



UNIVERSIDADE FEDERAL DE VIÇOSA - CAMPUS  
FLORESTAL  
INSTITUTO DE CIÊNCIAS EXATAS E  
TECNOLOGICAS  
CIÊNCIA DA COMPUTAÇÃO  
INTRODUÇÃO AOS SISTEMAS LÓGICOS DIGITAIS

## **LÓGICA COMBINACIONAL**

GUILHERME SÉRGIO 3854

LARISSA ISABELLE 3871

OTÁVIO SANTOS 3890

PEDRO CARDOSO 3877

FLORESTAL

2019

# 1. SUMÁRIO

|    |                                           |    |
|----|-------------------------------------------|----|
| 2. | INTRODUÇÃO .....                          | 3  |
| 3. | MÓDULO DE TRANSCRIÇÃO .....               | 4  |
| a. | TABELA VERDADE .....                      | 4  |
| b. | MINTERMOS E MAXTERMOS .....               | 4  |
| c. | SIMPLIFICAÇÃO COM MAPAS DE KARNAUGH ..... | 5  |
| d. | EXPRESSÕES SIMPLIFICADAS.....             | 7  |
| e. | CIRCUITO RESULTANTE .....                 | 7  |
| 4. | MÓDULO DE SIMULAÇÃO .....                 | 9  |
| 5. | IMPLEMENTAÇÃO NA FPGA .....               | 10 |
| 6. | CONCLUSÃO .....                           | 11 |

## **2. INTRODUÇÃO**

Este relatório consiste no primeiro trabalho prático da disciplina de Introdução aos Sistemas Lógicos Digitais. Neste trabalho será desenvolvida pelo grupo a lógica combinacional, usando para isso recursos da lógica de Boole e simplificações das expressões em circuitos lógicos desenvolvidos pelo programa Logisim, assim como a implementação do módulo de testes e geração das formas de ondas. Por fim, implementaremos o módulo de síntese e utilizaremos o código em uma FPGA.

### 3. MÓDULO DE TRANSCRIÇÃO

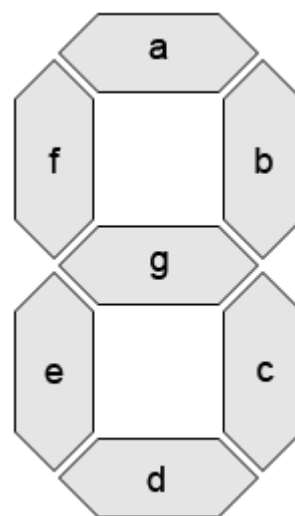
#### a. TABELA VERDADE

Consideramos o nível lógico 1 para o “Tom” alto e o nível lógico 0 para o “Tom” baixo. As entradas são: W, X, Y, Z. Quanto às sete saídas, atribuímos os nomes de a, b, c, d, e, f, g com base na figura ao lado da tabela.

OBS: Como descrito na especificação, o nível lógico 0 terá o status de aceso e o nível lógico 1 apagado. Além disso, a entrada 0 foi considerada inválida conforme solicitado.

Tabela Verdade das entradas W (Tom) e X, Y, Z (3 bits).

| W | X | Y | Z | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |



#### b. MINTERMOS E MAXTERMOS

Representação por Mintermos:

$$a \rightarrow \sum m = (0, 1, 2, 3, 4, 5, 6, 7, 8)$$

$$b \rightarrow \sum m = (0, 1, 2, 4, 8, 9, 12, 13, 14)$$

$$c \rightarrow \sum m = (0, 1, 3, 4, 5, 7, 8, 12, 15)$$

$$d \rightarrow \sum m = (0, 6, 8, 9, 10, 11, 13, 14, 15)$$

$$e \rightarrow \sum m = (0, 4, 6, 7, 8, 11, 12, 14)$$

$$f \rightarrow \sum m = (0, 1, 2, 3, 6, 8, 9, 11, 12)$$

$$g \rightarrow \sum m = (0, 5, 8, 10, 11, 13)$$

Representação por Maxtermos:

$$a \rightarrow \prod M = (9, 10, 11, 12, 13, 14, 15)$$

$$b \rightarrow \prod M = (3, 5, 6, 7, 10, 11, 15)$$

$$c \rightarrow \prod M = (2, 6, 9, 10, 11, 13, 14)$$

$$d \rightarrow \prod M = (1, 2, 3, 4, 5, 7, 12)$$

$$e \rightarrow \prod M = (1, 2, 3, 5, 9, 10, 13, 15)$$

$$f \rightarrow \prod M = (4, 5, 7, 10, 13, 14, 15)$$

$$g \rightarrow \prod M = (1, 2, 3, 4, 6, 7, 9, 12, 14, 15)$$

### c. SIMPLIFICAÇÃO COM MAPAS DE KARNAUGH

Foi escolhido pelo grupo usar a representação de soma de produtos.

| YZ \ WX | 00 | 01 | 11 | 10 |
|---------|----|----|----|----|
| 00      | 1  | 1  | 1  | 1  |
| 01      | 1  | 1  | 1  | 1  |
| 11      |    |    |    |    |
| 10      | 1  |    |    |    |

Saída “a”  $\rightarrow X'Y'Z' + W'$

| YZ \ WX | 00 | 01 | 11 | 10 |
|---------|----|----|----|----|
| 00      | 1  | 1  |    | 1  |
| 01      | 1  |    |    |    |
| 11      | 1  | 1  |    | 1  |
| 10      | 1  | 1  |    |    |

Saída “b”  $\rightarrow W'X'Z' + X'Y' + Y'Z' + WXZ' + WY'$

| YZ \ WX | 00 | 01 | 11 | 10 |
|---------|----|----|----|----|
| 00      | 1  | 1  | 1  |    |
| 01      | 1  | 1  | 1  |    |
| 11      | 1  |    | 1  |    |
| 10      | 1  |    |    |    |

Saída “c”  $\rightarrow Y'Z' + W'Z + XYZ$

| YZ \ WX | 00 | 01 | 11 | 10 |
|---------|----|----|----|----|
| 00      | 1  |    |    |    |
| 01      |    |    |    | 1  |
| 11      |    | 1  | 1  | 1  |
| 10      | 1  | 1  | 1  | 1  |

Saída “d”  $\rightarrow X'Y'Z' + WZ + WX' + XYZ'$

| YZ \ WX | 00 | 01 | 11 | 10 |
|---------|----|----|----|----|
| 00      | 1  |    |    |    |
| 01      | 1  |    | 1  | 1  |
| 11      | 1  |    |    | 1  |
| 10      | 1  |    | 1  |    |

Saída “e”  $\rightarrow Y'Z' + WX'YZ + W'XY + XYZ'$

| YZ \ WX | 00 | 01 | 11 | 10 |
|---------|----|----|----|----|
| 00      | 1  | 1  | 1  | 1  |
| 01      |    |    |    | 1  |
| 11      | 1  |    |    |    |
| 10      | 1  | 1  | 1  |    |

Saída “f”  $\rightarrow W'X' + W'YZ' + WY'Z' + WX'Z$

| YZ \ WX | 00 | 01 | 11 | 10 |
|---------|----|----|----|----|
| 00      | 1  |    |    |    |
| 01      |    | 1  |    |    |
| 11      |    | 1  |    |    |
| 10      | 1  |    | 1  | 1  |

Saída "g"  $\rightarrow X'Y'Z' + XY'Z + WX'Y$

#### d. EXPRESSÕES SIMPLIFICADAS

$$a \rightarrow X'Y'Z' + W'$$

$$b \rightarrow W'X'Z' + W'X'Y' + Y'Z' + WXZ' + WY'$$

$$c \rightarrow Y'Z' + W'Z + XYZ$$

$$d \rightarrow X'Y'Z' + WZ + WX' + XYZ'$$

$$e \rightarrow Y'Z' + WX'YZ + W'XY + XYZ'$$

$$f \rightarrow W'X' + W'YZ' + WY'Z' + WX'Z$$

$$g \rightarrow X'Y'Z' + XY'Z + WX'Y$$

#### e. CIRCUITO RESULTANTE

Após a simplificação das expressões booleanas usando mapas de Karnaugh, o circuito abaixo foi implementado utilizando a ferramenta Logisim.

