Reconfiguration Architectures

Pao-Ann Hsiung
Embedded Systems Laboratory
National Chung Cheng University
Chiayi, Taiwan, ROC.
http://www.cs.ccu.edu.tw/~pahsiung/courses/rc/
pahsiung@cs.ccu.edu.tw

Outline

- Introduction
- Fine-grained Architectures
- Coarse-grained Architectures
- Configuration Architectures
- References

Outline

- Introduction
- Fine-grained Architectures
- Coarse-grained Architectures
- Configuration Architectures
- References

Introduction

- Gerald Estrin proposed the first reconfigurable computer [1]
  - Fixed + Variable
  - Way ahead of its time
- Algotronix CHS2X4
  - First commercial reconfigurable computer
  - Completed in 1991
  - Not a commercial success
  - Bought by Xilinx (inventor of FPGA) and hired the Algotronix staff

Introduction

- Different RC architectures based on
  - Granularity
    - fine vs. coarse
  - Configuration time
    - static vs. dynamic
  - Configurable capacity
    - full vs. partial
  - Configuration model
    - column vs. tile

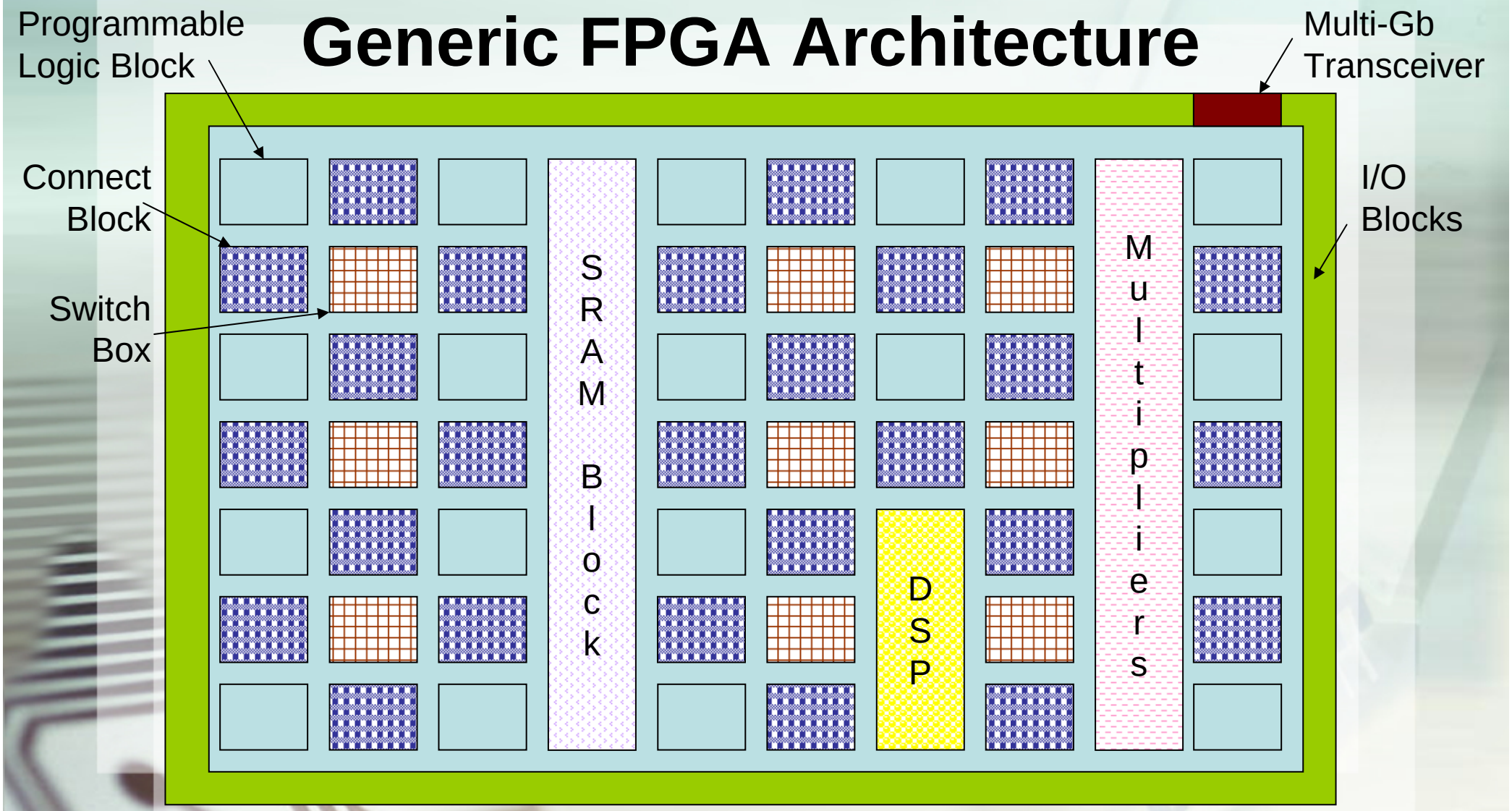
Outline

- Introduction
- Fine-grained Architectures
- Coarse-grained Architectures
- Configuration Architectures
- References

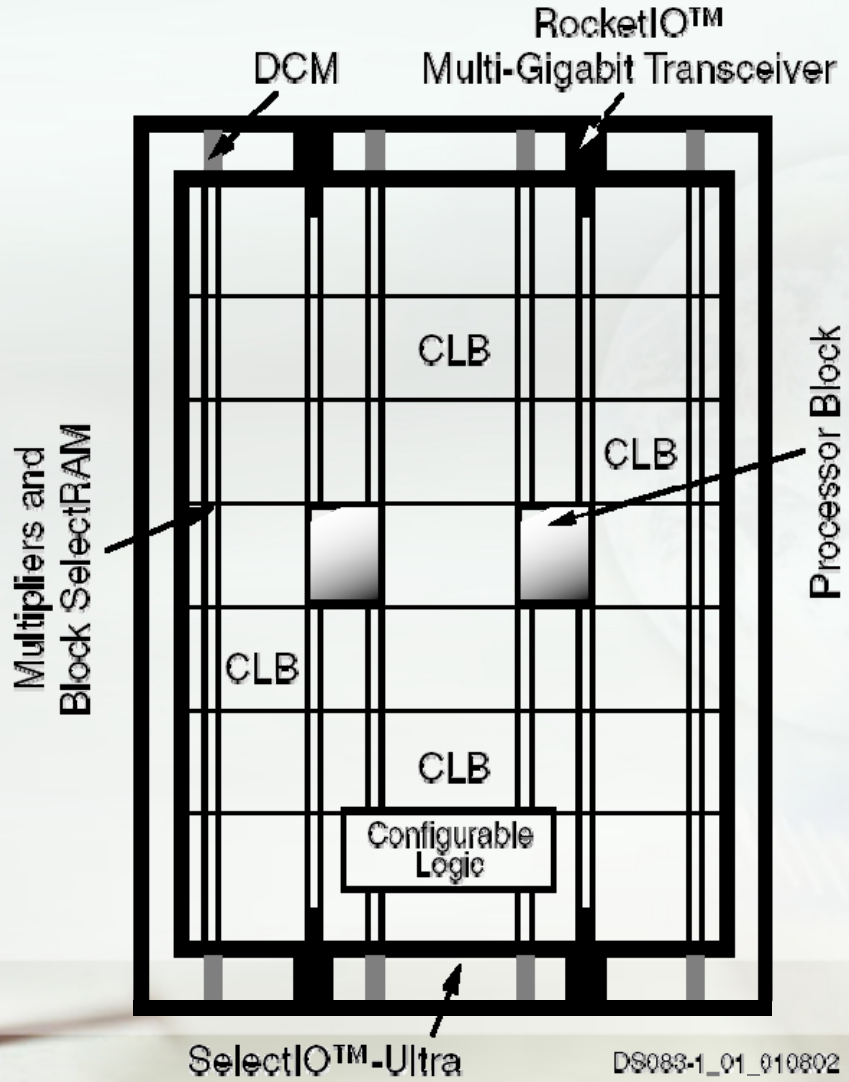
Fine-grained Architectures

- Field Programmable Gate Arrays (FPGA)
  - Arrays of simple logic functions and memories
  - Programmable interconnection networks
  - Enough logic density to perform significantly complex and large number of functions
    - 207,360 6-LUTs, 135,200 ALMs (8-Frac LUT)
  - Latest fabrication technology (65 nm)
  - Embedded Special Blocks
    - DSP, Multi-Gb serial I/O, embedded MP, SRAM

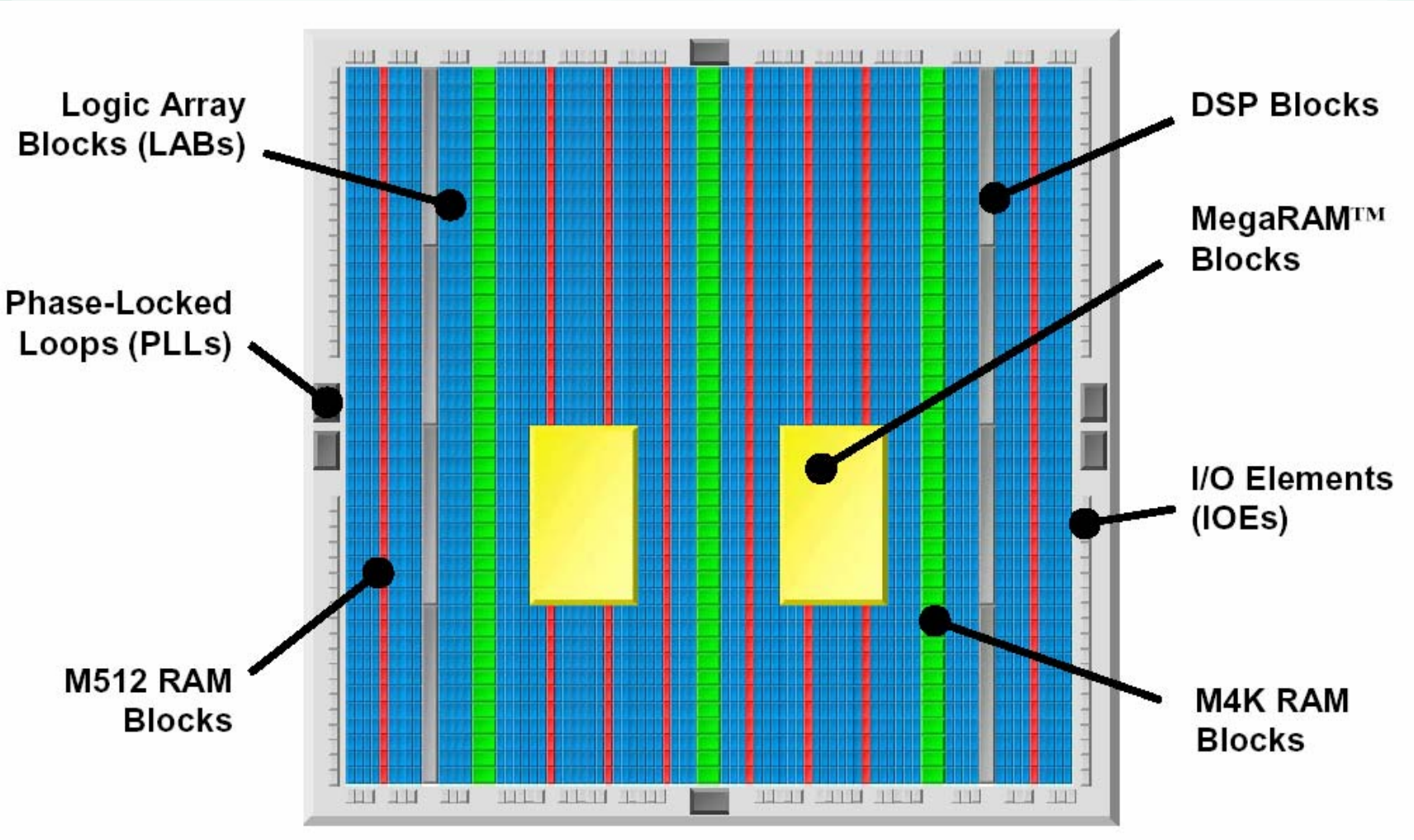
Generic FPGA Architecture



Xilinx Virtex Architecture



Altera Stratix Architecture



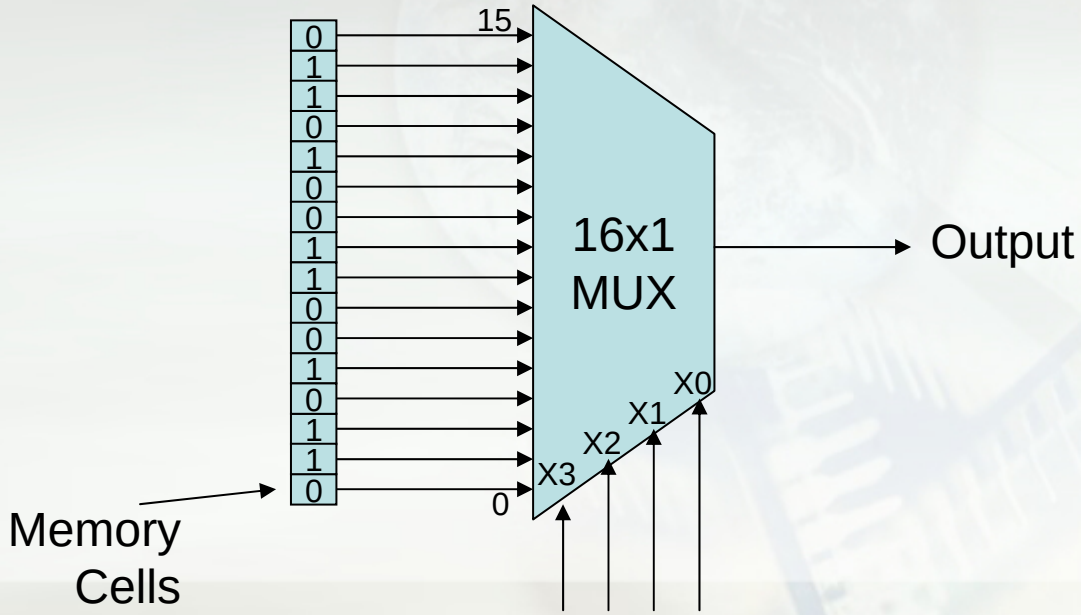
FPGA Architecture

- Combinational Logic
  - Lookup Table (LUT)
- Programmable Logic Block (PLB)
  - PLB = LUT + FF
- Programmable Logic Cluster (PLC)
  - PLC = PLB + Interconnects
- FPGA
  - FPGA = PLC + Routing Blocks + I/O Blocks + Special Blocks

FPGA Architecture

- Combinational Logic: LookUp Table (LUT)
- *N* address lines, 2^N memory locations

