RTL Design on a Laptop

Ed Grochowski
Senior Principal Engineer
Intel

18th Annual Electronic Design Process Symposium
April 7, 2011
Monterey, California

Evolution of Engineering Computing Environment at Intel

Decade	Projects	Computing Environment	Server Hardware/OS	Client Hardware/OS
1980s	Intel486™ Processor	Mainframe	IBM 3090 Amdahl UTS IBM AIX	Terminal Sun 386i SunOS
1990s	Intel® Pentium® Processor	Workstation	IBM RS/6000 IBM AIX HP PA-RISC HP-UX	IBM RS/6000 IBM AIX HP PA-RISC HP-UX
2000s	Intel® Core™ Processor	Cloud	Intel® Xeon® Processor Linux	Desktop/laptop PC Microsoft Windows

What's Wrong with Cloud Computing?

- Requires network connection
- Noticeable latency
- Underutilizes client hardware

A Bet

I was talking to my friend Naresh Sehgal about the compute environment

"Unpredictable due to sharing and network latency"

"It's clunky!"

"I bet my laptop would be faster!"

The race was on!

RTL Design on a Laptop Proof-of-Concept Experiment

Hardware (late 2009)

- HP Envy 15
- Intel® Core™ i7-820 processor (1.73GHz)
- 16GB physical memory (12GB available to Linux under VMWare)
- 320GB (2 x 160GB) of solid-state disks

Software

- SUSE Linux Enterprise Server 10 (x86_64)
- VMWare
- Microsoft Windows 7

Laptop Setup

- Set up standard RTL environment

- Collaborative effort between
  - Corporate IT (Naresh Sehgal, Satish Sammanna)
  - Project Design Automation (Amit Kashyap)
  - Project Architecture (me)

- Laptop appears identical to server

- Everything worked!
  - No laptop-specific bugs found

Evaluating the laptop - Tasks

Coded and debugged the RTL for second-level TLB

- Important new feature
- About 7K lines of new RTL code

Developed random snoop injectors

- For the L1 fill buffers and L1 instruction cache
- Reproduced hard-to-find RTL bug
  - Bug required two consecutive snoops to the fill buffers and instruction cache, respectively, to reproduce failure

Developed random snoop injectors

- For L1 data cache
- Try to reproduce a paging-related bug

Strengths

Instantaneous response for interactive tasks

- Editing code and viewing waveforms are extremely fast
- Helps productivity
  - Concentration not broken due to lag
  - Moving the cursor can momentarily stymie the server!

A server accessed via VNC offers mediocre response times

- Ping time is 45ms over VPN connection using my home DSL
- Ping time is 16ms at Intel site

Laptop is clear winner on interactive tasks

Weaknesses

For compute-bound tasks, laptop does not offer as much performance as the server

- I do most of my work at the core and CPUBOX levels

Quad-core CPU underutilized by single-threaded tools

- Could run four instances of the VCS simulator (on the core and CPUBOX models) without swapping

Laptop fan becomes noisy when CPU is busy

Laptop weaknesses are acceptable

Local Copies of Files

Laptop maintains copies of all necessary files on its local disks

- RTL model
- RTL and validation environments
- CAD tools

Compute servers mount NFS disks

- For security and performance reasons, we could not mount NFS disks on the laptop

Files were manually copied from the server to the laptop

- All three categories of files were rapidly changing
- Significant effort required to keep laptop in sync with server
- Automation required for wider laptop deployment

Limited Network Bandwidth

Most problems were due to slow (768Kbit) home DSL

- Complete checkout (bk clone) of the RTL model required 3.5 hours to transfer 800MBytes of data
- Incremental update (bk pull) takes one minute
- Tool updates require hours to a day
  - VCS version could be updated overnight, 3GBytes

