

Dispositivos lógicos programables

- Alternativas:
 - La suma de términos productos.
 - 16R6 MMI (PAL).
 - CoolRunner Xilinx (CPLD).
 - MAX 7000 Altera (EPLD).
 - Tablas de lookup.
 - Stratix Altera.
 - Cyclone Altera.
 - Sapartan Xilinx.
 - Virtex Xilinx.
 - Lógica basada en multiplexores.
 - ACT1, ACT2, ACT3 Actel

3

Programmable Logic Array (PAL)

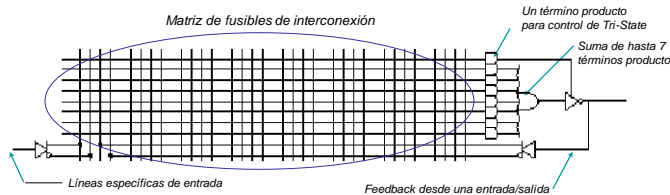


Img: [COM]

- 1978 H. T. Chua and John Birkner de Monolithic Memories.
- De PROM a PAL.
- Basadas en sumas de términos producto.

4

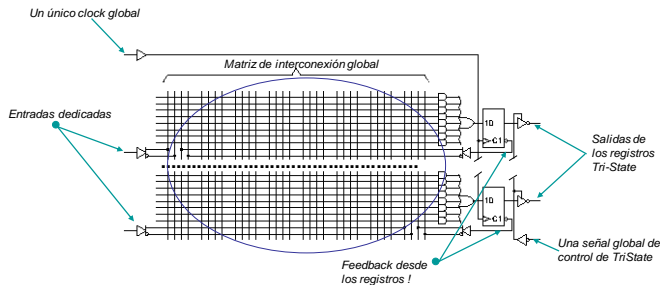
Programmable Logic Array



- Se distribuye a lo largo del CI una línea de metal con los literales de todas las variables de entrada (directa y complementada).
- Mediante fusibles se pueden generar cualquier término producto (AND) de las variables de entrada.
- Para sintetizar la función deseada sólo resta sumar (OR) los términos producto necesarios.

5

PAL con registros

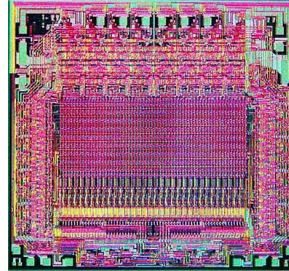


- Se incorporan registros, ejemplo 16R8, para sintetizar circuitos secuenciales.

6

PAL 16R8

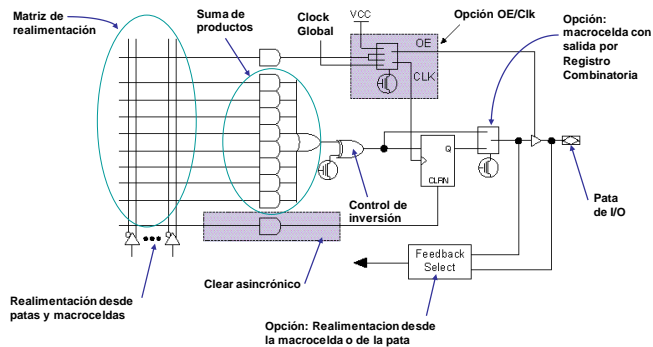
- Die photograph of the MMI PAL16R8



Img: [COM]

7

Macroelda



- Celda de síntesis lógica (macroelda) de PAL EP610.

8

PAL, conclusiones

- Productos lógicos con un gran número de variables. Muy útil en la síntesis de funciones con gran fan-in.
- A la compuerta OR llegan entre 6 y 8 términos producto.
- Es elevada el área que ocupa cada término producto, que también involucra un gran número de fusibles.
- Frente a las antiguas soluciones PROM el empleo de PALs significa una reducción de 300.

9

Ejemplo Macrocelda CPLD

No distribuye los literales por todo el integrado.

Programación EEPROM

- CoolRunner II
 - Hasta 56 términos producto.
 - Orex.
 - FF D/T, Latch o doble flanco.
 - Set y reset internos globales.

