



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
CENTRO DE TECNOLOGIA - CT
DEPARTAMENTO DE ENGENHARIA DE COMPUTAÇÃO E AUTOMAÇÃO

CIRCUITOS COMBINACIONAIS
DCA0212.1 - Grupo 09 - Laboratório 02

Igor Michael Araujo de Macedo
Isaac de Lyra Junior
Paulo Costa Braga
Pedro Henrique de Freitas Silva

Natal, 19 de novembro de 2021

1 INTRODUÇÃO

O presente relatório tem por objetivo desenvolver os conceitos teóricos de circuitos digitais de forma prática. Neste trabalho serão abordados os conceitos de circuitos combinacionais; para isso, usaremos outros conceitos como os de portas lógicas, equações booleanas, simplificações, dentre outras. O *software* ModelSim será utilizado para construir uma simulação digital dos circuitos que serão abordados, onde serão praticados estes conceitos com a linguagem de descrição de *hardware* VHD.

Neste trabalho, desejamos projetar um circuito que conta o número de bits iguais a 1 presente em três entradas (A, B, C) e, como saída, fornece esse número em binário, por meio de duas saídas S1 e S2.

2 DESNVOLVIMENTO

O desenvolvimento do projeto se dá primeiramente na especificação das entradas e saídas do circuito. Como o número de 1's pode variar no intervalo $[0, 3]$, precisamos de dois bits para a representação das três entradas, S2 e S1, sendo S2 o bit mais significativo.

Para prosseguir com o desenvolvimento do projeto, construímos a tabela verdade do circuito explicitando todas as entradas e todas as saídas que o circuito pode assumir. Como o circuito é um contador de 1's, se tivermos as entradas ABC como 000, o circuito deve devolver 00 como resposta, caso as entradas sejam 001, o circuito deve devolver 01 como resposta, e assim por diante. A tabela verdade do circuito pode ser observada na Figura 1.

Figura 1 – Tabela verdade do circuito contador de 1's.

Entradas			Saídas	
A	B	C	S2	S1
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Fonte: Elaboração própria.

Depois disso, foram retiradas, da tabela verdade, as equações de cada saída do circuito e simplificadas utilizando os postulados. As equações simplificadas podem ser vistas na Figura 2.

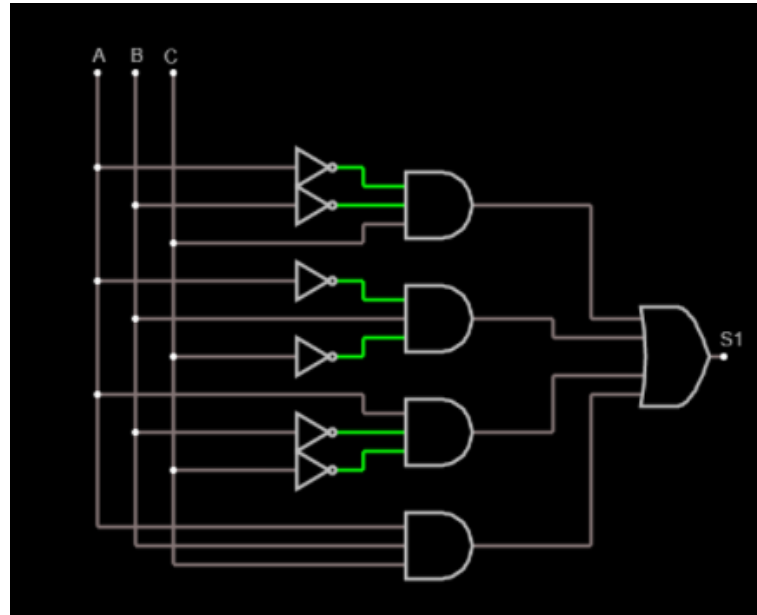
Figura 2 – Equações booleanas simplificadas das saídas do circuito.