Bitkeeper works well incrementally

CAD tools require network connection to talk to license servers

Impractical to share large trace files among engineers

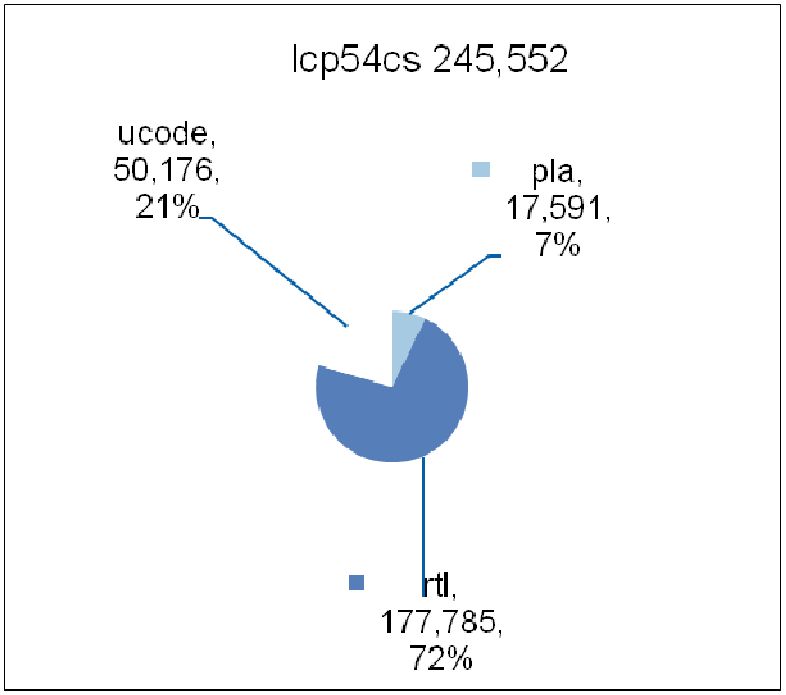
Network bandwidth limits the type of work that may be done

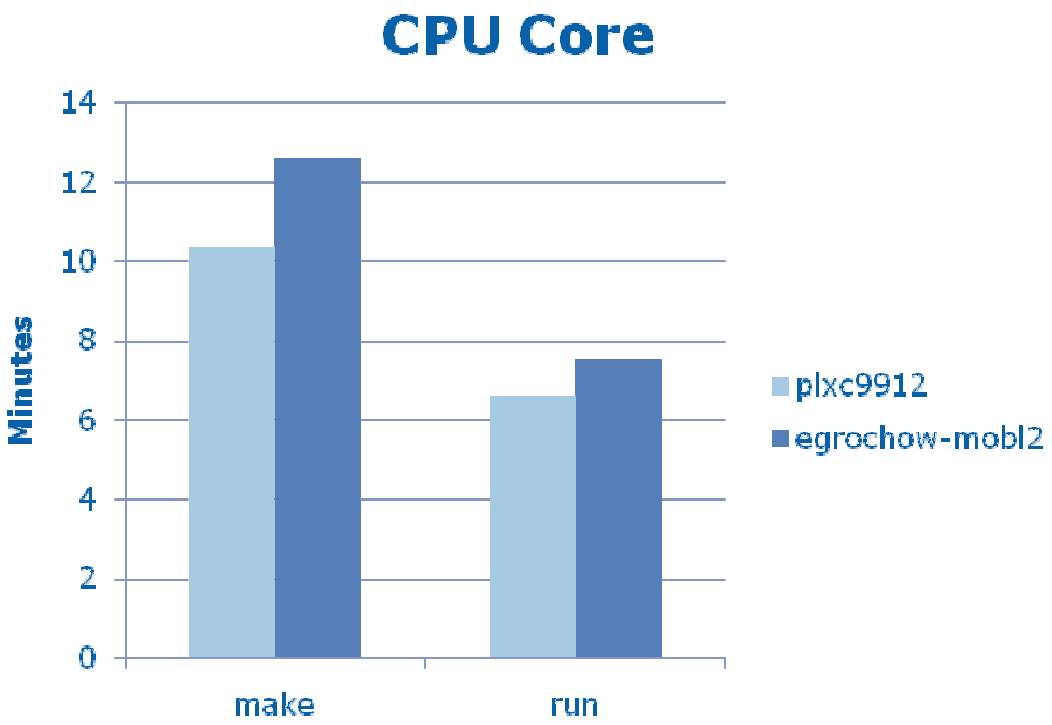
Benchmark Machines

Machine Name	Description
plxc9912	Intel(R) Xeon(R) CPU X5570 @ 2.93GHz stepping 05 74 GB physical memory
egrochow-mobl2	Intel(R) Core(TM) i7 CPU Q 820 @ 1.73GHz stepping 05 12 GB memory available under VM (16 GB physical memory)

