

# **ARQUITETURA E ORGANIZAÇÃO DE COMPUTADORES**

## **SISTEMAS DE NUMERAÇÃO: REPRESENTAÇÃO EM PONTO FLUTUANTE**

Prof. Dr. Daniel Caetano

2011 - 2

# Visão Geral

1

- O que é Ponto Flutuante

2

- Ponto Flutuante em Binário

3

- Codificação IEEE 754/2008

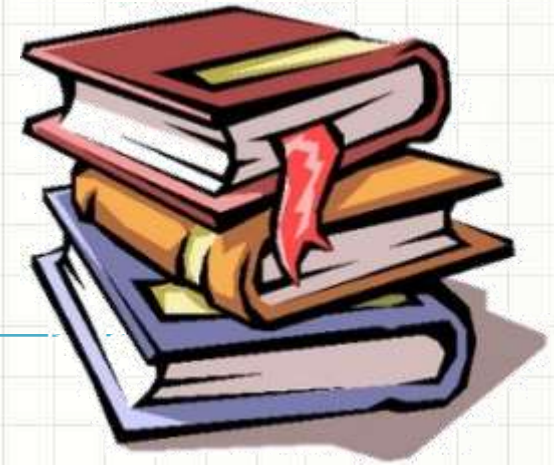
4

- Representação do Zero

5

- Representação de Caracteres

# Material de Estudo



---

## Material

## Acesso ao Material

Notas de Aula

<http://www.caetano.eng.br/aulas/aoc/>  
(Aula 5)

Apresentação

<http://www.caetano.eng.br/aulas/aoc/>  
(Aula 5)

Material Didático

-

Arquitetura e  
Organização dos  
Computadores

Biblioteca Virtual, páginas 289 a 292.

---

# Lembretes

- **Lista de Exercícios 1!**





# O QUE É PONTO FLUTUANTE?

# Notação em Ponto Flutuante

- Como guardar esse número?
- 653400000000000000000000000000000000
- Pode ser reescrito assim:

$$6,534 * 10^{+31}$$

- O que significa  $* 10^{+31}$  ?
- Significa que tenho que mudar a vírgula de lugar para obter o número real
- No caso, preciso andar 31 casas **à direita**



# Notação em Ponto Flutuante

- Qual o nome dessa representação?
- **$6,534 * 10^{+31}$**
- **$6,534 * 10^{-31}$**
- Notação científica ou...
- Notação em Ponto Flutuante
- Por que “ponto flutuante”?
- Por que a posição da vírgula muda (flutua) de acordo com o expoente da potência de 10



# Notação em Ponto Flutuante

- É comum especificar esses números assim:
- $6,534 * 10^{+31} \rightarrow 6,534E31$
- $6,534 * 10^{-31} \rightarrow 6,534E-31$
- Repare que essa representação tem 3 partes

6,534E-31

- Quais são os nomes destas partes?

