International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 4 (2015) pp. 11279-11295

http://www.ripublication.com

Enhanced Architecture for Digitally Controlled Delay Lines

[1]D.Preethi, [2] Dr. (Mrs.) R.Brindha
[1] P.G Scholar / II M.E.VLSI DESIGN, [2] Professor& Head of the Department
[1],[2] Department of Electronics and Communication Engineering
Avinashilingam Institute for Home Science and Higher Education for Women-University,
Coimbatore, India
[1]prettz_d@yahoo.in

Abstract

Digitally Controlled Delay Lines (DCDLs) are used for producing time reference to the movement of data and are intended to introduce delay in any clock distribution network. The variable delay of DCDLs can be controlled by using thermometric code obtained from time to digital converter. Since the conventional NAND based DCDLs caused glitches, architectures to eliminate the glitches were designed. Even though there were no glitches in such lines, the area and power were more due to the presence of large number of NAND gates in constructing a single delay cell. Hence multiplexer based DCDLs drew attention owing to their reduced area and power. In this paper, the architectures with further reduction in transistor count are proposed for both NAND and Multiplexer based DCDLs.

Keywords: NAND based DCDLs, Enhanced Transistor Count Architecture, Thermometer Code, Time to Digital Converter

Introduction

Delay lines can be widely categorized into two major sub-divisions namely the analog and digital [5]. The three main sub divisions of analog delay lines are shunt capacitor, current starved and variable resistor [7]. They are much suited for fine grain architecture. Any logic gate having a propagation delay can be used in such places but the difficult part was in programming specific delay until the use of NAND based delay lines. The constituents of such delay lines are voltage controlled delay element and a comparator circuit. The operation of shunt capacitor is shown in Figure 1.

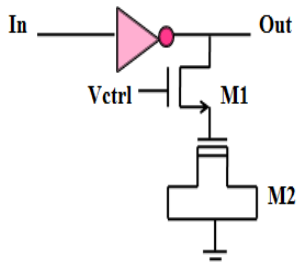


Figure 1: Shunt Capacitor [5]

In Figure1, the transistor M2 acts as a capacitor and the transistor M1 is used for controlling the charging and discharging of current in M2. Whenever the control voltage Vctrl is high, the transistor M1 switches ON and the input gets stored in the capacitor M2, thereby introducing a delay between the input and output. On the contrary when delay is not necessary the Vctrl is set to low so that the transistor M1 is OFF and hence the input reaches the output without any delay. This circuit suffers from a drawback of occupying large area and has fixed delay. Therefore digital delay lines are preferred when compared to analog delay lines.

The advantages of digital over analog are predictable and variable delay, improved speed and simple design [3], [13]. In digital electronics, digital delay lines have series of delay elements of variable length which determine the delay. A digital control code is given to the sequence of delay elements coupled with the clock to achieve a synchronous delay. Any logic gate can be used as a digital delay element but the difficult part was in programming variable delay. The conventional inverter based delay line suffered from certain limitations [6] such as: a) when the number of delay cells increases, larger delays cannot be realized and b) the resolution decreases when the frequency increases. Since the delay lines were constructed using deep sub-micrometer technology, definitely the frequency is high due to faster switching rates [4]. Hence NAND [1] and multiplexer [2] based delay lines are of prime choice. The logical effort of an inverter is one but that of a NAND gate is (n+2)/3. The value of logical effort decides the complexity of the circuit. Though the complexity of inverter is less when compared to that of NAND, they overcame the shortcomings of inverter based delay lines.

The initial NAND based delay lines suffered from glitches due to multiple path propagation which was eliminated in the architecture explained in section II. A glitch is an unwanted pulse [1] at the output of a combinational network. A circuit prone to glitches is said to be a hazard and are classified into three types namely static hazard, dynamic hazard and functional hazard. Since the architecture that eliminated the glitches in NAND based DCDLs occupied more area and power, multiplexer based DCDLs were preferred. In this work, the extension of the glitch free architectures is proposed for reduced transistor count. This method occupies comparatively less area and power.

In section II a brief review about the existing technique is discussed followed by the proposed work in section III where the reduced transistor count architecture is

designed and their operation is also explained. The section IV entails upon the simulation results and analysis and in section V, the entire work is being reviewed and concluded.

Existing Methodology

A. Thermometric Code

The thermometric code [1], also called the unary code depends on a single parameter. The pattern of this code is either rising zeros or rising ones. The table I represents the rising zeros. This code is given as a control code to the digital delay lines for controlling the variable delay. This code is generated using the time to digital converter [12] on measuring the time interval between the two instances with reference to the clock. For example in Figures 2(a) and 2(b), the control code is given as 'S_i'.

Table 1: Thermometer Code

T	Thermometer Code
1	0111111111
2	0011111111
3	0001111111
4	0000111111
5	0000011111
6	0000001111
7	0000000111
8	0000000011
9	00000000011

B. Glitches in NAND-Based DCDL

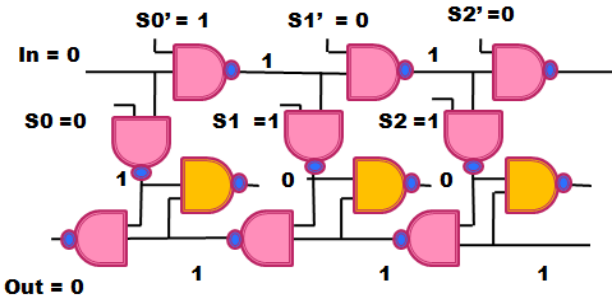
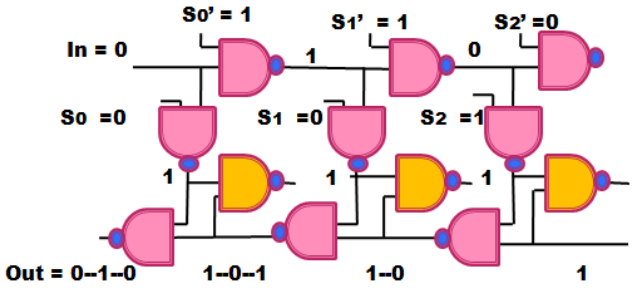


