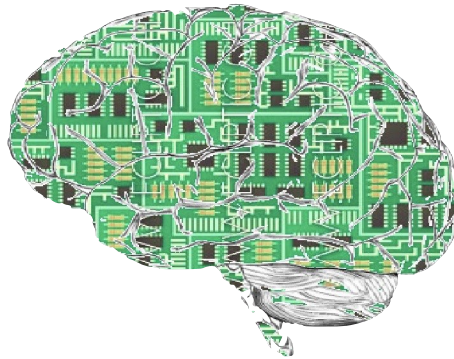


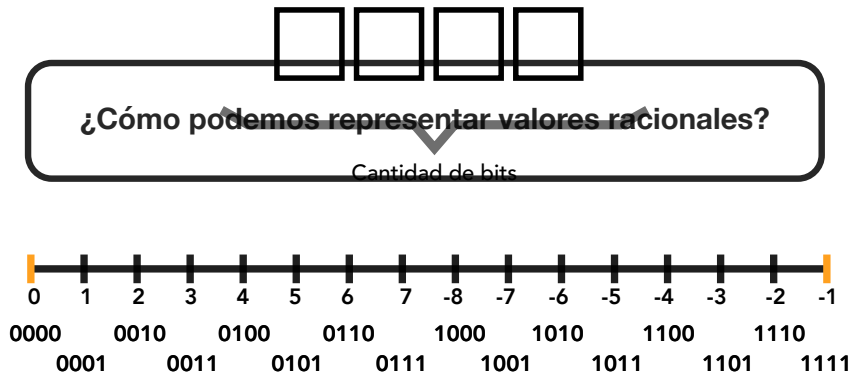
DEPARTAMENTO DE  
COMPUTACIÓN  
UNRC

## ORGANIZACIÓN DEL PROCESADOR



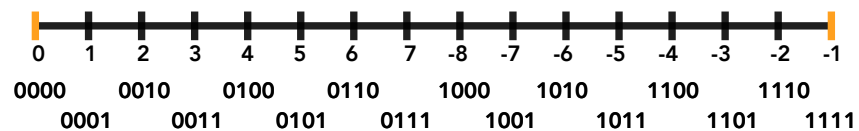
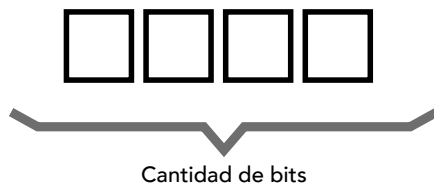
## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES



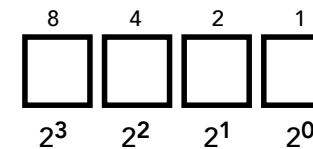
## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES



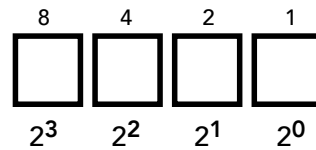
## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES



## ORGANIZACIÓN DEL PROCESADOR

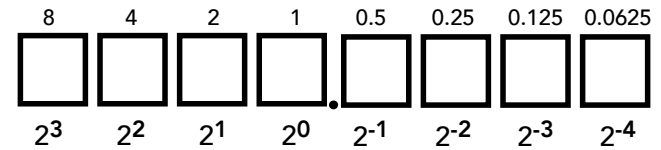
### REPRESENTACIÓN DE NÚMEROS RACIONALES



¿ 6,82 ?

## ORGANIZACIÓN DEL PROCESADOR

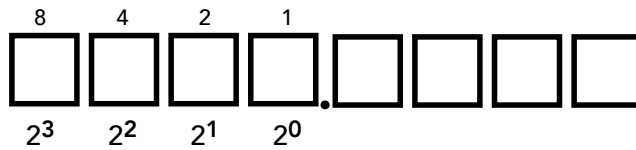
### REPRESENTACIÓN DE NÚMEROS RACIONALES



¿ 6,82 ?

## ORGANIZACIÓN DEL PROCESADOR

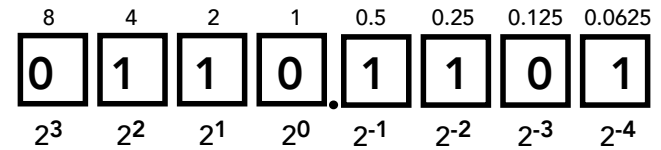
### REPRESENTACIÓN DE NÚMEROS RACIONALES



¿ 6,82 ?

