International Journal of Reconfigurable and Embedded Systems (IJRES)
Vol. 10, No. 1, March 2021, pp. 18~24
ISSN: 2089-4864, DOI: 10.11591/ijres.v10.i1.pp18-24

Characterization and hierarchical static timing analysis of mixed-signal design

Sowmya K. B., Thanushree M.
Department of Electronics and Communication Engineering, R V College of Engineering, Bengaluru, India

Article Info

Article history:

Received Jun 2, 2020
Revised Oct 1, 2020
Accepted Jan 10, 2021

Keywords:

HSTA flow
On-chip variation
Parasitic corners
Static timing analysis

ABSTRACT

As the technology grows, the tendency to increase the data rate also increases. Clocks with higher frequencies have to be generated to meet the increased data rate. Any mismatch between the clock rate and data rate will lead to the capture of the wrong data. Hence performing timing analysis for any design to validate the capture of correct data plays a major role in any System on chip. This paper explains the procedure followed to perform timing analysis for any mixed-signal design.

Corresponding Author:

Sowmya K. B.
Department of Electronics and Communication Engineering
R V College of Engineering
Bengaluru, Karnataka, India
Email: kb.sowmya@gmail.com

1. INTRODUCTION

The Static Timing Analysis (STA) is one of the techniques available to verify the timing of any design. An alternative approach that is also available to verify the timing is the timing simulation which can verify the functionality along with timing verification of any design.

STA is said to be static since the analysis of any design is statically performed and does not rely upon the data values which are being applied at the input pins. This contrasts with simulation-based timing analysis where different patterns of stimulus signal are applied at the input and then the resulting behavior for all patterns is observed and verified. Given a design with the definition of the external environment and a set of input clock definitions, the purpose of static timing analysis is to verify and validate if the design is capable to operate at the rated speed.

The design under analysis is defined using a hardware description language such as Verilog or VHDL. The external environment along with the clock definitions are defined using design constraints (SDC) or an equivalent format. The timing reports are in ASCII form with multiple columns. Each column shows one attribute of the path delay [1, 2].

The design performance and its functionality can be limited by the noise in mixed-signal designs. The noise occurs due to many reasons like noise on primary inputs or power supplies or due to crosstalk with other signals. The noise can limit the operating frequency of the design and can also cause functional failures. Thus, a design implemented must be verified to be robust which means that it should have the capability to withstand noise without affecting the design performance. Verification based upon logic simulation cannot handle the effects of noise, on-chip variations, and crosstalk.

2. ON-CHIP VARIATION

Different portions on a chip have different PVT conditions because of the variations in environmental and process parameters. These variations lead to violation in timing; changes in MOS characterization can affect wire and cell delays. To account for these PVT variations, OCV analysis must be done during STA. Since data and clock paths may be affected differently by on-chip variation, timing verification can model this variation effect by making PVT conditions for capture and launch paths to be different. In STA these variations can be modeled by derating the delays of paths either making it faster or slower as per the requirement. Net and cell delay can be derated while modeling OCV [3-5].

2.1. Derating OCV for setup check

The worst condition possible for setup violation due to OCV occurs when launching clock path and data path OCV leads to the longest delay while the capturing clock path has OCV conditions which lead to the shortest delay [6-8]. This condition is the most restrictive case for setup check which is also shown in Figure 1.

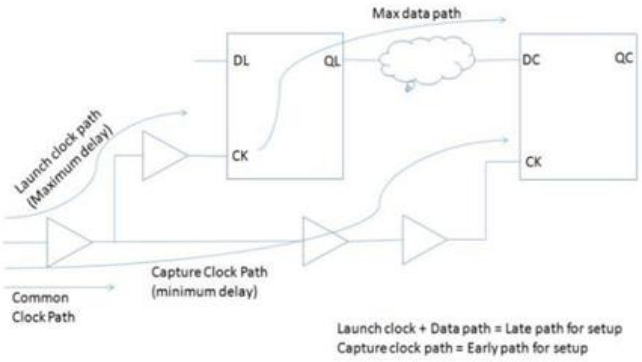

Figure 1. Derating on-chip variation for setup check [8]

To account for these violations, the following constraints are provided which will derate early/minimum paths by -10% i.e the path is made more faster and will derate late/maximum-paths by +10% i.e., the path is delayed more. set-timing-derate -early 0.9 set -timing-derate -late 1.1.

