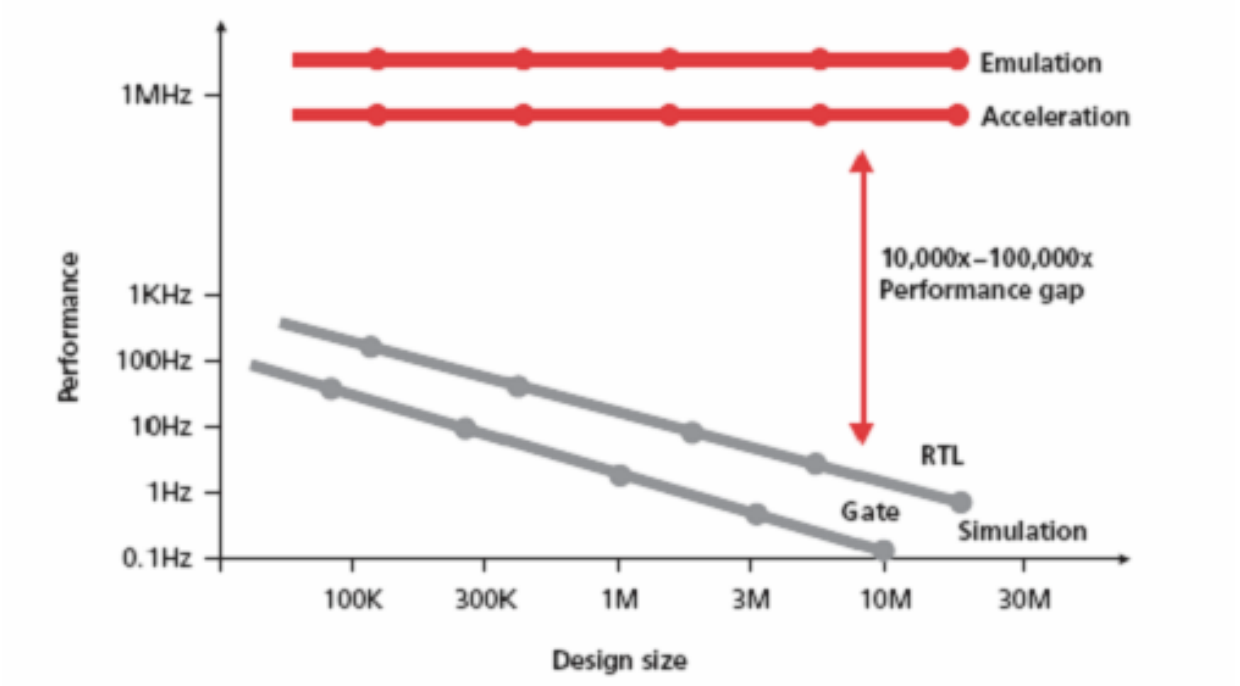
Hardware Acceleration

Dr. Shubhajit Roy Chowdhury,
Centre for VLSI and Embedded Systems Technology,
IIIT Hyderabad

