**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

****

**BÁO CÁO NGHIÊN CỨU KHOA HỌC ĐỀ TÀI**

***“*Thiết kế, chế tạo module phần cứng thu nhận và truyền dữ liệu cảm biến*”***

**Mã số ………….**

**HỢP ĐỒNG NGHIÊN CỨU KHOA HỌC CÔNG NGHỆ**

**Số: ……………**

|  |  |
| --- | --- |
| **Người thực hiện :** |  |
| **Chủ nhiệm đề tài:** |  |
|  |  |
|  |  |

Mục Lục

[CHƯƠNG I: GIỚI THIỆU CHUNG 3](#_Toc3413673)

[1. Giới thiệu chung 3](#_Toc3413674)

[CHƯƠNG 2: THIẾT KẾ SƠ ĐỒ NGUYÊN LÝ 5](#_Toc3413675)

[2.1. Sơ đồ khối 5](#_Toc3413676)

[2.2. Thiết kế nguyên lý khối nguồn cho mạch 6](#_Toc3413677)

[2.2.1. Khối hạ áp DC-DC sử dụng IC MP1584 6](#_Toc3413678)

[2.2.2. Khối ổn áp nguồn 3.3V sử dụng IC HT7333 8](#_Toc3413679)

[2.3. Thiết kế nguyên lý khối điều khiển trung tâm 10](#_Toc3413680)

[2.4. Thiết kế nguyên lý khối mạch cho module Lora Rhf76-052 15](#_Toc3413681)

[2.4.1. Giao tiếp UART 15](#_Toc3413682)

[2.4.2. Giới thiệu về RHF76-052: 17](#_Toc3413683)

[2.4.3. Thông số kỹ thuật của mô-đun: 20](#_Toc3413684)

[2.5. Tập lệnh AT cho mô-đun Rhf76-052: 20](#_Toc3413685)

[2.6. Thiết kế sơ đồ nguyên lý khối module Lo-ra: 24](#_Toc3413686)

[2.7. Thiết kế nguyên lý khối cảm biến nhiệt độ độ ẩm 25](#_Toc3413687)

[2.7.1. Giới thiệu cảm biến SHT75 25](#_Toc3413688)

[2.7.2. Block Diagram: 26](#_Toc3413689)

[2.7.3. Thông số kỹ thuật hiệu suất của cảm biến 26](#_Toc3413690)

[2.7.4. Thông số kỹ thuật giao diện kết nối 27](#_Toc3413691)

[2.7.5. Quá trình giao tiếp với MCU 28](#_Toc3413692)

[2.7.6. Thiết kế sơ đồ nguyên lý kết nối STM32 và SHT75 31](#_Toc3413693)

[2.8. Thiết kế nguyên lý khối đọc ADC 32](#_Toc3413694)

[2.9. Thiết kế nguyên lý khối EEPROM 32](#_Toc3413695)

[2.9.1. Giới thiệu về I2C 32](#_Toc3413696)

[2.9.2. Giới thiệu IC M24M01 38](#_Toc3413697)

[2.9.3. Thiết kế nguyên lý STM32 kết nối với M24M01 41](#_Toc3413698)

[CHƯƠNG 3: THIẾT KẾ LAYOUT 42](#_Toc3413699)

[3.1. Giới thiệu 42](#_Toc3413700)

[3.2. Thiết kế layout khối nguồn 43](#_Toc3413701)

[3.2.1. Thiết kế layout khối hạ áp DC-DC sử dụng IC MP1584 43](#_Toc3413702)

[3.2.2. Thiết kế layout khối ổn áp 3.3V 45](#_Toc3413703)

[3.3. Thiết kế layout khối điều khiển trung tâm 48](#_Toc3413704)

[3.4. Thiết kế layout khối EEPROM 49](#_Toc3413705)

[3.5. Thiết kế layout khối module Lora 50](#_Toc3413706)

[3.6. Thiết kế layout cho các khối cảm biến 52](#_Toc3413707)

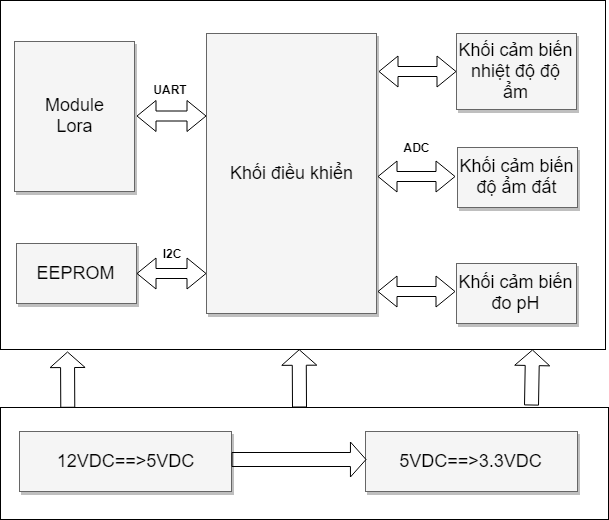
[3.7. Hình ảnh 3D của bo mạch 53](#_Toc3413708)

[CHƯƠNG 4: KẾT LUẬN 55](#_Toc3413709)

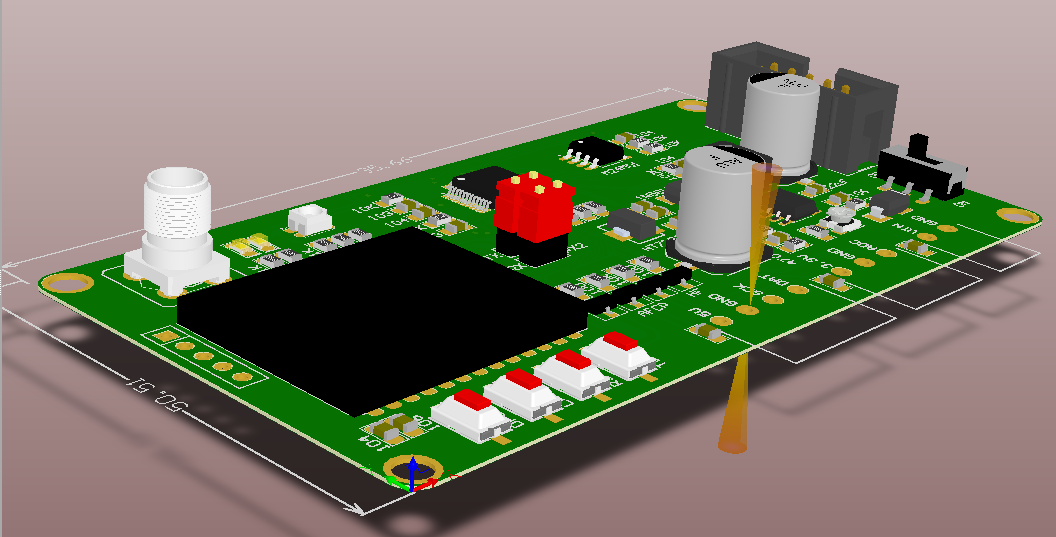
**CHƯƠNG I: GIỚI THIỆU CHUNG**

## 1. Giới thiệu chung

Để có thể thu thập các dữ liệu nhiệt độ độ ẩm và các thông số môi trường từ các cảm biến đặt trong lắp đặt trong hệ thống nông trại cần phải thiết kế một bo mạch để đọc dữ liệu từ các cảm biến đó lưu trữ vào bộ nhớ và truyền dữ liệu thông qua Lora. Bo mạch này sẽ sử dụng chíp STM32L152 giao tiếp với module Lora Rhf76-052 và đọc dữ liệu từ các cảm biến kết nối với nó. Dưới đây là sơ đồ khối và hình ảnh 3D của nó:

****

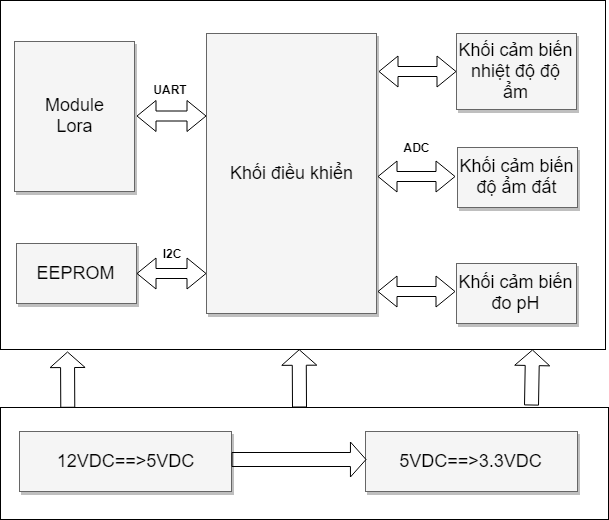
Hình 1. : Sơ đồ khối

****

Hình 1. : Hình ảnh 3D

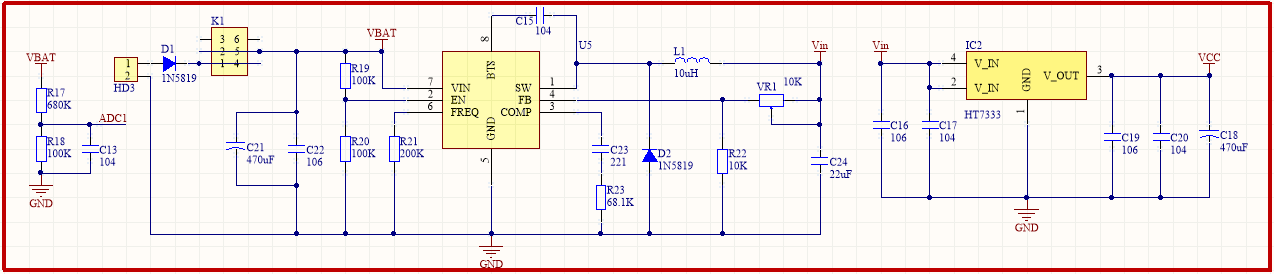
# CHƯƠNG 2: THIẾT KẾ SƠ ĐỒ NGUYÊN LÝ

## 2.1. Sơ đồ khối

****

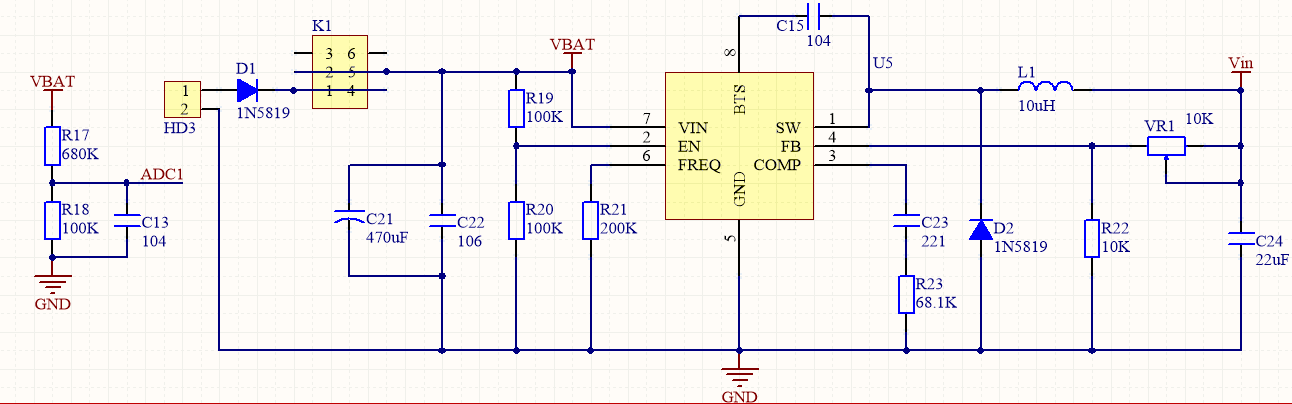
Hình 2. : Sơ đồ khối

## 2.2. Thiết kế nguyên lý khối nguồn cho mạch



Hình 2. : Khối nguồn cho toàn mạch

### 2.2.1. Khối hạ áp DC-DC sử dụng IC MP1584



Hình 2. : Khối hạ áp DC-DC sử dụng IC MP1584

Sử dụng cầu phân áp để đọc được điện áp còn lại của pin . Ngõ ra 12VDC từ ac quy 12VDC sẽ được đưa qua bộ chuyển đổi DC – DC để hạ áp xuống 5VDC. Trong thiết kế sử dụng IC chuyên dụng MP1584.

