

Esse é o décimo artigo da série escrita pelo engenheiro Ismael Lopes da Silva, exclusivamente para o site "www.embarcados.com.br". Nessa série focarei no Microcontrolador da STMicroelectronics, o MCU STM32F103C8T6, que é um ARM Cortex-M3. Os pré-requisitos para uma boa compreensão dos artigos é ter o domínio da Linguagem C Embedded e conceitos de eletrônica.

Mapa de Memória do Processador ARM Cortex-M3

O processador ARM Cortex-M3 possui um mapa de memória padrão, que fornece até 4 GB de memória endereçável, que é o máximo que um barramento de 32 bits pode endereçar, conforme ilustrado na figura 1:

Vendor-specific	0xFFFF.FFFF
Private peripheral bus - External	0xE010.0000 0xE00F.FFFF
Private peripheral bus - Internal	0xE004.0000 0xE003.FFFF
External device 1.0GB	0xE000.0000 0xDFFF.FFFF
External RAM 1.0GB	0xA000.0000 0x9FFF.FFFF
Peripheral 0.5GB	0x6000.0000 0x5FFF.FFFF
SRAM 0.5GB	0x4000.0000 0x3FFF.FFFF
Code 0.5GB	0x2000.0000 0x1FFF.FFFF
	0x0000.0000

Figura 1 – Mapa de memória ARM Cortex-M3

Nesse artigo, você observará que a memória de programa do nosso MCU STM32F103C8T6, é uma parte da memória de código (Code), que é a primeira seção do mapa, conforme ilustrado na parte inferior da figura 1. Essa faixa de código, de tamanho máximo de 0.5GB, é padrão para atender os mais variados modelos de microcontroladores, e seus recursos e tamanhos.

Mapa de Memória do Microcontrolador STM32F103C8T6

Esse artigo foi baseado na documentação oficial da STMicroelectronics, cujo link é <https://github.com/IsmaelLopesSilva/loT/blob/master/STM32/STM32F103C8T6/ST%20Docs/STM32F10xxx%20-%20PM0075%20Programming%20Manual%20Flash%20Memory.pdf>

