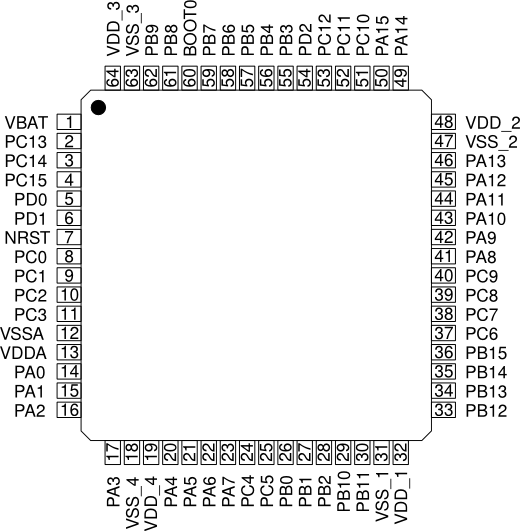
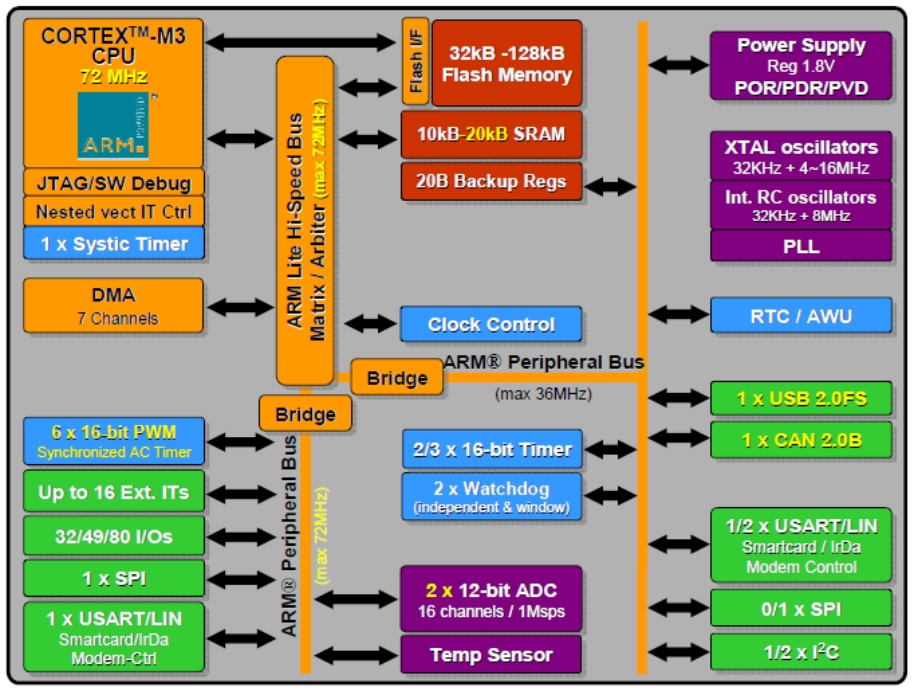
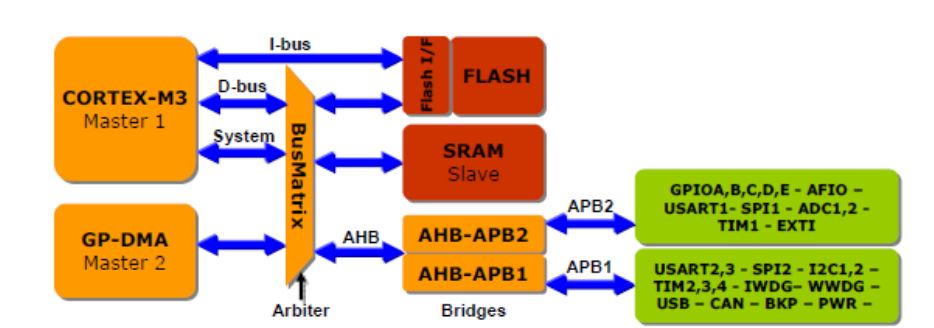
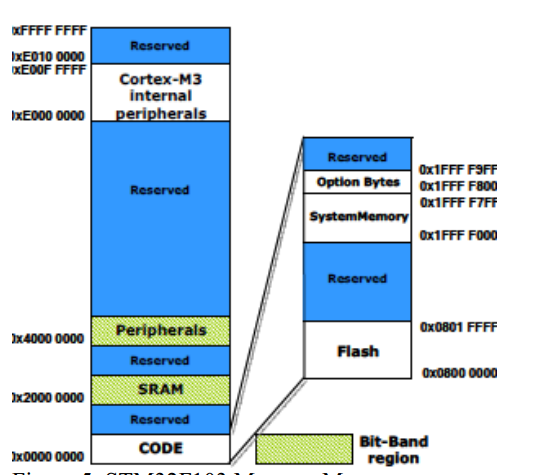
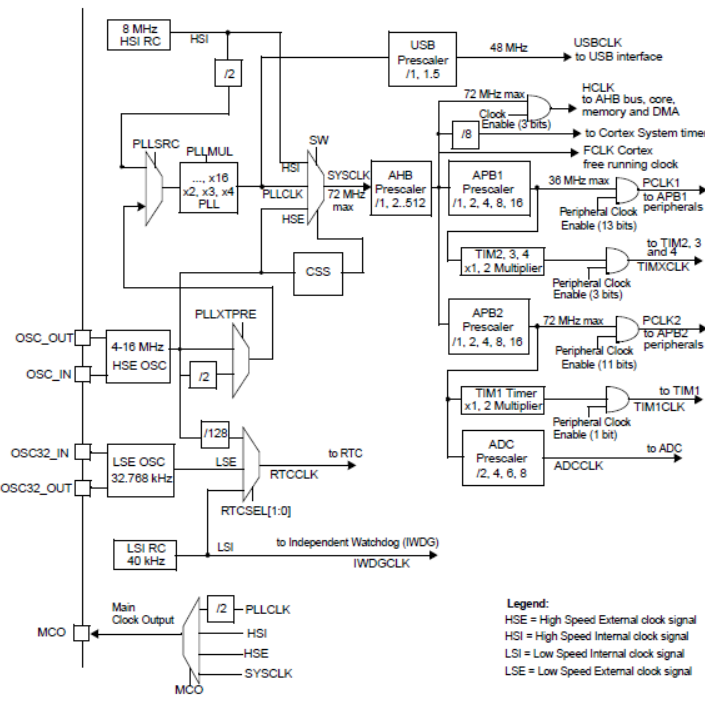
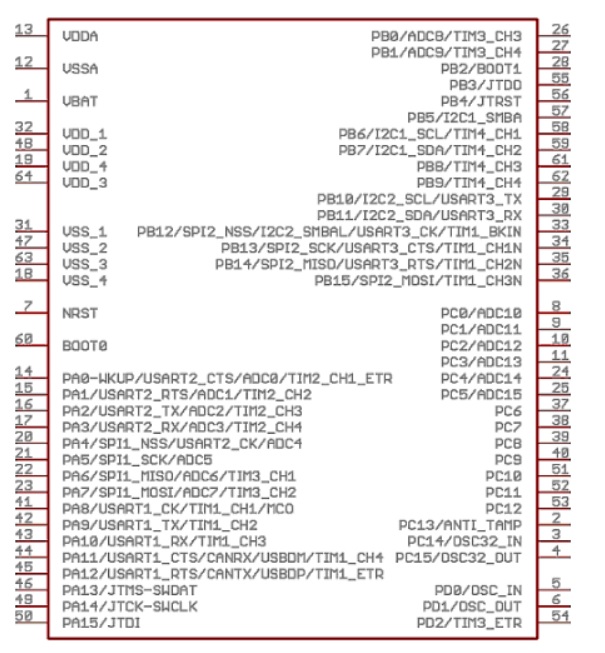
Vi điều khiển STM32F103XX được sản xuất bởi ST company và ứng dụng trong đồ án này. Baltic Engineering Co. quyết định sử dụng vi điều khiển STM32F103 trong phần lớn các dự án vì vi điều khiển này có rất nhiều tính năng cao cấp so với dòng vi điều khiển khác. Đây là một trong số những vi điều khiển tốt nhất trong dòng MCU 32-bit, hiệu suất cao nhất để điều khiển và kế nối trong các dự án điện tử, nó có thể dùng trong các giải pháp xử lý tín hiệu số ( nhờ vào xung nhịp cao ), ứng dụng năng lượng thấp để tiết kiệm năng lượng cho hệ thống, tốc độ của các ngoại vi tăng lên khi hiệu suất cao hơn.  
Overview  
Họ vi điều khiển của STM32F103xx bao gồm lõi ARM Cortex-M3 32-bit, bộ nhớ nhúng tốc độ cao ( Bộ nhớ Flash lên tới 128 Kbyte và SRAM lên tới 20Kb), I/Os và các ngoại vi được hoạt động đông thời nhờ vào kết nối đến 2 buses APB. Vi điều khiển STM32F103xx gồm rất nhiều ngoại vi như 2 ADC 12-bit, 1 bộ điểu khiển Timer cao cấp, 3 GP Timer 16 bit và có Timer PWM. Nó cũng mang đến 2 I2C và SPI, 3 USART, 1 USB và 1 CAN là giao thức truyền thông.   
Hình dưới đây cho thấy các chân ra của STM32F103xx được sử dụng trong project.  
   
