



FEDERAL UNIVERSITY
OF SANTA CATARINA

EEL5105 – Circuitos e Técnicas Digitais

Aula 1

Prof. Héctor Pettenghi

hector@eel.ufsc.br

<http://hectorpettenghi.paginas.ufsc.br>

Introdução

EEL5105: Circuitos e Técnicas Digitais

Sistemas Digitais

Representando Informações com Bits

Sistema Hexadecimal de Numeração

Sistema Octal de Numeração

Outros Códigos Importantes

2

Nesta primeira aula veremos algumas informações básicas sobre a disciplina, o que são os sistemas digitais, como as informações são representadas com bits, e alguns sistemas de numeração e codificação importantes.

EEL5105: Circuitos e Técnicas Digitais

Sistemas Digitais

Representando Informações com Bits

Sistema Hexadecimal de Numeração

Sistema Octal de Numeração

Sistema Quaternario de Numeração

Outros Códigos Importantes

3

Comecemos com as características básicas.

EEL5105

- Carga horária semanal
 - 2 horas-aula de **teoria**
 - 3 horas-aula de **laboratório**
- É oferecida para os seguintes cursos da **UFSC**:
 - Ciência da Computação
 - Engenharia Elétrica
 - Engenharia Eletrônica
 - Engenharia de Controle e Automação
 - Engenharia de Produção Elétrica

4

EEL5105 conta, semanalmente, com 2 horas-aula de teoria e 3 de laboratório e é oferecida para Ciências da Computação, Engenharia Elétrica, Engenharia Eletrônica, Engenharia de Controle e Automação e Engenharia de Produção Elétrica.

EEL5105

- **Bibliografia**

- **Básica (disponíveis a partir do site da biblioteca da UFSC):**
 - *Sistemas Digitais: Projeto, Otimização e HDLs*
Frank Vahid, 1a Edição, Bookman, 2007
 - <http://www.bu.ufsc.br/framebases.html> opção **Minha Biblioteca**
 - Se, após o login na **Minha Biblioteca**, o livro não aparecer na lista de livros, use o seguinte link:
<http://integrada.minhabiblioteca.com.br/books/9788577802371>
 - *Sistemas Digitais: Princípios e Aplicações*
Ronald J. Tocci, Neal S. Widmer e Gregory L. Moss
10a Edição, Pearson Prentice Hall, 2007
- **Complementar:**
 - *Synthesizable VHDL Design for FPGAs*
Eduardo Bezerra e Djones Lettnin, Springer
 - *Contemporary Logic Design*
Randy H. Katz e Gaetano Borriello, 2a Edição, Prentice Hall, 2004

5

Esta é a bibliografia básica e complementar para esta disciplina.

EEL5105

- **Avaliação**

- **Teoria:** 2 Provas (**P1** e **P2**), com

$$M_{\text{Teoria}} = P1 \times 0,5 + P2 \times 0,5$$

- **Lab:** Projeto (**P**), Minitestes (**M**) e Exercícios (**E**), com

$$M_{\text{Lab}} = P \times 0,3 + M \times 0,3 + E \times 0,4$$

- Nota do aluno = $(M_{\text{Teoria}} + M_{\text{Lab}}) / 2$

- Nota ≥ 6 para aprovação

- Frequência mínima: 75%

- **Não há recuperação!!!!**

6

A nota final é formada pela seguinte regra. Não há recuperação, afinal a maioria da carga-horária da disciplina é de caráter prático.

EEL5105: Circuitos e Técnicas Digitais

Sistemas Digitais

- Representando Informações com Bits
- Sistema Hexadecimal de Numeração
- Sistema Octal de Numeração
- Outros Códigos Importantes

7

Veremos agora o que são os sistemas digitais.

Sistemas Digitais

- Os sistemas digitais estão em praticamente todos os lugares



8

Um sistema digital é um sistema no qual os sinais possuem um conjunto finito de valores que podem assumir (sinal discreto). Em contraposição, em um sistema analógico os sinais possuem infinitos valor possíveis (sinal contínuo). Atualmente, os sistemas digitais estão em praticamente todos os lugares, como podemos ver com os exemplos deste slide.

Sistemas Digitais

- Os sistemas digitais estão em praticamente todos os lugares

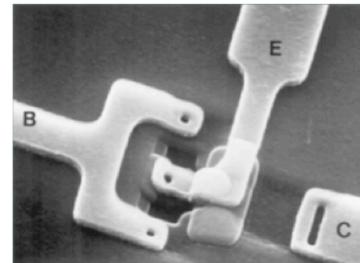


9

E os exemplos deste slide também.

Sistemas Digitais

- Início da revolução digital: invenção do **transistor**
- **Transistor:** “uma chave eletrônica, do tamanho de um vírus, capaz de controlar o fluxo de uma pequena corrente elétrica”
<https://www.youtube.com/watch?v=1qQE5Xwe7fs> 
- Funcionando como chave, o transistor pode então ficar em dois estados:
 - **desligado** ou **ligado**
 - **sim** ou **não**
 - **zero** ou **um**
 - Sistema **binário!**



10

O início da revolução digital se deu com a invenção do transistor, "uma chave eletrônica, do tamanho de um vírus, capaz de controlar o fluxo de uma pequena corrente elétrica". Funcionando como chave, o transistor pode então ficar em dois estados, que podem ser interpretados de diversas formas (desligado ou ligado, sim ou não, zero e um, e entre outros). Assim, ele permite a utilização do sistema binário para a representação da informação em sistemas digitais.