To avoid setup violation the data path has to be multiplied with late derate (delayed more) and capture path should be multiplied with early derate (made faster). Net and cell delays can also be derated using -net-delay and - cell-delay with set-timing-derate.

2.2. Derating OCV for hold check

The worst condition possible for hold violation due to OCV occurs when launching clock path and data path OCV leads to the shortest delay while the capturing clock path has OCV conditions which lead to the longest delay. This condition is the most restrictive case for hold check which is also shown in Figure 2.

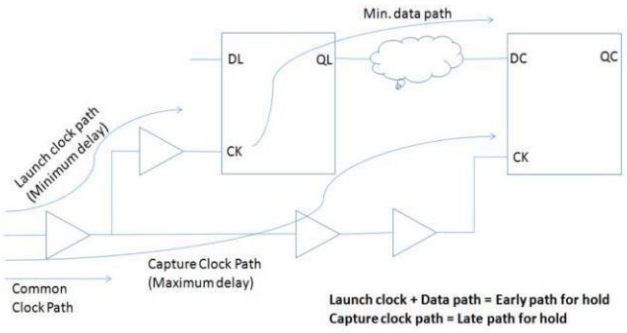

Figure 2. Derating on-chip variation for hold check [9]

To account for these violations, the following constraints are provided which will derate early/minimum paths by -10% i.e., the path is made more faster and will derate late/maximum-paths by +10% i.e the path is delayed more. set-timing-derate -early 0.9 set -timing-derate -late 1.1. To avoid hold violation the data path has to be multiplied with early derate (made faster) and capture path should be multiplied with late derate (delayed more).

2.3. Advanced on-chip variation

The concept of OCV modeling was based on fixed derating factor. This is a kind of pessimism. This can't be applied all the time. To overcome this the concept of Advanced On-Chip Variation (AOCV) is used. In AOCV, the derating factor isn't fixed whereas it is based on the depth and location of logic. AOCV differs from OCV where different derate values can be given for the cells.

a) Distance-based AOCV: This is used in modeling global variation effects. Different parts will have different variations. So as the distance increases variations also increases according to which derate value is decided.
b) Depth based AOCV: This is used in modeling local variation effects. These variations cancel out as the distance increases. So as depth increases the derate value decreases [10].

3. PARASITIC CORNERS

In Timing analysis has to be performed at various corners thus parasitic extraction is done at different corners. Interconnect wires in any design form two important devices in any circuit which are resistors and capacitors. This is because all the interconnect wires are modeled very nearer to each other [11].

3.1. Capacitance of interconnect wire

One of the devices formed by interconnect wire is a capacitor. Capacitance is dependent on various parameters such as Dielectric constant, Width of Metal (W), Thickness of Metal (T) Spacing between the metals (S) and Thickness of Dielectric (H) As the technology value decreases these parameters also decreases and the capacitance varies based on that [12].

3.2. Resistance of interconnect wire

All materials are associated with resistivity. Conductors have lower resistivity compared to semiconductors and insulators. Resistivity varies based on Temperature and the type of material. Resistance per unit length is given by ρ/ [(Thickness of metal) *(Width of metal)]. For a material at a particular temperature as the technology decreases the parameters such as width and thickness decrease and hence resistance increases which is not desired. Because circuit delay is directly proportional to resistance and if the delay value is high then it's impossible to design a high-speed circuit [13-16].

Hence to reduce resistance, resistivity can be made low by using different materials with lesser resistivity value. Interconnect parameters also depend on the number of drivers connected and the length of interconnect wire. High fanout indicates that the driver can drive more load and hence driver resistance is less (more current to drive more load) whereas low fanout indicates that the driver resistance is high. There may be 4 cases based on interconnect length and fanout of the driver as given in Table 1.