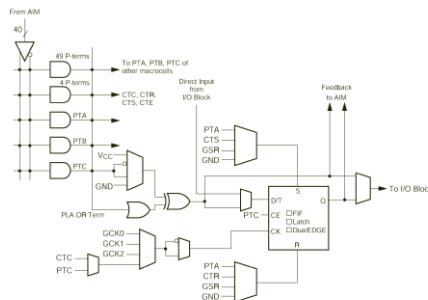


Figure 3: CoolRunner-II CPLD Macrocell

10

Ejemplo Macrocelda EPLD

No distribuye los literales por todo el integrado.
Programación EEPROM

- MAX7000. Macrocelda similar a las anteriores pero con:
 - Expansores compartidos.
 - Expansores paralelo.

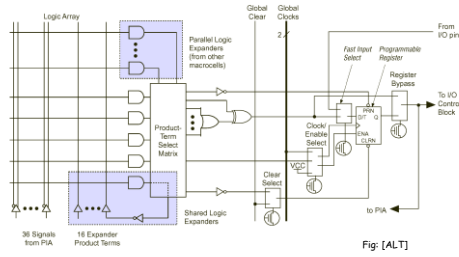


Fig. [ALT]

11

EPLDs, expansores compartidos

Ejemplo MAX7000

- 1 por cada macrocelda 16 por cada LAB.
- Se agrega un pequeño retardo (t_{SEXP}).
- Cada expansor puede alimentar varias macroceldas.
- Términos producto de más variables.
- Multifunciones.

Figure 5. Shareable Expanders

Shareable expanders can be shared by any or all macrocells in an LAB.

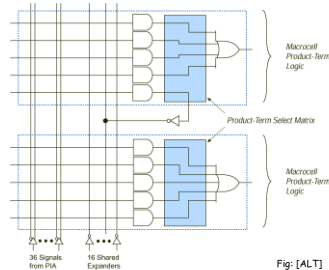


Fig. [ALT]

12

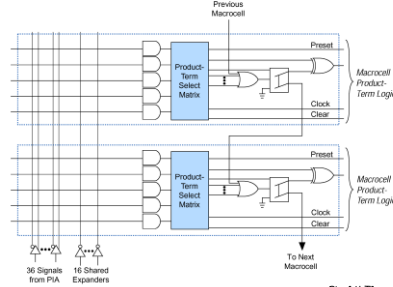
EPLDs, expansores paralelos

Ejemplo MAX7000

- Una OR puede tener hasta 20 términos producto (5 locales y 15 de celdas vecinas)
- Cada 5 términos se agrega un retardo t_{PEXP} .

Figure 6. Parallel Expanders

Unused product terms in a macrocell can be allocated to a neighboring macrocell.

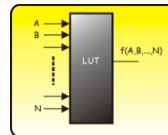


13

Elementos de síntesis lógica

Elementos de síntesis lógica:

- Basadas en sumas de términos producto, caso de las PAL.
- Alternativa, ¿Qué bloques lógicos permiten implementar cualquier función lógica?
 - Tablas de búsqueda (LUT Look-up Table), caso de las FPGAs:
 - Memoria.
 - Multiplexor.
 - Decodificador.

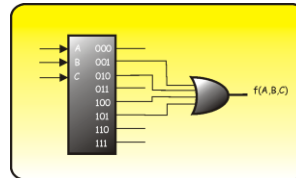
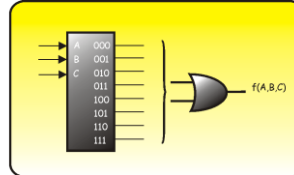


14

Síntesis mediante decodificador

Para sintetizar una función determinada se conectan a la entrada de la OR los minitérminos.

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

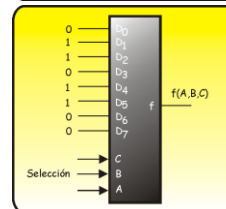
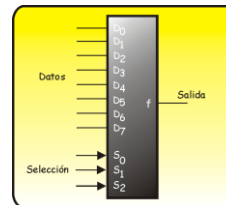


15

Síntesis mediante multiplexor

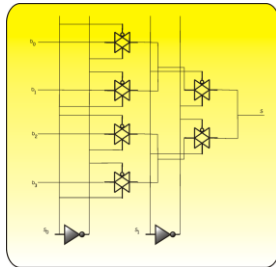
Se colocan en las entradas de datos los minitérminos y en las de selección las variables.

