

# Resumen Electrónica Digital

HOJA N° [1]

FECHA \_\_\_\_\_

## Sistemas Numéricos posicionales

Pasar de base  $b \geq$  decimal:

$$(d_3 d_2 d_1 d_0)_b = b^3 \cdot d_3 + b^2 \cdot d_2 + b^1 \cdot d_1 + b^0 \cdot d_0$$

Pasar de base decimal a binario

$$\begin{array}{r} 5 \\ | \\ 1 \end{array} \quad \begin{array}{r} 2 \\ | \\ 2 \end{array} \quad \begin{array}{r} 1 \\ | \\ 1 \end{array} \quad \begin{array}{r} 2 \\ | \\ 0 \end{array} \Rightarrow (101)_2$$

Pasar de base 8 a binario y viceversa

$$(523)_8 = (101010011)_2 \Rightarrow \text{c/u son 3 bit}$$

Pasar de base 16 a binario y viceversa

$$(153)_{16} = (000101010011)_2 \Rightarrow \text{c/u son 4 bit}$$

Pasar de base 8 a 16 y viceversa → Pasar 1º a binario

Sistema octal → 0523

Le indica a la computadora que está en octal

## Representación de no Significativos

→ Signo y Magnitud → No se usa en computadoras a nivel de hardware

S	Magnitud
---	----------

0 → 0 Binario Natural  
1 → 0

Rango: Es simétrico y hay 2 ceros (+0 -0)  
En sistema de 5 bits de [-15, 15]  
de 4 bits de [-7, 7]

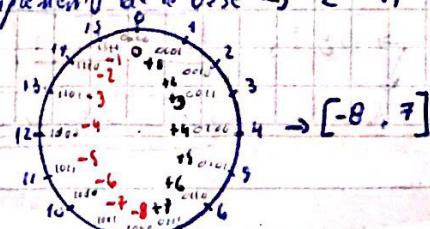
→ Complemento a la base → para sistema de n bits se define base.

$$b = 2^n$$

Si  $n > 0$  → se lo representa por su binario natural  
Si  $n < 0$  → se lo representa por su complemento de la base  $\rightarrow 2^n - n$

Rango: Asimétrico, tiene un negativo más

→ Sistema de 4 bits →



VOTA

→ El primer bit me puede decir si es negativo o positivo ( $0 \rightarrow +, 1 \rightarrow -$ ) pero no es un bit de signo →  $\begin{array}{r} 1 \\ \times \\ 1010 \\ \hline -1010 \end{array} = -6$

$$\begin{array}{l} 3 = 000111 \\ \downarrow \downarrow \downarrow \downarrow \downarrow \\ CB_3 = 111011 \\ -5 = \underline{\begin{array}{r} 111011 \\ -2^3 2^2 2^1 2^0 2^4 2^5 \end{array}} \end{array}$$

$6 = 001110$   $\Rightarrow D_2 \leftrightarrow$  copio  $\ominus$  hasta el  $2^0$  "1"  
 $CB_6 = 11010$  y después copio al revés  
 $\rightarrow D_0 \rightarrow$ , el resto  $\ominus$  y después todos  $\oplus$  y el resultado se lo sumo.

### Suma binaria



Suma  $2^n$  de m bits  $\Rightarrow$

$$A = \begin{array}{c} \text{Cout} \\ \downarrow \\ a_{n-1} \ a_{n-2} \dots a_2 \ a_1 \ a_0 \end{array}$$

$$B = \begin{array}{c} \text{Cout} \\ \downarrow \\ b_{n-1} \ b_{n-2} \dots b_2 \ b_1 \ b_0 \end{array}$$

Carry out se ignora

Cout	S <sub>n-1</sub>	S <sub>n-2</sub>	...	S <sub>1</sub>	S <sub>0</sub>
------	------------------	------------------	-----	----------------	----------------

↓ Resultado

$$\begin{array}{r} +4 \\ +3 \\ \hline +7 \end{array} \Rightarrow \begin{array}{r} 00100 \\ 00011 \\ \hline 00111 \end{array} \boxed{7} \quad \begin{array}{r} +4 \\ -3 \\ \hline +1 \\ CB_3 \end{array} \Rightarrow 3 = 00011 \quad \begin{array}{r} 00100 \\ 11101 \\ \hline 00001 \end{array} \boxed{1}$$

$$\begin{array}{r} +4 \\ -3 \\ \hline +1 \\ CB_3 \end{array} \Rightarrow \text{idem} \quad \begin{array}{r} +4 \\ -3 \\ \hline +1 \\ CB_3 \end{array} \Rightarrow \begin{array}{r} 00100 \\ 11101 \\ \hline 00001 \end{array} \boxed{7}$$

"7-2 → Del  $\ominus$  desde el 7 avanza 30 lugares y el 1 ignorado es para dar 1 undte"

$$7+CB_2 \\ 7+30$$

"Positivo  $\oplus$  Positivo  $\text{No}$  pueden ser en los negativos" } → Overflow  
 "Negativo  $\oplus$  negativo  $\text{No}$  pueden ser en los positivos" }  
 "positivo  $\oplus$  negativo siempre se pude"

### Análisis de Overflow

Si  $C_{out} = C_{n-1} \Rightarrow$  Resultado Correcto

Si  $C_{out} \neq C_{n-1} \Rightarrow$  Overflow

### Extensión de signo

+5 → 5 bits  $\boxed{00101}$   $\stackrel{\text{① se agrega } 0}{\rightarrow}$

-5 → 5 bits  $\boxed{11011}$

$\ominus$  se segregan " " → 5 bits  $\boxed{11011}$

16 bits  $\boxed{11111111111111011}$

NOTA

## Multiplicación de Números Signados

Multiplicando  
x Multiplicador  
=====

$\begin{array}{r} \text{+9} \\ \hline x + 6 \\ \hline +54 \end{array}$	$\xrightarrow{\quad}$	$\begin{array}{r} 01001 \\ 00000000110 \\ 0000000000111 \\ 00100010000 \\ 00100010000 \\ 00100010000 \\ 00010011000 \\ 00010011000 \\ 00010011000 \\ 00001001100 \\ 00001001100 \\ 00001001100 \end{array}$
$r9 = 01001$	$\xrightarrow{\quad}$	$\begin{array}{r} 00010011000 \\ 00010011000 \\ 00010011000 \\ 00001001100 \\ 00001001100 \\ 00001001100 \end{array}$

$+9$	$\xrightarrow{+}$	0 1 0 0 1	
$-6$	$\xrightarrow{+}$	0 0 0 0 0 1 1 0 1 0 0	
$-54$	$\xrightarrow{+}$	0 0 0 0 0 0 1 1 0 0 0	
$+81$	$\xrightarrow{+}$	0 1 0 0 1 0 1 0 1 1 0 1	
$-9$	$\xrightarrow{+}$	0 0 1 0 0 1 0 1 1 1 0 0	
$+54$	$\xrightarrow{+}$	0 0 0 1 0 0 1 0 1 0 1 0	
$-10$	$\xrightarrow{+}$	0 1 0 1 1 0 1 0 1 0 1 0	
$+6$	$\xrightarrow{+}$	0 0 1 0 1 1 0 1 0 1 0 1	
$-6$	$\xrightarrow{+}$	0 1 1 1 0 1 0 1 0 1 0 1	
$-6$	$\xrightarrow{+}$	0 0 0 1 1 1 0 1 0 1 0 1	