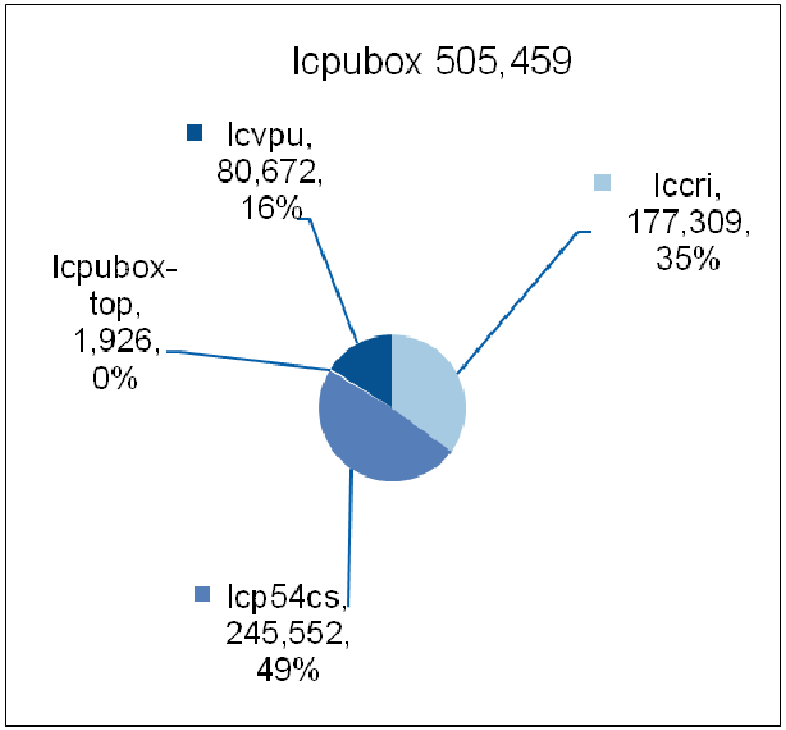
During these tests, I had exclusive use of the laptop, whereas the server was running background jobs

