



Universidade Federal do Rio Grande do Norte

Centro de Tecnologia - CT

Departamento de Engenharia Elétrica

CIRCUITO LÓGICO PARA EXIBIÇÃO DE CARACTERES EM UMA MATRIZ 8X8

ELE2715 - Grupo 04 - Projeto 01

Sthefania Fernandes Silva
Isaac de Lyra Junior
Vinícius Souza Fonsêca
Antônio Paulo Vinícius Barateiro de Sousa
Rafael Pereira de Alexandria Soares

Natal, 24 de janeiro de 2021

Resumo

Este trabalho tem como objetivo o projeto de um circuito lógico combinacional que represente - através de LEDs - alguns caracteres específicos e utilizando uma matriz 8x8 como espaço de referência. Dessa forma faz-se o uso do efeito POV (*Persistence of vision*) para acionar os LEDs da matriz de acordo com a escolha binária dos caracteres numéricos a serem exibidos. Para tal execução, foram montadas tabelas verdade e definida - de forma lógica - a posição de acendimento de cada led durante a exibição. Posteriormente, foram montados os circuitos lógicos que representam cada linha e coluna da matriz de LED, como também, as entradas, visando simular as saídas e testar a solução. A partir dos resultados obtidos foi possível notar a eficácia da solução encontrada para a exibição de caracteres em uma matriz 8x8.

Palavras-chaves: Efeito POV; Matriz de LEDs; Circuitos Digitais.

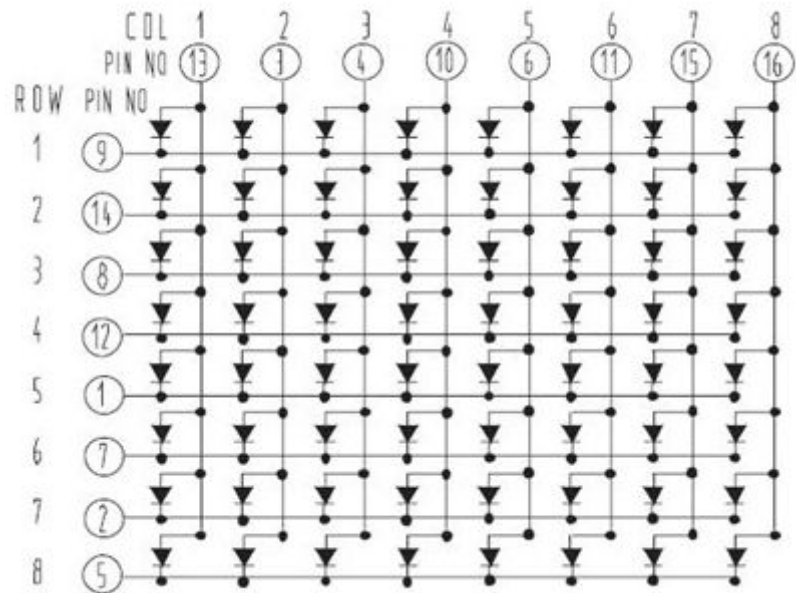
Sumário

1 Introdução	3
Figura 1 - Datasheet do modelo 1088AS	3
Figura 2 - Circuito para matriz de LEDs 8×8	3
2 Desenvolvimento	5
Figura 3 - Tabela verdade da linha 1	5
Figura 4 - (a) Circuito lógico para o acendimento de um LED (b) LED acendido por esse circuito	6
3 Conclusão	7
Referências	8
Anexo A - Tabelas Verdade	9

1 Introdução

A matriz de LEDs é um módulo que possui 64 LEDs distribuídos em 8 linhas e 8 colunas, formando um quadrado. A matriz é feita de modo que cada led seja acionado individualmente, portanto ela não possui 65 pinos (64 pinos + o terra) e sim 16. Isso acontece, pois os LEDs são interligados por linhas e colunas (GUIMARÃES, 2018). No modelo 1088AS, os LEDs de cada coluna tem os ânodos interligados e os LEDs de cada linha tem os cátodos interligados, como pode ser observado na figura 1.

