

An effective way to generate the shift timing constraints and sanity checks

Shaik Mahammad Ameer Afridi¹, Nagaraja Shylashree¹, Satish Tunga²,
Latha Bavikatte Nanjundappa³

¹Department of Electronics and Communication Engineering, Rashtreeya Vidyalaya College of Engineering, Bengaluru, India

²Department of Electronics and Telecommunication Engineering, M. S. Ramaiah Institute of Technology, Bengaluru, India

³Department of Electronics and Communication Engineering, Jagadguru Sri Shivarathreshwara Academy of Technical Education, Bengaluru, India

Article Info

Article history:

Received Jun 29, 2022

Revised Jan 18, 2023

Accepted Jan 26, 2023

Keywords:

Design for testability

Primitime

Sanity checks

Scan

Static timing analysis

Synopsys design constraints

ABSTRACT

Design for testability (DFT) is a technique, which facilitates a design to become testable after fabrication. As the technology node is shrinking, complexity of the system-on-chip (SoC) becomes high and inserting DFT and verifying its timing becomes complex. For these complex SoC, generating DFT timing constraints becomes difficult in shift mode and the time required for the generation of these timing constraints is also more. A new methodology proposed to overcome these issues. The main objective of this work is to propose the flow for generating DFT timing constraints for the complex SoC in shift mode, by dividing the design blocks into scan blocks and non-scan blocks. To target the whole design without getting all paths reported, relaxation of the setup, and hold time for non-scan blocks plays crucial role. If not, the time taken to generate DFT timing constraints would be more. Implemented methodology of this paper includes design setup, timing exceptions, and synopsys design constraints (SDC) generation for DFT timing. Design setup consists of all pre-requisites for design such as netlists, timing libraries, and exceptions. Synopsys primitime (PT Shell) is used for all the timing-related checks. Compared to conventional methods, the proposed flow reduces the overall time by 40% to generate constraints.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Nagaraja Shylashree

Department of Electronics and Communication, Rashtreeya Vidyalaya College of Engineering

R. V. Vidyanikethan Post, 8th Mile, Mysuru Road, Bengaluru 560059, Karnataka, India

Email: shylashreen@rvce.edu.in

1. INTRODUCTION

Design for testability (DFT) is the process, to detect all the defects of a chip after fabrication. DFT makes integrated circuits (ICs) testable which is the controllability and observability of internal nodes by adding additional circuitry to the functionality of the design. This makes the testing of chips cost-effective [1]. The transistor number on a system on chip (SoC) has increased from millions to billions in the last few years. The time to assess or verify a chip is obtained by the number of test patterns, test vectors [2] used, and the time to apply each test vector. As the number of test cases increases, the time taken to apply them also increases. So, the time taken to apply test vectors turned into an increasing worry. Currently, the very large-scale integration (VLSI) industry is experiencing a huge difficulty in testing complex SoC designs [3]. Scan design [4] is the DFT method that can test an extremely complex design with good fault detection. The goal of scan design is to make every flip-flop in the chip controllable and observable. In DFT, a scan-based design is used for testing to provide more fault coverage. Scan-based design for testability uses scan cells which are the

combination of a mux and a D flip-flop. Several scan cells are stitched together to make a scan chain. In the scan-based design approach, a few or all of the flip-flops are substituted with the scan cell, if all flip-flops in the design are updated with scan cells it is called full scan [5] testing where only some flip-flops are updated with scan cells it is known as partial scan testing [6]. High test coverage and fault coverage can be obtained by using full scan testing.

Timing constraints in a design are vital attributes to analyze the timing. In static timing analysis, one of the inputs for the static timing analysis (STA) tool is a constraint file and the format of the file is synopsys design constraints (SDC). This SDC file contains the timing constraint information related to the design and should be the same as the constraint file used for synthesis. SDC file includes the clock definitions for design, input/output delay, max/min delay, and timing exceptions [7] such as multicycle path, false path, set-case-analysis, and set disabling time. The SDC file defines all timing constraints included for both setup and hold. In designs first, all normal flip-flops replace with scan flops then scan stitching will be done. The nodes in the design must control and observe and control and observe the test vectors should send through scan chains. This testing will happen in three phases first load, second shift then capture [8], as each phase depends on the clock edge, clock cycles play a significant role in the timing. In the shift to test the functionality, mainly against any stuck-at faults [9], [10] of the designed chip, the sequential devices will be stitched in the chain order, and the test pattern to detect the fault is applied in the series.

