L12: Reconfigurable Logic Architectures

Acknowledgements:

➢ Lecture material adapted from R. Katz, G. Borriello, “Contemporary Logic Design” (second edition), Copyright 2005 Prentice-Hall/Pearson Education.

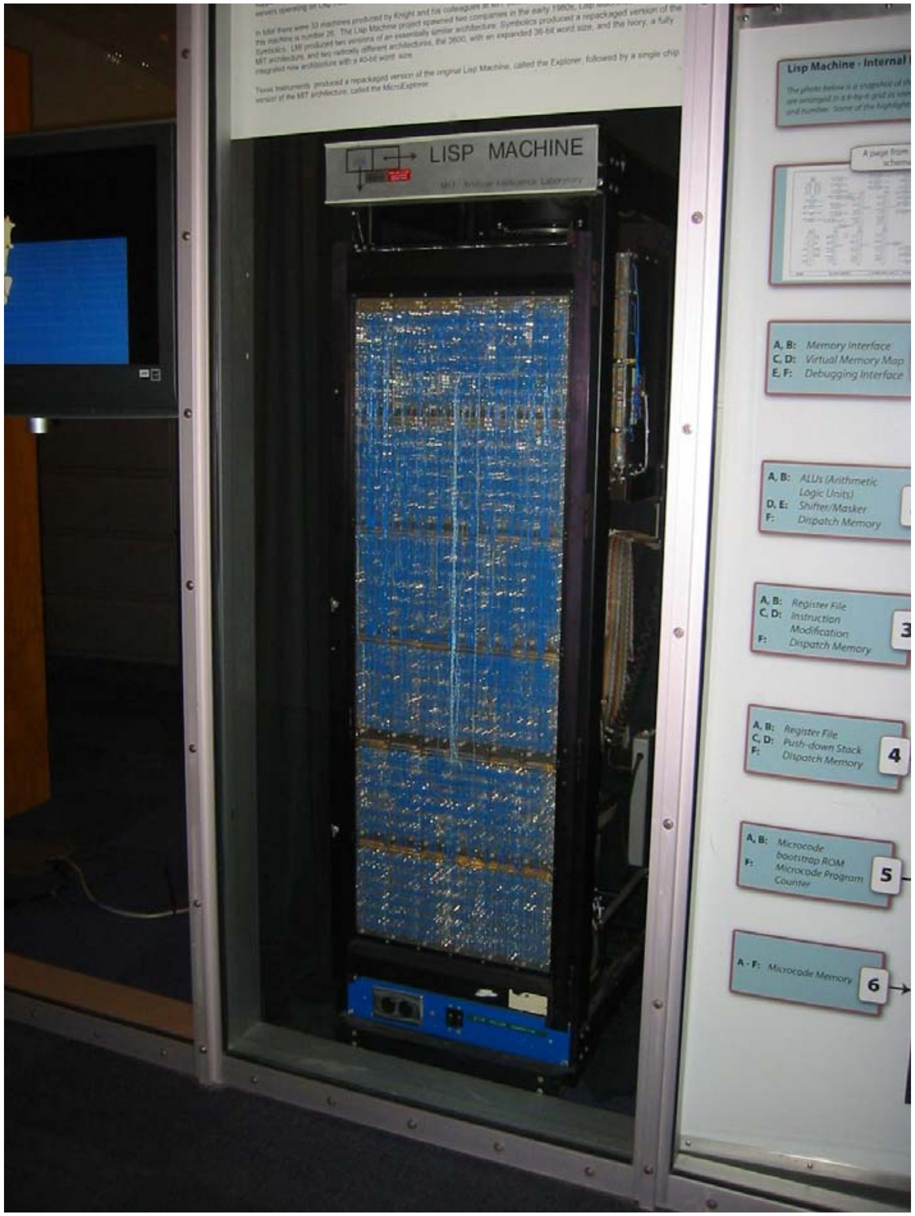
➢ Frank Honore

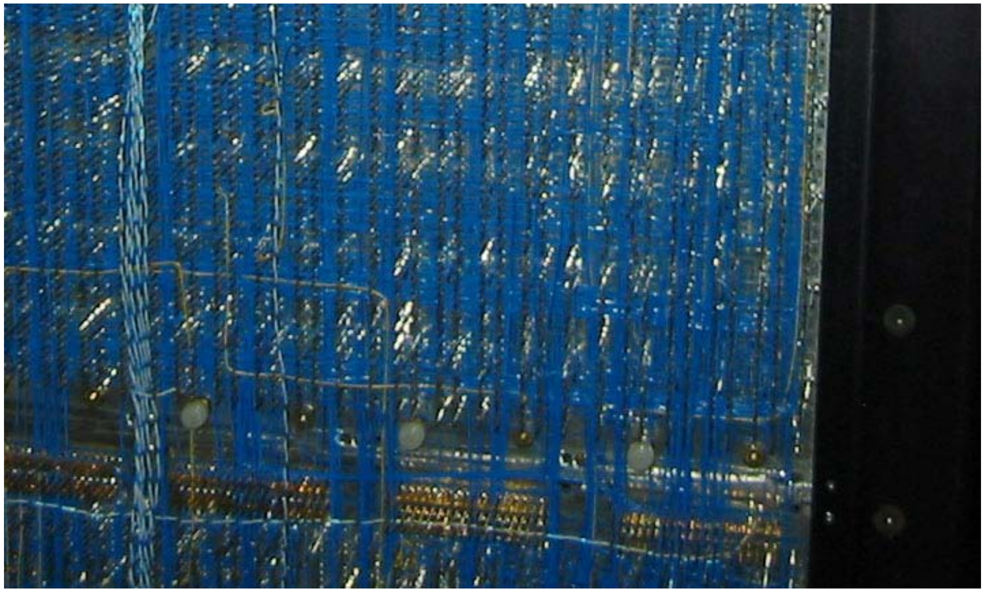
➢Lecture Notes prepared by Professor Anantha Chandrakasan

History of Computational Fabrics

- Discrete devices: relays, transistors (1940s-50s)
- Discrete logic gates (1950s-60s)
- Integrated circuits (1960s-70s)
  - □ e.g. TTL packages: Data Book for 100's of different parts
- Gate Arrays (IBM 1970s)
  - □ Transistors are pre-placed on the chip & Place and Route software puts the chip together automatically – only program the interconnect (mask programming)
- Software Based Schemes (1970's- present)
  - □ Run instructions on a general purpose core
- Programmable Logic (1980's to present)
  - □ A chip that be reprogrammed after it has been fabricated
  - □ Examples: PALs, EPROM, EEPROM, PLDs, FPGAs
  - □ Excellent support for mapping from Verilog
- ASIC Design (1980's to present)
  - □ Turn Verilog directly into layout using a library of standard cells
  - □ Effective for high-volume and efficient use of silicon area

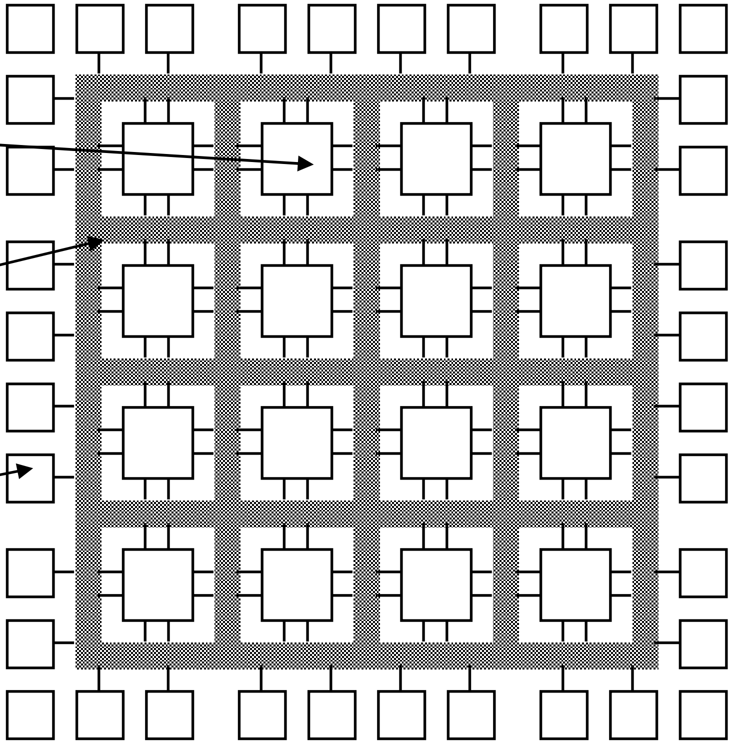
Lisp Machine



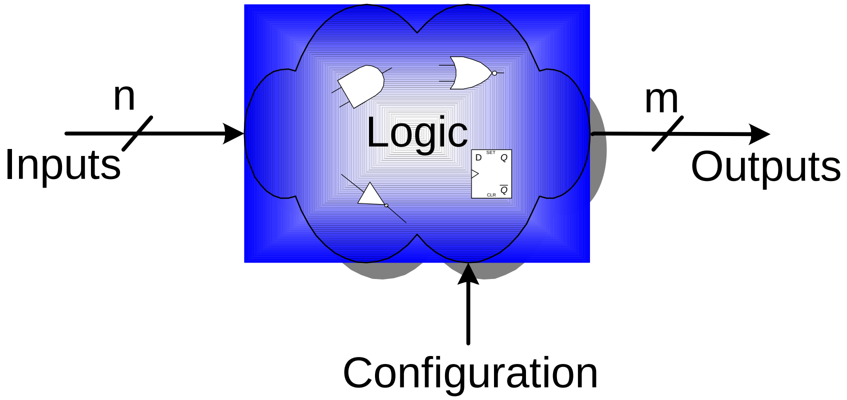


Reconfigurable Logic

- **Logic blocks**
  - □ **To implement combinational and sequential logic**
- **Interconnect**
  - □ **Wires to connect inputs and outputs to logic blocks**
- **I/O blocks**
  - □ **Special logic blocks at periphery of device for external connections**

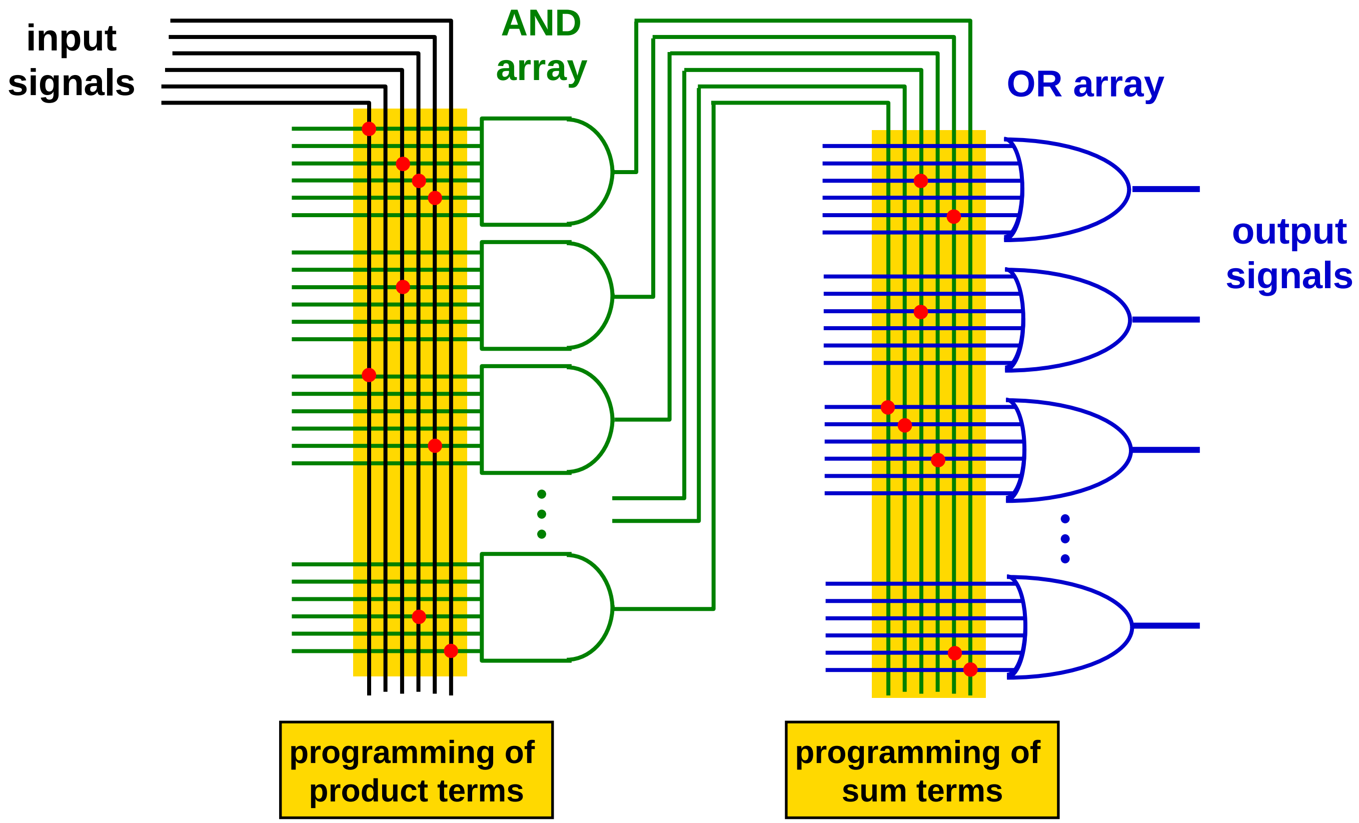


- **Key questions:**
  - □ **How to make logic blocks programmable? (after chip has been fabbed!)**
  - □ **What should the logic granularity be?**
  - □ **How to make the wires programmable? (after chip has been fabbed!)**
  - □ **Specialized wiring structures for local vs. long distance routes?**
  - □ **How many wires per logic block?**



