

# Tema 3

## Circuitos lógicos combinacionales

Juan J. Navarro  
[juanjo@ac.upc.edu](mailto:juanjo@ac.upc.edu)  
Tel: 93 4016983  
Despacho D6-205

1

Copyright © Juan J. Navarro. Departament d'Arquitectura de Computadors. Universitat Politècnica de Catalunya.

## 3. Circuitos Lógicos Combinacionales

- ♦ 3.1. Introducción
- ♦ 3.2. Análisis de circuitos combinacionales
- ♦ 3.3. Síntesis de circuitos combinacionales
- ♦ 3.4. Completando el modelo de un circuito combinacional.  
Tiempos de propagación
- ♦ Apéndice. Álgebra de conmutación.  
Circuitos equivalentes. Minimización.

2

Copyright © Juan J. Navarro. Departament d'Arquitectura de Computadors. Universitat Politècnica de Catalunya.

## 3.1. Introducción

- ♦ 3.1.1. Circuito Lógico Combinacional (CLC)
- ♦ 3.1.2. De la descripción a la tabla de verdad
- ♦ 3.1.3. Puertas Lógicas
- ♦ 3.1.4. Interconectando CLCs
- ♦ Actividad #3.1(a)

3

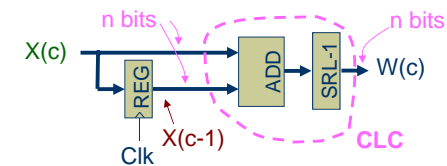
Copyright © Juan J. Navarro. Departament d'Arquitectura de Computadors. Universitat Politècnica de Catalunya.

## 3.1.1. Circuito Lógico Combinacional

### Motivación

En un computador, los CLC son los encargados de procesar (transformar) las señales binarias (información digital)

Ejemplo: parte del PPE que realiza un filtro paso bajo

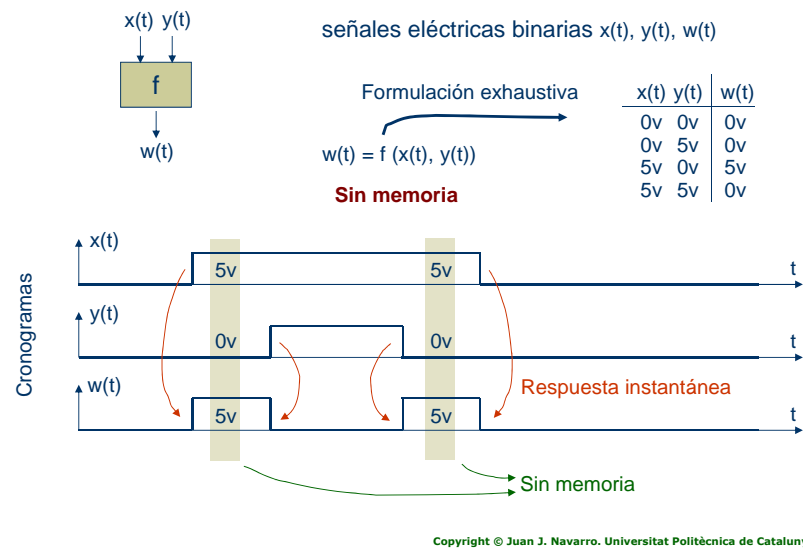


$$\text{Procesado: } W(c)_u = \text{int}( X(c)_u + X(c-1)_u ) / 2$$

4

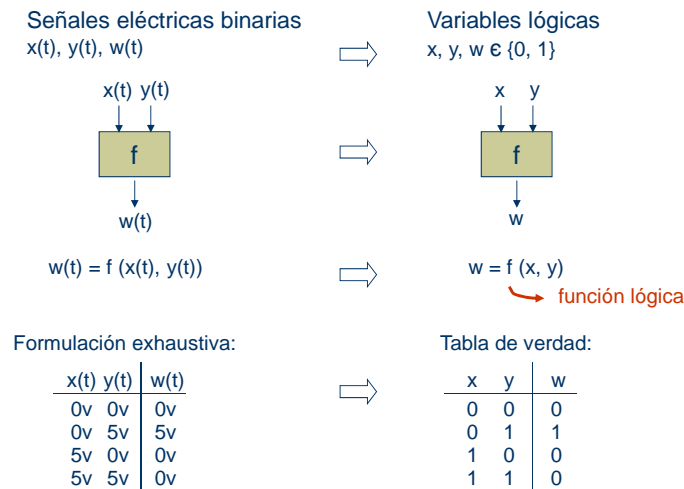
Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

# Definición. CLC: Sin memoria e Instantáneo



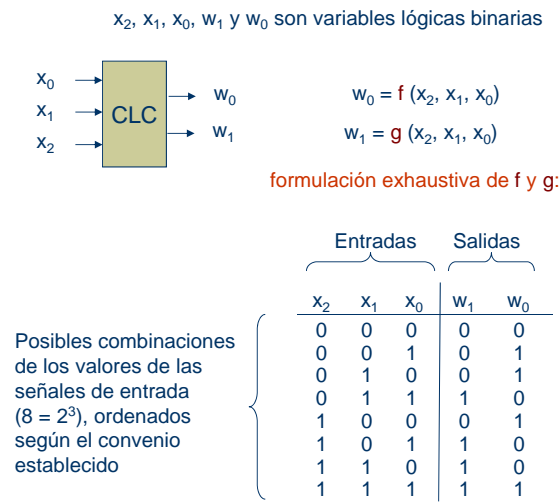
5

# Modelado matemático



6

# Tabla de verdad



7

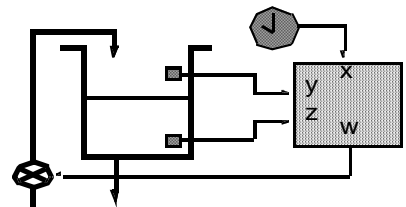
# 3.1.2. De la descripción a la TV

**Ejemplo: Sistema de llenado automático de un depósito de agua**

$x$ : 0 = noche, 1 = día

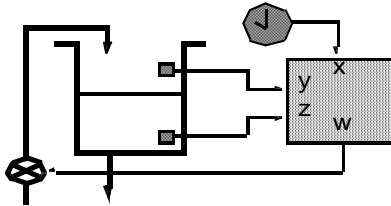
$y, z$ : 0 = sensor seco, 1 = sensor mojado

$w$ : 0 = no bombea, 1 = bombea



8

## Del texto a la Tabla de verdad



a)	b)	c)	d)
x y z   w	x y z   w	x y z   w	x y z   w
0 0 0	0 0 0   1	0 0 0   1	0 0 0   1
0 0 1	0 0 1	0 0 1   1	0 0 1   1
0 1 0	0 1 0	0 1 0	0 1 0   x
0 1 1	0 1 1   0	0 1 1   0	0 1 1   0
1 0 0	1 0 0   1	1 0 0   1	1 0 0   1
1 0 1	1 0 1	1 0 1   0	1 0 1   0
1 1 0	1 1 0	1 1 0   0	1 1 0   x
1 1 1	1 1 1   0	1 1 1   0	1 1 1   0

## 3.1.3. Puertas Lógicas

**Puertas lógicas:** los circuitos combinacionales más simples  
(de momento 1 o 2 entradas y 1 salida)

¿Cuántas puertas/funciones lógicas distintas hay con una sola entrada?