Figure 2: Glitching in Non- Inverting Toplogy (a) When Delay Path is One [1]



(b)

Figure 2: Glitching in Non- Inverting Toplogy (b) When Delay Path increases by One [1]

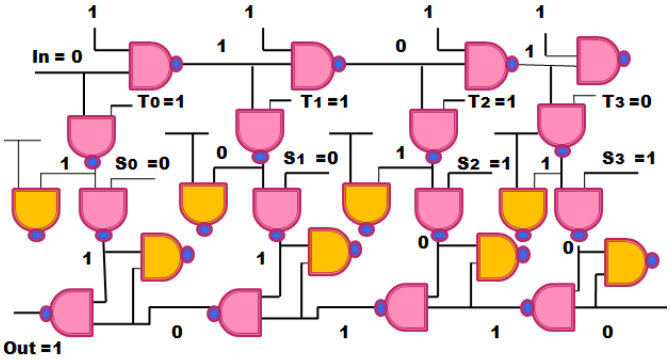
In the traditional NAND based DCDLs, delay was produced as intended but there were occurrence of glitches when more than one delay cell was included. The delay elements (NAND) are connected in series forming a delay line and they combine to form a delay path. In the following DCDLs, one of the inputs of the lower delay element is stuck-at-1.

When the Delay Path is one: The Figure 2(a) shows the glitching activity when the delay path is one; S_0:0, S_1:1 by the condition that S_i= 0 for i < c and S_i= 1 for i $\geq$ c where S represents the control bit, i is any integer varying from 1 to n and c is the delay control code based on which the thermometric code bits will be generated as shown in table 1 and are applied to the S bits [1]. For example since the value of c=1 in the Figure 3(a), S bits are 01. In this case there is no occurrence of glitches because only one path leads the input to output but the problem arises when multiple path propagation is involved.

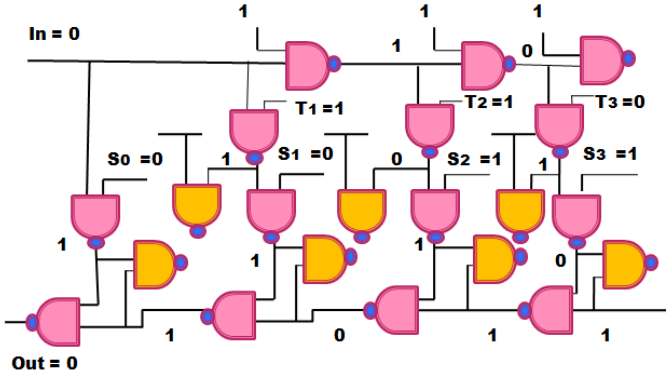
When the Delay Path increases by one: The Figure 2(b) shows the glitching activity when the control code is increased by more than one. The control bit at this stage would be S_0:0, S_1:0, S_2=1. Hence there will be momentary change in the output causing a glitch [1] to occur due to multiple path intrusion between the input and output. For instance if the input is '0' the output should have remained at 0 in non-inverting topology but it goes to 1 and then returns to 0. Therefore the intermediate signals contribute to glitch. The dotted lines represent the delay path.

C. Glitch Free NAND based DCDL

This architecture eliminated the glitches from the above delay line by introducing an additional control bit T_i with the conditions: S_i= 0 for i < c and S_i= 1 for i $\geq$ c ; T_i =1 for i$\neq$ c+1 and T_{c+1} =0 [1]. The last lower DE cell is stuck at '1' for non-inverting topology and stuck at '0' for inverting topology instead of the entire lower delay elements to be stuck at a value as in the previous case. When the control code word is two, the control bits are $S_{0\text{-}1}$:0, $S_{2:}$1, $T_{0\text{-}2}$:1, $T_{3:}$0. Here as the input is given '0' the output from the inverting topology is '1'as shown in Figure 3(a). The Figure 3(b) illustrates the operation of non-inverting topology where the Out is'1' for the In='1'. A non inverting delay cell topology is placed in front of the inverting topology.



(a)



(b)

Figure 3: Glitch-Free NAND-based DCDL (a) Inverting Topology [1], (b) Non-Inverting Topology[1]

1. Driving Circuit for Control-Bits Si and Ti:

The Figure 4 shows the driving circuit of the glitch free DCDL [1],[9], [10]. The thermometer code is transmitted through the driving circuit and their output is passed on as the control bit to the delay element. There are two delay flip flops for the two control bits Si and Ti and three NAND gates for any one of them in order to introduce three NAND gate delay. There is a $3xt_{nand}$ delay by using a three NAND gates for the control bit Si.