Sistemas Digitais

- **Mas, como é possível representar praticamente qualquer tipo de informação usando apenas dois símbolos (0 e 1)?**

11

Mas como isso é feito?

EEL5105: Circuitos e Técnicas Digitais
Sistemas Digitais

Representando Informações com Bits **O Sistema Binário de Representação**

Sistema Hexadecimal de Numeração
Sistema Octal de Numeração
Sistema Quaternario de Numeração
Outros Códigos Importantes

12

É isso que veremos agora.

Representando Informações com Bits

- Todo tipo de informação pode ser representada por números
 - **Som**: sequência de valores de pressão ao longo do tempo
 - **Imagem**: conjunto de pixels, cada um com um valor de brilho
 - **Video**: sequencia de imagens ao longo do tempo
 - **Distâncias, alturas, pesos, etc**: naturalmente numéricos
 - **etc...**

13

Todo tipo de informação pode ser representada por números, isso vale para o som, imagens, vídeos, distâncias, alturas, pesos e muitas outras.

Representando Informações com Bits

- Todo tipo de informação pode ser representada por números
 - **Som**: sequência de valores de pressão ao longo do tempo
 - **Imagem**: conjunto de pixels, cada um com um valor de brilho
 - **Video**: sequencia de imagens ao longo do tempo
 - **Distâncias, alturas, pesos, etc**: naturalmente numéricos
 - **etc...**
- Se com **dez** símbolos diferentes podemos representar infinitos valores, o mesmo pode ser feito com **dois** símbolos, basta usar uma lógica **posicional** similar à decimal...

14

Normalmente vemos todas essas informações representadas no sistema decimal. Se com dez símbolos diferentes podemos representar infinitos valores, o mesmo pode ser feito com apenas dois símbolos, basta usar uma lógica posicional similar à decimal.

Representando Informações com Bits

- **Sistema Decimal**
 - Base 10
 - 10 símbolos diferentes
 - 0, 1, 2, 3, 4, 5, 6, 7, 8 e 9

Com D dígitos decimais, quantos números diferentes podem ser representados?

$$\begin{aligned}1 + 1 &= 2 \\2 + 3 &= 5 \\1 + 9 &= 10 \\47+1 &= 48 \\99+1 &= 100\end{aligned}$$

15

O sistema decimal trabalha com números na base 10, ou seja, utiliza-se de 10 símbolos diferentes para a representação dos números. Assim, se temos D dígitos decimais, quantos números diferentes podem ser representados?

Representando Informações com Bits

- **Sistema Decimal**

- Base 10
- 10 símbolos diferentes
- 0, 1, 2, 3, 4, 5, 6, 7, 8 e 9

Com **D** dígitos decimais, **10^D** números diferentes podem ser representados.

$$\begin{aligned}1 + 1 &= 2 \\2 + 3 &= 5 \\1 + 9 &= 10 \\47+1 &= 48 \\99+1 &= 100\end{aligned}$$

Exemplo:

Com 3 dígitos decimais, podemos representar 1000 números:
0 a 999.

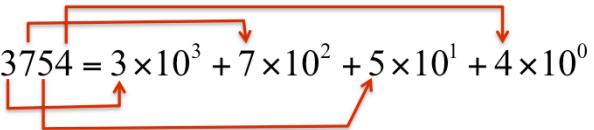
16

A resposta é 10^D (dez elevado a D). Assim, se tomarmos o exemplo deste slide, que leva em consideração 3 dígitos, podemos representar 10^3 (dez elevado a 3)=1000 números (0 a 999).

Representando Informações com Bits

- **Sistema Decimal**

- Posição do dígito tem efeito multiplicador sobre a base:

$$3754 = 3 \times 10^3 + 7 \times 10^2 + 5 \times 10^1 + 4 \times 10^0$$


17

Outra característica do sistema decimal é que a posição do dígito tem efeito multiplicador sobre a base. Veja o exemplo deste slide.

Representando Informações com Bits

- **Sistema Binário**

- Base 2
- 2 símbolos diferentes
- 0 e 1

- **Sistema Decimal**

- Base 10
- 10 símbolos diferentes
- 0, 1, 2, 3, 4, 5, 6, 7, 8 e 9

18

O sistema binário, que trabalha com números na base 2 e conta com 2 símbolos diferentes (0 e 1) funciona de forma análoga.

Representando Informações com Bits

- **Sistema Binário**

- Base 2
- 2 símbolos diferentes
- 0 e 1

$$\begin{aligned}0 + 1 &= 1 \\1 + 1 &= 10 \\10 + 1 &= 11 \\11 + 1 &= 100\end{aligned}$$

19

Veja que quando somamos 1+1 no sistema binário acontece o mesmo que quando somamos, por exemplo, 9+3, em decimal: soma-se 1 à próxima casa, da dezena, e recomeça-se a contagem (a partir de 0) dos dígitos na mesma casa, da unidade.