2. DESIGN SETUP TO GENERATE THE DFT TIMING CONSTRAINTS

DFT timing constraints [11] generation flow starts with design loading, to load the design prerequisites are needed, which are shown in Figure 1. These prerequisites are categorized as three blocks for simplicity: the timing exceptions [12] block, netlist block, and libraries block. Timing exceptions block consists of clock information and exceptions of soft IP in SDC format.

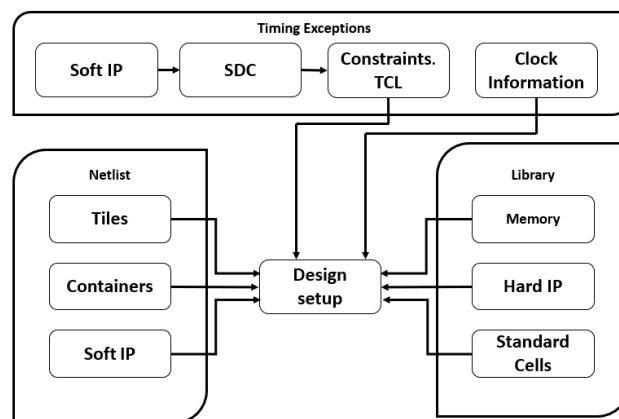


Figure 1. Design setup to generate DFT timing con constraints

Clock information consists of timeperiods of test clocks and functional clocks, again in test clocks, different frequency clocks are present. For soft IP, constraints present in SDC format. The focus of the paper is on timing exceptions in the shift mode, which is one of the test modes of DFT. In general, constraints are present for functional synthesis and timing [13]. Similarly, timing constraints for the DFT part must be taken care of to proceed further in the digital flow. One can expect errors in quality checks for timing due to constraints that are already applied in functional mode and make sure that whatever errors that are getting are due to exceptions that are already present. There is a tool command language (TCL) file called constraints. TCL is present in the timing exceptions block to impose all constraints on the design.

Netlist block consists of tiles, containers, and soft IP. All the information on these netlists is in verilog format. Containers are a group of tiles where some of the tiles are DFT-related. DFT-related tiles include clock domain logic control [14], [15] tiles in DFT mode and power domain control logic tiles in test mode and so on. All design-related information is covered in the netlist and libraries block. Libraries include memory, hard IP, and standard cells, these libraries can be in .db format.

3. PRIMETIME FLOW TO GENERATE THE DFT TIMING CONSTRAINTS

Design data from all the blocks should be loaded into the tool named build. The build tool plays a key role in setup the design. When it comes to build, this tool takes targets as inputs and runs them and each target contains the pre-defined task. Once the targets got completed, this session should be saved to further debug the timing violations and quality of the SDC generation. The session saved in build represents the primetime session, by using that saved session one can easily restore the session in synopsys prime time tool. The session that is saved after completing the targets in the build tool is restored in prime time. The design could be restored in prime time and timing analysis can be done but timing analysis needs to be done by using the session. The restored session is helpful to review whether the check timing is clean or not if it is clean proceed with report timing. Check timing command helps to find the missing definitions for clocking, unconstrained endpoints, undefined input arrival times, and undefined output constraints. If the check timing is not clean update the constraints and update the run area, the flow will be the same after updating the run area again session should be saved and restored in primetime. Once report timing is generated reports of paths that are violating cross-verify why the path is failing to meet the timing. The reasons for getting the slack violation can be constraints that are already provided by the functional design, if that is the reason waive the violations if not fix the violations. Extracted timing models (ETM) should be the latest for the design.