Correction + CB      Multibit      1 0 1 1 1 0 0 0 0 0

Multiplicando


$$+9 \times (-6) \Rightarrow +9 \times (2^m - 6)$$

Melw.  
C. perotii Notizie

$$\begin{array}{r} -9 \\ \underline{-6} \\ \hline 15 \end{array} \quad \text{Combining like terms}$$

## División con Restauración

→ "suel carries ö" → te pasaste → hoy que volver

Extension  
de  
Signo

Dividendo      Divisor  
Resto      Cociente

1	CB Divisor	
0 0 0 0 0		Dividendo
Resto		Quociente

Dividendo	Divisor	Coiciente	Resto
+	+	+	+
+	-	-	+
-	NOTA	+	-
-	-	-	-

Sea cucumber

$\text{se h}_2\text{co} + \text{L}^+$

### Correciones:

Correcciones:

• Si Dividendo y Divisor son de signo

• Si dividendo y divisor son de signos  $\neq$  Complemento cociente

13	S	+ 0	0 0 0 0 0 0 0 1 1 0 1
3	R	0	1 1 1 0 1 1 1 0 1
13: 0 1 1 0 1		R	0 0 0 0 0 0 0 1 1 0 1
S: 0 0 1 0 1	+ 0	0 0 0 0 0 0 0 1 0 1 0	
CDS: 1 1 0 1 1	R	1 1 1 1 0 0 1 0 1 0	
3: 0 0 0 1 1	+ 0	0 0 0 0 1 1 0 1 0 0 0	
2: 0 0 0 1 0	R	0 0 0 0 0 1 1 0 1 0 0	
	+ 1	0 0 0 0 0 0 1 1 0 0 0	
	R	0 0 0 0 0 0 1 1 0 0 0	
22 dec 20	+ 0	0 0 0 0 0 1 1 0 0 0 1	
+ +	R	1 1 1 1 1 0 0 0 0 1	
ones:		0 0 0 1 1 0 0 0 1 0	
	complementer	0 0 0 1 1 0 0 0 1 0	

## Coma fija

$$b_3 b_2 b_1 b_0 + b_{-1} b_{-2} b_{-3} = b_3 \times 2^3 + b_2 \times 2^2 + b_1 \times 2^1 + b_0 \times 2^0 + b_{-1} \times 2^{-1} + b_{-2} \times 2^{-2} + b_{-3} \times 2^{-3}$$

$$(1101,11)_2 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} = 8 + 4 + 0.2 + 1 + \frac{1}{2} + \frac{1}{4} = 13.75)_{10}$$

$$(27,625)_{10} \Rightarrow \text{binario por partes} \quad 27 = 11001$$

$$0,625 = 0,101$$

$$27,625 = 11001,101$$

0,625

$\times 2$

1,250

$\times 2$

0,500

$\times 2$

0,000

$\times 2$

0

Hay  $\oplus$  n° periódicos  $\rightarrow$  ej

$$\begin{array}{r} 0,4 \\ \times 2 \\ \hline 0,8 \\ \times 2 \\ \hline 0,6 \\ \times 2 \\ \hline 0,6 \\ \times 2 \\ \hline 0,2 \end{array} \rightarrow \begin{array}{r} 0,2 \\ \times 2 \\ \hline 0,4 \end{array}$$

$$0,4 = 0, \overline{0110}$$

## Coma flotante

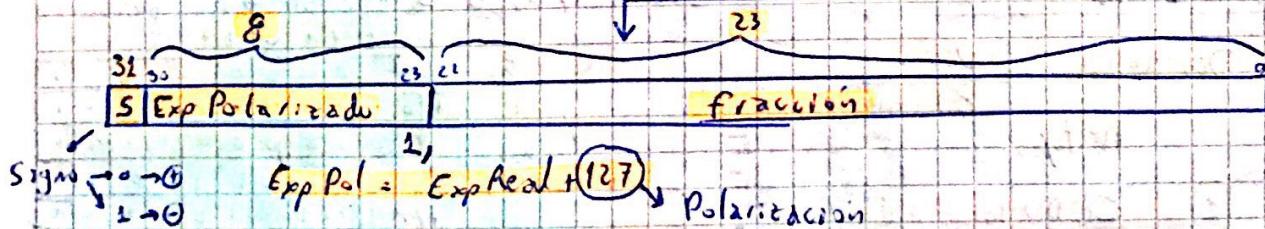
↳ Usar notación científica  $\rightarrow 1,23456 \times 10^5$   
 Normalizado

↳ Binario  $10111,011 \times 2^6$  Normalizado  $\rightarrow 1,0111011 \times 2^6$   
 Forma general: 1, fracción  $\times 2^{\text{exp}}$  Fracción

↳ Standard IEEE 754  $\rightarrow$  Precision Simple (32 bits)

Precision doble (64 bits)

Rango Dinámico: Codiente entre el n° más grande y el más representable



• Pasar a decimal  $\rightarrow$  Signo  $\oplus$  o  $\ominus$

$$\text{Exp pol.} = \dots, )_c = )_{10} \rightarrow )_{10} - 127 = \text{exp}$$

$$1, \text{fracción} \times 2^{\text{exp}} \rightarrow \underbrace{1}_{\text{Signo}} \underbrace{,}_{\text{Exp}} \underbrace{\text{fracción}}_{(\text{complemento})_{10}} \times 2^{\text{exp}} \text{ (complemento)}_{10}$$

• Pasar a binario

$$)_{10} \rightarrow \underbrace{\text{signo}}_{\text{NOTA}} \underbrace{,}_{\text{Exp}} \underbrace{\text{fracción}}_{\text{Normalizada}} \times 2^{\text{Exp}} \rightarrow 1, \text{fracción} \times 2^{\text{Exp}} \rightarrow p_0 (\text{signo}) \text{ exp.}$$

Aritmética en Punto Flotante

- Suma / Resta : ① Alinear exp (Llevar el menor al mayor) → en decimal  
 ② Sumar / Restar  
 ③ Normalizar → en decimal

- Producto : ① Multiplicar 1. fracciones y sumar exponentes  
 ② Restar 1 polarización  
 ③ Normalizar → en decimal

- Dividir : ① Dividir 1. fracciones y restar exponentes  
 ② Sumar 1 polarización  
 ③ Normalizar

Sistemas digitales (punto punto)Teorema del muestreo de Nyquist

→ Para analizar como cambia una señal a través del tiempo, en un gráfico de espacio de frecuencia

La Pueden recorrer idéntica si se muestran a doble & mitad de la frecuencia máxima

- Álgebra de Boole
- Función Booleana → cualquier combinación de variables directas o negadas
  - Variable booleana → puede tomar 0 o 1
  - Equivalencia:  $a = b \Rightarrow a$  equivale a  $b$
  - Dos operaciones
    - Suma lógica (or) → Neutral: 0
    - Producto lógico (and) → Neutral: 1
  - Elemento Negado →  $\bar{a} \rightarrow a + \bar{a} = 1$   
 $\bar{a} \cdot \bar{a} = 0$