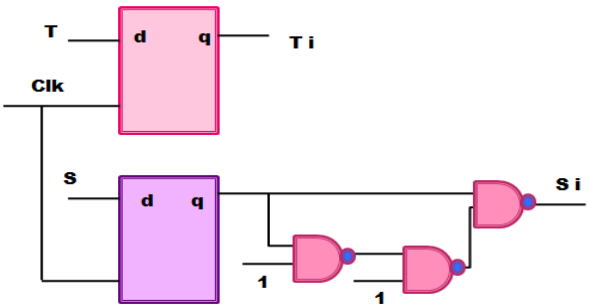


Figure 4: Driving Circuit [1]

D. Multiplexer Based Glitch Free Topology

To reduce both the larger area larger power consumption in the above DCDLs due to the additional driving circuit, multiplexer based delay lines [2] serve as an efficient alternative. For example the transistor level implementation of NAND based DCDL in inverting topology requires (6x4)+(3x4)+2(4x4)+(2x2)=72 transistors for a single delay element which includes the six NAND gates; three NAND gates and two delay flip flops for driving circuit. In case of multiplexers just (12x2)+(2x2)=28 transistors are needed, obviously reducing the area and power [15]. Both the architectures given below in Figures 5(a) and 5(b) are designed for 3 bit.

1. Inverting Topology:

The inverting multiplexer based delay lines shown in Figure 5(a) are formed by adding an inverter represented by the bubble at the output of the multiplexer and are controlled by one control bit (thermometer code) S_i. The control bits given are used as select lines for the multiplexer. The table II explains that whenever the control bits are either 0 or 1 the outputs would be with an inversion i.e., 1 or 0 respectively but with a delay. The select line of the Multiplexer at the top leftmost corner is kept at 0 to pass on the data which is always given at 0^{th} terminal of the Multiplexer both in the non-inverting and inverting topologies. In the Figure 5(a), for In= 0 and control code c= 2, the Out= 1 since it is an inverting topology. Though the control code increases by one, there are no glitches in the output as in the case of glitch free NAND based DCDLs.

2. Non- Inverting Topology:

The topology in Figure 5(b) shows the non-inverting multiplexer based delay line. The table III explains the operation of multiplexer. Whenever the control bit is '0' the data at the input terminal '0' of the MUX will be sent at the output and if the control bit is '1' the data at the input terminal '1'of the MUX will be sent at the output. In the Figure 5(b), for In= 0 and control code c= 2, the Out= 0 since it is a non-inverting topology and has no glitches.

Table 2: Inverting MUX

Terminal	**Data**	**Control Bit**	**MUX Output**
0	1	0	0
1	1	1	0

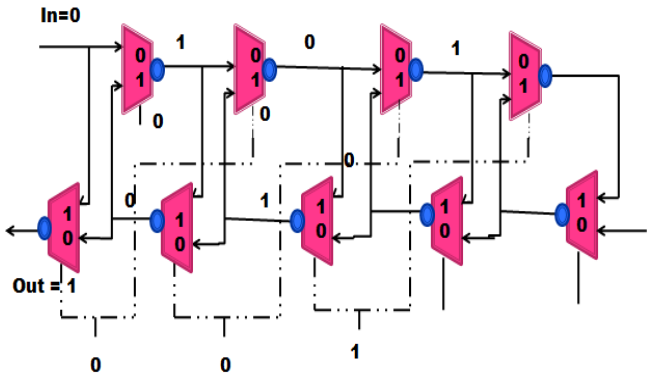


Figure 5: MUX based DCDL for 3 bit (a) Inverting Topology [2]

Table 3: Non-Inverting MUX

Terminal	Data	Control Bit	MUX Output
0	1	0	1
1	1	1	1

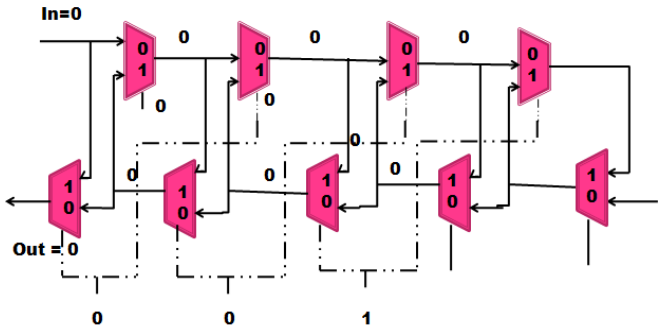


Figure 5: MUX based DCDL for 3 bit (b) Non-Inverting Topology [2]

Proposed Methodologies

A. NAND based DCDLs

The architecture of reduced transistor count is obtained by modifying the Figures 3(a) and 3(b) respectively. Since while representing the NAND gates using their equivalent transistor representation in CMOS logic, some of the transistors are found to be redundant. Therefore it paves way for eliminating such redundant transistors to save area and power. The thermometric code is used for controlling the delay lines digitally as in the previous case by the use of driving circuitry. The Figure 6(a) depicts the removal of redundant or unnecessary transistors in the inverting topology of glitch

free NAND based DCDLs of Figure 3(a) represented in red dashed lines. Similar removal is performed for non-inverting architecture.

The operation of the inverting topology in Figure 6(b) is explained for input and the control code 'c' being 2. When the input is 0, the p-transistor switches off and also the n-transistor which is controlled by the control bit S_0=0 switches off but turns on the p-transistor thereby transmitting the value of V_{dd} to the next consecutive n-transistor controlled by control bit T_0=1. Due to the value of T_0=1, the n-transistor switches on. Hence 1 is transmitted to one of the inputs of the NAND gate. Since by similar operation the second delay element has produced a value of 0, the inversion of the input 0 which is 1 is obtained at the output. There are no glitches at the output of this architecture.

The operation of the non-inverting topology in Figure 6(c) is explained for input and the control code 'c' being 2. When the input is 0, the p-transistor switches off and also the n-transistor which is controlled by the control bit S_1=1 switches on but turns off the p-transistor thereby disconnecting V_{dd} with the next consecutive n-transistor controlled by control bit T_1=1. Due to the value of T_1=1, the n-transistor switches on. Hence 1 is transmitted to one of the inputs of the NAND gate. By similar operation the second delay element has produced a value of 0, then the output from the non-inverting block is 1, hence the NAND of 1 and 1 results in the output 0 which is same as the input. Even though the control code is 2, there are no glitches in the output.

