CHAPTER III

PROPOSED XOR/XNOR CIRCUITS

Ever growing demand under explosive proportions for portable applications like mobile phones and laptops made designers to make great efforts to achieve in the lesser silicon area, high speed, longer battery life and high reliability. The XOR/XNOR is the basic building block of various circuits like adders, comparators, multipliers, parity generator/checkers, code converters and error controlling coders – to mention a few. Hence it is used widely in many VLSI systems as a part of the critical path that determines the altogether performance of the system. The performance of the complex logic is decided by the performance of individual XOR/XNOR blocks that are part and parcel of them. Careful design and analysis, therefore, is required for XOR/XNOR circuits to produce full rail-to-rail swing, dissipate less power, and have less delay in the critical path and component economy leading to smaller. In this part of the thesis, the proposed circuits are divided into four categories. Those are:

- full rail to rail swing, high performance and better PDP XOR/XNOR Circuits
- fault secure, full rail to rail swing, high performance and better PDP XOR/XNOR circuit
- low voltage, low power, full rail to rail swing and better PDP XOR/XNOR circuits
- multi valued logic, full rail to rail swing and better PDP XOR/XNOR circuits

3.1 Full Rail To Rail Swing, High Performance and Better PDP XOR/XNOR Circuits

In this category, the proposed circuits are:

(i) Two input differential XOR/XNOR circuits
(ii) Direct Three input differential XOR/XNOR circuits
(iii) Four input and Eight input differential XOR/XNOR circuits
(iv) General structure for multi input XOR/XNOR circuits
(v) Direct Three input XOR circuits
(vi) Two input XNOR circuits

3.1.1 Two input differential XOR/XNOR circuits

Over the years, various XOR/XNOR designs have been reported based on Complimentary Pass Transistor Logic (CPL), Pass Transistor Logic (PTL), Double

Pass Transistor Logic (DPL) and Transmission Gate (TG) logic to enhance the performance of time synchronized applications like wave-pipelined systems. Family of Pass Transistor Logic (PTL) had been a popular choice due to identical logic depth to this end. However, this logic style has its limitation due to data-dependent output level and input loading.

We have proposed a new XOR/XNOR structure ([1] from list of publications) with only two transistors and two inverters connected back to back. Number of components are thus reduced to 6. Yet the output gives full swing due to the inverter-latch. The gates can be realized (including true and complementary inputs) for full swing for both outputs with only 6, 8 and 10 transistors in 4-cases, using the same topology. Another topology proposed in the present work uses transmission gate (TG) to handle the threshold voltage issue. The second approach proposes a XOR/XNOR structure with only two TG and two inverters connected back to back so that full swing can be obtained at the output. Complete logic gates with only 10 and 12 transistors are realized which operate even at low supply overcoming thereby the threshold issue.

Here, two high performance, low voltage, full swing, less area and noise tolerant XOR/XNOR structures are presented . The 1st structure denoted as structure-1 has 4 versions that realize the logic using 10, 8, 8 and 6 transistors respectively, in four configurations. The 2nd structure, denoted as structure-2, realizes the logic using 10 and 12 transistors in each of its four configurations. The proposed gates are based on pass-transistor/transmission-gate logic and static CMOS inverter. The pass transistor design has lesser component-count and operates at very low supply. Since a nMOS passes a good ‘0’ and a degraded ‘1’ while a pMOS passes a good ‘1’ and a degraded logic ‘0’, so either complementary CMOS inverters as level-restorer or transmission gates at pass-level are used to produce a strong output logic level for all input combinations.

3.1.1.1 Proposed Structure-1 for XOR/XNOR circuit

Fig.3.1 (a), (b), (c) and (d) show the proposed circuit schematics of structure-1 for XOR/XNOR logic. Each uses back-to-back static CMOS inverters and two pass transistors. In all these circuits, depending on the input signal A, only one pass transistor gets switched ON and the corresponding input signal B will be passed on to one of the outputs (XOR or XNOR). Then through CMOS inverter the complementary output will be generated.

3.1.1.1.1 Proposed Structure-1 for XOR/XNOR Version-1

To understand the operation of the proposed logic depicted in Fig.3.1(a), let us first assume that A is at logic '0' and thus T_1 is ON and T_2 is OFF. So B^1 will be passed to the XNOR-output, which is also connected to the inverter I_2 and at the XOR-output B will appear. In this case if B is at logic '1', then a strong '0' is passed to XNOR-output. That strong '0' is connected to the inverter I_2 and a resultant strong logic '1' will be go to XOR-output.