Propiedad Asociativa:  $(a+b)+c = a+(b+c)$ ,  $(a \cdot b) \cdot c = a \cdot (b \cdot c)$

Propiedad Comutativa:  $a+b = b+a$ ,  $a \cdot b = b \cdot a$

Propiedad Distributiva:  $a \cdot (b+c) = a \cdot b + a \cdot c$ ,  $a+b \cdot c = (a+b) \cdot (a+c)$

Principio de Dualidad:
 
$$\begin{array}{l} a \cdot (b+c) = (a \cdot b) + (a \cdot c) \\ a + (b \cdot c) = (a+b) \cdot (a+c) \end{array}$$

Cubiertas  $\rightarrow a + b = 1 \quad , \quad a \cdot b = 1$

$$a + (a \cdot b) = 1$$

$$a \cdot (a + b) = 1$$

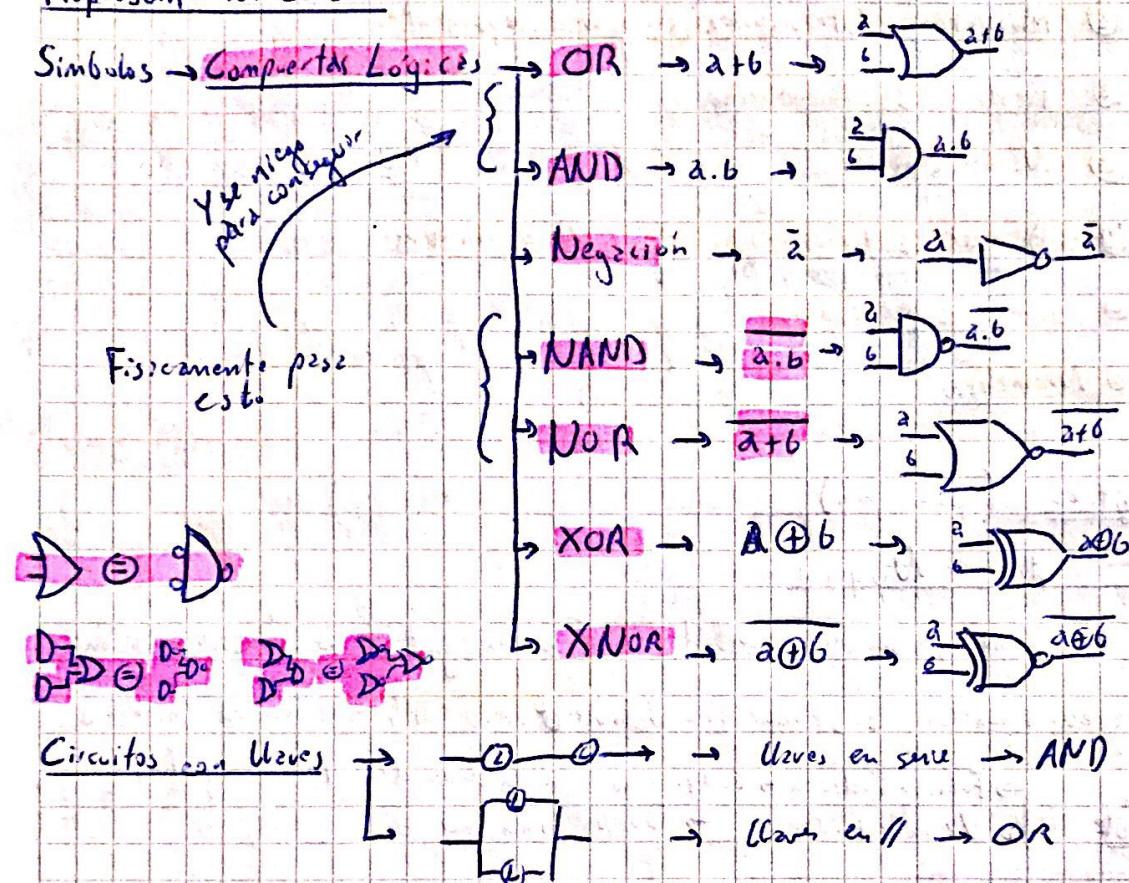
$$a \cdot (1 + b) = a$$

$$a \cdot 1 = a$$

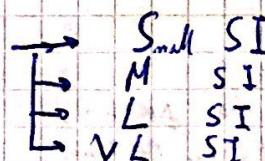
Teorema de De Morgan  $a \cdot b = \overline{a} + \overline{b}$ ,  $\overline{a + b} = \overline{a} \cdot \overline{b}$

### Representación Circuitual

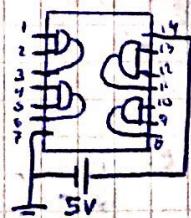
Símbolos  $\rightarrow$  Componentes Lógicas



### Cantidad de Transistores



### Circuito Integrado (chip)



Análisis de un circuito de Boole  $\rightarrow f(a, b, c)$

Representación estándar de funciones lógicas

Tabla de verdad

$a \cdot b \cdot c$	$f(a, b, c)$
0 0 0	0
0 0 1	0
0 1 0	0
0 1 1	1
1 0 0	0
1 0 1	0
1 1 0	0
1 1 1	1

NOTA

Suma de Miniterminos

Producto de todos los ver directas o negadas

 $m_3 = \text{Todos } \bar{0} \text{ excepto fila } 3 // \text{Negados que están en } '0'$ 

$$\rightarrow f(a,b,c) = m_0 + m_4 + m_7 = \Sigma(0,4,7)$$

$$= \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c + \bar{a}bc$$

Producto de Maxiterminos

Suma de todos los ver directas o negadas

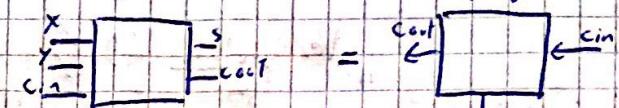
 $M_5 = \text{Todos en } '1' \text{ excepto filas } 5 // \text{Negados que están en } '1'$ 

$$\rightarrow f(a,b,c) = M_1 \cdot M_2 \cdot M_6 = \prod_M (1,2,6)$$

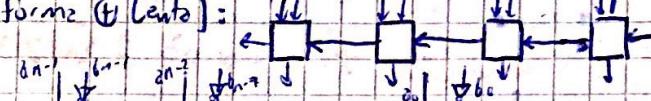
$$= (a+b+\bar{c}) \cdot (a+b+c) \cdot (\bar{a}+\bar{b}+c)$$

Sumadores Aritméticos

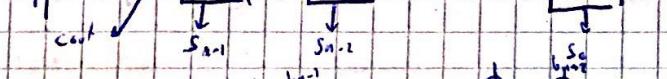
Full adder:



→ Sumador de Ripple (físico (4) lento):

