REPORT

THIẾT KẾ VÀ THỰC HIỆN HỆ THỐNG ĐẾM SỐ BƯỚC CHÂN

Design and Implementation of a Pedometer

Ver 1.0

31/12/2024

|  |  |  |  |
| --- | --- | --- | --- |
|  | Họ và tên (Full name) | Mã SV (ID) | Đóng góp (Contribution) |
| Thành viên 1 (Member 1) | Phạm Thái Dương | 21021407 | -Thiết kế và làm mạch  -Xây dụng code thư viện GPIO |
| Thành viên 2  (Member 2) | Lê Toàn Thắng | 21021407 | - Xây dựng code thư viện I2C  - Hoàn thiện code logic |
| Tên/Địa chỉ Repo trên Github | - Phạm Thái Dương: duongthai1801  - Lê Toàn Thắng: thanguetvnu1009 | | |

|  |
| --- |
| Tóm tắt (Abstract - from 5 to 10 lines) |
| Mục tiêu của dự án là vận dụng các kiến thức và kỹ năng đã học để thiết kế và triển khai một bộ đếm số bước chân. Hệ thống sẽ sử dụng mô-đun cảm biến gia tốc MMA8452Q và vi xử lý STM32F103C8T6 để thực hiện các chức năng đo đếm bước chân và hiển thị kết quả trên màn hình LCD. Bộ đếm này sẽ nhận tín hiệu từ cảm biến gia tốc để phát hiện và đếm số bước chân của người sử dụng khi di chuyển. Các chức năng điều khiển như bật/tắt bộ đếm và reset kết quả sẽ được thực hiện thông qua các nút nhấn SW1 và SW2, trong khi các chỉ thị trạng thái sẽ được hiển thị bằng đèn LED và số liệu đếm được sẽ được hiển thị trên màn hình LCD. |

|  |
| --- |
| Từ khóa (Keywords) |
| ARM (Advanced RISC Machine): kiến trúc vi xử lý với bộ lệnh RISC  LCD: Là màn hình được sử dụng để hiển thị thông tin về số bước chân đếm được  I2C: Giao thức truyền thông 2 dây dùng để kết nối giữa màn hình LCD và STM32F103C8T6 để hiển thị số bước chân đếm được. |

Mục lục (Table of Contents)

[Document History 3](#_Toc132787351)

[Table of Contents 4](#_Toc132787352)

[1. Giới thiệu (Introduction) 6](#_Toc132787353)

[2. Yêu cầu đối với thiết kế (Requirements) 6](#_Toc132787354)

[2.1. Yêu cầu đối với thiết kế 6](#_Toc132787355)

[2.2. Đặc tả kỹ thuật (Specification) 7](#_Toc132787356)

[3. Thực hiện hệ thống (Implementation) 7](#_Toc132787357)

[3.1. Kiến trúc phần cứng (Hardware Architecture) 7](#_Toc132787358)

[3.1.1. Khối xử lý trung tâm 8](#_Toc132787359)

[3.1.2. Cảm biến đầu vào 8](#_Toc132787360)

[3.1.3. Các LED 8](#_Toc132787361)

[3.1.4. LCD 8](#_Toc132787362)

[3.2. Lập trình phần mềm 8](#_Toc132787363)

[3.2.1. Thiết lập Clock cho hệ thống 8](#_Toc132787364)

[3.2.2. Khởi tạo các LED 9](#_Toc132787365)

[3.2.3. Khởi tạo các Switch 9](#_Toc132787366)

[3.2.4. Khởi tạo Systick Timer 9](#_Toc132787367)

[3.2.5. Thiết lập mức ưu tiên cho các ngắt 9](#_Toc132787368)

[3.2.6. Chương trình điều khiển (Hàm main()) 9](#_Toc132787369)

[4. Kiểm chứng (Validation) 9](#_Toc132787370)

[5. Kết luận (Conclusion) 9](#_Toc132787371)

[Danh sách hình 17](#_Toc132787375)

[Danh sách bảng 17](#_Toc132787376)