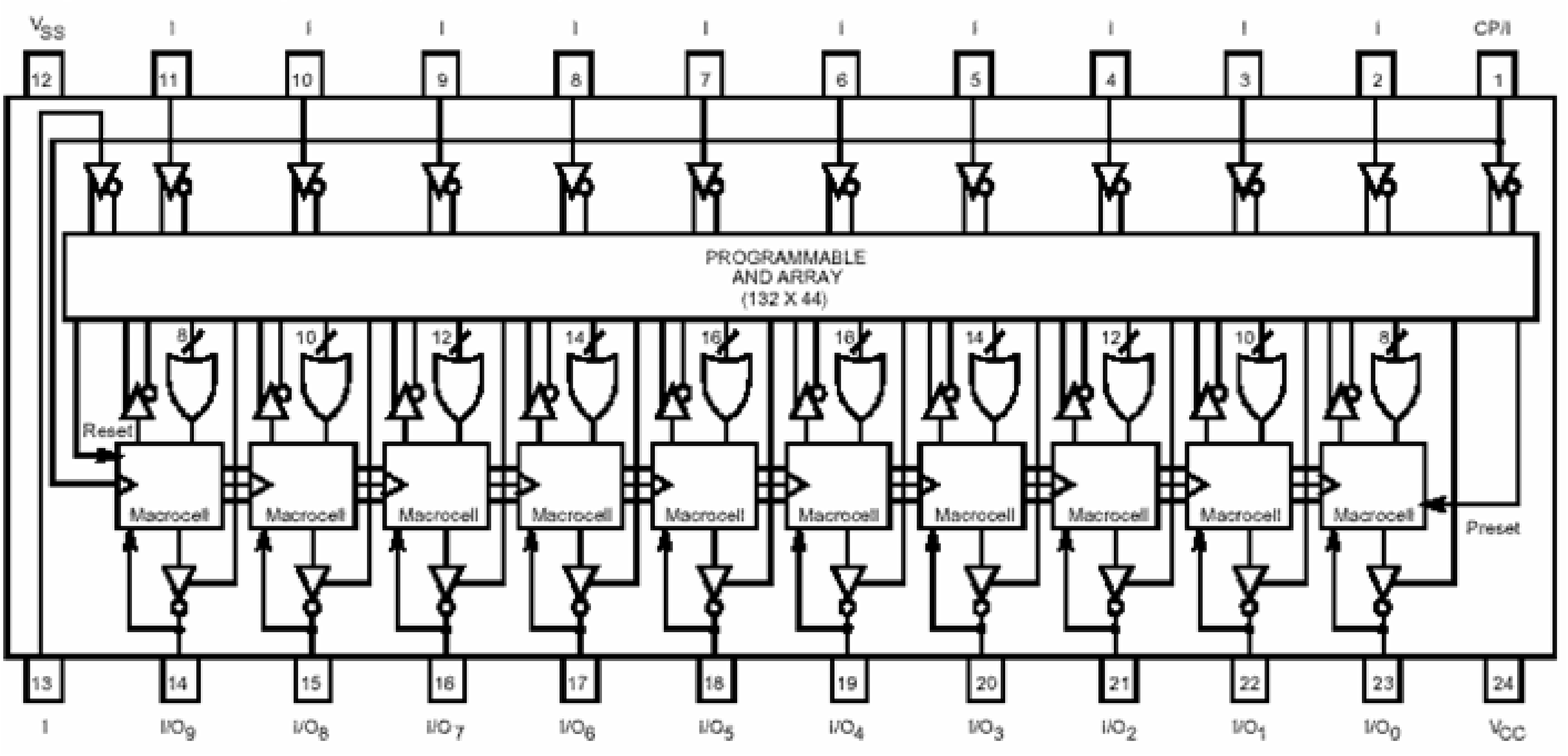
Programmable Array Logic (PAL)

- Based on the fact that any combinational logic can be realized as a sum-of-products
- PALs feature an array of AND-OR gates with programmable interconnect



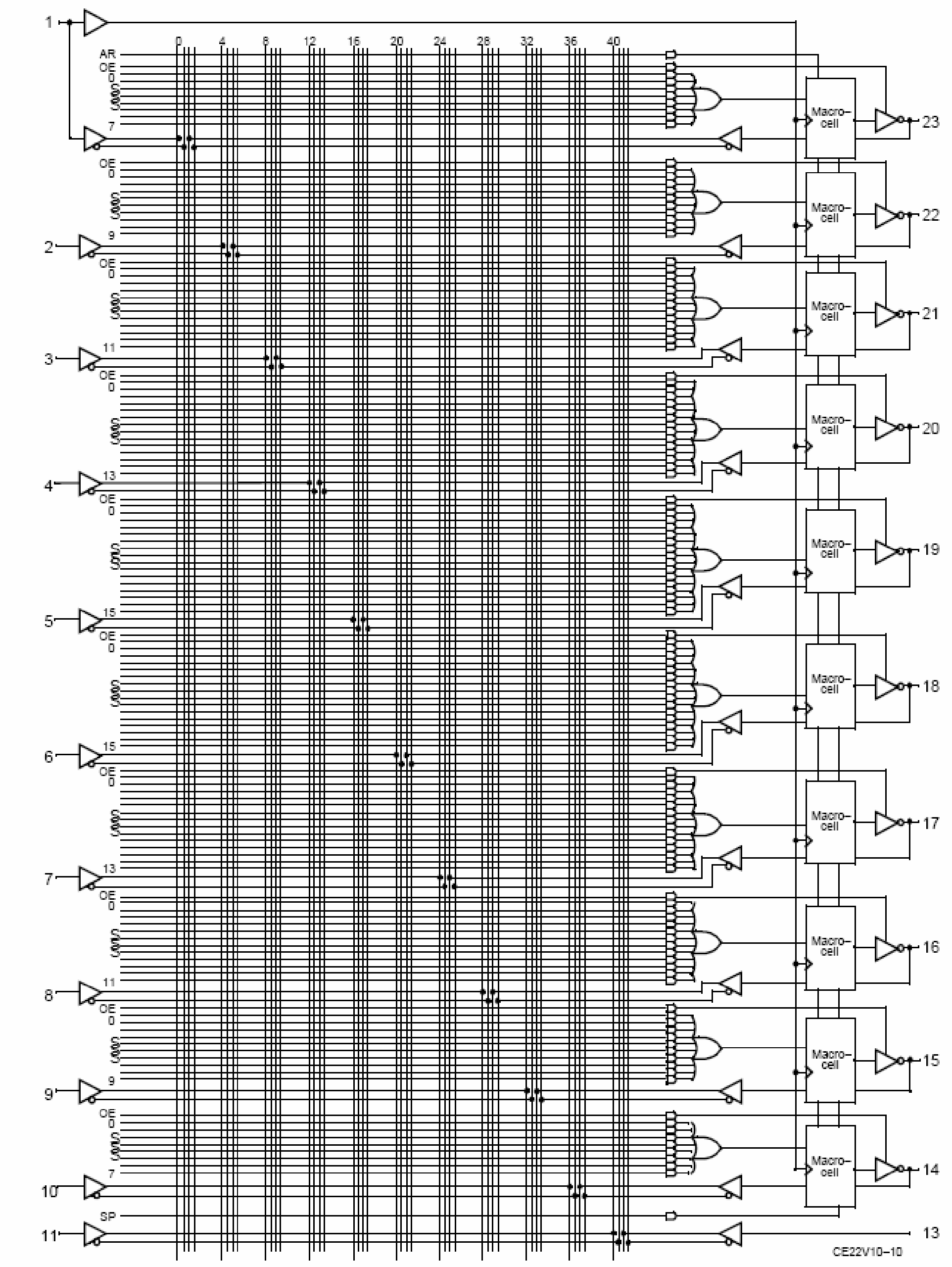
Inside the 22v10 PAL

- Each input pin (and its complement) sent to the AND array
- OR gates for each output can take 8-16 product terms, depending on output pin
- “Macrocell” block provides additional output flexibility...

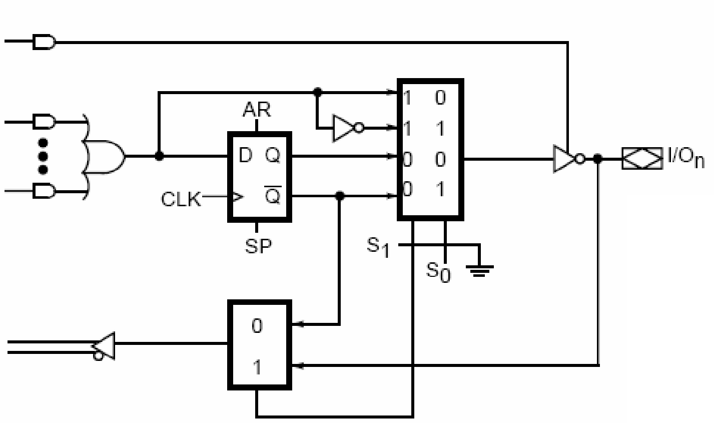


Cypress PAL CE22V10

Functional Logic Diagram for PALCE22V10 **From Cypress**



From Lattice Semiconductor

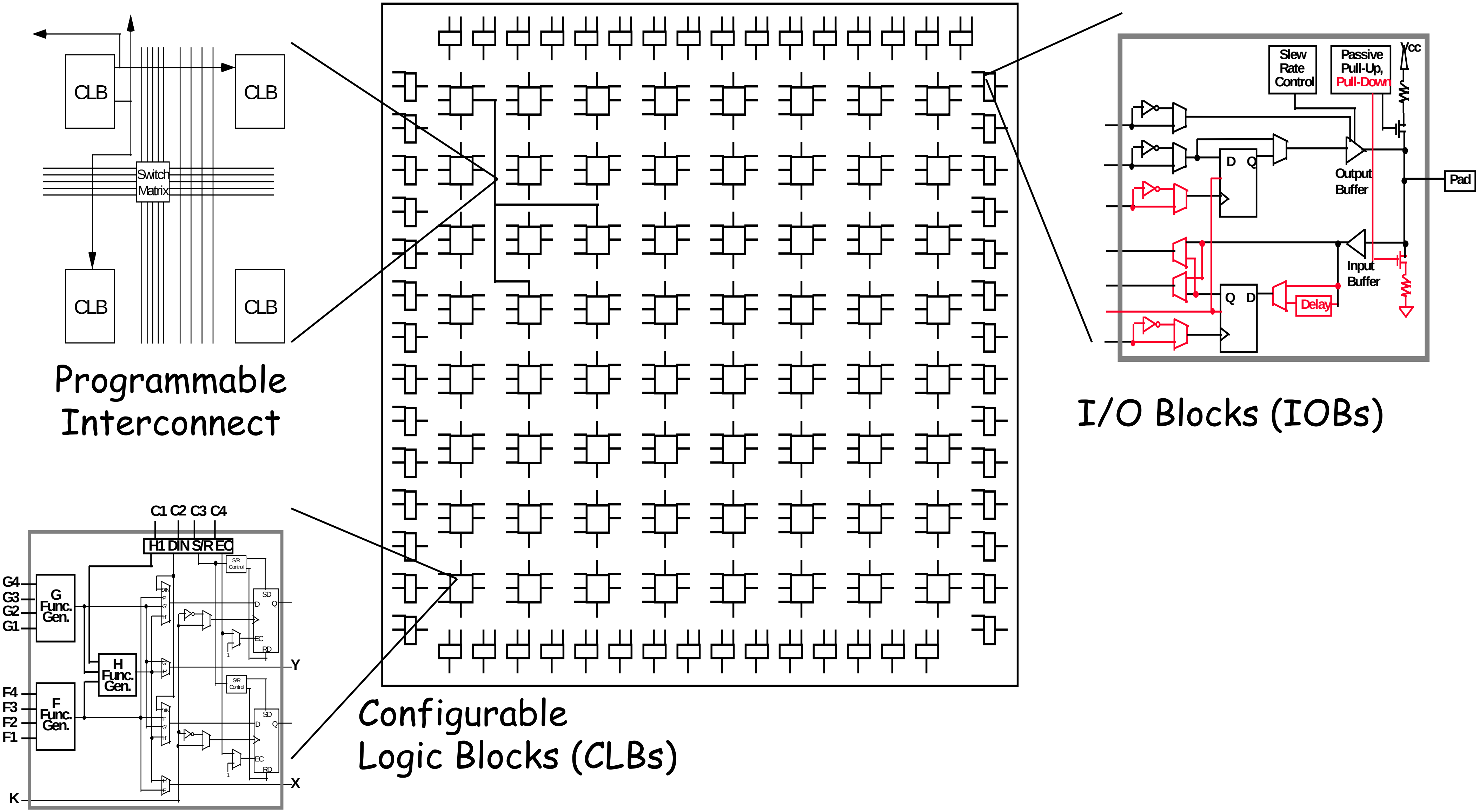


S_1	S_0	Output Configuration
0	0	Registered/Active Low
0	1	Registered/Active High
1	0	**Combinational/active low**
1	1	**Combinational/active high**

