

**UNIVERSIDADE ESTADUAL DE MARINGÁ**  
**INFORMÁTICA**

**FRASE EM DISPLAY DE 7 SEGMENTOS**

Olga Maria dos Santos R.A. 130002

Curso: Informática,

Disciplina Circuitos Digitais (9889-2022-T32)

Professor: Maurilio Martins Campano Junior

MARINGÁ

2022

## SUMÁRIO

<b>Introdução.....</b>	<b>3</b>
<b>Desenvolvimento.....</b>	<b>6</b>
<b>Justificativa.....</b>	<b>6</b>
<b>Decisões de projeto para a simulação.....</b>	<b>8</b>
○ Tabela verdade completa	
○ Simplificação das saídas por Mapa de Karnaugh	
○ Representação do Circuito lógico	
<b>Conclusão.....</b>	<b>11</b>
<b>Referências.....</b>	<b>12</b>

## Introdução

Este projeto tem como objetivo colocar em prática os conhecimentos de circuitos combinacionais em conjunto com circuitos sequencias, devemos criar uma frase, que parte de 4 entradas binárias, ou seja, 2bits, e devemos mostrar em um display de 7 segmentos, tanto em ordem crescente e decrescente da frase, utilizando nesta uma ideia de circuito sequencial, o contador.

A frase escolhida para ser mostrado foi “- - SUBI NO ONIBUS - -”, tendo 2 traços no inicio e fim para que chegasse no limite de opções que 4 entradas binárias tem de possibilidades, que são no caso 16 possibilidades, e utilizamos os 16 caracteres, a frase está com espaços, mas no display como é mostrado somente um caractere por vez, não utilizaremos espaços, logo não contam como caractere.

## Descrição

Vamos conceituar quais elementos utilizaremos neste projeto:

Utilizaremos o programa logisim, que é um simulador lógico que permite o desenho e a simulação de circuitos através de uma interface gráfica, nele utilizaremos elementos:

**Pinos de entrada:** recebem os valores 0 ou 1, que vão compor as entradas em binário, serão importantes para a criação do display.



**Display de 7 segmentos:** Display que mostrará os valores de binários, em letras.



**Clock:** Responsável por enviar um pulso ao contador, que irá assim alterar seu valor de 0 para 1 ou de 1 para 0.



**Distribuidores:** Eles recebem os valores e distribuem na quantidade de bits que precisamos, servirão para deixar os circuitos mais limpo e uma distribuição mais eficiente:



**Chips:** São “caixinhas” que compõe toda a lógica do que foi programado dentro delas, são maneiras resumidas, iremos utilizar estas nos contadores, e nos displays, vai ficar visualmente mais eficiente e de melhor entendimento, pois nomearemos os chips com qual sua função, e seu visual pode mudar dependendo do número de entradas e saídas:



**Porta AND:** A porta lógica AND, “e” na língua portuguesa, faz referência a uma operação lógica que aceita dois ou mais operandos, que sempre resultem em um valor lógico verdadeiro, se somente se todos os valores passados terem seu valor sendo verdadeiro, caso o contrário se existir um valor ou mais que sejam falsos a saída resultante é falsa.

#### Representação no Logisim

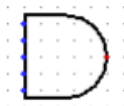
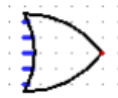


Tabela verdade da Porta AND

P	Q	$P \wedge Q$
0	0	0
0	1	0
1	0	0
1	1	1

**Porta Lógica OR:** na tradução para o português fica “ou”, faz referência a uma operação lógica que permite a entrada de um ou mais valores e que sempre retorna um valor verdadeiro se um desses valores forem verdadeiros.

### Representação no Logisim

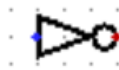


### Tabela verdade Porta OR

P	Q	P V Q
0	0	0
0	1	1
1	0	1
1	1	1

**Porta NOT:** A porta not, em português “não”, tem a função de trocar o valor passado por ela, por exemplo, se um valor verdadeiro é passado por uma porta not a saída será falsa, se a entrada é falsa, a saída será verdadeira. A ideia em si é sempre trocar os valores.