ETMs are used to reduce the reporting time, in other words, all internal timing arcs will be suppressed [16]. By suppressing all internal timing arcs, functionality can be secure and if the block is already met its timing at the gate level, then there is no need to check its timing at the chip level. These are the advantages of extracted timing models, as discussed earlier they mainly consist of interface timing arcs. If extracted timing models are the latest then proceed with report timing otherwise update the constraints as shown in Figure 2.

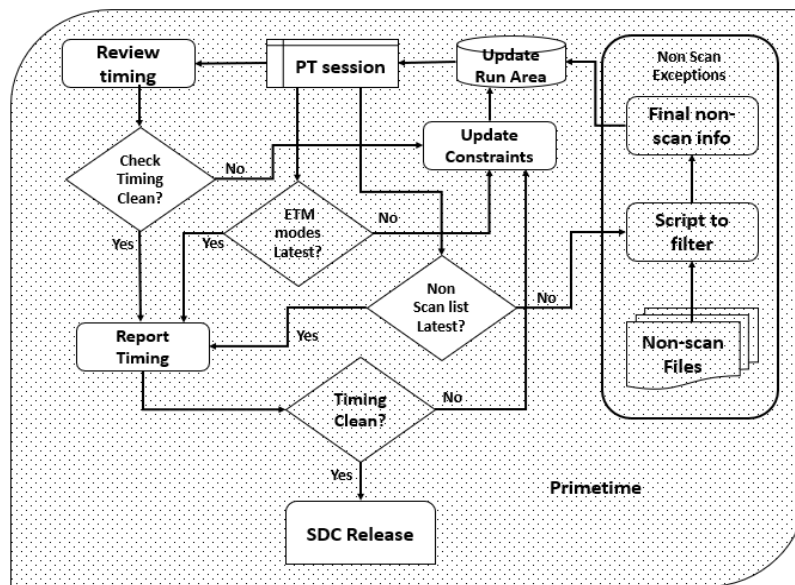


Figure 2. Design flow for SDC generation

Non-scan cells are the cells which don't involve in scan mode, for this there are multiple explanations. One of the reasons is some cells and IPs are provided by third parties which intend those vendors don't want their IPs to be scanned for security purposes, and power consumption [17] in scan mode can be reduced. If timing analysis is performed for these cells they all will be reported for timing, this process will take additional time and effort to segregate results and reports. To avoid this, make a list of all non-scan cells and write a script to exclude these cells from report timing. To segregate the non-scan list from the design the script is available. Similar to ETM check non-scan list is also the latest or not, if not pick up the latest list by using the script which is present in the non-scan exception block.

Update the constraints as per each block shown in Figure 2 and, update the run area after updating the constraints. Once the report timing is done check whether the timing is clean or not, if not update the constraints and then update the run area and follow the same steps again. Sanity checks are done to cross-verify the basic things like connections and timing definitions that must be present in design before proceeding further in the design cycle. Sanity checks can be done at different stages of the design cycle. In this paper,

these sanity checks are done at the DFT timing stage, and these checks include chip-level scan flop check, tile-level scan flop check memory check, and pipeline checks [18].

In chip-level scan flip-flop check, timing definitions and exceptions on the flops are being checked. To restore the pt session which was saved during the end of the build tool run, scan cells must exclude from the chip-level scan flip-flop check. To exclude them from design, keep all scan cells in a file and made list, and write a script that generates a scan cell report by taking the scan cell list as input. These output reports should be sourced in the pt session and then generate log files. Check for dominant exceptions in the log file, if any exceptions are present on the scan path, then remove that exception from non-scan exceptions and regenerate the SDCs. Chip-level sanity checks should be done for all the test modes. This process is explained using the flow chart in Figure 3.