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



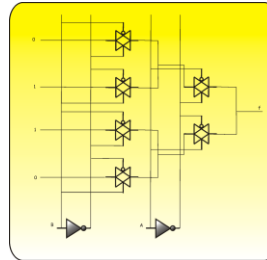
16

Multiplexores CMOS



A	B	f
0	0	0
0	0	1
0	1	1
0	1	0

Multiplexores CMOS → bajo número de transistores



17

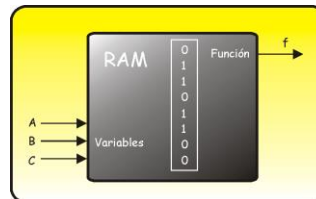
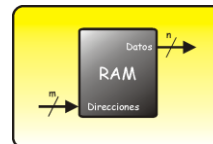
Síntesis mediante memorias

Líneas de dirección → variables

Salidas → funciones

Contenido → tabla de verdad

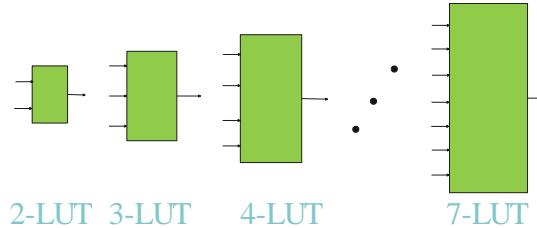
A	B	C	f
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0



18

Síntesis lógica mediante LUTs

- ¿Cuál es el número de entradas óptimo, 2, 3 4.. (K)?



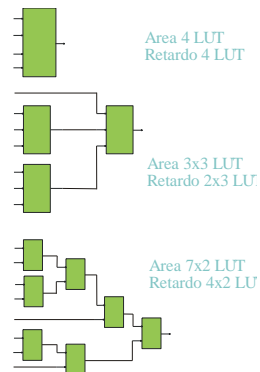
- Se miden los resultados de área y velocidad.

19

LUTs valor óptimo de K 1/2

Ejemplo:

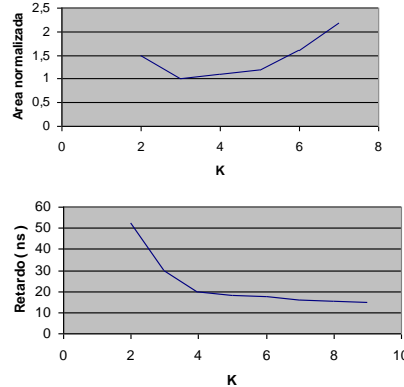
$$F = A.B.D + BCD' + A'B'C'$$



20

LUTs valor óptimo de K 2/2

- Se miden áreas y retardos según el número de entradas.
- Los resultados indican que normalmente se trabaje con LUTs de K=4 o 5.



21

Ejemplo: Configurable Logic Block (CLB) X3000

- Hasta 5 variables de entrada A, B, C, D, y E
- Línea de entrada directa de datos DI.
- Habilitación de reloj EC.
- Inversión de reloj K.
- Reset asincrónico RD.
- Dos salidas X e Y.

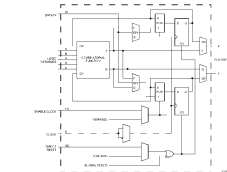


Figure 5: Configuración Logic Block

Fig: [XIL]

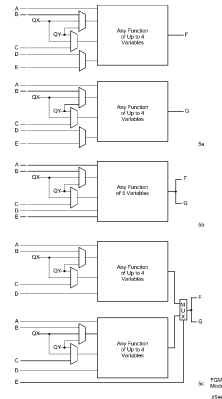


Figure 6: Combinational Logic Options

22

Ejemplo: Configurable Logic Block (CLB) X4000 1/2

- Cualquier función de hasta 4 variables más una segunda función de hasta 4 variables no relacionadas con las anteriores.
- Cualquier función de 5 variables.
- Cualquier función de 4 variables junto con algunas funciones de 6 variables.
- Algunas funciones de hasta 9 variables.

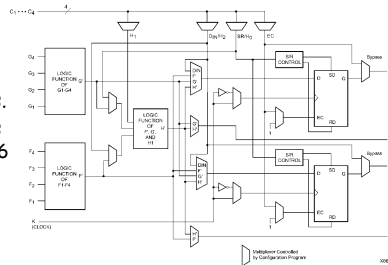


Figure 1: Simplified Block Diagram of XC4000 Series CLB (RAM and Carry Logic Functions not shown)

Fig: [XIL]

23

Ejemplo: Configurable Logic Block (CLB) X4000 2/2

Empleo como memoria RAM

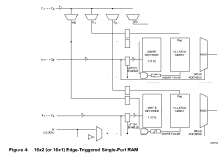


Figure 4: 16x16 Edge-Triggered Single-Port RAM

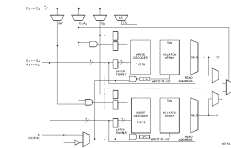


Figure 5: 16x16 Edge-Triggered Single-Port RAM (2 and 3 addresses are identical)

