
ECE 636

Reconfigurable Computing

Lecture 8

Reconfigurable Systems I



Overview

- **Field programmable interconnect chip**
- **Multi-FPGA system topologies**
- **An application: logic emulation**
- **An example multi-FPGA system**
- **Topology optimization**

Issues

- **Types of multi-FPGA systems.**
- **Multi-FPGA networks.**
- **Multi-FPGA partitioning**
- **Why are we interested in multi-FPGA systems?**
 - Numerous applications which require HUGE amounts of FPGA logic
 - Moore's Law
 - Greed



© Mentor Graphics

Types of systems

- **Can build a specialized multi-FPGA system.**
 - **Wired for one purpose.**
- **Can build reusable multi-FPGA system.**
 - **Emulators, other debugging systems.**
- **Our bridge to computation**
 - **Multi-FPGA systems are parallel computers in the traditional sense**
 - **Granularity of computation and communication are important**

Are Meshes Realistic?

- The number of wires leaving a partition grows with Rent's Rule

$$P = KG^B$$

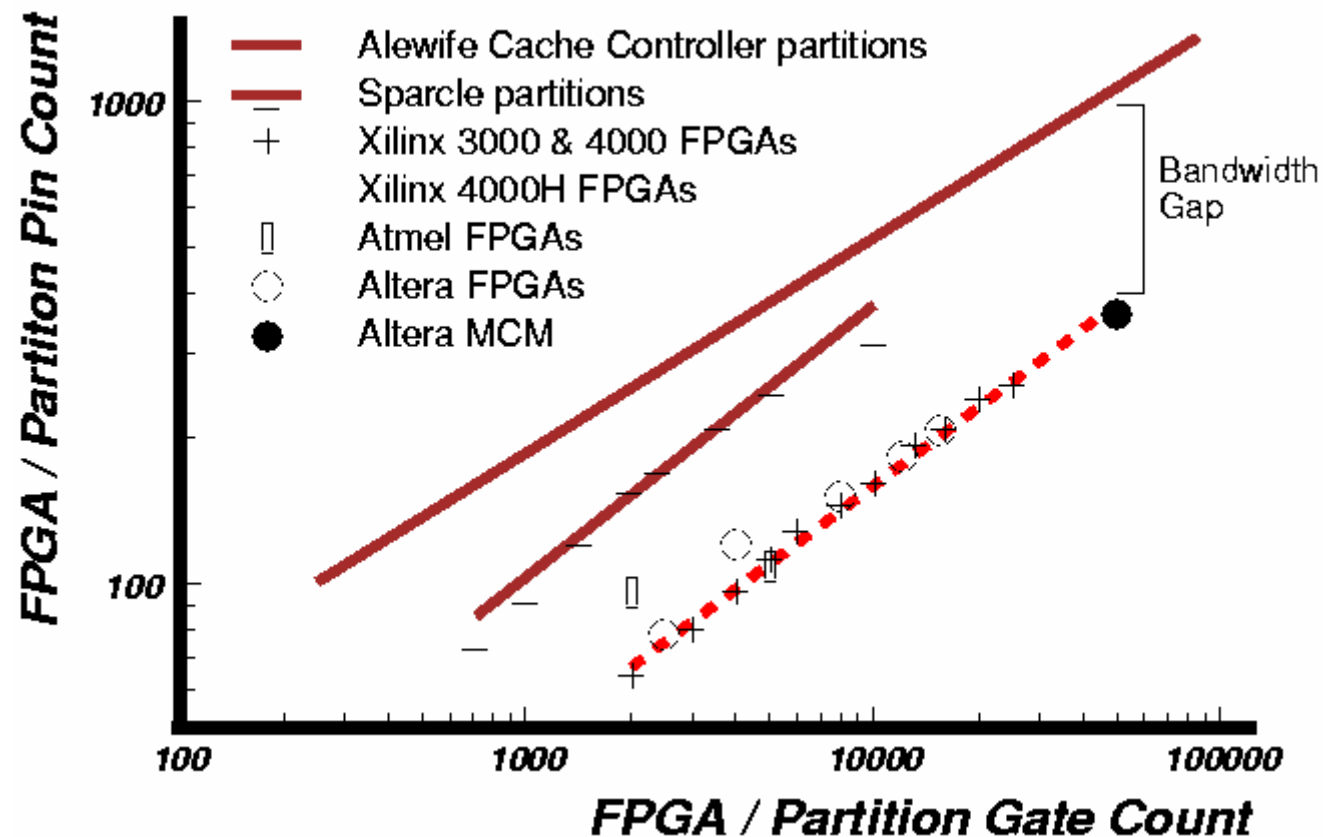
- Perimeter grows as $G^{0.5}$ but unfortunately most circuits grow at G^B where $B > 0.5$
- Effectively devices highly pin limited
- What does this mean for meshes?

Possible Device Scenarios

Not Limited <ul style="list-style-type: none">– unused FPGA pins– unused FPGA gates	Gate Limited <ul style="list-style-type: none">– some unused pins– no unused gates
Pin Limited <ul style="list-style-type: none">– no unused pins– some unused gates	Balanced <ul style="list-style-type: none">– no unused pins– no unused gates

- Rent's Rule indicates that pin limited situation is getting worse.
- Frequently some logic must be left unused leading to limited utilization
- Perhaps this logic can be "reclaimed"

Partition vs FPGA Pin Count

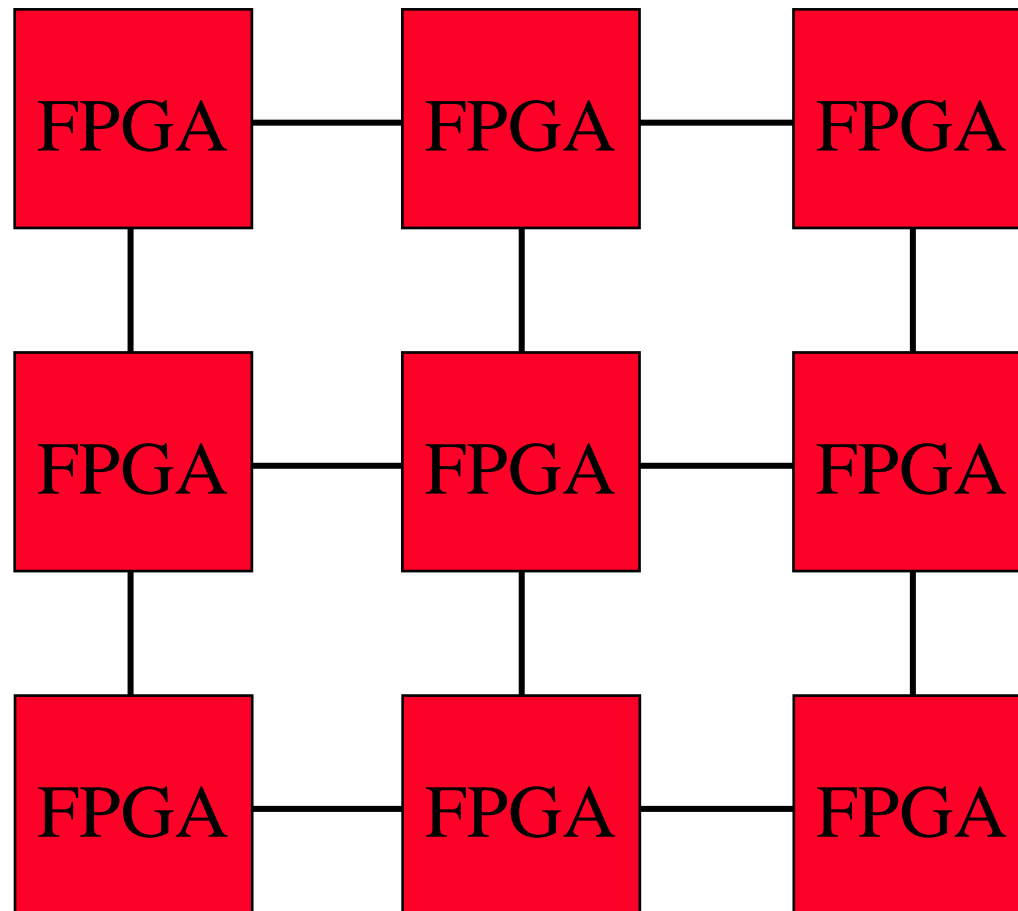


- FPGAs don't have enough pins
- Problem may or may not get worse depending on “structured” design.