4 posibles funciones lógicas de 1 entrada

x	$g_3$	$g_2$	$g_1$	$g_0$
0	1	1	0	0
1	1	0	1	0

$x \rightarrow g_i \rightarrow w$   
 $w = g_i(x)$

$x \rightarrow \neg \rightarrow w$   
**Not**  
 $w = !x$

10

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Puertas lógicas de dos entradas

¿Cuántas puertas/funciones lógicas distintas hay con dos entradas?

16 posibles funciones lógicas de 2 entradas

x y	$f_{15}$	$f_{14}$	$f_{13}$	$f_{12}$	$f_{11}$	$f_{10}$	$f_9$	$f_8$	$f_7$	$f_6$	$f_5$	$f_4$	$f_3$	$f_2$	$f_1$	$f_0$
0 0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
0 1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0
1 0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0
1 1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0

$w = f_i(x, y)$

**Or**  
 $w = x + y$




**And**  
 $w = x \cdot y$

Cualquier circuito combinacional se puede implementar interconectando adecuadamente puertas **Not**, **And** y **Or** (como veremos en este capítulo)

11

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Puertas Lógicas Not, And y Or

Nombre	Símbolo	Expresión lógica	Tabla de verdad	Explicación															
Not		$w = !x$ ( $w = \bar{x}$ ) ( $w = x'$ )	<table><tr><th>x</th><th>w</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	x	w	0	1	1	0	La salida es la <b>inversa</b> (o el complemento) de la entrada									
x	w																		
0	1																		
1	0																		
And-2		$w = x \cdot y$ ( $w = x \cdot y$ ) ( $w = x \&\& y$ )	<table><tr><th>x</th><th>y</th><th>w</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	x	y	w	0	0	0	0	1	0	1	0	0	1	1	1	La salida vale <b>1</b> cuando <b>todas</b> las entradas valen <b>1</b> (La salida vale <b>0</b> cuando <b>alguna</b> entrada vale 0)
x	y	w																	
0	0	0																	
0	1	0																	
1	0	0																	
1	1	1																	
Or-2		$w = x + y$ ( $w = x + y$ ) ( $w = x    y$ )	<table><tr><th>x</th><th>y</th><th>w</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	x	y	w	0	0	0	0	1	1	1	0	1	1	1	1	La salida vale 1 cuando <b>alguna</b> entrada vale 1 (La salida vale <b>0</b> cuando <b>todas</b> las entradas valen 0)
x	y	w																	
0	0	0																	
0	1	1																	
1	0	1																	
1	1	1																	

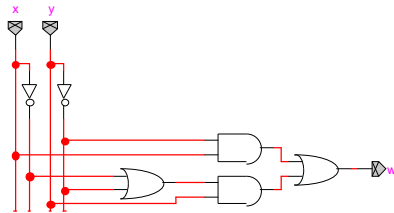
12

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

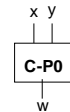
### 3.1.4. Interconectando CLCs

#### Diseño modular multinivel:

Implementación de dispositivos combinacionales grandes interconectando dispositivos pequeños



Circuito interno del dispositivo



Símbolo del dispositivo

13

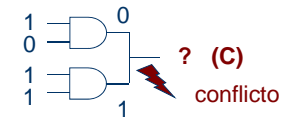
Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

### Reglas de interconexión

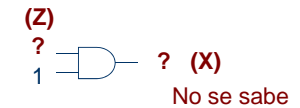
#### Implementación de CLC grandes interconectando CLCs pequeños

Reglas de interconexión:

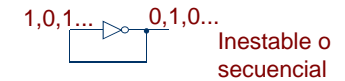
- No conectar directamente salidas entre sí



- No dejar entradas sueltas (sin valor)  
(Alta impedancia, Z, el valor lógico depende de la tecnología)



- No ciclos (inestabilidad o memoria –secuencial)

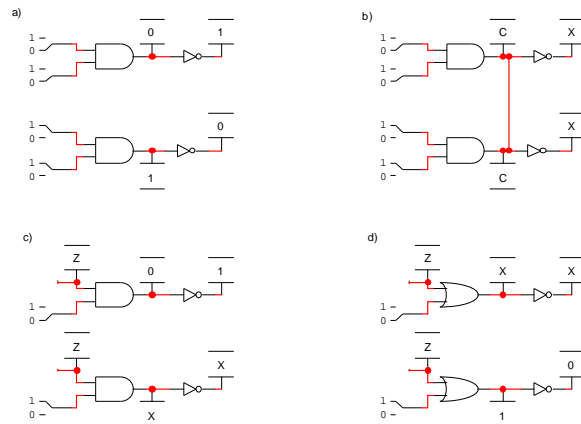


14

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

### Reglas de interconexión

#### Simulación de circuitos con Z, X o C



15

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

### Actividad #3.1(a)

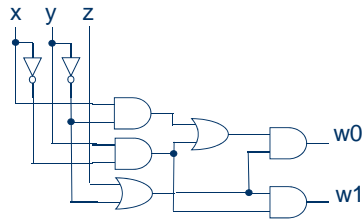
- Individual
- En una hoja suelta (para toda la actividad #3.1 (a, b, c, d y e). Nombre. Fecha. En el primer 1/3 de la hoja:
- Dibuja el esquema lógico de un CLC que tiene 3 entradas (x, y, z) y dos salidas (w1, w0), usando de 6 a 8 puertas Not, And y Or interconectadas como tu desees, pero siguiendo las reglas anteriores.

16

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Ejemplo de solución de la Actividad #3.1(a)

Un posible esquema:



17

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## 3.2. Análisis de circuitos combinacionales

- 3.2.1. Análisis de un CLC: T.V. por filas Actividad #3.1(b)
- 3.2.2. Análisis de un CLC: T.V. por columnas Actividad #3.1(c)

18

Copyright © Juan J. Navarro. Departament d'Arquitectura de Computadors. Universitat Politècnica de Catalunya.

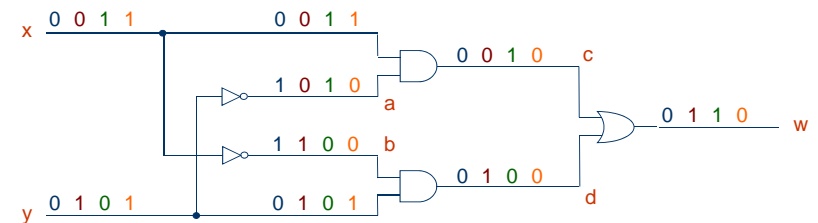
### 3.2.1. Análisis de un CLC, T.V. por filas

- Dado el esquema de un circuito lógico combinacional encontrar la tabla de verdad que especifica su funcionamiento.
  - Circuito con pocas entradas/salidas (tabla de verdad con pocas filas/columnas)
  - El circuito esta formado por puertas interconectadas. (más adelante ejemplos con circuitos formados por bloques combinacionales interconectados).