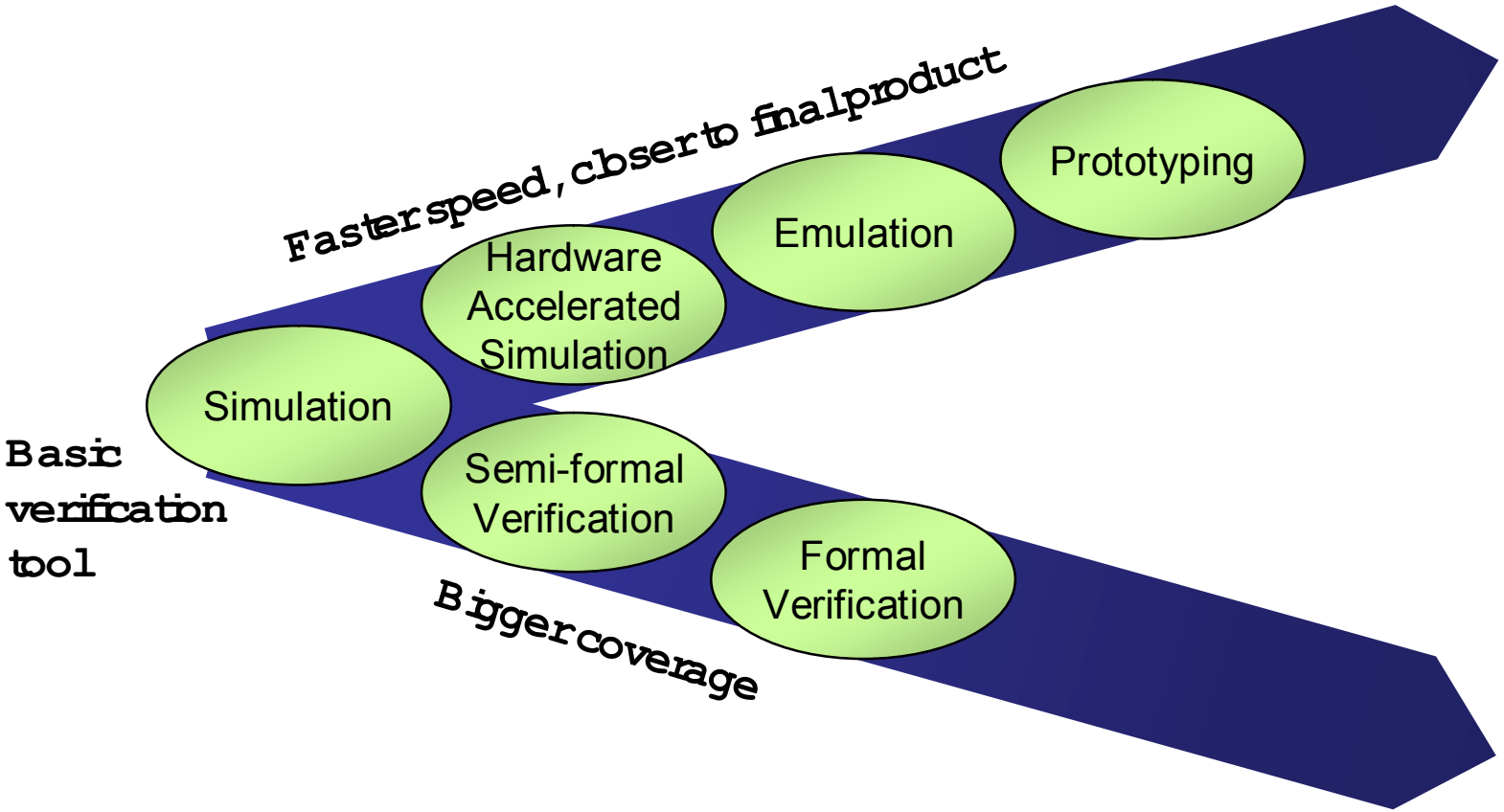
The Problem

1. Verification tools have not kept pace with the incredible rate at which design sizes are growing.
2. Each verification stage has its own methodology, tools, models and user inteface.
3. Embedded Software debug/integration is started late after first silicon/prototype.



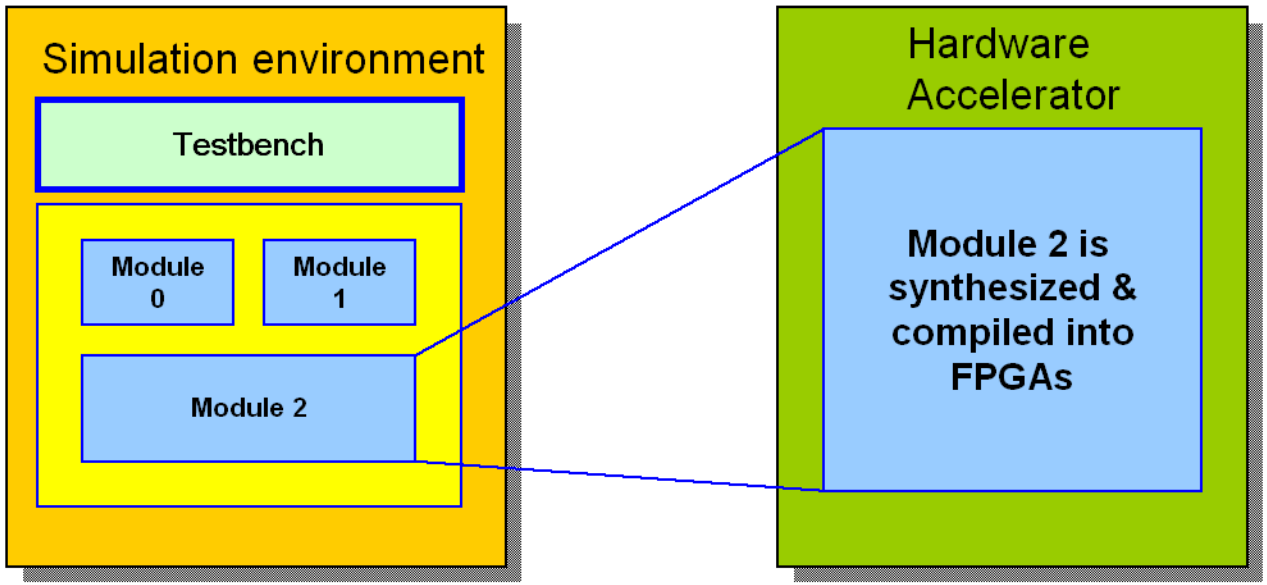
Design complexity

Methodologies



Hardware-Accelerated Simulation

- **Simulation performance is improved by moving the time-consuming part of the design to hardware.**
- **Usually, the software simulation communicates with FPGA-based hardware accelerator.**

