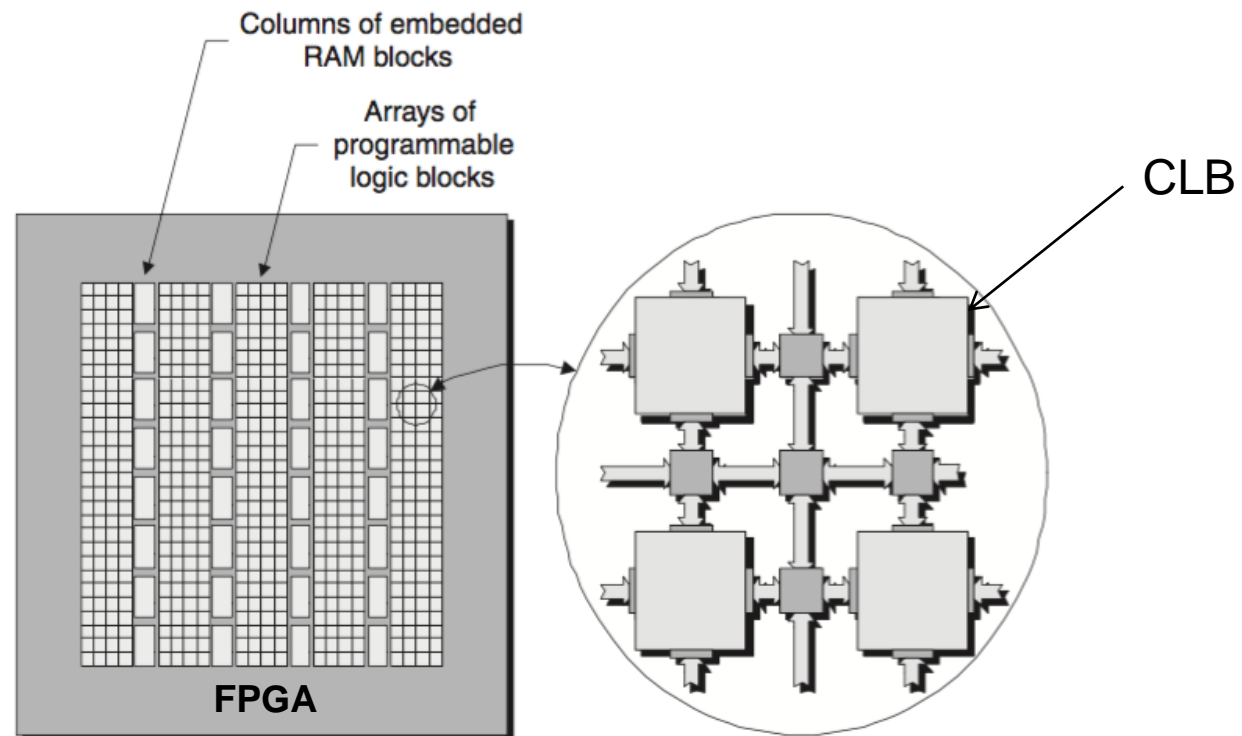


Trabalho 1 – parte 2

- A tarefa deste trabalho será fazer a prototipação em FPGA do processador R8
- Inicialmente será necessário trocar a memória a fim de que ela seja implementada utilizando as Block RAMs do FPGA



Trabalho 1 – parte 2

- ❑ A descrição da memória a ser utilizada está no *moodle*
 - ❑ Memory.vhd
 - ❑ Devido à maneira como foi descrita esta memória, o sintetizador (XST) fará a inferência de block RAMs
 - ❑ O número de posições da memória deve ser 2^{15} (32768) e a largura dos dados 16 bits

```
RAM: entity work.Memory
      generic map (
        ADDR_WIDTH  => 15,      -- 32768 posições (2^15)
        DATA_WIDTH => 16,      -- palavras de 2 Bytes
```

Capacidade de armazenamento da memória: 64KB

Block RAM disponível no FPGA Spartan-6 LX16: ~70KB (72000B)

Trabalho 1 – parte 2

- O arquivo de imagem desta memória deve conter o conteúdo de todas as posições da memória.
Uma palavra por linha

