



Arquitetura de Computadores

Prof. Dr. Joseffe Barroso de Oliveira



Introdução

Na natureza, todo tipo de informação pode assumir qualquer valor compreendido em um intervalo de $-\infty$ a $+\infty$. Você consegue distinguir, por exemplo, uma cor vermelha que esteja um pouco mais clara de outro tom de vermelho. Esse tipo de informação é informação analógica. Computadores lidam apenas com informações digitais, não abrindo espaço para dados estruturados de forma analógica, que podem assumir valores indefinidos.

Sistemas de Numeração

O sistema de numeração decimal está tão presente no nosso cotidiano que nem notamos sua presença. O kg (quilograma), o m (metro), o Km/h (quilômetro por hora), tudo isso é medido numa forma que compreendemos no modelo decimal. Entretanto, este não é o modelo utilizado pelos computadores, pois as posições de memória basicamente conseguem representar apenas 0 ou 1 (como se estivesse ligado ou desligado). Por isso, existem outras notações, como a binária e a hexadecimal.

Essas notações são obstáculos para os iniciantes. Tornam-se, entretanto, simples depois de recordarmos o sistema de base decimal.



Decimal

Quando falamos do número 123, imaginamos certo número de itens que esse número representa e esquecemos o seu significado matemático. Na realidade 123 representa:

$$(1 \times 10^2) + (2 \times 10^1) + (3 \times 10^0), \text{ ou seja: } 100 + 20 + 3 = 123$$

Observe que cada algarismo é multiplicado por uma potência de 10. Os expoentes de 10 são numerados da direita para a esquerda começando com 0 (zero).



Binário

O sistema binário funciona exatamente da mesma forma (nossa cabeça é que insiste em dar nó). O sistema decimal possui dez dígitos (de 0 a 9); o binário possui apenas dois (0 e 1). Fazemos potência de 10 para calcular o número no sistema decimal, então faremos potência de 2 para o sistema binário.

Por exemplo: o valor binário 11001010 representa o valor decimal 202, assim:

$$(1 \times 2^7) + (1 \times 2^6) + (0 \times 2^5) + (0 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0) + (0 \times 2^0), \text{ ou seja: } 128 + 64 + 0 + 0 + 8 + 0 + 2 + 0 = 202$$



Hexadecimal

O sistema hexadecimal representa os números em base 16. É usado na informática, pois os computadores costumam utilizar o byte como unidade básica da memória e com um byte podemos representar 256 valores possíveis, o que abrange todo alfabeto (maiúsculas e minúsculas), os números e vários caracteres especiais.

Como no sistema decimal dispomos de apenas dez dígitos, devemos incluir seis letras para representar o hexadecimal. O conjunto de algarismos hexadecimais fica, portanto, assim: Cada algarismo é multiplicado por uma potência de 16. Os expoentes de 16 são numerados da direita para a esquerda começando com 0 (zero).

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Nesse sistema, o **A** vale **10**, o **B** vale **11**, etc, até o **F**, que vale **15**.



Conversão de Decimal para Binário

Dividimos o número decimal seguidamente por 2 e utilizamos o resto de cada divisão para compor o número binário.

Observe o exemplo a seguir. Converter 45 (decimal) para binário:

$45 / 2 = 22$	sobra 1
$22 / 2 = 11$	sobra 0
$11 / 2 = 5$	sobra 1
$5 / 2 = 2$	sobra 1
$2 / 2 = 1$	sobra 0
$1 / 2 = 0$	sobra 1

Como não dá mais para dividir, paramos. Pegamos todos os restos olhando de baixo para cima. Concluimos que o número 45 em decimal equivale 101101 em binário.



Conversão de Decimal para Hexadecimal

Por um processo parecido com o que já vimos, dividimos o número decimal seguidamente por 16 e utilizamos o resto de cada divisão para compor o hexadecimal. A única diferença é que o divisor é o 16. Veja o exemplo a seguir:

Converter 970 (decimal) para hexadecimal:

$970 / 16 = 60,$	resto 10 = A
$60 / 16 = 3,$	resto 12 = C
$3 / 16 = 0,$	resto 3 = 3

Temos então que o número 970 (decimal) representa 3CA em hexadecimal (novamente pegamos os restos das divisões, de baixo para cima).



Conversão de Binário para Decimal

Você pode montar uma tabela com as seguintes linhas:

na primeira: escreva o número binário;

na segunda: escreva os expoentes de 2, da direita para a esquerda, começando com 0;

na terceira: calcule as potências de 2;

na quarta: multiplique a primeira linha pela terceira.

Some a quarta linha e terá o número binário convertido para decimal. Veja o exemplo de conversão do número binário 101101 observando a tabela abaixo:

número em binário	1	0	1	1	0	1
expoentes	25	24	23	22	21	20
potências	32	16	8	4	2	1
primeira x terceira	32	0	8	4	0	1



Conversão de Hexadecimal para Decimal

Você pode usar a mesma tabelinha, trocando 2 por 16. Vamos converter o hexadecimal 3F8 para decimal (tabela) a seguir:

número em hexadecimal	3	F	8
expoentes	16^2	16^1	16^0
potências	256	16	1
primeira x terceira	3×256	15×16	8



Tabela de conversão Decimal x Binário x Hexadecimal

Decimal	Binário	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F



Unidades de medida na computação

Unidade	Símbolo	Número de bytes
kilobyte	kB	$2^{10} = 1024$ bytes
megabyte	MB	$2^{20} = 1,048,576$ bytes
gigabyte	GB	$2^{30} = 1,073,741,824$ bytes
terabyte	TB	$2^{40} = 1,099,511,627,776$ bytes
petabyte	PB	$2^{50} = 1,125,899,906,842,624$ bytes
exabyte	EB	$2^{60} = 1,152,921,504,606,846,976$ bytes
zettabyte	ZB	$2^{70} = 1,180,591,620,717,411,303,424$ bytes
yottabyte	YB	$2^{80} = 1,208,925,819,614,629,174,706,176$ bytes



Códigos de representação - Tabela ASCII

Dec	Hex	Oct	Chr	Dec	Hex	Oct	HTML	Chr	Dec	Hex	Oct	HTML	Chr	Dec	Hex	Oct	HTML	Chr
0	0	000	NULL	32	20	040	 	Space	64	40	100	@	@	96	60	140	`	`
1	1	001	Start of Header	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	Start of Text	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	End of Text	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	End of Transmission	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5	005	Enquiry	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	Acknowledgment	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	Bell	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	Backspace	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9	011	Horizontal Tab	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A	012	Line feed	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	Vertical Tab	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	Form feed	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	Carriage return	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	Shift Out	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	Shift In	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	Data Link Escape	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	Device Control 1	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	Device Control 2	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	Device Control 3	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	Device Control 4	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	Negative Ack.	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	Synchronous idle	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	End of Trans. Block	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	Cancel	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	End of Medium	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	Substitute	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	Escape	59	3B	073	;	;	91	5B	133	[[123	7B	173	{	{
28	1C	034	File Separator	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	Group Separator	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	Record Separator	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	Unit Separator	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		Del

asciicharstable.com