



Relatório – Projeto de detector de magnitude relativa

Disciplina: ELE0518 – Laboratório de Sistemas Digitais

Alunos: Bruno Matias de Sousa

Data: 05/04/2019

Levy Gabriel da Silva Galvão

Pedro Henrique de Souza Fonsêca dos Santos

1. Introdução

É comumente observado no cotidiano, que a resolução de problemas que envolvem comparação de determinadas variáveis é primordial para análise dos mais variados sistemas, não sendo diferente nos sistemas digitais, que as utilizam para operações lógicas e aritméticas. Esse tipo de comparação serve para determinar se um número é maior, menor ou igual que o outro. Um comparador digital é imensamente usado em sistemas combinacionais e é especialmente projetado para comparar a magnitude relativa de números binários.

No projeto proposto de detecção de magnitude relativa faz a comparação das variáveis de entrada, de dois números binários de dois bits, compara a igualdade e caso haja diferença, olha se o número X é maior que Y. Para isso, foi necessário utilizar tudo que foi aprendido durante a disciplina, em resumo, foi feita a tabela verdade do sistema, e em seguida foi retirada a expressão lógica pelo o mapa de Karnaugh. Depois foi obtida o circuito lógico das expressões obtidas anteriormente. O circuito lógico foi montado em protoboard e analisado suas saídas com a tabela verdade. Para a montagem utilizou-se os CI com as portas lógicas OR, AND, XOR e NOT. Além disso foi utilizada os dois LED's para verificação das saídas do sistema.

2. Referencial teórico

O comparador de magnitude relativa é um circuito combinacional que compara dois números binários. No projeto a seguir foram utilizadas saídas para indicar com o acendimento de LEDs individuais apenas se os números em comparação são iguais ($X=Y$), ou um maior que o outro ($X>Y$). Fica implícito que no fato de não ocorrer o acendimento de nenhum dos LEDs, um comparador irá indicar que o segundo número é maior que o primeiro ($Y>X$), permitindo que, de certa forma se poupem recursos para a indicação dessa condição.

Esses compradores podem utilizar números das mais variadas quantidade de bits, sejam eles de 2, 4, 8 bits, etc, dependendo da aplicação requerida. Eles estão disponíveis na família de circuitos integrados TTL e CMOS, entre eles está incluído o CI 7485 (comparador de 4 bits), o CI 4585 (comparador de 4 bits da família CMOS) e o CI 74AS885 (comparador de 8 bits). Porém, o desejo desse projeto é implementar um circuito capaz de fazer a comparação de números de 2 bits e mostrar que não existe tamanhas dificuldades em sua implementação.

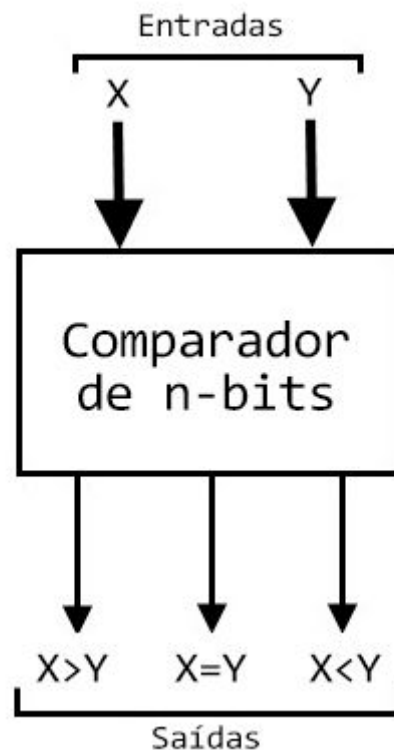


Figura 1 - Diagrama ilustrativo de um comparador com duas palavras de n-bits.

O referencial teórico do sistema analisado se baseia nas seguintes etapas:

1. Construir uma tabela-verdade a partir da interpretação do problema;
2. Montar o mapa de Karnaugh para as saídas a partir da tabela e dele retirar as equações booleanas necessárias;
3. Implementar o circuito para a expressão final, simplificada, em nível de portas lógicas.

