

# Controlador de DDR para FPGA

Miguel A. Sagreras

Los dispositivos denominados FPGAs comienzan a desplazar en aplicaciones de alta prestación a los procesadores tradicionales. En este contexto, se requiere del uso de memoria masiva DDR DRAM. El controlador, la interfaz entre la aplicación y la misma, se programa en la FPGA y debe ser capaz de transferir datos desde y hacia la memoria de manera de alcanzar las velocidades máximas que ofrecen los recursos de la FPGA y de esta forma aprovechar todo su potencial.

# Contenidos

# Números

Decimal	Hexadecimal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9

# Números

Decimal	Hexadecimal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	A

# Números

Decimal	Hexadecimal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	A
11	B
12	C
13	D
14	E
15	F

# Números

Decimal	Hexadecimal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	A
11	B
12	C
13	D
14	E
15	F
16	10

# Números

Decimal	Hexadecimal	Binario
0	0	0
1	1	1
2	2	
3	3	
4	4	
5	5	
6	6	
7	7	
8	8	
9	9	
10	A	
11	B	
12	C	
13	D	
14	E	
15	F	
16	10	

# Números

Decimal	Hexadecimal	Binario
0	0	0
1	1	1
2	2	10
3	3	
4	4	
5	5	
6	6	
7	7	
8	8	
9	9	
10	A	
11	B	
12	C	
13	D	
14	E	
15	F	
16	10	

# Números

Decimal	Hexadecimal	Binario
0	0	0
1	1	1
2	2	10
3	3	11
4	4	
5	5	
6	6	
7	7	
8	8	
9	9	
10	A	
11	B	
12	C	
13	D	
14	E	
15	F	
16	10	

# Números

Decimal	Hexadecimal	Binario
0	0	0
1	1	1
2	2	10
3	3	11
4	4	100
5	5	
6	6	
7	7	
8	8	
9	9	
10	A	
11	B	
12	C	
13	D	
14	E	
15	F	
16	10	

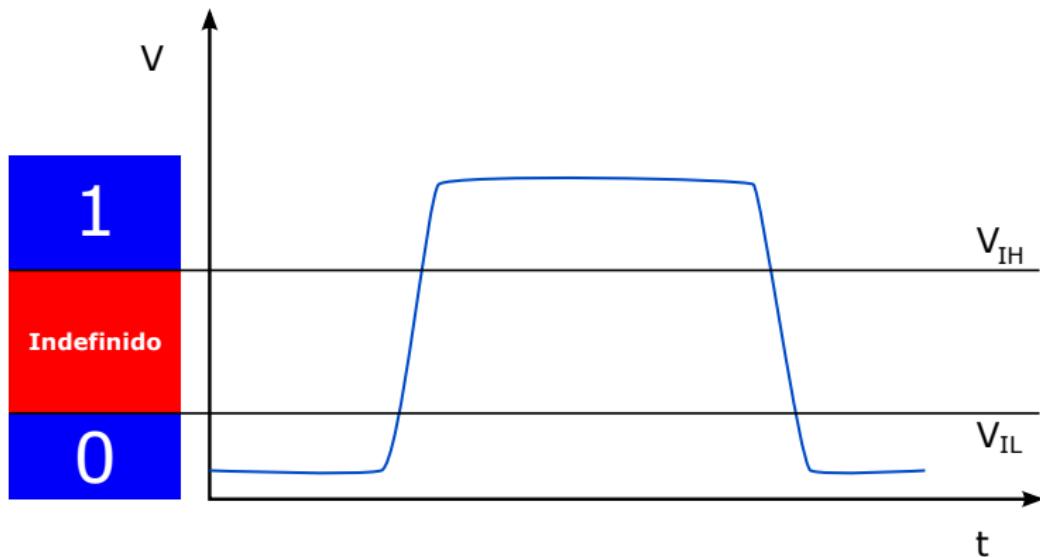
# Números

Decimal	Hexadecimal	Binario
0	0	0
1	1	1
2	2	10
3	3	11
4	4	100
5	5	101
6	6	
7	7	
8	8	
9	9	
10	A	
11	B	
12	C	
13	D	
14	E	
15	F	
16	10	

