Circuitos Electrónicos Digitales

Universidad de Sevilla
2011-2012

Tema II

Representación binaria

Índice

1. Sistemas de numeración
2. Códigos binarios
3. Aritmética Binaria
4. Representación de números con signo

Representación Posicional de Magnitudes

$$5734 = 5 \times 10^3 + 7 \times 10^2 + 3 \times 10^1 + 4 \times 10^0$$

Base

Unidades de millar

Peso

Posición:	3	2	1	0
	5	7	3	4

Base: Número de dígitos distintos que pueden emplearse para representar una magnitud con el sistema utilizado.

Representación Posicional de Magnitudes

Bases interesantes

Base 2: Binario → 0, 1

$010011_{(2} = 19_{(10}$

Base 8: Octal → 0, 1, 2, 3, 4, 5, 6, 7

$47_{(8} = 39_{(10}$

Base 16: Hexadecimal → 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

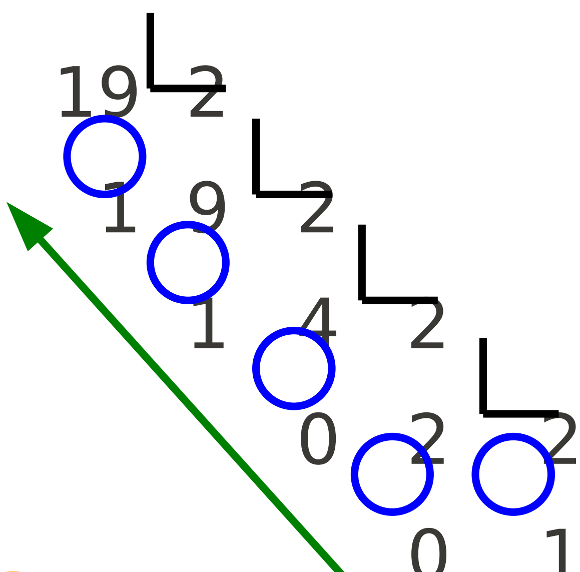
$2A_{(16} = 42_{(10}$

Transformaciones entre bases

Base 2 a Base 10:

$$010011_{(2} = 0 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 19_{(10}$$

Base 10 a Base 2: $19_{(10} = 10011_{(2}$



Representación Posicional de Magnitudes

Transformaciones especiales

Base 2 a Base 16: $\boxed{16 = 2^{4}}$

$010011_{(2} = 0001\ 0011_{(2} = 13_{(16}$

Base 16 a Base 2:

$A\ 7_{(16} = 1010\ 0111_{(2}$

Base 2 a Base 8: $\boxed{8 = 2^{3}}$

$010011_{(2} = 010\ \ 011_{(2} = 2\ 3_{(8}$

Base 8 a Base 2:

$3\ 7_{(8} = 011\ \ 111_{(2}$

Representación parte fraccionaria

Entera fraccionaria

Ejemplo: 0 1 0 1 1 . 1 1 0

Base 2 a Base 10:

$$0.110_{(2} = 1 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3} = 0.75_{(10}$$

Base 10 a Base 2:

$$0.65 \times 2 = 1\ .\ 30$$

$$0.30 \times 2 = 0\ .\ 60$$

$$0.75_{(10} = 0.\ 1\ 0_{(2}$$

Índice

Códigos Binarios

- BCD
- 7 Segmentos
- Gray
- Detectores de Errores
- ASCII

Códigos Binarios

BCD

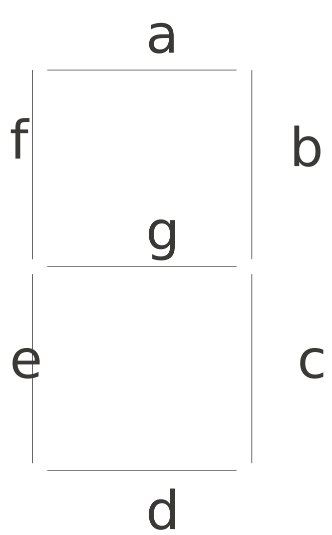
Dígito	Código BCD
0	0000
1	0001
2	0010
3	0011
4	0100

Dígito	Código BCD
5	0101
6	0110
7	0111
8	1000
9	1001

Ejemplo: $16_{(10} = 0001\ 0110_{(BCD}$

Códigos Binarios

7 Segmentos



Dígito	Código 7-Seg abcdefg
0	1111110
1	0110000
2	1101101
3	1111001
4	0110011

Dígito	Código 7-Seg abcdefg
5	1011011
6	0011111
7	1110000
8	1111111
9	1110011

Códigos Binarios

Gray

Dígito	Código Gray 1 bit
0	0
1	1

Dígito	Código Gray 2 bits
0	0 0
1	0 1
2	1 1
3	1 0

Espejo

Códigos Binarios

Detectores de errores

- Bit de Paridad: Se añade un bit (más significativo) al código binario, denominado **bit de paridad**. Puede hacerse de dos formas:

1. Paridad Par: El número total de 1s debe ser par.

2. Paridad impar: El número total de 1s debe ser impar.

Código	Bit Paridad Par	Código con Paridad Par	Bit Paridad Impar	Código con Paridad Impar
0000	0	0 0000	1	1 0000
1011	1	1 1011	0	0 1011

Códigos Binarios

ASCII

American Standard Code for Information Interchange (ASCII)

	$B_7B_6B_5$							
$B_4B_3B_2B_1$	000	001	010	011	100	101	110	111
0000	NULL	DLE	SP	0	@	P	`	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	'	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	|
1101	CR	GS	-	=	M	]	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DEL