Veja: Em 9+3, recomeçamos a contar a casa da unidade a partir de 0, o que resulta em 2. Como precisamos recomeçar a contar a casa da unidade, soma-se 1 à casa da dezena, que valia 0 e agora vale 1. O resultado disso tudo é 12. Agora tente reproduzir a mesma lógica para as somas binárias apresentadas neste slide.

Representando Informações com Bits

- **Sistema Binário**

- Base 2
- 2 símbolos diferentes
- 0 e 1

Com **D** dígitos binários,
quantos números
diferentes podem ser
representados?

$$\begin{aligned}0_2 + 1_2 &= 1_2 \\1_2 + 1_2 &= 10_2 \\10_2 + 1_2 &= 11_2 \\11_2 + 1_2 &= 100_2\end{aligned}$$

Base 2

20

Para representar que um número está na base 2, basta subescrever o número 2 após o número.

Agora, com D dígitos binários, quantos números diferentes podem ser representados.

Representando Informações com Bits

- **Sistema Binário**

- Base 2
- 2 símbolos diferentes
- 0 e 1

Com **D** dígitos binários,
 2^D números diferentes
podem ser
representados.

$$\begin{aligned}0_2 + 1_2 &= 1_2 \\1_2 + 1_2 &= 10_2 \\10_2 + 1_2 &= 11_2 \\11_2 + 1_2 &= 100_2\end{aligned}$$

Exemplo:

Com **3** dígitos
binários, podemos
representar **8**
números:
 0_2 a 111_2 .

21

Resposta: 2^D (dois elevado a D) números diferentes. Assim, com 3 dígitos binários, podemos representar 2^3 (dois elevado a 3)=8 números (de 0 a 111).

Representando Informações com Bits

- **Sistema Binário**

- Posição do dígito tem efeito multiplicador sobre a base:

$$100110_2 = 1 \times 10_2^5 + 0 \times 10_2^4 + 0 \times 10_2^3 + 1 \times 10_2^2 + 1 \times 10_2^1 + 0 \times 10_2^0$$

- Convertendo para **decimal**:

$$100110_2 = 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = \boxed{38}$$

22

Da mesma forma que no sistema decimal, a posição do dígito também tem efeito multiplicador sobre a base.

Representando Informações com Bits

- **Sistema Binário**

- Relação entre binários e decimais:

binário	decimal
0 ₂	0
1 ₂	1
10 ₂	2
11 ₂	3
100 ₂	4
101 ₂	5
110 ₂	6
111 ₂	7
1000 ₂	8
1001 ₂	9
1010 ₂	10
1011 ₂	11
1100 ₂	12
⋮	⋮
1111111 ₂	255
⋮	⋮ 23

Esta tabela apresenta uma relação entre números binários e decimais.

Representando Informações com Bits

- Sistema Binário
 - Mas, como converter decimal para binário?
 1. Método de divisão
 2. Método de subtração

24

Mas, como converter um número decimal para um número binário? Existem dois métodos para isso, o da divisão e o da subtração.

Representando Informações com Bits

- **Sistema Binário**

- Mas, como converter **decimal** para **binário**? → **METODO DE DIVISÃO**
 - **Binários** e **decimais** são equivalentes. Se um **decimal** inteiro é dividido por **10**, o resto de tal divisão corresponde ao **dígito mais à direita** do decimal:

$$\begin{array}{r} 493 \quad | \quad 10 \\ - 490 \quad \quad 49 \quad | \quad 10 \\ \hline \quad \quad \quad 9 \quad | \quad 0 \quad | \quad 0 \\ \quad \quad \quad \quad \quad 4 \end{array}$$

A arrow points from the remainder '3' in the first division step to the rightmost digit '0' in the binary result.

25

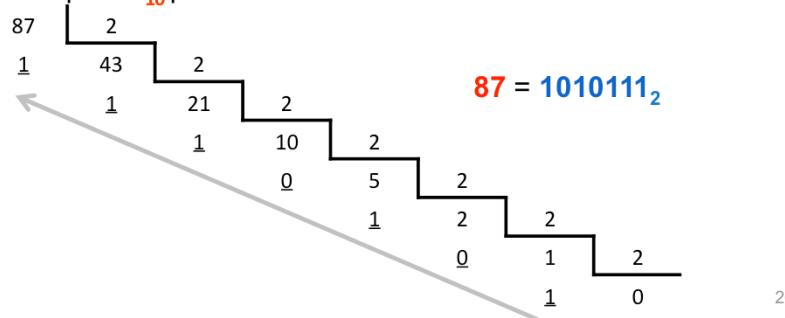
O método da divisão leva em consideração o fato de que se um decimal inteiro é dividido por 10, o resto de tal divisão corresponde ao dígito mais à direita do decimal. Sendo binários e decimais equivalentes, podemos utilizar esta observação para obter números binários a partir de decimais.

Representando Informações com Bits

- **Sistema Binário**

- Mas, como converter **decimal** para **binário**? → **METODO DE DIVISÃO**
 - Então, para converter um número **decimal** para **binário**, basta:
 - 1) Dividir o número sucessivamente por 2
 - 2) Agrupar os restos das divisões de trás para frente

- Exemplo: 87_{10} para binário



26

Assim, basta dividir o número decimal sucessivamente por 2 e agrupar os restos das divisões de trás para frente. Ao final, teremos o número binário.