Các chân chức năng của MP1584:

|  |  |  |
| --- | --- | --- |
| Chân số | Tên | Mô tả |
| 1 | SW | Điểm node kết nối giữa 2 N-FET : Low side và High side, yêu cầu kéo xuống đất qua một đi-ôt Schottky. Điôt cần phần đặt gần chân SW để giảm sự chuyển mạch đột ngột. |
| 2 | EN | chân cho phép đầu vào.Kéo xuống dưới mức quy đinh chíp sẽ tắt. Kéo lên trên mức quy định để cho phép chíp. |
| 3 | COMP | Đầu ra của bộ khuếch đại lỗi. |
| 4 | FB | Đầu vào của bộ khuếch đại lỗi.Điện áp đầu ra Vout sẽ được đặt bởi một cầu chia áp nối với chân này.(VFB tham chiếu =0.8V) |
| 5 | GND | Đất. Nó nên được kết nối càng gần càng tốt với tụ điện đầu ra để rút ngắn đường dẫn chuyển đổi dòng điện cao |
| 6 | FREQ | Chân đặt tần số chuyển đổi đầu vào. Kết nối một điện trở từ chân này xuống đất để đặt tần số chuyển đồi. |
| 7 | VIN | Đầu vào nguồn cung cấp.Một tụ cách ly cần phải đặt gần chân này để giảm chuyển mạch đột ngột. |
| 8 | BST | Bootstrap. Đây là nguồn cung cấp năng lượng tích cực cho MOSFET bên trong.Nối một tụ bypass giữa chân này và chân SW. |

Bảng1: Mô tả các chân chức năng của MP1584

Tính năng của MP1584:

* Hoạt động ở dải điện áp đầu vào rộng 4.5V-28V.
* Điện áp đầu ra 0.8V đến 25V.
* Có thể thay đổi tần số chyển mạch từ 100 kHz đến 1.5MHz.
* Hoạt động ở nhiệt độ -20 °C đến 125 °C**.**
* Dòng điện cung cấp tối đa 3A.

Tính toán giá trị các linh kiện trong mạch để điện áp đầu ra của IC MP1584 khoảng 5V:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| L(µH) | C13(µF) | R19(kΩ) | C20(pF) | C19(µF) | R15(kΩ) | R16(kΩ) | R18(kΩ) | C10(kΩ) |
| 15-22 | 22 | 100 | 150 | 10 | 100 | 27 | 100 | 100 |

Bảng : Giá trị các linh kiện để đầu ra 5V

Theo sơ đồ mạch điện điện áp chân FB của IC:

\*.

* .

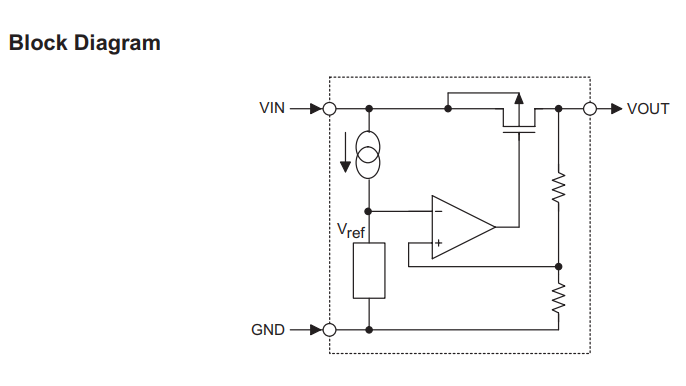
|  |  |  |
| --- | --- | --- |
| **Tham số** | **Ký hiệu** | **Giá trị** |
| Điện áp ngõ vào | VDC | 12 VDC |
| Điện áp ngõ ra | Vin | 0.8 -5 VDC |
| Dòng điện rõ ra |  | 3A ( max ) |
| Tụ ngõ ra | C13 | 22uF/25V |
| Cuộn dây ngõ ra | L23 | 15 uH/6.8A |

Bảng 3: Tham số và giá của các linh kiện

### 2.2.2. Khối ổn áp nguồn 3.3V sử dụng IC HT7333

Điện áp khoảng 5VD ở đầu ra của IC MP1854 sẽ được đưa qua khối ổn áp 3.3V để cấp nguồn cho vi điều khiển STM32 và Lora hoạt động.

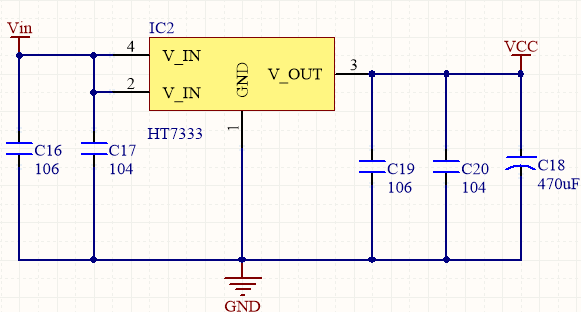
IC HT7333 là ic ổn áp thuộc họ HT73xx cho điện áp đầu ra ổn định ở 3.3V.Dòng điện đầu ra tối đa 250 mA. Điện áp đầu vào tối đa 12V.



Hình 2. : Block Diagram của HT7333

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Symbol | Parameter | Test | | Min. | Typ. | Max. | Unit |
|  | Conditions |
|  | Output Voltage | 4.3V | =40mA | 3.201 | 3.3 | 3.399 | V |
| (MAX) | Maximum Output Current | 4.3V | 2.97V | 250 |  |  | mA |
| \* | Load Regulation | 4.3V | 1mA IOUT 80mA |  | 45 | 98 | mV |
| \*\* | Dropout Voltage |  | =40mA |  | 90 |  | mV |
|  | Quiescent Current | 4.3V | No load |  | 4 | 8 | µA |
|  | Line Regulation |  | =40mA 4.3V VIN 12V |  | 0.2 | 0.3 | %/V |
|  | Input Voltage |  |  |  |  | 12 | V |
|  | Temperature Coefficient | 4.3V | =40mA 40°C <<85°C |  | ± 0 .7 |  | mV/°C |

Bảng 4: Khoảng làm việc HT7333

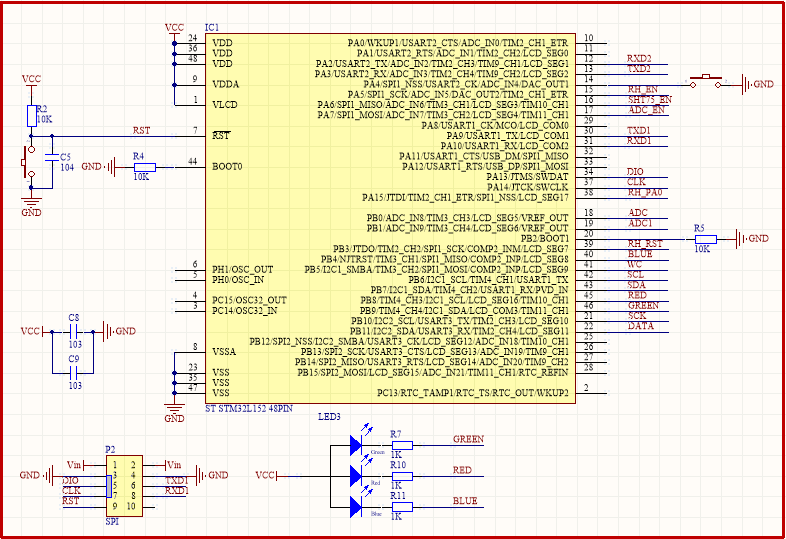


Hình 5- Sơ đồ nguyên lý khối nguồn ổn áp 3.3V

|  |  |  |
| --- | --- | --- |
| **Tham số** | **Ký hiệu** | **Giá trị** |
| Điện áp ngõ vào | Vin | ~5 VDC |
| Điện áp ngõ ra | VCC | 3.3 VDC |
| Dòng điện rõ ra |  | 250mA ( max ) |
| Tụ ngõ vào | C16,C17 | ,104,106 |
| Tụ ngõ ra | C18,C19,C20 | 470uF/25V , 104,106 |

Bảng 5: Thông số kỹ thuật nguồn ổn áp HT7333

## 2.3. Thiết kế nguyên lý khối điều khiển trung tâm



Hình 2. : Sơ đồ nguyên lý khối vi điều khiển smt32

Khối điều khiển trung tâm sử dụng vi điều khiển STM32L152. Đây là MCU ARM 32-bit Cortex-M3.Chíp tiết kiệm năng lượng có 128KB Flash, 16KB SRAM, 4KB EEPROM, hỗ trợ LCD, USB, ADC, DAC.

Một số tính năng của STM32L152:

* Sử dụng nguồn năng lượng thấp:
  + Nguồn cung cấp: 1.65 – 3.6 V
  + Hoạt động ở dải nhiệt độ -40 \*C đến 105\*C
  + 0.3 µA Standby mode (3 wakeup pins)
  + 0.9 µA Standby mode + RTC
  + 0.57 µA Stop mode (16 wakeup lines)
  + 1.2 µA Stop mode + RTC
  + 9 µA Low-power run mode
  + 214 µA/MHz Run mode
  + 10 nA ultra-low I/O leakage
  + < 8 µs wakeup time
* Core: ARM® Cortex®-M3 32-bit CPU:
  + Từ 32 kHz lên đến 32 MHz max.
  + 1.25 DMIPS/MHz.
  + Memory protection unit.
* Nguồn clock:
  + Tần số thạch anh từ 1 đến 24 MHz
  + Thạch anh 32 kHz cho RTC(thời gian thực)
  + PLL cho clock của CPU và USB (48 MHz)
* Có thể sử dụng nhiều nhất 83 chân IO, tất cả đều được ánh xạ trên 16 vector ngắt ngoài.
* Bộ nhớ:
  + 128 Kbytes Flash
  + 16 Kbytes RAM
  + 4 Kbytes EEPROM
  + 80 byte thanh ghi dữ trự
* Rất nhiều ngoại vi Analog:
  + 24 kênh ADC 12 bit
  + 2 kênh DAC 12 bit với bộ đệm đầu ra
* 8x giao diện kết nối ngoại vi:
  + 1x USB 2.0 (internal 48 MHz PLL)
  + 3x USARTs
  + 2x SPIs 16 Mbit/s
  + 2x I2Cs (SMBus/PMBus)
* 10x Timers:

Để tiết kiệm năng lượng và chi phí thiết kế sử chíp STM32L152(loại đóng gói 48 PIN). Chíp hỗ trợ 7 chế độ hoạt động để sử dụng năng lượng một cách phù hợp nhất.

* **Sleep mode:**

ở chế độ này chỉ CPU ngừng hoạt động.Tất cả các ngoại vi khác vẫn hoạt động bình thường và có thể đánh thức CPU khi có một ngắt hoặc sự kiện.

* Low power run mode:

Chế độ này đạt được với bộ dao động RC đa năng bên trong (MSI) được đặt ở mức xung nhịp tối thiểu (65 kHz), thực hiện từ bộ nhớ SRAM hoặc Flash và bộ điều chỉnh bên trong ở chế độ này để giảm thiểu dòng hoạt động của bộ điều chỉnh. Trong chế độ này, tần số xung nhịp và số lượng thiết bị ngoại vi được bật đều bị giới hạn.

* Low power sleep mode:

Chế độ này đạt được bằng cách vào Sleep mode với bộ điều chỉnh điện áp bên trong ở chế độ Nguồn điện thấp để giảm thiểu dòng điện hoạt động của bộ điều chỉnh.Trong chế độ này cả tần số lẫn số lượng ngoại vi cho phép đều bị giới hạn. Khi đánh thức được kích hoạt bởi một sự kiện hoặc ngắt, hệ thống sẽ trở lại run mode với bộ điều chỉnh bật.

