Outline

- Field Programmable Gate Arrays
  - Historical perspective
- Programming Technologies
- Architectures
  - PALs, PLDs, and CPLDs
  - FPGAs
    - ✓Programmable logic
    - ✓Interconnect network
    - ✓I/O buffers
    - ✓Specialized cores
- Programming Interfaces

History

- Programmable Logic Arrays ~ 1970
  - Can implement any set of sum-of-products logic equations
  - Incorporated in VLSI devices
- Programmable Logic Devices ~ 1980
  - MMI Programmable Array Logic (PAL)
    - ✓ 16L8 – combinational logic only
    - ✓ 16R8 – sequential logic only
  - AMD 22V10 and Lattice 16V8
  - Complex PLDs – arrays of PLDs with routing network
- Field Programmable Gate Arrays ~ 1985
  - Xilinx Logic Cell Array (LCA)
- CPLD & FPGA architectures became similar ~ 2000

Programming Technologies

- PLAs were mask programmable
- PALs used fuses for programming
- Early PLDs & CPLDs used floating gate technology
  - Erasable Programmable Read Only Memory (EPROM)
    - ✓ Ultra-violet erasable (UVEPROM)
    - ✓ Electrically erasable (EEPROM)
    - ✓ Flash memory came later and was used for CPLDs
- FPGAs used RAM for programming
- Later trends
  - Fuses were replaced with anti-fuses
    - ✓ Better reliability
  - Large CPLDs went to RAM-based programming

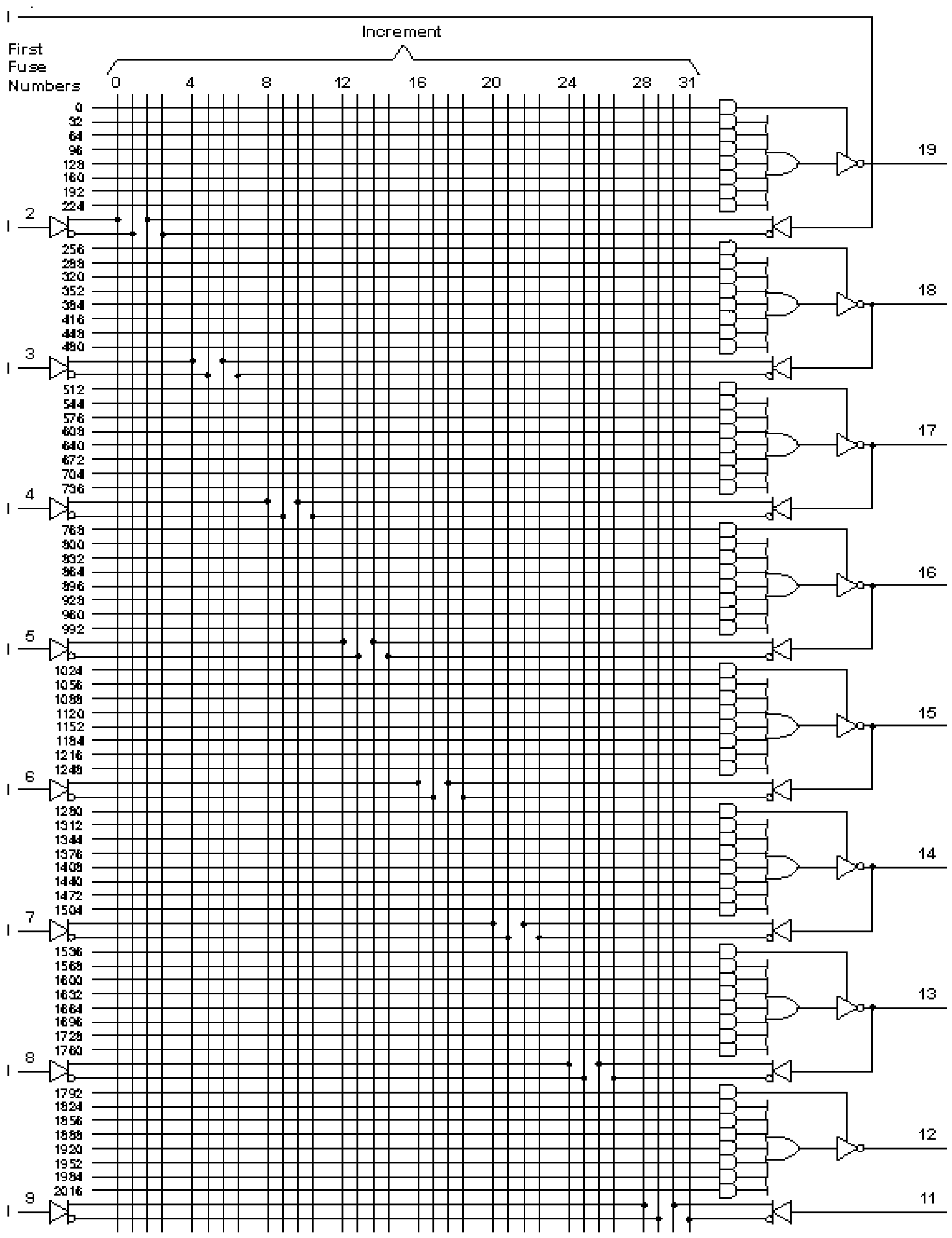
Programming Technologies

- RAM
  - Volatile – must configure after power-up
  - In-System Re-programmable (ISR)
  - Run-Time Reconfiguration (RTR)
    - ✓ dynamic reconfiguration while system is operating
- Floating gate technologies
  - Non-volatile but re-usable
    - ✓ UV EPROM, EEPROM, and flash memory
  - In-System Programmable (ISP)
    - ✓ EEPROM and flash memory
  - In-System Re-programmable (ISR)
    - ✓ Flash memory
- Fuse/anti-fuse
  - Non-volatile but not re-usable
  - One Time Programmable (OTP)

PALs

16L8 – combinational logic

- ❑ 10 to 16 inputs, each with true and complement signal
- ❑ 2 to 8 outputs, each with
  - ❖ 7 product terms can AND any of up to 16 inputs or their complements
  - ❖ Tri-state control product term for inverting output buffer
    - ✓ When output in tri-state, I/O pin can be used as input
      - ➢ High impedance output with no signal driven

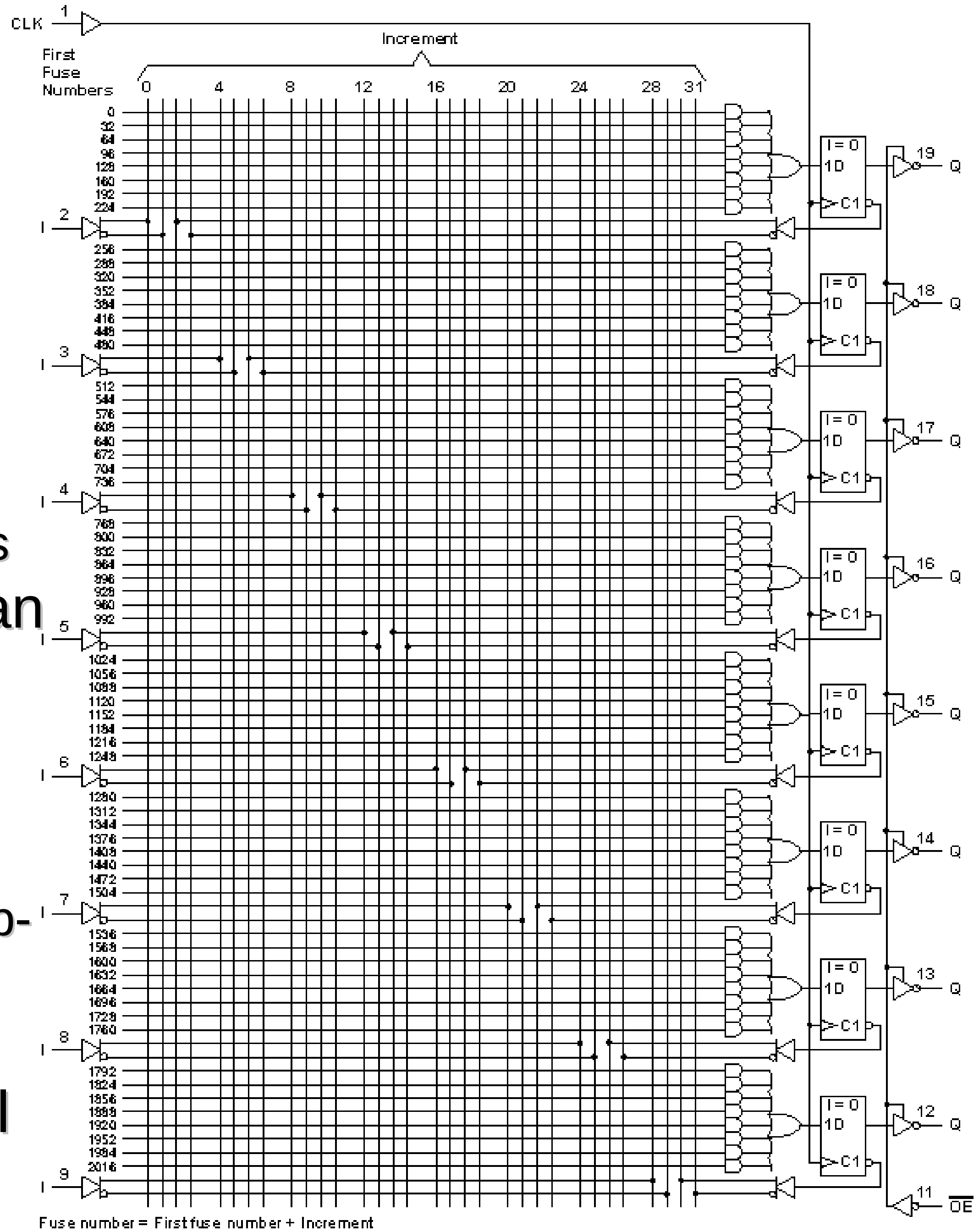


Fuse number = First fuse number + Increment

PALs

16R8 – sequential logic

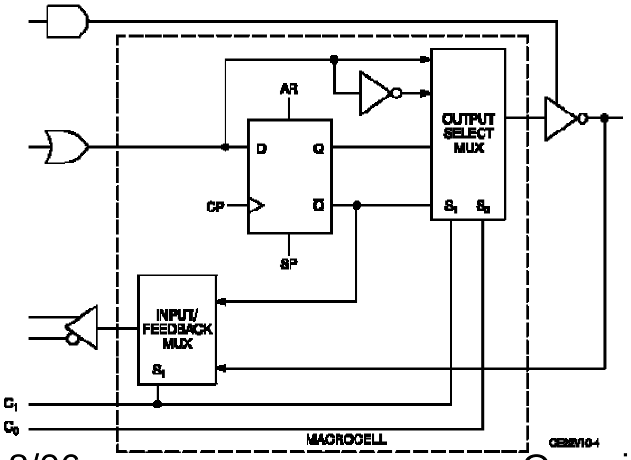
- 8 inputs, each with true & complement
- 8 outputs, each with
  - D flip-flop
    - ✓ With feedback for FSMs
  - 8 product terms that can AND any of:
    - ✓ 8 inputs or their complements
    - ✓ 8 feedbacks or their complements from D flip-flops
- One clock for all FFs
- One tri-state control for all outputs