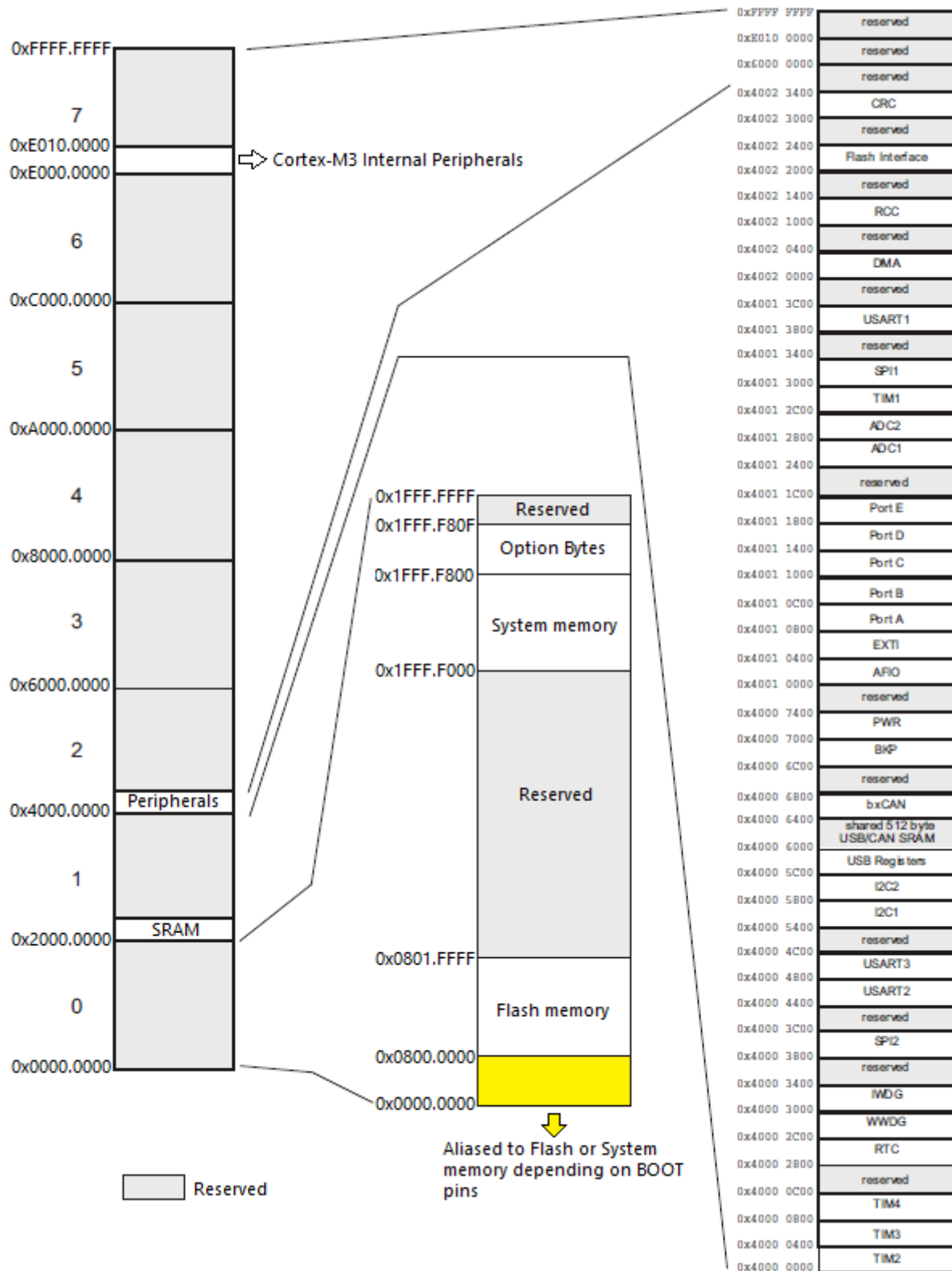


Figura 2 – Mapa de memória do MCU STM32F103C8T6

✓ Memória FLASH interna do Microcontrolador STM32F103C8T6

O MCU STM32F103C8T6, é um dispositivo com média densidade de memória FLASH, mais especificadamente 64 kbytes. A memória FLASH é usada para armazenar o código da aplicação, dados somente de leitura do programa e para armazenar a tabela de vetores (vector table). Cada deslocamento do ponteiro são quatro bytes ($4 \times 8 \text{ Bytes} = 32 \text{ bits}$), portanto, cada posição endereçável da memória FLASH contém 8 Bytes de comprimento.

Tamanho: 64 kbytes (65.536)

Endereço inicial: 0x0800.0000

Endereço final: 0x0800.FFFF

Tipo: Não volátil

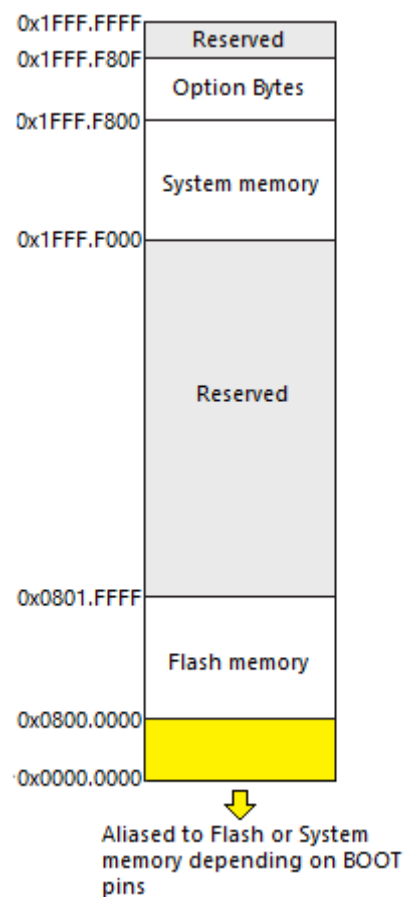


Figura 3 – Mapa da memória FLASH com média densidade 64k/128 kBytes

Segue o mapa das páginas da memória FLASH principal para o MCU STM32F103C8T6, de média densidade, com 64 kBytes.

