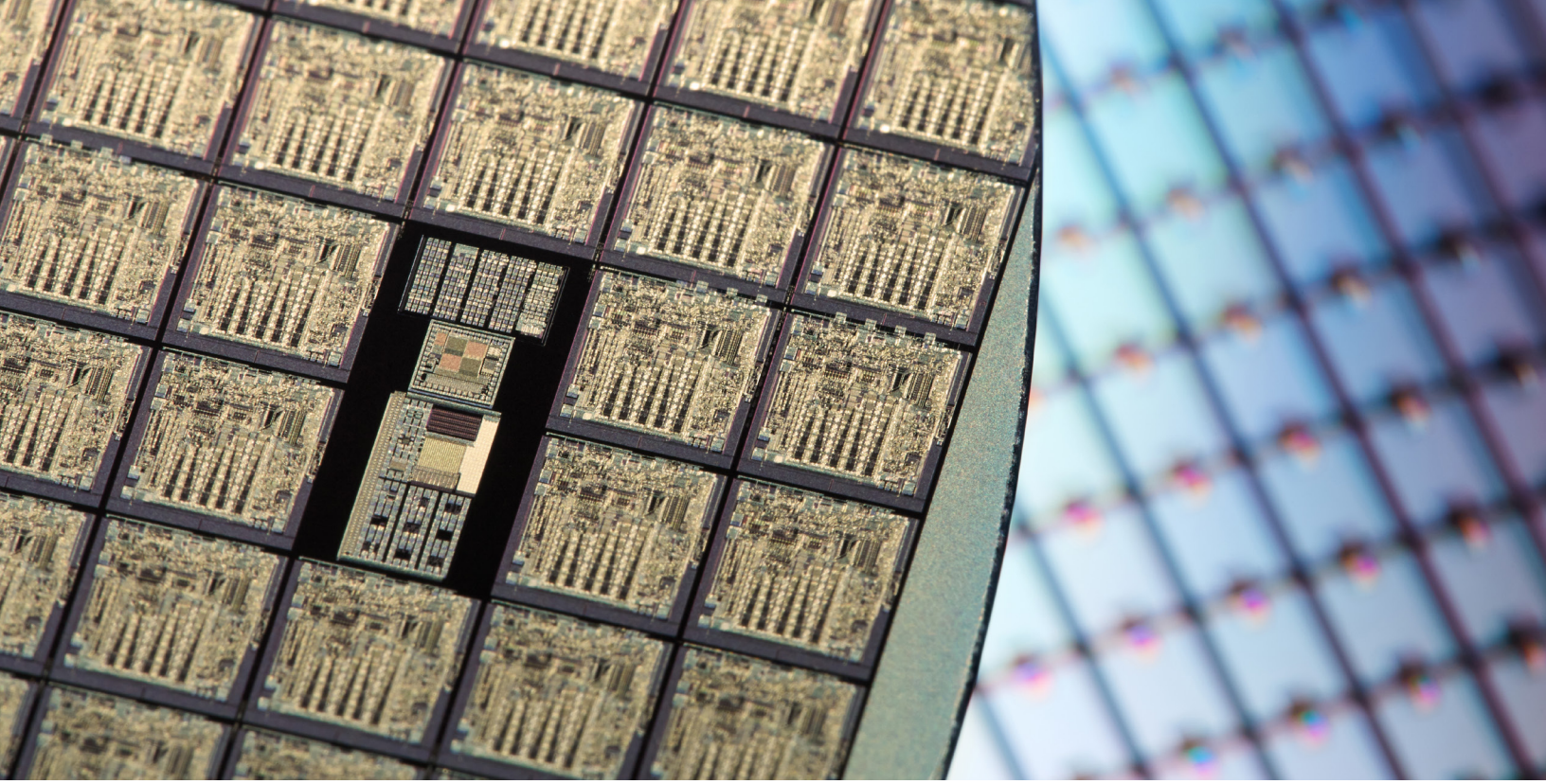

DIGITAL INDUSTRIES SOFTWARE

2022 Wilson Research Group FPGA functional verification trends

Executive summary
This report presents those results from the 2022 Wilson Research Group Functional Verification Study that were focused on the Field-Programmable Gate Array (FPGA) segment. The findings from this study provide invaluable insight into the state of today's FPGA market in terms of both design and verification trends.

Harry D. Foster
Chief Scientist Verification

SIEMENS

Contents

I. Introduction

This report presents field-programmable gate array (FPGA) functional verification trends based on the 2022 Wilson Research Group functional verification study. While multiple studies focused on general IC/ASIC functional verification trends have been published,[1, 2, 3, 4, 5] to our knowledge our 2018 study was the first to specifically focus on FPGA functional verification trends.[6, 7] Our 2022 study builds on our previous studies by providing the latest industry trends.

A. The global FPGA semiconductor market

IBS estimates that the global semiconductor market was valued at $547 billion in 2021. While the market is expected to decline to $545 billion in 2023, it is optimistically projected to grow to a value of $635 billion by 2025. The FPGA portion of the semiconductor market is valued at about $5.3 billion.[8] The FPGA semiconductor market is expected to reach a value of $8.1 billion by 2025. The growth in this market is driven by new and expanding end-user applications related to data center computing, networking, and storage, as well as communications.

Historically, FPGAs have offered two primary advantages over ASICs. First, due to their low NRE, FPGAs are generally more cost effective than IC/ASICs for low volume production. Second, FPGAs' rapid prototyping capabilities and flexibility can reduce the development schedule since a majority of the verification and validation cycles have traditionally been performed in the lab. More recently, FPGAs offer advantages related to performance for certain accelerated applications by exploiting hardware parallelism (e.g., AI Neural Networks).

The IC/ASIC market in the mid- to late-2000s timeframe underwent growing pains to address increased verification complexity. Similarly, we find today's FPGA market is forced to address growing verification complexity. With the increased capacity and capability of today's complex FPGAs and the emergence of high-performance SoC programmable FPGAs (e.g., Xilinx Zynq® UltraSCALE+, Intel® Stratix®, and Microsemi SmartFusion®), traditional lab-based approaches to FPGA verification and validation are becoming less effective. In Section II, we quantify the ineffectiveness of today's FPGA verification processes in terms of nontrivial bug escapes into production.

B. Study background

The study results presented in this report are a continuation of a series of industry studies on functional verification. This series includes the previously published 2012, 2014, 2016, 2018, and 2020 Wilson Research Group functional verification studies. Each of these studies was modeled after the 2002 and 2004 Collett International Research, Inc. studies.[1, 2]

For the purpose of our study, a randomized sampling frame was constructed from multiple acquired industry lists. This enabled us to cover all regions of the world and all relevant electronics industry market segments. It is important to note that we did not include our own account team's customer list in the sampling frame. This was done in a deliberate attempt to prevent vendor bias in the final results. While we architected the study in terms of questions and then compiled and analyzed the results, we commissioned Wilson Research Group to execute our study. After data cleaning the results to remove inconsistent, incomplete, or random responses, the final sample size consisted of 980 eligible participants (i.e., n=980).

Fig. 1 shows the percentage of 2022 study participants (i.e., design projects) by targeted implementation for both IC/ASIC and FPGA projects.

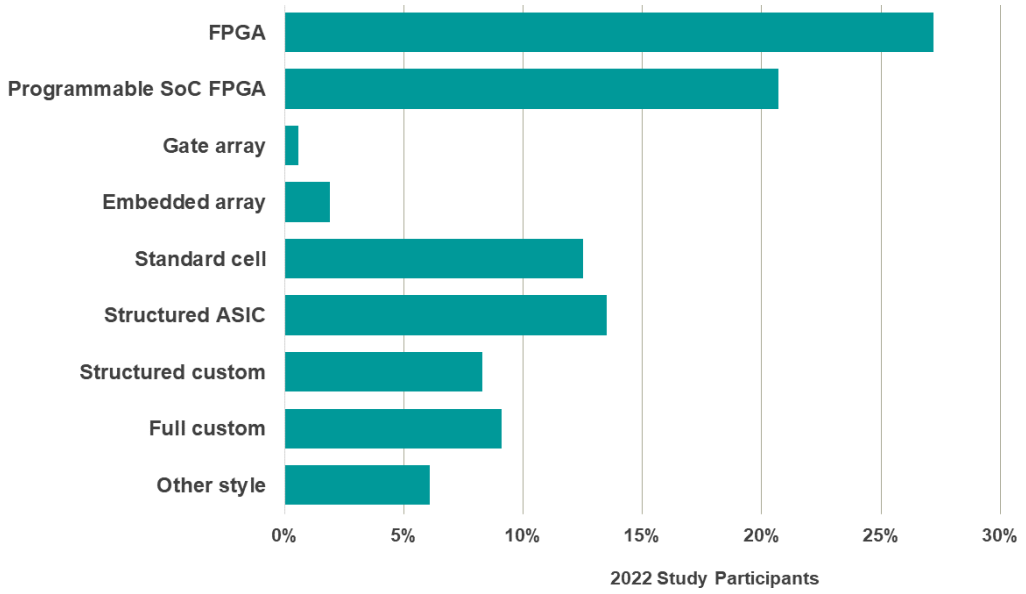

Fig. 1. Study participants by targeted implementation.

C. Study confidence interval

Since all survey-based studies are subject to sampling errors, we attempt to quantify this error in probabilistic terms by calculating a confidence interval. For our study, we determined the overall margin of error to be ±3.7% using a 95% confidence interval. In other words, this confidence interval tells us that if we were to take repeated samples from a population, 95% of the samples would fall inside our margin of error of ±3.7%, and only 5% of the samples would fall outside.

D. Study bias

When architecting a study, three main bias concerns must be addressed to ensure valid results: sample validity bias, nonresponse bias, and stakeholder bias. We have adopted multiple techniques to minimize these biases. However, the study demographics, as shown below in fig. 2, saw a 58 percentage points decline in participation from India but an increase in participation from Europe and North America. This raises some interesting questions. For example, was the decline due to more aggressive spam filters than in previous years, preventing the invitation from reaching potential study participants? Regardless, the shift in balance in the study demographics can introduce potential non-response biases in the findings that need to be considered. For example, regional shifts in participation can influence the findings for design and verification language adoption trends. Regional bias from year to year can contribute to lumpiness in trends across multiple years. Hence, any potential biases of concern in the data will be highlighted when appropriate.

E. Report organization

The remainder of this report is organized as follows. In Section II, we discuss the study findings related to FPGA verification effectiveness. In Section III, we discuss trends in terms of FPGA project resources. In Section IV, we discuss the study results specifically related to various aspects of FPGA design to illustrate growing complexity. In Section V, we examine FPGA verification technology adoption trends. In addition, this section presents adoption trends for various design and verification language and methodology standards. Finally, in Section VI, we draw some conclusions and discuss various aspects of this year's study.

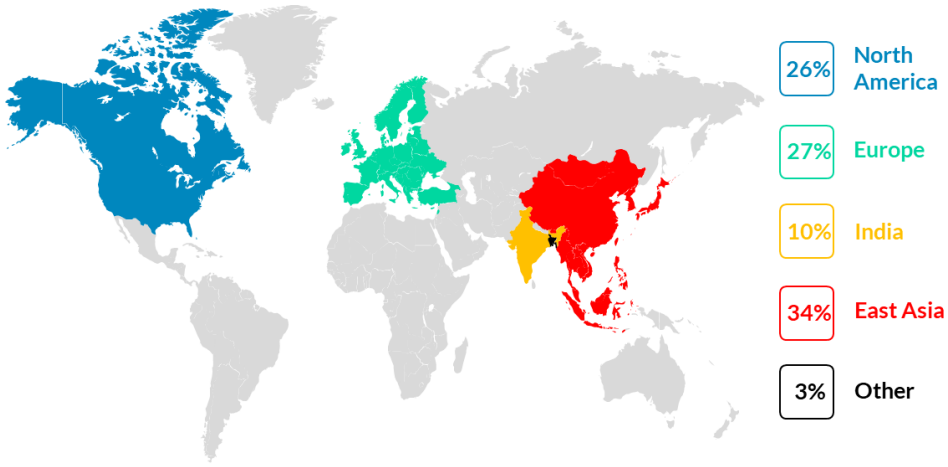

Fig. 2. 2022 study demographics.

II. FPGA verification effectiveness

A. Non-trivial bug escapes

IC/ASIC projects have often used the metric "number of required spins before production" as a benchmark to assess a project's verification effectiveness. Historically, about 30% of IC/ASIC projects achieved first silicon success, and most successful designs were productized on the second silicon spin. Unfortunately, FPGA projects have no equivalent metric. As an alternative to IC/ASIC spins, our study asked the FPGA participants "how many non-trivial bugs escaped into production?" The results, shown below in fig. 3, are somewhat disturbing. In 2022, only 17% of all FPGA projects were able to achieve no bug escapes into production, which is worse than IC/ASIC in terms of first silicon success.

Furthermore, for some market segments, the cost of field repair can be significant. For example, in the aerospace market, once a cover has been removed on a system to upgrade the FPGA, the entire system needs to be revalidated.

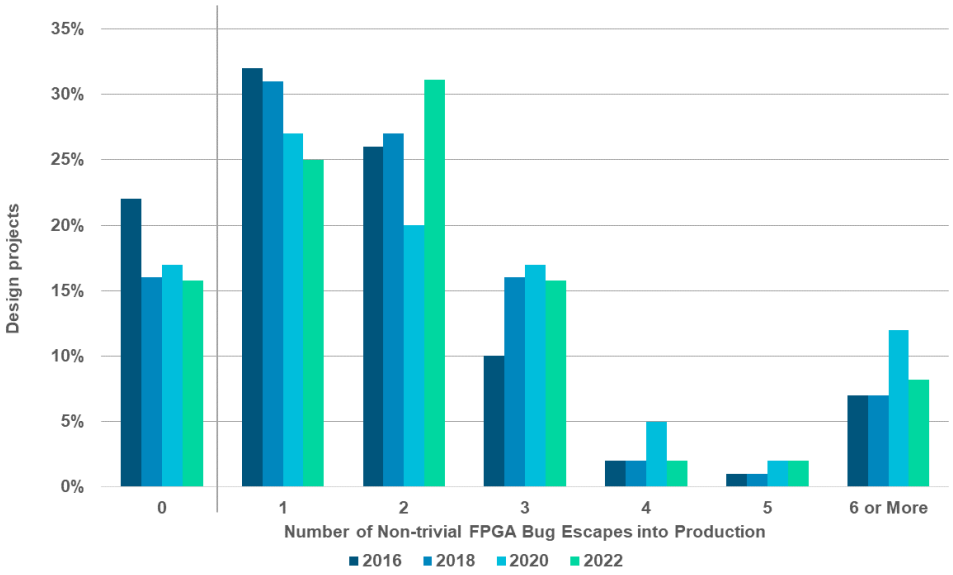

Fig. 3. FPGA non-trivial bug escapes into production.

B. Types of flaws resulting respins

Fig. 4 shows various categories of design flaws contributing to FPGA non-trivial bug escapes. "Logic or functional flaws" remain the leading cause of bugs. New flaws tracked in the 2022 study are associated with safety (8%) and security (6%) features. Obviously multiple flaws can contribute to bug escapes, which is the reason the total percentage of flaws sums to more than 100%.

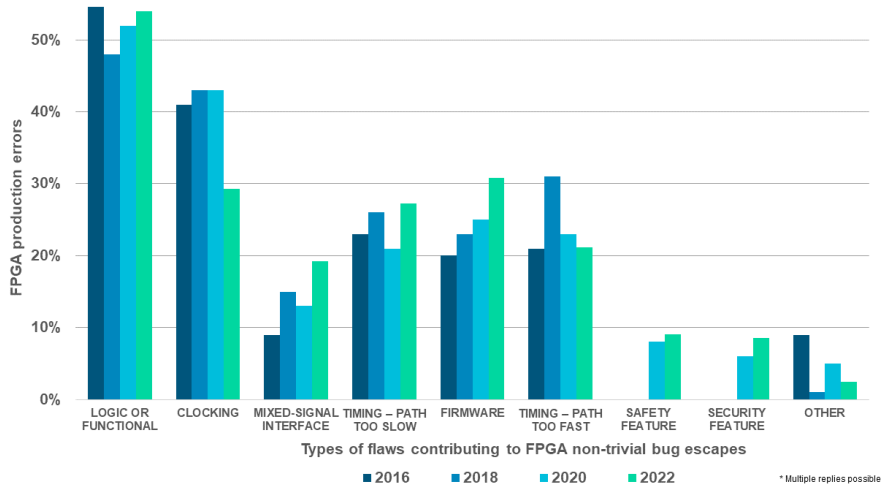

Fig. 4. Types of flaws contributing to respins.

Fig. 5 below demonstrates the root cause of logical or functional flaws by various categories. Historically the data suggest design errors are the leading cause of functional flaws, yet the situation might be improving as FPGA projects mature their verification processes to address complexity. In addition, problems associated with changing, incorrect, or incomplete specifications are a common concern often voiced by verification engineers and project managers, as is clearly identified in the study.

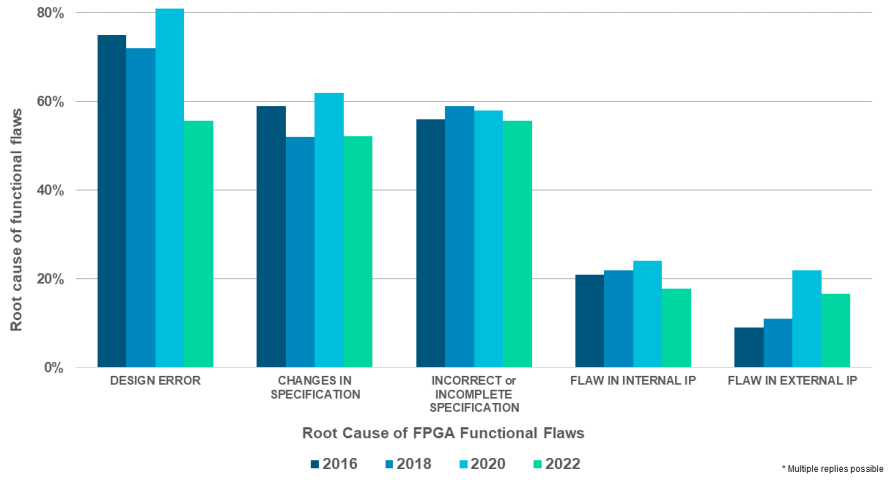

Fig. 5. Root cause of functional flaws.

C. Design completion compared to original schedule

In addition to bug escape metrics that we used to determine an FPGA project's effectiveness, another metric we tracked was project completion compared to the original schedule, as shown in fig. 6. Here we found that 70% of FPGA projects were behind schedule. One indication of growing design and verification complexity is reflected in the increasing number of FPGA projects missing schedule by more than 50% in 2022.

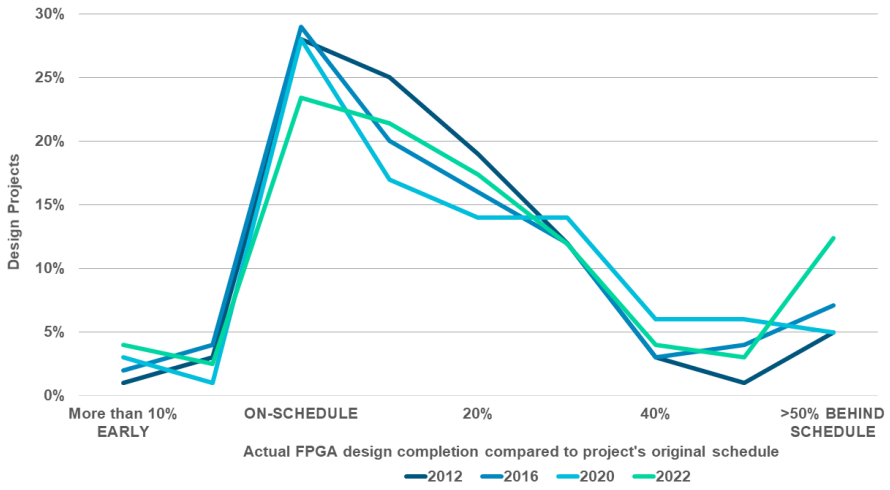

Fig. 6. FPGA actual completion compared to original schedule.

III. FPGA verification effort

In this section, we discuss trends in terms of FPGA project time and resources.

A. Percentage of project time spent in verification

Fig. 7 shows the percentage of total FPGA project time spent in verification. You can see two extremes in this graph. In general, projects that spend very little time in verification are typically designs with a good deal of existing, pre-verified design IP, which is integrated to create a new product. On the other extreme, projects that spend a significant amount of time in verification often have a high percentage of newly developed design IP that must be verified.

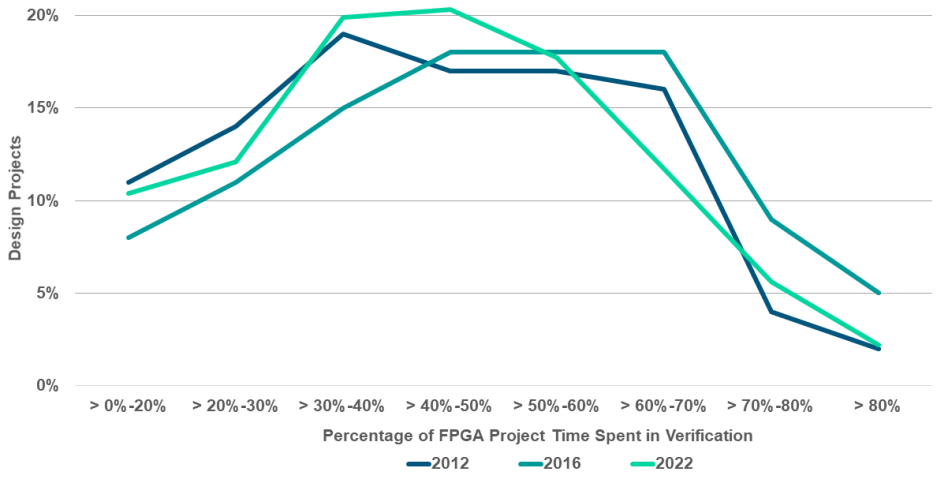

Fig. 7. Percentage of FPGA project time spent in verification.

B. Mean peak number of engineers

Perhaps two of the biggest challenges today are controlling cost and engineering headcount, which means identifying FPGA design and verification solutions that increase productivity. To illustrate the need for productivity improvement, we discuss the trend in terms of increasing engineering headcount.
Fig. 8 below shows the mean peak number of FPGA engineers working on a project.

While, on average, the demand for FPGA design engineers grew at about a 1.6% CAGR between 2012 and 2022, the demand for FPGA verification engineers grew at a 3.3% CAGR. It is worth noting that during the period 2007 through 2014, the IC/ASIC market went through similar growth demands related to verification engineers to address growing verification complexity.[3]

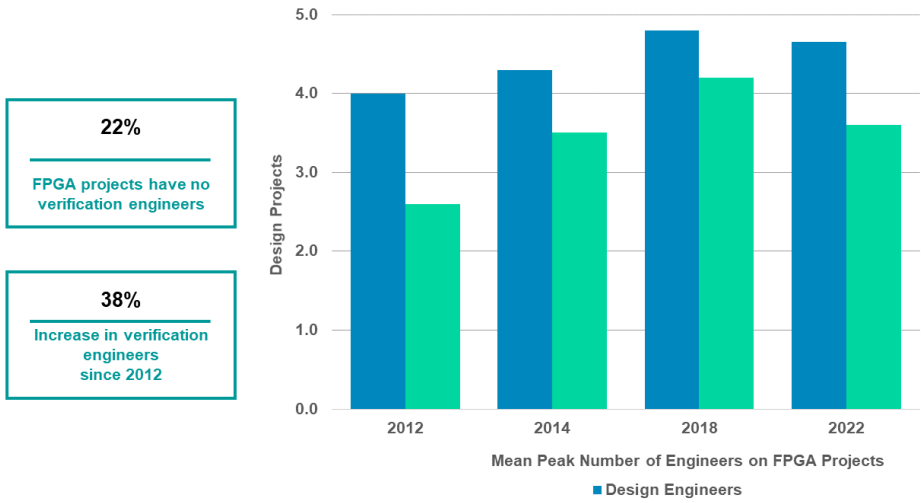

Fig. 8. Mean peak number of FPGA engineers.

But verification engineers are not the only project stakeholders involved in the verification process. Design engineers spend a significant amount of their time in verification too.

In 2022, design engineers spent on average 58% of their time involved in design activities and 42% of their time in verification. However, when compared to 2014 and 2018, the data indicate a trend showing that FPGA design engineers are now spending less time involved in verification tasks. There are two reasons for this trend. First, many FPGA projects have added verification engineers to their teams, which means design engineers can focus most of their effort on design. Second, in general, there has been increased adoption of larger, more complex FPGAs, which has increased the design engineer's workload.

Fig. 9 shows where verification engineers spend their time (on average). We do not show trends here since there were no statistically significant changes in the FPGA results during the period 2014 through 2022.

Our study found that FPGA verification engineers spend more of their time debugging than on any other activity. From a management perspective, this can be a significant challenge when planning future projects' effort and schedule based on previous projects' data since debugging is unpredictable and varies significantly between projects.

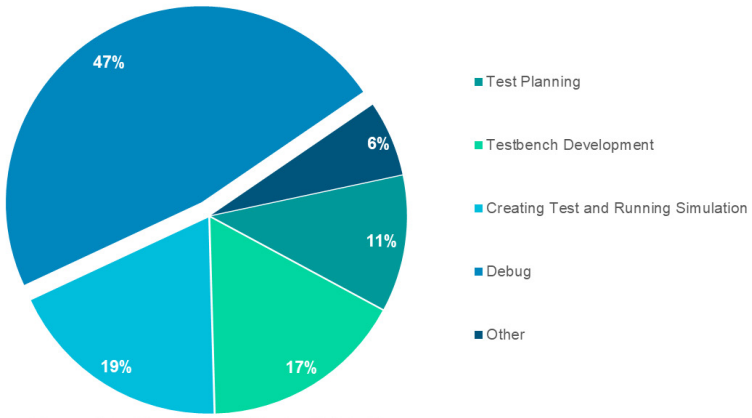

Fig. 9. Where FPGA verification engineers spend their time.

IV. FPGA design trends

One industry driver that has had a substantial impact on FPGA design and verification complexity is the emergence of new layers of design requirements (beyond basic functionality), which did not exist years ago for example, clocking requirements, security requirements, safety requirements, and requirements associated with hardware-software interactions. In this section, we examine trends related to various aspects of growing FPGA design complexity.

A. Embedded processor cores

What has changed significantly in FPGA designs in the last 15 years is the movement toward SoC-class designs. For example, our study found that 65% of all projects targeted their design at an FPGA containing one or more embedded processors, as shown in fig. 10. Furthermore, 43% of all FPGA designs today contain two or more embedded processors, while 4% include eight or more embedded processors.

SoC-class designs add a new layer of verification complexity—due to an increased number of design requirements—to the verification process[9, 10] that did not exist

with traditional non SoC-class designs. For example, SoC-class designs often require verification of hardware and software interactions, new coherency architectures, and complex network-on-a-chip interconnects.

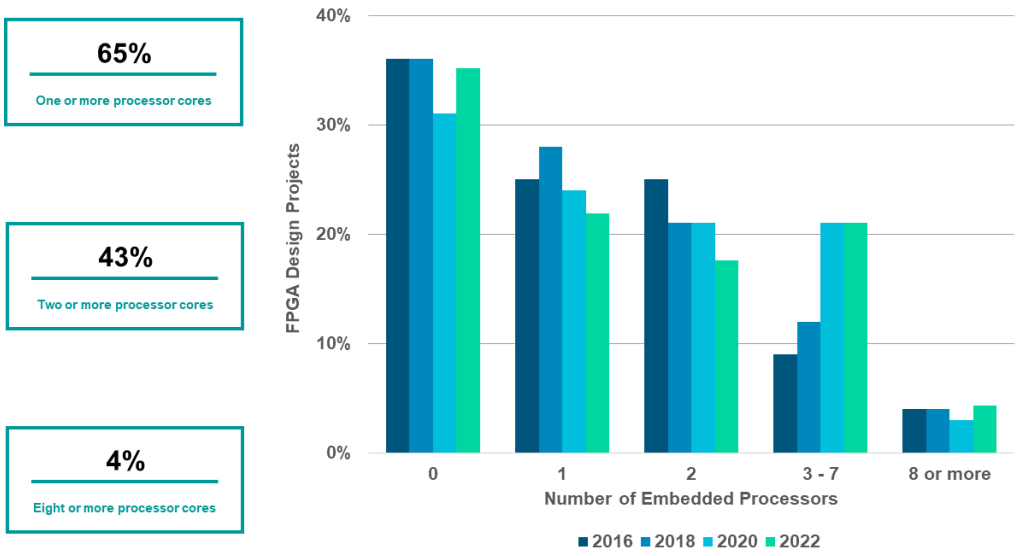

Fig. 10. Number of embedded processor cores.

Our 2022 study also tracked the percentage of FPGA projects that have incorporated a RISC-V processor in their design, which was 22%. In addition, we tracked the percentage of FPGA projects that have incorporated some type of AI accelerator processor (e.g., TPU, etc.), which was 23% in our current study.

B. Asynchronous clock domains

In fig. 11 below we see that 92% of designs being implemented as FPGAs contain two or more asynchronous clock domains, where the average number of clock domains is between 3 to 4.

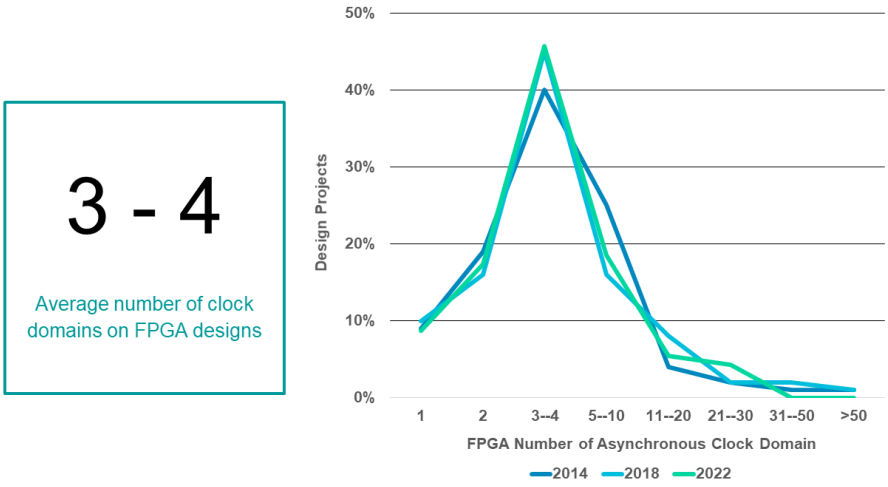

Fig. 11. Number of asynchronous clock domains.

Verifying requirements associated with multiple asynchronous clock domains has increased both the verification workload and complexity. For example, a class of metastability bugs cannot be demonstrated on an RTL model using simulation. To simulate these issues requires a gate-level model with timing, which is often not available until later stages in the design flow. Furthermore, clocking metastability bugs are generally difficult to reproduce and find in the lab. To address these issues, static clock-domain crossing (CDC) verification tools have emerged and are being adopted to help identify clock domain issues directly on an RTL model at earlier stages in the design flow.

C. Security features

Today we find that 49% of FPGA projects add security features to their designs. Examples of security features include security assurance hardware modules (e.g., a security controller) that are designed to safely hold sensitive data, such as encryption keys, digital right management (DRM) keys, passwords, and biometrics reference data. These security features add requirements and complexity to the verification process.

D. Safety-critical design

Another example of increasing requirements contributing to complexity relates to safety-critical designs. In 2022, we found that 42% of all FPGA projects were working under one of multiple safety-critical development process standards or guidelines.

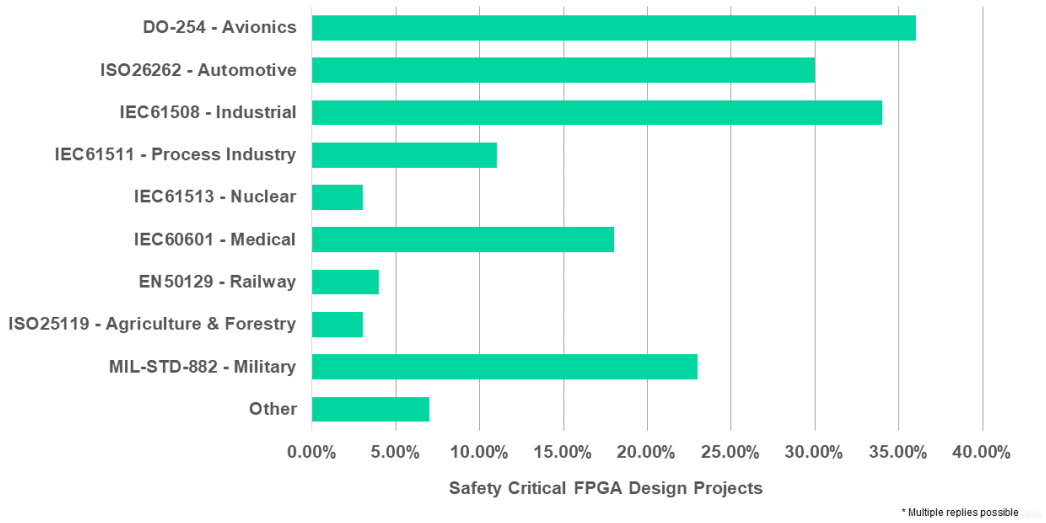

Fig. 12. Safety-critical development standard used on FPGA projects.

For those projects working under a safety-critical development process standard or guideline, in fig. 12 we show the specific breakdown for the various standards. Note that some projects are required to work under multiple safety standards or guidelines (e.g., IEC61508 and IEC61511), which is why the percentage of adoption sums to more than 100%.

Fig. 13 shows the percentage of overall project time spent in functional safety activities. The median percentage of time is between 25% and 50%. Fig. 14 below shows the biggest challenges associated with functional safety as reported by this year's study participants.

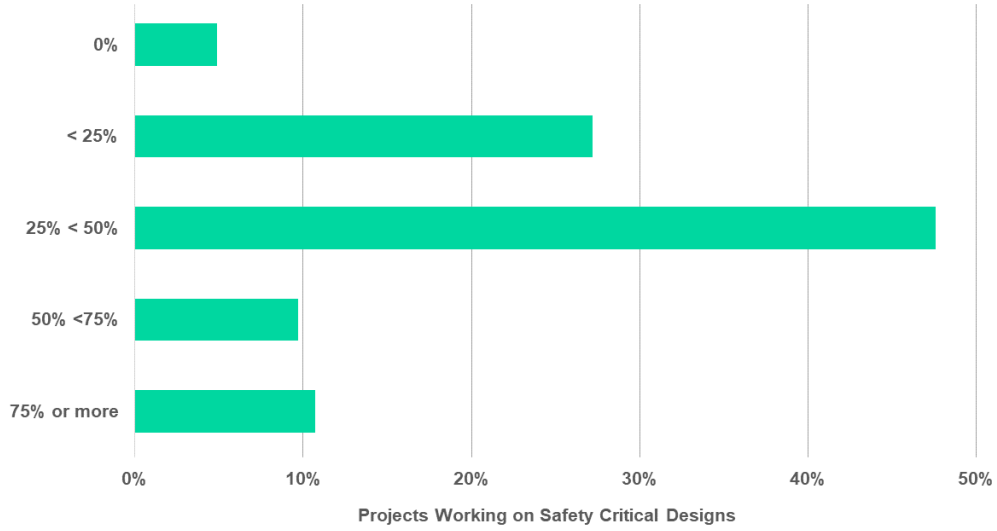

Fig. 13. Percentage of overall project time spent on functional safety.

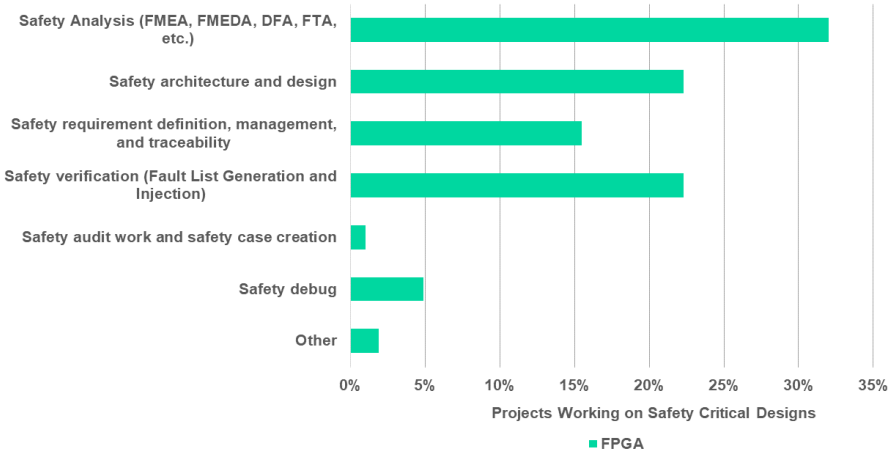

Fig. 14. Biggest functional safety project challenge

V. FPGA verification adoption trends

To address growing verification complexity, we found that many FPGA projects are starting to mature their pre-lab functional verification processes. In this section, we present FPGA trends related to the adoption of various verification techniques, which are fairly standard practice today on most IC/ASIC projects.

A. Verification languages and methodolgy adoption trends

Fig. 15 below shows the adoption trends for languages used to build testbenches. Also, we needed to consider potential 2022 regional biases associated with the findings in fig. 14, as discussed in the Introduction Section D pertaining to India; although the findings suggest there is not much impact on this year's results since they are in line with what we expected.

Historically, VHDL was the predominant language used for FPGA testbench development, but we have recently seen increasing interest in SystemVerilog adoption. Today, it is not unusual to find that the RTL design was created using VHDL, while the testbench was created using other languages, such as SystemVerilog.

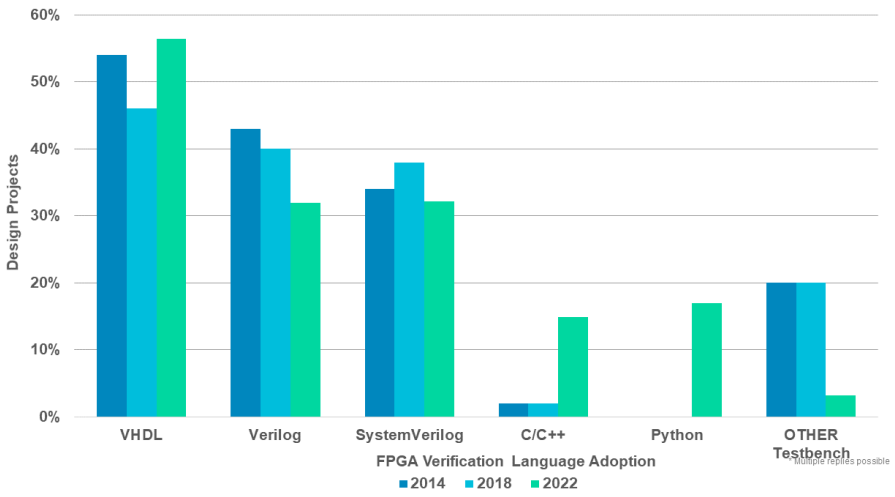

Fig. 15. FPGA project verification language adoption.

In the 2022 data, note the decline in Verilog adoption. Technically, the IEEE Std. 1364 Verilog is no longer a standard, and all its capability was incorporated in the IEEE Std. 1800 SystemVerilog standard. Also note that we show the adoption levels for the emerging Accellera Portable Test and Stimulus Standard (PSS). Finally, we recently started tracking the adoption of Python for testbench development as shown in this graph.

The adoption trends for various methodology standards are shown in fig. 16, and we found that the Accellera UVM is currently the predominant standard adopted to create FPGA testbenches. In 2018, we first started tracking the Open Source VHDL Verification Methodology™ (OSVVM) and the Universal VHDL Verification Methodology (UVVM), and in 2022 we were starting to trends. In addition, note that we recently started tracking Python-based methodologies, such as *cocotb*.

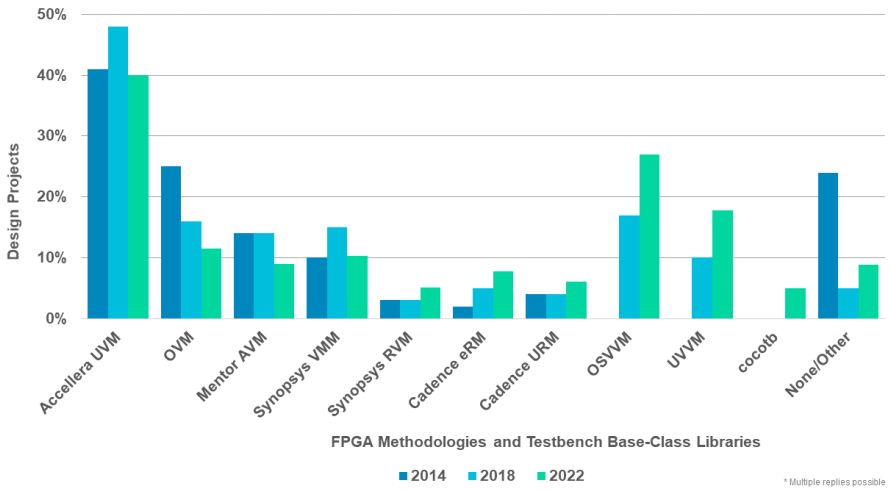

Fig. 16. FPGA project methodology and base-class libraries adoption.

Finally, FPGA project adoption trends for various assertion language standards are shown in fig. 17. SystemVerilog Assertions (SVA) is the predominant assertion language in use today. Like languages used to build testbenches, it is not unusual to find FPGA projects create their RTL in VHDL and then create their assertions using SVA.

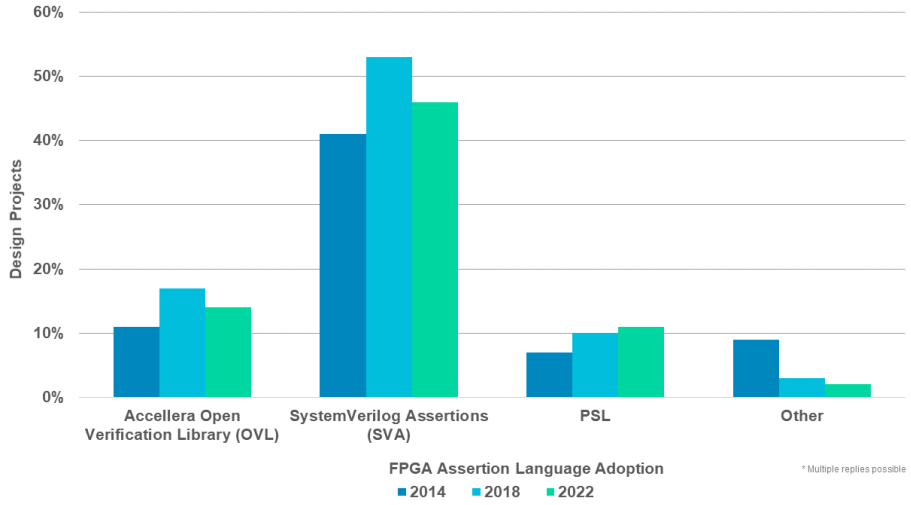

Fig. 17. FPGA project assertion lanauge adption.

B. Verification technology adoption trends

The adoption trends for formal property checking (e.g., model checking) and automatic formal applications are shown in fig. 18. We found that the adoption of both formal property checking and automatic formal applications on FPGA projects is growing at a 7.5% CAGR. Historically, the formal property checking process has required specialized skills and expertise. However, the recent emergence of automatic formal applications provides narrowly focused solutions that do not require specialized skills for adoption. In general, formal solutions (i.e., formal property checking combined with automatic formal applications) is one of the fastest growing segments in functional verification in terms of project adoption.

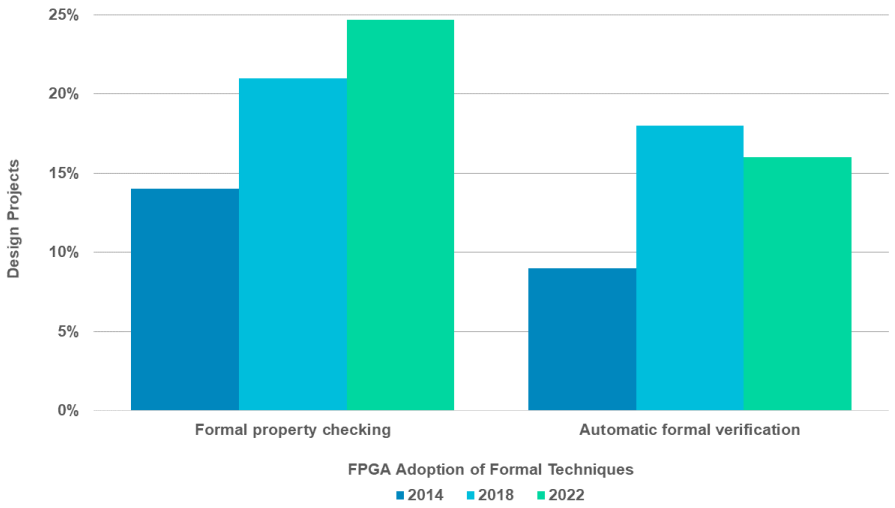

Fig. 18. FPGA project formal technology adoption trends.

Fig. 19 shows the FPGA project adoption trends for various simulation-based techniques from 2012 through 2022, which include code coverage, functional coverage, assertions, and constrained-random simulation. The data suggest that the FPGA market has slowly matured its functional verification processes during this ten-year span.

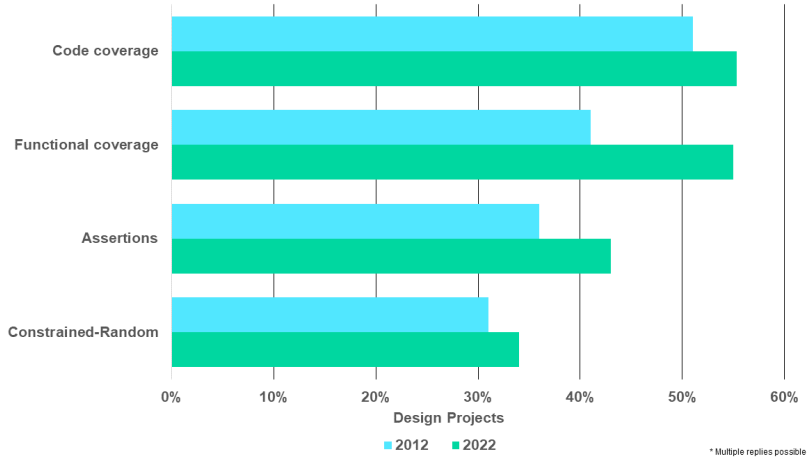

Fig. 19. FPGA project simulation technique trends.

VI. Conclusion

In this report, we presented FPGA design and verification trends based on a recent, large industry study. FPGAs have recently grown in complexity equal to many of today's IC/ASIC designs. We quantified the impact of this growing complexity in terms of verification effectiveness and effort.

Perhaps the most disturbing finding from this year's study relates to the number of FPGA projects with nontrivial bug escapes into production, as discussed in Section II. We did find an interesting correlation between the improvement of reduced functional flaws contributing to non-trivial bug escapes, as shown in fig. 3, and the maturing of FPGA projects' functional verification processes, as discussed in Section V.

The data suggest that projects that are more mature in their functional verification processes will likely experience fewer bug escapes. To test this claim, we partitioned the study participants into two independent groups: FPGA projects with no bug escapes and FPGA projects that experienced a bug escape. We then examined the percentage of adoption of various verification techniques, and the results are shown below in fig. 20. What we are unable to measure from our study is how effective a project was in adopting any of these processes. Nonetheless, these findings are statistically significant in that the group with no bug escapes tended to have higher adoption of various verification techniques, which suggests they are more mature in their verification process.

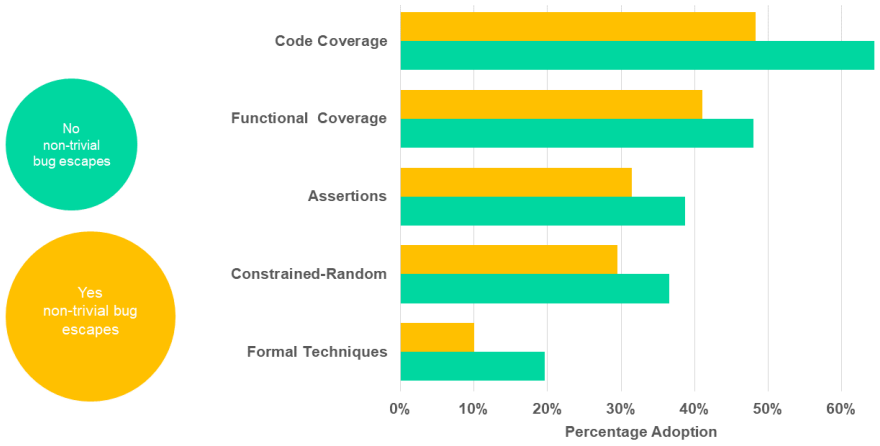

Fig. 20. FPGA simulation technique adoption vs non-trival bug escapes.

References

1 R. Collett, "2002 IC/ASIC functional verification study," Industry Report from Collett International Research, Inc. 2003.

2 R. Collett, "2004 IC/ASIC functional verification study," Industry Report from Collett International Research, Inc. 2005.

3 H. Foster, "Why the design productivity gap never happened" in Proceedings of the International Conference on Computer-Aided Design (ICCAD), IEEE Press pp. 581-584 2013

4 H. Foster, "Trends in functional verification: a 2014 industry study," Proceedings of the 52nd Annual Design Automation Conference (DAC), p.1-6, June 07-11, 2015, San Francisco, California.

5 H. Foster, "Trends in functional verification: a 2016 industry study," DVCon 2017, San Jose, California.

6 H. Foster, "2018 FPGA Functional Verification Trends," in 19th International Workshop on Microprocessor and SOC Test and Verification (MTV), Austin, TX, USA, Dec. 2018.

7 H. Foster, "Prologue: The 2020 Wilson Research Group Functional Verification Study," Siemens Digital Industries Software Blogs (blog), October 27, 2020, https://blogs.sw.siemens.com/verificationhorizons/2020/10/27/prologuethe-2020-wilson-research-group-functional-verification-study/

8 International Business Strategies, Semiconductor Market Analysis, January, 2023.

9 S. Trimberger, Three ages of FPGAs: a retrospective on the first thrity years of FPGA Technology, Proceedings of the IEEE, Vol 103, Issue 3, March 2015.

10 W. Chen, Member, S. Ray, M. Abadir, J. Bhadra, Li-C Wang, Challenges and Trends in Modern SoC Design Verification, IEEE Design & Test,Vol 34, Issue: 5, Oct

Siemens Digital Industries Software

Americas: 1 800 498 5351

EMEA: 00 800 70002222

Asia-Pacific: 001 800 03061910

For additional numbers, click here.

About Siemens Digital Industries Software

Siemens Digital Industries Software is driving transformation to enable a digital enterprise where engineering, manufacturing and electronics design meet tomorrow. Siemens Xcelerator, the comprehensive and integrated portfolio of software, hardware and services, helps companies of all sizes create and leverage a comprehensive digital twin that provides organizations with new insights, opportunities and levels of automation to drive innovation. For more information on Siemens Digital Industries Software products and services, visit siemens.com/software or follow us on LinkedIn, Twitter, Facebook and Instagram.

Acknowledgment

The author would like to thank Zachary Wilson (from Wilson Research Group) and Susan Hanson (from Siemens EDA) for their expertise and guidance in conducting very large industry studies.

siemens.com/software

85144-D5 01/23 K