

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

INF01058 - CIRCUITOS DIGITAIS 2022/1

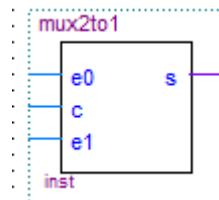
Professor: Renato Perez Ribas
Aluno: Matheus Souza da Silveira
Nº Cartão: 316271

Relatório Projeto Neander

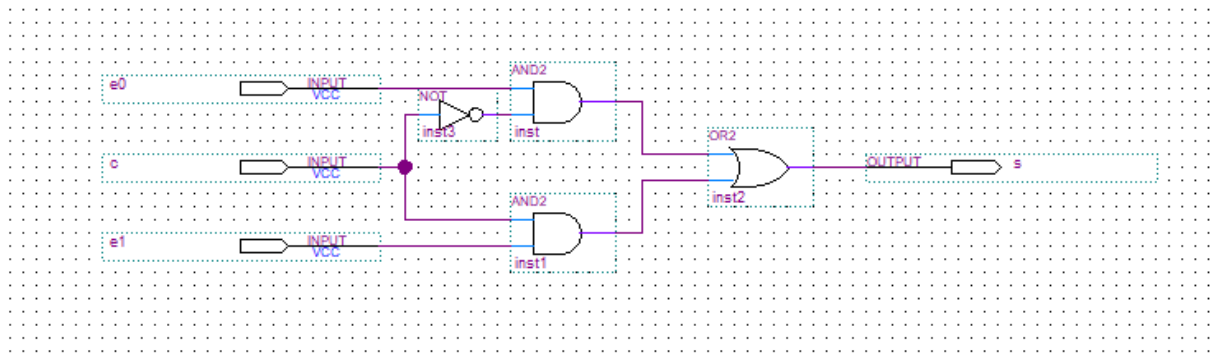
Nome: mux2to1

Descrição: Selecionar uma entre duas entradas (e0 e e1) do circuito como saída (s) com base num sinal de entrada (c).

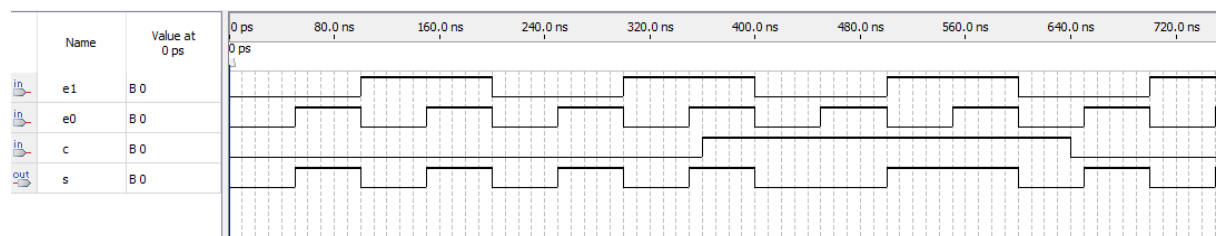
Símbolo:



Esquemático:



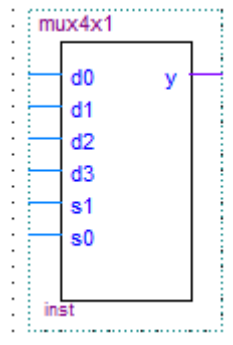
Simulação:



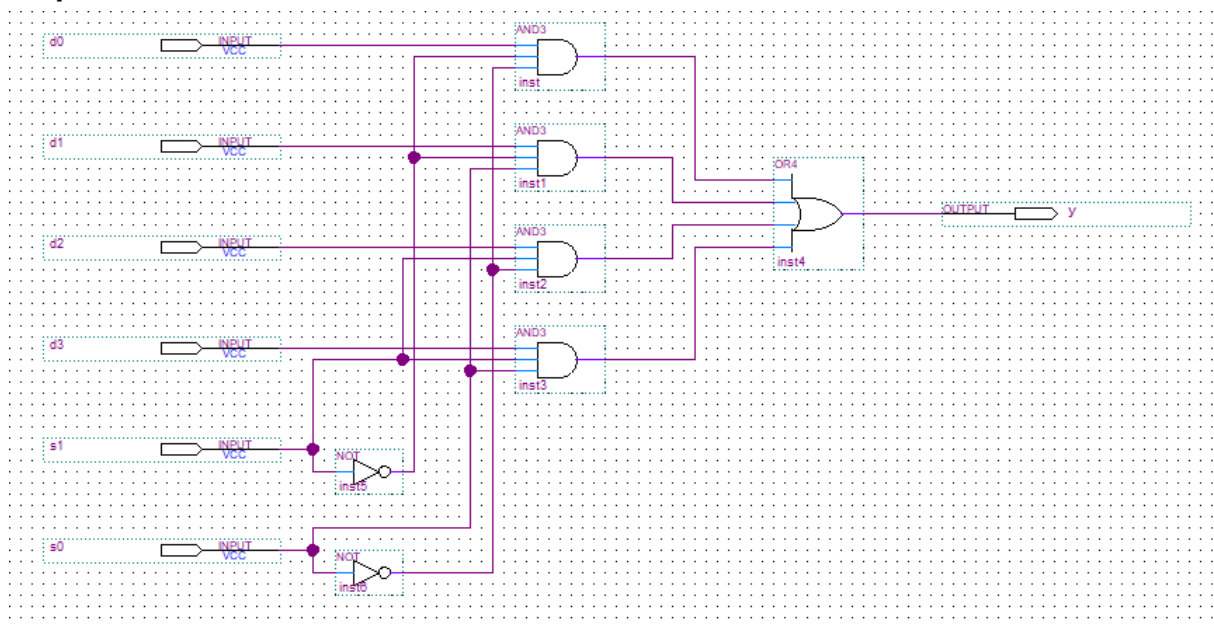
Nome: mux4x1

Descrição: Selecionar uma entre quatro entradas (d0, d1, d2 e d3) do circuito como saída (y) com base em dois sinais de entrada (s0 e s1).

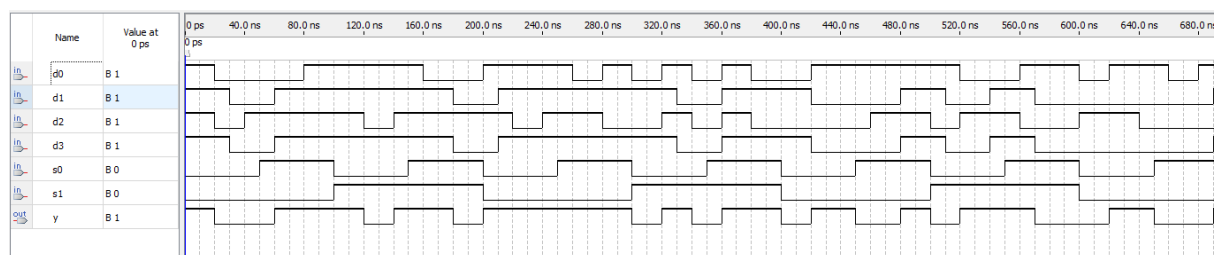
Símbolo:



Esquemático:



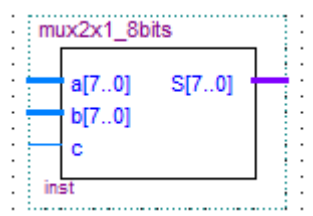
Simulação:



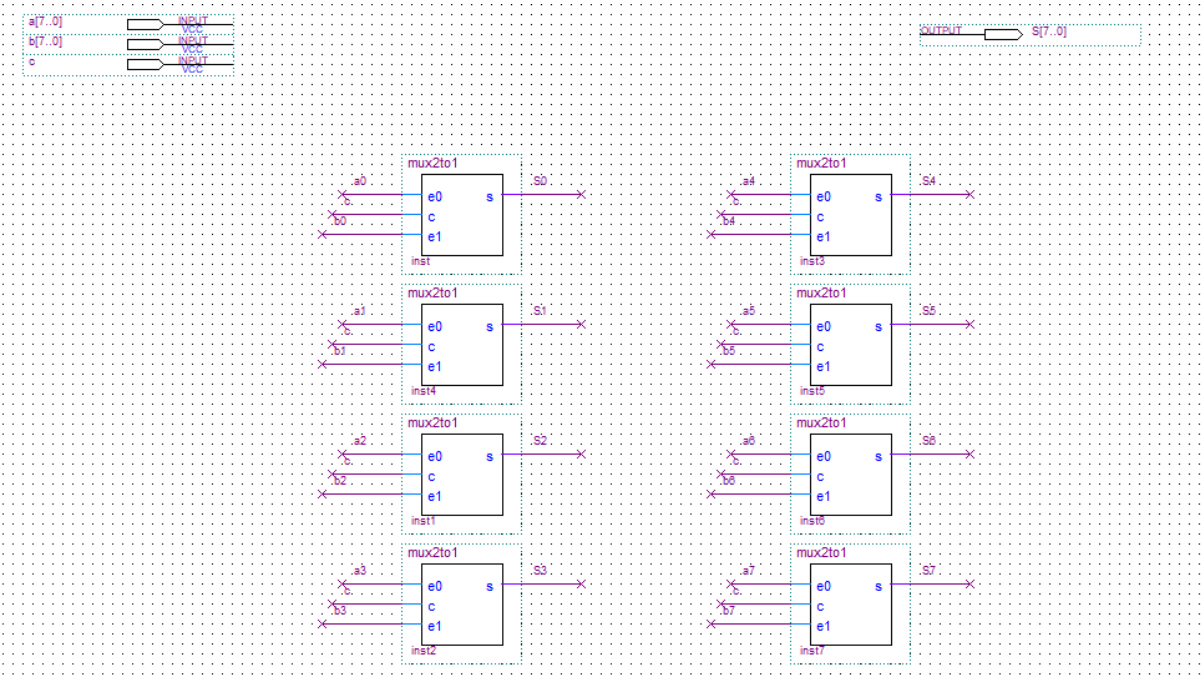
Nome: mux2x1_8bits

Descrição: Utiliza a lógica do mux2x1 para o tratamento de um vetor com 8 bits.

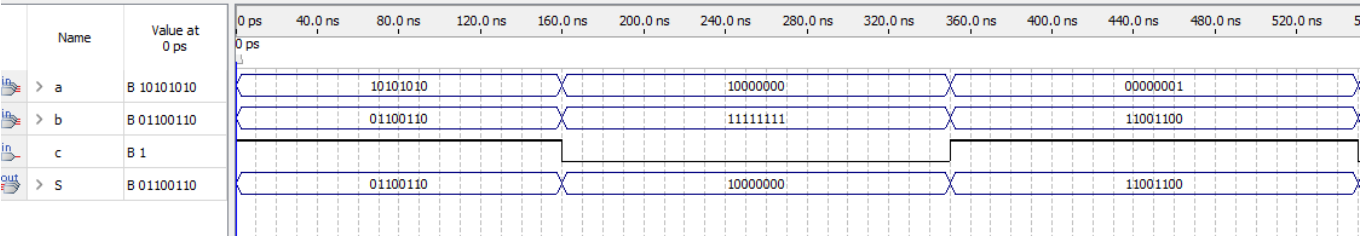
Símbolo:



Esquemático:



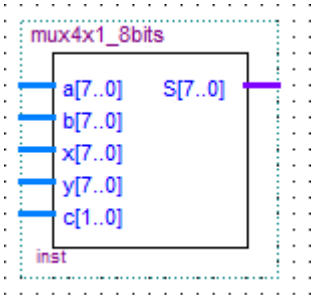
Simulação:



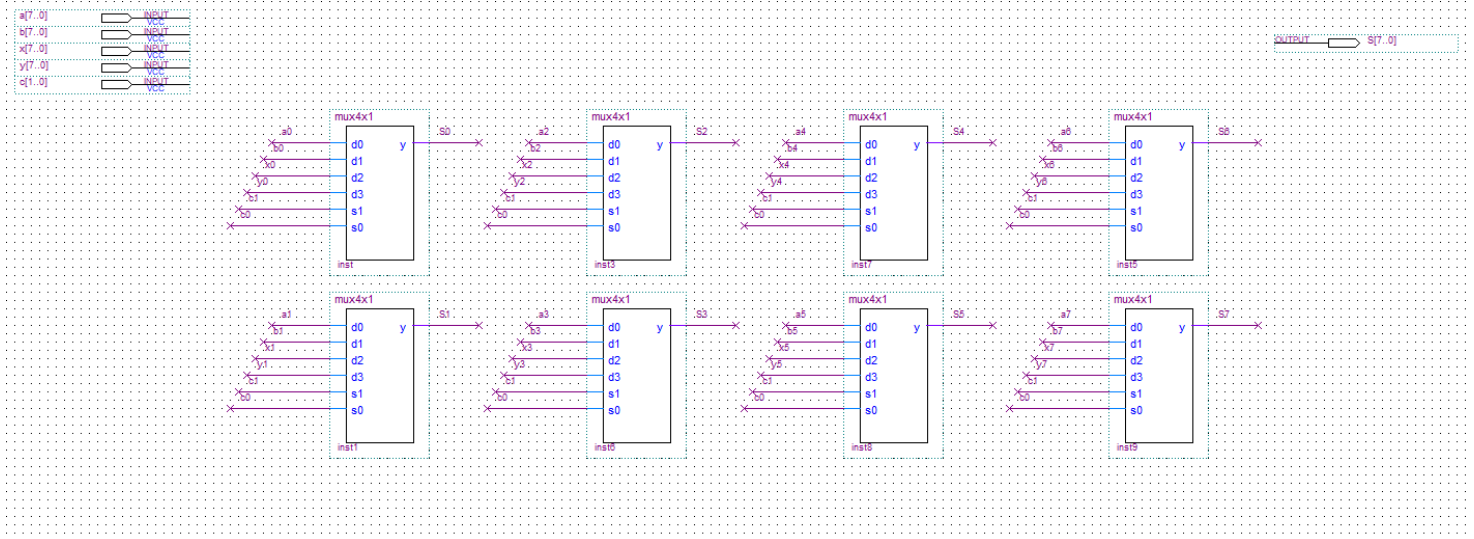
Nome: mux4x1_8bits

Descrição: Utiliza a lógica do mux4x1 para o tratamento de um vetor com 8 bits.

