

ProtoFlex: FPGA-Accelerated Hybrid Simulator

Eric S. Chung, Eriko Nurvitadhi
James C. Hoe, Babak Falsafi, Ken Mai

Computer Architecture Lab at
Carnegie Mellon

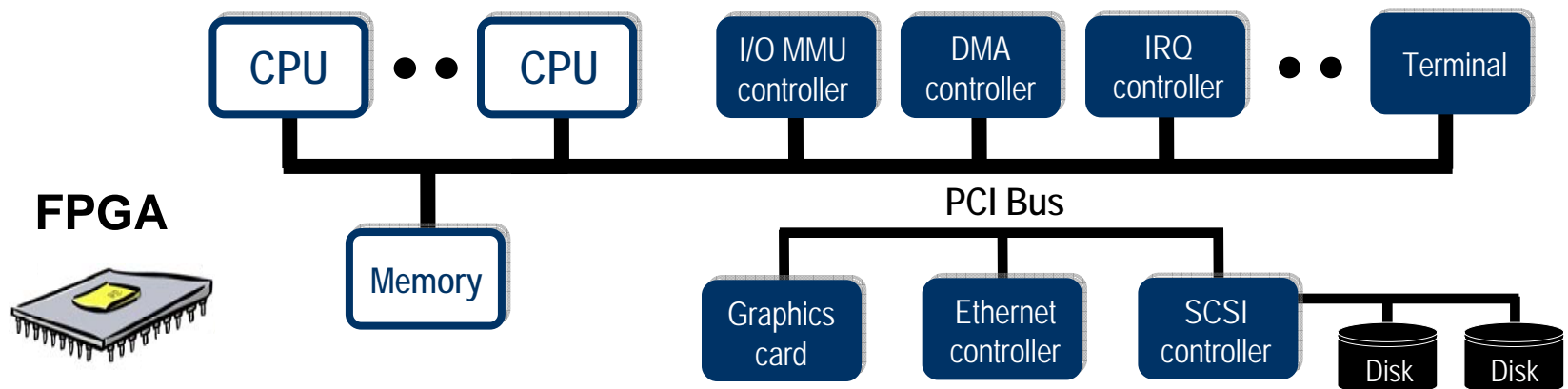
Our work in this area has been funded
in part by NSF, Intel, and IBM.



Multiprocessor Emulation

- Need fast MP emulators to study future revolutionary changes in HW and SW
- Hardware concurrency of FPGA emulation can scale up multiprocessor simulation speed
- BUT, we also want full-system fidelity (OS and I/O support)

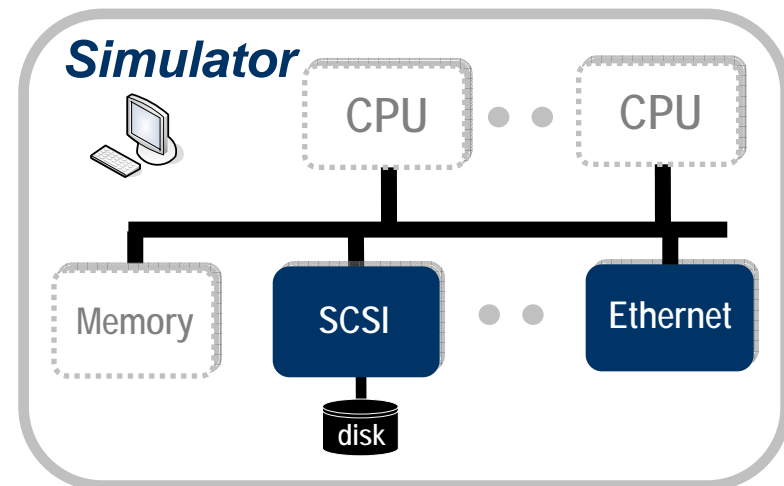
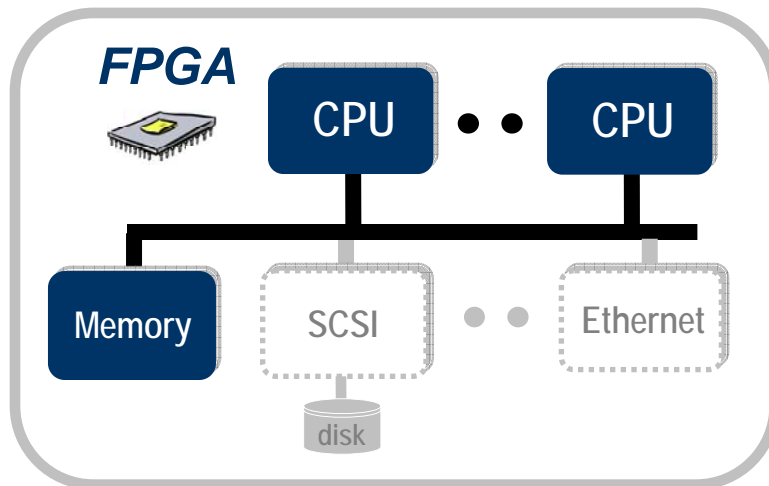
Can we really build a 1000-node fully detailed MP system, FPGA or not?