## ORGANIZACIÓN DEL PROCESADOR

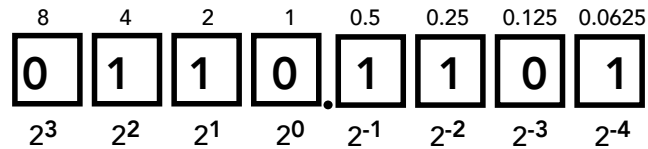
### REPRESENTACIÓN DE NÚMEROS RACIONALES



¿ 6,82 ?

## ORGANIZACIÓN DEL PROCESADOR

### REPRESENTACIÓN DE NÚMEROS RACIONALES

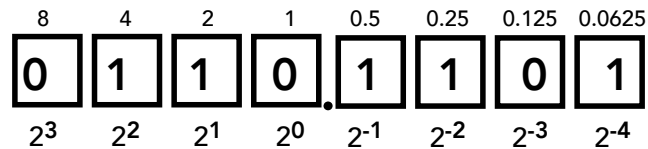


$$0+4+2+0+0.5+0.25+0+0.625$$

¿ 6,82 ?

## ORGANIZACIÓN DEL PROCESADOR

### REPRESENTACIÓN DE NÚMEROS RACIONALES

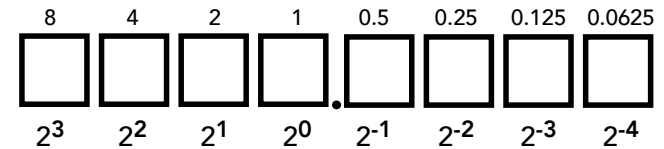


$$0+4+2+0+0.5+0.25+0+0.625$$

¿ 6,82 ?  $\simeq$  6,8125

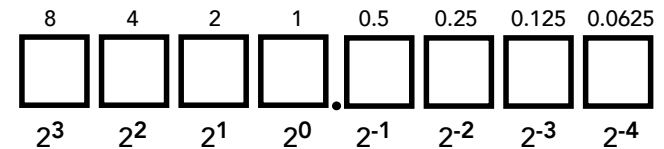
## ORGANIZACIÓN DEL PROCESADOR

### REPRESENTACIÓN DE NÚMEROS RACIONALES



## ORGANIZACIÓN DEL PROCESADOR

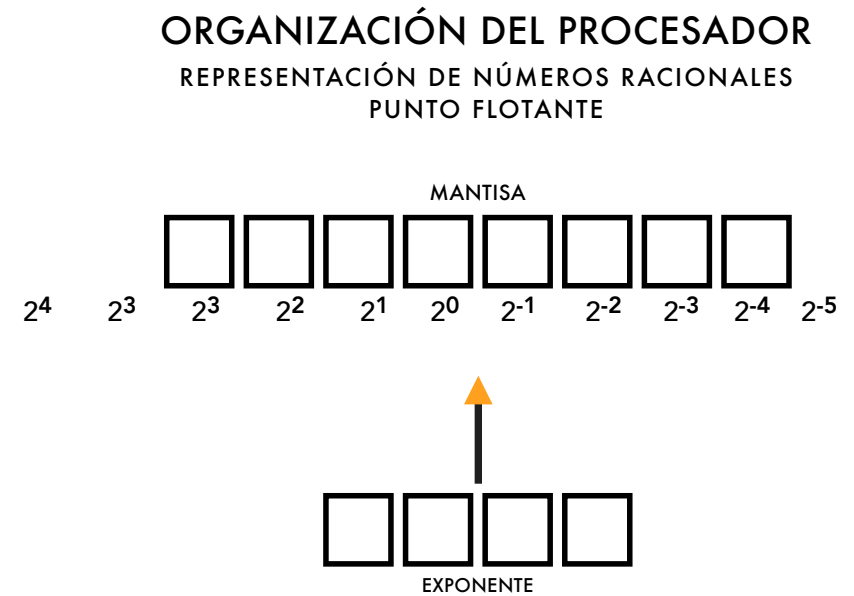
### REPRESENTACIÓN DE NÚMEROS RACIONALES



Necesitamos muchos bits para lograr tener una representación relativamente aceptable

