độ gửi dữ liệu, v.v… chúng ta có thể chọn giá trị hợp lý để tối  
ưu quá trình truyền nhận qua LoRa.

Hiện nay các chirp LoRa do Semtech sản xuất có tần số thuộc dải băng tần  
ISM (dải băng tần miễn phí cho các ứng dụng không dây) từ 137MHz đến  
1020MHz tùy thuộc vào từng khu vực khác nhau trên thế giới:

* 430MHz: Châu Á
* 780MHz: Trung Quốc
* 433MHz hoặc 868MHz: Châu Âu
* 915MHz: Mỹ

Ở dải ISM tần số cao 868 và 915 MHz có 3 lựa chọn cho độ rộng dải băng  
thông sử dụng là 125, 250 hoặc 500 kHz, trong khi ở dải ISM tần số thấp hơn là  
7.8, 10.4, 15.6, 20.8, 31.2, 62.5kHz. Mỗi gói tin truyền nhận có thể từ 2-255 byte  
với mỗi gói tin và tốc độ truyền dữ liệu có thể đạt từ 0.018 – 37.5 kbps.

Nếu coi thời gian truyền một *Symbol* mang thông tin là Ts thì dựa và băng  
thông và hệ số trải phổ ta tính được:

Do đó tỉ lệ *Symbol* được truyền trong một giây, ký hiệu (Rs):

Cuối cùng tốc độ bit theo công thức:

Ví dụ: với băng thông là BW = 125 kHz, hệ số trải phổ SF = 7, và tỉ lệ mã  
code = 4/5, sẽ cho ta tốc độ truyền bit là Rb = 5.5 kbps.

## Kỹ thuật đo RTK trong hệ thống GNSS

### Tổng quan về hệ thống GNSS

Trong xã hội hiện nay, sóng vệ tinh gần như bao phủ mọi mặt của đời sống. Dễ dàng hình dung nhất chính là tivi và điện thoại. Một chiếc điện thoại có sóng vệ tinh có thể xác định chính xác tọa độ của người đang sử dụng nó. Tọa độ này sẽ được xác định nhờ vào hệ thống GNSS. Vậy GNSS là gì?

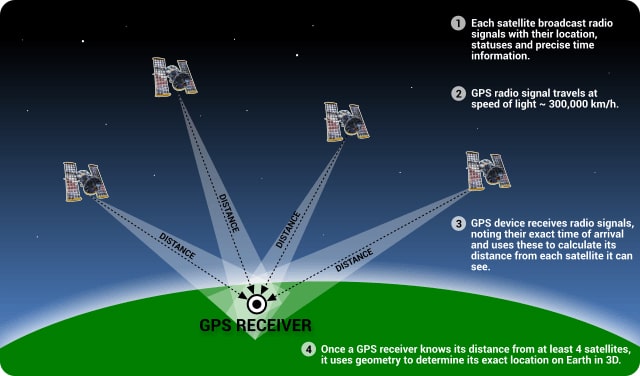
GNSS là tên viếttắt của Global Navigation Satellite System – hệ thống vệ tinh định vị toàn cầu. Hiểu một cách đơn giản thì nó gồm tập hợp tất cả các vệ tinh nhân tạo trên quỹ đạo ngoài Trái Đất. Các vệ tinh này sẽ di chuyển liên tục xung quanh địa cầu. Chúng sẽ xác định vị trí của các đối tượng trên mặt đất và định vị chính xác theo tọa độ.

GNSS hoạt động 24/24 và bất chấp điều kiện thời tiết nào. Nó cũng hoạt động mọi nơi trên Trái Đất và chỉ cần một điểm, một vị trí trên Trái Đất có thể xác định được khoảng cách đến 3 vệ tinh GNSS thì chúng ta sẽ tính được tọa độ của vị trí đó.

Cấu tạo của một hệ thống GNSS:

* Phần không gian: Là các vệ tinh bay trên quỹ đạo ngoài Trái Đất. Các vệ tinh này hoạt động bằng năng lượng Mặt Trời và có tuổi thọ khoảng 10 năm.
* Phần điều khiển: Gồm các trạm quan sát trung tâm và trạm con, có nhiệm vụ phân tích dữ liệu từ vệ tinh.
* Phần người sử dụng: Thiết bị thu vệ tinh là khu vực có phủ sóng vệ tinh. Để có thể sử dụng được các dữ liệu từ GNSS, người dùng cần 1 máy thu GNSS và ăng ten tương ứng (ví dụ truyền hình cần có đầu thu, ăng ten).