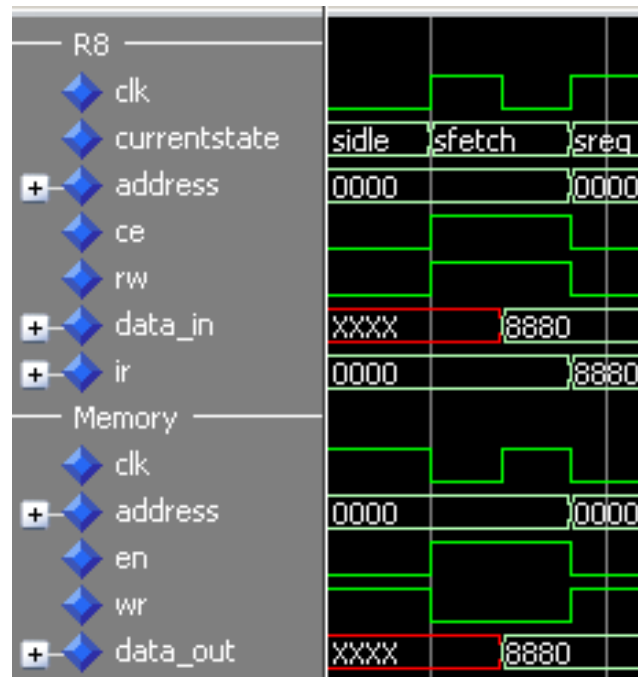
8880	}	32768 linhas
7800		
8100		
7102		
72F0		
...		

Trabalho 1 – parte 2

- ❑ Para gerar a nova imagem, uma nova versão do simulador estrá disponível no *moodle*
- ❑ R8_sim.jar
- ❑ Além de gerar a imagem no formato até então utilizada, será gerada também automaticamente uma imagem com sufixo _BRAM
- Exemplo: Todas_Instrucoes_R8.asm
 - Todas_Instrucoes_R8.txt
 - Todas_Instrucoes_R8_BRAM.txt

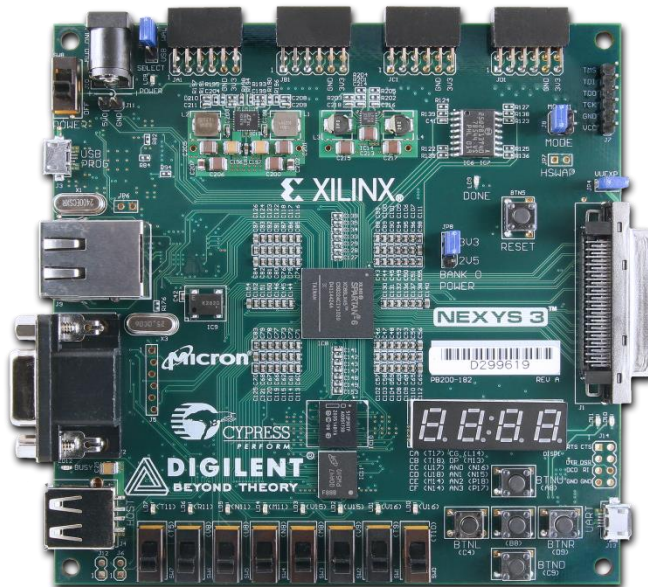
Trabalho 1 – parte 2

- ❑ Visto que a nova memória tem a leitura síncrona, será necessário que ela opere nas bordas de descida do *clock* do processador para fins de sincronia
 - ❑ Basta setar o *clock* da memória como o complemento do *clock* do processador



Trabalho 1 – parte 2

- ❑ A prototipação será feita utilizando a placa Nexys 3 (NUPEDEE)
 - ❑ Nesta placa o FPGA opera a 100MHz, no entanto nosso projeto deve operar a 50 MHz
 - ❑ Para dividir a frequência da placa utilizaremos um dos DCMs (*Digital Clock Manager*) do FPGA