Table 3.1 State of all the transistors in Fig.3.1(a)

A B	**T_1**	**T_2**	**I_1**		**I_2**		**XOR**	**XNOR**
			T_3	**T_4**	**T_5**	**T_6**		
0 0	ON	OFF	ON	OFF	OFF	ON	0	1
0 1	ON	OFF	OFF	ON	ON	OFF	1	0
1 0	OFF	ON	OFF	ON	ON	OFF	1	0
1 1	OFF	ON	ON	OFF	OFF	ON	0	1

If B is at logic '0' then a weak '1' (V_{dd}-V_{tn}) is initially passed on to the XNOR-output. However the inverter I_2 generates a strong '0', since (V_{dd}-V_{tn}) is set to be greater than its switching-threshold value of $V_{dd}/2$ and the strong '0' is connected to the inverter I_1 which will generate a strong logic '1' to override the weak '1' passed on by T_1. Thus, due to cross coupling of inverters (designed to have a positive feedback with a loop gain >1), even a weak signal ('1' or '0') is converted to a strong signal at the output.

Similarly, when A is at logic '1', B^l is passed to the XOR and B is passed on to the XNOR through the inverter I_1. Once again any degraded output, due to the back-to-back connection of inverters will eventually become strong to override the weak signal at the output. Table 3.1 gives the operation of the Fig.3.1(a) by defining the active states of all the transistors for complete set of inputs.

3.1.1.1.2 Proposed Structure-1 for XOR/XNOR Version-2

The proposed structure-1 version-2 is shown in Fig.3.1(b). To understand the operation of Fig.3.1 (b), let us first assume that A is at logic '0'and thus T_1 is ON and T_2 is OFF. So B^1 will be passed to the XNOR-output, which is also connected to the inverter I_2 and at the XOR-output B will appear. In this case if B is at logic '0', then a strong '1' is passed to XNOR-output. That strong '1' is connected to the inverter I_2 and a resultant strong logic '0' will be go to XOR-output.

If B is at logic ‘1’ then a weak ‘0’ (V_{dd}-V_{tn}) is initially passed on to the XNOR-output. However, the inverter I_2 generates a strong ‘1’, since (V_{dd}-V_{tn}) is set to be greater than its switching-threshold value of $V_{dd}/2$ and the strong ‘0’ is connected to inverter I_1 which will generate a strong logic ‘0’ to override the weak ‘0’ passed on by T_1. Thus, due to cross coupling of inverters (designed to have a positive feedback with a loop gain >1), even a weak signal (‘1’ or ‘0’) is converted to a strong signal at the output.

Table 3.2 State of all the transistors in Fig.3.1(b)

A B	**T_1**	**T_2**	**I_1**		**I_2**		**XOR**	**XNOR**
			T_3	**T_4**	**T_5**	**T_6**		
0 0	ON	OFF	ON	OFF	OFF	ON	0	1
0 0	ON	OFF	OFF	ON	ON	OFF	1	0
1 0	OFF	ON	OFF	ON	ON	OFF	1	0
1 1	OFF	ON	ON	OFF	OFF	ON	0	1

Similarly, when A is at logic ‘1’, B^l is passed to the XOR and B is passed on to the XNOR through the inverter I_1. Once again any degraded output, due to the back-to-back connection of inverters will eventually become strong to override the weak signal at the output. Table 3.2 gives the operation of the Fig.3.1(b) by including the active states of all the transistors for complete set of inputs.

3.1.1.1.3 Proposed Structure-1 for XOR/XNOR Version-3

Fig.3.1(c) gives the proposed structure-1 version-3. To understand the operation of the Fig.3.1(c), let us first assume that A is at logic ‘0’and thus T_1 is ON and T_2 is OFF. So B will be passed to the XOR-output, which is also connected to the inverter I_2 and at the XNOR-output B^l will appear. In this case if B is at logic ‘0’, then a strong ‘0’ is passed to XOR-output. That strong ‘0’ is connected to the inverter I_2 and a resultant strong logic ‘1’ will be go to XNOR-output.