# Números

Decimal	Hexadecimal	Binario
0	0	0
1	1	1
2	2	10
3	3	11
4	4	100
5	5	101
6	6	110
7	7	111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111
16	10	10000

# Valores lógicos

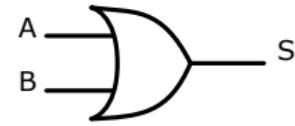


# Compuertas

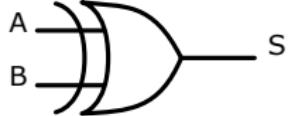
A	$\wedge$	B	S
F(0)	F(0)	F(0)	F(0)
T(1)	F(0)	F(0)	F(0)
F(0)	T(1)	F(0)	F(0)
T(1)	T(1)	T(1)	T(1)



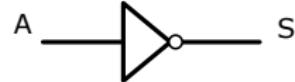
A	$\vee$	B	S
F(0)	F(0)	F(0)	F(0)
T(1)	F(0)	F(0)	T(1)
F(0)	T(1)	T(1)	T(1)
T(1)	T(1)	T(1)	T(1)



A	$\vee\!\vee$	B	S
F(0)	F(0)	F(0)	F(0)
T(1)	F(0)	T(1)	T(1)
F(0)	T(1)	T(1)	T(1)
T(1)	T(1)	F(0)	F(0)

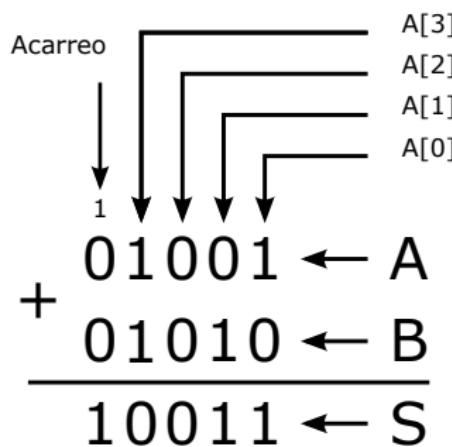


$\bar{A}$	S
F(0)	T(1)
T(1)	F(0)

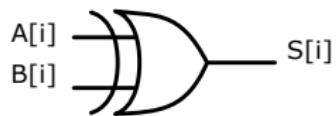


# Aritmética

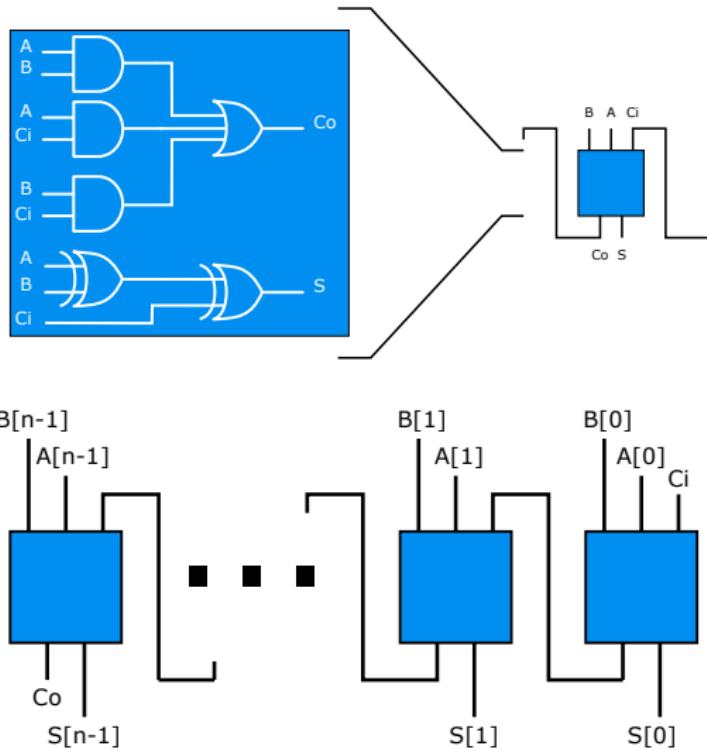
$$\begin{array}{r} 9 \\ + 10 \\ \hline 19 \end{array}$$