0 = Programmed EE bit
1 = Erased (charged) EE bit

- **Outputs may be registered or combinational, positive or inverted**

RAM Based Field Programmable Logic - Xilinx

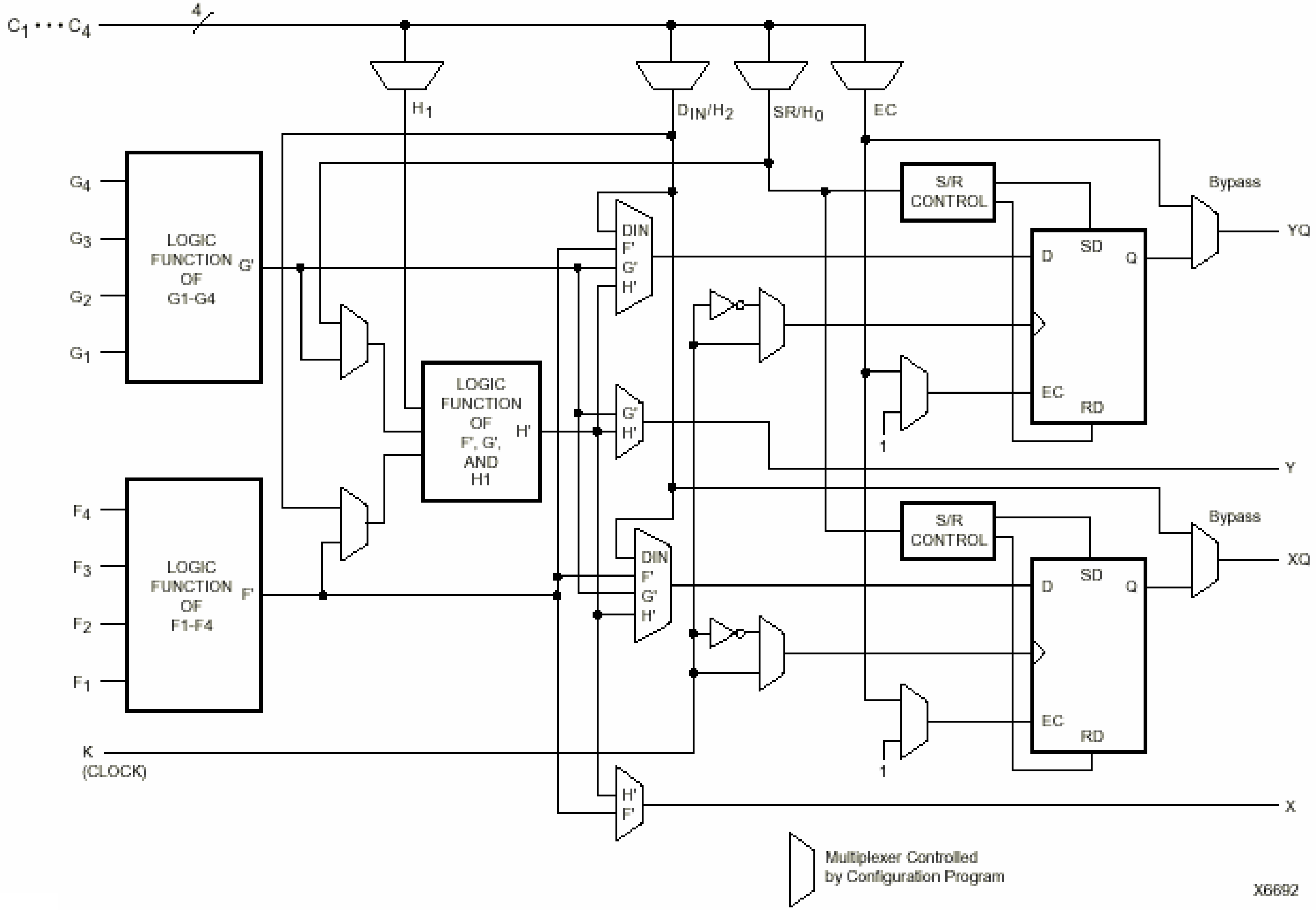


Programmable Interconnect

I/O Blocks (IOBs)

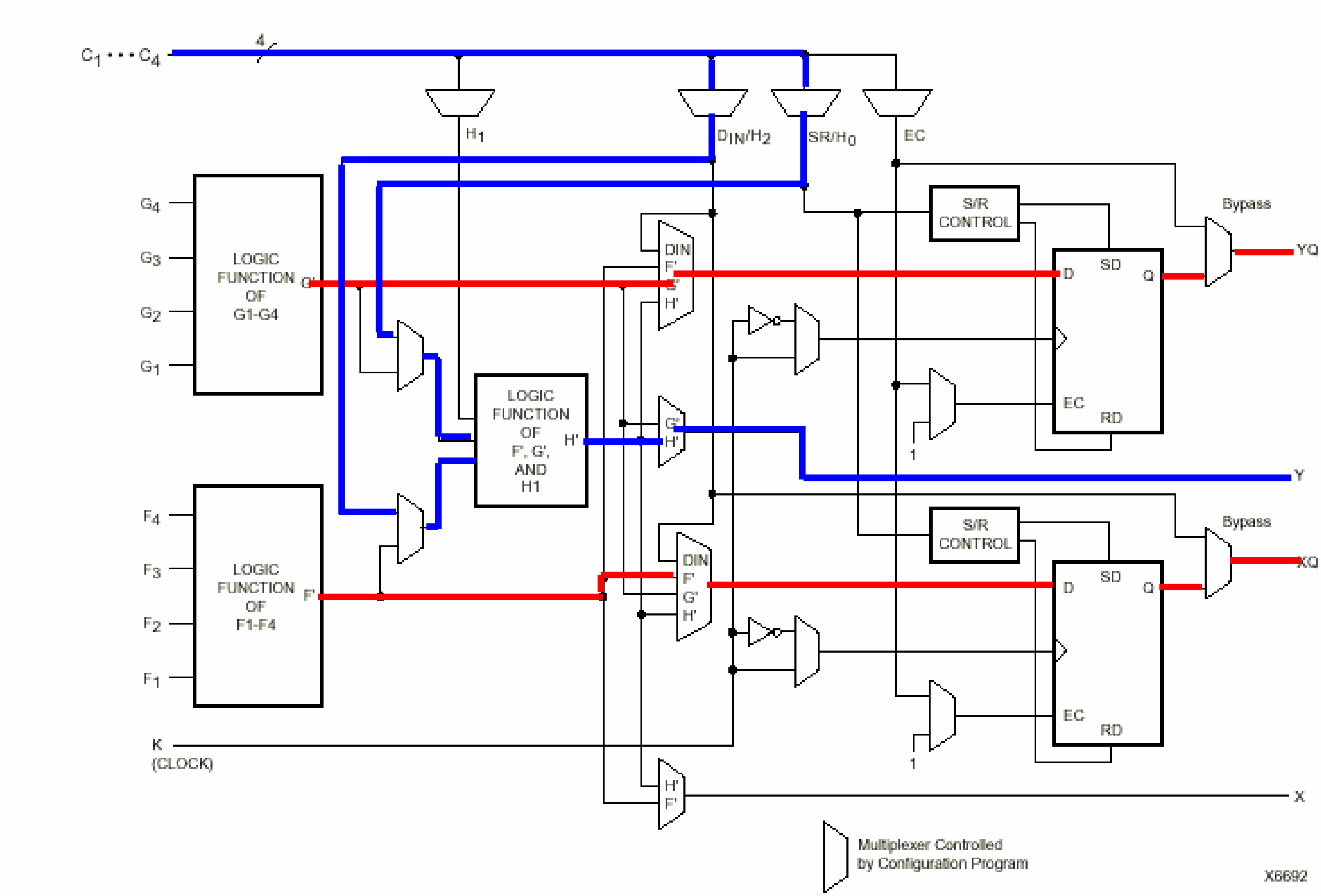
Configurable Logic Blocks (CLBs)

The Xilinx 4000 CLB



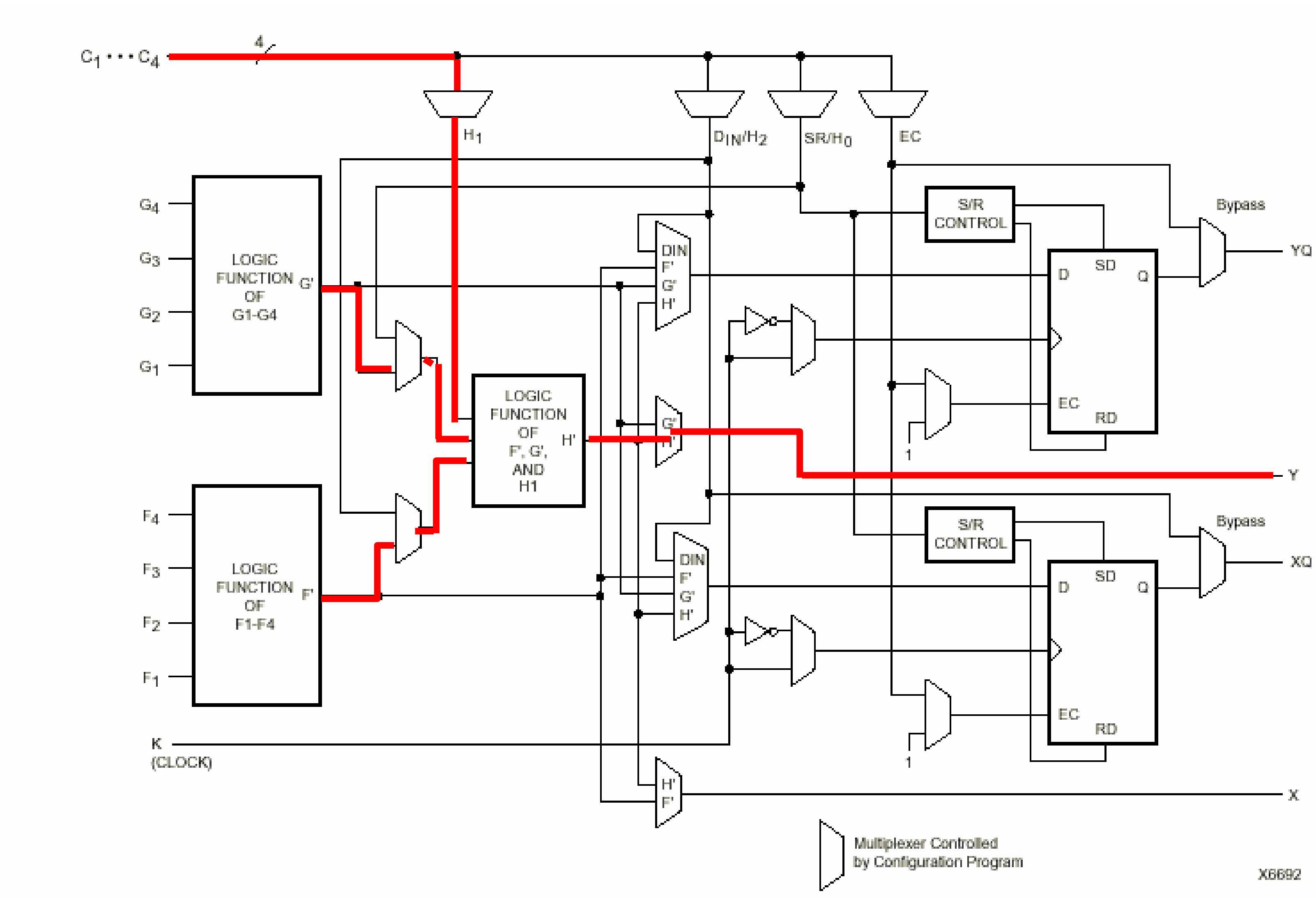
Simplified Block Diagram of XC4000 Series CLB (RAM and Carry Logic functions not shown)

Two 4-input Functions, Registered Output and a Two Input Function



Simplified Block Diagram of XC4000 Series CLB (RAM and Carry Logic functions not shown)