- 4-input LUT configured as the XOR function



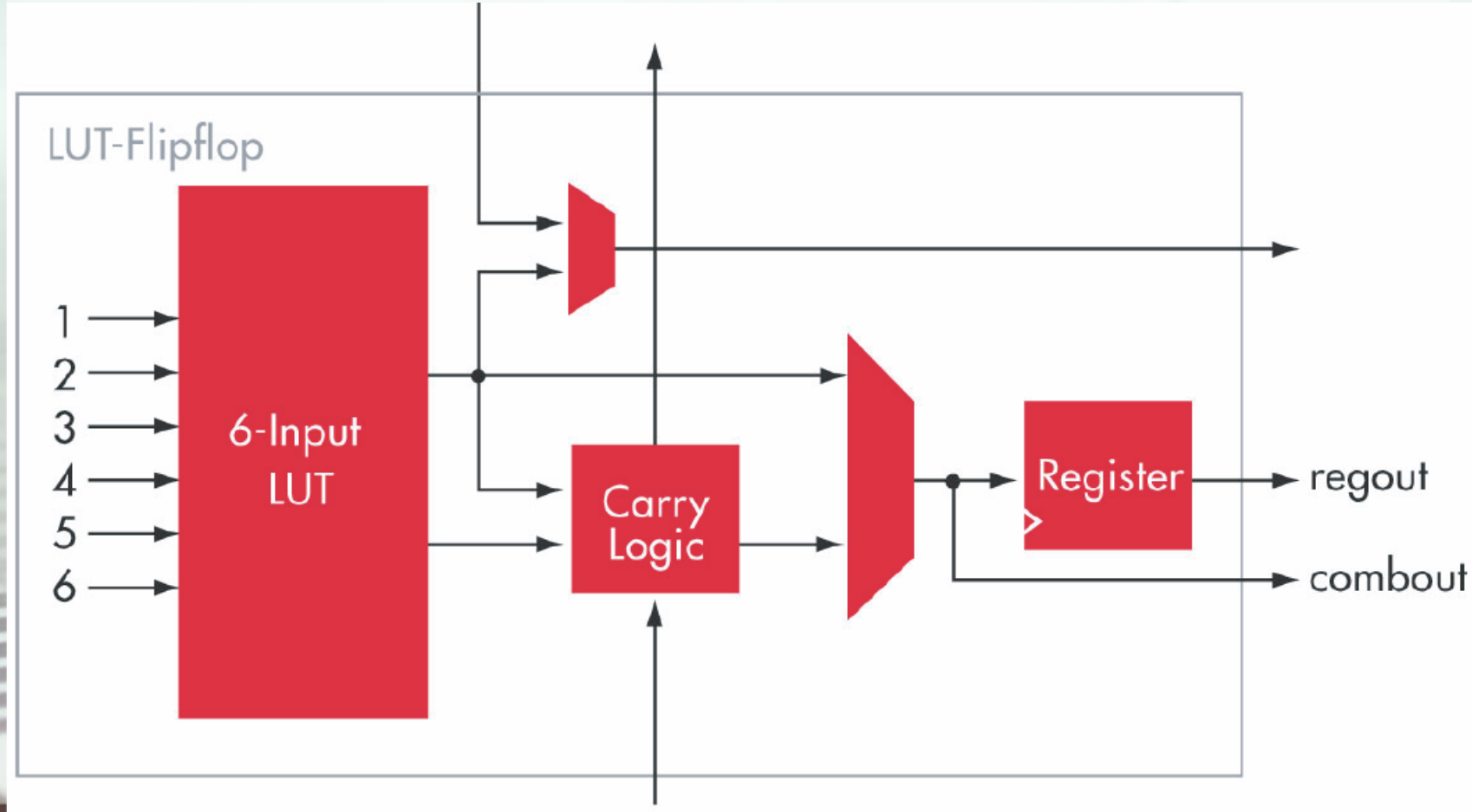
FPGA Architecture

- Programmable Logic Block

Basic Design
- – Combinational Logic: LookUp Table (LUT)
- – Memory: Flip-Flop (FF) or Latch

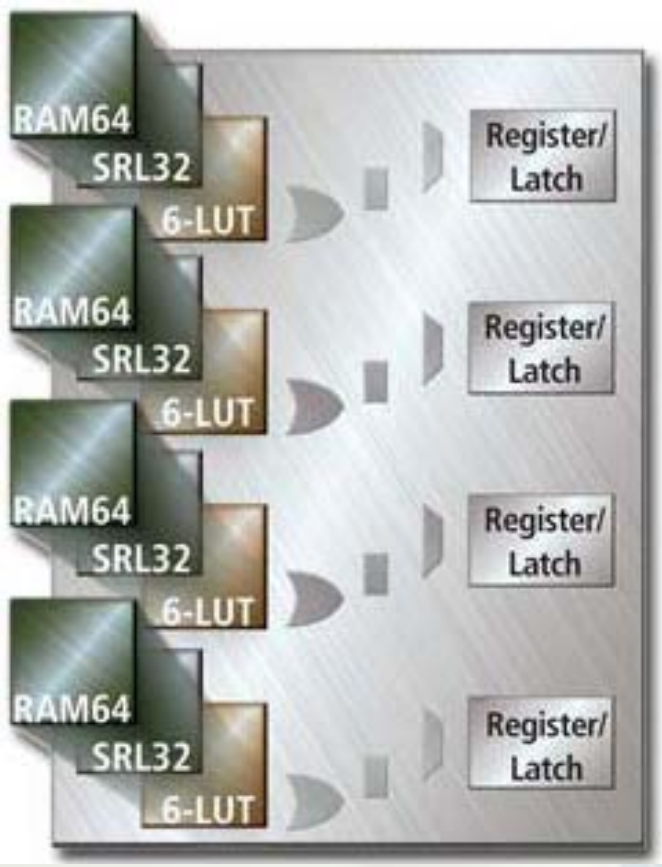
Extra Logic Design
- – Arithmetic Logic: Carry Logic, Adder, …
- – Control Logic: Asynchronous/synchronous sets and resets
- – Trigger Logic: negative-edge or positive-edge

Xilinx LUT-FF Pair in Virtex-5



Xilinx Slice in Virtex-5

- 4 LUT-FF pairs



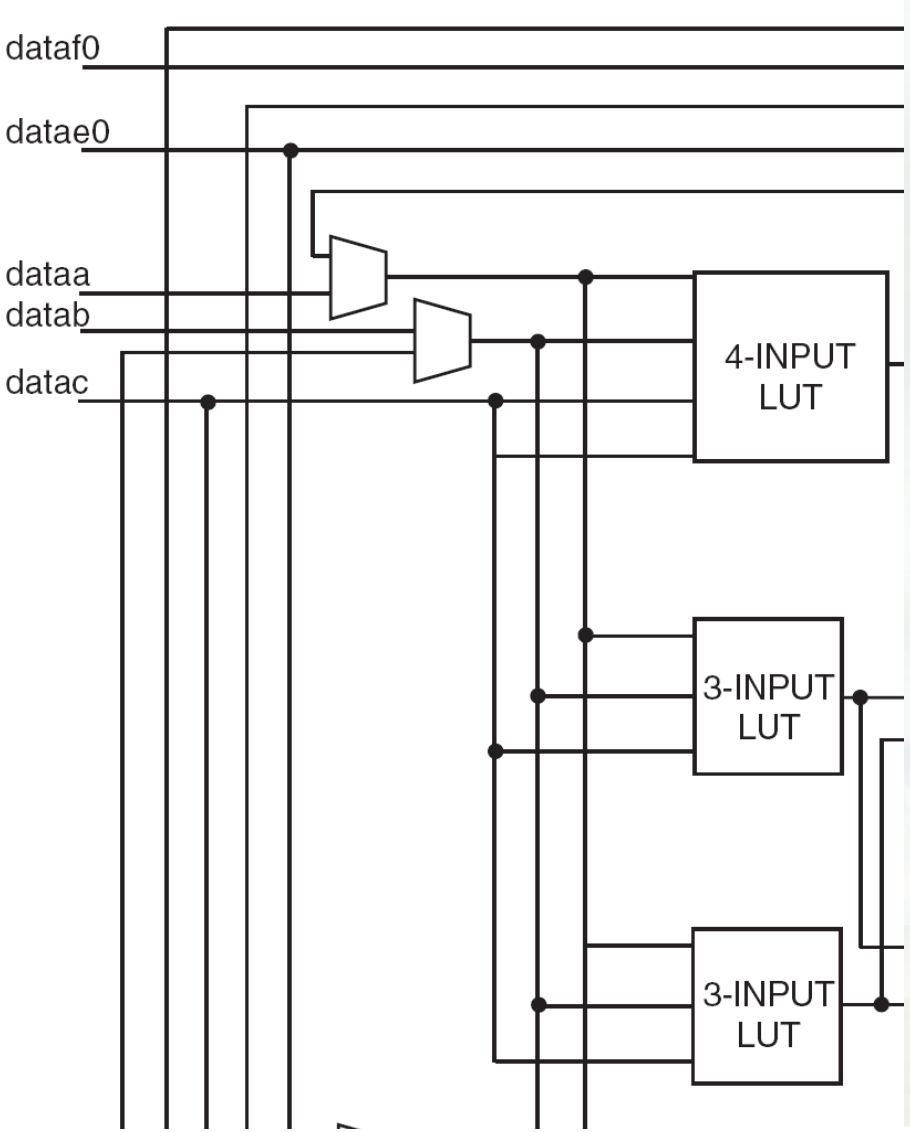
Altera Fracturable LUTs

Shared:

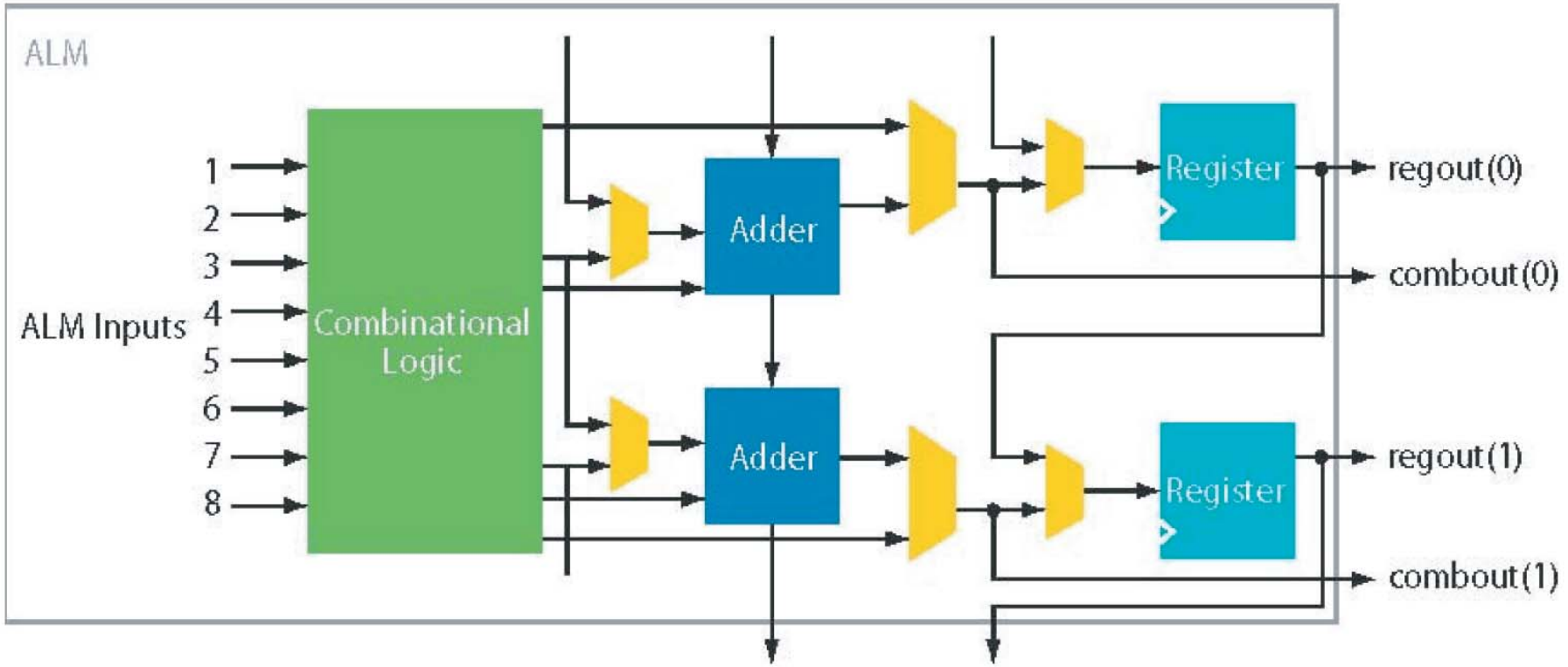
dataa, datab, datac, datad

Unshared:

datae0, datae1, dataf0, dataf1



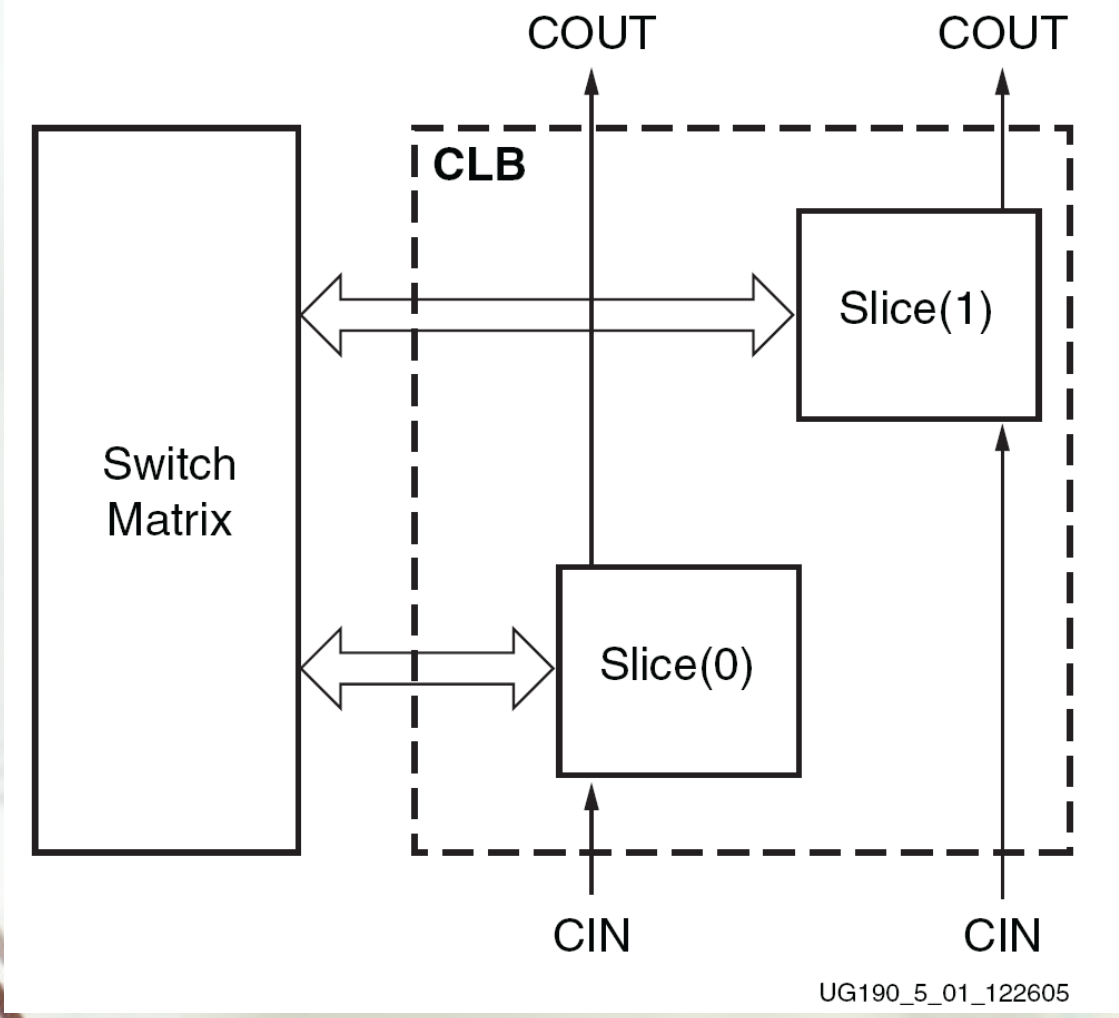
Altera Adaptive Logic Module (ALM) in Stratix III



FPGA Architecture

- Programmable Logic Cluster
  - Xilinx Configurable Logic Block (CLB)
    - Virtex-4
      - Slice: Two 4-LUT, two FF
      - CLB: 4 slices
    - Virtex-5
      - Slice
        - » SLICEL: Four 6-LUT, four FF
        - » SLICEM: 6-LUT can be configured as 64 bit LUTRAM or 32-bit shift registers
      - CLB: 2 slices with 0 or 1 SLICEM (i.e., 256 bits distributed RAM or 128 bit shift registers)

Xilinx Configurable Logic Block (CLB)