Trabalho 1 – parte 2

- ❑ Para utilizar o DCM do FPGA, deve-se adicionar ao projeto componente *ClockManager*
 - ❑ ClockManager.vhd (moodle)

```
entity ClockManager is
  port (
    -- Board clock (100MHz)
    clk_in      : in    std_logic;

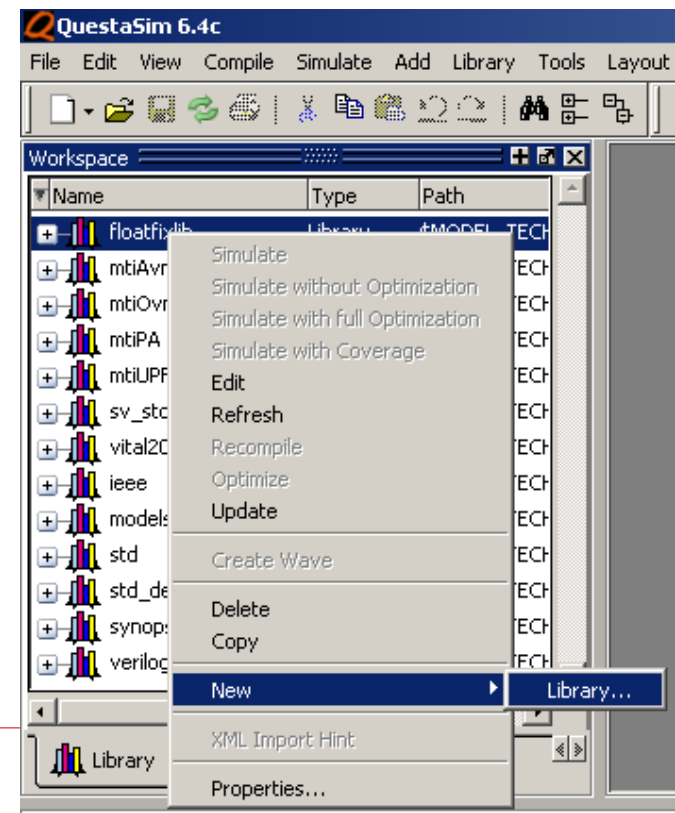
    -- Project clock
    clk_out     : out   std_logic;
    clk_out_n   : out   std_logic;
    clk_div2    : out   std_logic
  );
end ClockManager;
```

Não utilizaremos estes

clk_in / 2 (processador)

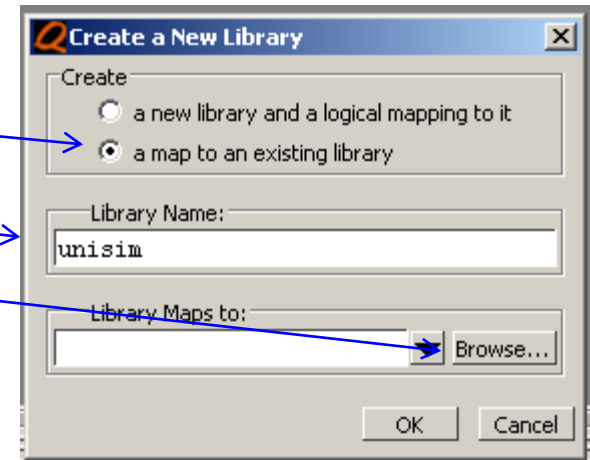
Trabalho 1 – parte 2

- ❑ Para simular componentes internos do FPGA (e.g. DCM) será necessário utilizar a biblioteca UNISIM
 - ❑ Moodle: unisim.zip (descompactar)
- Para adicionar biblioteca no Modelsim
 - ❑ Abrir o menu de contexto do painel *Library* e selecionar *New → Library*



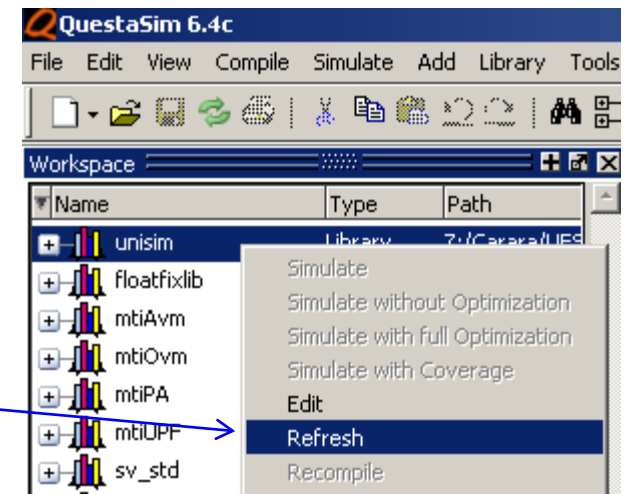
Trabalho 1 – parte 2

- ❑ Para simular componentes internos do FPGA (e.g. DCM) será necessário utilizar a biblioteca UNISIM
 - ❑ Moodle: unisim.zip (descompactar)
- Para adicionar biblioteca no Modelsim
 - ❑ Abrir o menu de contexto do painel *Library* e seleccionar *New* → *Library*
 - ❑ Marcar a opção "*a map to an existing library*",
Adicionar o nome da *library*
Seleccionar o directório da *library*