Figura 1 - Datasheet do modelo 1088AS

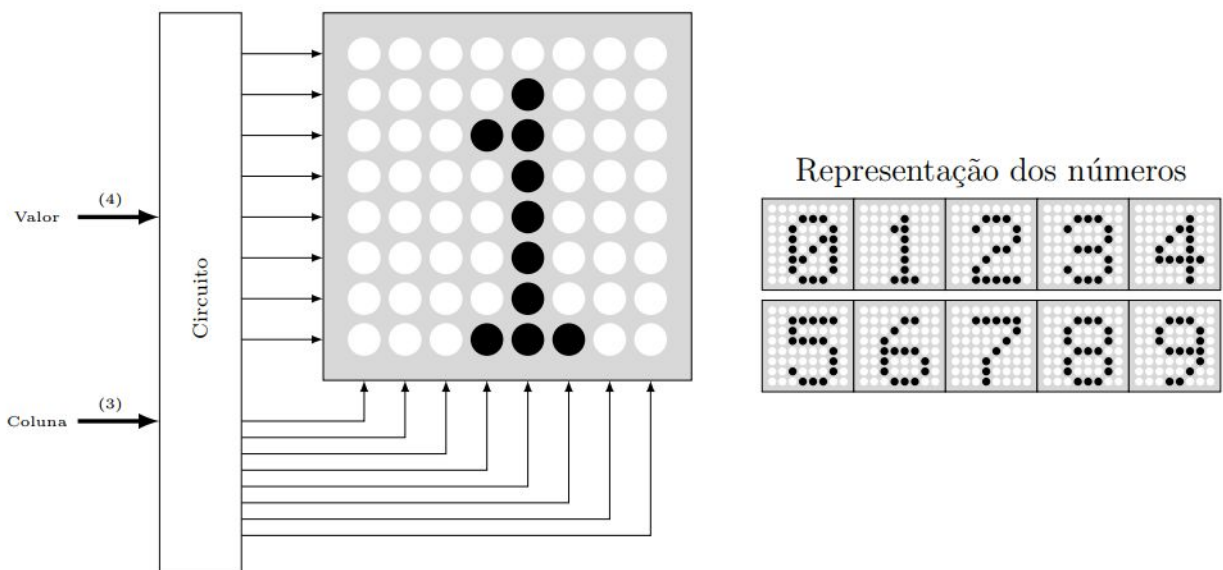


Fonte: DatasheetBank

Essa configuração impossibilita que certas combinações sejam criadas nos LEDs, visto que, o acionamento de um LED pode interferir no funcionamento de outro.

Diante dessa problemática, o presente relatório busca exibir em uma matriz de LEDs 8x8 dez algarismos, como mostrado na figura 2.

Figura 2 - Circuito para matriz de LEDs 8x8



Fonte: Autoria própria (2021).

Para isso, primeiramente foi preciso entender o efeito *Persistence Of Vision* - POV. O POV é um fenômeno que causa a ilusão de movimento quando uma determinada imagem, vista pelo olho humano, permanece na retina por uma pequena fração de segundo logo após sua captura. Se, por exemplo, forem visualizadas dezesseis ou mais imagens continuamente em um intervalo de tempo de 1 segundo, o olho humano será incapaz de perceber a mudança entre elas, criando um movimento que se assemelha a uma imagem (CALADO, 2015).

Além disso, outros fundamentos teóricos também foram utilizados, como: a lógica Booleana. Ela foi desenvolvida no século 19 pelo matemático George Boole, seu objetivo era formalizar a lógica e o raciocínio humano através de métodos algébricos. Hoje seu arranjo matemático é utilizado na construção de circuitos digitais (VAHID, 2008).

A álgebra Booleana possui operadores que manipulam as variáveis para retornar um valor falso (0) ou verdadeiro (1) de acordo com a sentença. Esses operadores são denominados AND; OR e NOT. O primeiro é usado para garantir que ambas as variáveis sejam verdadeiras, o segundo para assegurar que pelo menos um das variáveis seja verdadeira e o último retorna verdadeiro se a variável for falsa e vice-versa (VAHID, 2008).