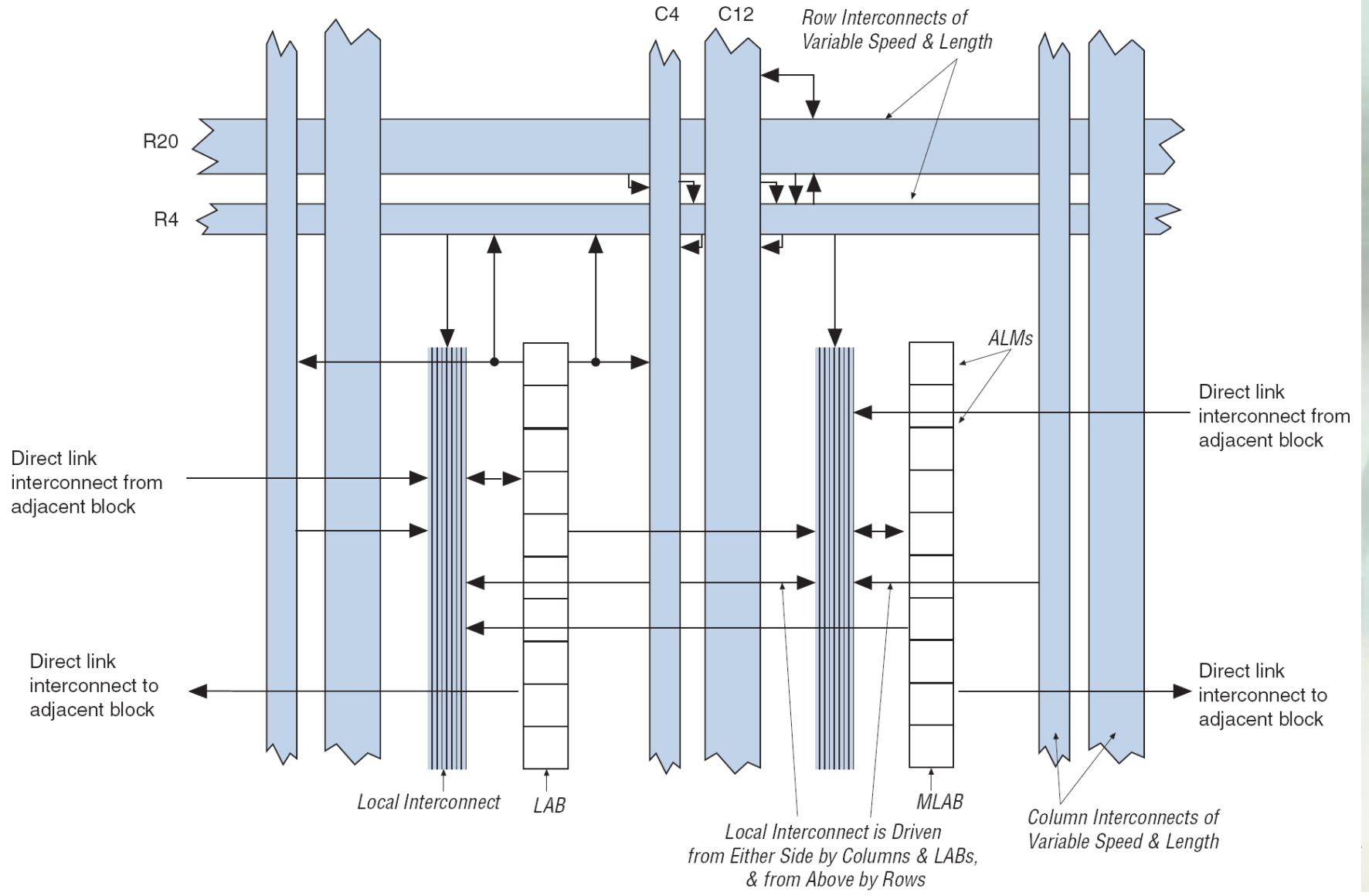


FPGA Architecture

- Programmable Logic Cluster
  - Altera Logic Array Block (LAB)
    - Stratix III
      - ALM: Adaptive Logic Module
        - » Adaptive LUT: four 3-LUT, two 4-LUT (all 6-input functions, certain 7-input functions, two independent functions of (5, 3), (4, 4), (4, 3), (3, 3) inputs)
        - » Two programmable registers
        - » Two full adders
        - » A carry chain, a shared arithmetic chain, a register chain
      - LAB: 10 ALMs
      - MLAB: 10 ALMs (each ALM can be configured as a 64-bit dual port SRAM)

(2007 Copyright @Pao-Ann Hsiung)

Altera Logic Array Block (LAB)



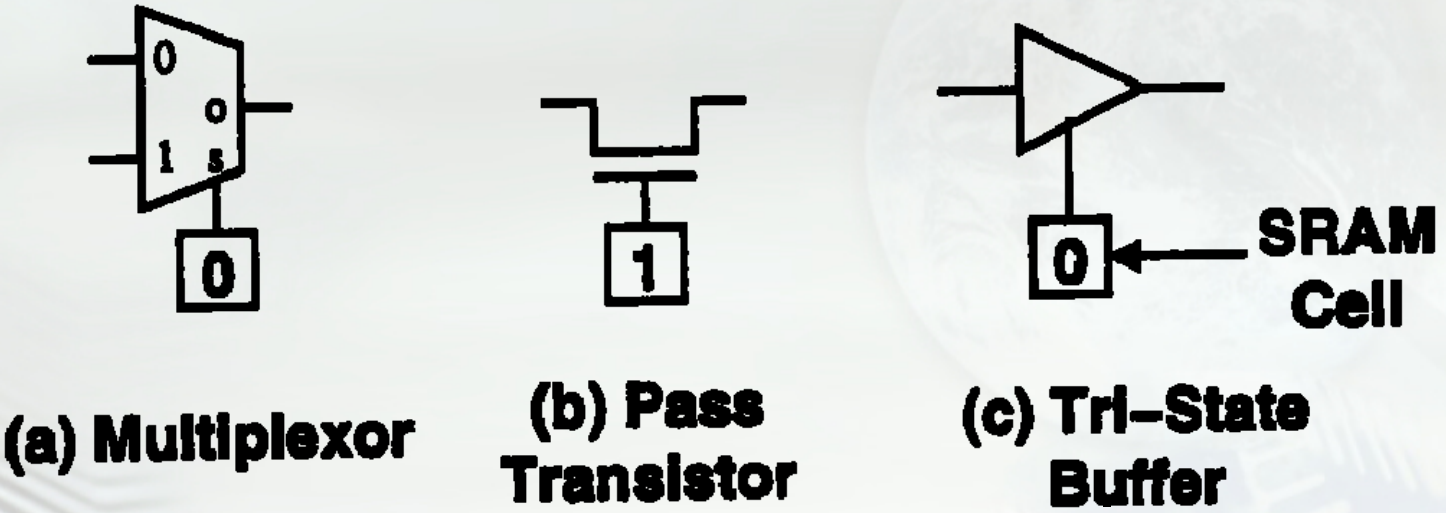
Programmable Logic Cluster Size

Device	Year	LUT	Cluster	Size
Xilinx XC2000	1985	4	CLB	1
Xilinx XC3000	1987	4	CLB	2
Xilinx XC4000	1990	3 & 4	CLB	1 & 2
Altera FLEX 8K	1992	4	LAB	8
Altera FLEX 10K	1995	4	LAB	8
Xilinx Virtex	1998	4	CLB	4
Altera Apex 20K	1998	4	LAB	10
Xilinx Virtex-II	2000	4	CLB	8
Altera Apex II	2001	4	LAB	10
Altera Stratix	2002	4	LAB	10
Xilinx Virtex-4	2004	4	CLB	8
Altera Stratix II	2004	3 & 4	LAB	24 & 16
Xilinx Virtex-5	2006	6	CLB	8
Altera Stratix III	2006	3 & 4	LAB	40 & 20

(2007 Copyright @Pao-Ann Hsiung)

Reconfigurable Routing

- 3 Basic Programmable Switch Types



What is a pass transistor?
http://www.everything2.com/index.pl?node_id=1324909

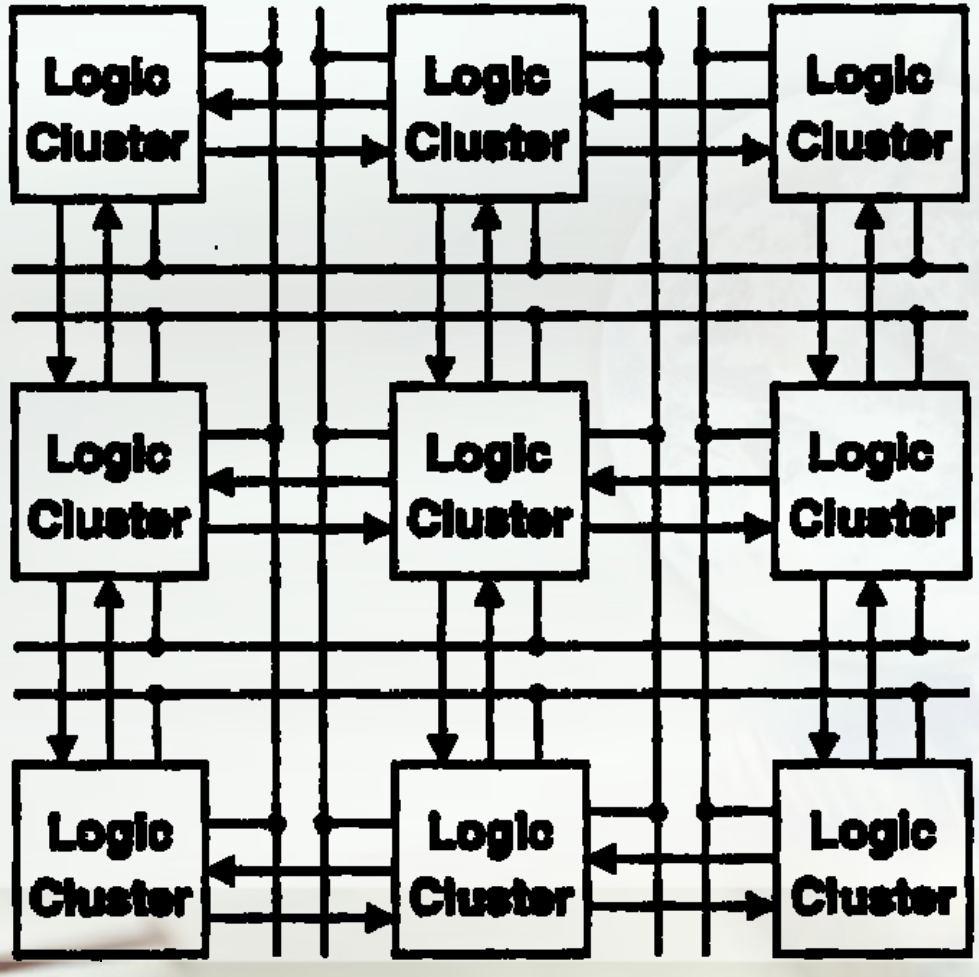
Reconfigurable Routing Architectures

- Mesh based cellular architectures
- Island style architectures
- Long line routing architectures
- Row based architectures
- Diagonal architectures
- MultiTrack interconnect architectures
- Systolic architectures

Mesh based Cellular Routing

- Rich in local routing between logic clusters
- Few global routing segments
- Very simple logic clusters
  - Can be used as part of routing network
- Examples
  - Algotronix CAL FPGA (Xilinx XC6200)
  - CLi/Atmel 6000 FPGA
  - Plessey/Pilkington ERA
  - Triptych, Montage

Mesh based Cellular Routing



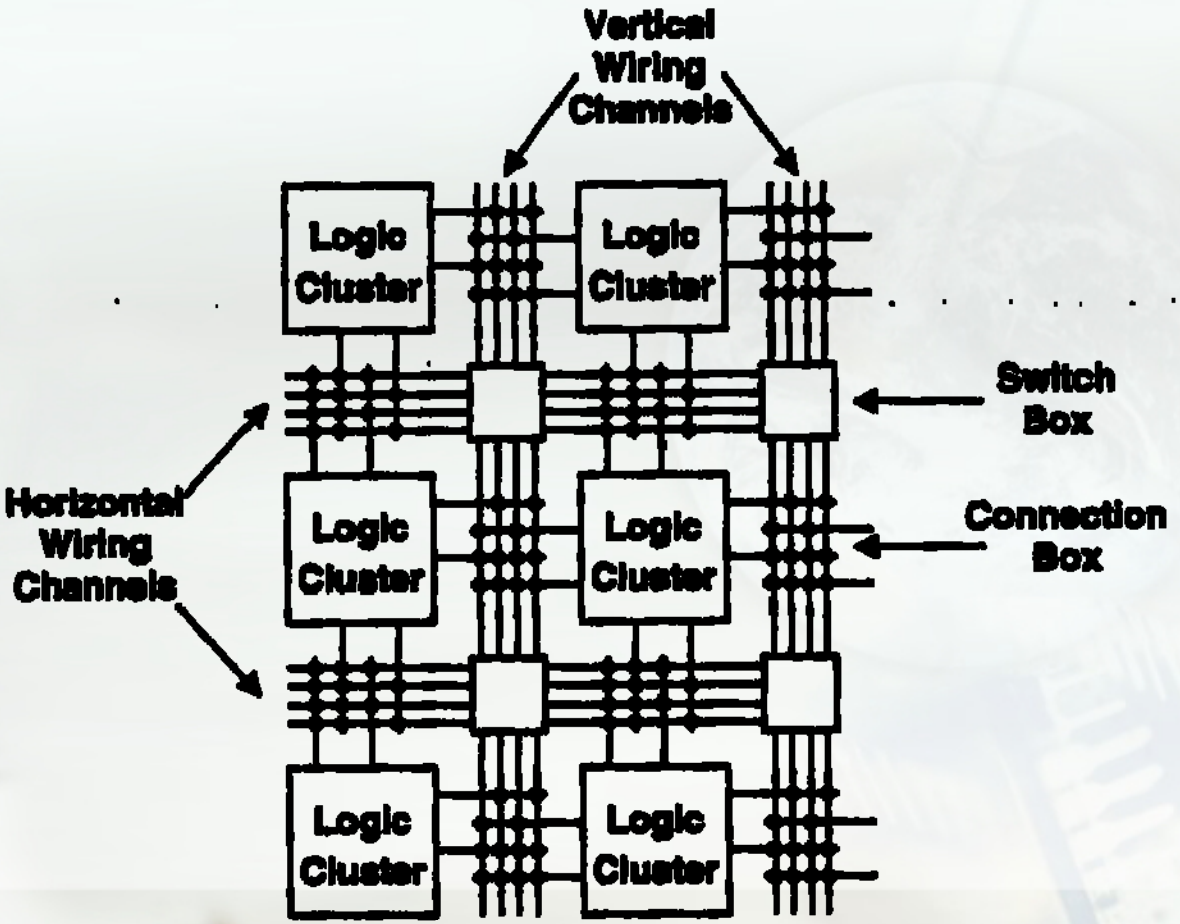
Mesh based Cellular Routing

- Advantages
  - Simple fabrication due to regularity
  - Cell Matrix for nanotechnology fabrication
- Disadvantages
  - Long combinational path delays
  - Poor routing results
  - Large routing resource requirements

Island Style Routing

- Segmented horizontal and vertical routing channels
- Connection boxes
  - For logic clusters to connect to network
- Switch boxes
  - For connection between two segments
- Powerful logic clusters
- Example
  - Xilinx FPGA