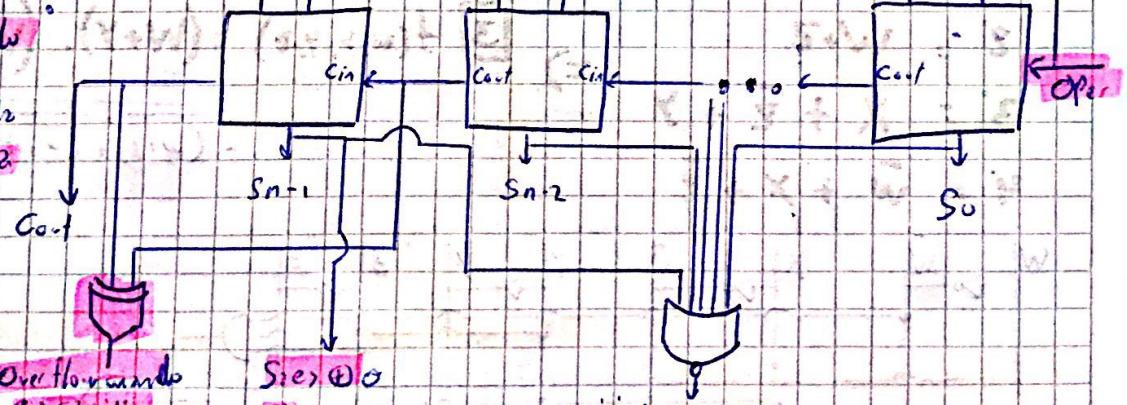


→ Restador de Ripple:

Suma y Resta en Uno Solo

Si oper = 0 → Suma

... " 1 → Resta

Unidad Aritmética Lógica

Trabaja por tabla de Verdad

Circuitos de Sistemas Digitales

Circuitos Combinacionales

Salida exclusivamente determinada por entradas

NOTAS:

Circuitos Secuenciales

Dependiendo de las entradas actuales y de lo que sucedió en el pasado

Tiempos

# Diagramas de Karnaugh

Simplificar circuitos

Diseñar circuitos mínimos en 2 niveles  $\rightarrow$  Circuitos rápidos

	$wx$	$z$	$w$	$x$	$y$	$z$	
	00	01	11	10	00	01	11
	00	00	11	11	00	01	11
	01	0*	1*	1*	1*	0	*
	11	1*	1	1	1*	0	*
	10	0*	0	1	1	*	1
		$x$					

• Implementación primaria: Agrupación de vectores más cercanos

• Las primeras columnas es vecinos de la última

[1] Míndalos implicantes primarios  $\rightarrow$  en orden

[2] Marca los que están en 1 solo implicante primario  
↳ estos van siendo:

$$[3] f(w, x, y, z) = x_1 + x_2 + x_3 \quad \bar{z}$$

iii) Ver que todos los 1's dependen de  $w$ ,  
están tapados!!! No Tapa Nada

$$[1] 1 = \bar{w} \cdot y \cdot z$$

$$2 = x \cdot y \cdot z$$

$$3 = w \cdot x$$

$$4 = w \cdot \bar{z}$$

$$[3] f(w, x, y, z) = \bar{w}y\bar{z} + w\bar{x} + w\bar{z}$$

$$= \overline{(\bar{w}y\bar{z})} + \overline{(w\bar{x})} + \overline{(w\bar{z})}$$

$$[1] 1 = w \cdot y$$

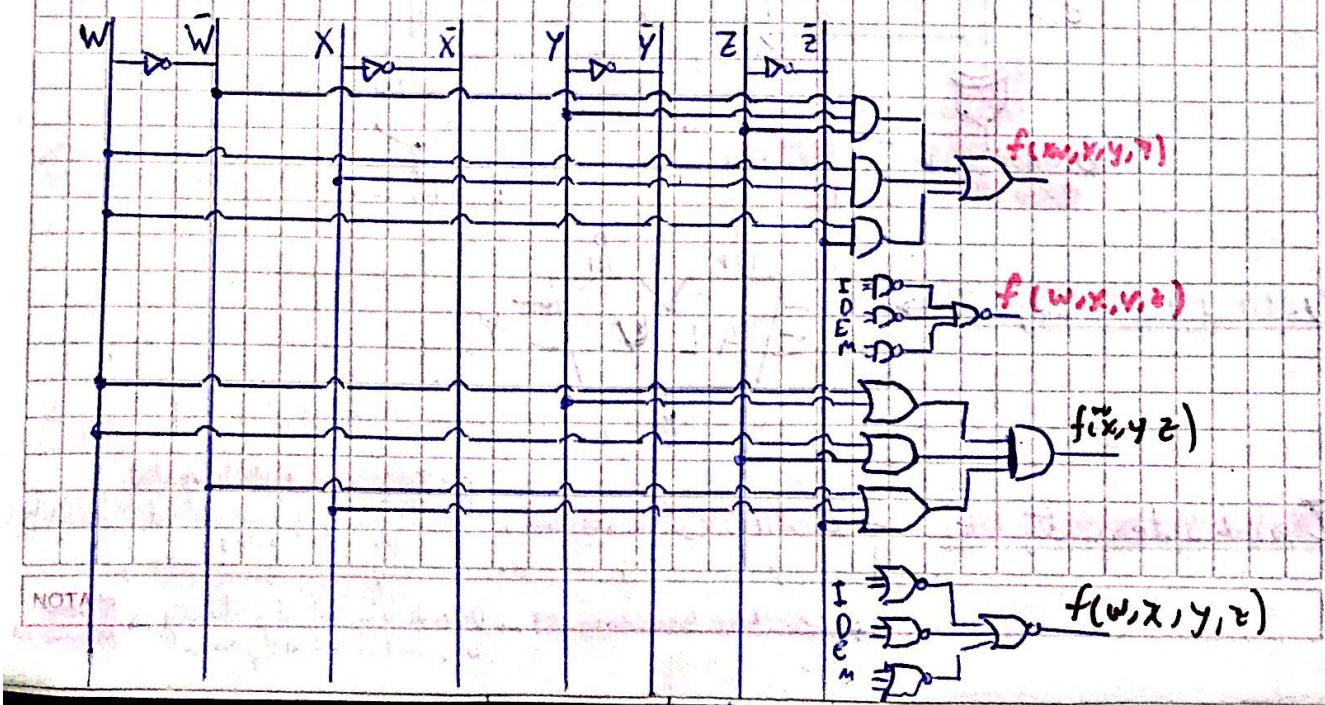
$$2 = w \cdot \bar{z}$$

$$3 = x + \bar{z} + y$$

$$4 = \bar{w} + x + \bar{z}$$

$$[3] f(w, x, y, z) = (w+y) \cdot (w+\bar{z}) \cdot (\bar{w}+x+\bar{z})$$

$$= \overline{(w+y)} + \overline{(w+\bar{z})} + \overline{(\bar{w}+x+\bar{z})}$$



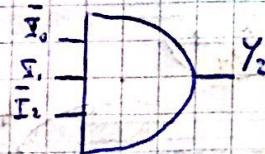
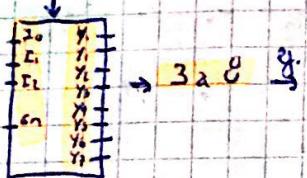
Condiciones Sin importancia (Redundancias) ( $\rightarrow$  "X")  $\rightarrow$  Garantiza q' nunca va a pasar