19

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

### Ejemplo de análisis, T.V. por filas



x	y	w
0	0	0
0	1	1
1	0	1
1	1	0

**Xor**  
La salida vale 1 cuando las dos entradas tienen distinto valor  
(La salida vale 1 cuando hay un número impar de unos en las entradas)

20

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

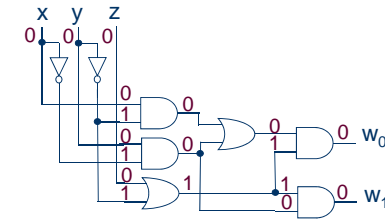
## Actividad #3.1(b)

- ♦ Individual
- ♦ Intercambiad la hoja de la actividad anterior con vuestro compañero.
- ♦ Poned vuestro nombre. En el segundo 1/3 de la hoja:
- ♦ Obtén la tabla de verdad del CLC que dibujo tu compañero por el procedimiento anterior (caso por caso, T.V. por filas).

21

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Ejemplo de solución de la Actividad #3.1(b)



x	y	z	w <sub>1</sub>	w <sub>0</sub>
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	0
1	1	1	0	0

22

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

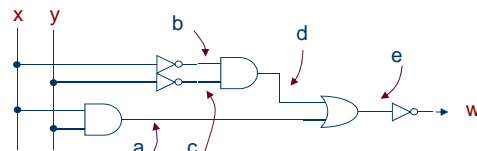
## 3.2.2. Análisis de un CLC, T.V. por columnas

Obtención de la TV de las variables lógicas intermedias usando la TV de la Not, And y Or

x	!x
0	1
1	0

x	y	x·y
0	0	0
0	1	0
1	0	0
1	1	1

x	y	x+y
0	0	0
0	1	1
1	0	1
1	1	1

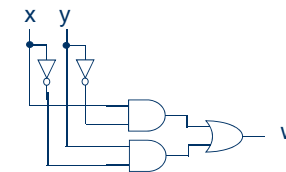


x	y	x·y	!x	!y	b·c	a+d	!e
		a	b	c	d	e	w
0	0	0	1	1	1	1	0
0	1	0	1	0	0	0	1
1	0	0	0	1	0	0	1
1	1	1	0	0	0	1	0

23

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

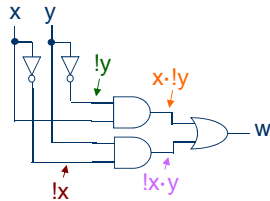
## Ejemplo de análisis de un CLC por columnas. Enunciado



24

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Ejemplo de análisis de un CLC. Solución



Solución:

x	y	$\neg y$	$\neg x$	$x \cdot \neg y$	$\neg x \cdot y$	w
0	0	1	1	0	0	0
0	1	0	1	0	1	1
1	0	1	0	1	0	1
1	1	0	0	0	0	0

25

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

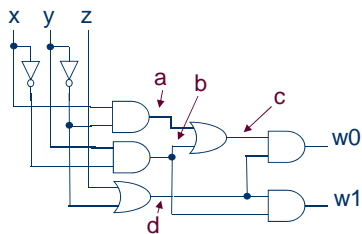
## Actividad #3.1(c)

- ♦ Individual
- ♦ Pasad la hoja de la actividad anterior a vuestro compañero.
- ♦ En el tercer 1/3 de la hoja (pon tu nombre):
- ♦ Obtén la tabla de verdad del CLC que dibujaste ahora por columnas.

26

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Solución de la actividad #3.1(c)



T.V. de cada variable lógica intermedia  
(salida de cada puerta)

x	y	z	$x \cdot y$ a	$x \cdot z$ b	$y \cdot z$ c	$a + b + c$ d	d · b w <sub>1</sub>	c · d w <sub>0</sub>
0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0
0	1	1	0	0	1	1	0	0
1	0	0	0	0	0	0	0	0
1	0	1	0	1	0	1	0	0
1	1	0	1	0	0	1	0	0
1	1	1	1	1	1	1	1	1

27

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## 3.3. Síntesis de circuitos

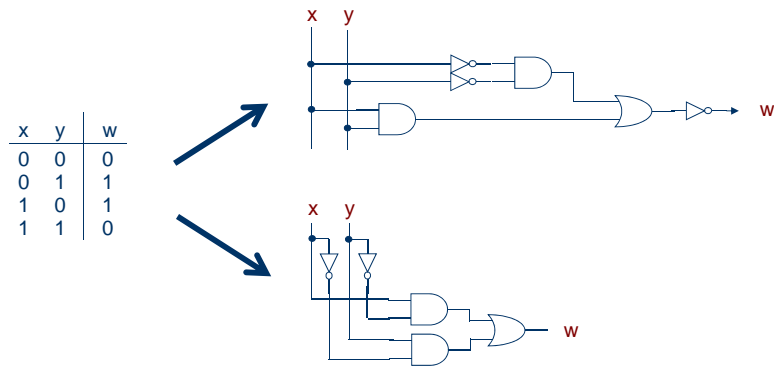
- ♦ 3.3.1. Síntesis. Definición. Dificultad.
- ♦ 3.3.2. Casos sencillos en suma de minterms
- ♦ 3.3.3. Algunas propiedades de la And y Or. Puertas n entradas.
- ♦ 3.3.4. Síntesis en suma de minterms. Caso general  
Actividad #3.1.(d)  
Del problema al circuito vía TV
- ♦ 3.3.5. Síntesis con decodificador y puertas Or
- ♦ 3.3.6. Síntesis con ROM  
Actividad #3.1 (e) y Actividad #3.2

28

Copyright © Juan J. Navarro. Departament d'Arquitectura de Computadors. Universitat Politècnica de Catalunya.

### 3.3.1. Síntesis. Definición. Dificultad

**Síntesis:** de la TV al esquema lógico  
(de momento con Not, And y Or)

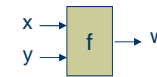


29

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

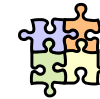
### Dificultad

Síntesis: de la TV al esquema lógico con Not, And y Or



La salida vale 1 cuando  
las dos entradas son iguales

x	y	w
0	0	1
0	1	0
1	0	0
1	1	1



¡Sintetizar el circuito con puertas Not, And y Or!  
⇒ muy difícil sin herramientas



30

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

### 3.3.2. Casos sencillos en suma de minterms

x	y	w
0	0	0
0	1	0
1	0	0
1	1	1



$$x \text{ AND } y \rightarrow m_3(x,y) = x \cdot y$$



31

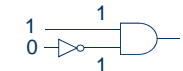
Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

### Caso del minterm 2 de dos variables

x	y	w
0	0	0
0	1	0
1	0	1
1	1	0



$$x \text{ AND } \neg y \rightarrow m_2(x,y) = x \cdot \neg y$$



x	y	$\neg y$	$x \cdot \neg y$
0	0	1	0
0	1	0	0
1	0	1	1
1	1	0	0

32

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.