Island Style Routing



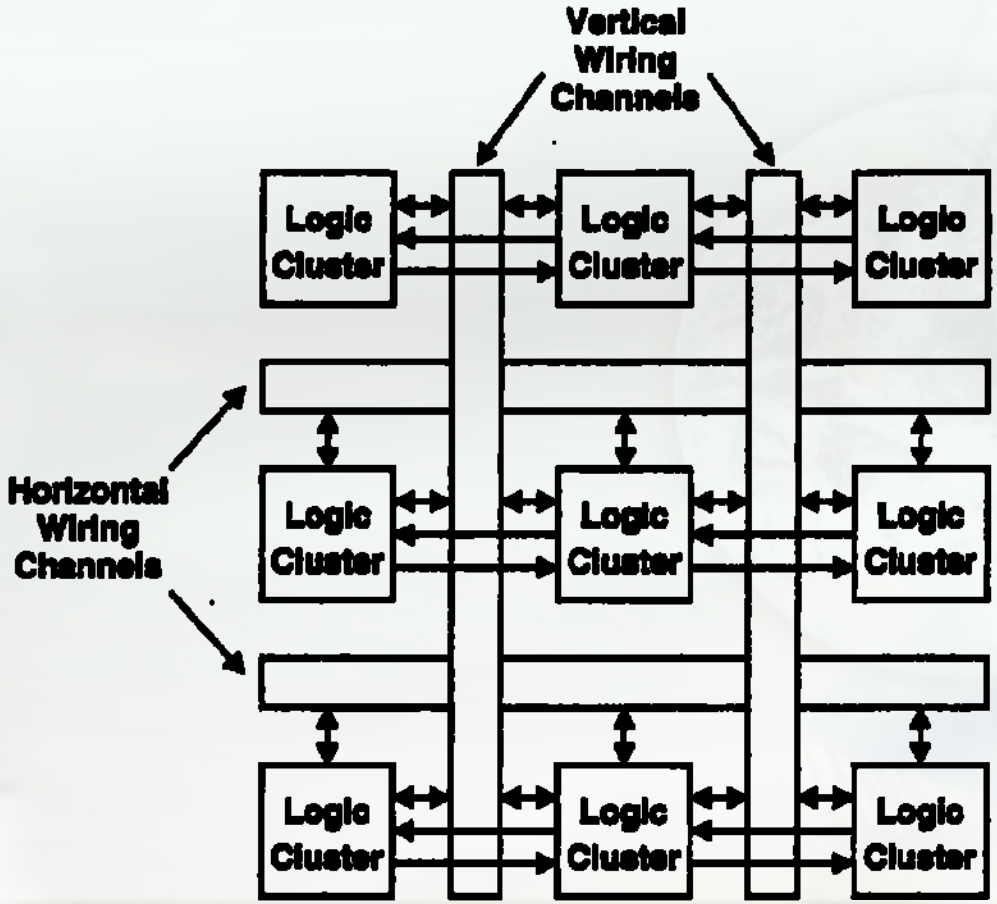
Island Style Routing

- Advantages
  - Versatile: can achieve good routing results
  - Speed: small delays through efficient mappings
- Disadvantages
  - Expensive: requires large amount of routing resources

Long Line Routing

- Horizontal and vertical routing channels
  - Multiple wires per channel
  - Span the width or height of entire chip
- To connect any two logic clusters
  - Only 1 horizontal and 1 vertical lines are needed
  - Transition between a horizontal and a vertical line
    - By using internal routing of a logic cluster at intersection
- Examples
  - Altera FPGAs, Actel ProASIC FPGA

Long Line Routing



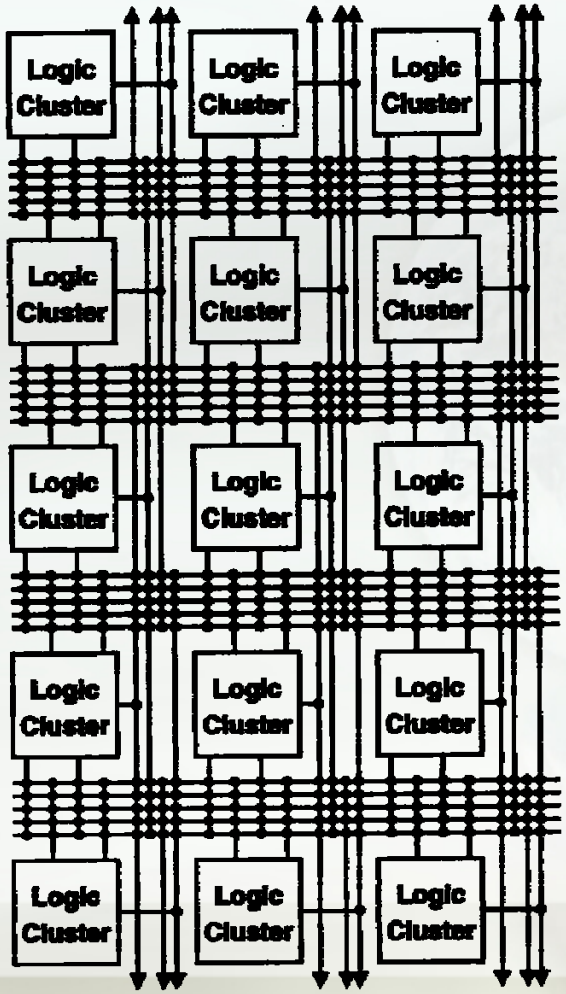
Long Line Routing

- Advantages
  - Simpler routing strategies
  - Fewer routing resources
- Disadvantages
  - Long wires cause delays
    - Altera Stratix, Stratix II introduced smaller length segments

Row based Routing

- Mostly horizontal interconnect channels
  - Segmented wires to reduce routing delays for short paths
- Few vertical channels
- One time programmable FPGAs
- Example
  - Actel Act-1 FPGA

Row based Routing



Row based Routing

- Advantages
  - Simple routing
  - Fewer routing resources
- Disadvantages
  - Inflexible
  - Inefficient mapping to routing resources

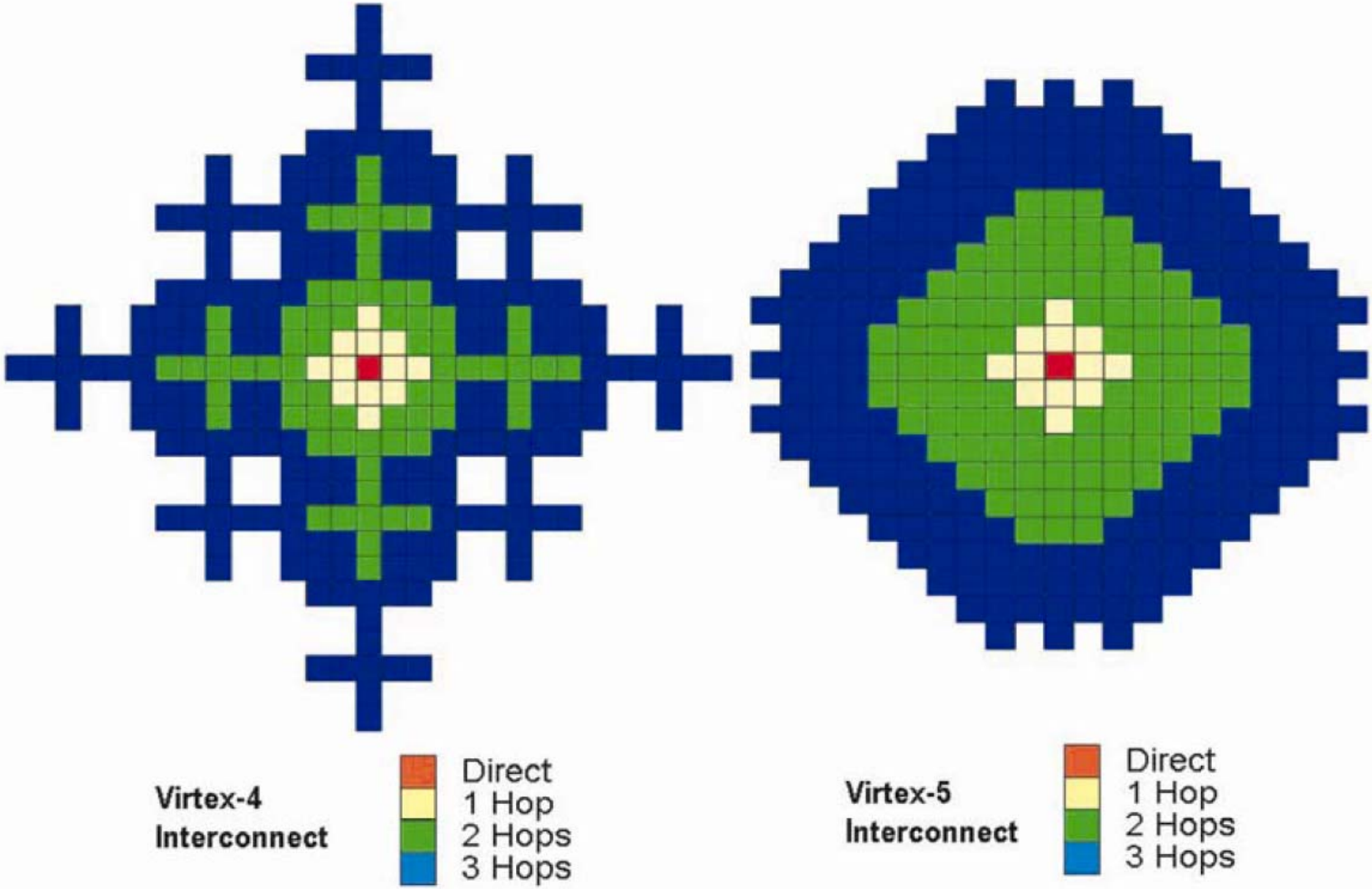
Predictability vs. Connectivity

- Predictability
  - Equal distance ➔ equal routing length
  - Xilinx Diagonal Routing
- Connectivity
  - Larger number of connections in fewer number of hops
  - Altera MultiTrack Interconnect Routing

Diagonal Routing

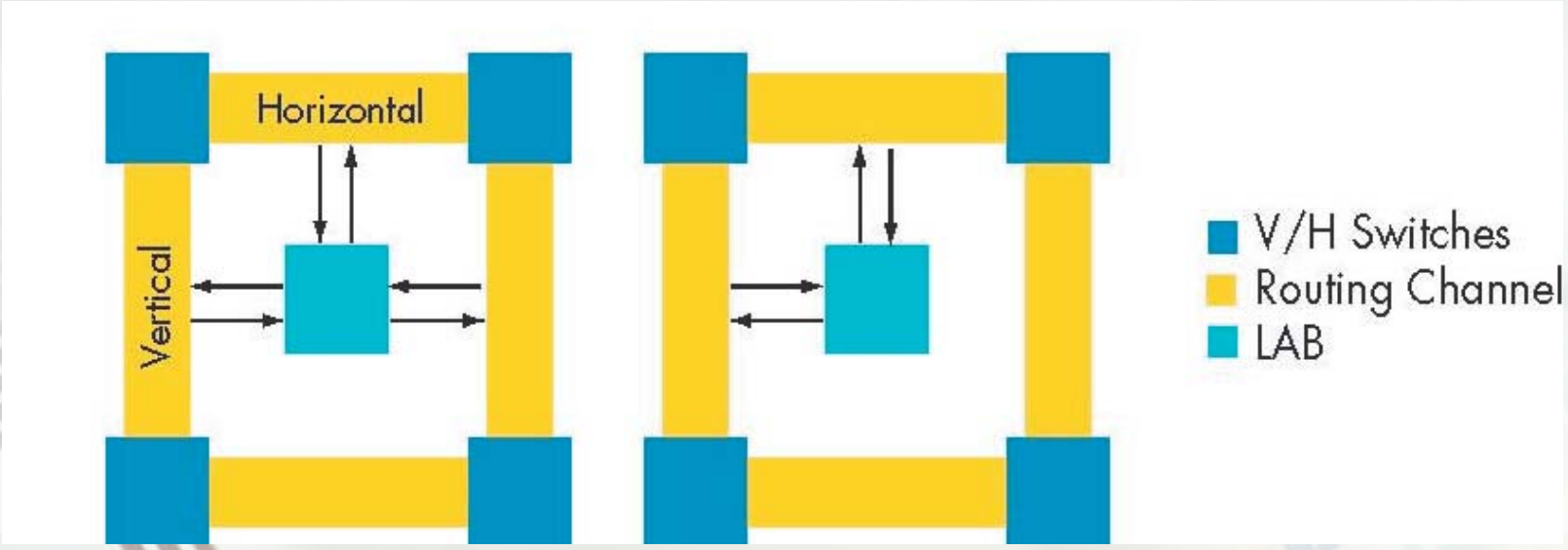
- Block-to-block connectivity with minimal hops
  - More logic connections in fewer hops
- Regular routing pattern allows more optimal routes
- Fast, predictable routing based on distance

Diagonal Routing in Xilinx Virtex

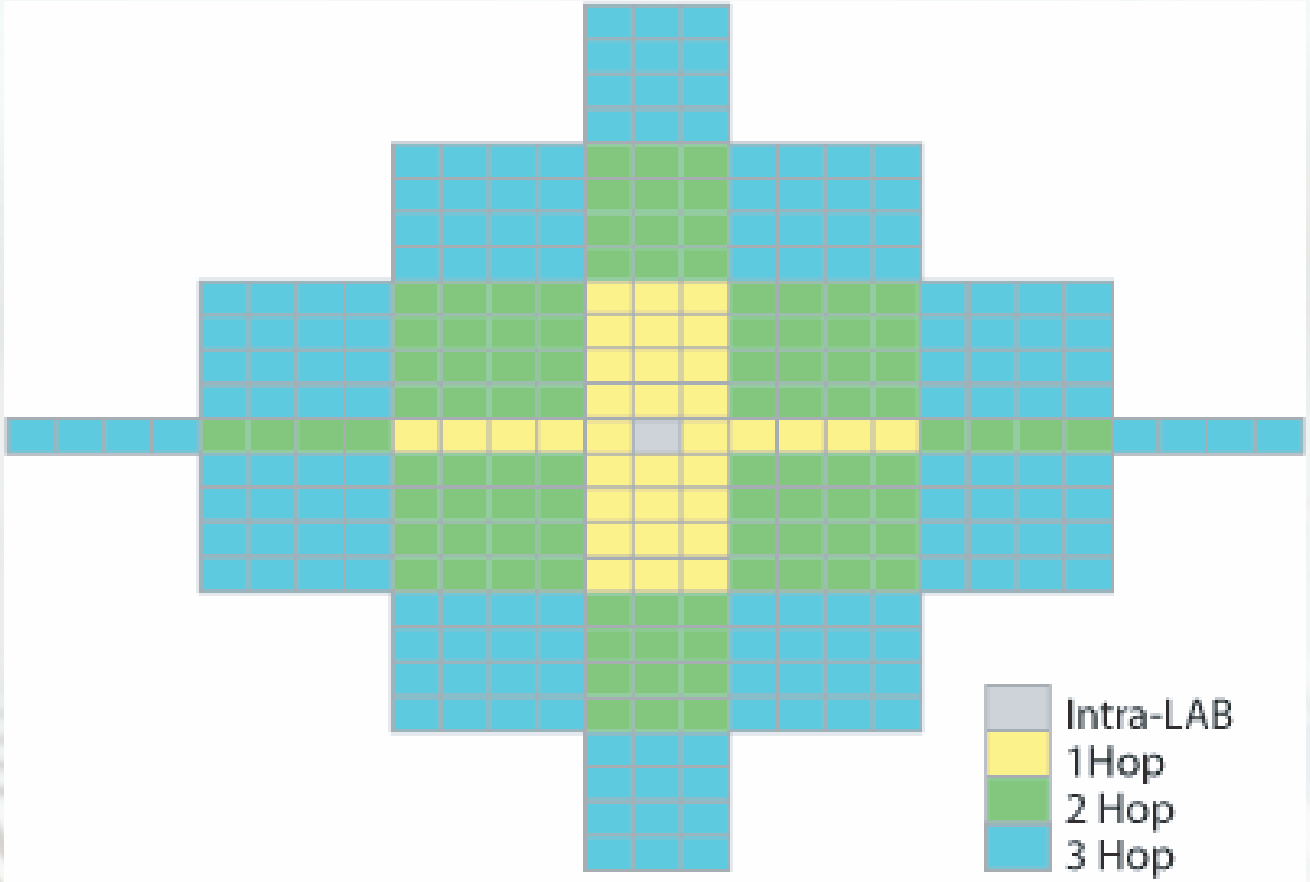


MultiTrack Interconnect

- Altera Stratix series
  - Larger connectivity to all wires on multiple sides



Multitrack Interconnect



Stratix III vs. Virtex-5 Connectivity

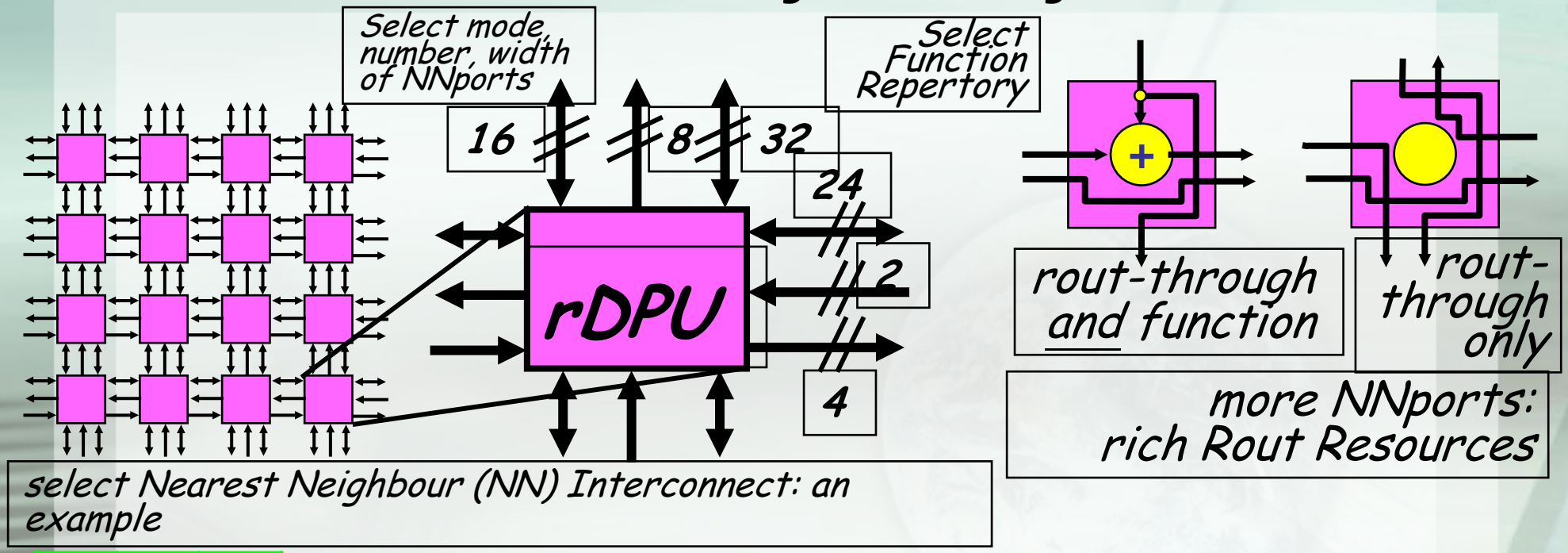
Hops	Number of LABS/CLBs Reachable		Number of LEs Reachable		Ratio of Stratix III LEs to Virtex-5 LEs
	Stratix III	Virtex-5	Stratix III *(1)*	Virtex-5	
1	34	12	850	132	6.4
2	96	96	2,400	1,056	2.3
3	160	180	4,000	1,980	2.0
Total	290	288	7,250	3,168	2.3