# Notação em Ponto Flutuante

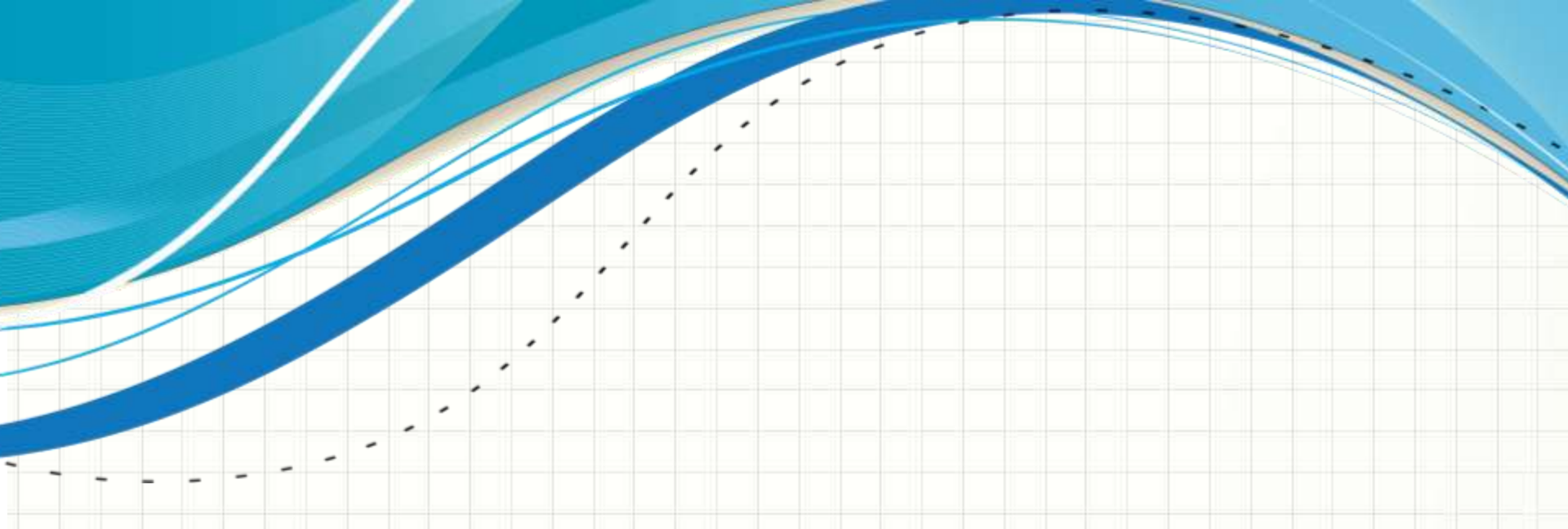
6,534E-31

- Característica
- Mantissa
- Expoente
- Qualquer número pode ser escrito assim?

# Notação em Ponto Flutuante

- Vejamos alguns exemplos

Tradicional	Científica	Característica	Mantissa	Expoente
0,234	2,34E-1	2	34	-1
0,054334	5,4334E-2	5	4334	-2
1	1,0E0	1	0	0
10	1,0E1	1	0	1
100	1,0E2	1	0	2
1000	1,0E3	1	0	3
125	1,25E2	1	25	2
...	1,25E56	1	25	56
...	1,25E-56	1	25	-56



# **PONTO FLUTUANTE EM BINÁRIO**

# Notação em Ponto Flutuante

- Será que isso funciona com binários?
- 101100000000000000000000000000000000b
- Pode ser reescrito assim:

$$1,011b * 2^{+31}$$

- O que significa  $* 2^{+31}$  ?
- Significa que tenho que mudar a vírgula de lugar para obter o número real
- No caso, andar 31 casas binárias à **direita**



# Notação em Ponto Flutuante

- $1,011b * 2^{+31}$
- $1,011b * 2^{-31}$
- Os nomes permanecem os mesmos
- Característica
- Mantissa
- Expoente

# Notação em Ponto Flutuante

- Vejamos alguns exemplos

Tradicional	Científica	Característica	Mantissa	Expoente
100b	$1,00b * 2^2$	1b	00b	2
101b	$1,01 * 2^2$	1b	01b	2
11,101b	$1,1101b * 2^1$	1b	1101b	1
1b	$1,0b * 2^0$	1b	0b	0
0,1001b	$1,001b * 2^{-1}$	1b	001b	-1
...	$1,0101b * 2^{56}$	1b	0101b	56
...	$1,0101b * 2^{-56}$	1b	0101b	-56

- O que tem de estranho aí?



# Notação em Ponto Flutuante

- Vejamos alguns exemplos

Tradicional			Expoente
100b	1		
101b			
11,101b			
1b			
0,1001b			-1
...	1,010	1b	56
...	1,01	01b	56

**Se é sempre 1b,  
não precisamos  
guardar!**

- O que tem de estranho aí?