No mantiene si salida

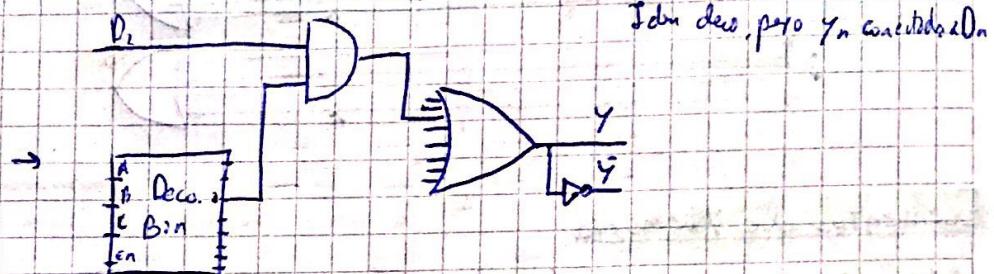
- Usa "X" para armar implementos primos
- Nunca pongo donde "X"

### Circuitos Estándar MSI

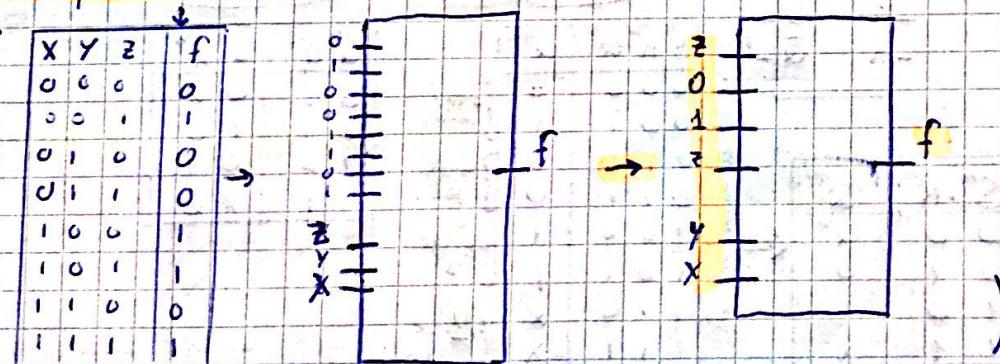
Decodificador Binario  $\rightarrow$  [Entra 1 numero en binario natural y sale por "y" el equivalente]



Multiplexor  $\rightarrow$  [Sale el bit seleccionado por los entradas selectores]  $\rightarrow$  Muestra  $D_n$ ,  $n = CBA$



(Multiplexor + eficiente)



### ROM

Memoria de Lectura, no es modificable  
Estructura de Memoria No modificable de Lectura, No volátil (no necesita energía)

$\neq$

RAM  $\rightarrow$  Random access memory, se puede escribir y es volátil (al apagar la computadora se pierde)

PRAM

EPRAM

E<sup>2</sup>PRAM

Flesh

A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>	D <sub>1</sub>	D <sub>2</sub>	D <sub>3</sub>	D <sub>4</sub>	D <sub>5</sub>	D <sub>6</sub>	D <sub>7</sub>
0	0	0	0	1	0	1	0	0	0	0
0	0	1	0	0	1	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0

A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>	D <sub>1</sub>	D <sub>2</sub>	D <sub>3</sub>	D <sub>4</sub>	D <sub>5</sub>	D <sub>6</sub>	D <sub>7</sub>
0	0	0	0	1	0	1	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0

A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>	D <sub>1</sub>	D <sub>2</sub>	D <sub>3</sub>	D <sub>4</sub>	D <sub>5</sub>	D <sub>6</sub>	D <sub>7</sub>
1	1	1	1	1	1	0	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0

A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>	D <sub>1</sub>	D <sub>2</sub>	D <sub>3</sub>	D <sub>4</sub>	D <sub>5</sub>	D <sub>6</sub>	D <sub>7</sub>
1	1	1	1	1	1	0	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0

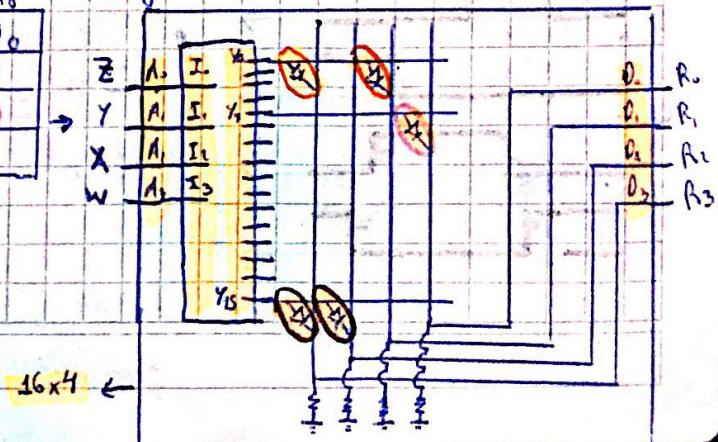
A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>	D <sub>1</sub>	D <sub>2</sub>	D <sub>3</sub>	D <sub>4</sub>	D <sub>5</sub>	D <sub>6</sub>	D <sub>7</sub>
1	1	1	1	1	1	0	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0

→ Tabla → Entradas  $\oplus$  Fijo desfaseable

(Vwlr)

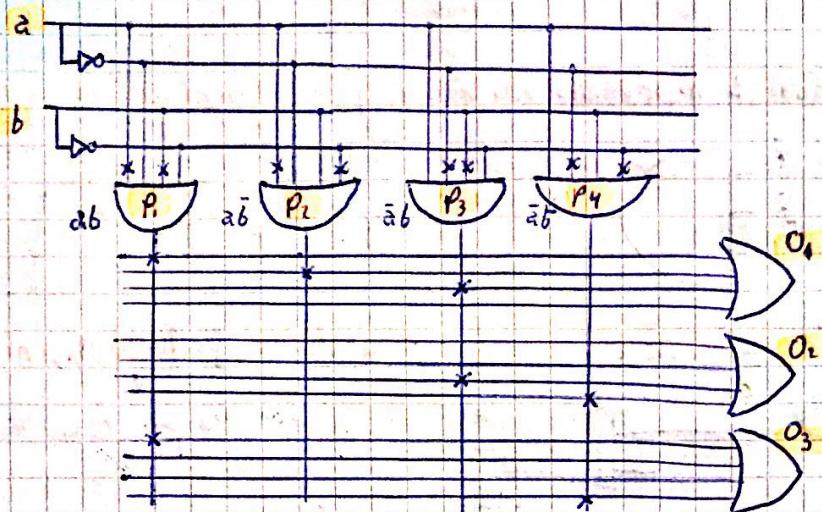
→ Usar Dados → Solo permiten tensión en dirección

