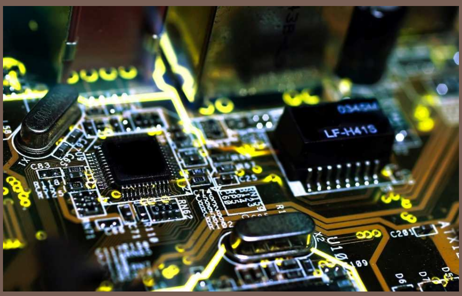


SISTEMAS DIGITAIS

Circuitos combinatórios

Circuitos combinatórios

2

Sumário:

- Sistemas de numeração
- Códigos
- Codificadores
- Descodificadores
- *Multiplexers*
- *Desmultiplexers*
- Comparadores
- Somadores
- Subtractores

Circuitos combinatórios

3

Sistemas de numeração

Sistemas de Numeração são formas de **representação das grandezas quantitativas**.

Exemplos de sistemas de numeração:

- Decimal (0,1,2,3,4,5,6,7,8,9)
- Binário (0,1)
- Octal (0,1,2,3,4,5,6, 7)
- Hexadecimal (0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F)

Circuitos combinatórios

4

Sistemas de numeração Binário

- Neste sistema de numeração utilizam-se somente dois símbolos (0,1);
- Normalmente designa-se por sistema de numeração de base 2 ou binário natural;
- Cada dígito binário designa-se por bit;
- Cada coluna do código binário tem um peso diferente ($2^{\text{nº de ordem da coluna}}$);
- Os diferentes pesos são somados por forma a obter o número desejado.

$5 = 4 + 1$

Código Decimal	Código Binário				
	2^4	2^3	2^2	2^1	2^0
	16	8	4	2	1
5	0	0	1	0	1

peso

Código Decimal	Código Binário				
	2^4	2^3	2^2	2^1	2^0
	16	8	4	2	1
0	0	0	0	0	0
1	0	0	0	0	1
2	0	0	0	1	0
3	0	0	0	1	1
4	0	0	1	0	0
5	0	0	1	0	1
6	0	0	1	1	0
7	0	0	1	1	1
8	0	1	0	0	0
9	0	1	0	0	1
10	0	1	0	1	0
....	....	...	....	...	....

Circuitos combinatórios

5

Sistemas de numeração Octal

- Este sistema também pode ser designado por base 8;
- Só tem 8 dígitos: 0, 1, 2, 3, 4, 5, 6, 7.
- As oito combinações do código octal são representadas da mesma forma que as primeiras oito combinações do código decimal.

Código Decimal	Código Octal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7

Como distinguir representações iguais de sistemas de numeração diferentes?

- $1_{(10)}$ → Número 1 na base 10
- $1_{(8)}$ → Número 1 na base 8

Circuitos combinatórios

6

Sistemas de numeração Hexadecimal

- Este sistema também pode ser designado por base 16;
- Só tem 16 dígitos: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F;
- As dez primeiras combinações do código hexadecimal são representadas da mesma forma que as primeiras dez combinações do código decimal.

Como distinguir representações iguais de sistemas de numeração diferentes?

- $1_{(10)}$ → Número 1 na base 10
- $1_{(16)}$ → Número 1 na base 16

Código Decimal	Código Hexadecimal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	A
11	B
12	C
13	D
14	E
15	F

Circuitos combinatórios

7

É possível representar a mesma grandeza nos vários sistemas de numeração.

O número $8_{(10)}$ no sistema de numeração decimal é representado por:

- $10_{(8)}$ → no sistema de numeração octal
- $8_{(16)}$ → no sistema de numeração hexadecimal
- $1000_{(2)}$ → no sistema de numeração binário

Código Decimal	Código Octal	Código Hexadecimal	Código Binário				
			2^4	2^3	2^2	2^1	2^0
			16	8	4	2	1
0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	1
2	2	2	0	0	0	1	0
3	3	3	0	0	0	1	1
4	4	4	0	0	1	0	0
5	5	5	0	0	1	0	1
6	6	6	0	0	1	1	0
7	7	7	0	0	1	1	1
8	10	8	0	1	0	0	0
9	11	9	0	1	0	0	1
10	12	A	0	1	0	1	0
11	13	B	0	1	0	1	1
12	14	C	0	1	1	0	0
13	15	D	0	1	1	0	1
14	16	E	0	1	1	1	0
15	17	F	0	1	1	1	1

Circuitos combinatórios

8

Como converter da base 2 para a base 8?

- Divide-se o número binário em grupos de três, da direita para a esquerda;
- A soma ponderada entre cada dígito e o respectivo peso, dá um algarismo no sistema octal.

Converter o seguinte número representado na base 2 para a base 8.

$$110011_{(2)}$$

$$\underline{110}\,\underline{011}_{(2)}$$

$$011_{(2)} = 0*2^2 + 1*2^1 + 1*2^0 = 2+1 = 3_{(8)}$$

$$110_{(2)} = 1*2^2 + 1*2^1 + 0*2^0 = 4+2 = 6_{(8)}$$

$$110011_{(2)} = 63_{(8)}$$

Circuitos combinatórios

9

Como converter da base 2 para a base 10?

- A soma ponderada entre cada dígito e o respectivo peso, dá um algarismo no sistema decimal;
- Expoentes à esquerda da vírgula são positivos e expoentes à direita da vírgula são negativos.

Converter o seguinte número representado na base 2 para a base 10.

$$1101,011_{(2)}$$

$$\begin{aligned} 1101,011_{(2)} &= 1*2^3 + 1*2^2 + 0*2^1 + 1*2^0 + 0*2^{-1} + 1*2^{-2} + 1*2^{-3} = \\ &= 8 + 4 + 0 + 1 + 0 + 0,25 + 0,125 = \\ &= 13,375_{(10)} \end{aligned}$$

Circuitos combinatórios

10

Como converter da base 2 para a base 16?

- Divide-se o número binário em grupos de quatro, da direita para a esquerda;
- A soma ponderada entre cada dígito e o respectivo peso, dá um algarismo no sistema hexadecimal.

Converter o seguinte número representado na base 2 para a base 16.

$$110001011010_{(2)}$$

$$\underbrace{1100}\,\underbrace{0101}\,\underbrace{1010}_{(2)}$$

$$1010_{(2)} = 1*2^3 + 0*2^2 + 1*2^1 + 0*2^0 = 8 + 2 = A_{(16)}$$

$$0101_{(2)} = 0*2^3 + 1*2^2 + 0*2^1 + 1*2^0 = 4 + 1 = 5_{(16)}$$

$$1100_{(2)} = 1*2^3 + 1*2^2 + 0*2^1 + 0*2^0 = 8 + 4 = C_{(16)}$$

$$110001011010_{(2)} = C5A_{(16)}$$

Circuitos combinatórios

11

Conversão	Regra para a conversão
Base 2 → Base 8	• Divide-se o número binário em grupos de três, da direita para a esquerda; • A soma ponderada entre cada dígito e o respectivo peso, dá um algarismo no sistema octal.
Base 2 → Base 10	• A soma ponderada entre cada dígito e o respectivo peso, dá um algarismo no sistema decimal; • Expoentes à esquerda da vírgula são positivos e expoentes à direita da vírgula são negativos.
Base 2 → Base 16	• Divide-se o número binário em grupos de quatro, da direita para a esquerda; • A soma ponderada entre cada dígito e o respectivo peso, dá um algarismo no sistema hexadecimal.