* Stop mode với RTC:

Chế độ này đạt mức tiêu thụ năng lượng thấp nhất trong khi vẫn giữ RAM và thanh ghi nội dung và đồng hồ thời gian thực. Tất cả các đồng hồ trong miền VCORE bị dừng, bộ dao động tinh thể PLL, MSI RC, HSI RC và HSE bị vô hiệu hóa. LSE hoặc LSI vẫn đang chạy. Bộ điều chỉnh điện áp ở chế độ năng lượng thấp.

* Stop mode mà không có RTC:

Chế độ này đạt mức tiêu thụ năng lượng thấp nhất trong khi vẫn giữ RAM và thanh ghi nội dung. Tất cả các đồng hồ trong miền VCORE bị dừng, bộ dao động tinh thể PLL, MSI RC, HSI RC và HSE bị vô hiệu hóa. LSE hoặc LSI vẫn đang chạy. Bộ điều chỉnh điện áp ở chế độ năng lượng thấp.

* Standby mode với RTC:

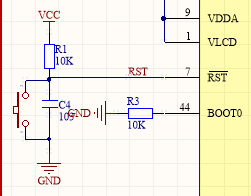
Chế độ này được sử dụng để đạt được mức tiêu thụ điện năng thấp nhất và đồng hồ thời gian thực. Bộ điều chỉnh điện áp bên trong được tắt để toàn bộ miền VCORE bị tắt. Các bộ dao động tinh thể PLL, MSI RC, HSI RC và HSE cũng bị tắt. LSE hoặc LSI vẫn đang chạy. Sau khi vào chế độ Chờ, RAM và nội dung thanh ghi bị mất ngoại trừ các thanh ghi trong mạch chờ (wakeup logic, IWDG, RTC, LSI, LSE Crystal 32K osc, RCC\_CSR).

* Standby mode mà không có RTC:

Chế độ này được sử dụng để đạt được mức tiêu thụ điện năng thấp nhất. Bộ điều chỉnh điện áp bên trong được tắt để toàn bộ miền VCORE bị tắt. Các bộ dao động tinh thể PLL, MSI, RC, HSI và LSI RC, HSE và LSE cũng bị tắt. Sau khi vào chế độ Chờ, RAM và nội dung thanh ghi bị mất ngoại trừ các thanh ghi trong mạch chờ (logic đánh thức, IWDG, RTC, LSI, LSE Crystal 32K osc, RCC\_CSR).

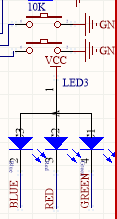
Thiết bị thoát khỏi chế độ này trong 60 Lời mời khi xảy ra cài đặt lại bên ngoài (chân NRST) hoặc cạnh tăng trên một trong ba chân WKUP.

Trong thiết kế sử dụng chíp này để giao tiếp với lora RHF76052 bằng giao tiếp UART,giao tiếp với cảm biến SHT75, giao tiếp với EEPROM M24M01 (I2C),Cảm biến đo pH qua ADC.



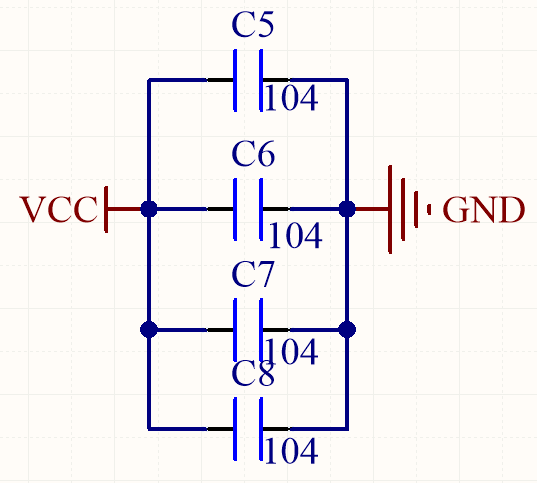
Hình 2. : Khối reset cho mạch STM32

Một mạch Reset được nối với chân RST của chíp.Bình thường chân này được kéo lên mức cao qua điện trở 10K, khi nhấn nút reset nó sẽ kéo chân này xuống mức thấp để reset chíp.



Hình 2. : Led RGB và nút nhấn

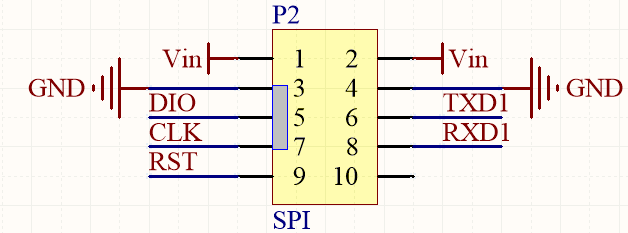
Trên mạch có tích hợp 1 số nút nhấn và một đèn led RGB để thuận tiện cho việc debug code cho chip sau này..



Hình 2. : Các tụ lọc nguồn cho STM32

Một số tụ lọc để gần nguồn của chíp để nhằm mục đích chống nhiễu cho chíp.

Khi layuot cần đặt càng gần chân nguồn của vi điều khiển càng tốt.



Hình 2. : Header để nạp code

Kết nối với 1 header 5x2 để thuận tiện cho quá trình nạp code cho vi điều khiển và debug cho UART1.

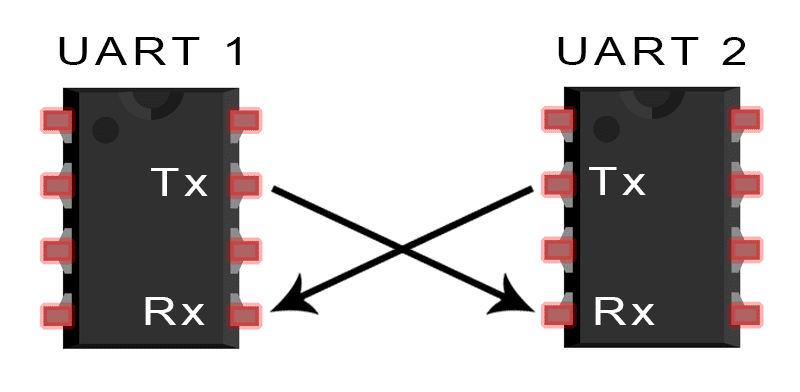
## 2.4. Thiết kế nguyên lý khối mạch cho module Lora Rhf76-052

### 2.4.1. Giao tiếp UART

Thuật ngữ UART trong tiếng anh là viết tắt của cụm từ: Universal Asynchronous serial Reveiver and Transmitter, nghĩa là bộ truyền nhận nối tiếp không đồng bộ.

Trong giao tiếp UART cả 2 thiết bị kết nối trực tiếp với nhau. UART truyền chuyển đổi dữ liệu song song từ thiết bị điều khiển như CPU ​​thành dạng nối tiếp, truyền nó nối tiếp đến UART nhận, sau đó chuyển đổi dữ liệu nối tiếp thành dữ liệu song song cho thiết bị nhận.Do đó chỉ cần đường dây kết nối để truyền và nhận dữ liệu giữa 2 thiết bị có hỗ trợ UART. Dữ liệu được truyền từ chân TX của thiết bị truyền sang chân RX của thiết bị nhận. UART truyền dữ liệu không đồng bộ,điều đó có nghĩa là không cần một chân clock để đồng bộ hóa giữa 2 thiết bị. Thay vì tín hiệu clock, UART truyền thêm các bit start và stop cho gói dữ liệu được truyền. Các bit này xác định điểm bắt đầu và kết thúc của gói dữ liệu để UART nhận biết khi nào bắt đầu đọc các bit.

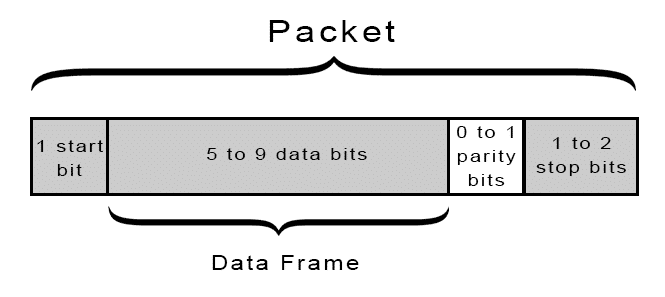
Khi thiết bị nhận phát hiện bit start, nó bắt đầu đọc các bit đến ở một tần số cụ thể được gọi là tốc độ baund. Tốc độ truyền là thước đo tốc độ truyền dữ liệu, được biểu thị bằng bit trên giây (bps). Cả hai thiết bị UART phải hoạt động ở cùng tốc độ baund. Cả hai UART cũng phải được cấu hình để truyền và nhận cùng một cấu trúc gói dữ liệu.



Hình 2. : 2 thiết bị giao tiếp UART

**Start bit:** Đường truyền dữ liệu UART thường được giữ ở mức điện áp cao khi nó không truyền dữ liệu. Để bắt đầu truyền dữ liệu, UART truyền sẽ kéo đường truyền từ cao xuống thấp trong một chu kỳ xung nhịp. Khi UART nhận phát hiện sự chuyển đổi điện áp cao sang thấp, nó bắt đầu đọc các bit trong khung dữ liệu ở tần số của tốc độ truyền.

**Data Frame (khung truyền):** do truyền thông nối tiếp mà nhất là nối tiếp không đồng bộ rất dễ mất hoặc sai lệch dữ liệu, quá trình truyền thông theo kiểu này phải tuân theo một số quy cách nhất định. Bên cạnh tốc độ baud, khung truyền là một yếu tốc quan trọng tạo nên sự thành công khi truyền và nhận. Khung truyền bao gồm các quy định về số bit trong mỗi lần truyền, các bit “báo” như bit Start và bit Stop, các bit kiểm tra như Parity, ngoài ra số lượng các bit trong một data  cũng được quy định bởi khung truyền. Nó có thể dài 5 bit đến 8 bit nếu sử dụng một bit chẵn lẻ. Nếu không có bit chẵn lẻ nào được sử dụng, khung dữ liệu có thể dài 9 bit. Trong hầu hết các trường hợp, dữ liệu được gửi với bit ít thấp nhất trước tiên.



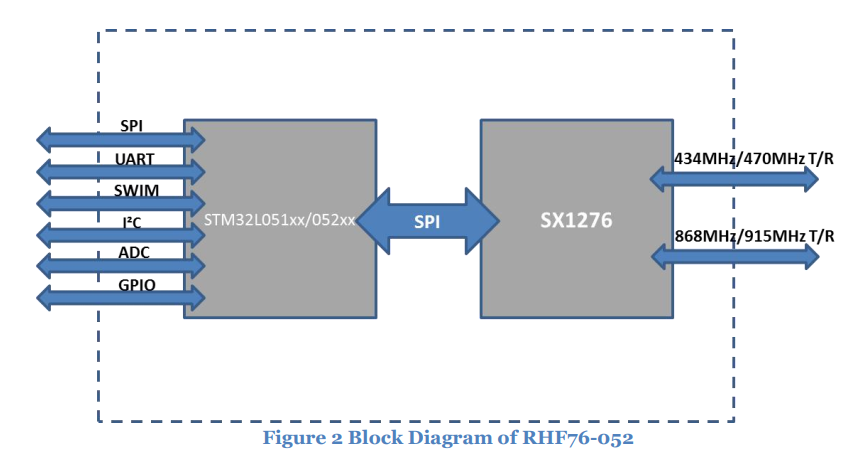
Hình 2. : Hình 12- Khung truyền dữ liệu trong giao tiếp UART

**Parity bit:** Parity là bit dùng kiểm tra dữ liệu truyền đúng không (một cách tương đối). Có 2 loại parity là parity chẵn (even parity) và parity lẻ (odd parity). Bởi vì Các bit có thể được thay đổi bằng bức xạ điện từ, tốc độ truyền không khớp hoặc truyền dữ liệu đường dài. Sau khi UART nhận đọc khung dữ liệu, nó đếm số bit có giá trị là 1 và kiểm tra xem tổng số là số chẵn hay số lẻ. Nếu bit chẵn lẻ là 0 (chẵn lẻ), thì số bít 1 trong khung dữ liệu sẽ có tổng số chẵn. Nếu bit chẵn lẻ là 1 (chẵn lẻ), thì số bít 1 trong khung dữ liệu sẽ tổng thành một số lẻ. Khi bit chẵn lẻ khớp với dữ liệu, UART biết rằng việc truyền không có lỗi. Nhưng nếu bit chẵn lẻ là 0 và tổng là số lẻ; hoặc bit chẵn lẻ là 1 và tổng số chẵn, UART biết rằng các bit trong khung dữ liệu đã thay đổi.