Nguyên lý hoạt động của hệ thống GNSS: Nguyên lý hoạt động của hệ thống GNSS khá phức tạp về mặt kỹ thuật, nhưng lại rất dễ dàng giải thích ở mức tổng quát, dễ hiểu hơn:



Đầu tiên, các vệ tinh được phóng lên quỹ đạo, bay vòng quanh trái đất với thời gian bay là 11 giờ, 58 phút và 2 giây. Các vệ tinh này được vận hành, theo dõi bởi các trạm điều khiển mặt đất. Các trạm điều khiển này sẽ gửi tín hiệu về thời gian, vị trí một cách chính xác cho vệ tinh. Các vệ tinh này nhận tín hiệu từ các trạm điều khiển mặt đất và sẽ biết được chính xác vị trí của chúng. Khi đã biết được vị trí chính xác của mình, các vệ tinh sẽ gửi các tín hiệu xuống trái đất cho các máy thu GNSS. Máy thu GNSS sẽ nhận tín hiệu, tính toán được chính xác vị trí của mình khi nhận được tín hiệu của ít nhất 3 vệ tinh.

# THIẾT KẾ HỆ THỐNG GPS RTK

## Thiết kế phần cứng hệ thống

A red circuit board with a white square and black square and a gold square

Description automatically generated with medium confidence

# PHÁT TRIỂN HỆ THỐNG GPS RTK

## Sơ đồ kiến trúc hệ thống và chức năng

A diagram of a system diagram

Description automatically generated

Hình 6‑3 Sơ đồ khối mô tả hệ thống

Sơ đồ khối mô tả một cách khái quát về mối liên kết giữa các thành phần của hệ thống, chúng được tách ra làm 3 phần riêng biệt:

* Khối Gateway: nhận dữ liệu nhiệt độ từ các Sensor Node và đẩy dữ liệu đó lên Cloud Server thông qua giao thức MQTT.
* Khối Sensor Node: giám sát và thu thập dữ liệu nhiệt độ và gửi dữ liệu đó về Gateway để đánh giá thông qua công nghệ truyền thông không dây LoRa.
* Khối Cloud Server: quản lý và hiển thị dữ liệu theo giao diện thống nhất.

Nhóm chúng em sẽ đi vào demo sơ đồ triển khai chi tiết từng khối phía dưới và phân tích rõ chức năng từng khối trong hệ thống này.

## Sơ đồ triển khai chi tiết từng khối

### Sơ đồ triển khai chi tiết thiết bị Rover

#### Khối nguồn

Với yêu cầu về khối nguồn của dự án, nhóm em sử dụng nguồn từ Pin Lithium và đồng thời thiết kế mạch sạc để có thể sạc trực tiếp, điện áp đầu vào để sạc cho Pin Lithium là 5V. Nguồn sạc cho Pin được cung cấp bởi IC TP4056, đây là loại IC chuyên dụng để sạc cho 1 cell Pin Lithium, dòng sạc tối đa là 1A. Đồng thời IC TP4056 có thể tự động ngừng sạc khi dòng sạc nhỏ hơn 100mA. Để bảo vệ Pin khi điện áp Pin bị sụt quá mức cho phép làm giảm tuổi thọ của Pin, nhóm em sử dụng IC DW01A và IC FS8205A. Bản thân IC FS8205A là 2 mosfet kênh P có cổng Drain nối với nhau, khi điện áp Pin sụt xuống còn 3V, chân OD của IC FS8205A nối với cổng Gate mosfet sẽ xuất ra logic 0 để ngắt Pin khỏi tải, nhờ đó Pin sẽ được bảo vệ.

Trong sơ đồ hình 6 – 1, = 0.4và = 100nF được dùng để lọc nhiễu điện áp đầu vào. Theo datasheet thì chân PROG của TP4056 dùng để chọn dòng sạc cho pin theo công thức:

trong đó V nên với dòng sạc 1A thì = 1.2k.

Hai chân CHRG và STDBY có vai trò cho biết trạng thái sạc của pin. Khi ngừng sạc thì chân STDBY có mức logic 0, chân CHRG float còn trong quá trình sạc thì ngược lại. Vậy nên nhóm em sử dụng LED L1 có 2 màu để hiển thị trạng thái sạc, điện trở và dùng để hạn dòng cho LED. Đầu ra của IC TP4056 sẽ được lọc qua tụ = 10uF và đến cực dương của Pin.