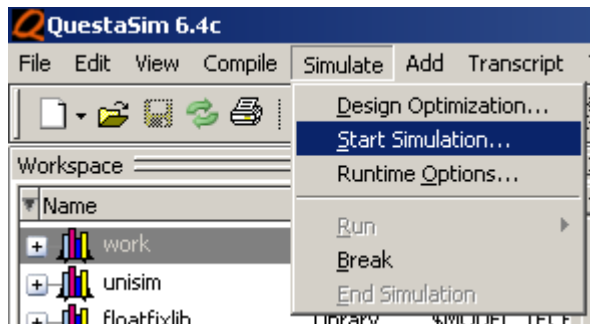
Trabalho 1 – parte 2

- ❑ Para simular componentes internos do FPGA (e.g. DCM) será necessário utilizar a biblioteca UNISIM
 - ❑ Moodle: unisim.zip (descompactar)
- Para adicionar biblioteca no Modelsim
 - ❑ Abrir o menu de contexto do painel *Library* e selecionar *New → Library*
 - ❑ Marcar a opção *"a map to an existing library"*, Adicionar o nome da *library*
Selecionar o diretório da *library*
 - ❑ Compilar (*Refresh*)
 - É necessário compilar apenas uma vez

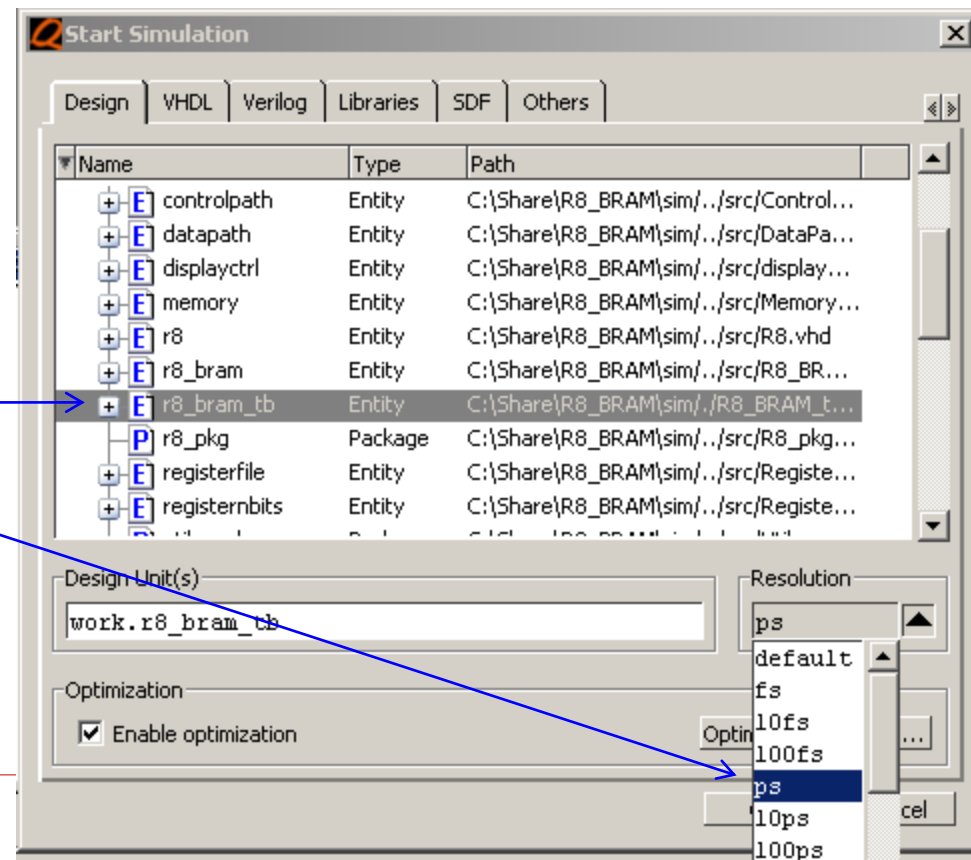


Trabalho 1 – parte 2

- ❑ Para simular projetos que utilizam o DCM, é necessário alterar a resolução do simulador Modelsim **antes de iniciar a simulação**