A álgebra booleana pode ser representada através de equações, circuitos e tabelas verdade. A última representa todas as combinações de entradas e saídas de uma função (VAHID, 2008).

Em posse desses conhecimentos foi possível desenvolver uma solução para o problema proposto.

2 Desenvolvimento

A questão inicial foi entender como acender LEDs específicos, levando em consideração o efeito POV. Diante disso, primeiramente foi considerado que a operação de acender e apagar os LEDs precisa ser coordenada, de forma que seja identificada quais linhas e colunas específicas são necessárias para exibir um número.

Como no experimento executado os algarismos não chegam a ocupar todo o espaço disponível no módulo, buscou-se manusear somente as 7 linhas e 5 colunas que possuem LEDs acesos. Diante disso, foram criadas tabelas verdade para cada uma das linhas: as entradas foram os 4 bits que representam em binário os algarismos de 0 a 9 e as saídas foram as colunas que acendem (ou não) quando aquele algarismo é exibido na matriz.

Para simplificar as expressões algébricas, de cada tabela verdade, o Mapa de Karnaugh foi utilizado. No anexo A, todas as tabelas verdade estarão disponíveis para consulta. Na figura 3 é possível observar uma dessas tabelas.

Figura 3 - Tabela verdade da linha 1

L1											
char	bits (in)				C2	C3	C4	C5	C6	Eq. bool.	
0	0	0	0	0	0	1	1	1	0	C2	BD
1	0	0	0	1	0	0	1	0	0	C3	$A + BD + B'C + B'D'$
2	0	0	1	0	0	1	1	1	0	C4	$D + C + B'$
3	0	0	1	1	0	1	1	1	0	C5	$D' + C + B + A$
4	0	1	0	0	0	0	0	1	0	C6	BD
5	0	1	0	1	1	1	1	1	1		
6	0	1	1	0	0	0	1	1	0		
7	0	1	1	1	1	1	1	1	1		
8	1	0	0	0	0	1	1	1	0		
9	1	0	0	1	0	1	1	1	0		

Fonte: Autoria própria (2021).

A partir das expressões foi possível montar o circuito lógico do acendimento dos LEDs da matriz. No entanto, diante da configuração da matriz 8x8, para que haja o correto funcionamento de um LED em específico, é necessário que a linha esteja em *High*, enquanto a coluna precisará estar em *Low*, ou vice-versa. Essa dinâmica torna viável o efeito POV.

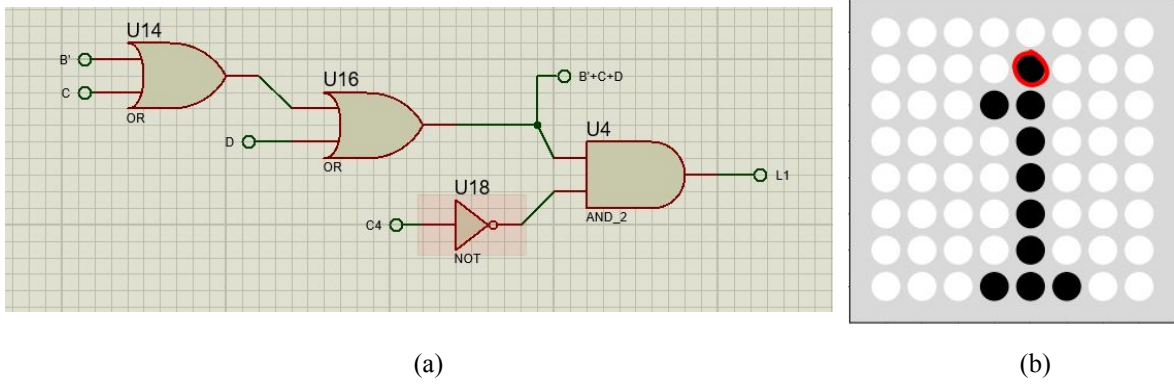
Como o problema proposto possui duas entradas: uma de 4 bits que representa o número a ser exibido e uma de 3 bits que representa as colunas da matriz, a solução encontrada foi manipular a entrada de 3 bits para acender os LEDs.