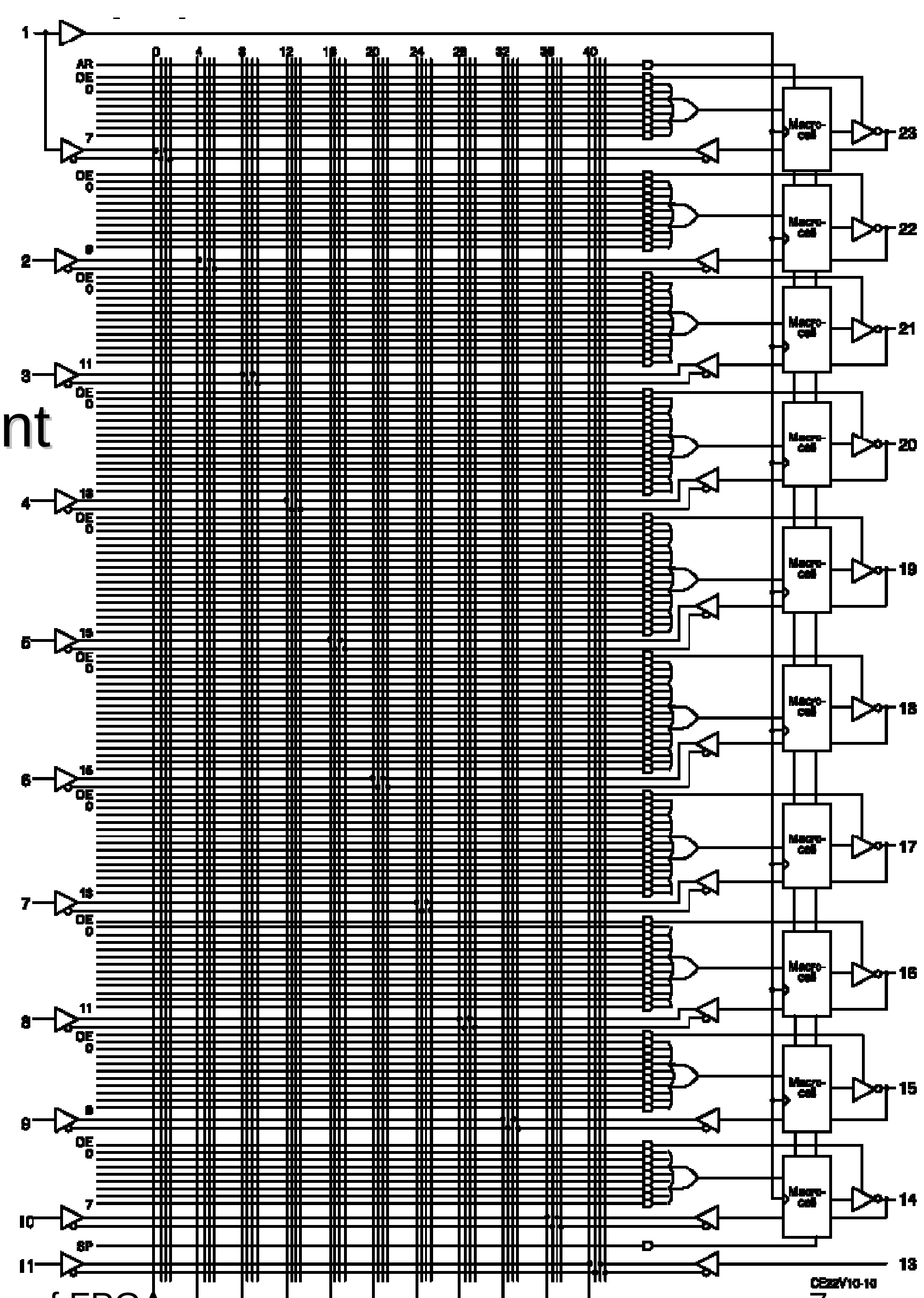


PLDs

22V10 replaced all PALs

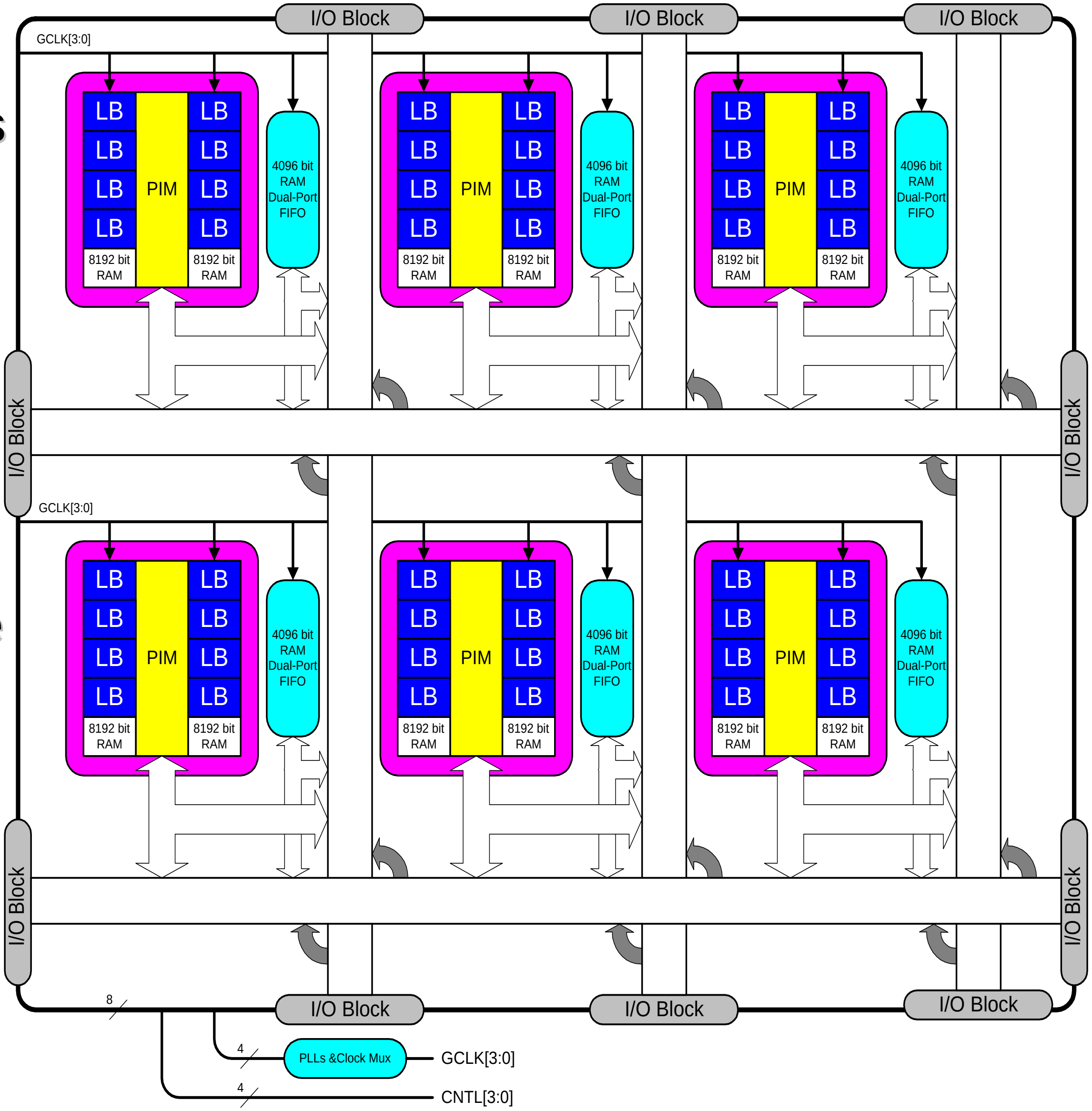
- Combinational and/or sequential logic
  - Macrocell program bits C0, C1
- Up to 22 inputs w/complement
- Up to 10 outputs, each with
  - Macrocell
  - 8-16 product terms
  - Tri-state control product term
- Global
  - preset & clear PTs
  - clock





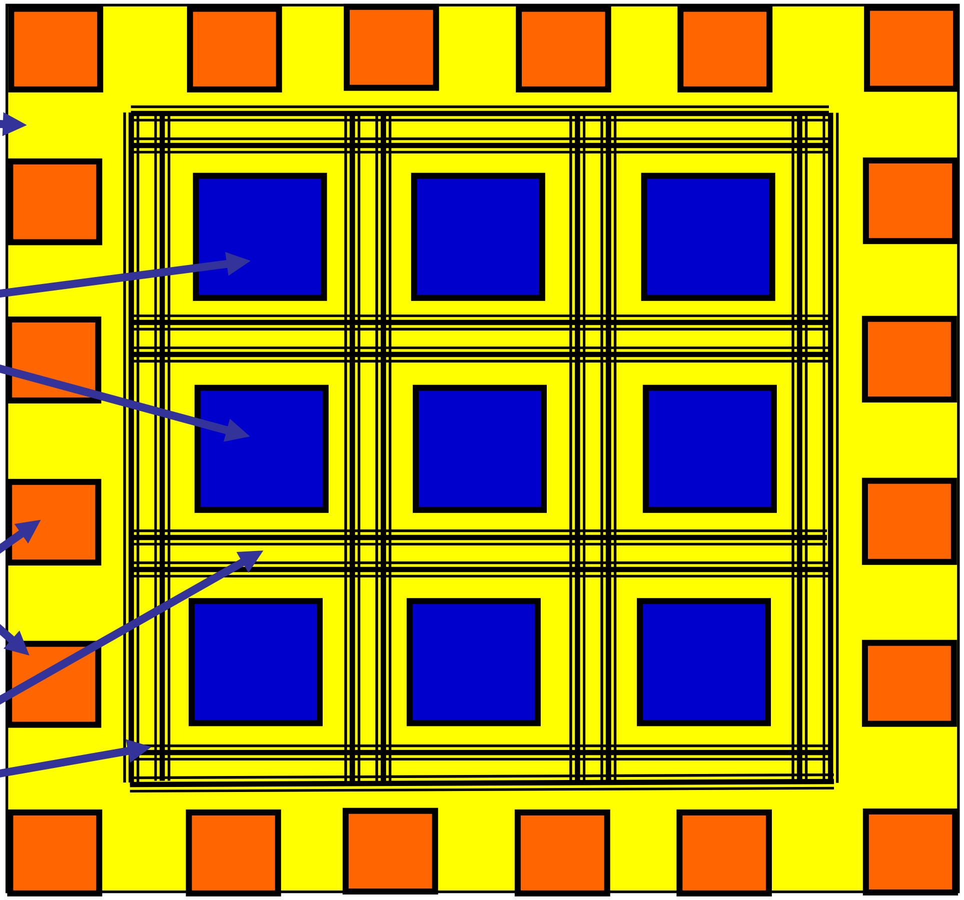
CPLDs

- An array of PLDs
  - Global routing resources for connections
    - ✓ PLDs to other PLDs
    - ✓ PLDs to/from I/O pins
- Example: Cypress 39K
  - Each Logic Block (LB) similar to a 22V10
  - Each cluster of 8 LBs has two 8K RAMs & one 4K dual-port RAM/FIFO
    - ✓ Programmable Interconnect Modules (PIMs) provide interconnections
  - Array of up to 24 clusters with global routing



Field Programmable Gate Arrays

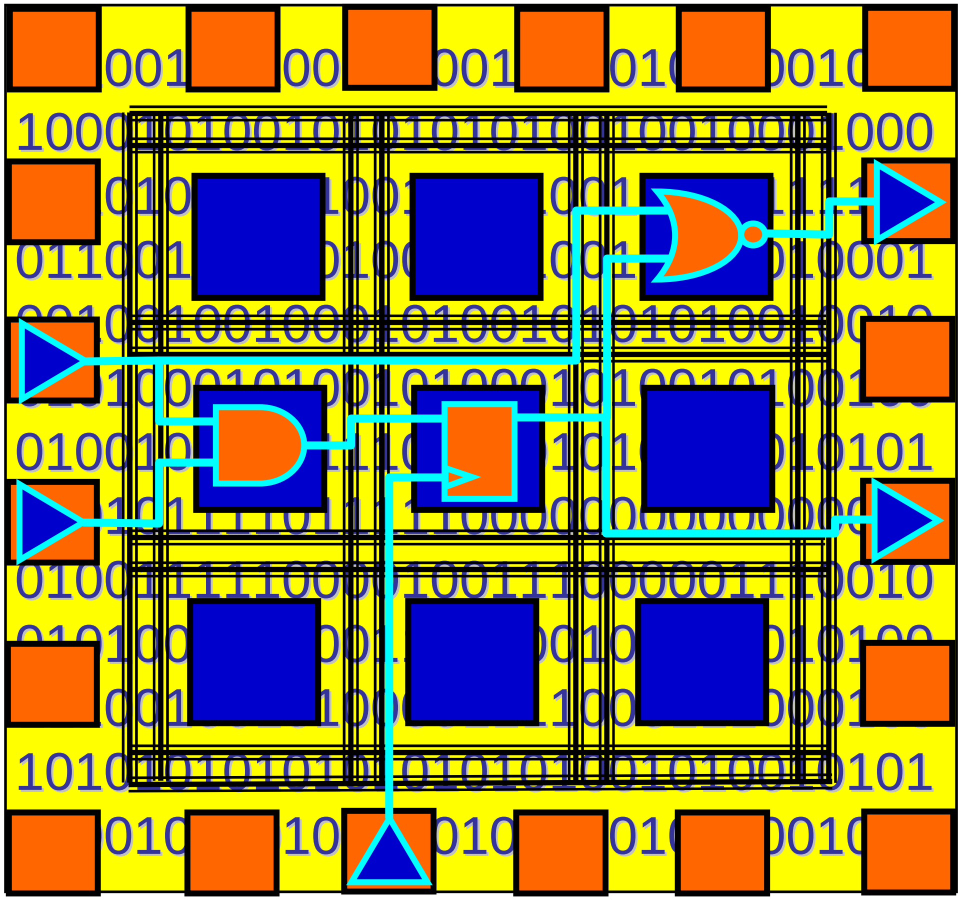
- Configuration Memory
- Programmable Logic Blocks (PLBs)
- Programmable Input/Output Cells
- Programmable Interconnect



Typical Complexity = 5 million – 1 billion transistors

Basic FPGA Operation

- Writing configuration memory ⇒ defines system function
  - Input/Output Cells
  - Logic in PLBs
  - Connections between PLBs & I/O cells
- Changing configuration memory data ⇒ changes system function
  - Can change at anytime
  - Even while system function is in operation