Table 1. Different cases for interconnect parameters

Driver Fanout	Interconnect wire length
High	Short
Low	Short
High	Long
Low	Long

Among the above cases only 2nd and 3 rd cases are important, because in the 1st case resources are not utilized properly and in the 4 the circuit doesn't work properly. The driver lacks in its driving capability till the end of the nets. Let Rd and Rw be the driver and wire resistance. In the 2nd case, Rd is greater than Rw whereas in 3rd case Rw is greater than Rd. Hence in 2nd case Driver resistance is dominant and in 3 rd Interconnect resistance is dominant. Driver resistance dominated paths mean Interconnect resistance is less which also means Interconnect capacitance is large. In terms of capacitance it is Interconnect capacitance dominated paths and the other is Interconnect resistance dominated path. Based on these there are two corners defined which are: C-best known as Cmin (minimum C, maximum R) and C-worst known as Cmax (maximum C, minimum R). There are other corners too which are RCbest, RCworst and typical.

a) *Cbest:* This refers to corners with minimum capacitance hence known as Cmin corner. Interconnect resistance value is higher than the Typical corner. This corner results in the shortest delay for paths with short nets and is used for min path analysis [17-20].
b) *Cworst:* This corner refers to those which has maximum capacitance. Hence known as Cmax corner. Interconnect resistance is smaller compared to the Typical corner. This corner results in the largest paths with short nets and is used for max path analysis.
c) *RCbest:* This corner refers to those which minimize interconnect RC product hence known as RCmin corner. This corner has the shortest path delay for paths with long interconnects and is used for min path analysis.
d) RCworst: This corner refers to those which maximize *the* interconnect RC product. Hence also known as RCmax corner. This corner has the highest path delay value for paths with long interconnects and is used for max path analysis.
e) *Typical:* This corner refers to the nominal value of interconnect capacitance and resistance. Timing analysis is performed at each different corner and violations are observed and corrected accordingly.

4. HSTA FLOW

Static Timing Analysis of any mixed-signal design is carried out using HSTA flow which is shown in Figure 3.

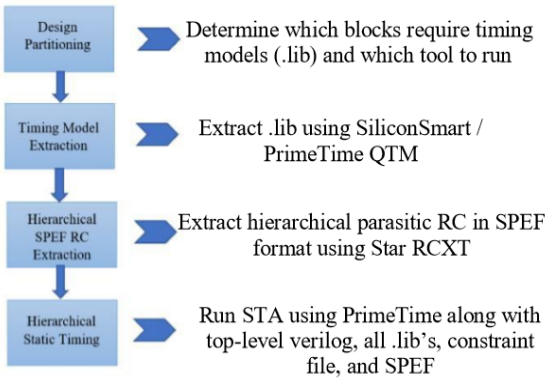

Figure 3. HSTA flow

The design is partitioned hierarchically and decision on which cells should undergo .lib and QTM generation is made. Parasitic are also extracted at few corners. With all of the above data along with the netlists of top-level cell and the cells for which parasitic are extracted, static Timing analysis is performed using either Synopsys Prime Time or Nano time [21-23].

4.1. Generation of .libs

Characterization of lower-level cells is done using SiliconSmart. The generated .libs has all the required timing information. .LIBS generated for one of the cell cDqOmux. The cell is characterized at one of the corners and the various parameters can be observed. The .lib contains a description of each pin and power supplies used.

4.2. Generation of QTM libs

Lib extraction for analog cells follows a separate procedure. Characterization of such cells is done using the Quick Timing Model. PrimeTime helps in generating QTM libs for Analog blocks. The QTM libs generated for analog block is as shown in Listing. 3. for one of the cells cZQCal. QTMs are considered as dummy .libs as they do not have any timing information. They only contain the direction of each pin of a cell with capacitance at each pin as shown in Listing 1.

```
cell( cZQCal ) {
  dont_use : true ;
  dont_touch : true ;
  interface_timing : true;
  timing_model_type : "qtm";

pin("iodrvoff") {
        direction : input ;
        capacitance : 0.005000 ;
} /* end of pin iodrvoff */

pin("sc_iddq") {
        direction : input ;
        capacitance : 0.005000 ;
} /* end of pin sc_iddq */

pin("sc_pwron") {
        direction : input ;
        capacitance : 0.005000 ;
} /* end of pin sc_pwron */

pin("sc_sm") {
        direction : input ;
        capacitance : 0.005000 ;
} /* end of pin sc_sm */

pin("sc_vrefEXTsel") {
        direction : input ;
        capacitance : 0.005000 ;
} /* end of pin sc_vrefEXTsel */
```

Listing 1. QTM lib of cZQCal cell

With all the .libs and QTM libs generated for lower-level cells in prior, timing analysis at the top level takes less time. All the netlists, parasitics and .libs are provided as inputs for either PrimeTime or Nano time to carry timing analysis [24, 25].