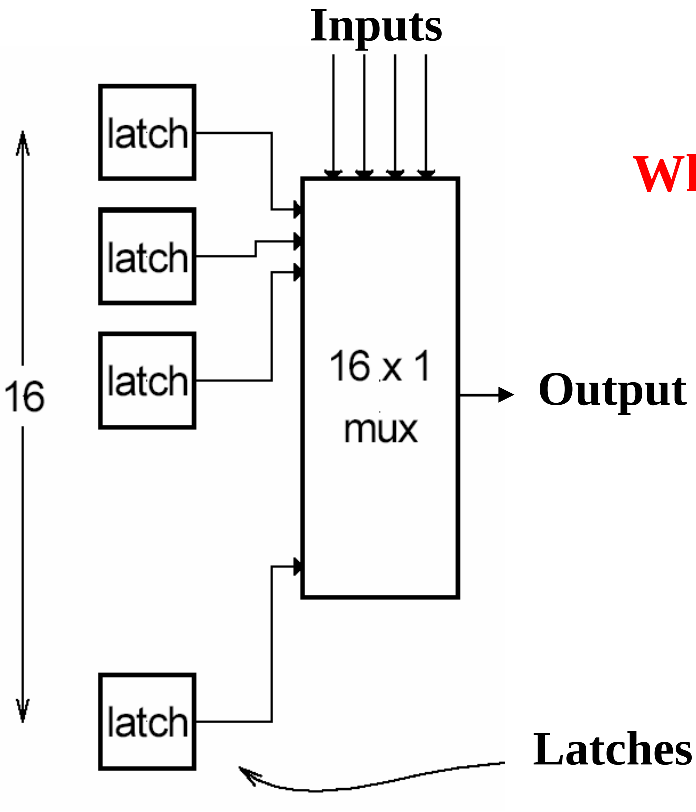
5-input Function, Combinational Output



Simplified Block Diagram of XC4000 Series CLB (RAM and Carry Logic functions not shown)

LUT Mapping

- N-LUT direct implementation of a truth table: any function of n-inputs.
- N-LUT requires 2^N storage elements (latches)
- N-inputs select one latch location (like a memory)

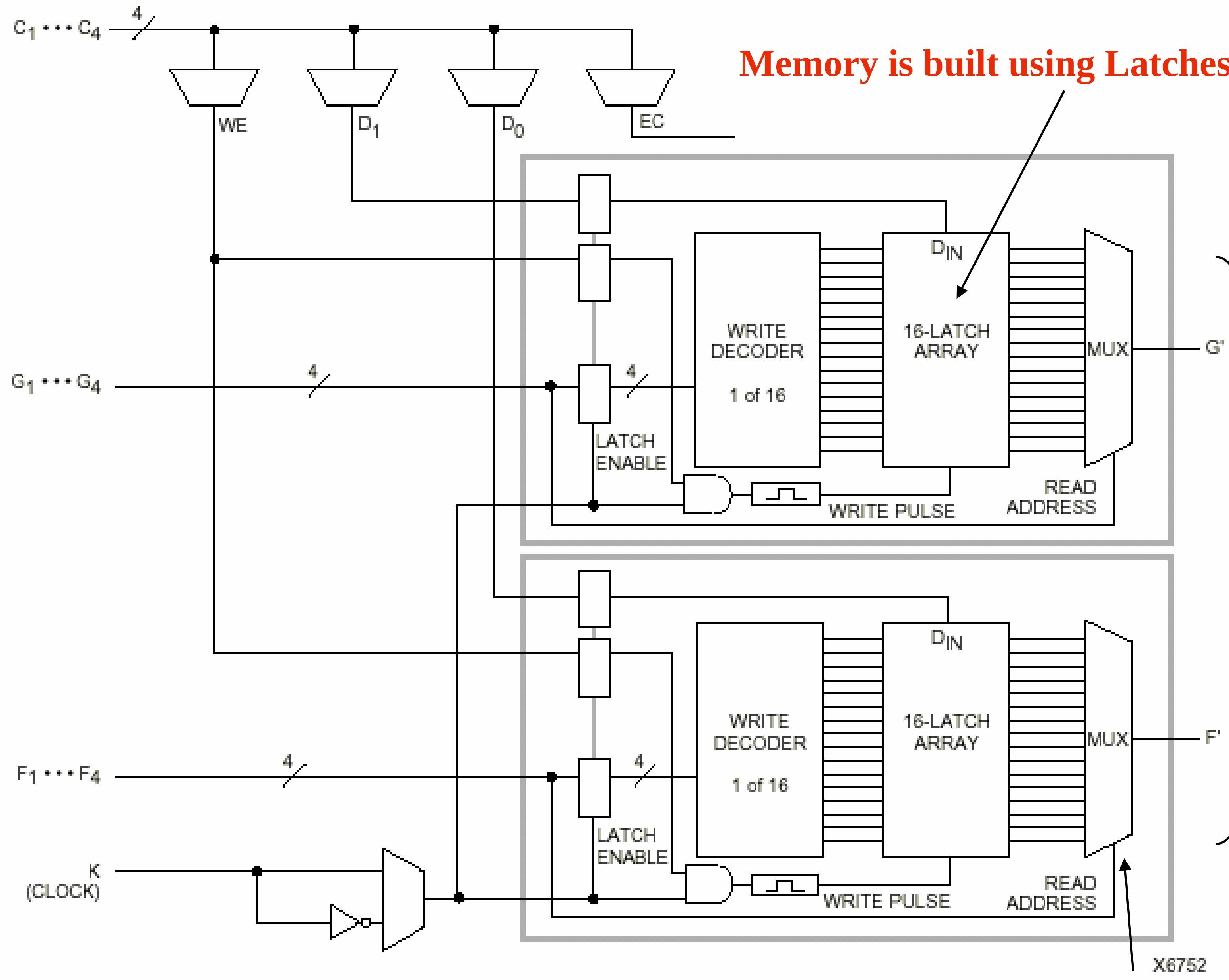


Why Latches and Not Registers?

Latches set by configuration bitstream

4LUT example

Configuring the CLB as a RAM

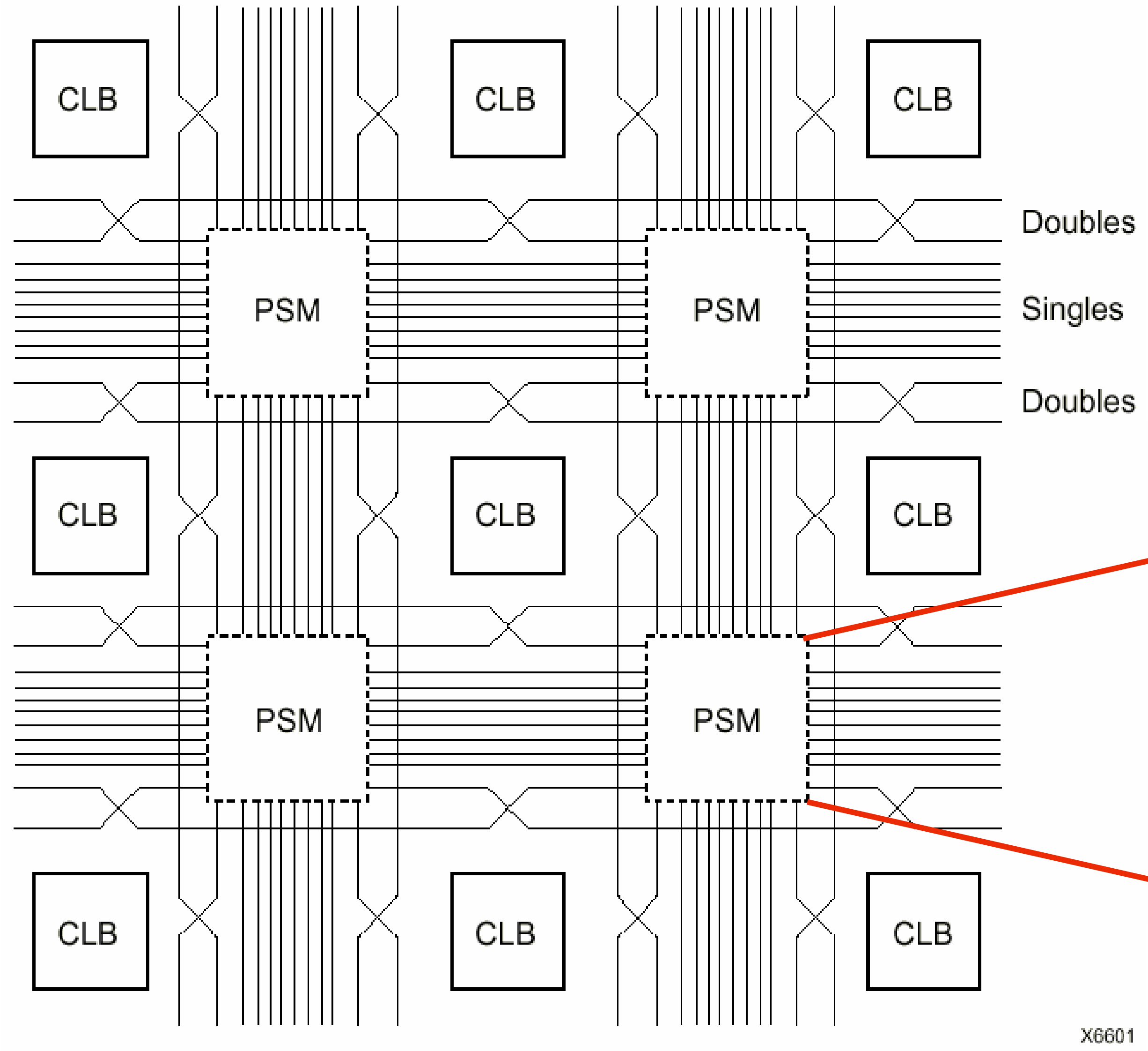


Memory is built using Latches not FFs

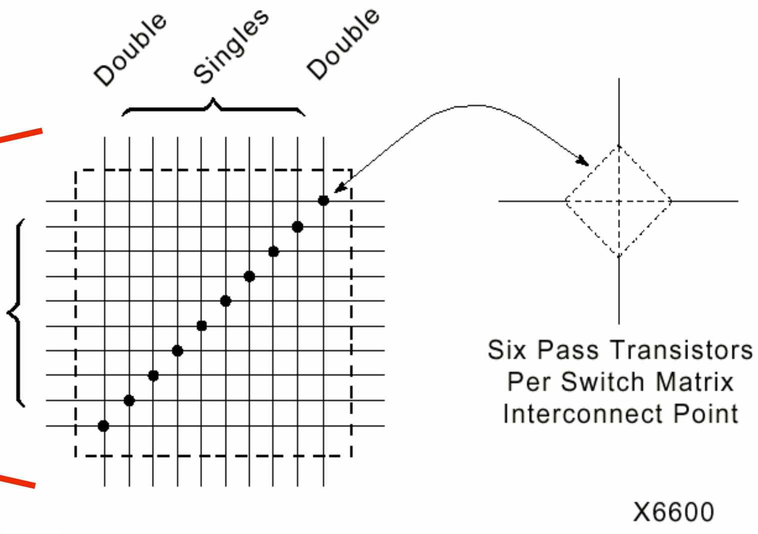
16x2

Read is same a LUT Function!