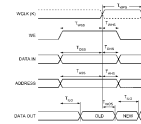


Figure 3: Edge-Triggered RAM Write Timing

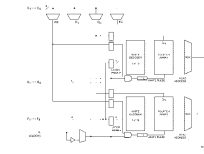


Figure 7: 16x16 Edge-Triggered Dual-Port RAM

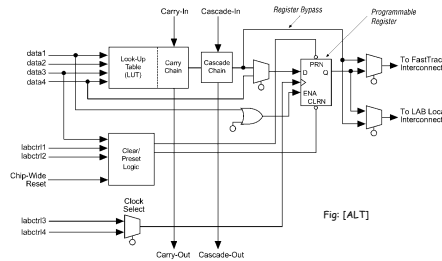
Fig: [XIL]

24

FLEX10K Logic Element (LE)

Figure 6. FLEX 10K Logic Element

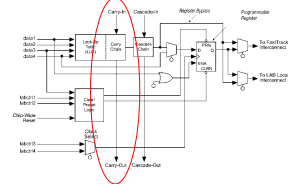
- Permite Register Packing. Se puede trabajar el ff separado de la parte combinacional.
- Carry chain.
- Cascade chain.
- Operación modo
 - Normal.
 - Aritmético.
 - Contador.



25

FLEX10K, carry chain 1/2

Figure 6. FLEX 10K Logic Element



- Transmisión muy rápida de acarreo.
- La 4-LUT se divide en dos 3-LUT.
- Se genera el acarreo y la suma por separado.

Figure 7. Carry Chain Operation (n-bit Full Adder)

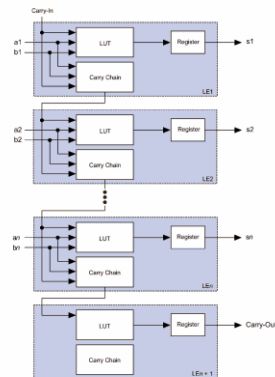


Fig. [ALT]

26

FLEX10K, carry chain 2/2

Ejemplo: multiplicador de 8×8

	<u>EPF81500A-2</u>		<u>EPF10K10-3</u>	
	<i>Elementos lógicos</i>	<i>Mayor retardo</i>	<i>Elementos lógicos</i>	<i>Mayor retardo</i>
	(LEs)	(ns)	(LEs)	(ns)
<i>Sin cadena</i>	161	56,9	161	57,4
<i>Con acarreo</i>	136	31,1	136	36,4

El empleo del acarreo encadenado puede

- mejorar la performance
- reducir el consumo de lógica
- los LEs se agrupan y permanecen juntos

27

FLEX10K, cascade chain 1/2

Figure 6. FLEX10K Logic Element

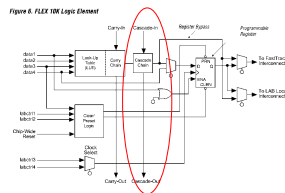
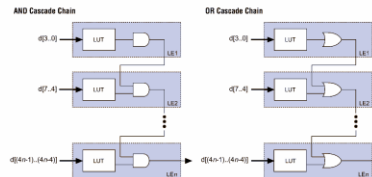


Figure 8. Cascade Chain Operation



- Pensado para implementar funciones de gran fanin.

Fig: [ALT]

28

FLEX10K, cascade chain 2/2

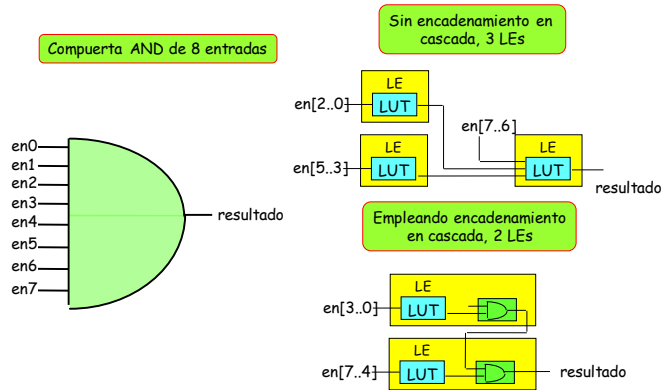


Fig: [IAQ]

29

FLEX10K, modos de funcionamiento

- Normal.
- Aritmético.
 - Carry chain.
- Contador ascendente/descendente.
 - Counter enable.
 - Clock enable.
 - Control sincrónico up/down.
 - Carga de cuenta.
- Contador borrable.
 - Similar al anterior pero con borrado sincrónico.