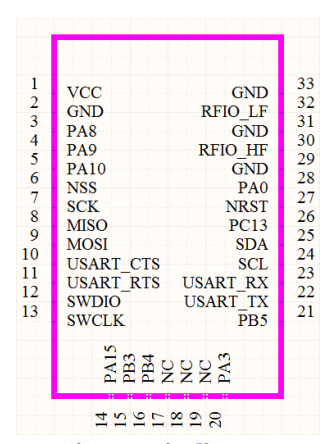
**Stop bits:** Để báo hiệu sự kết thúc của gói dữ liệu, UART gửi sẽ điều khiển đường truyền dữ liệu từ mức thấp đến mức cao trong ít nhất hai bit.

### 2.4.2. Giới thiệu về RHF76-052:

Mô-đun RisingHF ™ LoRaWAN ™ RHF76-052 được nhúng với LoRaWAN stack. Người sử dụng có thể sử dụng một host MCU để điều khiển mô-đun với tập lệnh AT đơn giản qua giao tiếp UART.Trong thiết kế host MCU là STM32L152.



Hình 2. : Block diagram của Rhf-76052



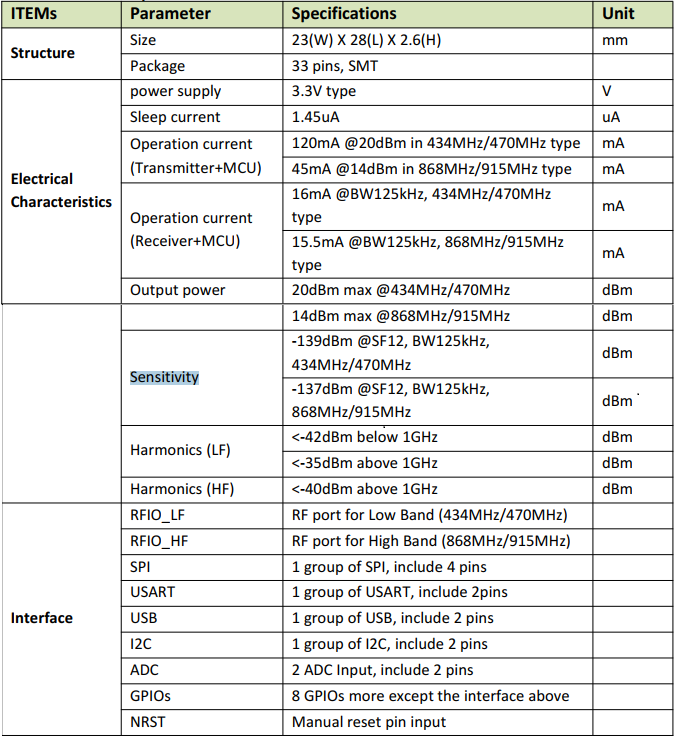
Hình 2. : Sơ đồ chân của module rhf76-052

**Phạm vi hoạt động:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ký hiệu | Mô tả | Min | Max | Unit |
| VCC | Nguồn cung cấp | 1.8 | 3.6 | V |
| TOP | Nhiệt độ | -40 | 85 | °**C** |
| POP | RF input level |  | +10 | dBm |

Bảng : Phạm vi hoạt động của module Rhf76-05

### 2.4.3. Thông số kỹ thuật của mô-đun:

****

Hình 2. : Thông số kỹ thuật của mô-đun UART

## 2.5. Tập lệnh AT cho mô-đun Rhf76-052:

**Quy ước:**

* Tất cả các lệnh đều có phản hồi
* Độ dài lệnh không bao giờ vượt quá 528 ký tư
* Một lệnh AT có giá trị phải két thúc với ‘\n’,’r\n’
* Nếu tính năng timeout được bật, kết thúc ‘\n’ sẽ không bắt buộc
* <LF> có nghĩa là ký tự dòng mới. <CR> có nghĩa là vận chuyển trở lại
* UART được được cấu hình “9600,8,n,1” (8 bits data, no parity,1 bít stop)

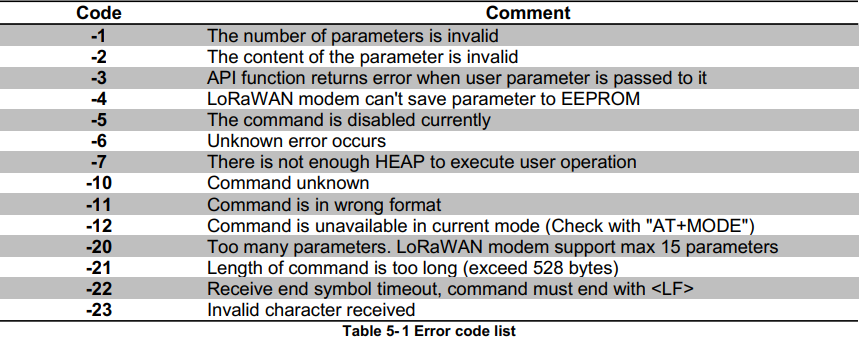
**Ký hiệu**

* **= ==>** cài đặt giá trị cho lệnh
* **?**  ==> Truy vấn
* **:** ==> bắt đầu một danh sách tham số đầu vào
* **+ ==>**  Tiền tố của lệnh
* **, ==>** Dấu phân cách các tham số
* **Space ==>** Ký tự trống,có thể sự dụng để format lệnh

**Định dạng(Format):**

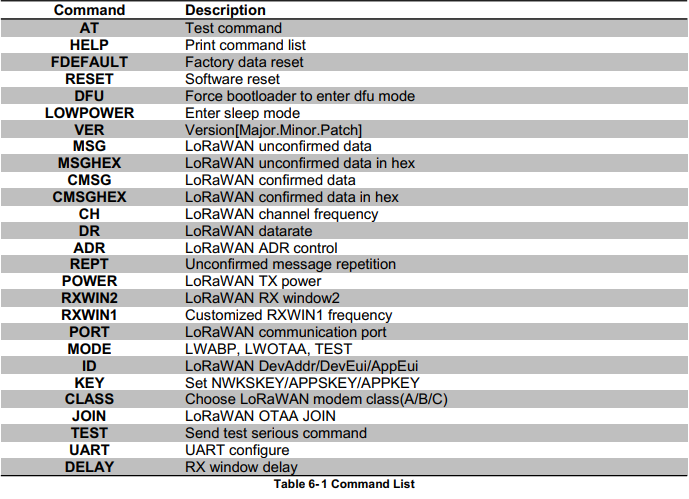
* **Truy vấn:** sử dụng lệnh truy vấn để kiểm tra cấu hình modem LoRaWAN,cũng như cấu hình kênh,trạng thái ADR, nguồn TX,v..v..
  + **AT+COMMAND**
  + **AT+COMMAND?**
  + **AT+COMMAND=?**
* **Cấu hình/Điều khiển:** sử dụng lệnh cấu hình/ điều khiển để cài đặt cấu hình mới hoặc điều khiển
  + **AT+COMMAND=DATA**
* **Return:** Trả về dữ liệu với định dạng như sau "+CMD: RETURN DATA"
  + **+COMMAND: “RETURN DATA”**

**Mã lỗi(Error):**

****

Hình 2. : Mã lỗi và mô tả các lỗi của RHf76-052

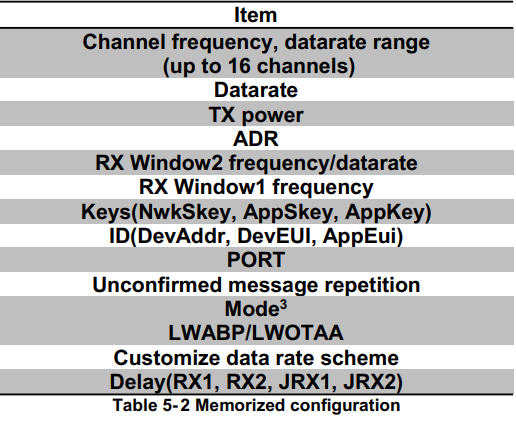
**Mô tả các lệnh AT:**

****

Hình 2. : Mã lệnh AT cho mo-đun Rhf76-052

**EEPROM**

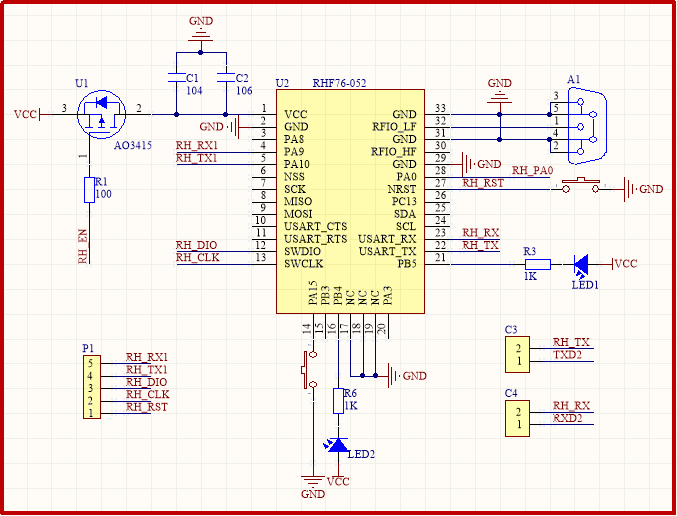
Các mục bên dưới sẽ được đồng bộ hóa với EEPROM của modem LoRaWAN sau khi thay đổi thành công, điều này làm cho chế độ LoRaWAN được ghi nhớ, người dùng không cần phải cấu hình lại tham số sau khi tắt và mở lại nguồn, Modem LoRaWAN giúp giữ nó.



Hình 2. : Các mục được lưu trong EEPROM của mo-đun

### 

## 2.6. Thiết kế sơ đồ nguyên lý khối module Lo-ra:

****

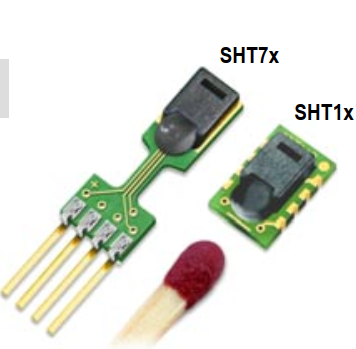
Hình 2. : Sơ đồ nguyên lý khối module Lora Rhf76-052

* Sử dụng anten thu ở dải tần số thấp 434MHz/470MHz nối vào chân số 32
* Chân 22 và chân 23 của modem được kết nối với các chân TX,RX của MCU STM32(host) thông qua 2 jump.
* Chân 4 và chân 5 của modem được sử dụng để upgare FW dựa trên kết nối UART.
* Chân 14 được sử dụng để kích hoạt chế độ DFU cho upgrage FW.
* Chân 16 được sử dụng để hiển thị trạng thái của xử lý LoraWAN.
* Chân 21 được dành riêng để kết nối LED
* Ngoài ra sử dụng 1 mosfet để làm chân ENABLE cho module.

## 2.7. Thiết kế nguyên lý khối cảm biến nhiệt độ độ ẩm

Để đo được độ ẩm và nhiệt độ của môi trường trong khu vực sử dụng IC cảm biến nhiệt độ độ ẩm chuyên dụng SHT75 họ SHT7x(SHT71,SHT75).