Xilinx 4000 Interconnect

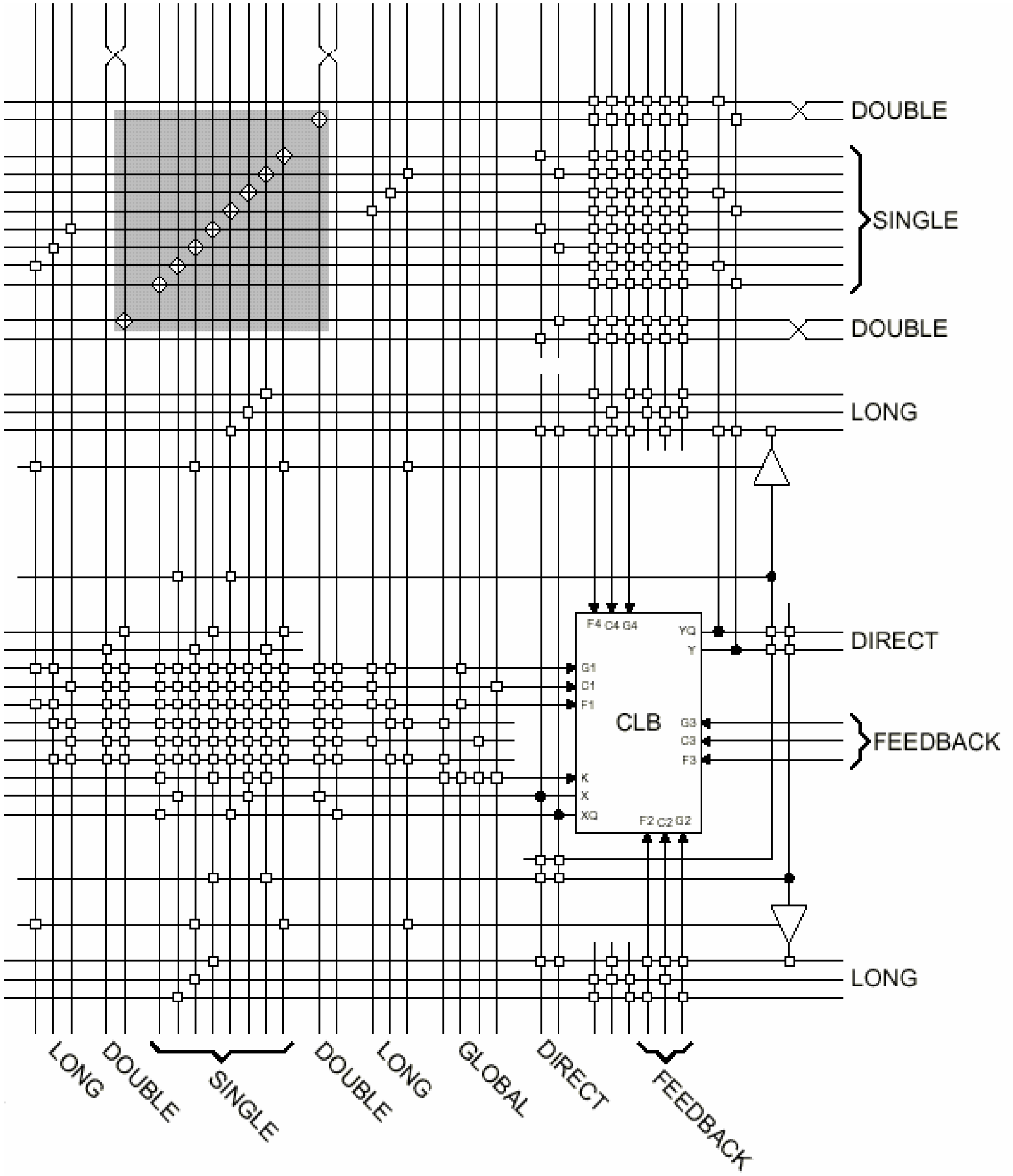


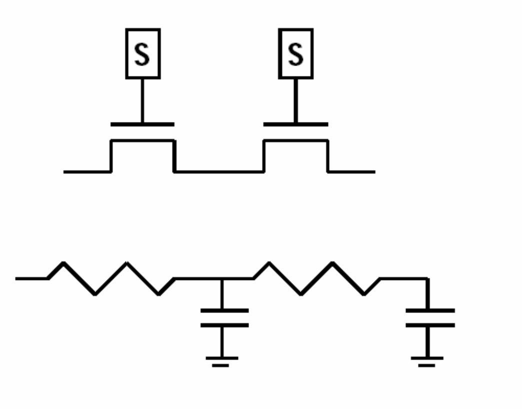
Single- and Double-Length Lines, with Programmable Switch Matrices (PSMs)



Programmable Switch Matrix (PSM)

Xilinx 4000 Interconnect Details



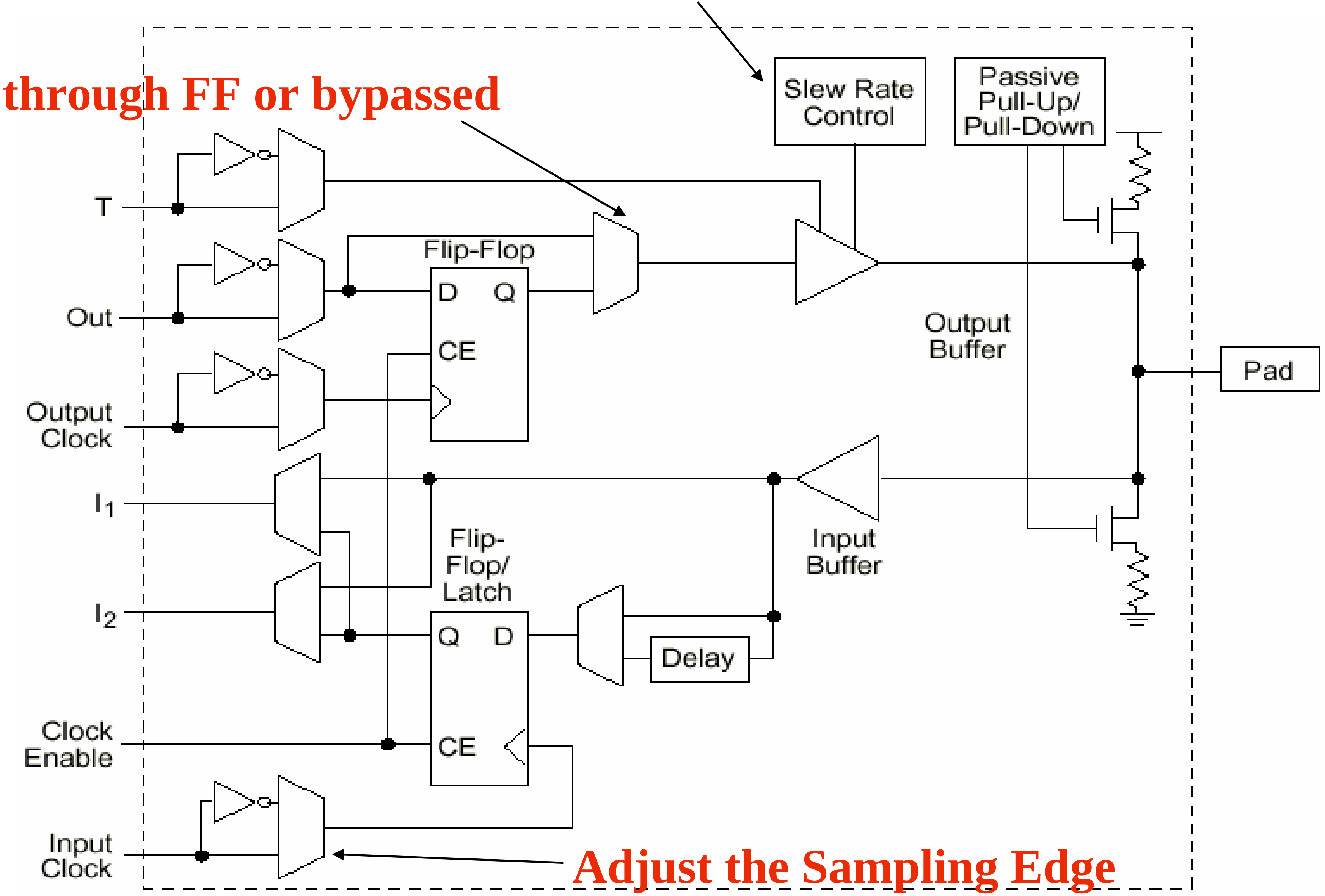


Wires are not ideal!