Representando Informações com Bits

- **Sistema Binário**

- Mas, como converter **decimal** para **binário**? → **METODO DE SUBTRAÇÃO**

- Então, para converter um número **decimal** para **binário**, podemos:

- 1) Subtrair o número sucessivamente pela potencia de 2 imediatamente inferior.

- 2) As potencias de 2 usadas na subtração correspondem com as posições dos uns do código.

- Exemplo: **87₁₀** para binário

$$\begin{array}{r} - 87 \\ - \underline{64 \rightarrow 2^6} \\ - 23 \\ - \underline{16 \rightarrow 2^4} \\ - 7 \\ - \underline{4 \rightarrow 2^2} \\ - 3 \\ - \underline{2 \rightarrow 2^1} \\ - 1 \\ - \underline{1 \rightarrow 2^0} \\ \hline 0 \end{array}$$

87 = 1010111₂

27

Já no método da subtração devemos subtrair o número sucessivamente pela potência de 2 imediatamente inferior, como ilustrado no slide. As potências de 2 usadas na subtração correspondem com as posições dos 1's do código.

Representando Informações com Bits

- Sistema Binário

Exemplo de Conversão Decimal a binário:

$$\begin{array}{r} 206 \\ - 128 \rightarrow 2^7 \\ \hline 78 \\ - 64 \rightarrow 2^6 \\ \hline 14 \\ - 8 \rightarrow 2^3 \\ \hline 6 \\ - 4 \rightarrow 2^2 \\ \hline 2 \\ - 2 \rightarrow 2^1 \\ \hline 0 \end{array}$$

Método de subtração

$$\begin{array}{r} 206 \\ \hline 0 \ 103 \ 2 \\ | \quad | \quad | \\ 1 \ 51 \ 25 \\ | \quad | \quad | \\ 1 \ 12 \ 6 \\ | \quad | \quad | \\ 0 \ 3 \ 1 \\ | \quad | \quad | \\ 1 \ 1 \ 0 \end{array}$$

Método de divisão

28

Aqui temos 206, em decimal, sendo transformado para a sua versão binária pelos dois métodos apresentados.

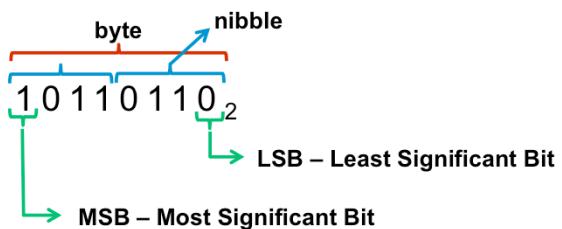
Representando Informações com Bits

- Sistema Binário

- Conceitos:

bit → um dígito binário
nibble → 4 bits
byte → 8 bits

- Exemplo:



29

Os números binários podem ainda ser fatiados em bit (um dígito binário), nibble (4 bits) e byte (8 bits). Há também o MSB (Most Significant Bit – bit mais significativo) e o LSB (Least Significant Bit – bit menos significativo).

EEL5105: Circuitos e Técnicas Digitais
Sistemas Digitais
Representando Informações com Bits

Sistema Hexadecimal de Numeração

Sistema Octal de Numeração
Sistema Quaternario de Numeração
Outros Códigos Importantes

30

A seguir, veremos o sistema hexadecimal de numeração.

Sistema Hexadecimal de Numeração

- A conversão entre **binário** e **decimal** não é "direta". É preciso fazer várias divisões por 2 ou expandir → Solução: sistema **hexadecimal** (Base 16)
- 16 símbolos diferentes
- 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, **A, B, C, D, E, F**

31

A conversão entre binário e decimal não é "direta", afinal é preciso fazer várias divisões por 2 ou expandir. A solução para isso é a representação de informações em hexadecimal.

O sistema hexadecimal conta com 16 símbolos diferentes (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F).

Sistema Hexadecimal de Numeração

- Base 16
- 16 símbolos diferentes
- 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

$$F3_{16} = F \times 10_{16}^1 + 3 \times 10_{16}^0$$

32

Como no sistema binário e decimal, a posição do dígito também tem efeito multiplicador sobre a base.

Sistema Hexadecimal de Numeração

- Base 16
- 16 símbolos diferentes
- 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

$$F3_{16} = F \times 10_{16}^1 + 3 \times 10_{16}^0$$

 p/ decimal

$$F3_{16} = 15 \times 16^1 + 3 \times 16^0 = \boxed{243}$$

33

Convertendo isso para decimal temos.

Sistema Hexadecimal de Numeração

- Como $16 = 2^4$, 1 dígito hexadecimal representa um nibble e 2 dígitos hexadecimais representam um byte.

1 1 1 1 0 0 1 1₂
F₁₆ 3₁₆

- Números hexadecimais são muito usados para representar bytes.
 - Exemplo: representação de cores RGB em HTML e CSS.

