Universidade Católica de Santos
Ciências da Computação

Professores: Walter Oliveira e Vitor Kuribara
walter.oliveira@unisantos.com
Disciplina: Circuitos Digitais

Prática 2: Circuito Multiplexador e Demultiplexador

Nesta prática tem como o objetivo de estudar os tipos de codificação de Verilog na construção de módulos multiplexadores e demultiplexadores.

(1) Estilos de Codificação

Existem três tipos de estilos de codificação, que se diferem em relação ao nível de abstração:

1.) Comportamental: Com maior nível de abstração, descreve o circuito em forma de algoritmos, onde os procedimentos são realizados passo-a-passo

2.) Fluxo de Dados: Nível intermediário de abstração, efetua a descrição do circuito por meio da transferência de dados.

3.) Estrutural: Menor nível de abstração, descreve o circuito por à nível de componentes (por exemplo, portas lógicas), ou dispositivos, e realiza a interconexão entre eles.

Comportamental

```
//(...)
      input  s, a0, a1;
      output reg y;
      always @ (s or a0 or a1) begin
               if(s == 1'b0)
                        y = a0;
               else
                        y = a1;
       end
//(...)
```

Fluxo de dados

```
//(...)
        input  s, a0, a1;
        output y;
        assign y = s & a0 | ~s & a1
//(...)
```

Estrutural

```
//(...)
      input  s, a0, a1;
      output y;
      wire   sn, a0_sn, a1_s;

      not i0 (sn, s);
      and i1 (a0_sn, a0, sn);
      and i2 (a1_s, a1, s);
      or  i3 (y, a0_sn, a1_s);
//(...)
```

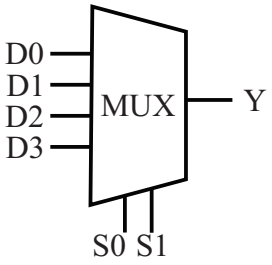
Figura 1: Estilos de Codificação

De acordo com o seu problema, cada estilo é mais adequado.

(2) Multiplexador:

O circuito multiplexador (também chamado de MUX) é um dos mais comuns em toda a circuitaria digital, sendo um dos blocos bases para sistemas mais complexos. Um multiplexador possui **um número de dados de entrada**, **uma ou mais entradas de seleção** e **uma saída**. Desta forma, **a saída do MUX corresponde a qual dado de entrada foi selecionado**.

Uma das principais aplicações do MUX está em controlar o fluxo de dados em sistemas, diminuindo o número de conexões necessárias para realizar esse controle. Por exemplo, a tabela-verdade de MUX com 4 dados de entrada e o bloco usual de um MUX é ilustrado na figura 2:



S1	S0	Y
0	0	D0
0	1	D1
1	0	D2
1	1	D3

Figura 2: Tabela-Verdade e bloco de um circuito MUX 4:1

Sendo assim, temos a seguinte função lógica associada ao MUX:

$$Y = D_0\overline{S_0}\overline{S_1} + D_1\overline{S_1}S_0 + D_2S_1\overline{S_0} + D_3S_0S_1$$

A partir da expressão lógica, é possível obter o seguinte circuito equivalente da figura 3:

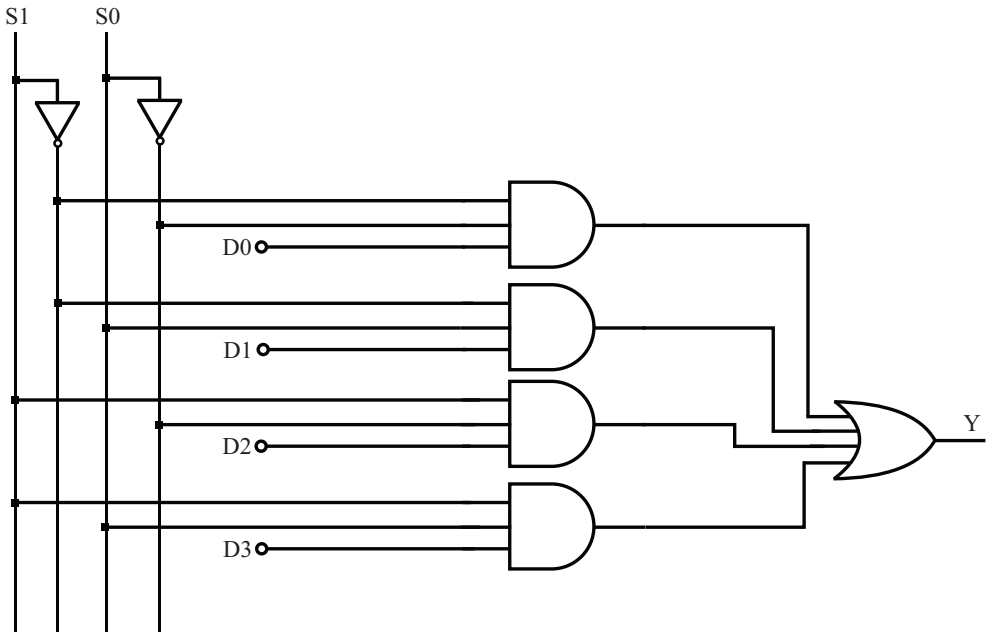
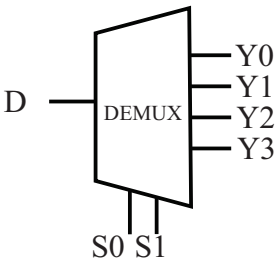