### 2.7.1. Giới thiệu cảm biến SHT75

****

Hình 2. : Hình ảnh cảm biến SHT7x,SHT1x

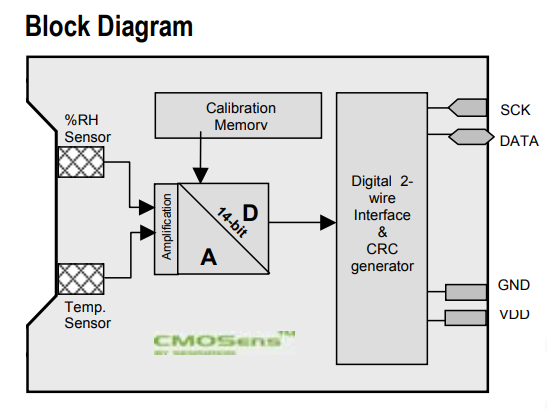
Đây là loại cảm biến chuyên dụng được sản xuất với công nghệ (CMOSens®) để đảm bảo độ tin cậy cao và thời gian hoạt động lâu dài,ngoài ra nó còn có các tính năng nổi bật như sau:

* có đầu ra số(digital output)
* ổn định lâu dài
* không yêu cầu thêm các thành phần chuyển đổi bên ngoài
* tiêu thụ năng lượng cực thấp
* Thiết kế nhỏ gọn(4 pin)
* Tự động ngắt nguồn
* Dải đo nhiệt độ -40 đến 120 độ C
* Độ ẩm: 0 đến 100 %

Cảm biến này được ứng dụng cho:

* Trong ô tô, máy tự động
* HVAC
* Trạm thời tiết
* Công nghệ tự động
* Y khoa
* Máy tạo độ ẩm

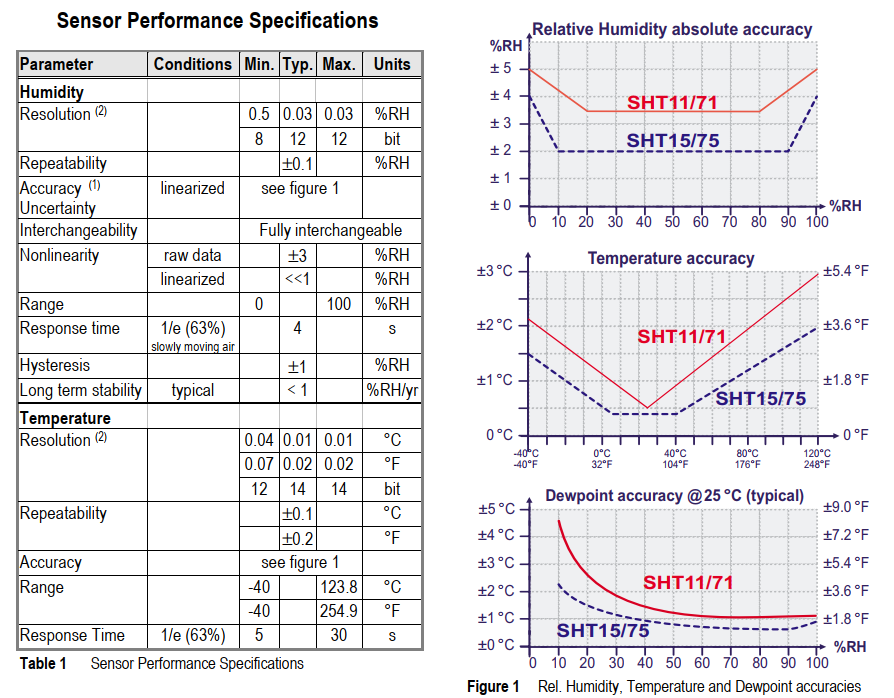
**2.7.2. Block Diagram:**



Hình 2. : Block Diagram của SHT75

Ở bên trong SHT75 có 2 cảm biến nhiệt độ độ ẩm riêng biệt sau khi đọc được dữ liệu từ môi trường chúng được đưa qua bộ chuyền đôi A/D 14 bít để biến thành tín hiệu số sau đó đưa qua giao diện 2 dây số để giao tiếp với vi điều khiển.

**2.7.3. Thông số kỹ thuật hiệu suất của cảm biến**



Hình 2. : Hình 21- Thông số kỹ thuật hiệu suất cảm biến SHT75

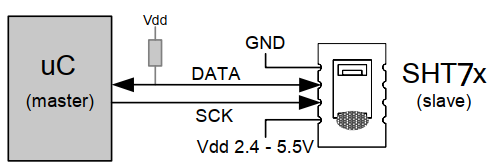
**2.7.4. Thông số kỹ thuật giao diện kết nối**

Sơ đồ chân SHT75:

|  |  |  |
| --- | --- | --- |
| Chân số | Tên | Mô tả |
| 1 | GND | Đất |
| 2 | DATA | Dữ liệu nối tiếp 2 chiều |
| 3 | SCK | Clock nối tiếp,input |
| 4 | VDD | Nguồn cung cấp 2.4-5.5 V |

Bảng : Sơ đồ chân SHT75

SHT75 giao tiếp với MCU thông qua giao tiếp 2 dây nối tiếp (2 dây 2 chiều) để tối ưu hóa:

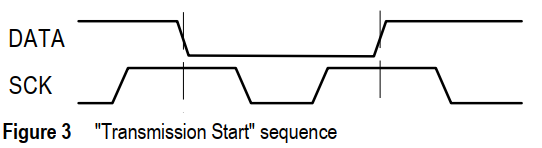


Hình 2. : Sơ đồ kết nối MCU và SHT7x

* *Serial clock input(SCK):* Chân SCK được sử dụng để đồng bộ giao tiếp giữa MCU và SHT75.
* *Serial data(DATA):* Chân DATA được sử dụng để truyền dữ liệu vào ra cho cảm biến(2 chiều). Dữ liệu thay đổi sua sườn xuống và có giá trị ở cạnh dương của chân SCK. Cần có 1 điện trở pull-up cho chân này.

**2.7.5. Quá trình giao tiếp với MCU**

Để bắt đầu quá trình truyền nhận, MCU cần phải gửi một tín hiệu bắt đầu truyền(Transmission Start) cho SHT75. Tín hiệu này được tạo ra bằng cách kéo chân DATA xuống mức thấp trong khi chân SCk vẫn ở mức cao giữ nó ở mức thấp cho đến khi mức cao của chu kỳ tiếp theo của chân SCK.



Hình 2. : Tín hiệu bắt đầu truyền nhận

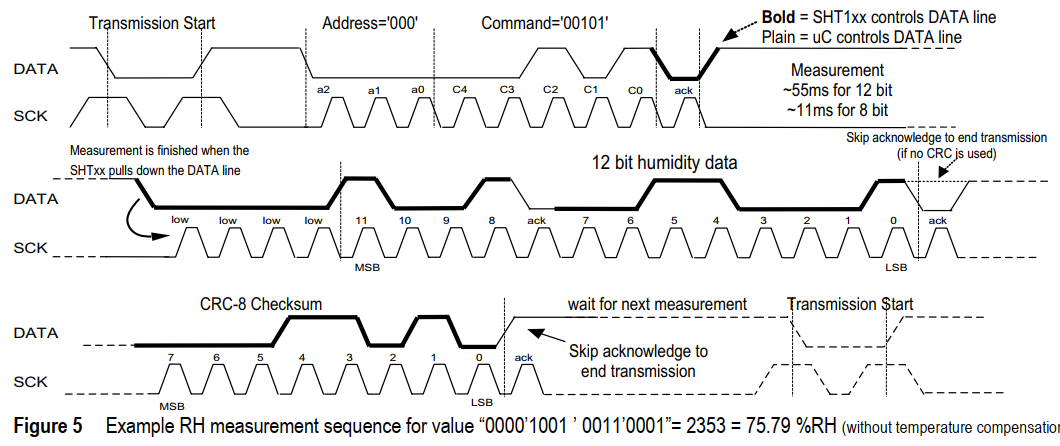
Tiếp theo MCU sẽ gửi 3 bít địa chỉ(000) và 5 bít lệnh. SHT75 cho biết việc nhận lệnh thích hợp bằng cách kéo pin DATA xuống thấp (bit ACK) sau sườn xuống SCK thứ 8.

Chân DATA sẽ lại được kéo lên mức cao sau sườn âm thứ 9 của SCK.

|  |  |
| --- | --- |
| Lệnh | Mã lệnh |
| Reserved | 0000x |
| Đo nhiệt độ | 00011 |
| Đo độ ẩm | 00101 |
| Đọc thanh ghi trạng thái | 00111 |
| Ghi vào thanh ghi trạng thái | 00110 |
| Soft reset |  |
|  |  |

Bảng : lệnh và mã lệnh cho SHT75

Tiếp theo là quá trình đọc dữ liệu từ cảm biến. Sau khi gửi lệnh đo (‘00000101, đối với RH, 00 00000011 triệt cho Nhiệt độ), MCU phải chờ cho phép đo hoàn thành. Quá trình này mất khoảng 11/55/210 ms cho phép đo 8/12/14bit. Để báo hiệu việc hoàn thành phép đo, SHT75 kéo DÂT xuống mức thấp(ack). MCU phải chờ tín hiệu sẵn sàng dữ liệu này trước khi bắt đầu bật lại SCK.



Hình 2. : Ví dụ về đo độ ẩm bằng SHT75

Quá trình truyền được đảm bảo bởi 8 bít checksum nó dùng để đảm bảo rằng bất kỳ lỗi nào trên đường truyền cũng được phát hiện và loại bỏ.

Ngoài ra sht75 còn có 1 thanh ghi trạng thái 8 bit để cấu hình 1 vài chức năng nâng cao như cấu hình độ phân giải cho bộ A/D.

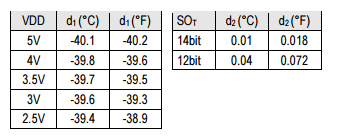
- Chuyển đổi dữ liệu 12 bit ra giá trị nhiệt độ và độ ẩm:

Với nhiệt độ Nhiệt độ được chuyển đổi theo công thức:

T = d1 + d2.SOT

T là giá trị nhiệt độ

Giá trị d1 ,d2 tham khảo bảng sau:

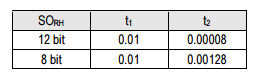


SOT là giá trị 12 bit dữ liệu nhiệt độ đọc về Với độ ẩm Độ ẩm được chuyển đổi theo công thức:

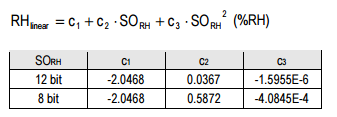
**HM = (T - 25) \* (t1+t2\*SORH) + RHliner**

T là nhiệt độ đo được tính theo độ C

Giá trị t1,t2 các bạn xem trong bảng

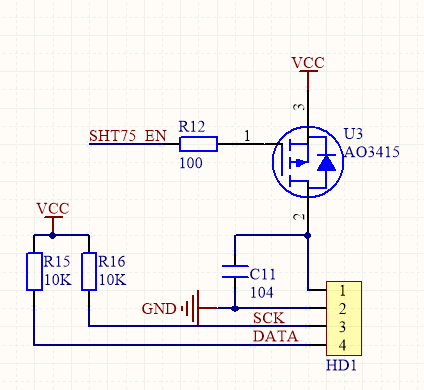


SORH là giá trị 12 bit độ ẩm đọc được. RHliner là giá trị được tính theo công thức :



Giả sử mình đọc được 12 bit độ ẩm là 0100 0011 0001 quy đổi nó ra là bằng 1073 vậy SOrh = 1073 vậy ta có RHlinear = -2.0468 + 0.0367\*1073 + 9.633.10^(-7)\*1073^2 = 38.44 Gia sử nhiệt độ đo được bằng SHT75 lúc đó là 30 độ C thì HM = (30 - 25)\*(0.01 + 0x00008\*1073) + 38.44 = 38.9 %.

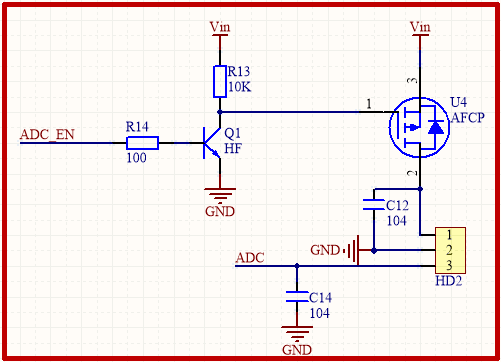
**2.7.6. Thiết kế sơ đồ nguyên lý kết nối STM32 và SHT75**



Hình 2. : Hình 25- Sơ đồ kết nối giữa STM32 và SHT75

Trong thiết kế này sử dụng 1 con mosfet AO3145 để làm chân điều khiển cho phép cảm biến hoạt động hay không(SHT75 enable). Sử dụng 2 trở 10k kéo lên cho 2 chân SCK và chân DATA. Tụ 104 để lọc nhiễu nguồn cho SHT75.

