

Memórias - Breve Introdução

1 Introdução

Projetos envolvendo FPGAs em geral utilizam três tipos de memórias:

- **Banco de registradores:** Acesso instantâneo e paralelo, extremamente caro (em termos de recursos lógicos) se utilizado em excesso;
- **Memória *on-chip*:** Blocos de memória existentes dentro do FPGA, oferece acesso praticamente instantâneo, mais barato que registradores, porém em quantidade limitada (em torno de 1 MB);
- **Memória *off-chip*:** Memórias foras do FPGA. Podem ser acessadas através de interfaces. Acesso mais lento, porém com maior capacidade (exemplo: memórias SSRAM, SDRAM DDR, etc.).

2 Memória on-chip

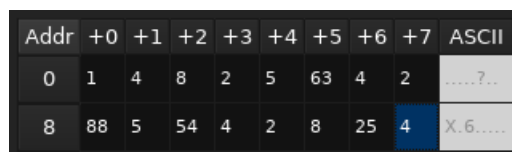
Memórias *on-chip* são compostas de blocos de memória já existentes dentro do FPGA, oferece acesso praticamente instantâneo (em geral o dado é disponibilizado após um ciclo de *clock*). São compostas praticamente de quatro portas:

- **clock;**
- **wren:** Habilitador de escrita. Se 1, o dado em *data* é escrito no endereço *address*. Se 0, o dado no endereço *address* é exibido em *q*;
- **data:** Dado a ser escrito, se **wren** for 1;
- **address:** Endereço de leitura ou escrita;
- **q:** Dado do endereço *address* após a requisição de uma leitura.

Mais portas são possíveis (endereçador de byte, 2 portas paralelas de escrita ou leitura, etc.). Focaremos aqui apenas no básico.

3 Utilizando memória on-chip no Quartus

1. Crie ou abra o projeto onde a memória será utilizada;
2. Para definir os dados iniciais de sua memória, um arquivo de inicialização de memória deve ser criado:
 - Vá em **File > New...**
 - Selecione **Hexadecimal (Intel-Format) File** ou **Memory Initialization File**;
 - Defina o número de palavras da sua memória e quantos bits terá cada palavra;
 - Preencha as palavras (por padrão, o preenchimento é na base 10). Cada campo é uma palavra da memória;

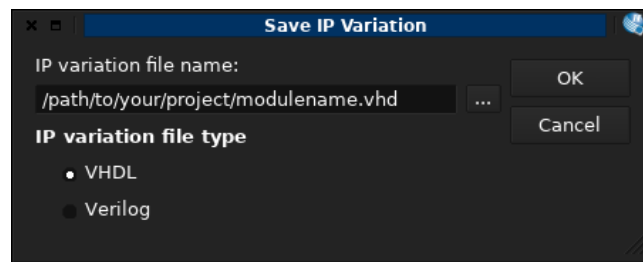


Addr	+0	+1	+2	+3	+4	+5	+6	+7	ASCII
0	1	4	8	2	5	63	4	2?..
8	88	5	54	4	2	8	25	4	X.6.....

- Salve o arquivo;
3. Abra a janela de criação de memória:
 - (a) Quartus anterior ao 14.0:
 - i. Vá em **Tools > MegaWizard Plug-In Manager**;
 - ii. Selecione **Create a new custom megafunction variation**;
 - iii. Na lista, buscar por **RAM: 1-PORT**;
 - (b) Quartus superior ou igual ao 14.0:
 - i. Vá em **Tools > IP Catalog**;

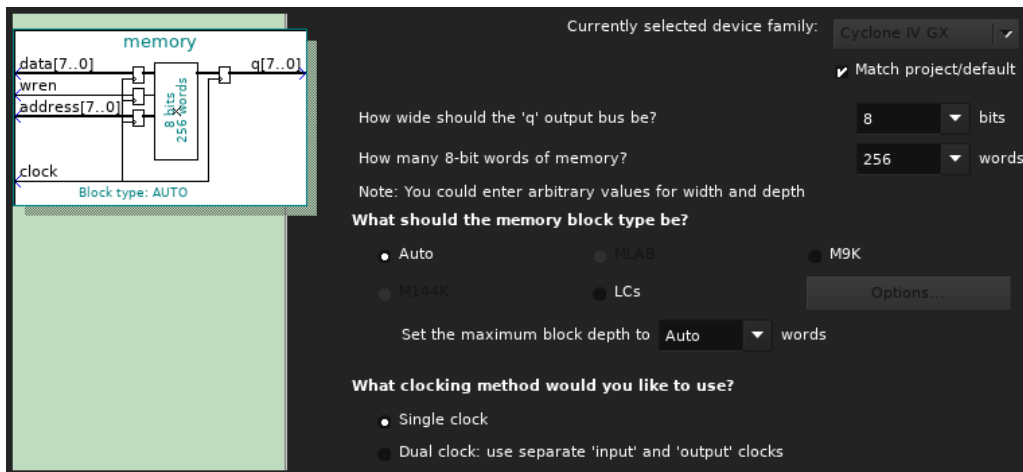
ii. No IP Catalog, buscar e selecionar RAM: 1-PORT;

