



Prof. Dr. Stefan Michael Blawid
Departamento de Engenharia de Computação
Centro de Informática

Tel. (81) 2126-8430 r: 4328

sblawid@cin.ufpe.br

<https://sites.google.com/a/cin.ufpe.br/if817/>

P1: Primeiros passos em Verilog

1. Dadas as declarações simultâneas da Verilog:

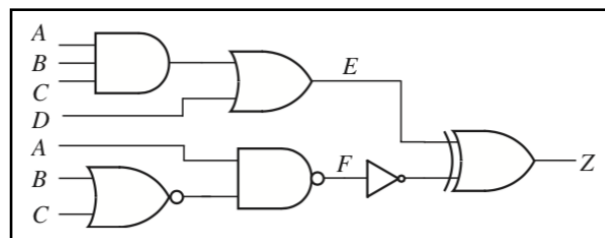
```
assign #3 B = A & C;
```

```
assign #2 C = !B;
```

(a) **(0.4p)** Desenhe o circuito representado pelas instruções.

(b) **(0.6p)** Desenhe um diagrama de tempo se inicialmente $A=B=0$ e $C=1$, e A mudar para 1 no tempo 5ns.

2. **(1p)** Escreva uma descrição Verilog do seguinte circuito combinacional usando instruções simultâneas. Cada porta tem um atraso de 5ns, excluindo o inversor, que tem um atraso de 2ns.



3. (a) **(2p)** Escreva pelo menos dois módulos Verilog diferentes que sejam equivalentes ao seguinte pseudo-código:

```
A = B1 when C = 1 else B2 when C = 2 else B3 when C = 3 else 0;
```

(b) **(1p)** Desenhe um circuito para implementar a seguinte declaração:

```
A = B1 when C1 = 1 else B2 when C2 = 1 else B3 when C3 = 1 else 0;
```

4. **(1p)** Supondo que B seja guiado pelo comando simulador:

`force B 0 0, 1 4, 0 10, 1 15, 0 20, 1 30, 0 40.` Desenhe um diagrama de tempo ilustrando A , B e C se as seguintes instruções simultâneas forem executadas:

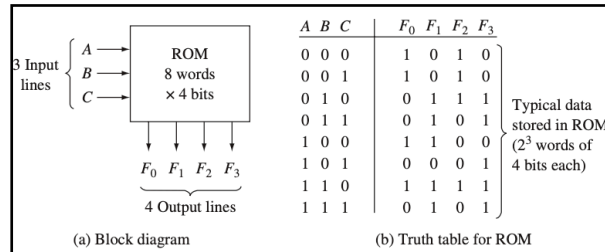
(a)

```
always @(A,B)
begin
C <= #5 A && B;
end
```

(b)

```
always @(A,B)
begin
C = #5 A && B;
end
```

5. Uma memória somente leitura (ROM) consiste em uma matriz de dispositivos semicondutores que são interconectados para armazenar uma matriz de dados binários. Uma vez que os dados binários são armazenados na ROM, eles podem ser lidos sempre que desejado, mas os dados armazenados não podem ser alterados em condições normais de operação. A figura abaixo mostra uma ROM que possui três linhas de entrada e quatro linhas de saída.



Uma ROM pode implementar qualquer circuito combinacional. Essencialmente, se as saídas de todas as combinações de entradas são armazenadas na ROM, as saídas podem ser “consultadas” na tabela armazenada na ROM. O método ROM também é chamado de método Look-Up Table (LUT) por esse motivo.

- (a) **(1p)** Escreva um módulo Verilog completo que realize a função $F = \bar{A}\bar{B} + B\bar{C}$ e $G = AC + \bar{B}$ usando uma ROM de palavras de 8 bits \times 2. O LUT deve obrigatoriamente ser armazenado em um vetor de elementos de dois bits.
- (b) **(1p)** Escreva um módulo Verilog para uma LUT com quatro entradas e três saídas. A saída de 3 bits deve ser um número binário igual ao número de 1s na entrada da LUT. Escreva uma bancada de teste para verificar o funcionamento correto do módulo e mostrar o resultado;
- (c) **(1p)** Escreva um módulo Verilog para um circuito que conte o número de 1s em um número de 12 bits. Use três dos módulos de (b).
- (d) **(1p)** Simule seu código e teste-o para as seguintes entradas de dados: 111111111111, 010110101101, 100001011100.