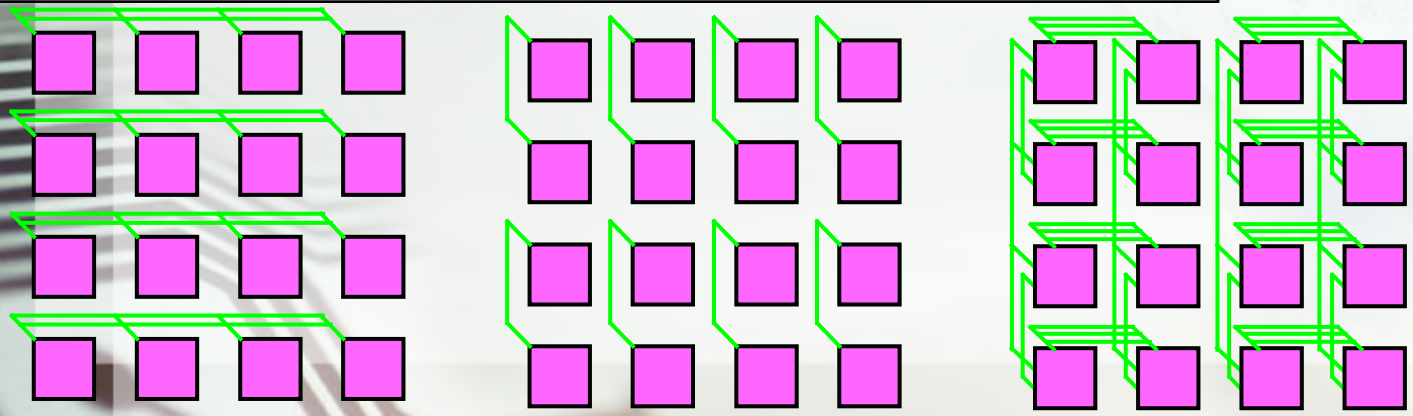
1 ALM = 2.5 LEs, 1 LE = 4-LUT-FF pair

Systolic Routing

- Process data in rhythmic fashion
- Data passes through many processing elements before returning to memory
- Data streams driven by data counter (no PC)
- Kung and Leiserson [1978] proposed systolic arrays for routing in VLSI
- Example
  - KressArray by Rainer Kress

KressArray Family





Examples of 2nd Level Interconnect: layouted over rDPU cell - no separate routing areas !

http://kressarray.de

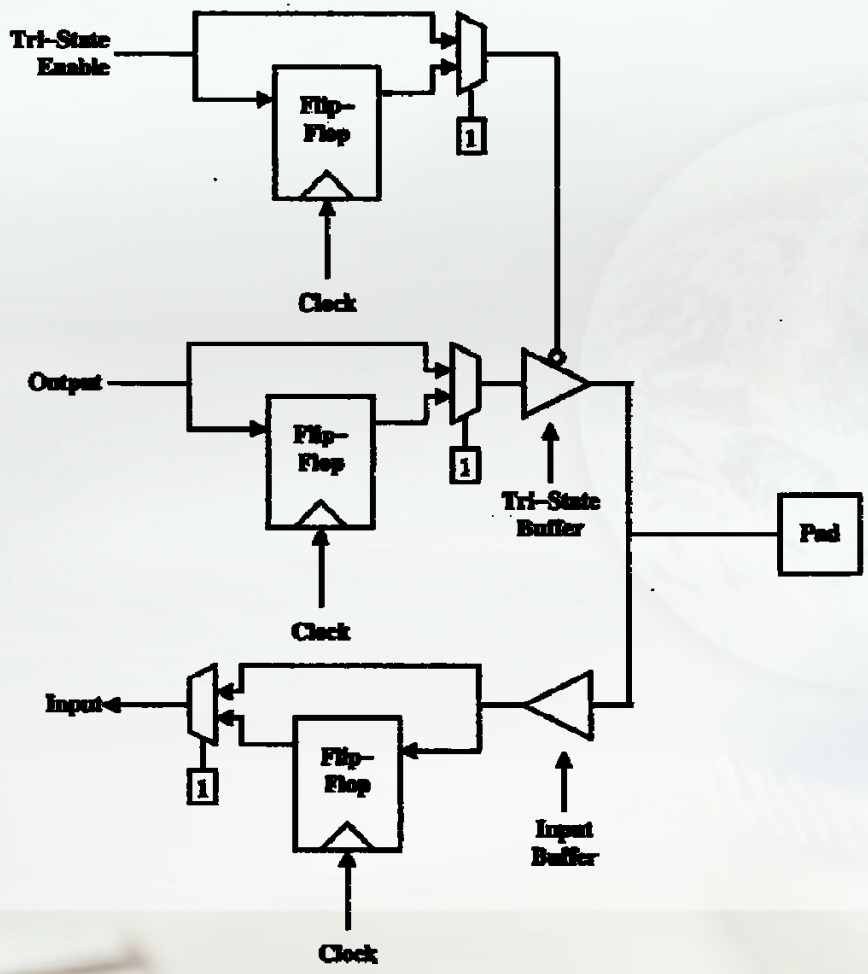
Reconfigurable Routing

- Basically, it is a tradeoff between
  - Routing simplicity vs. routing versatility
    - The more versatile a routing is, the more complex it becomes
  - Resource area vs. communication performance
    - The higher communication performance a routing has, the more routing resource area is required

Programmable I/O

- Three buffers
  - Tri-state buffers for outputs
  - Tri-state enable signal buffer
  - Input buffers for inputs
- Buffers can be registered individually or left unregistered
- Enhancements made in modern I/O blocks

Programmable I/O Block



Enhancements in Programmable I/O Block

- Xilinx Virtex-5 IOB
  - Variations of I/O signaling standards
    - Digitally controlled impedance (DCI) active termination (instead of terminating resistors) to eliminate transmission line effects
    - Differential signaling to improve signal integrity
    - 40 I/O per bank (more independent standards)
    - 1,200 packaged SelectIO (more parallel I/O)
  - DDR (double-data rate) registering
  - Programmable input/output delays

Specialized Function Blocks

- Embedded memories
- Embedded DSP blocks
- High speed serial I/O
- Embedded microprocessors

Embedded Memories

- Flip-flops
- LUTs with memory capabilities
- SRAM blocks

Embedded Memory: Flip-flops

- Flip-flops
  - Available in each programmable logic block
    - Xilinx LUT-FF pair: 1 in each pair
      - 8 in each Xilinx Virtex-5 CLB
    - Altera ALM: 2 in each ALM
      - 20 in each Altera LAB
  - Inefficient for creating memories of large depth

Embedded Memory: LUTs

- LUTs
  - used as (a)synchronous RAM, dual-ported RAMs, shift registers
  - 1st FPGA: Xilinx XC4000
  - Xilinx Virtex 5 Slice (with 4 LUT-FF pairs)
    - 64x4 = 256 bit distributed RAM
    - 32x4 = 128 bit shift register
  - Altera Stratix III ALM
    - 64 bit dual port SRAM

Embedded Memory: SRAM blocks

- 1st FPGA with SRAM blocks: Altera FLEX 10K
- Altera Stratix III TriMatrix EMB provides totally 16,272 Kbits embedded SRAM at 600 MHz with 3 granularities
  - 144-Kbit M144K blocks: for code storage, packet buffer, video frame buffer
  - 9-Kbit M9K blocks: for general purpose memory applications
  - 640-bit MLAB blocks: for filter delay lines, small FIFO buffers, shift registers

Embedded Memory: SRAM blocks

- Xilinx Virtex 5 provides totally 11.6 Mbits of dual-port RAM at 550 MHz
- Other FPGAs: Mostly 1 to 4 Kbits
- Aspect ratio can be programmed
  - Xilinx Virtex-4 4-Kb RAM (*depth* x *width* in bits)
    - 4096x1, 2048x2, 1024x4, 512x8, 256x16
  - Xilinx Virtex-5 36-Kb RAM
    - 32Kx1, 16Kx2, 8Kx4, 4Kx9, 2Kx18, 1Kx36
- Can function as CAMs, FIFOs, SRL, …

CAM = Content Addressable Memories

SRL: Shift Register Logic

(2007 Copyright @Pao-Ann Hsiung)

Embedded Memory

- Largest memory is around 2 MB
- Large number of ports
  - Large aggregate memory bandwidth
  - Stratix II (EP2S180)
    - 30 Gb/s
    - 3414 ports of 1707 RAMs (assuming all dual-ported and operating at maximum frequency)

Embedded DSP Blocks

- Optimized blocks for DSP applications
- Applications
  - WiMAX, 3GPP, WCDMA, CDMA2000, VoIP, H.264 video compression, HDTV
- Functions
  - FIR, complex FIR, IIR, FFT, DCT

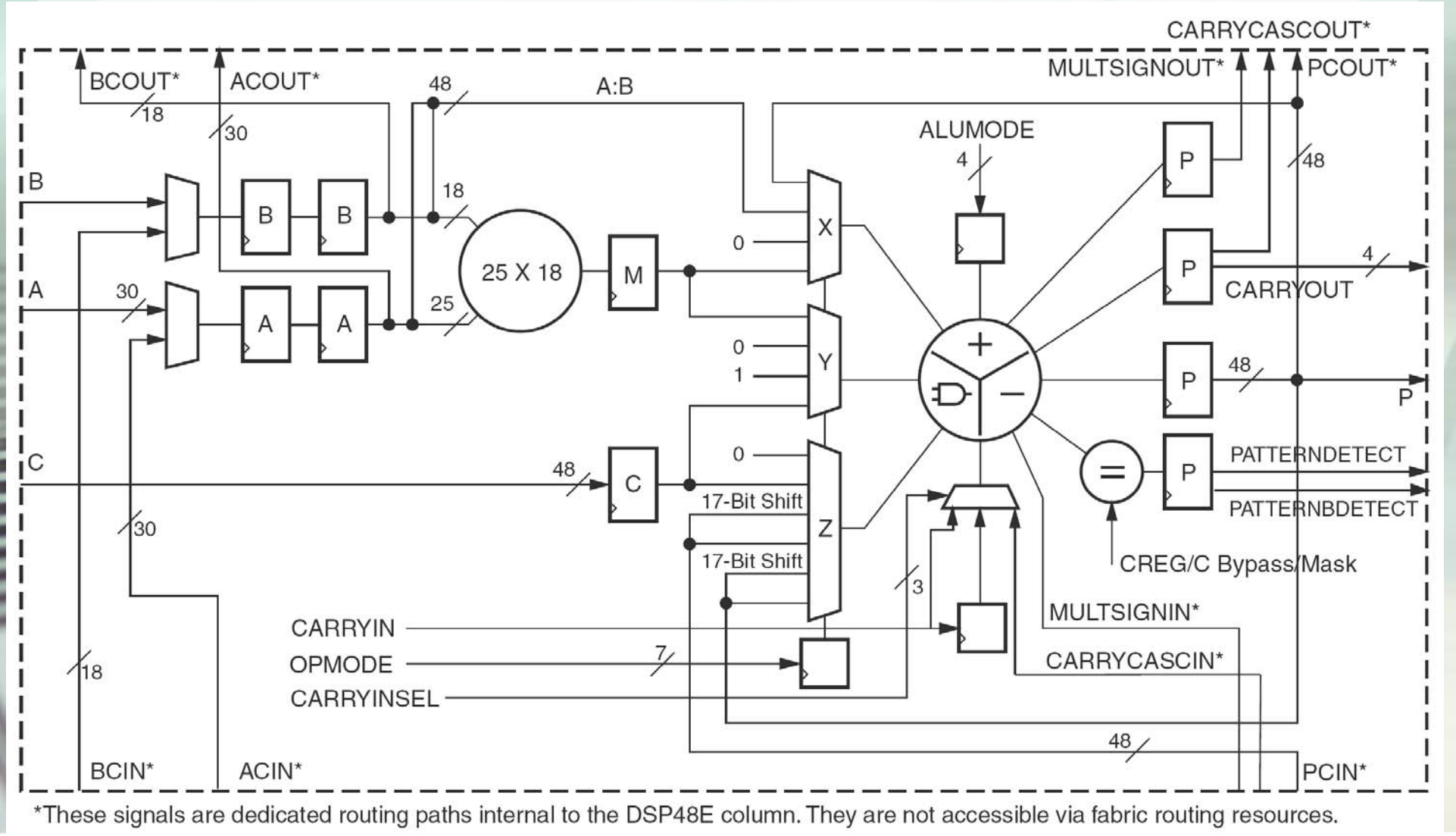
Embedded DSP Blocks

- Logics
  - Adders/subtractors
  - Multipliers, Complex Multipliers
  - MAC (Multiply Accumulate)
  - Arithmetic rounding and saturation units
  - Barrel shifter
  - Loopback capability
  - Pipeline
  - Cascading

Embedded DSP Blocks

- Xilinx Virtex-5 FPGA
  - XtremeDSP DSP48E Slice
    - 25x18 two's complement multiplier
    - 48-bit MAC, logic unit
    - Integrated adder
    - Pipeline, full cascading
    - Adder/subtractor/logic unit
      - With dual 24-bit SIMD mode or quad 12-bit SIMD mode
    - Pattern detector
      - For rounding, underflow, overflow