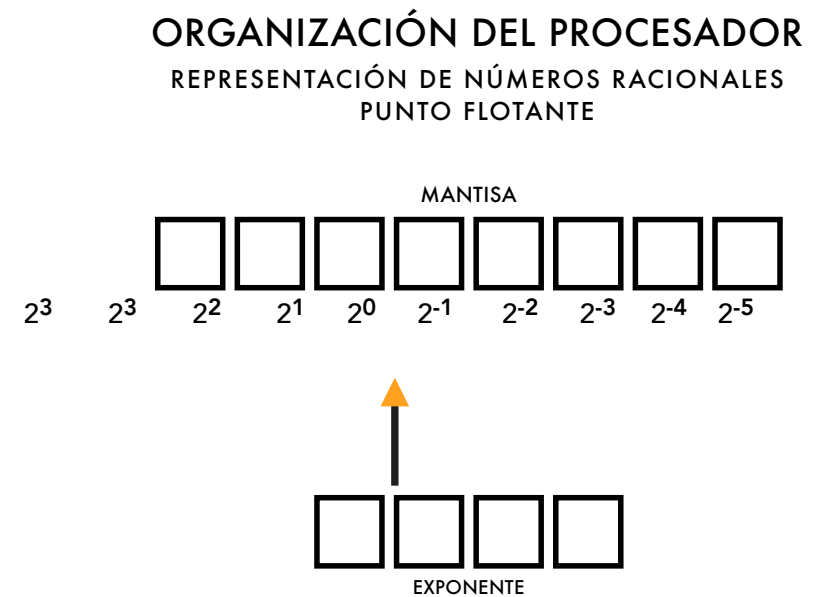
## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES  
PUNTO FLOTANTE



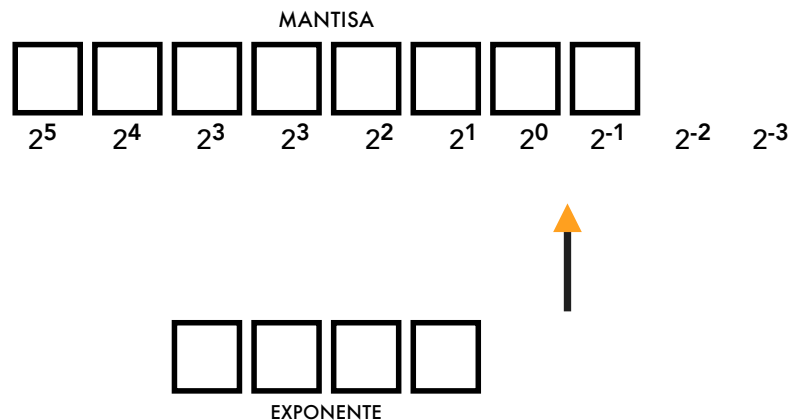
## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES  
PUNTO FLOTANTE



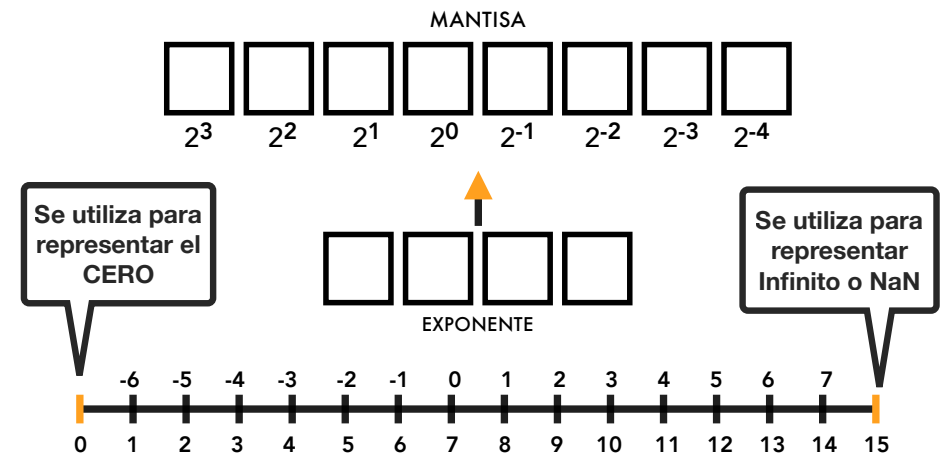
## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES  
PUNTO FLOTANTE