34

Como $16=2^4$ (dois elevado a quatro), um dígito hexadecimal representa um nibble e dois dígitos hexadecimais representam um byte. Por essa facilidade na conversão (representada neste slide) é que os hexadecimais são muito usados para representar bytes, o que pode ser visto, por exemplo, na representação de cores RGB em HTML e CSS.

Sistema Hexadecimal de Numeração

- Outro exemplo:

10111101100011110000100₂

35

Veja este outro exemplo de conversão de binário para hexadecimal (e, consequentemente, vice e versa).

Sistema Hexadecimal de Numeração

- Outros exemplos:

$\begin{array}{ccccccc} 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ \hline 5_{16} & E_{16} & C_{16} & 7_{16} & 8_{16} & 4_{16} \end{array} \quad _2$

$5EC784_{16}$

$\begin{array}{ccccccc} 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ \hline 7_{16} & 7_{16} & 0_{16} & 3_{16} & F_{16} & 6_{16} \end{array} \quad _2$

$7703F6_{16}$

36

E este último exemplo também.

Sistema Hexadecimal de Numeração

- Resumo: conversão entre sistemas numeração
 - Decimal para binário:
 - Binario para decimal:
 - Binário para hexa:
 - Hexa para binário:
 - Decimal para hexa:
 - Hexa para decimal:

37

Assim, em resumo...

Sistema Hexadecimal de Numeração

- **Resumo: conversão entre sistemas numeração**
 - **Decimal** para **binário**: método de divisão ou subtração
 - **Binario** para **decimal**: expandir o número e converter
 - **Binário** para **hexa**: agrupa de 4 em 4 e converte
 - **Hexa** para **binário**: cada dígito gera 4 bits
 - **Decimal** para **hexa**: decimal para binário e binário para hexa
 - **Alternativa**: dividir sucessivamente por 16 e agrupar os restos
 - **Hexa** para **decimal**: expandir o número e converter

38

temos estes métodos para a conversão de base.

PROBLEMAS

Problema 1.1. Faça as seguintes conversões:

- a) Converter $378_{(10)}$ para hexadecimal e depois binário.
- b) Converter $0110_{(2)}$ para hexadecimal e decimal.
- c) Converter $010110010100000100001111_{(2)}$ para hexadecimal.

PROBLEMAS

Solução:

a) Obtemos a solução em hexadecimal e a seguir em binário:

$$\begin{array}{r}
 378 \mid 16 \\
 10 \quad 23 \quad 16 \\
 \downarrow \quad \downarrow \quad \downarrow \\
 7 \quad 1 \quad 0
 \end{array}
 \rightarrow 17A_{(16)} = 00010111010_{(2)}$$

Verificamos a solução:

$$\begin{array}{r}
 378 \\
 -256 \rightarrow 2^8 \\
 \hline
 122 \\
 -64 \rightarrow 2^6 \\
 \hline
 58 \\
 -32 \rightarrow 2^5 \\
 \hline
 26 \\
 -16 \rightarrow 2^4 \\
 \hline
 10 \\
 -8 \rightarrow 2^3 \\
 \hline
 2 \\
 -2 \rightarrow 2^1 \\
 \hline
 0
 \end{array}
 \rightarrow 101111010_{(2)}$$

b) $0110_{(2)} = 6_{(16)} = 6_{(10)}$

c)

5	9	4	1	0	F	
010	100	010	000	0000	111	\rightarrow Solução: 59410F ₍₁₆₎

Bit menos significativo

EEL5105: Circuitos e Técnicas Digitais
Sistemas Digitais
Representando Informações com Bits
Sistema Hexadecimal de Numeração

Sistema Octal de Numeração

Sistema Quaternário de Numeração
Outros Códigos Importantes

41

Agora veremos o sistema octal de numeração.

Sistema Octal de Numeração

- Base **8**
- **8** símbolos diferentes
- 0, 1, 2, 3, 4, 5, 6, 7

$$164_8 = 1 \times 10_8^2 + 6 \times 10_8^1 + 4 \times 10_8^0$$

- Convertendo para decimal:

$$164_8 = 1 \times 8^2 + 6 \times 8^1 + 4 \times 8^0 = 116$$

42

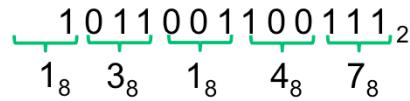
O sistema octal segue os mesmos conceitos dos outros sistemas, mas com 8 símbolos diferentes (0, 1, 2, 3, 4, 5, 6, 7).

Sistema Octal de Numeração

- Sistema Octal

- Como $8 = 2^3$, um grupo de três bits corresponde a apenas um dígito octal.

binário	octal
000_2	0_8
001_2	1_8
010_2	2_8
011_2	3_8
100_2	4_8
101_2	5_8
110_2	6_8
111_2	7_8
1000_2	10_8


101 110 001 110 001 111₂
1₈ 3₈ 1₈ 4₈ 4₈ 7₈

$$1011001100111_2 = 13147_8$$

43

Assim, como $8=2^3$ (2 elevado ao cubo), um grupo de três bits corresponde a apenas um dígito octal. Veja no slide.

PROBLEMAS

Problema 1.2. Pretende-se converter o número $38_{(10)}$ para outras bases.