# Notação em Ponto Flutuante

- Assumimos que a característica é sempre 1b

Tradicional	Científica	Mantissa	Expoente
100b	$1,00b * 2^2$	00b	2
101b	$1,01 * 2^2$	01b	2
11,101b	$1,1101b * 2^1$	1101b	1
1b	$1,0b * 2^0$	0b	0
0,1001b	$1,001b * 2^{-1}$	001b	-1
...	$1,0101b * 2^{56}$	0101b	56
...	$1,0101b * 2^{-56}$	0101b	-56

# Representação em Ponto Flutuante

- Como representar estes números na memória?
- Em ponto fixo, reservamos um bit para o sinal... E os demais bits para a magnitude

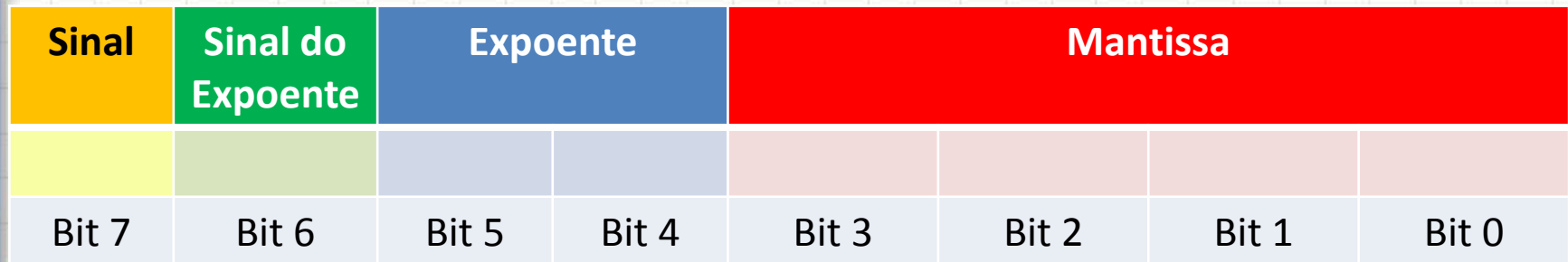
Bit	7	6	5	4	3	2	1	0
Dígito Binário	0	1	1	0	0	0	0	1

+ ou -  
Bit 7: Sinal

0 a 127  
Bits 0 a 6: Magnitude

# Representação em Ponto Flutuante

- O princípio é o mesmo, mas agora vamos reservar bits para várias outras coisas



- Observe que ficaram poucos bits para cada indicação
- Expoente de -3 a +3
- Mantissa de 0000b a 1111b
- Característica -1b ou +1b

# Representação em Ponto Flutuante

- Vamos representar o número 2,25 nessa notação

- Primeiro vamos converter para binário

- Parte inteira:

$$2 = 10b$$

- Parte fracionária:

$$0,25 = 0,01b$$

- Logo...  $2,25 = 10,01b$

- Em notação científica:

$$10,01b = 1,001b * 2^1$$

# Representação em Ponto Flutuante

- Bem, então  $2,25 = 1,001b * 2^1$
- Vamos dividir este número em partes
  - Sinal:** 0 (positivo)                      **Característica:** 1b
  - Mantissa:** 001b
  - Sinal Expoente:** 0 (positivo)   **Expoente:** 1b
- Agora... Na memória:

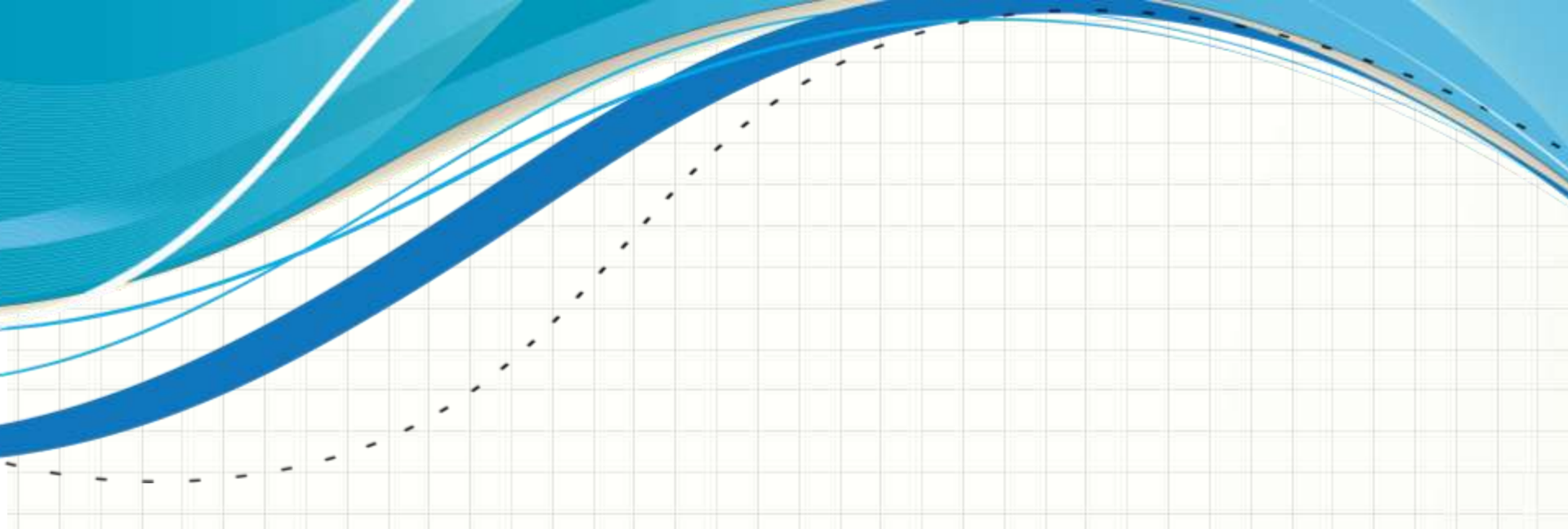
Sinal	Sinal do Expoente	Expoente		Mantissa			
0	0	0	1	0	0	1	0
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

# Representação em Ponto Flutuante

- Bem, em 25
- Vamos
- Sinal.
- Manti
- Sinal Expoen
- Agora... Na mem

**Nunca remova  
zeros à esquerda  
na mantissa!**

Sinal	Sinal do Expoente	Expoente		Mantissa			
0	0	0	1	0	0	1	0
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0



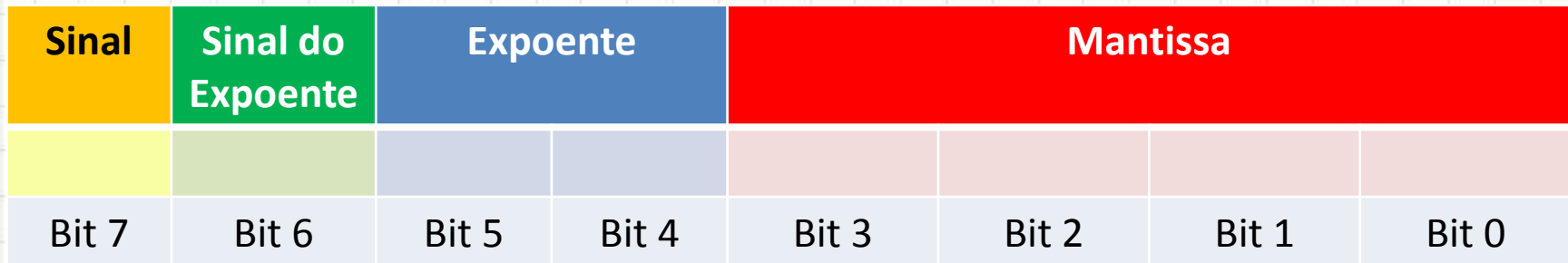
# **NOTAÇÃO IEEE**

## **754/2008**



# Representação IEEE 754/2008

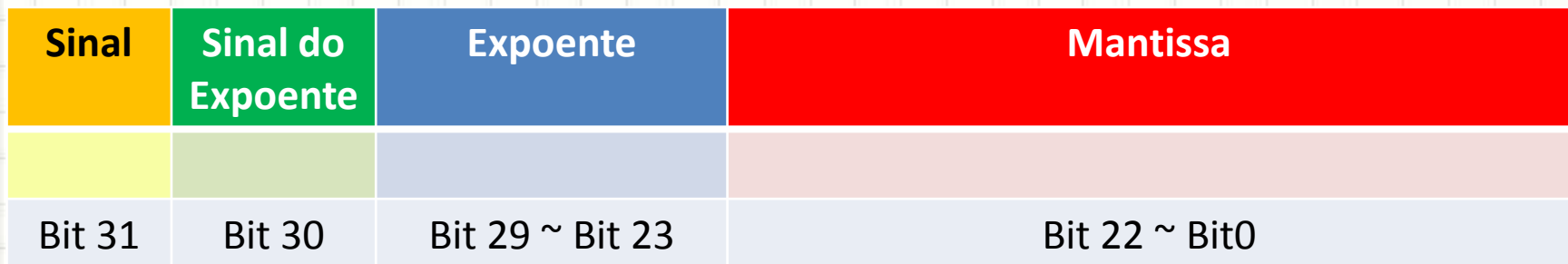
- Usar 8 bits é didático, mas muito limitado para a prática