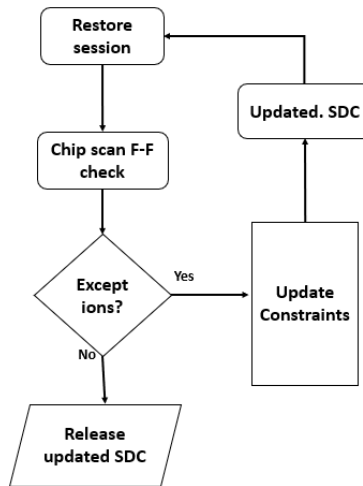


Figure 3. Sanity check for chip-level scan flops

Tile-level scan flip-flops are also having some timing constraints, in this check exceptions on the tile-level scan, flops can be detected. If there are any exceptions found for tile scan flops verify whether those exceptions came because of timing definitions from the point of functionality or because of an issue in design. If the exceptions are because of design update the constraints and update the SDC as well. Repeat the same procedure as shown in Figure 4. Release or generate the SDC, if no need to update constraints. The same procedure is applicable to pipeline flops which are shown in Figure 5.

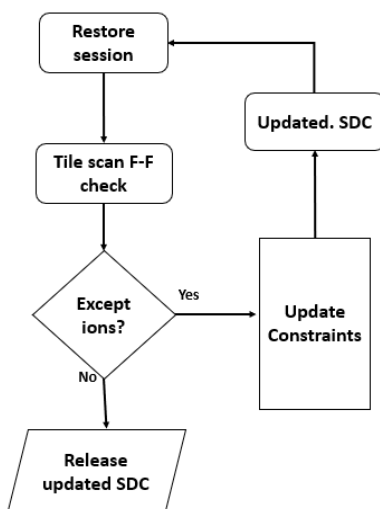


Figure 4. Sanity check for tile-level scan flops

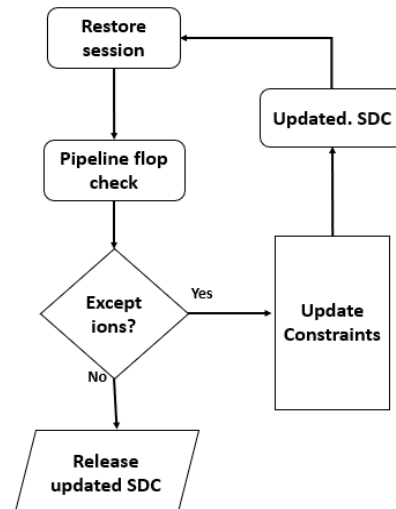


Figure 5. Sanity check for pipeline flops

Memories occupy a large part of VLSI designs, and the purpose of memories is to store a large amount of data. As memories do not consist of any logic gates and flip-flops, as a result, comparatively different and new fault models and methodologies have to use for memory testing. As a part of DFT, memory is one of the major things to test or verify. To write data into memory blocks and read them back connections should be in such a way to access all rows and columns in memory, a memory interface is needed. In this interface on some of the sequential cells, timing exceptions can be found. This check is going to verify the timing exception on the memory interface as shown in Figure 6. Restore the session first after setup the design then checks whether there are any exceptions on the memory interface. If any exceptions are found update the constraints and update the SDC if not generate the updated SDC.