Block	Name	Base addresses	Size (bytes)
Main memory	Page 0	0x0800 0000 - 0x0800 03FF	1 Kbyte
	Page 1	0x0800 0400 - 0x0800 07FF	1 Kbyte
	Page 2	0x0800 0800 - 0x0800 0BFF	1 Kbyte
	Page 3	0x0800 0C00 - 0x0800 0FFF	1 Kbyte
	Page 4	0x0800 1000 - 0x0800 13FF	1 Kbyte
	⋮	⋮	⋮
	Page 63	0x0800 FC00 - 0x0800 FFFF	1 Kbyte
Information block	System memory	0x1FFF F000 - 0x1FFF F7FF	2 Kbytes
	Option Bytes	0x1FFF F800 - 0x1FFF F80F	16
Flash memory interface registers	FLASH_ACR	0x4002 2000 - 0x4002 2003	4
	FLASH_KEYR	0x4002 2004 - 0x4002 2007	4
	FLASH_OPTKEYR	0x4002 2008 - 0x4002 200B	4
	FLASH_SR	0x4002 200C - 0x4002 200F	4
	FLASH_CR	0x4002 2010 - 0x4002 2013	4
	FLASH_AR	0x4002 2014 - 0x4002 2017	4
	Reserved	0x4002 2018 - 0x4002 201B	4
	FLASH_OBR	0x4002 201C - 0x4002 201F	4
	FLASH_WRP	0x4002 2020 - 0x4002 2023	4

Figura 4 – Organização da memória FLASH do STM32F103C8T6 em páginas

✓ **Memória SRAM interna do Microcontrolador STM32F103C8T6**

A SRAM é usada para armazenar dados globais e variáveis estáticas. Também usada para os propósitos de Stack e Heap (alocação dinâmica).

Tamanho: 20 kbytes (20.480)

Endereço inicial: 0x2000.0000

Endereço final: 0x2000.4FFF

Tipo: Volátil

Observação: Código também pode ser executado na memória SRAM.

✓ **Memória do Sistema (System memory) do Microcontrolador STM32F103C8T6**

Função: O Bootloader é armazenado nessa memória

Tamanho: 2 kBytes (2.048)

Endereço inicial: 0x1FFF.F000

Endereço final: 0x1FFF.F7FF

Tipo: Somente de leitura

Por padrão, o MCU não executa nenhum código de programa na memória do Sistema, mas, pode ser configurado para fazer o Boot ou executar o Bootloader. Ver figura 4.

✓ **Memória Options Bytes do Microcontrolador STM32F103C8T6**

Essa memória são bytes de opção, configuráveis pelo usuário, que podem controlar recursos do MCU. Essas opções fornecem aos usuários uma maneira conveniente de alterar as configurações de seu dispositivo. Ver figura 4.

Tamanho: 16 Bytes

Endereço inicial: 0x1FFF.F800

Endereço final: 0x1FFF.F80F

Código e Dados de Programa

O que exatamente é um programa? Um programa é uma coleção de código e dados. Um código de programa opera os dados. Onde armazenamos o código? O código, que são as instruções, são armazenados na memória de programa, que em nosso caso, é a memória FLASH, interna do microcontrolador. É muito importante entender os diferentes tipos de dados do programa.

Onde armazenamos os dados do programa? Os dados podem ser armazenados na memória FLASH ou na memória SRAM, dependendo da natureza dos dados. Se for um dado constante, não faz sentido armazená-lo na memória SRAM.