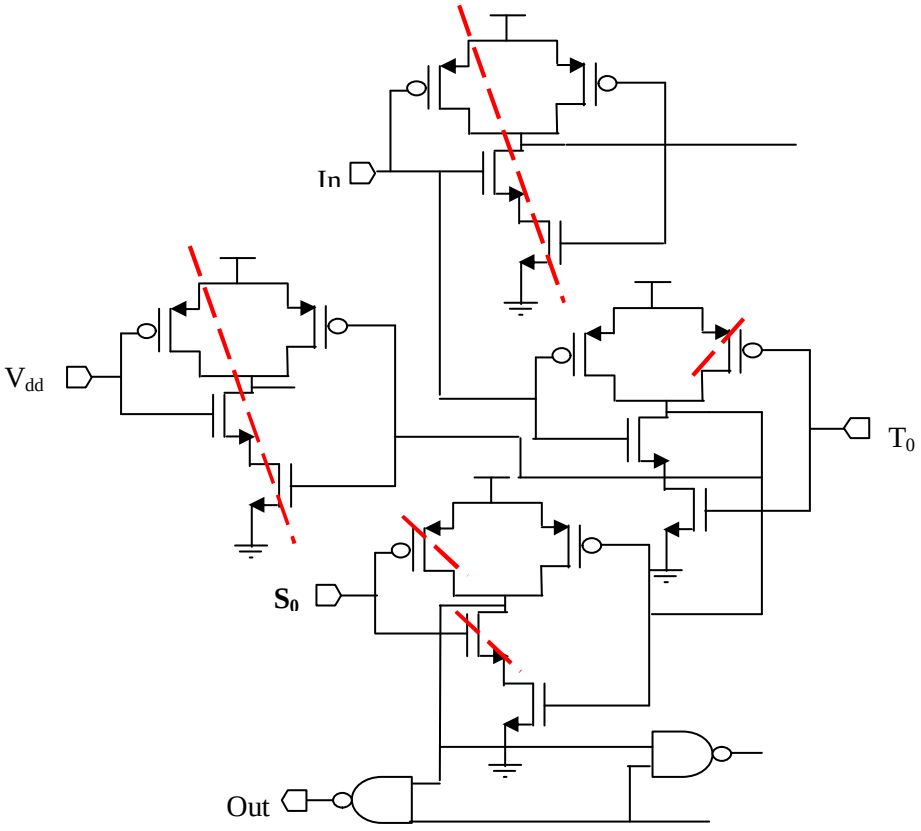


Figure 6(a): Removal of Redundant Transistors in Inverting Topology of Glitch Free NAND-based DCDL