Para descrever como a saída de um circuito lógico depende dos níveis lógicos presentes nas entradas dos circuitos, usamos a técnica da tabela-verdade. Essa tabela relaciona todas as combinações possíveis de níveis lógicos das entradas com seus correspondentes níveis lógicos de saída. Uma tabela verdade de N entradas tem 2^N sendo seu número de combinações de entrada. É simples completar uma tabela verdade sem esquecer nenhuma combinação visto que a lista de todas as combinações possíveis é uma sequência de contagem binária. É a partir da tabela verdade que obtemos a expressão booleana para o circuito requerido.

Em seguida foi utilizado o Mapa de Karnaugh para a obtenção das expressões lógicas do sistema. O mapa de Karnaugh conhecido também como Mapa-K, é um método gráfico utilizado na conversão para um circuito lógico de uma tabela verdade ou para simplificar uma equação lógica. Cada quadrado no Mapa de Karnaugh representa uma linha da tabela verdade. A numeração dos quadrados do Mapa-K está relacionada com a linha correspondente na tabela verdade. Cada quadrado contém em si o nível lógico de saída da combinação da linha correspondente na tabela verdade. Podemos ver logo abaixo alguns Mapas-K, para 2, 3, e 4 entradas.

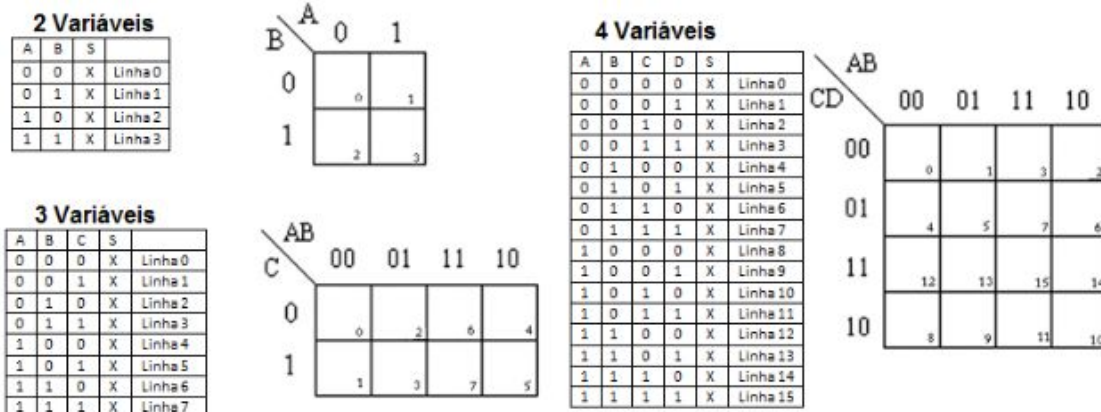


Figura 2 - Mapas-K para 2, 3 e 4 entradas.

No projeto foi utilizado o de 4 entradas para as saídas respectivas A e B.

Para o uso do Mapa-K na simplificação de uma expressão booleana pode ser seguido o procedimento aplicável à maioria dos casos: Colocar Um nos quadrados correspondentes aos níveis lógicos altos da tabela-verdade. Nos quadrados restantes, colocar zero. Faça os agrupamentos com maior número de Uns possível. Comece procurando octetos, depois quartetos, pares e isolados, nessa ordem, para máxima simplificação. Por fim, Forme somas de todos os termos gerados por cada grupo.

Com a simplificação do sistema, obtemos as expressões simplificadas para as Saídas A e B, logo foi possível escolher os CI adequados para cada problema.

A porta lógica utilizada foi a NOT, AND, XOR e OR e o CIs respectivo dessas portas são os CI's 7404, 7408, 7486 e 7432. As informações mais necessárias do *datasheet* são como o circuito é por dentro e a sua tabela-verdade.

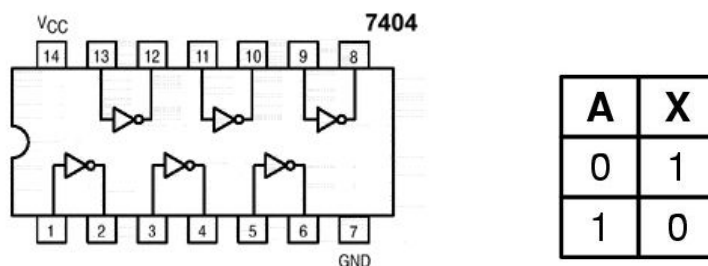


Figura 3 - Circuito integrado 7404 e tabela-verdade da porta NOT.