4. Dê um nome para o seu módulo, escolha onde será salvo e em qual linguagem será criado:



5. Configure as características da sua memória:

- **How wide should the 'q' output bus be?** Quantidade de bits para cada palavra na memória;
- **How many X-bit words of memory?** Quantas palavras devem existir na memória;
- Deixar o restante com os valores padrão;



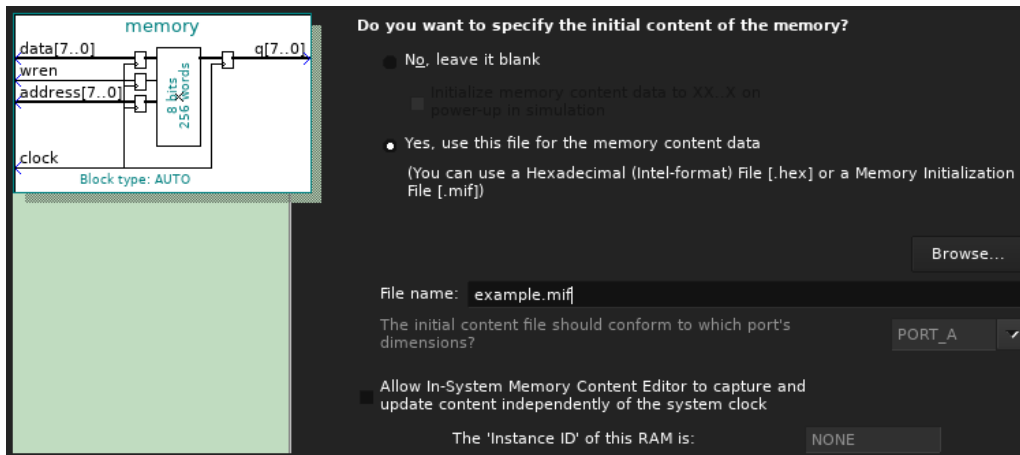
6. Pressione Next;

7. Para **Which ports should be registered?**, desativar 'q' output port;

8. Pressiona Next duas vezes até chegar na página Mem Init;

9. Se quiser inicializar a memória com os dados preenchidos no arquivo de inicialização do passo 2:

- (a) **Do you want to specify the initial content of the memory?** Selecionar **Yes, use this file for the memory content data**;
- (b) Em **File name**, colocar o nome do arquivo de inicialização gerado anteriormente;



10. Clicar em **Finish**. Uma lista dos arquivos a serem criados será apresentado. Selecione o arquivo **memory.bsf** (o nome será diferente dependendo do nome dado no passo 4). Isto gerará o bloco **bsf** a ser utilizado no projeto;

11. Clicar em **Finish**, clicar em **Yes**.

Nota: Os arquivos de inicialização também podem ser preenchidos manualmente:

- **Para o formato Hexadecimal (Intel Format), ver:**
http://quartushelp.altera.com/14.1/mergedProjects/reference/glossary/def_hexfile.htm
- **Para o formato Memory Initialization File, ver:**
http://quartushelp.altera.com/15.0/mergedProjects/reference/glossary/def_mif.htm

4 Exemplo de uso

Arquivo de inicialização utilizado:

- Modo texto:

Listing 1: memory.mif

```
-- Quartus II generated Memory Initialization File (.mif)

WIDTH=4;
DEPTH=32;

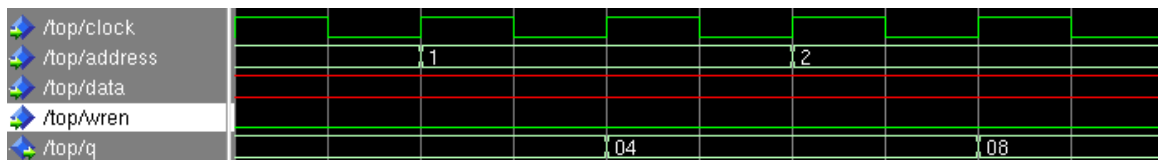
ADDRESS_RADIX=UNS;
DATA_RADIX=UNS;

CONTENT BEGIN
  [0..27] : 0;
  28 : 8;
  29 : 9;
  30 : 4;
  31 : 5;
END
```

- Modo gráfico:

Addr	+0	+1	+2	+3	+4	+5	+6	+7	ASCII
0	1	4	8	2	5	63	4	2?..
8	88	5	54	4	2	8	25	4	X.6.....

4.1 Leitura



4.2 Escrita