Assim sendo, a metodologia criada para o circuito funcionar é a seguinte: a entrada de 4 bits receberá o número que o usuário quer exibir na matriz de LEDs, enquanto a entrada de 3 bits receberá quais colunas deverão acender para o número ser exibido.

Apesar disso, para que o efeito POV aconteça, as colunas devem ser acesas uma após a outra, muito rápido. Nesse caso, montando o circuito de forma que todas as vezes que a coluna esteja no *High*, a linha esteja no *Low*, será possível exibir um número na matriz de LEDs. Logo, toda a coluna será ligada, mas somente os LEDs necessários para exibir o número aparecerão.

Para exemplificar o funcionamento, na figura 4 está o circuito para o acendimento do LED - circulado em vermelho - do número 1. Note que, além das expressões algébricas extraídas da tabela verdade para linha 1, a coluna 4 também é uma entrada para o circuito. Para esse LED acender essa coluna deve estar ligada, por essa razão ela também deve ser uma entrada do circuito lógico.

Figura 4 - (a) Circuito lógico para o acendimento de um LED (b) LED acendido por esse circuito



(a)

(b)

Fonte: Autoria própria (2021).

Usando esse método é possível ligar todos os LEDs necessários para exibir os 10 algarismos.

3 Conclusão

O problema de exibir números de 0 a 9 em uma matriz 8x8, na qual os LEDs estão interligados e o funcionamento de um interfere no outro, exigiu um bom embasamento teórico e muita discussão em grupo para elaboração da solução.

Após entender o desafio inicial e as exigências requeridas para a execução do projeto, foi possível desenvolver um método lógico para acender os LEDs respeitando a configuração da matriz 8x8.

Nossa solução consiste em inserir na entrada de 4 bits o valor que se deseja exibir na matriz, enquanto a entrada de 3 bits - que representa as colunas - varia rapidamente, permitindo que o efeito POV aconteça e o número seja visualizado.

Referências

DATASHEETBANK. 1088AS Datasheet. Disponível em:
<https://ru.datasheetbank.com/datasheet/ETC/1088AS.html>.

Acesso em: 23 jan. 2021.

GUIMARÃES, Fábio. **Matriz de LED – O que é e como usar**. 2018. Disponível em:
<http://mundoprojetado.com.br/matriz-de-led-o-que-e-e-como-usar/>. Acesso em: 23 jan. 2021.

VAHID, Frank. **Sistemas digitais: projeto, otimização e HDLs** / Frank Vahid; tradução: Anatólio Laschuk. - Porto Alegre: Artmed, 2008. 560p.

Anexo A - Tabelas Verdade

L1											
char	bits (in)				C2	C3	C4	C5	C6	Eq. bool.	
0	0	0	0	0	0	1	1	1	0	C2	BD
1	0	0	0	1	0	0	1	0	0	C3	$A + BD + B'C + B'D'$
2	0	0	1	0	0	1	1	1	0	C4	$D + C + B'$
3	0	0	1	1	0	1	1	1	0	C5	$D' + C + B + A$
4	0	1	0	0	0	0	0	1	0	C6	BD
5	0	1	0	1	1	1	1	1	1		
6	0	1	1	0	0	0	1	1	0		
7	0	1	1	1	1	1	1	1	1		
8	1	0	0	0	0	1	1	1	0		
9	1	0	0	1	0	1	1	1	0		

L2											
char	bits (in)				C2	C3	C4	C5	C6	Eq. bool.	
0	0	0	0	0	1	0	0	0	1	C2	$A + B'D' + B'C + BC'D$
1	0	0	0	1	0	1	1	0	0	C3	$BCD' + A'B'C'D$
2	0	0	1	0	1	0	0	0	1	C4	$BC'D' + A'B'C'D$
3	0	0	1	1	1	0	0	0	1	C5	$BC'D$
4	0	1	0	0	0	0	1	1	0	C6	$A + B'D' + CD$
5	0	1	0	1	1	0	0	0	0		
6	0	1	1	0	0	1	0	0	0		
7	0	1	1	1	0	0	0	0	1		
8	1	0	0	0	1	0	0	0	1		
9	1	0	0	1	1	0	0	0	1		