# Giới thiệu (Introduction)

(Dự án này nhằm thiết kế và thực thi một bộ đếm số bước chân sử dụng STM32F103C8T6 và cảm biến gia tốc MMA8452. Hệ thống sẽ đo và đếm số bước đi của người sử dụng, hiển thị kết quả trên màn hình LCD qua giao thức I2C. Mục tiêu là vận dụng các kiến thức đã học về lập trình vi điều khiển, giao tiếp I2C và xử lý tín hiệu từ cảm biến để tạo ra một sản phẩm thực tế, giúp sinh viên nâng cao kỹ năng trong thiết kế hệ thống nhúng.

# Yêu cầu đối với thiết kế (Requirements)

## Yêu cầu đối với thiết kế

Yêu cầu thiết kế:

* Chức năng: Đếm số bước chân khi người sử dụng thiết bị di chuyển và hiển thị kết quả trên màn LCD.
* Các đầu vào:
  + Hệ thống sử dụng cảm biến gia tốc MMA8452 để đo số bước chân.
  + SW1 để chuyển đổi bộ đếm bước chân hoạt động giữa trạng thái hoạt động và trạng thái dừng. Khi bộ đếm dừng hoạt động, nếu bấm SW1 bộ đếm chuyển sang trạng thái hoạt động, và ngược lại. Khi chuyển từ trạng thái dừng sang trạng thái hoạt động, bộ đếm tiếp tục đếm từ số bước chân đang hiển thị trên LCD.
  + SW2 để xóa (reset) kết quả đếm của bộ đếm về không.
* Các đầu ra:
  + Hệ thống có 2 lối ra trạng thái: LED xanh nhấp nháy với tần số 1Hz khi bộ đếm hoạt động; tắt khi hệ thống dừng hoạt động. LED đỏ sáng khi hệ thống dừng hoạt động; tắt khi hệ thống hoạt động.
  + LCD: Hiện thị số bước chân
* Bộ đếm sử dụng một timer để xác định khoảng thời gian nhấp nháy LED xanh.

## Đặc tả kỹ thuật (Specification)

1. Các thành phần chính

|  |  |  |
| --- | --- | --- |
| Thành phần | Miêu tả | Số lượng |
| STM32F103C8T6 | Dùng để xử lý tín hiệu từ cảm biến MMA8452Q, đồng thời điều khiển màn hình LCD qua giao thức I2C | 1 |
| MMA8452Q | Cảm biến gia tốc 3 trục để phát hiện chuyển động và đếm số bước chân | 1 |
| LCD (I2C) | Hiển thị số bước chân | 1 |
| LEDs | Đèn led xanh nhấp nháy với tần số 1Hz khi bộ đếm đang hoạt động, tắt khi bộ đếm dừng  LED đỏ sáng khi bộ đếm dứng, tắt khi bộ đếm hoạt động | 1 |
| Nút nhấn | SW1: Chuyển dổi giữa trạng thái dừng và hoạt động  SW2: Reset bộ đếm | 2 |
| Điện trở 220Ohm |  | 2 |
| Điện trở 10Kohm |  | 2 |
| Board mạch |  | 1 |