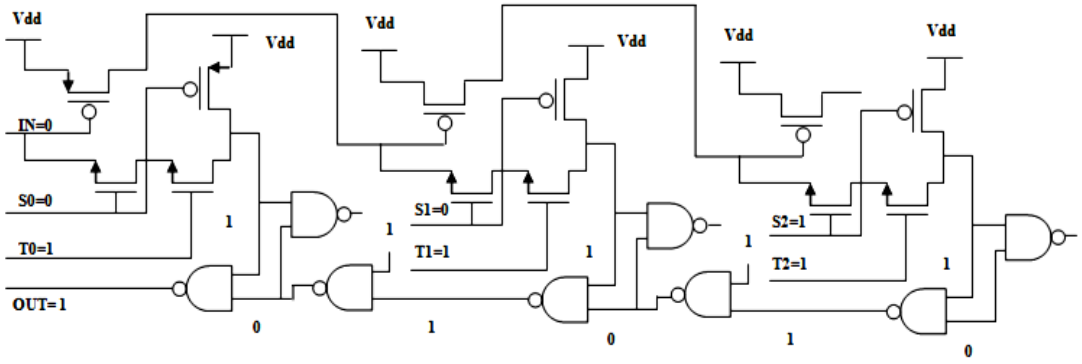


Figure 6(b): Inverting Topology of Enhanced NAND-based DCDL

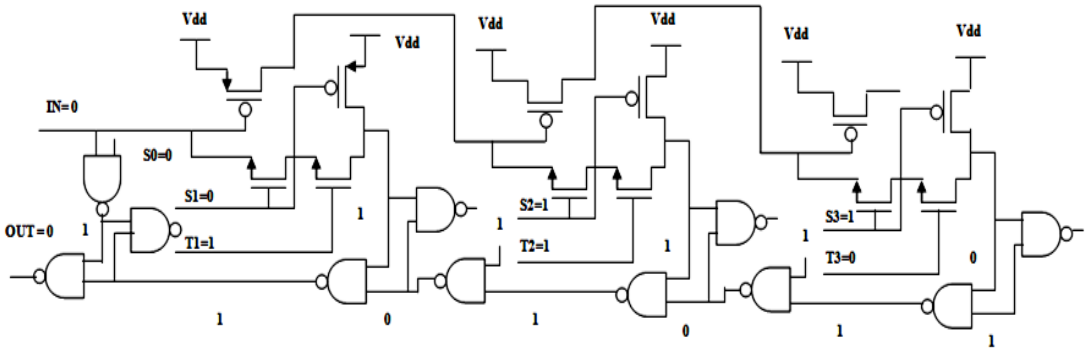


Figure 6(c): Non-Inverting Topology of Enhanced NAND-based DCDL

B. MUX based DCDLs

This modified architecture for 3 bit is derived from the design in Figures 5(a) and 5(b). From the diagram it can be observed that there is a possibility of reducing the multiplexer count. The first MUX which has its select line always set at zero and the one which provides the turn state at the end of the delay path could be eliminated. The removal of multiplexers as shown in Figures 7(a) and 7(c) has no hindrance to the usual operation of the multiplexer based DCDLs. The length of the select line interconnect is also reduced to within the delay cell itself instead of stretching to the consecutive delay cell as shown in Figures 7(b) and 7(d). The operation is similar to the existing architecture where the input is transmitted to the output based on the control code given as the select line for the multiplexer. The delay elements are replaced with the Pass Transistor Logic [11], Pseudo NMOS Logic and the following results are tabulated in tables V, VI and VII of section IV. The Pass Transistor Logic and Pseudo NMOS Logic are considered to be an effective replacement of delay elements in place of CMOS logic [14]. The architecture is designed for control code 2 and there are no glitches at the output.

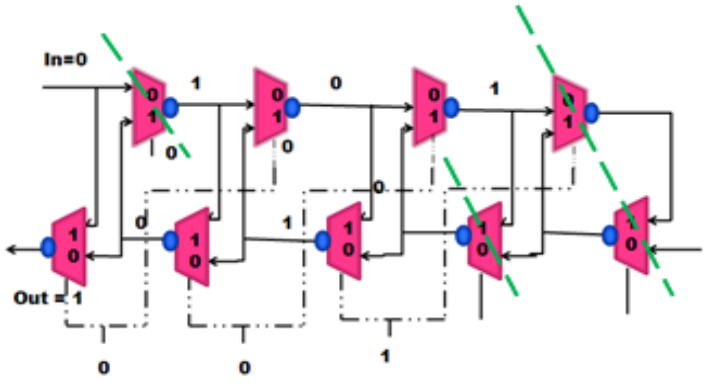


Figure 7: MUX based DCDL for 3 bit (a) Removal of MUX in Inverting Topology [2]

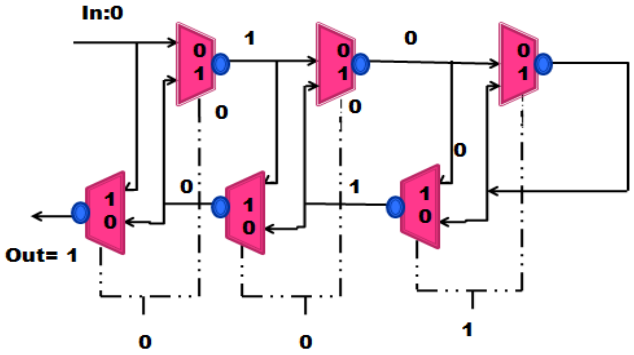


Figure 7: MUX based DCDL for 3 bit (b) Enhanced Inverting Topology [2]

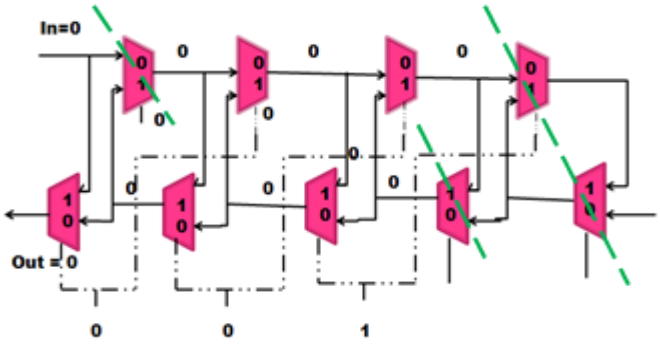


Figure 7: MUX based DCDL for 3 bit (c) Removal of MUX in Non-Inverting Topology [2]

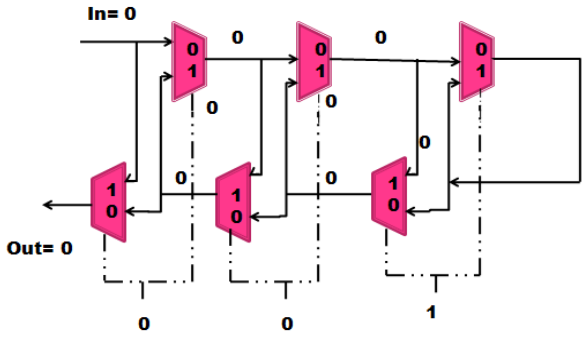


Figure 7: MUX based DCDL for 3 bit (d) Enhanced Non-Inverting Topology

Simulation Results and Discussion

The simulations are performed using 250 nm technology of Tanner Tools v13.0, the backend tool. The Figures 8(b) and 8(c) shows the screenshots of inverting and non- inverting topologies respectively. The input is given to the In terminal of the Figures of 2(a), 2(b), 3(a), 3(b), 5(a), 5(b), 6(b), 6(c), 7(b) and 7(d). A clock of amplitude 5V is given as the input and output is obtained from the Out terminal. The clock has equal rise and fall times of 1ns and the pulse width and pulse period of 10ns and 20ns are chosen respectively. The transient analysis is performed for the maximum time of 10ns and total simulation time of 100ns. Each and every screenshot has the output waveform at the top and input at the bottom.