If B is at logic ‘1’ then a weak ‘1’ (V_{dd}-V_{tn}) is initially passed on to the XOR-output. However the inverter I_2 generates a strong ‘0’, since (V_{dd}-V_{tn}) is set to be greater than its switching-threshold value of $V_{dd}/2$ and the strong ‘0’ is connected to the inverter I_1 which will generate a strong logic ‘1’ to override the weak ‘1’ passed on by T_1. Thus, due to cross coupling of inverters (designed to have a positive feedback with a loop gain >1), even a weak signal (‘1’ or ‘0’) is converted to a strong signal at the output. Similarly, when A is at logic ‘1’, B is passed to the XNOR and B^l is passed on to the XOR through the inverter I_1. Once again any degraded output, due to the back-to-back

connection of inverters will eventually become strong to override the weak signal at the output. Table 3.3 gives the operation of the Fig.3.1(c) by including the active states of all the transistors for complete set of inputs.

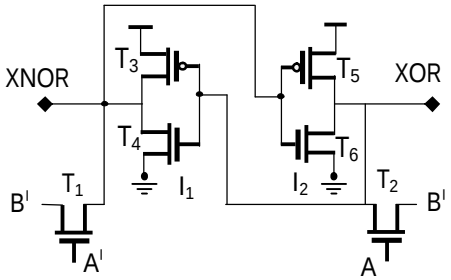


Fig.3.1(a) Proposed Structure-1 for XOR/XNOR Version-1

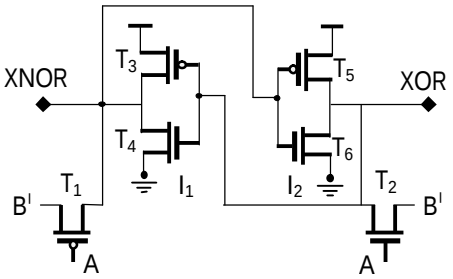


Fig.3.1(b) Proposed Structure-1 XOR/XNORVersion-2

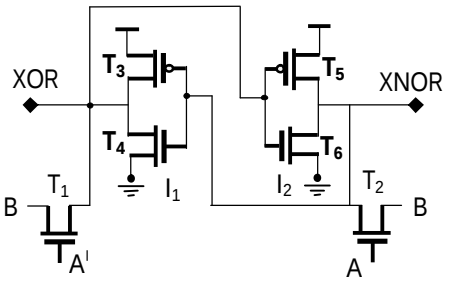


Fig.3.1(c) Proposed Structure-1 XOR/XNORVersion-3

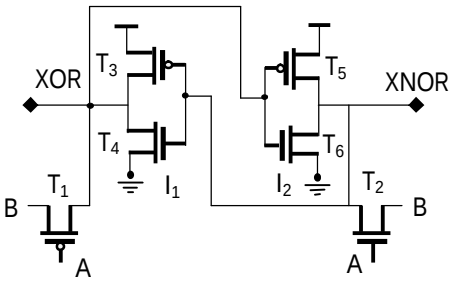


Fig.3.1(d) Proposed Structure-1 XOR/XNORVersion-4

Table 3.3 State of all the transistors in Fig.3.1(c)

A B	T_1	T_2	I_1		I_2		XOR	XNOR
			T_3	T_4	T_5	T_6		
0 0	ON	OFF	OFF	ON	ON	OFF	0	1
0 1	ON	OFF	ON	OFF	OFF	ON	1	0
1 0	OFF	ON	ON	OFF	OFF	ON	1	0
1 1	OFF	ON	OFF	ON	ON	OFF	0	1

3.1.1.1.4 Proposed Structure-1 for XOR/XNOR Version-4

The proposed structure-1 version-4 is shown in Fig.3.1(d). To understand its operation, let us first assume that A is at logic '0'and thus T_1 is ON and T_2 is OFF. So B will be passed to the XOR-output, which is also connected to the inverter I_2 and at the XNOR-output B^l will appear. In this case if B is at logic '1', then a strong '1' is passed to XOR-output. That strong '1' is connected to the inverter I_2 and a resultant strong logic '0' will be go to XNOR-output.