Essa é a razão pela qual podemos armazenar dados constante na memória FLASH. A memória FLASH pode ser considerada como uma memória somente de leitura (ROM). Podemos modificar o conteúdo da memória FLASH, mas, não em tempo de execução. Existem alguns procedimentos para alterar o conteúdo da memória FLASH.

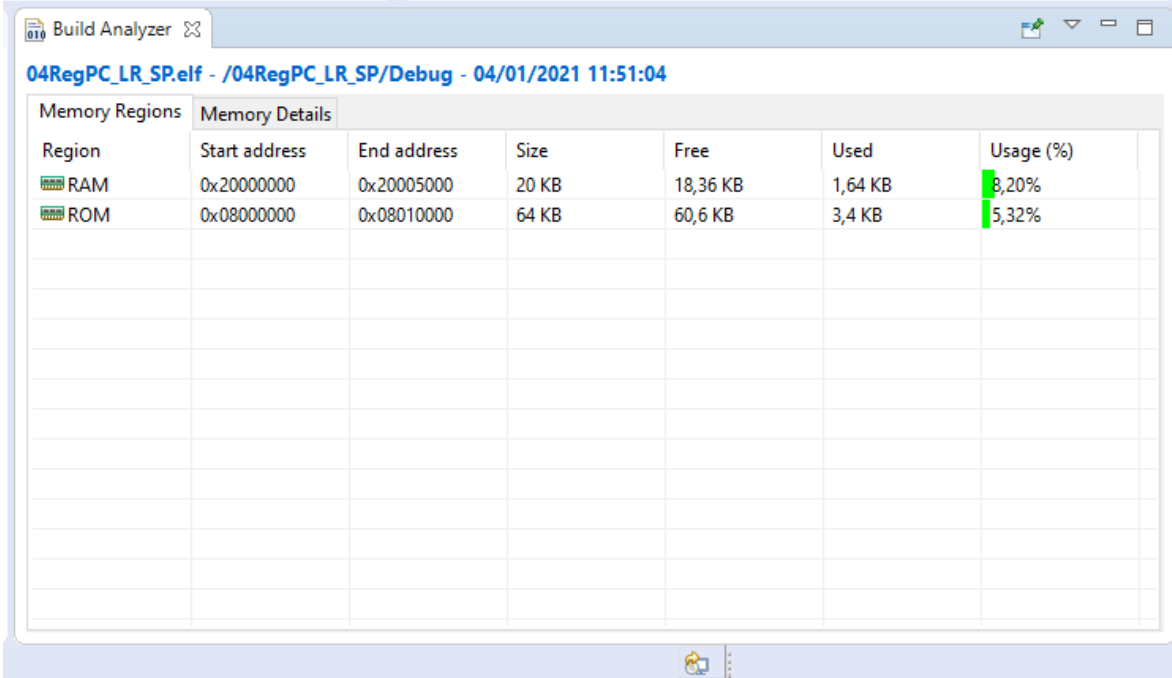
É por isso que podemos tratar a FLASH como memória somente de leitura (ROM) e a SRAM como memória de leitura e escrita (RAM). E quanto às variáveis do programa? As variáveis do programa são chamadas de dados de leitura e escrita, porque o valor das variáveis pode ser alterado em qualquer ponto durante a execução do programa.

Analisar as Memórias FLASH e SRAM no STM32CubeIDE

Vamos reutilizar o programa do oitavo artigo, somente para analisarmos as memórias FLASH e SRAM dessa aplicação. Usando o STM32CubeIDE, no mesmo workspace que criamos os artigos

anteriores, então, na janela "Projetc Explorer", clique com o botão direito do mouse sobre o projeto "04RegPC_LR_SP", depois dê um "Clean Project" e "Build Project".

