

Does Design Verification Have a Future?

One consultant's point of view

Current Trends Indicate

Not a very good one!

The important question is
“Does design have a future?”

Current Trends

1. Boredom
2. ASIC designs are way down
3. Few large companies headquartered in our area
4. Few startups
5. Signing authority is way down
6. Consulting is getting better
7. “Technology” indicates other things
8. Little mention of this industry in state politics
9. Nice, free trinkets are now rare
10. Lower pay

Does Design Have a Future?

- A lot of IP in use
- A lot of reuse
- Heavy pricing pressures
- A lot of things that often changed are becoming static
- A lot of consumer products
- Seems like a lot of designs are no longer on the Moore's Law curve

Some Basic Observations - 1980

- 1980 starting salary was about \$21K
 - About \$51K in today's dollars
- Apple II, Northstar, etc. were the available home computer
- Basic and Fortran were the common languages
- Design was primarily SSI using some available LSI
 - Verification was simply lab debug
- ASIC design was rare
 - The name was not even in use
- The calculator and digital watch booms were fading

Some Basic Observations - 1980

- Which technology would win was still debatable
 - ECL, TTL, DTL, CMOS, NMOS
- ASIC (Gate Array) design was typically a few hundred gates
 - and yet it still took 18 months
- 64Kbit single voltage DRAMs were just appearing
 - Multi voltage
- PALs were just coming into their own
- Graphics was limited
- Which OS would win was debatable

Some Basic Observations - 1980

- Startups were relatively rare
- Stock options were rare
 - What to do with them was a mystery
 - Stock purchase plans were available
- IPO was the goal

Some Basic Observations - 1980

- A typical mistake by a typical engineer could cost the company hundreds of dollars
- A typical engineer did diverse things
 - Power calculations
 - Hand done timing
 - Board design
 - Signal integrity
 - Power integrity
 - Soldering
 - Vector generation

Some Basic Observations - 1993

- My first laptop - 1993
 - 4 Meg of RAM
 - 80 Meg of disk
 - 33 MHz
 - \$1675 – \$2336 in 2006 dollars
- My current laptop - 2005
 - 1 Gig of RAM – 256X
 - 80 Gig of disk – 1000X
 - 1.3 GHz – 39X
 - About \$1900 – 0.81X

What am I trying to show?

- Hardware seems more static than in the past
 - Few design options – ASIC, FPGA, CMOS
 - Easily predicted path – Moore's Law
 - Fewer platforms
 - Easy to segment the jobs
- Software is a little less static
 - Lots of languages
 - Lots of free software
 - Still a lot of old software
 - The basic products are static

Another Observation

- In the past hardware seemed to lead the consumer market
- Today, hardware follows the consumer market
- The resulting business climate is not necessarily friendly to engineers
 - Razor thin margins are unfriendly

What have caused the biggest impacts?

1. Synthesis tools applied to a modeling language
2. More than two signal layers on a board and in VLSI
3. Advanced C compilers
 - Helps at so many levels
4. Advanced packaging
5. Scan
6. Emacs and VI

What have also contributed?

Fast printers

Ethernet

Workstations

Verification languages

Waveform viewers

Phones

Faster computers

Faster simulators

More memory

Word processors

Coverage

Randomization

Things that don't seem to be helping

- Behavioral Synthesis
 - Abstract up a little and you get too many knobs
 - Abstract up a lot and you have IP
- Formal verification
 - Can't get people to do it
- Differences in the verification languages
 - Verilog, VHDL, C/C++, SystemC, E, SystemVerilog, Vera

What do all these have in common?

They are all old innovations

Silly Things I've Heard and Seen

- We can't outsource to a site 20 miles away
 - Today this same group outsources to India
- Over budget project with 5 consultants kept on for 6 months “just in case”
- One company developing 3 essentially identical routers at the same time
 - Different markets

Silly Continued

- In college I heard that we simply needed 2X performance and 4X memory and we would have AI.
- Plans to buy an accelerator to speed up 20 minutes of simulations.
- Plans to use an emulator to get 300X when the simulation environment could be sped up 20X with a week of work

More Silliness

- A 5 million gate ASIC's verification being considered done with 2 CPU hours worth of tests
- Deciding RTL is reusable when it was not designed to be, has no documentation, and the designers are gone
- Nroff used as a benchmark on a supercomputer

More

- A bi-coastal project where one coast said that async reset can't work and the other said sync reset can't work.
- Not doing scan to save money
- Picking a startup company to go to simply on the volume (number) of shares
- The fact that almost every day to day term has a slightly different definition at every company

And more

- Multi-cast is required
- IPV6 is required and IPV4 is about to be abandoned – this in 1997
- Threads are so important that we need special hardware in the cache and address translation units – this in 1983
- Video conferencing will be ubiquitous
- Madonna will be coming to a warehouse turned into office space for an interview

And more

- The fact that it cost more to buy the parts than what the complete machine sold for was not a problem
 - I suggested saving money by going to a kit form
- Busses that will all be the next major bus
 - FutureBus+, SCI, Infiniband, etc.

What is the thread

- Money wasted!

What to do?

- Don't be a DV engineer – be a hardware engineer
- Try to save money on the project
 - Manpower is the key here
- Suggest project ideas that meet the companies' goals – not yours
- Chop excess features and complexity
 - In design and verification
- Schedule properly -- some projects should die
- Don't fret over off shoring
- Try to remember that your company is not a democracy

What to do?

- Innovate for good reasons
- Drop Useless Innovation
- Think back about things you have been involved with that were a waste of money