- IEEE define dois padrões de números de ponto flutuante:
  - **Precisão Simples (32 bits)**
  - **Precisão Dupla (64 bits)**

# Representação IEEE 754/2008

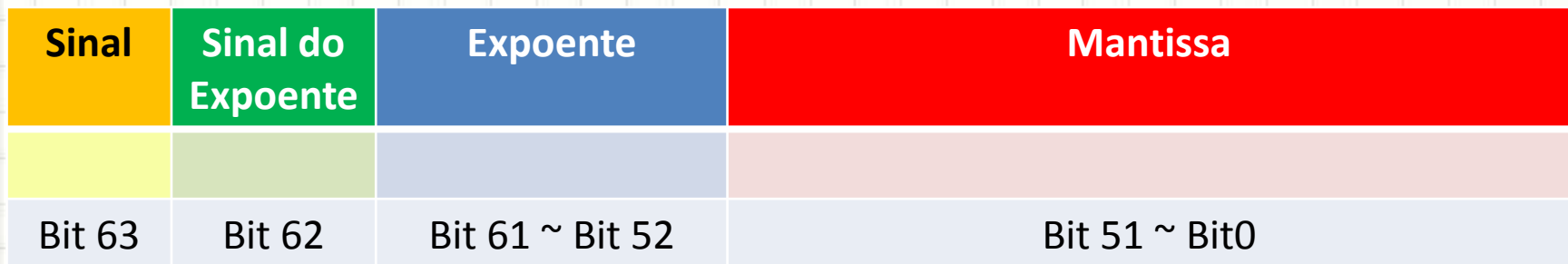
- Precisão Simples (32 bits)



- 1 bit para o sinal do número
- 8 bits para o expoente (incluindo sinal)
- 23 bits para a mantissa

# Representação IEEE 754/2008

- Precisão Dupla (64 bits)



- 1 bit para o sinal do número
- 11 bits para o expoente (incluindo sinal)
- 52 bits para a mantissa

# Representação IEEE 754/2008

- Comparativo

Tipo	Bits de Sinal	Bits de Expoente	Bits de Mantissa
Simple	1	8	23
Dupla	1	11	52

- Por que aumentar mais os bits de mantissa que os de expoente?
- PRECISÃO

# Representação IEEE 754/2008

- Com 11 bits de expoente, já é possível indicar 1024 casas binárias...
- É dígito que não acaba mais!
- Entretanto, não adianta ter 1 milhão de bits de expoente e ter apenas 1 bit de mantissa, pois é a mantissa que dá “detalhes” do número

# Representação IEEE 754/2008

- **Nenhum bit de mantissa:**
  - No caso extremo, se não tivéssemos nenhum bit de mantissa, poderíamos representar apenas potências de 2, o que é bem limitado
- **Nenhum bit de expoente:**
  - Por outro lado, se não tivermos nenhum bit de expoente, caímos no caso da representação em ponto fixo, que está muito longe de ser inútil!