### Representação no Logisim



### Tabela verdade NOT

A	A'
0	1
1	0

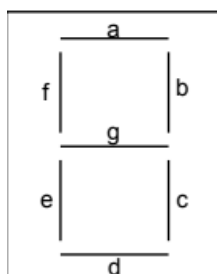
## Desenvolvimento

### Justificativa

O projeto consiste em criar uma tabela, com as quatro entradas binárias possíveis, sendo 16 possibilidades, e na possibilidade 0, ou chamaremos de 1º posição (0000), o segmentos do display que acenderam devem coincidir com a 1ª posição da nossa frase, que é um (-), é a mesma lógica para as demais posições, seguindo a tabela:

S4	S3	S2	S1		POSIÇÃO
0	0	0	0	-	1
0	0	0	1	-	2
0	0	1	0	S	3
0	0	1	1	U	4
0	1	0	0	B	5
0	1	0	1	I	6
0	1	1	0	N	7
0	1	1	1	O	8
1	0	0	0	O	9
1	0	0	1	N	10
1	0	1	0	I	11
1	0	1	1	B	12
1	1	0	0	U	13
1	1	0	1	S	14
1	1	1	0	-	15
1	1	1	1	-	16

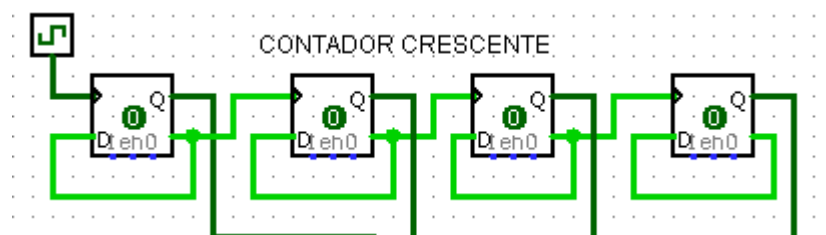
Logo depois definimos quais segmentos acenderiam e determinadas posições, baseado no display:



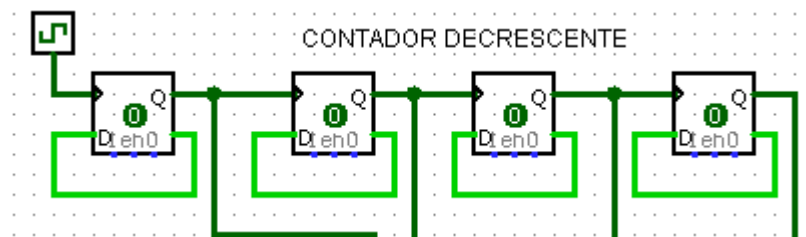
Chegamos na tabela:

--SUBI NO ONIBUS--											
S4	S3	S2	S1	A	B	C	D	E	F	G	
0	0	0	0	0	0	0	0	0	0	1	-
0	0	0	1	0	0	0	0	0	0	1	-
0	0	1	0	1	0	1	1	0	1	1	S
0	0	1	1	0	1	1	1	1	1	0	U
0	1	0	0	0	0	1	1	1	1	1	B
0	1	0	1	0	0	0	0	1	1	0	I
0	1	1	0	1	0	1	0	1	0	1	N
0	1	1	1	1	1	1	1	1	1	0	O
1	0	0	0	1	1	1	1	1	1	0	O
1	0	0	1	1	0	1	0	1	0	1	N
1	0	1	0	0	0	0	0	1	1	0	I
1	0	1	1	0	0	1	1	1	1	1	B
1	1	0	0	0	1	1	1	1	1	0	U
1	1	0	1	1	0	1	1	0	1	1	S
1	1	1	0	0	0	0	0	0	0	1	-
1	1	1	1	0	0	0	0	0	0	1	-

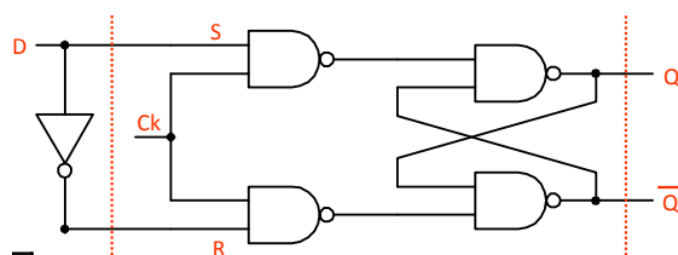
Foi necessário montarmos o contador, este que iria controlar a ordem de aparecimento dos caracteres, para montar o **contador crescente** utilizamos o flip-flop D:



O **contador decrescente** utilizamos o flip-flop D também:



Sendo a lógica interna de cada flip-flop D, a seguinte:



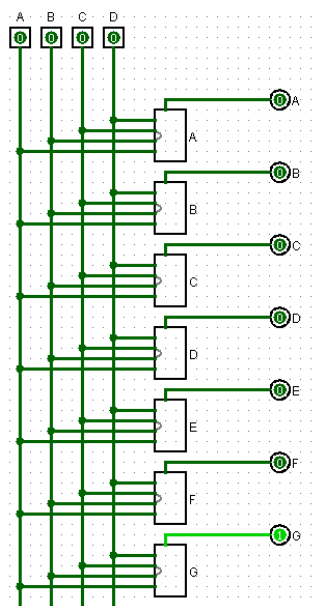
O clock, foi conectado somente no primeiro flip flop, o restante sempre recebia a saída do outro, isso que nos garante que irão seguir a ordem binária.

