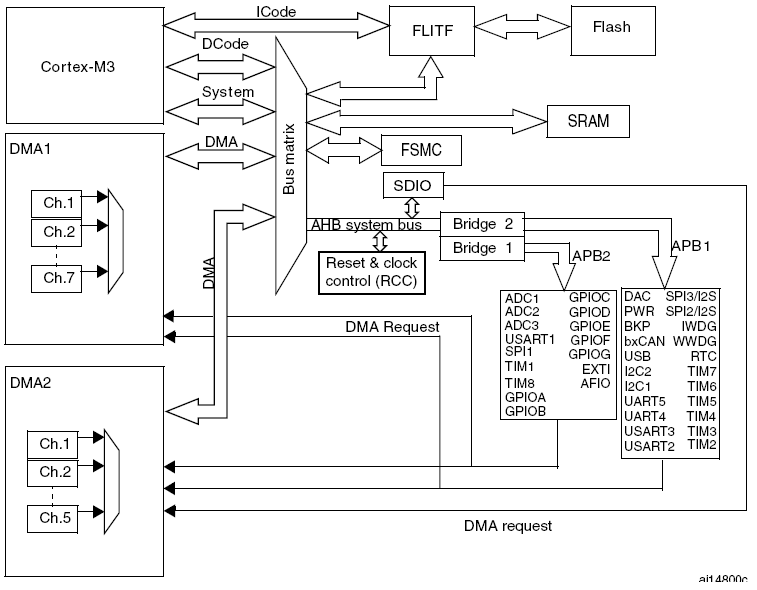
**System Architecture**



**AHB – A**dvanced **H**igh-Performance **B**us

**APB – A**dvanced **P**eripheral **B**us

**FSMC –** Flexible Static Memory Controller

**FLITF – FL**ash memory **I**n**T**er**F**ace

Main System consist of

* Four Masters
  + Cortex – M3 Core DCode Bus (D-bus)
  + Cortex – M3 Core System Bus(S-bus)
  + GP-DMA 1 (General-purpose DMA)
  + GP-DMA 2 (General-purpose DMA)
* Four Slaves
  + Internal SRAM
  + Internal Flash Memory
  + FSMC
  + AHB to APBx (APB1 or APB2) - Which connects to all peripherals.

**ICode Bus**

* Connects instruction bus of Cortex M3 core to Flash Memory instruction interface.

**DCode Bus**

* Connects Cortex-M3 DCode bus (Literal Load and Debug access) to the Flash Memory Data Interface.

**System Bus**

* Connects the system bus (peripherals bus) of the Cortex-M3 core to BusMatrix which manages the arbitration between the core and DMA.

**DMA Bus**

* Connects AHB Master Interface of DMA to BusMatrix which manages access of CPU DCode and DMA to SRAM, Flash Memory and Peripherals.

**BusMatrix**

* Manages the arbitration between the core system bus and DMA master bus.
* Manages access of CPU DCode and DMA to SRAM, Flash Memory and Peripherals.
* This arbitration is Round Robin based.
* Four Masters
  + CPU DCode
  + System Bus
  + DMA1 bus
  + DMA2 bus
* Four Slaves
  + FLITF
  + SRAM
  + FSMC
  + AHB2APB Bridge

**AHB/APB Bridge (APB)**

* Two AHB/APB Brides provide full synchronous connection between AHB and two APB Buses
* APB1 is limited to 36MHz
* APB2 can be @ 72Mhz
* AHB Peripherals are connected to system bus through BusMatrix to allow DMA access.
* After reset, all peripherals clocks are disabled, (except for SRAM and FLITF), so we must enable ***RCC\_AHBENR, RCC\_APB2ENR and RCC\_APB1ENR.***