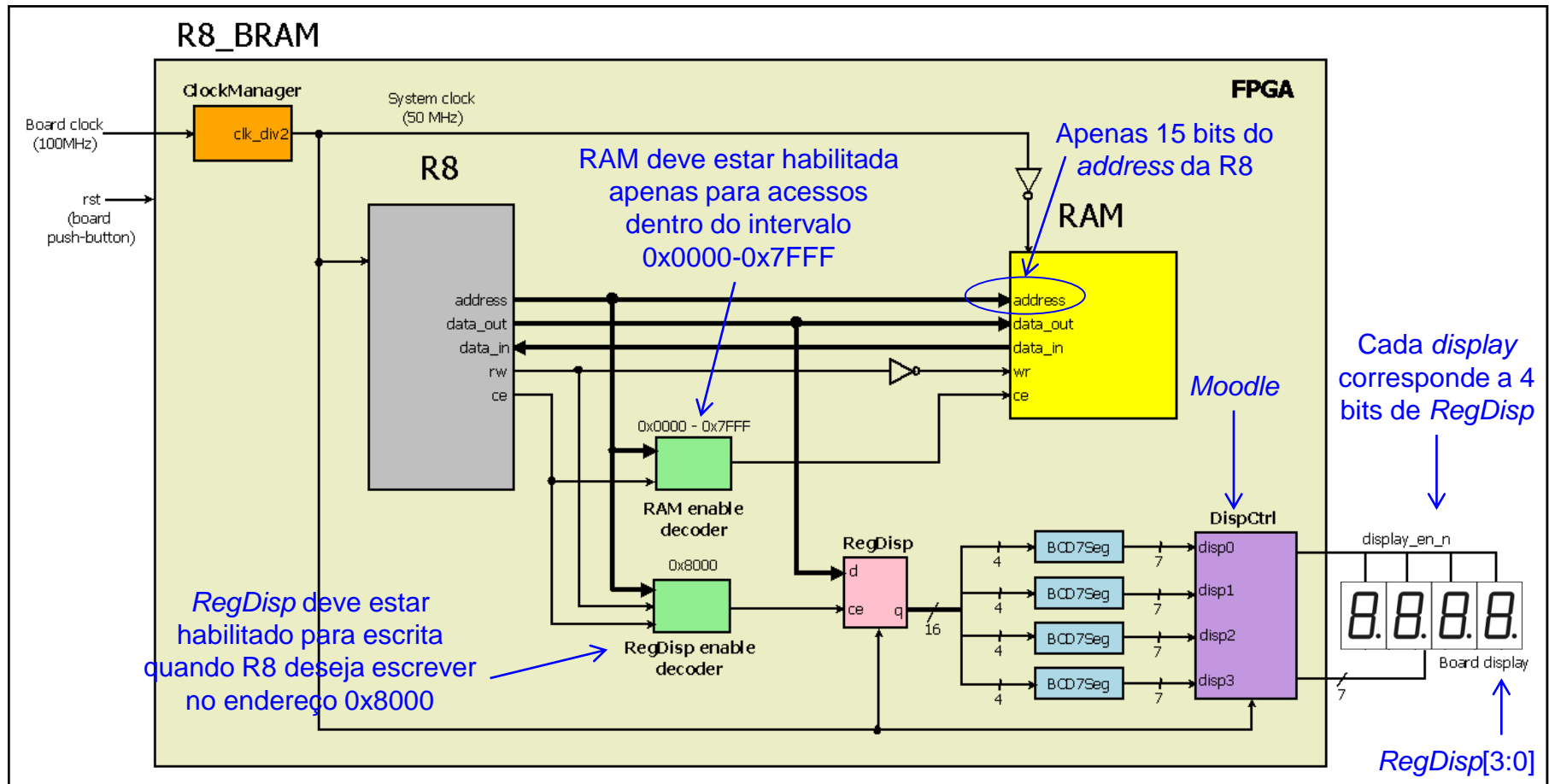
Selecionar o *test bench*
Setar a resolução para *ps*



Pode-se setar pela linha de comando
`vsim -t ps work.r8_bram_tb`

Trabalho 1 – parte 2

- ❑ Para que seja possível verificar o funcionamento da prototipação utilizaremos os *displays* da placa