The glitches when the input is fed into the In terminal of Figure 2(b) and the control code= 2 is shown in Figure 8(a) where the Out=1 instead of 0 for non-inverting topology. In the Figure 8(b), the glitch free output of non-inverting topology is shown where the same input appears across the output with the delay of 0.37ns (In=10.58ns; Out=10.95ns) for the control code =1. The Figure 8(c) shows the glitch free output of non-inverting topology for the control code =2 with a delay of 0.73ns (In=10.58ns; Out=11.31ns). The Out is measured by placing the vertical cursor bars at the start(x1) and end(x2) of the pulse. The value of dx is the difference between x1and x2. The In is also measured by the same procedure. The above delays increase as the control code increases. Since the control codes are applied to the Figure 3(b), which is a glitch free circuit there are no glitches at the output.

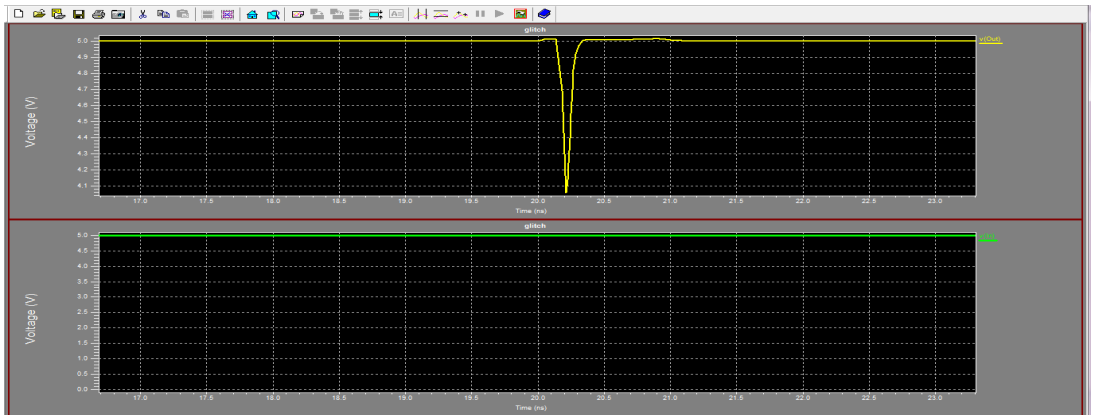


Figure 8(a): Presence of Glitches in NAND based DCDLs

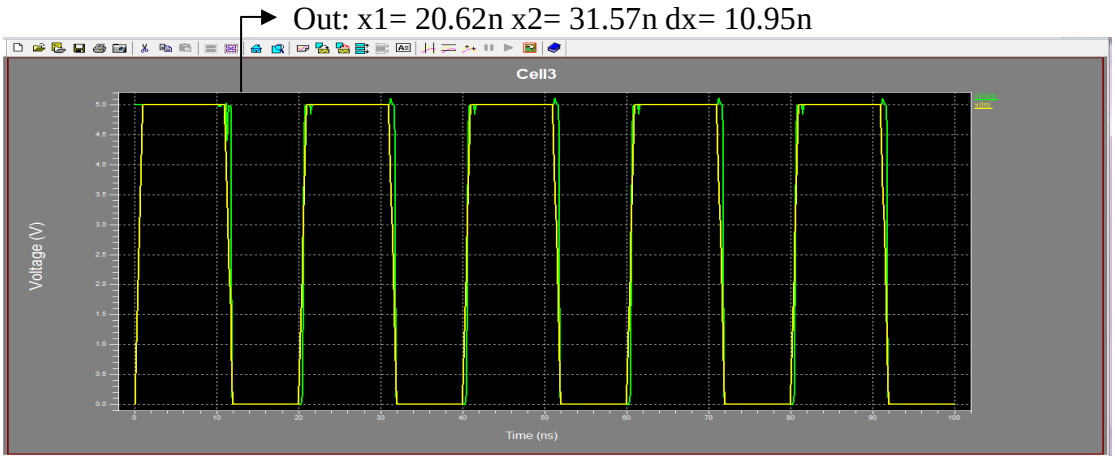


Figure 8(b): Glitch Free NAND based DCDL when Control Code= 1

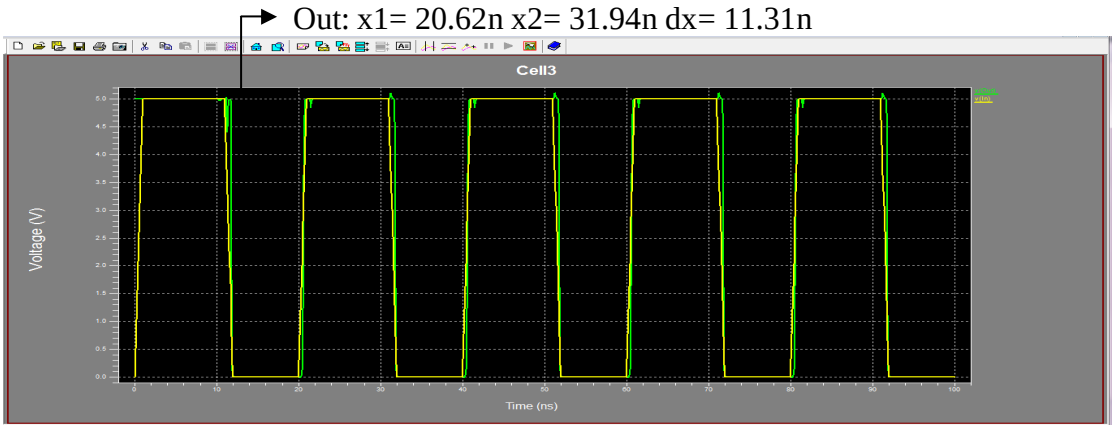


Figure 8(c): Glitch Free NAND based DCDL when Control Code= 2

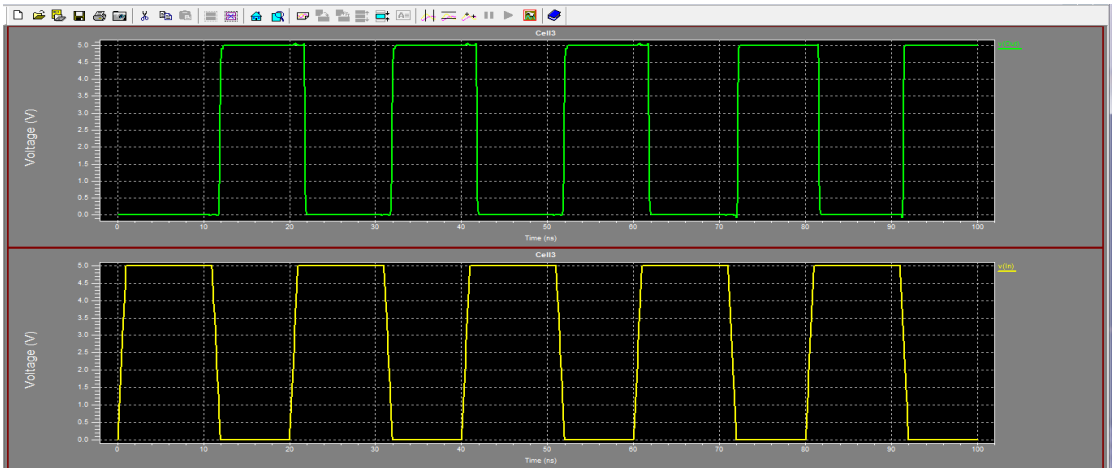


Figure 8(d): Input and Output of Inverting Topology of NAND and Multiplexer based DCDLs

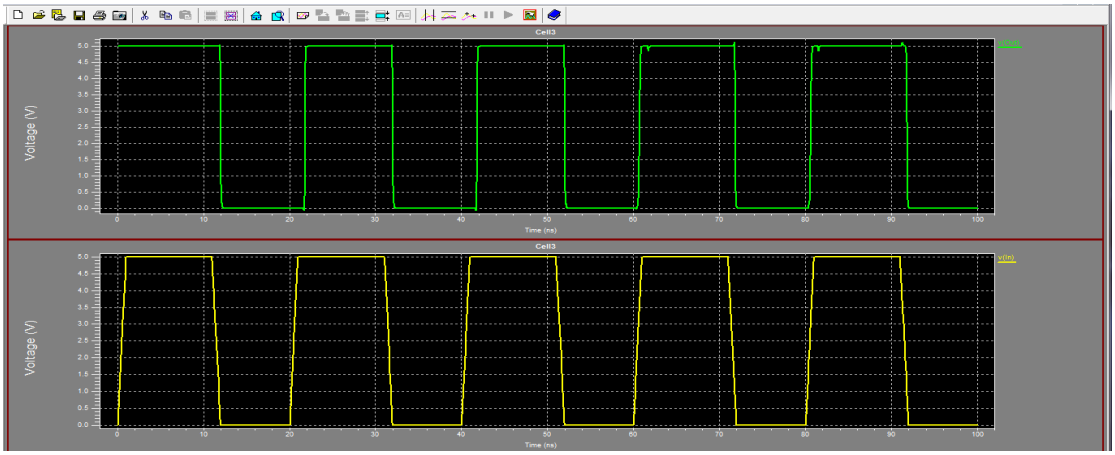


Figure 8(e): Input and Output of Non- Inverting Topology of NAND and Multiplexer based DCDLs