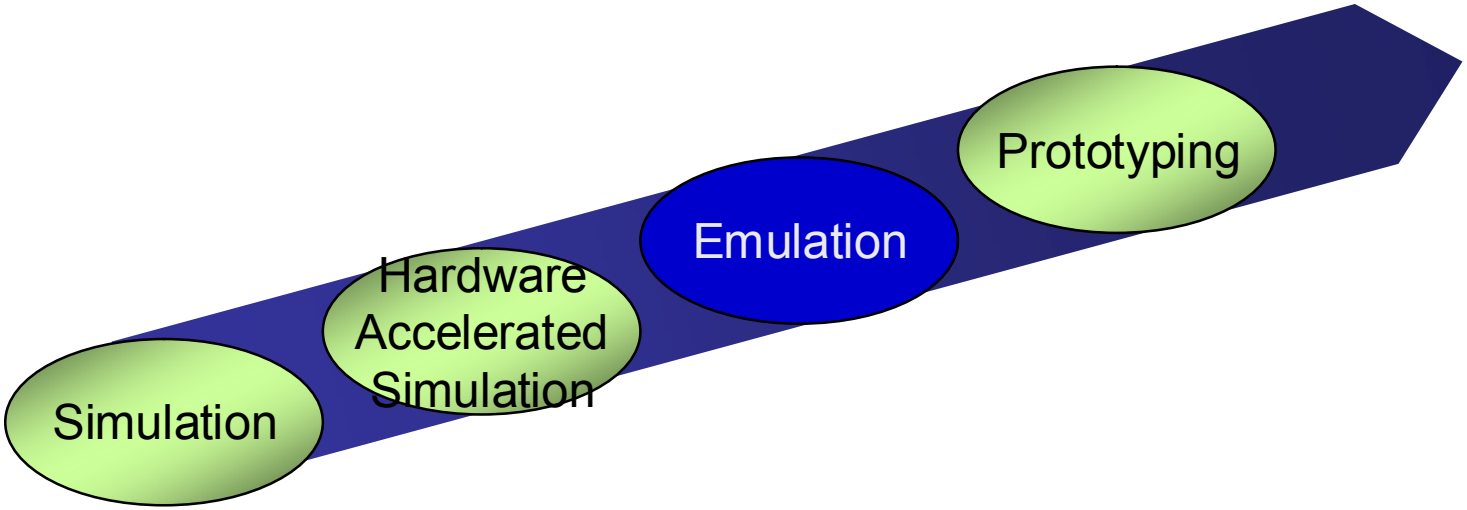


Hardware-Accelerated Simulation

- Advantages
  - Fast (100K cycles/sec)
  - Cheaper than hardware emulation
  - Debugging is easier as the circuit structure is unchanged.
  - Not an Overhead : Deployed as a step stone in the gradual refinement
- Disadvantages (Obstacles to overcome)
  - Set-up time overhead to map RTL design into the hardware can be substantial.
  - SW-HW communication speed can degrade the performance.
  - Debugging of signals within the hardware can be difficult.