STM32F103xx pinout  
  
Họ vi điều khiển này chứa 3 port là PA,PB,PC là các port của MCU, mỗi port có 16 chân I/O. Vss, Vss và Vbat được dùng để tính sai số của vi điều khiển bằng cách dùng điện áp đầu vào tham chiếu.  
  
Họ vi điều khiển của STM32F103xx   
  
Họ vi điều khiển STM32F103xx được chia thành 3 nhóm:  
Low-density: STM32F103x4 và STM32F103x6 là các thiết bị low-density.  
Medium-density: STM32F103x8 và STM32F103xB là các thiết bị medium-density.  
High-density: STM32F103xC, STM32F103xD và STM32F103xE là các thiết bị high-density.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Pinout | Low-denstisy | | Medium-density | | High-density | | |
| 16 KB Flash | 32 KB Flash | 64 KB Flash | 128 KB Flash | 256 KB Flash | 384 KB Flash | 512 KB Flash |
| 6 KB RAM | 10 KB RAM | 20 KB RAM | 20 KB RAM | 48 KB RAM | 64 KB RAM | 64 KB RAM |
| 144 |  |  |  |  | 5 x USART  4 x 16-bit timers, 2 x basic timers  3 x SPI, 2 x I2S, 2 x I2C USB, CAN, 2 x PWM timer 3 x ADC, 1 x DAC, 1 x SDIO  FSMC ( 100 và 144 chân ) | | |
| 100 |  |  | 3 x USART  3 x 16-bit timer  2 x SPI, 2 x I2C, USB, CAN, 1 x PWM timer, 2 x ADC | |
| 64 | 2 x USART  2 x 16-bit timer  1 x SPI, 1 x I2C, USB, CAN, 1 x PWM timer  2 x ADC | |
| 48 |  |  |  |
| 36 |  |  |  |  |  |

Ba nhóm trên được chia theo chức năng của họ vi điều khiển STM32F103xx  
Vi điều khiển low-dentisy có bộ nhớ Flash thấp hơn và RAM, ít timer và ngoại vi hơn 2 nhóm còn lại. Vì vậy, Medium-density và High-dentisy bao gồm bộ nhớ Flash cao hơn, RAM tốt hơn và có nhiều ngoại vi mở rộng hơn.  
Dòng Low-density bao gồm 16 KB đến 32 KB bộ nhớ Flash và 6 KB đến 10 KB RAM. Nó chứa 1 CAN, 1 USB, 1 PWM timer, 1 I2C, 1 SPI và 2 ADC, 2 USART , 2 timer 16 bit. Sự khác nhau giữa các dòng Low-density là số lượng gói chân ra của nó. Có 3 loại gói chân ra là 36,48,64 chân. Bộ nhớ flash cũng tăng từ 64 KB đến 128 KB và RAM 20 KB.  
Dòng Medium-density có nhiều đặc tính hơn so với dòng low-density. Số lượng ngoại vi và chân ra của chúng được tăng lên. Có tất cả 1 CAN, 1 USB, 1 PWM timer, 2 I2C, 2 SPI, 2 ADC, 3 USART, 3 timer 16 bit. Nó có tổng cộng 3 loại gói chân gồm 48,64,100 chân.   
Dòng High-density thì hơn hoàn toàn các dùng khác và có nhiều ngoại vi hơn. Vi dụ như: 1 × CAN, 1 × USB, 1 × PWM timer, 2 × I²Ss (SPI), 2 × I²C, 10 3 × SPI, 2 × ADCs, 1 × DAC, 5 × USARTs, 2 × basic timers, 4 × 16-bits timers, 1 × SDIO (Secure Digital Input Output), and 1 ×FSMC (Flexible Static Memory Controller).   
Vi điều khiển STM32F103xx thực sự thuận tiện để sử dụng cho rất nhiều các ứng dụng khác nhau như trong công nghiệp, điều khiển, thiết bị y tế, Thiết bị chơi game trên PC, GPS, Liên lạc qua video, hệ thống cảnh báo …  
Cấu trúc của MCU STM32F103  
Phần này giải thích về lõi của vi điều khiển, bộ nhớ, chân IO và các ngoại vi có trong sơ đồ dưới đây:  
  