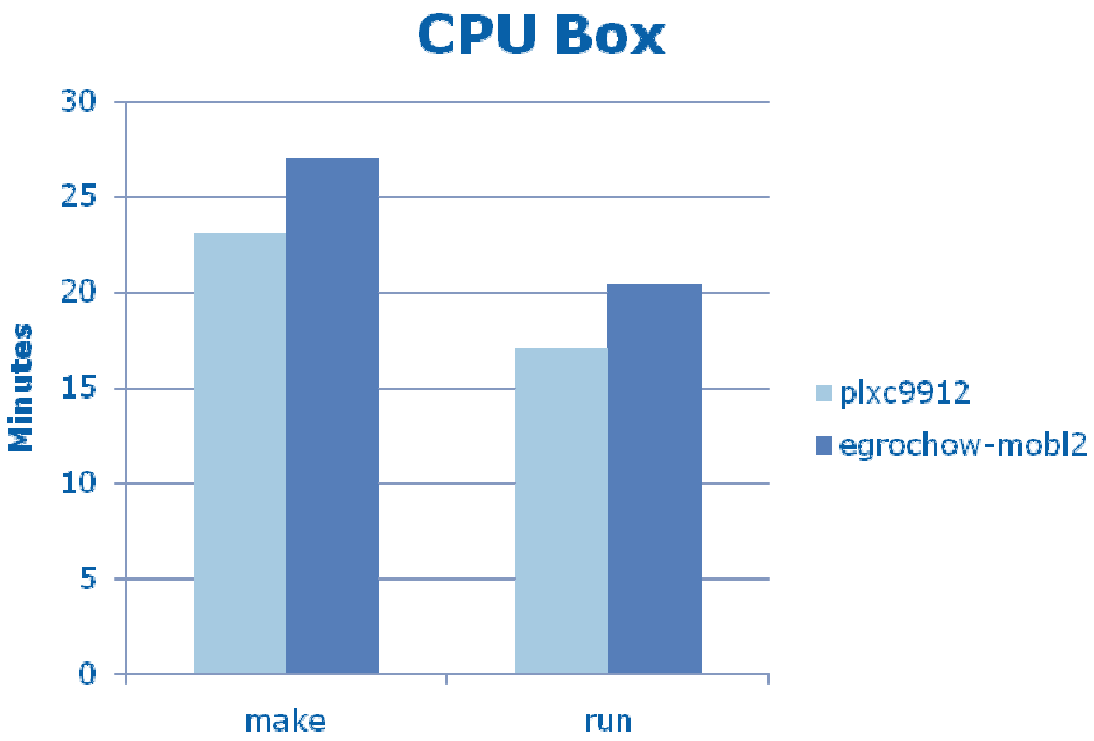
Benchmark Results – CPU Core Model



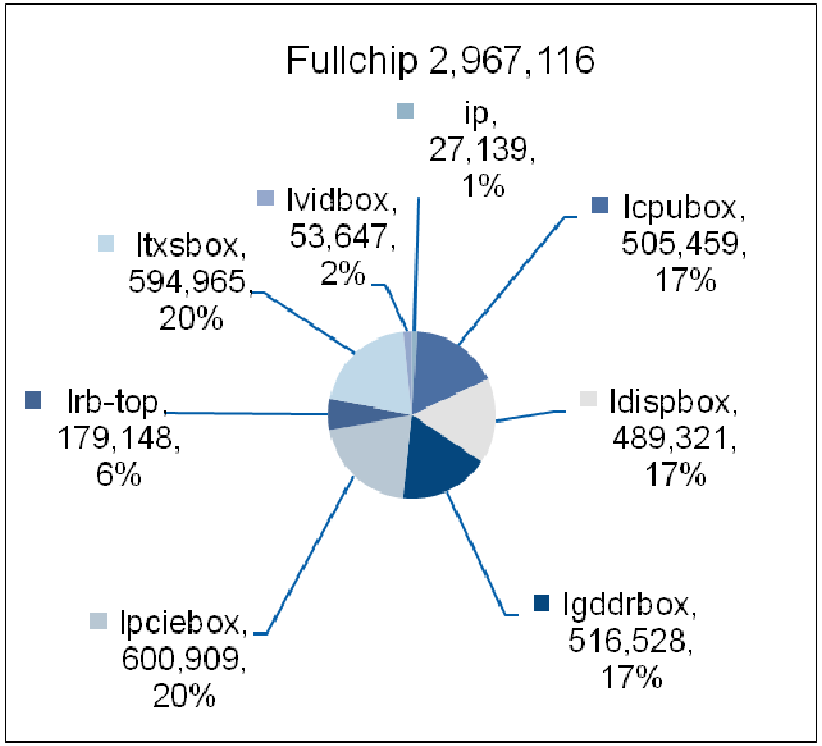


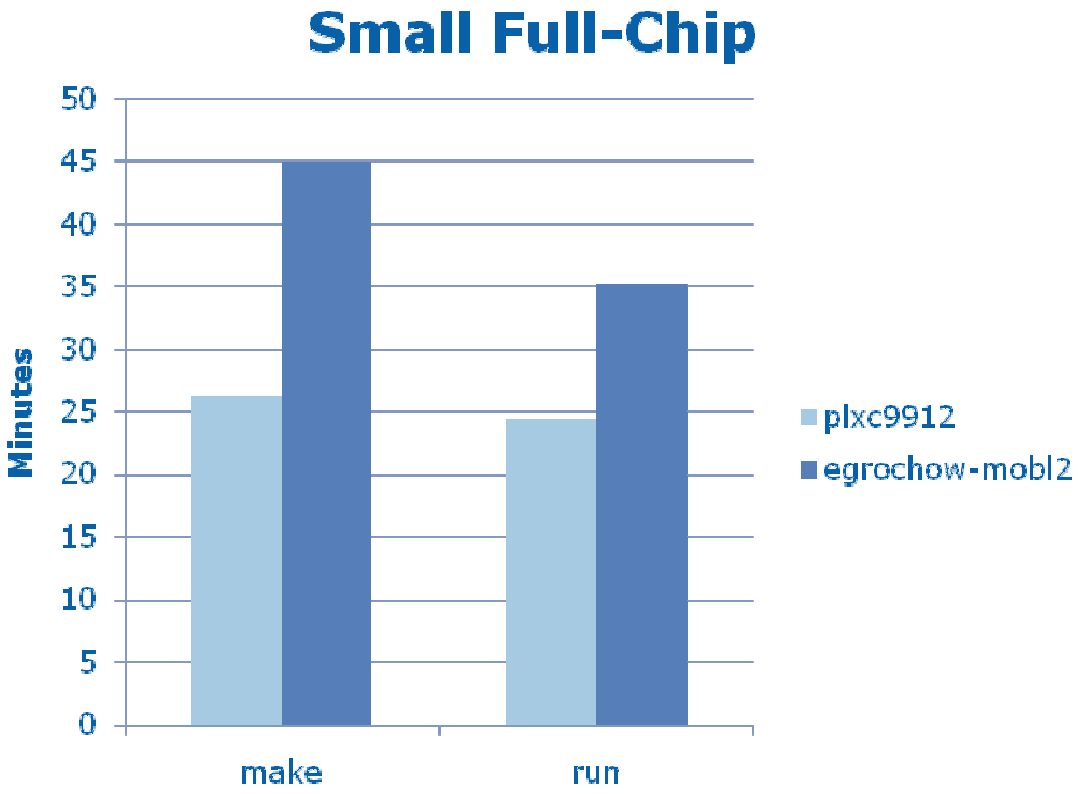
Benchmark Results - CPU Box Model





Benchmark Results – Small Full-Chip Model





Conclusion

Laptop is better	Server is better
Interactive design and debug	Running regressions and netbatch Sharing large amounts of data

The best computing environment would make use of both!

Follow-up the RTL laptop PoC experiment with a larger laptop deployment on a future CPU design project