
GFM015 – Introdução à Computação

Noções básicas sobre os componentes de
microcomputadores (cont...) -
Terminologia: Sw, Hw, Programas, Bit, Byte
e Códigos

Ilmério Reis da Silva

ilmerio@facom.ufu.br

www.facom.ufu.br/~ilmerio/ic

UFU/FACOM

Programa

1. Noções básicas sobre os componentes de micro computadores

1.1 Unidades componentes de micro computadores (Placa mãe, Dispositivos de Armazenamentos, Dispositivos de Entrada e Dispositivos de Saida);

1.2 Terminologia (hardware, software, programa, bit, byte, códigos: Binário e ASCII);

1.3 Sistemas operacionais: tipos e características;

2. Uso de aplicativos

3. Algoritmos

4. Fundamentos de programação

5. Estrutura de Dados

6. Modularização de programas

Terminologia – Hw, Sw e Programas

- Hardware e Dispositivos:
 - processador, memória, controladores;
 - monitor, teclado, unidades de disco, impressora, etc.
- Software e Programas:
 - Sistema Operacional – software básico do computador responsável por gerenciamento de memória, controle de dispositivos, etc. Exemplo: Linux, Windows, MacOS, SunOS, etc.
 - Software embarcado: programas que funcionam em máquinas específicas. Ex: semáforos, eletrodomésticos
 - Aplicativos: programas com foco no usuário
 - Desktop: Editor de texto, Planilha, Apresentação
 - Serviços Web: Máquina de Busca, Wikipédia, Sítios de Automação Bancária, etc..

Terminologia – Programas de Aplicação



Terminologia – Linguagem de Máquina

- O programas são executados em Linguagem de Máquina, que é um conjunto de instruções com significado para uma determinada arquitetura de computador.

- Por exemplo, algumas instruções na arquitetura 32 bits:

- 1 0 0 1 0 1 0 1 1 0 0 1 0 1 0 1 1 0 0 1 0 1 0 1 1 0 0 1 0 1 0 1
- 1 0 0 1 0 1 0 1 0 1 1 0 1 1 1 1 0 0 0 1 0 0 0 1 0 0 1 1 0 0 1 1
- 1 0 0 1 0 1 0 1 0 1 1 0 1 1 1 1 1 0 0 1 0 1 0 1 1 0 0 1 0 1 0 1
- 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 1 0 1 0 1 0 1 0 0 1 0 0 1 1 0 0

Terminologia – Linguagem Assembly

- Linguagem de baixo nível (Assembly): representação da linguagem de máquina por meio de mneumônicos.

```
LOOP    MOV A , 3
        INC A
        JGE 12 , SEG01
        JMP LOOP
SEG01   ADD B, A
```

* Montador: traduz linguagem assembly para linguagem de máquina

Terminologia – Linguagens de alto nível

- Linguagem de alto nível: é uma linguagem próxima do nível de abstração do ser humano, por exemplo:
 - C,
 - C++,
 - Java,
 - Python,
 - Delphi,
 - Visual Basic

Terminologia – Compilador

Exemplo de código em linguagem de alto nível (Ling. C):

```
#include <stdio.h>
mail() {
printf(“Hello, World!\n”);
}
```

Compilador – traduz a linguagem de alto nível em linguagem de máquina, ou seja, em código executável.

Terminologia – Interpretador

Um interpretador lê uma instrução em linguagem de alto nível, vinda de um programa ou interface interativa, gera o código executável relativo à instrução, executa a instrução e repete o processo enquanto tiver novas instruções.