  
STM32F103 Performance line Block Diagram  
  
Cấu trúc hệ thống bao gồm Buses, DMA, SRAM nội, Bộ nhớ Flash nội, một vài trong số đó được cân nhắc làm master và còn lại là slave.   
Hình dưới cho thấy cấu trúc bus cho họ vi điều khiển STM32. Việc truy cập giữ bus lõi hệ thống và bus DMA được điều khiển bởi BusMatrix.  
  
  
System Architecture   
  
Có 4 khối master và 3 khối slave trong cấu trúc được nói đến ở trên  
  
Master:  
I-bus : Kết nối lõi Cortex M3 tới cấu trúc bộ nhớ flash để thực thi.  
D-bus: Kết nối lõi M3 đến bộ nhớ flash về dữ liệu giao diện  
S-bus: kết nối bus ngoại vi của M3 đến BoxMatrix để điều khiển thành phần ản giữa DMA và Lõi.  
GP-DMA: Kết nối CPU và DMA đến bộ nhớ flash, SRAM và ngoại vi thông qua BoxMatrix để tạo ra kết nối giữa chúng.  
  
Slave:  
SRAM nội  
Bộ nhớ Flash nội  
AHB đến APB: cầu chia bus AHP vào 2 bus, APB1,APB2. APB1 để dùng cho ngoại vi mà có tần số 36 MHz và APB2 dùng cho ngoại vi chạy trên 72 MHz.  
  