Figura 3: Circuito equivalente a um MUX de 4 entradas

(3) **Demultiplexador:** O circuito demultiplexador (ou DEMUX) realiza a função inversa do MUX. Ou seja, a partir de **uma entrada**, **uma ou mais entradas de seleção** e **diversas saídas**. Desta forma, a **saída do DEMUX corresponde em 0, para as saídas não selecionadas, e D, na saída selecionada**. Por exemplo, a tabela-verdade de DEMUX com 4 saídas e o bloco usual de um DEMUX é ilustrado na figura 4:



S1	S0	Y0	Y1	Y2	Y3
0	0	D	0	0	0
0	1	0	D	0	0
1	0	0	0	D	0
1	1	0	0	0	D

Figura 4: Tabela-Verdade e bloco de um circuito DEMUX 1:4

Sendo assim, temos a seguinte função lógica associada ao DEMUX para cada saída:

$$Y0 = D\overline{S_0}\overline{S_1}$$

$$Y1 = D\overline{S_1}S_0$$

$$Y2 = DS_1\overline{S_0}$$

$$Y3 = DS_0S_1$$

A partir da expressão lógica, é possível obter o seguinte circuito equivalente da figura 5:

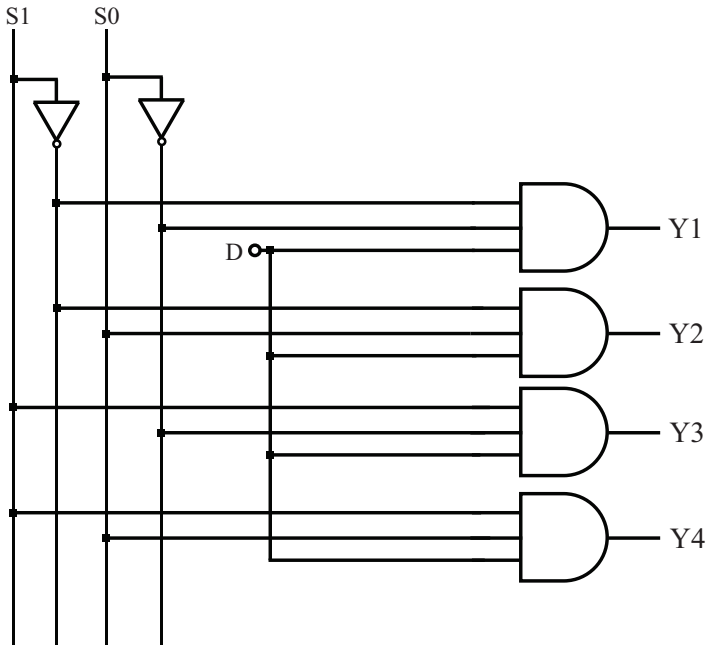


Figura 5: Circuito equivalente a um DEMUX de 4 saídas

(4) Tarefas:

1.) Em Verilog, realize a implementação de um MUX de 4 entradas utilizando os três estilos de codificação.

2.) Verifique o funcionamento dos módulos a partir um testbench e as formas de onda no GTKWave. Elas correspondem a função lógica descrita?

3.) Importe os módulos gerados nessa prática em http://digitaljs.tilk.eu/ e veja o esquemáticos gerados para cada um dos módulos. Eles são iguais? Porquê?

4.) Realize a implementação de um DEMUX de 4 entradas (escolha o estilo que achar conveniente) e verifique seu funcionamento no GTKWave.

5.) **Opcional**: Crie um módulo que conecte a saída de um MUX a entrada de um DEMUX, e interligue as entradas de seleção. O que acontece?