- Exemplo de linguagens interpretadas:
 - Shell script
 - Perl

Terminologia – Programação

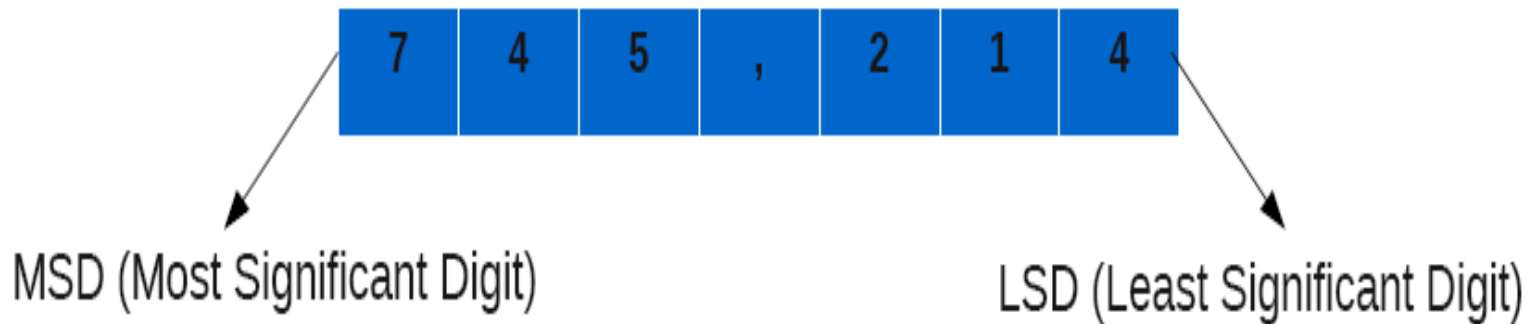
- Programação de Computadores é o processo de escrita, teste e manutenção de um programa de computador
- A programação pode ser feita em linguagem de alto nível, linguagem assembly ou mesmo linguagem de máquina
- Utilizada para solucionar problemas em diferentes áreas, por exemplo:
 - Simulação de experimentos,
 - Cálculos,
 - Análise de imagens,
 - Relatórios comerciais,
 - etc.

Terminologia – Sistemas Numéricos

- Sistema Decimal - composto por 10 numerais ou símbolos
(0, 1, 2, 3, 4, 5, 6, 7, 8, 9)
- Sistema Binário - composto por 2 numerais ou símbolos
(0, 1) – cada dígito é denominado “bit” - “binary digit”
- Sistema Octal - composto por 8 numerais ou símbolos
(0, 1, 2, 3, 4, 5, 6, 7)
- Sistema Hexadecimal - composto por 16 símbolos
(0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F)

Terminologia – Valor posicional base decimal

Cada número é uma soma de produtos do valor de cada dígito pelo seu valor posicional



•Logo:

$$\begin{aligned}(745,214)_{10} &= 7 \times 10^2 + 4 \times 10^1 + 5 \times 10^0 + 2 \times 10^{-1} + 1 \times 10^{-2} + 4 \times 10^{-3} \\ &= 700 + 40 + 5 + 0,2 + 0,01 + 0,004\end{aligned}$$

Terminologia – Valor posicional binário, etc

Conversão para base decimal:

- Sistema Binário:

$$\begin{aligned}(11011,011)_2 &= 2^4 + 2^3 + (0 \times 2^2) + 2^1 + 2^0 + (0 \times 2^{-1}) + 2^{-2} + 2^{-3} \\ &= (27,375)_{10}\end{aligned}$$

- Sistema Octal

$$(764)_8 = (7 \times 8^2) + (6 \times 8^1) + (4 \times 8^0) = (500)_{10}$$

- Sistema Hexadecimal:

$$\begin{aligned}(2AF3)_{16} &= (2 \times 16^3) + (10 \times 16^2) + (15 \times 16^1) + (3 \times 16^0) \\ &= (10.995)_{10}\end{aligned}$$

Terminologia – Conversão dec. para binário

Conversão de decimal para binário

$$(27)_{10}$$

$$27 / 2 = 13 \text{ resto } 1$$

$$13 / 2 = 6 \text{ resto } 1$$

$$6 / 2 = 3 \text{ resto } 0$$

$$3 / 2 = 1 \text{ resto } 1$$

$$1 / 2 = 0 \text{ resto } 1$$

$$(27)_{10} = (11101)_2$$

Terminologia – Conversão dec. p/ binário (2)

Conversão de decimal para binário (após a vírgula)

$$(27,375)_{10} = (27)_{10} + (0,375)_{10}$$

$$(0,375)_{10}$$

$$0,375 \times 2 = 0,750 \text{ pegue o } 0$$

$$0,750 \times 2 = 1,500 \text{ pegue o } 1 \text{ e zere a unidade}$$

$$0,500 \times 2 = 1,000 \text{ pegue o } 1 \text{ e zere a unidade}$$

$$0 = \text{pare.}$$

$$(0,375)_{10} = (0,011)_2$$

$$\text{Como: } (27)_{10} = (11101)_2$$

$$\text{Logo: } (27,375)_{10} = (11101,011)_2$$

Terminologia – Conversões da base decimal

Conversão de decimal para outras bases seguem procedimentos análogos, por exemplo:

$$(8734)_{10} = (?)_{16}$$

$$8734 / 16 = 545 \text{ resto } 14 = \text{E}$$

$$545 / 16 = 34 \text{ resto } 1$$

$$34 / 16 = 2 \text{ resto } 2$$

$$2 / 16 = 0 \text{ resto } 2$$

$$(8734)_{10} = (221\text{E})_{16}$$

Terminologia – Outras Conversões de base

Hexadecimal para Binário:

- Cada dígito hexadecimal representa quatro bits

$$(2AF3)_{16} = (0010\ 1010\ 1111\ 0011)_2 = (1001011110011)_2$$

Binário para Hexadecimal:

- Processo inverso, preenchendo com zeros à esquerda se número de bits não for múltiplo de 4

$$(10101011110011)_2 = (0010\ 1010\ 1111\ 0011)_2 = (2AF3)_{16}$$

Terminologia – Tabela de Códigos - ASCII

ASCII – Padrão desde 1963, modificado em 1986, usa 128 caracteres. Código mais utilizado na Web até 2007

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

Terminologia – Tabela de Códigos – ISO Latin 1 ou ISO 8859-1

Cada caracter é codificado por um byte, extendendo a tabela ASCII para incluir caracteres de línguas europeias, por exemplo:

Char	Code	Name	Description
À	192	Agrave	A grave
Á	193	Aacute	A acute
Â	194	Acirc	A circumflex
Ã	195	Atilde	A tilde
Ä	196	Auml	A umlaut
Å	197	Aring	A ring
Æ	198	AElig	AE ligature
Ç	199	Ccedil	C cedilla
È	200	Egrave	E grave
É	201	Eacute	E acute
Ê	202	Ecirc	E circumflex
Ë	203	Euml	E umlaut
Ì	204	Igrave	I grave
Í	205	Iacute	I acute
Î	206	Icirc	I circumflex
Ï	207	Iuml	I umlaut

Char	Code	Name	Description
Ð	208	ETH	ETH
Ñ	209	Ntilde	N tilde
Ò	210	Ograve	O grave
Ó	211	Oacute	O acute
Ô	212	Ocirc	O circumflex
Õ	213	Otilde	O tilde
Ö	214	Ouml	O umlaut
×	215	times	Multiplication sign
Ø	216	Oslash	O slash
Ù	217	Ugrave	U grave
Ú	218	Uacute	U acute
Û	219	Ucirc	U circumflex
Ü	220	Uuml	U umlaut
Ý	221	Yacute	Y acute
Þ	222	THORN	THORN
ß	223	szzlig	sharp s

Terminologia – Tabela de Códigos – ISO Latin 1 ou ISO 8859-1 (cont.)

Char	Code	Name	Description
à	224	agrave	a grave
á	225	aacute	a acute
â	226	acirc	a circumflex
ã	227	atilde	a tilde
ä	228	auml	a umlaut
å	229	aring	a ring
æ	230	aelig	ae ligature
ç	231	ccedil	c cedilla
è	232	egrave	e grave
é	233	eacute	e acute
ê	234	ecirc	e circumflex
ë	235	euml	e umlaut
ì	236	igrave	i grave
í	237	iacute	i acute
î	238	icirc	i circumflex
ï	239	iuml	i umlaut

Char	Code	Name	Description
ð	240	eth	eth
ñ	241	ntilde	n tilde
ò	242	ograve	o grave
ó	243	oacute	o acute
ô	244	ocirc	o circumflex
õ	245	otilde	o tilde
ö	246	ouml	o umlaut
÷	247	divide	division sign
ø	248	oslash	o slash
ù	249	ugrave	u grave
ú	250	uacute	u acute
û	251	ucirc	u circumflex
ü	252	uuml	u umlaut
ý	253	yacute	y acute
þ	254	thorn	thorn
ÿ	255	yuml	y umlaut

Terminologia – Tabela de Códigos - UTF-8

UTF-8 – Unicode Transformation Format-8

- É um padrão para transferência de textos em Unicode, com milhares de códigos
- Usa de um a quatro bytes por código
- Seus primeiros caracteres são compatíveis com o ASCII
- É o mais utilizado na Web (cerca de 50% das páginas da Web estão em UTF-8)
- Usado na maioria das distribuições Linux
- Windows XP, Windows Vista e Windows 7 usam, como representação interna, o UTF-16 com dois códigos de 16 bits, podendo representar até 1.112.064 símbolos.

Terminologia – Conversões de Base - Exercícios

Converta os seguintes números para as bases decimal, binário e hexadecimal

$$(54DF)_{16}$$

$$(554)_{10}$$

$$(1001011110110101)_2$$

$$(679)_{10}$$

$$(110011011010)_2$$

$$(ABCD)_{16}$$

Bibliografia

Disponível em:

http://www.facom.ufu.br/~ilmerio/ic/ic_s1b_bitByteCodigos.pdf

Material de apoio em:

http://www.facom.ufu.br/~ilmerio/ic/ic_introducao.pdf

FIM – Componentes