Indice

Aritmética binaria

Suma aritmética

$$
\begin{array}{r}
1\;0\;0\;1\;1\;0 \\
+\quad 1\;1\;0\;1 \\
\hline
\end{array}
$$

Aritmética binaria

Suma aritmética

	0	1	1	0	0		Acarreo
	1	0	0	1	1	0	Sumando 1
+			1	1	0	1	Sumando 2
	1	1	0	0	1	1	Suma

Aritmética binaria

Resta aritmética

$$
\begin{array}{rcccccc}
 & 1 & 0 & 0 & 1 & 1 & 0 \\
- & & & 1 & 1 & 0 & 1 \\
\hline
\end{array}
$$

Aritmética binaria

Resta aritmética

	1	0	0	1	1	0	Minuendo
-			1	1	0	1	Sustraendo
	1	1	0	0	1		Acarreo
	0	1	1	0	0	1	Resta

Aritmética binaria

Multiplicación

$$\begin{array}{r} 1\;0\;0\;1\;1\;0 \\ \times \quad 1\;1\;0\;1 \\ \hline \end{array}$$

Aritmética binaria

Multiplicación

			1	0	0	1	1	0		Multiplicando
			X			1	1	0	1	Multiplicador
			1	0	0	1	1	0		
		0	0	0	0	0	0			
	1	0	0	1	1	0				
1	0	0	1	1	0					
1	1	1	1	0	1	1	1	0		Multiplicación

Aritmética binaria

División

1 0 0 1 1 0	1 1 0 1

Aritmética binaria

División

Dividendo 1 0 0 1 1 0 | 1 1 0 1 Divisor

\- 1 1 0 1 ↓ 1 0 Cociente

0 1 1 0 0

\- 0 0 0 0

1 1 0 0 Resto

Índice

Aritmética binaria

Representación de números con signo:

- Signo-Magnitud
- Complemento a 1
- Complemento a 2

Aritmética binaria

Signo-Magnitud

Signo Magnitud

$+90_{(10}$ →	0	1	0	1	1	0	1	0
$-90_{(10}$ →	1	1	0	1	1	0	1	0

- Signo: Positivo (0) ; Negativo (1)
- Representable con n bits: $(2^n - 1)$ números
- Doble representación del cero (4 bits): 0000 y 1000

Aritmética binaria

Complemento a 1 (Ca1)

- Números positivos
  - Se representa el número en base 2
  - El MSB debe ser siempre un cero
  - Ejemplo: + $90_{(10} \rightarrow$ 01011010
- Números negativos
  - Se obtiene el Ca1 de su representación como número positivo.
  - El MSB siempre será un uno
  - Ejemplo: $-90_{(10} \rightarrow$ 1010010

- Obtención del Ca1: se complementan todos los bits
- Representable con n bits: (2^n – 1) números
- Doble representación del cero (4 bits): 0000 y 1111

Aritmética binaria

Complemento a 2 (Ca2)

- Números positivos
  - Se representa el número en base 2
  - El MSB debe ser siempre un cero
  - Ejemplo: + $90_{(10}$ → 01011010
- Números negativos
  - Se obtiene el Ca2 de su representación como número positivo.
  - El MSB siempre será un uno
  - Ejemplo: $-90_{(10}$ → 10100110

- Obtención del Ca2: comenzando por el LSB, se conservan los bits a cero y el primer uno, complementando el resto.
- Representable con n bits: (2^n) números
- Representación única del cero (4 bits): 0000

Aritmética binaria

Representación de números con signo, ejemplo 4 bits

- Para los números > 0
  - Son iguales
- Para los números < 0
  - $N_{(CA2} = N_{(CA1} + 1$

	S-M	Ca1	Ca2
7	**0111**	**0111**	**0111**
6	**0110**	**0110**	**0110**
5	**0101**	**0101**	**0101**
4	**0100**	**0100**	**0100**
3	**0011**	**0011**	**0011**
2	**0010**	**0010**	**0010**
1	**0001**	**0001**	**0001**
0	**0000/1000**	**0000/1111**	**0000**
-1	**1001**	**1110**	**1111**
-2	**1010**	**1101**	**1110**
-3	**1011**	**1100**	**1101**
-4	**1100**	**1011**	**1100**
-5	**1101**	**1010**	**1011**
-6	**1110**	**1001**	**1010**
-7	**1111**	**1000**	**1001**
-8	**-**	**-**	**1000**

Aritmética binaria

- Aritmética en S-M
  - Para obtener la suma de dos números en S-M:
    - Si ambos tienen el mismo signo
      - La magnitud del resultado coincide con la suma de las magnitudes.
      - El bit de signo del resultado es el mismo que el de cualquiera de los sumandos.
    - Si los números tienen distinto signo
      - La magnitud del resultado se obtiene restando la magnitud menor de la mayor.
      - El signo del resultado se corresponde con el signo que tenga la magnitud mayor.

Aritmética binaria

- Aritmética en Ca1 y Ca2
  - Con la notación en Ca1 y Ca2:

    $A-B = A+(-B) = A+B_{(Ca1}$

    $A-B = A+(-B) = A+B_{(Ca2}$
  - En Ca1 si C_{OUT} =1 se añade al resultado

1 1 0 0 0	**acarreos**	**0 0 1 1**	**acarreos**
1 1 0 1 0	**-5**	**1 0 0 1 1**	
1 1 0 0 1	**-6**	**+ 1**	
1 0 0 1 1	**-12**	**1 0 1 0 0**	**-11 (OK)**

  - En Ca2 si C_{OUT} =1 se desprecia

1 1 0 1 0	**acarreos**
1 1 0 1 1	**-5**
1 1 0 1 0	**-6**
1 0 1 0 1	**-11 (OK)**

  - Esto hace que Ca2 sea la más utilizada