Xilinx DSP48E Slice Architecture



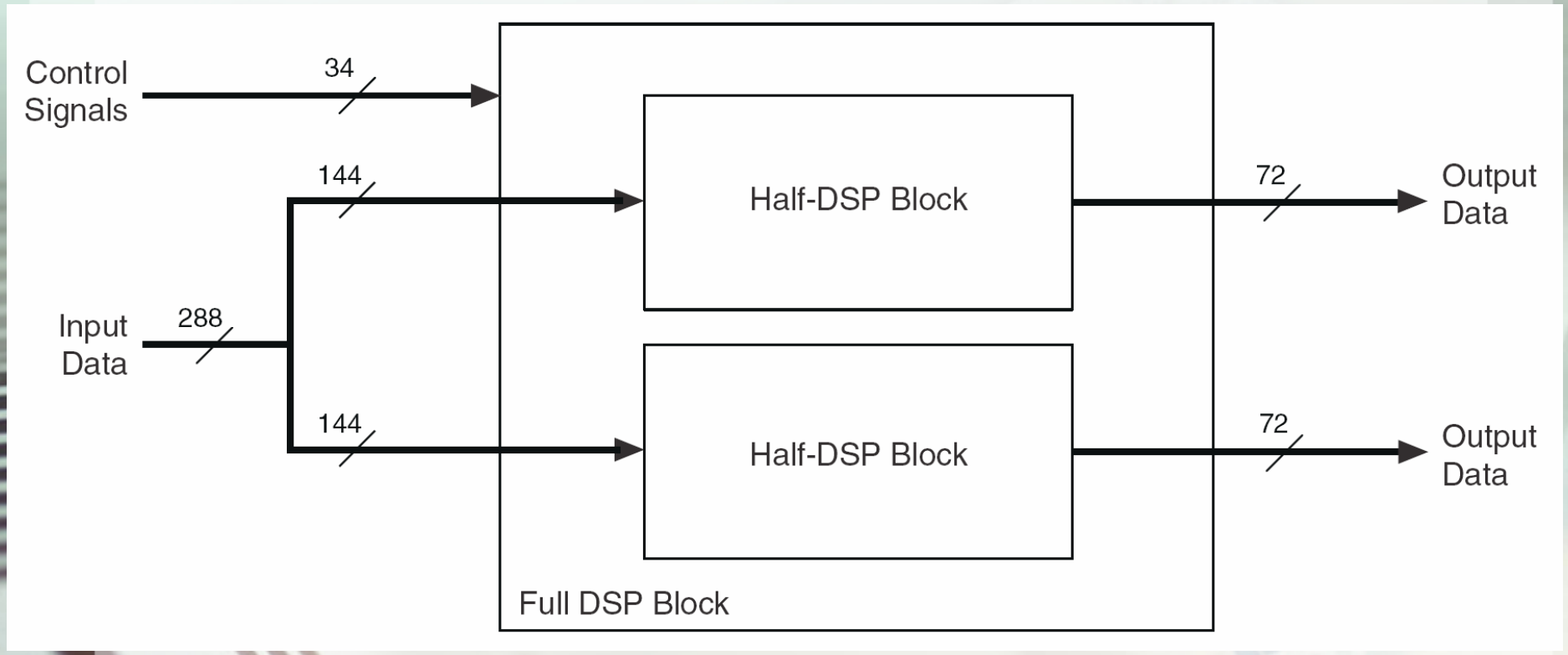
*These signals are dedicated routing paths internal to the DSP48E column. They are not accessible via fabric routing resources.

Embedded DSP Blocks

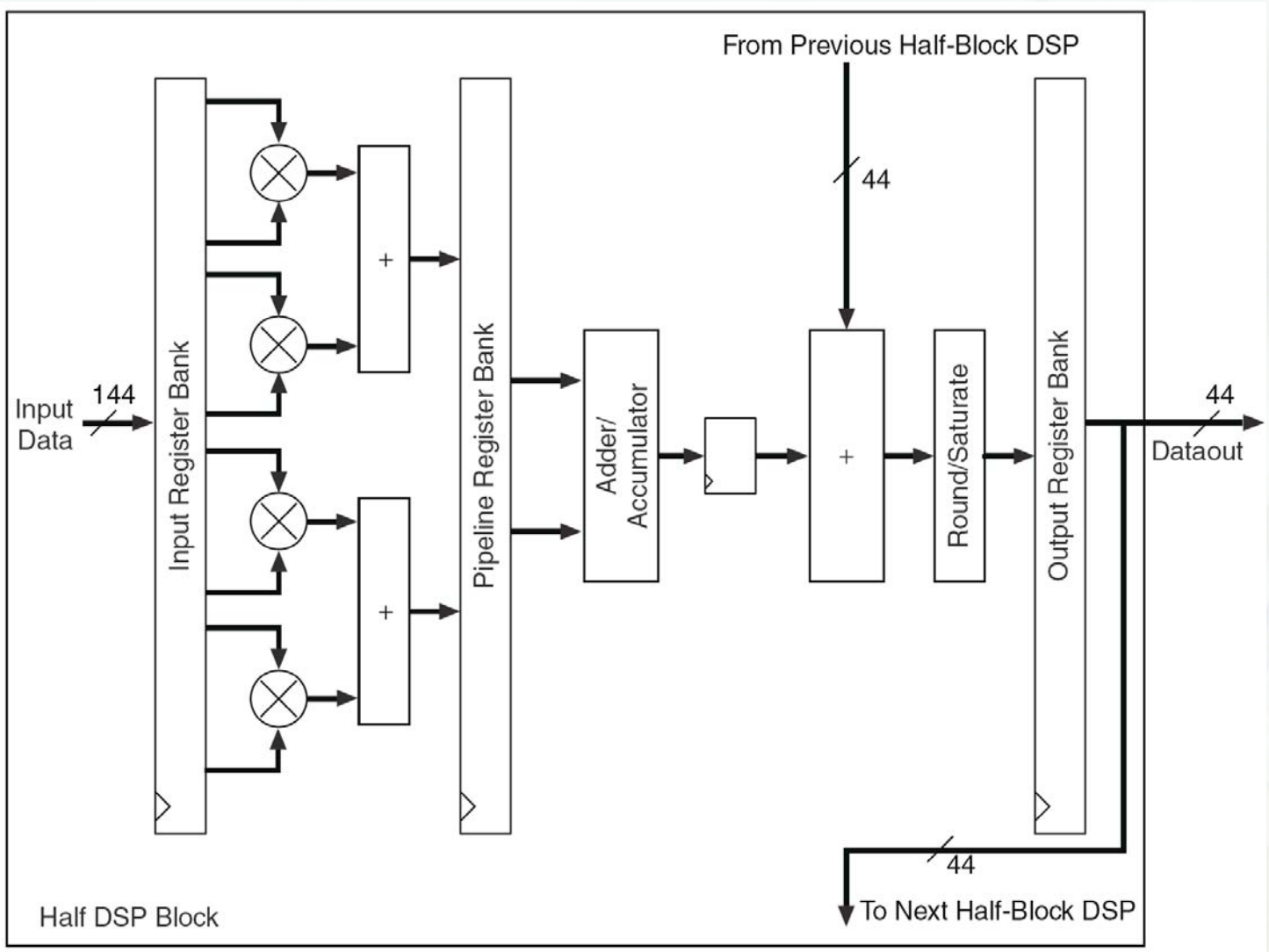
- Altera Stratix III FPGA
  - DSP Block
    - 9-bit, 12-bit, 18-bit, 36-bit multipliers
    - 18-bit complex multiplier
    - Adder, subtractor, MAC
    - Arithmetic rounding and saturation units
    - Barrel shifter
    - Loopback capability for adaptive filtering
  - EP3SE110: 112 DSP blocks, 896 18x18 multipliers

Altera DSP Block Architecture

- 1 DSP block = 2 identical half-DSP blocks



Altera Half-DSP Block Architecture



High-speed Serial I/O

- For high-throughput telecommunications
- Multi-gigabit serial transceivers (MGSTs)
  - Full-duplex serialization and deserialization (SERDES) functions
  - Encoding/decoding functions (8B/10B)
  - Error control logic
  - Support for various I/O standards
- Ethernet MACs for serial I/O solutions

What is 8B/10B decoding?
http://www.xilinx.com/ipcenter/catalog/logicore/docs/decode_8b10b.pdf

(2007 Copyright @Pao-Ann Hsiung)

RocketIO GTP Transceivers

- In Xilinx-5 FPGA chips
- 8-24 channels
- 100 Mb/s to 3.2 Gb/s
- Embedded support:
  - Out of band signaling: Serial ATA
  - PCI Express

High Speed Differential I/O with DPA

- In Altera Stratix III FPGA chips
- Up to 1.25 Gb/s
- Embedded support
  - Data realignment
  - Dynamic phase aligner (DPA)
  - Synchronizer (FIFO buffer)
  - Phase-Locked Loops (PLLs)

Embedded Microprocessors

- For low-bandwidth, control-intensive functions
  - Implementing TCP/IP stacks
- Altera Excalibur
  - First to integrate microprocessor with FPGA
  - ARM 32-bit RISC processor core
  - APEX-20KE FPGA
  - Communication: AHB or dual-port SRAM

Embedded Microprocessors

- Altera
  - Processor Stripe
  - Dedicated hard AHB buses
- Xilinx
  - Embedded Processor Island
  - No dedicated hard buses

Outline

- Introduction
- Fine-grained Architectures
- Coarse-grained Architectures
- Configuration Architectures
- References

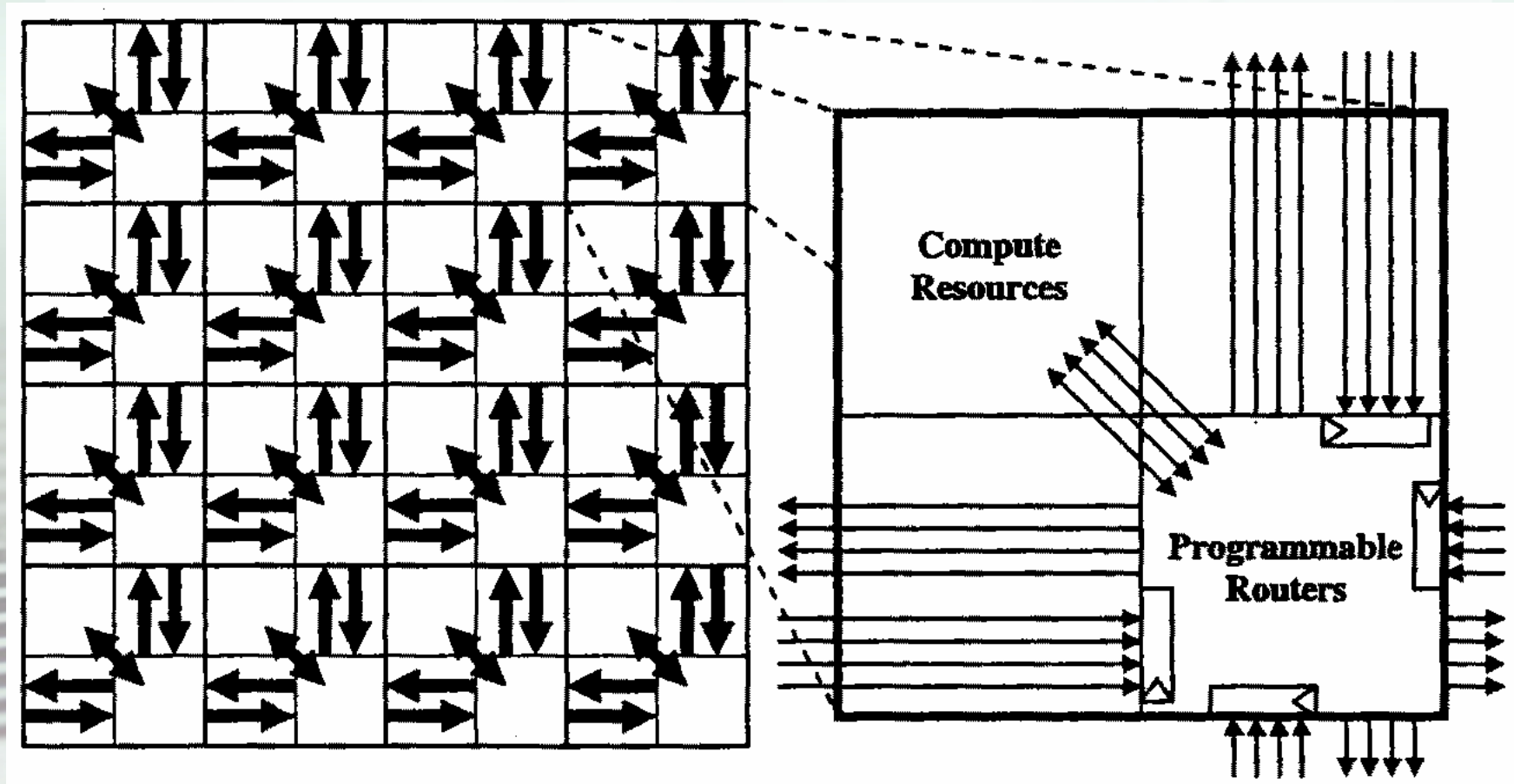
Coarse-Grained Architectures

- Coarse-Grained Reconfigurable Arrays (CGRA)
  - Reconfiguration of functional blocks
  - Blocks optimized for large computations
  - Lesser flexibility, lower power, faster configuration
  - Examples
    - KressArray, RAW, PipeRench, RaPiD, PACT XPP, MathStar FPOA, ACM, DAPDNA-2, MRC6011, PC102, DFA1000, rDSP MATRIX, NAPA, PADDI

RAW

- 2D array of 32-bit MIPS
- Instruction and data caches
- 32-bit pipelined FPU
- Routers and wiring channels for 2D mesh
  - Switched network between processors
  - Two mesh networks statically scheduled
    - For high performance, low latency, low overhead
  - Two mesh networks dynamically scheduled using wormhole routing
    - For data bursts such as cache miss, occasional I/O

RAW



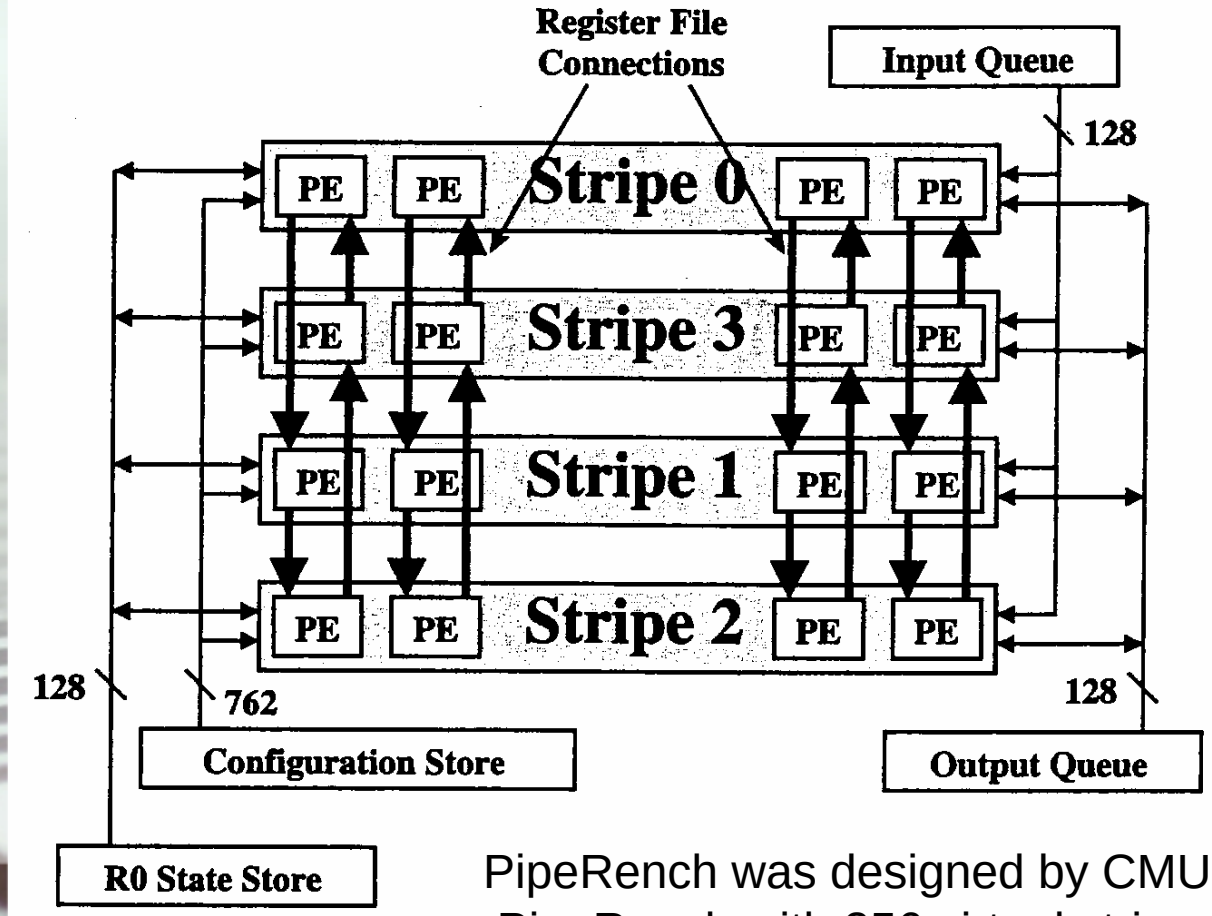
RAW was designed by MIT in 1997
and a 4x4 RAW was fabricated in 2002