FPGAs

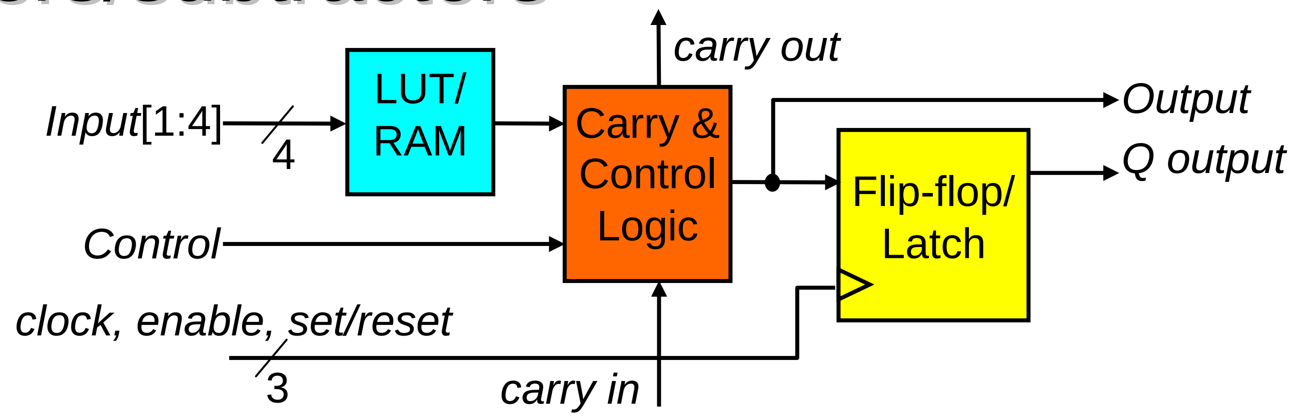
- Configuration memory (42K bits to 80M bits)
- Array of Programmable Logic Blocks (PLBs)
  - 250 to over 25,000 PLBs per FPGA
- Programmable routing network
  - Wire segments (45 to over 400 per PLB)
  - Programmable switches (130 to over 3,600 per PLB)
- Programmable I/O cells around periphery
  - Bi-direction buffer w/ flip-flops/latches (60 to over 1,200 per FPGA)
- Recent trend is to incorporate specialized cores
  - RAMs – single-port, dual-port, FIFOs
    - ✓ 128 bits to over 36K bits per RAM
    - ✓ 4 to over 575 per FPGA
  - DSPs – 18x18-bit multiplier, 48-bit accumulator, etc.
    - ✓ up to 512 per FPGA
  - Microprocessors and/or microcontrollers (up to 2 per FPGA)

Ranges of Resources

FPGA Resource		Small FPGA	Large FPGA
Logic	PLBs per FPGA	256	25,920
	LUTs and flip-flops per PLB	1	8
Routing	Wire segments per PLB	45	406
	PIPs per PLB	139	3,462
Specialized Cores	Bits per memory core	128	36,864
	Memory cores per FPGA	16	576
	DSP cores	0	512
Other	Input/output cells	62	1,200
	Configuration memory bits	42,104	79,704,832

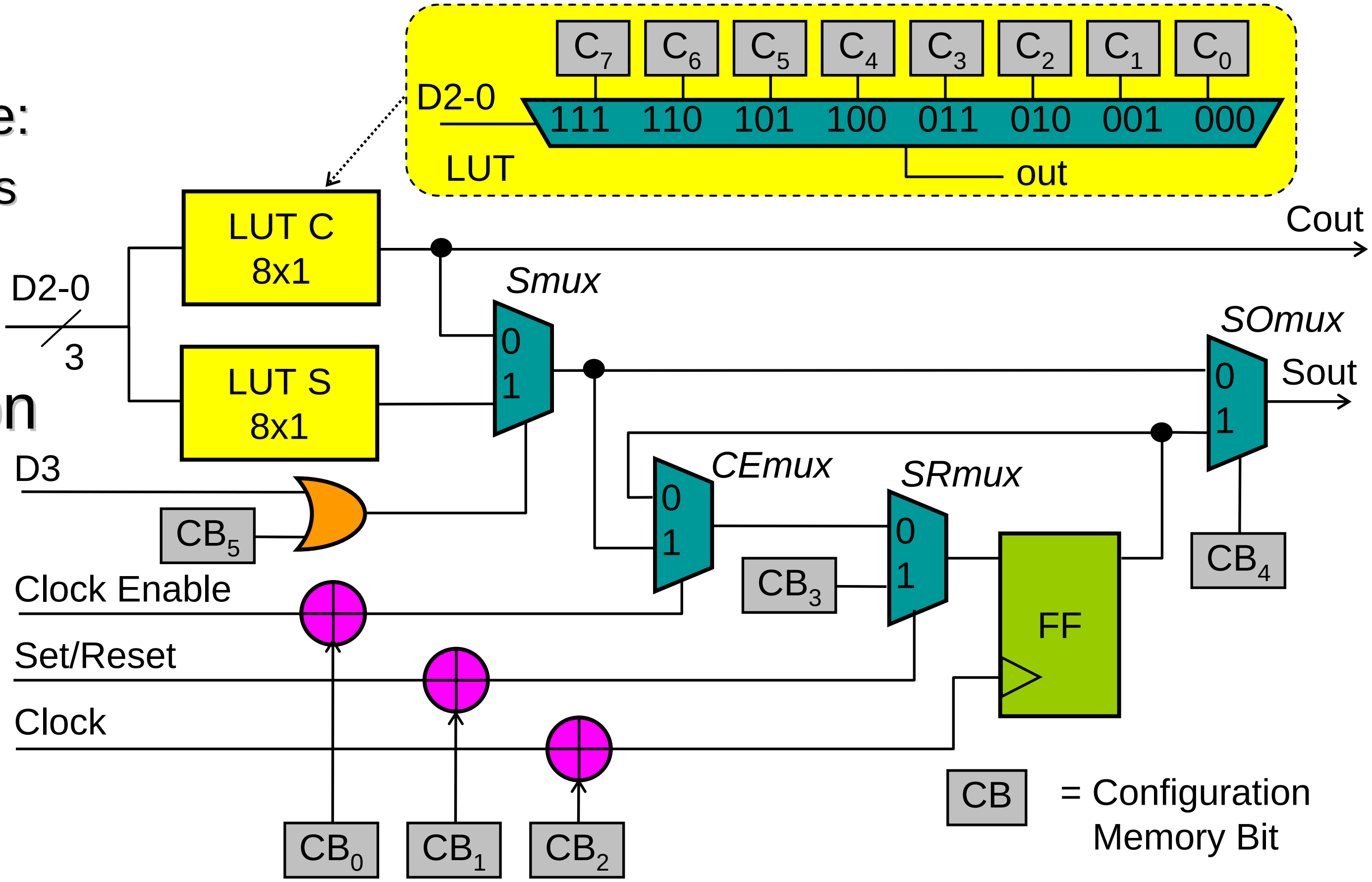
Basic PLB Architecture

- Look-up Table (LUT) implements truth table
- Memory elements:
  - Flip-flop/latch
  - Some FPGAs - LUTs can also implement small RAMs
- Carry & control logic implements fast adders/subtractors



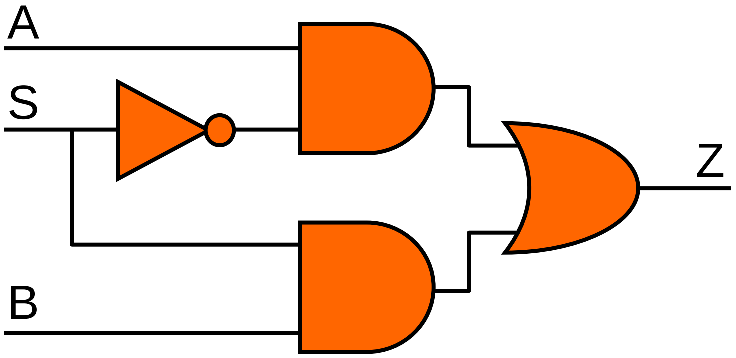
A Simple PLB

- Two 3-input LUTs
  - Can implement any 4-input combinational logic function
- 1 flip-flop
  - Programmable:
    - ✓ Active levels
    - ✓ Clock edge
    - ✓ Set/reset
- 22 configuration memory bits
  - 8 per LUT
    - ✓ C0-7
    - ✓ S0-7
  - 6 controls
    - ✓ CB0-7



Combinational Logic Fucntions

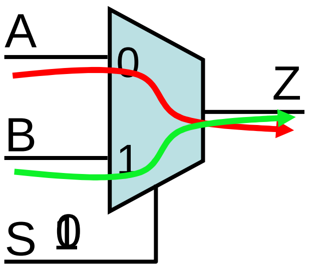
- Gates are combined to create complex circuits
- Multiplexer example
  - If S = 0, Z = A
  - If S = 1, Z = B
  - Very common digital circuit
  - Heavily used in FPGAs
    - ✓S input controlled by configuration memory bit
    - ✓We'll see it again



Truth table

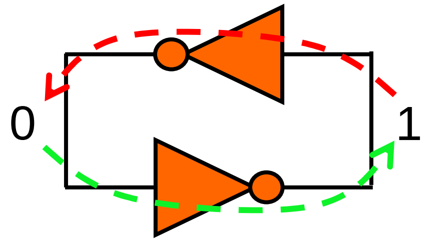
S	A	B	Z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

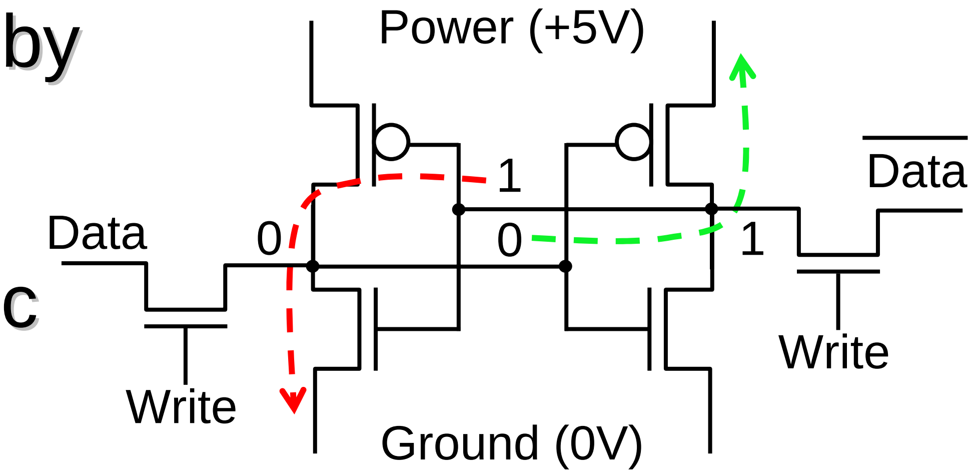
Logic symbol

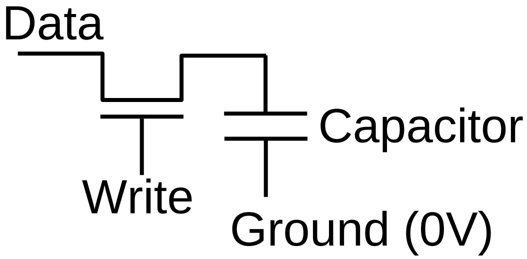


Memory Elements

- Static memory based on cross-coupled logic gates
  - Storage obtained by feedback and amplification
  - Flip-flops are static
- Dynamic memory based on capacitance
  - No feedback or amplification
    - ✓Voltage decays in time







Flip-Flops

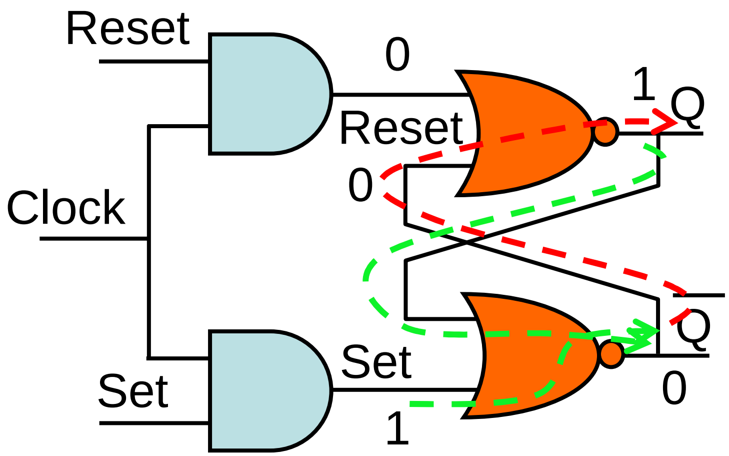
- Use a collection of gates to enable a sample control signal called Clock
- Basic structure is cross-coupled NOR gates
- AND gates provide for sampling control signal - Clock (CK)
- This is a latch – flip-flops are more complicated