Figure 9: FLEX10K LE Operating Modes

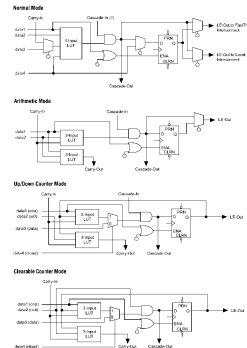


Fig: [ALT]

30

FLEX10K, clear y preset

- Clear asincrónico.
- Preset asincrónico.
- Preset y clear asincrónico.
- Carga asincrónica con clear.
- Carga asincrónica con preset.
- Carga asincrónica sin preset o clear.

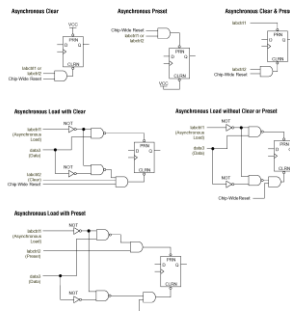


Fig: [ALT]

31

Stratix Logic Element (LE)

Figure 2-5. Stratix LE

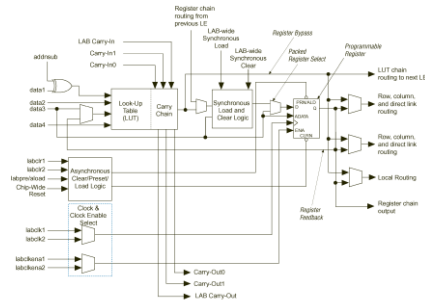


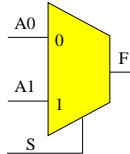
Fig: [ALT]

- Modo dinámico de suma y sustracción.
- Registro programable como D, T, JK, o RS.
- Register packing.
- Register feedback.
- Encadenamiento de LUT, permiten conectar en cascada LUTs del mismo LAB para funciones de gran fan in.

32

Síntesis lógica mediante multiplexores

A0	A1	S	Función
0	0	0	0
B	0	A	$A' B$
0	1	A	A'
A	0	B	$A B'$
0	1	B	B'
0	B	1	B (BUF)
0	B	A	$A B$
0	A	1	A (BUF)
B	1	A	$A + B$
1	1	1	'1'



- Un único **MUX 2:1** permite la síntesis de ocho posibles funciones no obvias de dos variables A y B , además de las obvias '1' y '0':
 - $\text{NOT } A$ y $\text{NOT } B$ (A' y B')
 - $\text{BUF } A$ y $\text{BUF } B$
 - $A + B$ y $A \cdot B$
 - $A \cdot B'$
 - $A' \cdot B$
- El multiplexor y el negador son dispositivos fáciles de realizar con CMOS

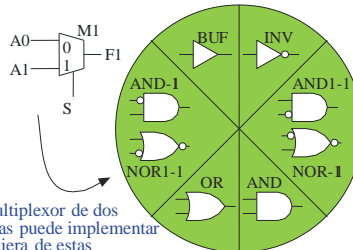
Ref: [IAQ]

35

Síntesis lógica mediante multiplexores

- Mediante un multiplexor de 2:1 se puede implementar cualquiera de las siguientes funciones: $A \cdot B$, $A' \cdot B$

A0	A1	S	Función
0	0	0	0
B	0	A	$/A B$
0	1	A	$/A$
A	0	B	A/B
0	1	B	$/B$
0	B	1	B (BUF)
0	B	A	$A B$
0	A	1	A (BUF)
B	1	A	$A + B$
1	1	1	1



Un multiplexor de dos entradas puede implementar cualquiera de estas funciones, seleccionadas por $A0$, $A1$ y S .

Fig: [SMI]

36

Teorema de expansión de Shannon

Sirve para descomponer una función de muchas variables en partes más pequeñas.

$$F = f(A, B, C, \dots) = A \cdot f(1, B, C, \dots) + A' \cdot f(0, B, C, \dots)$$

$$\text{Ej: } F(A, B, C) = A' \cdot B + A \cdot B \cdot C' + A' \cdot B' \cdot C = A \cdot (B \cdot C') + A' \cdot (B + B' \cdot C)$$

$$\text{Se puede seguir: } F(1, B, C, \dots) = B \cdot f(1, 1, C, \dots) + B' \cdot f(1, 0, C, \dots) \\ \text{y } F(0, B, C, \dots) = B \cdot f(0, 1, C, \dots) + B' \cdot f(0, 0, C, \dots)$$

etc.

