

## Comparador de Números Binários com 2 Bits

**Aluno: Gabriel Cavalcanti Coelho**

Esta atividade tem como objetivo implementar, simular e analisar um circuito comparador de números binários com 2 bits utilizando a linguagem SystemVerilog. O circuito foi descrito utilizando submódulos para as portas lógicas NOT, XOR e AND, conforme especificado no enunciado:

```
module porta_not (
    input A,
    output X
);

    assign X = ~A;

endmodule

module porta_xor (
    input A,
    input B,
    output X
);

    assign X = A^B;

endmodule

module porta_and (
    input A,
    input B,
    output X
);

    assign X = A&B;

endmodule

module comparador(
    input [1:0]A,
    input [1:0]B,
    output X
);
```

```

    logic [3:0]aux;

    porta_xor u0(A[0],B[0],aux[0]);
    porta_xor u1(A[1],B[1],aux[1]);
    porta_not u2(aux[0],aux[2]);
    porta_not u3(aux[1],aux[3]);
    porta_and u4(aux[2],aux[3],X);

endmodule

// iverilog -g2012 -o vars_comparador tb_comparador.sv
// vvp vars_comparador
// gtkwave

```

Em seguida, foi criado um módulo testbench para testar o funcionamento do comparador:

```

`timescale 1ns/1ns
`include "comparador.sv"

module tb_comparador;

    logic [1:0]a;
    logic [1:0]b;
    logic x;

    comparador uut0(.A(a),.B(b),.X(x));

    initial begin
        $dumpfile("waveform.vcd");
        $dumpvars(0,tb_comparador);
        a[1] = 1;
        a[0] = 0;
        b[1] = 1;
        b[0] = 1;
        #10;
        a[1] = 0;
        a[0] = 1;
        b[1] = 0;
        b[0] = 1;
        #10;
        a[1] = 1;

```

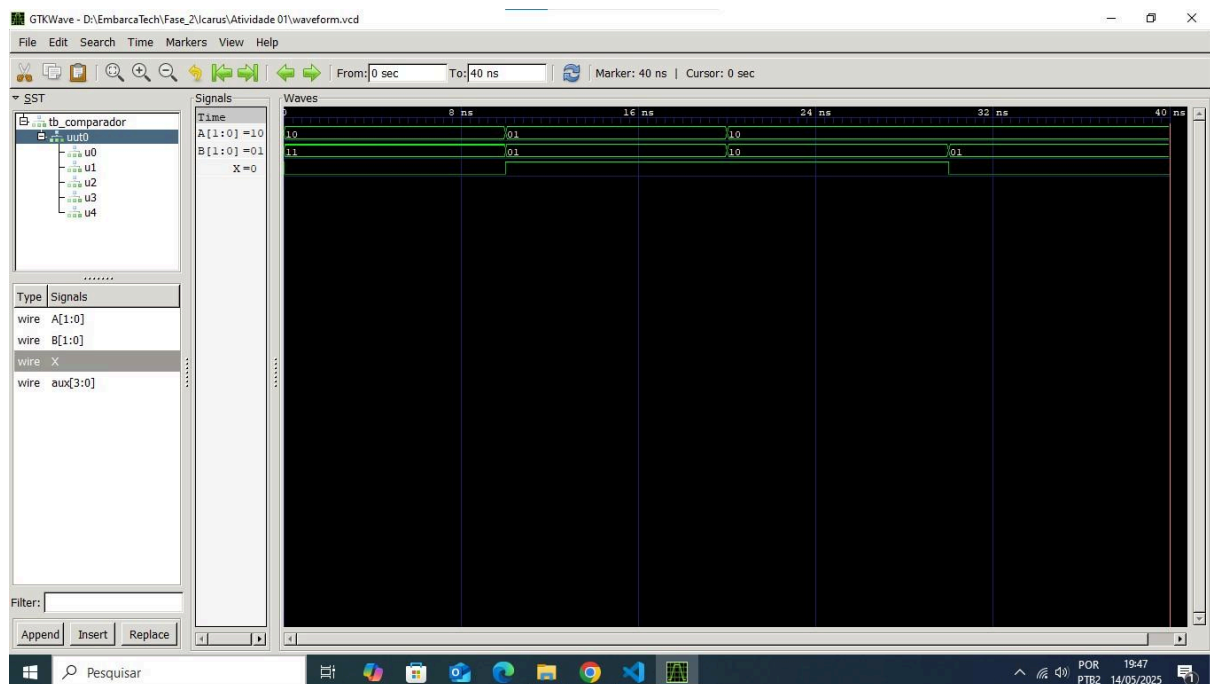
```

    a[0] = 0;
    b[1] = 1;
    b[0] = 0;
    #10;
    a[1] = 1;
    a[0] = 0;
    b[1] = 0;
    b[0] = 1;
    #10;
    $finish;
end

endmodule

```

A simulação foi realizada com o compilador Icarus Verilog e as formas de onda foram visualizadas com a ferramenta GTKWave. A imagem abaixo mostra as formas de onda geradas, evidenciando o funcionamento correto do circuito comparador:



Observa-se que as entradas A e B variam a cada 10 ns conforme os valores definidos no testbench. A saída X é ativada (valor 1) apenas quando os valores de A e B são iguais, como esperado para um comparador. Isso pode ser verificado nos intervalos de 10 ns a 20 ns e de 20 ns a 30 ns, onde  $A = B$ . Nos demais intervalos, a saída permanece em 0, indicando desigualdade.

Link do repositório no GitHub com todos os arquivos:

[https://github.com/GabrielC248/FPGAs\\_Atividade\\_01.git](https://github.com/GabrielC248/FPGAs_Atividade_01.git)