**Memory organization**

* Program memory, Data memory, registers and I/O ports are organized within the same linear 4-GByte address space.
* uses Little Endian
* Divided into 8 main blocks, each of 512MB. (8 x 512MB = 4GB)

|  |  |  |
| --- | --- | --- |
| **Boundary address** | **Peripheral** | **Bus** |
| 0xA000 0000 - 0xA000 0FFF | FSMC | AHB |
| 0x5000 0000 - 0x5003 FFFF | USB OTG FS |
| 0x4003 0000 - 0x4FFF FFFF | Reserved |
| 0x4002 8000 - 0x4002 9FFF | Ethernet |
| 0x4002 3400 - 0x4002 7FFF | Reserved |
| 0x4002 3000 - 0x4002 33FF | CRC |
| 0x4002 2000 - 0x4002 23FF | Flash memory interface |
| 0x4002 1400 - 0x4002 1FFF | Reserved |
| 0x4002 1000 - 0x4002 13FF | Reset and clock control RCC |
| 0x4002 0800 - 0x4002 0FFF | Reserved |
| 0x4002 0400 - 0x4002 07FF | DMA2 |
| 0x4002 0000 - 0x4002 03FF | DMA1 |
| 0x4001 8400 - 0x4001 FFFF | Reserved |
| 0x4001 8000 - 0x4001 83FF | SDIO |
| 0x4001 5800 - 0x4001 7FFF | Reserved | APB2 |
| 0x4001 5400 - 0x4001 57FF | TIM11 timer |
| 0x4001 5000 - 0x4001 53FF | TIM10 timer |
| 0x4001 4C00 - 0x4001 4FFF | TIM9 timer |
| 0x4001 4000 - 0x4001 4BFF | Reserved |
| 0x4001 3C00 - 0x4001 3FFF | ADC3 |
| 0x4001 3800 - 0x4001 3BFF | USART1 |
| 0x4001 3400 - 0x4001 37FF | TIM8 timer |
| 0x4001 3000 - 0x4001 33FF | SPI1 |
| 0x4001 2C00 - 0x4001 2FFF | TIM1 timer |
| 0x4001 2800 - 0x4001 2BFF | ADC2 |
| 0x4001 2400 - 0x4001 27FF | ADC1 |
| 0x4001 2000 - 0x4001 23FF | GPIO Port G |
| 0x4001 1C00 - 0x4001 1FFF | GPIO Port F |
| 0x4001 1800 - 0x4001 1BFF | GPIO Port E |
| 0x4001 1400 - 0x4001 17FF | GPIO Port D |
| 0x4001 1000 - 0x4001 13FF | GPIO Port C |
| 0x4001 0C00 - 0x4001 0FFF | GPIO Port B |
| 0x4001 0800 - 0x4001 0BFF | GPIO Port A |
| 0x4001 0400 - 0x4001 07FF | EXTI |
| 0x4001 0000 - 0x4001 03FF | AFIO |
| 0x4000 7800 - 0x4000 FFFF | Reserved | APB1 |
| 0x4000 7400 - 0x4000 77FF | DAC |
| 0x4000 7000 - 0x4000 73FF | Power control PWR |
| 0x4000 6C00 - 0x4000 6FFF | Backup registers (BKP) |
| 0x4000 6400 - 0x4000 67FF | bxCAN1 |
| 0x4000 6800 - 0x4000 6BFF | bxCAN2 |
| 0x4000 6000 - 0x4000 63FF | Shared USB/CAN SRAM 512 bytes |
| 0x4000 5C00 - 0x4000 5FFF | USB device FS registers |
| 0x4000 5800 - 0x4000 5BFF | I2C2 |
| 0x4000 5400 - 0x4000 57FF | I2C1 |
| 0x4000 5000 - 0x4000 53FF | UART5 |
| 0x4000 4C00 - 0x4000 4FFF | UART4 |
| 0x4000 4800 - 0x4000 4BFF | USART3 |
| 0x4000 4400 - 0x4000 47FF | USART2 |
| 0x4000 4000 - 0x4000 43FF | Reserved |
| 0x4000 3C00 - 0x4000 3FFF | SPI3/I2S |
| 0x4000 3800 - 0x4000 3BFF | SPI2/I2S |
| 0x4000 3400 - 0x4000 37FF | Reserved |
| 0x4000 3000 - 0x4000 33FF | Independent watchdog (IWDG) |
| 0x4000 2C00 - 0x4000 2FFF | Window watchdog (WWDG) |
| 0x4000 2800 - 0x4000 2BFF | RTC |
| 0x4000 2400 - 0x4000 27FF | Reserved |
| 0x4000 2000 - 0x4000 23FF | TIM14 timer |
| 0x4000 1C00 - 0x4000 1FFF | TIM13 timer |
| 0x4000 1800 - 0x4000 1BFF | TIM12 timer |
| 0x4000 1400 - 0x4000 17FF | TIM7 timer |
| 0x4000 1000 - 0x4000 13FF | TIM6 timer |
| 0x4000 0C00 - 0x4000 0FFF | TIM5 timer |
| 0x4000 0800 - 0x4000 0BFF | TIM4 timer |
| 0x4000 0400 - 0x4000 07FF | TIM3 timer |
| 0x4000 0000 - 0x4000 03FF | TIM2 timer |