Networks

- **Ad hoc.**
 - Best suited for specialized systems.
- **Crossbar.**
 - Fully connected.
- **Specialized crossbars.**
- **Multi-stage.**
 - Not often used in multi-FPGA systems.

Nearest Neighbor Interconnection



Near-neighbor meshes

◦ Advantages:

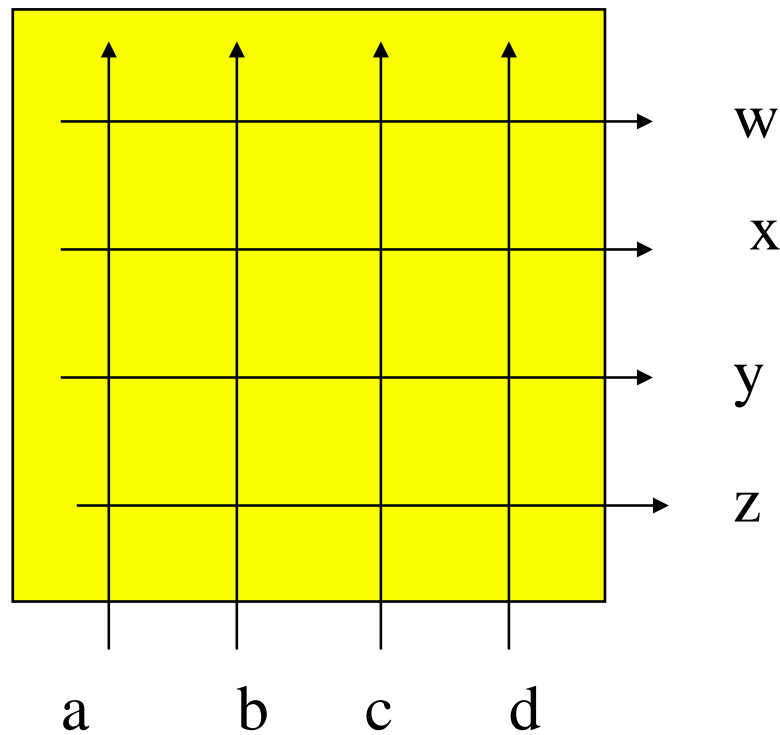
- Uniform: all chips the same.
- Easy to lay out on PCB.

◦ Disadvantages:

- Routing is easily blocked.
- Through pins limit logic utilization of FPGAs.
- Long and unpredictable delays.
- No natural hierarchical extension.

Crossbar

- **Fully connected:**
 - Single source/destination.
 - Multi-point.
- **n^2 area.**

