



CIÊNCIA DA COMPUTAÇÃO

CIRCUITOS DIGITAIS I (6878)

Trabalho: Projeto de circuito combinacional - Parte I

Data: 22/11/2019

Professor: Nardênio Almeida Martins

Discentes

R.A.	Nome
112679	Guilherme Panobianco Ferrari
115735	Sergio Alvarez da Silva Junior



SUMÁRIO

1. Introdução e Objetivos.....	3
1.1) <i>Introdução.....</i>	<i>3</i>
1.2) <i>Objetivos de Experiência.....</i>	<i>3</i>
2. Fundamentação Teórica.....	3
2.1) <i>Componentes Utilizados.....</i>	<i>3</i>
2.2) <i>Descrição dos Componentes Utilizados.....</i>	<i>3</i>
3. Procedimentos Executados.....	9
3.1) <i>Descrição dos Procedimentos Executados.....</i>	<i>9</i>
3.1.1) <i>Controle.....</i>	<i>9</i>
3.1.2) <i>Entradas.....</i>	<i>11</i>
3.1.3) <i>Operadores Lógicos.....</i>	<i>11</i>
3.1.4) <i>Procedimentos executados.....</i>	<i>11</i>
3.1.5) <i>Apresentando o circuito somador.....</i>	<i>15</i>
3.1.6) <i>Apresentando o circuito subtrator.....</i>	<i>15</i>
4. Conclusão.....	16
5. Referências Bibliográficas.....	17



1. Introdução e Objetivos

1.1) Introdução

Neste relatório, será apresentado o desenvolvimento de uma ULA de 1 bit, tal como todo seu processo de desenvolvimento e a explicação para cada operação nele feita, assim como todos os componentes e conhecimentos usados.

1.2) Objetivos da experiência

O relatório tem como objetivo compreender e simular um projeto de circuito combinacional ULA (Unidade Lógica e Aritmética) de 1 bit.

2. Fundamentação Teórica

2.1 Componentes Utilizados

- 6 Chaves Lógicas
- 4 Portas NOT
- 8 Portas 3-in-AND
- 17 Portas 2-in-AND
- 1 Porta 2-in-NAND
- 3 Portas 2-in-OR
- 2 Portas 4-in-OR
- 1 Porta 2-in-NOR
- 5 Portas 2-in-XOR
- 1 Porta 2-in-XNOR
- 2 LEDs

2.2 Descrição dos componentes utilizados

- **Chaves lógicas:** Permite a escolha do nível de entrada, podendo variar entre 0 (sem corrente) e 1 (com corrente).



Figura 1. Chave lógica com valor 0.

- **Porta NOT:** Inverte o nível lógico da corrente, como exemplificado na tabela-verdade abaixo:

Entrada A	Saída
0	1
1	0

Tabela 1. Tabela-Verdade do NOT.

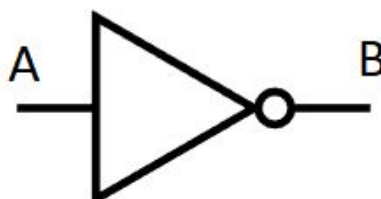


Figura 2. Porta NOT (inversor).

Em Álgebra de Boole, pode ser representado como $S = \bar{A}$. Leia-se A barra ou A barrado.

- **Porta AND:** Porta com duas ou mais entradas, retorna 1 somente quando todas as suas entradas receberem nível lógico igual a 1. Em Álgebra de Boole,



pode ser representada como $S = A \cdot B$. Leia-se A e B. Seu comportamento é descrito pela tabela-verdade abaixo.

Entrada A	Entrada B	Saída
0	0	0
0	1	0
1	0	0
1	1	1

Tabela 2. Tabela-Verdade da porta AND.



Figura 3. Porta AND.

- **Porta NAND:** Porta com duas ou mais entradas, sua saída é 0 somente quando todas suas entradas forem iguais a 1. Em Álgebra de Boole, pode ser representado como $S = \overline{A \cdot B}$. Leia-se A e B barrado. Seu comportamento é descrito na tabela-verdade abaixo.

Entrada A	Entrada B	Saída
0	0	1
0	1	1
1	0	1
1	1	0

Tabela 3. Tabela-verdade porta NAND.



Figura 4. Porta NAND.