**Lõi ARM M3** là CPU của vi điều khiển và là 1 trong những phần quan trọng nhất của của vi điều khiẻn. Lõi này là bản mới nhất của vi xử lý ARM được ứng dụng cho hệ thống nhúng. Nó hoạt động nhanh nhất trên xung nhịp 72 MHz, 90DIMPS với 1.25 DIMPS/MHz, hiệu suất tại trạng thái truy cập bộ nhớ 0.   
Hệ thống nhớ của vi điều khiển bao gồm 2 phần là nhớ flash và nhớ SRAM. Bộ nhớ flash để lưu trữ dữ liệu và chương trình và có thể nâng lên thành 128 Kb. Bộ nhớ SRAM là bộ nhớ đọc nghi của CPU với trạng thái đợi 0 để lưu trữ dữ liệu cho việc xử lý bởi USB và có thể lên tới 20 Kb.  
NVIC: Nó có thể được dùng để điều khiển lên tới 43 kênh ngắt ( là những ngắt đặc biệt có thể enable/disable hay điều khiển bởi chương trình) và, nó có 16 mức ưu tiên lập trình. Nó được dùng để đặt ưu tiên kênh IRQ.  
EXTI : nó có 19 lines bắt sườn để tạo ngắt/ sự kiện yêu cầu. Để bắt được trigger ngoại có thể sử sụng, có thể để bắt thể bắt theo sườn lên hoặc sườn xuống hoặc cả lên và xuống. Nó có thể kết nối đến 80 GPIO đến 69 ngắt ngoài để bắt trigger ngoại ( giống như ngắt ).  
Clock system: bao gồm các nguồn clock các nhau trong vi điều khiển như HIS, HSE, PLL, LSI RC và LSE.  
HSI là tín hiệu clock nội tốc độ cao có 8 Mhz RC nội. Thường được ứng dụng như clock hệ thống cho MCU mặc định.  
HSE là tín hiệu clock ngoại tốc độ cao và nó có thể sử dụng 4 -6 MHz xung ngoại để cung cấp tín hiệu clock ổn định để cho clock hệ thống.  
PLL có thể gen tín hiệu ra ổn định và chính xác từ một nguồn tần số thấp cố định. Nó kết hợp xung ra từ HIS RC hay clock thạch anh đầu ra HSE để tạo ra những tần số khác nhau cho vi điều khiển và các ngoại vi được dùng.  
Clock LSI là tín hiệu clock nội tốc độ thấp và tần số clock cả nó khoảng 40 KHz. Clock LSI RC có thể được dùng cho nguồn clock thấp và nó rất hữu dụng trong chế độ nghỉ hoặc dừng.  
Dao động LSE là thạch anh ngoại tốc độ thấp, tần số của nó vào khoảng 32 KHz và dùng cho clock thời gian thực. Nguồn clock này cấp tần số clock cho mỗi phần của vi điều khiển như CPU, các ngoại vi …   
**Startup clock** là clock hệ thống cho vi điều khiển trong trạng thái khởi động. clock này tạo ra bởi HSI RC mặc định và clock này lf khoảng 8 MHz.  
**Boot modes** xác định làm sao để boot CPU để làm việc. Có tất cả 3 chế độ boot cho vi điều khiển. Map bộ nhớ của vi điều khiển được chỉ ra dưới đây:   
Có thể sử dụng các phần khác nhau của vi điều khiển để boot CPU   
  
STM32F103 Memory Map  
  
Một trong các chế độ boot này được giới thiệu ở dưới có thể dduwwojc dùng trong Startup để boot CPU,  
User Flash: CPU sẽ boot từ Flash  
Bộ nhớ hệ thống: CPU sẽ boot từ bộ nhớ hệ thống  
SRAM: CPU sẽ boot từ SRAM.  
  
Nên chương trình bootloader là rất cần để có thể chạy và khởi chạy chương trình trên Flash bằng cách dùng USART. Bootloader được để ở bộ nhớ hệ thống.  
**Power Supply** là để cấp cho vi điều khiển để từ đó cấp cho các phần khác nhau của hệ thống. Nó được mô tả chân phải kết nối đến bộ nguồn:   
Vdd: Pin cấp nguồn cho IO và ổn áp. Vdd nên được nối đến nguồn 2.0 đến 3.6V và điện áp này cấp qua chân Vdd trên MCU.  
Vssa,Vdda: chân này cấp nguồn 1 chiều ngoài cho ADC, khối Reset, RCs và PLL. Các chân này nên được nối đến điện áp 2.0 đến 3.6 V   
Vi điều khiển này bao gồm POR/PDR, đây là một mạch tích hợp và nó thường được kích hoạt để luôn giữ 2.0 V. Khi Vdd nhỏ hơn 1 ngưỡng nào đó, vi điều khiển không thể hoạt động và nó sẽ giữ ở trạng thái reset. PVD so sánh Vdd với ngưỡng Vpvd. ISR báo đến vi điều khiển hoặc đưa nó vào trạng thái an toàn, nếu Vdd lên hoặc xuống so với Vpvd  
**RTC** và thanh ghi Backup cấp một bộ đếm chương trình 32 bit cho MCU để cấp đặt bộ đếm hoạt động liên tục. Nó được dùng để cấp cho hàm lập lịch, ngắt chu kì và ngắt cảnh báo. Nguồn được cấp bởi Vdd hay Vbat cho RTC và thanh ghi Backup.  
**General Purpose Timers (TIMx)** tạo ra 3 timer chuẩn có thể đồng bộ để dùng cho MCU. Mỗi một timer bao gồm bộ đếm counter 16 bit, bộ băm xung.  
**Advanced Control Timer (TIM1)** giống với TIMx nếu nó được cấu hình như timer 16 bit chuẩn, nhưng được hoàn thiện hơn. Nó có thể dùng 4 kênh cho Input capture, Output compare, PWM… Nó cũng có thể làm việc với 3 pha băm xung kết hợp trên 6 kênh.  
**I2C** là giao diện bus và có thể đưa ra trong 2 chế độ là master và slave. Vi điều khiển này chưa lên đên 2 bus I2C.   
GPIO là giao diện của MCU để kết nối đến các thiết bị bên ngoài có thể sử dụng như là đầu vào để đọc dữ liệu và dùng như đầu ra để ghi dữ liệu… Trên đây có sẵ 3 GPIO trên MCU ( STM32F103 với 80 chân. Nó nên được cấu hình mềm và được cân nhắc trong suốt lúc làm việc với MCU. Tất cả các ngoại vi của MCU có thể kết nối ra bên ngoài hoặc các thiết bị ngoài khác bở GPIO  
  