**2.8. Thiết kế nguyên lý khối đọc ADC**

****

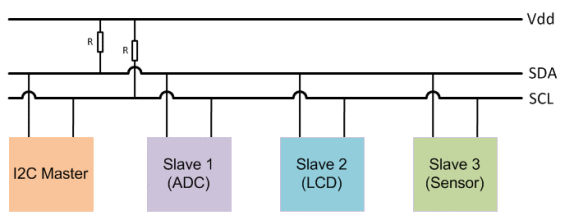
Hình 2. : Sơ đồ kết nối đọc ADC với STM32

Khối đọc ADC được thiết kế để có thể sử dụng để đọc dữ liệu từ cảm biến đo độ pH và cảm biến đo độ ẩm của đất.Khi muốn sử dụng cảm biến nào ta có thể cắm loại đó. Với cảm biến đo độ ẩm đất còn có thể kết nối bằng RS485.Thiết kế sử dụng mạch tran và mosfet để làm chân ENABLE khối đọc. Tụ lọc 104 để kéo chống nhiễu cho đường đọc ADC. Ở đây sử dụng thêm transistor Q1 để tránh mosfet Q4 hiểu nhầm mức 3.3V là mức 0, khó kích thông mosfet.

**2.9. Thiết kế nguyên lý khối EEPROM**

**2.9.1. Giới thiệu về I2C**

I2C (Inter–Integrated Circuit) là một chuẩn truyền nhận tín hiệu có sử dụng xung clock. Đây là một chuẩn kết nối ngắn sử dụng cặp dây SCL, SDA. Giao tiếp  đến các ngoại vi bằng cách gọi địa chỉ . Một Master có thể kết nối với nhiều slave, mỗi Slave  trên đường truyền có địa chỉ không giống nhau.



Hình 2. : Hình 27- Sơ đồ giao tiếp I2C

* **Tốc độ giao tiếp:**

Chuyển dữ liệu hai chiều:

* 100 Kbit/s  : Standard – Mode
* 400Kbit/s  : Fast – Mode
* 1 Mbit/s    : Fast – Mode – Plus
* 3.4 Mbit/s : High – speed Mode

Chuyển dữ liệu một chiều : 5 Mbit/s   :  Ultra – Fast – Mode

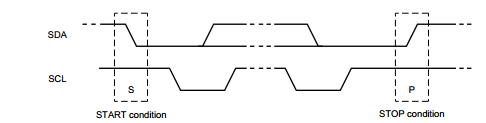
* **Cơ sở truyền nhận:**

Tín hiệu start ,Restart , stop:

Start : SCL ở mức 1 và SDA được kéo từ  mức 1 xuống mức 0

Restart : Start + delay + Start;

Stop: SCL ở mức 1 SDA được kéo từ 0 lên 1



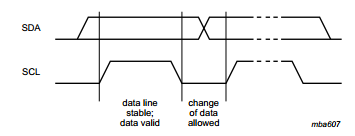
Hình 2. : Tín hiệu Start và tín hiệu Stop

1 bít dữ liệu được truyền :

Khi SCL = 0 dữ liệu trên SDA thay đổi không ảnh hưởng

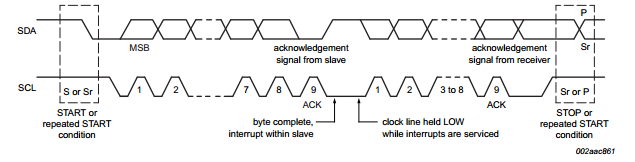
Khi SCL =1 dữ liệu 1 bit là trạng thái trên chân SDA

Kéo lại chân SCL xuống 0 thì một bit đã truyền thành công.



Hình 29-Định dạng 1 byte

Bit MSB là Bit cao được truyền trước. Khi Master truyền đủ một khung 8 bit Slave nhận được 8 bít thì Slave truyền lại cho Master một trạng thái ACK báo hiệu đã nhận thành công 8 bit.



Hình 2. : Quá trình I2C truyền nhận nhiều byte

ACK(Acknowledge : thừa nhận) và NACK (Not Acknowledge : Không thừa nhận) :

 ACK : 1 bit =0 : SCL=0, SDA=0, SCL=1,SCL=0 (SDA =0 trong một xung trên)

NACK: 1 bit =1 : SCL=0, SDA=1, SCL=1,SCL=0 (SDA =1 trong một xung trên)

Tín hiệu báo hiệu khi nhận được một byte dữ liệu . Tín hiệu trả về . Nên được đọc và xử lý nó để tối ưu hóa nhiễu trên đường truyền .

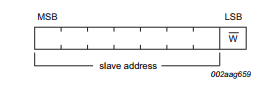
Vậy khi truyền 8 bit thì cần 9 xung clock

Địa chỉ trong I2C :

Có 2 chuẩn ; 7 bit địa chỉ và 10 bit địa chỉ . Chỉ nói đến chuẩn 7 bit địa chỉ

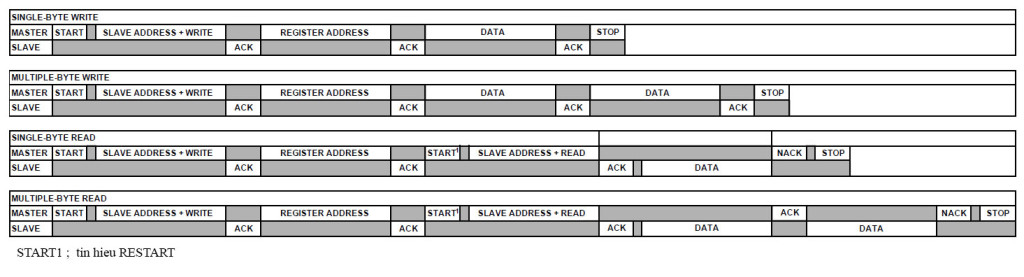
Với 7 bit địa chỉ ta có thể mã hóa vào 128 loại thiết bị . 1 master có thể kết nối tối đa 127 Slave nhưng trong I2C có 16 địa chỉ sử dụng để dự trữ nên chỉ có thể kết nối tối đa 111 slave (112 thiết bị trên đường truyền).

7 bit địa chỉ được thêm vào sau đó một bit chế độ W/R (đọc/ghi ) để tạo thành 1 byte đầu tiên được truyền đi sau mỗi lệnh start hoặc Restart.



Hình 2. : Byte đầu tiên được gửi sau lệnh start trong I2C

Một số khung truyền :

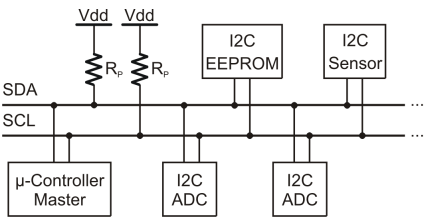


Hình 2. : Một số khung truyền trong I2C

Điện trở treo:

Điện trở treo  Rp (Pullup Resistor) là điện trở không thể bỏ trong I2C.

Việc tính toán điện trở này là cần thiết trong mỗi dự án , với mỗi tốc độ bộ I2C , nguồn Vdd , dòng cấp cho SDA , SCL Cần một điện trở khác nhau để đạt được hiệu suất cao nhất , ít nhiễu nhất (mất tín hiệu ).



Hình 2. : Điện trở kéo lên trong I2C

Chọn Rp như thế nào

Rp(min) : phù hợp với mức logic

Rp(max) : phù hợp thời gian rise time (thời gian tăng giảm 1 cạnh)

Giá trị Rp(min): để IC nhận đúng mức logic thì điện áp chân phải lớn hơn VOL(điện áp cao nhất mức LOW).

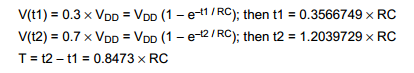
https://nguyentiensk.files.wordpress.com/2016/09/rpmin.png

Giá trị Rp(max) : I2C là Bus truyền nhận tín hiệu nên vấn đề thời gian rise time rất cần phải được quan tâm . Nhất là với các mode ở tốc độ cao .

Mạch kết nối được coi là một mạch RC vì có xuất hiện tụ kí sinh trong mạch do đó  giá trị V được tính :

https://nguyentiensk.files.wordpress.com/2016/09/v.png?w=150&h=22

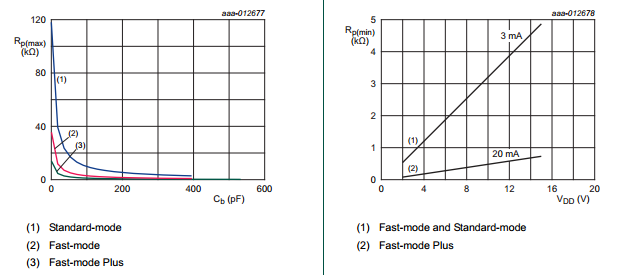
Hai giá trị VIH = 0.7 Vdd  , VIL = 0.3 Vdd  thông thường là như vậy (theo cách chế tạo IC ) do đó thời gian để chuyển trạng thái :



Ta tính ngược lại giá trị R :

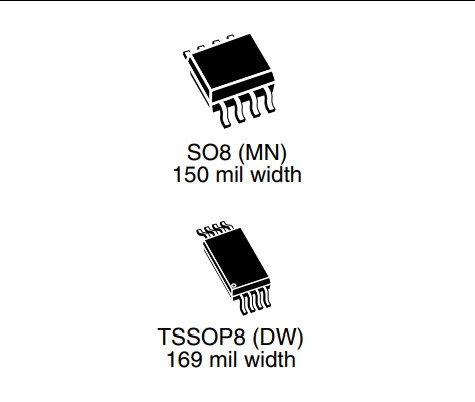
https://nguyentiensk.files.wordpress.com/2016/09/rpmax.png

Có thể chọn giá trị Rp dựa vào biểu đồ sau :



Hình 2. : Biểu đồ lựa chọn điện trở treo

**2.9.2. Giới thiệu IC M24M01**



Hình 2. : Hình ảnh của IC M24M01

Một số tính năng cơ bản:

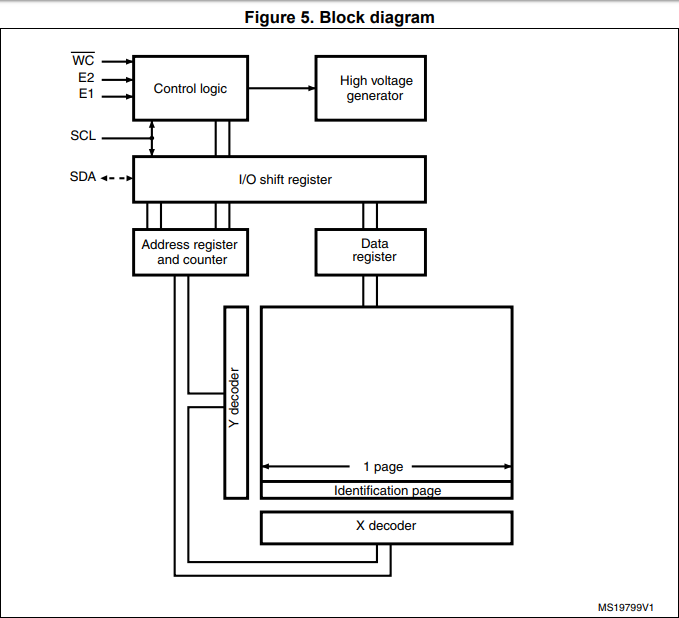
* Tương thích với tất cả các chế độ bus I2C:
  + 1 Mhz
  + 400 kHz
  + 100 kHz
* Mảng bộ nhớ:
  + 1 Mbit (128 Kbyte) of EEPROM
  + Page size: 256 byte –
  + Additional Write lockable page (M24M01-D order codes)
* Nguồn cung cấp và tốc độ cao:
  + 1 MHz clock from 1.7 V to 5.5 V
* Dải nhiệt đô hoạt động:
  + -40 °C đến +85 °C
* Lưu giữ dữ liệu hơn 200 năm
* Hơn 4 triệu chu kỳ viết