- Faça a conversão para base 16 directamente.
- Faça a conversão para base 16 através da base 2 (converta primeiro para base 2 e depois para base 16).
- Faça a conversão para base 8 (octal) a partir da base 2.

Solução:

a)

$$\begin{array}{r} 38 \\ \times 16 \\ \hline 6 & 2 \\ 2 & 0 \end{array}$$

$26_{(16)}$

b)

$$\begin{array}{r} 38 \\ -32 \rightarrow 2^5 \\ \hline 6 \\ -4 \rightarrow 2^2 \\ \hline 2 \\ -2 \rightarrow 2^1 \\ \hline 0 \end{array}$$

$100110_{(2)}$

$2 \quad 6$

$00100110_{(2)}$

c)

$$\begin{array}{r} 4 & 6 \\ 100110_{(2)} \end{array}$$

Solução: $26_{(16)}$

44

EEL5105: Circuitos e Técnicas Digitais
Sistemas Digitais
Representando Informações com Bits
Sistema Hexadecimal de Numeração
Sistema Octal de Numeração

Sistema Quaternario de Numeração

Outros Códigos Importantes

45

A seguir, veremos o sistema quaternário de numeração

Sistema Quaternario de Numeração

- Base **4**
- **4** símbolos diferentes
- 0, 1, 2, 3

$$32_4 = 3 \times 10^1 + 2 \times 10^0$$

- Convertendo para decimal:

$$32_4 = 3 \times 4^1 + 2 \times 4^0 = 14$$

46

Este sistema também segue os mesmos conceitos dos demais, mas possui 4 símbolos diferentes (0, 1, 2, 3).

Sistema Quaternario de Numeração

- Sistema Quaternario
- Como $4 = 2^2$, um grupo de dois bits corresponde a apenas um dígito quaternario.

binário	quaternario
00_2	0_4
01_2	1_4
10_2	2_4
11_2	3_4
100_2	10_4

1010010011_2
 $2_4 \quad 2_4 \quad 1_4 \quad 0_4 \quad 3_4$

$$1010010011_2 = 22103_4$$

47

E como $4=2^2$ (dois elevado ao quadrado), um grupo de dois bits corresponde a apenas um dígito quaternário.

PROBLEMAS

Problema 1.3. Faça a conversão de $010010010110001101010111_{(2)}$ para:

- a) Hexadecimal.
- b) Octal.
- c) Quaternário.

Solução:

a)

4 9 6 3 5 7
 $0100\boxed{1001}0110\boxed{0011}0101\boxed{0111}_{(2)}$

b)

2 2 2 6 1 5 2 7
 $0100\boxed{1001}0110\boxed{0011}0101\boxed{0111}_{(2)}$

Solução: $496357_{(16)}$

c)

1 0 2 1 1 2 0 3 1 1 1 3
 $0100\boxed{1001}0110\boxed{0011}0101\boxed{0111}_{(2)}$

Solução: $102112031113_{(8)}$

EEL5105: Circuitos e Técnicas Digitais
Sistemas Digitais
Representando Informações com Bits
Sistema Hexadecimal de Numeração
Sistema Octal de Numeração

Outros Códigos Importantes

49

Por último, veremos alguns outros códigos importantes para esta disciplina.

Outros Códigos Importantes

Código BCD

- BCD – *binary-coded-decimal*
- Cada dígito decimal é codificado com 4 bits

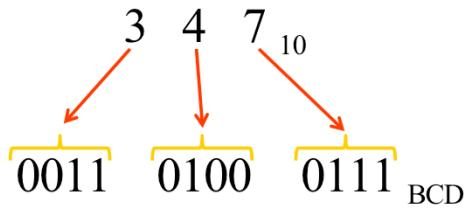
50

O primeiro é o BCD (Binary Coded Decimal), no qual cada dígito decimal é codificado com 4 bits. Originalmente, este código era utilizado em calculadoras.

Outros Códigos Importantes

Código BCD

- BCD – *binary-coded-decimal*
- Cada dígito decimal é codificado com 4 bits
- Exemplo:



51

Assim, este é um exemplo de conversão de decimal para BCD.

Outros Códigos Importantes

Código BCD

- BCD – *binary-coded-decimal*
- Cada dígito decimal é codificado com 4 bits
- Exemplo:

347_{10}

 001101000111_{BCD}

52

Observe que, como cada dígito decimal só vai até 9, cada nibble não pode valer mais do que isso também.

PROBLEMAS

Problema 1.4. Faça as seguintes conversões:

- 3980₍₁₀₎ para BCD e binário.
- 98015₍₁₀₎ para BCD.
- 10000111000001011001_{BCD} para decimal.

Solução:

a)

$$\begin{array}{r}
 3980 \\
 -2048 \rightarrow 2^{11} \\
 \hline
 1932 \\
 -1024 \rightarrow 2^{10} \\
 \hline
 908 \\
 -512 \rightarrow 2^9 \\
 \hline
 396 \\
 -256 \rightarrow 2^8 \\
 \hline
 140 \\
 -128 \rightarrow 2^7 \\
 \hline
 12 \\
 -8 \rightarrow 2^3 \\
 \hline
 4 \\
 -4 \rightarrow 2^2 \\
 \hline
 0
 \end{array}$$

$111110001100_{(2)}$

b)
$$\begin{array}{r}
 9 & 8 & 0 & 1 & 5 \\
 \hline
 10011000000001010 \\
 \hline
 1000011100000101100 \\
 \hline
 8 & 7 & 0 & 5 & 9
 \end{array}$$

53

PROBLEMAS

Problema 1.5. A seguinte sequência de bits pode representar um número BCD?