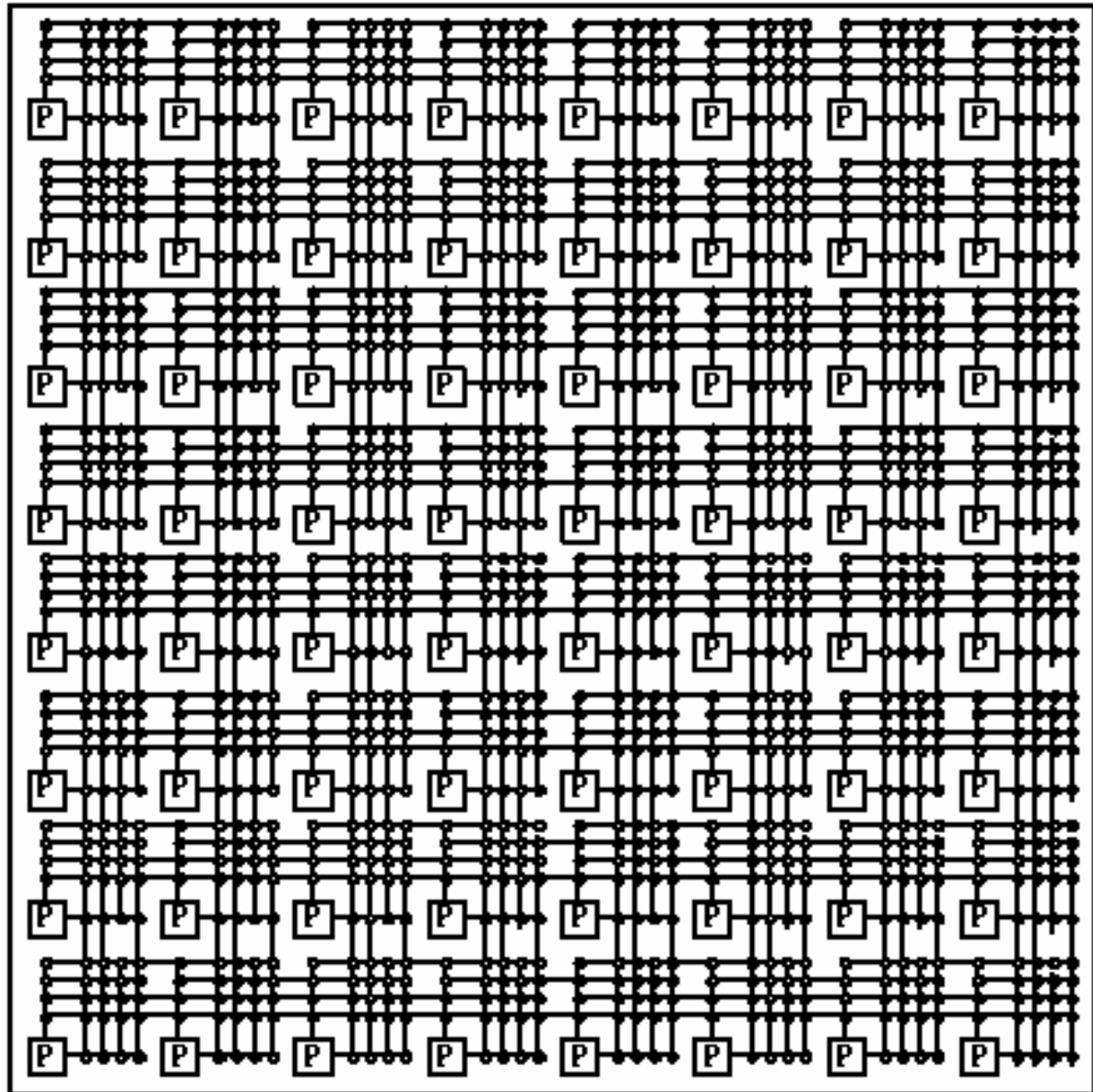


Field-programmable Interconnect Components

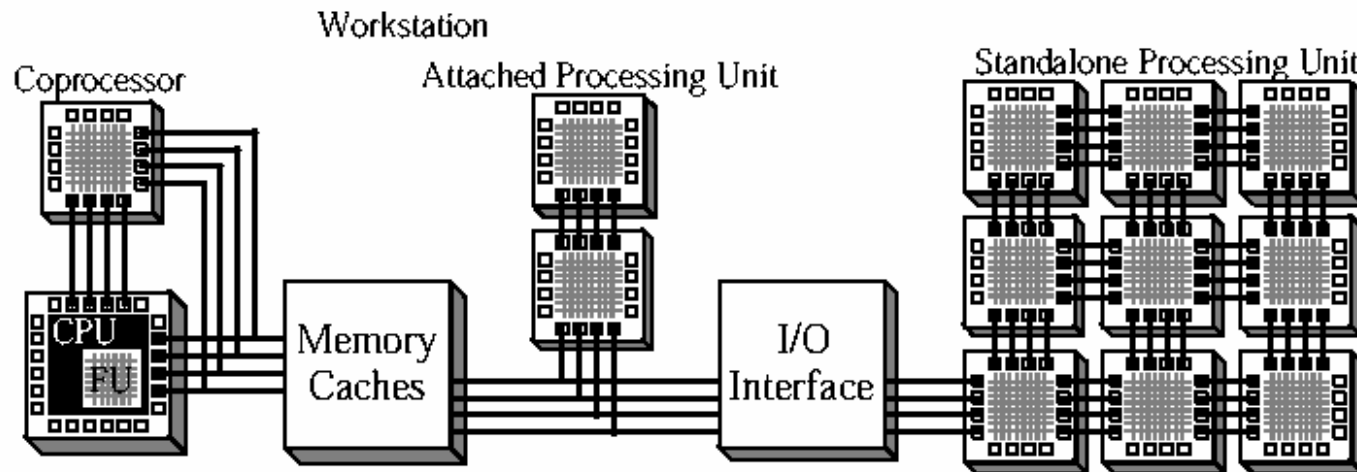
- **Field-programmable parts used for interconnection only.**
- **Effectively FPGA with logic removed.**
 - **Lack of connection blocks leads to fewer transistors, better performance.**
- **Frequently in competition with FPGA devices for interconnect.**
- **Exhibit expanded pin counts.**

FPICs

- High internal connectivity
- Not always cost effective



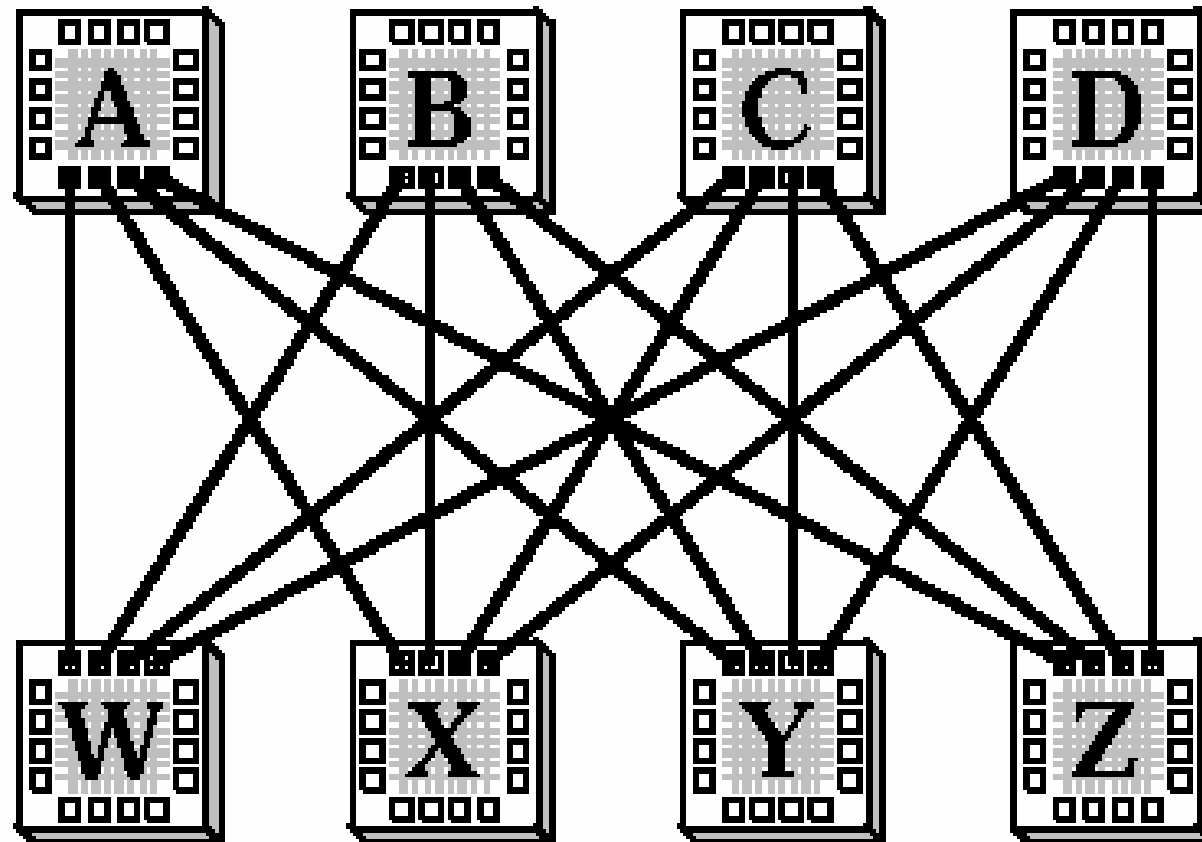
Reconfigurable Processing



From Hauck: Role of FPGAs

- Many places to put reconfigurable computing components
- Most implementations involve multiple discrete devices
- How should these devices be connected together?

Full Crossbar Topology



- Devices A-D are routing only
- Gives predictable performance
- Potential waste of resources for near-neighbor connections

Hierarchical Crossbar

- Full connectivity occurs at top level
- Routing between FPGAs requires determining level at which source and destination share an ancestor.
- Simplifies routing