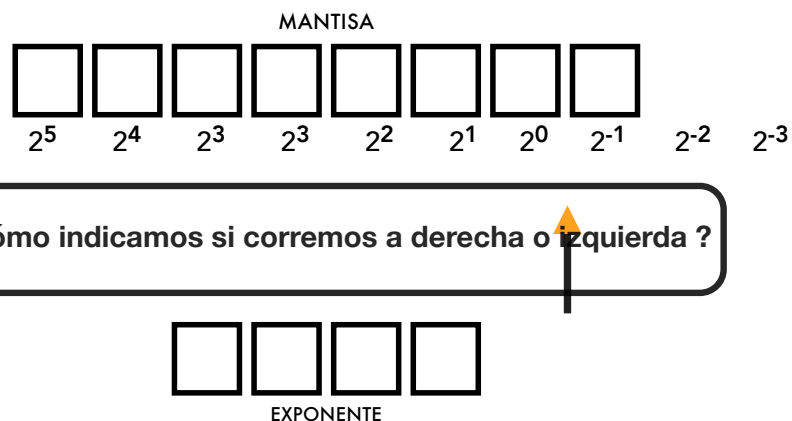
## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES  
PUNTO FLOTANTE - POLARIZACIÓN



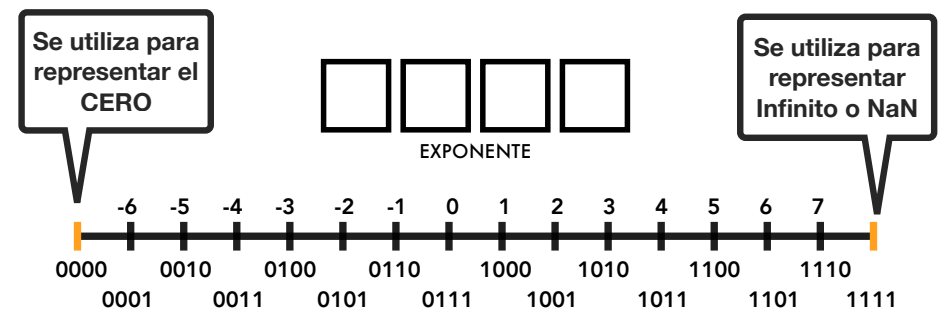
## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES  
PUNTO FLOTANTE



## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES  
PUNTO FLOTANTE - POLARIZACIÓN

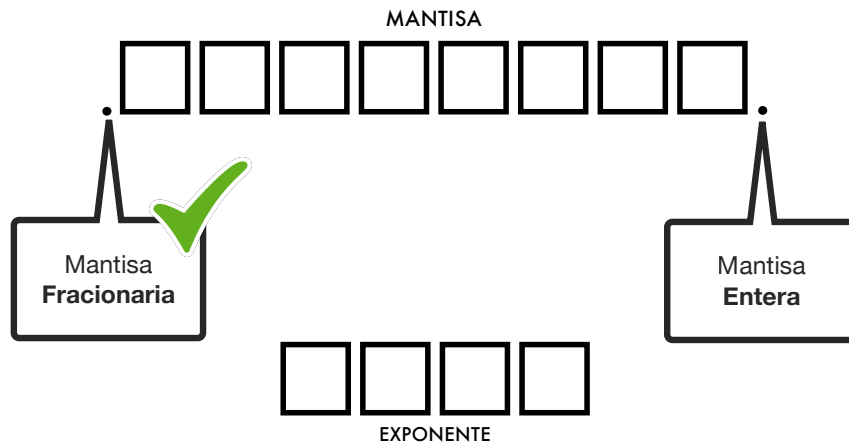


$$\text{exponente} = \text{exponente real} + 7$$

El exponente representado se calcula sumando al exponente el valor que representa el bit más significativo del tamaño destinado al exponente -1 (IEEE estandar)

## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES  
PUNTO FLOTANTE - POLARIZACIÓN



## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES  
PUNTO FLOTANTE - ESTÁNDARES

Estandares (IEEE 1985)

Single: 32bits

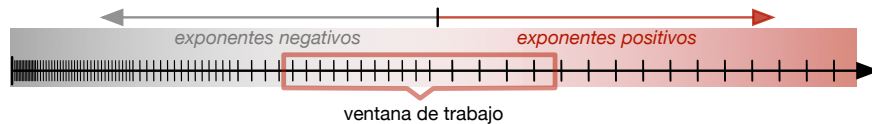
- 1 bit Signo
- 8 bits Exponente.
- 23 bits Mantisa