A diagram of a circuit

Description automatically generated

Hình 6‑4 Sơ đồ khối sạc pin

Với điện áp Pin từ 3.7 – 4.2V, nhóm em sử dụng IC LDO RT9013 để hạ áp xuống 3.3V để cấp nguồn cho các khối khác. IC RT9013 cho dòng cung cấp tối đa là 0.5A khi tối thiểu 250mV. Tụ và dùng để lọc nhiễu, chân EN của IC RT9013 sẽ được kéo trở cao để luôn cho phép hoạt động. Ở Sensor Node sẽ có 2 IC RT9013 để có được dòng 1A đảm bảo cung cấp nguồn ổn định.

A diagram of a circuit

Description automatically generated

Hình 6‑5 Sơ đồ nguyên lý khối nguồn LDO RT9013

#### Khối xử lý trung tâm

Điện áp IC RT9013 sẽ được lọc qua các tụ bypass theo yêu cầu trong datasheet của STM32 trước khi cấp nguồn.

Diagram, schematic

Description automatically generated

Hình 6‑6 Sơ đồ nguyên lý tụ bypass cho STM32

Như đã nói trong phần 6.3, ở Sensor Node nhóm em sử dụng vi điều khiển là STM32F103C8T6 có tần số thạch anh 72MHz, bộ nhớ Flash 64KB, SRAM 20KB. Để vi điều khiển có thể nạp bằng mạch nạp ST – Link, chân BOOT0 sẽ được kéo xuống GND qua trở 100k để chọn chế độ Main flash memory. Chân SENSOR\_DATA là chân để giao tiếp với cảm biến DS18B20. Các chân DSIN, CS, CLK để giao tiếp với IC MAX7219 để điều khiển 4 LED 7 thanh.

Diagram, schematic

Description automatically generated

Hình 6‑7 Sơ đồ nguyên lý vi điều khiển STM32

Mặc dù bản thân MCU STM32F103C8T6 đã được hãng trang bị bộ tạo dao động nội HSI lên đến 8MHz tuy nhiên bộ dao động này có độ chính xác không cao, HSI tốc độ chỉ có 8MHz không đáp ứng được yêu cầu hệ thống. Thạch anh HSE 8MHz được mắc vào 2 chân OSC\_IN và OSC\_OUT qua bộ PLL có thể nhân tần số hoạt động của chip lên tối đa 72Mhz. Các tụ C18 và C20 được thêm vào để giúp thạch anh hoạt động ổn định và chính xác.

Diagram

Description automatically generated with medium confidence

Hình 6‑8 Sơ đồ nguyên lý khối thạch anh ngoài cho vi điều khiển STM32

Do điện áp của Pin có thể lên đến 4.2V vượt quá dải điện áp cho phép từ 0 – 3.3V của STM32 nên cần được phân áp trước khi đến chân ADC của STM32. Ở đây em chọn 2 điện trở là k và 22k, khi 4.2V thì V nằm trong dải điện áp cho phép.

Diagram, schematic

Description automatically generated

Hình 6‑9 Khối giám sát điện áp pin LiPo

Module SX1278 sẽ được cấp nguồn 3.3V được lọc qua tụ . Vi điều khiển sử dụng truyền thông SPI để giao tiếp với module với Baud Rate là 8Mbit/s, 2 chân LORA\_NSS và LORA\_IO0 được sử dụng để Reset và gửi tín hiệu ngắt đến vi điều khiển.

Diagram

Description automatically generated with medium confidence

Hình 6‑10 Sơ đồ nguyên lý module Ra-02 LoRa

Ngoài ra nhóm em thiết kế thêm khối Debug để in ra dữ liệu cần thiết, thuận tiện khi lập trình và kiểm tra lỗi. USB Micro dùng để chuyển tiếp dữ liệu tới máy tính đồng thời cấp nguồn sạc cho Pin. Hai chân D\_N và D\_P sẽ được nối với chân UD+ và UD- ở IC CH340 , IC này sẽ chuyển đổi dữ liệu sang UART để giao tiếp với vi điều khiển.