4.3. Analysis coverage

Analysis coverage of any design on performing timing analysis looks like as shown in Listing 2.

Type of Check	Total	Met	Violated	Untested
setup	10782	10054 (93%)	256 (2%)	472 (4%)
hold	10782	10012 (93%)	298 (3%)	472 (4%)
recovery	4341	0 (0%)	0 (0%)	4341 (100%)
removal	4341	0 (0%)	0 (0%)	4341 (100%)
min_pulse_width	40558	31798 (78%)	0 (0%)	8760 (22%)
clock_gating_setup	15	4 (27%)	0 (0%)	11 (73%)
clock_gating_hold	15	4 (27%)	0 (0%)	11 (73%)
out_setup	340	170 (50%)	170 (50%)	0 (0%)
out_hold	340	340 (100%)	0 (0%)	0 (0%)
All Checks	71514	52382 (73%)	724 (1%)	18408 (26%)

Listing 2. Analysis coverage report

It can be observed that 2% of the paths are violated due to setup check and 3% of the paths are violated due to hold check and similarly 4% of the paths are untested due to setup and hold check. All these violations should be corrected in-order to capture correct data.

5. CONCLUSION

Hierarchical Static Timing Analysis flow has helped in analyzing violations at each lower level and has helped in a great reduction of analysis time. The division of entire design to different levels and parasitic extraction separately for such levels skipping QTM and .lib cells for which timing information is already available reduces the execution time to a great extent. From the timing reports obtained, the violation paths can be easily identified which will usually be a register to register path. The slack value indicates the difference between the data arrival time and the time which was expected for the data to reach the input of the register. This value indicates the extent of the violation and will be usually in nanoseconds. Based on

whether it is setup violation or hold violation buffer cells whose cell delay corresponds to the slack value is added either in the data path or in a clock path. From this the data is exactly sampled at either rising or falling edge of clock based on the type of edge triggering. All the data is properly written or read from the memory without any loss of data. Thus, timing analysis plays a key role in any VLSI design.

REFERENCES

[1] Y. Shim and D. Oh, "Impact of dynamic power supply noise induced by clock networks on clock jitter and timing margin," in *2016 IEEE International Symposium on Electromagnetic Compatibility (EMC)*, Ottawa, pp. 62-66, 2016.

[2] K. B. Sowmya, Jose Alex Mathew, "Discrete wavelet transform based on coextensive distributive computation on FPGA," *Materials Today: Proceedings*, vol. 5, no. 4, Part 3, pp. 10860-10866, 2018.

[3] Sowmya K.B., Mathew J.A, "Adept-disseminated arithmetic-based discrete cosine transform," In: Abraham A., Dutta P., Mandal J., Bhattacharya A., Dutta S. (eds) *Emerging Technologies in Data Mining and Information Security*, Advances in Intelligent Systems and Computing, vol. 755. Springer, Singapore, 2019.

[4] L. Thomas and V. Kiran, "Timing convergence techniques in digital VLSI designs," in *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, Chennai, pp. 2882-2886, 2017.

[5] H. Kwon and Y. H. Kim, "Calculating the probability of timing violation of F/F-controlled paths with timing variations," *2016 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)Jeju*, pp. 514-517, 2016.

[6] "Tips and techniques on how to ensure high quality of results in static timing analysis," *London Jin Toshiba America Electronic Components, Inc.*, SNUG San Jose 2000.

[7] "AN 554: How to Read HardCopy PrimeTime Timing Reports," ALTERA, Mar. 2010.

[8] "Synopsys, SiliconSmart ACE User Guide," Version L-2016.06-SP1, Sep. 2016.

[9] "Synopsys, PrimeTime User Guide," Version O-2018.06-SP4, Dec. 2018.

[10] Lily Aggarwal, "Static timing analysis of custom mixed-signal circuits," *SNUG Silicon Valley 2018 Proceedings*, 2018.