Double: 64bits

- 1 bit Signo
- 11 bits Exponente.
- 52 bits Mantisa

## ORGANIZACIÓN DEL PROCESADOR

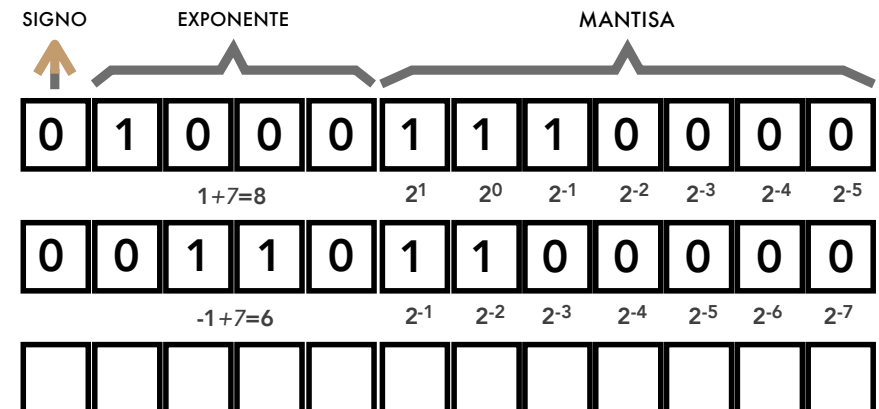
REPRESENTACIÓN DE NÚMEROS RACIONALES  
PUNTO FLOTANTE



## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES  
PUNTO FLOTANTE - SUMA

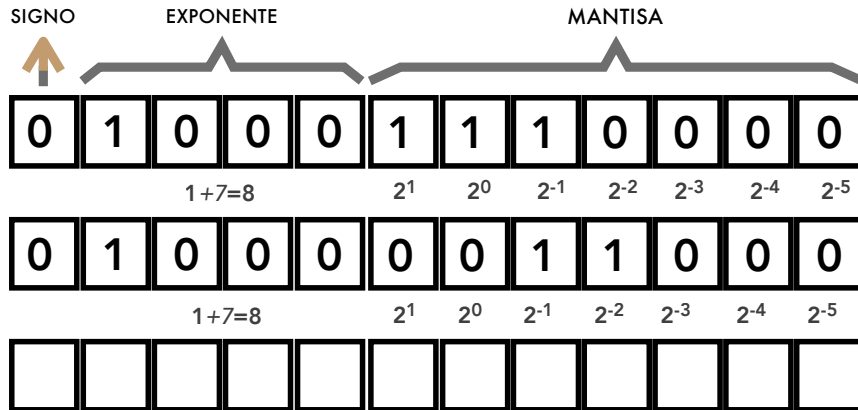
$$3,50 + 0,75 = 4,25$$



## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES  
PUNTO FLOTANTE - SUMA

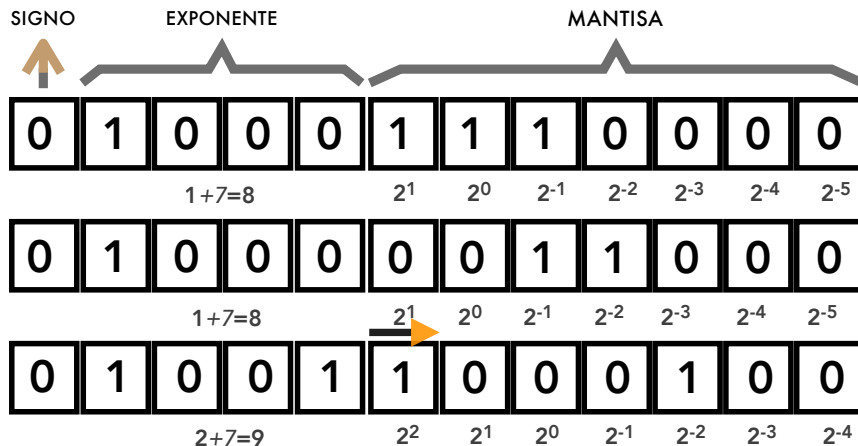
$$3,50 + 0,75 = 4,25$$



## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES  
PUNTO FLOTANTE - SUMA

$$3,50 + 0,75 = 4,25$$



## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES  
PUNTO FLOTANTE - SUMA/RESTA

### Algoritmo de Suma/Resta:

- 1) Elegir el operando con **menor exponente**, *shiftear* la mantisa hacia la derecha incrementando el exponente hasta que éste coincida con el del otro operando.
- 2) Operar con las mantisas como la **suma/resta** de enteros.
- 3) Modificar el **signo** si fuese necesario.
- 4) **Normalizar** si fuese necesario.