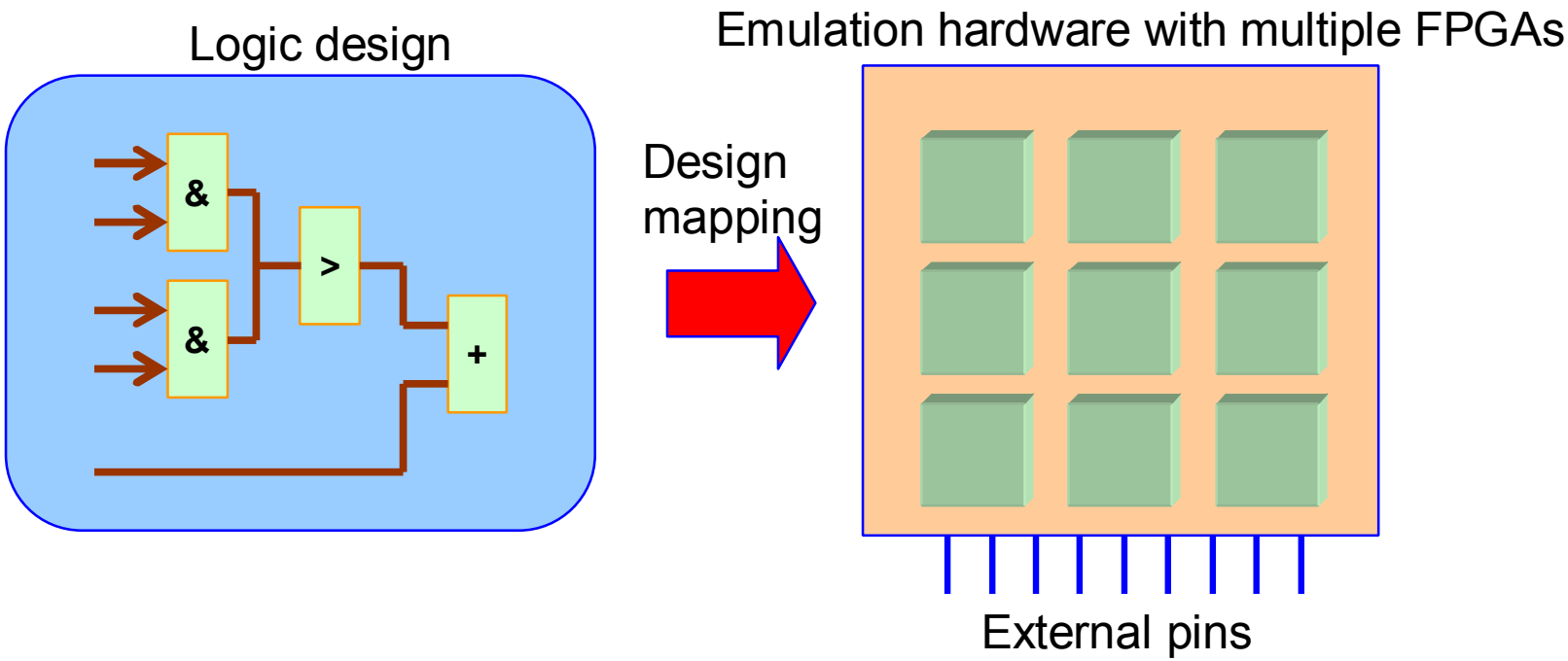
Emulation

- Imitating the function of another system to achieve the same results as the imitated system.
- Usually, the emulation hardware comprises an array of FPGA's (or special-type processors) and interconnection scheme among them.
- About 1000 times faster than simulation.



Emulation

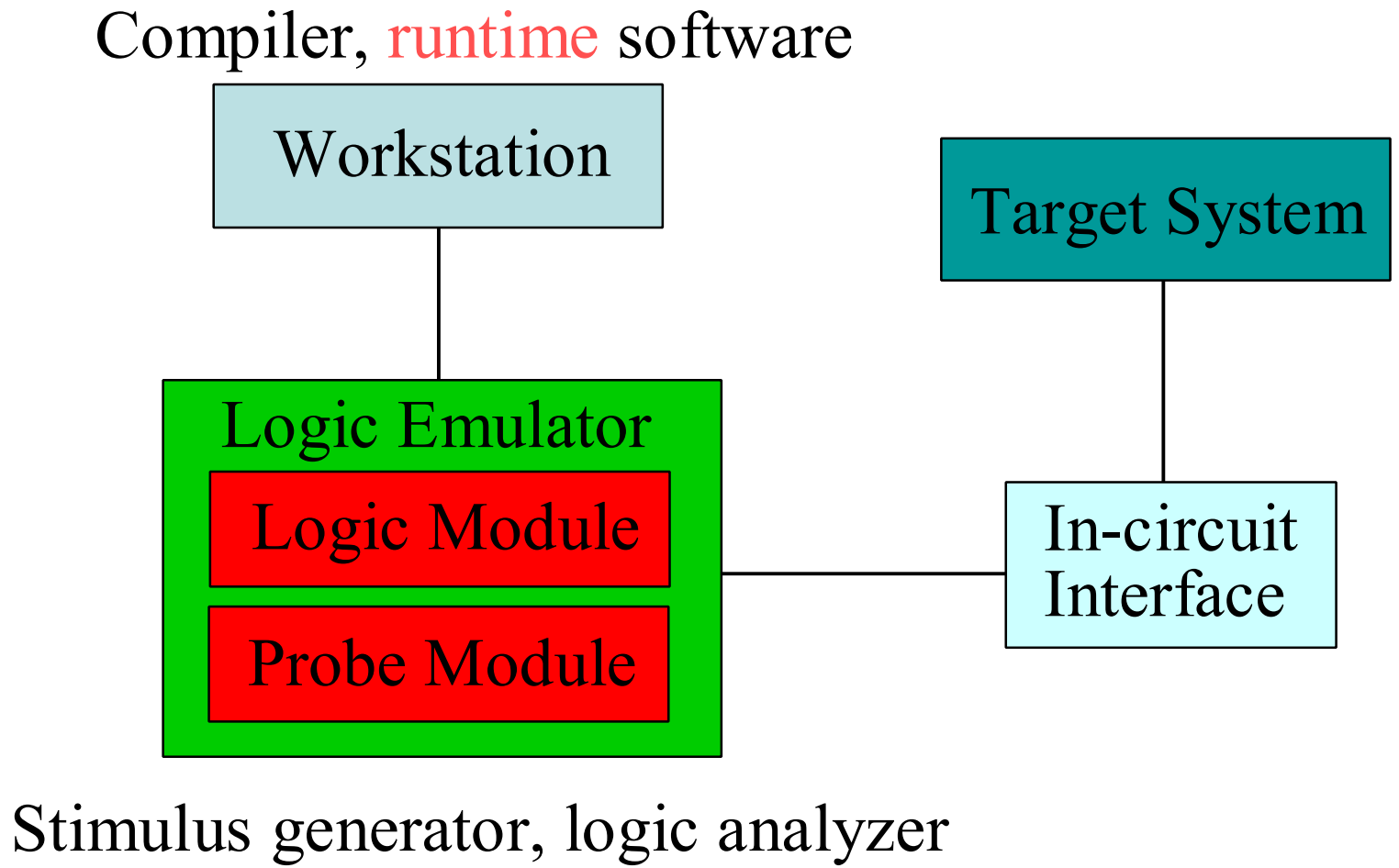
- User logic design is mapped to emulation board with multiple FPGA's or special processors.
- The emulation board has external interconnection hardware that emulates the pins of final chip.



Emulation

- Pros
  - Fast (500K cycles/sec)
  - Verification on real target system.
- Cons
  - Setup time overhead to map RTL design into hardware is very high.
  - Many FPGA's + resources for debugging
    → high cost
  - Circuit partitioning algorithm and interconnection architecture limit the usable gate count.