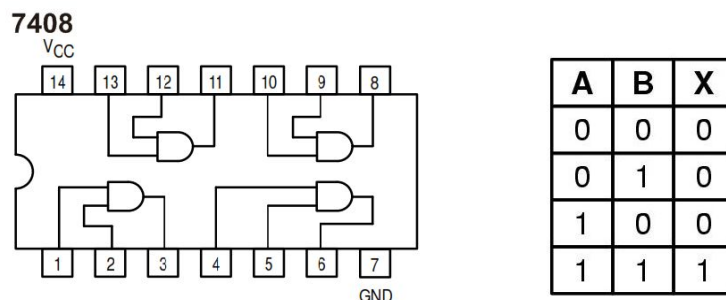


Figura 4 - Circuito integrado 7408 e tabela-verdade da porta AND.

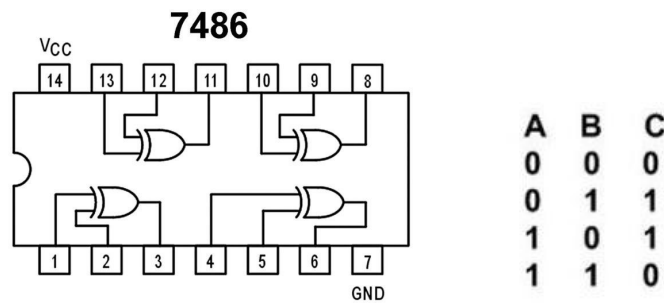


Figura 5 - Circuito integrado 7486 e tabela-verdade da porta XOR.

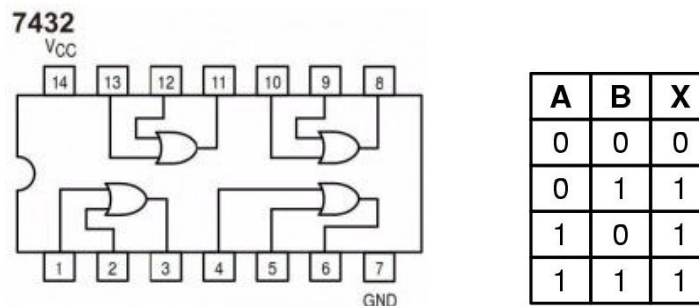


Figura 6 - Circuito integrado 7432 e tabela-verdade da porta OR.

Resumidamente o detector de magnitude relativa pode ser utilizado em qualquer lugar e aplicação, porém o que se deve saber é que para esse circuito funcionar ele deve receber sinais binários, ou seja, zero e um, somente assim vai poder executar as comparações na lógica empregada. Um dos locais mais comuns de se ter aplicado esse tipo de circuito é em uma CPU, nessa aplicação o circuito é capaz de realizar comparações com uma infinidade de valores de entrada e para diversas saídas.

3. Metodologia

3.1 Proposta do projeto

O projeto a ser realizado é referente a um bloco chamado detector de magnitude relativa. Este recebe dois números de dois bits e determina se eles são iguais e, em caso contrário, indica se o número X é maior que o Y.

As saídas possíveis do sistema são A e B, tal que A = 1 apenas se os dois números binários de entrada (X e Y) forem iguais; e B = 1 apenas se o número binário X for maior que Y.

Assim, o circuito a ser montado deve possuir quatro entradas para os números, duas para X e duas para Y. Ele também deve possuir duas saídas, uma para A e outra para B.

3.2 Materiais

O presente projeto fez uso dos seguintes equipamentos:

- Protoboard;
- Fonte de tensão DC;
- Fios e conexões;
- Dois resistores de 290Ω ;
- Um LED verde;
- Um LED vermelho;
- Um CI 7404 (NOT);
- Um CI 7486 (XOR);
- Um CI 7432 (OR);
- Dois CI's 7408 (AND).

As quantidades e os usos de cada um dos CI's serão explicitado no tópico a seguir.

