

LÓGICA DIGITAL (1001351)

EXPERIMENTO NR.3

Implementação de circuitos em Verilog para FPGAs. ¹

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

- Introdução à ferramenta Xilinx Vivado 2017.4;
- Manipulação introdutória de arquivos básicos correspondentes ao código de descrição de circuito e arquivo de restrições (*constraints*);
- Execução de experimentos no kit Digilent Zybo Z7-20.

3 Materiais e Equipamentos

- Xilinx Vivado 2017.4;
- Kit Digilent Zybo Z7-20.

4 Orientações para a prática no laboratório

1. Observe atentamente os códigos fornecidos e seu comportamento antes de alterar qualquer coisa;
2. Faça uma cópia dos códigos a serem modificados, assim você pode comparar com os originais sempre que for preciso;
3. Procure associar cada linha do código ao circuito que você está implementando/-testando;
4. Faça alterações por etapas, assim você não corre o risco de modificar muito o código de uma vez e inserir vários erros difíceis de detectar.

¹Prof. Mauricio Figueiredo e Prof. Ricardo Menotti (Revisão: 19 de agosto de 2022)

5 Fundamentos teóricos

Nesta seção, será descrito o básico sobre a ferramenta e a linguagem Verilog necessários para o seu laboratório.

5.1 Xilinx Vivado

Esta ferramenta cobre o fluxo completo para desenvolvimento e prototipação de hardware usando FPGAs:

- i) **Síntese:** geração do circuito RTL² a partir da descrição fornecida;
- ii) **Simulação funcional:** geração de estímulos e verificação das saídas apresentadas (*não usada neste exercício*);
- iii) **Implementação:** alocação o roteamento dos recursos necessários no dispositivo alvo para implementar o circuito desejado;
- iv) **Simulação/Análise temporal:** medida dos tempos de resposta do circuito e frequência máxima de operação (*não usada neste exercício*);
- v) **Programação:** envio do *bitstream* para o dispositivo por meio de um cabo.

Na indústria, geralmente são usadas ferramentas distintas para cada etapa do projeto.

5.2 Verilog

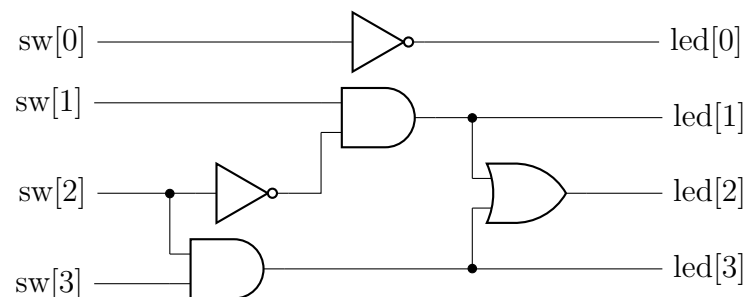
Como já mencionamos antes, o Verilog é apenas uma ferramenta que usamos para compreender a lógica digital e seus circuitos. Nosso objetivo não contempla compreender toda a sintaxe da linguagem durante o curso. Um subconjunto dela já é suficiente para atingir o nosso objetivo.

Nosso exemplo inicial usa barramentos (bus) como entradas e saídas, eles são como arranjos de fios simples e possuem um índice para acessar cada fio separadamente. No exemplo a seguir, tanto a entrada quanto a saída possuem 4 fios cada. Vamos atribuir todas juntas inicialmente, ou seja, cada chave vai comandar um led.

```
1 module top (  
2     input [3:0] sw,  
3     output [3:0] led);  
4     assign led = sw;  
5 endmodule
```

Após testar o código inicial na placa, modifique-o para que corresponda ao circuito a seguir:

²Resistor-Transistor Logic



Agora os leds precisam ser atribuídos individualmente, como no modelo a seguir. Atribua as expressões para cada um dos leds e teste novamente:

```

1 module top (
2     input [3:0] sw,
3     output [3:0] led);
4     assign led[0] = ~sw[0];
5
6     ...
7
8 endmodule
  
```

6 Procedimentos Experimentais

Passo 1: Crie um novo projeto no Vivado:

Etapa a): Use o primeiro código fornecido como *top-level*;

Etapa b): Adicione o arquivo de *constraints*;

Etapa c): Faça a síntese/implementação;

Etapa d): Programe a placa e verifique o funcionamento.

Passo 2: Modifique o código para corresponder ao circuito fornecido:

Etapa a): Use o modelo e complete-o para modificar o *top-level*;

Etapa b): Faça a síntese/implementação novamente;

Etapa c): Reprograme a placa e verifique o funcionamento.

Passo 3: 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.