[11] "Using the Synopsys® Design Constraints Format Application Note," Dec 2017.

[12] Y. Blaquiere, M. Dagenais and Y. Savaria, "Fast timing analysis of VLSI circuits: a dynamic and hierarchical approach," *1991 IEEE International Sympoisum on Circuits and Systems*, vol. 4, pp. 2398-2402, 1991.

[13] Krishnan-talkad Sukumar, Naveen Chandra Srivastava, Hariprasad TT, Pankaj Kumar, David Newmark, Mitch Sengson and Nagarjuna Vallamkonda, "Determining the Preferred Methodology for Timing Analysis and Signoff of Mixed Signal Designs and Timing Methodology Correlation Challenges in FinFET technology nodes," in *SNUG Silicon Valley 2018 Proceedings*, 2018.

[14] Venkatnarayan Hariharan, Satish Damaraju, Aaron Deno, Harini Suresh and Iqbal Rajwani, "Improved methodology for modeling clock skew in hold analysis," *SNUG Silicon Valley 2017 Proceedings*, 2017.

[15] Sowmya K. B., Deshpande N., Mathew J. A., "Implementation of Kernel-Based DCT with controller unit," In: Fiaidhi J., Bhattacharyya D., Rao N. (eds) *Smart Technologies in Data Science and Communication*. Lecture Notes in Networks and Systems, vol. 105. Springer, Singapore, 2020.

[16] N. P. Jouppi, "Timing analysis for nMOS VLSI," in *20th Design Automation Conference Proceedings*, 1983, pp. 411-418.

[17] B. Korte and J. Vygen, *Combinatorial Optimization: Theory and Algorithms*, Berlin:Springer-Verlag, 2006.

[18] S. Kourtev and E. G. Friedman, *Timing optimization through clock skew scheduling*, MA, Boston:Kluwer, 2000.

[19] H. Schneider and M. H. Schneider, "Max-balancing weighted directed graphs and matrix scaling," *Math. Oper. Res.*, vol. 16, no. 1, pp. 208-222, 1991.

[20] C.-Y. Yeh and M. Marek-Sadowska, "Delay budgeting in sequential circuit with application on FPGA placement," in *Proceedings of the 40th annual Design Automation Conference*, pp. 202-207, 2003.

[21] N. E. Young, R. E. Tarjan and J. B. Orlin, "Faster parametric shortest path and minimum balance algorithms," *Networks*, vol. 21, no. 2, pp. 205-221, 1991.

[22] T.-W. Huang and M. D. Wong, "UI-timer 1.0: An ultrafast path-based timing analysis algorithm for CPPR," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 11, pp. 1862-1875, 2016.

[23] P.-Y. Lee, I. H.-R. Jiang and T.-C. Chen, "Fastpass: fast timing path search for generalized timing exception handling," in *2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2018, pp. 172-177.

[24] S. Chen Hu, X. Zhao and X. Chen, "TAU 2016 contest on macro modeling," in *2016 ACM International Workshop on Timing Issues in the Specification and Synthesis of Digital Systems*, 2016.

[25] T.-Y. Lai, T.-W. Huang and M. D. Wong, "LibAbs: An efficient and accurate timing macro-modeling algorithm for large hierarchical designs," in *Proceedings of the 54th Annual Design Automation Conference 2017*, 2017, pp. 1-6.

BIOGRAPHIES OF AUTHORS

Sowmya K B received the B. E degree in Electronics and Communication Engineering from the Vivekananda College of Engineering and Technology, Puttur, India, in 2006, M.Tech degree in Electronics from the Sir. M. Visveswaraya Institute of Technology, Bengaluru, India, in 2012, bagging rank from VTU, Belagavi. She was working as Assistant Professor in PA College of Engineering, Mangaluru, India for Nine years. Currently she is working as Assistant Professor in R V College of Engineering, Bengaluru, India. Her research interests include VLSI and Signal Processing, Image processing, Low power Architectures and VLSI Design. She has contributed many National and International journal papers to various reputed journals and conference.

Thanushree M is doing her Bachelor of Engineering in Electronics and Communication Engineering Department, RV College of Engineering, Bengaluru, India.