AND

A B	Z
0 0	0
0 1	0
1 0	0
1 1	1

NOR

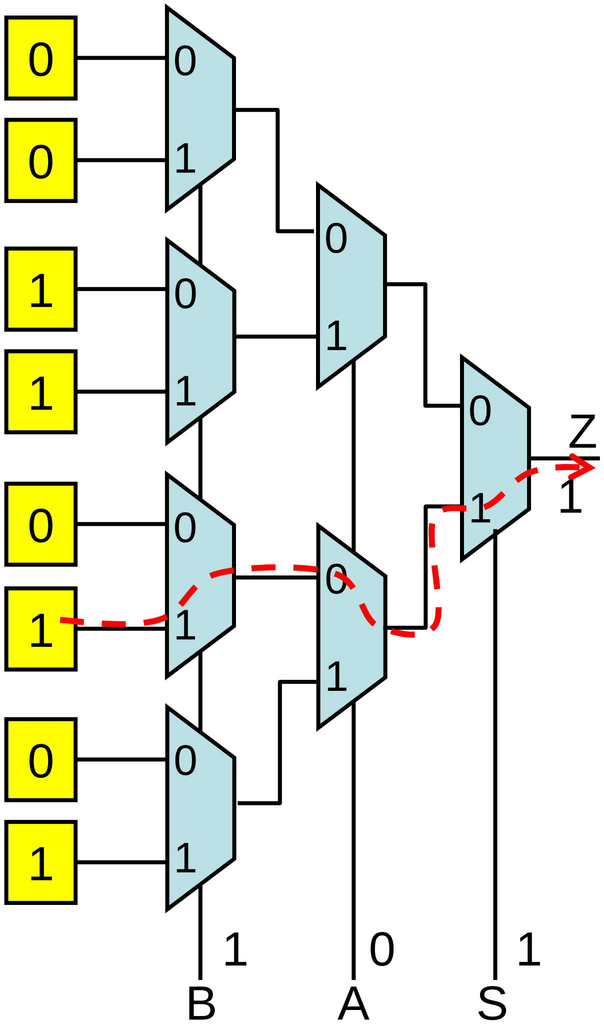
A B	Z
0 0	1
0 1	0
1 0	0
1 1	0



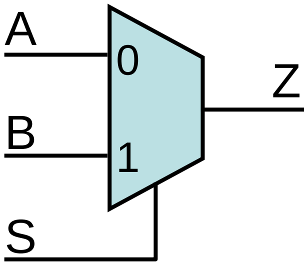
CK	R S	Q
1	0 0	Q
1	0 1	1
1	1 0	0
0	X X	Q

Look-up Tables

- Recall multiplexer example
- Configuration memory holds outputs for truth table
- Internal signals connect to control signals of multiplexers to select value of truth table for any given input value



Multiplexer

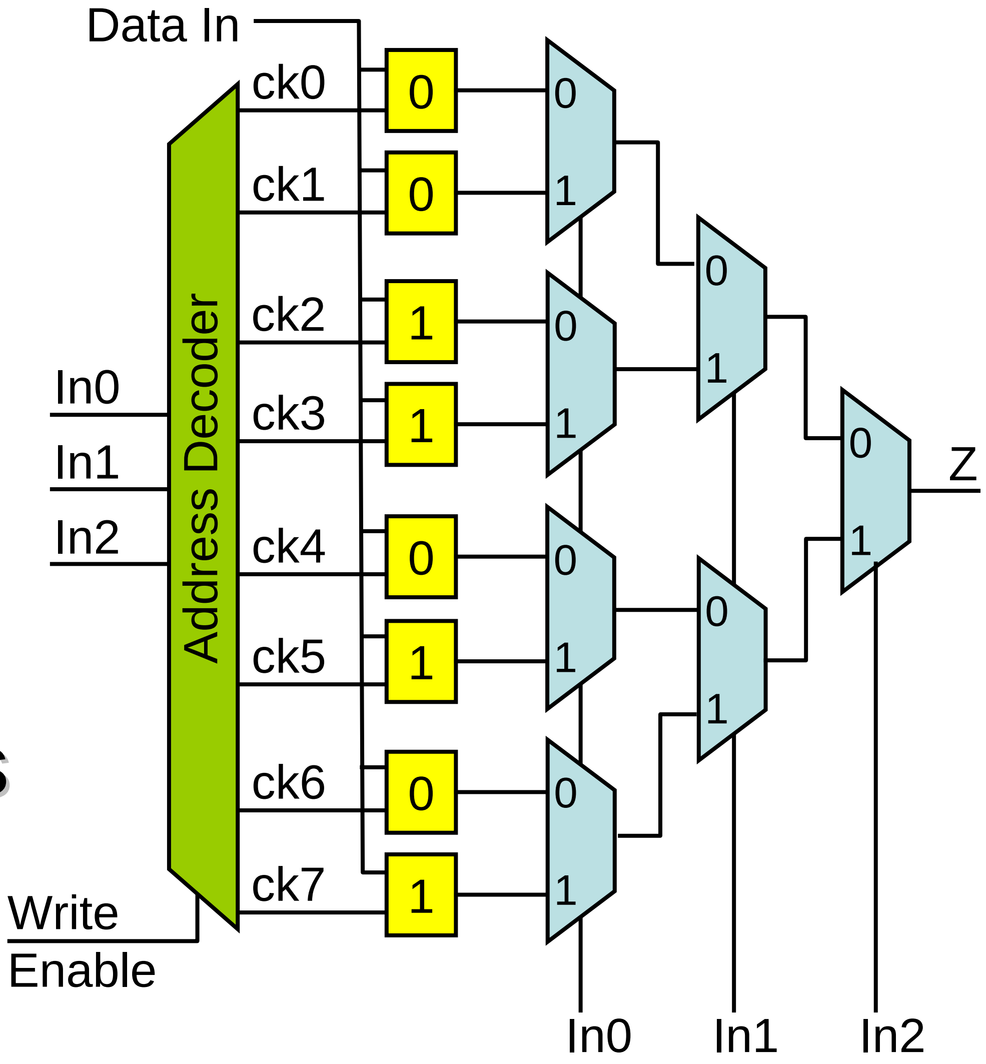


Truth table

S A B	Z
0 0 0	0
0 0 1	0
0 1 0	1
0 1 1	1
1 0 0	0
1 0 1	1
1 1 0	0
1 1 1	1

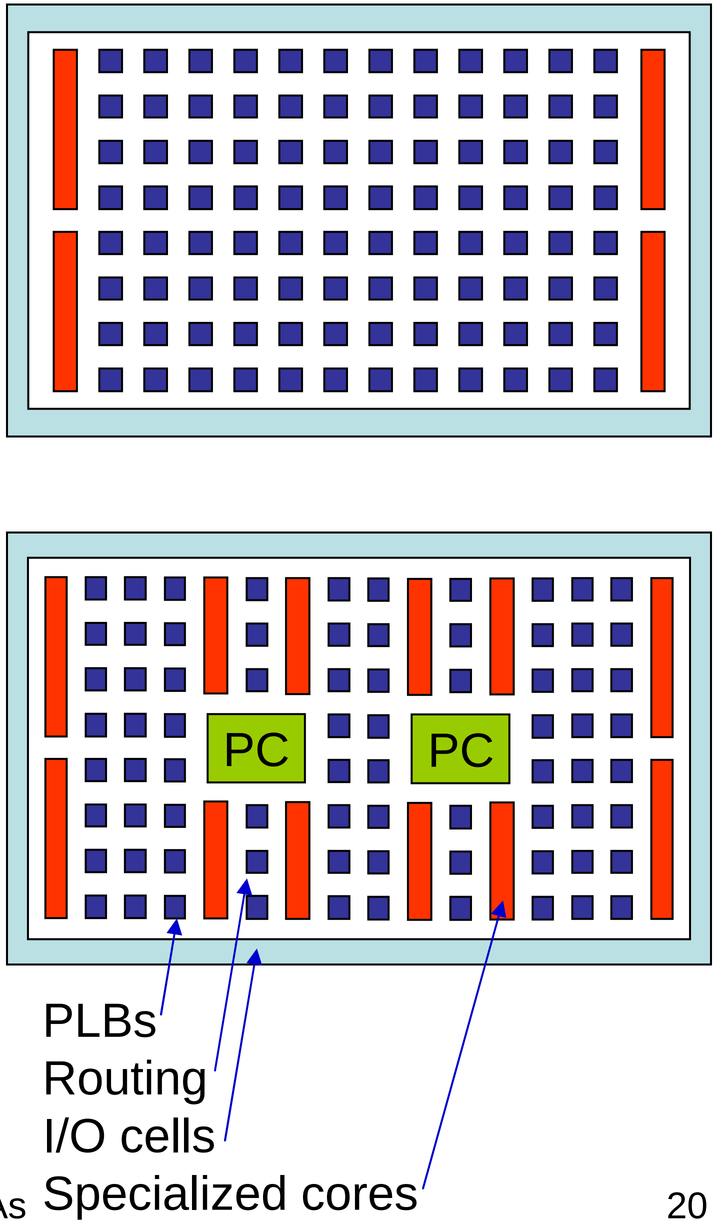
Look-up Table Based RAMs

- ❑ Normal LUT mode performs read operations
- ❑ Address decoder with write enable generates clock signals to latches for write operations
- ❑ Small RAMs but can be combined for larger RAMs



Xilinx FPGAs

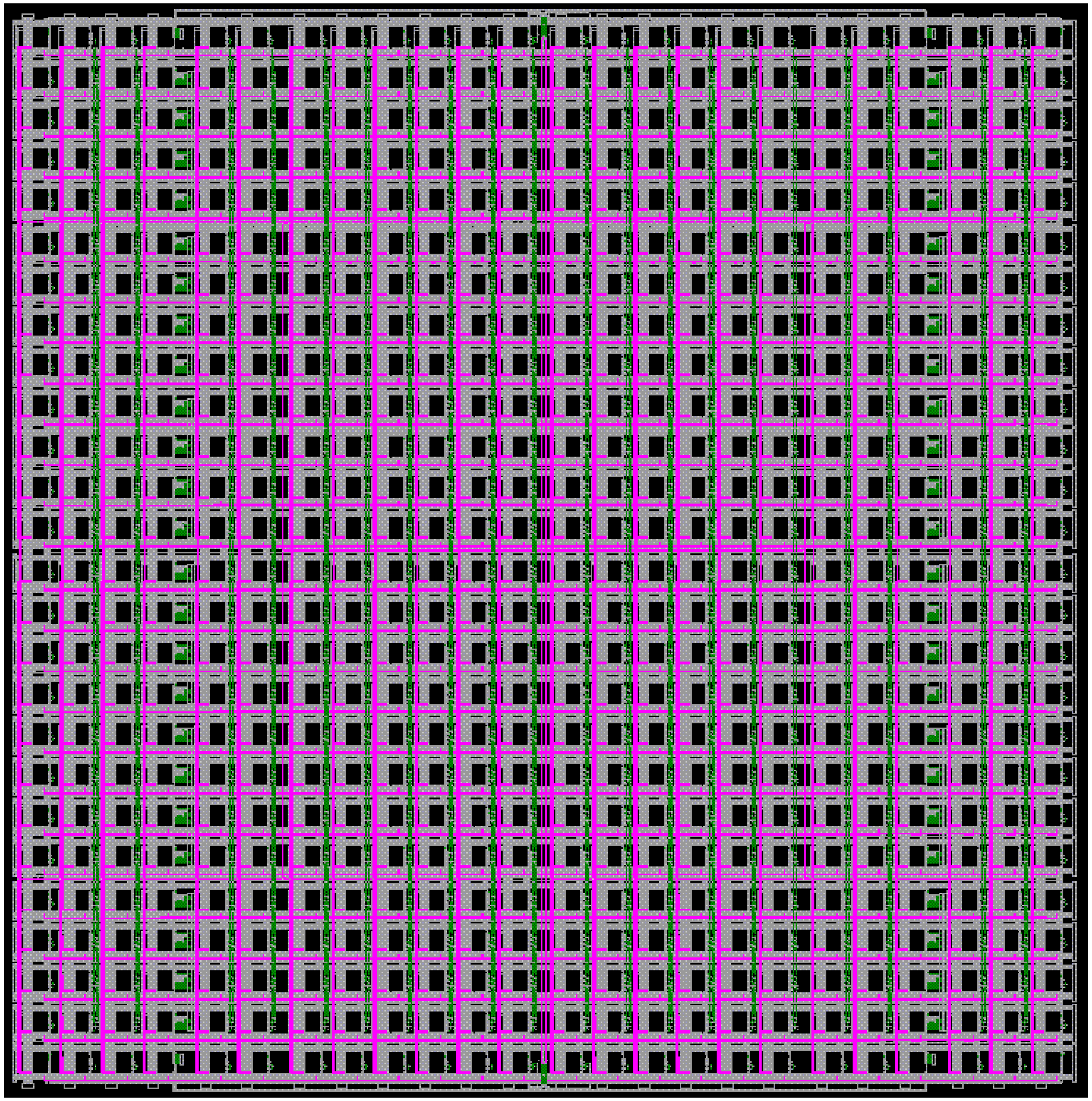
- Virtex and Spartan II
  - Array of 96 to 6,144 PLBs
    - ✓ 4 LUTs/RAMs (4-input)
    - ✓ 4 FF/latches
  - 4 to 32 4K-bit dual-port RAMs
- Virtex II, Virtex II Pro, and Spartan 3
  - Array of 192 to 11,204 PLBs
    - ✓ 8 LUTs/RAMs (4-input)
    - ✓ 8 FF/latches
  - 4 to 444 18K-bit dual-port RAMs
  - 4 to 444 18×18-bit multipliers
  - 0 to 2 PowerPC processor cores
- Virtex 4
  - Array of 1,536 to 22,272 PLBs
    - ✓ 4 LUTs/RAMs (4-input)
    - ✓ 4 LUTs (4-input)
    - ✓ 8 FF/latches
  - 48 to 552 18K-bit dual-port RAMs
    - ✓ Also operate as FIFOs
  - 32 to 512 DSP cores 48-bits
  - 0 to 2 PowerPC processor cores