## Todos los minterms de dos variables

x	y	$\neg x$	$\neg y$	$m_3 = x \cdot y$	$m_2 = x \cdot \neg y$	$m_1 = \neg x \cdot y$	$m_0 = \neg x \cdot \neg y$
0	0	1	1	0	0	0	1
0	1	1	0	0	0	1	0
1	0	0	1	0	1	0	0
1	1	0	0	1	0	0	0

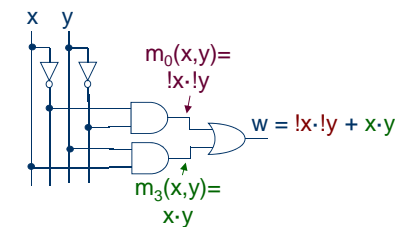
33

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## La suma de minterms

x	y	w	$m_3$	$m_0$	$w = m_3 + m_0$
0	0	1	0	1	1
0	1	0	0	0	0
1	0	0	0	0	0
1	1	1	1	0	1

Síntesis  
en suma de  
minterms



34

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

### 3.3.3. Algunas propiedades de la And y Or. Puertas de n entradas.

- Comutatividad:  $x \cdot y = y \cdot x$   
( $0 \cdot 1 = 1 \cdot 0 = 0$ )

- Asociatividad:  $(x \cdot y) \cdot z = x \cdot (y \cdot z)$

x	y	z	$x \cdot y$	$(x \cdot y) \cdot z$	$y \cdot z$	$x \cdot (y \cdot z)$
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
0	1	1	0	0	1	0
1	0	0	0	0	0	0
1	0	1	0	0	0	0
1	1	0	1	0	0	0
1	1	1	1	1	1	1

x	y	$x \cdot y$
0	0	0
0	1	0
1	0	0
1	1	1

Idem para la Or

35

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Puertas And y Or de n entradas

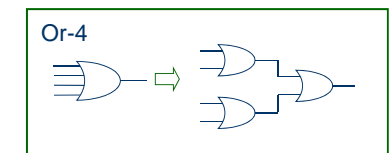
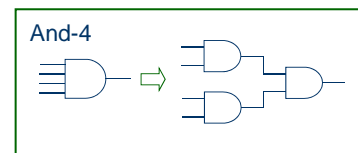
- Comutatividad y asociatividad de la And y de la Or:

$$(x \cdot y) \cdot z = x \cdot (y \cdot z) = x \cdot y \cdot z = x \cdot z \cdot y = y \cdot x \cdot z = y \cdot z \cdot x = z \cdot x \cdot y = z \cdot y \cdot x$$

⇒ Puerta And de n entradas   $w = x \cdot y \cdot z$  And-3

$$(x + y) + z = x + (y + z) = x + y + z = x + z + y = \dots = z + y + x$$

⇒ Puerta Or de n entradas   $w = x + y + z$  Or-3



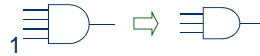
36

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Puertas de n-1 entradas a partir de puertas de n entradas

### ♦ Elemento neutro del producto lógico (And):

- $x \cdot 1 = x \Rightarrow x \cdot y \cdot z \cdot 1 = x \cdot y \cdot z$   
 $\Rightarrow$  Puerta **And-3** a partir de una **And-4**



### ♦ Elemento neutro de la suma lógica (Or):

- $x + 0 = x \Rightarrow x + y + z + 0 = x + y + z$   
 $\Rightarrow$  Puerta **Or-3** a partir de una **Or-4**



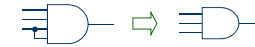
37

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Puertas de n-1 entradas a partir de puertas de n entradas

### ♦ Idempotencia del producto lógico (And):

- $x \cdot x = x \Rightarrow x \cdot y \cdot z \cdot z = x \cdot y \cdot z$   
 $\Rightarrow$  Puerta **And-3** a partir de una **And-4**



### ♦ Idempotencia de la suma lógica (Or):

- $x + x = x \Rightarrow x + y + z + z = x + y + z$   
 $\Rightarrow$  Puerta **Or-3** a partir de una **Or-4**



38

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

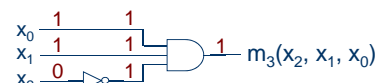
## 3.3.4. Síntesis en suma de minterms

### ♦ Síntesis de funciones **minterm**

- Una función minterm vale 1 sólo para una combinación concreta de las entradas, para el resto de combinaciones vale 0
- Función minterm  $i$  de las variables  $x_2, x_1, x_0$ :  $m_i(x_2, x_1, x_0)$   
La salida vale 1 para la fila  $i$  de la tabla de verdad

$x_2$	$x_1$	$x_0$	$m_0$	$m_1$	$m_2$	$m_3$	$m_4$	$m_5$	$m_6$	$m_7$
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

$$\Rightarrow m_3(x_2, x_1, x_0) = !x_2 \cdot x_1 \cdot x_0$$



39

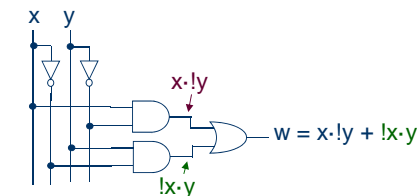
Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Síntesis en suma de minterms

### Ejemplo:

$x$	$y$	$w$
0	0	0
0	1	1
1	0	1
1	1	0

$x$	$y$	$m_2$	$m_1$	$w$
0	0	0	0	0
0	1	0	1	1
1	0	1	0	1
1	1	0	0	0



$$w(x,y) = \sum_m(1,2) = m_1(x,y) + m_2(x,y) = !x \cdot y + x \cdot !y$$

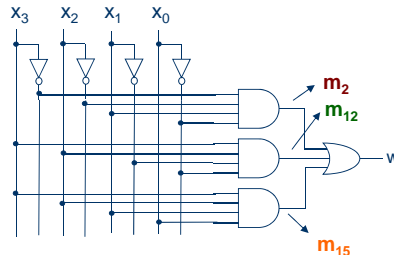
40

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Ejemplo con 4 variables

$$w(x_3, x_2, x_1, x_0) = \sum m(2, 12, 15)$$

$x_3$	$x_2$	$x_1$	$x_0$	$w$
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1



41

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Actividad #3.1(d)

- ♦ Individual
- ♦ Intercambiad la hoja de la actividad #3.1 con vuestro compañero.
- ♦ En la parte de atrás de la hoja (segundo 1/3):
- ♦ A partir de la tabla de verdad de las salidas  $w_0$  y  $w_1$  de la actividad #3.1(b) o (c) (si las T.V. de los dos apartados no coinciden elige la que consideres correcta), dibuja el esquema lógico del circuito que se obtiene directamente de las expresiones en suma de minterms de las salidas.

42

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Solución de la actividad #3.1(d)

$x$	$y$	$z$	$w_1$	$w_0$
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	0
1	1	1	0	0

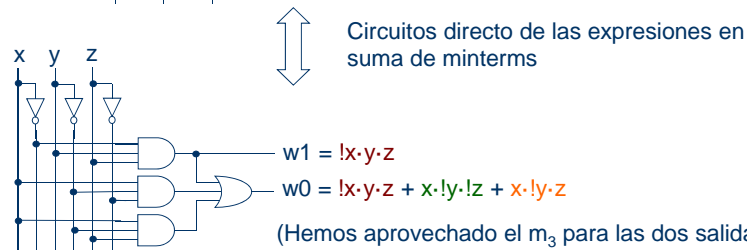
$$m_3(x, y, z) = !x \cdot y \cdot z$$

$$m_4(x, y, z) = x \cdot !y \cdot !z$$

$$m_5(x, y, z) = x \cdot !y \cdot z$$

$$w_0 = !x \cdot y \cdot z + x \cdot !y \cdot !z + x \cdot !y \cdot z$$

$$w_1 = !x \cdot y \cdot z$$



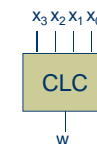
43

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Del problema al circuito vía TV

### Problema: detector de códigos

- ♦ Dibujar el esquema lógico interno, de la implementación directa en suma de minterms, de un CLC que detecta, activando la salida  $w$ , cuando en la entrada de 4 bits se encuentra codificado en binario el número natural 3, 4 o 13.



