Evaluation of Dual Rail Complete Detection Circuitry using Asynchronous Delay Insensitive Frameworks

J. Sudhakar
e-mail: sudhakar.jyo@gmail.com

A.L. Durga
e-mail: aldurga@gmail.com

K. Sushma
e-mail: koyeladasushma@gmail.com

Department of ECE
Vignan's Institute of Engineering for Women
Vishakapatnam-A.P
India.

***Abstract* - This paper proposes a testable design of Delay-Insensitive nanoscale framework using different registers such as normal Sleep Convention Logic (SCL) register, modified SCL register and SCL scan cell. Combinational logic blocks in nanoscale frameworks cannot rouse until a complete DATA set is accessible at the input of their preceding register, input-completeness to DATA is also unequivocally ensured. These are characterized in terms of speed and power. Dual rail pipelined versions are developed, and those comparisons are carried out by several parameters mainly focus on power dissipation, delay, slew rate, rise time and fall time.**

Keywords - sleep convention logic, dual rail delay insensitive multi-threshold null convention logic, delay, power dissipation, slew rate.

I. INTRODUCTION

Over the traditional synchronous circuits delay insensitive paradigms, like SCL [1], MTNCL, and MTD³L have been the target of rehabilitated research effort for the advantages which are likely reduced power dissipation, delay, and robustness to noise[2]. The delay-insensitive method could be sorted into numerous sub-categories such as pure Delay Insensitive (DI), Quasi-Delay Insensitive (QDI) and Speed Independent circuits (SI) [2]. In QDI circuits [1], delays can be unbounded. To achieve delay insensitive behaviour, SCL uses an expression that depends on the associations of the ciphers present in the expression without a reference to their time of valuation [3] [4]. In particular, dual-rail, quad-rail signals can be used to integrate data into one miscellaneous signal path to eradicate time reference [5] [6].

The most popular delay insensitive encoding is dual-rail, D consists of two wires D_0 and D_1, which may presume any value from the set {DATA0, DATA1, NULL}.

TABLE 1: LOGIC VALUES OF DUAL RAIL ENCODING

Dual-Rail Encoding (D_0, D_1)	Logic Value(D)
(0,0) (1,0) (0,1) (1,1)	NULL DATA0 DATA1 INVALID

D is Boolean logic 0 corresponds to DATA0 when D_0=1 & D_1=0, is logic 1 corresponds to DATA1 when D_0=0 & D_1=1 and the NULL state corresponds to the empty set when D_0, D_1=0 [7].

The rest of the paper organized as follows: Section II presents different types of registers, focusing on performance metrics. Section III describes various methodologies in order to design delay insensitive framework. The obtained results are presented in section IV. Section V concludes the paper.

II. LITERATURE REVIEW

In this paper, we are constructing delay insensitive frameworks using sleep convention logic register, scan cell and modified sleep convention logic registers and also demonstrating the performance metrics like power dissipation, delay, slew rate, rise time and fall time. Register block is used to harmonize contiguous computational blocks to ensure the exact data communication. It controls the DATA/NULL wave fronts [4].

A. Sleep Convention Logic (SCL) Register:

The implementation of single bit dual rail SCL register in transistor level is shown in figure 1. Where I_0 is input rail and O_0 is output rail [9]. A single bit SCL register is comprised two inverters with a feedback path. They can pass a DATA value from input to output when K_i is requested for data (high) and pass NULL when K_i is requesting for null (low). K_i is the request signal which comes from the output of the subsequent stage of completion detection circuit.

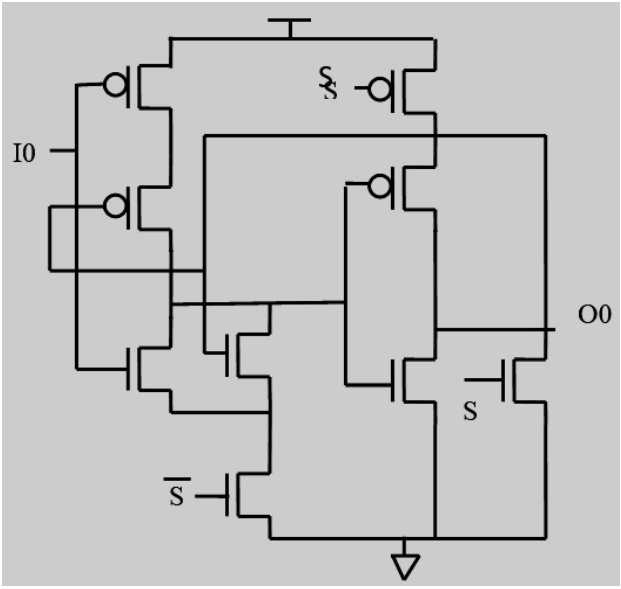

Figure 1: SCL register for single rail.