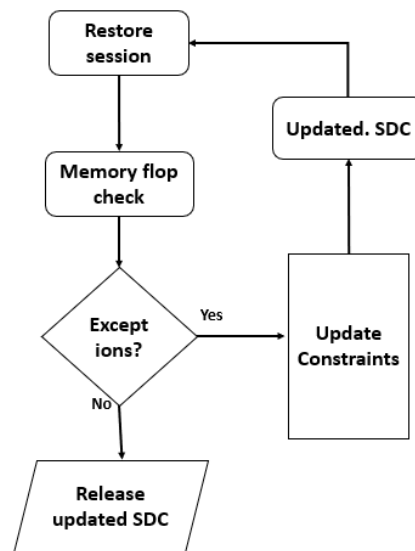


Figure 6. Sanity check for memory interface

4. RESULTS AND DISCUSSION

An effective methodology is used to generate the DFT timing constraints and sanity checks for a complex SoC in which there are thousands of scan flip-flops at the tile level and chip level and several memory interfaces used for memories also a lot of pipeline flops were used. Results are plotted in terms number of elements that had exceptions on them for sanity checks and elements that had default exceptions as shown in Figure 7 along with timing reports [19]. The vertical axis represented the number of scan flops that had default exceptions and exceptions that are found are described in order of thousands, and the horizontal axis represents tile level, chip level scan flops, pipeline flops, and memory interface. Four thousand three hundred scan flip-flops are having exceptions before generating the timing constraints, those are predefined constraints. After generating constraints, 1200 flip-flops got reported as they have timing [20] exceptions on them. If this issue moves forward in the design flow it will lead to timing failure in lateral stages, the same is applicable to tile-level flops and pipeline flops as well. This paper completely eliminates this propagation of timing exceptions towards the next stage with an effective method. This method finds exceptions on flops by using the primetime tool. Eliminated those exceptions based on the type of exception by updating the constraints file or using timing fixations [21]. Bidirectional signalling [22] methodology has reduced overall test time by more than 40% but it will work for only 3-D stacked IC designs. Similar to bidirectional signalling, a recursive hierarchical methodology [23] was proposed which reduces the overall run time but this work will be a burden to the later stage in the design cycle.

Another improved test methodology [24] was proposed to reduce the test time unfortunately due to usage of on-chip controller it will be applicable to only multi clock domain testing in SoC. When compared to all these methodologies proposed methodology has no limitations over other parameters. Once set the design and all inputs, with in less time will get timing constraints.

The time taken to generate the DFT timing constraints [25] using the proposed methodology is 40% less time than the general approach which is shown in Figure 8. The vertical axis represents the two different methodologies, and the horizontal axis represents the time in percentage. These results are calculated based on different tools that take the time to generate constraints and sanity reports.