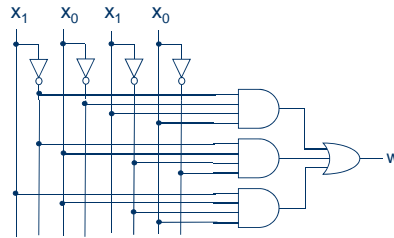
44

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Solución detector de códigos

$$w(x_3, x_2, x_1, x_0) = \sum m(3, 4, 13)$$

$x_3$	$x_2$	$x_1$	$x_0$	$w$
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0



45

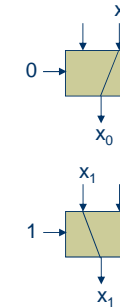
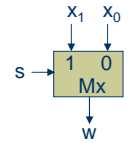
Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Ejemplo. Multiplexor. Funcionalidad

### ♦ Paso 1. De la descripción funcional a la TV

#### a) Descripción funcional:

if ( $s=0$ )  $w=x_0$  else  $w=x_1$ ;



#### b) Tabla de verdad

s	$x_1$	$x_0$	w
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

46

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

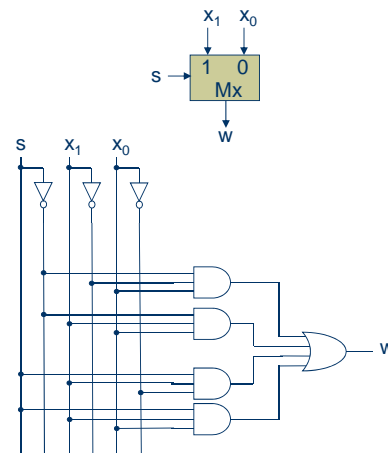
## Ejemplo. Multiplexor. Síntesis

### ♦ Paso 2. Síntesis

De la TV a la expresión/circuito en suma de minterms

s	$x_1$	$x_0$	w
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

$$\begin{aligned} m_1 &= !s \cdot !x_1 \cdot x_0 \Rightarrow \\ m_3 &= !s \cdot x_1 \cdot x_0 \Rightarrow \\ m_6 &= s \cdot x_1 \cdot !x_0 \Rightarrow \\ m_7 &= s \cdot x_1 \cdot x_0 \Rightarrow \end{aligned}$$



$$w(s, x_1, x_0) = m_1 + m_3 + m_6 + m_7 = !s \cdot !x_1 \cdot x_0 + !s \cdot x_1 \cdot x_0 + s \cdot x_1 \cdot !x_0 + s \cdot x_1 \cdot x_0$$

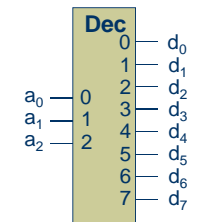
47

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## 3.3.5. Síntesis con decodificador y puertas Or

Un **decodificador** de n entradas y  $2^n$  salidas (Dec-n- $2^n$ ) implementa cada uno de los  $2^n$  **minterms** de n entradas

$a_2$	$a_1$	$a_0$	$d_0$	$d_1$	$d_2$	$d_3$	$d_4$	$d_5$	$d_6$	$d_7$
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

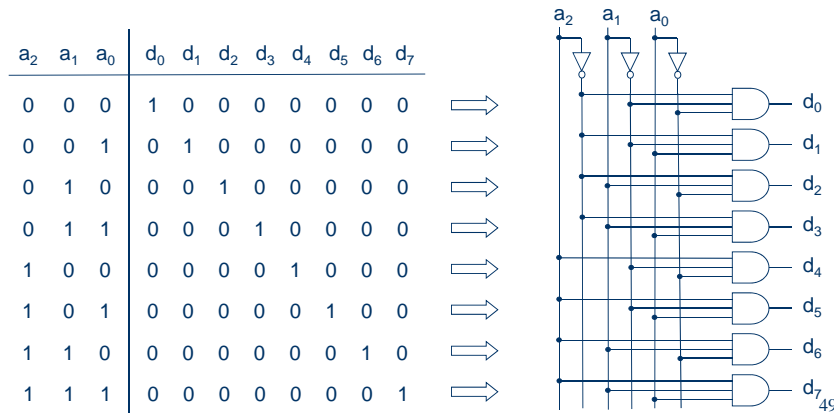


48

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Síntesis con decodificador y puertas Or

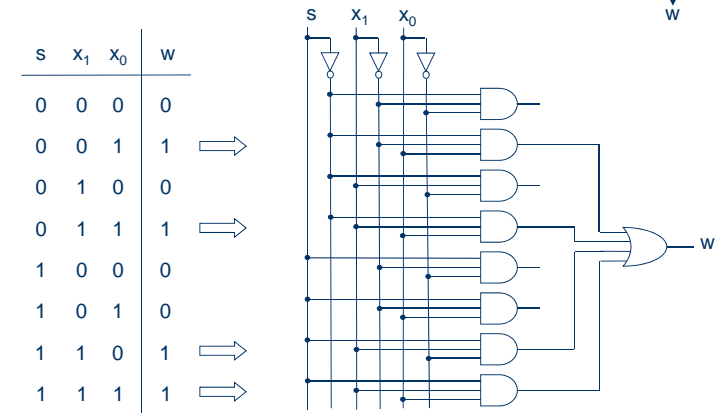
Un **decodificador** de  $n$  entradas y  $2^n$  salidas (**Dec- $n-2^n$** ) implementa cada uno de los  $2^n$  **minterms** de  $n$  entradas



Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Síntesis con decodificador y puertas Or

- Ejemplo: Multiplexor  
if  $s=0$  then  $w=x_0$  else  $w=x_1$

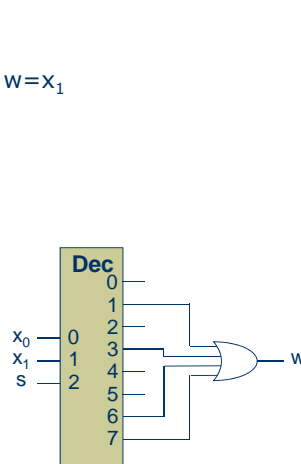


Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Síntesis con decodificador y puertas Or

- Ejemplo: Multiplexor  
if  $s=0$  then  $w=x_0$  else  $w=x_1$

s	x <sub>1</sub>	x <sub>0</sub>	w
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1



51

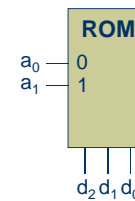
Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## 3.3.6. Síntesis con ROM

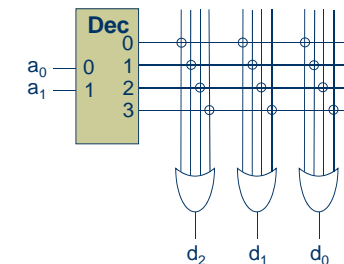
Una **ROM** de  $n$  entradas de dirección y  $m$  salidas de datos puede implementar  $m$  funciones lógicas cualesquiera de  $n$  entradas (en suma de minterms). **Ejemplo con  $n=2$  y  $m=3$ :**