Circuitos combinatórios

12

Como converter da base 8 para a base 2?

- É necessário recorrer às tabelas dos sistemas de numeração de base 8 e de base 2;
- Converte-se cada algarismo em binário com três dígitos.

Converter o seguinte número representado na base 8 para a base 2.

$$63_{(8)} = 110011_{(2)}$$

Código Octal	Código Binário		
	4	2	1
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Circuitos combinatórios

Como converter da base 8 para a base 10?

- A soma ponderada entre cada dígito e o respectivo peso, dá um algarismo no sistema decimal;
- A base do expoente é agora 8;
- Expoentes à esquerda da vírgula são positivos e expoentes à direita da vírgula são negativos.

Converter o seguinte número representado na base 8 para a base 10.

$$30,5_{(8)}$$

$$30,5_{(8)} = 3*8^1 + 0*8^0 + 5*8^{-1} = 24,625_{(10)}$$

Circuitos combinatórios

Como converter da base 8 para a base 16?

- Primeiro converte-se da base 8 para a base 10;
- Depois converte-se da base 10 para a base 16, através do método das divisões sucessivas.

Converter o seguinte número representado na base 8 para a base 16.

$$30_{(8)}$$

Primeiro converte-se da base 8 para a base 10.

$$30_{(8)} = 3*8^1 + 0*8^0$$
$$= 24_{(10)}$$
$$30_{(8)} = 24_{(10)}$$

Segundo converte-se da base 10 para a base 16.

$$\begin{array}{l|l} 24_{(10)} & 16 \\ \hline 8 & 1 \end{array}$$

Método das divisões sucessivas

$$24_{(10)} = 18_{(16)}$$

$$30_{(8)} = 18_{(16)}$$

Circuitos combinatórios

15

Conversão	Regra para a conversão
Base 8 → Base 2	• É necessário recorrer às tabelas dos sistemas de numeração de base 8 e de base 2; • Converte-se cada algarismo em binário com três dígitos.
Base 8 → Base 10	• A soma ponderada entre cada dígito e o respectivo peso, dá um algarismo no sistema decimal; • A base do expoente é 8; • Expoentes à esquerda da vírgula são positivos e expoentes à direita da vírgula são negativos.
Base 8 → Base 16	• Primeiro converte-se da base 8 para a base 10; • Depois converte-se da base 10 para a base 16, através do ***Método das Divisões Sucessivas***.

Circuitos combinatórios

16

Como converter da base 10 para a base 2?

- Parte inteira:
  - Divide-se sucessivamente o número representado no sistema decimal por 2 até que o valor do quociente seja menor que o divisor;
  - O resto obtido e o último quociente constituem o número no sistema binário;
  - A leitura do número do sistema binário é efectuada da direita para a esquerda.

- Parte fraccionária:
  - Multiplica-se por dois a parte fraccionária, até que a parte fraccionária do resultado seja igual a zero;
  - Os resultados de todas as partes inteiras constituem o número no sistema binário;
  - A leitura do número no sistema binário é efectuada da esquerda para a direita.

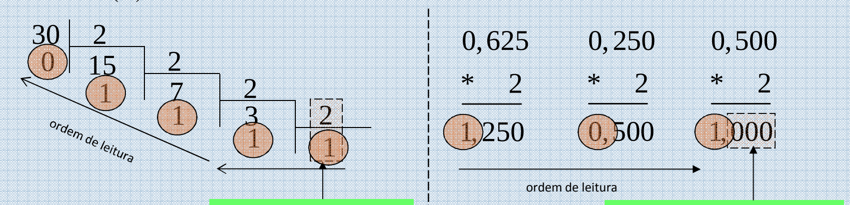
Circuitos combinatórios

17

Como converter da base 10 para a base 2?

Converter o seguinte número representado na base 10 para a base 2.

$30,625_{(10)}$



$30_{(10)} = 11110_{(2)}$

$,625_{(10)} = ,101_{(2)}$

$$30,625_{(10)} = 11110,101_{(2)}$$

Circuitos combinatórios

18

Como converter da base 10 para a base 8?

- Parte inteira:
  - Divide-se sucessivamente o número representado no sistema decimal por 8 até que o valor do quociente seja menor que o divisor;
  - O resto obtido e o último quociente constituem o número no sistema octal;
  - A leitura do número do sistema octal é efectuada da direita para a esquerda.

- Parte fraccionária:
  - Multiplica-se por oito a parte fraccionária, até que a parte fraccionária do resultado seja igual a zero;
  - Os resultados de todas as partes inteiras constituem o número no sistema octal;
  - A leitura do número no sistema octal é efectuada da esquerda para a direita.

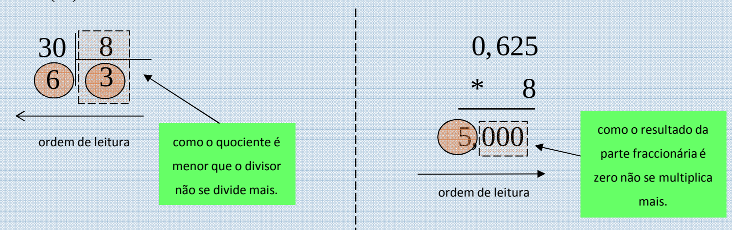
Circuitos combinatórios

19

Como converter da base 10 para a base 8?

Converter o seguinte número representado na base 10 para a base 8.

$30,625_{(10)}$



$30_{(10)} = 36_{(8)}$

$,625_{(10)} = ,5_{(8)}$

$30,625_{(10)} = 36,5_{(8)}$

Circuitos combinatórios

20

Como converter da base 10 para a base 16?

- Parte inteira:
  - Divide-se sucessivamente o número representado no sistema decimal por 16 até que o valor do quociente seja menor que o divisor;
  - O resto obtido e o último quociente constituem o número no sistema hexadecimal;
  - A leitura do número do sistema hexadecimal é efectuada da direita para a esquerda.

- Parte fraccionária:
  - Multiplica-se por dezasseis a parte fraccionária, até que a parte fraccionária do resultado seja igual a zero;
  - Os resultados de todas as partes inteiras constituem o número no sistema hexadecimal;
  - A leitura do número no sistema hexadecimal é efectuada da esquerda para a direita.

Circuitos combinatórios

21

Como converter da base 10 para a base 16?

Converter o seguinte número representado na base 10 para a base 6.

$$30,625_{(10)}$$

$$\begin{array}{r|l} 30 & \boxed{16} \\ \hline 14 & \boxed{1} \end{array}$$

como o quociente é menor que o divisor não se divide mais.

ordem de leitura

$$30_{(10)} = 1E_{(16)}$$

$$\begin{array}{r} 0,625 \\ *\quad 16 \\ \hline 10,\boxed{000} \end{array}$$

como o resultado da parte fraccionária é zero não se multiplica mais.

ordem de leitura

$$,625_{(10)} =, A_{(16)}$$

$$30,625_{(10)} = 1E,A_{(16)}$$

Circuitos combinatórios

22

Conversão	Regra para a conversão
Base 10 → Base 2	• Parte inteira: • Método das divisões sucessivas (divisor = 2). • Parte fraccionária: • Método das multiplicações sucessivas (multiplicador = 2).
Base 10 → Base 8	• Parte inteira: • Método das divisões sucessivas (divisor = 8). • Parte fraccionária: • Método das multiplicações sucessivas (multiplicador = 8).
Base 10 → Base 16	• Parte inteira: • Método das divisões sucessivas (divisor = 16). • Parte fraccionária: • Método das multiplicações sucessivas (multiplicador =16).