$A[i]$	$\vee$	$B[i]$	$S[i]$
0		0	0
1		0	1
0		1	1
1		1	0



# Sumador



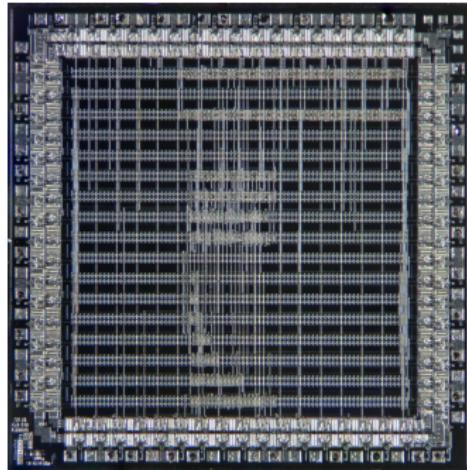
## Dificultades

- Tiempos de propagación
  - Compuertas
  - Cables
- Número de componentes



# Gate Arrays o Arreglos de Compuertas

- C.I. con compuertas y bloques prearmados.
- Ventajas
  - Fabricación de gran escala
  - Sólo requieren las máscaras de interconexión
  - Reduce el costo de la aplicación específica



"S-MOS Systems ASIC SLA6140"  
by Antoinebercovici

# FPGAs Field Programmable Gate Arrays

- 1985 Xilinx
- Gate Arrays cuya red de interconexión es programable.
- Son los impulsores del incremento en la escala de integración
- Tecnología de exportación restringida

# Ventajas

## FPGA vs procesadores

- Más rápidas
- Menor consumo a igual aplicación
- Se adaptan a aplicaciones específicas

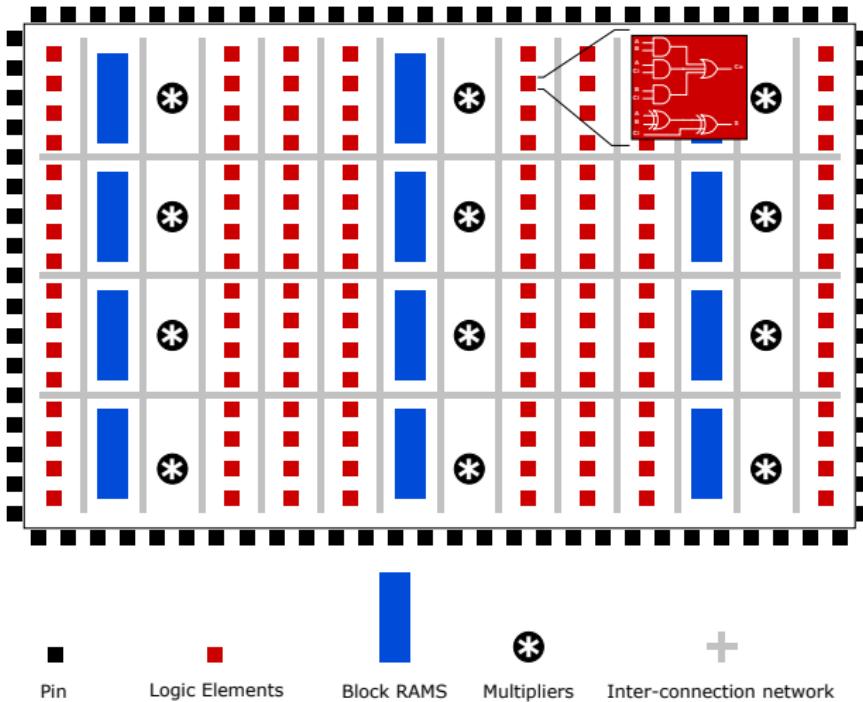
## FPGA vs ASIC

- Menor costo de fabricación para bajo volumen
- Reprogramables para actualizaciones de la aplicación

# Aplicaciones

- Aeroespacio y Defensa
- Electromedicina.
- Prototipado de ASIC.
- Audio
- Automotor
- Radiodifusión
- Electrónica de Consumo
- Data Center
- Video & Procesamiento de Imagen
- Computación de Alta Prestación
- Industrial
- Medicina
- Instrumentación Científica
- Seguridad
- Comunicaciones alámbricas
- Comunicaciones inalámbricas

