Is SystemVerilog Useful for FPGA Design & Verification?

("Burn and Learn" versus "Learn and Burn")

Stuart Sutherland

SystemVerilog Wizard

Sutherland HDL, Inc.

Training engineers to be HDL wizards

www.sutherland-hdl.com

About the Presenter...

- Stuart Sutherland
  - SystemVerilog design and verification consultant
    - Founder and President of Sutherland HDL, Inc.
    - Specializes in providing Verilog/SystemVerilog training
    - Involved in hardware design & verification since 1982
    - Has been using Verilog since 1988
    - Bachelors in Computer Science with Electronic Engineering minor
    - Master's in Education with emphasis in e-learning
  - Involved in IEEE Verilog and SystemVerilog standards since 1993
    - Editor of IEEE 1364 Verilog and IEEE 1800 SystemVerilog Language Reference Manuals (LRMs)
    - Author of multiple books on Verilog and SystemVerilog

You can follow up with the author at stuart@sutherland-hdl.com

Presenting SystemVerilog to FPGA Designers…

- **In 2004 I gave a paper on SystemVerilog to FPGA designers**
  - Nobody cared, nobody listened, no one asked questions

This relief is over the door of one of the entrances to the Yale Law School building

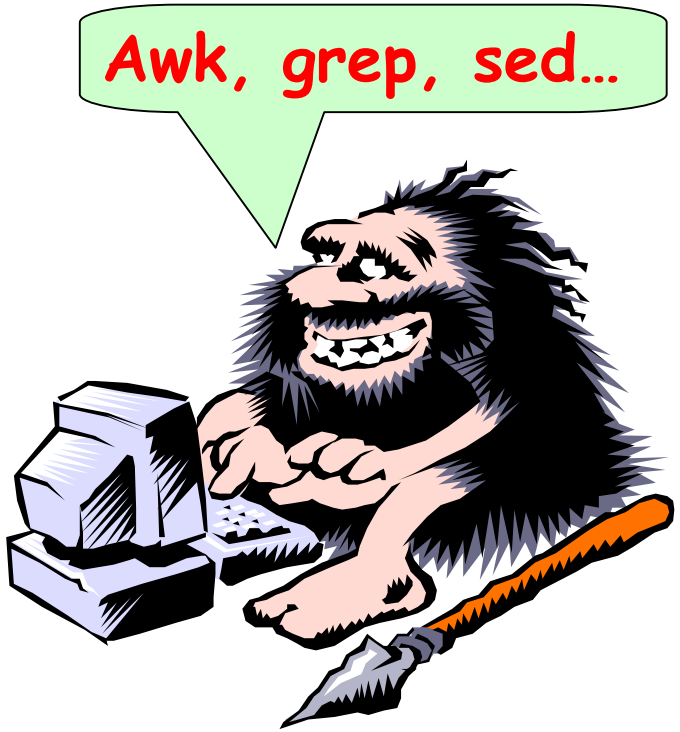
- **Has anything changed after 5 years?**
- **3 months ago, I attended an FPGA conference…**
  - Synopsys had several presentations about SystemVerilog
  - Almost nobody cared, listened, or asked questions

FPGA Design and Verification Then and Now

- FPGA tools have changed since 2004, but…
  - Are FPGA designers still using obsolete languages?

THEN (2004 AD)