NOTA  $\oplus$  q' un OR pero  $\oplus$  berata



## PLA (Programmable Logic Array)

- Se programan las conexiones,  $q'$   $\rightarrow$  Ver entre  $a \wedge p$   
 ↓  
 P entre  $a \wedge c \wedge o$
- +2 opciones  $\rightarrow$  Instrumento está todo conectado  $\rightarrow$  se cortan los  $q'$  no quiero  
 u " " " desconectado  $\rightarrow$  se conectan los  $q'$  que quiero
- No puede haber  $O$  terminos  $\neq q' \neq p$



$$O_1 = \bar{a} \cdot b + a \cdot \bar{b} + a \cdot b$$

$$O_2 = \bar{a} \cdot \bar{b} + \bar{a} \cdot b$$

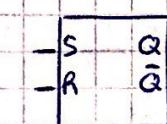
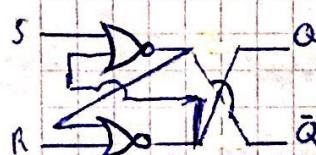
$$O_3 = a \cdot \bar{b} + \bar{a} \cdot \bar{b}$$

PLA de 2 Var

4 & terminos de producto  
3 Salidas de Funciones

## Elementos de Memoria

### LATCH Set. Reset (S-R)



$S^m$	$R^m$	$Q^m$	$\bar{Q}^m$
0	0	Q <sup>m</sup>	
0	1	0	
1	0	1	
1	1	X	

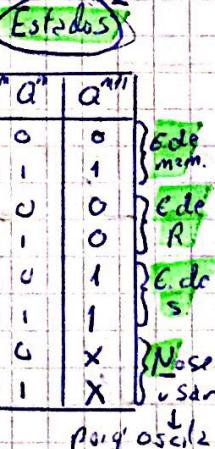
Estados

$S^m$	$R^m$	$Q^m$	$\bar{Q}^m$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	X	X

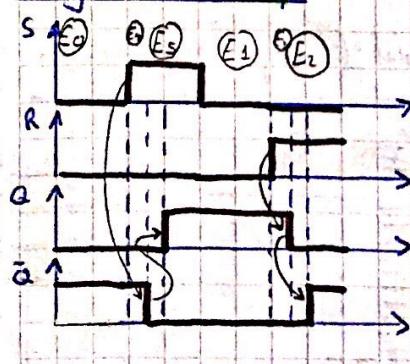
Tabla de

Transiciones

$Q^m$	$\bar{Q}^m$	$S^m$	$R^m$
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0



## Diagrama de tiempo



Set activa sobre  $Q$

Reset activa sobre  $\bar{Q}$

NOTA

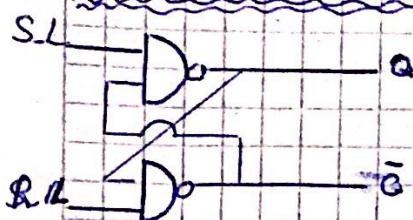
Latch S-R con entradas  $S$  y  $R$  activas

Tabla de Estados

$S-L^m$	$R-L^m$	$Q^{m+1}$
0	0	X
0	1	1
1	0	0
1	1	$Q^m$

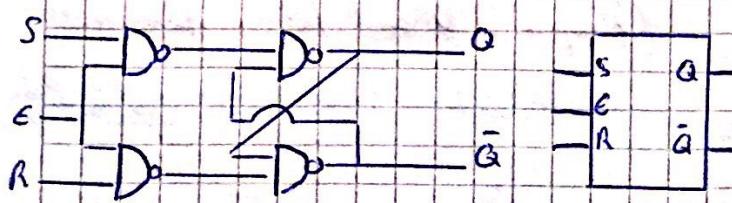
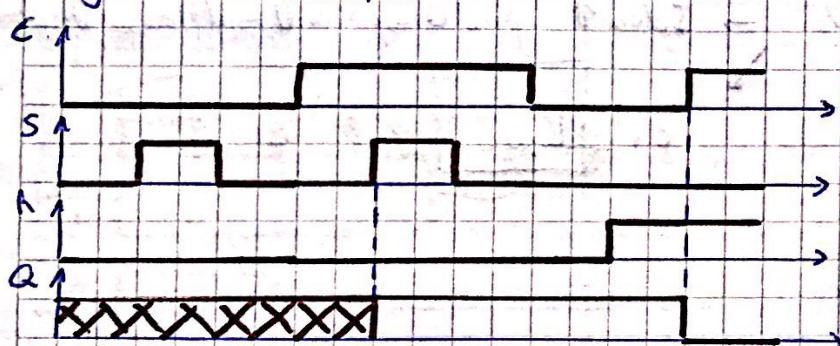
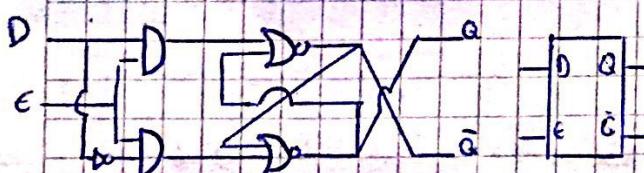
LATCH S-R con habilitación

Tabla de Estados

$E^m$	$S^m$	$R^m$	$Q^{m+1}$
0	X	X	$Q^m$
1	0	0	$Q^m$
1	0	1	0
1	1	0	1
1	1	1	X

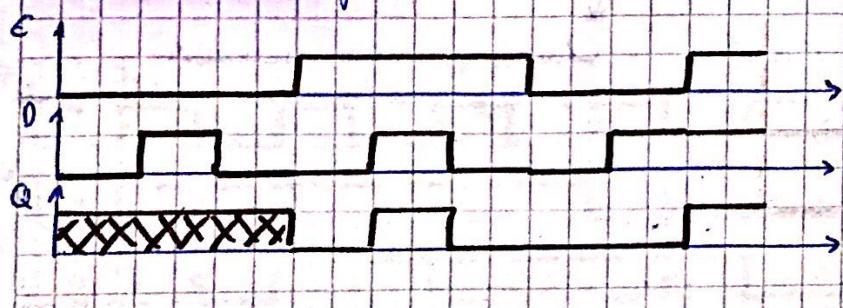
Diagrama de TiempoLATCH D con habilitación

Estados

Transiciones

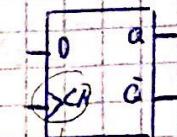
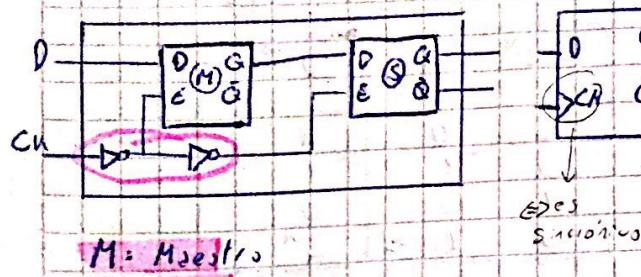
$E^m$	$D^m$	$Q^{m+1}$
0	X	$Q^m$
1	0	0
1	1	1

$E^m$	$Q^m$	$Q^{m+1}$	$D$
0	0	0	0
0	0	1	1
1	0	0	0
1	1	1	1

Diagrama de Tiempo

## Sincronizadores por señal de Reloj

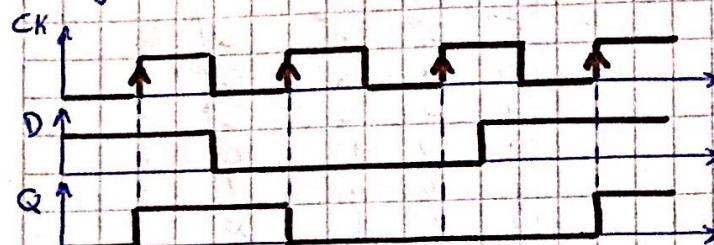
### Flip-Flop D de Flanco Ascendente (FFD)