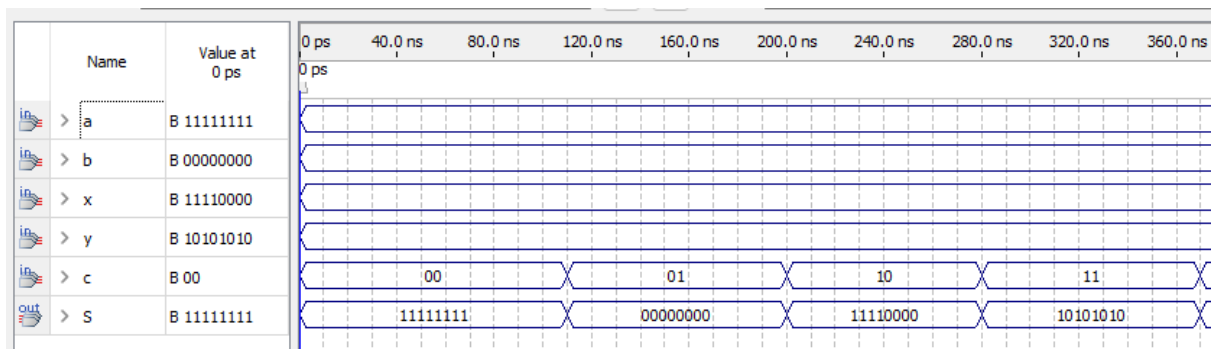
Símbolo:



Esquemático:



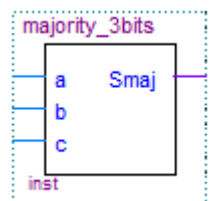
Simulação:



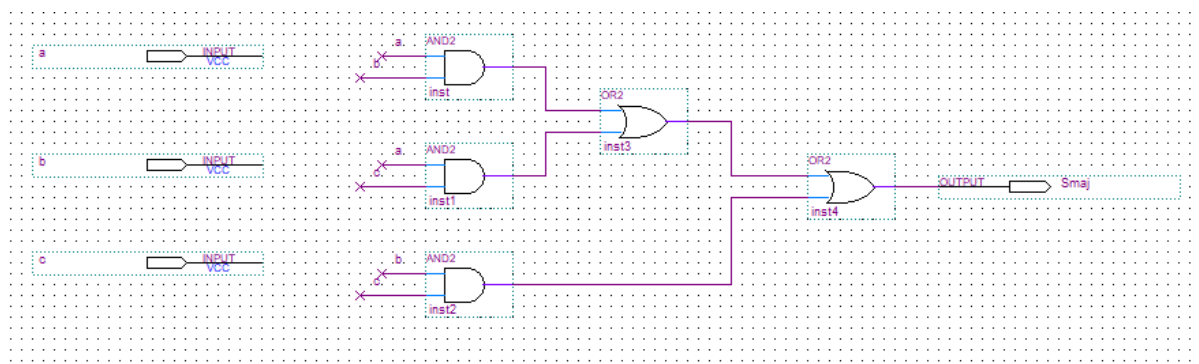
Nome: majority_3bits

Descrição: Efetua a função booleana majoritária entre 3 bits.

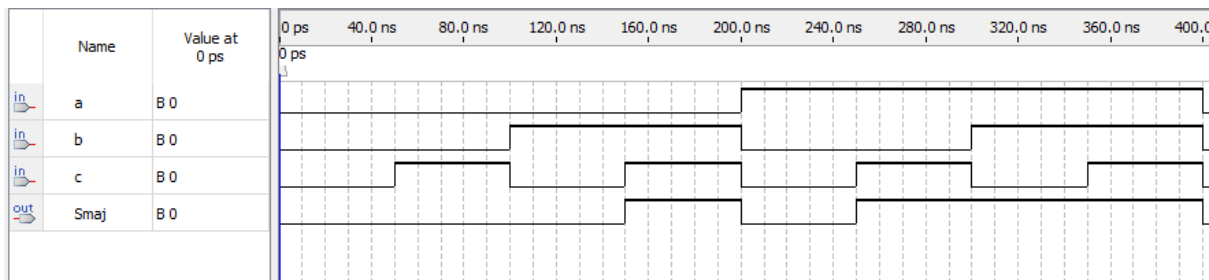
Símbolo:



Esquemático:



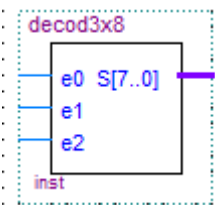
Simulação:



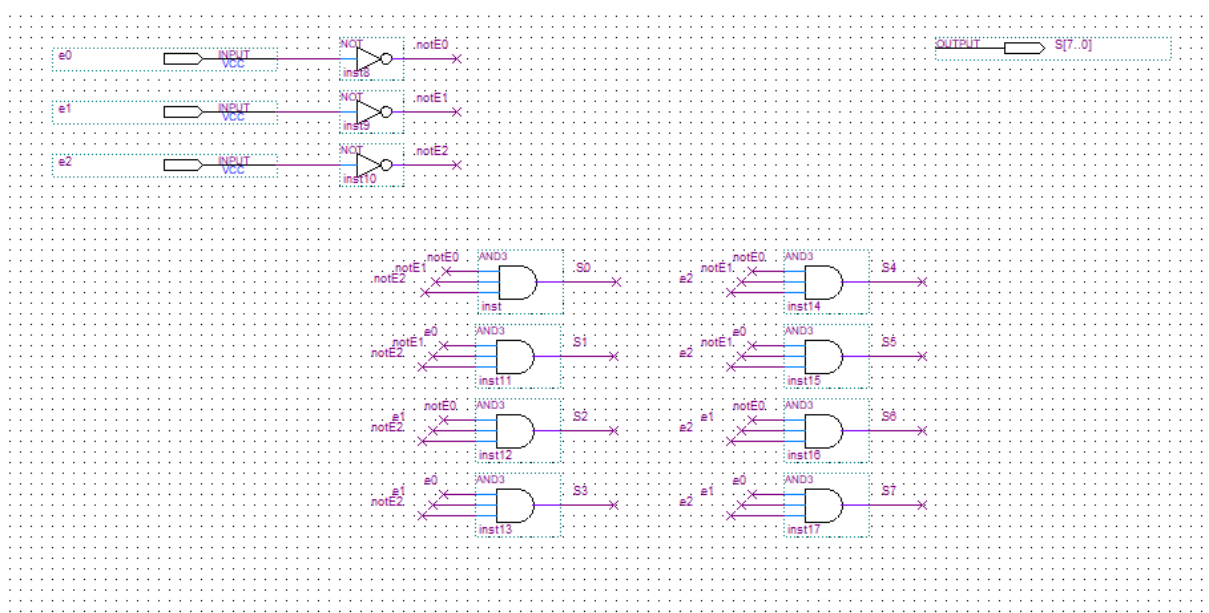
Nome: decod3x8

Descrição: Através de um vetor binário de 3 bits (e0, e1 e e2), seleciona uma única saída entre oito (S0 .. S7) como verdadeira.

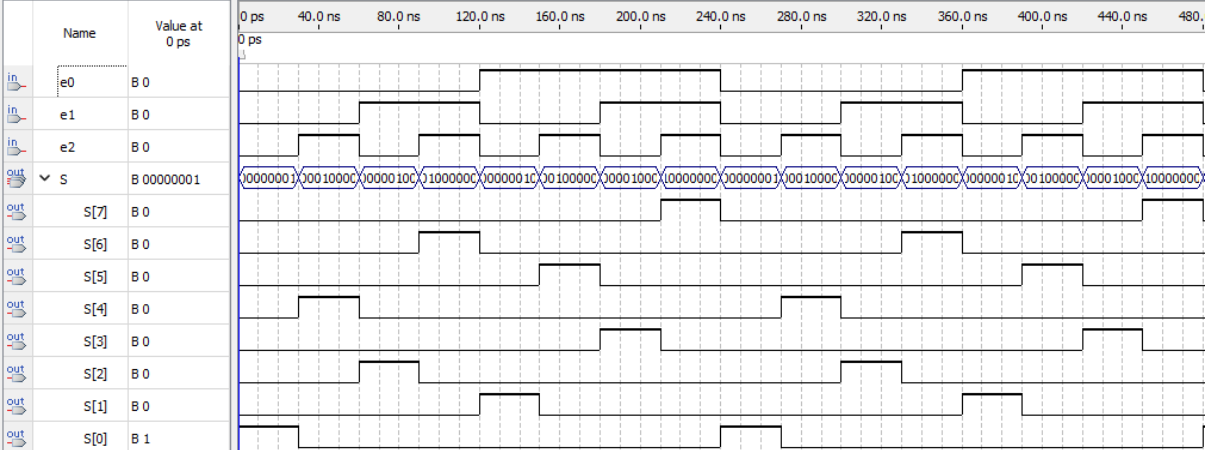
Símbolo:



Esquemático:



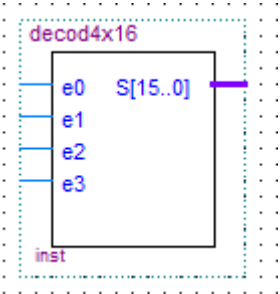
Simulação:



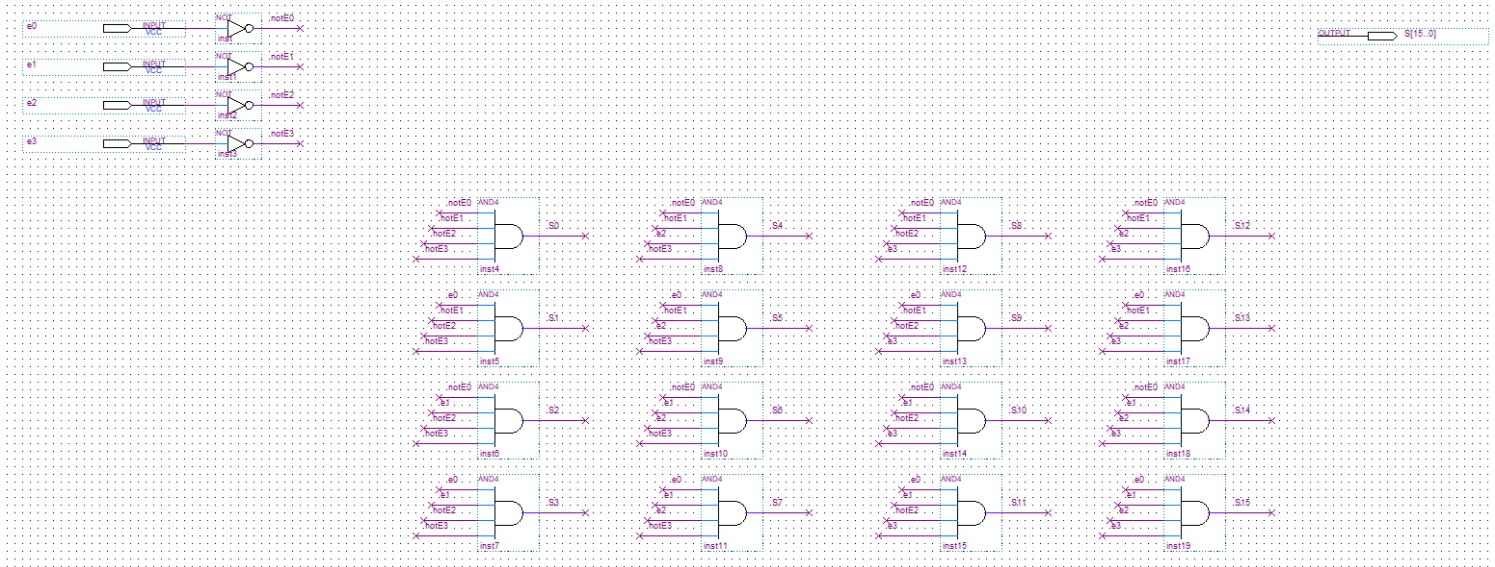
Nome: decod4x16

Descrição: Através de um vetor binário de 4 bits (e0, e1, e2 e e3), seleciona uma única saída entre dezesseis (S0 .. S15) como verdadeira.