- **Porta OR:** Porta com duas ou mais entradas, sua saída é 1 quando ao menos uma das entradas é igual a 1. Pode ser escrita em álgebra de Boole como $A + B$. Leia-se A ou B.
O comportamento da porta OR é indicado pela tabela abaixo.

Entrada A	Entrada B	Saída
0	0	0
0	1	1
1	0	1
1	1	1

Tabela 4. Tabela-Verdade da porta OR.

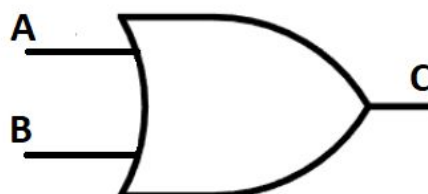


Figura 5. Porta OR.



- **Porta NOR:** Porta com duas ou mais entradas, sua saída é 1 quando todas entradas forem iguais a 0. Em álgebra de Boole pode ser escrita como $\overline{A + B}$. Leia-se A ou B barrado. O comportamento da porta NOR é descrito pela tabela-verdade abaixo.

Entrada A	Entrada B	Saída
0	0	1
0	1	0
1	0	0
1	1	0

Tabela 5. Tabela-Verdade da porta NOR.

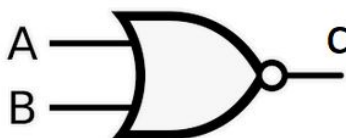


Figura 6. Porta NOR.

- **Porta XOR:** porta com duas ou mais entradas, sua saída é 1 quando suas entradas possuem valores lógicos diferentes. Em álgebra de Boole pode ser escrita como $A \oplus B$. Leia-se A ou exclusivo B. Seu comportamento é descrito na tabela-verdade abaixo.

Entrada A	Entrada B	Saída
0	0	0
0	1	1
1	0	1
1	1	0

Tabela 6. Tabela-Verdade da porta XOR.

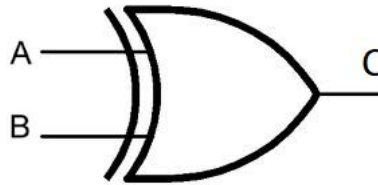


Figura 7. Porta XOR.

- **Porta XNOR:** Porta com duas ou mais entradas. Sua saída é 1 quando as entradas forem iguais. Em álgebra de Boole, é escrita como $A \odot B$. Leia-se não ou exclusivo. Seu comportamento é descrito pela tabela-verdade abaixo

Entrada A	Entrada B	Saída
0	0	1
0	1	0
1	0	0
1	1	1

Tabela 7. Tabela-Verdade da porta XNOR.

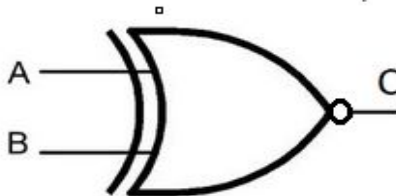


Figura 8. Porta XNOR.

- **LED:** É um diodo emissor de luz, utilizado no circuito para verificar quando a saída é igual a 1 (acende) ou 0 (apaga).

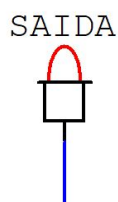


Figura 9. LED (apagado).

3. Procedimentos executados

3.1) Descrição do Funcionamento do circuito

3.1.1) Controle

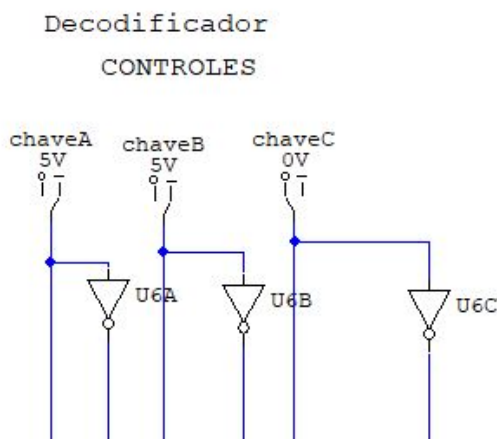


Figura 10. Controles.

A função do **controlador** é comandar se o circuito executa uma operação lógica, de soma ou subtração.

A escolha é ditada como na tabela verdade abaixo.

Con2	Con1	Con0	AND	NAND	OR	NOR	XOR	XNOR	SOMA	SUB
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

Tabela 8. Tabela-Verdade das operações aritméticas.

