

# LÓGICA DIGITAL (1001351)

## EXPERIMENTO NR.5

### Somador de 1 bit com saída em display. <sup>1</sup>

---

## 1 Instruções Gerais

- Grupos definidos no AVA, só incluir os nomes de quem efetivamente participou;
- Ler atentamente todo o procedimento desta experiência antes de realizá-la;

## 2 Objetivos da Prática

- Implementação de um somador de 1 bit utilizando como saída o display de 7 segmentos.
- Realizar a hierarquia de módulos.
- Implementação do projeto no Kit de desenvolvimento Zybo Z7-20.

## 3 Materiais, Equipamentos e Arquivos

- Kit de desenvolvimento Zybo Z7-20,
- Módulo de Pmod SSD com dois display de 7 segmentos,

## 4 Fundamentos teóricos

### 4.1 Módulo meio somador - *adder*

Na Figura 1 é apresentado o circuito do módulo meio somador com sua tabela da verdade, equação booleana, e a sua forma de representação. O código Verilog do circuito da Figura 1 é dado pelo Código 1.

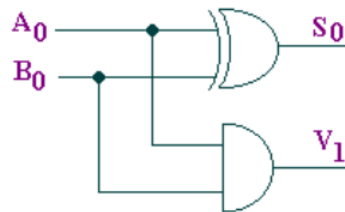
---

<sup>1</sup>Revisão 11/07/2022: Prof. Mauricio Figueiredo e Prof. Ricardo Menotti.

	$A_0$	$B_0$	$S_0$	$V_1$
$A_0$	0	0	0	0
$B_0$	0	1	1	0
$V_1$	1	0	1	0
$S_0$	1	1	0	1

$$S_0 = \bar{A} \cdot B + A \cdot \bar{B} = A \oplus B$$

$$V_1 = A \cdot B$$



Representação:



Figura 1: Módulo meio somador - adder.

```

1 module adder(
2     input a, b,
3     output s0, s1);
4
5     assign s0 = a ^ b;
6     assign s1 = a & b;
7 endmodule

```

Código 1: Código Verilog de um módulo meio somador.

## 4.2 Display de 7 segmentos

Na Figura 2 é apresentado o display de 7 segmentos. O código Verilog que realiza a decodificação dos bits de entrada para o sete segmentos do display da Figura 2 é dado pelo Código 2. Note que esta decodificação está simplificada apenas para 00, 01, 10 e 11.

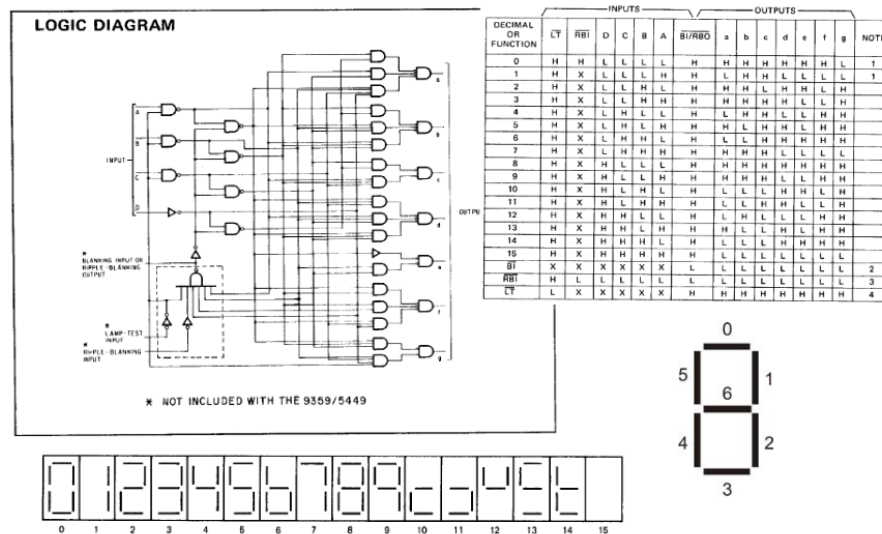


Figura 2: Display de 7 segmentos.

```

1 module display(
2     input s1, s0,
3     output a, b, c, d, e, f, g);
4
5     assign a = ~s0;
6     assign b = 1'b1;
7     assign c = ~s1;
8     assign d = ~s0;
9     assign e = ~s0;
10    assign f = ~s1 & ~s0;
11    assign g = s1 & ~s0;
12 endmodule

```

Código 2: Código Verilog para decodificar o resultado de um meio somador.