B. Scan Cell:

Figure 2 shows the implementation of SCL scan cell. The realization is made of two D-latches, one of them is original SCL register and another one is modified SCL register [10]. In this register, Din is the main input, and Dout is the output of the scan cell. M is the test mode selection signal. When M=1, scan cell enters in test mode. When M=0, scan cell is in normal mode. The scan cell works exactly like SCL register in normal mode; but in test mode, it works as Level Sensitive Scan Design (LSSD) type scan cell [11]. Through the non overlapping clock signals (CL0 & CL1) the data can be shifted from Din to Dout. Lastly, scan cell is in normal mode when sleep signal puts the register in sleep mode.

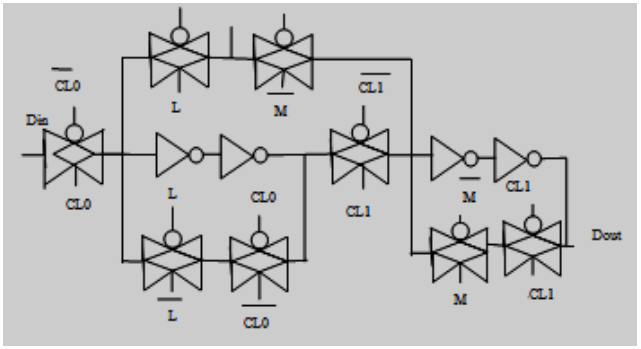

Figure 2: Scan Cell.

C. Modified SCL Register:

The transistor level implementation of Modified SCL is shown in figure 3. It consists of three additional transistors to cut the feedback path [12].

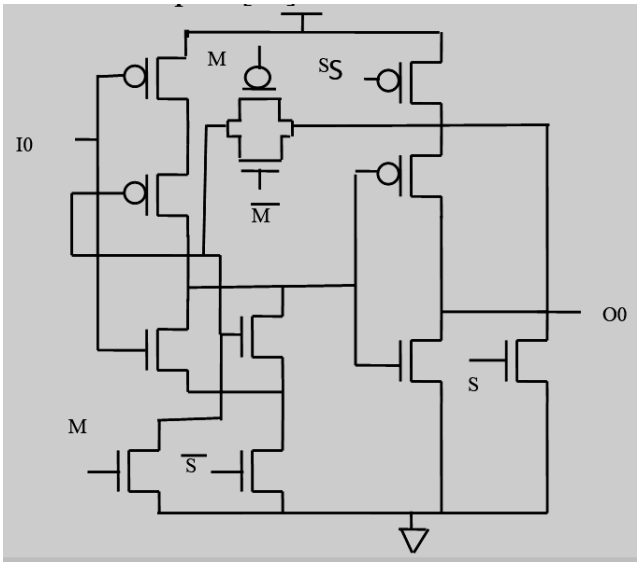

Figure 3: Modified SCL register

In this paper, we are signifying the several parameters like power dissipation, delay, slew rate, rise time and fall time. These are valuable to recognize the preeminent methodology in delay insensitive nanoscale framework [13].

III. PROPOSED WORK

Over the NCL circuits, SCL & MTD³L circuits have several advantages. Those are: reducing the power consumption due to the high threshold voltage transistors and reducing input completeness, results in a significant area reduction & boosts the performance of the circuits.

In our paper, we propose the nine different methodologies in order to obtain delay insensitive framework with complete auto detection feasibility. The framework is as shown in figure 4. In these methodologies, each pipeline stage contains a Functional block (Fi), Register block (Ri), Completion detection block (CDi) and Ci element (F.A. Parsan, J. Zhao, S.C. Smith, 2014.)