Vejamos o funcionamento com um exemplo:

Para observar o caminho percorrido pelo circuito observa a imagem abaixo

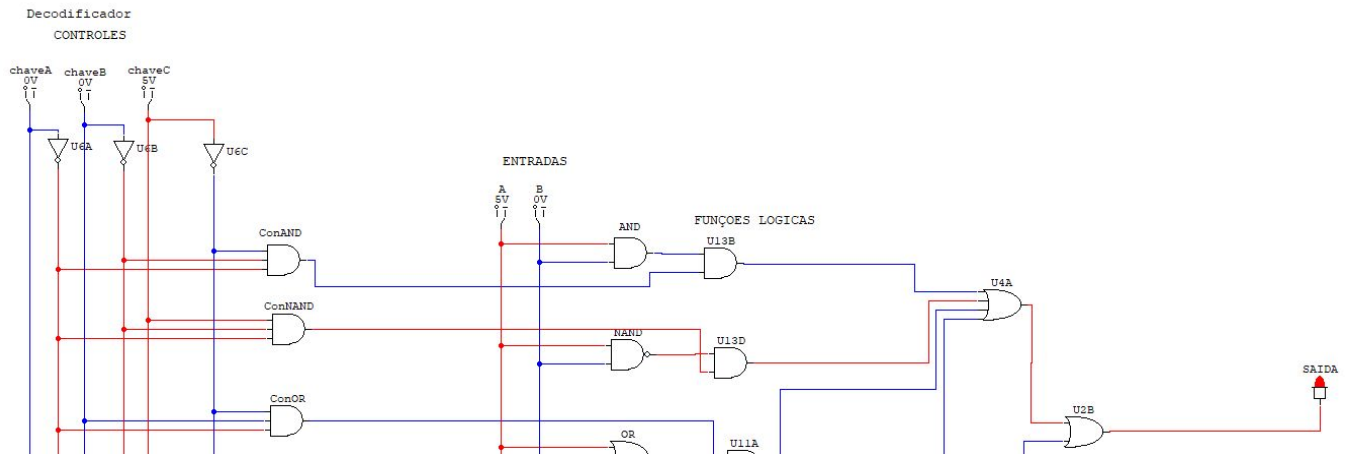


Figura 11. Circuito com as portas A, B e C com valores lógicos igual a 0, 0 e 1.

Para escolher a porta NAND, escolhemos os controles A como 0, B como 0 e C como 1, quando a corrente passar pela porta 3-in-AND, a única que terá valor lógico 1 é a “conNAND”, fazendo com que a corrente continue seu percurso até as funções lógicas. A corrente das entradas passarão por todas as portas lógicas e serão operadas, como a NAND foi escolhida, a porta AND garante que somente ela siga, e assim chega na saída.



3.1.2) Entradas

ENTRADAS

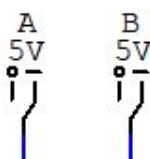


Figura 12. Entradas A e B do circuito.

As entradas definem o valor lógico a ser utilizado pelos operadores, elas podem assumir os valores

A	B
0	0
0	1
1	0
1	1

Tabela 9. Tabela verdade das entradas A e B

3.1.3) Operadores Lógicos

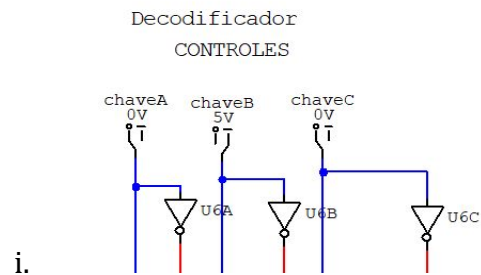
Os operadores lógicos presentes no circuito, por ordem são:

AND, NAND, OR, NOR, XOR E XNOR.

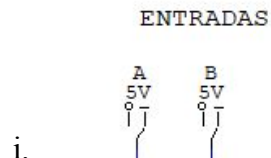
As entradas decididas, passam por todas as portas e a resposta passará para frente apenas no circuito escolhido no **Controle**, isso é possível por meio da porta **AND** que liga a saída da porta lógica e a saída do controle.