Sincronizante

(FFD)

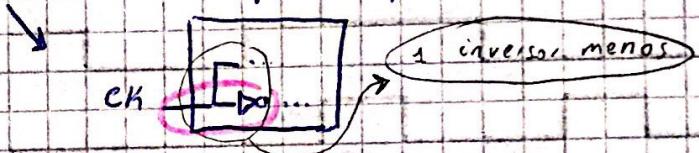
### Diagrama de Tiempo



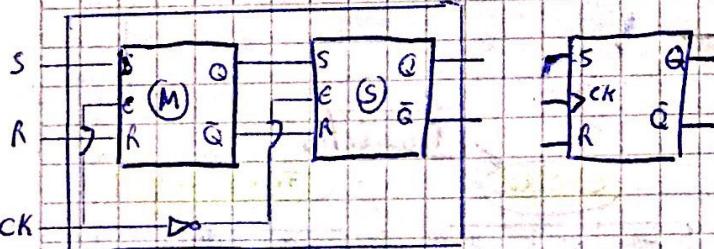
Q copia a D en el flanco Ascendente

### Flip-Flop D de Flanco Descendente

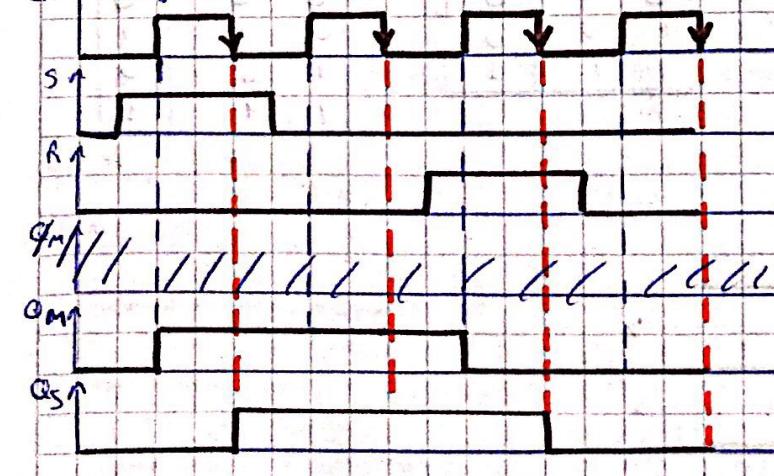
→ Idem ↑ pero Q copia a D en el flanco Descendente



### Flip-Flop S-R M-S



### Diagrama de Tiempo

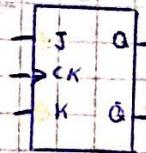
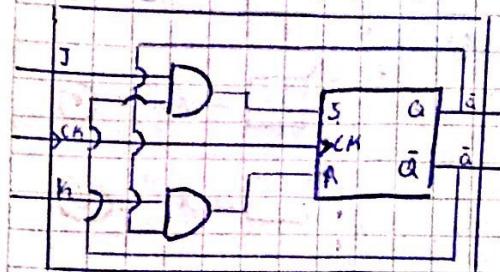


Q sensa en ascendente y cambia en descendente

Si R No cambian con CK en alto

Qm cambia en ascendente y Qs copia a maestro en descendente

FLip - FLop J - K M - S



J	K	Q <sup>n+1</sup>	(S, R)
0	0	0	M
0	1	1	X, 0
1	0	0	0, X
1	1	0	R
1	0	1	S
1	0	1	1, 0
1	1	0	Cambio
1	1	0	0, 1

Q <sup>n+1</sup>	J	K
0	0	0
0	1	1
1	0	1
1	1	0

Estados

Transiciones

Contador de Anillo  $\rightarrow$  Módulo  $m$

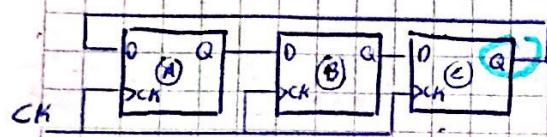


Tabla del contador

A	B	C
1	0	0
0	1	0
0	0	1

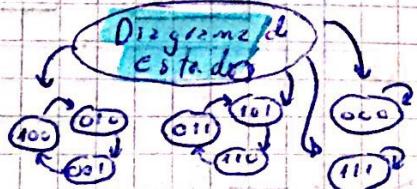
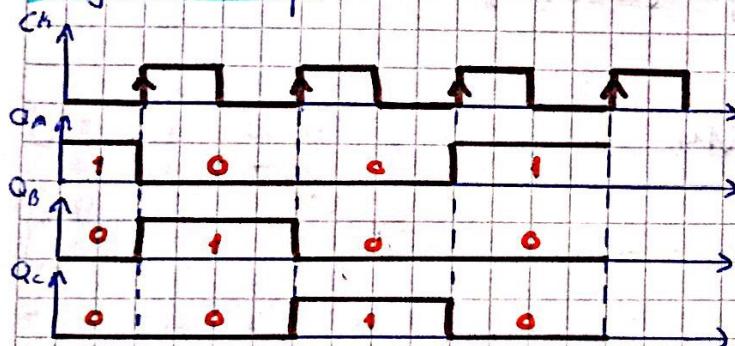


Diagrama de tiempo



Johnson  $\rightarrow$  Módulo  $2m$

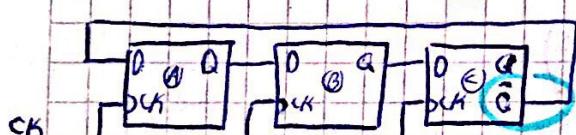


Tabla del contador

A	B	C
0	0	0
1	0	0
1	1	0
1	1	1
0	1	1
0	0	1

Diagrama de estados

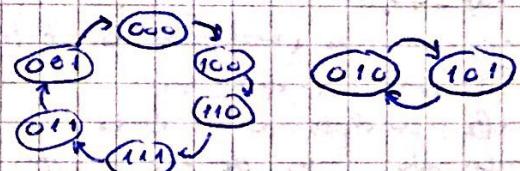
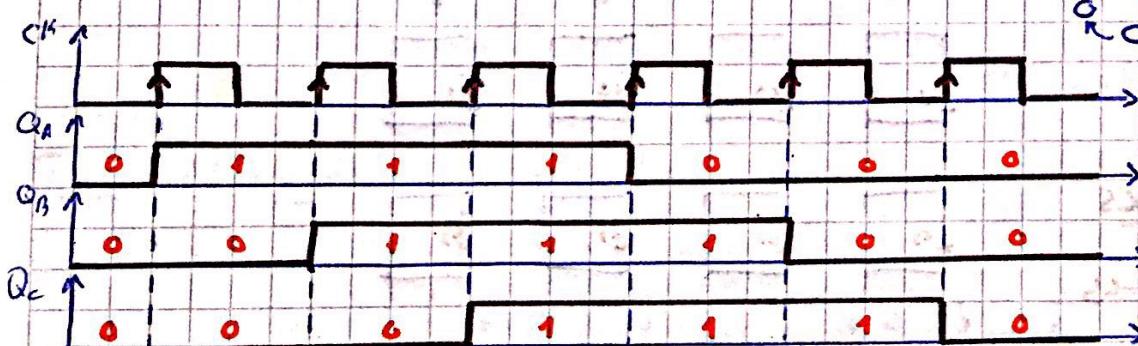