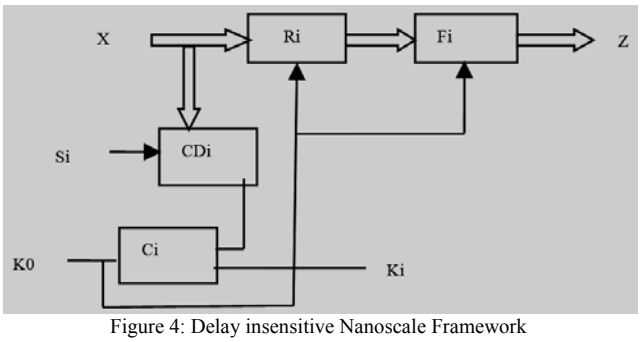

Figure 4: Delay insensitive Nanoscale Framework

A. SCL Functional Block:

Functional blocks in SCL are comprised of a family of

threshold gates. SCL gate consists of a set block & a hold0 block [9].

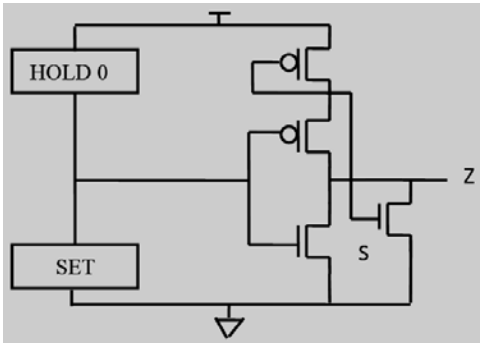

Figure 5. SCL gate Implementation

In which reset & hold1 blocks are removed. Reset block functionality is performed through the sleep mechanism and hold1 block job was to add hysteresis to the gates in order to ensure input completeness. For example, TH23w2 has three inputs (A, B, C) & its threshold is 2 with weight 2, as shown in fig 6. The Boolean expression of the TH23w2 gate can be described as A+BC.

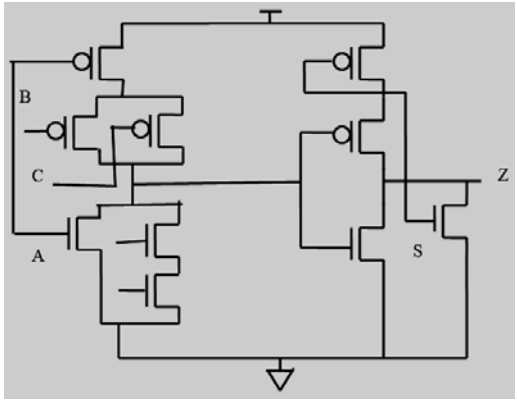

Figure 6: Implementation of TH23W2 SCL

B. MTNCL Functional Block:

NCL gates consume more power & delay to implement Boolean functions. During NULL cycle, this technique is to set the functional block in sleep mode [10]. From the output signal of the previous register stage, the sleep transistors can be demoralized globally. During active mode, the NCL gate produces DATA outputs throughout RESET block. It requires more transistors than SCL per stage.

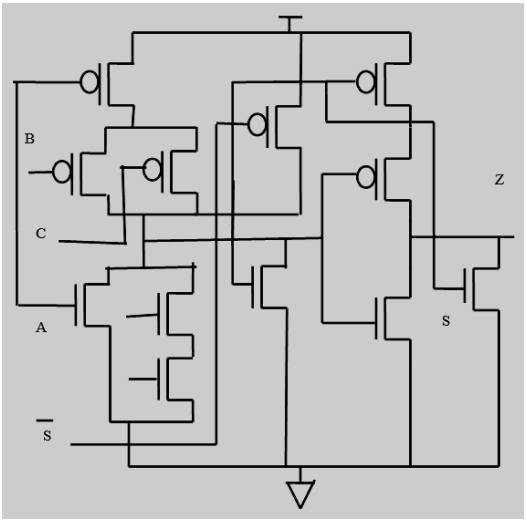

Figure 7. Schematic of TH23W2 MTNCL

The implementation of TH23W2 MTNCL is shown in fig 7 whose inputs n=2, the threshold of the gate m=3. All the threshold gates will become logic 0 when hold1 circuit put into sleep mode, so no longer hysteresis required [11].

C. MTD^{3}L Functional Block:

D^{3}L methodology lucratively implements the dual spacer dual rail delay insensitive protocol, it suffers from high overhead [12]. To eradicate those, MTNCL can be pooled with D^{3}L, to form MTD^{3}L [13]. When S0 is asserted, the circuit sleeps to all zero state. When S1 is asserted, the circuit is going to all one spacer. If both sleep signals are to be asserted at once, then the circuit will not be operating properly [14]. Different methodologies in order to obtain delay insensitive framework with complete detection feasibility are shown in Table 3.