Saída	Equação booleana
S1	$A'B'C + A'BC' + AB'C' + ABC$
S2	$BC + AC + AB$

Fonte: Elaboração própria.

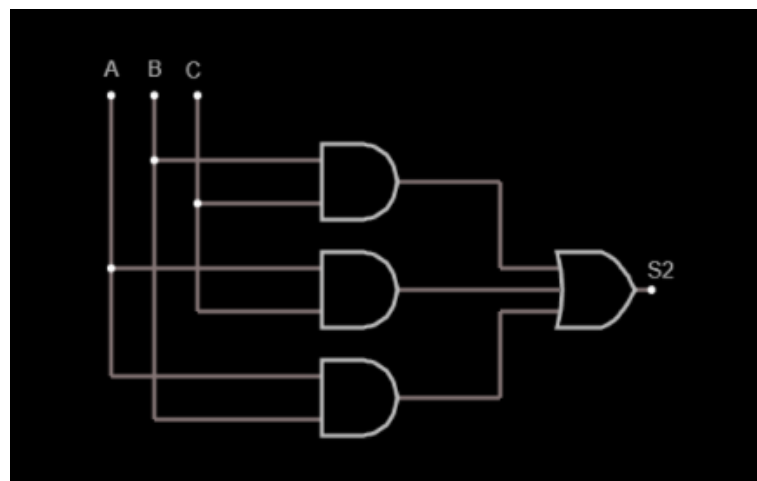
Feito isso, foi representado o esquemático do circuito, desenhando a combinação de cada saída. Estas representações podem ser observadas nas Figuras 3 e 4.

Figura 3 – Circuito combinacional para a saída S1.



Fonte: Elaboração própria.

Figura 4 – Circuito combinacional para a saída S2.



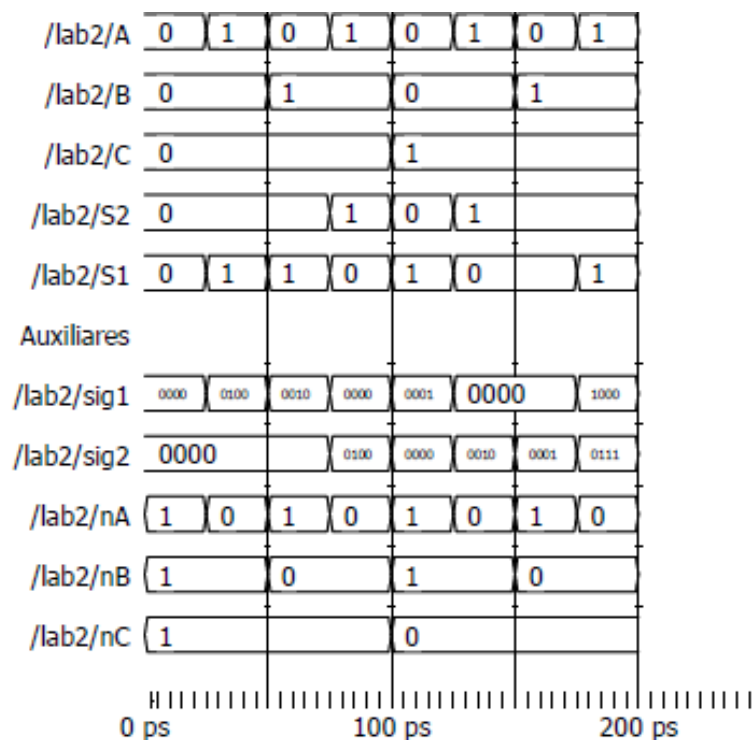
Fonte: Elaboração própria.

Em seguida, o próximo passo foi descrever o circuito em VHDL. Para isso, foi utilizado a ideia de componentes para facilitar o processo. Nas equações booleanas retiradas da tabela verdade, há operações lógicas AND entre três entradas e operação OR entre 4 entradas, sendo assim, dois componentes foram criados, um para cada um desses dois tipos de portas. Eles foram utilizados na arquitetura principal que descreve o circuito contador de 1's. O código fonte completo pode ser visto no Anexo A.