(2007 Copyright @Pao-Ann Hsiung)

PipeRench

- For run-time reconfiguration of hardware
- Pipeline stages called “stripes”
  - Each stripe consists of 16 8-bit interconnected PEs
  - Stripes interleaved into a ring structure
- Each PE consists of 8 3-LUTs, one for each bit of the operand width, and some carry logic
- Configuration bits
  - PE: 42 bits
  - Stripe: 42x16 = 672 bits
- A virtual stripe
  - can be swapped into any physical stripe by loading register R0,
  - can be swapped out by storing register R0

PipeRench



PipeRench was designed by CMU in 2000 and a 16 stripe PipeRench with 256 virtual stripes was fabricated in 2002

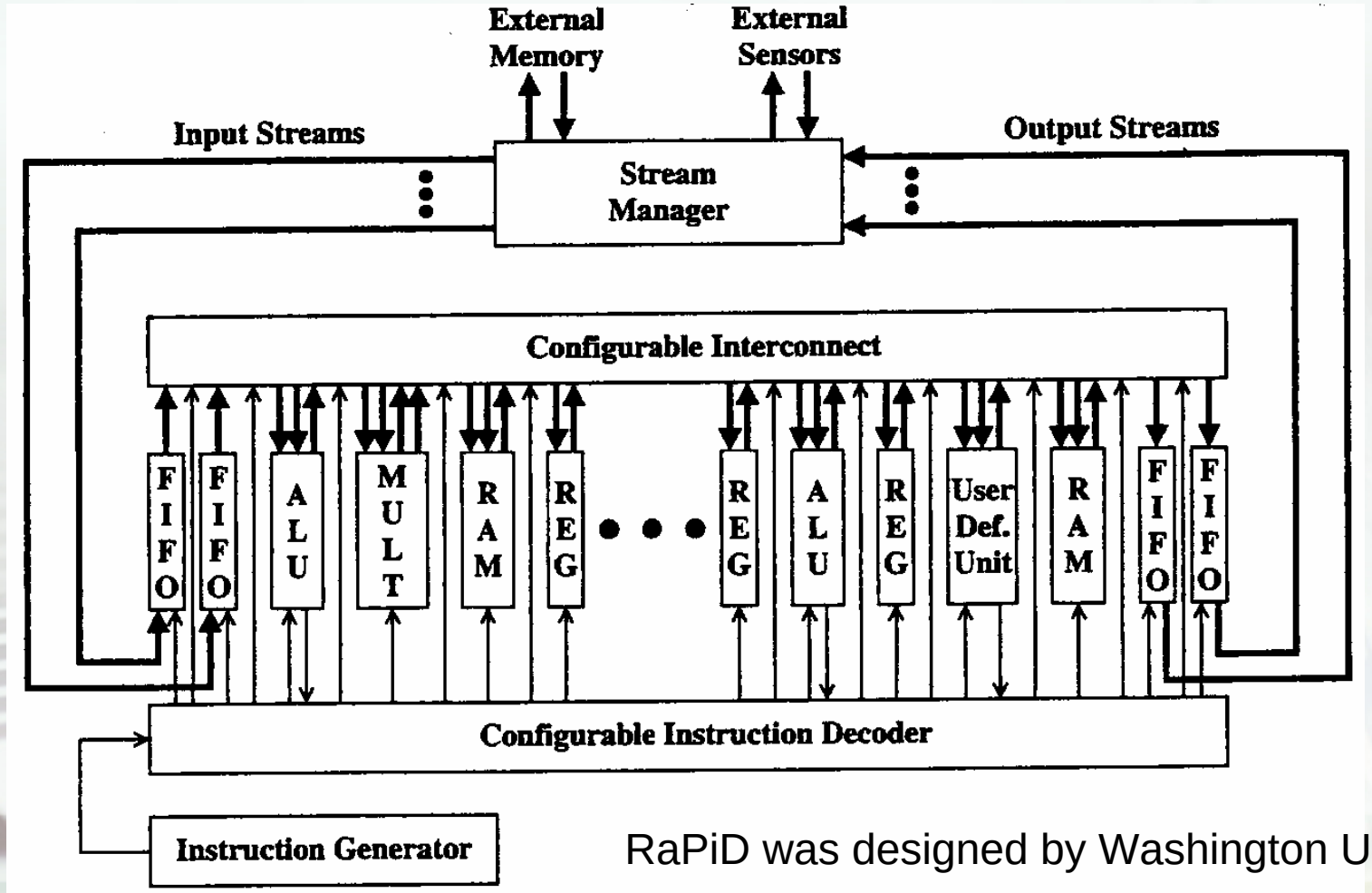
RaPiD

- Two goals
  - Domain-specific high-performance CGRA
  - High-level system design

RaPiD

- RaPiD architecture consists of:
  - A mix of coarse-grained function units
    - with common data width,
    - with a configurable routing channel, and
    - data is streamed through them
  - A Streams Manager
    - provides the input data from external sources
    - helps output data to external devices
  - Instruction Generator and Configurable Instruction Decoder
    - dynamically configures function units and routing interconnects
    - cycle by cycle scheduling (sequencing)
  - Synchronization mechanism
    - Halted upon read of empty input FIFO or write of full output FIFO

RaPiD



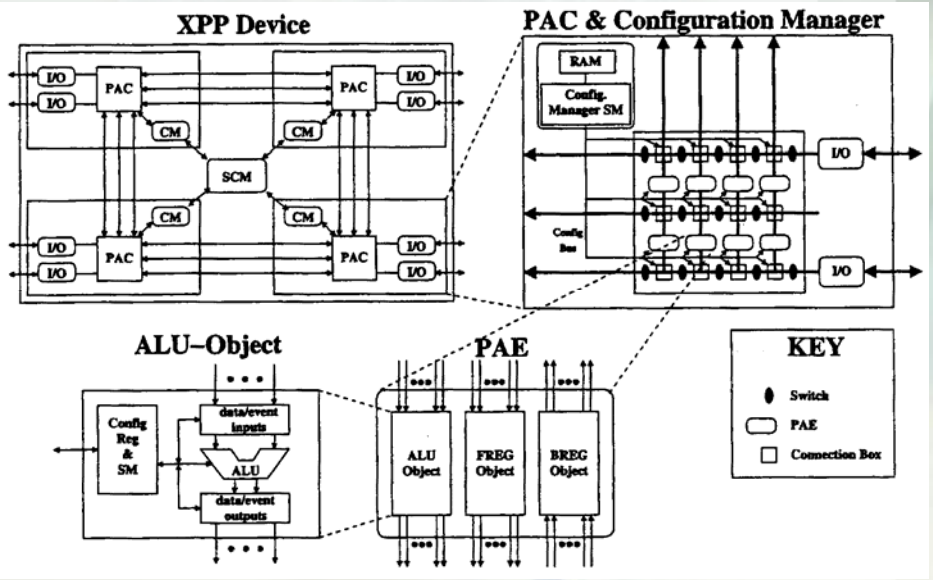
RaPiD was designed by Washington Univ. in 1996

PACT XPP

- XPP: eXtreme Processing Platform
- Commercial CGRA by PACT Informationstechnologie GmbH
- Data streaming for signal and media processing
- Hierarchical configuration and management

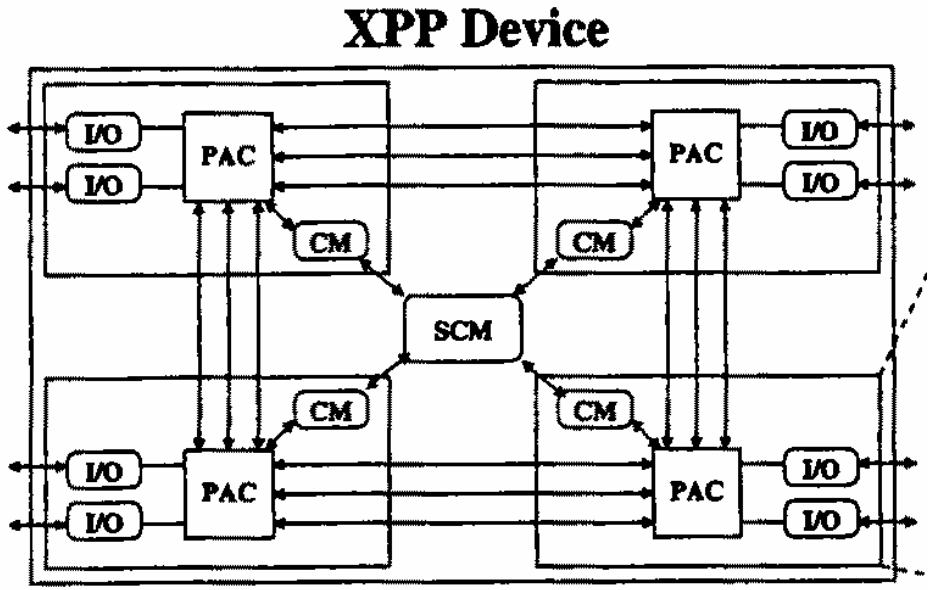
PACT XPP

- Four different levels
  - System level
  - PAC and CM level
  - PAE level
  - Function object level



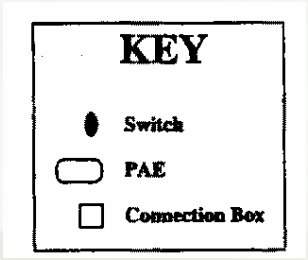
PACT XPP: System Level

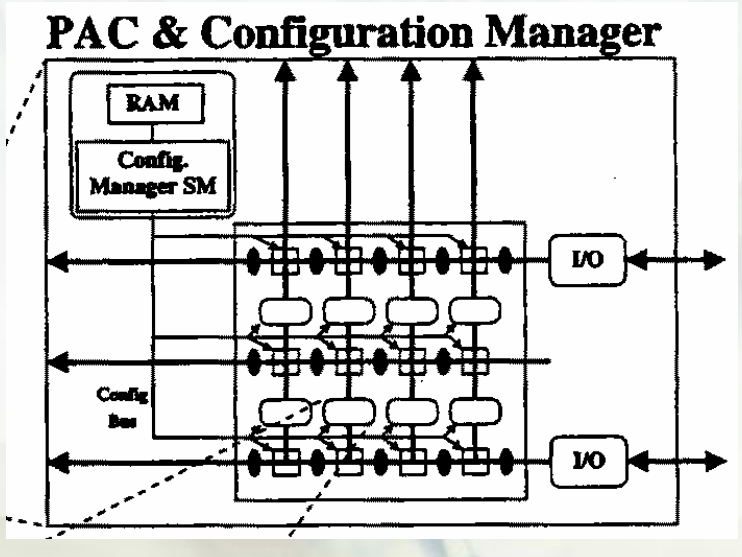
- Processing array clusters (PACs)
- Configuration managers (CMs)
  - State-machine controller and
  - Local RAM
- Supervising CM (SCM)
  - Controls overall configuration of XPP
  - Can be controlled by other external SCMs



PACT XPP: PAC and CM Level

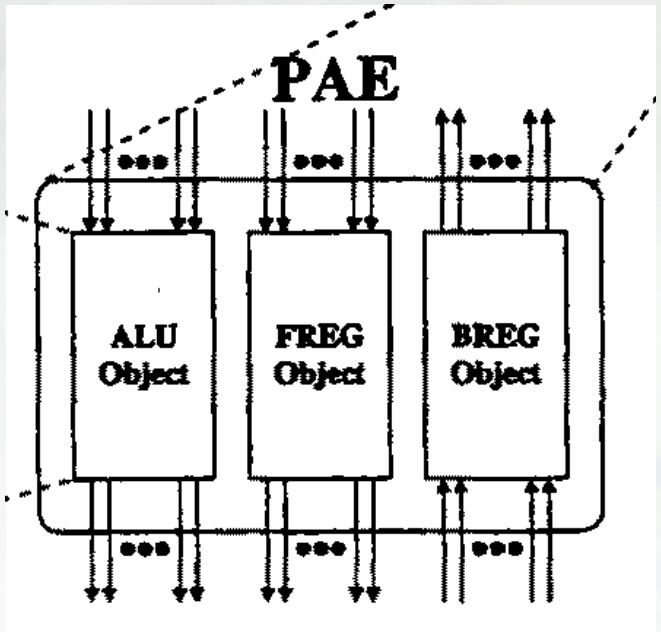
- Array of processing array elements (PAEs)
- Connection boxes for routing between vertical and horizontal buses
- Switches for segmenting horizontal buses
- I/O resources





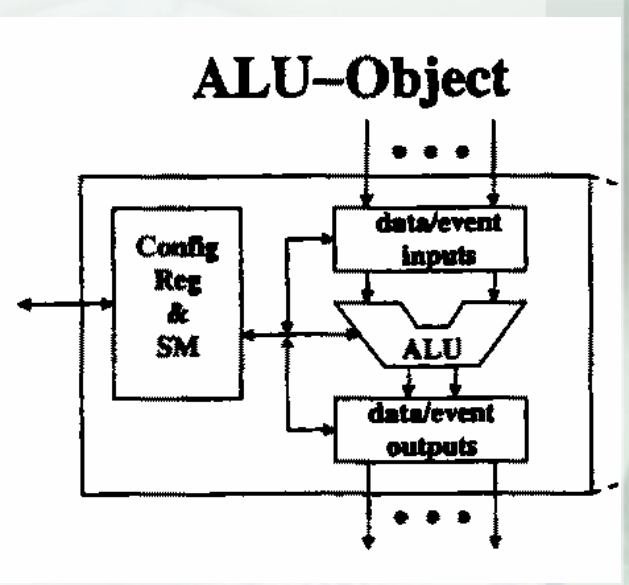
PACT XPP: PAE Level

- Function objects such as ALU, RAM, …
- Forward register (FREG)
  - Vertical routing support
  - Data control flow
  - Counters
- Backward register (BREG)
  - Vertical routing support
  - Adders/subtracters
  - Barrel shifters