○ posible conexión

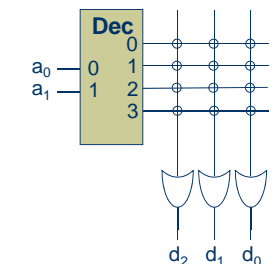
Símbolo:



Esquema lógico:



Esquema simplificado:



52

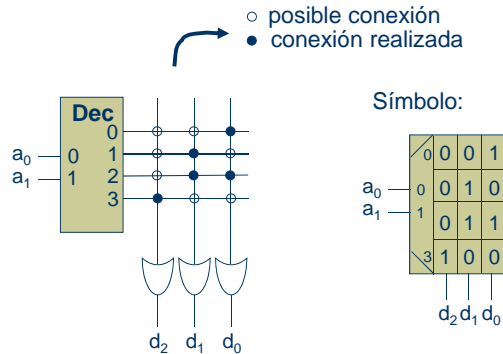
Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Síntesis con ROM

Una ROM de  $n$  entradas de dirección y  $m$  salidas de datos puede implementar  $m$  funciones lógicas cualesquiera de  $n$  entradas (en suma de minterms)

Ejemplo con  $n=2$  y  $m=3$ :  
Incrementador

$a_1$	$a_0$	$d_2$	$d_1$	$d_0$
0	0	0	0	1
0	1	0	1	0
1	0	0	1	1
1	1	1	0	0



53

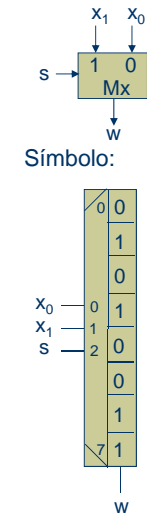
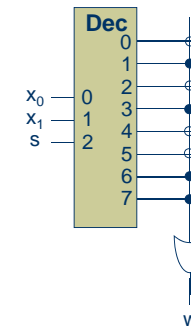
Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Síntesis con ROM

- Ejemplo: Multiplexor  
if  $s=0$  then  $w=x_0$  else  $w=x_1$

$s$	$x_1$	$x_0$	$w$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Esquema interno:



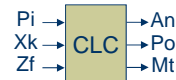
54

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Ejemplo de síntesis con ROM

Enunciado:

$P_i$	$X_k$	$Z_f$	$A_n$	$P_o$	$M_t$
0	0	0	1	1	0
0	0	1	1	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	0	0	0
1	0	1	0	1	0
1	1	0	0	0	1
1	1	1	1	0	0



55

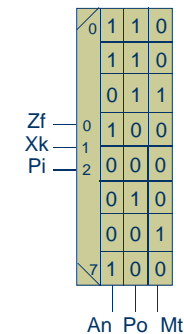
Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Ejemplo de síntesis con ROM

Solución:

$P_i$	$X_k$	$Z_f$	$A_n$	$P_o$	$M_t$
0	0	0	1	1	0
0	0	1	1	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	0	0	0
1	0	1	0	1	0
1	1	0	0	0	1
1	1	1	1	0	0

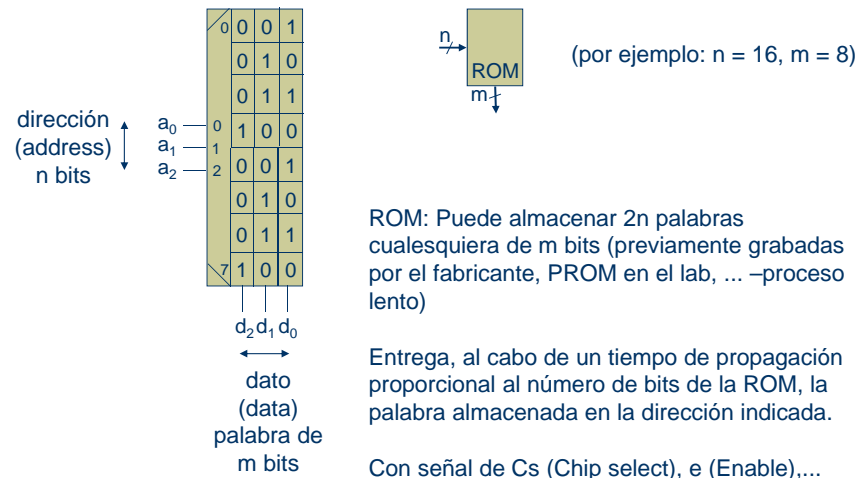
¡ojo con el etiquetado de los nombres!



56

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## (ROM como memoria de solo lectura)



57

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Actividad #3.1(e)

- ♦ Individual
- ♦ Intercambiad la hoja de la actividad #3.1 con vuestro compañero.
- ♦ En la parte de atrás de la hoja (tercer 1/3):
- ♦ A partir de la tabla de verdad de las salidas  $w_0$  y  $w_1$  de la actividad #3.1(b) o (c) (si las T.V. de los dos apartados no coinciden elige la que consideres correcta), dibuja el esquema lógico que las implementa (a) con un decodificador y puertas Or y (b) con una memoria ROM.

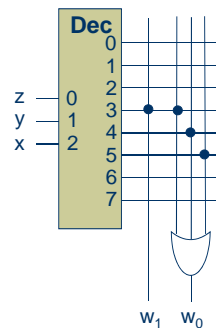
58

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Solución de la actividad #3.1(e). (a)

x	y	z	$w_1$	$w_0$
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	0
1	1	1	0	0

(a) Con Decodificador y puertas Or



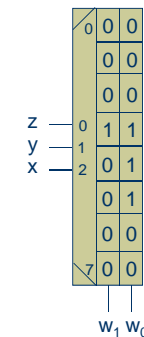
59

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Solución de la actividad #3.1(e). (b)

x	y	z	$w_1$	$w_0$
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	0
1	1	1	0	0

(b) Con una ROM



60

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

### 3.3.9. Actividad #3.2(a)

- ♦ Individual
- ♦ En una hoja. Nombre. Fecha.
- ♦ Escribid la tabla de verdad del siguiente CLC e implementad el CLC con un decodificador y puertas Or.
- ♦ En grupo de 3 estudiantes. Comparad vuestras soluciones y acordar una solución correcta.

61

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

### Funcionamiento del sistema

- ♦ Descripción del CLC: Diseñar el CLC del piloto automático de un avión encargado de controlar la dirección del avión durante el aterrizaje.
- ♦ Para que el aterrizaje sea seguro, el avión debe estar situado sobre una línea recta, prolongación de la pista de aterrizaje, unos cuantos kilómetros antes de llegar a la pista.
- ♦ Para que el piloto automático sepa la posición del avión respecto de esta línea imaginaria, se disponen dos filas de antenas, una a cada lado de esta línea. Emiten muy direccionales hacia el cielo.
- ♦ Las antenas de la izquierda emiten una señal de radio con frecuencia  $F_i$  y las de la derecha con frecuencia  $F_d$ .
- ♦ Dos receptores de radio, I y D, sintonizados a  $F_i$  y  $F_d$  generan las señales binarias  $S_i$  y  $S_d$ , que se activan al recibir su frecuencia.