## ORGANIZACIÓN DEL PROCESADOR

REPRESENTACIÓN DE NÚMEROS RACIONALES  
PUNTO FLOTANTE - MULTIPLICACIÓN

### Algoritmo de Multiplicación:

- 1) **Sumar** los exponentes, restando una vez la polaridad para mantenerla. (Ejemplo IEEE Single -127)
- 2) **Multiplicar** las mantisas
- 3) Modificar el **signo** si fuese necesario
- 4) **Normalizar** si fuese necesario

# ORGANIZACIÓN DEL PROCESADOR

## REPRESENTACIÓN DE NÚMEROS RACIONALES

### PUNTO FLOTANTE - DIVISIÓN

#### Algoritmo de División:

- 1) **Restar** los exponentes, sumando una vez la polaridad para mantenerla. (Ejemplo IEEE Single -127)
- 2) **Dividir** las mantisas
- 3) Modificar el **signo** si fuese necesario
- 4) **Normalizar** si fuese necesario

# ORGANIZACIÓN DEL PROCESADOR

## REPRESENTACIÓN DE CARACTERES

ABC

# ORGANIZACIÓN DEL PROCESADOR

## REPRESENTACIÓN DE CARACTERES

ASCII (*American Standard Code for Information Interchange*) en 1967.

Fue sucesor de los códigos Baudot (telegrafía 5bits) y el código Murray (desarrollado para las máquinas de escribir “typewriter”).

- 32 caracteres de control
- 10 dígitos
- 52 letras (mayúsculas y minúsculas)
- 32 caracteres especiales
- 1 espacio

# ORGANIZACIÓN DEL PROCESADOR

## REPRESENTACIÓN DE CARACTERES

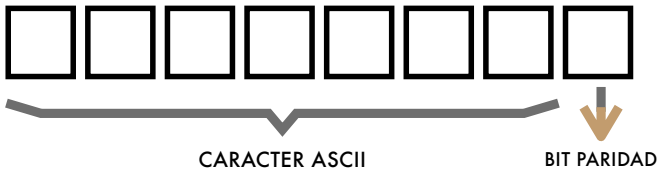
ASCII 7 bits											
Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(	72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29	)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[ENG OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[	123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D	]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]



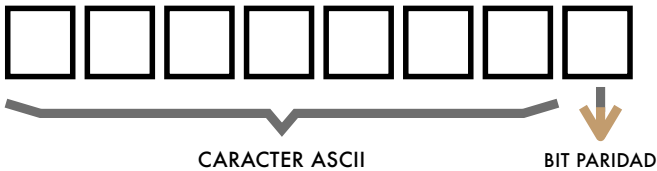
ORGANIZACIÓN DEL PROCESADOR  
REPRESENTACIÓN DE CARACTERES

ASCII 7 bits											
Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	\$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(	72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29	)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[ENG OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[	123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D	]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

ORGANIZACIÓN DEL PROCESADOR  
REPRESENTACIÓN DE CARACTERES ASCII

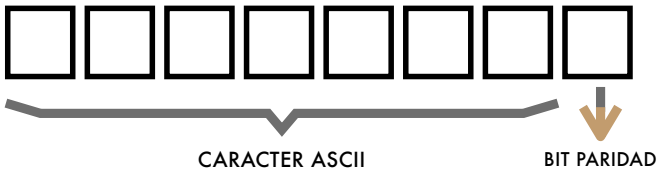


ORGANIZACIÓN DEL PROCESADOR  
REPRESENTACIÓN DE CARACTERES ASCII



( + + + + + + + ) mod 2

ORGANIZACIÓN DEL PROCESADOR  
REPRESENTACIÓN DE CARACTERES ASCII



( + + + + + + + ) mod 2

Bit para verificación

# ORGANIZACIÓN DEL PROCESADOR

## REPRESENTACIÓN DE CARACTERES ASCII EXTENDIDO

**ASCII Extendido** en 1980.

Debido a la masificación de las computadoras se comenzó a utilizar el último bit para poder representar una mayor cantidad de símbolos: ñ Ñ Ç etc.

256 Símbolos (compatibles con ASCII)

# ORGANIZACIÓN DEL PROCESADOR

## REPRESENTACIÓN DE CARACTERES UNICODE

- **Unicode** fue presentado en 1991
- utiliza **16 bits**
- permite codificar todos los símbolos de mundo
- provee un mecanismo de extensión que permite codificar millones de caracteres
- es compatible con ASCII