

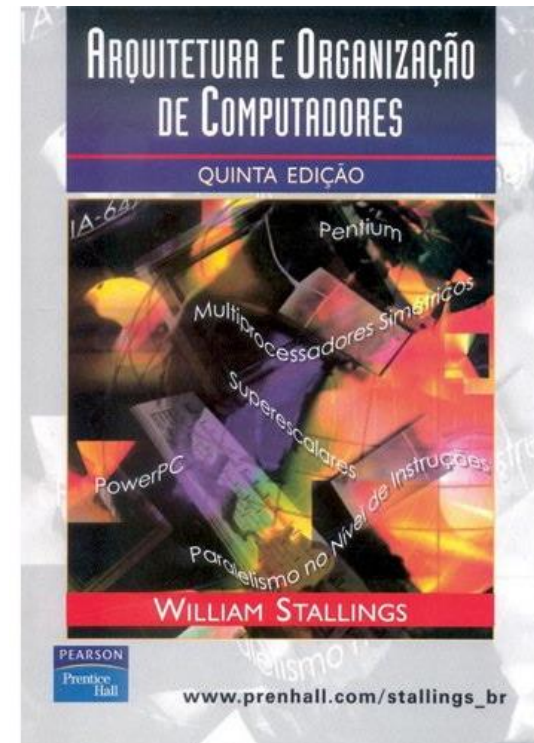
## Aula 02 – Códigos Binários

Prof. João Fernando Mari

*joaof.mari@ufv.br*

# Referências

- STALLINGS, W. **Arquitetura e Organização de Computadores**, 5. Ed., Pearson, 2010.
  - Apêndice A
- ICEA, Sistemas Numéricos e Códigos.
  - Disponível em:
    - <http://www.icea.gov.br/ead/anexo/21401.htm>
  - Acesso em: Mar/2011.



# Roteiro

- Códigos Binários
- Código BCD 8421
  - Conversão binário para BCD
- Código ASCII
  - Conversão em ASCII
- Exercícios

# Códigos Binários

- Conversão de um número decimal em seu equivalente binário
  - Codificação
- Sistema numérico binário como conhecemos
  - Aula anterior!!!
  - Código Binário PURO
    - Diferenciar dos outros códigos binários

# Códigos Binários

- Sistema numérico decimal
  - Conveniente para os seres humanos.
- Sistema numérico binário
  - Conveniente para computadores.
  - (BEM) menos conveniente para os seres humanos.
- Exemplo:
  - $1010011_2$  em decimal ???
    - Processo de conversão simples, porém tedioso → consome muito tempo.
- BCD – Forma especial de código binário MAIS compatível com o sistema decimal.

# Código BCD 8421

- BCD – *Binary Coded Decimal*
  - Binário Codificado em Decimal.
  - Representa os dígitos decimais de 0 a 9 com um código binário de 4 dígitos.
  - Usa o sistema de pesos posicionais 8421 do código binário puro
    - $d_B \times 2^3 + d_B \times 2^2 + d_B \times 2^1 + d_B \times 2^0$
    - $d_B \times 8 + d_B \times 4 + d_B \times 2 + d_B \times 1$
  - Exemplo: Decimal  $\rightarrow$  BCD
    - $834_{10}$  em BCD = 1000 0011 0100
    - 0.764 em BCD = 0.0111 0110 0100
  - Exemplos: BCD  $\rightarrow$  Decimal
    - 0110 0010 1000.1001 0101 0100 = 628.954

- Vantagens BCD
  - Simples manipulação e conversão
  
- Desvantagens
  - Menos eficiente que o código binário puro. Utiliza maior número de bits.
  - Maior complexidade dos circuitos, maior consumo de energia, ...
  - As operações aritméticas consomem mais tempo.

# Código BCD 8421

DECIMAL	BCD 8421	BINÁRIO
0	0000	0000
1	0001	0001
2	0010	0010
3	0011	0011
4	0100	0100
5	0101	0101
6	0110	0110
7	0111	0111
8	1000	1000
9	1001	1001
10	0001 0000	1010
11	0001 0001	1011
12	0001 0010	1100
13	0001 0011	1101
14	0001 0100	1110
15	0001 0101	1111



# Conversão BCD → Binário

- Converte de BCD para Binário puro
  - 1) Converte BCD para decimal
  - 2) Decimal é convertido para binário
  - Exemplo:
    - **1001 0110.0110 0010 0101 = 96,625**

Inteiro	Resto	Posição	Fração	Inteiro	Posição
$96 \div 2 = 48$	<b>0</b>	-> LSB	$0,625 \times 2 = 1,25 = 0,25$	<b>1</b>	<- MSB
$48 \div 2 = 24$	<b>0</b>		$0,250 \times 2 = 0,50 = 0,50$	<b>0</b>	
$24 \div 2 = 12$	<b>0</b>		$0,500 \times 2 = 1,00 = 0$	<b>0</b>	<- LSB
$12 \div 2 = 06$	<b>0</b>				
$06 \div 2 = 03$	<b>0</b>				
$03 \div 2 = 01$	<b>1</b>				
$01 \div 2 = 00$	<b>1</b>	<- MSB			
<b><math>96_{10} = 1100000_2</math></b>			<b><math>0,625_{10} = 0.101_2</math></b>		

$$\mathbf{96,625_{10} = 96_{10} + 0,625_{10} = 1100000_2 + 0.101_2 = 1100000.101_2}$$