Diagram

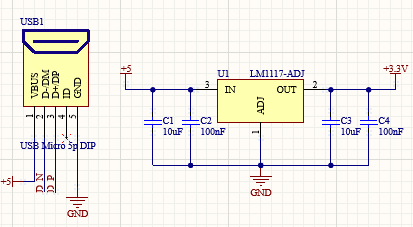
Description automatically generated

Hình 6‑11 Sơ đồ nguyên lý khối debug

### Sơ đồ triển khai chi tiết thiết bị Base

#### Khối nguồn

Gateway được cung cấp nguồn đầu vào 5V được lọc nhiễu qua tụ và , để hạ áp xuống 3.3V cấp nguồn cho khối xử lý trung tâm thì nhóm em sẽ sử dụng IC LDO LM1117 có khả năng cung cấp dòng điện lên đến 1A đủ đáp ứng cho các khối còn lại.



Hình 6‑14 Sơ đồ nguyên lý khối nguồn LDO

#### Khối xử lý trung tâm

Ở Gateway, nhóm em sử dụng vi điều khiển ESP32. Đây là dòng vi điều khiển hiệu năng rất tốt, tần số hoạt động lên tới 240MHz. Đồng thời hãng đã trang bị cho nó bộ nhớ Flash 4M, SRAM 520KB thích hợp để sử dụng hệ điều hành FreeRTOS. Vi điều khiển được cấp nguồn 3.3V được lọc nhiễu qua 2 tụ và theo yêu cầu trong datasheet. Hai chân LCD\_SDA và LCD\_SCK được dùng để giao tiếp với màn hình LCD 1602. Để Gateway đảm bảo kết nối với WiFi nhất thì có thể sử dụng antenna rời, không sử dụng antenna on – chip.

Diagram, schematic

Description automatically generated

Hình 6‑15 Sơ đồ nguyên lý vi điều khiển ESP32

Như đã đề cập ở yêu cầu 8, Gateway sẽ có 3 nút nhấn chức năng để thuận tiện thao tác vật lý, đồng thời có 2 LED để chỉ thị trạng thái hoạt động.

A picture containing table

Description automatically generated

Hình 6‑16 Sơ đồ nguyên lý các nút nhấn và LED

Để phục vụ quá trình nạp firmware và debug, một khối chuyển đổi UART sang USB và một cổng MicroUSB được sử dụng giống như ở Sensor Node.

Diagram

Description automatically generated

Hình 6‑17 Sơ đồ nguyên lý khối debug và nạp code

Khối phụ trợ cho khối nạp firmware, có nhiệm vụ chính là đưa module ESP32 vào chế độ boot bằng cách lần lượt kéo chân EN và GPIO0 xuống GND. Hai nút bấm để reset module và để kéo GPIO0 xuống GND trong trường hợp khối nạp firmware hoạt động không đúng mong đợi.

Diagram, schematic

Description automatically generated

Hình 6‑18 Sơ đồ nguyên lý khối hỗ trợ nạp

Module SX1278 vẫn sẽ được sử dụng để truyền thông với Sensor Node, các chân được sử dụng giống với thiết kế ở Sensor Node.

Diagram

Description automatically generated with medium confidence

Hình 6‑19 Sơ đồ nguyên lý module Ra-02 LoRa

## Công suất tiêu thụ của hệ thống

## Thiết kế phần cứng hệ thống

# THIẾT KẾ GIAO TIẾP GIỮA THIẾT BỊ BASE – ROVER

## Giới thiệu về platform STM32CubeIDE và VSCode

## Cấu hình thiết bị Rover

### ADC

#### Cơ bản về ADC

<https://tapit.vn/chuc-nang-adc-su-dung-vi-dieu-khien-stm32f103c8t6/>

Trong các ứng dụng vi điều khiển – hệ thống nhúng, bộ chuyển đổi tương tự-số (ADC) là 1 thành phần rất quan trọng để có thể chuyển đổi các dữ liệu dạng analog từ môi trường (nhiệt độ, độ ẩm, độ sáng, …) sang dạng digital để vi điều khiển có thể xử lý được.

STM32F103C8 có tích hợp sẵn các bộ chuyển đổi ADC với độ phân giải 12bit. Có 12 kênh cho phép đo tín hiệu từ 10 nguồn bên ngoài và 2 nguồn nội bên trong.

Trong bộ chuyển đổi ADC, có 2 thuật ngữ mà chúng ta cần chú ý đến, đó là độ phân giải (resolution) và thời gian lấy mẫu (sampling time).