3.1.4) Procedimentos executados

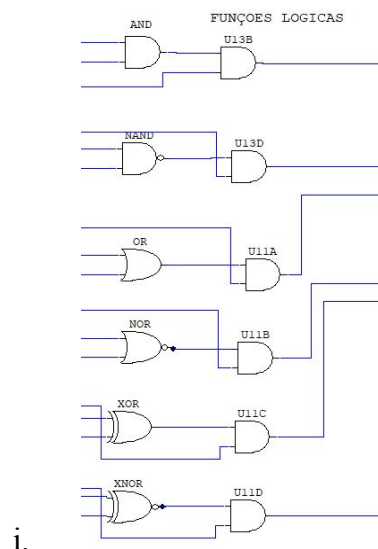
1. Primeiro foi adicionado os controladores, denominados como chaveA, chaveB e chaveC.



2. As chaves (controladores) foram ligados em 8 portas AND, todas com 3 entradas e 1 saída, sendo 6 separadas em operações AND, NAND, OR, NOR, XOR e XNOR, respectivamente. E as outras 2 responsáveis pelas operações de Soma e Subtração.
3. Depois foram adicionadas duas entradas, separadas em A e B



4. As entradas foram ligadas em portas lógicas, responsáveis pela função lógica do circuito
5. Para as funções lógicas foram utilizados 12 portas lógicas, sendo 6 portas AND, 1 NAND, 1 OR, 1 NOR, 1 XOR E 1 XNOR.