Circuitos combinatórios

23

Como converter da base 16 para a base 2?

- É necessário recorrer às tabelas dos sistemas de numeração de base 16 e de base 2;
- Converte-se cada algarismo em binário com quatro dígitos.

Código Hexadecimal	Código Binário				
	2^4	2^3	2^2	2^1	2^0
	16	8	4	2	1
0	0	0	0	0	0
1	0	0	0	0	1
2	0	0	0	1	0
3	0	0	0	1	1
4	0	0	1	0	0
5	0	0	1	0	1
6	0	0	1	1	0
7	0	0	1	1	1
8	0	1	0	0	0
9	0	1	0	0	1
A	0	1	0	1	0
B	0	1	0	1	1
C	0	1	1	0	0
D	0	1	1	0	1
E	0	1	1	1	0
F	0	1	1	1	1

Converter o seguinte número representado na base 16 para a base 2.

$$C5A_{(16)} = 110001011010_{(2)}$$

Circuitos combinatórios

24

Como converter da base 16 para a base 8?

- Primeiro converte-se da base 16 para a base 10;
- Depois converte-se da base 10 para a base 8, através do método das divisões sucessivas.

Converter o seguinte número representado na base 16 para a base 8.

$30_{(16)}$

Primeiro converte-se da base 16 para a base 10.

$$30_{(16)} = 3*16^1 + 0*16^0$$
$$= 48_{(10)}$$
$$30_{(16)} = 48_{(10)}$$

Segundo converte-se da base 10 para a base 8.

$$\begin{array}{r|l} 48_{(10)} & 8 \\ \hline 0 & 6 \end{array}$$

Método das divisões sucessivas

$$48_{(10)} = 60_{(8)}$$

$$30_{(16)} = 60_{(8)}$$

Circuitos combinatórios

25

Como converter da base 16 para a base 10?

- A soma ponderada entre cada dígito e o respectivo peso, dá um algarismo no sistema decimal;
- A base do expoente é agora 16;
- Expoentes à esquerda da vírgula são positivos e expoentes à direita da vírgula são negativos.

Converter o seguinte número representado na base 16 para a base 10.

$$30,5_{(16)}$$

$$30,5_{(16)} = 3*16^{1} + 0*16^{0} + 5*16^{-1} = 48,3125_{(10)}$$

Circuitos combinatórios

26

Conversão	Regra para a conversão
Base 16 → Base 2	• É necessário recorrer às tabelas dos sistemas de numeração de base 16 e de base 2; • Converte-se cada algarismo em binário com quatro dígitos.
Base 16 → Base 8	• Primeiro converte-se da base 16 para a base 10; • Depois converte-se da base 10 para a base 8, através do ***Método das Divisões Sucessivas***.
Base 16 → Base 10	• A soma ponderada entre cada dígito e o respectivo peso, dá um algarismo no sistema decimal; • A base do expoente é agora 16; • Expoentes à esquerda da vírgula são positivos e expoentes à direita da vírgula são negativos.

Circuitos combinatórios

27

Códigos

- Um código é um conjunto de unidades de informação relacionadas de forma sistemática e biunívoca com outro conjunto de sinais e símbolos segundo determinadas regras de tradução pré – fixadas;
- Os códigos utilizados nos sistemas digitais são binários, isto é, combinações de 1’s e 0’s;
- Os 1’s e 0´s representam níveis altos ou baixos de tensão;
- O processo de transformação de informação perceptível aos seres humanos em informação perceptível aos sistemas digitais é denominado por ***codificação***;
- O processo de transformação de informação perceptível aos sistemas digitais em informação perceptível aos seres humanos é denominado por ***descodificação***.

Circuitos combinatórios

28

Código Decimal Codificado em Binário Natural - BCD Natural

- É um código baseado nas primeiras 16 combinações do código binário;
- É um código que é construído com as primeiras 10 combinações do código binário por ordem crescente (0, ..., 9).

	Código BCD Natural			
	2^3	2^2	2^1	2^0
	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Código Decimal	Código Binário			
	2^3	2^2	2^1	2^0
	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Circuitos combinatórios

29

Código Decimal Codificado em Binário Excesso 3 - BCD Excesso 3

- É um código baseado nas primeiras 16 combinações do código binário;
- É um código que não utiliza as primeiras três combinações nem as três últimas combinações do código binário.

	Código BCD Excesso 3			
	2^3	2^2	2^1	2^0
	8	4	2	1
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0

Código Decimal	Código Binário			
	2^3	2^2	2^1	2^0
	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Circuitos combinatórios

30

Código Decimal Codificado em Binário Aiken - BCD Aiken

- É um código baseado nas primeiras 16 combinações do código binário;
- É um código que utiliza as primeiras cinco combinações e as cinco últimas combinações do código binário.

	Código BCD Aiken			
	2^3	2^2	2^1	2^0
	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Código Decimal	Código Binário			
	2^3	2^2	2^1	2^0
	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Circuitos combinatórios

31

Código progressivo Gray

- A característica fundamental dos códigos progressivos é que uma combinação difere da combinação anterior e da combinação seguinte exclusivamente num bit;
- É um código cíclico porque a última combinação difere da combinação anterior e da primeira combinação exclusivamente num bit;
- Podem existir vários códigos Gray.

Código Gray			
0	0	0	0
0	0	0	1
0	0	1	1
0	0	1	0
0	1	1	0
0	1	1	1
0	1	0	1
0	1	0	0
1	1	0	0
1	1	0	1
1	1	1	1
1	1	1	0
1	0	1	0
1	0	1	1
1	0	0	1
1	0	0	0

Circuitos combinatórios

32

Códigos detectores de erros

- Existem códigos complexos que detectam os erros na informação;
- Nos códigos mais vulgares, o erro é detectado se ocorre apenas num bit da combinação, uma vez que a probabilidade de haver erro na transmissão em dois bits é muito pequena;
- O número mínimo de bit que estes códigos usam é de cinco;
- Os códigos detectores mais utilizados são os de ***paridade par*** e os de ***paridade ímpar***;
- Estes códigos formam-se acrescentando mais um bit aos códigos da família BCD;
- O bit paridade é gerado por um circuito denominado por gerador de paridade, que é construído com portas OU Exclusivo.

Circuitos combinatórios

33

Códigos detectores de erros de paridade ímpar

- A detecção é realizada (através de um circuito detector formado por portas OU Exclusivo) de maneira que o número de 1's é sempre ímpar;
- Nos códigos de paridade ímpar o número de 1's tem de ser ímpar, entrando em conta com o bit de paridade;
- Estes códigos são sempre baseados nos códigos BCD;
- No exemplo da figura o código utilizado foi o código BCD Excesso 3.