62

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

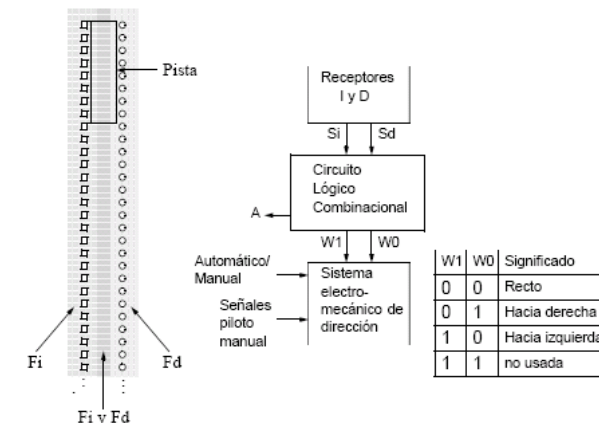
### Funcionamiento del sistema

- ♦ El CLC a diseñar recibe  $S_i$  y  $S_d$  y genera  $W_1$  y  $W_0$  que van al subsistema electromecánico que gobierna la dirección del avión y A (Alarma) que va a un sistema de alarma. Las señales  $W_1$  y  $W_0$  le indican al sistema electromecánico si debe dirigir el avión recto, hacia la derecha o hacia la izquierda, según el código de la tabla.
- ♦ El piloto visualiza la recepción de las frecuencias de radio y cuando el avión está situado en la línea de aterrizaje ( $F_i = 1$  y  $F_d = 1$ ) activa la señal Automático/Manual.
- ♦ Pueden venir vientos racheados que desplacen el avión hacia una dirección distinta de la deseada. Si no se recibe ninguna frecuencia el CLC debe sacar un 1 en la señal A para que se emita una alarma y el piloto aterrice manualmente. Cuando A vale 1, las señales  $W_1$  y  $W_0$  deben indicarle al sistema electromecánico de dirección que vaya recto. Cuando el piloto tome el control manual, el sistema electromecánico hará caso omiso de las señales  $W_1$  y  $W_0$ .

63

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

### Dibujo de la pista y sistema a grandes bloques



64

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.



## Solución actividad #3.2

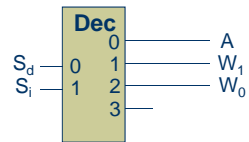
$S_i$	$S_d$	A	$W_1$	$W_0$
-------	-------	---	-------	-------

0	0	1	0	0
0	1	0	1	0
1	0	0	0	1
1	1	0	0	0

→ Alarma, no importa  
 → Izquierda  
 → Derecha  
 → Recto

**Objetivo:** dado un texto que describe un CLC encontrar su tabla de verdad

**Objetivo:** dada una tabla de verdad dibujar el esquema del CLC con Decodificador y Or



¡En este caso concreto no hacen falta puertas Or!

65

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## 3.4. Completando el modelo de un CLC: Tiempo de propagación

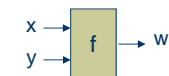
- 3.4.1. Tiempo de propagación.
- 3.4.2. Cronogramas. Ejercicio
- 3.4.3. Camino crítico  
Actividad #3.3
- 3.4.4. Una curiosidad: Glitches  
Actividad #3.4

66

Copyright © Juan J. Navarro. Departament d'Arquitectura de Computadors. Universitat Politècnica de Catalunya.

### 3.4.1. Tiempo de propagación

#### Modelo lógico:



~~$w(t) = f(x(t), y(t))$~~   
 respuesta inmediata

Tabla de verdad:

x	y	w
0	0	0
0	1	1
1	0	0
1	1	0

En un CLC real la **respuesta no es inmediata**.

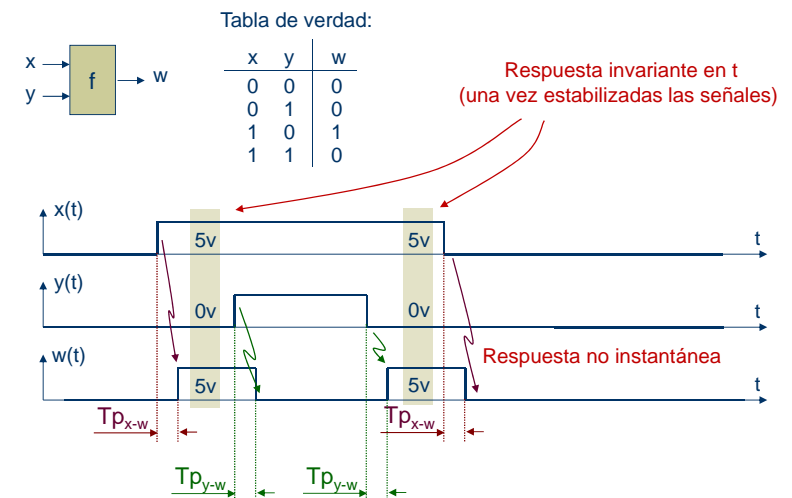
**Tiempo de propagación** de la entrada **e** a la salida **s**,  $T_{p_{e-s}}$ :

Tiempo desde que se produce un cambio en la entrada **e** hasta que la salida **s** se estabiliza al valor que indica la tabla de verdad para la nueva combinación de los valores de las entradas.

67

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

### Tiempo de propagación de **e** a **s**: $T_{p_{e-s}}$

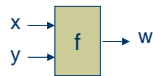


68

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## En nuestro modelo de tiempos




- ♦  $T_{p_{x-w}}$  es el mismo cuando  $w$  pasa de 0  $\rightarrow$  1 que cuando pasa de 1  $\rightarrow$  0  
(la realidad puede no ser así. V.g. TTL:  $T_{pHL_{x-w}} \neq T_{pLH_{x-w}}$ )
- ♦ Cuando  $T_{p_{x-w}} = T_{p_{y-w}} \rightarrow T_p$   
 **$T_p$ : Tiempo de propagación**
- ♦ (Modelo simplificado:  $T_p = \max(T_{p_{x-w}}, T_{p_{y-w}})$ )



69

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Puertas Lógicas, modelo completo

Nombre	Símbolo	Expresión lógica	Tabla de verdad	Tiempo de propagación															
Not		$w = !x$	<table><tr><th>x</th><th>w</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	x	w	0	1	1	0	$T_p(\text{Not}) = 10 \text{ u.t.}$									
x	w																		
0	1																		
1	0																		
And-2		$w = x \cdot y$	<table><tr><th>x</th><th>y</th><th>w</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	x	y	w	0	0	0	0	1	0	1	0	0	1	1	1	$T_p(\text{And-2}) = 20 \text{ u.t.}$
x	y	w																	
0	0	0																	
0	1	0																	
1	0	0																	
1	1	1																	
Or-2		$w = x + y$	<table><tr><th>x</th><th>y</th><th>w</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	x	y	w	0	0	0	0	1	1	1	0	1	1	1	1	$T_p(\text{Or-2}) = 20 \text{ u.t.}$
x	y	w																	
0	0	0																	
0	1	1																	
1	0	1																	
1	1	1																	