Typical Logic Emulation Environment



Design Verification Issues

- Simulation-based verification methods have run out of steam when chip complexity grows.

- Emulation is a verification technology that grows along with design size.

Why we need Logic Emulation?

- Design verification issues.
- Real-time operation.
- System-level testing.
- Rapid prototyping.

Real-Time Operation

- Simulation requires test vector development which is costly and difficult.
  - **Verification depends on test vector correctness**.
- Certain applications must be verified in real time - human perception: **audio** and **video**.
- Emulation connected to actual hardware can run:
  - real diagnostic code,
  - operating systems, and
  - applications.

System-Level Testing

- Often the chip meets its specifications but it fails in the system.
- We have to verify the system-level interactions between the chip and other components. They are hard to formalize.
- Internal probing is impossible when the chip is fabbed and placed in a system
- But it is possible using emulation.

Rapid Prototyping

- Once emulated design is debugged it is available for immediate use by *software developers* for software debugging.

- Emulated design is available for demo and experiments with architecture on real applications and data.

Considerations for programmable interconnect

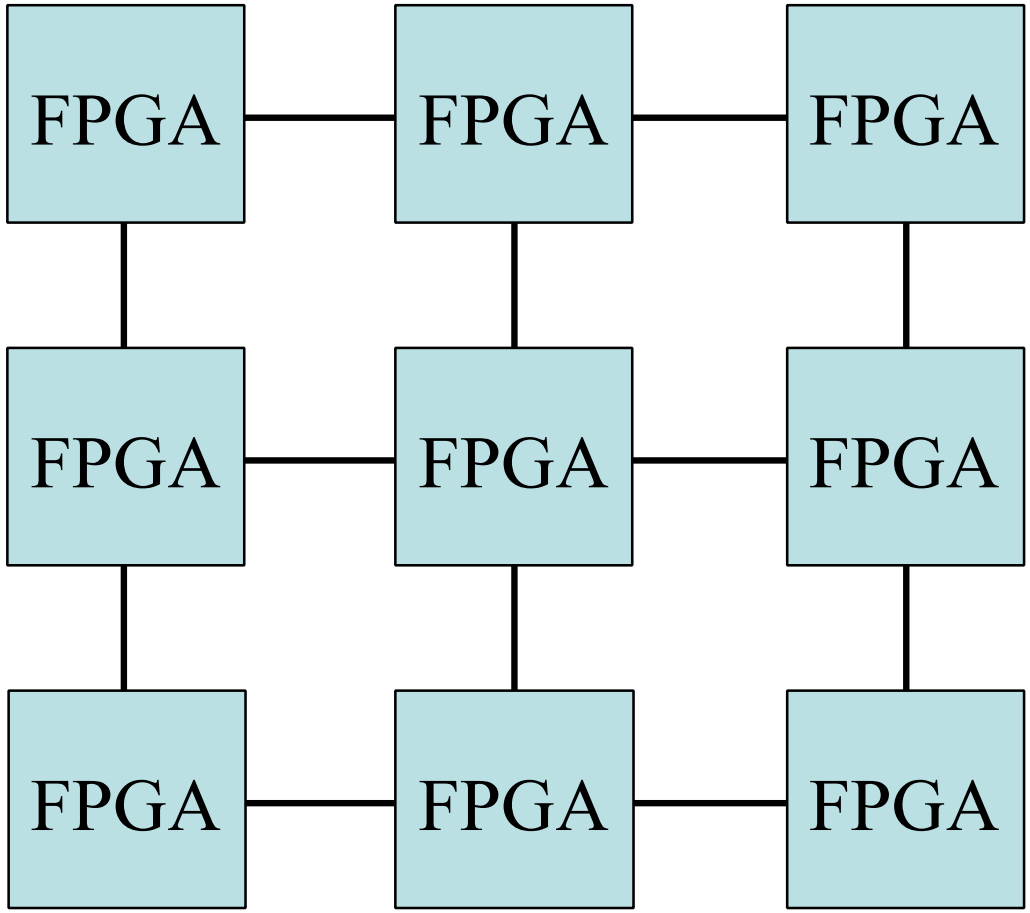
- The capacity of logic and interconnection depends on package constraints.

- The interconnect structure must:
  1. Provide successful connectivity,
  2. Maximize FPGA utilization, and
  3. Minimize delay and skew.

Structures of Multi-FPGA Systems

- **Topologies:**
  - Mesh - nearest neighboring.
  - Crossbar - full and partial.