	Código Detector de Erros de Paridade Ímpar				
	Código BCD Excesso 3				Bit de Paridade
	2^3	2^2	2^1	2^0	
	8	4	2	1	
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	0
8	1	0	0	0	0
9	1	0	0	1	1
10	1	0	1	0	1
11	1	0	1	1	0
12	1	1	0	0	1

Circuitos combinatórios

34

Códigos detectores de erros de paridade par

- A detecção é realizada (através de um circuito detector formado por portas OU Exclusivo) de maneira que o número de 1's é sempre par;
- Nos códigos de paridade paro número de 1's tem de ser par, entrando em conta com o bit de paridade;
- Estes códigos são sempre baseados nos códigos BCD;
- No exemplo da figura o código utilizado foi o código BCD Excesso 3.

	Código Detector de Erros de Paridade Par				
	Código BCD Excesso 3				Bit de Paridade
	2^3	2^2	2^1	2^0	
	8	4	2	1	
3	0	0	1	1	0
4	0	1	0	0	1
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	0

Circuitos combinatórios

35

Código ASCII

- É usado para representar informação de letras, números e sinais especiais;
- Existem códigos ASCII de 6 e 7 bits, mais 1 bit de paridade para detecção de erros;
- Permitem diversas ordens de controlo de periféricos (impressoras, monitores, etc.).

P	b_7	b_6	b_5	b_4	b_3	b_2	b_1

Formato da combinação do código ASCII

				b_7	0	0	0	0	1	1	1	1
				b_6	0	0	1	1	0	0	1	1
				b_5	0	1	0	1	0	1	0	1
b_4	b_3	b_2	b_1		0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	`	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	10	LF	SUB	*	:	J	Z	j	z
1	0	1	1	11	VT	ESC	+	;	K	[	k	{
1	1	0	0	12	FF	FS	,	<	L	\	l	|
1	1	0	1	13	CR	GS	–	=	M	]	m	}
1	1	1	0	14	SO	RS	.	>	N	^	n	~
1	1	1	1	15	SI	US	/	?	O	_	o	DEL

Circuitos combinatórios

36

Códigos detectores e correctores de erros

- Existem códigos complexos que detectam e corrigem os erros na informação;
- Nos códigos mais vulgares, o erro é detectado se ocorre apenas num bit da combinação, uma vez que a probabilidade de haver erro na transmissão em dois bits é muito pequena;
- O código detector e corrector mais utilizado é o código de ***Hamming***;
- O código de Hamming é construído sobre os códigos da família BCD;

Circuitos combinatórios

37

Código detector e corrector de erros - Hamming

- O código de Hamming dá-nos o lugar do bit incorrecto e através de um circuito adequado pode corrigir automaticamente a falha na informação recebida;
- O código de Hamming é formado por sete bits;
- O código de Hamming é construído a partir da família BCD;

	Código BCD Natural			
	2^3	2^2	2^1	2^0
	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Circuitos combinatórios

38

- As colunas B7, B6, B5 e B3 correspondem ao código BCD Natural;
- As colunas B4, B2 e B1 são preenchidas separadamente;
- As colunas B4, B2 e B1 são construídas por forma a que o número de 1's seja par em cada uma das seguintes combinações:
  - B1 – B3 – B5 – B7
  - B2 – B3 – B6 – B7
  - B4 – B5 – B6 – B7

	Código de Hamming						
	B7	B6	B5	B4	B3	B2	B1
	2^3	2^2	2^1		2^0		
	8	4	2		1		
0	0	0	0	0	0	0	0
1	0	0	0	0	1	1	1
2	0	0	1	1	0	0	1
3	0	0	1	1	1	1	0
4	0	1	0	1	0	1	0
5	0	1	0	1	1	0	1
6	0	1	1	0	0	1	1
7	0	1	1	0	1	0	0
8	1	0	0	1	0	1	1
9	1	0	0	1	1	0	0

Circuitos combinatórios

39

- As colunas B7, B6, B5 e B3 correspondem ao código BCD Natural;
- As colunas B4, B2 e B1 são preenchidas separadamente;
- As colunas B4, B2 e B1 são construídas por forma a que o número de 1’s seja par em cada uma das seguintes combinações:
  - **B1 – B3 – B5 – B7**
  - B2 – B3 – B6 – B7
  - B4 – B5 – B6 – B7

	Código de Hamming						
	B7	B6	B5	B4	B3	B2	B1
	2^3	2^2	2^1		2^0		
	8	4	2		1		
0	0		0		0		0
1	0		0		1		1
2	0		1		0		1
3	0		1		1		0
4	0		0		0		0
5	0		0		1		1
6	0		1		0		1
7	0		1		1		0
8	1		0		0		1
9	1		0		1		0

Circuitos combinatórios

40

- As colunas B7, B6, B5 e B3 correspondem ao código BCD Natural;
- As colunas B4, B2 e B1 são preenchidas separadamente;
- As colunas B4, B2 e B1 são construídas por forma a que o número de 1’s seja par em cada uma das seguintes combinações:
  - B1 – B3 – B5 – B7
  - **B2 – B3 – B6 – B7**
  - B4 – B5 – B6 – B7

	Código de Hamming						
	B7	B6	B5	B4	B3	B2	B1
	2^3	2^2	2^1		2^0		
	8	4	2		1		
0	0	0			0	0	
1	0	0			1	1	
2	0	0			0	0	
3	0	0			1	1	
4	0	1			0	1	
5	0	1			1	0	
6	0	1			0	1	
7	0	1			1	0	
8	1	0			0	1	
9	1	0			1	0	

Circuitos combinatórios

41

- As colunas B7, B6, B5 e B3 correspondem ao código BCD Natural;
- As colunas B4, B2 e B1 são preenchidas separadamente;
- As colunas B4, B2 e B1 são construídas por forma a que o número de 1's seja par em cada uma das seguintes combinações:
  - B1 – B3 – B5 – B7
  - B2 – B3 – B6 – B7
  - **B4 – B5 – B6 – B7**

	Código de Hamming						
	B7	B6	B5	B4	B3	B2	B1
	2^3	2^2	2^1		2^0		
	8	4	2		1		
0	0	0	0	0			
1	0	0	0	0			
2	0	0	1	1			
3	0	0	1	1			
4	0	1	0	1			
5	0	1	0	1			
6	0	1	1	0			
7	0	1	1	0			
8	1	0	0	1			
9	1	0	0	1			

Circuitos combinatórios

42

- As sete colunas (B1 a B7) estão relacionadas pelas seguintes equações:
  - $c_1 = b_1 \oplus b_3 \oplus b_5 \oplus b_7$
  - $c_2 = b_2 \oplus b_3 \oplus b_6 \oplus b_7$
  - $c_3 = b_4 \oplus b_5 \oplus b_6 \oplus b_7$
- Quando não há erros, o valor das funções C1, C2 e C3 é zero;
- Quando há erro, o número decimal equivalente à combinação binária C3C2C1 indicará o bit incorrecto.