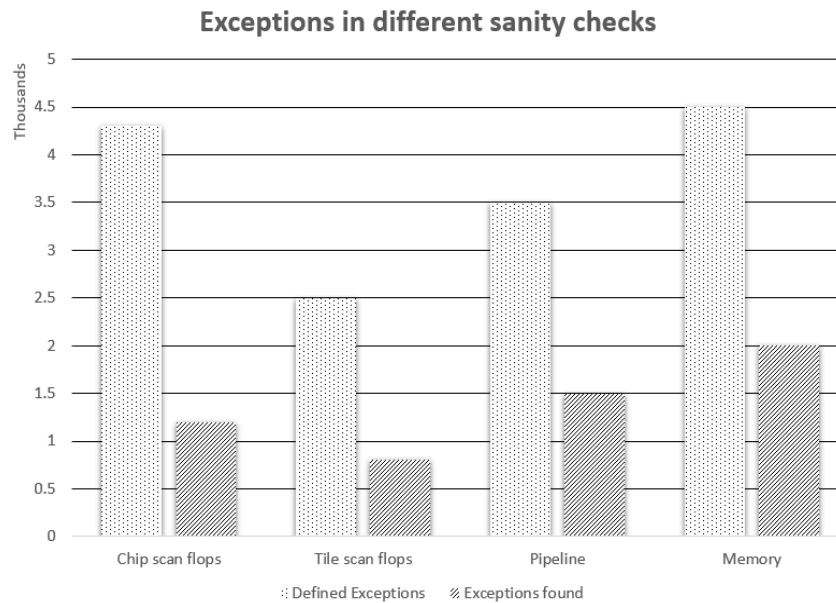


Figure 7. Exceptions in different sanity checks

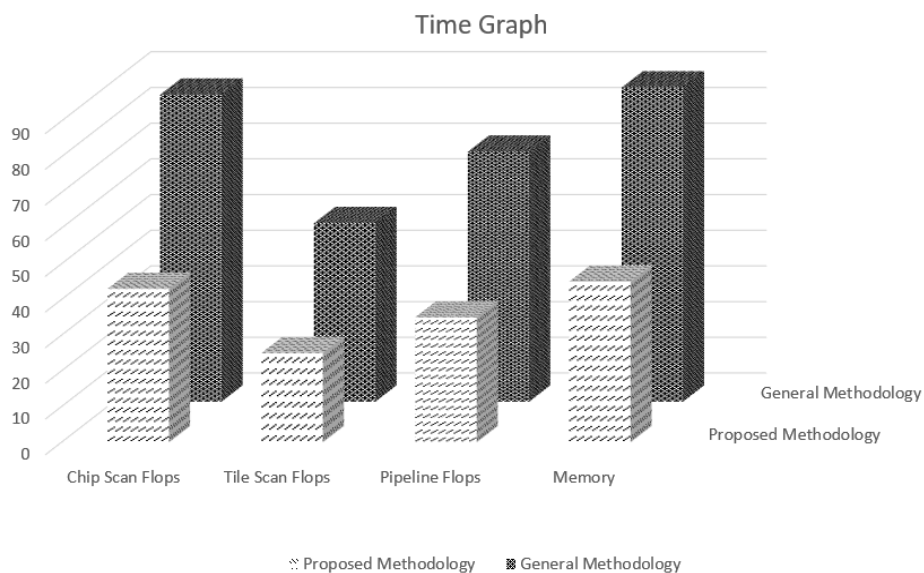


Figure 8. Time graph for different scenarios using proposed methodology

5. CONCLUSION

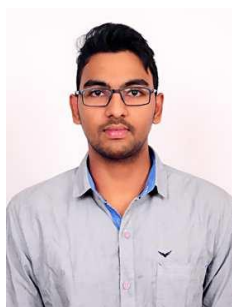
A new methodology and effective way proposed in order to generate the timing constraints especially in DFT. As there are many approaches present the time taken by them are very high compared to proposed methodology, this work successfully reduced the time taken to generate constraints and sanity checks. The main advantage of this method is it will completely stop the propagation of reported exceptions to move forward in design cycle, by doing this we can avoid engineering change order (ECO) in the last stages of project. If inputs of the tools that have been used in this experiment are automated using proper scripting further time will get reduced and fewer time constraints will be generated.




REFERENCES

- [1] B. Esen, A. Coyette, N. Xama, W. Dobbelaere, R. Vanhooren, and G. Gielen, "A very low cost and highly parallel DFT method for analog and mixed-signal circuits," in *Proceedings of the European Test Workshop*, May 2017, pp. 1–2, doi: 10.1109/ETS.2017.7968225.