Nearest Neighbor Interconnection

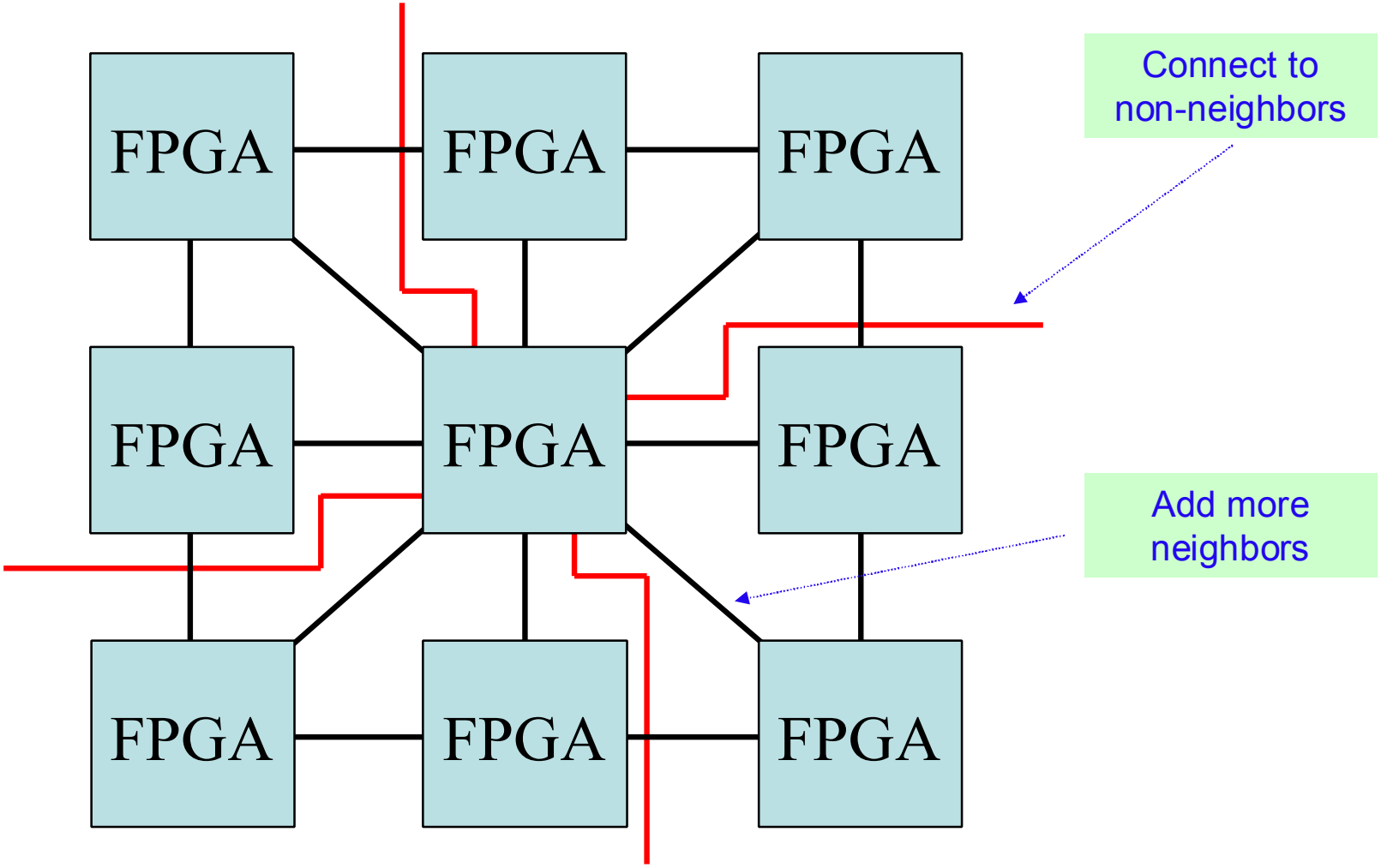


Advantages and Disadvantages of Nearest Neighbor Interconnection

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

- Disadvantages:
  - Routing is easily blocked.
  - The "through pins" limit the logic utilization of FPGAs.
  - Long and unpredictable delays.
  - No natural hierarchical extension.