Xilinx 4000 Flexible IOB

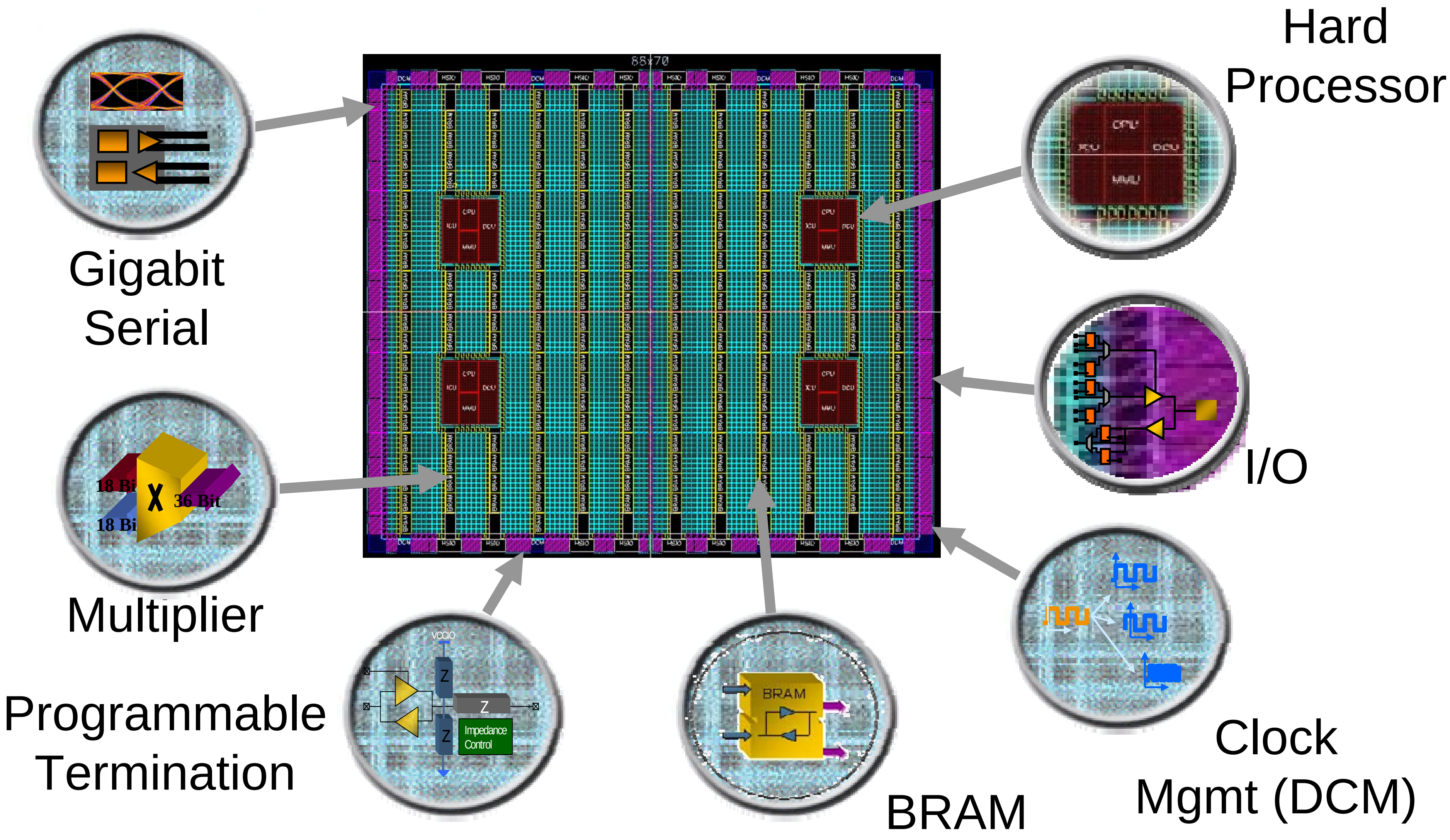
Adjust Transition Time

Outputs through FF or bypassed



Adjust the Sampling Edge

Add Bells & Whistles



Hard Processor

Gigabit Serial

I/O

Multiplier

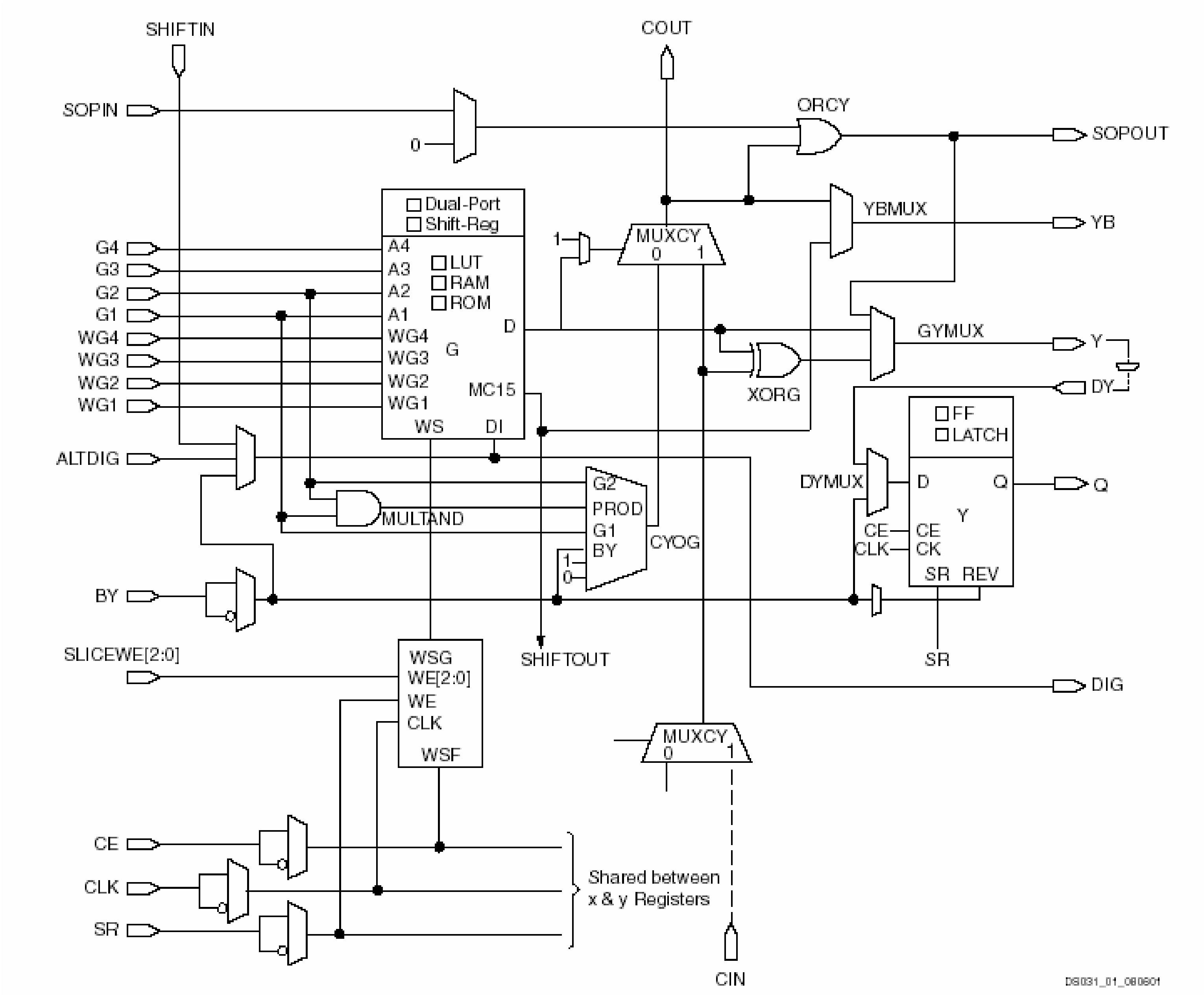
Programmable Termination

BRAM

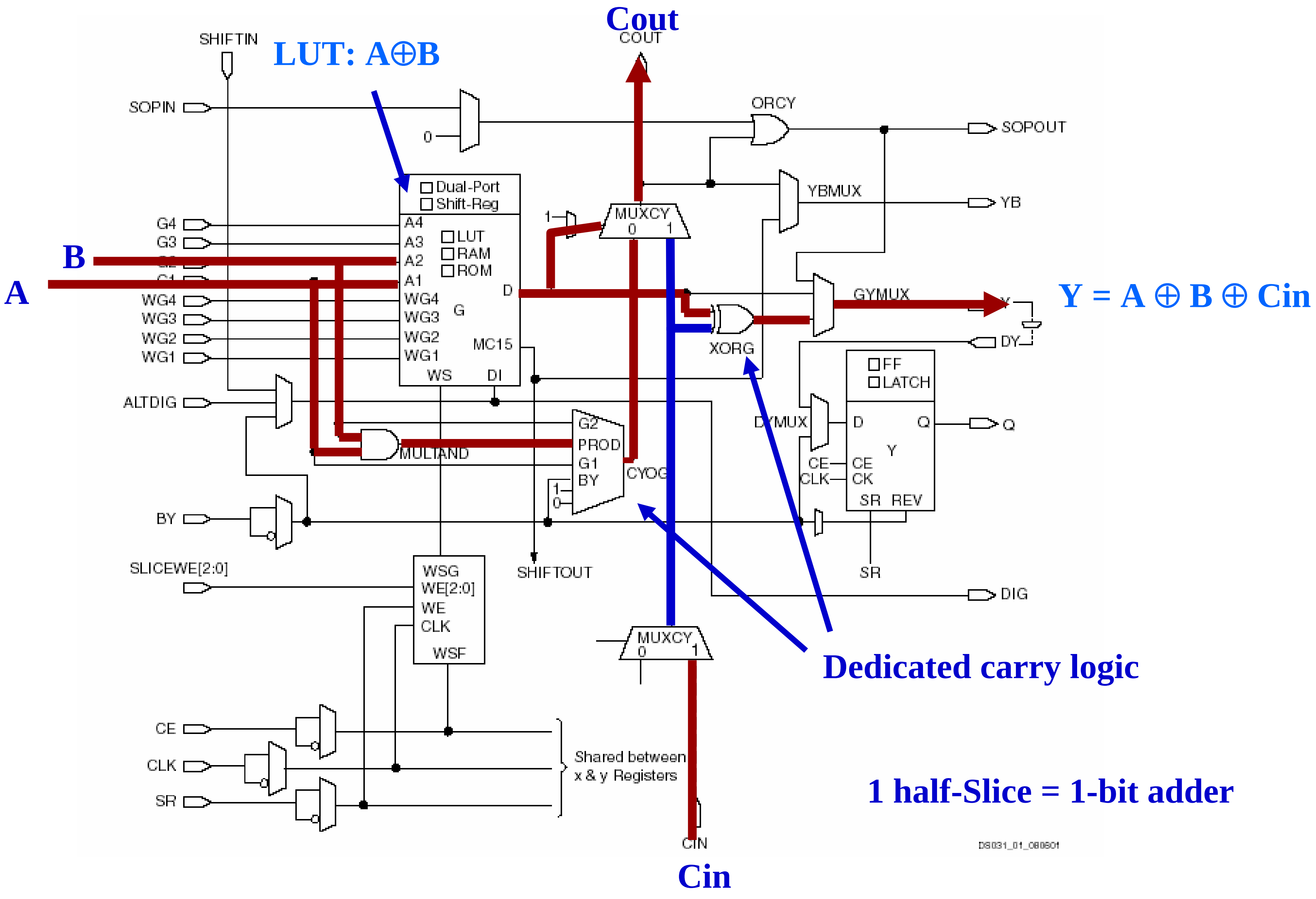
Clock Mgmt (DCM)

Courtesy of David B. Parlour, ISSCC 2004 Tutorial,
“The Reality and Promise of Reconfigurable Computing in Digital Signal Processing”

The Virtex II CLB (Half Slice Shown)



Adder Implementation



Cout

LUT: A⊕B

B

A

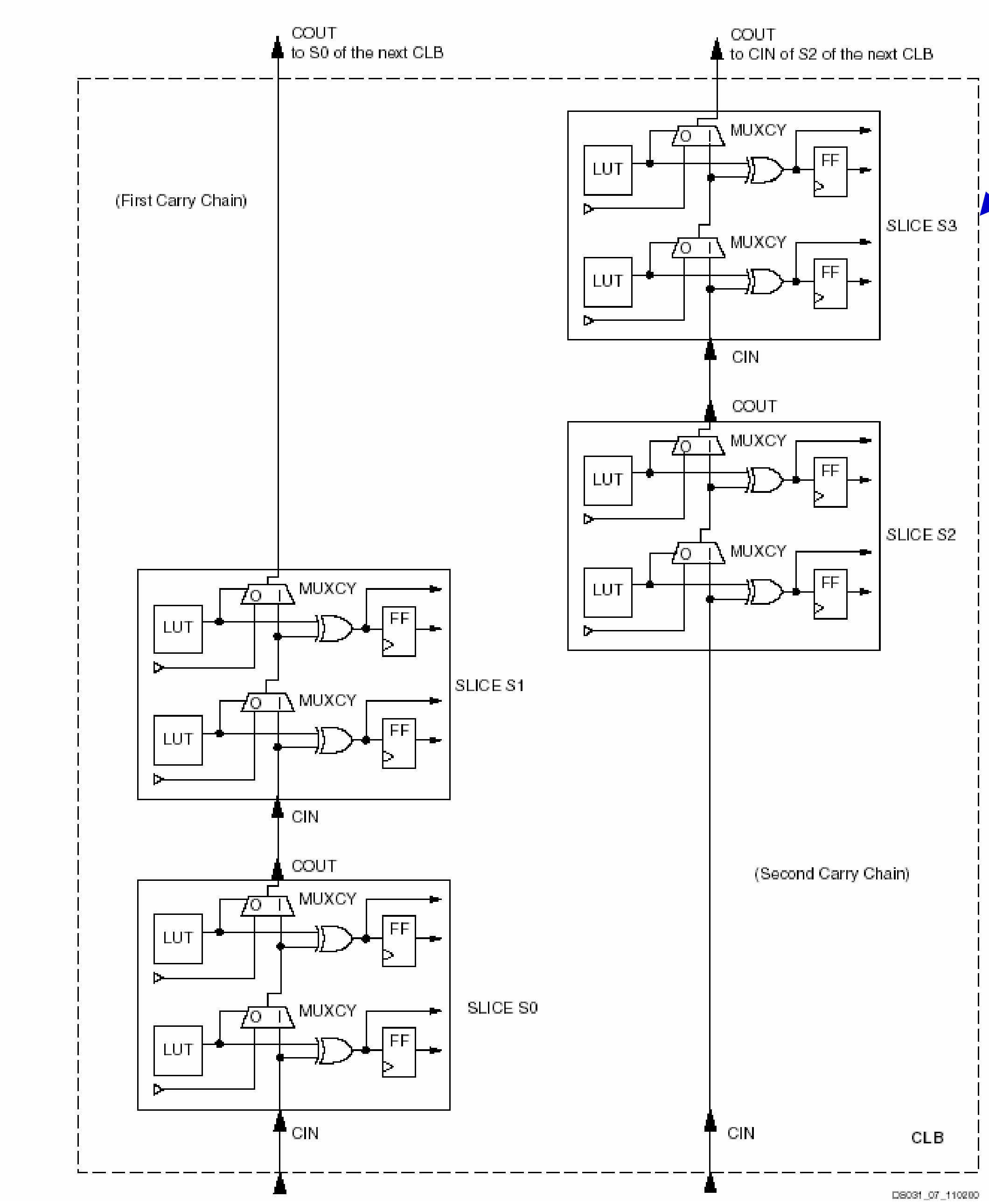
Y = A ⊕ B ⊕ Cin

Dedicated carry logic

1 half-Slice = 1-bit adder

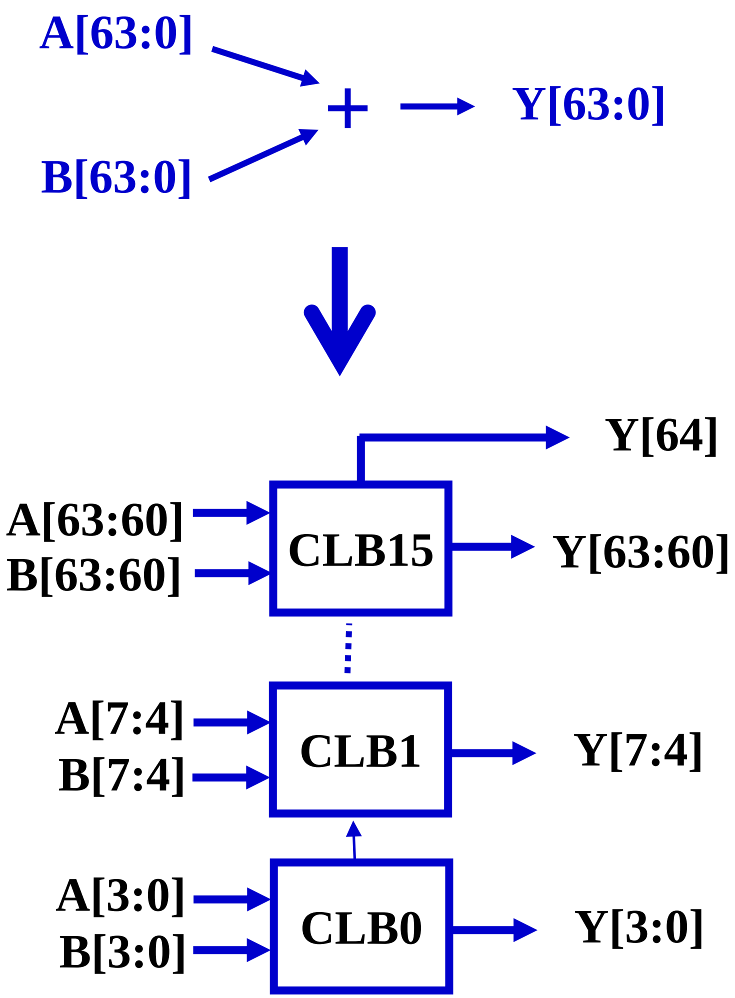
Cin

Carry Chain



1 CLB = 4 Slices = 2, 4-bit adders

64-bit Adder: 16 CLBs


A[63:0]

B[63:0]

+

Y[63:0]

Y[64]

A[63:60]

B[63:60]

CLB15

Y[63:60]

A[7:4]

B[7:4]

CLB1

Y[7:4]

A[3:0]