3.3 Métodos

O primeiro passo na montagem do circuito é estabelecer sua lógica combinacional. Com isso em vista, fora desenvolvida a tabela-verdade para cada uma das saídas (A e B). Como A e B vão depender de das duas palavras de dois bits, X e Y, a tabela verdade vai possuir 4 entradas.

A condição para A só será verdadeira quando o par de bits de um número for igual ao par de bits do outro, assim permitindo preencher a tabela-verdade de acordo com a configuração da figura 7 na penúltima coluna. A última coluna é para B e está preenchida com 1s apenas quando o número X é maior que o Y, por exemplo: $B = 1$, se as palavras forem, em binário, $X = 01$ e $Y = 00$ ou $X = 11$ e $Y = 01$, e assim por diante seguindo a lógica, permitindo preencher a última coluna da tabela-verdade da figura 7.

X_1	X_0	Y_1	Y_0	A	B
0	0	0	0	1	0
0	0	0	1	0	0
0	0	1	0	0	0
0	0	1	1	0	0
0	1	0	0	0	1
0	1	0	1	1	0
0	1	1	0	0	0
0	1	1	1	0	0
1	0	0	0	0	1
1	0	0	1	0	1
1	0	1	0	1	0
1	0	1	1	0	0
1	1	0	0	0	1
1	1	0	1	0	1
1	1	1	0	0	1
1	1	1	1	1	0

Figura 7 - Tabela-verdade para as saídas A e B.

Com a tabela-verdade pronta, o próximo passo do projeto é definir a equação booleana - de preferência simplificada - para que possa seguir ao último passo: implementar um circuito baseado em portas lógicas.

O método mais simples utilizado foi o do mapa de Karnaugh com quatro variáveis. Assim, permitindo dispor as combinações das entradas da tabela verdade em uma matriz que permite a simplificação da equação booleana para a saída.

A figura 8 mostra o mapa de Karnaugh para a saída A. Apesar de seu resultado ser esperado, com o preenchimento apenas da diagonal principal com 1s devido à sua lógica combinacional, ainda fora utilizado o mapa de Karnaugh para tornar mais evidente o tipo de situação. Assim não é permitido fazer nenhuma simplificação para remover variáveis, mas pela lógica, dá para implementar portas XOR. A equação booleana resultante pode ser constatada na expressão abaixo:

$$A = (X_1 \oplus Y_1)' * (X_0 \oplus Y_0)'$$

A	$\bar{Y}_1\bar{Y}_0$	\bar{Y}_1Y_0	Y_1Y_0	$Y_1\bar{Y}_0$
$\bar{X}_1\bar{X}_0$	1	0	0	0
\bar{X}_1X_0	0	1	0	0
X_1X_0	0	0	1	0
$X_1\bar{X}_0$	0	0	0	1

Figura 8 - Mapa-K para a saída A.

A figura 9 mostra o mapa de Karnaugh para a saída B. Diferente da saída A, esta possui grupos de 1s que podem ser agrupados para simplificação. A equação booleana para a saída B está

disposta logo abaixo. Na parte inferior da tabela podem ser agrupados quatro 1s, eliminando as variáveis X_0 e Y_0 resultando no primeiro mintermo da expressão abaixo. O segundo mintermo pode ser encontrado agrupando os dois 1s acima do grupo anterior, eliminando apenas a variável X_1 . O último mintermo foi encontrado agrupando o 1 restante presente na borda com o da outra extremidade, eliminando a variável Y_1 .

$$B = X_1 Y'_1 + X_0 Y'_1 Y'_0 + X_1 X_0 Y'_0$$

B	$\bar{Y}_1 \bar{Y}_0$	$\bar{Y}_1 Y_0$	$Y_1 Y_0$	$Y_1 \bar{Y}_0$
$\bar{X}_1 \bar{X}_0$	0	0	0	0
$\bar{X}_1 X_0$	1	0	0	0
$X_1 X_0$	1	1	0	1
$X_1 \bar{X}_0$	1	1	0	0

Figura 9 - Mapa-K para a saída B.

Com as equações booleanas montadas e simplificadas, o próximo passo do projeto é implementar o circuito com portas lógicas a partir delas. O circuito da figura 10 resulta na saída A e o circuito da figura 11 resulta na saída B.