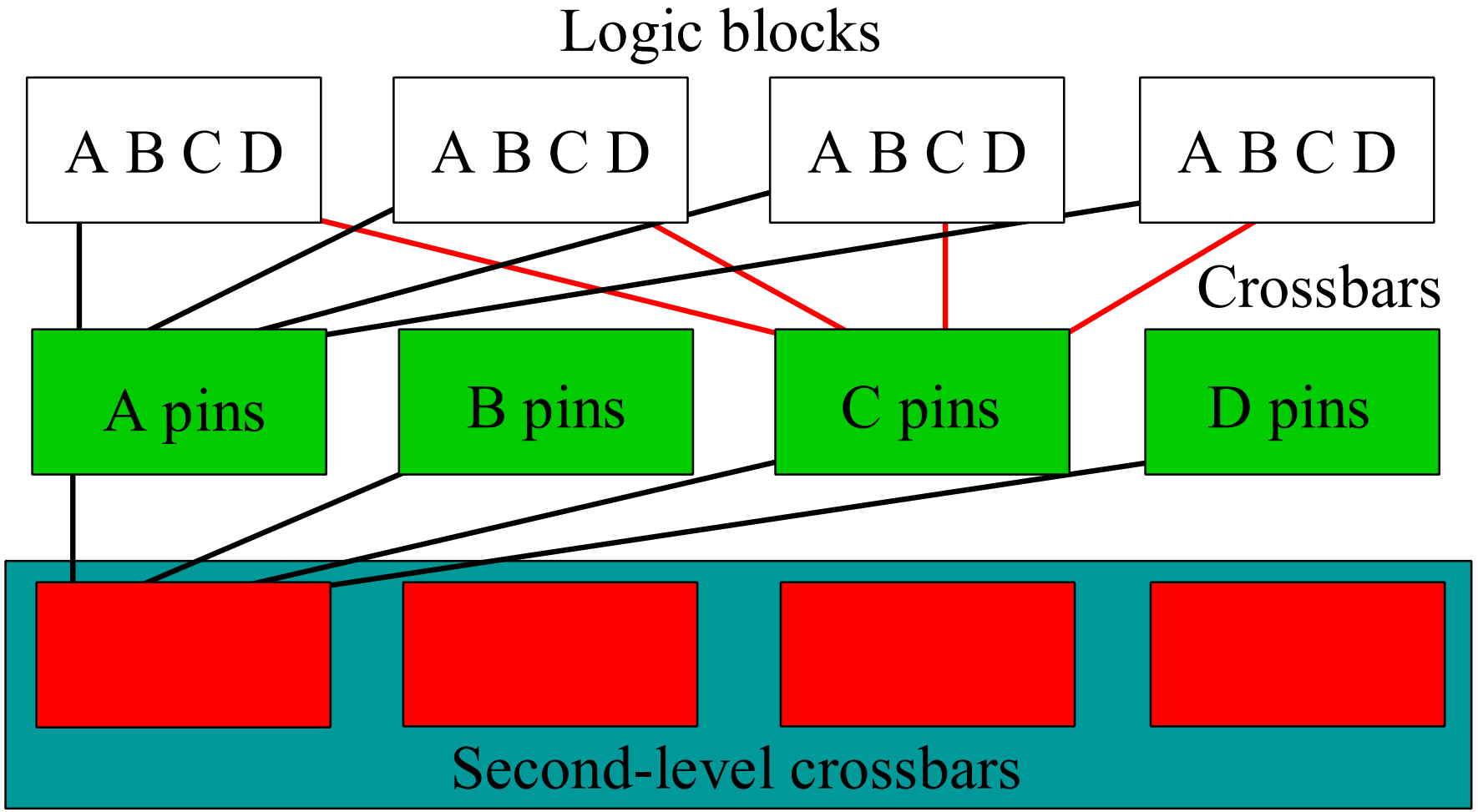
Nearest Neighbor Extensions



Advantages and Disadvantages of nearest-neighbor extended architectures

- Advantages:
  - More choices for router by adding diagonal lines & skip lines.
- Disadvantages:
  - More complex PCB.
  - More complex routing software.

Partial Crossbar Interconnect



Partial Crossbar Interconnect

- Partial crossbar consists of a **set of small full crossbars,**
  - connected to logic blocks
  - but not to each other.

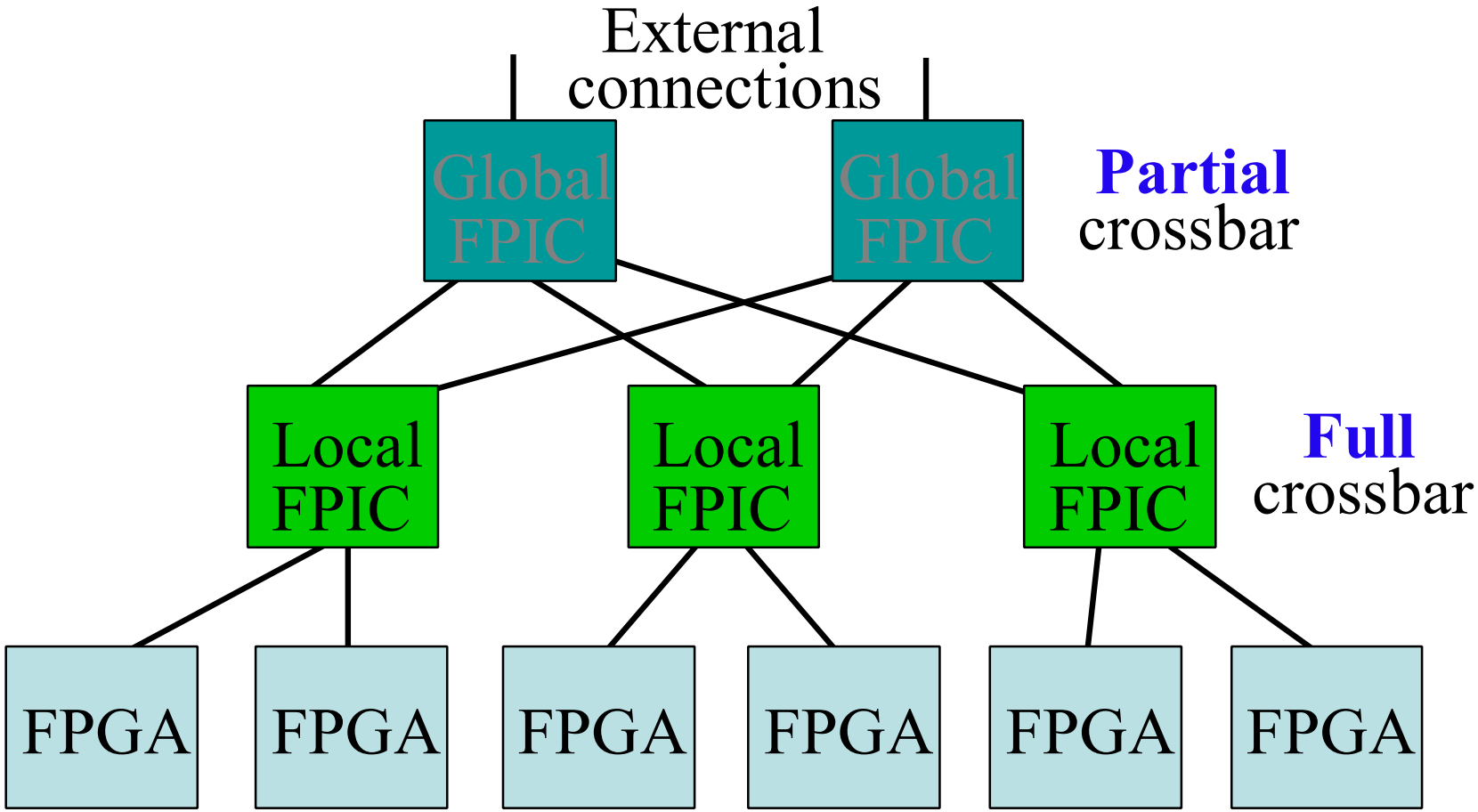
- I/O pins of each FPGA are divided into subsets.
  - Each subset is connected by a full crossbar circuit switch.