Bảng 1. Bảng linh kiện

# 

# Thực hiện hệ thống (Implementation)

## Kiến trúc phần cứng (Hardware Architecture)

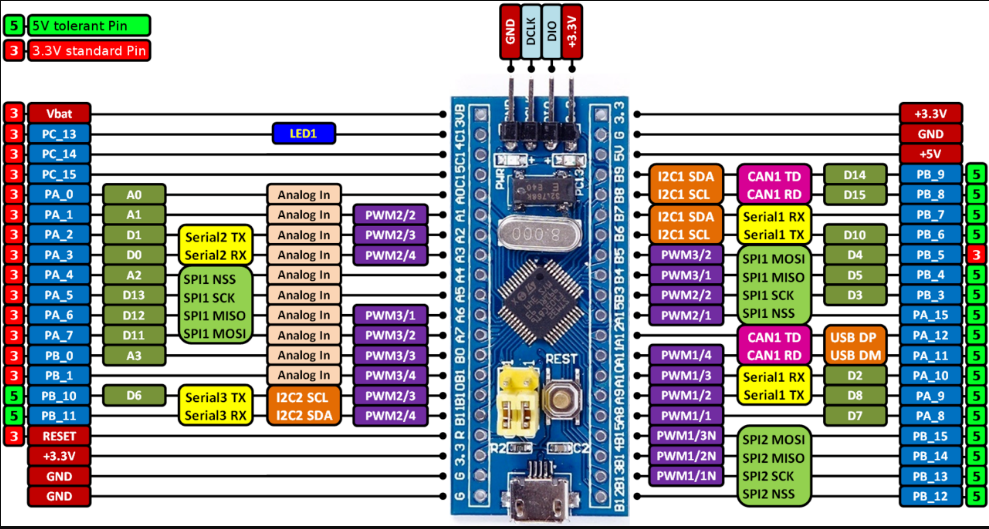
Miêu tả chức năng của các ngoại vi được sử dụng để thực hiện hệ thống đếm bước chân (Bảng 2);

|  |  |  |
| --- | --- | --- |
| STT | Tên ngoại vi | Chức năng |
| 1 | SW1 | Start / Stop bộ đếm |
| 2 | SW2 | Reset bộ đếm |
| 3 | LED | Bóng đèn 2 màu |
| 4 | Điện trở | Điện trở 220Ohm và 10KOHm |

Bảng 2. Tên ngoại vi sử dụng

### Khối xử lý trung tâm

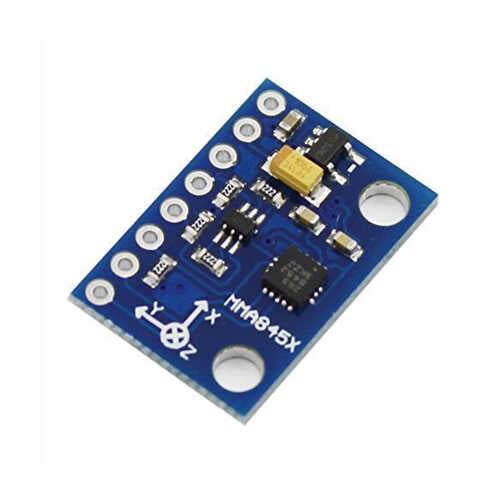
STM32F103C8T6 là vi điều khiển thuộc dòng STM32 của STMicroelectronics, sử dụng kiến trúc ARM Cortex-M3 với tốc độ xung nhịp lên đến 72 MHz. Vi điều khiển này có bộ nhớ Flash 64 KB và SRAM 20 KB, cung cấp 37 chân I/O có thể cấu hình linh hoạt. STM32F103C8T6 hỗ trợ các giao tiếp như USART, SPI, I2C, và ADC 12-bit với 16 kênh, tốc độ tối đa 1 MSPS. Nó cũng tích hợp nhiều bộ định thời (timers) 16-bit và 32-bit, hỗ trợ chế độ PWM. Vi điều khiển này có hệ thống ngắt ngoại vi mạnh mẽ và hỗ trợ các chế độ tiết kiệm năng lượng, rất phù hợp cho các ứng dụng nhúng, điều khiển công nghiệp, cảm biến, IoT, và robot. Với tính năng linh hoạt và hiệu suất cao, STM32F103C8T6 là lựa chọn phổ biến cho các dự án yêu cầu chi phí hợp lý và hiệu quả năng lượng.



Hình 1. STM32F103C8T6

### Cảm biến đầu vào

MMA8452Q là một cảm biến gia tốc 3 trục do NXP Semiconductors sản xuất. Đây là một cảm biến MEMS (Micro-Electro-Mechanical System), cho phép đo gia tốc trên ba trục X, Y và Z, với khả năng hoạt động ở nhiều chế độ và có độ phân giải cao.



Hình 2. Cảm biến gia tốc MMA8452Q

.