6. Para a função responsável pela soma, foram utilizados 6 portas, sendo 2 XOR, 1 OR e 3 AND.

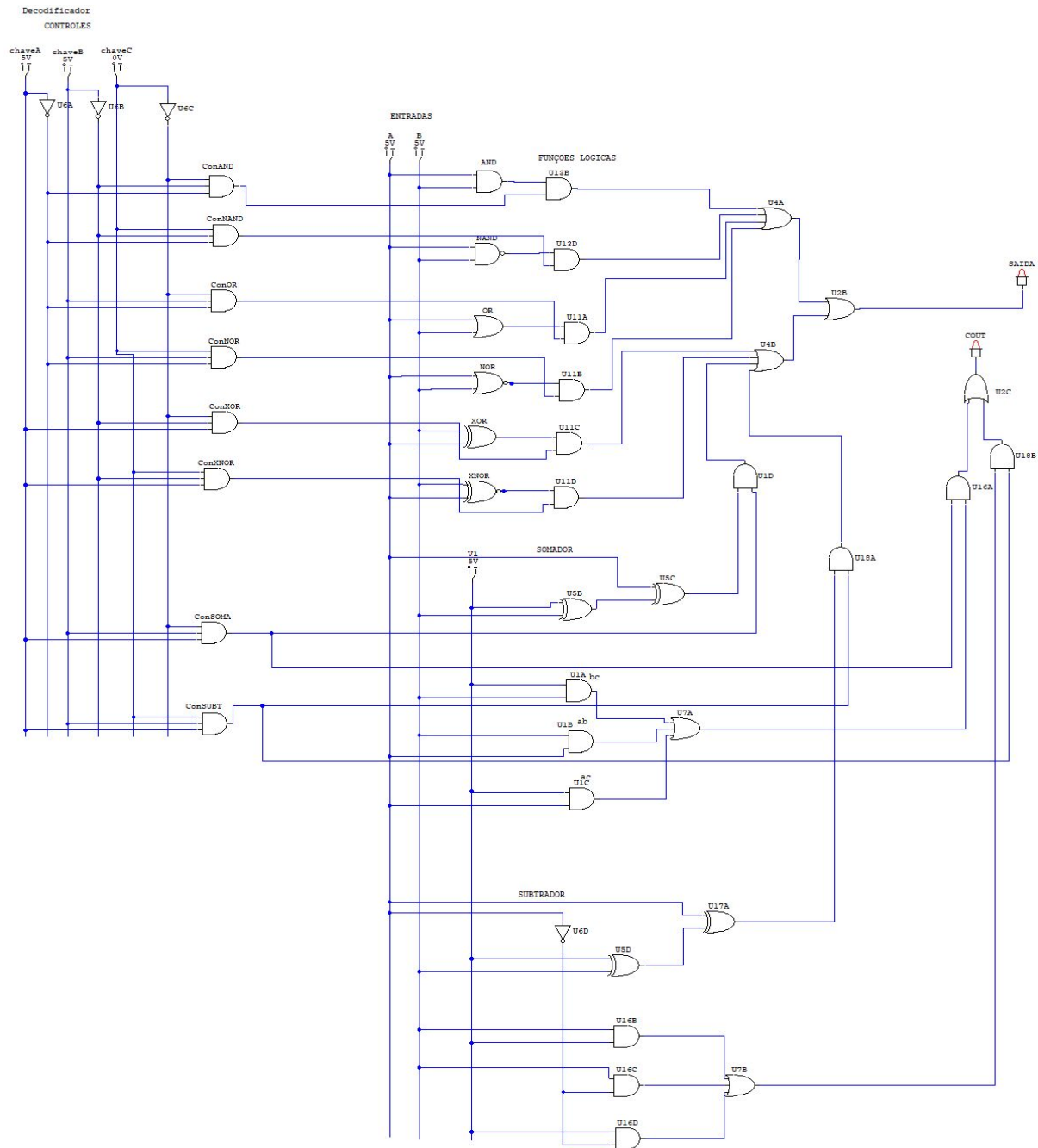


Figura 13. Circuito ULA de 1 bit.



3.1.5) Apresentando o circuito somador

O **circuito somador** tem como objetivo somar os bits de entrada, seu funcionamento pode ser descrito pela tabela abaixo.

Cin	A	B	SAIDA	COUT
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1

Tabela 10. Tabela-Verdade do circuito somador

As expressões booleanas do circuito somador é expressa pela seguinte forma:

$$\text{Cout} = AB + BC_{in} + AC_{in}$$

$$S = A \oplus B \oplus C_{in}$$

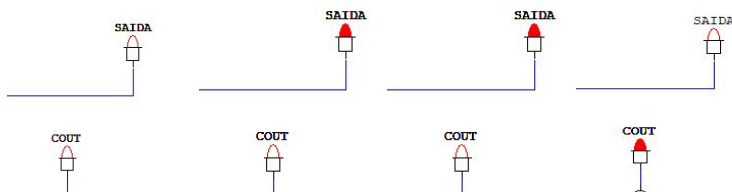


Figura 14. Representação dos LEDs conforme as saídas da Tabela-Verdade da soma.

3.1.6) Apresentando o circuito subtrator

O **circuito subtrator** subtrai os bits de entrada, seu funcionamento pode ser descrito pela tabela-verdade abaixo.



Cin	A	B	S	Cout
1	0	0	0	0
1	0	1	1	1
1	1	0	0	1
1	1	1	0	0

Tabela 11. Tabela-Verdade da operação subtração

As expressões booleanas do circuito somador é expressa pela seguinte forma:

$$\text{Cout} = AB + BC_{in} + A$$

$$S = A \oplus B \oplus C_{in}$$

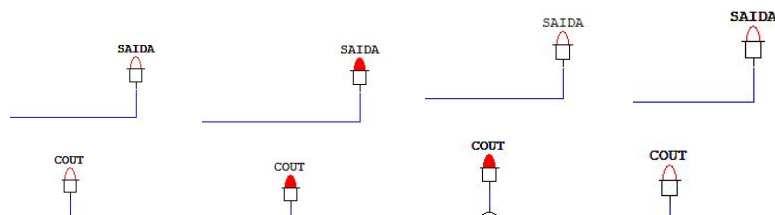


Figura 15. Representação das saídas da Tabela-Verdade da subtração

4. Conclusão

O objetivo deste projeto foi a criação de uma ULA de 1 bit e apresentar o seu funcionamento.

Conclui-se a partir deste projeto da Unidade lógica aritmética, ULA, de 1 bit, que toda unidade de processamento, até mesmo as mais simples como esta, carrega consigo uma grande carga de conhecimento.

Composta por um decodificador, funções lógicas, somador e subtrator, ao ser escalada, a ULA permitiu a criação dos computadores como conhecemos, As ULAs nada mais são do que a UCP (unidade central de processamento) que se equivale ao cérebro dos computadores modernos. Assim os objetivos desse projeto foi atingido - O desenvolvimento da ULA e a apresentação de seu funcionamento.



5. Referências Bibliográficas

Pimenta, Tales. **Circuitos digitais: Análise e Síntese Lógica - Aplicações em FPGA**. Editora GEN LTC., edição 1, 2016

Francisco G. Capuano; Ivan V. Idoeta. **Elementos de Eletrônica Digital**. Editora Érica, Edição 41^a, 1997.

K. Alexander, Charles. **Fundamentos de Circuitos Elétricos**. Editora AMGH, Edição 5, 2013.

Floyd, Thomas L. **Sistemas Digitais - Fundamentos e Aplicações**. Editora Bookman, Edição 9, 2007.