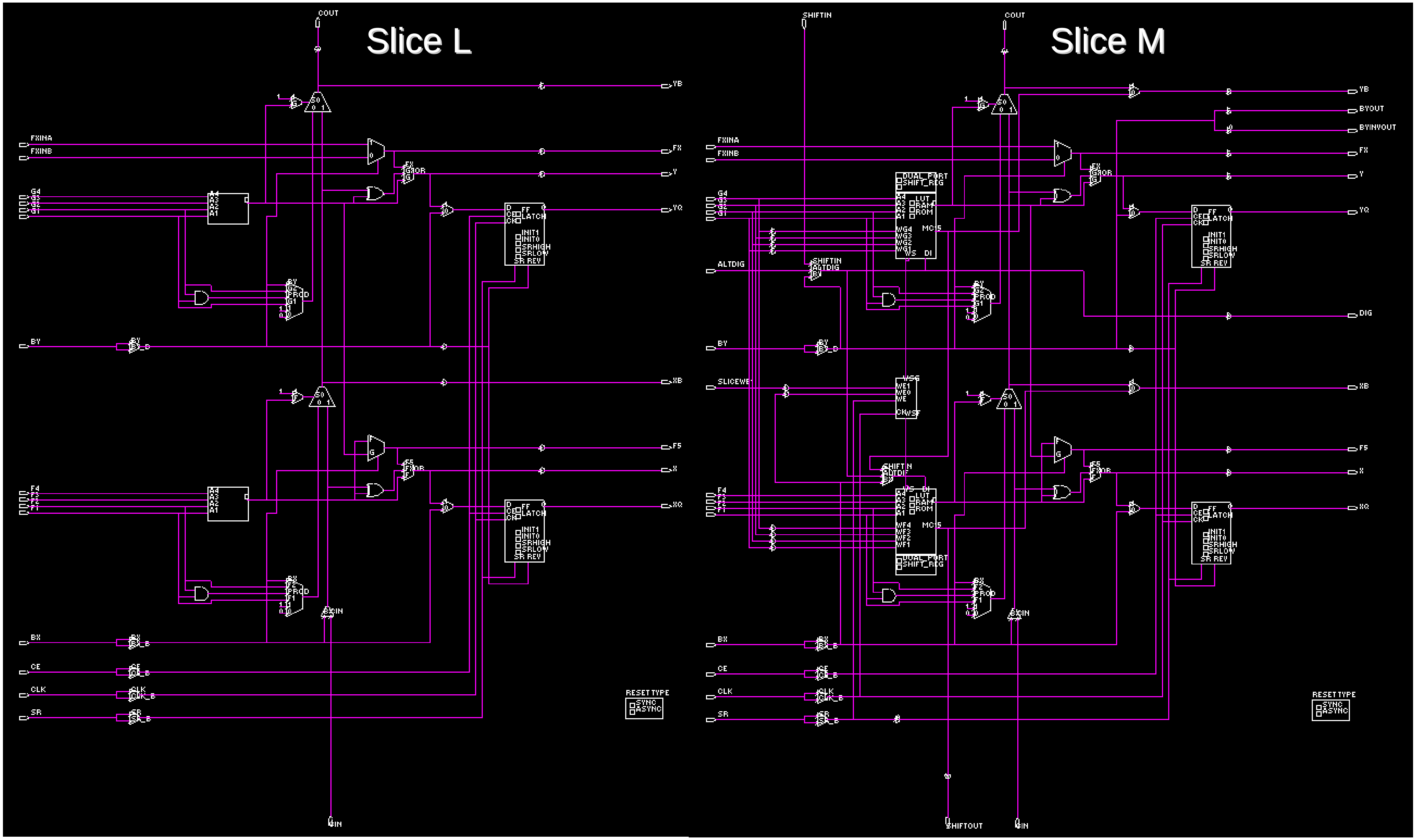
PLBs
Routing
I/O cells
Specialized cores

Spartan 3 (XC3S200)

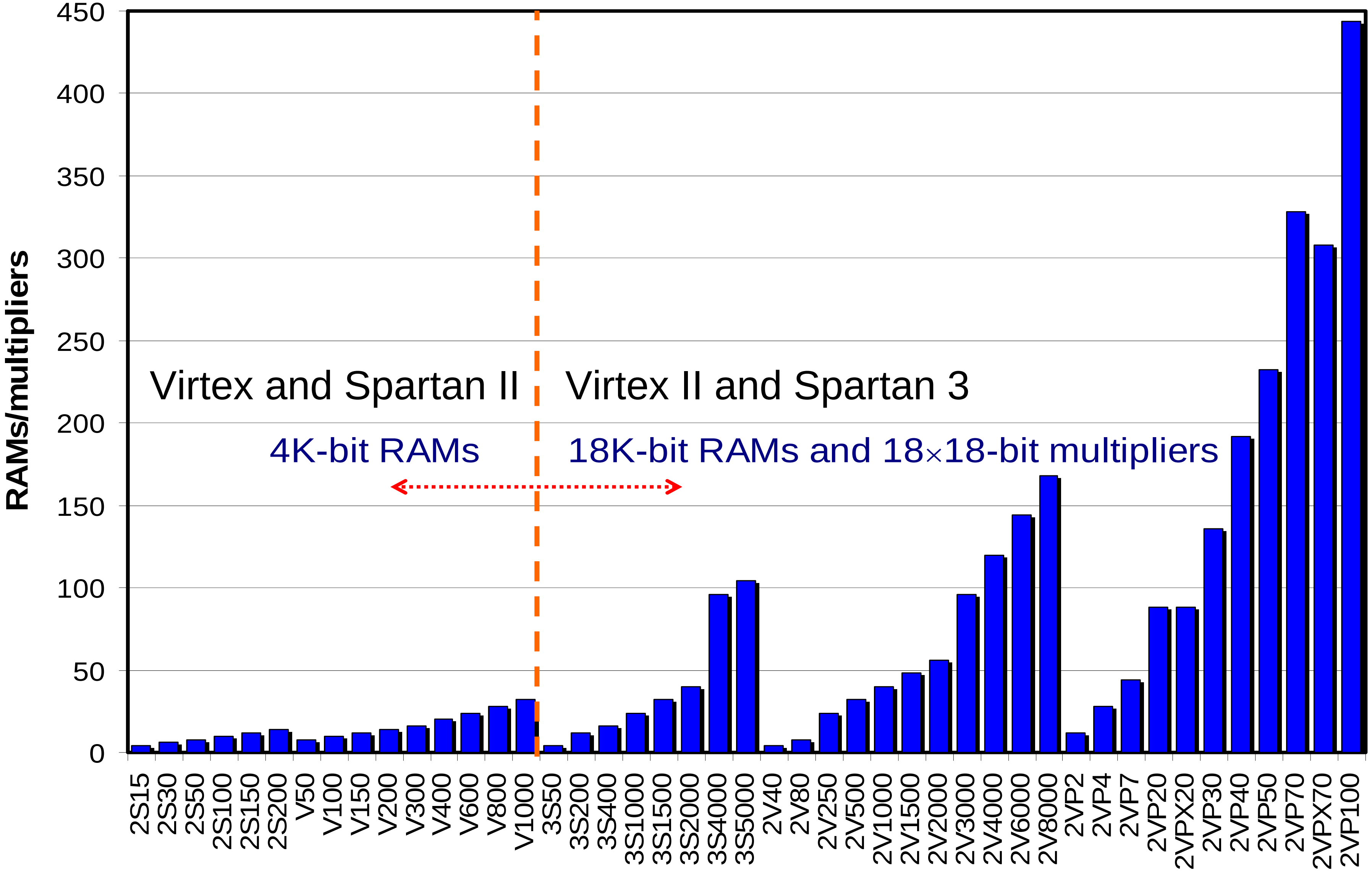
- PLBs = 24 rows x 20 columns = 480
  - 4 slices/PLB
    - ✓2 L slices
      - L= logic
    - ✓2 M slices
      - M= memory
- RAMs = 12 18Kbit dual port RAMs
- Multipliers = 12 18x18-bit signed