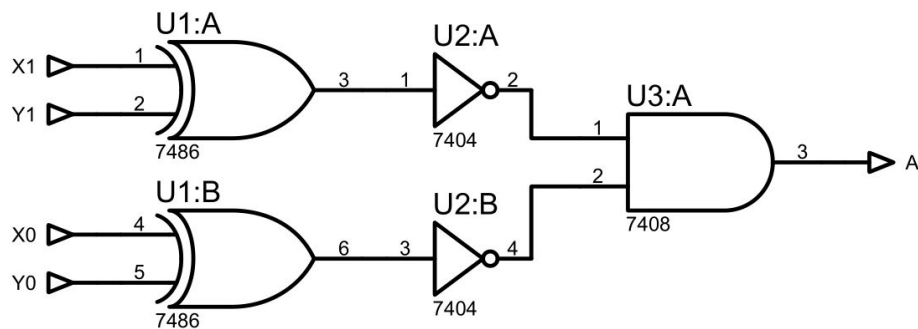


Figura 10 - Circuito para a saída A.

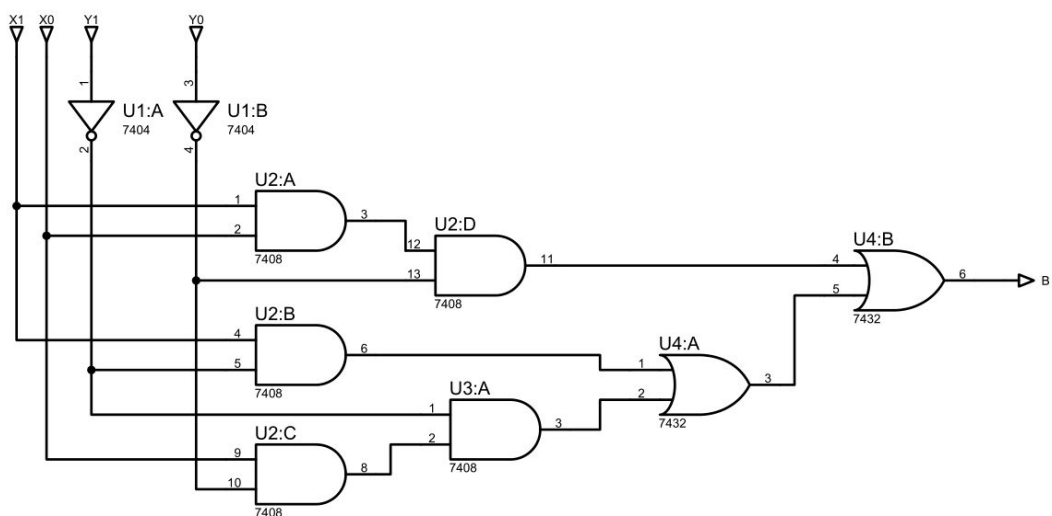


Figura 11 - Circuito para a saída B.

Com o projeto pronto, basta executar a montagem na protoboard tendo em vista a quantidade de CI's que serão utilizados, previamente definidos no tópico anterior.

Contabilizando ambos os circuitos, são utilizados apenas 4 porta NOT (inversoras), assim como cada CI 7404 dispõe de 6 portas, basta utilizar apenas um CI. A quantidade de portas XOR são apenas duas, como cada CI 7486 contém 4 dessas portas, basta utilizar um CI. O mesmo se aplica para as 2 portas OR e o uso de apenas um CI 7432. Já as portas AND estão em maior quantidade, possuindo 6, mas cada CI 7408 só contém 4 dessas portas, sendo necessário utilizar dois.

Por fim, a saída de A foi conectada em série com um resistor e um LED verde. A saída B foi conectada em série com um resistor e um LED vermelho. Ambos os resistores possuem um valor arbitrário de 290Ω com a finalidade de limitar a corrente enviada para o LED. O circuito final na protoboard pode ser observado na figura 12.