TABLE 2: MTD^{3}L SLEEP SIGNALS

Sleep Signals (S0, S1)	Output (z)
(0,0)	Normal
(1,0)	All Zero Spacer
(0,1)	All One Spacer
(1,1)	Invalid

Similar to the MTNCL, it requires SET & HOLD0 blocks. The amendment required is the addition of sleep transistors[15]. These transistors are prescribed by a pair of sleep signals S0 and complement of S1 (ns1).

The circuit will sleep to the appropriate value if either of the inputs is asserted. Fig 8 shows the MTD^{3}L design [16].

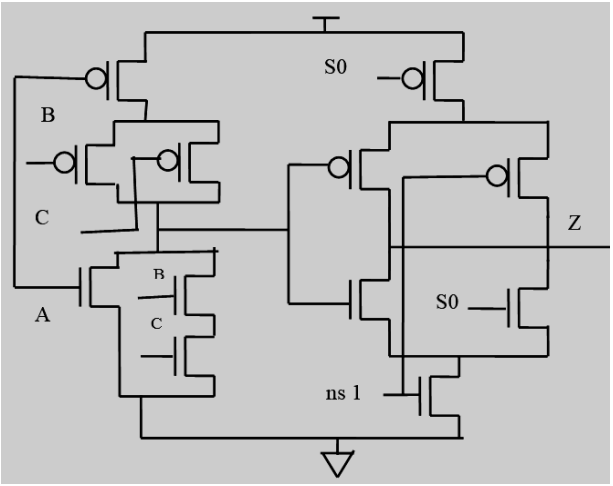

Figure 8: Implementation of TH23W2 MTD^{3}L

Figure 9 shows a typical SCL pipeline with two primary inputs named as A & B and two primary outputs named as Y & Z. This functional block diagram consists of Functional block, scan cells, completion detector and C element[17].

TABLE 3: FRAMEWORK METHODOLOGIES

Version	Functional Block	Register Block
1	SCL	SCL
2	SCL	Scan Cell
3	SCL	Modified SCL
4	MTNCL	SCL
5	MTNCL	Scan Cell
6	MTNCL	Modified SCL
7	MTD^{3}L	SCL
8	MTD^{3}L	Scan Cell
9	MTD^{3}L	Modified SCL

As we have seen in traditional scan chain design, scan cells here form separate long shift registers in test mode [18]. Here that the test vectors can be shifted in and results are shifted out.