### Các LED

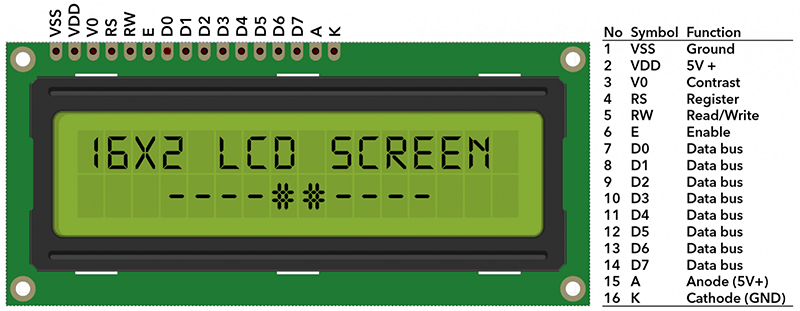
|  |  |
| --- | --- |
| LED | Chức năng |
| Red | STOP |
| Green | START |

Bảng 3. Các chân LED được sử dụng

### LCD và module PCF8574

1. Giới thiệu về màn hình LED LCD 1602

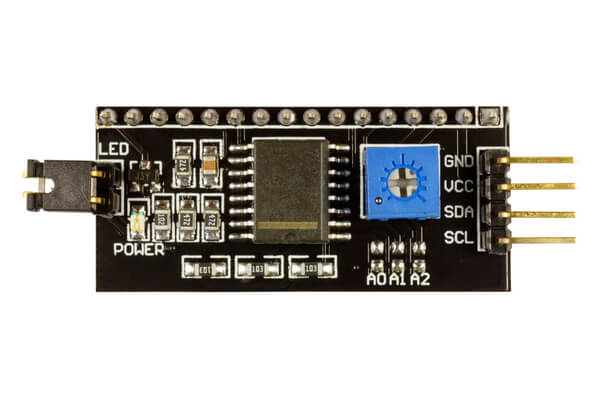
Màn hình LED LCD 1602 là một loại màn hình hiển thị dạng ký tự. Với kích thước 16 cột và 2 hàng, nó có thể hiển thị tối đa 32 ký tự, thường được sử dụng để hiển thị thông tin văn bản như dữ liệu, trạng thái, hoặc các thông báo. Màn hình này sử dụng công nghệ LCD và có đèn nền LED giúp hiển thị rõ ràng trong điều kiện ánh sáng yếu. Đặc biệt, màn hình có thể giao tiếp qua I2C hoặc parallel, giúp dễ dàng tích hợp với vi điều khiển. Với giá thành hợp lý, tiết kiệm năng lượng và dễ sử dụng, màn hình LCD 1602 là lựa chọn lý tưởng cho các ứng dụng đơn giản cần hiển thị thông tin dạng văn bản.