10001110000110110000001_{BCD}

Solução:

1000 111 0000 1101 1000 0001_(BCD)
8 ? 0 ? 8 1

Não pode ser uma sequencia BCD

Problema 1.6. Quantos bits são necessários para representar os números decimais de 0 a 999 em binário puro e usando o código BCD?

Solução: Em binário puro precisamos de 10 bits por enquanto em BCD de 12 bits.

Outros Códigos Importantes

Código Gray

- Princípio: entre um número e o próximo, apenas 1 bit é modificado.

- 3 bits:

	Decimal	Binário	Gray
0	000	000	
1	001	001	
2	010	011	
3	011	010	
4	100	110	
5	101	111	
6	110	101	
7	111	100	

55

No Código Gray, entre um número e o próximo, apenas 1 bit é modificado. Neste slide temos um exemplo com 3 bits.

Outros Códigos Importantes

Código ASCII

- *American Standard Code for Information Exchange*
- Codificação alfanumérica
- 7 ou 8 bits por símbolo

56

O Código ASCII (American Standard Code for Information Exchange) é utilizado para traduzir dados alfanuméricos para código binário, sendo que cada símbolo tem 7 ou 8 bits.

Outros Códigos Importantes

Código ASCII

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0 000	000	NUL (null)	32	20	040	 	Space	64	40	100	@	 	96	60	140	`	`
1	1 001	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2 002	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3 003	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4 004	004	EOT (end of transmission)	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5 005	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6 006	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7 007	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8 010	010	BS (backspace)	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9 011	011	TAB (horizontal tab)	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A 012	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B 013	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C 014	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D 015	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E 016	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F 017	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10 020	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11 021	021	DCL (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12 022	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13 023	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14 024	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15 025	025	NAK (negative acknowledgement)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16 026	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17 027	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18 030	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19 031	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A 032	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B 033	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[[123	7B	173	{	{
28	1C 034	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D 035	035	GS (group separator)	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E 036	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F 037	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		DEL

57

Ele é composto por uma tabela, apresentada neste slide, onde cada caracter alfanumérico é endereçado a um código binário (aqui representado em decimal, hexadecimal, octal e HTML).

PROBLEMAS

Problema 1.8. Decodifique a seguinte mensagem que está codificada usando o código ASCII:

01010011010101000100111101010000_(ASCII)

Solução: Obtemos a representação em hexadecimal da mensagem:

0101 0011 0101 0100 0100 1111 0101 0000

5 3 5 4 4 F 5 0₍₁₆₎

Olhando a tabela conseguimos obter a mensagem codificada.

53 54 4F 50₍₁₆₎ → STOP

PROBLEMAS

Problema 1.10 (Prova 2019.1). Represente os valores apresentados a seguir nas formas numéricas indicadas:

- a) $10100000111101010110_{(2)}$ em hexadecimal;
- b) $+69_{(10)}$ em octal;
- c) $CADE8_{(16)}$ em quaternário;
- d) $+67_{(10)}$ na representação BCD.

Solução:

a)

A	0	F	5	6
$10100000111101010110_{(2)}$				

\uparrow
Bit menos significativo

\rightarrow Solução: $A0F56_{(16)}$

b)

C	A	D	E	8
$11001010110111101000_{(2)}$				

↓
Passo 2

3	0	2	2	3	1	3	2	2	0
$11001010110111101000_{(2)}$									

\rightarrow Solução: $3022313220_{(4)}$

c)

$\frac{69}{-64} \rightarrow 2^6$	$\frac{5}{-4} \rightarrow 2^2$	$\frac{1}{-1} \rightarrow 2^0$
$1000101_{(2)}$		

\uparrow
 $1000101_{(2)}$

\rightarrow Solução: $105_{(8)}$

d)

$67_{(10)}$	\rightarrow	$0110\boxed{0111}_{BCD}$	\rightarrow	Solução: $01100111_{(BCD)}$
-------------	---------------	--------------------------	---------------	-----------------------------

59



FEDERAL UNIVERSITY
OF SANTA CATARINA

EEL5105 – Circuitos e Técnicas Digitais

Aula 1

Prof. Héctor Pettenghi

hector@eel.ufsc.br

<http://hectorpettenghi.paginas.ufsc.br>

Exercícios

(retirados principalmente de diferentes edições do livro do Tocci)

1) Realize as seguintes conversões:

- (a) $1417_{10} = \text{_____}_2$
- (b) $255_{10} = \text{_____}_2$
- (c) $11010001_2 = \text{_____}_{10}$
- (d) $1110101000100111_2 = \text{_____}_{10}$
- (e) $2497_{10} = \text{_____}_8$
- (f) $511_{10} = \text{_____}_8$
- (g) $235_8 = \text{_____}_{10}$
- (h) $4316_8 = \text{_____}_{10}$
- (i) $7A9_{16} = \text{_____}_{10}$
- (j) $3E1C_{16} = \text{_____}_{10}$
- (k) $1600_{10} = \text{_____}_{16}$
- (l) $38,187_{10} = \text{_____}_{16}$
- (m) $865_{10} = \text{_____} \text{ (BCD)}$
- (n) $100101000111 \text{ (BCD)} = \text{_____}_{10}$
- (o) $465_8 = \text{_____}_{16}$
- (p) $B34_{16} = \text{_____}_8$
- (q) $01110100 \text{ (BCD)} = \text{_____}_2$
- (r) $111010_2 = \text{_____} \text{ (BCD)}$

61

Exercícios

(retirados principalmente de diferentes edições do livro do Tocci)

2) Muitas calculadoras utilizam o código BCD tanto para armazenar valores conforme eles são digitados quanto para apresentar os valores no display.

- a) Se a calculadora é projetada para lidar com números decimais de 8 dígitos, qual o número de bits necessário para o armazenamento de cada número?
- b) Quais bits são armazenados quando o número 4127 é digitado?

3) Um determinado processador usa o código octal para representar os seus endereços de memória de 12 bits.

- a) Quantos dígitos são necessários para armazenar cada endereço?
- b) Qual a faixa de endereços em octal.
- c) Quantas posições de memória estão disponíveis?

Exercícios

(retirados principalmente de diferentes edições do livro do Tocci)

2) Muitas calculadoras utilizam o código BCD tanto para armazenar valores conforme eles são digitados quanto para apresentar os valores no display.

- a) Se a calculadora é projetada para lidar com números decimais de 8 dígitos, qual o número de bits necessário para o armazenamento de cada número? **R: $8 \times 4 = 32$ bits.**
- b) Quais bits são armazenados quando o número 4127 é digitado? **R: $00004127_{10} = 0000\ 0000\ 0000\ 0000\ 0100\ 0001\ 0010\ 0111_{BCD}$.**

3) Um determinado processador usa o código octal para representar os seus endereços de memória de 12 bits.

- a) Quantos dígitos octais são necessários para representar cada endereço? **R: $12 / 3 = 4$ dígitos.**
- b) Qual a faixa de endereços em octal. **R: 0000_8 até 7777_8 .**
- c) Quantas posições de memória estão disponíveis?
R: $8^4 = 2^{12} = 4096$ posições de memória.

Exercícios

(retirados principalmente de diferentes edições do livro do Tocci)

4) Um computador utiliza um número de 20 bits para representar cada uma das suas posições de memória.

- a) Quantos dígitos hexadecimais são necessários para representar um endereço de memória?
- b) Qual a faixa de endereços possíveis?
- c) Qual o número total de posições de memória?

5) Quantos bits são necessários para representar números decimais inteiros entre 0 e 1999 usando a representação binária pura? E usando a representação BCD?

6) Represente o valor decimal 47 em cada uma das seguintes formas:

- | | | |
|-----------------|----------|----------------|
| a) binário puro | b) BCD | c) hexadecimal |
| d) ASCII | e) octal | |

64

Exercícios

(retirados principalmente de diferentes edições do livro do Tocci)

4) Um computador utiliza um número de 20 bits para representar cada uma das suas posições de memória.

- a) Quantos dígitos hexadecimais são necessários para representar um endereço de memória? **R: 20 / 4 = 5 dígitos.**
- b) Qual a faixa de endereços possíveis? **R: 00000₁₆ a FFFF₁₆.**
- c) Qual o número total de posições de memória? **R: 16^5 = 2^20.**

5) Quantos bits são necessários para representar números decimais inteiros entre 0 e 1999 usando a representação binária pura? **R: 11 bits.** E usando a representação BCD? **R: 4x4 = 16 bits.**

6) Represente o valor decimal 47 em cada uma das seguintes formas:

- | | | |
|-----------------|----------|----------------|
| a) binário puro | b) BCD | c) hexadecimal |
| d) ASCII | e) octal | |

65

Exercícios

- Os exercícios da **1^a edição** do livro do **Vahid** indicados abaixo são os recomendados:
 - **1.1, 1.2, 1.6, 1.12-1.23, 1.26**
- **A versão digital da 1^a edição do livro do Vahid está disponível no site da BU**
 - <http://www.bu.ufsc.br/framebases.html> opção **Minha Biblioteca**
 - Se, após o login na **Minha Biblioteca**, o livro não aparecer na lista de livros, use o seguinte link:
<http://integrada.minhabiblioteca.com.br/books/9788577802371>

Exercícios

- Os exercícios da **10ª edição** do livro do **Tocci** indicados abaixo são os recomendados (dê preferência aos exercícios que tem resposta):
 - **2.1 a 2.23;**
 - **2.30 a 2.36;**
 - Muito interessantes: **2.37 e 2.39.**
- **A versão digital da 10ª edição do livro do Tocci está disponível no site da BU**
 - Mais especificamente em:
 - <http://www.bu.ufsc.br/framebases.html>
 - Acessar a **Biblioteca Virtual 3.0** com seu login e senha da BU

67