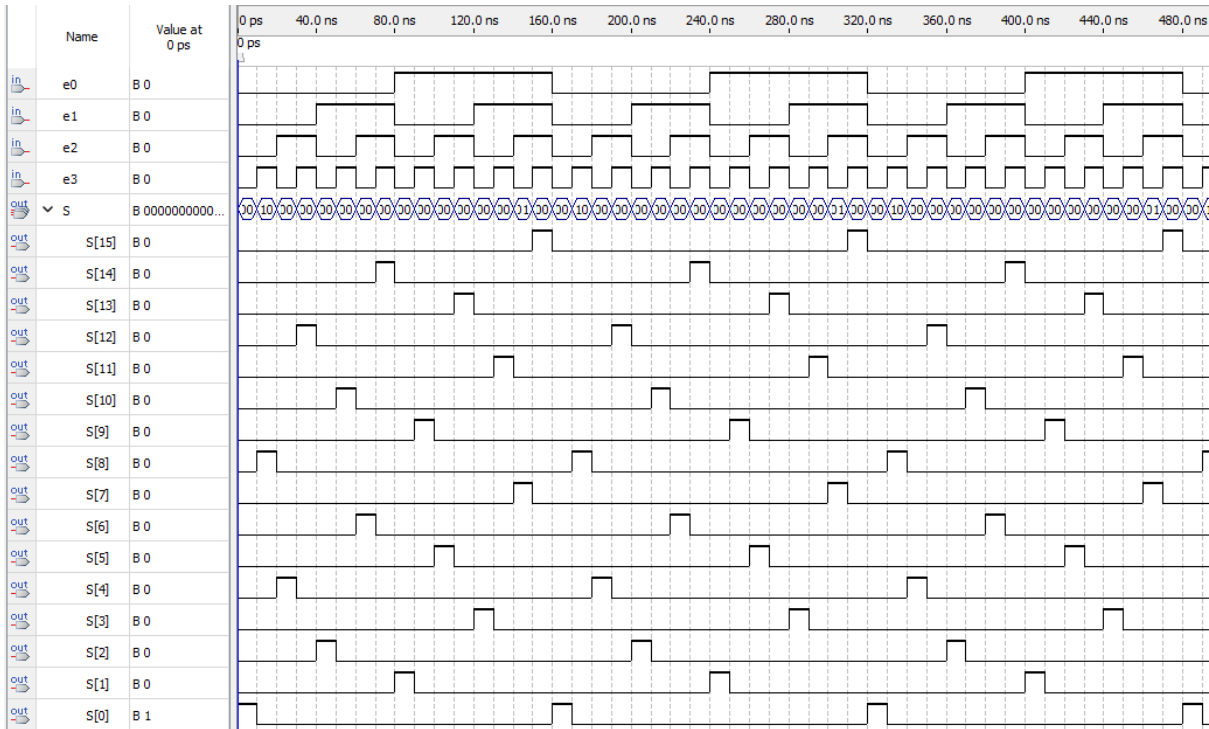
Símbolo:



Esquemático:



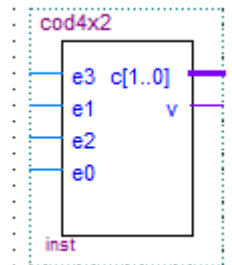
Simulação:



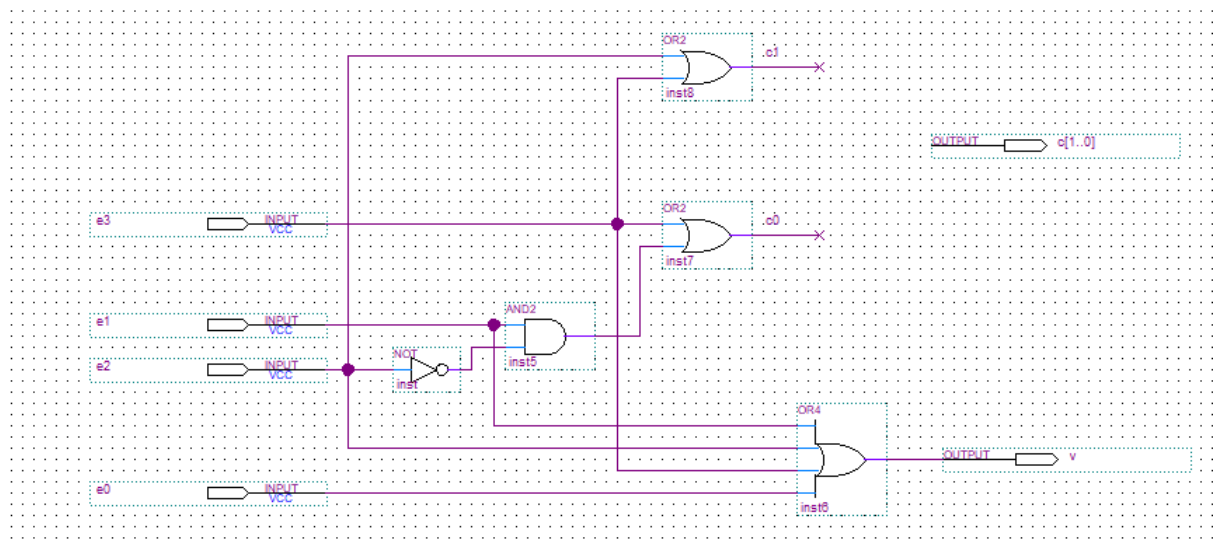
Nome: cod4x2

Descrição: Reduz a informação de um vetor binário de 4 bits (e0, e1, e2 e e3) para outro com a sua codificação em 2 bits (c0 e c1).

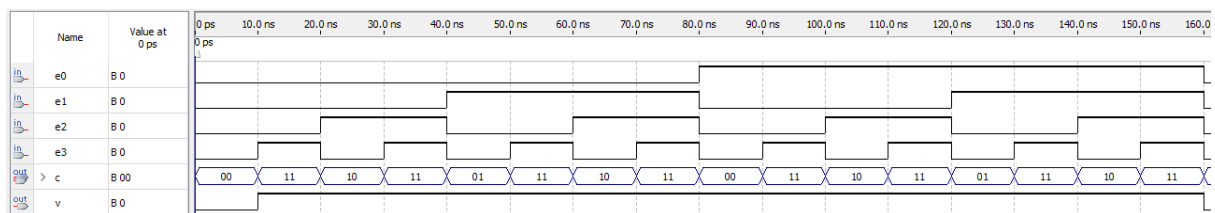
Símbolo:



Esquemático:



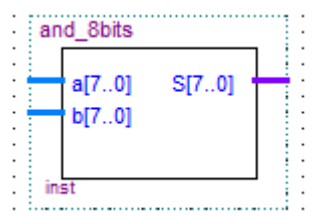
Simulação:



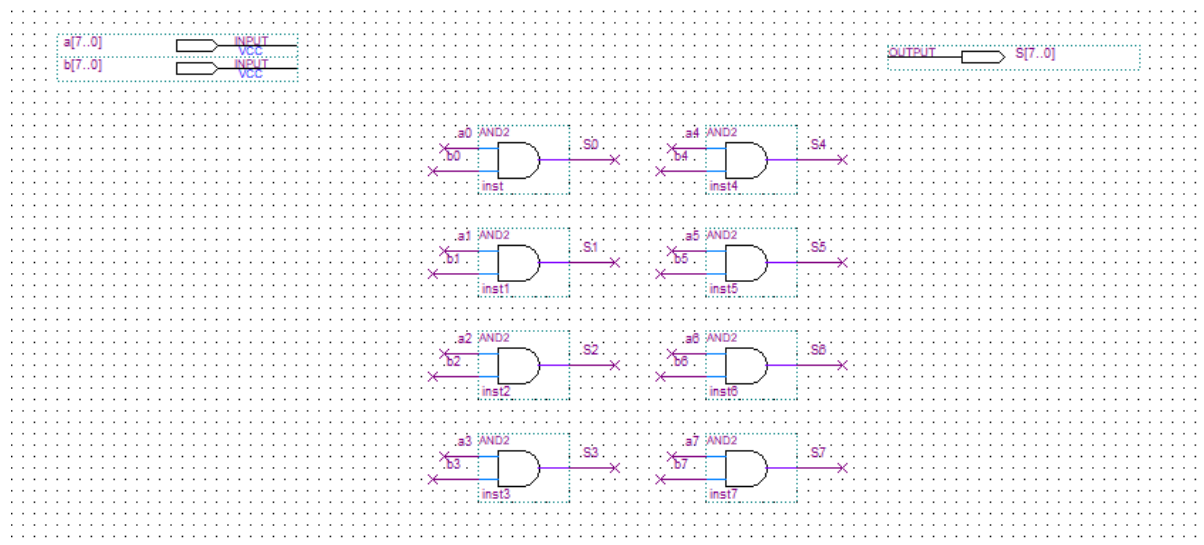
Nome: and_8bits

Descrição: Efetua a função booleana AND entre dois vetores de 8 bits.

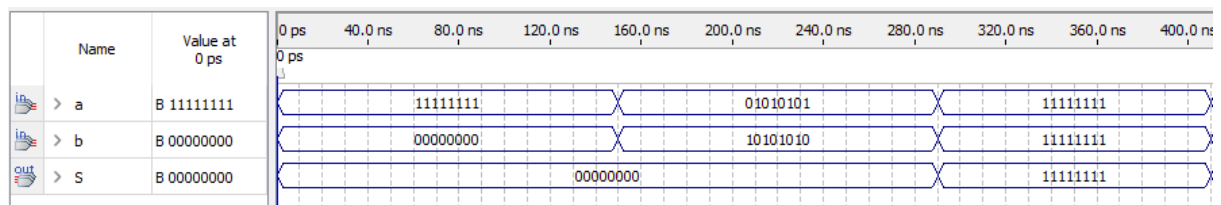
Símbolo:



Esquemático:



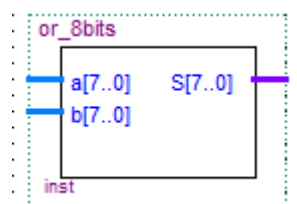
Simulação:



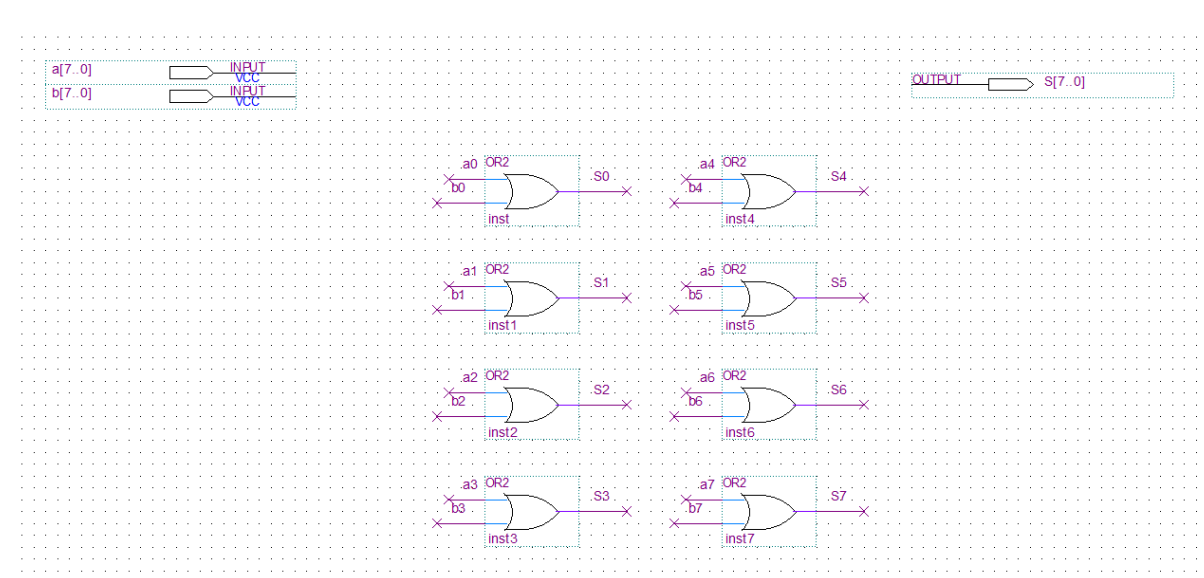
Nome: or_8bits

Descrição: Efetua a função booleana OR entre dois vetores de 8 bits.

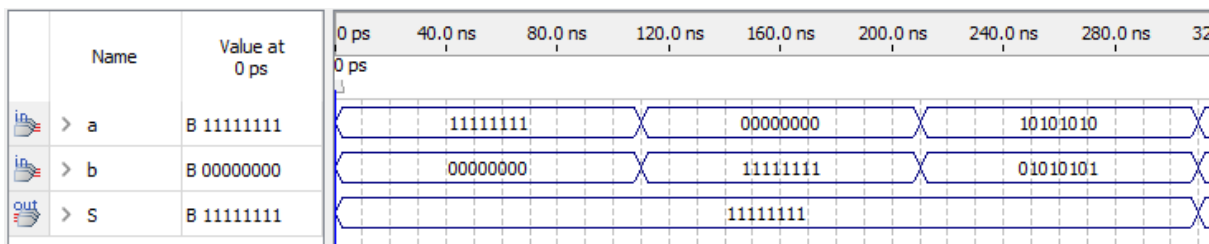
Símbolo:



Esquemático:



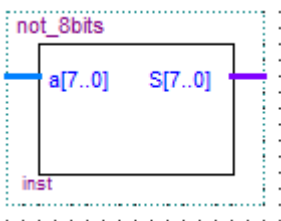
Simulação:



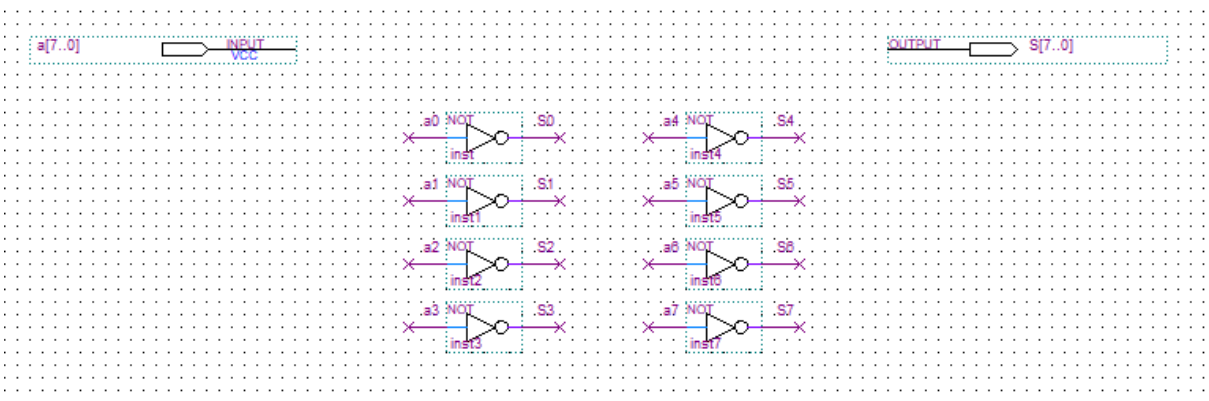
Nome: not_8bits

Descrição: Efetua a função booleana NOT num vetor de 8 bits.