**Embedded SRAM**

* 96KBytes max
* starts @0x2000 0000

**Embedded Flash – see datasheet**

* 1 Mbytes max
  + bank 1: max up to 512Kbytes
  + bank 2: max up to 512 Kbytes
* Organized as Main Block, Information Block and Flash Memory Interface (FLITF) registers.
  + Main Memory Block

|  |  |  |  |
| --- | --- | --- | --- |
| **Memory Type** | **Divided into** | **#pages** | **Size per page** |
| Connectivity | 32KBx64bits | 128 pages | 2KB |
| XL-Density | 128KBx64bits | 512 pages | 2KB |
| High-Density | 64KBx64bits | 256 pages | 2KB |
| Medium-Density | 16KBx64bits | 128 pages | 1KB |
| Low-Density | 4KBx64bits | 32 pages | 1KB |

* + Information Block size

|  |  |
| --- | --- |
| **Memory Type** | **Size** |
| Connectivity | 2360x64bits |
| XL-Density | 770x64bits |
| High, Medium, Low - Density | 258x64bits |

* + Flash Memory Interface registers for
    - Read interface with prefetch buffer
    - Option Byte loader
    - Flash Program/ Erase operation
    - Read/ Write Protection

**Boot Configuration**

|  |  |  |
| --- | --- | --- |
| **BOOT[1:0]** | **Boot mode** | **Aliasing** |
| x0 | Main Flash Memory | Main Flash memory selected as boot space |
| 01 | System Memory | System Memory selected as boot space |
| 11 | Embedded SRAM | Embedded SRAM selected as boot space |

* The value of ***BOOT*** pins are latched on 4th rising cycle of SYSCLK after reset.
* User should set BOOT [1:0] pins after reset to select required boot mode.
* BOOT pins are also sampled after Standby Mode, so they must be kept in boot configuration in Standby Mode.
* Instruction area is 0x0000 0000 (accessed through ICode/DCode busses) – may be Main Flash, System or SRAM memory based on BOOT pins.
  + Boot from Main Flash Memory: the main Flash Memory is aliased in Boot memory space (0x0000 0000) and also accessible through its original memory space (0x800 0000).
  + Boot from System Memory: the system memory is aliased in Boot memory space (0x0000 0000) and also accessible through its original memory space (0x1FFF F0000).
  + Boot from Embedded SRAM: the SRAM is accessible only through 0x2000 0000. User relocate the vector table in SRAM using NVIC exception table and offset.
* Data area (SRAM) is 0x2000 0000 (accessed through system Bus).

**Embedded Boot Loader**

* Embedded Boot Loader is located in System Memory (programmed by ST during production).
  + Helps program the
    - Low, medium, high using USART1 interface
    - XL using USART1 or USART2 interface
    - Connectivity using USART1, USART2, CAN, USB OTG interface