

Computação Reconfigurável

Aula prática 8

Problemas para resolver

Exercício 8.1. Usar *IP Integrator* para criar um projeto que é baseado no divisor de frequência que permite gerar frequências ~1 Hz (mostra em led0), 2 Hz (mostra em led1) e 4 Hz (mostra em led2).

Exercício 8.2. Usar *IP Integrator* para criar um projeto que permite converter códigos binários (sw(15 downto 0)) para valores decimais. Mostrar valores decimais em displays de segmentos.

Exercício 8.3. Usar *IP Integrator* para criar um projeto que permite ler palavras de ROM (usar um ficheiro COE para programar a ROM: 8 palavras de 16 bits cada) e mostrar em displays em decimal (quando pressiona o botão btnL) e hexadecimal (quando pressiona o botão btnR). Escolher uma palavra por interruptores sw(2 downto 0).

Exercício 8.4. Usar *IP Integrator* para criar um projeto que permite ler palavras de ROM (usar um ficheiro COE para programar a ROM: 8 palavras de 16 bits cada) e encontrar o peso de *Hamming* de todas as palavras. Mostrar o peso em displays em decimal (4 displays esquerdos) e hexadecimal (4 displays direitos).

Exercício 8.5. Usar *IP Integrator* para criar um projeto que permite ler palavras de ROM (usar um ficheiro COE para programar a ROM: 16 palavras de 64 bits cada), encontrar o peso de *Hamming* de cada palavra e escrever na memória RAM (*dual port*). Usar a segunda porta da memória RAM para mostrar resultados em displays. Mostrar o peso em displays em decimal (4 displays esquerdos) e hexadecimal (4 displays direitos). Escolher uma palavra da memória RAM por sw(3 downto 0).

Exercício 8.6. Usar *IP Integrator* para criar um projeto que permite ler palavras de ROM (usar um ficheiro COE para programar a ROM: 32 palavras de 16 bits cada), encontrar o valor máximo e mínimo das palavras e mostrar os resultados em displays em hexadecimal (4 displays esquerdos – o máximo, 4 displays direitos - o mínimo).

Exercício 8.7. Usar *IP Integrator* para criar um projeto que permite ler palavras de ROM (usar um ficheiro COE para programar a ROM: ~~256~~³² palavras de 16 bits cada), ordenar dados e escrever na memória RAM (*dual port*). Mostrar os resultados em displays em decimal (quando pressiona o botão btnL) e hexadecimal (quando pressiona o botão btnR). Escolher uma palavra da memória RAM por sw(7 downto 0).