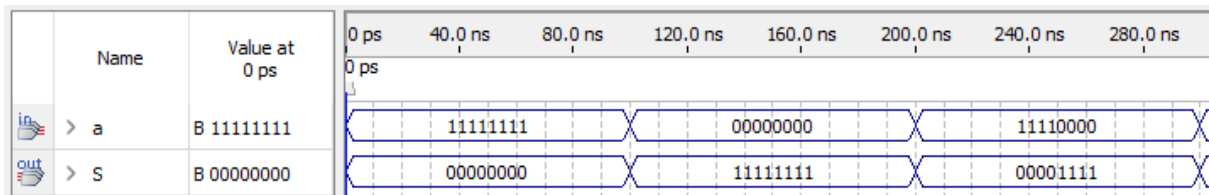
Símbolo:



Esquemático:



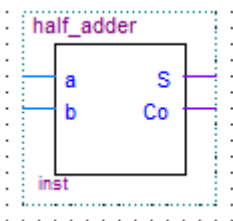
Simulação:



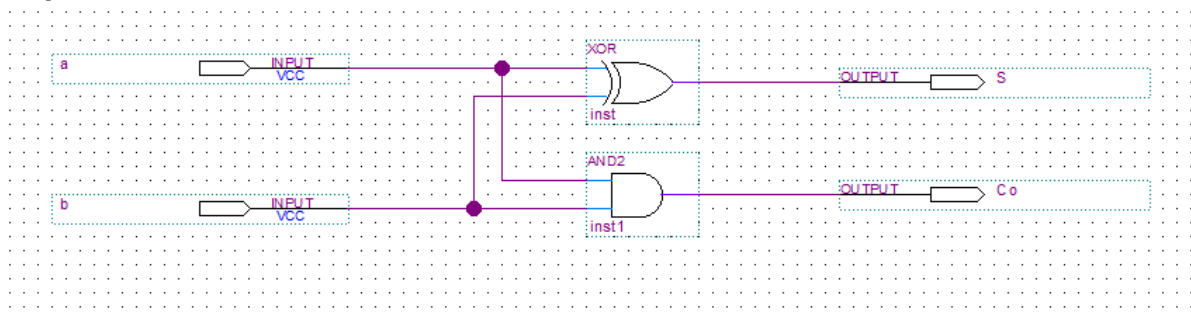
Nome: half_adder

Descrição: Meio somador.

Símbolo:



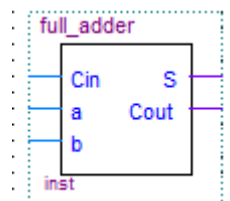
Esquemático:



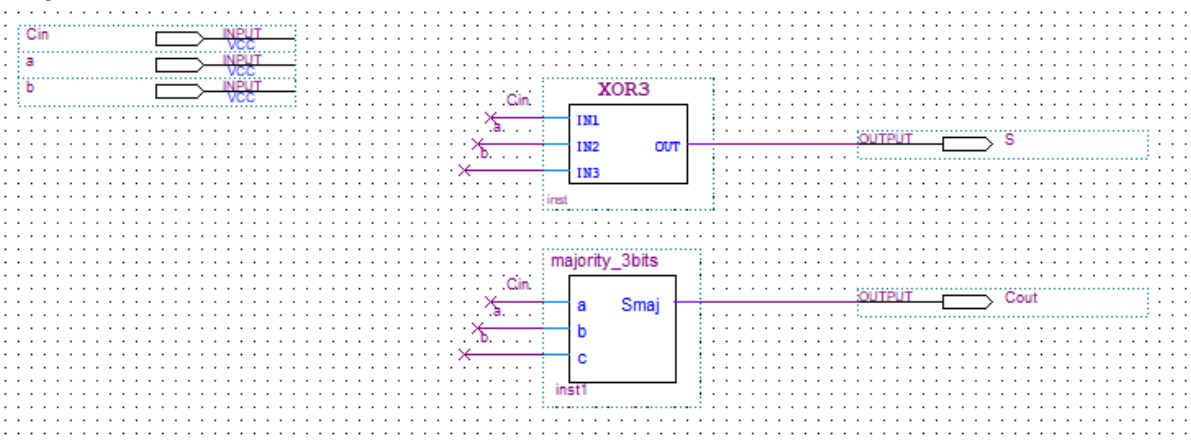
Nome: full_adder

Descrição: Somador completo.

Símbolo:



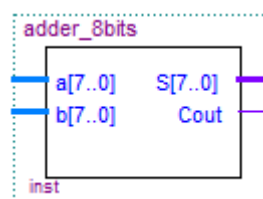
Esquemático:



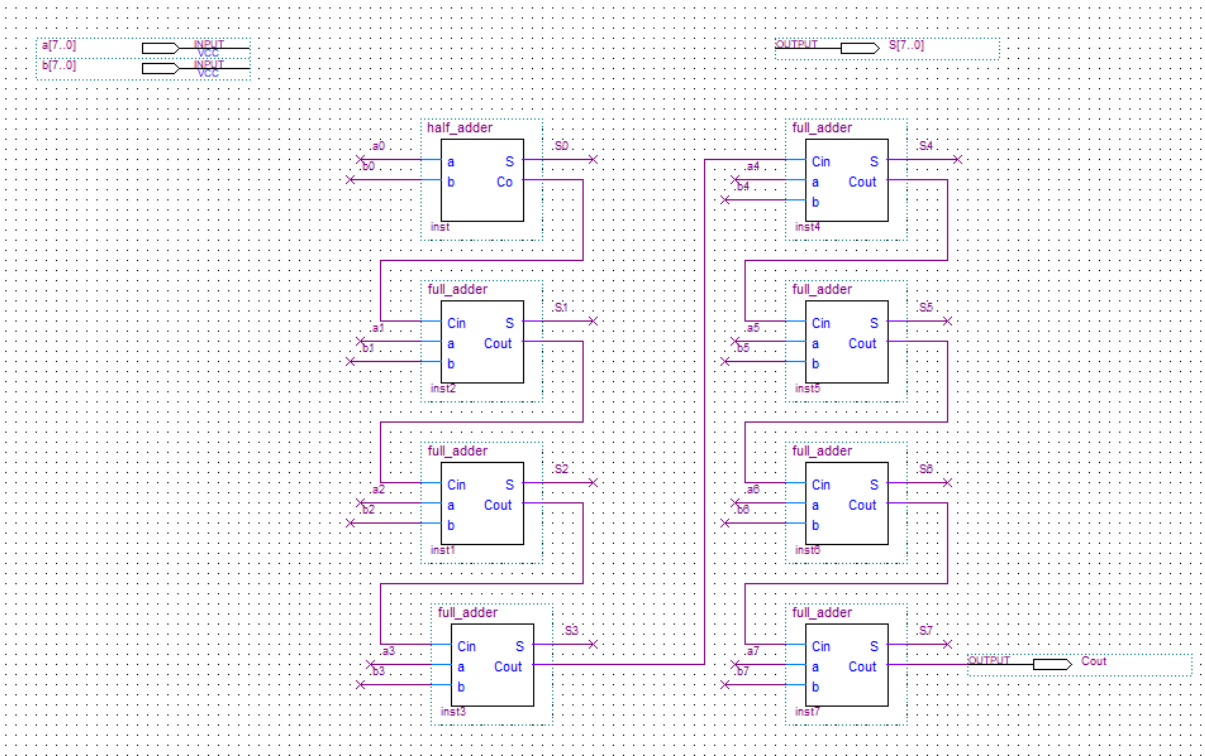
Nome: adder_8bits

Descrição: Somador de dois vetores de 8 bits.

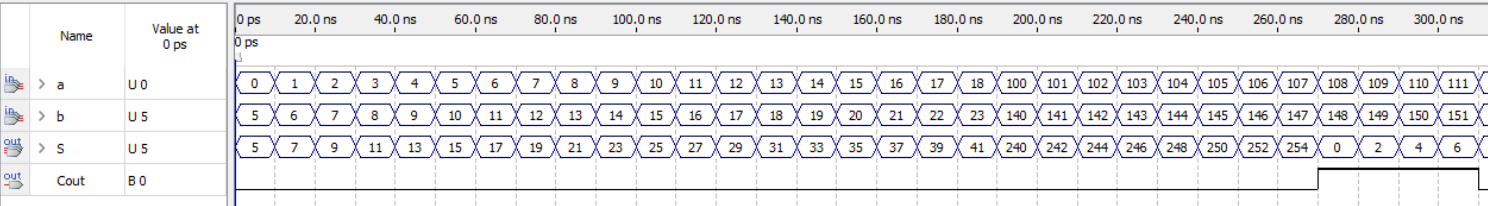
Símbolo:



Esquemático:



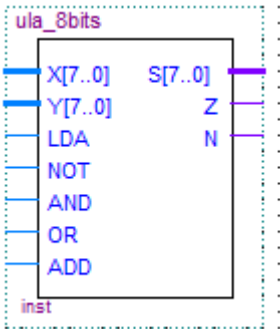
Simulação:



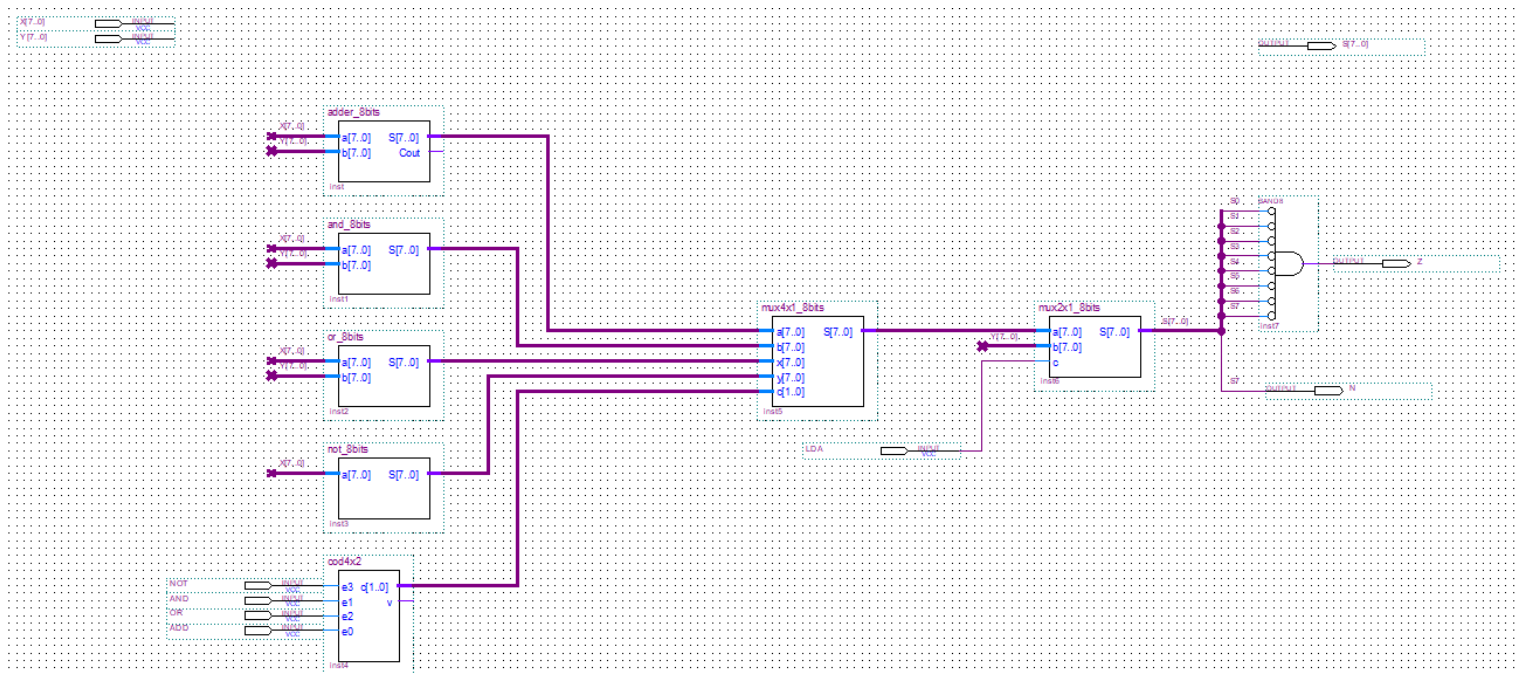
Nome: ula_8bits

Descrição: Unidade lógica e aritmética do processador Neander.

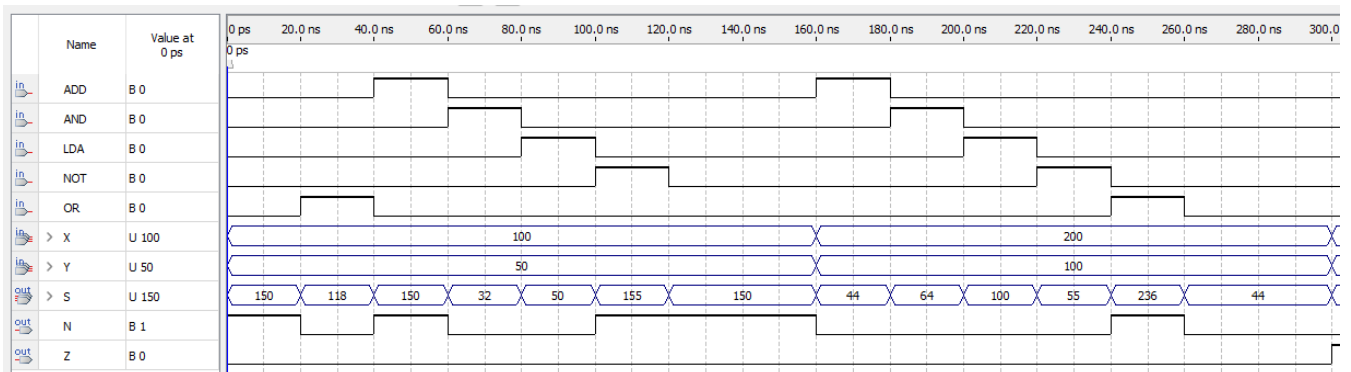
Símbolo:



Esquemático:



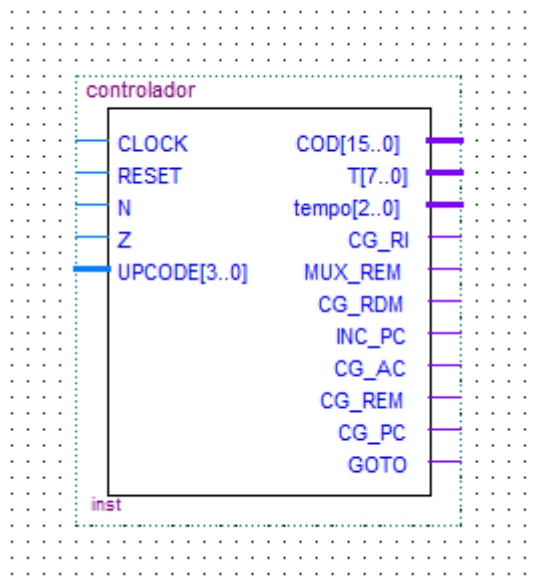
Simulação:



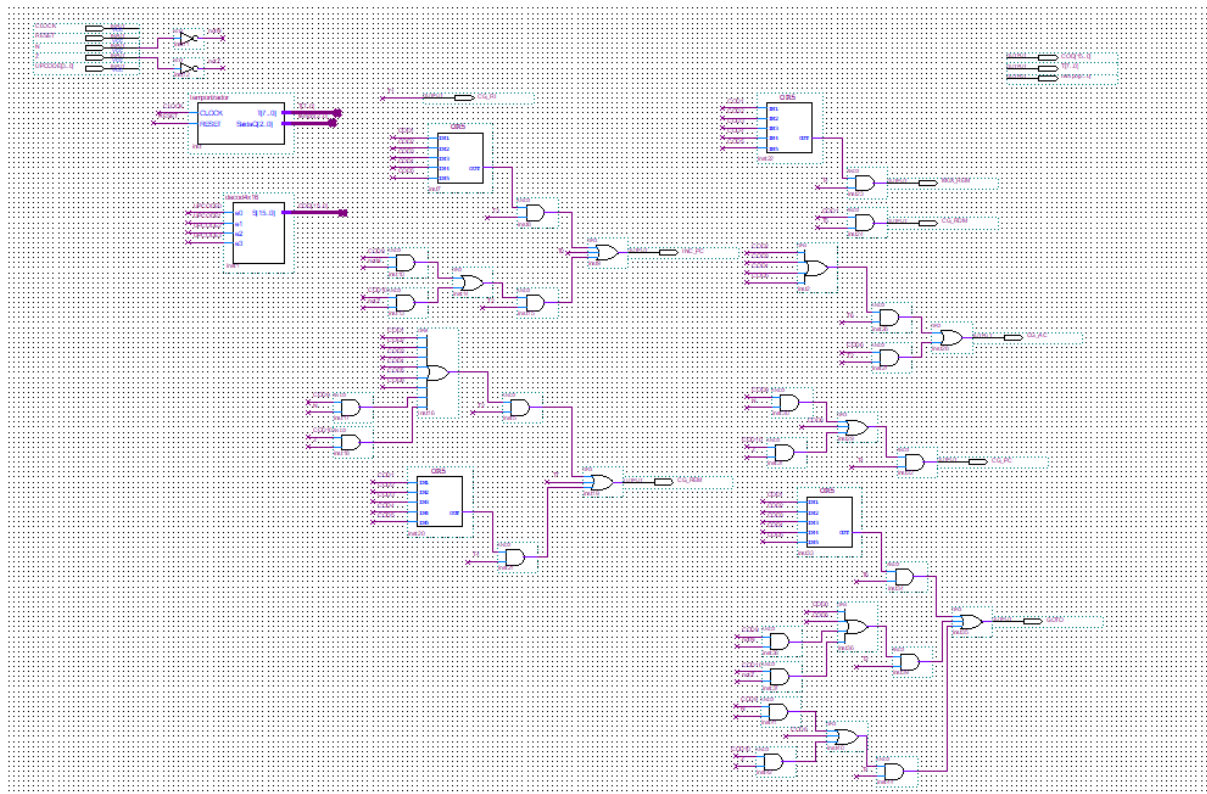
Nome: controlador

Descrição: Efetuar a seleção dos sinais de controle das operações do Neander com base no código de instruções de 4 bits.

Símbolo:

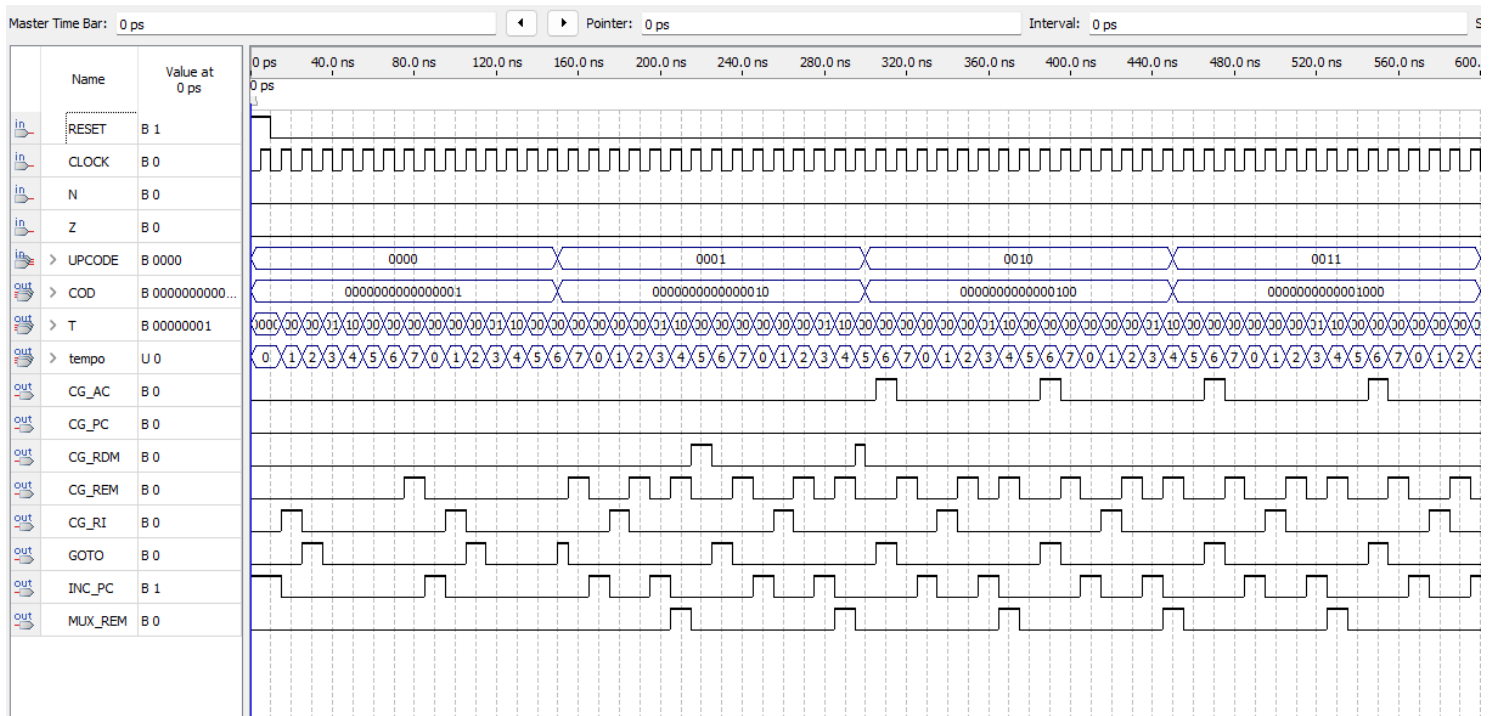


Esquemático:

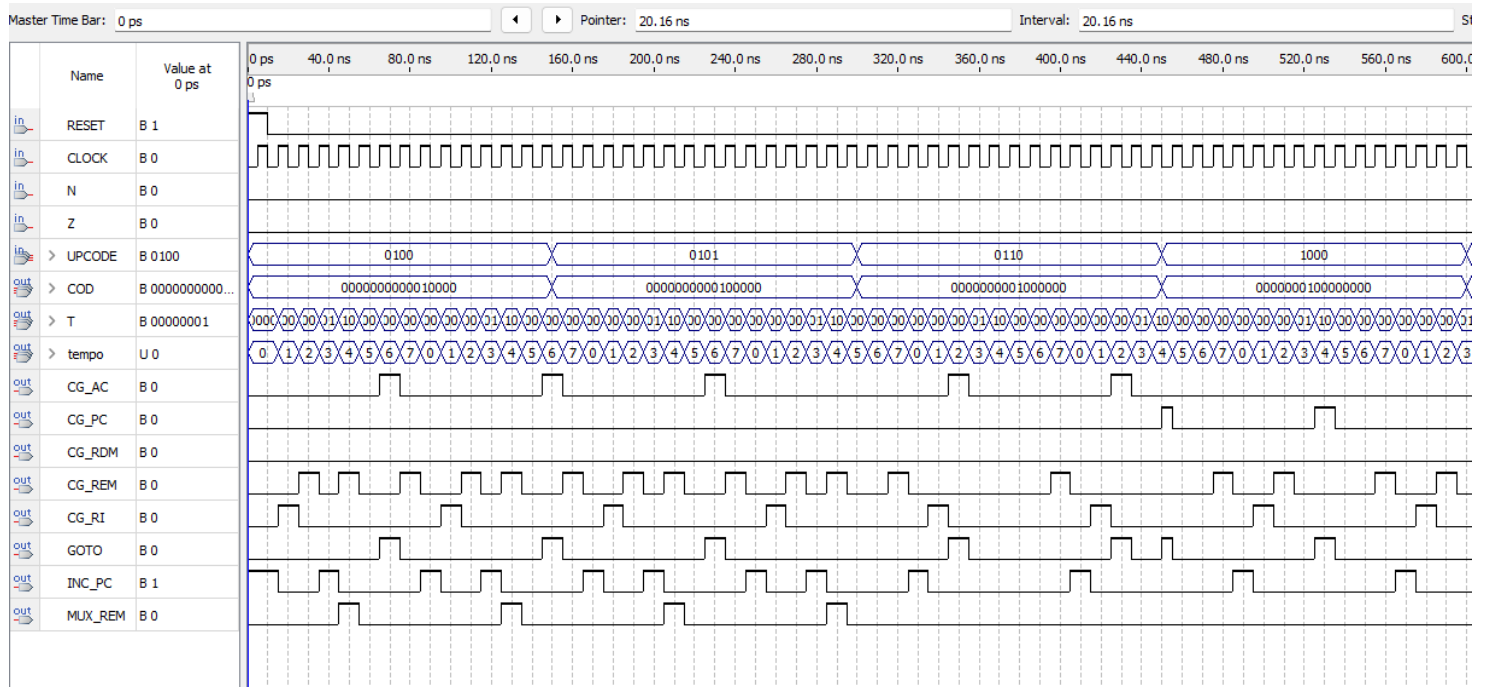


Simulação:

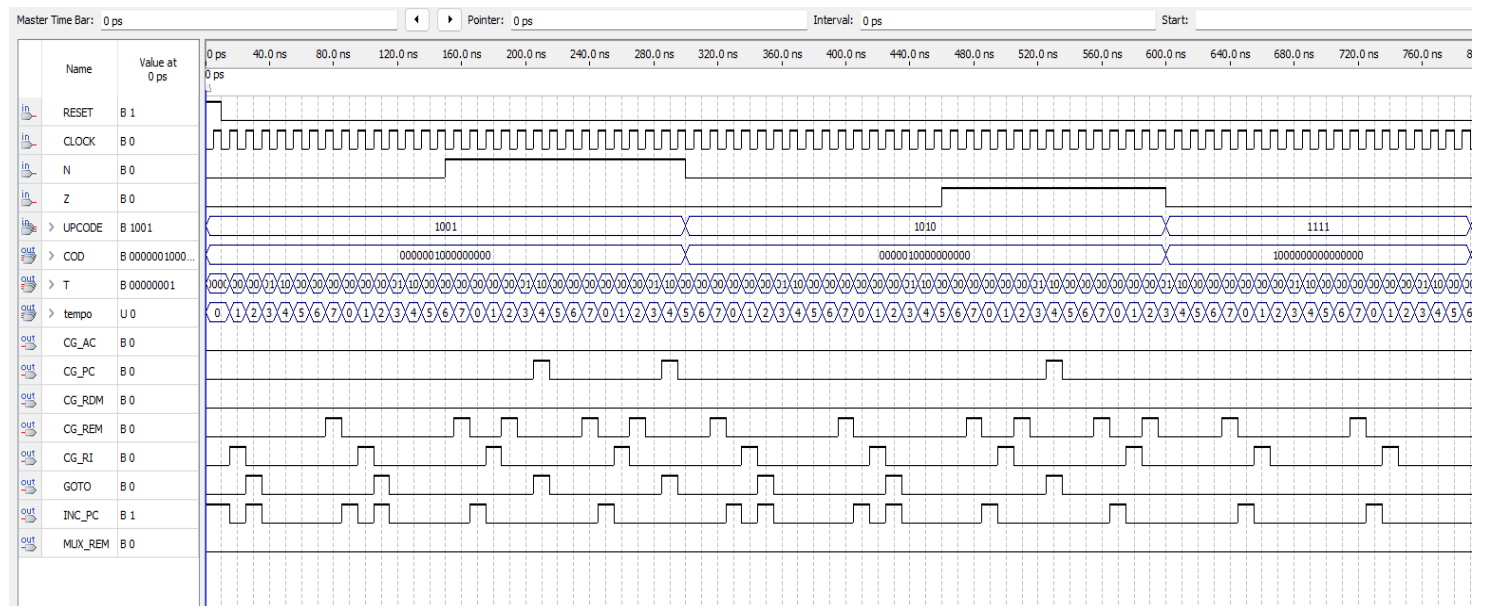
NOP, STA, LDA, ADD



OR, AND, NOT, JMP



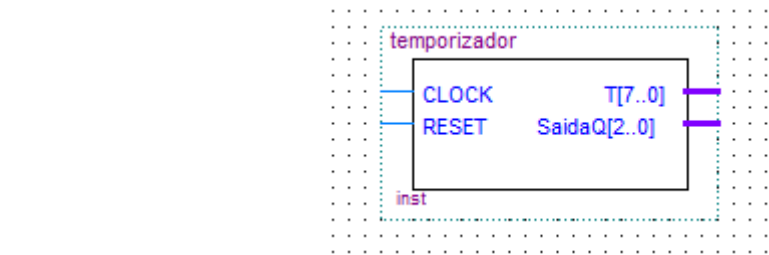
JN (N = 0), JN (N = 1), JZ (Z = 0), JZ (Z = 1), HLT



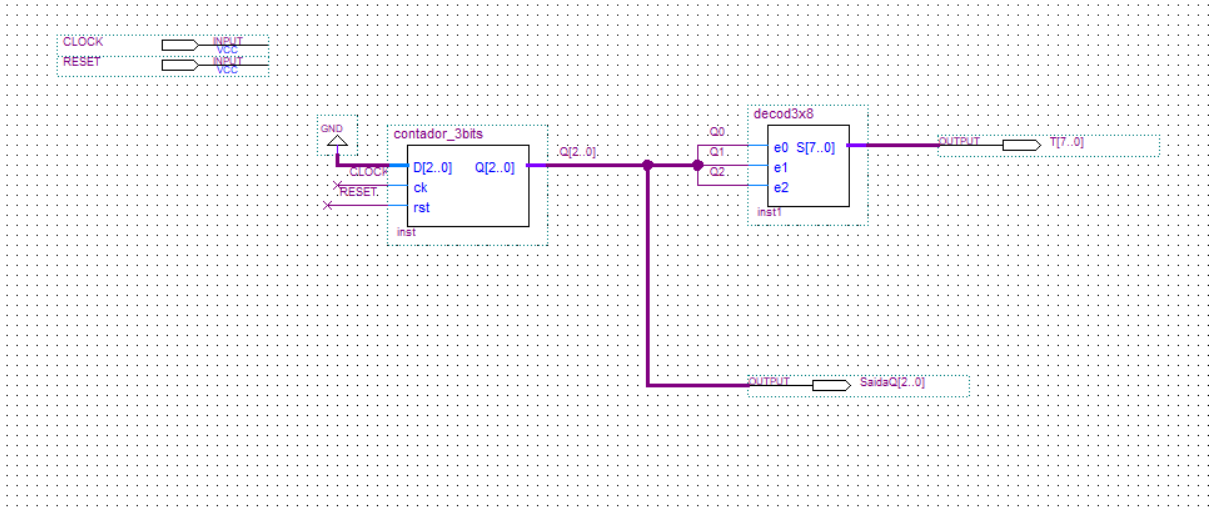
Nome: temporizador

Descrição: possui como entrada os sinais de CLOCK e RESET e como saída um vetor de 8 bits referente à saída do decodificador 3x8 indicando qual o tempo atual, e um vetor de 3 bits referente ao valor de saída do contador de 3 bits.