## 5 Procedimentos Experimentais

Deseja-se implementar um somador de 1 bit utilizando como saída o display de 7 segmentos como mostrado na Figura 3.

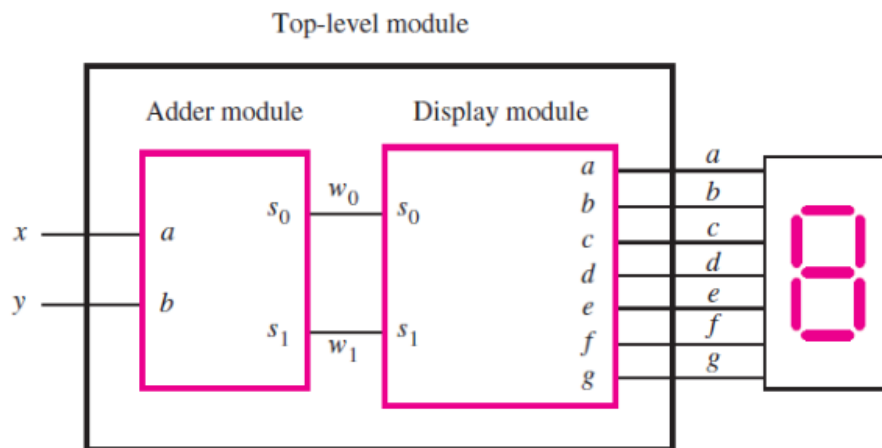


Figura 3: Somador de 1 bit com saída em display.

Para isso, siga os seguintes passos:

1. Crie um projeto novo no Vivado utilizando a placa Zybo Z7-20.
2. Crie o arquivo `adder.v`, e implemente módulo somador conforme Código 1.
3. Crie o arquivo `display.v`, e implemente módulo display conforme Código 2.
4. Crie o arquivo `adder_display.v` conforme Código 3. Instancie nele os outros dois módulos e faça as ligações correspondentes.

```

1 module adder_display(x, y, a, b, c, d, e, f, g );
2     input x, y;
3     output a, b, c, d, e, f, g;
4     // Instancie e ligue os componentes abaixo
5
6 endmodule

```

Código 3: Template para desenvolvimento do módulo top-level do somador com display.

5. Adicione o arquivo de restrição `ZYBO_z7_Master.xdc` e faça as configurações necessárias para que as entradas  $x$  e  $y$  sejam chaves; e  $a$ ,  $b$ ,  $c$ ,  $d$ ,  $e$ ,  $f$ , e  $g$  acionem o módulo Pmod SSD que será conectado nas portas JC e JD do kit de desenvolvimento Zybo Z7-20. Adicione o código alterado no relatório, e explique as alterações.

6. Conecte o módulo Pmod SSD nas portas JC e JD do kit de desenvolvimento Zybo Z7-20.
7. Gere o arquivo de bitstream do Código 3 e grave no kit de desenvolvimento Zybo Z7-20.
8. Envie apenas o arquivo *top-level* no AVA, incluindo os nomes dos participantes como comentário no início dele.

Etapa a): Se terminou durante a aula, chame o professor para corrigir;

Etapa b): Senão, envie também – como comentário no código – link para um vídeo demonstrando o correto funcionamento.

## Referências

[1] <https://digilent.com/reference/pmod/pmodssd/reference-manual>

[2] <https://digilent.com/reference/programmable-logic/zybo-z7/start>