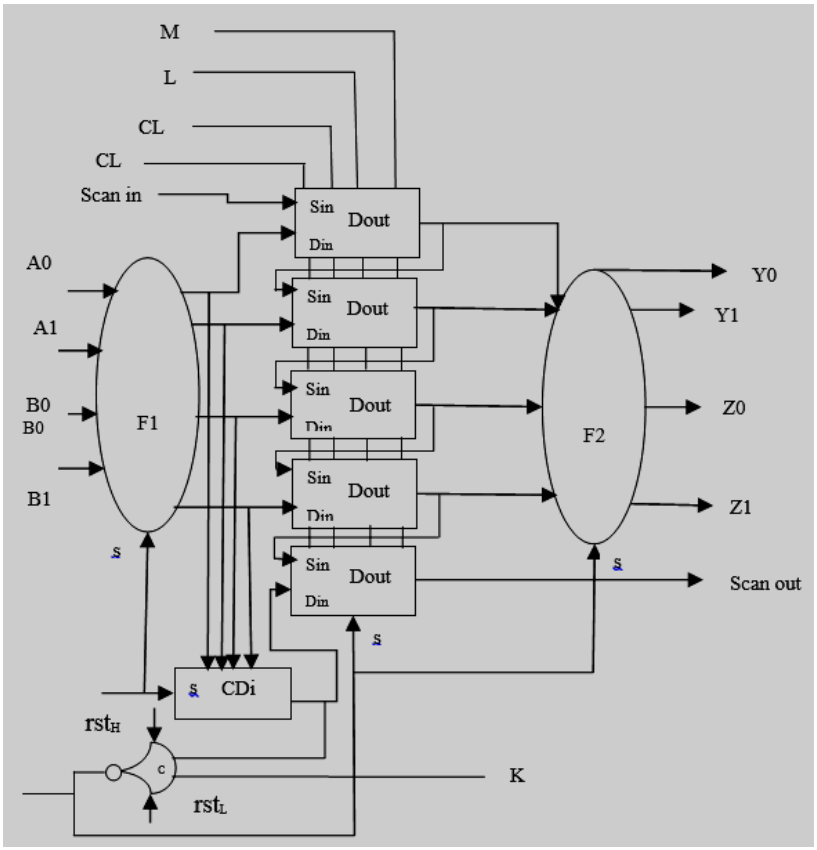

Figure 9: Architecture of SCL scan chain

Different methodologies in order to obtain delay insensitive framework with complete detection feasibility are shown in below figures. We implemented version 1 using SCL functional block & SCL register block in Mentor Graphics tool. In this implementation, we use four blocks they are register block, functional block, completion detector & C – element. The function of the register is to hoard the data as per the output of the circuit and it is worked with the acknowledgment signals.

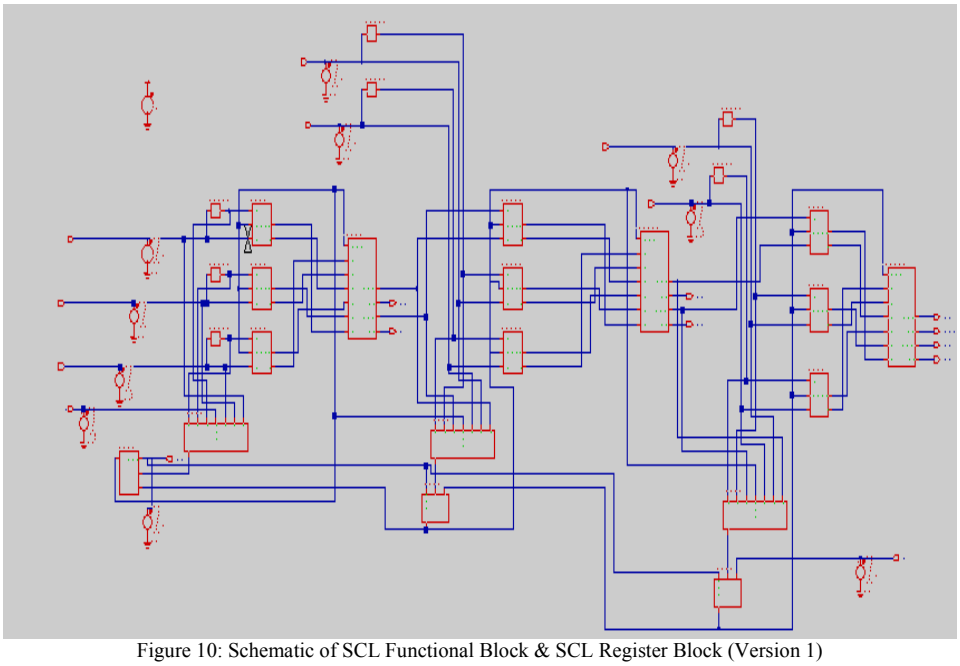

Figure 10: Schematic of SCL Functional Block & SCL Register Block (Version 1)

The outputs of the register block are given as inputs to the functional block. The inputs that are given to the register block are fed to the completion detector as inputs. Lastly, the complete framework generates outputs.

Version 5 framework is designed by using MTNCL functional block & Scan cell register. It is established by the aid of full adder. The complete architecture functionality is performed by dual rail only. This implementation is developed in mentor graphics tool. In this version, the SCL registers are replaced with scan cells & the SCL functional blocks with MTNCL.