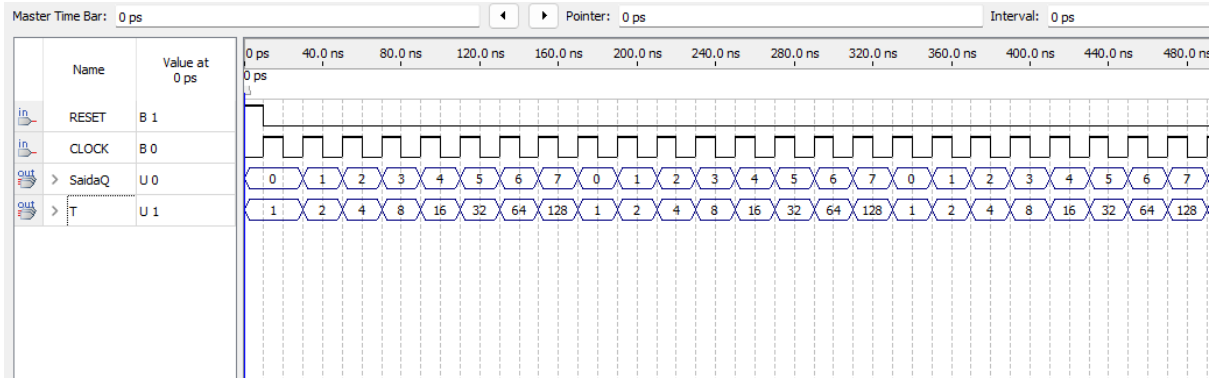
Símbolo:



Esquemático:

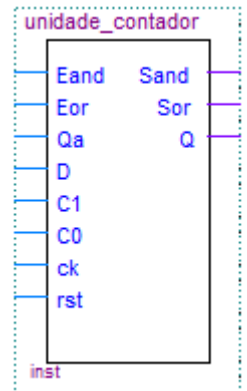


Simulação:

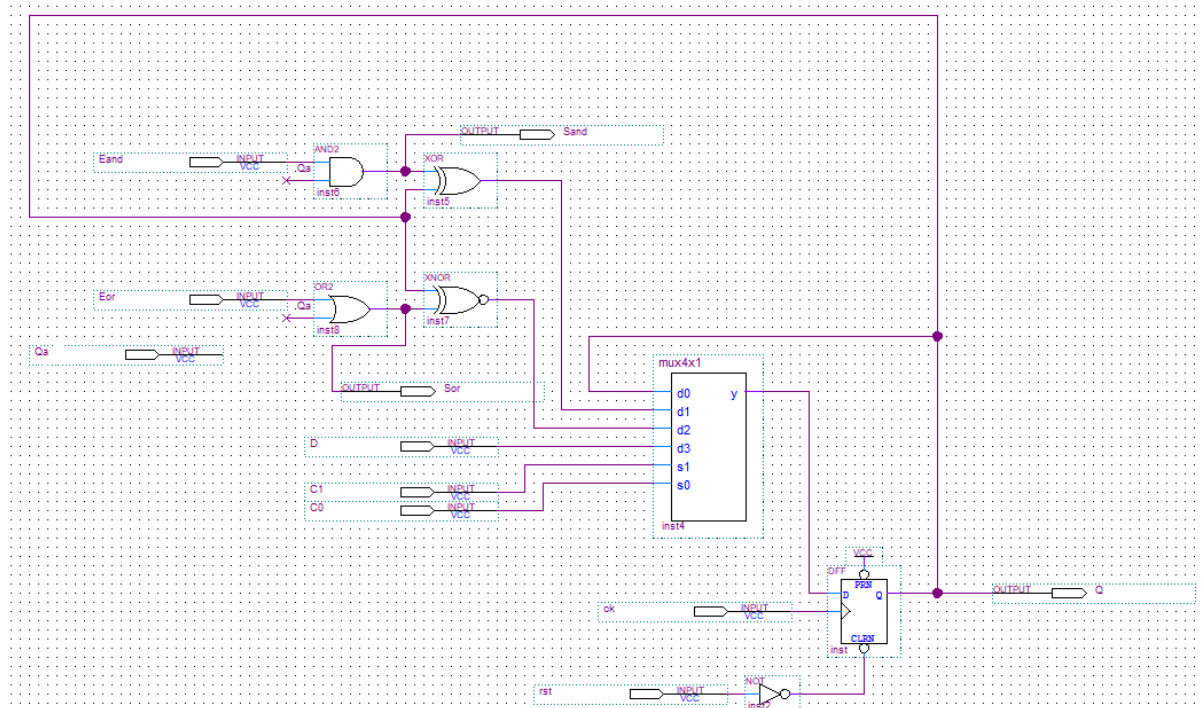


Descrição: Unidade de registrador de 1 bit com a possibilidade de incremento, decremento, dado e carga com base nos sinais de controle (C0 e C1).

Símbolo:



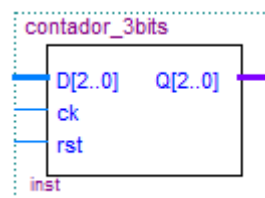
Esquemático:



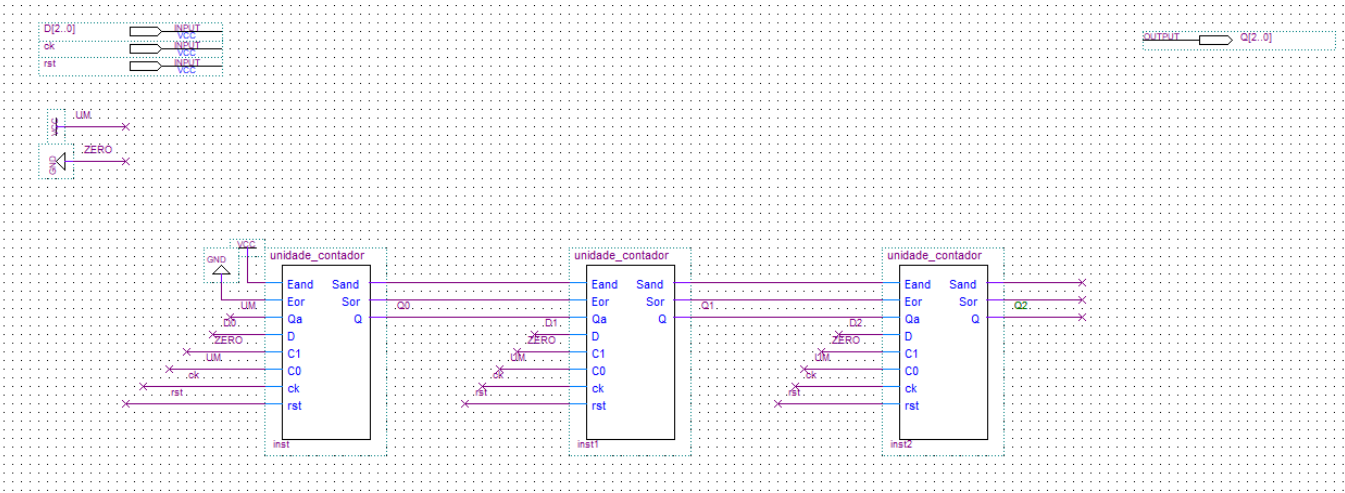
Nome: contador_3bits

Descrição: Conjunto de três unidade_contador definidas com o modo de incremento (C0 = 1 e C1 = 0) para efetuar a contagem de 0 a 7.

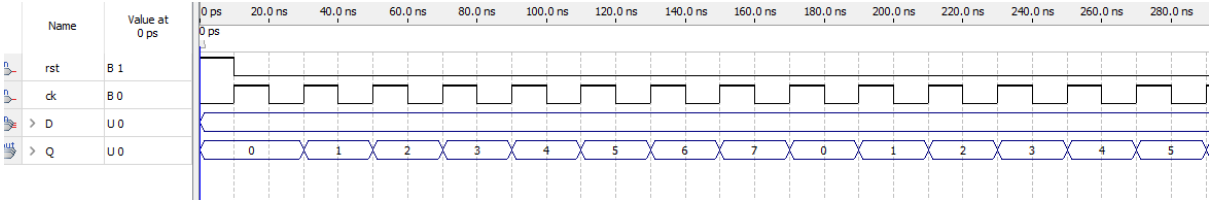
Símbolo:



Esquemático:



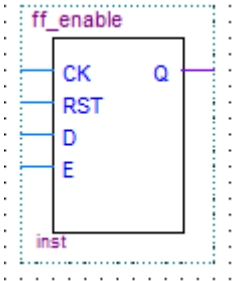
Simulação:



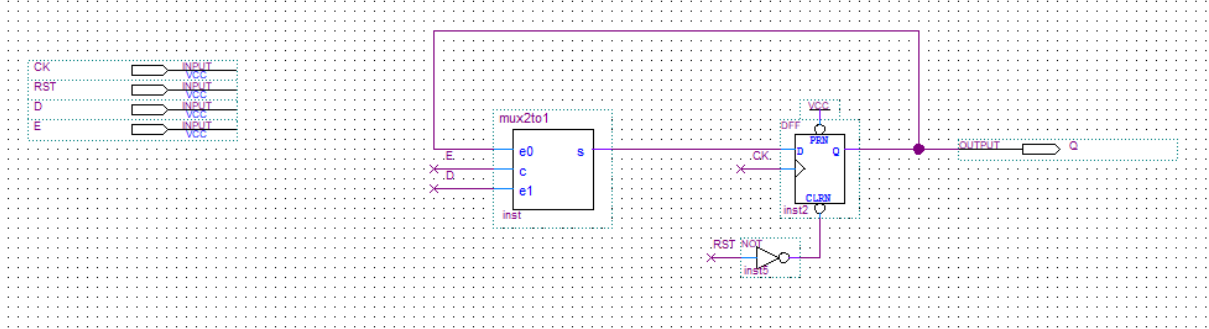
Nome: ff_enable

Descrição: Flip-flop com enable.

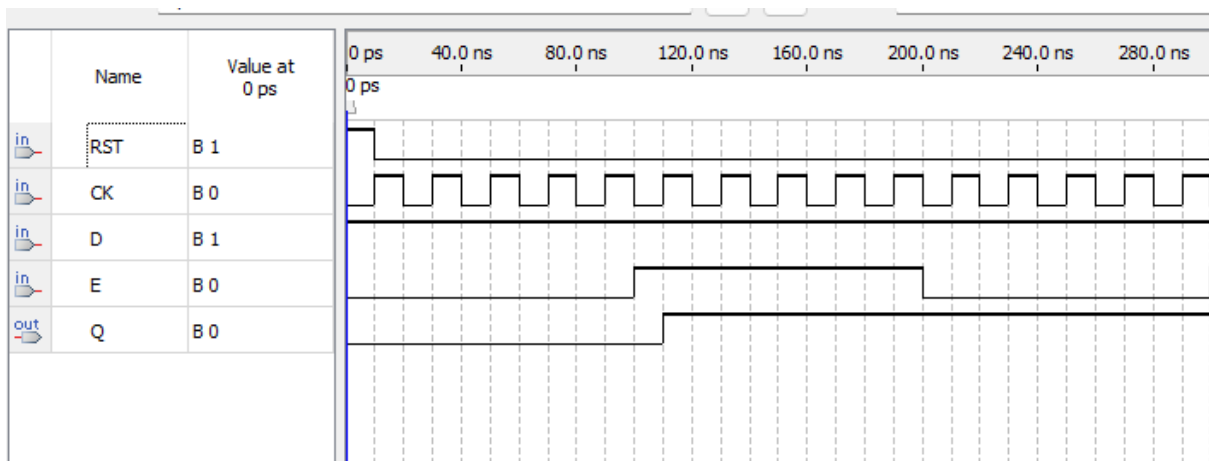
Símbolo:



Esquemático:



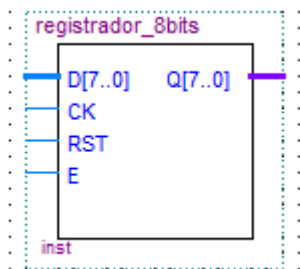
Simulação:



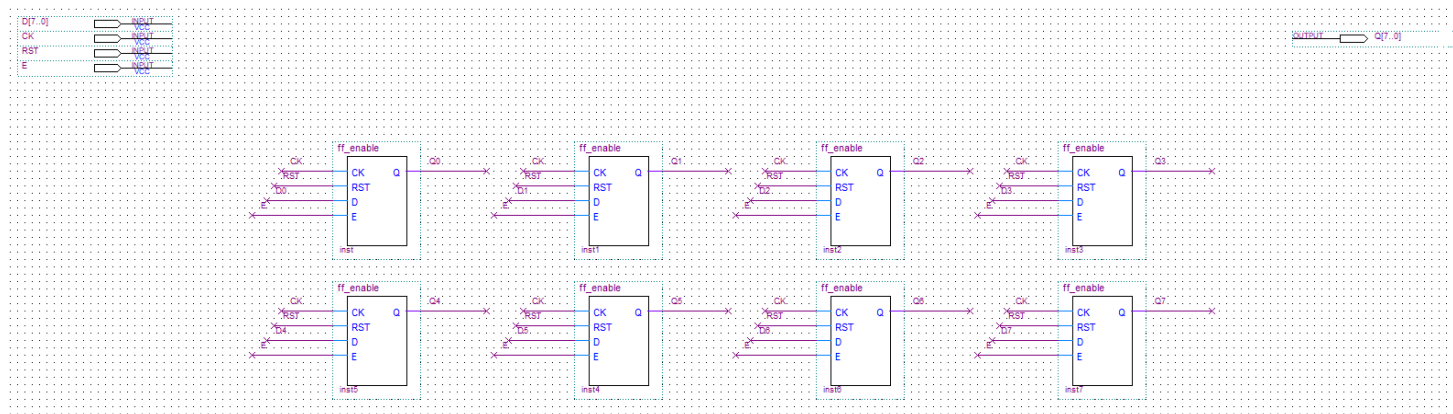
Nome: registrador_8bits

Descrição: Conjunto de oito flip-flops com enable para o armazenamento ou carregamento de dados.