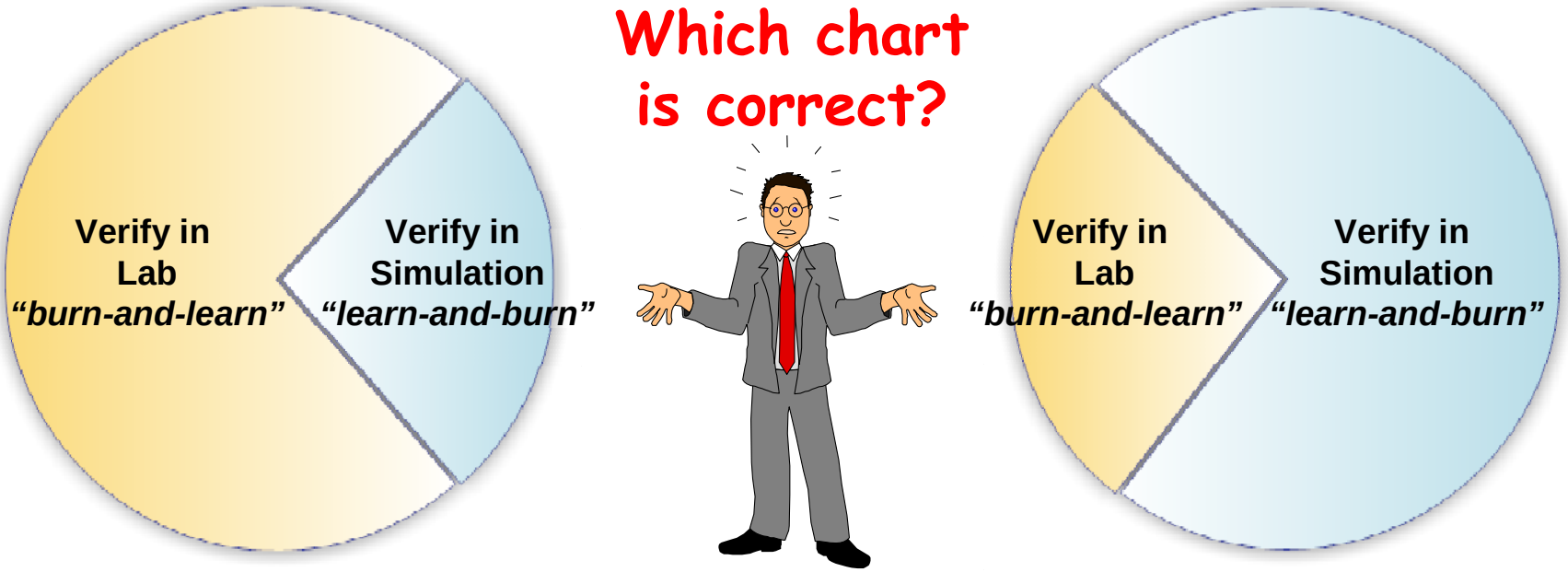
NOW (2009 AD)



Adapted from a cartoon strip, circa 1987

Asking the Right Question

- Do FPGA designers use advanced ASIC methodologies?



- This paper does not answer this question!!!
  - The real question is…
    "Is SystemVerilog useful for FPGA design and Verification"

SUG SAN JOSE 2009

What are the Obstacles?

- The purpose of this paper is to understand the obstacles
  - Are SystemVerilog features useful for FPGA design/verification?
  - Are there deficiencies with tools that limit its usefulness?
  - Are there deficiencies in the language that limit its usefulness?
- What needs to change to enable using SystemVerilog???

Finding the Answers

SUTHERLAND HDL
training engineers to be Verilog and SystemVerilog wizards
www.sutherland-hdl.com

- To determine what FPGA designers think about SystemVerilog:

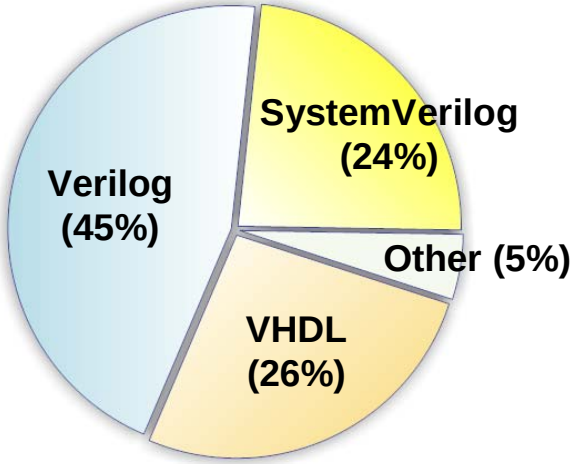
  - A **survey** was sent to companies involved in designing FPGAs
    - More than 35 companies from over 11 countries responded
  - One-on-one **interviews** were conducted with several engineers

- Note: The engineers involved already use VHDL or Verilog
  - Responses are from engineers who understand using HDLs for FPGA design and verification

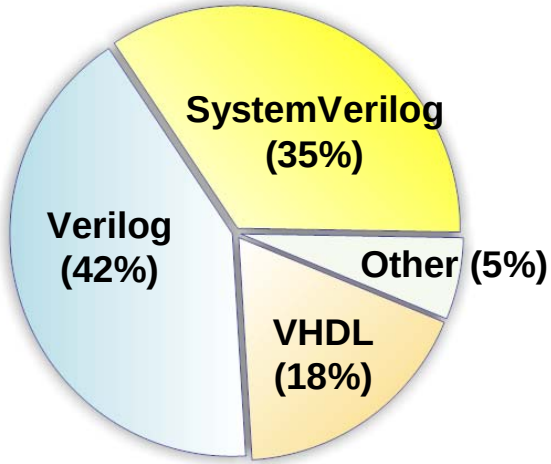
Which HDL Is Used?

- Participants were asked which HDL they use to design and verify FPGAs:

For Synthesis



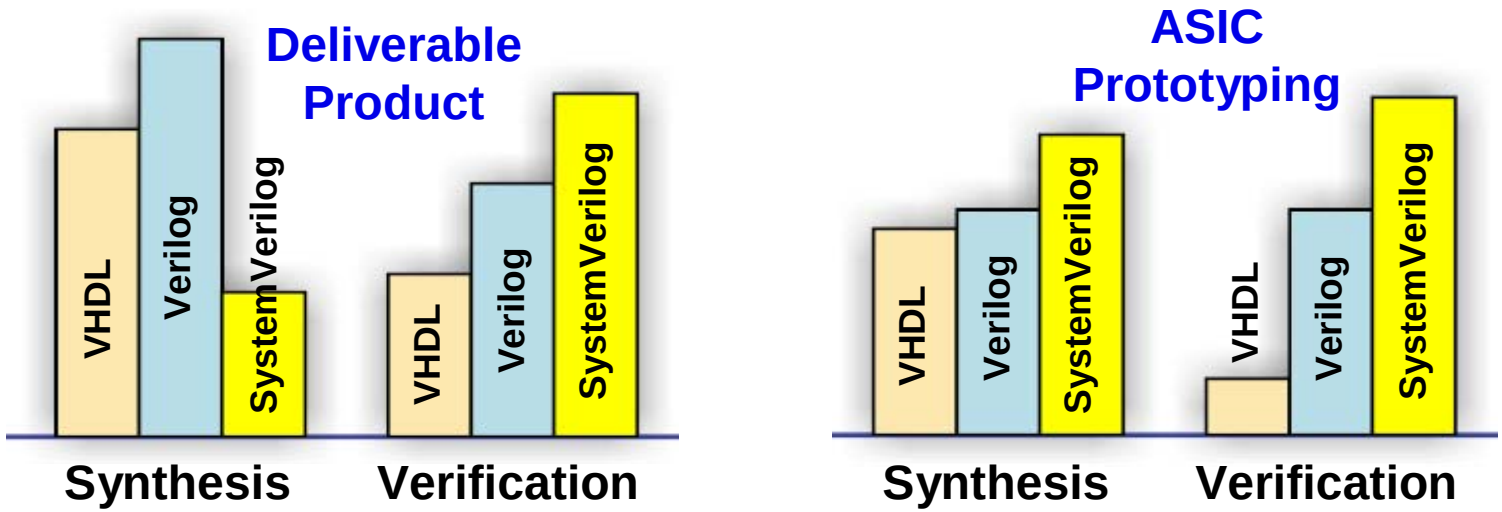
For Verification



- What this data shows…
  - There are companies using SystemVerilog to create FPGAs!
- What this data does *not* show…
  - What types of FPGAs are being created using SystemVerilog (e.g. end products versus ASIC prototyping)

HDL versus FPGA Usage

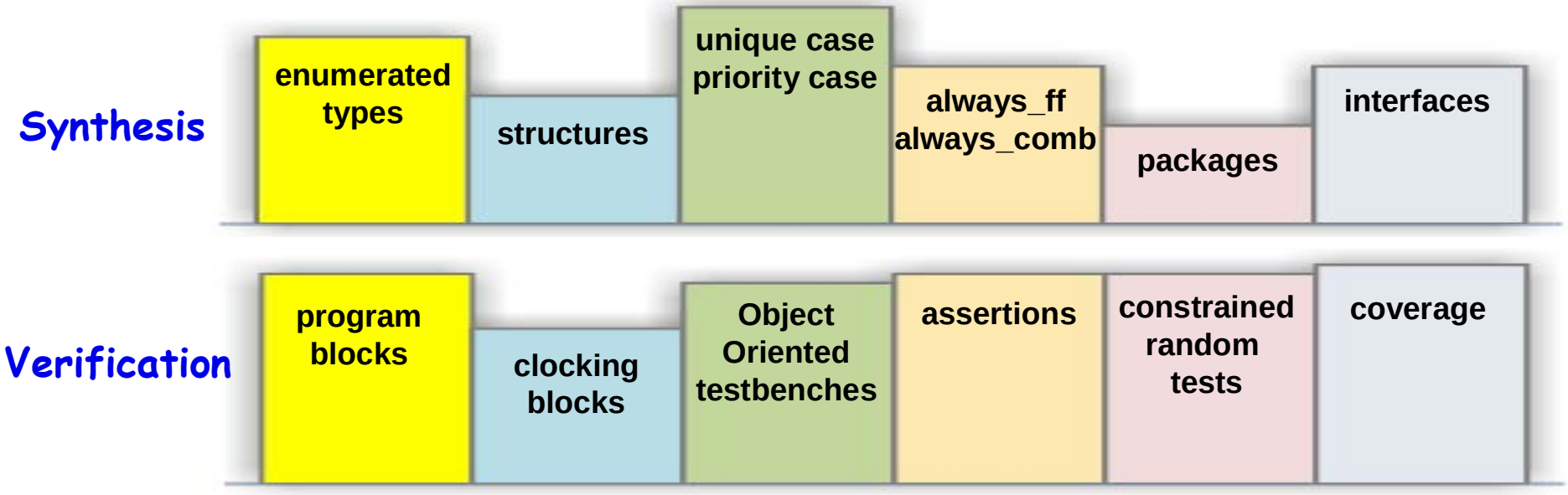
- SystemVerilog cross coverage was used to determine which HDLs were most often used for which types of FPGA
  - (Not really – the correlation of data was done by hand)



- This data indicates that:
  - SystemVerilog is used more to verify FPGAs than for synthesis
  - SystemVerilog is used more for prototyping than for end products

Features of SystemVerilog Being Used for FPGAs

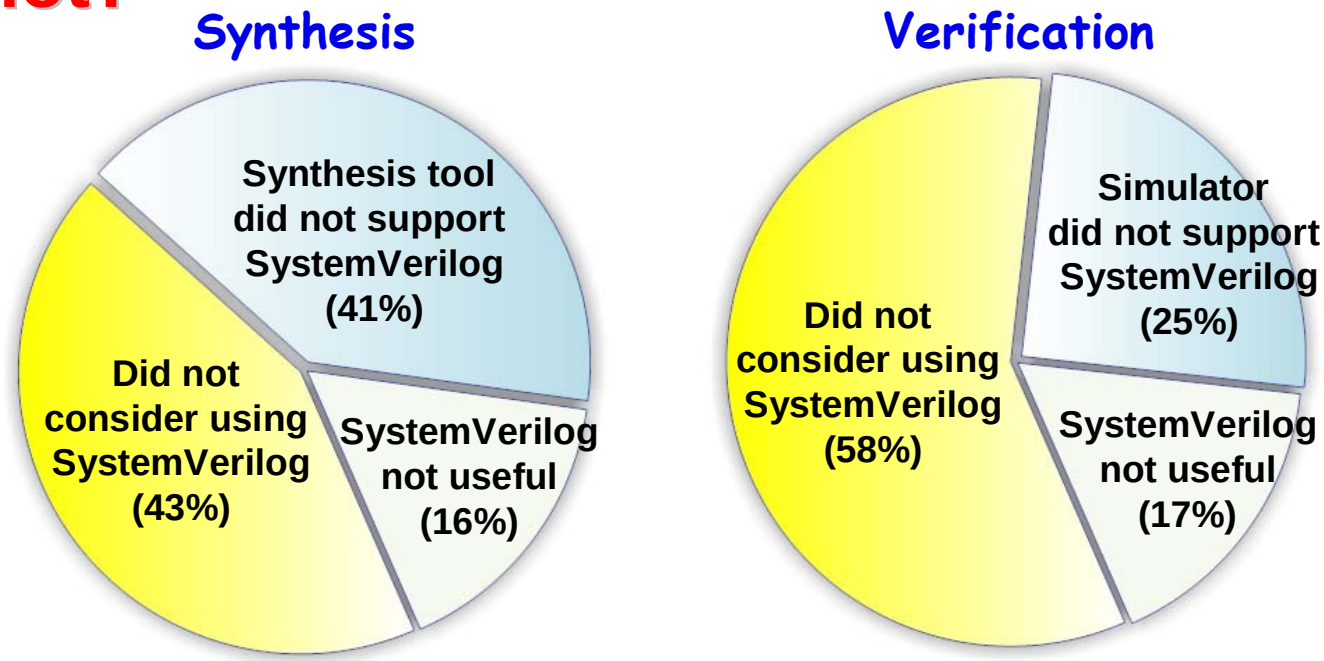
- Respondents who *do use* SystemVerilog were asked what constructs they use:



- This data indicates that:
  - Engineers who are using SystemVerilog are taking advantage of all aspects of the language

Reasons SystemVerilog Is Not Being Used

- Respondents who *do not use* SystemVerilog were asked, **"Why not?"**



- The additional comments made by engineers explain the obstacles for why SystemVerilog was not used

Survey and Interview Comments

- Several hundred comments provide the answers as to what obstacles might be preventing the usage of SystemVerilog
  - The comments were analyzed to identify specific themes
    - Five themes appeared frequently
      - Presented on the next few slides, in order of least frequently to most frequently occurring theme

Synthesis Support

SUTHERLAND HDL
training engineers to be Verilog and SystemVerilog wizards
www.sutherland-hdl.com

- Nearly every engineer who had tried using SystemVerilog for synthesis commented that:
  - Synthesis compilers from FPGA vendors (e.g. Xilinx, Altera, Actel) did not support SystemVerilog at all

> **From an engineer designing FPGAs for consumer products:**
>
> "[We need] better support in [our] vendor's synthesis tool ([FPGA vendor name omitted] is noteworthy in not supporting it at all in their synthesis tool)."

  - Synthesis compilers from EDA vendors (e.g. Synopsys, Synplicity) had very limited support for SystemVerilog

> **From an engineers using FPGAs to prototype ASICs (paraphrased from interview):**
>
> In order to emulate the ASIC in FPGAs, we have to make substantial changes to the RTL code to manually re-write SystemVerilog constructs as Verilog.

- Are these statements valid?
  - That question is addressed later in this presentation

Other Common Themes

- Many of the comments from participants indicated a lack of awareness of SystemVerilog capabilities

> "Seems like it [SystemVerilog] is just as useful as Verilog since it is a superset of Verilog....We can 'get by' without learning something new."

> "SV inherits too much of the low level nature of Verilog and does not allow as high-level design as VHDL."

- Cost came up a number of times

> "Support [for SystemVerilog] would have to come through the simulation tools we now own. And don't…bump up my maintenance costs to cover that capability."