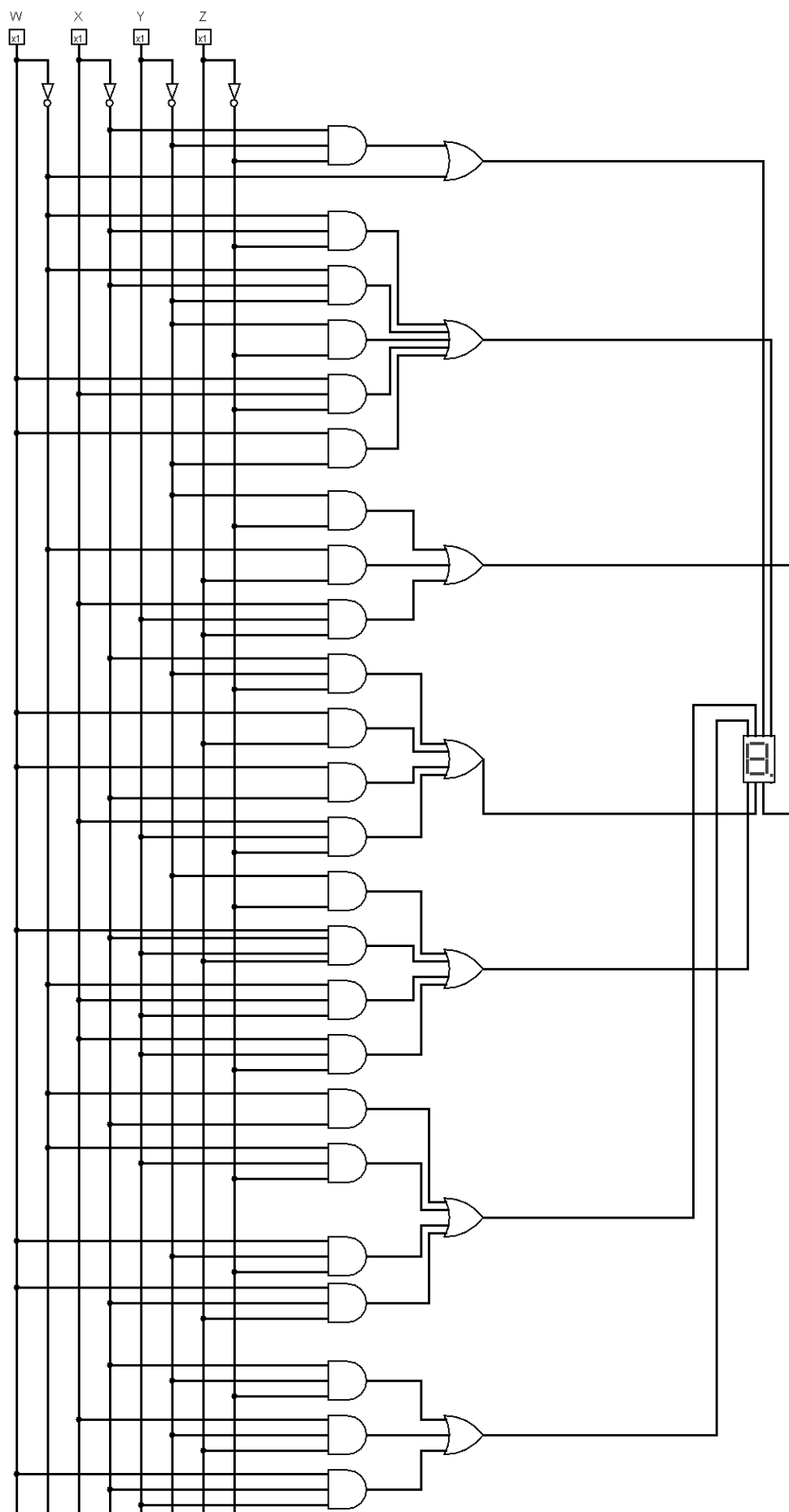


Figura 1: Circuito gerado no Logisim



A partir do circuito resultante foi escrito o código em Verilog de descrição do circuito.