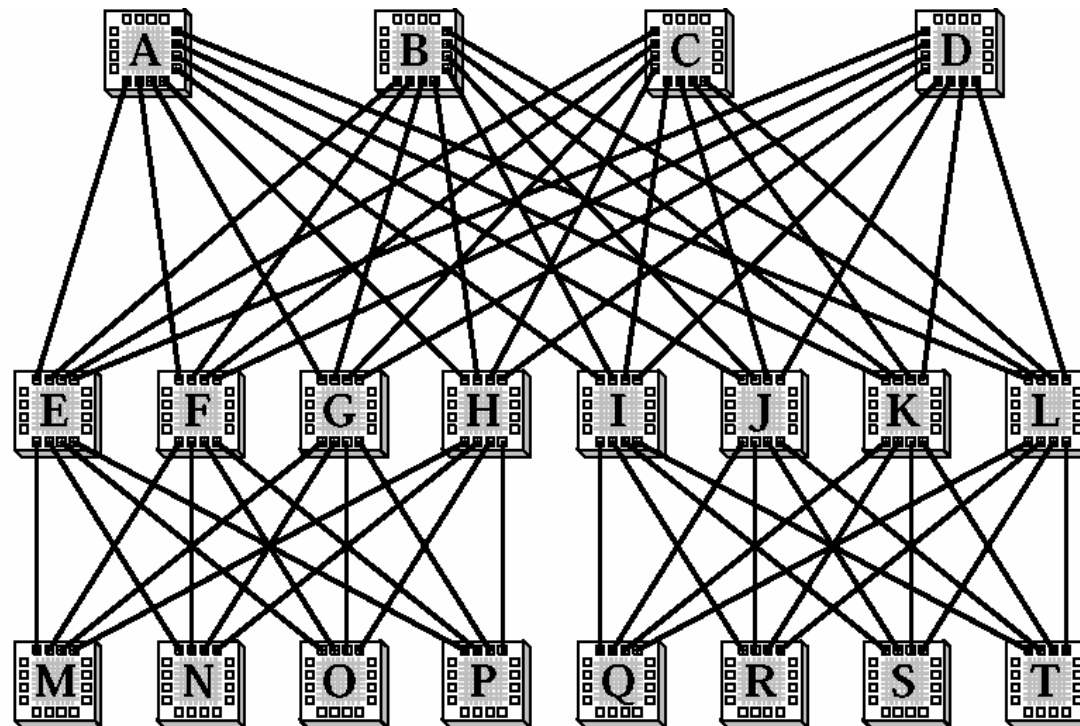
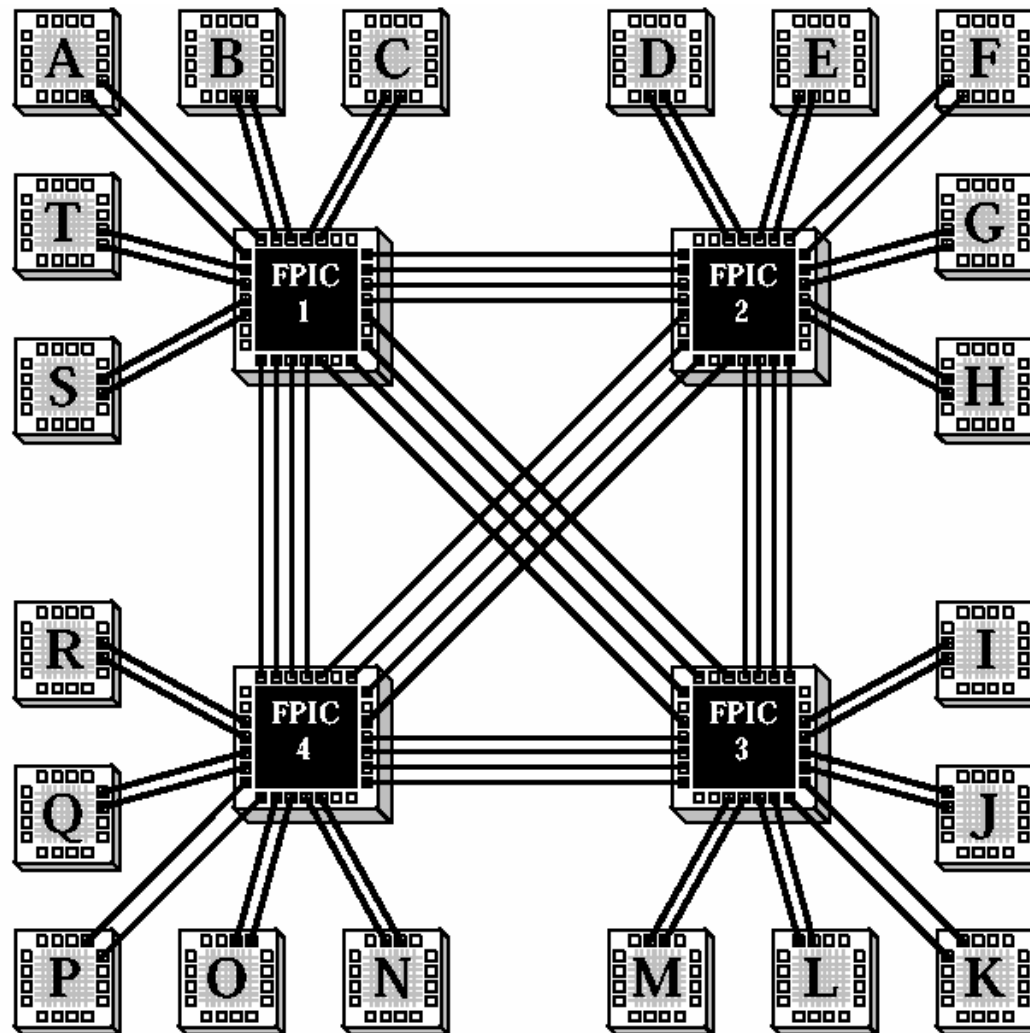


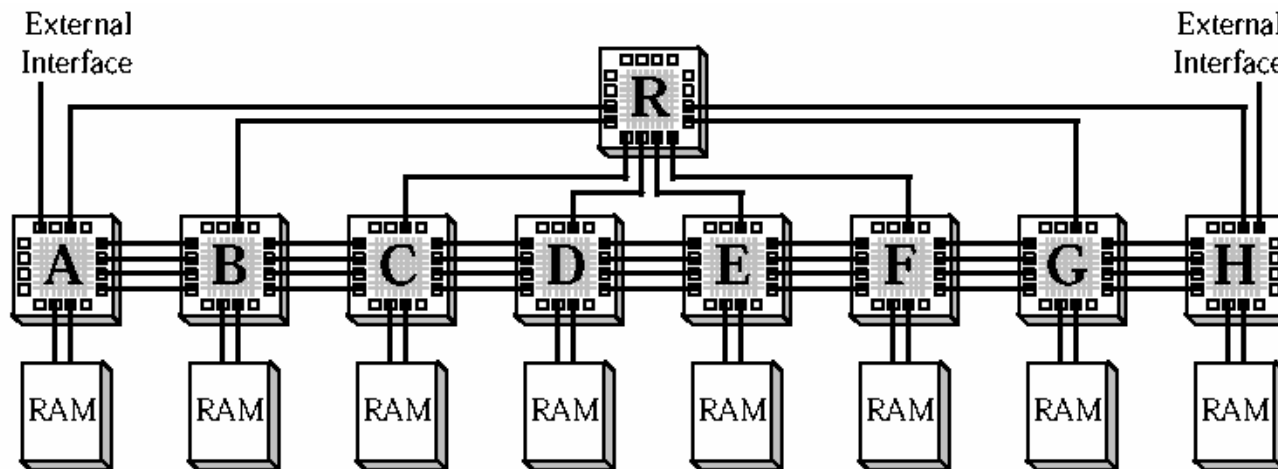
Figure 13. A hierarchy of crossbars. FPGAs M-T hold all the logic in the system. Chips E-H and I-J form two first-level crossbars, and chips A-D form a second-level crossbar.

Two-level Hierarchy

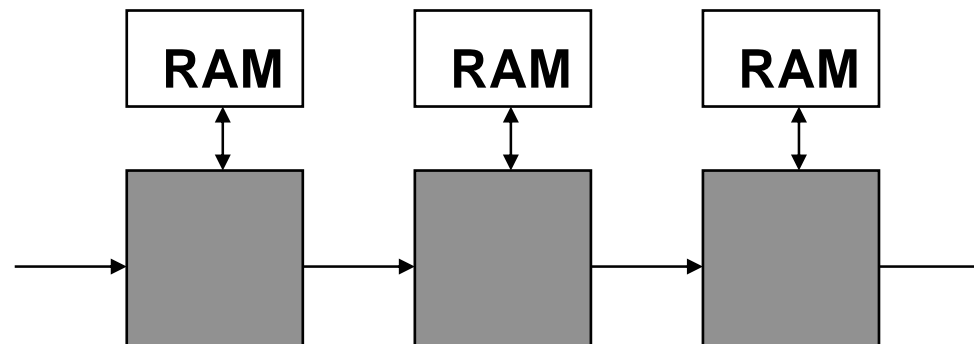
- Inter-FPGA signals travel through at most two FPICs
- Maximum distance in mesh topology is $N^{0.5}$ for N FPGAs