Hình 3. Màn hình LED LCD 1602

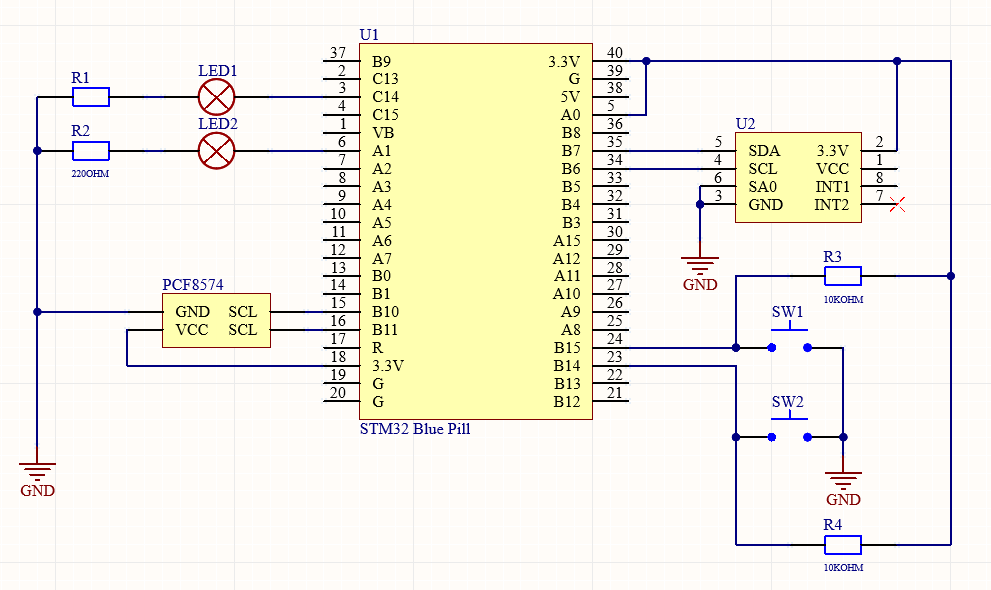
1. Giới thiệu về module PCF8574

Module PCF8574 là một module mở rộng I/O sử dụng giao thức I2C, cung cấp 8 chân vào/ra (I/O). Nó cho phép tiết kiệm chân kết nối và dễ dàng mở rộng số lượng I/O mà không cần nhiều dây nối. Các chân của module có thể được cấu hình làm đầu vào hoặc đầu ra, rất hữu ích cho việc điều khiển các thiết bị như đèn LED, công tắc, cảm biến, hay relay.