	Código de Hamming						
	B7	B6	B5	B4	B3	B2	B1
	2^3	2^2	2^1		2^0		
	8	4	2		1		
0	0	0	0	0	0	0	0
1	0	0	0	0	1	1	1
2	0	0	1	1	0	0	1
3	0	0	1	1	1	1	0
4	0	1	0	1	0	1	0
5	0	1	0	1	1	0	1
6	0	1	1	0	0	1	1
7	0	1	1	0	1	0	0
8	1	0	0	1	0	1	1
9	1	0	0	1	1	0	0

Circuitos combinatórios

43

Exemplo:

- Vamos supor que ao transmitir a informação relativa ao número $3_{(10)}$ em vez de receber a informação correcta de $0011110_{(2)}$ recebemos a informação errada de $0011010_{(2)}$.
- Como provar que o erro está no terceiro bit a contar da direita?

$$c_1 = b_1 \oplus b_3 \oplus b_5 \oplus b_7 = 0 \oplus 0 \oplus 1 \oplus 0 = 1$$
$$c_2 = b_2 \oplus b_3 \oplus b_6 \oplus b_7 = 1 \oplus 0 \oplus 0 \oplus 0 = 1$$
$$c_3 = b_4 \oplus b_5 \oplus b_6 \oplus b_7 = 1 \oplus 1 \oplus 0 \oplus 0 = 0$$

$$c_3c_2c_1 = 011 = 3$$ → este valor indica que o terceiro bit a contar da direita é o bit que está errado.

	Código de Hamming						
	B7	B6	B5	B4	B3	B2	B1
	2^3	2^2	2^1		2^0		
	8	4	2		1		
0	0	0	0	0	0	0	0
1	0	0	0	0	1	1	1
2	0	0	1	1	0	0	1
3	0	0	1	1	1	1	0
4	0	1	0	1	0	1	0
5	0	1	0	1	1	0	1
6	0	1	1	0	0	1	1
7	0	1	1	0	1	0	0
8	1	0	0	1	0	1	1
9	1	0	0	1	1	0	0

Para corrigir o erro bastava usar uma por NOT para inverter o bit.

Circuitos combinatórios

44

Circuitos combinatórios

- Circuito combinatório é um circuito formado por funções lógicas elementares, que tem um conjunto de entradas e outro conjunto de saídas;
- Os valores das saídas dependem exclusivamente do valor lógico das entradas e da sua constituição interna;
- Exemplos de circuitos combinatórios: Codificadores; Descodificadores; Multiplexers; Desmultiplexers; Comparadores; Detectores de paridade; Conversores de código; etc.

Circuitos combinatórios

45

Codificadores

- Um codificador é um circuito combinatório formado por um número de entradas menor ou igual a $2^{\text{nº de saídas}}$
- Quando uma entrada adopta um determinado valor lógico, as saídas representam em binário o número de ordem da entrada que foi activada;
- O valor lógico que pode activar uma entrada é 0 ou 1.

Circuitos combinatórios

46

Entradas								Saídas			
E7	E6	E5	E4	E3	E2	E1	E0	S2	S1	S0	
								2^2	2^1	2^0	
N.º de ordem das entradas											
7	6	5	4	3	2	1	0				
X	X	X	X	X	X	X	0	0	0	0	0
X	X	X	X	X	X	0	1	0	0	1	1
X	X	X	X	X	0	1	1	0	1	0	2
X	X	X	X	0	1	1	1	0	1	1	3
X	X	X	0	1	1	1	1	1	0	0	4
X	X	0	1	1	1	1	1	1	0	1	5
X	0	1	1	1	1	1	1	1	1	0	6
0	1	1	1	1	1	1	1	1	1	1	7

N.º de saídas = 3
N.º de entradas = $2^{\text{N.º de saídas}} = 2^3 = 8$
Valor lógico responsável por activar as entradas = 0

Quando o valor lógico 0 activa a entrada E1, na saída aparece a combinação binária referente ao número de ordem da entrada que foi activada, isto é, o número 1 (001).

Circuitos combinatórios

47

Entradas								Saídas			
E7	E6	E5	E4	E3	E2	E1	E0	S2	S1	S0	
								2^2	2^1	2^0	
N.º de ordem das entradas											
7	6	5	4	3	2	1	0				
X	X	X	X	X	X	X	1	0	0	0	0
X	X	X	X	X	X	1	0	0	0	1	1
X	X	X	X	X	1	0	0	0	1	0	2
X	X	X	X	1	0	0	0	0	1	1	3
X	X	X	1	0	0	0	0	1	0	0	4
X	X	1	0	0	0	0	0	1	0	1	5
X	1	0	0	0	0	0	0	1	1	0	6
1	0	0	0	0	0	0	0	1	1	1	7

N.º de saídas = 3

N.º de entradas = $2^{\text{N.º de saídas}} = 2^3 = 8$

Valor lógico responsável por activar as entradas = 1

Quando o valor lógico 1 activa a entrada E1, na saída aparece a combinação binária referente ao número de ordem da entrada que foi activada, isto é, o número 1 (001).

Circuitos combinatórios

48

Entradas								Saídas			
E7	E6	E5	E4	E3	E2	E1	E0	S2	S1	S0	
								2^2	2^1	2^0	
N.º de ordem das entradas											
7	6	5	4	3	2	1	0				
0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	1	0	0	0	1	1
0	0	0	0	0	1	0	0	0	1	0	2
0	0	0	0	1	0	0	0	0	1	1	3
0	0	0	1	0	0	0	0	1	0	0	4
0	0	1	0	0	0	0	0	1	0	1	5
0	1	0	0	0	0	0	0	1	1	0	6
1	0	0	0	0	0	0	0	1	1	1	7

N.º de saídas = 3

N.º de entradas = $2^{\text{N.º de saídas}} = 2^3 = 8$

Valor lógico responsável por activar as entradas = 1

Em vez de X pode-se colocar o valor lógico. O que acontece é quando escrevermos a função lógica referente à tabela de verdade esta será mais extensa.

Circuitos combinatórios

49

- Existem dois tipos de codificadores:
  - ***Codificadores sem prioridade***;
  - ***Codificadores com prioridade***.

- Os ***codificadores sem prioridade***, são circuitos que não admitem a activação simultânea de mais do que uma entrada, porque se isso acontece aparecem códigos errados nas suas saídas;

- Os ***codificadores com prioridade***, são circuitos que no caso de ocorrer activação simultânea de várias das suas entradas, aparecerá nas suas saídas o código do número de ordem da entrada de maior prioridade;

- Nos codificadores com prioridade, a prioridade pode ser dada tanto às entradas de maior peso como às entradas de menor peso.