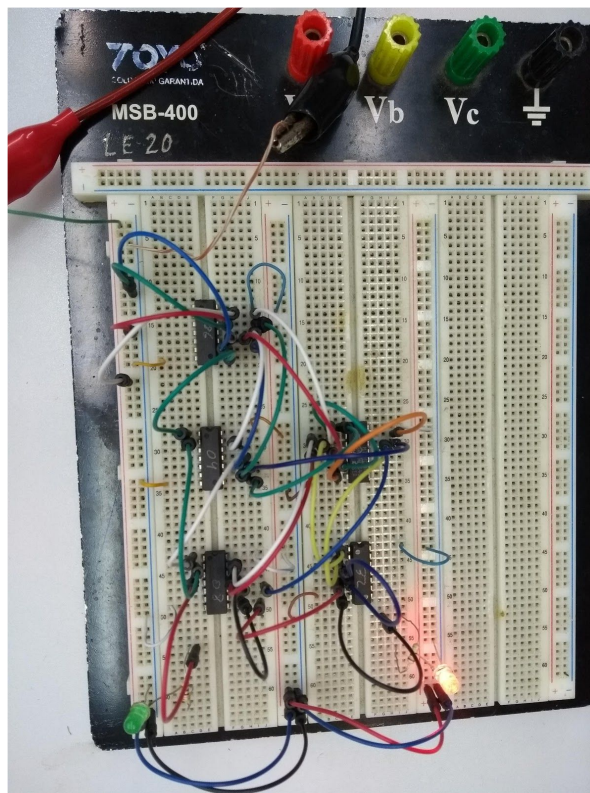


Figura 12 - Circuitos montados no protoboard.

4. Resultados práticos

Na expressão do circuito da saída A, obtemos um fácil entendimento de seu funcionamento. A porta NXOR, que foi feita por meio de uma NOT após uma XOR, verifica a igualdade dos bits. Dessa forma, para comparar as entradas X e Y, comparamos o valor de X_1 com Y_1 e de X_2 com Y_2 , assim, se ambos os valores são iguais, verificando com uma porta AND no final, o número X é igual ao número Y.

Já na expressão do circuito da saída B, na primeira AND, ele nos mostra que sempre que a entrada X_1 é 1 e Y_1 é 0, sua saída será 1. Isso mostra que 10 e 11 serão sempre maiores que 00 e 01. Na segunda AND, ele garante que qualquer valor de X que não seja 00 será maior que Y quando ele for 00, e na última AND, garante que X igual a 11 será maior que qualquer Y, desde que não seja 11 também.

É importante notar que os circuitos são independentes, de forma que um não irá interferir no outro. Além disso, são exclusivos, ou seja, quando a saída de um circuito é 1, a outra não pode ser 1, o que facilita na hora de encontrar erros.

Após a montagem do circuito, foram testados alguns valores indicados pelo professor e por meio da indicação dos LEDs, vimos se os valores eram iguais ou com $X > Y$.

5. Conclusão

Os resultados obtidos neste projeto satisfizeram a expectativa e se mostraram iguais aos teóricos. As formas de simplificação de circuitos, como Mapa-K, ajudaram de forma satisfatória na criação do circuito.

6. Referências Bibliográficas

ABNT, Associação Brasileira de Normas Técnicas. **NBR 10719 – Apresentação de relatórios técnico-científicos**. Rio de Janeiro: ABNT, Copyright © 1989.

ELECTRONICS HUB. **Digital Comparator and Magnitude Comparator**. Disponível em: <<https://www.electronicshub.org/digital-comparator-and-magnitude-comparator/>>, acesso em 24 de março de 2019.

Fairchild Semiconductor, “**Hex Inverting Gates**,” DM7404 datasheet, Aug. 1986 [Revised Feb. 2000].

Fairchild Semiconductor, “**Quad 2-Input AND Gates**,” DM7408 datasheet, Aug. 1986 [Revised Jul. 2001].

Fairchild Semiconductor, “**Quad 2-Input OR Gate**,” DM74LS32 datasheet, Jun. 1986 [Revised Mar. 2000].

MARCONI, Marina de A. & LAKATOS, Eva M. **Fundamentos de metodologia científica**. 5 ed. Editora Atlas. São Paulo, 2003.

Nexperia, “**Quad 2-input EXCLUSIVE-OR gate**,” 74HC86 datasheet, Dec. 2015 [Revised Dec. 2015].