- [2] H. Chen, D. Chen, J. Ye, W. Cao, and L. Gao, "An integrated automatic test generation and executing system," in *2011 IEEE AUTOTESTCON*, Sep. 2011, pp. 383–390, doi: 10.1109/AUTEST.2011.6058726.
- [3] P. K. D. Jagannadha *et al.*, "Advanced test methodology for complex SoCs," in *2016 IEEE International Test Conference (ITC)*, Nov. 2016, pp. 1–10, doi: 10.1109/TEST.2016.7805857.
- [4] H. Yi, S. Kundu, S. Cho, and S. Park, "A scan cell design for scan-based debugging of an SoC with multiple clock domains," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 57, no. 7, pp. 561–565, Jul. 2010, doi: 10.1109/TCSII.2010.2049923.
- [5] H. Vranken, T. Waayers, H. Fleury, and D. Lelouvier, "Enhanced reduced pin-count test for full-scan design," *Journal of Electronic Testing: Theory and Applications (JETTA)*, vol. 18, no. 2, pp. 129–143, 2002, doi: 10.1023/A:1014989408897.
- [6] J. Rearick, "The case for partial scan," *Proceedings International Test Conference 1997*, Washington, DC, USA, 1997, doi: 10.1109/TEST.1997.639722.
- [7] D. Goswami, K.-H. Tsai, M. Kassab, and J. Rajski, "Test generation in the presence of timing exceptions and constraints," in *Proceedings of the 44th annual conference on Design automation - DAC '07*, 2007, doi: 10.1145/1278480.1278653.
- [8] S. Wu *et al.*, "Using launch-on-capture for testing scan designs containing synchronous and asynchronous clock domains," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 30, no. 3, pp. 455–463, Mar. 2011, doi: 10.1109/TCAD.2010.2092510.
- [9] P. C. Reddy, K. SaiTulasi, T. J. Anuja, R. Rajarajeswari, and N. Mohan, "Automatic test pattern generation of multiple stuck-at faults using test patterns of single stuck-at faults," in *Proceedings of the 5th International Conference on Trends in Electronics and Informatics, ICOEI 2021*, Jun. 2021, pp. 71–75, doi: 10.1109/ICOEI51242.2021.9452949.
- [10] C. -H. Wu and K. -J. Lee, "An efficient diagnosis pattern generation procedure to distinguish stuck-at faults and bridging faults," *2014 IEEE 23rd Asian Test Symposium*, Hangzhou, China, 2014, pp. 306–311, doi: 10.1109/ATS.2014.56.
- [11] A. Lamkanfi and S. Demeyer, "Filtering bug reports for fix-time analysis," in *2012 16th European Conference on Software Maintenance and Reengineering*, Mar. 2012, pp. 379–384, doi: 10.1109/CSMR.2012.47.
- [12] V. Bhargava, K. Kapoor, and S. S. Iqbal, "Enhanced timing closure using latches," *EDN Magazine*, Jun. 2014.
- [13] S. Savugathali, M. Mustapa, M. Sharazel Razali, and F. Faiz Zakaria, "Timing violation reduction in the FPGA prototyped design using failed path fixes and time borrowing techniques," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 14, no. 2, pp. 628–636, May 2019, doi: 10.11591/ijeecs.v14.i2.pp628-636.
- [14] A. Majumdar and B. Jayadev, "Handling clock-domain crossings in dual clock-edge logic for DfX features," in *2018 IEEE 27th Asian Test Symposium (ATS)*, Oct. 2018, pp. 36–41, doi: 10.1109/ATS.2018.00018.
- [15] C. H. Ang, "Single test clock with programmable clock enable constraints for multi-clock domain SoC ATPG testing," *2013 22nd Asian Test Symposium*, Yilan, Taiwan, 2013, pp. 195–200, doi: 10.1109/ATS.2013.44.
- [16] C. W. Moon, H. Kriplani, and K. P. Belkhal, "Timing model extraction of hierarchical blocks by graph reduction," in *Proceedings of the 39th conference on Design automation - DAC '02*, 2002, doi: 10.1145/513918.513957.
- [17] S. Mitra and D. Das, "A comprehensive review on applications of don't care bit filling techniques for test power reduction in digital VLSI systems," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 12, no. 3, pp. 941–949, Dec. 2018, doi: 10.11591/ijeecs.v12.i3.pp941-949.
- [18] H. Morisita, K. Inakagata, Y. Osana, N. Fujita, and H. Amano, "Implementation and evaluation of an arithmetic pipeline on FLOPS-2D: multi-FPGA system," *ACM SIGARCH Computer Architecture News*, vol. 38, no. 4, pp. 8–13, Sep. 2010, doi: 10.1145/1926367.1926370.
- [19] T. Thiel, "Have I really met timing? - validating primetime timing reports with SPICE," in *Proceedings Design, Automation and Test in Europe Conference and Exhibition*, 2004, vol. 3, pp. 114–119, doi: 10.1109/DATE.2004.1269216.
- [20] N. Shylashree, V. S. Bharadwaj, D. Yashas, V. Kulkarni, A. Bharadwaj, and V. Nath, "Comprehensive design and timing analysis for high speed master slave D flip-flops using 18 nm FinFET technology," *IETE Journal of Research*, pp. 1–8, Aug. 2021, doi: 10.1080/03772063.2021.1948925.
- [21] B. Li, N. Chen, and U. Schlichtmann, "Timing model extraction for sequential circuits considering process variations," *2009 IEEE/ACM International Conference on Computer-Aided Design - Digest of Technical Papers*, San Jose, CA, USA, 2009, pp. 333–343.
- [22] I. A. Soomro, M. Samie, and I. K. Jennions, "Test time reduction of 3-D stacked ICs using ternary coded simultaneous bidirectional signaling in parallel test ports," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 12, pp. 5225–5237, Dec. 2020, doi: 10.1109/TCAD.2020.2977604.
- [23] D. Trock and R. Fisette, "Recursive hierarchical DFT methodology with multi-level clock control and scan pattern retargeting," in *Proceedings of the 2016 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2016, pp. 1128–1131, doi: 10.3850/9783981537079_1131.
- [24] E. M. Ooi and C. H. Ang, "Improved test methodology for multi-clock domain SoC ATPG testing," in *Fifth Asia Symposium on Quality Electronic Design (ASQED 2013)*, Aug. 2013, pp. 33–38, doi: 10.1109/ASQED.2013.6643560.
- [25] H. Chen, B. Lu, and D.-Z. Du, "Static timing analysis with false paths," in *Proceedings 2000 International Conference on Computer Design*, 2000, pp. 541–544, doi: 10.1109/ICCD.2000.878336.