Spartan 3 PLB Slices



Specialized Cores



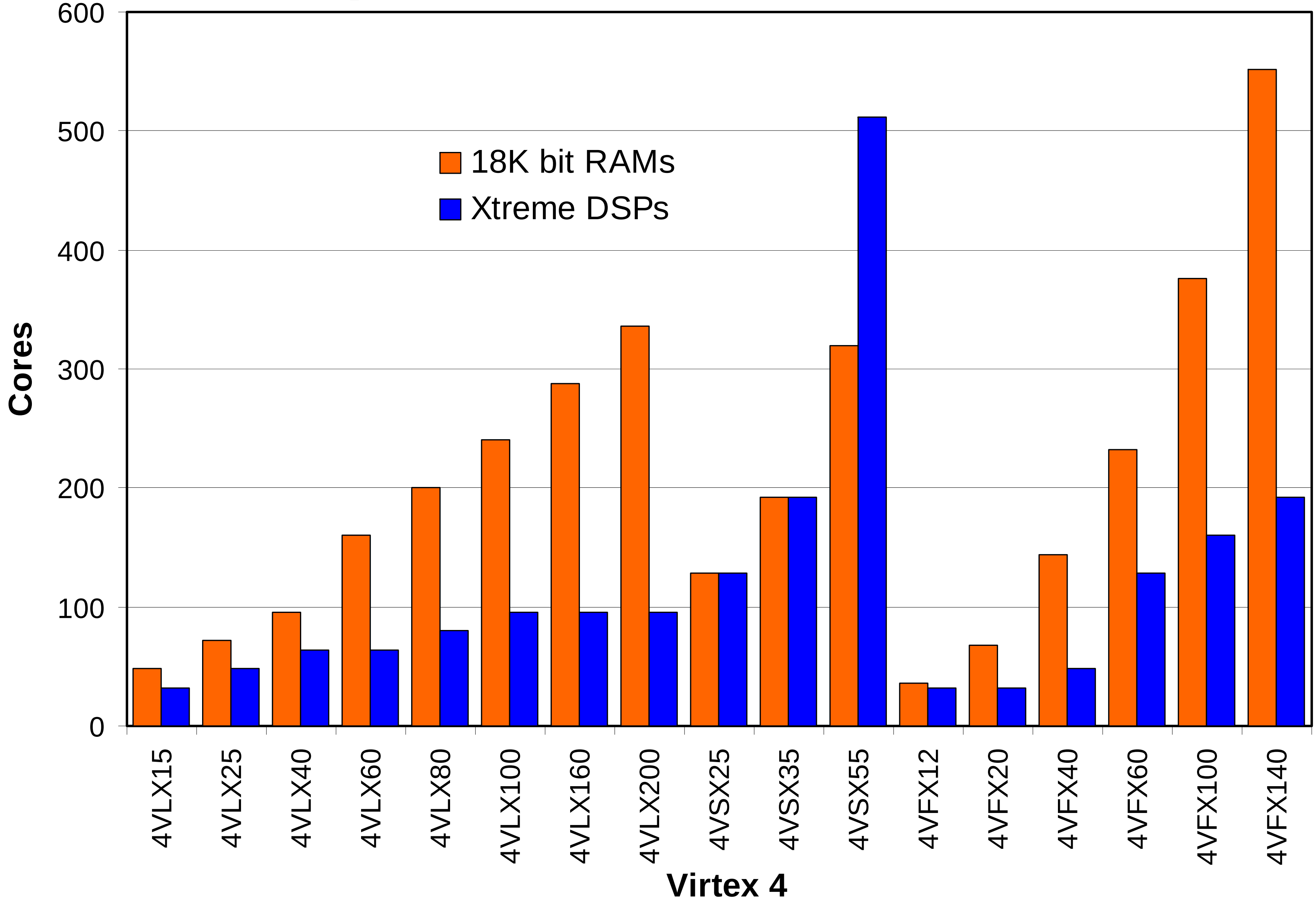
Virtex and Spartan II

4K-bit RAMs

Virtex II and Spartan 3

18K-bit RAMs and 18×18-bit multipliers

Specialized Cores

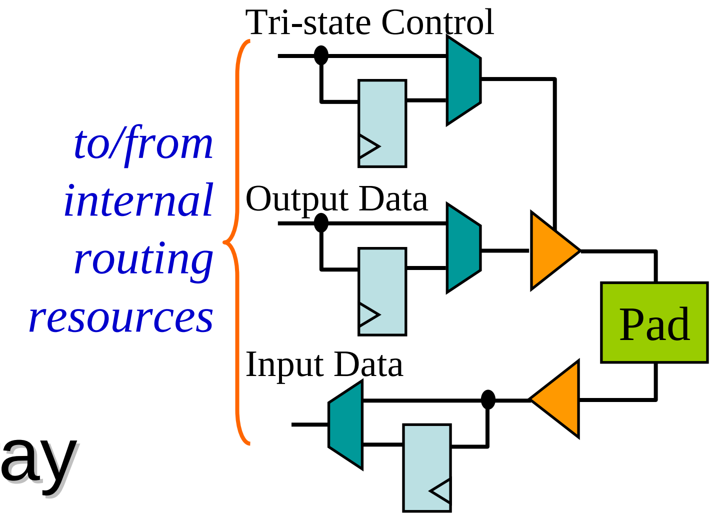


Programmable RAMs

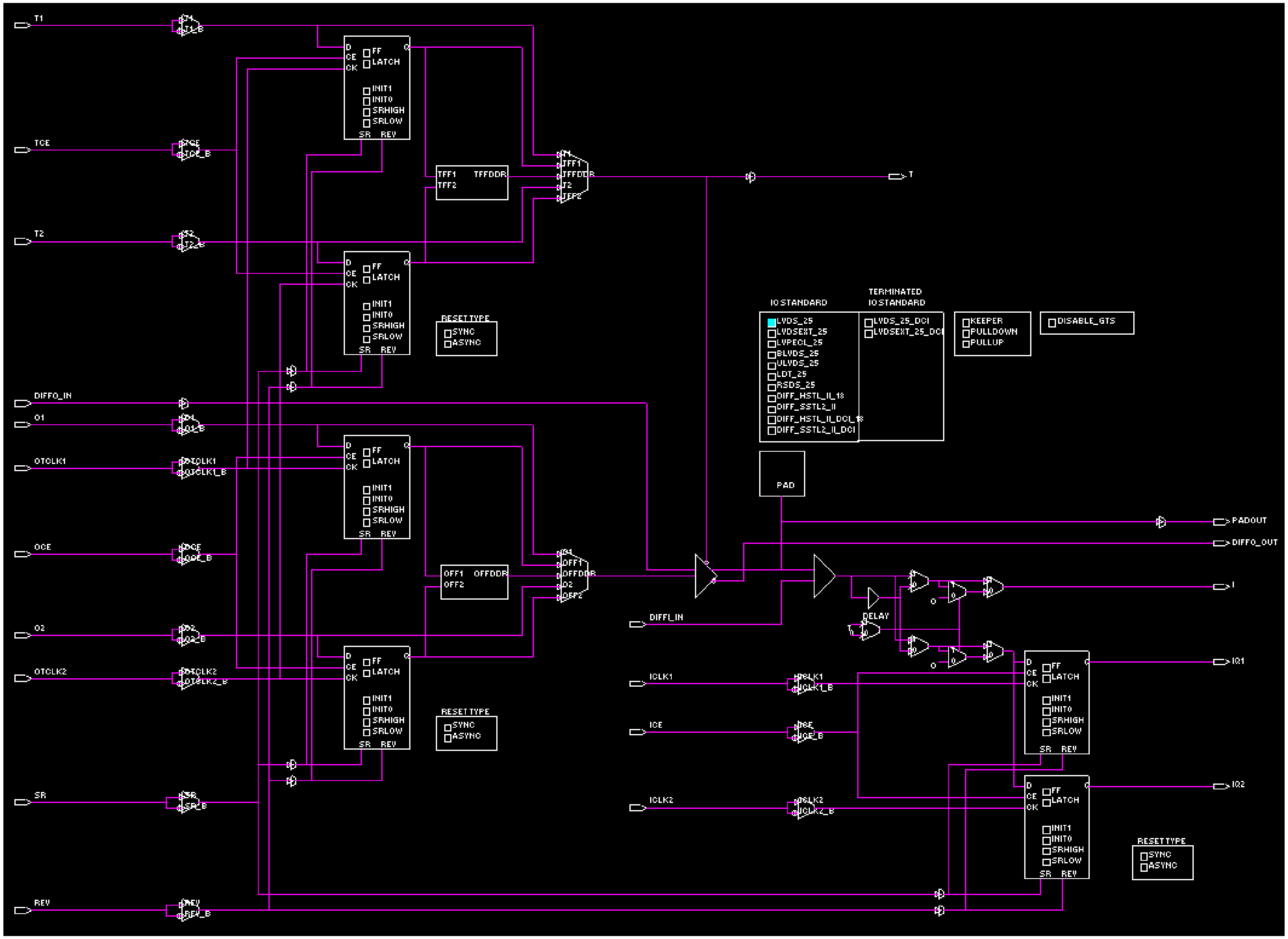
- 18 Kbit dual-port RAM
- Each port independently configurable as
    - 512 words x 36 bits
        - ✓ 32 data bits + 4 parity bits
    - 1K words x 18 bits
        - ✓ 16 data bits + 2 parity bits
    - 2K words x 9 bits
        - ✓ 8 data bits + 1 parity bit
    - 4K words x 4 bits (no parity)
    - 8K words x 2 bits (no parity)
    - 16K words x 1 bit (no parity)
- Each port has independently programmable
    - clock edge
    - active levels for write enable, RAM enable, reset

Input/Output Cells

- ❑ Bi-directional buffers
  - ❖ Programmable for input or output
  - ❖ Tri-state control for bi-directional operation
  - ❖ Flip-flops/latches for improved timing
    - ✓ Set-up and hold times
    - ✓ Clock-to-output delay
  - ❖ Pull-up/down resistors
- ❑ Routing resources
  - ❖ Connections to core of array
- ❑ Programmable I/O voltage & current levels

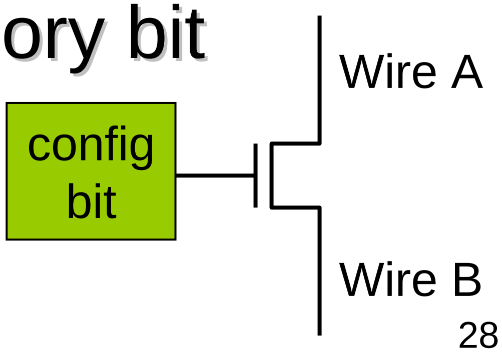


Spartan 3 I/O Cell



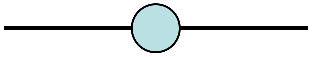
Interconnect Network

- Wire segments of varying length
  - x*N* = *N* PLBs in length
    - ✓1, 2, 4, and 6 are most common
  - xH = half the array in length
  - xL = length of full array
- Programmable Interconnect Points (PIPs)
    - ✓Also known as Configurable Interconnect Points (CIPs)
  - Transmission gate connects to 2 wire segments
  - Controlled by configuration memory bit
    - ✓0 = wires disconnected
    - ✓1 = wires connected

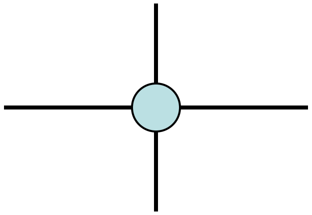


PIPs

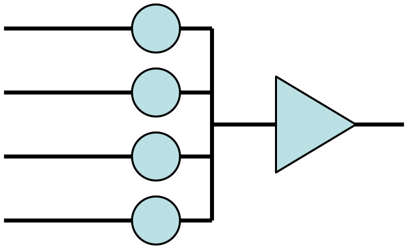
- Break-point PIP
  - Connect or isolate 2 wire segments



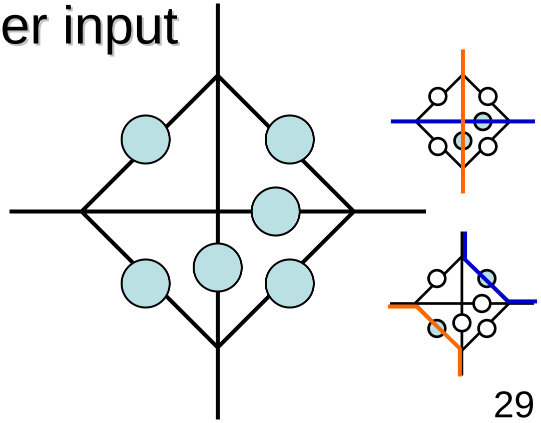
- Cross-point PIP
  - Turn corners



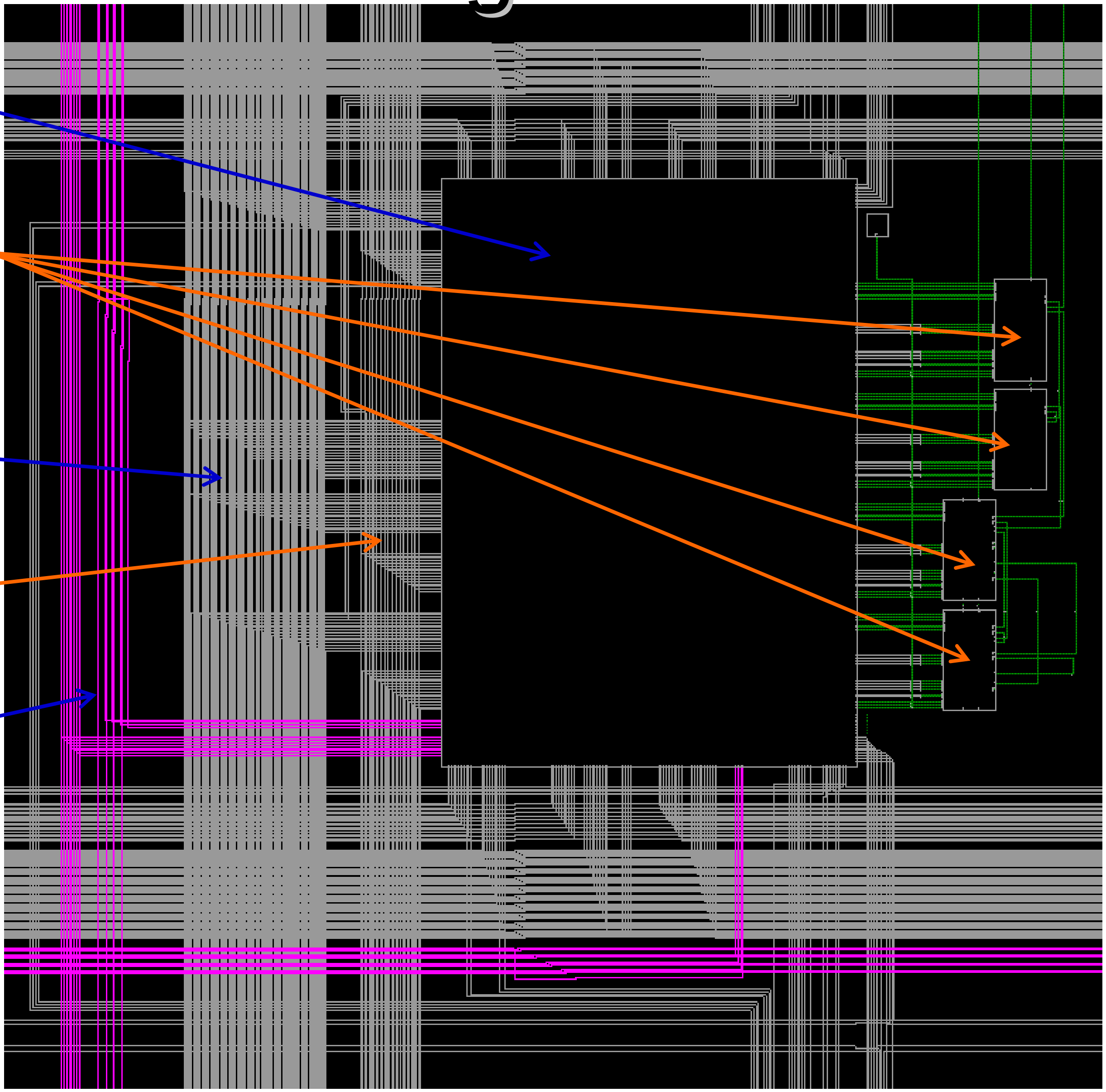
- Multiplexer PIP
  - Directional and buffered
  - Select 1-of-*N* inputs for output
    - ✓ Decoded MUX PIP – *N* config bits select from 2^N inputs
    - ✓ Non-decoded MUX PIP – 1 config bit per input



- Compound cross-point PIP
  - Collection of 6 break-point PIPs
    - ✓ Can route to two isolated signal nets