# Representação IEEE 754/2008

- **DETALHE MUITO IMPORTANTE**
- Até agora, Bit de Sinal:
  - 0 é positivo
  - 1 é negativo
- Na notação IEEE, isso vale para o sinal do número...
- Mas no sinal do expoente, o bit de sinal é invertido, ou seja:
  - 0 é negativo
  - 1 é positivo

# Representação IEEE 754/2008

## • DETALHES IMPORTANTES

- Até 2008, a representação de números em ponto flutuante era feita de forma diferente.
  - 0 é positivo
  - 1 é negativo
- Na notação científica, o sinal do número
- No sinal do expoente, o bit de sinal é invertido, ou seja:
  - 0 é negativo
  - 1 é positivo

**Ué?! Por quê?  
Para complicar a  
vida?**

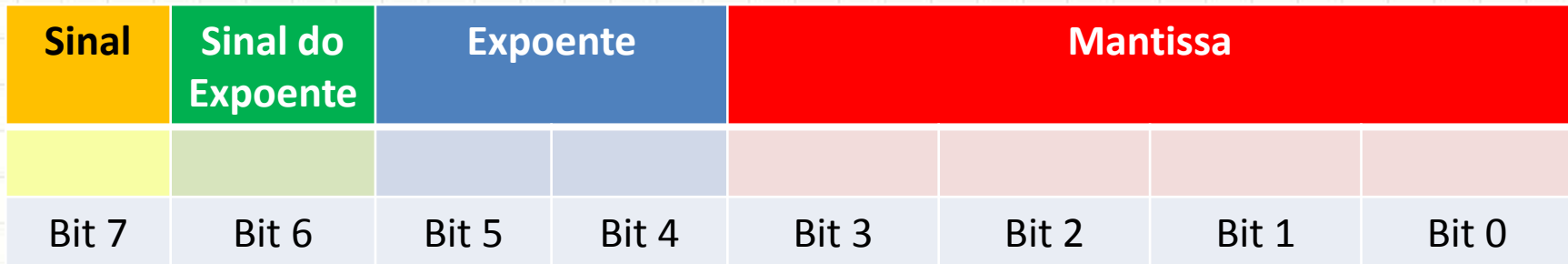




# REPRESENTAÇÃO DO ZERO

# Representação do Zero

- Voltemos ao nosso caso com apenas 8 bits



- Zero seria  $0,0b * 2^0$

**Sinal:** 0 (positivo)

**Característica:** 0b

**Mantissa:** 0b

**Sinal Expoente:** 1 (positivo) **Expoente:** 0b

- **Como indicar característica 0b !?!?!?**

# Representação do Zero

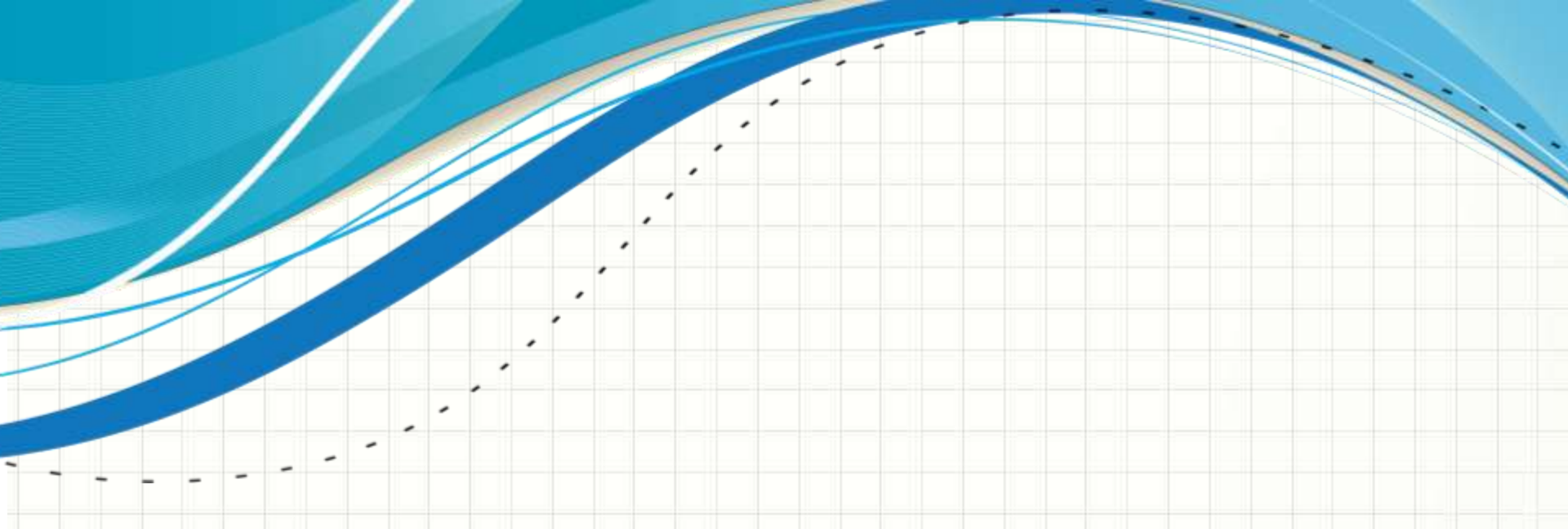
- IEEE determinou que se o expoente é “-0”...
- ... Então a característica é considerada 0b!
- Assim, o número ZERO pode ser escrito como