Circuitos combinatórios

50

- Codificador sem prioridade

	Entradas								Saídas			
	E7	E6	E5	E4	E3	E2	E1	E0	S2	S1	S0	
									2^2	2^1	2^0	
	N.º de ordem das entradas											
	7	6	5	4	3	2	1	0				
	0	0	0	0	0	0	0	1	0	0	0	0
	0	0	0	0	0	0	1	0	0	0	1	1
	0	0	0	0	0	1	0	0	0	1	0	2
ERRO	0	0	1	0	1	0	0	0	?	?	?	?
	0	0	0	1	0	0	0	0	1	0	0	4
	0	0	1	0	0	0	0	0	1	0	1	5
	0	1	0	0	0	0	0	0	1	1	0	6
	1	0	0	0	0	0	0	0	1	1	1	7

N.º de saídas = 3

N.º de entradas = $2^{\text{N.º de saídas}} = 2^3 = 8$

Valor lógico responsável por activar as entradas = 1

Circuitos combinatórios

51

- Codificador com prioridade à entrada de menor peso

A entrada de menor peso é a entrada E3

Entradas								Saídas			
E7	E6	E5	E4	E3	E2	E1	E0	S2	S1	S0	
								2^2	2^1	2^0	
N.º de ordem das entradas											
7	6	5	4	3	2	1	0				
0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	1	0	0	0	1	1
0	0	0	0	0	1	0	0	0	1	0	2
0	0	1	0	1	0	0	0	0	1	1	3
0	0	0	1	0	0	0	0	1	0	0	4
0	0	1	0	0	0	0	0	1	0	1	5
0	1	0	0	0	0	0	0	1	1	0	6
1	0	0	0	0	0	0	0	1	1	1	7

N.º de saídas = 3

N.º de entradas = $2^{\text{N.º de saídas}} = 2^3 = 8$

Valor lógico responsável por activar as entradas = 1

Circuitos combinatórios

52

- Codificador com prioridade à entrada de maior peso

A entrada de maior peso é a entrada E5

Entradas								Saídas			
E7	E6	E5	E4	E3	E2	E1	E0	S2	S1	S0	
								2^2	2^1	2^0	
N.º de ordem das entradas											
7	6	5	4	3	2	1	0				
0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	1	0	0	0	1	1
0	0	0	0	0	1	0	0	0	1	0	2
0	0	1	0	1	0	0	0	1	0	1	5
0	0	0	1	0	0	0	0	1	0	0	4
0	0	1	0	0	0	0	0	1	0	1	5
0	1	0	0	0	0	0	0	1	1	0	6
1	0	0	0	0	0	0	0	1	1	1	7

N.º de saídas = 3

N.º de entradas = $2^{\text{N.º de saídas}} = 2^3 = 8$

Valor lógico responsável por activar as entradas = 1

Circuitos combinatórios

53

Exemplo: Construir o diagrama lógico de um codificador de 4 entradas sem prioridade.

N.º de saídas = 2

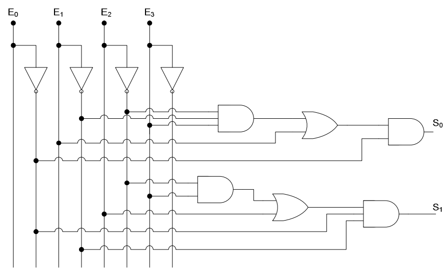
N.º de entradas = $2^{\text{N.º de saídas}} = 2^2 = 4$

Valor lógico responsável por activar as entradas = 1

Entradas				Saídas		
E3	E2	E1	E0	S1	S0	
				2^1	2^0	
X	X	X	1	0	0	0
X	X	1	0	0	1	1
X	1	0	0	1	0	2
1	0	0	0	1	1	3

$$S_0 = E_1 \bullet \overline{E_0} + E_3 \bullet \overline{E_2} \bullet \overline{E_1} \bullet \overline{E_0} = \overline{E_0} \bullet \left(E_1 + \overline{E_1} \bullet \overline{E_2} \bullet E_3\right)$$

$$S_1 = E_2 \bullet \overline{E_1} \bullet \overline{E_0} + E_3 \bullet \overline{E_2} \bullet \overline{E_1} \bullet \overline{E_0} = \overline{E_1} \bullet \overline{E_0} \bullet \left(E_2 + \overline{E_2} \bullet E_3\right)$$



Circuitos combinatórios

54

Descodificadores

- Os descodificadores realizam a função inversa dos codificadores;
- Um descodificador selecciona uma das saídas dependendo da combinação binária presente nas entradas;
- Um descodificador é um circuito combinatório formado por um número de saídas menor ou igual a $2^{\text{nº de entradas}}$
- o valor lógico que pode activar uma saída é 0 ou 1.

Circuitos combinatórios

55

	Entradas		Saídas			
	E1	E0	S3	S2	S1	S0
	2^1	2^0				
0	0	0	1	1	1	0
1	0	1	1	1	0	1
2	1	0	1	0	1	1
3	1	1	0	1	1	1

N.º de entradas = 2

N.º de saídas = $2^{\text{N.º de entradas}} = 2^2 = 4$

Valor lógico responsável por activar as saídas = 0

Quando a combinação binária referente ao número 1 (01) está presente na entrada, a saída activada será a saída com o mesmo número de ordem, isto é, S1.

	Entradas		Saídas			
	E1	E0	S3	S2	S1	S0
	2^1	2^0				
0	0	0	0	0	0	1
1	0	1	0	0	1	0
2	1	0	0	1	0	0
3	1	1	1	0	0	0

N.º de entradas = 2

N.º de saídas = $2^{\text{N.º de entradas}} = 2^2 = 4$

Valor lógico responsável por activar as saídas = 1

Quando a combinação binária referente ao número 1 (01) está presente na entrada, a saída activada será a saída com o mesmo número de ordem, isto é, S1.

Circuitos combinatórios

56

Exemplo 1: Construir o diagrama lógico de um descodificador de 2 entradas.

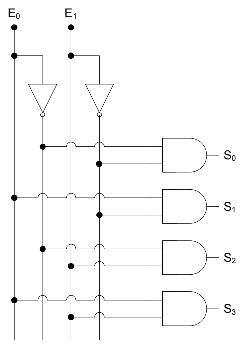
N.º de entradas = 2

N.º de saídas = $2^{\text{N.º de entradas}} = 2^2 = 4$

Valor lógico responsável por activar as saídas = 1

	Entradas		Saídas			
	E1	E0	S3	S2	S1	S0
	2^1	2^0				
0	0	0	0	0	0	1
1	0	1	0	0	1	0
2	1	0	0	1	0	0
3	1	1	1	0	0	0

$$S_0 = \overline{E_0} \bullet \overline{E_1}$$
$$S_1 = E_0 \bullet \overline{E_1}$$
$$S_2 = \overline{E_0} \bullet E_1$$
$$S_3 = E_0 \bullet E_1$$



Circuitos combinatórios

57

Exemplo 2: Dada a tabela de verdade, implementar a função lógica correspondente com descodificadores.

C	B	A	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

1º Determinar a função lógica:

$$F = \overline{C} \bullet \overline{B} \bullet A + \overline{C} \bullet B \bullet A + C \bullet \overline{B} \bullet \overline{A} + C \bullet B \bullet A$$

2º Determinar quantas entradas o nosso descodificador necessita:

N.º entradas = n.º de variáveis = 3

3º Identificar quais as combinações presentes nas entradas que levam a função lógica a tomar o valor 1 lógico:

001; 011; 100; 111

4º Identificar qual o valor e qual a saída activada dependendo das combinações presentes nas entradas:

$$001_{(2)} = 1_{(10)} = S_1; \quad 011_{(2)} = 3_{(10)} = S_3;$$
$$100_{(2)} = 4_{(10)} = S_4; \quad 111_{(2)} = 7_{(10)} = S_7$$

Circuitos combinatórios

58

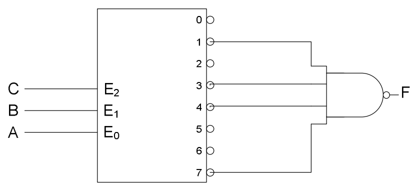
5º Escolher se a implementação vai ser levada a cabo com portas NAND ou portas OR.