How to get full-system fidelity without building the full system?

Combining Simulators & FPGAs

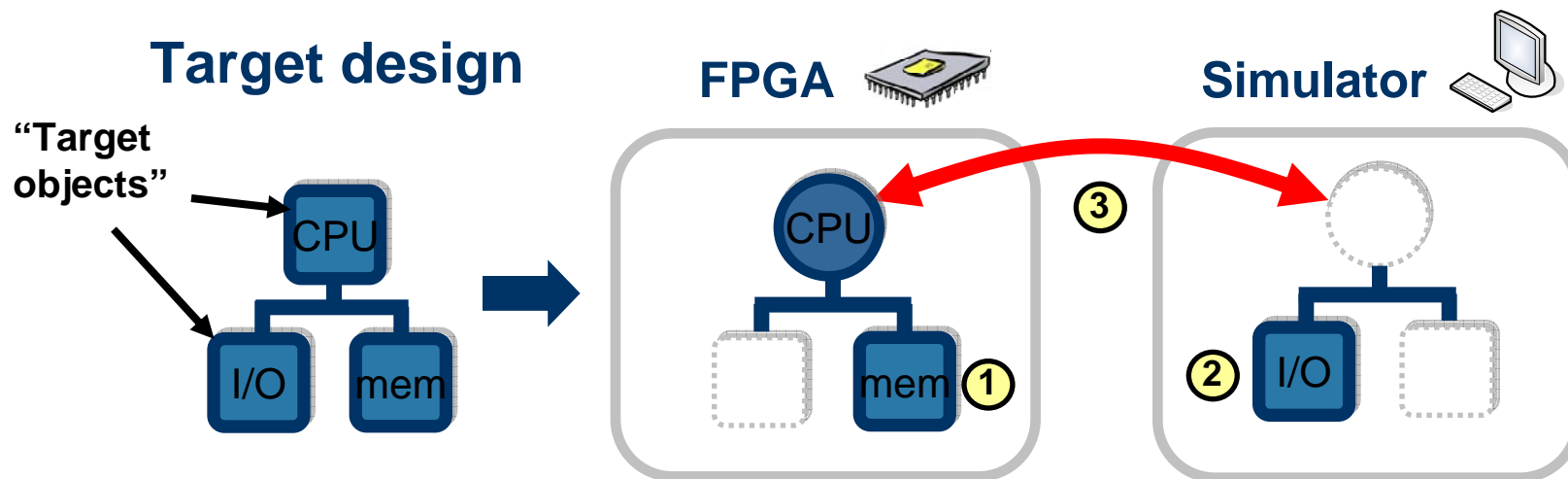
- **Simulators already provide full-system behaviors**
 ⇒ why not just simulate infrequent behaviors (e.g., I/O devices)?