The screenshots in Figures 8(d) and 8(e) represents the inverting and non-inverting topologies of the glitch free NAND and Multiplexer based DCDLs obtained while giving the clock input to the Figures 3(a), 3(b), 5(a), 5(b), 6(b), 6(c), 7(b) and 7(d). In the inverting topology the input appears across the output with an inversion but in case of non-inverting there is no inversion and also in both the cases there are no glitches in any of the Figures listed above. The reason for absence of glitches at the output is due to the elimination of multiple path propagation on additional control bit generated and passed through the driving circuit after which they are applied into the S and T bits. The screenshots in Figures 8(d) and 8(e) and the analysis report are obtained by assigning the value of control code to be 10 and there are no glitches.

Table 4: Analysis Report of Glitch Free NAND and MUX Based DCDLs

Parameters	**Glitch free DCDLs**			
	NAND		**MUX**	
	Inverting	**Non-Inverting**	**Inverting**	**Non-Inverting**
Power (watts)	3.87e-002	2.04e-002	6.86e-003	5.44e-003
Number of Active Devices	720	732	294	252
Simulation Time (sec)	14.47	10.87	8.87	4.18

Table 5: Power Report of Enhanced Architecture for DCDLs

Transistor Logic	**Modified Architecture for DCDLS**			
	NAND		**MUX**	
	Inverting	**Non-Inverting**	**Inverting**	**Non-Inverting**
CMOS	1.09e-002	1.55e-002	6.53e-003	5.16e-003
Pseudo NMOS	-	-	2.53e-003	5.82e-003
Pass Transistor	-	-	2.19e-003	1.58e-003

Table 6: Area Report of Enhanced Architecture for DCDLs

Transistor Logic	**Enhanced Architecture for DCDLS**			
	NAND		**MUX**	
	Inverting	**Non-Inverting**	**Inverting**	**Non-Inverting**
CMOS	160	172	280	240
Pseudo NMOS	-	-	105	147
Pass Transistor	-	-	80	120