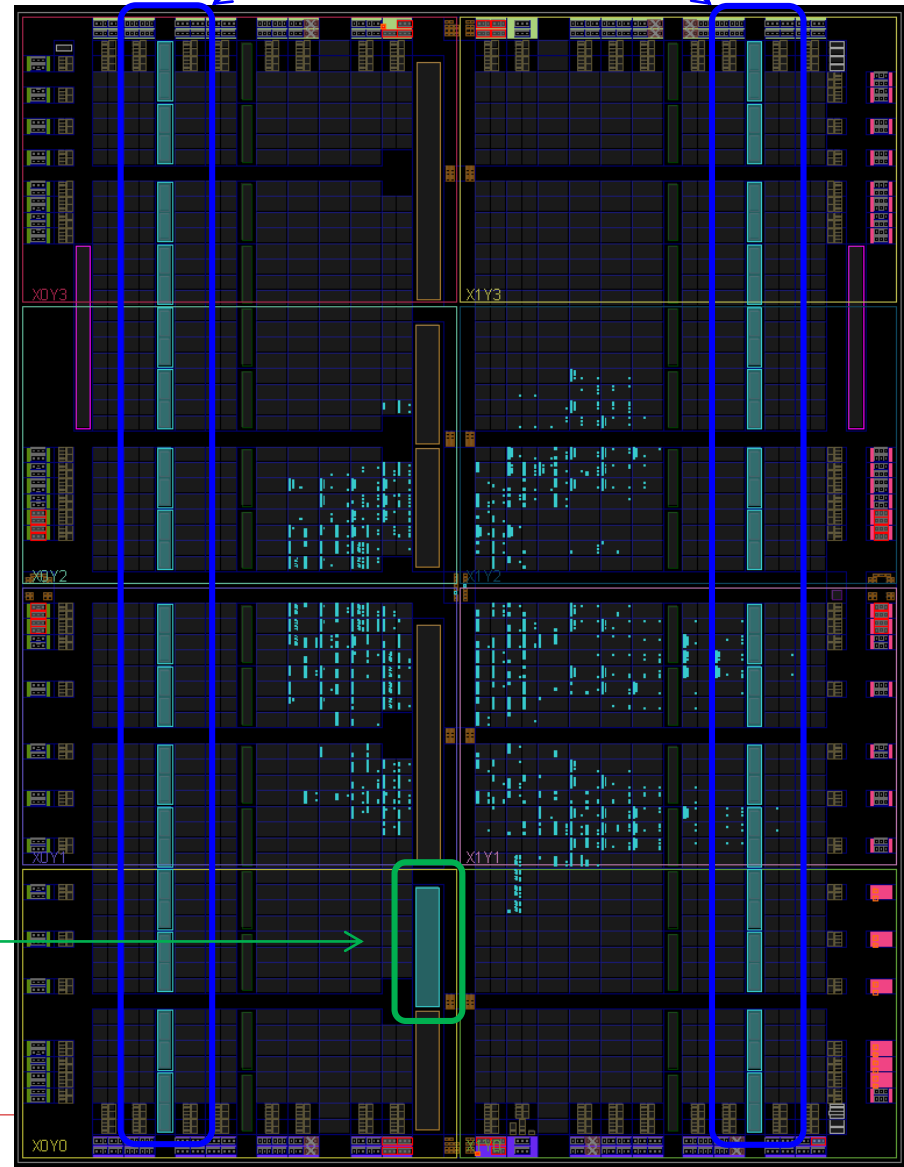


Trabalho 1 – parte 2

Block RAMs

- ❑ *Implemented design*
- ❑ Spartan 6 (LX16)

DCM



Trabalho 1 – parte 2

□ Aplicação

- Implementar um contador com incremento de 1 segundo
- Considerar o tempo de execução das instruções e a frequência a fim de atingir um incremento **o mais próximo possível** de 1 segundo

□ Sequência sugerida para o desenvolvimento

1. Trocar memória
 2. Adicionar o DCM
 3. Adicionar a parte relativa à interface com os *displays*
 4. Prototipar
 - O arquivo com a imagem da memória deve estar no mesmo diretório do arquivo Memory.vhd
 - **Atenção aos *warnings*!**
-

Trabalho 1 – parte 2

□ Grupos de 2 alunos

- Apresentação da descrição do funcionando será **IMPRETERIVELMENTE** dia 31/3
- A nota do trabalho dará **ENORME ÊNFASE** à execução correta da simulação
 - Se a simulação não funciona, não há o que apresentar
- A apresentação será oral, teórico-prática, frente ao computador, onde o grupo deverá explicar ao professor o projeto, a simulação e a implementação
- Em relação às duvidas, sejam pontuais