As portas NAND utilizam-se para descodificadores com saídas activas por nível baixo (0 lógico).

As portas OR utilizam-se para descodificadores com saídas activas por nível alto (1 lógico).

5º Escolher se a implementação vai ser levada a cabo com portas NAND ou portas OR. (cont.)

Vamos supor que as saídas deste descodificador são activas por nível baixo (0 lógico), logo teremos que utilizar na implementação a porta lógica NAND e teremos que negar todas as saídas do descodificador.

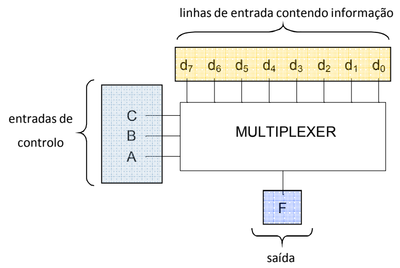


Circuitos combinatórios

59

Multiplexers

- A função do multiplexer consiste em transmitir por um só canal de saída uma das informações presentes nas várias linhas de entrada;
- A relação entre as entradas e as saídas é dada por: n.º de linhas presentes na entrada = $2^{\text{N.º de entradas de controlo}}$.



	Entradas de controlo			F
	A	B	C	
0	0	0	0	d0
1	0	0	1	d1
2	0	1	0	d2
3	0	1	1	d3
4	1	0	0	d4
5	1	0	1	d5
6	1	1	0	d6
7	1	1	1	d7

linhas de entrada contendo informação

Circuitos combinatórios

60

Exemplo 1: Implementar a função lógica F, com um multiplexer de **4 entradas de controlo**.

$$F = \overline{A}\bullet\overline{B}\bullet\overline{C}\bullet D + \overline{A}\bullet\overline{B}\bullet C\bullet D + \overline{A}\bullet B\bullet C\bullet\overline{D} +$$
$$+ \overline{A}\bullet B\bullet C\bullet D + \overline{A}\bullet B\bullet\overline{C}\bullet D + \overline{A}\bullet B\bullet\overline{C}\bullet\overline{D} +$$
$$+ A\bullet B\bullet C\bullet\overline{D} + A\bullet B\bullet\overline{C}\bullet\overline{D} + A\bullet\overline{B}\bullet\overline{C}\bullet D$$

1º Identificar quantas variáveis tem a função lógica
N.º de variáveis = 4
N.º de entradas de controlo = n.º de variáveis = 4
N.º de linhas presentes na entrada = $2^{\text{n.º de entradas de controlo}}$ = 2^4 = 16

2º Identificar qual o valor das linhas de entrada que é activo pelas diversas combinações das variáveis de controlo.

$$\overline{A}\bullet\overline{B}\bullet\overline{C}\bullet D = 0001_{(2)} = 1_{(10)} \rightarrow D_1$$
$$\overline{A}\bullet\overline{B}\bullet C\bullet D = 0011_{(2)} = 3_{(10)} \rightarrow D_3$$
$$\overline{A}\bullet B\bullet C\bullet\overline{D} = 0110_{(2)} = 6_{(10)} \rightarrow D_6$$
$$\overline{A}\bullet B\bullet C\bullet D = 0111_{(2)} = 7_{(10)} \rightarrow D_7$$
$$\overline{A}\bullet B\bullet\overline{C}\bullet D = 0101_{(2)} = 5_{(10)} \rightarrow D_5$$
$$\overline{A}\bullet B\bullet\overline{C}\bullet\overline{D} = 0100_{(2)} = 4_{(10)} \rightarrow D_4$$
$$A\bullet B\bullet C\bullet\overline{D} = 1110_{(2)} = 14_{(10)} \rightarrow D_{14}$$
$$A\bullet B\bullet\overline{C}\bullet\overline{D} = 1100_{(2)} = 12_{(10)} \rightarrow D_{12}$$
$$A\bullet\overline{B}\bullet\overline{C}\bullet D = 1001_{(2)} = 9_{(10)} \rightarrow D_9$$

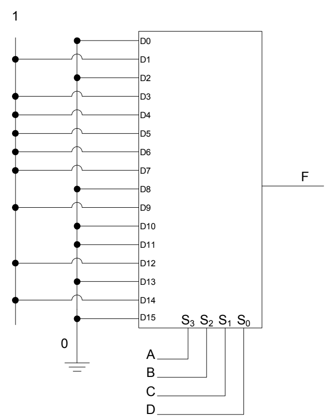
Circuitos combinatórios

61

3º Implementar a função lógica utilizando um multiplexer:

1 lógico $\longrightarrow D_1, D_3, D_4, D_5, D_6, D_7, D_9, D_{12}, D_{14}$

0 lógico $\longrightarrow D_0, D_2, D_8, D_{10}, D_{11}, D_{13}, D_{15}$



Circuitos combinatórios

62

Exemplo 2: Implementar a função lógica F, com um multiplexer de ***3 entradas de controlo***.

$$F = \overline{A}\bullet\overline{B}\bullet\overline{C}\bullet D + \overline{A}\bullet\overline{B}\bullet C\bullet D + \overline{A}\bullet B\bullet C\bullet\overline{D} + \overline{A}\bullet B\bullet C\bullet D + \overline{A}\bullet B\bullet\overline{C}\bullet D +$$
$$+ \overline{A}\bullet B\bullet\overline{C}\bullet\overline{D} + A\bullet B\bullet C\bullet\overline{D} + A\bullet B\bullet\overline{C}\bullet\overline{D} + A\bullet\overline{B}\bullet\overline{C}\bullet D$$

1º Identificar quantas variáveis tem a função lógica

N.º de variáveis = 4

N.º de entradas de controlo = n.º de variáveis = 4

N.º de linhas presentes na entrada = $2^{\text{n.º de entradas de controlo}} = 2^4 = 16$

Atenção:

O multiplexer só tem 3 entradas de controlo

N.º de linhas presentes na entrada = $2^{\text{n.º de entradas de controlo}} = 2^3 = 8$

Circuitos combinatórios

63

$$F = \overline{A}\bullet\overline{B}\bullet\overline{C}\bullet D + \overline{A}\bullet\overline{B}\bullet C\bullet D + \overline{A}\bullet B\bullet C\bullet\overline{D} + \overline{A}\bullet B\bullet C\bullet D + \overline{A}\bullet B\bullet\overline{C}\bullet D +$$
$$+\overline{A}\bullet B\bullet\overline{C}\bullet\overline{D} + A\bullet B\bullet C\bullet\overline{D} + A\bullet B\bullet\overline{C}\bullet\overline{D} + A\bullet\overline{B}\bullet\overline{C}\bullet D$$

2º Construir uma tabela na qual se representam com 1 lógico as combinações das variáveis de controlo que intervêm na função lógica e com 0 lógico as restantes combinações que não intervêm na função lógica.