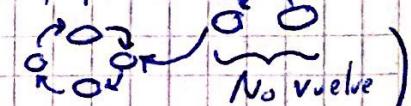
Diagrama de tiempo



NOTA

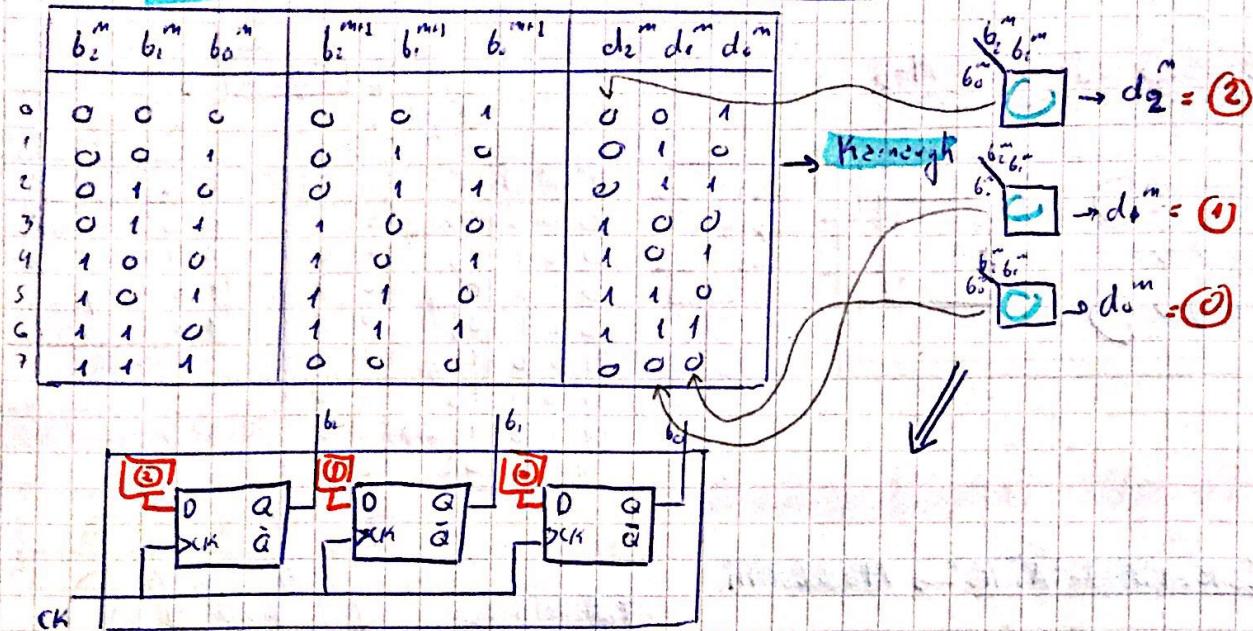
Se debe usar el modo de conteo

(Puede seguirse:



No vuelve

Contador binario de 3 bits usando FFD de flanco ↑



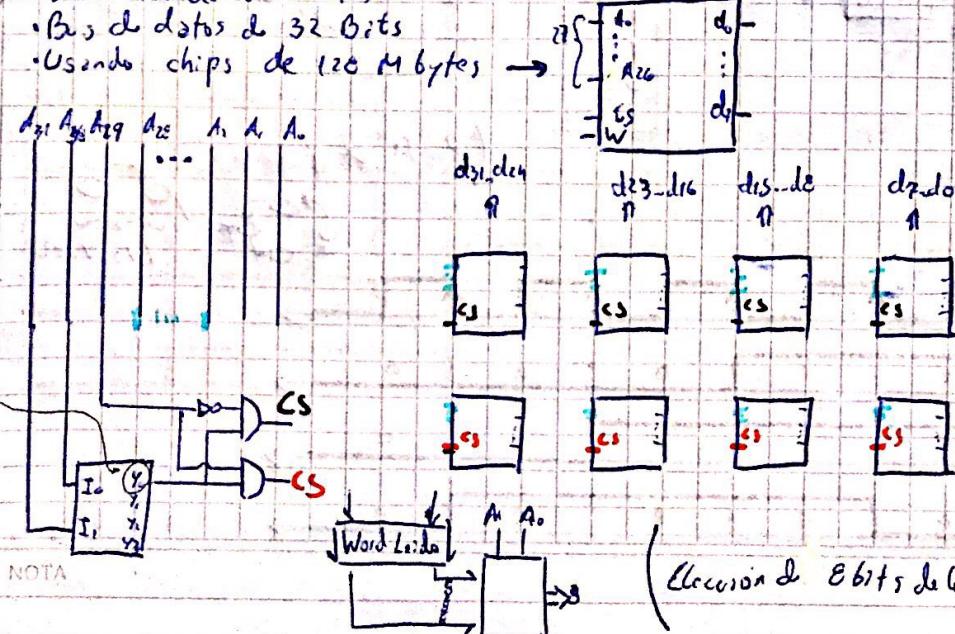
- Shift → Derecha
  - Izquierdo
  - Aritmético → Extensión de signo (Sólo der)
  - Lógico → Siempre pone 0

Todos los programas se ejecutan desde la RAM

- Memoria
  - Código
  - Datos
  - Estática
  - Dinámica
  - STACK

### Bosquejo de Memoria

- de 1 Gbyte
- para mips 32 (en el primer Gigabyte de espacio [4 Gbytes total])
- 32 líneas de direcciones



Mips  $\rightarrow$   $\text{Q} \times \text{Q}$   
Direcciones      Datos en bits/bytes

Giga  $\rightarrow 2^{30}$   
Mega  $\rightarrow 2^{20}$   
Kilo  $\rightarrow 2^{10}$

SUBIU: (\$4, \$5, 33)

SELrs, REGout, Yin

IR\_immedout, ALUsub, Zin

Zout, SELrd, REGin, JUM(start)

SLT: (\$6, \$7, \$8)

SELrs, REGout, Yin

SELrt, REGout, ALUslt, Zin

Zout, SELrd, REGin, JUMP(start)

SW \$s, offset(\$t):

SELrt, REGout, Yin

IR\_offsetout, ALUadd, Zin

Zout, MARin

SELrs, REGout, MDRin, MEMwrite, Jump(start)

LW \$s, offset(\$t) : #Carga ("load word") en el registro \$s la palabra contenida  
en la dirección de memoria almacenada en el registro \$t

SELrt, REGout, Yin

IR\_offsetout, ALUadd, Zin

Zout, MARin, MEMread

UNTIL mfc

MDRout, SELrs, REGin, Jump(start)

BNE \$s, \$t, Label : # Si el contenido del Registro \$s es  $\neq$  del de \$t, entonces salta a la dirección indicada por la etiqueta Label

SELrs, REGout, Yin

SELrt, REGout, ALUxor, Zin

IR\_offsetout, PC:nsf()

NOTA

