

Experiência Nº 6

1 – Usando VHDL, implemente um detector de sequência de bits (10110001) de duas maneiras: a) usando um registrador de deslocamento; b) usando uma máquina de estado. Mostre as simulações no relatório e demonstre o funcionamento na placa usando um botão para entrada 0, outro para entrada 1. O pressionamento desses botões deve causar a amostragem do sinal de entrada e funcionar como sinal de load ou clock. Um LED deverá sinalizar quando a sequência correta tiver sido entrada. [RSD]¹.

2 – Usando VHDL, implemente um detector de sequência de palavras de 4 bits em Hexa (CADA1) usando 4 registradores de deslocamento e a solução do exercício 1. Mostrar simulações no relatório e demonstrar, agora com as entradas em 4 dip switches amostradas por um botão. Os valores entrados devem ser mostrados em um display de 7-seg. Um LED deverá sinalizar quando a sequência correta tiver sido entrada. [RSD]

3 – Projetar, em VHDL, uma memória RAM de 64 B com operação de RD e WR usando **mega funções** disponíveis para a Flex 10K70. [RD]

3.a - Salvar o componente para usar no item (b). Esta RAM deverá ter a seguinte pinagem:

- Um pino E = Chip enable
- Um pino W = Write enable
- Um pino G = Output enable (controle de tri-state)
- As linhas de endereço
- Pinos de entradas e saídas de dados.

3.b - Projetar um sistema de memória RAM com 4 x (64 B), usando 4 chips do experimento (3a). Levando em conta que o barramento de endereços tem 10 bits, projete o decodificador de endereços, considerando que os endereços devem ser mapeados para a região inferior do espaço de endereçamento. Demonstre o funcionamento do sistema de memória na placa. [RSD]

¹ R= relatório; S= simulação; D = demonstração

4 - Projetar usando VHDL uma via de dados como descrito na figura, com 8 registradores de 8 bits cada. Na implementação, use um dip switch como OPCODE, que definirá a operação a ser realizada (como descrito na figura). Outro dip switch como sinal de ENTRADA. A saída do barramento deverá ser mostrada no display de 7 segmentos. Usar um clock manual através de um push button e sinal de reset global. O barramento deverá ser tri-state e pode ser implementado, por exemplo, com o buffer BTRI. [RSD]