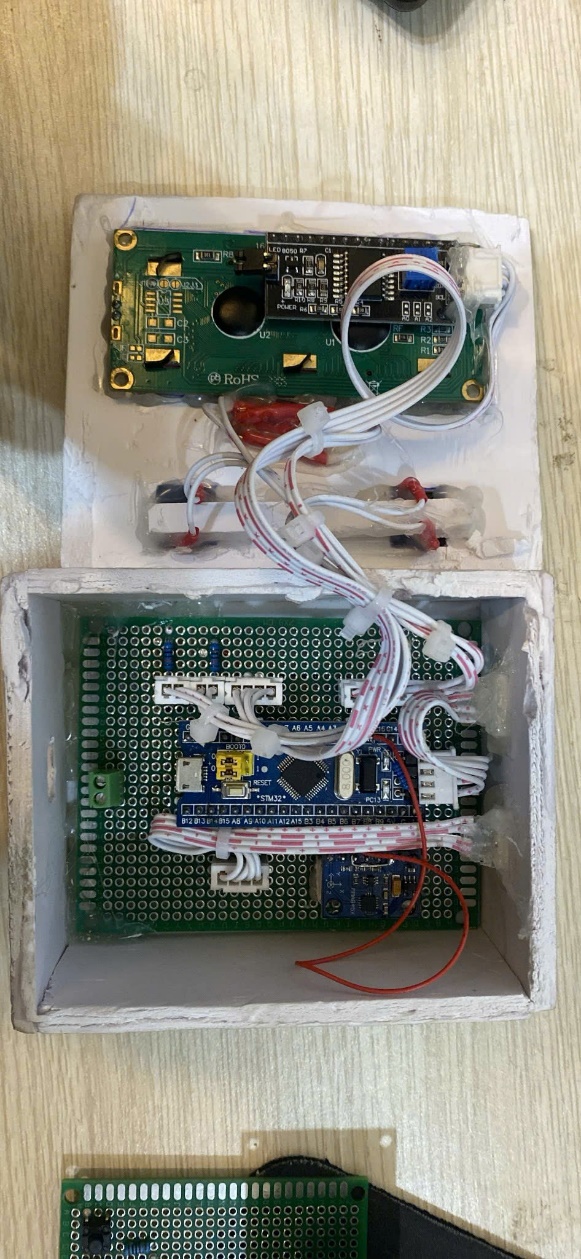
Hình 4. Module PCF8574

### Mạch nguyên lý thiết kế bằng Altium



Hình 5. Sơ đồ nguyên lý được thiết kế trên Altium

### Mạch sau khi hoàn thiện phần hộp kỹ thuật



Hình 6. Mạch sau khi hoàn thiện

## Lập trình phần mềm

### Thiết lập GPIO

1. Giới thiệu về GPIO

GPIO là từ viết tắt của General purpose I/O ports tạm hiểu là nơi giao tiếp chung giữa tín hiệu ra và tín hiệu vào. Ở STM32 thì các chân GPIO chia ra làm nhiều Port vd: PortA, PortB….. Số lượng Port phụ thuộc vào số lượng chân (pin) và cách gọi phụ thuộc vào nhà sản xuất. Mỗi Port thường có 16 chân đánh số từ 0 -> 15 tương ứng với mỗi chân là 1bit. Mỗi chân có 1 chức năng khác nhau như analog input, external interrupt.. hay đơn thuần chỉ là xuất tín hiệu on/off ở mức 0,1. Chức năng của mỗi chân thì chúng ta cần tra datasheet của nhà sản xuất trước khi lập trình hoặc thiết kế mạch.

1. Chức năng của các chân GPIO

* Input floating: Chân I/O cấu hình là đầu vào, không có trở kéo lên hay kéo xuống (nổi).
* Input pull-up: Chân I/O cấu hình là đầu vào, có trở kéo lên (tới VDD).
* Input pull-down: Chân I/O cấu hình là đầu vào, có trở kéo xuống (tới GND).
* Analog: Chân I/O cấu hình là chế độ tương tự, dùng cho ADC/DAC.
* Output open-drain: Chân I/O cấu hình là đầu ra, khi control = 0, chân nối GND; khi control = 1, chân để nổi.
* Output push-pull: Chân I/O cấu hình là đầu ra, khi control = 0, chân nối GND; khi control = 1, chân nối VDD.
* Alternate function push-pull: Chân I/O vừa có thể làm đầu vào, vừa làm đầu ra, output giống push-pull, không có điện trở kéo lên/kéo xuống, dùng cho remap.
* Alternate function open-drain: Chân I/O vừa có thể làm đầu vào, vừa làm đầu ra, output giống open-drain, không có điện trở kéo lên/kéo xuống, dùng cho remap.

1. Một số thanh ghi quan trọng

- GPIOx\_CRx - Port configuration register:

Thanh ghi mode:

00: Input mode (reset state).

01: Output mode, max speed 10 MHz.

10: Output mode, max speed 2 MHz.

11: Output mode, max speed 50 MHz.

Thanh ghi CNF:

Khi là input:

00: Analog mode.

01: Floating input (reset state).

10: Input with pull-up / pull-down.

11: Reserved.

Khi là output:

00: General purpose output push-pull.

01: General purpose output Open-drain.

10: Alternate function output Push-pull.

11: Alternate function output Open-drain.

- GPIOx\_IDR - Port input data register.

Đây là thanh ghi đọc giá trị đầu vào của từng chân. Khi đầu vào ở chân nào đó có mức logic là 1 thì bit tương ứng với chân đó sẽ có giá trị là 1. Và ngược lại là 0.

- GPIOx\_ODR - Port output data register

Đây là thanh ghi quyết định mức logic của đầu ra trên chân STM32 tương ứng với mode output. Giá trị của bit nào ở mức cao thì output của nó sẽ ở mức cao nếu sử dụng mode output push pull và ngược lại. Riêng mode output open drain thì có chút khác biệt.

1. Cấu hình ví dụ cho LED on board

void GPIO\_Config(void)

{

// Bật đồng hồ cho GPIOC (RCC)

RCC->APB2ENR |= RCC\_APB2ENR\_IOPCEN; // Bật đồng hồ GPIOC

// Cấu hình C13 làm đầu ra (Push-pull)

GPIOC->CRH &= ~GPIO\_CRH\_MODE13; // Xóa chế độ hiện tại của C13

GPIOC->CRH |= GPIO\_CRH\_MODE13\_1; // Cấu hình tốc độ 2MHz cho C13 (Output mode)