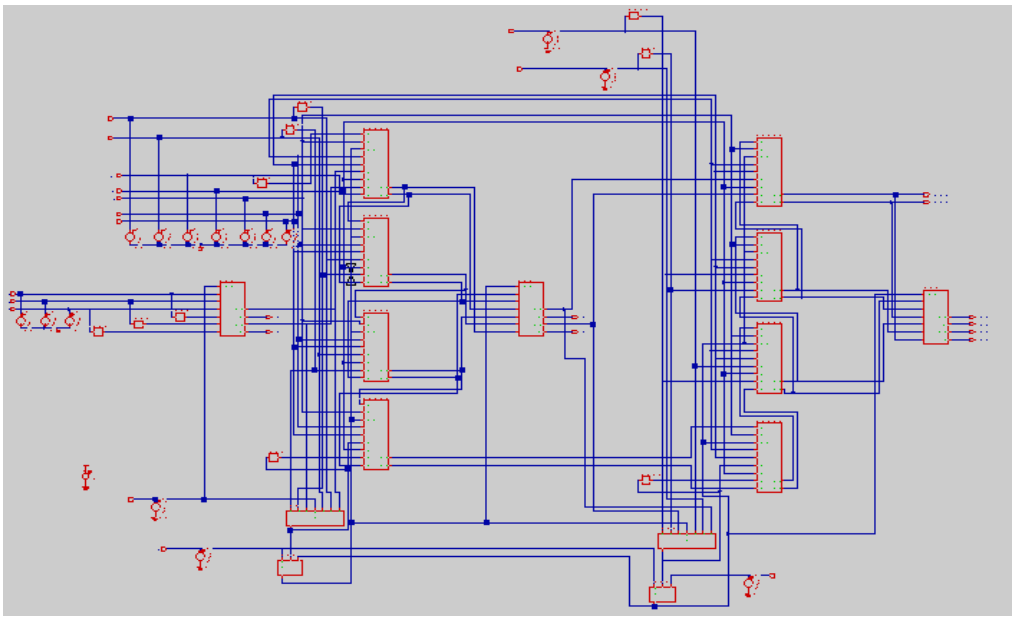

Figure 11: Schematic of MTNCL Functional Block & Scan Cell Register (Version 5)

In which, the outputs of the completion detector and functional block are given as inputs to the scan cells. In order to crack the problem, here we are accumulating an extra scan cell to the completion detector because the output of this block is not readily accessible for surveillance. Therefore the area overhead is abandoned due to this the power is reduced in this version compared to all other versions.

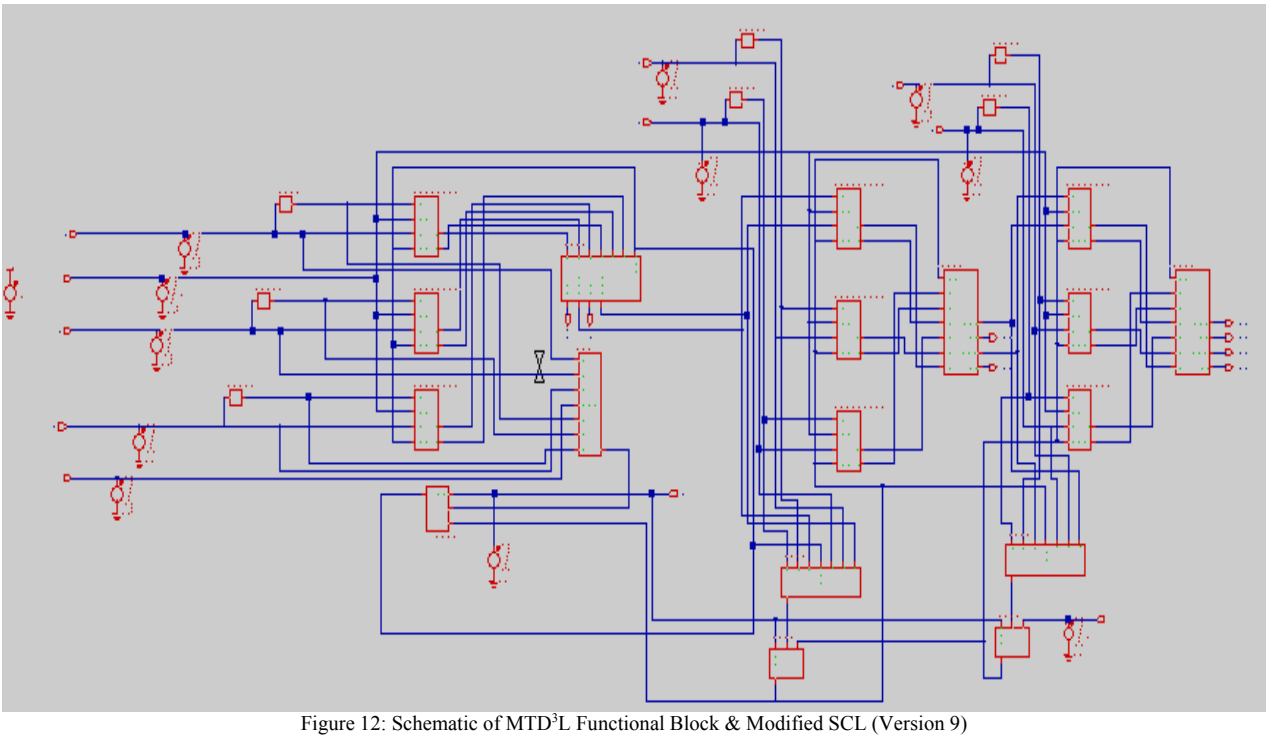

Figure 12: Schematic of MTD^{3}L Functional Block & Modified SCL (Version 9)

In the implementation of version 9, we use the combinational block as MTD^{3}L & register block as modified SCL. This adapted SCL register makes use of three supplementary transistors to engrave the feedback path. When M =1 the register appear as an inverter.