Sơ đồ chân của M24M01:

|  |  |  |
| --- | --- | --- |
| Chân tín hiệu | Chức năng | Hướng dữ liệu |
| E1,E2 | Chíp enable | Input |
| SDA | Serial data | I/O |
| SCL | Serial Clock | Input |
|  | Write Control | Input |
|  | Nguồn cung cấp | - |
|  | Đất | - |

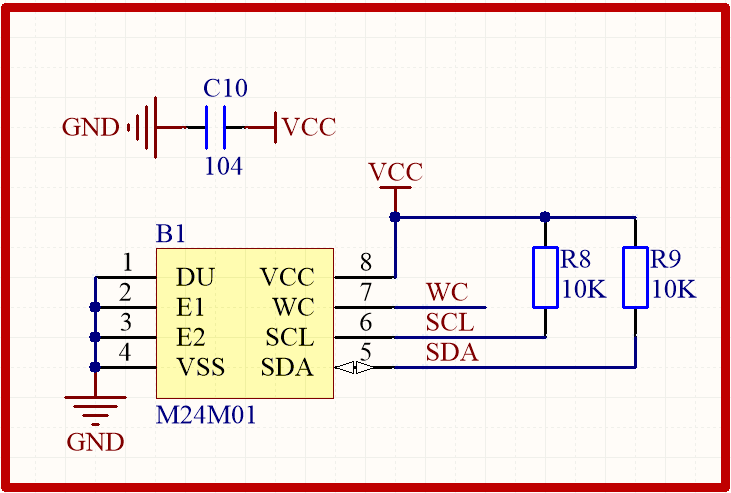
Bảng 9: Sơ đồ chân của M24M01

Tổ chức bộ nhớ như hình dưới đây:



Hình 2. : Block diagram của M24M01

**2.9.3. Thiết kế nguyên lý STM32 kết nối với M24M01**



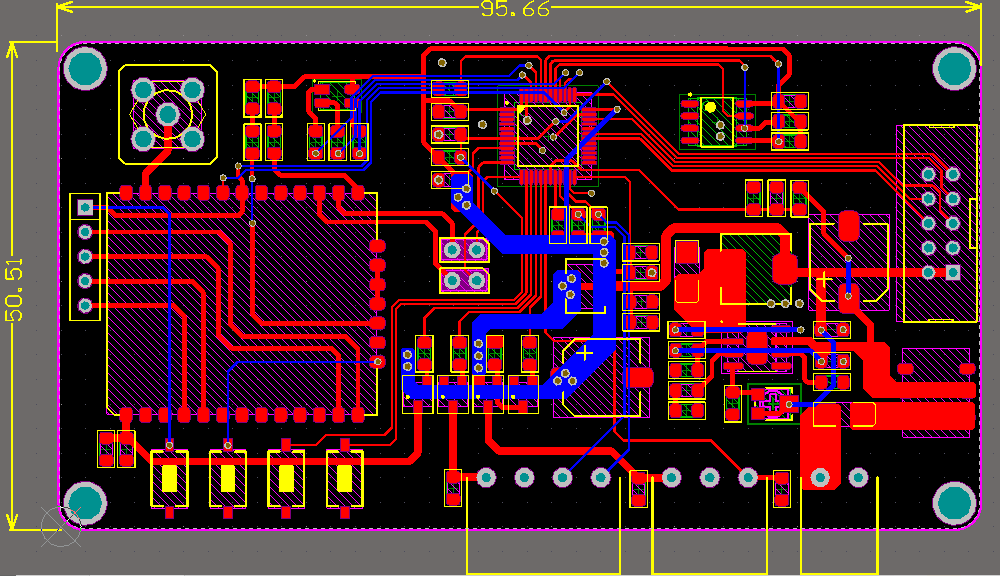
Hình 36- Sơ đồ kết nối STM32 và M24M01

2 chân SDA và SCL của IC M24M01 được kết nối tới bộ I2C 1 của STM32 có sử dụng 2 trở 10K kéo lên nguồn. Chân WC của ic được kết nối với 1 chân GPIO của STM32.

# CHƯƠNG 3: THIẾT KẾ LAYOUT

## 3.1. Giới thiệu

Thông thường PCB được cấu tạo từ các tấm đồng rất mỏng được ép với nhau, xen giữa chúng là các chất nền đóng vai trò là chất cách điện, thường là các sợi thủy tinh được bao bởi nhựa epoxy. Tùy theo mức độ gia nhiệt, độ cứng mà các chất nền này được gọi là PrePreg hoặc Core. Core là chất nền được xử lý dưới nhiệt độ và áp suất cao hơn, nên cứng và cách điện tốt hơn so với PrePreg. Trong thiết kế này để phù hợp với yêu cầu thực hiện thiết kế mạch 2 lớp với kích thước chiều ngang và dọc như hình dưới đây(đơn vị mm, đã ẩn lớp phủ đồng).



Hình 37- PCB toàn bộ bo mạch(đã ẩn toàn bộ lớp phủ đồng)

*Các thông số của vật liệu tạo nên PCB gồm:*

* Hằng số điện môi (Relative Dielectric Constant), được ký hiệu là εr hoặc Dk: Hằng số điện môi càng cao sẽ có một số ưu điểm sau:
  + Khả năng cách ly tốt hơn
  + Các tín hiệu có thể truyền với tốc độ nhanh hơn
  + Trở kháng đường truyền cao hơn với cùng một kích thước, từ đó giúp giảm độ rộng của dây dẫn, tiết kiệm diện tích trong khi layout các mạch có mật độ cao.
  + Các thành phần cảm kháng ký sinh giảm.
* Hệ số suy hao (Loss Tangent), ký hiệu tan(δ) hoặc Df: Đo lường sự suy hao của tín hiệu khi được dẫn truyền trên đường truyền. Hệ số suy hao càng thấp thì càng tốt.
* Kiểu dệt của sợi thủy tinh (Fiberglass Weave Composition): Mật độ các sợi thủy tinh càng dày thì trở kháng của các đường tín hiệu càng đồng đều, càng tốt.
* Hiệu ứng bề mặt (Skin Effect): Tần số tín hiệu càng cao, trở kháng của đường dây dẫn càng lớn, gây cản trở tín hiệu. Vì vậy phải tính toán kích thước dây dẫn phù hợp với tần số làm việc để bảo đảm trở kháng chính xác.

Như vậy, có thể đưa ra kết luận, với tín hiệu có tần số càng cao thì chất lượng PCB càng phải tốt: hằng số điện môi lớn, hệ số suy hao nhỏ, vải sợi thủy tinh trong các chất nền nên có mật độ cao, đồng đều.

**3.2. Thiết kế layout khối nguồn**

**3.2.1. Thiết kế layout khối hạ áp DC-DC sử dụng IC MP1584**

Trong các hệ thống với các vi xử lý, các nguồn DCDC Convert luôn đóng vai trò là nguồn cung cấp chính nhờ hiệu suất cao. Tuy nhiên, các nguồn này có nhược điểm là sinh ra các nhiễu chuyển mạch tần số cao trong quá trình hoạt động. Nếu layout không tốt sẽ gây ra các hệ quả như làm giảm hiệu suất, hoạt động thiếu ổn định, gây ra các vấn đề EMI. Về nguyên tắc, nên sử dụng các đường dẫn có kích thước rộng và ngắn cho các đường cấp nguồn chính.

*Quy tắc layout chung cho switching converter*

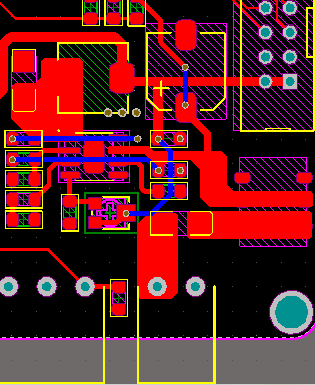
Điện áp trên một cuộn cảm được tính theo công thức

Từ đó suy ra, điện áp không phụ thuộc vào cường độ dòng điện mà phụ thuộc vào tốc độ thay đổi cường độ dòng điện. Một dòng điện nhỏ nhưng thay đổi nhanh có thể gây ra một thay đổi điện áp lớn dù giá trị cảm kháng là nhỏ. Trong DC-DC Converter, giá trị dI/dt cao trên các đường AC thường xảy ra khi thay đổi trạng thái đóng ngắt của các bộ chuyển mạch.

Do dI/dt rất khó can thiệp, vì vậy các giảm V hiệu quả nhất là giảm L. Việc đặt các phần tử linh kiện lại gần nhau sẽ giúp giảm độ dài dây dẫn, giúp giảm cảm kháng ký sinh trên dây nhiều hơn là việc mở rộng kích thước của dây dẫn. Ưu tiên cao nhất trong layout là các tụ đầu vào cần đặt gần sát DCVIN và DCGND. Tiếp theo là đặt các cuộn cảm lại gần SW và tụ đầu ra lại gần cuộn cảm để giảm tối thiểu diện tích Current Loop, giúp giảm EMI. Diện tích Current Loop nhỏ cũng có thể làm giảm nhiễu ảnh hưởng tới các đường có điện áp nhạy cảm như đường DCVOUT feedback.

Không nên sử dụng các lỗ xuyên pcb (via) để nối các đường quan trọng do chúng có trở kháng ký sinh cao.

Nên đặt các tụ đầu vào có kích thước lớn ở trên cùng một mặt của PCB với IC và sử dụng các vias để nối các tụ tới mặt phẳng Power Plane và Ground Plane, nên sử dụng nhiều via thay vì một via.

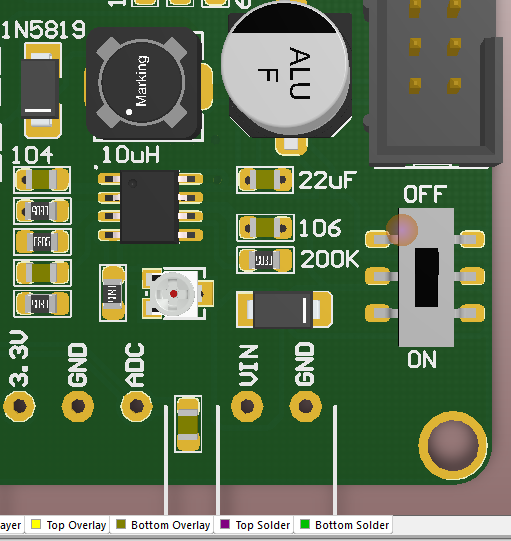


Hình 38- PCB khu vực nguồn DC-DC Converter

Dây DC đầu vào và dây nguồn pin VBAT chịu dòng lớn nên được phủ lớp coper nhằm tăng khả năng chịu dòng. Tương tự đường Vin cũng được đi dây với độ rộng 40 mil. Các dây khác với độ rộng 20 mil.

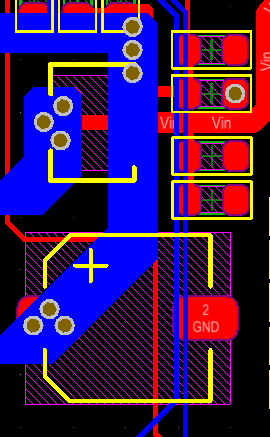
Một số khuyến cáo của nhà sản xuất:

* Giữ đường dẫn chuyển đổi dòng điện ngắn và giảm thiểu diện tích vòng lặp được hình thành bởi jack đầu vào,MOSFET phía cao và diode chuyển mạch.
* Các tụ gốm bypass được đề nghị đặt gần với chân VIN
* Đảm bảo các đường feedback là ngắn nhất và trực tiếp. Đặt các điện trở feedback càng gần chíp càng tốt.
* Đi dây SW cách xa khu vực analog.
* Kết nối IN, SW và đặc biệt là GND tương ứng với một khu vực đồng lớn để làm mát chip để cải thiện hiệu suất nhiệt và độ tin cậy lâu dài.