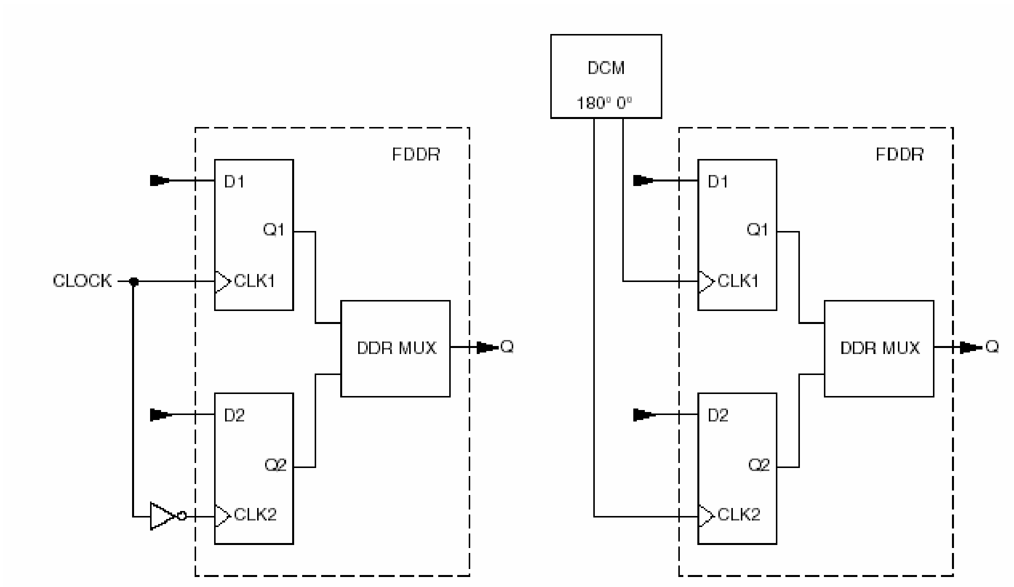
B[3:0]

CLB0

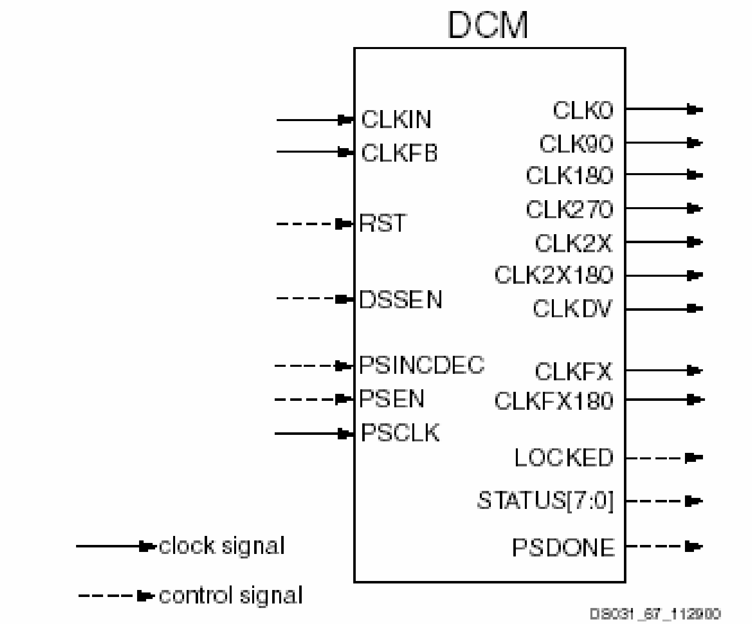
Y[3:0]

CLBs must be in same column

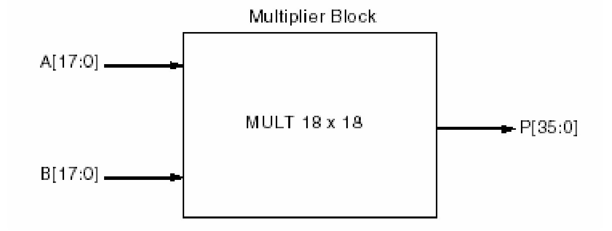
Virtex II Features



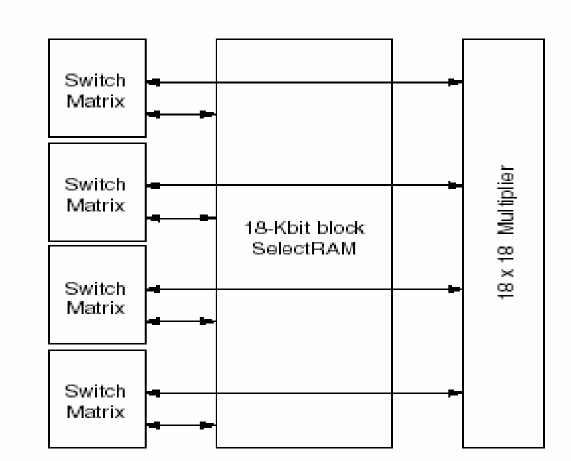
Double Data Rate registers



Digital Clock Manager



Embedded Multiplier

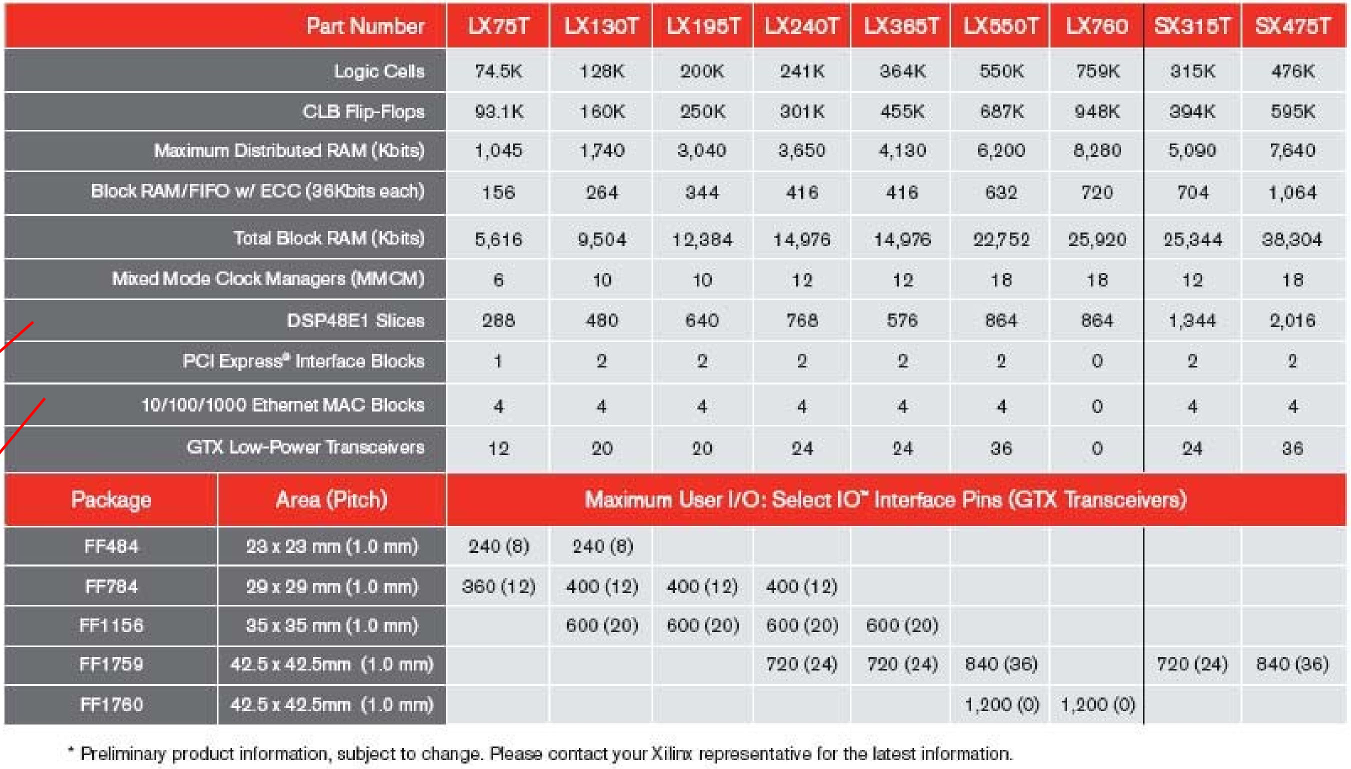


Block SelectRAM

The Latest Generation: Virtex-6

DSP with 25x18 multiplier

Gigabit ethernet support



Part Number	LX75T	LX130T	LX195T	LX240T	LX365T	LX550T	LX760	SX315T	SX475T
Logic Cells	74.5K	128K	200K	241K	364K	550K	759K	315K	476K
CLB Flip-Flops	93.1K	160K	250K	301K	455K	687K	948K	394K	595K
Maximum Distributed RAM (Kbits)	1,045	1,740	3,040	3,650	4,130	6,200	8,280	5,090	7,640
Block RAM/FIFO w/ ECC (36Kbits each)	156	264	344	416	416	632	720	704	1,064
Total Block RAM (Kbits)	5,616	9,504	12,384	14,976	14,976	22,752	25,920	25,344	38,304
Mixed Mode Clock Managers (MMCM)	6	10	10	12	12	18	18	12	18
DSP48E1 Slices	288	480	640	768	576	864	864	1,344	2,016
PCI Express® Interface Blocks	1	2	2	2	2	2	0	2	2
10/100/1000 Ethernet MAC Blocks	4	4	4	4	4	4	0	4	4
GTX Low-Power Transceivers	12	20	20	24	24	36	0	24	36

Package	Area (Pitch)	Maximum User I/O: Select IO™ Interface Pins (GTX Transceivers)								
FF484	23 x 23 mm (1.0 mm)	240 (8)	240 (8)							
FF784	29 x 29 mm (1.0 mm)	360 (12)	400 (12)	400 (12)	400 (12)					
FF1156	35 x 35 mm (1.0 mm)		600 (20)	600 (20)	600 (20)	600 (20)				
FF1759	42.5 x 42.5mm (1.0 mm)				720 (24)	720 (24)	840 (36)		720 (24)	840 (36)
FF1760	42.5 x 42.5mm (1.0 mm)						1,200 (0)	1,200 (0)		

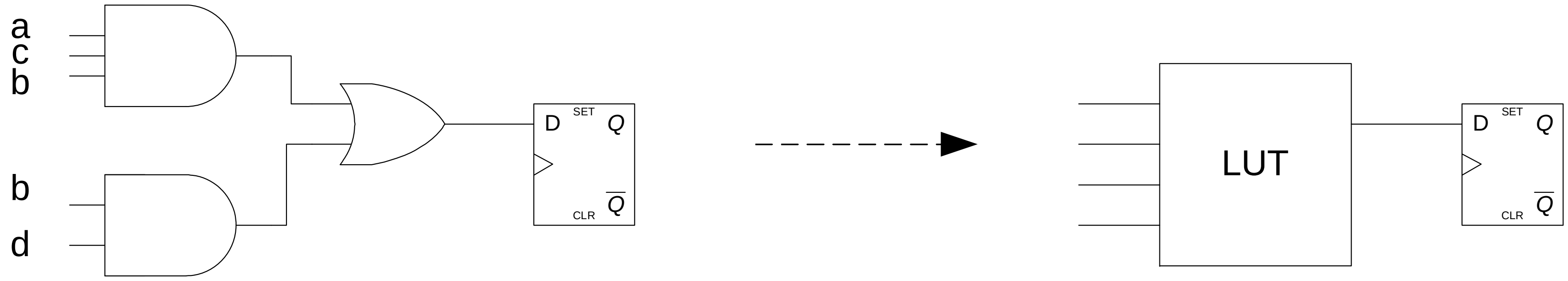
* Preliminary product information, subject to change. Please contact your Xilinx representative for the latest information.

	CLB	Dist RAM	Block RAM	Multipliers
Virtex 2*	8,448	1,056kbit	2,592kbit	144 (18x18)
Virtex 6*	667,000	6,200kbit	22,752kbit	1,344 (25x18)
Spartan 3E	240	15kbit	72kbit	4 (18x18)

* Compare 2nd most performance

Design Flow - Mapping

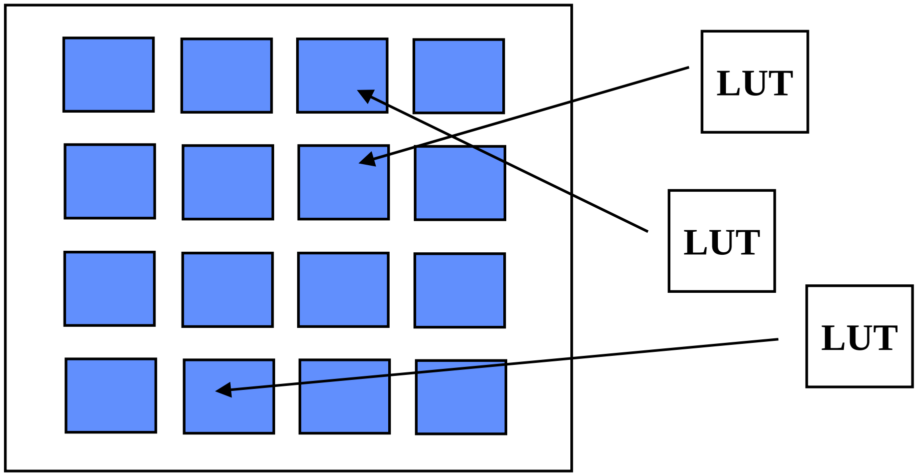
- **Technology Mapping: Schematic/HDL to Physical Logic units**
- **Compile functions into basic LUT-based groups (function of target architecture)**