D. Completion Detection (CDi):

Completion Detection uses N, K0 lines to detect the complete DATA/NULL wavefronts at the output of every stage & request the next NULL & DATA set, respectively. The output of the completion component is connected to all Ki lines of the previous register for full word completion. For an N bit register, the number of logic levels in the completion component is $\log_4^N$. Hence the maximum input threshold gate is the TH44 gate.

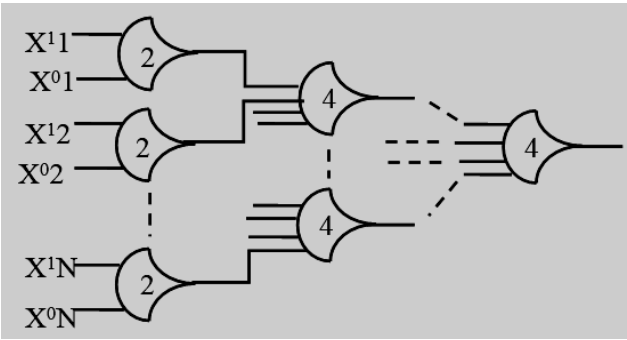

Figure 13: Completion Detector.

E. The C- Element:

To synchronize the sleep signals, the resettable C – element with inverted output is required [9]. Therefore, it cannot be put to sleep during pipeline operation.

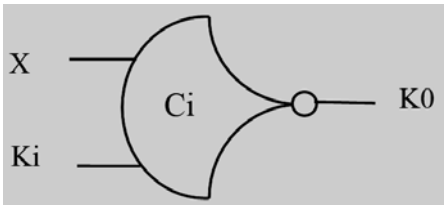

Figure 14: C – Element

This C- element is the only gate in the pipeline that does not have sleep capability. A C- element is working as: the output is asserted when all the inputs are asserted; the output is relics asserted until all inputs are reasserted.

IV. Results & Discussions

In this paper, delay insensitive nanoscale framework using different versions are designed to evaluate the delay, power dissipation, slew rate, rise time and fall time. Table 4 shows the comparison of proposed methodologies. Each version has been analyzed in terms of power, delay, slew rate, rise time, fall time and values are measured.

Power is a foremost characteristic and it is clear that power dissipation has greatly reduced in version 4 compared to other versions. Thereby the version 4 circuit

will operate without any leakage problems & produce outputs accurately. Rise time is an analog parameter of essential significance in low power and high-speed applications. Therefore, it is a assess of the potential of a circuit to act in response to prompt input signals. We can observe that the rise time is high in version 8. Fall time is also known as pulse delay time. It is the time taken for the amplitude of a signal to lessen from a precise value. The fall time is high in Version 8. It reduces noise &glitches. From table 4 we noticed that, compared to all versions, version 2(SCL functional block & scan cell register) offers the best delay & achieves a higher speed of operation. Version 4 (MTNCL functional block & SCL register) gives finer performance compared to other versions in terms of power & version 9(MTD3L functional bock & modified SCL register) provides superior performance in terms of slew rate. So, these are the better versions to design delay insensitive nanoscale framework.

V. CONCLUSION

From the analysis of the above various types of delay-insensitive frameworks, we can reach to a conclusion that the power dissipation is low for version 4 (MTNCL functional block & SCL register) and delay is stumpy for version 2(SCL functional block & Scan cell register). So version 2 & version 4 are the best methods to optimize the power and delay. In addition to these, version 9 (MTD^{3}L functional block & Modifies SCL register) is also a preferred alternative because of high slew rate. In version 8(MTD^{3}L functional block & Scan cell register) we can improve both rise time & fall time, so it can also be a good choice. Combinational block and register block are the important blocks in these in these frameworks.