Sinal	Sinal do Expoente	Expoente		Mantissa			
0	0	0	0	0	0	0	0
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

- Note que o uso de bit de sinal invertido no expoente (0 é negativo) permite que o valor 0 seja representado como “tudo 0”!

# Representação do Zero

- Expoente  $\neq -0$ : **notação normalizada**
  - Característica é 1b
- Expoente = -0: **notação não-normalizada**
  - Característica é 0b



# **CONTAS EM PONTO FLUTUANTE**

# Cálculos em Ponto Flutuante

- Uma série de transformações são necessárias...
- Basicamente, os elementos precisam ter o mesmo expoente para serem somados
- Complicado!
- Por isso a maioria dos computadores tem uma...
- FPU – Floating Point Unit



# REPRESENTAÇÃO DE CARACTERES

# Representação de Caracteres

- Vimos como representar números...
- Mas como representar letras?
- Problema antigo: surgiu com a computação
- Tabela ASCII
  - American Standard for Computer Information Interchange
- Cada um dos códigos visuais de caracteres são mapeados para um número



# Representação de Caracteres

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	<b>NUL</b> (null)	32	20	040	&#32;	Space	64	40	100	&#64;	@	96	60	140	&#96;	`
1	1	001	<b>SOH</b> (start of heading)	33	21	041	&#33;	!	65	41	101	&#65;	A	97	61	141	&#97;	a
2	2	002	<b>STX</b> (start of text)	34	22	042	&#34;	"	66	42	102	&#66;	B	98	62	142	&#98;	b
3	3	003	<b>ETX</b> (end of text)	35	23	043	&#35;	#	67	43	103	&#67;	C	99	63	143	&#99;	c
4	4	004	<b>EOT</b> (end of transmission)	36	24	044	&#36;	\$	68	44	104	&#68;	D	100	64	144	&#100;	d
5	5	005	<b>ENQ</b> (enquiry)	37	25	045	&#37;	%	69	45	105	&#69;	E	101	65	145	&#101;	e
6	6	006	<b>ACK</b> (acknowledge)	38	26	046	&#38;	&	70	46	106	&#70;	F	102	66	146	&#102;	f
7	7	007	<b>BEL</b> (bell)	39	27	047	&#39;	'	71	47	107	&#71;	G	103	67	147	&#103;	g
8	8	010	<b>BS</b> (backspace)	40	28	050	&#40;	(	72	48	110	&#72;	H	104	68	150	&#104;	h
9	9	011	<b>TAB</b> (horizontal tab)	41	29	051	&#41;	)	73	49	111	&#73;	I	105	69	151	&#105;	i
10	A	012	<b>LF</b> (NL line feed, new line)	42	2A	052	&#42;	*	74	4A	112	&#74;	J	106	6A	152	&#106;	j
11	B	013	<b>VT</b> (vertical tab)	43	2B	053	&#43;	+	75	4B	113	&#75;	K	107	6B	153	&#107;	k
12	C	014	<b>FF</b> (NP form feed, new page)	44	2C	054	&#44;	,	76	4C	114	&#76;	L	108	6C	154	&#108;	l
13	D	015	<b>CR</b> (carriage return)	45	2D	055	&#45;	-	77	4D	115	&#77;	M	109	6D	155	&#109;	m
14	E	016	<b>SO</b> (shift out)	46	2E	056	&#46;	.	78	4E	116	&#78;	N	110	6E	156	&#110;	n
15	F	017	<b>SI</b> (shift in)	47	2F	057	&#47;	/	79	4F	117	&#79;	O	111	6F	157	&#111;	o
16	10	020	<b>DLE</b> (data link escape)	48	30	060	&#48;	0	80	50	120	&#80;	P	112	70	160	&#112;	p