A \ BCD	000	001	010	011	100	101	110	111
0	0	1	0	1	1	1	1	1
1	0	1	0	0	1	0	1	0
	D0	D1	D2	D3	D4	D5	D6	D7

3º Analisar a tabela.

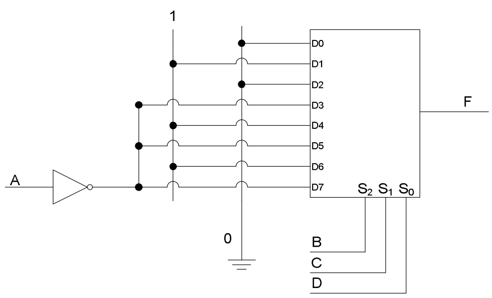
- 1 logico $\rightarrow D_1, D_4, D_6$
- 0 logico $\rightarrow D_0, D_2$
- inverso de A $\rightarrow D_3, D_5, D_7 = \overline{A}$

Circuitos combinatórios

64

4º Implementação.

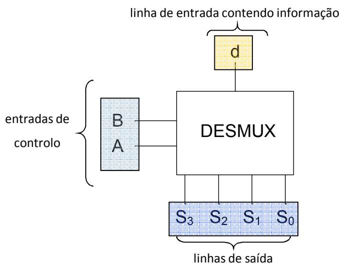
- 1 logico $\rightarrow D_1, D_4, D_6$
- 0 logico $\rightarrow D_0, D_2$
- inverso de A $\rightarrow D_3, D_5, D_7 = \overline{A}$



Circuitos combinatórios

Desmultiplexers

- Os desmultiplexers são circuitos com uma só entrada, n linhas de saídas e n entradas de controlo;
- A informação de entrada é transmitida à linha de saída selecciona pelas entradas de controlo;
- A relação entre as saídas e as entradas é dada por: n.º de linhas presentes na saída = $2^{\text{N.º de entradas de controlo}}$.

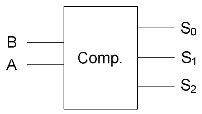


	Entradas de Controlo		Saídas			
	A	B	S0	S1	S2	S3
0	0	0	D	0	0	0
1	0	1	0	D	0	0
2	1	0	0	0	D	0
3	1	1	0	0	0	D

Circuitos combinatórios

Comparadores

- Os comparadores são circuitos combinatórios que, ao colocarmos nas suas entradas duas palavras de n bits, detectam se são ou não iguais, e neste caso qual das entradas é maior ou menor.



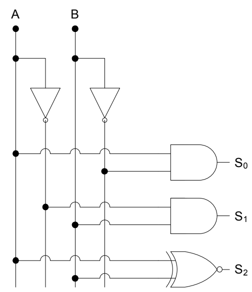
S0 = 1 lógico → A > B

S1 = 1 lógico → A < B

S2 = 1 lógico → A = B

	Entradas		Saídas		
	A	B	S0	S1	S2
0	0	0	0	0	1
1	0	1	0	1	0
2	1	0	1	0	0
3	1	1	0	0	1

$$S_0 = A \bullet \overline{B}$$
$$S_1 = \overline{A} \bullet B$$
$$S_2 = \overline{A \oplus B}$$



Circuitos combinatórios

67

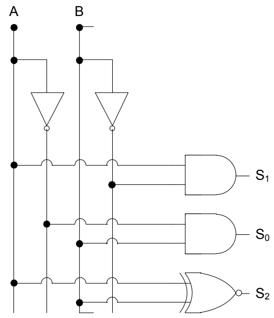
Assumindo outros pressupostos:

- S0 = 1 lógico → A < B
- S1 = 1 lógico → A > B
- S2 = 1 lógico → A = B

a tabela de verdade, as funções lógicas e o circuito são diferentes.

	Entradas		Saídas		
	A	B	S0	S1	S2
0	0	0	0	0	1
1	0	1	1	0	0
2	1	0	0	1	0
3	1	1	0	0	1

$$S_0 = \overline{A} \bullet B$$
$$S_1 = A \bullet \overline{B}$$
$$S_2 = \overline{A \oplus B}$$



Circuitos combinatórios

68

Circuitos operativos

- Os circuitos operativos são circuitos combinatórios que permitem realizar operações matemáticas com circuitos digitais;

- Para realizar operações matemáticas em primeiro lugar é necessário transformar os dados em expressões codificadas.

- Estes circuitos comtemplam, para além da(s) saída(s) para o resultado da operação, uma outra saída para o ***transporte*** que poderá resultar da operação efectuada:
  - Numa operação de adição o transporte tem o nome de ***carry***;
  - Numa operação de subtracção o transporte tem o nome de ***Borrow***;

Circuitos combinatórios

69

A tabela de verdade referente à soma lógica de duas palavras de 1 bit cada é a seguinte:

Soma lógica			
Entradas		Saídas	
A	B	S=A+B	C = Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

A tabela de verdade referente à subtracção lógica de duas palavras de 1 bit cada é a seguinte:

- Na subtracção temos de ter em atenção a ordem pela qual as palavras são subtraídas, pois obtemos resultados diferentes, isto é, **A-B ≠ B-A.**

Subtracção lógica			
Entradas		Saídas	
A	B	S=A-B	B = Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Circuitos combinatórios

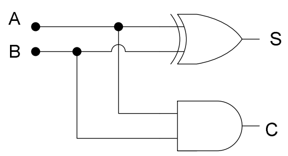
70

Circuitos operativos semi somadores

- É o somador mais simples que existe;
- Soma duas palavras de um bit cada;
- Fornece nas saídas o resultado da soma e o respectivo transporte ou "vai-um" (*carry*).

Entradas		Saídas	
A	B	S=A+B	C = Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$S = A \oplus B$$
$$C = A \bullet B$$



Circuitos combinatórios

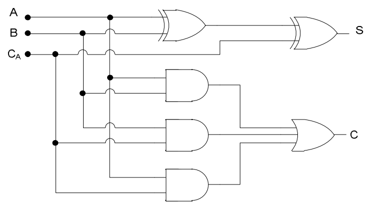
71

Circuitos operativos somadores completos

- O circuito somador completo, é um circuito aritmético que efectua a soma binária dos dois dígitos de entrada com o transporte de entrada procedente do andar anterior;
- Possui, as mesmas saídas que um circuito semi – somador, no entanto tem uma entrada a mais (CA) para poder contemplar o transporte de entrada procedente do andar anterior.

Entradas			Saídas	
A	B	C_A	S= A + B + C_A	C = Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$S=(A\oplus B)\oplus C_A; \quad C=A\bullet C+B\bullet C+A\bullet B$$



Circuitos combinatórios

72

Circuitos operativos subtractores completos

- A estrutura destes circuitos é muito parecida com a dos somadores completos, com a diferença de calcularem a subtracção binária dos dígitos de entrada, e os de transporte;
- Tanto o transporte de saída como o de entrada, recebem o nome de empréstimo (*borrow*).

Entradas			Saídas	
A	B	B_A	S= A - B - B_A	B = Borrow
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

$$S=(A\oplus B)\oplus B_A; \quad B=\left(B_A\bullet\left(\overline{A\oplus B}\right)\right)+\left(\overline{A}\bullet B\right)$$