# Diagrama en bloques



# NU HORIZONS Spartan 3A DSP board



# Descripción

## NU HORIZONS Spartan 3A DSP board

- 2008
- Spartan 3A DSP
  - 37440 LE equivalente a 14 procesadores de 32 Bits.
  - 84 BRAM de 18Kbits
  - 84 Unidades de DSP
  - 250 Mhz
  - Tecnología de exportación restringida
- 32 MBytes de Memoria DDR1 333 MT/s
- 2 ADC
  - Diferencial
  - 14 bits
  - 120 MS/s
  - 500 Mhz
- Video DAC salida diferencial
- Ethernet
- Precio USD 200

# Algoritmo en Lenguaje C y en Assembly Optimizado

## Ejemplo de aplicación : Convolución

$$y[n] = \sum_{i=0}^{N=84-1} h[i] * x[n - i]$$

```
/*
 * Convolucion *
 */
#include <stdlib.h>

typedef int sample;

sample convolution (x, h, n)
sample x[];
sample h[];
int n;
{
    sample y;
    y = 0;
    for (int i = 0; i < 84; i++)
        y += h[i]*x[n-i];
    return y;
}

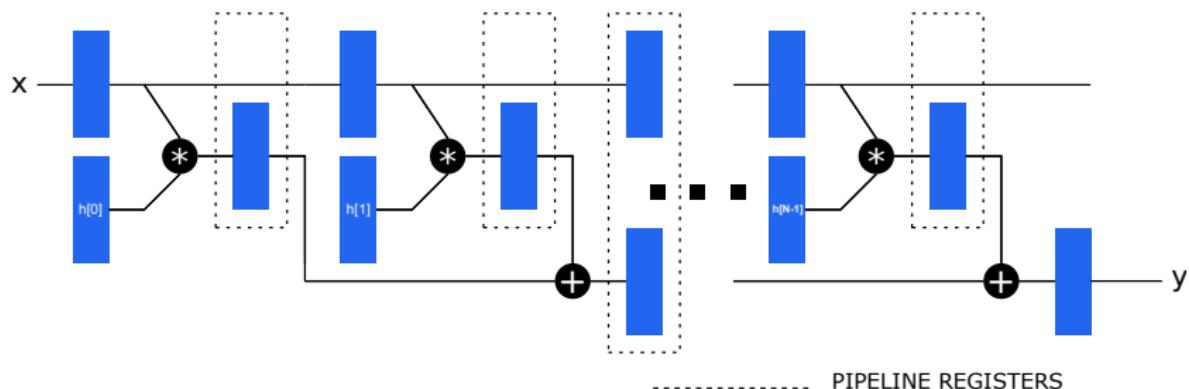
;   GCC 4.8.4 -O2 -S -std=gnu99 -mtune=corei7 -avx
;
; ++ Aportan al cálculo en el lazo
; -- NO aportan al cálculo en el lazo
;

convolution:
    movslq    %edx, %rax
    salq     $2, %rax
    leaq      (%rdi,%rax), %rdx
    leaq      -336(%rdi,%rax), %rdi
    xorl     %eax, %eax
.L3:
    movl      (%rsi), %ecx ; -- <---+ 84 veces
    subq     $4, %rdx ; -- | 7 instr.
    addq     $4, %rsi ; --
    imull    4(%rdx), %ecx ; ++
    addl     %ecx, %eax ; ++
    cmpq     %rdi, %rdx ; --
    jne   .L3 ; -- >>--+
    rep ret
```

# Diagrama en bloque para FPGA

Ejemplo de aplicación : Convolución

$$y[n] = \sum_{i=0}^{N=84-1} h[i] * x[n - i]$$



# Comparación estimativa entre FPGA y Procesador

Ejemplo de aplicación : Convolución

## Spartan 3A DSP

- Tecnología 2004
- 90 nm
- 250 Mhz o ciclo de 4 ns
- Un resultado por ciclo
- Un resultado cada 4 ns