17	11	021	<b>DC1</b> (device control 1)	49	31	061	&#49;	1	81	51	121	&#81;	Q	113	71	161	&#113;	q
18	12	022	<b>DC2</b> (device control 2)	50	32	062	&#50;	2	82	52	122	&#82;	R	114	72	162	&#114;	r
19	13	023	<b>DC3</b> (device control 3)	51	33	063	&#51;	3	83	53	123	&#83;	S	115	73	163	&#115;	s
20	14	024	<b>DC4</b> (device control 4)	52	34	064	&#52;	4	84	54	124	&#84;	T	116	74	164	&#116;	t
21	15	025	<b>NAK</b> (negative acknowledge)	53	35	065	&#53;	5	85	55	125	&#85;	U	117	75	165	&#117;	u
22	16	026	<b>SYN</b> (synchronous idle)	54	36	066	&#54;	6	86	56	126	&#86;	V	118	76	166	&#118;	v
23	17	027	<b>ETB</b> (end of trans. block)	55	37	067	&#55;	7	87	57	127	&#87;	W	119	77	167	&#119;	w
24	18	030	<b>CAN</b> (cancel)	56	38	070	&#56;	8	88	58	130	&#88;	X	120	78	170	&#120;	x
25	19	031	<b>EM</b> (end of medium)	57	39	071	&#57;	9	89	59	131	&#89;	Y	121	79	171	&#121;	y
26	1A	032	<b>SUB</b> (substitute)	58	3A	072	&#58;	:	90	5A	132	&#90;	Z	122	7A	172	&#122;	z
27	1B	033	<b>ESC</b> (escape)	59	3B	073	&#59;	:	91	5B	133	&#91;	[	123	7B	173	&#123;	{
28	1C	034	<b>FS</b> (file separator)	60	3C	074	&#60;	<	92	5C	134	&#92;	\	124	7C	174	&#124;	
29	1D	035	<b>GS</b> (group separator)	61	3D	075	&#61;	=	93	5D	135	&#93;	]	125	7D	175	&#125;	}
30	1E	036	<b>RS</b> (record separator)	62	3E	076	&#62;	>	94	5E	136	&#94;	^	126	7E	176	&#126;	~
31	1F	037	<b>US</b> (unit separator)	63	3F	077	&#63;	?	95	5F	137	&#95;	_	127	7F	177	&#127;	DEL

# Representação de Caracteres

- Esta tabela define os caracteres de 0 a 127
- Os caracteres de 128 a 255 são “extras”
- Cada país implementou a sua extensão, para os seus acentos, chamada “codepage”
- Isso criou muita confusão e, então, criaram os padrões mundiais UNICODE
- Os tipos comuns são UTF-8, UTF-16 e UTF-32

# Representação de Caracteres

- UTF: Unicode Transformation Format
  - UTF-8: 256 caracteres
  - UTF-16: 65536 caracteres
  - UTF-32: 4 bilhões de caracteres
- UTF-8 é compatível com os 128 primeiros caracteres do ASCII
- UTF-16 é compatível com UTF-8
- UTF-32 é compatível com UTF-16



**CONCLUSÕES**

# Resumo

- É possível representar números muito grandes e muito pequenos com o uso de ponto flutuante
- O formato padrão de representar ponto flutuante no computador é o estabelecido pela IEEE 754/2008.
- Os caracteres também são representados como números ... E existem várias codificações possíveis
- **TAREFA**
  - Lista 1!

# Próxima Aula



- Ok... Mas como é o lugar onde estes números ficam?
  - Como eles são escritos?
  - Como eles são acessados?



**PERGUNTAS?**



**BOM DESCANSO  
A TODOS!**