- **Advantages**

- avoid implementing infrequent behaviors
 ⇒ simplify full-system emulator development
- low impact on scalability and performance acceleration

Transplanting Hybrid Simulator



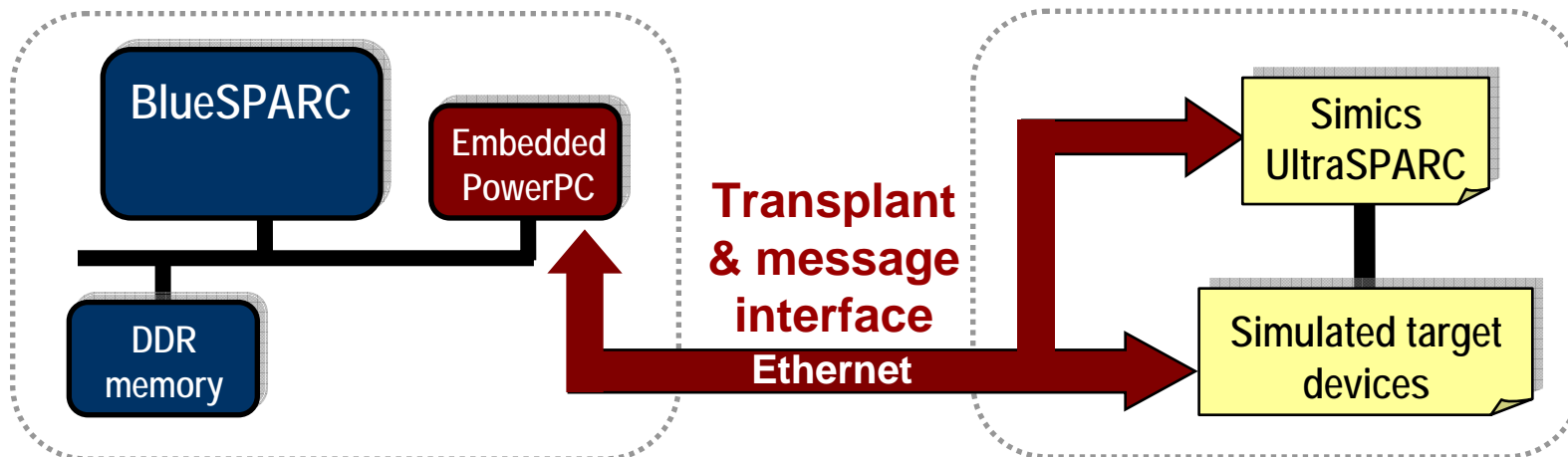
- **3 ways to map target object to hybrid-simulation host**
 Emulation-only ① Simulation-only ② Transplantable ③
- **Transplantable objects**
 - switch modes between FPGA & simulator hosts
 - *complete behavior* need not be implemented in FPGA
i.e., implement only the frequently used ISA subset in FPGA

It Really Works

Xilinx XUP Virtex-II Pro 30



Virtutech Simics
(commercial simulator)