## Intel Corei7 920

- Tecnología 2008
- 45 nm
- 2.66 Ghz o ciclo de 0.375 ps
- Un resultado cada 84 ciclos del lazo
- Un lazo 7 instrucciones
- 0.44<sup>a</sup> ciclos por instrucción.
- Un resultado cada  $84*7*0.44*0.375 = 97$  ns

<sup>a</sup>Computer Architecture A Quantitative Approach  
5th edition. Hennessy Patterson. Figura 3.43

## Aumento de velocidad

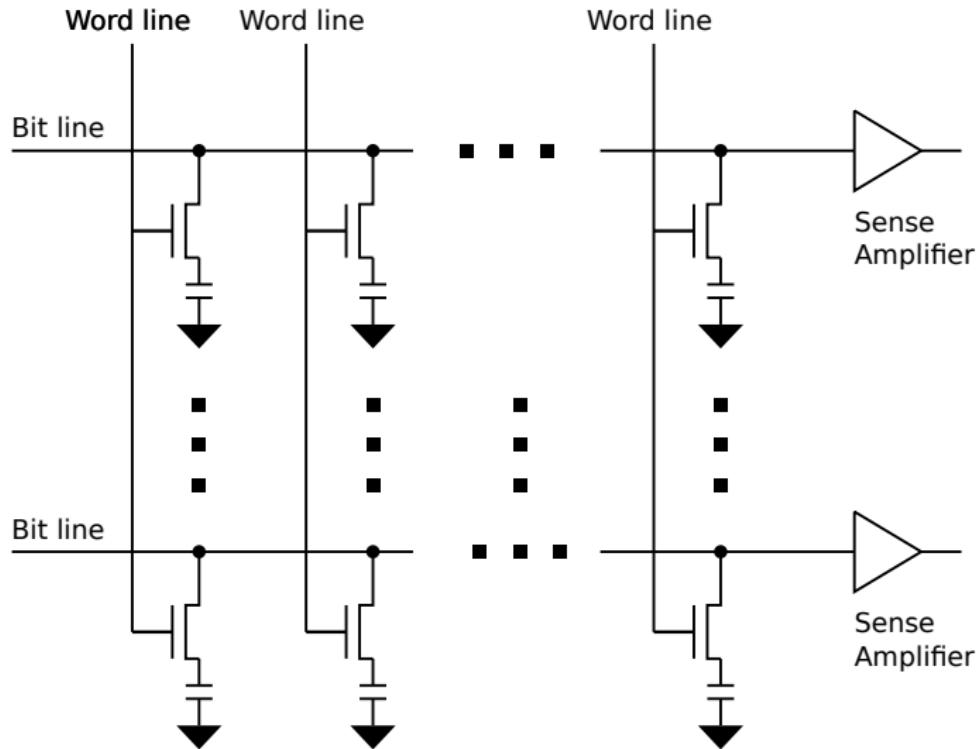
$$\frac{\text{Spartan3A DSP}}{\text{Intel Corei7}} = \frac{97\text{ns}}{4\text{ns}} = 24$$

# Memoria externa

- La FPGA tiene memoria interna pero no es masiva
- Se requiere de un controlador de memoria DRAM
- El controlador debe sostener la velocidad de transferencia de datos entre la aplicación y la memoria DRAM