BIOGRAPHIES OF AUTHORS






Shaik Mahammad Ameer Afridi    is a student at RV College of Engineering. He is pursuing his post-graduation in VLSI Design and Embedded Systems. He got his graduation degree from IIIT Rajiv Gandhi University of Knowledge and Technologies, Rk valley, idupulapaya in Electronics and communication engineering. His research line is static timing analysis in the DFT domain. He can be contacted at email: shaikmaafriidi.lvs20@rvce.edu.in.



Dr. Nagaraja Shylashree    is currently working as Associate Professor in the Department of Electronics and Communication Engineering at RV College of Engineering, Bengaluru. She is having 15 years of teaching experience. She was a recipient of the best Ph.D. thesis award for the year 2016-2017 in Electronics and Communication Engineering from BITES. She has received the best IEEE researcher award in IEEE-AGM meeting held during 2021 from Bangalore IEEE section. She has also received the best paper award in IEEE-ICERECT held during 2015 at Mandya. She has research publication in 20 International Journals (out of which 8 journals are SCI journals), 6 Springer book chapters and 9 International conferences. She received one US patent grant, one European patent grant, one Indian patent grant and published one Indian patent in the area of cryptography. She has also received 2 Indian patent grants in the area of VLSI. She is also the co-author of the network theory, engineering statistics and linear algebra and control engineering textbook. she has funded projects and consultancy projects and has delivered many technical talks on VLSI. She has delivered lectures as a subject matter expert in VTU e-shikshana and EDUSAT program. She is a recipient of international travel grant under SERB young research scholar category. She is a life member of ISTE, IETE, fellow member of ISVE, senior member of IEEE and Secretary of IEEE CAS, Bangalore section. Her areas of interest include cryptography and network security, network analysis, analysis and design of digital circuits, digital VLSI design, analog and mixed mode VLSI design, low power VLSI design, statistics and linear algebra and control engineering. She can be contacted at email: shylashreen@rvce.edu.in, shylashashi@gmail.com.



Dr. Satish Tunga    received his Ph.D. in Electronics Engineering from Jain University, Bangalore in 2018. He did his B.E. and M.E. in Electronics, in 1984 and 1991, respectively, from University Visvesvaraya College of Engineering, Bangalore. He is presently working as an associate professor in Department of Electronics and Telecommunication Engineering, M S Ramaiah Institute of Technology, Bangalore. He has published more than 10 papers in various international conferences and journals. His areas of interests are image processing, signal processing, communication systems, antennas, and electronic circuits. He can be contacted at email: satish.tunga@msrit.edu.



Prof. Latha Bavikatte Nanjundappa    is currently working as Assistant Professor in the Department of Electronics and Communication Engineering at JSS academy of technical education, Bangalore. She has 30 years of teaching experience Her areas of interest are signal processing, power electronics, computer networks, HDL, and control engineering. She has completed BE and M. Tech from Mysore University and NITK, Surthkal, Mangalore University in the year 1990 and 1997 respectively. She can be contacted at email: lathabn@jssateb.ac.in; lathajss@gmail.com.