## 4. MÓDULO DE SIMULAÇÃO

Após a implementação do módulo de simulação no Verilog, o código foi testado e, utilizando o comando \$monitor, foi possível observar as alterações nos valores das variáveis, comprovando que seu resultado é idêntico à tabela verdade montada no início do trabalho.

```
[Running] tpISL.v
VCD info: dumpfile simula_traducao.vcd opened for output.
0 W=0 X=0 Y=0 Z=0 | a=1 b=1 c=1 d=1 e=1 f=1 g=1
1 W=0 X=0 Y=0 Z=1 | a=1 b=1 c=1 d=0 e=0 f=1 g=0
2 W=0 X=0 Y=1 Z=0 | a=1 b=1 c=0 d=0 e=0 f=1 g=0
3 W=0 X=0 Y=1 Z=1 | a=1 b=0 c=1 d=0 e=0 f=1 g=0
4 W=0 X=1 Y=0 Z=0 | a=1 b=1 c=1 d=0 e=1 f=0 g=0
5 W=0 X=1 Y=0 Z=1 | a=1 b=0 c=1 d=0 e=0 f=0 g=1
6 W=0 X=1 Y=1 Z=0 | a=1 b=0 c=0 d=1 e=1 f=1 g=0
7 W=0 X=1 Y=1 Z=1 | a=1 b=0 c=1 d=0 e=1 f=0 g=0
8 W=1 X=0 Y=0 Z=0 | a=1 b=1 c=1 d=1 e=1 f=1 g=1
9 W=1 X=0 Y=0 Z=1 | a=0 b=1 c=0 d=1 e=0 f=1 g=0
10 W=1 X=0 Y=1 Z=0 | a=0 b=0 c=0 d=1 e=0 f=0 g=1
11 W=1 X=0 Y=1 Z=1 | a=0 b=0 c=0 d=1 e=1 f=1 g=1
12 W=1 X=1 Y=0 Z=0 | a=0 b=1 c=1 d=0 e=1 f=1 g=0
13 W=1 X=1 Y=0 Z=1 | a=0 b=1 c=0 d=1 e=0 f=0 g=1
14 W=1 X=1 Y=1 Z=0 | a=0 b=1 c=0 d=1 e=1 f=0 g=0
15 W=1 X=1 Y=1 Z=1 | a=0 b=0 c=1 d=1 e=0 f=0 g=0
[Done] exit with code=0 in 0.234 seconds
```

Figura 2: Resultado do teste do módulo de simulação Verilog

Dessa forma, gerado o arquivo simula\_traducao.vcd e rodando-o no GTKWave, obteve-se o resultado exibido abaixo, condizente com o que era esperado.



Figura 3: Formas de ondas gerados pelo GTKWave

## 5. IMPLEMENTAÇÃO NA FPGA

Após concluídos os passos anteriores e verificado que o circuito estava com o funcionamento desejado, restou apenas utilizar, o circuito na FPGA. Para isso usamos programas como o Quartus e o System Builder para agilizar o processo, criar o módulo de síntese e compilar o código. É possível ver abaixo um exemplo do funcionamento do circuito na FPGA:

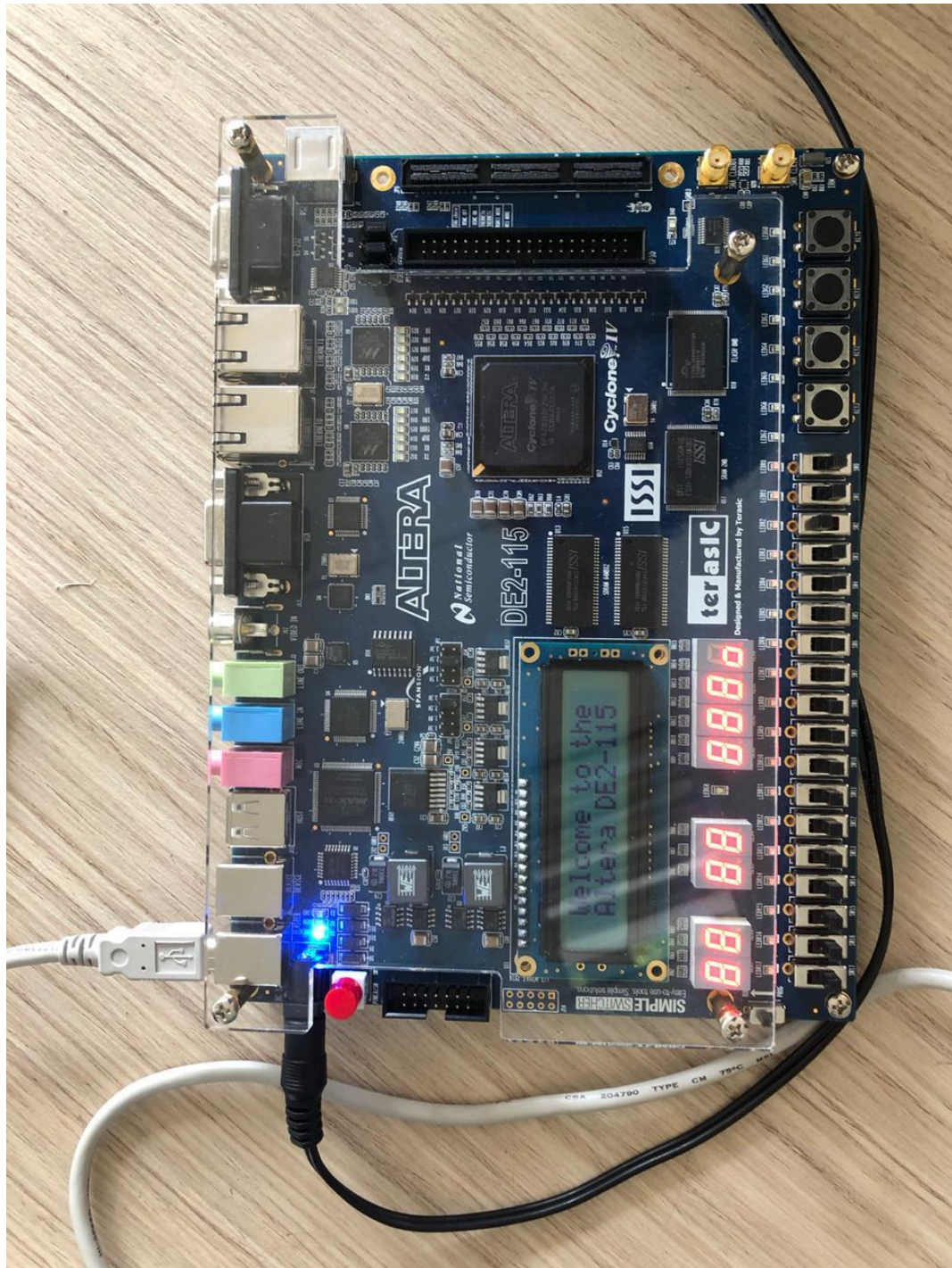


Figura 4: Funcionamento do circuito na FPGA

## **6. CONCLUSÃO**

Ao fim deste trabalho, concluímos o projeto de construção circuito de lógica combinacional proposto, para isso utilizamos conceitos teóricos, como simplificação de expressões, e também conceitos mais práticos, como na implementação dos módulos em uma linguagem de descrição de hardware. Assim, a elaboração do trabalho se mostrou de extrema importância, pois além de aprimorar os conceitos já conhecidos, exigiu a busca por mais conhecimento.