TABLE 4. COMPARISON OF DIFFERENT VERSIONS OF FRAMEWORKS

Version	Power (µW)	Delay	Slew rate	Rise time (ps)	Fall time (ps)
1	9.33	27.64ms	337.46 v/s	42.58	3.45
2	30.80	3.01µs	10.21 v/s	163.24	19.49
3	10.31	30.99ms	332.64 v/s	42.38	3.22
4	8.31	28.27ms	293.88 v/s	44.75	3.95
5	29.82	41.26s	722.67 v/ms	60.31	25.52
6	9.33	32.38ms	288.10 v/s	48.75	4.67
7	10.36	0.37s	27.75 v/µs	95.61	170.32
8	32.01	1.67s	19.08 v/µs	372.36	751.56
9	11.93	0.35s	33.65 v/µs	105.57	211.36

ACKNOWLEDGMENT

This research was supported by Science & Engineering Research Board (SERB), Department of Science & Technology (DST), Government of India, under ECRA grant (ECR/2017/000142)

REFERENCES

[1] J.Di and S. C. Smith, "ultra-low power multi-threshold asynchronous circuit design," U.S patent, No.7977972 (2011).
[2] Farad A. Parson, Scott C.Smith "Design for testability of sleep convention logic", IEEE Transactions on VLSI 10.1109/TVLSI.2015.2419816.
[3] S.C. Smith "Designing asynchronous circuits Using null convention logic," synth lect digital. Circuit syst, vol4, no.1. 2009.
[4] J. McCardle, and D. Chester, "Measuring an asynchronous processor's power and noise," in Proc. Synopsys Users Group Conf. (SNUG), 2001, pp. 66–70.
[5] S.C. Smith, R.F. DeMara, J.S. Yuan, M. Hagedorn, and D. Ferguson, "Delay-insensitive gate level pipelining," Integration, the VLSI Journal, Vol. 30/2 pp. 103-131, October 2001.
[6] J. Sudhakar, K.Sushma, " Energy efficient IEEE 754 floating point multiplier using dual spacer dual rail delay insensitive log (MTD3L)", Circuit World, Vol. 43, no. 2, pp. 72-79, April 2017.
[7] K. Sushma, J. Sudhakar " Energy-aware IP shifter for DSP processors using MTD3L asynchronous approach", IOSR – JVSP, vol 6, issue 6, Nov – Dec 2016.
[8] Gerald Edward Sobelman, "Null Convention Logic with flash, set and reset capability", US 6043674 A, 2010.
[9] Ortega, C.; Tse, J, Manohar, R., "Static power reduction techniques for asynchronous circuits," Asynchronous circuits and systems (ASYNC), 2010 IEEE Symposium on, Vol., no., pp,52,61, 3-6 May 2010.
[10] M. Shams, J.C. Ebergen, and M.I. Almasy, "Modeling and comparing CMOS implementations of the C-element," IEEE Transactions on VLSI Systems, Vol. 6, no. 4,pp. 563-567,1998.
[11] E. Yahya and L.Fesquet, "Asynchronous design: A Promising paradigm for electronic circuits and systems." The 16th IEEE International Conference on Electronics, circuits, and systems, p.339 (2009).
[12] C.L.Seitz, .System timing, An introduction to VLSI systems, Addison Wesley, pp.218-262, 1980.
[13] T.Verhoff, Delay insensitive codes. An overview, Distributed computing, Vol.3, pp. 1-8, 1998.
[14] Gerald E. Sobelman and Karl M. Fant, CMOS circuit design of threshold gates with hysteresis, IEEE International Symposium on circuits and systems (II), pp.61-65, 1998.
[15] M. Shams, J. C. Ebergen, and M. I. Elmasry, "Modeling and comparing CMOS implementations of the C-element," Very Large Scale Integ. (VLSI) Syst., IEEE Trans. on, vol. 6, no. 4, pp. 563–567, Dec. 1998.
[16] S. Yancey, and S. C. Smith, "A differential design for C-elements and NCL gates," in Circ. and Syst. , 53rd IEEE Int. Midwest Symp. on, Aug. 2010, pp. 632–635.
[17] Harish Gopalakrishnan, "Energy Reduction for Asynchronous Circuits in Soc Applications", Ph.D. dissertation, Wright State University, 2005.
[18] F.A. Parsan, J. Zhao, S.C. Smith, "SCL Design of a Pipelined 8051 ALU", Circuits and Systems (MWSCAS),2014 IEEE 57th International Midwest Symposium on, Aug 2014.
[19] David Rocklin, "Optimum Transistor Sizing of MTNCL Threshold Gates for Various Design Constraints", M.S thesis, University du Maine, 2008