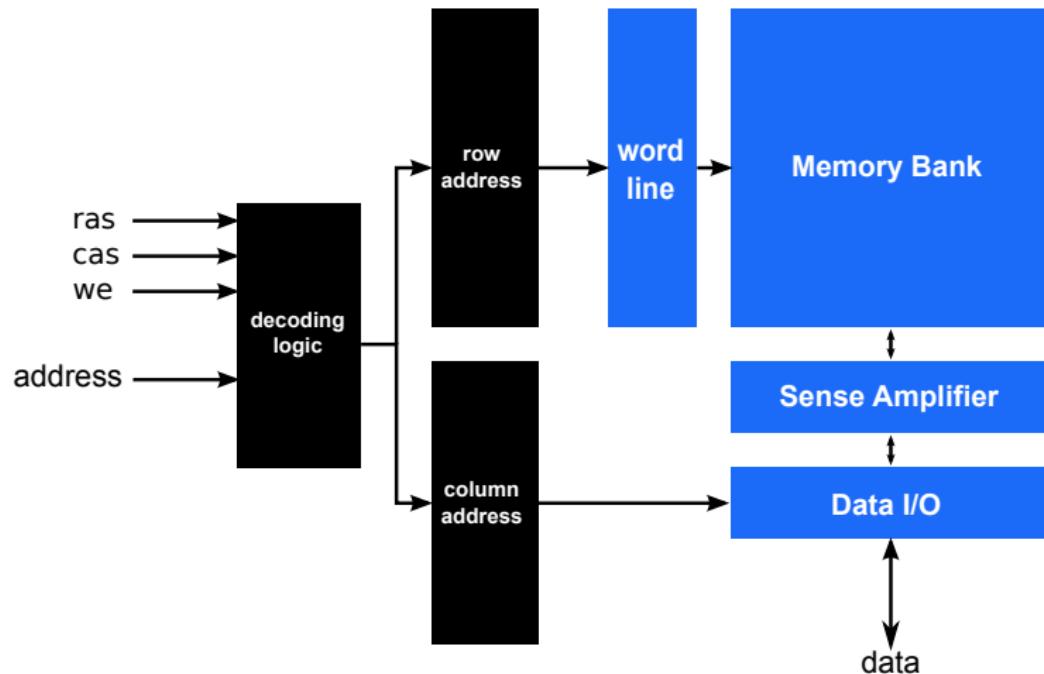


# Banco de Memoria de un solo transistor

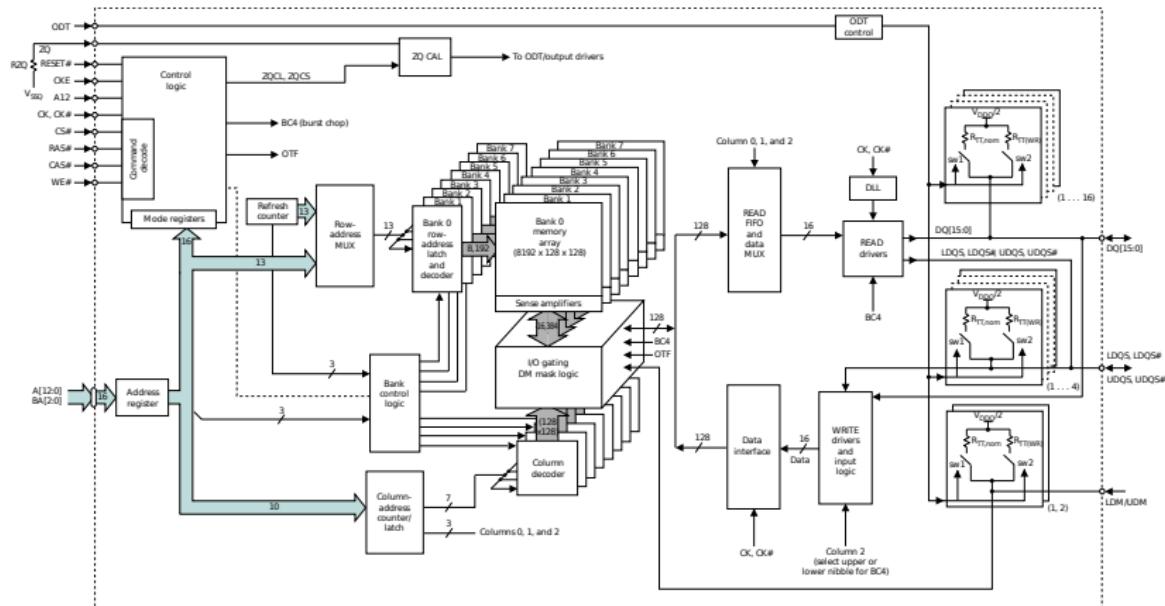


# Diagrama en Bloques

MOSTEK DRAM

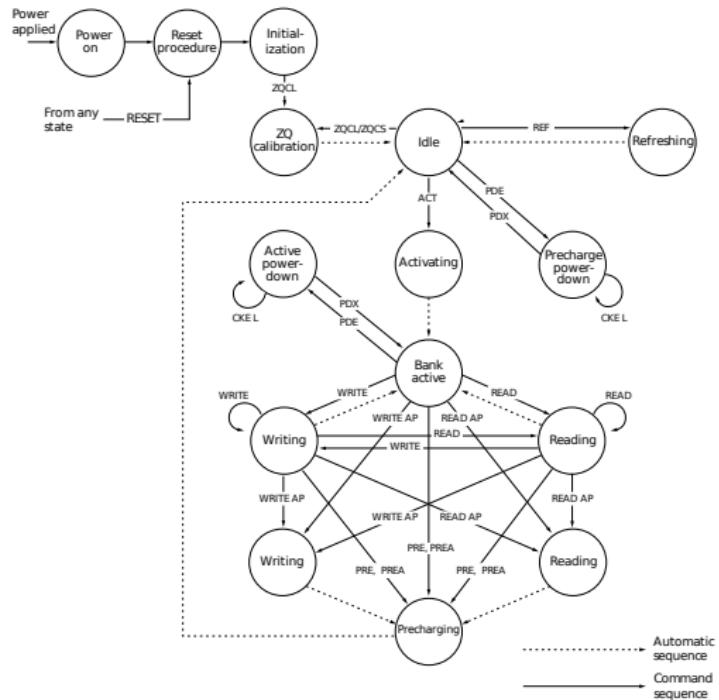


# Diagrama en bloque detallado de un chip de DDR3 DRAM



# Diagrama de estados

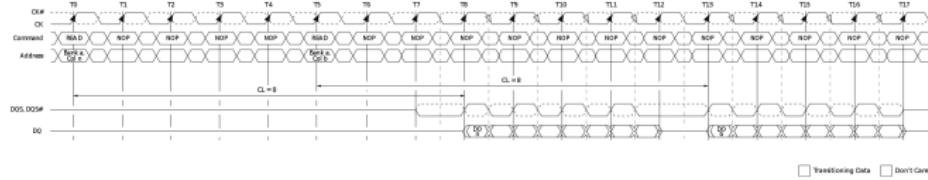
## Especificación DDR3



# Diagrama de tiempos

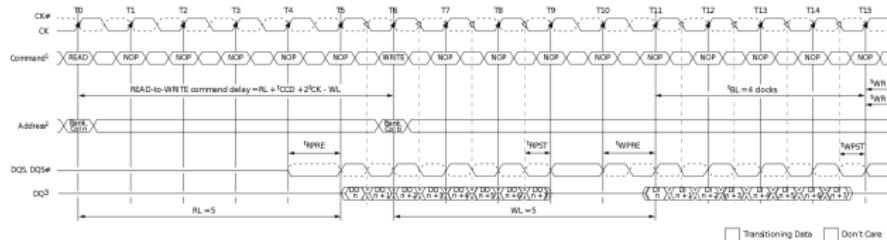
## Especificación DDR3

### Nonconsecutive READ Bursts



- Notes:
1. AL = 0, RL = 8.
  2. DO n (or b) = data-out from column n (or column b).
  3. Seven subsequent elements of data-out appear in the programmed order following DO n.
  4. Seven subsequent elements of data-out appear in the programmed order following DO b.