Spartan 3 Routing Resources



switch matrix

over 2,400 PIPs mostly MUX PIPs

PLB consists of 4 slices

x6 wire segments

x2 wire segments

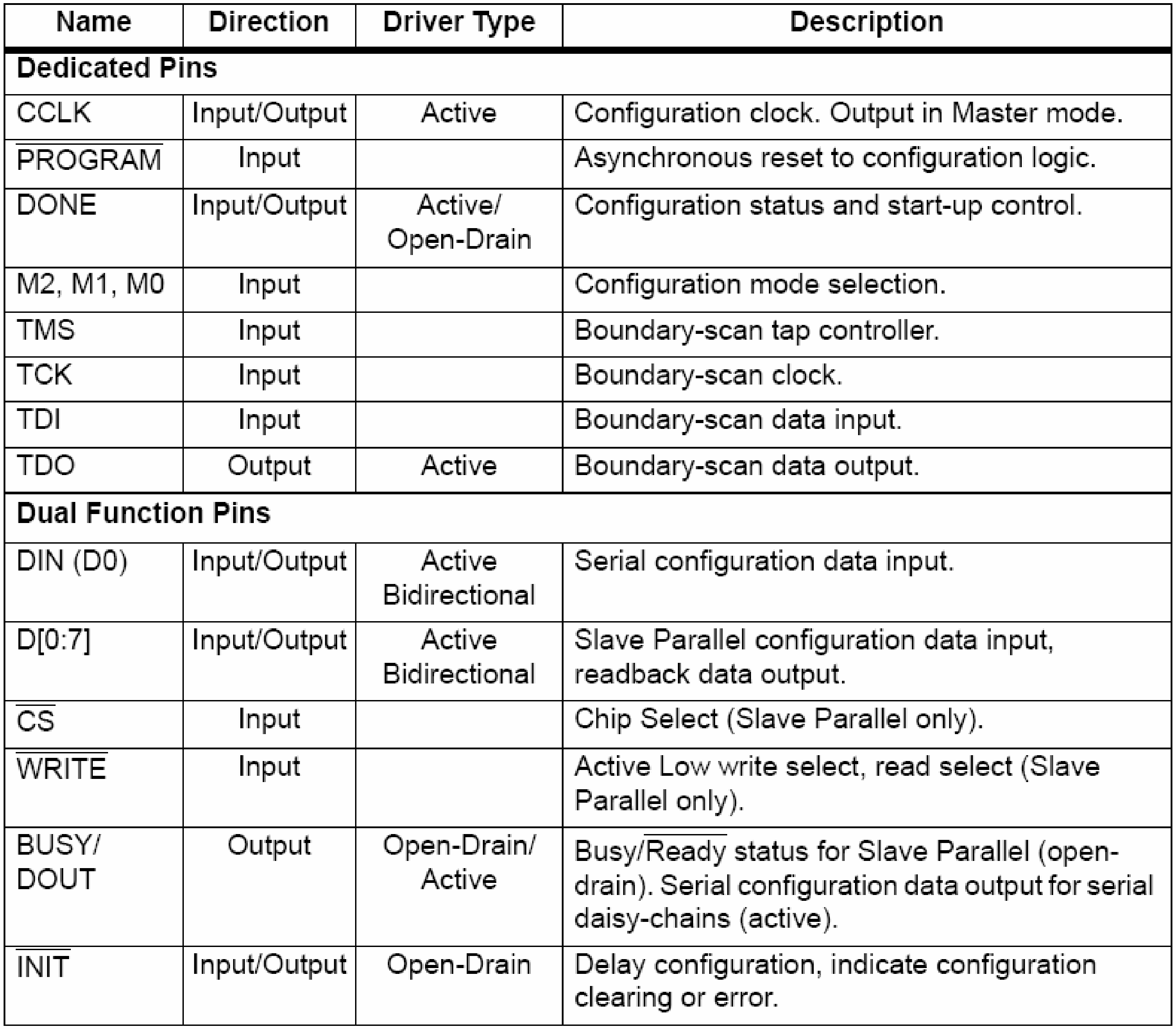
xH & xL wire segments

over 450 total wire segments in PLB

Configuration Interfaces

- Master – FPGA retrieves its own configuration from ROM after power-up
  - Serial or Parallel options
- Slave – FPGA configured by external source (i.e., a μP)
  - Serial or Parallel options
  - Used for dynamic reconfiguration
  - Can also read configuration memory contents
- Boundary Scan Interface
  - 4-wire IEEE standard serial interface for testing
  - Write and read access to configuration memory
    - ✓ Not available in all FPGAs
    - ✓ Used for dynamic partial reconfiguration
  - Interfaces to FPGA core
    - ✓ Not available in all FPGAs
    - ✓ Connections between Boundary Scan Interface and internal routing network and PLBs (Xilinx provides 2 of these ports)
- Other configuration interfaces in some FPGAs

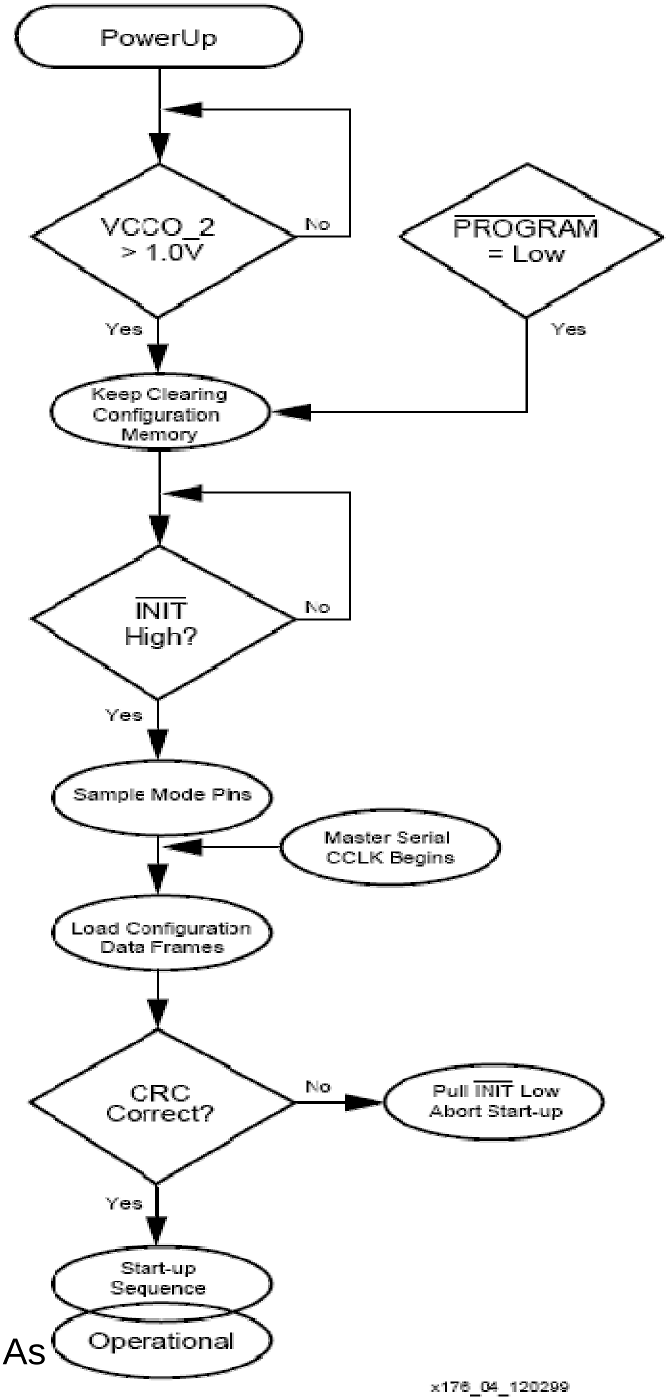
Xilinx Configuration Interface Pins



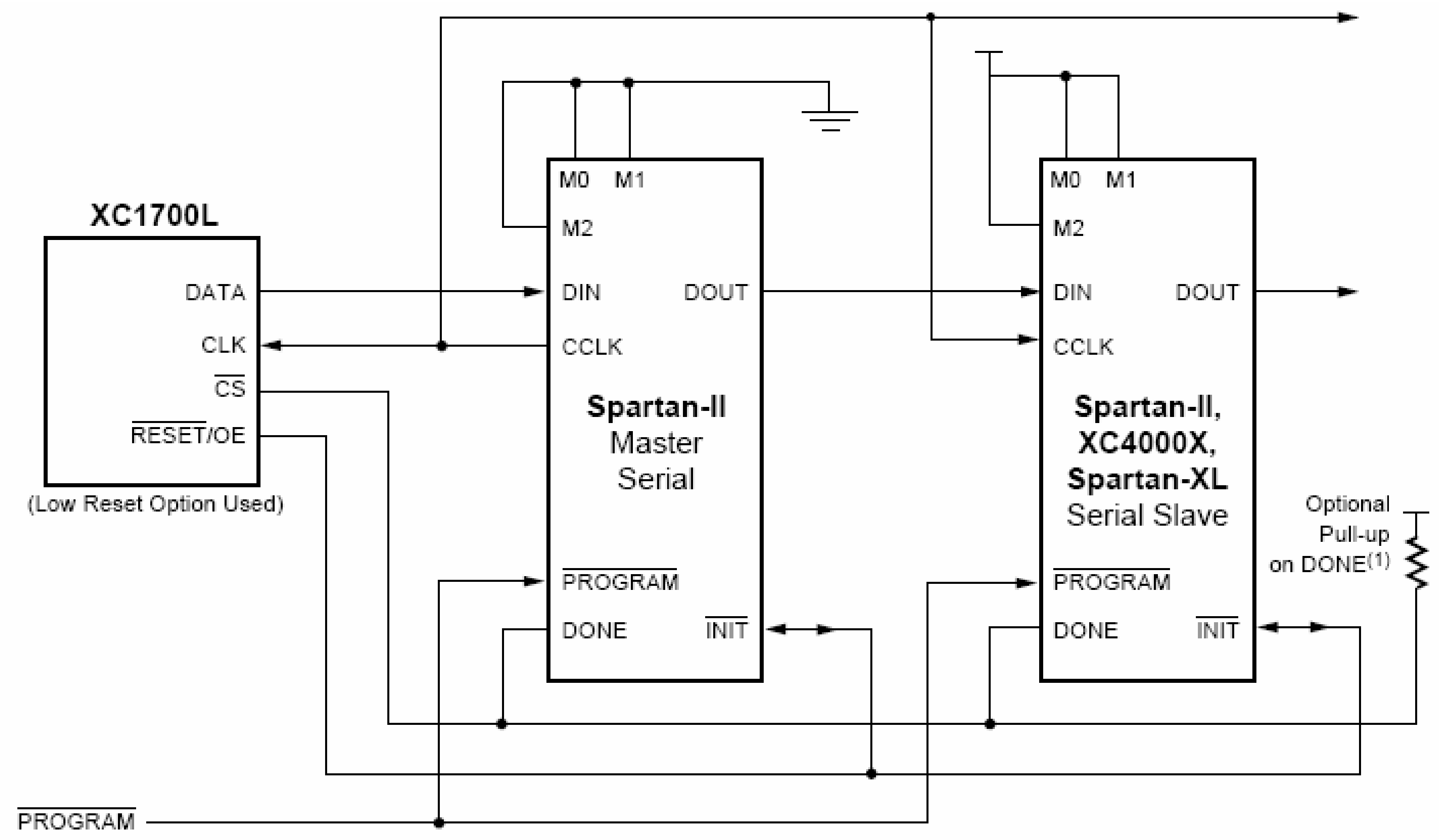
Name	Direction	Driver Type	Description
Dedicated Pins			
CCLK	Input/Output	Active	Configuration clock. Output in Master mode.
$\overline{\text{PROGRAM}}$	Input		Asynchronous reset to configuration logic.
DONE	Input/Output	Active/ Open-Drain	Configuration status and start-up control.
M2, M1, M0	Input		Configuration mode selection.
TMS	Input		Boundary-scan tap controller.
TCK	Input		Boundary-scan clock.
TDI	Input		Boundary-scan data input.
TDO	Output	Active	Boundary-scan data output.
Dual Function Pins			
DIN (D0)	Input/Output	Active Bidirectional	Serial configuration data input.
D[0:7]	Input/Output	Active Bidirectional	Slave Parallel configuration data input, readback data output.
$\overline{\text{CS}}$	Input		Chip Select (Slave Parallel only).
$\overline{\text{WRITE}}$	Input		Active Low write select, read select (Slave Parallel only).
BUSY/ DOUT	Output	Open-Drain/ Active	Busy/$\overline{\text{Ready}}$ status for Slave Parallel (open-drain). Serial configuration data output for serial daisy-chains (active).
$\overline{\text{INIT}}$	Input/Output	Open-Drain	Delay configuration, indicate configuration clearing or error.