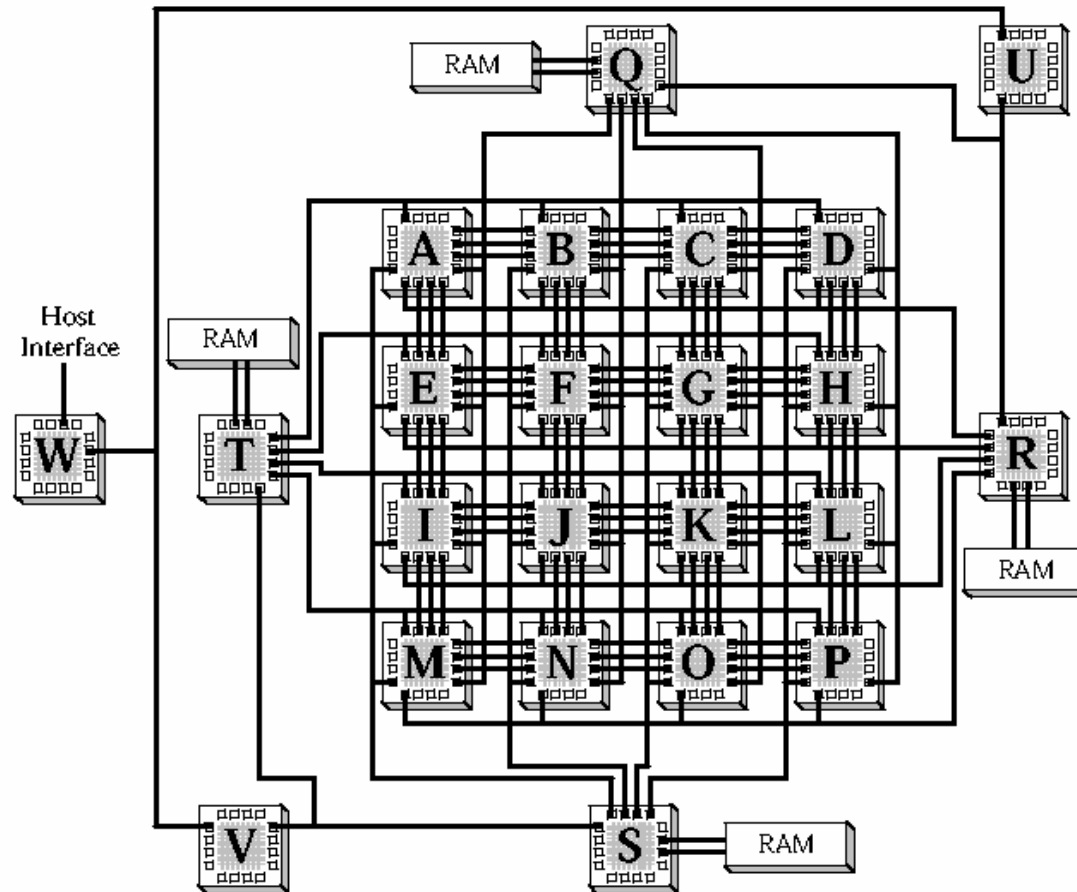
Linear Array



- **Current hardware**
- **Programs implemented as systolic array**
 - **Input key**
 - **Search each RAM bank for sequence**



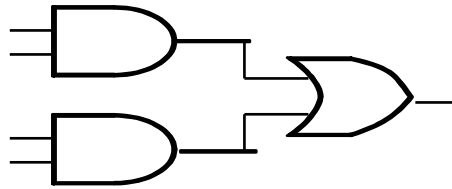
Hybrid Architecture



- Buses connect groups of FPGAs to SRAM
- Extra devices used for RAM controller and map to external interface.

Logic Emulation

- An application of multi-FPGA systems.
- One of several approaches to verify the functionality of a new ASIC
 - **Simulation** – use a microprocessor to verify functionality of a device.



- **Emulation** – physically implement the functionality of the design using FPGAs