* Độ phân giải (resolution): dùng để chỉ số bit cần thiết để chứa hết các mức giá trị số (digital) sau quá trình chuyển đổi ở ngõ ra. Dễ dàng nhận thấy với một bộ chuyển đổi có độ phân giải càng thấp, quá trình chuyển đổi sẽ cho ra kết quả là một điện áp càng biến dạng ở ngõ ra so với ngõ vào và ngược lại. Bộ chuyển đổi ADC của STM32F103 có độ phân giải mặc định là 12 bit, tức là có thể chuyển đổi ra 212 = 4096 giá trị ở ngõ ra số.
* Thời gian lấy mẫu (sampling time): khái niệm được dùng để chỉ thời gian giữa 2 lần số hóa của bộ chuyển đổi.

A diagram of a computer network

Description automatically generated

Vi điều khiển STM32F103 được tích hợp bộ chuyển đổi ADC 12bit loại SAR. Bộ chuyển đổi này cho phép lựa chọn 1 trong 19 kênh đầu vào, trong đó có 16 kênh đầu vào ngoại (là các chân vi điều khiển) và 3 kênh đầu vào nội là điện áp nguồn cung cấp cho vi điều khiển, cảm biến nhiệt độ được tích hợp sẵn của vi điều khiển và một kênh đo điện áp chân V\_BAT. Các kênh đầu vào này có thể chia làm 2 nhóm chuyển đổi: regular (thông thường) hoặc injected (chèn vào khi 1 ADC khác đang chuyển đổi).

#### Triển khai phần cứng

#### Cấu hình ở CubeMX:

* Data Alignments: Bộ ADC của STM32F103 có độ phân giải là 12bit mà ta sẽ phải cần lưu trữ vào một thanh ghi 32 bit, do đó sẽ còn thừa 20 bit. Chúng ta sẽ cấu hình việc căn lề cho 12 bits này nằm bên phải hay bên trái trong thanh ghi 32 bits đó.
* Scan Conversion Mode: được sử dụng để “quét” qua lần lượt các kênh ADC trong quá trình đọc dữ liệu.

Nếu sử dụng chế độ đơn kênh, chế độ này sẽ không có tác dụng.

* Continous Conversion Mode: quyết định cho bộ ADC của chúng ta có sử dụng chế độ chuyển đổi liên tục hay không.

Nếu chúng ta không *enable* mode này, sau mỗi lần chuyển đổi, ta sẽ phải gọi lại lệnh đọc giá trị ADC để bắt đầu quá trình chuyển đổi mới.

* Sampling Time: chọn thời gian lấy mẫu trong quá trình số hóa.

Tùy vào ứng dụng mà chúng ta có thể chọn thời gian lấy mẫu cho phù hợp. Thời gian lấy mẫu càng ngắn, việc tái thiết tín hiệu càng chính xác nhưng năng lượng tiêu tốn sẽ càng cao.

* Rank: là thứ tự lấy mẫu của các kênh ADC (đối với chuyển đổi ADC đa kênh), cao nhất là 1 lớn nhất là 10.

Bộ ADC sẽ scan từ kênh có rank là 1 đến n với n là số kênh các bạn cần chuyển đổi.

* NVIC Settings:

**Các chế độ chuyển đổi:**

* Chế độ đơn kênh, sử dụng interrupt để báo hiệu quá trình chuyển đổi hoàn tất.

<https://tapit.vn/chuc-nang-adc-su-dung-vi-dieu-khien-stm32f103c8t6/>

<https://tapit.vn/hieu-va-doc-gia-tri-adc-stm32f411-bo-chuyen-doi-tin-hieu-tuong-tu/> (xem ở phần single channel, single conversion mode)

* Chế độ nhiều kênh, chuyển đổi 1 lần - Multi-channel (scan), single conversion: Ở chế độ này, bộ ADC sẽ lần lượt chuyển đổi một chuỗi các kênh đầu vào đã được cấu hình từ trước, tối đa có thể chuyển đổi được 16 kênh. kết quả chuyển đổi của kênh cuối cùng sẽ được lưu vào thanh ghi ADC\_DR, cờ báo kết thúc quá trình chuyển được được bật lên, yêu cầu ngắt sẽ được sinh ra nếu người dùng cấu hình cho phép ngắt.

Chế độ này cần được kết hợp với tính năng DMA để có thể nhận đầy đủ kết quả chuyển đổi của các kênh.

Thứ tự chuyển đổi của các kênh có thể được người dùng sắp xếp và thời gian lấy mẫu của các kênh có thể được cấu hình độc lập.