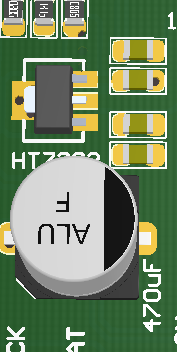
Hình 39- 3D khu vực DC-DC Converter

**3.2.2. Thiết kế layout khối ổn áp 3.3V**



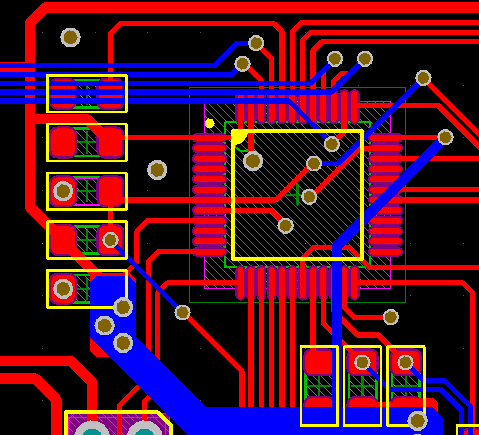
Hình 40- 2D khối ổn áp 3.3V

Khối hạ áp nhằm mục đích đưa nguồn từ 5VDC-3.3VDC để cung cấp cho các chíp hoạt động. Các linh kiện trong khối này được sắp xếp gần nhau và được căn chỉnh hợp lý để bo mạch trong thẩm mỹ hơn. Dây 3.3V cũng được phủ 1 lớp copper..



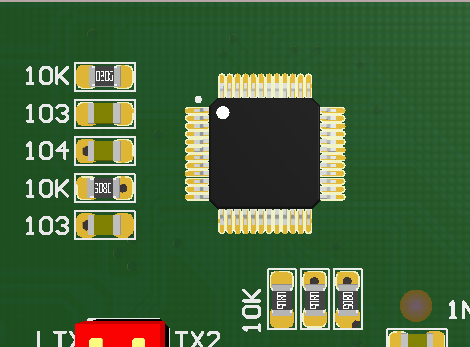
Hình 41- 3D khối ổn áp 3.3V

**3.3. Thiết kế layout khối điều khiển trung tâm**

****

Hình 42- 2D khu vực chíp STM32

Vi điều khiển STM32 được đặt ở phần trung tâm phí trên của bo mạch cách xa các khối khác . Từ đây kéo ra các bus để kết nối với các linh kiện của các khối khác. Các tụ lọc bypass được bố trí gần với chân nguồn của vi điều khiển nhằm chống nhiễu. sử dụng các via để kết nối giữa 2 lớp.



Hình 43- 3D khu vực chíp điều khiển STM32

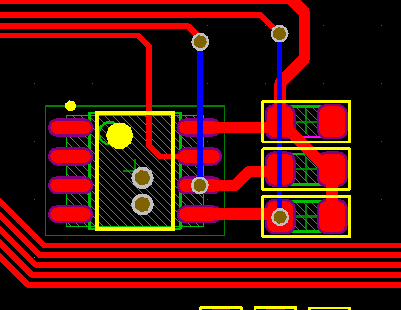
**3.4. Thiết kế layout khối EEPROM**

Một số yêu cầu khi layout chuẩn tốc độ cao I2C:

|  |  |
| --- | --- |
| Tham số | Yêu cầu |
| Tần số tối đa | 100kHz (Standard Mode)  400kHz (Fast Mode) |
| Số thiết bị trên đường truyền | Nhiều tải, 400pF (Standard Mode), 100pF (Fast Mode) |
| Mặt phẳng tham chiếu | GND hoặc PWR, không yêu cầu stitching via khi mặt phẳng tham chiếu là PWR |
| Trở kháng đường truyền | 50Ω ±15% single ended |
| Độ dài đường truyền trên PCB | <450mm (standard mode), <200mm (fast mode) |
| Số via tối đa | Càng ít càng tốt |

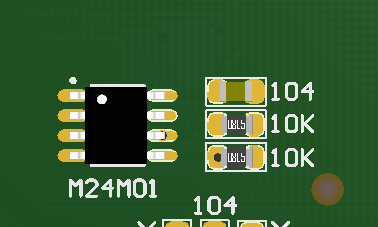
Bảng 10: Một số yêu cầu khi layout I2C tốc độ cao

Trong thiết kế IC M24M01 giao tiếp với vi điều khiển STM32 bằng giao tiếp I2C. Được layout bố trí như hình dưới đây.

****

Hình 44- 2D khu vực M24M01

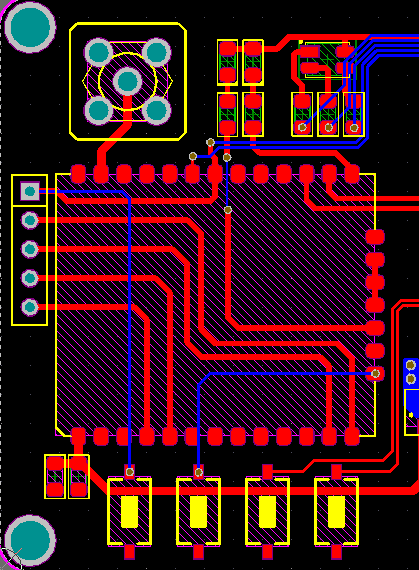
Dưới đây là hình ảnh 3D:



Hình 45- 3D khu vực M24M01

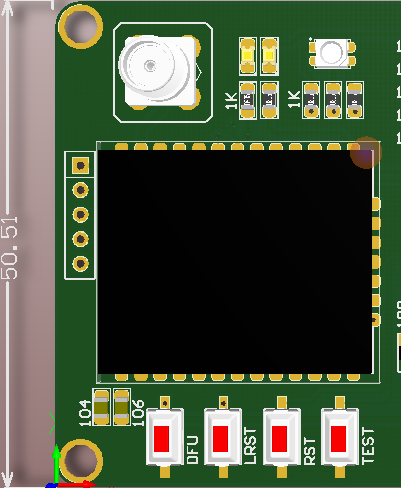
**3.5. Thiết kế layout khối module Lora**

Mô-đun lora được bố trí ở bên trái mạch đặt đế anten của mo-đun cách ở góc mạch tránh xa các linh kiện khác. Các nút bấm reset và các đèn led được sắp xếp hợp lý. Ngoài ra còn có 1 jump để kết nối 1 số chân của module.



Hình 46- 2D module Lora và đế anten

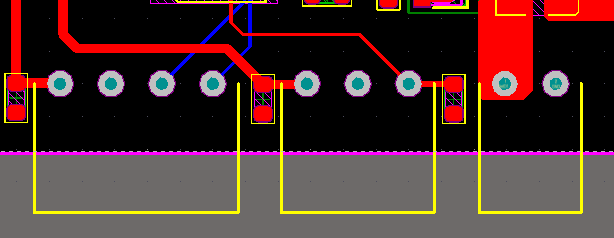
Dưới đây là hình ảnh 3D:



Hình 47- 3D module Lora và đế anten trên bo mạch

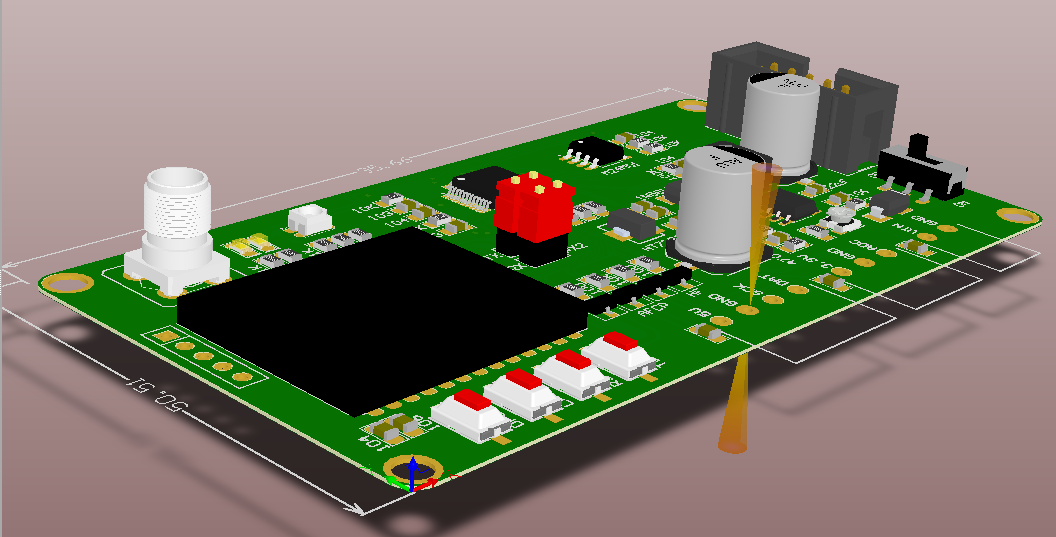
**3.6. Thiết kế layout cho các khối cảm biến**

Để thuận tiện cho việc kết nối các cảm biến ra ngoài, bo mạch bố trí các jack ở bên dưới ở giáp bên dưới mạch bên cạnh jack nguồn. Để có thể dễ dàng cắm các cảm biến vào khu vực đó.

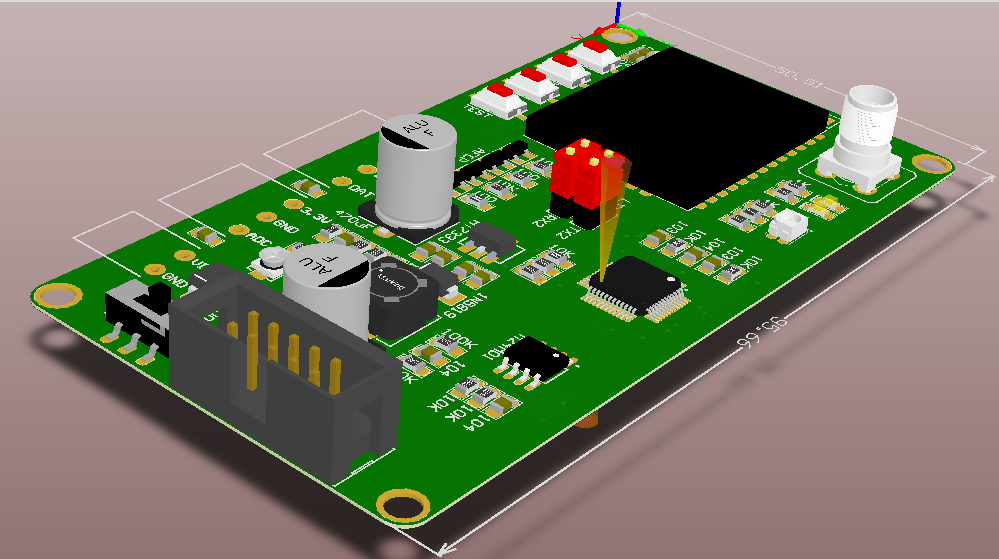


Hình 48- 2D các jack kết nối cảm biến trên bo mạch

**3.7. Hình ảnh 3D của bo mạch**

****

Hình 49- Hình ảnh 3D của bo mạch



Hình 50- Hình ảnh 3D của bo mạch

# CHƯƠNG 4: KẾT LUẬN

**Kết luận:**

Thiết kế nguyên lý và layout là hai bước quan trọng nhất khi thiết kế một mạch điện. Trong đó thiết kế mạch nguyên lý là việc lựa chọn giá trị các linh kiện và kết nối các biểu tượng của các linh kiện thành các sơ đồ mạch điện các chức năng cụ thể. Thiết kế mạch nguyên lý sai sẽ dẫn tới việc mạch thực tế không thể hoạt động hoặc hoạt động sai chức năng. Layout là việc sắp xếp và kết nối các linh kiện sẽ xuất hiện thực tế trên PCB. Layout cần tuân theo các quy tắc, đặc biệt là đối với các tín hiệu tốc độ cao hoặc các tín hiệu tương tự nhạy cảm với nhiễu. Bo mạch đã thiết kế có một số ưu điểm sau:

* Bo mạch có kích thước nhỏ gọn
* Các bo mạch có thể sản xuất số lượng lớn với chi phí hợp lý
* Bo mạch hoạt động ổn định trong điều kiện môi trường ngoài trời