

Computação Reconfigurável

Aula prática 3

Problemas para resolver

Exercício 3.1. Descrever um circuito que permite encontrar o número máximo de uns consecutivos num vetor binário `sw(15 downto 0)`. Usar máquinas de estados finitos. Mostrar o resultado em displays de segmentos em decimal e hexadecimal. Usar *IP block integrator*.

Exercício 3.2. Mostrar os resultados do divisor máximo comum em displays de segmentos em decimal. Entrar dados (intervalo 0-255) utilizando miniteclado. Usar *IP block integrator*.

Exercício 3.3. Descrever um circuito que permite encontrar o número máximo de uns consecutivos num vetor binário de 1000 bits. Usar máquinas de estados finitos. Gerar vetor binário aleatoriamente. Mostrar o resultado em displays de segmentos em decimal. Usar *IP block integrator*.

Exercício 3.4. Gerar um vetor binário de 1024 bits utilizando JAVA ou C/C++ num ficheiro COE. Contar o número de uns no vetor em programa JAVA ou C/C++. Preencher memória distribuída utilizando este ficheiro. Descrever um circuito que permite encontrar o número de uns no vetor. Usar máquinas de estados finitos. Mostrar o resultado em displays de segmentos em decimal. Comparar os resultados em software e hardware. Usar *IP block integrator*.

Exercício 3.5. Descrever um circuito para uma calculadora que execute operações +, -, *, / sobre operandos `sw(15 downto 8)` e `sw(15 downto 8)`. Cada operação deve ser executada utilizando memórias do tipo ROM e cálculos baseados em tabelas. Preencher memórias propriamente utilizando programas em JAVA ou C/C++. Ver aula teórica 5 para detalhes. Usar *IP block integrator*.