PACT XPP: Function Level

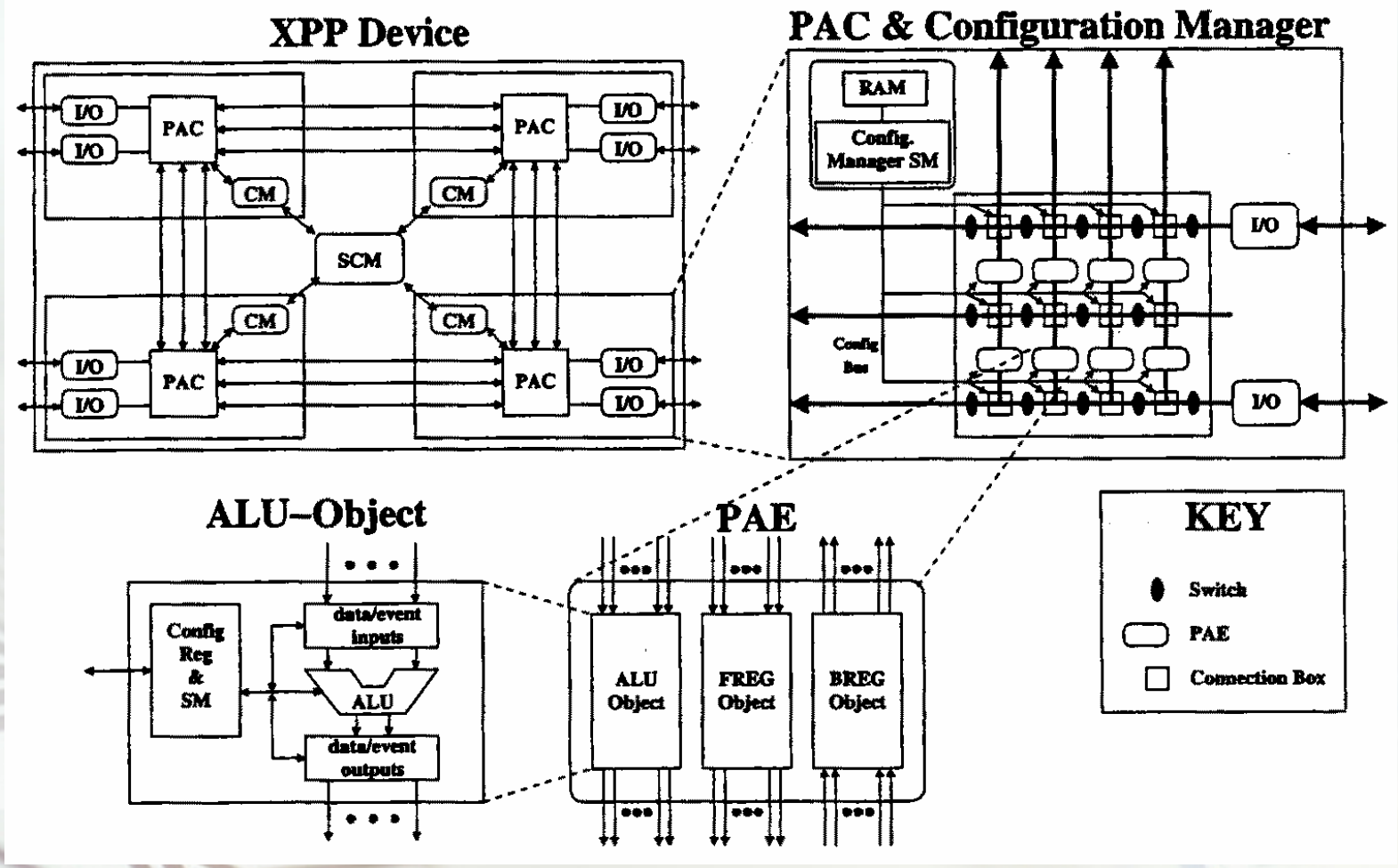
- Consume and produce data
- Two kinds of packets
  - Data packets (24 or 32 bits)
    - Computation results
  - Event packets (few bits)
    - Condition or state bits
- Event-driven reconfiguration
  - Data and event packets affect configurations



PACT XPP

- Advantages of hierarchical configuration
  - Configuration scaling of large systems
  - Independent configurations in different chip regions
  - Self-configuration can be local or global

PACT XPP


XPP Device
I/O
PAC
CM
SCM
PAC & Configuration Manager
RAM
Config. Manager SM
Config Bus
ALU–Object
Config Reg & SM
data/event inputs
ALU
data/event outputs
PAE
ALU Object
FREG Object
BREG Object
KEY
Switch
PAE
Connection Box

MathStar FPOA

- FPOA: Field-Programmable Object Array
- Commercial CGRA by MathStar
- Domain-specific CGRA
  - Fabrication in less than a month
  - 1 GHz internal operation speeds
- Silicon Objects
  - Function units
  - Can be placed in any position of the 2D array

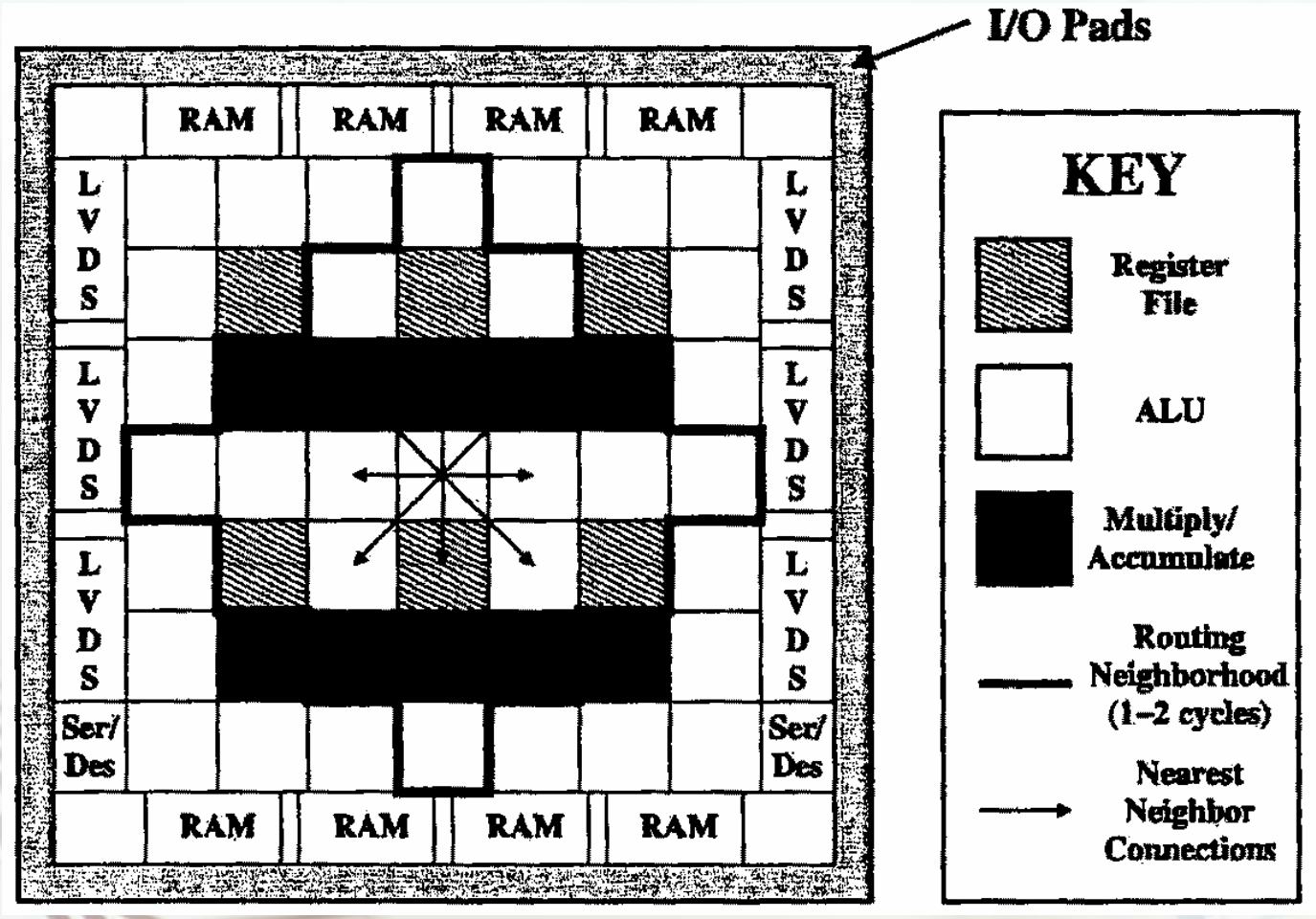
MathStar FPOA

- Silicon objects consists of a mix of
  - Register files
  - ALUs
  - MACs
  - Logic block of 4 4-LUTs
  - CRC generator
  - CAMs
  - External memory interfaces

MathStar FPOA

- Routing in FPOA
  - Each silicon object connect directly to its 8 immediate neighbors
    - Four unique values
    - Single clock cycle communication
  - One level of pipelining
    - 24 other cells within its extended neighorhood
  - More levels of pipelining
    - Rest of FPOA

MathStar FPOA



Outline

- Introduction
- Fine-grained Architectures
- Coarse-grained Architectures
- Configuration Architectures
- References

Configuration Architectures

- Configuration method
- Configuration bandwidth
- Configuration granularity
- Configuration time
- Configuration access

Configuration Method

- Requires
  - an external controller or microprocessor, or
  - a configuration device, or
  - a download cable
- Generic configuration methods
  - Serial
    - Joint Test Action Group (JTAG) Boundary-Scan mode
    - PS and AS in Stratix III, SPI and Serial in Virtex-5
  - Parallel
    - FPP in Stratix III, SelectMAP and BPI in Virtex-5

Configuration Method

- Configuration methods for Stratix III FPGA:
  - Fast Passive Parallel (FPP): byte-wide
  - Fast Active Serial (AS): supports remote upgrade
  - Passive Serial (PS)
  - Joint Test Action Group (JTAG): no decompression/security
- Enhanced features
  - Decompression of configuration data in reconfigurable device
  - Remote secure and reliable system upgrade

Configuration Method

- Configuration methods for Virtex-5 FPGA
  - Slave/Master serial mode
  - Slave/Master SelectMAP mode
  - JTAG Boundary-Scan mode
  - Serial Peripheral Interface (SPI) mode (std Flash)
  - Byte-wide Peripheral Interface (BPI-up/BPI-down) modes (x8 or x16 NOR Flash)
- Configuration options
  - 256-bit AES bitstream decryption, multi-bitstream management, bus width auto-detection, CRC, ECC

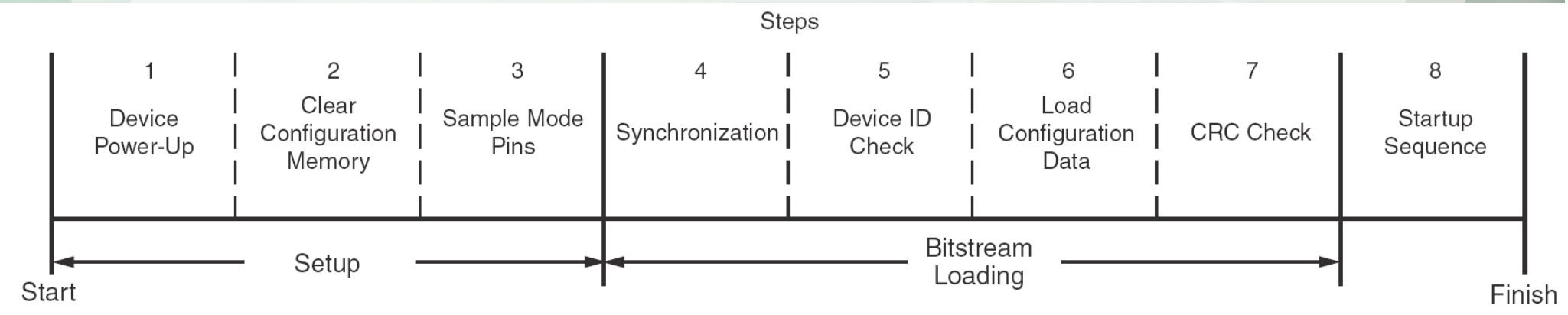
Master: CCLK from internal oscillator

Slave: CCLK from external clock source

Configuration Modes in Virtex-5

Configuration Mode	M[2:0]	Bus Width	CCLK Direction
Master Serial(2)	000	1	Output
Master SPI(2)	001	1	Output
Master BPI-Up(2)	010	8, 16	Output
Master BPI-Down(2)	011	8, 16	Output
Master SelectMAP(2)	100	8, 16	Output
JTAG	101	1	Input (TCK)
Slave SelectMAP	110	8, 16, 32	Input
Slave Serial	111	1	Input

Virtex-5 Configuration Process



Configuration Bandwidth

- The rate at which configuration data can be loaded into configuration memory in the FPGA chips
- Rate = ConfigClock x PortWidth
- Xilinx Virtex-5 in slave SelectMAP mode
  - Rate = 60 MHz x 32 bits = 1,920 Mb/s
- Altera Stratix III in FPP mode
  - Rate = 100 MHz x 8 bits = 800 Mb/s

Configuration Granularity

- Amount of resources configured by the smallest amount of configuration data
  - Xilinx Virtex-5: 1312 bits (41 32-bit words)
- The more addressability that is required in data, the more logic that is needed to provide the addressability

Configuration Granularity

- Full configuration
  - Full chip is configured at a time
    - Raw bitstream sizes could be very large
      - Xilinx Virtex-5 XC5VLX330T: 82 Mbits = 10 MB
      - Altera Stratix III EP3SL340: 120 Mbits =15 MB
- Partial configuration
  - Part of a chip is configured at a time
    - Column-based configuration
    - Tile-based configuration

Partial Configuration in Xilinx FPGA

- Column-based configuration in Virtex-2Pro
  - A full column, spanning the width of a chip, must be configured at a time
- Tile-based configuration in Virtex-4/5
  - A chip is divided into several rows and a tile, spanning the width of a row, must be configured at a time
    - A tile is configured using 1312 bits as follows:
    - 1312 = 41 x 32 = (20 x 32) + 32 + (20 x 32)

To configure 20 CLBs above HCLK | To configure HCLK | To configure 20 CLBs below HCLK

Configuration Time

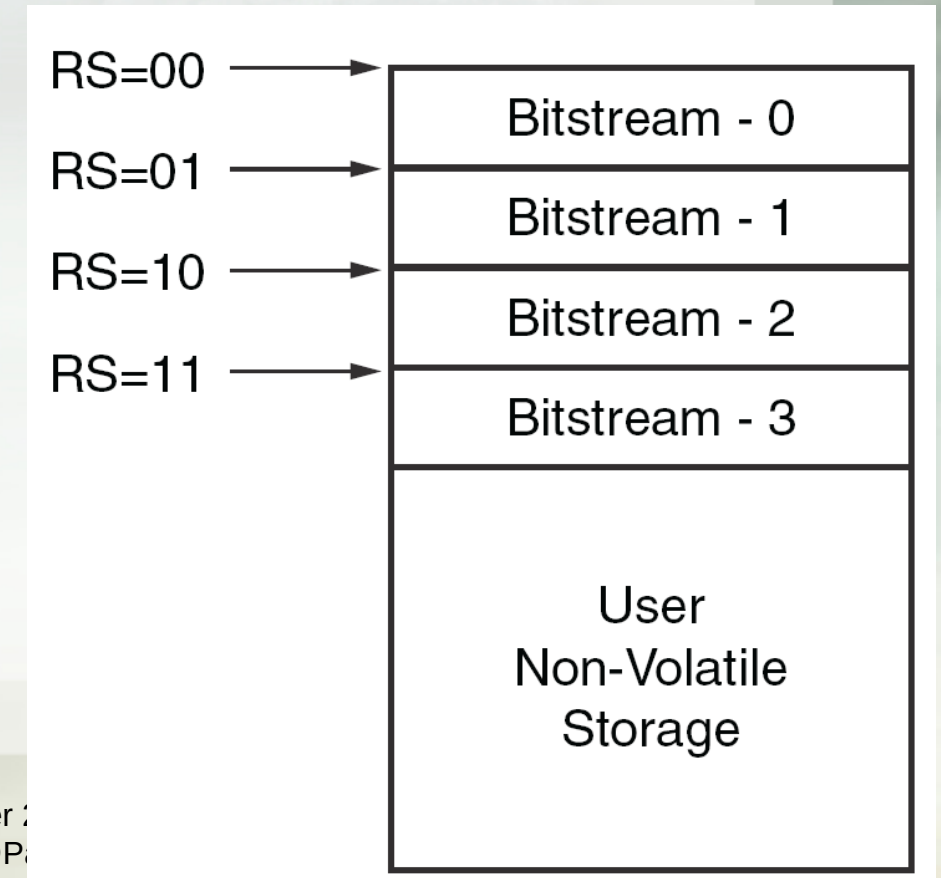
- Static configuration
  - FPGA must be reset and then configured
  - Example: Altera FPGA chips
- Dynamic (run-time) configuration
  - FPGA can be configured without reset
  - Example: Xilinx Virtex FPGA series

Configuration Access

- Ability to read out configuration data
  - Called "readback"
  - Disabled when data is encrypted
- Purposes
  - To ensure the data is correct
    - Example: in space
  - To extract hardware state of execution
  - To communicate between FPGA and host
- Example: Xilinx Virtex FPGA series

Configuration Access

- Fallback Reconfiguration
  - Allows multiple bitstreams to be stored on external memory for
    - Multiple booting
    - Error recovery



Xilinx Virtex-5 Fallback Reconfiguration

(2007 Copyright @P

Outline

- Introduction
- Fine-grained Architectures
- Coarse-grained Architectures
- Configuration Architectures
- References

References

1. G. Estrin, “Organization of Computer Systems – The Fixed Plus Variable Structure Computer,” Proc. Western Joint Computer Conf., New York, 1960, pp. 33-40.
2. Xilinx documents, http://www.xilinx.com
3. Altera documents, http://www.altera.com