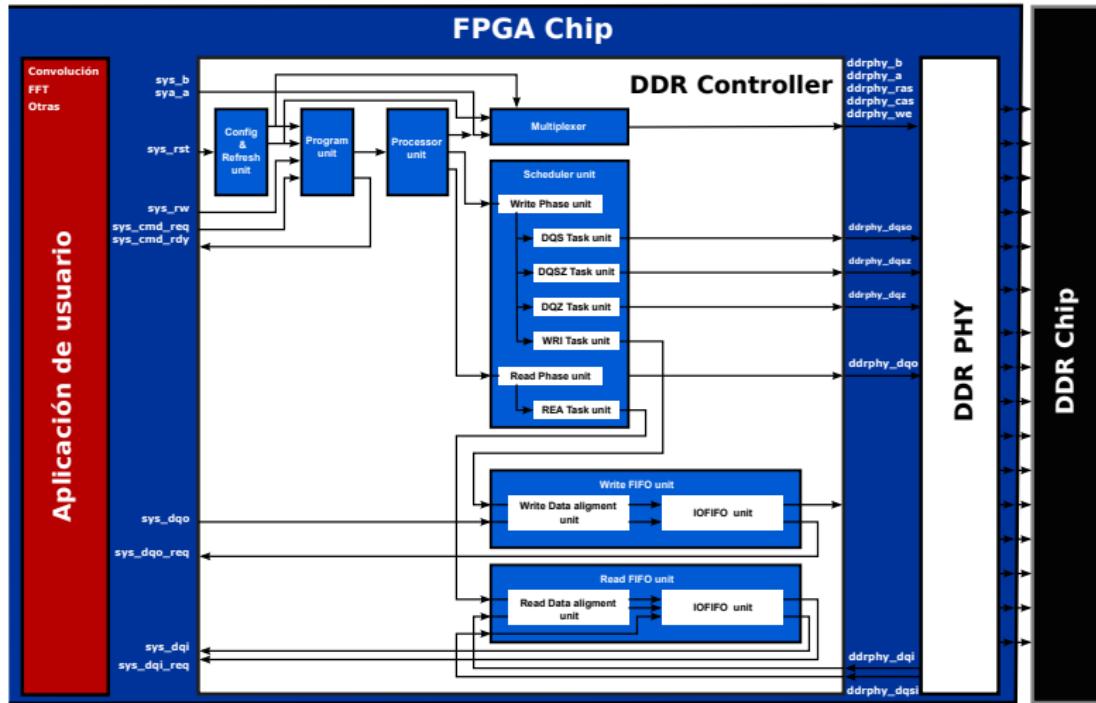
### READ (BL8) to WRITE (BL8)



- Notes:
1. NOP commands are shown for ease of illustration; other commands may be valid at these times.
  2. The BL8 setting is activated by either MRO[1:0] = 00 or MRO[1:0] = 01 and A12 = 1 during the READ command at T0, and the WRITE command at T6.
  3. DO n = data-out from column, DI b = data-in for column b.
  4. BL8, RL = 5 (AL = 0, CL = 5), WL = 5 (AL = 0, CWL = 5).



# Arquitectura del controlador de DDR DRAM obtenido para FPGA

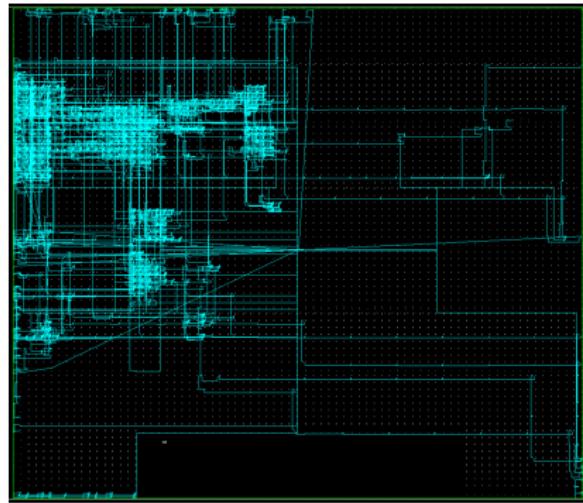


# Constraste con fabricantes

Familia	Spartan 3A DPS	Virtex 5	LatticeECP3
Chip	xc3sd1800a-4cs484	xc5vlx110t-1ff1136	LFE3-35EA-7FN484C
Fabricante	Xilinx	Xilinx	LatticeSemi
Escala de integración	90 nm	65 nm	65 nm
Grado	4	1	8
Tipo de DRAM	DDR1	DDR2	DDR3
<b>Módulo del fabricante</b>			
Velocidad MT/s	266	533	800
Código portable	no	no	cerrado
<b>Módulo obtenido</b>			
Velocidad MT/s	333	650	1000
Código portable	sí	sí	sí
<b>Aumento de velocidad</b>	25%	22%	25%

# Síntesis

LatticeECP3 Versa Kit



# Conclusiones

- Código Portable.
- Descripción a nivel de compuerta en VHDL.
- Más de un 20% de aumento de velocidad de transferencia comparado contra el módulo de los respectivos fabricantes.