37

Síntesis mediante ACT1

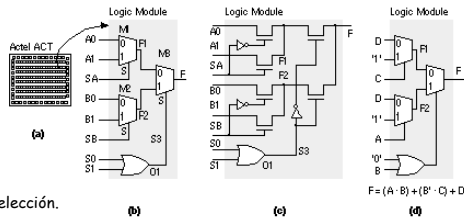
Ejemplo: con el conjunto de multiplexores de una ACT1:

$$F = A \cdot B + /B \cdot C + D$$

Se selecciona B

$$F = B \cdot (A + D) + /B \cdot (C + D) = B \cdot F2 + /B \cdot F1$$

que es un mul. de 2:1 con B como selección.



Expandiendo F1 y F2:

$$F2 = A + D = A \cdot 1 + A' \cdot D$$

$$F1 = C + D = C \cdot 1 + C' \cdot D$$

38

Celda de síntesis de ACTEL ACT2

- ACT2 C-Module similar a ACT1 pero puede implementar funciones de 5 variables.
- S-Module le agrega un registro a la parte combinacional.

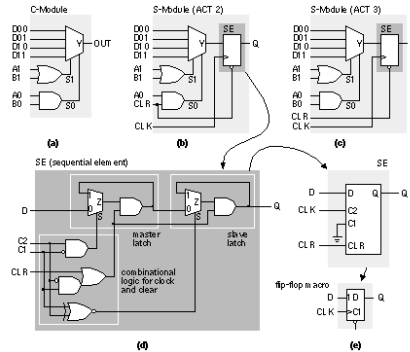


Fig. [SMI]

39

Virtex II Pro CLB's y Slice 1/9

- CLB's organizados en arreglos.
 - Conectado a una matriz de ruteo.
- Cada CLB 4 slices.
 - Agrupados en dos columnas.
 - Dos cadenas independientes de carry.
 - Una cadena de desplazamiento.

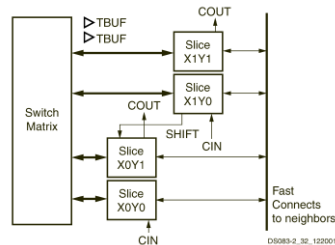


Figure 32: Virtex-II Pro CLB Element

Fig. [XIL]

40

Virtex II Pro CLB's y Slice 2/9

- Slice:
 - 2 LUT4.
 - Lógica para generación de carry.
 - Multiplexores.
 - LUTs como memoria distribuida.

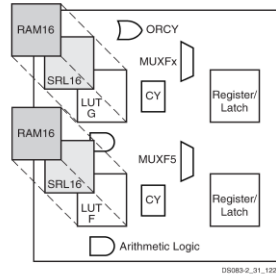


Figure 33: Virtex-II Pro Slice Configuration

Fig: [XII]

41

Virtex II Pro CLB's y Slice 3/9

Vista detallada de un slice

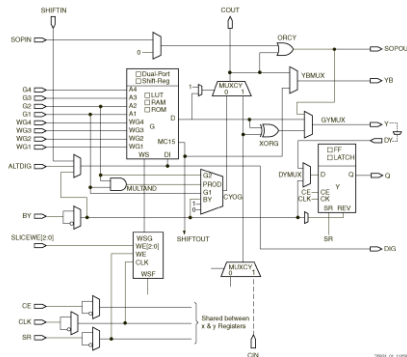


Figure 34: Virtex-II Pro Slice (Top Half)

Fig: [XII]

42

Virtex II Pro CLBs y Slice 4/9

- Registros
 - Como ffd o latch.
- Reset y clear
 - Sin set ni reset.
 - Set sincrónico.
 - Reset sincrónico.
 - Set y reset sincrónico.
 - Set asincrónico (preset).
 - Reset asincrónico (clear).
 - Set y reset asincrónico (preset y clear).

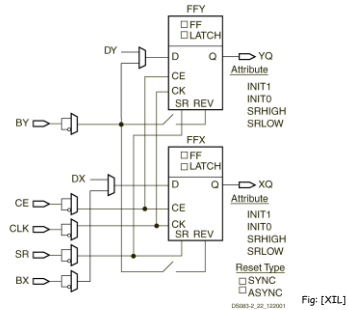


Figure 35: Register / Latch Configuration in a Slice