# Código ASCII

- "American Standart Code for Information Interchange" – **ASCII**
  - Forma especial de código binário.
  - Largamente utilizado.
  - 7 bits pode-se representar um total de  $2^7 = 128$  caracteres diferentes.
    - Números decimais de **0 até 9**
    - **Letras maiúsculas e minúsculas do alfabeto**
    - Outros **caracteres especiais** usados para pontuação e controle de dados.

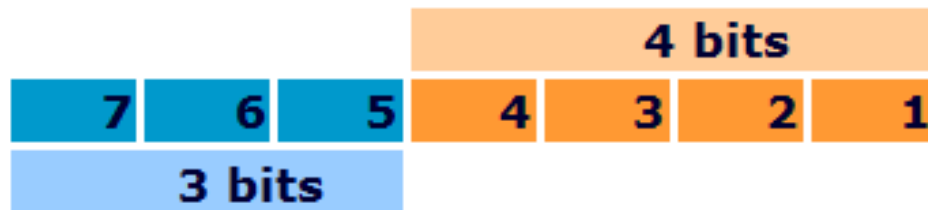
# Tabela ASCII completo ou ASCII estendido

<b>NULL</b>	Null	<b>DLE</b>	Data Link Escape
<b>SOH</b>	Start of Heading	<b>DC1</b>	Device Control 1
<b>STX</b>	Start of Text	<b>DC2</b>	Device Control 2
<b>ETX</b>	End of Text	<b>DC3</b>	Device Control 3
<b>EOT</b>	End of Transmission	<b>DC4</b>	Device Control 4
<b>ENQ</b>	Enquiry	<b>NAK</b>	Negative Acknowledge
<b>ACK</b>	Acknowledge	<b>SYN</b>	Synchronous Idle
<b>BEL</b>	Bell (audible signal)	<b>ETB</b>	End Transmission Block
<b>BS</b>	Backspace	<b>CAN</b>	Cancel
<b>HT</b>	Horizontal Tabulação (punched card skip)	<b>EM</b>	End of Medium
		<b>SUB</b>	Substitute
<b>LF</b>	Line Feed	<b>ESC</b>	Escape
<b>VT</b>	Vertical Tabulation	<b>FS</b>	File Separator
<b>FF</b>	Form Feed	<b>GS</b>	Group Separato
<b>CR</b>	Carriage Return	<b>RS</b>	Record Separator
<b>SO</b>	Shift Out	<b>US</b>	Unit Separator
<b>SI</b>	Shift In	<b>DEL</b>	Delete
<b>SP</b>	Space (blank)		

	coluna								
	bits	0	1	2	3	4	5	6	7
linha	7654321	000	001	010	011	100	101	110	111
0	0000	NUL	DLE	SP	0	@	P	`	p
1	0001	SOH	DC1	!	1	A	Q	a	q
2	0010	STX	DC2	"	2	B	R	b	r
3	0011	ETX	DC3	#	3	C	S	c	s
4	0100	EOT	DC4	\$	4	D	T	d	t
5	0101	ENQ	NAK	%	5	E	U	e	u
6	0110	ACK	SYN	&	6	F	V	f	v
7	0111	BEL	ETB	'	7	G	W	g	w
8	1000	BS	CAN	(	8	H	X	h	x
9	1001	HT	EM	)	9	I	Y	i	y
10	1010	LF	SUB	*	:	J	Z	j	z
11	1011	VT	ESC	+	;	K	[	k	{
12	1100	FF	FS	,	<	L	\	l	
13	1101	CR	GS	-	=	M	]	m	}
14	1110	SO	RS	.	>	N	^	n	~
15	1111	SI	US	/	?	O	_	o	DEL

# Conversão em ASCII

- Composto por 2 grupos:
  - Um de 4 bits e outro de 3 bits.
- O grupo de 4 bits está a direita e o bit 1 é o LSB.
  - LSB: Bit Menos Significativo. MSB: Bit Mais Significativo



- Exemplo: Código ASCII para a letra **L** é **1001100**.
  - Localizado na **coluna 4, linha 12**.
  - O grupo de 3 bits é 100 e o grupo de 4 bits é 1100.
    - Código ASCII: 100 1100.

# Conversão em ASCII

- No código ASCII de 7 bits,
- O oitavo bit é geralmente usado como um bit de paridade.
  - Para determinar se o dado (caractere) foi transmitido corretamente.
  - Determinado pelo tipo de paridade desejado.
    - Paridade par → a soma de todos os 1's, incluindo o bit de paridade, é um número par.
  - EXEMPLO:
    - Caractere G – código ASCII é 1000111
      - 4 bits UM – O bit de paridade é 0 → **0**1000111