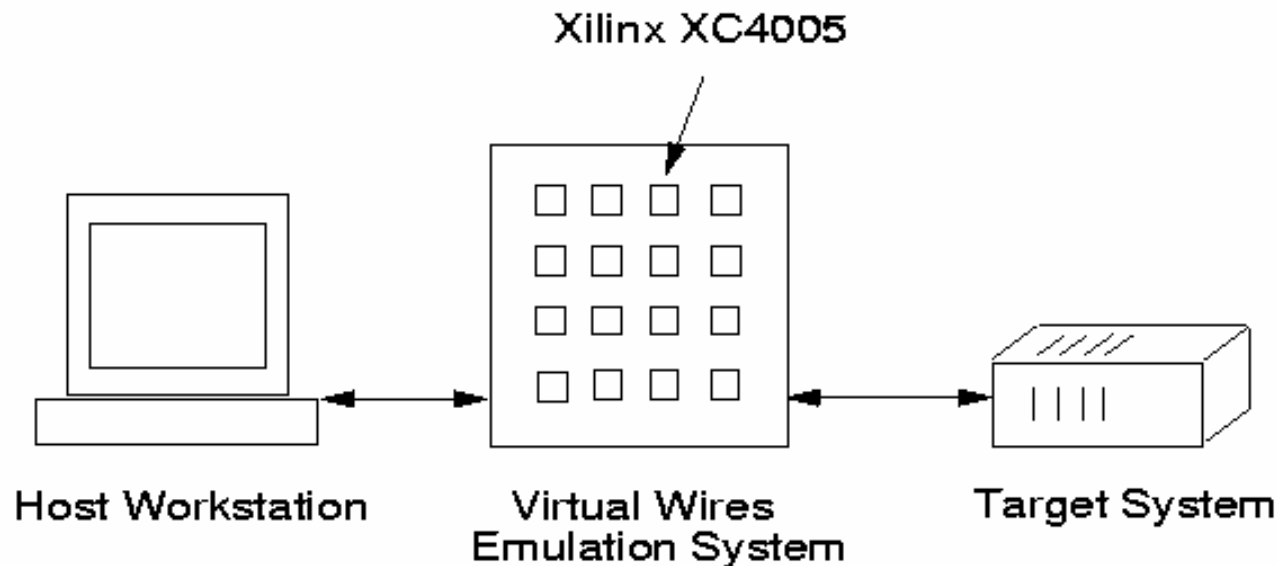


Example System: Virtual Wires

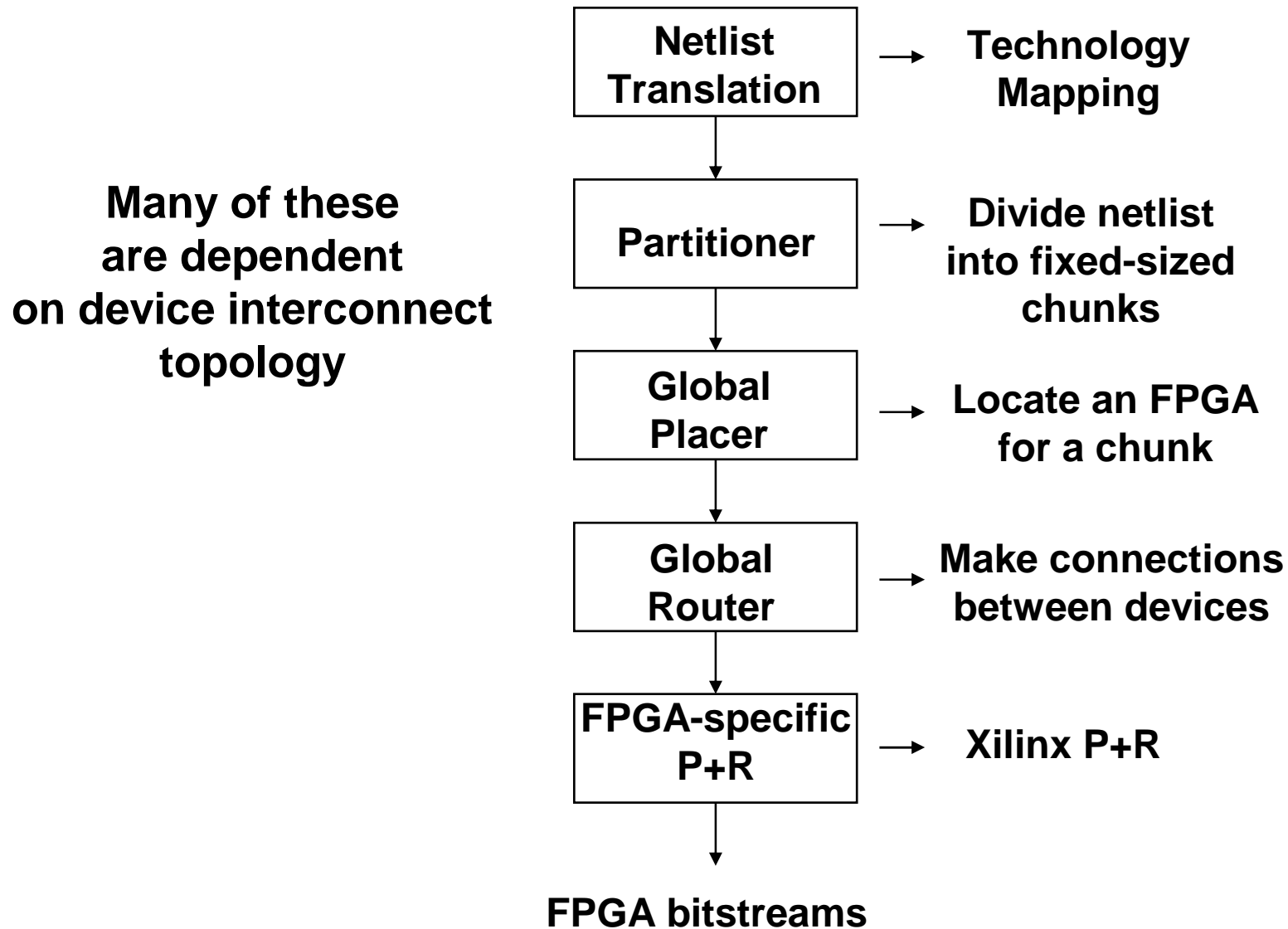
- Goal is to take an ASIC design and map it to multi-FPGA hardware
- Can replace new chip in target system to allow for software development.
- Important issues include
 - How is system interfaced to workstation.
 - What is interface to target system
 - How can memory be emulated
 - Logic analysis / debugging

Emulation System Configuration

- Pod interface to target system
- Serial or Sbus interface to host workstation
- (not shown) Physical connection to logic analyzer also a possibility
- Target system must be slowed down to accommodate emulation