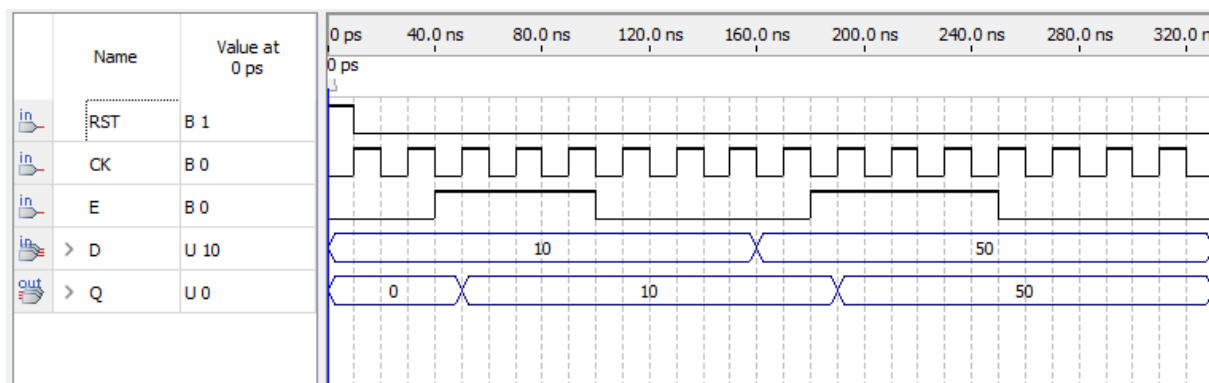
Símbolo:



Esquemático:



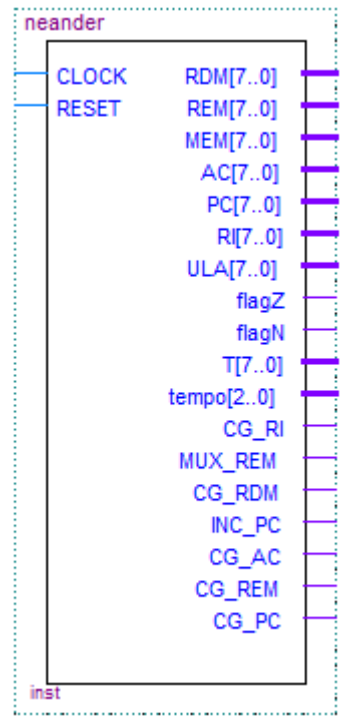
Simulação:



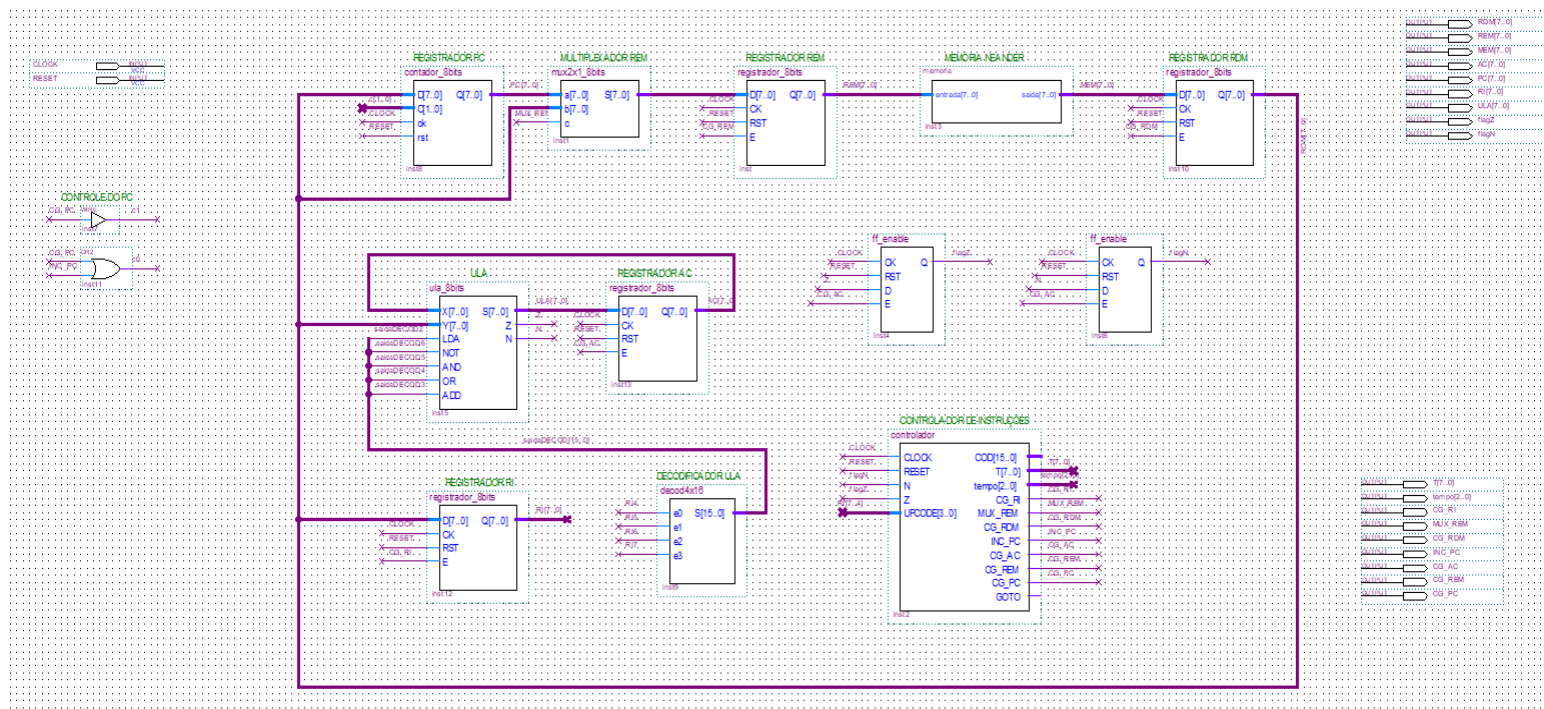
Nome: neander

Descrição: Neander completo.

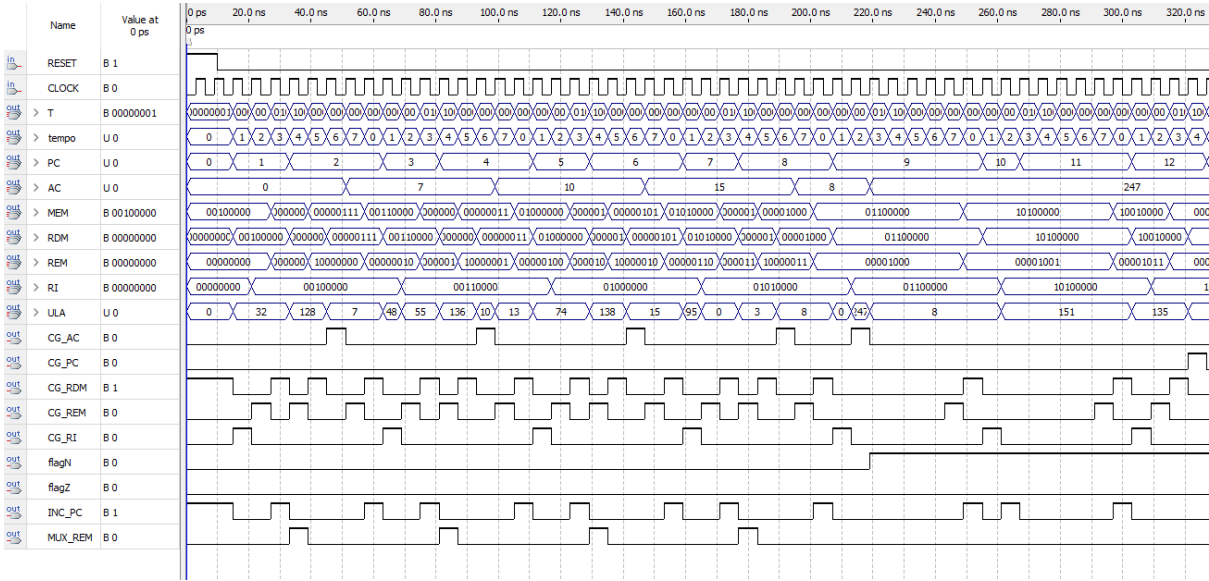
Símbolo:



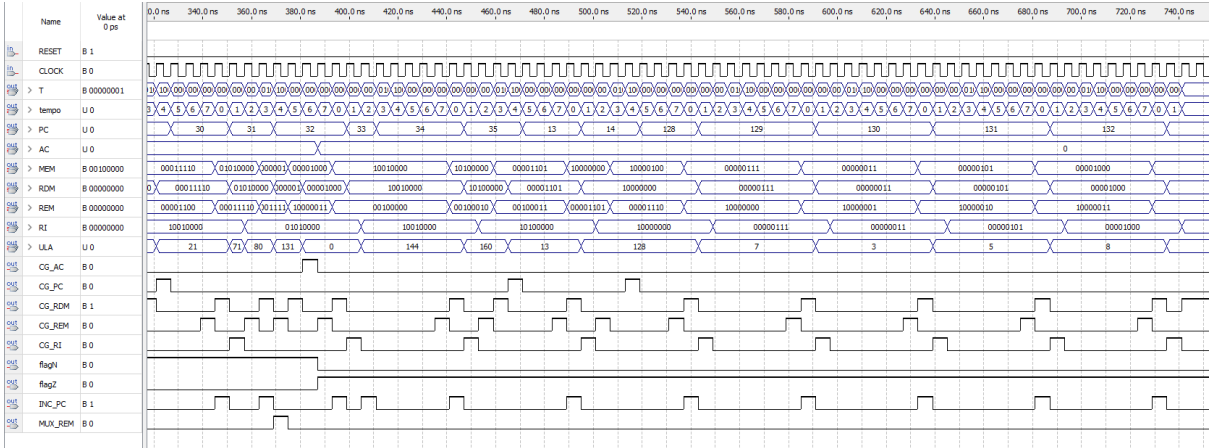
Esquemático:



Simulação com memória testbench:



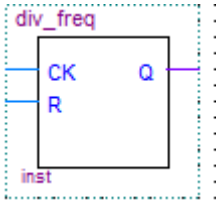
Continuação...



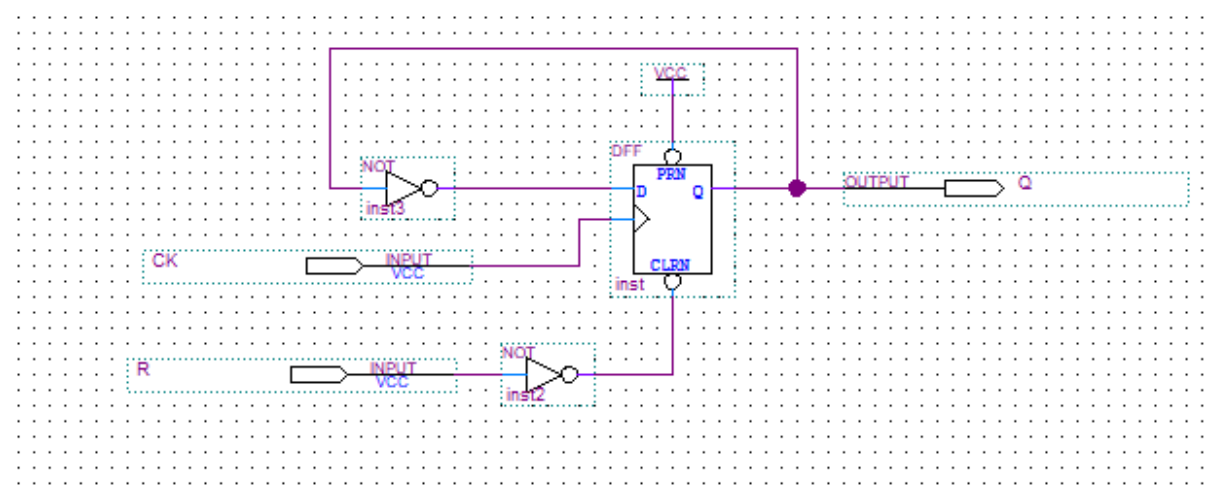
Nome: div_freq

Descrição: Divisor de frequência.