- Partial crossbar is a potentially blocking network.

Characteristics of Partial Crossbar Architecture

- Partial crossbar's size is proportional to the number of FPGA **pins**.
- All interconnections go through one/three crossbar chips for a one-level/two-level partial crossbar interconnect –
  - **delays are uniform and bounded**.

Mixed Full and Partial Crossbar



Products

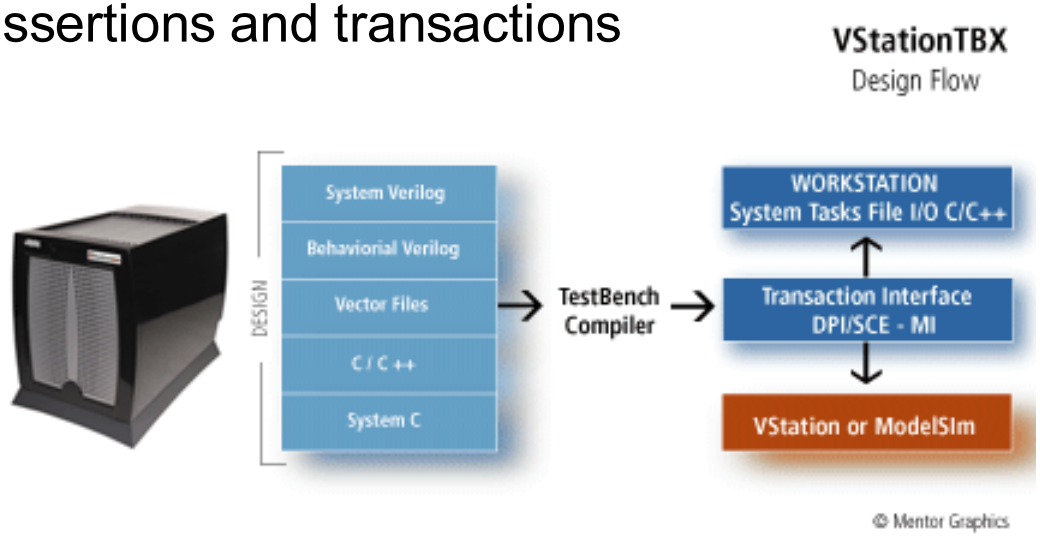
Cadence - PALLADIUM

- Simulation acceleration and in-circuit emulation in one system
- natively supports Verilog, VHDL, System Verilog, SystemC, SystemC Verification Library, OVL, and PSL/Sugar assertions.
- Dynamic probes/events allow interactive debug during run-time
- Comprehensive verification with "live" data
- Support for assertions in hardware with no performance degradation
- High Capacity -
  - Provides up to 100x-10,000x RTL performance (run four days of simulations over lunch at 100x).
  - Compiles up to 30 million gates per hour on a single workstation
  - Maximum capacity of 256 million gates
  - Supports up to 32 users in local or remote access
  - Highest throughput for HW/SW co-verification

Products

Mentor - VStationTBX

- Single verification environment for simulation and verification
- High-performance verification accelerator for designs and testbenches
- Built on standard languages: SystemC, SystemVerilog, C/C++
- Behavioral Verilog testbench and memory compiler
- VHDL and Verilog RTL debug
- HDL acceleration 10x-100x faster than co-simulation
- Accelerates the entire design process with advanced techniques, such as assertions and transactions



Other solution

- Homebrew with FPGAs –
  - Partition design to number of FPGAs.
  - Use FPGA on-chip processors or configurable processors to drive testbench.
- Pros
  - Low cost solution.
  - fast
- Cons
  - Most embedded processor compilers supports C/C++ languages.
  - Requires two verification environments.
  - Need to establish debug connection between WS and FPGAs

Conclusions

Hardware Acceleration/Emulation

Simulation Acceleration

Verification Acceleration

Project Acceleration

Thank you