### Decisões de projeto para a simulação

Depois de já termos montado os contadores, e testado principalmente, a tabela estava definida, era necessário retirar da tabela e transformar em circuito, foi retirado pelo método de mapa de karnaugh, e chegamos nas seguintes expressões:

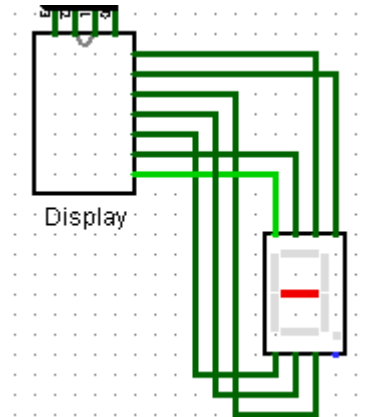
<b>A =</b>	$\bar{A}C\bar{D} + A\bar{C}D + A\bar{B}\bar{C} + \bar{A}BC$
<b>B =</b>	$\bar{A}CD + A\bar{C}\bar{D}$
<b>C =</b>	$\bar{A}C + A\bar{C} + B\bar{C}\bar{D} + \bar{B}CD$
<b>D =</b>	$\bar{A}CD + B\bar{C}\bar{D} + A\bar{C}\bar{D} + \bar{A}BC + AB\bar{C} + \bar{B}CD$
<b>E =</b>	$\bar{A}B + A\bar{B} + \bar{A}CD + B\bar{C}\bar{D}$
<b>F =</b>	$B\bar{C} + \bar{B}C + \bar{A}CD + A\bar{C}\bar{D}$
<b>G =</b>	$\bar{A}\bar{D} + AD + B\bar{C}\bar{D} + \bar{B}CD$

Uma expressão para cada segmento do display, logo após foi feito uma biblioteca com todos os 7 circuitos de cada letra, e com o objetivo de filtrarmos e deixarmos mais limpo, foi utilizado cada circuito de cada letra, em outro circuito denominado “Display”, ficando:



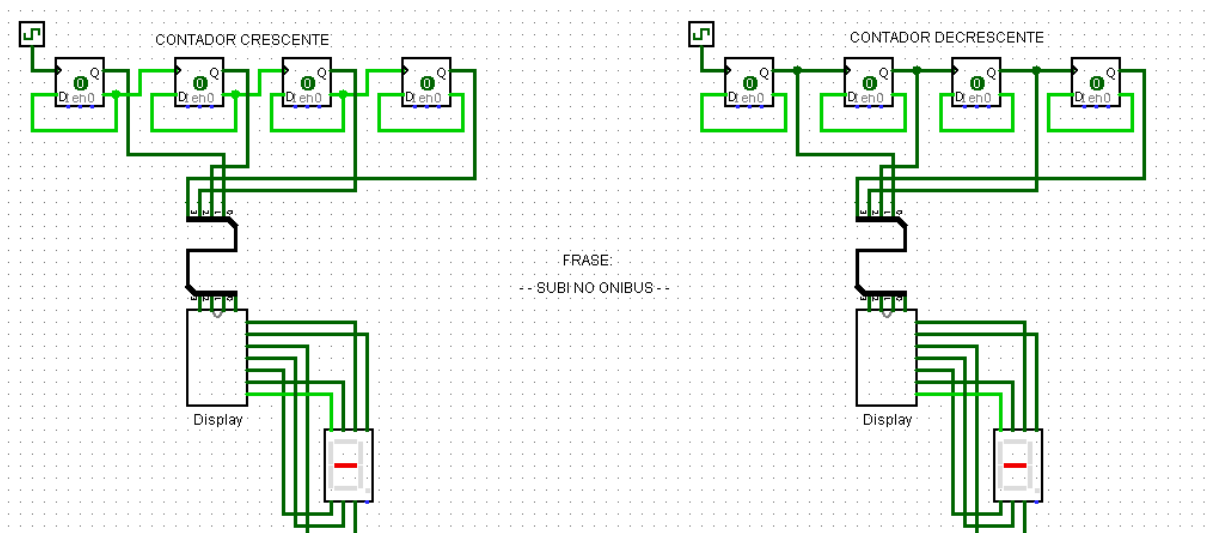


Agora tínhamos um arquivo que tem 4 entradas, e 7 saídas, cada uma responsável por um segmento do display, logo colocamos ele como chip no display principal “Frase”:



Conectamos a saída do circuito “Display” as entradas do display de 7 segmentos, e as entradas do “Display”, são as saídas do contador, assim seria acesso os segmentos, em ordem binária, ou também na ordem decrescente binária.

### Circuito Final



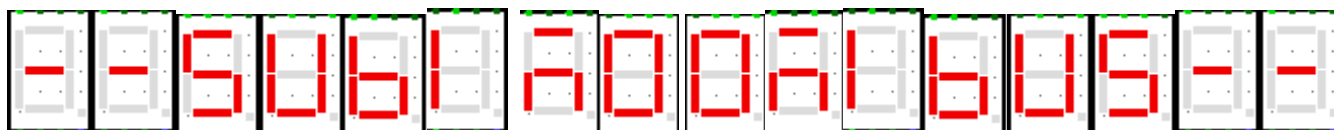
### Resultados

Os resultados, como a frase de trás para frente e em ordem de frente para trás, é a mesma “ - - SUBI NO ONIBUS - -”, mostraremos a tabela, da ordem binária crescente e decrescente que os contadores partiram, e quais os resultados nos displays:

CRESCENTE				Resultado Display	DECRESCENTE			
S4	S3	S2	S1		S4	S3	S2	S1
0	0	0	0	-	1	1	1	1
0	0	0	1	-	1	1	1	0
0	0	1	0	S	1	1	0	1
0	0	1	1	U	1	1	0	0
0	1	0	0	B	1	0	1	1
0	1	0	1	I	1	0	1	0
0	1	1	0	N	1	0	0	1
0	1	1	1	O	1	0	0	0
1	0	0	0	O	0	1	1	1
1	0	0	1	N	0	1	1	0
1	0	1	0	I	0	1	0	1
1	0	1	1	B	0	1	0	0
1	1	0	0	U	0	0	1	1
1	1	0	1	S	0	0	1	0
1	1	1	0	-	0	0	0	1
1	1	1	1	-	0	0	0	0

Resultados das letras no display foi:

“- - SUBI NO ONIBUS - -”



## **Conclusão**

Este projeto nos levou a entender a conexão entre circuitos combinacionais e sequencias, de certa forma, desenvolvemos conceitos estudados em sala de aula, na prática, definirmos as entradas de dados, e por ela onde deveríamos chegar, que seria os segmentos a acender, logo após é necessário entender os conceitos de flip flops, RS, JK, D e T, para entender qual o melhor para o projeto, chegamos na conclusão que o D seria mais simples, e unir a saída dele, a entrada do display utilizando distribuidores, foi o que conectou os dois tipos de circuitos combinacionais e sequenciais e o que uniu o nosso projeto, e tudo, levou a um extensão do nosso conhecimento sobre os circuitos digitais.

## Referências

Cefuve, Flip flop contador binário, 2015, disponível:

<https://www.youtube.com/watch?v=SXCXcZuqDjU>

GV Ensino, Eletrônica digital, entendendo contado assíncrono, 2020 disponível:

[https://www.youtube.com/watch?v=c1o46eJ\\_ocA](https://www.youtube.com/watch?v=c1o46eJ_ocA)

Galdino, Jean, Organização de computadores, 2016, disponível:

<https://docente.ifrn.edu.br/jeangaldino/disciplinas/2016.1/organizacao-de-computadores/aula-14-logisim-interligando-as-partes>

Cantu, Evandro, simulador de circuitos lógicos, sistemas digitais 2016, disponível em: [wiki.foz.ifpr.edu.br/wiki/index.php/Simulador\\_de\\_Circuitos\\_L%C3%B3gicos\\_-\\_Logisim](http://wiki.foz.ifpr.edu.br/wiki/index.php/Simulador_de_Circuitos_L%C3%B3gicos_-_Logisim)

Thiago Lima, Portas lógicas, 2015, disponível:

<https://embarcados.com.br/portas-logicas/>