GPIOC->CRH &= ~GPIO\_CRH\_CNF13; // Cấu hình chế độ push-pull

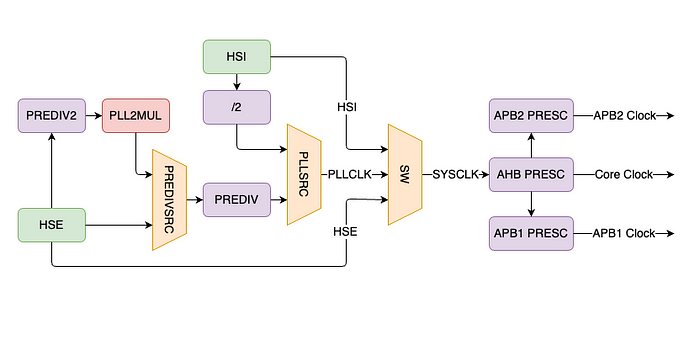
}

### Thiết lập Clock cho hệ thống

1. Sơ lược về lý thuyết

Clock là một phần quan trọng của vi điều khiển. Bất kì 1 ngoại vi nào cũng cần clock để hoạt động. Phần mềm lập trình cũng như mạch nạp có thông dụng, dễ sử dụng, giá cả ra sao…... Một phần quan trọng nữa đó clock của hệ thống, clock hệ thống quan trọng cũng như nguồn nuôi của mạch điện, không có clock hệ thống vi xử lí sẽ không hoạt động được. Clock hệ thống quyết định tốc độ xử lí của vi điều khiển. Vi điều khiển có clock hệ thống càng lớn(tính theo Hz) thì tốc độ xử càng nhanh. Ví dụ: Khi các bạn thay đổi clock thì tốc độ delay của hàm delay tương đối sẽ thay đổi theo. Clock tối đa của STM32F103C8 là 72Mhz.

Khi có thạch anh ngoài thì tốc độ tối đa sẽ là 72Mhz(thường thạch anh có giá trị 8Mhz) và đó cũng chính là tốc độ mặc định khi chưa cấu hình gì. Khi không có thạch anh ngoài thì tốc độ tối đa sẽ là 64Mhz nếu được cấu hình và tốc độ mặc định sẽ là 8Mhz nếu không cấu hình gì



Hình 7. Clock Configuration

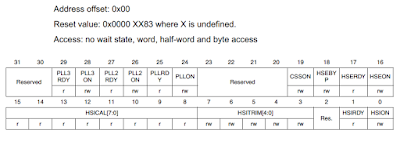
Các nguồn clock bao gồm:

* + Nguồn clock ngoài bên ngoài chip:
    - HSE: là nguồn clock chính để nuôi MCU và các ngoại vi. Thường sử dụng thạch anh 8Mhz.
    - LSE : là nguồn clock cung cấp cho bộ RTC hoạt động và sử dụng cho backup domain khi nguồn VDD bị mất. Thường sử dụng thạch anh 32.768Khz.
  + Nguồn clock ngoài bên ngoài chip:
    - LSI RC: sử dụng các mạch dạo động RC để tạo nguồn clock phụ nhằm để bộ RTC hoạt động.
    - HSI RC: sử dụng các mạch dao động RC để tạo nguồn clock cung cấp cho MCU và các ngoại vi khi không có thạch anh hay clock bên ngoài.

Chúng ta nên sử dụng các thạch anh hay nguồn clock bên ngoài để tạo được clock hệ thống tối đa cũng như để mạch hoạt động ổn định hơn.

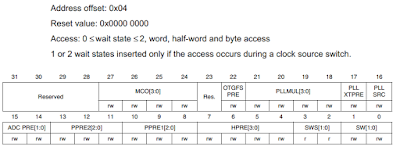
1. Một số thanh ghi quan trọng.