Cấu hình vi điều khiển   
Cấu hình vi điều khiển đặt xung clock cho CPU của vi điều khiển và các ngoại vi của nó. Cấu hình cho kiểu bộ dao động của vi điều khiển và đặt chính các tần số cho mỗi một phần của vi điều khiển. Nó cũng xác định chân vi điều khiển cho các ngoại vi như là I/O, chế độ hoạt động … . Nó chỏ cấu hình những ngoại vi cần thiết, những cáu mà nó cần và được chọn ra trong dự án. Với đề tài này thì chúng ta chỉ chọn GPIO.   
Cấu hình RCC   
Bước đầu tiên để cấu hình là chuẩn bị clock hệ thống cho vi điều khiển và nó rất quan trong trong lần đầu tiên vì clock hệ thống bật lên trên vi xử lý và việc kích hoạt ngoại vi cũng là rất cần thiết. Bằng việc cấu hình RCC, nó có thể chạy clock hệ thống.   
HSE được xác định như clock nguồn cho vi điều khiển, bởi vì nó có thể hoạt động ở 72 MHz (tần số clock hệ thống cực đại) nếu HSE được dùng như clock PLL đầu vào. Một lý do khác là nó có thể đưa ra tần số clock cho MCU rất chính xác. Vậy nên, HSE được chuẩn bị để đưa ra như tần số clock nguồn cho CPU và mỗi ngoại vi trên vi điều khiển.  
  
  
Clock tree for microcontroller STM32F103xx  
  
Sauk hi cấu hình, tần số clock của CPU và các ngoại vi có thể được set. Sau đó CPU có thể bắt đầu làm việc và các ngoại vi thì sẵn sàng sử dụng.  
Hình trên cũng đưa ra máy dao động HSE được dùng như clock đầu vào PLL và đầu ra của PLL là nguồn clock hệ thống. Tần số SYSCLK nên thay đổi, căn chỉnh theo mỗi phần khác nhau của vi điều khiển, bởi vì mỗi ngoại vi làm việc với những tần số clock riêng biệt.  
Cấu hình GPIO  
Cấu hình GPIO chọn ra và kích hoạt các port của vi điều khiển hay bất kì chân nào của vi điều khiển để dùng cho vi điều khiền trong các task. Cấu hình này xác định IO và trạng thái của mỗi port hay mỗi chân.   
Hình dưới sẽ đưa ra cấu hình chân của vi điều khiển STM32F103. Việc cấu hình cho các chân này là rất quan trong để cấy hình các port hay chân của vi điều khiển cho chương trình và nó nên được cân nhắc. Số lượng chân và tính chất của nó được nhắc đến ở hình dưới và nó rất hữu ích cho việc cấu hình các chân của vi điều khiển.  
  
  
Pin Configuration of the STM32F103  
  
Các ngoại vi của vi điều khiển có sẵn từ các chân của Port A, B, C. Vậy nên, việc kích hoạt các port này bởi chương trình là rất cần thiết theo yêu cầu công việc và nó có được ở việc cấu hình GPIO. Trong cấu hình, clock cho GPIO được kích hoạt cho tất cả các port để kích hoạt port của nó.   
Cấu hình NVIC  
Trong việc cấu hình NVIC, ưu tiên ngắt của các ngoại vi được đặt theo chương trình.