70

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## 3.4.2. Cronogramas de CLC

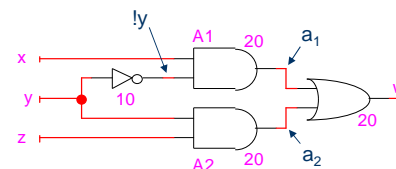
**Cronograma de una puerta** = comportamiento lógico retardado  $T_p$

71

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Cronograma de un circuito con puertas

**Cronograma de un CLC formado por puertas** = cronograma de la salida de cada puerta



x	y	z	w
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

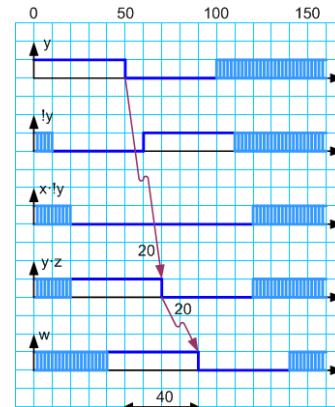
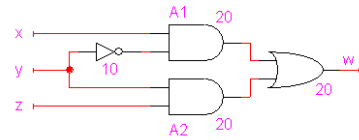
La entrada y cambia 1  $\rightarrow$  0

La salida w debe cambiar

72

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Cronograma de un circuito con puertas



$T_{p_{y-w}}$ : 40 u.t.  
(salida estable)

73

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## 3.4.3. Camino crítico

### $T_{p_{x-w}}$ de un CLC formado por CLC interconectados

**Camino de la entrada x a la salida w:** trayectoria (recorrido) de x a w pasando por CLC interconectados. Al pasar por un CLC se entra por una entrada y se sale por una salida:



**Camino crítico de la entrada x a la salida w:** el camino más largo de x a w. La longitud se mide como la suma de los  $T_p$  de cada entrada a cada salida de los CLC atravesados por el camino. Puede haber varios.

**Tiempo de propagación de x a w ( $T_{p_{x-w}}$ )**

=

**Tiempo (longitud) del camino crítico de x a w.**

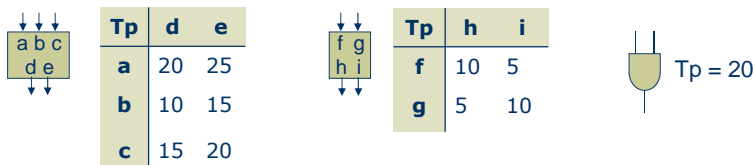
74

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

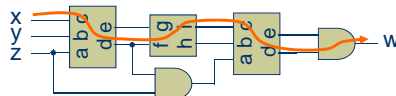
## Actividad #3.3

### Cálculo del $T_{p_{x-w}}$ de un CLC:

$T_p$  de los CLC internos que forman el CLC:



**Caminos de x a w:** hay 10 caminos diferentes ¡Encontradlos! (y sus tiempos)

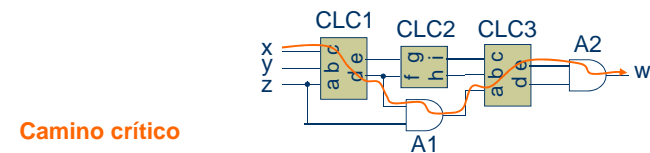


$$T = T_{p_{cd}}(\text{CLC1}) + T_{p_{fi}}(\text{CLC2}) + T_{p_{cd}}(\text{CLC3}) + T_p(\text{And-2}) = 15 + 5 + 15 + 20 = 55$$

75

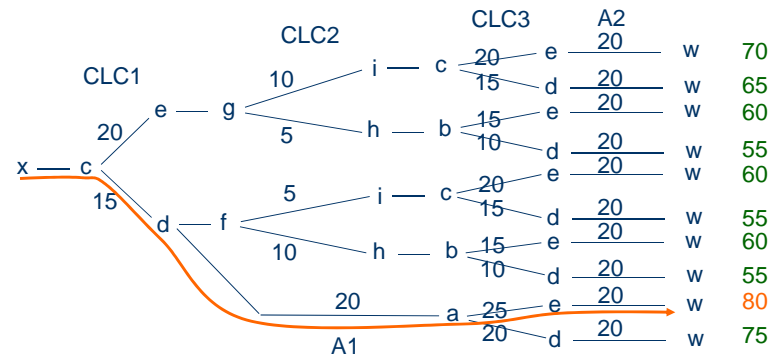
Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

## Solución Actividad #3.3



**Camino crítico**

**$T_{p_{xw}} = 80$**

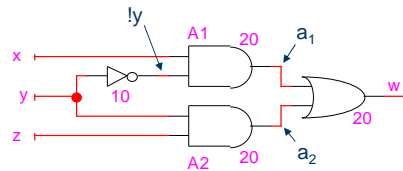


76

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

### 3.4.4. Una curiosidad: Glitches

Si hay dos caminos con distinto retardo de una entrada a una salida puede ocurrir que antes de que la salida se establezca al valor que indica la T.V. se produzcan pulsos indeseados: ¡Glitches!



z	y	x	w
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

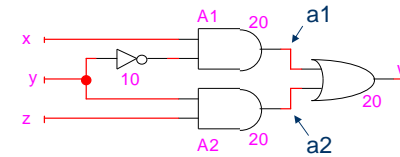
La entrada  
y  
cambia  
1 → 0

La salida  
w  
no debe  
cambiar

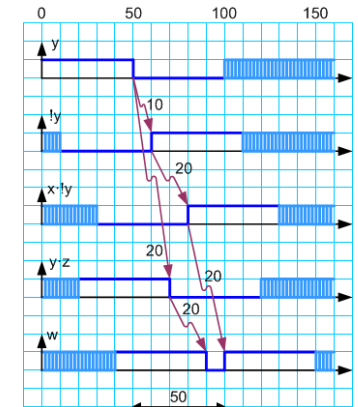
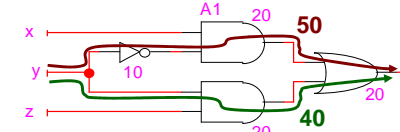
77

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

### ¿Por qué aparecen Glitches?



Dos caminos convergentes con distinto retardo



$T_{p_{y-w}}$ : 50 u.t.  
(salida estable)

78

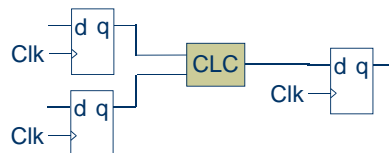
Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.

### Tp y glitches: ¡no son un problema!

Consideramos que la señal es correcta después de pasado el tiempo de propagación, cuando la salida es estable al valor correcto.

¿Cómo?

- Trabajaremos con circuitos secuenciales **síncronos**.
- La señal de salida de un subcircuito combinacional se almacena en un biestable cuando llega la señal del flanco ascendente del reloj.
- Nos aseguraremos, al diseñar el tiempo de ciclo, que el flanco ascendente del reloj no llegue antes de que las señales en las entradas D de los biestables estén estabilizadas.



Tiempo de ciclo mayor que  
 $T_p(\text{Flip-flop}) + T_p(\text{CLC})$

79

Copyright © Juan J. Navarro. Universitat Politècnica de Catalunya.