L3											
char	bits (in)				C2	C3	C4	C5	C6	Eq. bool.	
0	0	0	0	0	1	0	0	1	1	C2	$A + B'C'D' + BC'D + BCD'$
1	0	0	0	1	0	0	1	0	0	C3	BC'
2	0	0	1	0	0	0	0	0	1	C4	$A'C'D$
3	0	0	1	1	0	0	0	0	1	C5	$BD + A'C'D'$
4	0	1	0	0	0	1	0	1	0	C6	$A + B'D' + B'C$
5	0	1	0	1	1	1	1	1	0		

6	0	1	1	0	1	0	0	0	0	
7	0	1	1	1	0	0	0	1	0	
8	1	0	0	0	1	0	0	0	1	
9	1	0	0	1	1	0	0	0	1	

L4											
char	bits (in)				C2	C3	C4	C5	C6	Eq. bool.	
0	0	0	0	0	1	0	1	0	1	C2	$BD' + A'C'D'$
1	0	0	0	1	0	0	1	0	0	C3	$A + B'CD + BCD'$
2	0	0	1	0	0	0	1	1	0	C4	$B' + C$
3	0	0	1	1	0	1	1	1	0	C5	$A + B'C + BD'$
4	0	1	0	0	1	0	0	1	0	C6	$AD + BC'D + A'B'C'D'$
5	0	1	0	1	0	0	0	0	1		
6	0	1	1	0	1	1	1	1	0		
7	0	1	1	1	0	0	1	0	0		
8	1	0	0	0	0	1	1	1	0		
9	1	0	0	1	0	1	1	1	1		

L5											
char	bits (in)				C2	C3	C4	C5	C6	Eq. bool.	
0	0	0	0	0	1	1	0	0	1	C2	$C'D' + BD'$
1	0	0	0	1	0	0	1	0	0	C3	$A'B'D' + A'C'D' + BCD$
2	0	0	1	0	0	1	0	0	0	C4	$A'C'D$
3	0	0	1	1	0	0	0	0	1	C5	$BC'D'$
4	0	1	0	0	1	1	1	1	1	C6	$A + C'D' + BC' + BD' + B'CD$
5	0	1	0	1	0	0	0	0	1		
6	0	1	1	0	1	0	0	0	1		
7	0	1	1	1	0	1	0	0	0		
8	1	0	0	0	1	0	0	0	1		
9	1	0	0	1	0	0	0	0	1		

L6											
char	bits (in)				C2	C3	C4	C5	C6	Eq. bool.	
0	0	0	0	0	1	0	0	0	1	C2	$B'D' + B'C + CD' + BC'D$
1	0	0	0	1	0	0	1	0	0	C3	BCD
2	0	0	1	0	1	0	0	0	0	C4	$A'B'C'D$
3	0	0	1	1	1	0	0	0	1	C5	$AD + BC'D'$
4	0	1	0	0	0	0	0	1	0	C6	$B'C'D' + B'CD + BC'D + BCD'$
5	0	1	0	1	1	0	0	0	1		
6	0	1	1	0	1	0	0	0	1		
7	0	1	1	1	0	1	0	0	0		
8	1	0	0	0	1	0	0	0	1		
9	1	0	0	1	0	0	0	1	0		

L7											
char	bits (in)				C2	C3	C4	C5	C6	Eq. bool.	
0	0	0	0	0	0	1	1	1	0	C2	$B'CD'$
1	0	0	0	1	0	1	1	1	0	C3	$B' + D + C$
2	0	0	1	0	1	1	1	1	1	C4	$B' + C'D + CD'$
3	0	0	1	1	0	1	1	1	0	C5	$D' + A'B' + A'C'$
4	0	1	0	0	0	0	0	1	0	C6	$B'CD'$
5	0	1	0	1	0	1	1	1	0		
6	0	1	1	0	0	1	1	1	0		
7	0	1	1	1	0	1	0	0	0		
8	1	0	0	0	0	1	1	1	0		
9	1	0	0	1	0	1	1	0	0		