Table 7: Time Report of Enhanced Architecture for DCDLs

Transistor Logic	**Modified Architecture for DCDLS**			
	NAND		**MUX**	
	Inverting	**Non-Inverting**	**Inverting**	**Non-Inverting**
CMOS	2.60	2.48	5.92	4.12
Pseudo NMOS	-	-	3.14	2.56
Pass Transistor	-	-	2.99	2.41

In table IV, the analysis report of Existing Glitch Free NAND and MUX based DCDLs are obtained on implementing the circuits 3(a), 3(b), 5(a) and 5(b) using Tanner with a control code of 10. On implementing the Figures 6(b), 6(c), 7(b) and 7(d) further reduction in the power, area and simulation time are obtained and the results are tabulated in tables V, VI and VII. In table V, the power report of inverting and non- inverting topologies of Enhanced NAND and MUX based DCDLs are discussed. Compared to the table IV, table V has resulted in minimized power consumption for the same control code of 10 due to reduced number of active devices. The area report of table VI is the number of active devices when a control code of 10 is selected. There is a drastic reduction in the transistor count due to the removal of redundant or unnecessary transistors in the existing architecture. In table VII, the time report shows that the simulation time taken for executing the control code of 10. Since in table IV, the results of existing glitch free DCDLs are obtained on CMOS implementation, the alternate transisor logics for delay elements as proposed are also tabulated in tables V, VIand VII and has proved to be more efficient than CMOS. Among the three transisor logics such as CMOS, Pseudo NMOS and Pass Transistor, the maximum optimizations are obtained from Pass Transistor logic.

Conclusion

The glitches in the conventional DCDLs had embarked upon the need for glitch-free NAND based DCDLs. To reduce area and power in the NAND and multiplexer based glitch free delay lines, reduced transistor count architecture is proposed. The results are obtained using 250nm technology of Tanner Tools v13.0. From the results, the observation made is that better area, reduction in power and time optimizations are obtained using the proposed architecture.

References

[1]. Davide De Caro, Senior Member, IEEE, 'Glitch-Free NAND-Based Digitally Controlled Delay-Lines' IEEE Transactions On Very Large Scale Integration (VLSI) Systems, Vol. 21, No. 1, January 2013.

[2]. Jean Barbier, Montpellier, 'Programmable Delay Line Circuit With Glitch Avoidance', Patent published on sep.30,2010.

[3]. R. B. Staszewski, K.Muhammad, D. Leipold, C.-M. Hung, Y.-C. Ho, J. L. Wallberg, C. Fernando, K. Maggio, R. Staszewski, T. Jung, J. Koh, S. John, I. Y. Deng, V. Sarda, O. Moreira-Tamayo, V. Mayega, R. Katz, O. Friedman, O. E. Eliezer, E. de-Obaldia, and P. T. Balsara, 'All-digital TX frequency synthesizer and discrete-time receiver for bluetooth radio in 130-nm CMOS,' IEEE J. Solid-State Circuits, vol. 39, no. 12, pp. 2278–2291, Dec. 2004.

[4]. R. B. Staszewski and P. T. Balsara, 'All Digital Frequency Synthesizer in Deep Submicron CMOS', New York: Wiley, 2006.

[5]. Mohammad Maymandi-Nejad and Manoj Sachdev, 'A Digitally Programmable Delay Element: Design and Analysis,' IEEE Trans. VLSI, Reg. Papers, vol. 11, no. 5, Oct. 2003.

[6]. Soumya Shivakumar Begur ' A 45nm CMOS, low jitter, all-digital delayed locked loop with a circuit to dynamically vary phase to achieve fast lock', Northeastern University, January 01, 2011.

[7]. Goran S. Jovanović1 and Mile K. Stojčev2,' Linear Current Starved Delay Element', IEEE J. Solid-State Circuits.

[8]. S. Damphousse, K. Ouici, A. Rizki, and M. Mallinson, 'All digital spread spectrum clock generator for EMI reduction,' IEEE J. Solid- State Circuits, vol. 42, no. 1, pp. 145–150, Jan. 2007.

[9]. B. Nikolic, V. G. Oklobdzija, V. Stajanovic, W. Jia, J. K. Chiu, and M. M. Leung, 'Improved sense-amplifier based flip-flop: Design and measurements,' IEEE J. Solid-State Circuits, vol. 35, no. 6, pp. 876–883, Jun. 2000.

[10]. G. M. Strollo, D. De Caro, E. Napoli, and N. Petra, 'A novel high speed sense amplifier based flip-flop,' IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 13, no. 11, pp. 1266–1274, Nov. 2005.

[11]. D. Markovic´, B. Nikolic and V.G. Oklobdzˇijac, 'A general method in synthesis of pass-transistor circuits', Microelectronics Journal 31 (2000) 991–998.

[12]. Guansheng Li, Yahya M. Tousi,Arjang Hassibi and Ehsan Afshari, 'Delay- Line- Based Analog-to-Digital Converters,' IEEE J. IEEE Trans on circuits and systems—II: express briefs, vol. 56, no. 6, June 2009.

[13]. Ali Kabiri, Qing He, Mohammed H. Kermani and Omar M. Ramahi,' 'Design of a Controllable Delay Line', in IEEE Transactions on Advanced Packaging, Volume 33, November 2010, pp.1080- 1087.

[14]. Preethi.D and Dr. R.Brindha, 'Efficient Implementations of Delay Elements for Digitally Controlled Delay Lines' in International Conference on Electrical, Instrumentation and Communication Engineering, at Sri Krishna College of Technology, January 2015.

[15]. Preethi.D, 'Area and Power Efficient Glitch-less Digitally Controlled Delay Lines', in International Journal of Emerging Trends and Technology, Volume 9, Number 14, March 2014, IJETT-V9P332.