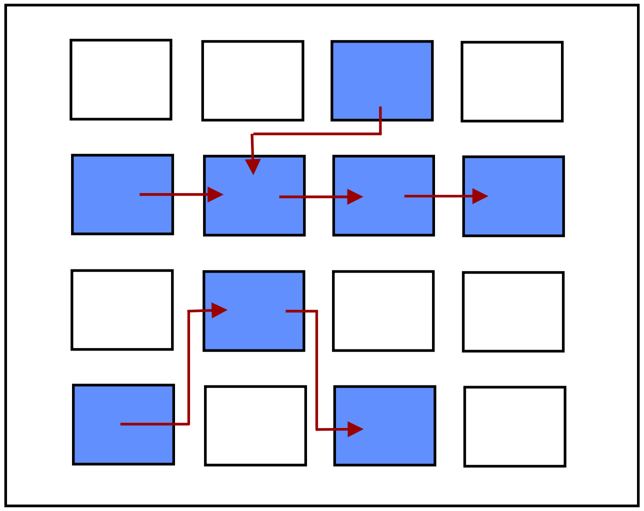
```
always @ (posedge clock)
  begin
    if (reset)  q <= 0;
    else q <= (a & b & c) | (b & d);
  end
```

Design Flow – Placement & Route

- Placement – assign logic location on a particular device



- Routing – iterative process to connect CLB inputs/outputs and IOBs. Optimizes critical path delay – can take hours or days for large, dense designs



Iterate placement if timing not met

Satisfy timing? → Generate Bitstream to config device

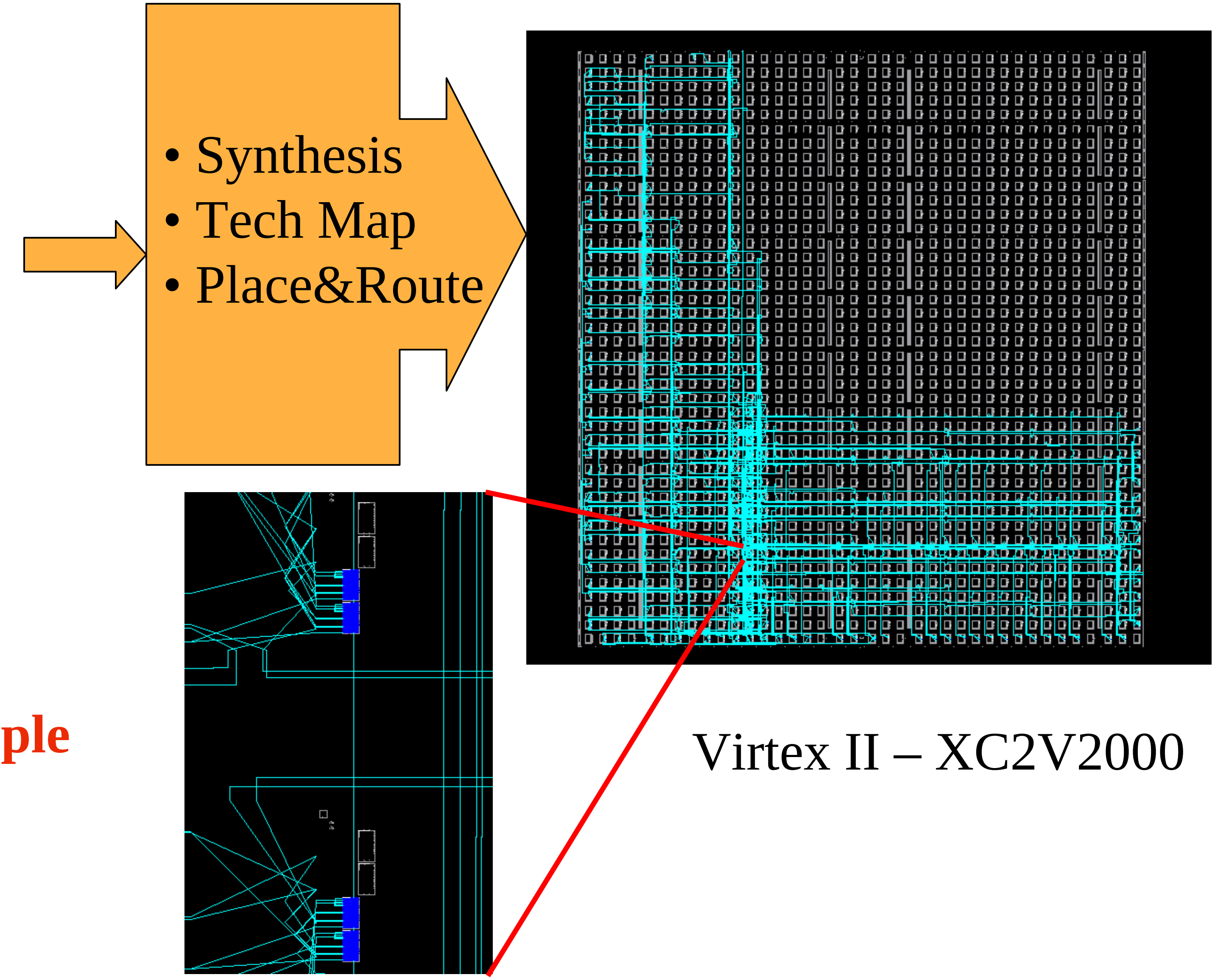
Challenge! Cannot use full chip for reasonable speeds (wires are not ideal).

Typically no more than 50% utilization.

Example: Verilog to FPGA

```
module adder64 (a, b, sum);
  input  [63:0] a, b;
  output [63:0] sum;

  assign sum = a + b;
endmodule
```



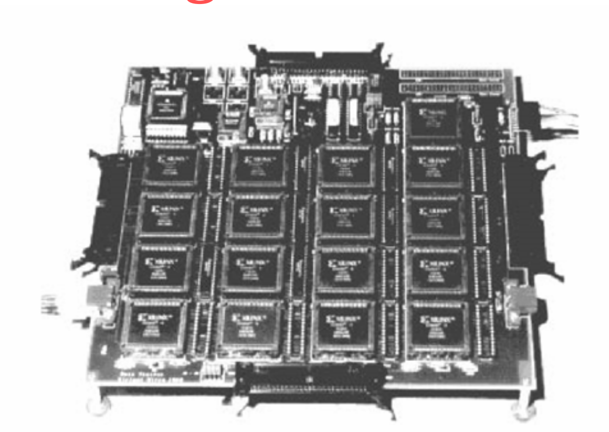
- Synthesis
- Tech Map
- Place&Route

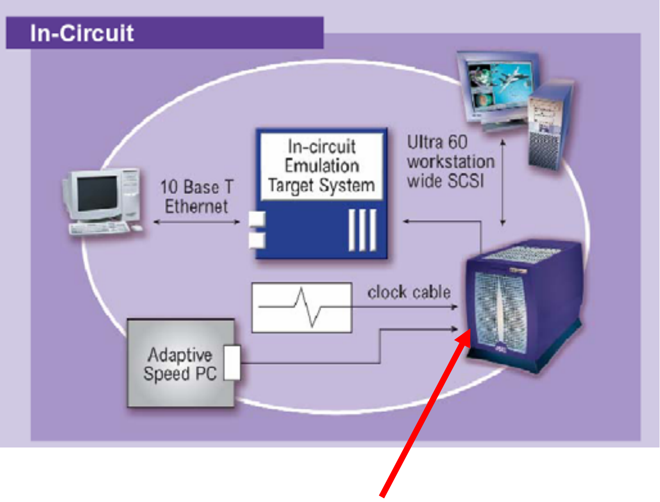
Virtex II – XC2V2000

64-bit Adder Example

How are FPGAs Used?

Logic Emulation



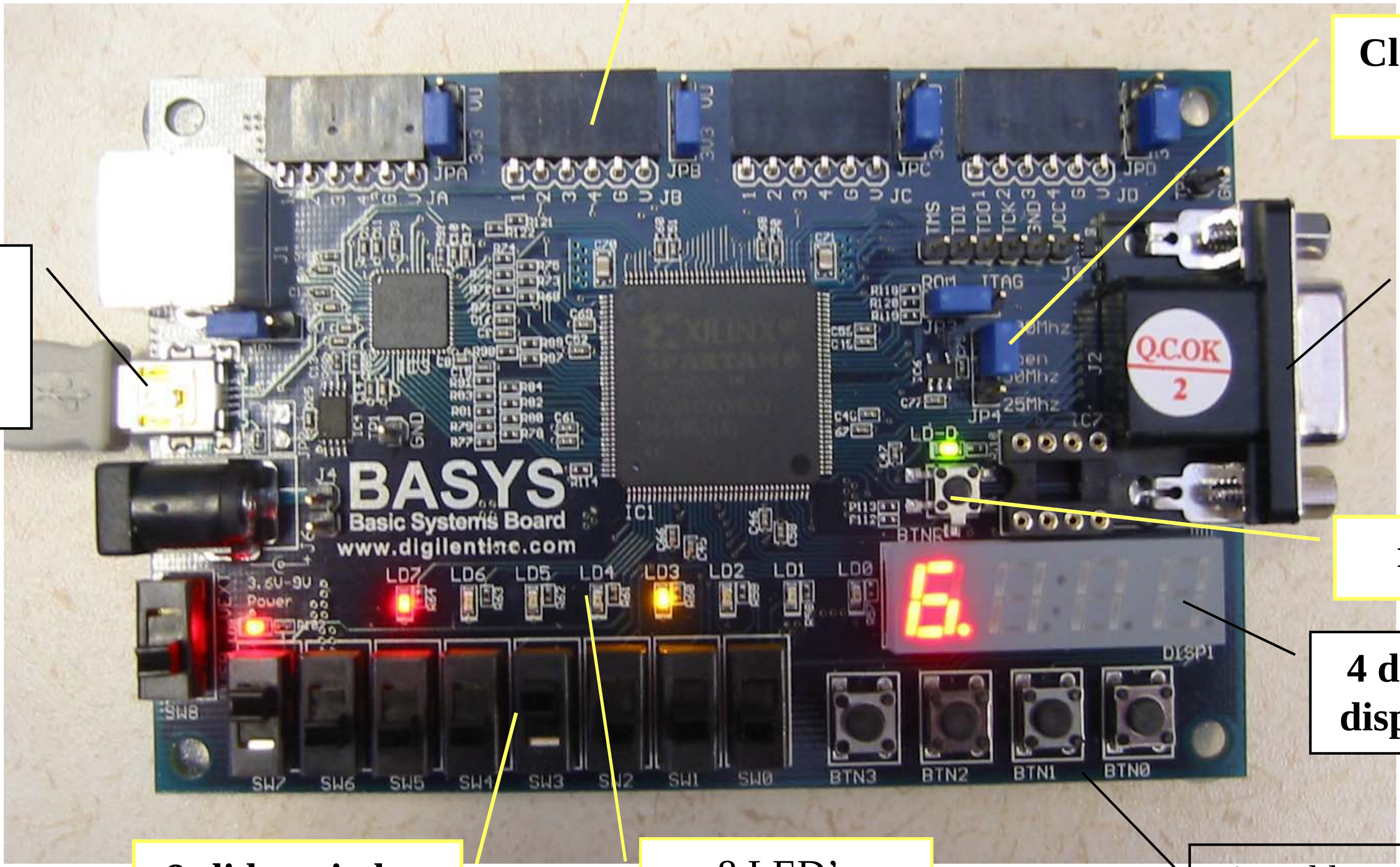

In-Circuit
10 Base T Ethernet
In-circuit Emulation Target System
Ultra 60 workstation wide SCSI
clock cable
Adaptive Speed PC

FPGA-based Emulator

(courtesy of IKOS)

- **Prototyping**
  - □ Ensemble of gate arrays used to emulate a circuit to be manufactured
  - □ Get more/better/faster debugging done than with simulation
- **Reconfigurable hardware**
  - □ One hardware block used to implement more than one function
- **Special-purpose computation engines**
  - □ Hardware dedicated to solving one problem (or class of problems)
  - □ Accelerators attached to general-purpose computers (e.g., in a cell phone!)

Personal FPGA - BASYS



4 user i/o
16 total

Clock freq
select

USB prog.
port &
power

VGA
port

reset

4 digit
display

8 slide switches

8 LED's

4 pushbottons

FPGA Software

- **Xilinx ISE Web-pack**
  - A free, downloadable design environment for both Microsoft Windows and Linux - but it's a 2.25GB download.
  - All the tools and features of ISE Foundation, including the Xilinx CORE Generator™ system and FPGA Editor
  - Support for Xilinx FPGA families, including the Virtex-5 Family of platform FPGAs (does not support big FPGA's like XCV6000!)
- **Modelsim available for student download (Modelsim PE) with 6 month license.**
- **Most prototyping boards can be programmed via USB.**

Summary

- FPGA provide a flexible platform for implementing digital computing
- A rich set of macros and I/Os supported (multipliers, block RAMS, ROMS, high-speed I/O)
- A wide range of applications from prototyping (to validate a design before ASIC mapping) to high-performance spatial computing
- Interconnects are a major bottleneck (physical design and locality are important considerations)

"College students will study concurrent programming instead of "C" as their first computing experience."

-- David B. Parlour, ISSCC 2004 Tutorial