+

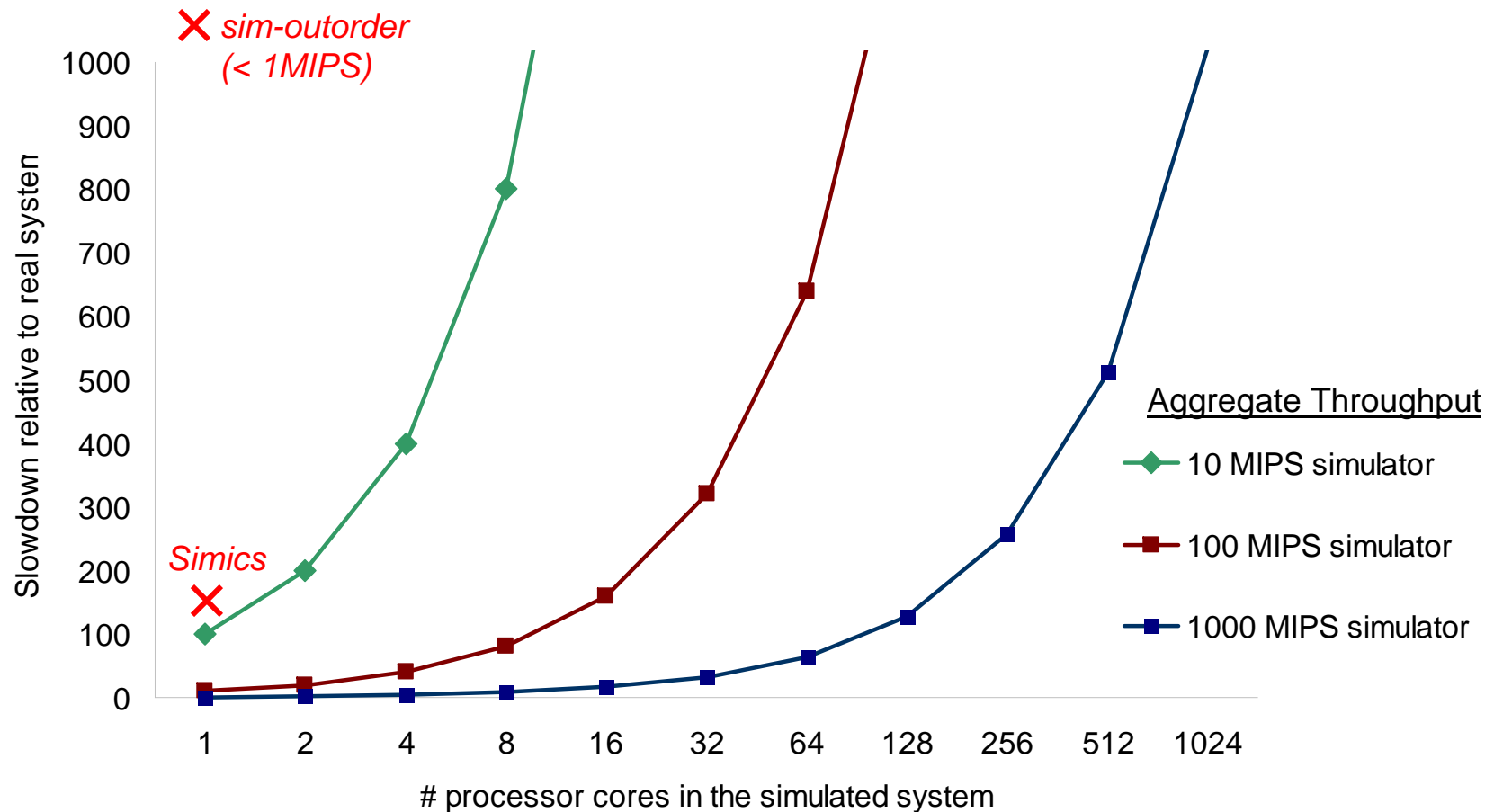


= SUN 3800 Server
(1x UltraSPARC III,
Solaris8)

1 graduate student in 6 months

**How to build a 1K-node MP emulator,
without building 1024 nodes?**

How fast do you need to simulate?



In the uniprocessor world

- up to 100x slowdown usable for interactive SW research (e.g. Simics)
- 1k to 10k slowdown usable for design exploration (e.g. SimpleScalar)

Different ways to simulate 1K cores

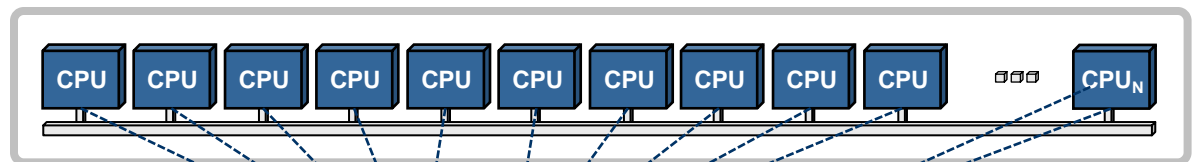
- Even for a 1K-node MP, only need 1000 to 10,000 MIPS (in aggregate) to do useful work
- The naïve approach
 - build a fast ISA core (estimate 100 MIPS per core)
 - physically replicate the core 1000 times
 - ⇒ *10x to 100x faster than it needs to be*
 - ⇒ *Why spend effort on performance I don't need*
- The better approach—think in terms of MIPS
 - build a 100-MIPS ISA core with a statically interleaved pipeline that can support multiple contexts
 - interleave 100 contexts per core to emulate a 1K-node system with just 10 physical cores
 - ⇒ *the parameters and the effort required can be tuned to make the emulator just fast enough and not more*