Símbolo:



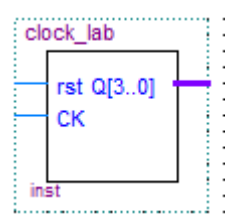
Esquemático:



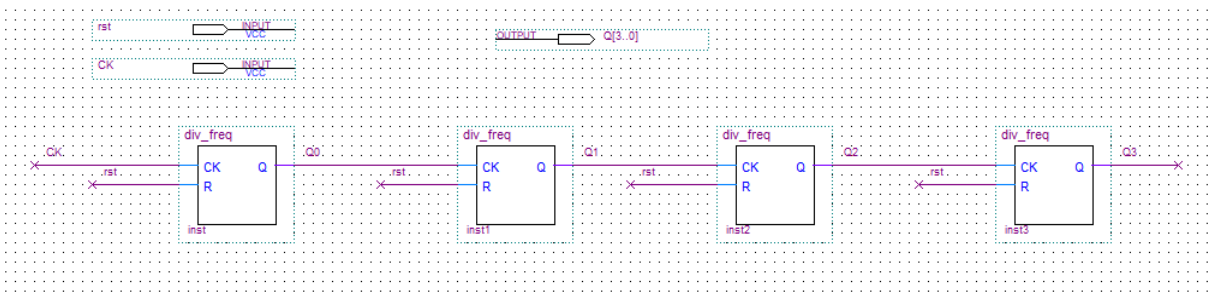
Nome: clock_lab

Descrição: Bloco que divide o sinal de entrada do clock pela metade em quatro estágios, cada um sendo a metade do clock anterior.

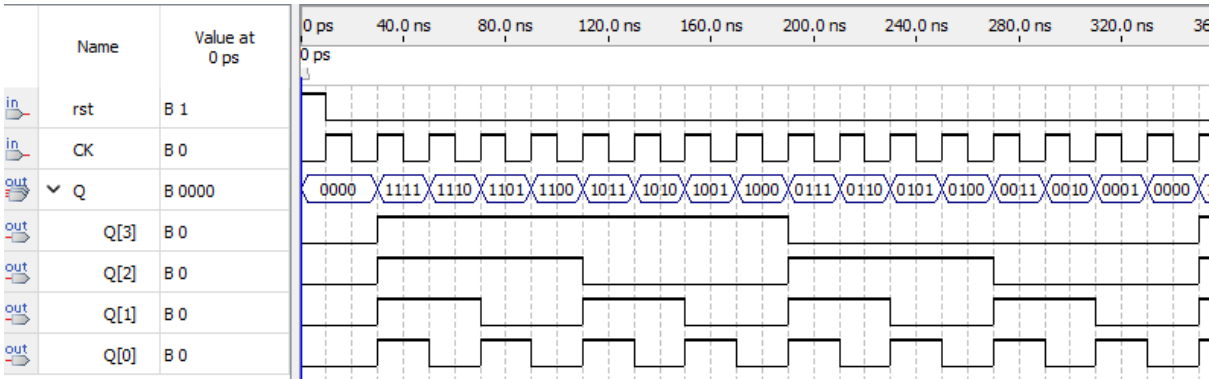
Símbolo:



Esquemático:



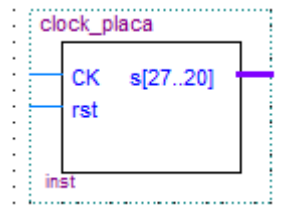
Simulação:



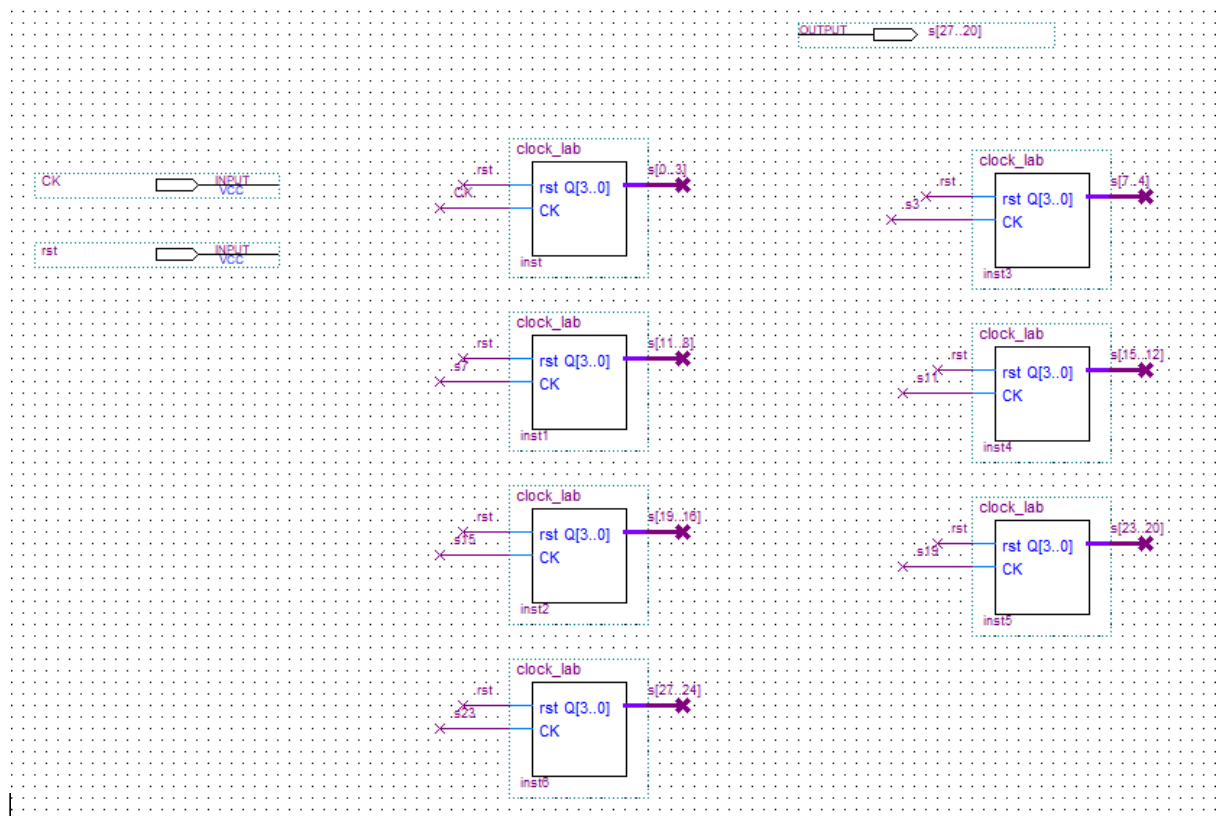
Nome: clock_placa

Descrição: Conjunto de sete clock_lab para dividir o clock padrão da placa e obter uma visualização próxima de 1 segundo no seu display.

Símbolo:

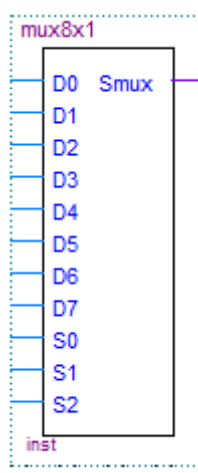


Esquemático:

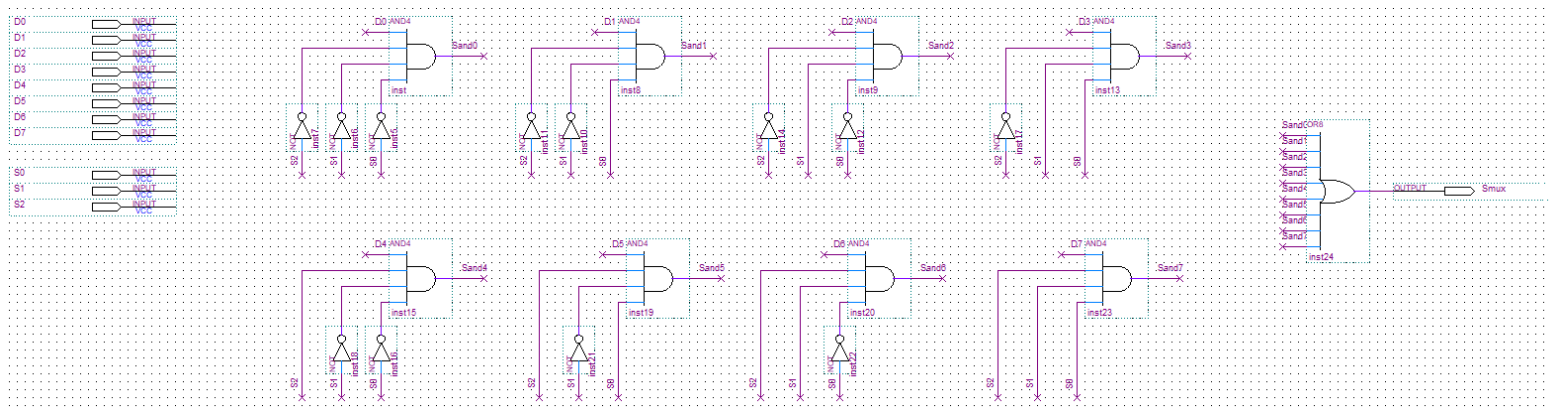


Descrição: Selecionar uma entre oito entradas (D0 .. D7) do circuito como saída (Smux) com base em 3 sinais de entrada (S0, S1 e S2).

Símbolo:



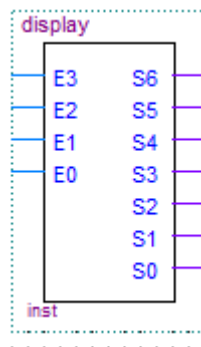
Esquemático:



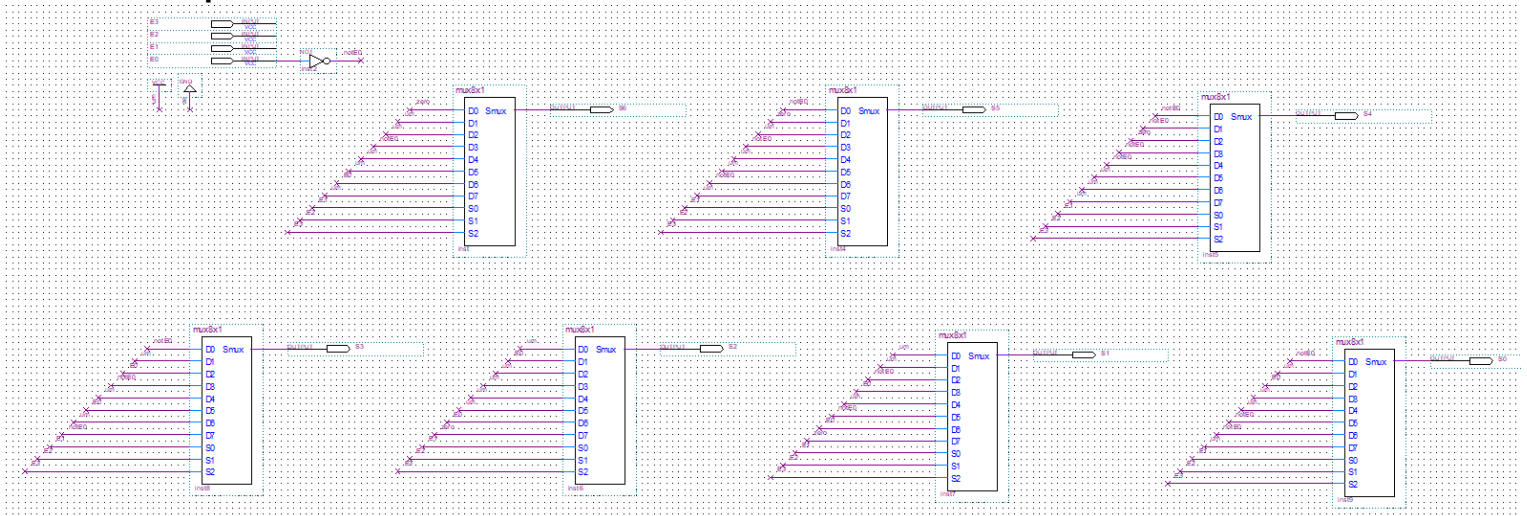
Nome: display

Descrição: Conjunto de sete multiplexadores 8x1 contendo uma lógica para a exibição dos números de 0 a 15 em hexadecimal.

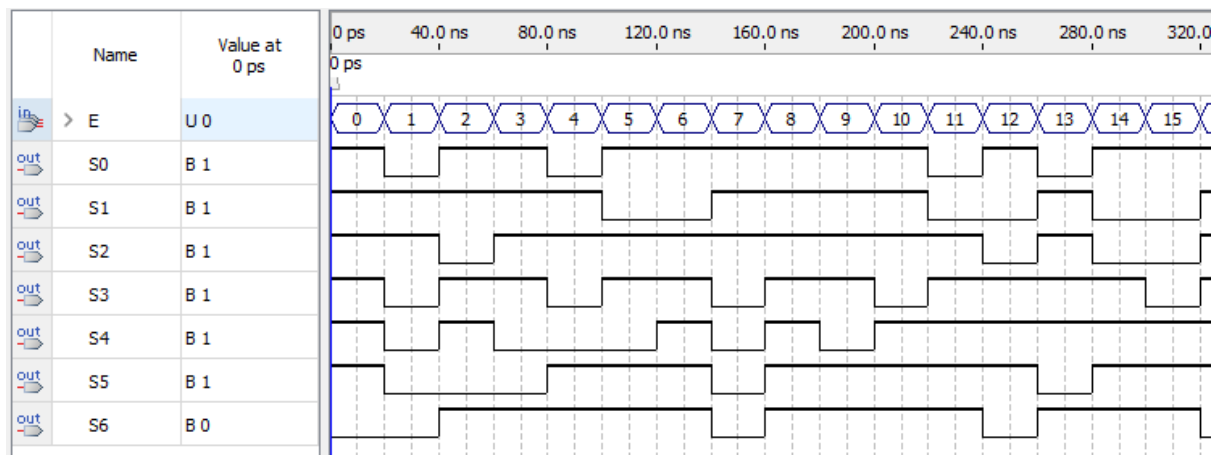
Símbolo:



Esquemático:



Simulação:



Nome: neander_placa

Descrição: Neander completo na placa.

Esquemático:

