Siemens EDA

2020 Wilson Research Group functional verification study

FPGA functional verification trend report

Executive summary

This report presents the results from the 2020 Wilson Research Group Functional Verification Study 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 Foster

Contents

I. Introduction 3
A. The global FPGA semiconductor market 3
B. Study background 3
C. Study confidence interval 4
D. Study bias 4
E. Report organization 4

II. FPGA verification effectiveness 5
A. Non-trivial bug escapes 5
B. Types of flaws resulting in non-trivial bug escapes .. 5
C. Design completion compared to original schedule... 5

III. FPGA verification effort 6
A. Percentage of project time spent in verification 6
B. Mean peak number of engineers 6

IV. FPGA design trends 7
A. Embedded processor cores 7
B. Asynchronous clock domains 7
C. Security features 8
D. Safety-critical design 8

V. FPGA verification adoption trends 9
A. Verification languages and methodolgy adoption trends 9
B. Verification technology adoption trends 10

VI. Conclusion and discussion 11

Acknowledgment 12

References 12

I. Introduction

This report presents field-programmable gate array (FPGA) functional verification trends based on the 2020 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] Our 2020 study builds on our previous studies by providing the latest industry trends .

A. The global FPGA semiconductor market

The 2019 global semiconductor market was valued at $385.4 billion after experiencing a 15 percent decline due to a 32 percent drop in the memory IC market, which is expected to recover in 2021.[7] The FPGA portion of the semiconductor market is valued at about $5 billion.[8] The FPGA semiconductor market is expected to reach a value of $7.5 billion by 2030, growing at a compounded annual growth rate (CAGR) of 4.4 percent during this forecast period. The growth in this market is being driven by new and expanding end-user applications related to data center computing, networking, and storage, as well as communication.

Historically, FPGAs have offered two primary advantages over ASICs. First, due to their low NRE,[9] 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-2000 timeframe underwent growing pains to address increased verification complexity. Similarly, we find today's FPGA market is being 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®2), traditional lab-based approaches to FPGA verification and validation are becoming less effective. In this report, we quantify the ineffectiveness of today's FPGA verification processes in terms of non-trivial 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, and 2018 Wilson Research Group Functional Verification Study.[3, 4, 5, 6] Each of these studies was modeled after the 2002 and 2004 Collett International Research, Inc. studies[1, 2] and focus on the IC/ASIC market. While we began studying the FPGA market in 2012, we waited until we had sufficient multi-year data points to identify verification trends before formally publishing the findings.

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 final 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 1492 eligible participants (i.e., n = 1492).

Fig. 1 compares the percentage of 2020 and 2018 study participants (i.e., design projects) by targeted implementation for both IC/ASIC and FPGA projects.

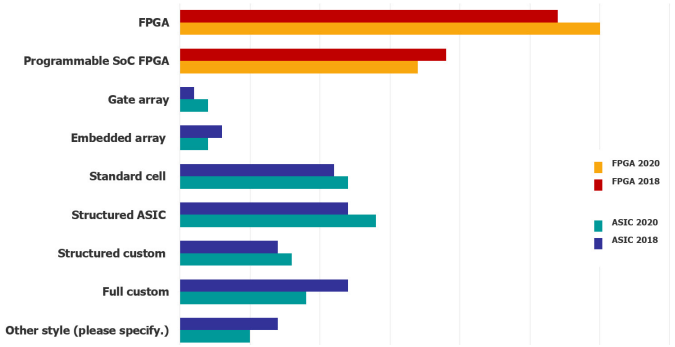

Fig. 1: Study participants by targeted implementation.

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 percent using a 95 percent confidence interval. In other words, this confidence interval tells us that if we were to take repeated samples from a population, 95 percent of the samples would fall inside our margin of error ±3 percent, and only 5 percent 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, non-response bias, and stakeholder bias. We have adopted multiple techniques to minimize these biases. However, the 2020 study demographics, as shown in fig. 2, saw an 11 percentage points decline in participation from North America but an increase in participation from Europe and India. This raises some interesting questions. Was the decline in North American participation due to COVID-19, which was peaking in the US during the June-July timeframe, as many employees shifted their work routine to a home environment? Or were spam filters more aggressive 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. Potential biases 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 from this year's study.

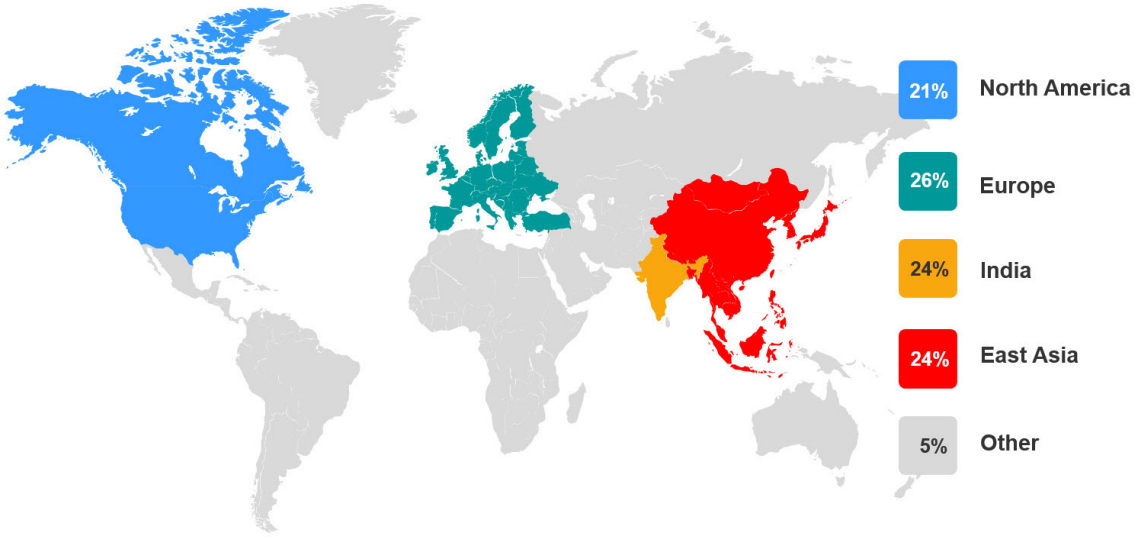

Fig. 2: 2020 study demographics.

II. FPGA verification effectiveness

In this section, we present FPGA project results in terms of 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 percent of IC/ASIC projects are able to achieve first silicon success, and most successful designs are 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 in fig. 3 are somewhat disturbing. In 2020, only 17 percent 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, and for some market segments, the cost of field repair can be significant. For example, in the mil-aero market, once a cover has been removed on a system to upgrade the FPGA, the entire system needs to be revalidated.

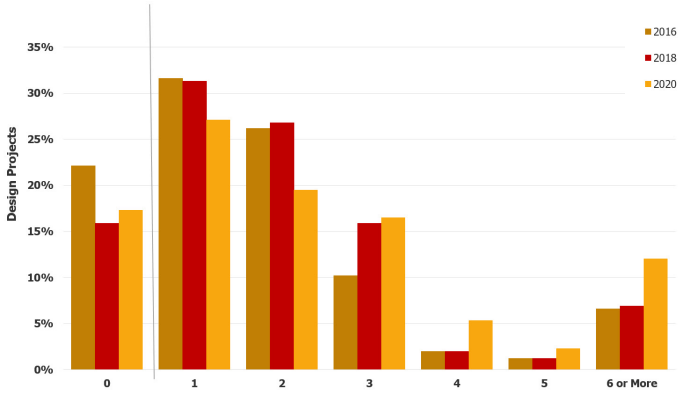

Fig. 3: Non-trivial FPGA bug escapes into production.

B. Types of flaws resulting in non-trivial bug escapes

Fig. 4 shows various categories of design flaws contributing to FPGA non-trivial bug escapes. The percentage of "logic or functional flaws" remains the leading cause of bugs. New flaws being tracked in the 2020 study are associated with safety (8 percent) and security (6 percent) features. Obviously multiple flaws can contribute to bug escapes, which is the reason the total percentage of flaws sums to more than 100 percent.

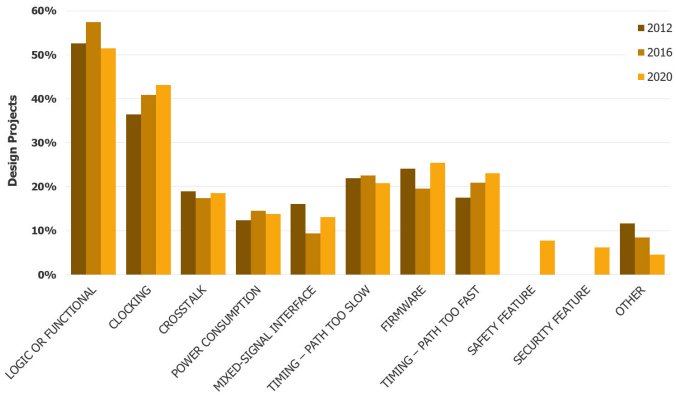

Fig. 4: Types of flaws resulting in FPGA bug escapes.

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. 5. Here we found that 68 percent 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 percent during the period 2014 through 2020.

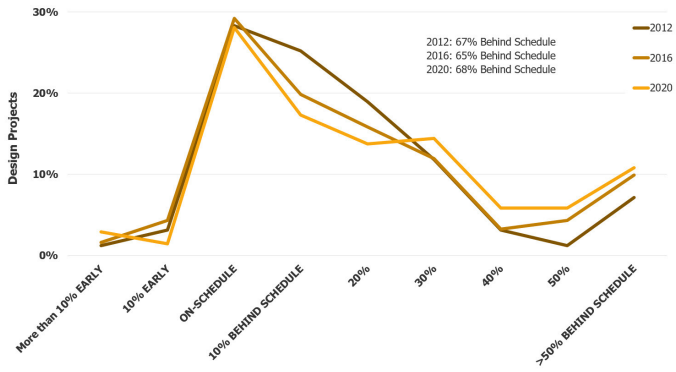

Fig. 5: Actual FPGA project completion compared to original schedule.

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. 6 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 working on 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.

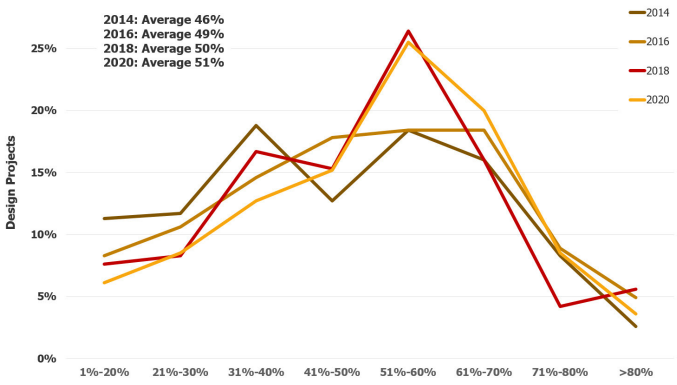

Fig. 6: Percentage of FPGA project time spent in verification.

Overall, we found an increase in the average percentage of FPGA project time spent in verification during the period 2014 through 2020. Again, this is an indication of growing design and verification complexity.

B. Mean peak number of engineers

Perhaps one of the biggest challenges today is to control 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. 7 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 1.5 percent CAGR between 2012 and 2020, the demand for FPGA verification engineers grew at a 5.5 percent 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]

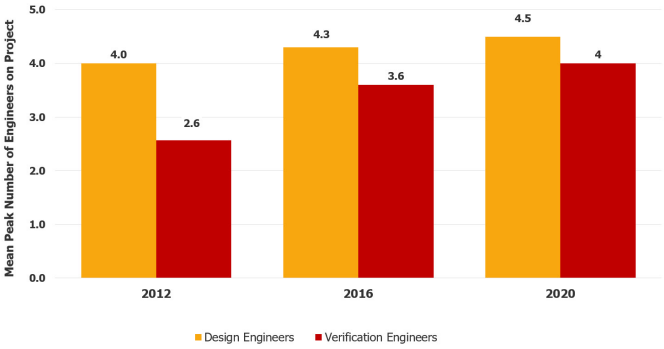

Fig. 7: Mean peak number of FPGA engineers.

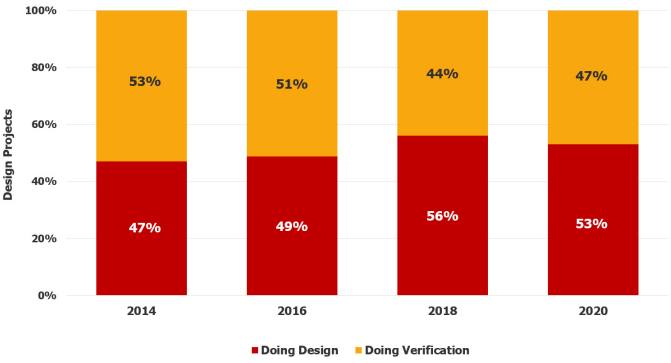

Fig. 8: Where FPGA design engineers spend their time.

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, as shown in fig. 8. In 2020, design engineers spent on average 53 percent of their time involved in design activities and 47 percent of their time in verification. However, when compared to 2014 and 2016, the data indicate a trend showing that FPGA design engineers are now spending slightly 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 2020.

Our study found that FPGA verification engineers spend more of their time debugging than with 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.

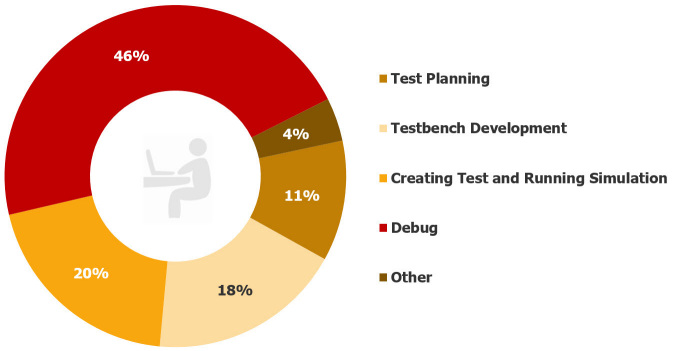

Fig. 9: Where FPGA verification engineers spend their time.

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 69 percent of all projects targeted their design at an FPGA containing one or more embedded processors, as shown in fig. 10. Furthermore, 45 percent of all FPGA designs today contain two or more embedded processors, while 3 percent include eight or more embedded processors.

SoC-class designs add a new layer of verification complexity to the verification process[10] that did not exist with traditional non SoC-class designs due to increased number of design requirements. For example, SoC-class designs often require verification of hardware and software interactions, new coherency architectures, and complex network-on-a-chip interconnect.

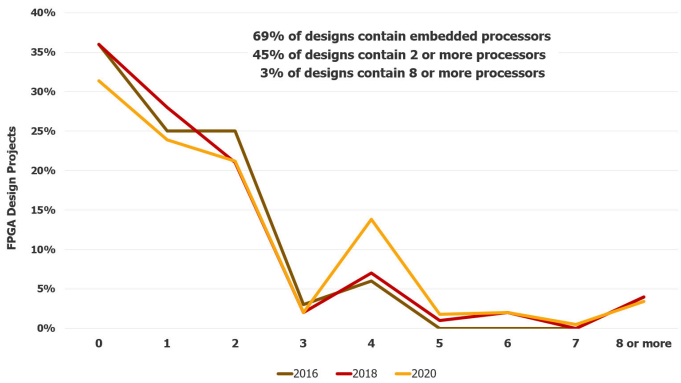

Fig. 10: Number of embedded processor cores.

Our 2020 study, for the first time, tracked the number of FPGA projects that have incorporated a RISC-V processor in their design, which was 23 percent. In addition, we tracked the number of FPGA projects that have

incorporated some type of AI accelerator processor (e.g., TPU, etc.), which was 19 percent.

B. Asynchronous clock domains

In fig. 11, we see that 92 percent of designs being implemented as FPGAs contain two or more asynchronous clock domains.

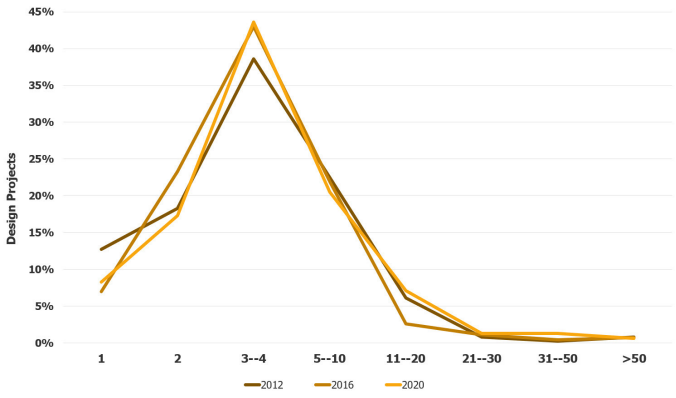

Fig. 11: Number of asynchronous clock domains.

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 43 percent 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 2020, we find that 40 percent of all FPGA projects are working under one of multiple safety-critical development process standards or guidelines. However, our 2018 study found that 56 percent of all FPGA projects were working under one of multiple safety-critical development standards, which we suspect was too an anomaly in the 2018 study. We know that the automotive industry is currently experiencing a contraction due to COVID-19, which might account for some of this year's decline in terms of safety critical projects. In addition, there are potentially some regional biases as discussed in the Introduction section D that should be considered.

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, which is why the percentage adoption sums to more than 100 percent. For example, IEC61508 and IEC61511.

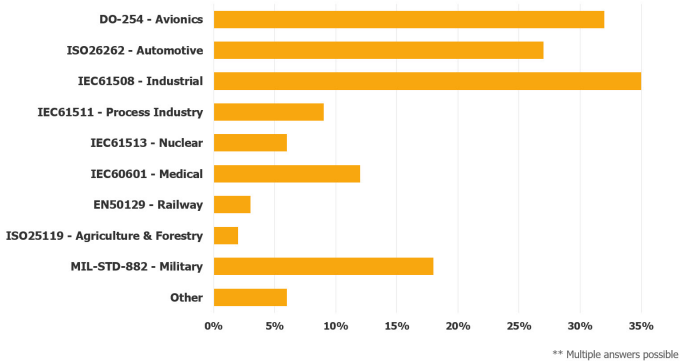

Fig. 12: Safety-critical development standard used on FPGA project.

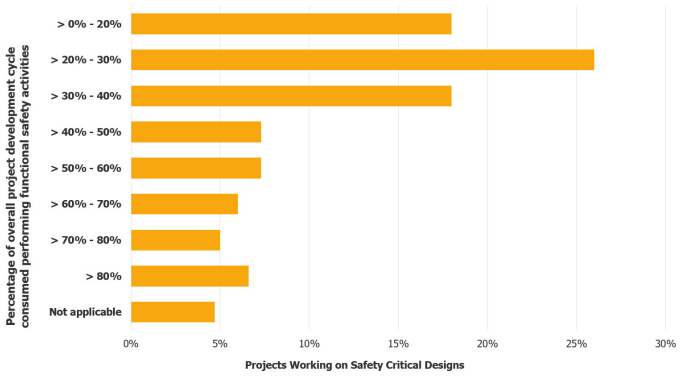

Fig. 13: Percentage of overall project time spent on functional safety.

Fig. 13 shows the percentage of overall project time spent in functional safety activities. The median

percentage of time is between 30 percent-40 percent. While fig. 14 shows the biggest challenges associated with functional safety as reported by this year's study participants.

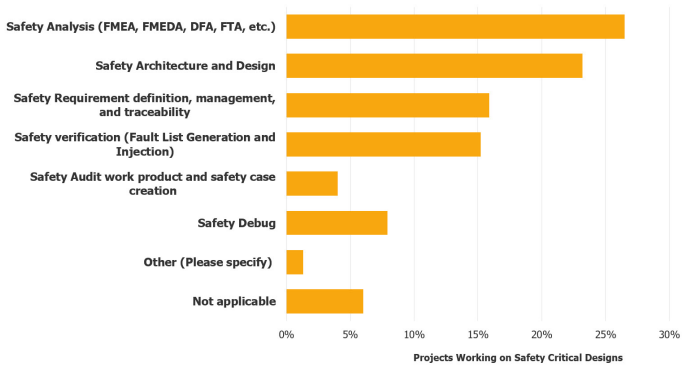

Fig. 14: Biggest functional safety project challenge.

V. FPGA verification adoption trends

To address growing verification complexity, we find 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

In fig. 15, we show the adoption trends for languages to build testbenches. Also, we need to consider potential 2020 regional biases associated with the findings in fig. 14 as discussed in the Introduction Section D.

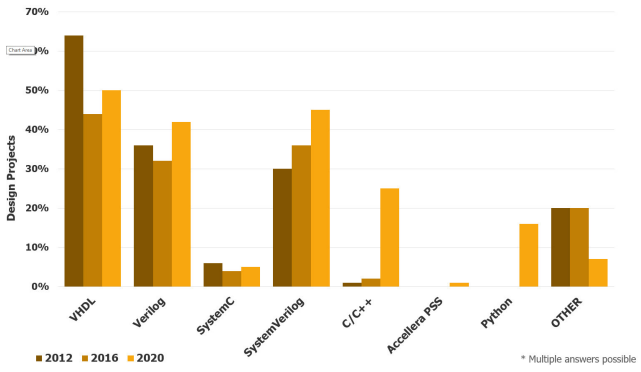

Fig. 15: FPGA project verification language adoption.

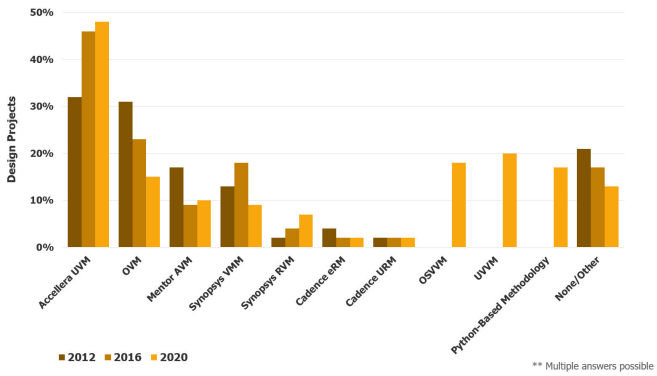

Fig. 16: FPGA project methodology and base-class libraries adoption.

It is not uncommon for FPGA projects to use multiple languages when constructing their testbenches, which is why the percentage adoption sums to more than 100 percent. This practice of adopting multiple languages is often due to legacy code as well as purchased verification IP written in a different language.

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, and the testbench was created using SystemVerilog.

What is unusual in the 2020 data was the huge increase in C/C++ for testbench development compared to previous years. It is unknown at this point if this was an anomaly in this year's study or an emerging trend. In addition, the 2018 level of adoption for the Accellera Portable Test and Stimulus Standard (PSS) was likely a misunderstanding by the study participants since its standardization was occurring at the same time as the 2018 study and few vendors supported it at that point in time. Finally, in 2020, for the first time, we explicitly asked about the adoption of Python for testbench development. In previous studies, Python was included with OTHER, which we now see has declined after moving Python to its own option.

The adoption trends for various base-class library and methodology standards are shown in fig. 16, and we found that the Accellera UVM is currently the predominant standard that has been 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 2020 we are showing trends for the first time. In addition, for the 2020 study, we track Python-based methodologies, such as cocotb, for the first time.

Finally, FPGA project adoption trends for various assertion language standards are shown in fig. 17. This is another example of potential 2020 regional biases as discussed in the Introduction Section D that needs to be considered when comparing trends across multiple studies.

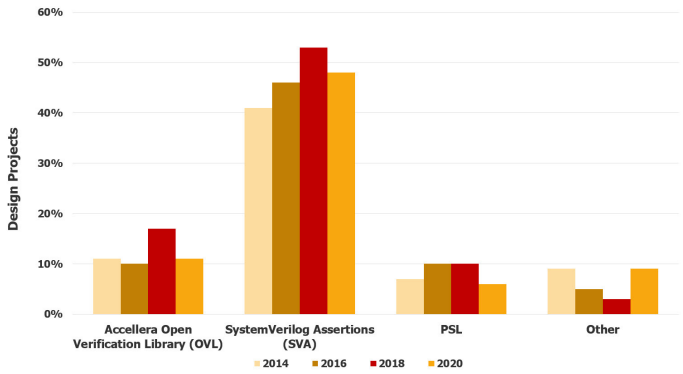

Fig. 17: FPGA project assertion lanauge adption.

SystemVerilog Assertions (SVA) is the predominant assertion language in use today. Similar to 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.

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.

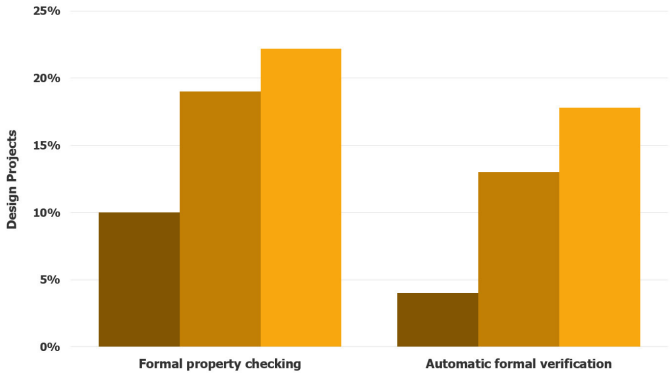

Fig. 18: FPGA project formal technology adoption trends.

We found that the adoption of formal property checking on FPGA projects is growing at an impressive 10 percent CAGR, and the adoption of automatic formal applications is growing at a 21 percent 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 and does 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. 19 shows the FPGA project adoption trends for various simulation-based techniques from 2012 through 2018, which include code coverage, functional coverage, assertions, and constrained-random simulation.

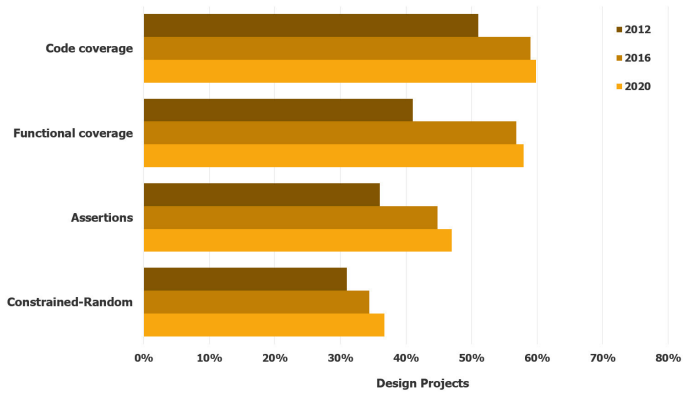

Fig. 19: FPGA project simulation technique trends.

One observation from these adoption trends is that the FPGA market has matured its verification processes. This maturity is likely due to the growing complexity of designs as discussed in the previous section.

VI. Conclusion and discussion

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 verification effectiveness and effort.

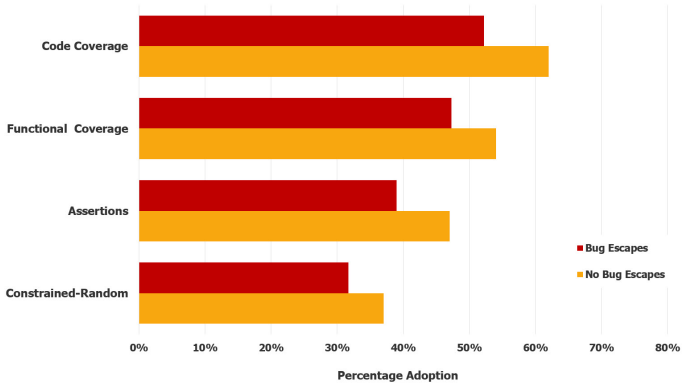

Fig. 20: FPGA simulation technique adoption vs non-trival bug escapes.

Perhaps the most disturbing finding from this year's study relates to the number of FPGA projects with non-trivial 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 adoption of various verification techniques and the results are shown 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.

Acknowledgment

The author would like to thank Zachary Wilson (from Wilson Research Group), in addition to Merlyn Brunken and Susan Hanson (from Siemens Digital Industries Software) for their expertise and guidance in conducting very large industry studies.

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. IC-Insights, April Update to The McCLEAN REPORT 2020 EDITION.
8. International Business Strategies, Semiconductor Market Analysis, Volume 29, No. 1, January, 2020.
9. S. Trimberger, Three ages of FPGAs: a retrospective on the first thirty 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. 2017

Siemens Digital Industries Software

Headquarters
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 972 987 3000

Americas
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 314 264 8499

Europe
Stephenson House
Sir William Siemens Square
Frimley, Camberley
Surrey, GU16 8QD
+44 (0) 1276 413200

Asia-Pacific
Unit 901-902, 9/F
Tower B, Manulife Financial Centre
223-231 Wai Yip Street, Kwun Tong
Kowloon, Hong Kong
+852 2230 3333

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. Xcelerator, the comprehensive and integrated portfolio of software and services from Siemens Digital Industries Software, 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. Siemens Digital Industries Software – Where today meets tomorrow.

siemens.com/eda

82392-C3 9/20 N/TGB