3 RESULTADOS

Por fim, a última etapa do trabalho foi fazer a simulação do circuito descrito em VHDL. Para isso, também foi utilizado o *software* ModelSim e o código para a simulação pode ser visto no Anexo B. Nele é forçada às entradas A, B e C todas as combinações possíveis a fim de testar todas as possibilidades nas saídas S2 e S1. Na Figura 5 é possível ver o resultado.

Figura 5 – Simulação do circuito no ModelSim.



Fonte: Elaboração própria.

Os sinais abaixo de “Auxiliares” têm finalidade apenas de, como o próprio nome já diz, auxiliar durante o processo de desenvolvimento do circuito.

Observando os valores de S2 e S1, sendo S2 o mais significativo, é possível notar que o resultado bate com o esperado, por exemplo: quando ABC=000, implica em S2S1=00; quando ABC=101, implica em S2S1=10; quando ABC=111, implica em S2S1=11; e assim por diante.

4 CONCLUSÃO

O trabalho teve como objetivo pôr em prática os conhecimentos estudados sobre circuitos combinacionais e testar a implementação e simulação destes por meio do VHDL. Para isso, foi sugerido um projeto de um circuito que recebe três entradas e duas saídas, de 1 bit cada, onde a saída indica a quantidade de 1's (níveis lógicos altos) da entrada. Para a solução foi construída uma tabela verdade para o problema, desenvolvida e simplificada as equações booleanas, desenhado o esquemático do circuito, desenvolvido em VHDL e simulado. Após todo o processo, a simulação retornou os resultados esperados e, com isso, podemos concluir que foi possível extrair o máximo de conhecimento e proveito do problema.

Referências

VAHID, F. *Sistemas Digitais: Projeto, Otimização e HDLs*. [S.l.]: Artmed Bookman, 2008.

ANEXO A – Código VHDL

```

— AND_3 —
entity AND_3 is
    port(A, B, C: in bit;
          S: out bit);
end AND_3;

architecture logic of AND_3 is

    begin
        S <= A and B and C;
end logic;

— OR_4 —
entity OR_4 is
    port(A, B, C, D: in bit;
          S: out bit);
end OR_4;

architecture logic of OR_4 is

    begin
        S <= A or B or C or D;
end logic;

— CIRCUITO CONTADOR DE 1s —
entity lab2 is
    port(A,B,C: in bit;
          S2, S1: out bit);
end lab2;

architecture logic of lab2 is

    component AND_3
        port(A, B, C: in bit;
              S: out bit);
    end component;

    component OR_4
        port(A, B, C, D: in bit;
              S: out bit);
    end component;

```



```
signal sig1,sig2: bit_vector(3 downto 0);
signal nA,nB,nC: bit;

begin

nA <= NOT A;
nB <= NOT B;
nC <= NOT C;

SIGNAL1: AND_3 port map(nA, nB, C, sig1(0));
SIGNAL2: AND_3 port map(nA, B, nC, sig1(1));
SIGNAL3: AND_3 port map(A, nB, nC, sig1(2));
SIGNAL4: AND_3 port map(A, B, C, sig1(3));

SAIDA1: OR_4 port map (sig1(0), sig1(1), sig1(2), sig1(3), S1);

SIGNAL5: AND_3 port map('1', B, C, sig2(0));
SIGNAL6: AND_3 port map(A, '1', C, sig2(1));
SIGNAL7: AND_3 port map(A, B, '1', sig2(2));

SAIDA2: OR_4 port map(sig2(0), sig2(1), sig2(2), '0', S2);

end logic;
```

ANEXO B – Código para simulação

```
vsim lab2

add wave *

force A 0 0, 1 25 -repeat 50
force B 0 0, 1 50 -repeat 100
force C 0 0, 1 100

run 200
```