* Chế độ đơn kênh, chuyển đổi liên tục – Single-channel, continous conversion mode: Ở chế độ này kênh được chọn sẽ được chuyển đổi liên tục mà không cần sự can thiệp của CPU trong mỗi lần chuyển đổi. Chức năng DMA với mode circular nên được sử dụng trong chế độ này để giảm tải cho CPU.
* Chế độ đa kênh, chuyển đổi liên tục – Multi-channel (scan), continous conversion mode: Ở chế độ này các kênh được chọn sẽ được chuyển đổi liên tục mà không cần sự can thiệp của CPU trong mỗi lần chuyển đổi. Chức năng DMA với mode circular kết hợp với việc tăng bộ nhớ tự động nên được sử dụng trong chế độ này để giảm tải cho CPU.

<https://khuenguyencreator.com/lap-trinh-stm32-adc-nhieu-kenh-su-dung-dma/>

A diagram of a program

Description automatically generated

**Tính năng DMA (Direction Memory Access):** hỗ trợ ngoại vi có thể chuyển dữ liệu trực tiếp đến bộ nhớ mà không cần sự can thiệp của CPU.

<https://www.youtube.com/watch?v=MYWiCRR2oHc&ab_channel=HuyB%C3%B9iV%C4%83n>

<https://www.youtube.com/watch?v=MMqaF6qgrP0&ab_channel=WonyeobChrisPark>

**Cấu hình DMA:**

* DMA Settings:
* Mục DMA Request thêm kênh ADC1 các mục còn lại sẽ tự động được cài đặt theo.
* Mode: Circular với mode này dữ liệu sẽ liên tục được gửi qua lại giữa ADC và Vùng nhớ sử dụng DMA
* Data Width: Half Word 16 bit

**Ứng dụng trong project:**

<http://www.efton.sk/STM32/STM32_VREF.pdf>

<https://stackoverflow.com/questions/58328342/calibrating-stm32-adc-vrefint>

<https://stackoverflow.com/questions/57752689/how-to-use-the-vrefint-in-stm32f103-bluepill>

Từ những khó khăn được nêu ra trong tài liệu STM32\_VREF, trong dự án này, để đo được điện áp của pin, em sử dụng chế độ đa kênh, chuyển đổi liên tục kết hợp tính năng DMA đối với bộ chuyển đổi ADC. Cụ thể:

* Chế độ đa kênh: kênh ADC1\_IN1 và kênh Vrefint.
* Cấu hình ADC: chế độ đa kênh, chuyển đổi liên tục.

A screenshot of a computer

Description automatically generated

|  |  |  |
| --- | --- | --- |
| ADC1 Clock | 12MHz | 12MHz |
| Channel | ADC1\_IN1 | ADC1\_Vrefint |
| Thời gian lấy mẫu | 239.5 cycles ≈ 20 µs | 239.5 cycles ≈ 20 µs |

* Cấu hình DMA: giống như cấu hình trên, tuy nhiên ở mục “Data Width” chọn thông số “Word” tương ứng 32 bits.

### LoRa

#### Cơ bản về giao thức SPI

<https://khuenguyencreator.com/lap-trinh-stm32-voi-giao-thuc-spi/#Che_do_hoat_dong_cua_giao_thuc_SPI>

<https://tapit.vn/chuan-giao-tiep-spi-tren-stm32f4/>

<http://www.hocavr.com/2018/06/bai-7-giao-tiep-spi.html>

A screenshot of a computer

Description automatically generated

#### Triển khai phần cứng

Module SX1278 sẽ được cấp nguồn 3.3V được lọc qua tụ . Vi điều khiển sử dụng truyền thông SPI để giao tiếp với module với Baud Rate là 8 Mbit/s, 2 chân LORA\_RST và LORA\_IO0 được sử dụng để reset và gửi tín hiệu ngắt đến vi điều khiển.

Diagram

Description automatically generated with medium confidence

**LORA\_RST:** chân này dùng cho manual reset.

A screenshot of a computer

Description automatically generated

LORA\_DI0:

A green and white table with black text

Description automatically generated

A screenshot of a computer

Description automatically generated

#### Cấu hình ở CubeMX

SPI2 Mode and Configuration:

* Mode: Full-Duplex Master
* Hardware NSS Signal: Disable (chân CS em sẽ sử dụng chân GPIO khác để dễ dàng điều khiển nếu nhiều slave)
* Chân PA8: cấu hình cho chân CS của SPI2

PA15: nối với chân LORA\_RST

PB12: nối với chân LORA\_DI0

### Cấu hình GPS

## Cấu hình thiết bị Base

### Cấu hình GPS

## Thiết kế phần mềm giao tiếp

# ĐÁNH GIÁ, KIỂM THỬ