- RCC\_CR–Clock control register



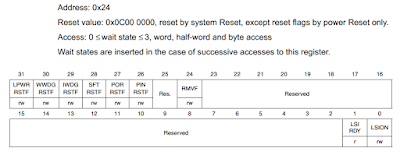
* PLLx: bộ nhân tần số clock, PLLxDRY cờ báo trạng thái khóa(=1) hay không khóa(=0) bộ nhân tần sô đó, PLLxON là bit cho phép(=1) hoặc không cho phép(=0) bộ nhân đó hoạt động.
* CSSON : bit cho phép kiểm tra clock hệ thống có sử dụng HSE hay không.HSEBYP/HSERDY : cho phép sử dụng HSE với bypass clock source hoặc crystal/ceramic resonator.
* HSEON: bit cho bật HSE hay không.
* HSIRDY: cờ báo bộ HSI đã sẵn sang hay chưa.
* HSION : bit cho phép bộ HSI hoạt động hay không.

- RCC\_CFGR – clock configuration register.



* MCO[3:0]: chọn nguồn clock output ra trên chân PA8 là nguồn nào như SYSCLK, HSI,HSE... tham khảo thêm trong reference manual.
* OTGFSPRE: bộ chia clock sử dụng cho giao tiếp USB.
* PLLMUL[3:0] : bộ nhân clock sau khi chọn lựa clock là HSE hay HSI.
* PLLXTPRE : bit này điều khiển thanh ghi RCC\_CFGR2.
* PLLSRC: bit chia đầu vào của bộ PLL.
* ADCPRE[1:0]: bit chia clock cho bộ ADC.
* PPRE2[2:0]: bit chia clock cho bộ APB2.
* PPRE1[2:0] : bit chia clock cho bộ APB1.
* HPRE[3:0] : bit chia clock cho bộ APB(bao gồm APB1 và APB2).
* SWS[1:0] : cờ báo trạng thái sử dụng clock hệ thống từ nguồn nào HSI, HSE, PLL.
* SW[1:0] : bit báo chuyển trạng thái sử dụng clock hệ thống nào.

- RCC\_CSR - Control/status register



* LPWRRSTF : cờ báo reset do VDD nhỏ hơn giới hạn dưới.
* WWDGRSTF : cờ báo reset do bộ window watchdog.
* IWDGRSTF : cờ báo reset do bộ Independent watchdog.
* SFTRSTF :cờ báo reset do phần mềm.
* PORRSTF: cờ báo reset khi khởi động.
* PINRSTF: cờ báo reset do tác động trên chân reset.
* RMVF: cờ báo reset do tác động bằng phần mềm.
* LSIDRY : cờ báo bộ LSI nội đã được kích hoạt.
* LSION: cờ cho phép bộ LSI hoạt động.

### Khởi tạo các LED

Code ví dụ cho LED:

    // Configure PC14 (LED Stop) as output push-pull, max speed 50 MHz

    LED\_PORT\_STOP->CRH &= ~(unsigned int)(0xF << ((14 - 8) \* 4)); // Clear configuration for PC14

    LED\_PORT\_STOP->CRH |= (0x3 << ((14 - 8) \* 4));  // Output mode, 50 MHz

### Khởi tạo các Switch

#### Cấp xung clock cho cổng GPIO:

#### Thiết lập ngắt cho Switches:

### Khởi tạo Systick Timer

### Thiết lập mức ưu tiên cho các ngắt

### Chương trình điều khiển (Hàm main())

# Kiểm chứng (Validation)

# Kết luận (Conclusion)

Hình được sử dụng

[Hình 1. STM32F103C8T6 7](#_Toc186591510)

[Hình 2. Cảm biến gia tốc MMA8452Q 8](#_Toc186591511)

[Hình 3. Màn hình LED LCD 1602 9](#_Toc186591512)

[Hình 4. Module PCF8574 10](#_Toc186591513)

[Hình 5. Sơ đồ nguyên lý được thiết kế trên Altium 10](#_Toc186591514)

[Hình 6. Mạch sau khi hoàn thiện 11](#_Toc186591515)

[Hình 7. Clock Configuration 14](#_Toc186591516)

Bảng được sử dụng

[Bảng 1. Bảng linh kiện 6](#_Toc186590506)

[Bảng 2. Tên ngoại vi sử dụng 7](#_Toc186590507)

[Bảng 3. Các chân LED được sử dụng 9](#_Toc186590508)