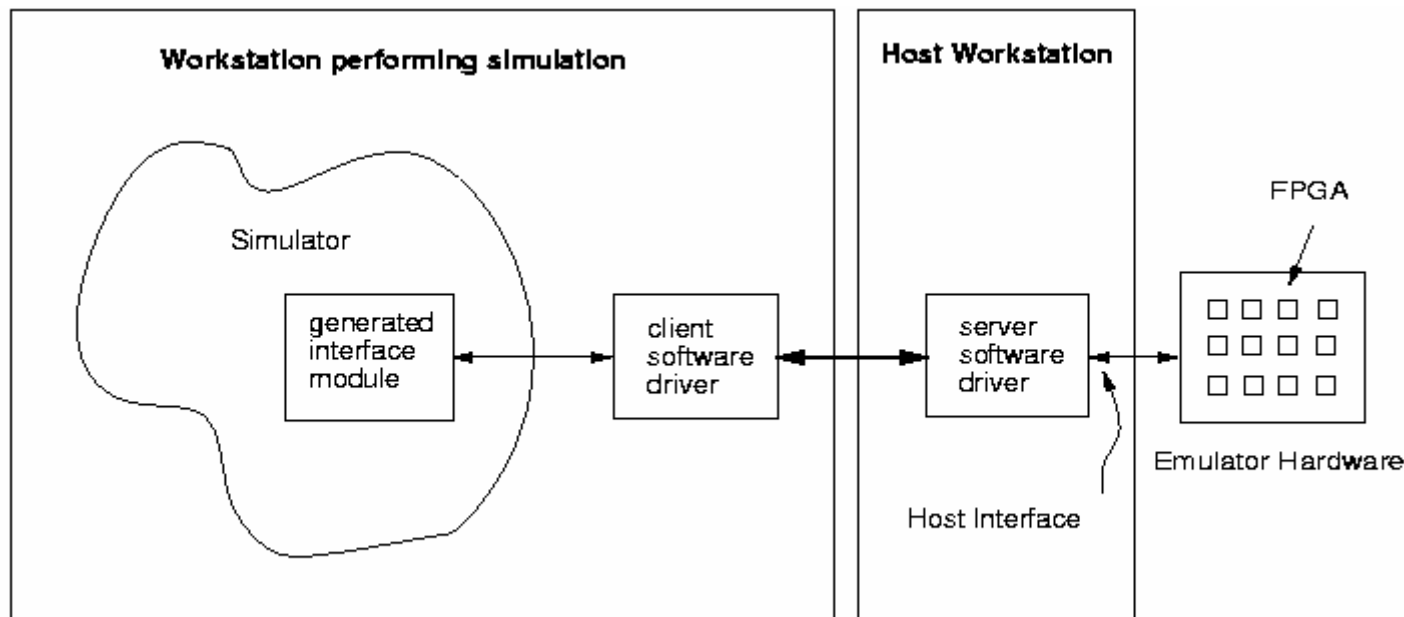


Emulation Software Steps



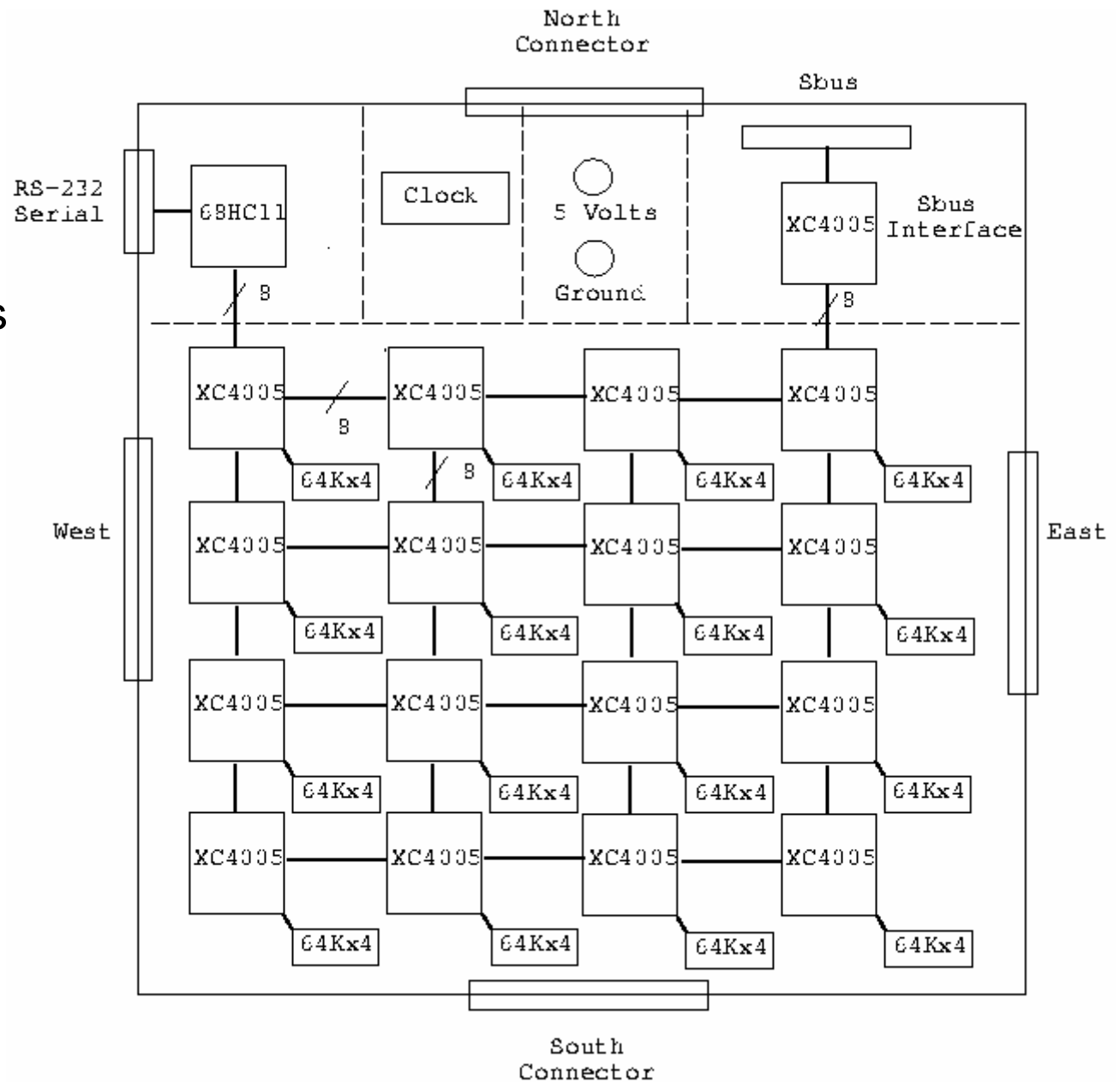
Simulation Acceleration

- **FPGA system takes the place of one portion of simulated design**
- **Inputs transported to FPGA system.**
- **Outputs returned from FPGA system.**



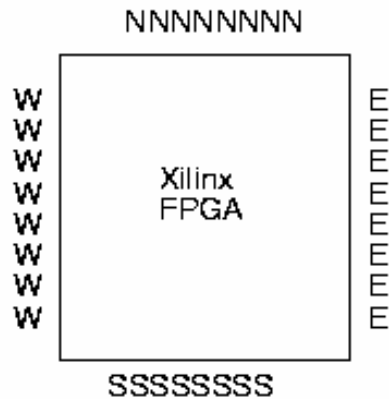
Emulation Board

- Pod connectors located along perimeter
- Two host interfaces
- Near-neighbor communication

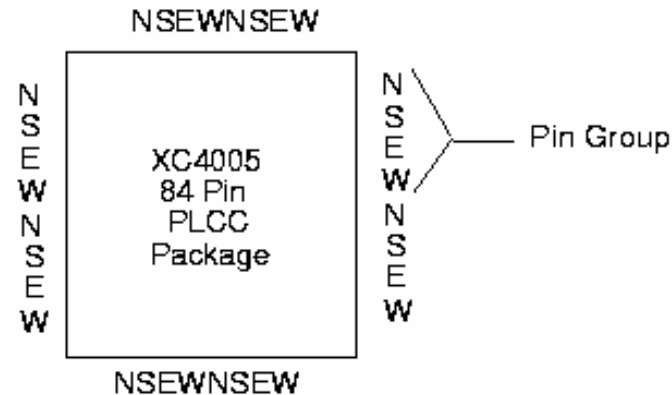


Device Pin Layout

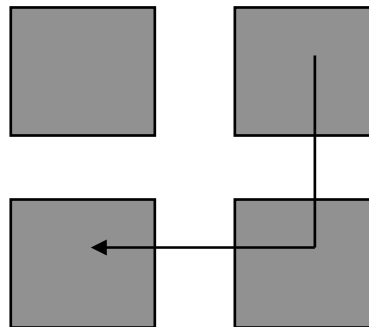
Logical I/O signal distribution



Physical I/O signal distribution



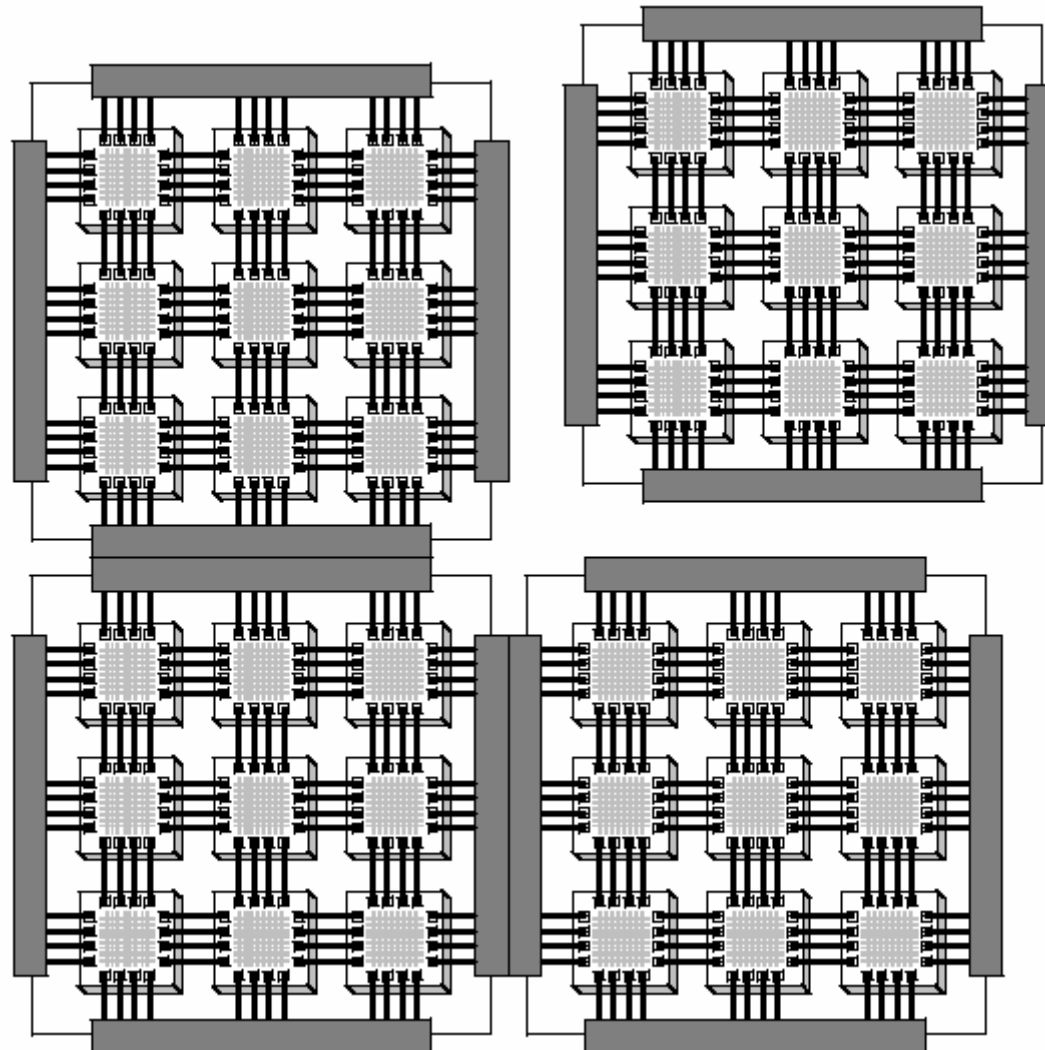
- Many nets may pass through an intermediate FPGA in traversing source to destination.



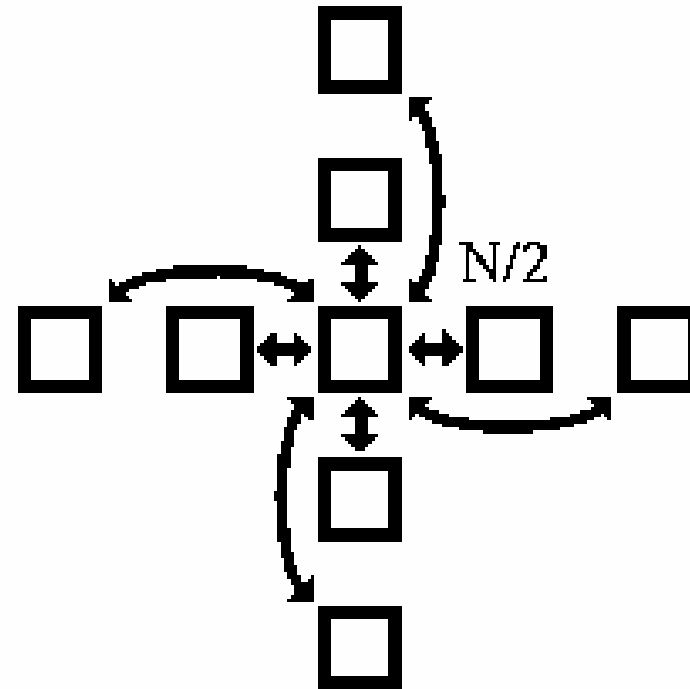
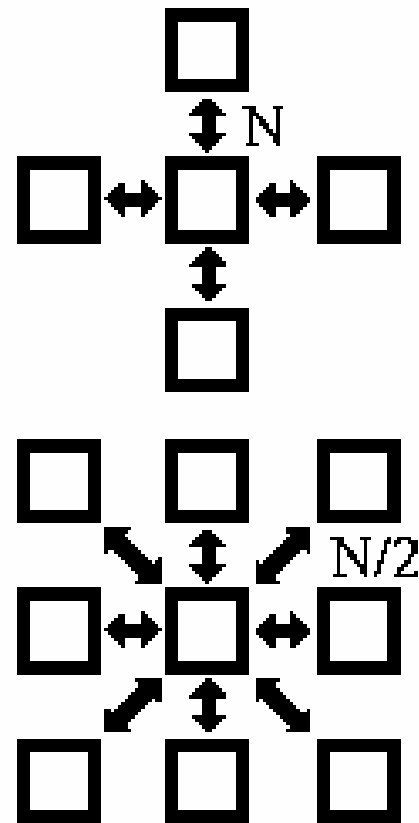
- Physical assignment of IO to pins important to allow device routability at the expense of board routability.

System Scalability

- Attach boards together to form a larger array
- Clock distribution for high-speed signalling an issue.

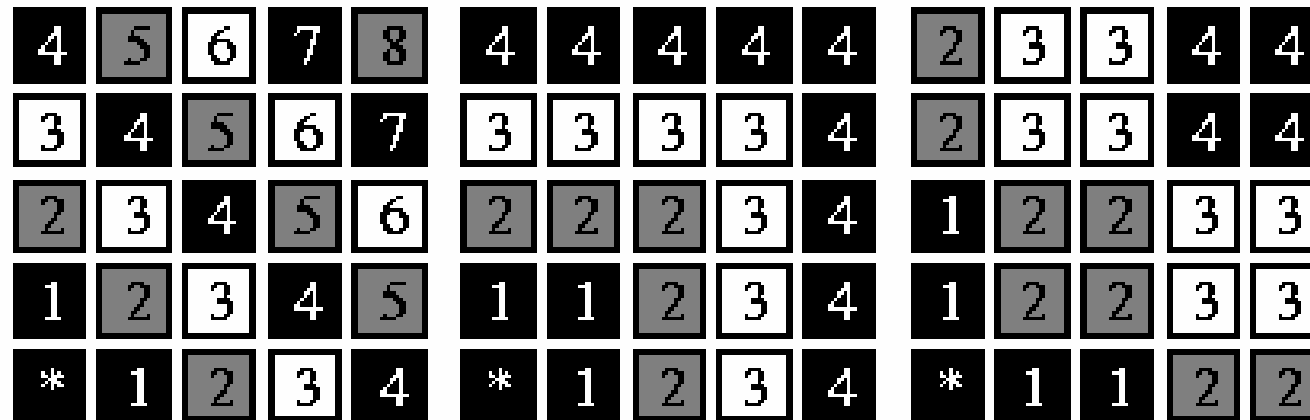


Topology Alternatives



- Near-neighbor
- 8-way interconnect
- One-hop interconnect

Mesh Routing Distances



4-way

8-way

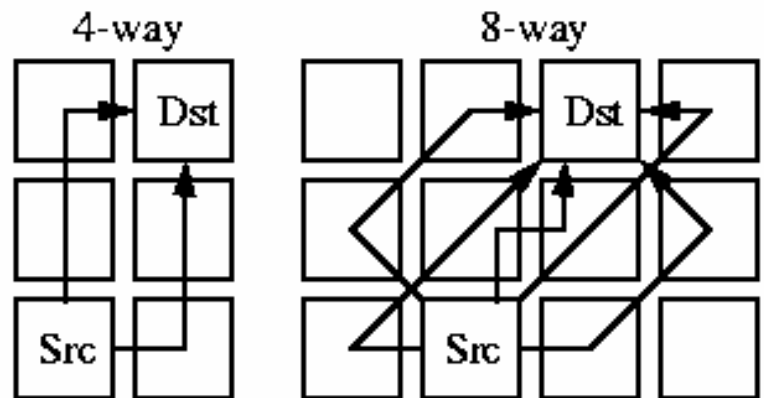
1-hop

- Note that 8-way and 1-hop distances are similar
- Longer wires restrict board level clock rates
- Board routing complexity generally not an issue.

Bandwidth Summary

- Consider connection to device 3 IO connections away
- Bandwidth between devices is 50% less for 8-way interconnect
- Example

8-way vs. 4-way					1-hop vs. 4-way				
1.5	1.3	2.0	2.0	2.0	3.0	2.0	2.0	2.0	2.0
1.5	1.3	1.8	1.8	2.0	1.5	2.0	2.0	2.0	2.0
1.5	1.3	1.5	1.8	2.0	1.0	1.3	2.0	2.0	2.0
0.5	0.8	1.3	1.3	1.3	0.5	0.5	1.3	2.0	2.0
*	0.5	1.5	1.5	1.5	*	0.5	1.0	1.5	3.0



Summary

- **Most FPGA systems require multiple devices. Topologies affect performance and use.**
- **One common use of multi-FPGA systems is logic emulation**
- **An example system (virtual wires) uses a near-neighbor mesh with several external interfaces.**
- **Topology is still an active area of research as “devices” migrate inside the chip.**