Master mode

- Configuration sequence during power-up of device
  - Typically from
    - Serial EPROM
      - Master Serial
    - Parallel EPROM
      - Master Parallel
        - 8-bit
        - 32-bit

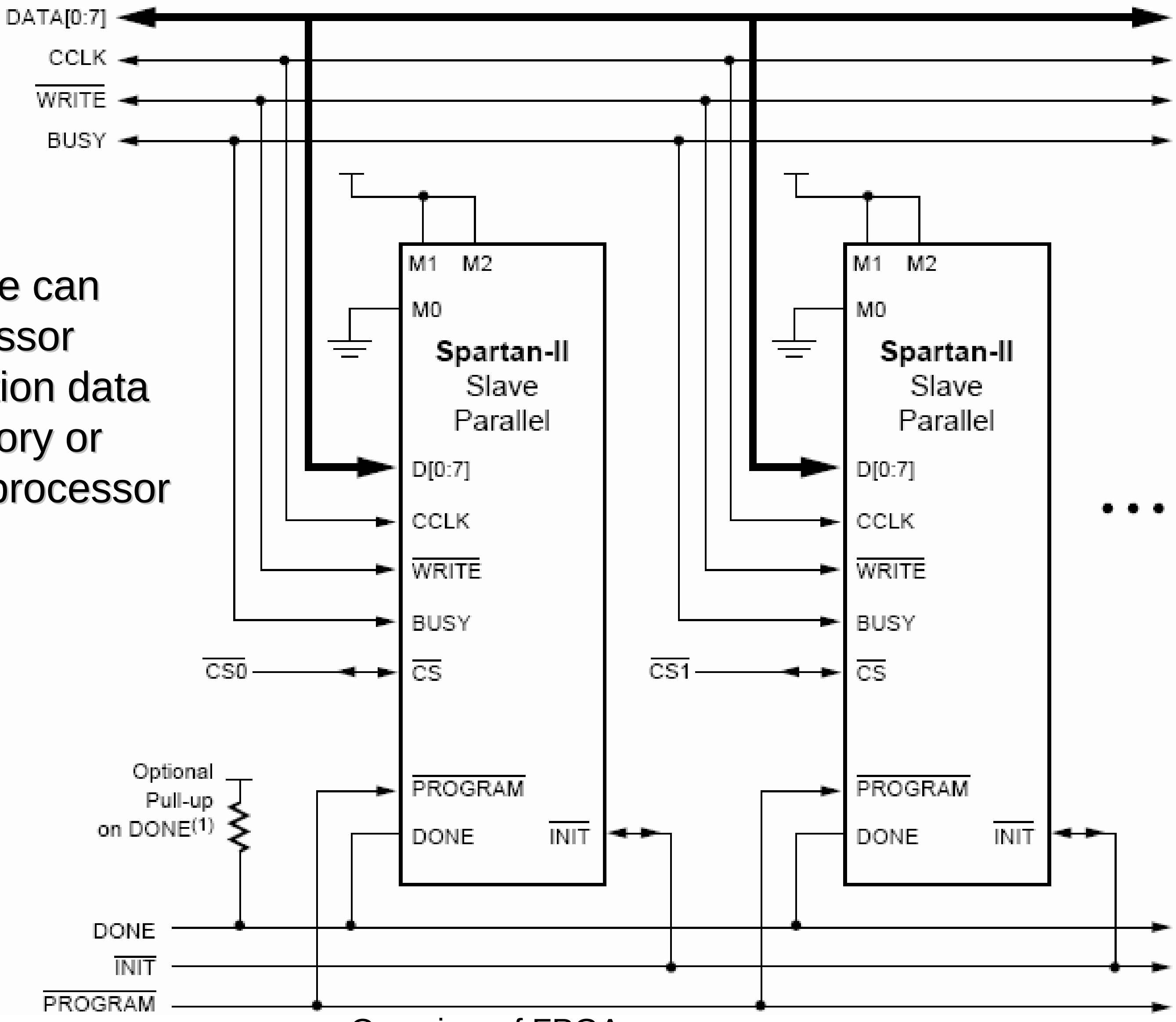


Daisy Chain Configuration



Slave Configuration

External source can be microprocessor with configuration data stored in memory or generated by processor

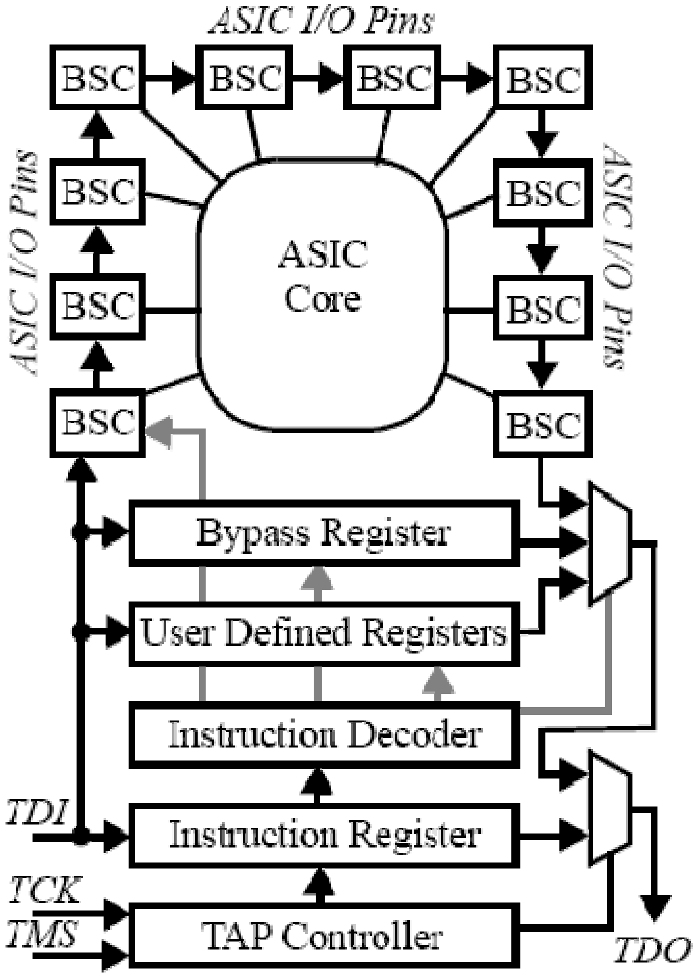


Boundary Scan Interface

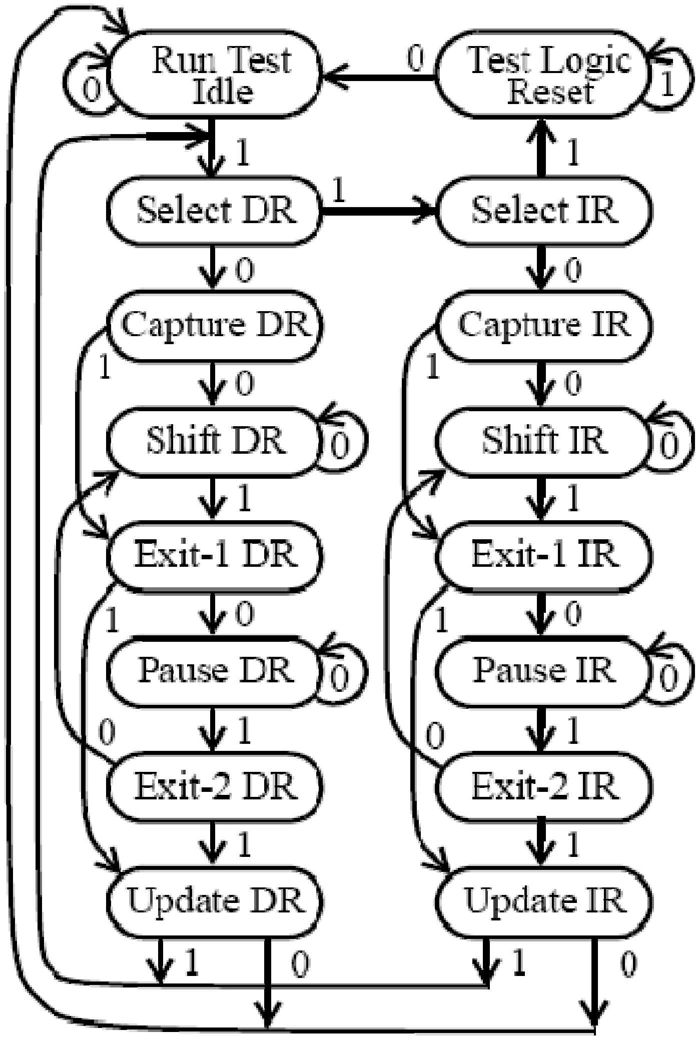
User registers can include:

a) configuration memory

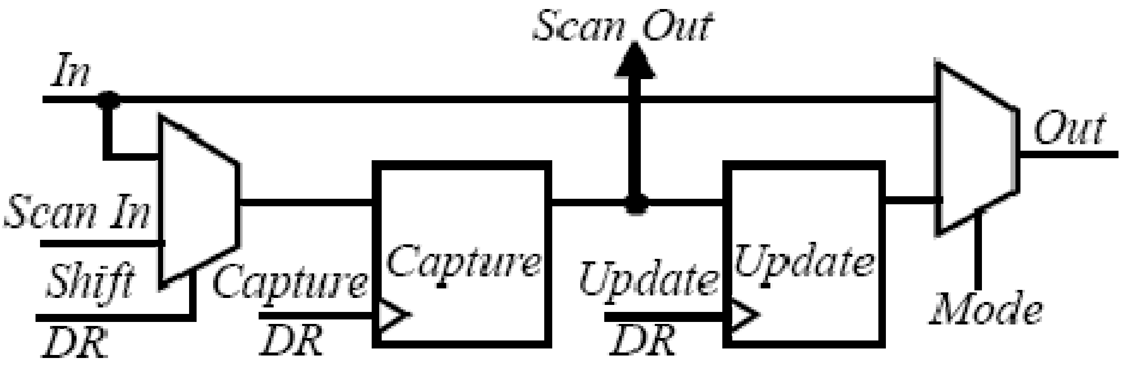
b) access to FPGA core



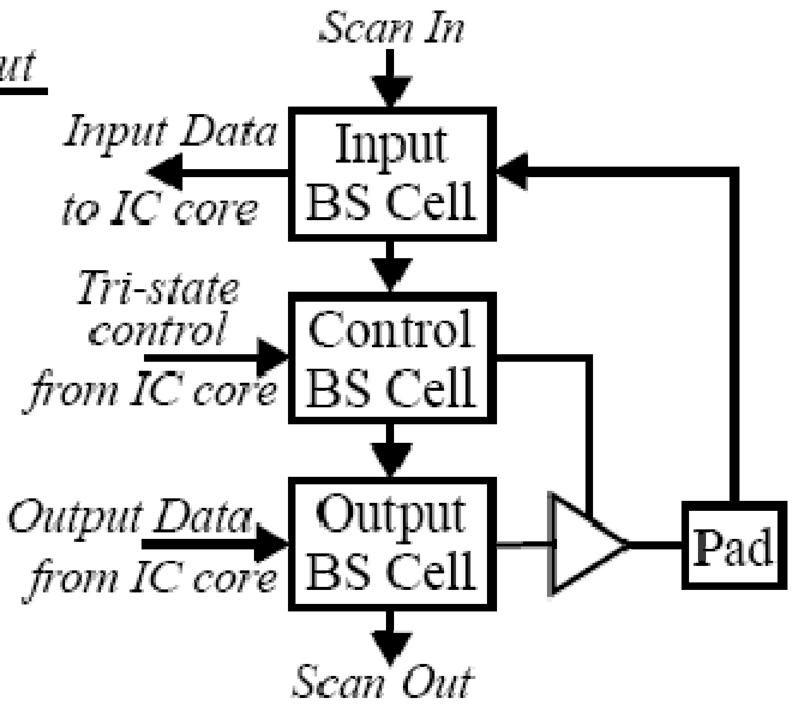
TAP controller state diagram



Typical Boundary Scan Cell



Operation Mode	Data Transfer
Normal	Out <= In
Scan	Capture <= Scan In
Capture	Capture <= In
Update	Update <= Capture



Typical Boundary Scan Cell

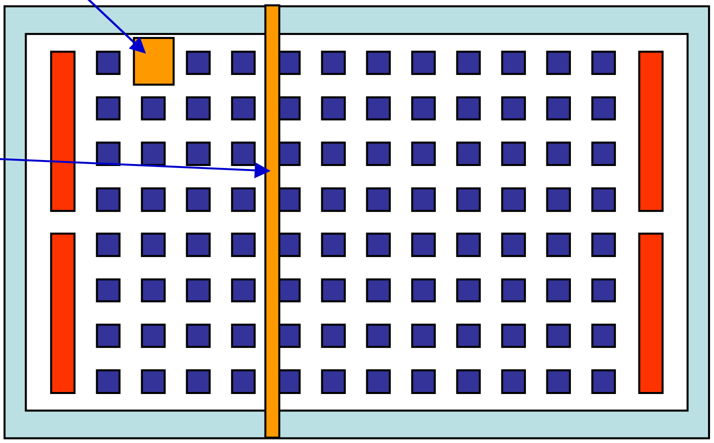
- Primarily designed to test external surface mount connections on printed circuit board
  - EXTEST instruction
    - ✓ Capture logic values at BSCs
      - Output responses from previous test patterns
    - ✓ Shift in test patterns through BSCs
      - As we shift out captured logic values
    - ✓ Update test patterns into BSCs
      - New test patterns are applied
- Optional: can be used to test inside chip
  - INTEST instruction
    - ✓ Same basic procedure as EXTEST but test patterns and responses are to/from device (FPGA) core

Boundary Scan Access to FPGA

- Configuration memory
  - Write and read usually separate instructions
  - Other registers for partial reconfiguration
    - ✓Command, frame address, etc.
- Core access
  - Ports from TAP to PLB and routing resources
    - ✓Good interface for accessing FPGA core
    - ✓Not supported by all FPGA manufacturers
  - Includes TCK, TDI, TDO access to core with active enable when core access instruction is in IR

FPGA Configuration Memorys

- PLB addressable
  - Good for partial reconfiguration
  - X-Y coordinates of PLB location to be written
    - ✓Requires tag to identify which resources will be configured



- Frame addressable
  - Vertical or horizontal frame
  - Access to all PLBs in frame
    - ✓Only portion of logic and routing resources accessible in a given frame
    - ✓Many frames to configure PLBs
      - Major address for column, minor address for frame