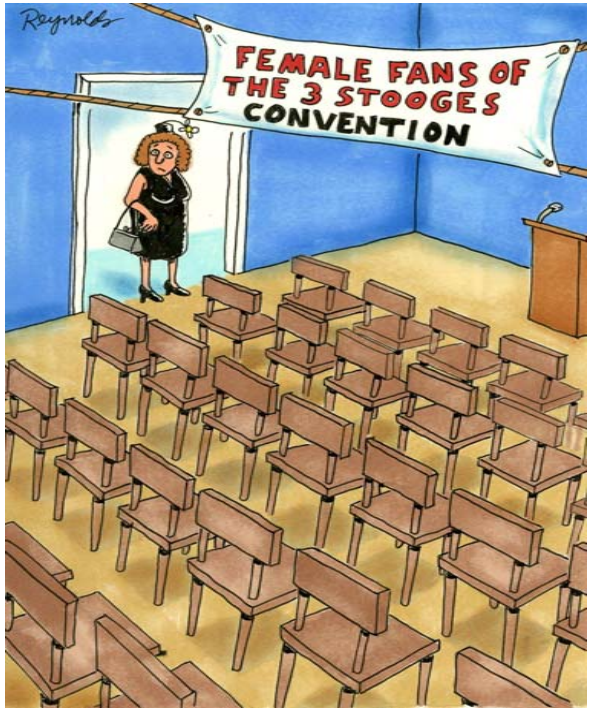
- Learning SystemVerilog was a concern

> "We considered SystemVerilog, but did not have adequate training and did not want a disruption."

Fact or Fiction? FPGAs Don't Need SystemVerilog

- Some survey respondents indicated that SystemVerilog is ***too complex*** for FPGA design and verification

> "Some of our Verilog users have been reluctant to move away from Verilog 95."



- **Is this a valid concern?**
  - FPGAs can be as complex as ASICs
  - SystemVerilog can be adopted in phases
    - Can benefit from many features without learning OO programming or other advanced features
- **Recommendations:**
  - Synopsys needs to improve the marketing of SystemVerilog for FPGA design and verification

Fact or Fiction? FPGA Synthesis Doesn't Support SystemVerilog

- Almost every survey and interview respondent said that synthesis is the biggest obstacle to using SystemVerilog
  - Yet almost all vendors claim to support SystemVerilog synthesis
    - **Who is right?**
- Two reasons engineers perceive synthesis is an obstacle are:

1. A older version of the synthesis compiler was evaluated

> "Our last assessment [of SystemVerilog synthesis] was done 2-1/2 years ago. We have not re-assessed since that time."

2. ASIC and FPGA synthesis tools support different SV subsets

> "The [RTL code] must be done differently for each…synthesis compiler…because the…support for SystemVerilog is quite different for each compiler."

- **Recommendation for Synopsys:**
  - DC & Synplify-Pro need to support *the same* SV constructs!!!

Fact or Fiction? SystemVerilog Tools Cost Too Much

- Some FPGA designers stated that they need low-cost tools

> "[SystemVerilog needs] support from a free open source simulator. I use [simulator name omitted], which presently only supports Verilog 95."

> "I can't see many teams taking up [SystemVerilog for verification]….A simulation seat is easily more expensive than all other…FPGA design software combined."

- Is this a valid concern?
  - Historically, FPGA tools have cost less than ASIC tools
  - Advanced verification tools are expensive to create & maintain
- **Recommendations:**
  - Project managers needs to recognize that …
    The more complex the FPGA, the more expensive the tools
  - EDA/FPGA vendors can reduce cost with "lite" versions of tools

Fact or Fiction? SystemVerilog Is Too Difficult to Learn

- Several comments indicated that the cost or time required to learn SystemVerilog is an obstacle to adopting SystemVerilog

> "We considered SystemVerilog, but did not have adequate training and did not want a disruption."

- Is this a valid concern?
  - Mastering every aspect of SystemVerilog requires a lot of learning
  - Do not need to be an expert OO programmer to gain huge benefits from SystemVerilog
- **Recommendations:**
  - Engineers can learn SystemVerilog incrementally
    - E.g.: First learn synthesis constructs or assertions
    - As needs increase, learn more advanced language features

Conclusions

- SystemVerilog *is* useful for FPGA design and verification
  - Many FPGA engineers are already using SystemVerilog
  - ASIC prototyping is where SystemVerilog is most often used
- There are valid barriers that have slowed the adoption of SV
  - Synthesis/simulation support was an obstacle, but not any more
  - Lack of awareness about SystemVerilog is the real obstacle!
- The paper conclusions are based on grounded data
  - Survey and interviews were used to determine the facts
    - 66 engineers at more than 35 companies in 11+ countries
    - Quantitative and qualitative data collection and analysis
- The paper includes all comments received
  - Both positive and negative comments – only names are removed

Thank You!

- FPGA tools have changed since 2004, AND…
  - You're no longer using obsolete languages, right?

THEN (2004 AD)

NOW (2009 AD)