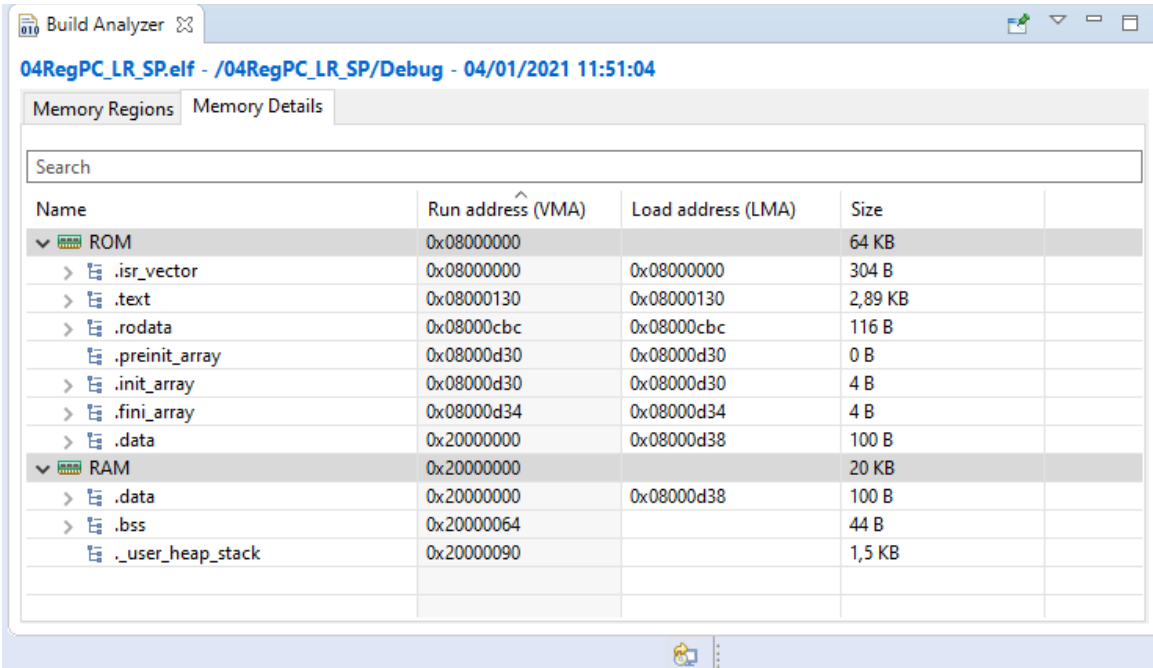
Disponibilize a visualização da janela "Build Analyzer", conforme ilustrado na figura 5. Note que a aba "Memory Regions" está selecionada e podemos ver os limites (tamanhos) da memória FLASH (ROM) e SRAM (RAM).



Region	Start address	End address	Size	Free	Used	Usage (%)
RAM	0x20000000	0x20005000	20 KB	18,36 KB	1,64 KB	8,20%
ROM	0x08000000	0x08010000	64 KB	60,6 KB	3,4 KB	5,32%

Figura 5 – Janela Build Analyzer

Agora selecione a aba "Memory Details". Podemos ver na figura 6 os detalhes das seções e seus endereços correspondentes para cada tipo de memória (ROM e RAM).



Name	Run address (VMA)	Load address (LMA)	Size
ROM	0x08000000		64 KB
> .isr_vector	0x08000000	0x08000000	304 B
> .text	0x08000130	0x08000130	2,89 KB
> .rodata	0x08000cbc	0x08000cbc	116 B
.preinit_array	0x08000d30	0x08000d30	0 B
> .init_array	0x08000d30	0x08000d30	4 B
> .fini_array	0x08000d34	0x08000d34	4 B
> .data	0x20000000	0x08000d38	100 B
RAM	0x20000000		20 KB
> .data	0x20000000	0x08000d38	100 B
> .bss	0x20000064		44 B
.user_heap_stack	0x20000090		1,5 KB

Figura 6 – Detalhes da memória FLASH e SRAM