PROTOFLEX^{MP}

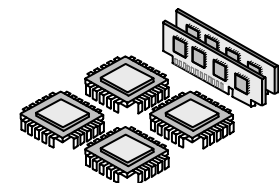
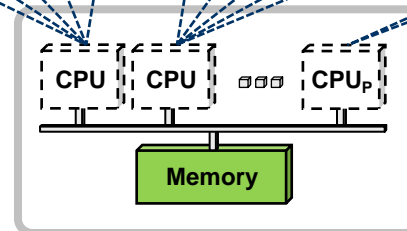
- Build a 1000-MIPS simulator from 10s of FPGAs
 - maximize throughput per emulation engine to be share by multiple interleaved contexts
 - multiplex a large number of emulated contexts onto a few emulation engines

Base the number of emulation engines you need on how much performance you need, and not on how many nodes you are emulating

N-way target system

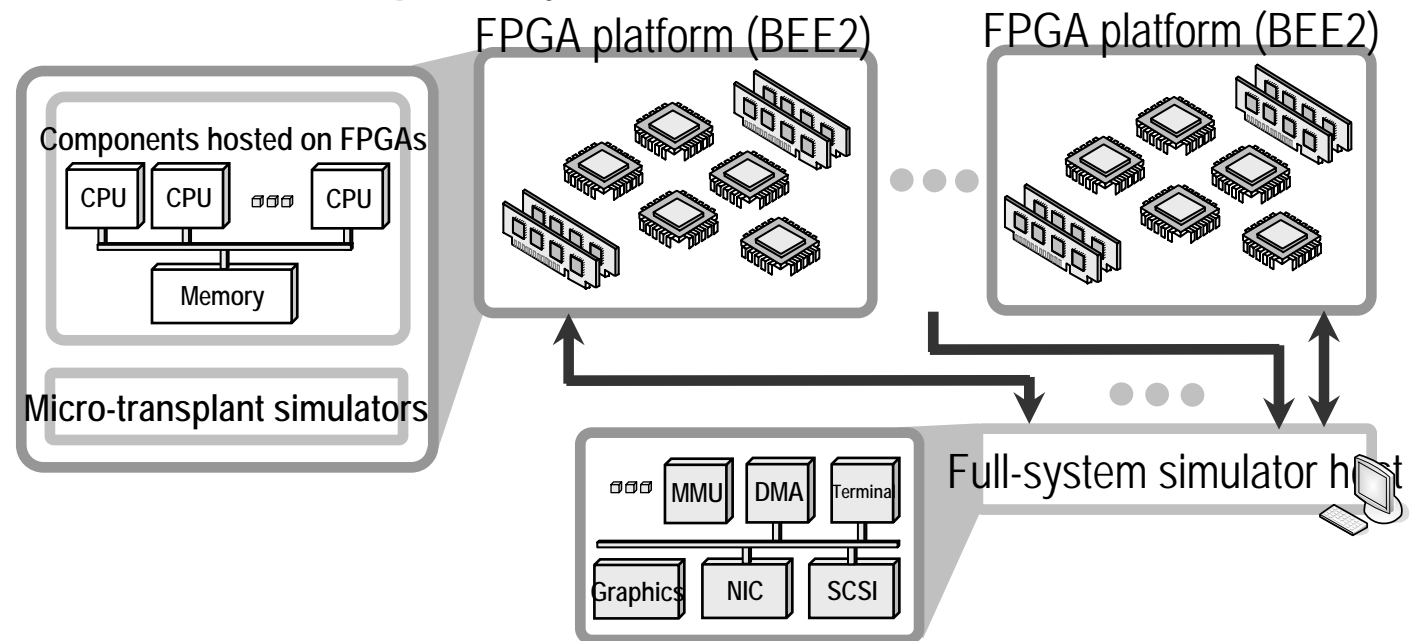


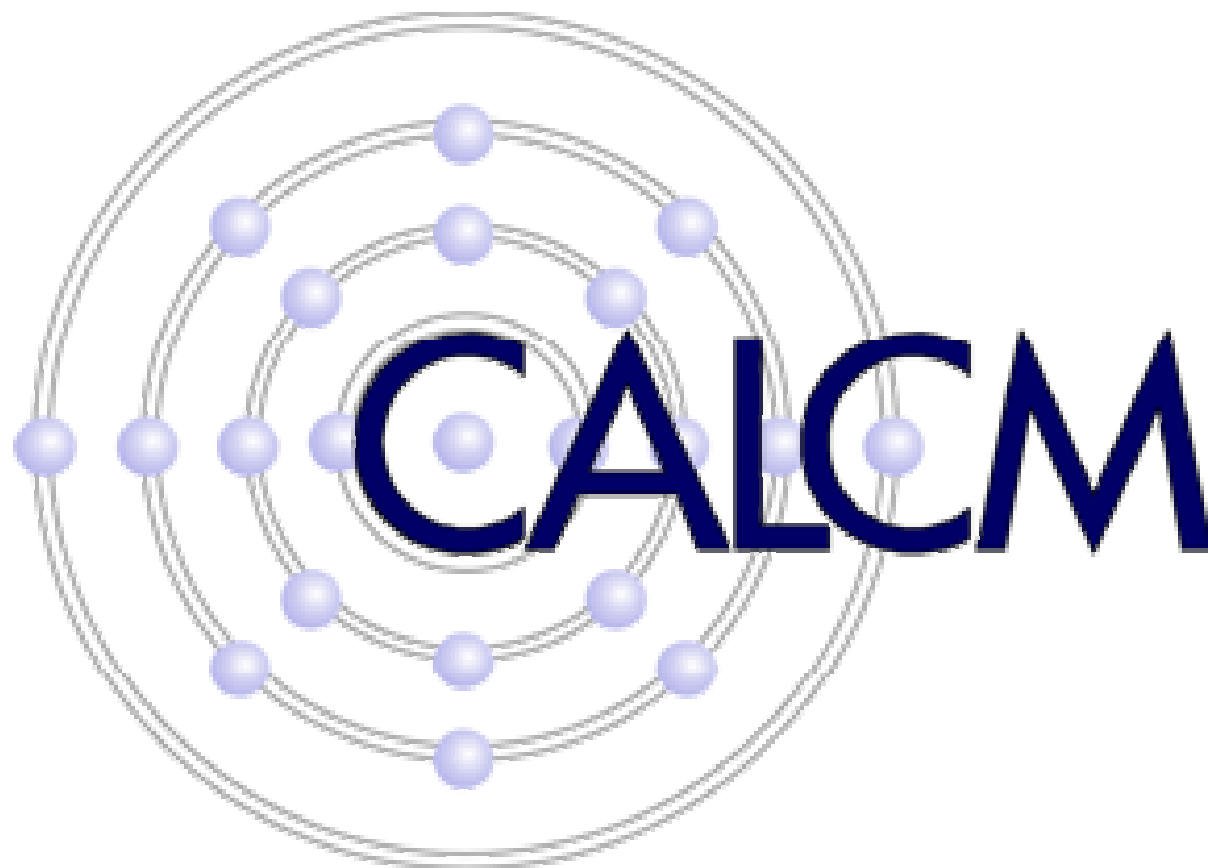
P-way FPGA emulation engines, $P \ll N$



Conclusions

- Technology to build a large-scale full-system multicore/multiprocessor simulator
 - Use hybrid transplantation to avoid a full-system construction effort
 - Use interleaved emulation cores to reduce physical system size and complexity





RAMP

ProtoFlex
Computer Architecture Lab (CALCM)
<http://www.ece.cmu.edu/CALCM>