Table 3.4 State of all the transistors in Fig.3.1(d)

A B	T_1	T_2	I_1		I_2		XOR	XNOR
			T_3	T_4	T_5	T_6		
0 0	ON	OFF	OFF	ON	ON	OFF	0	1
0 1	ON	OFF	ON	OFF	OFF	ON	1	0
1 0	OFF	ON	ON	OFF	OFF	ON	1	0
1 1	OFF	ON	OFF	ON	ON	OFF	0	1

If B is at logic '0' then a weak '0' (V_{dd}-V_{tn}) is initially passed on to the XOR-output. However, the inverter I_2 generates a strong '1', since (V_{dd}-V_{tn}) is set to be greater than its switching-threshold value of $V_{dd}/2$ and the strong '1' is connected to the inverter I_1 which will generate a strong logic '0' to override the weak '0' passed on by T_1. Thus, due to cross coupling of inverters (designed to have a positive feedback with a loop gain >1), even a weak signal ('1' or '0') is converted to a strong signal at the output. Similarly, when A is at logic '1', B is passed to the XNOR and B^l is passed on to the XOR through the inverter I_1. Once again any degraded output, due to the back-to-back connection of inverters will eventually become strong to override the weak signal at

the output. Table 3.4 gives the operation of the Fig.3.1(d) by including the active states of all the transistors for complete set of inputs.

3.1.1.2 Proposed Structure-2 for XOR/XNOR circuit

Fig.3.2 (a), (b), (c) and (d) show the proposed circuit schematics of structure-2 for XOR/XNOR logic. Once again each variation uses back-to-back connected static CMOS inverters where as the pass transistors are substituted by transmission gates. In each version, depending on the input signal A, only one TG goes ON at a time and the corresponding input signal (B or B^{l}) will be passed on to one of the outputs XOR or XNOR through a CMOS inverter. The complementary output is generated through the other CMOS inverter.

3.1.1.2.1 Proposed Structure-2 for XOR/XNOR Version-1

In Fig.3.2(a), the TG_2 turns ON and TG_1 turns OFF when A is logic '0' and B^{l} is passed to the XNOR-output. However B^{l} is also connected to the inverter I_1 and the logic B is this passed on to the XOR-output. The opposite happens when A is logic '1' and the XOR and XNOR output gets logic B^{l} and B respectively. The circuit gives full output swing for every input pattern. Table 3.5 gives the operation of the Fig.3.2(a) by including the active states of all the transistors for complete set of inputs.

Table 3.5 States of all the transistors in Fig.3.2(a)

A B	TG_1	TG_2	I_1		I_2		XOR	XNOR
			T_3	T_4	T_5	T_6		
0 0	OFF	ON	OFF	ON	ON	OFF	0	1
0 1	OFF	ON	ON	OFF	OFF	ON	1	0
1 0	ON	OFF	ON	OFF	OFF	ON	1	0
1 1	ON	OFF	OFF	ON	ON	OFF	0	1

3.1.1.2.2 Proposed Structure-2 for XOR/XNOR Version-2

Fig.3.2(b) shows the proposed structure-2 version-2. Table 3.6 gives its operation by including the active states of all the transistors for complete set of inputs. In Fig.3.2(b), the TG_2 turns ON and TG_1 turns OFF when A is logic '1' and B is passed to the XNOR-output. However B is also connected to the inverter I_1 and the logic B^{l} is this passed on to the XOR-output. The opposite happens when A is logic '0' and the XNOR and XOR output gets logic B^{l} and B respectively. The circuit gives full output swing for every input pattern.

Table 3.6 State of all the transistors in Fig.3.2(b)

A B	TG_1	TG_2	I_1		I_2		XOR	XNOR
			T_3	T_4	T_5	T_6		
0 0	ON	OFF	OFF	ON	ON	OFF	0	1
0 1	ON	OFF	ON	OFF	OFF	ON	1	0
1 0	OFF	ON	ON	OFF	OFF	ON	1	0
1 1	OFF	ON	OFF	ON	ON	OFF	0	1

3.1.1.2.3 Proposed Structure-2 for XOR/XNOR Version-3

Fig.3.2(c) displays the proposed structure-2 version-3. In Fig.3.2(c), the TG_2 turns ON and TG_1 turns OFF when B is logic '1' and A^1 is passed to the XOR-output. However, A^1 is also connected to the inverter I_1 and the logic A is this passed on to the XNOR-output. The opposite happens when B is logic '0' and the XNOR and XOR output gets logic A^1 and A respectively. The circuit gives full output swing for every input pattern. Table 3.7 gives the operation of the Fig.3.2(c) by defining the active states of all the transistors for complete set of inputs.

Table 3.7 State of all the transistors in Fig.3.2(c)

A B	TG_1	TG_2	I_1		I_2		XOR	XNOR
			T_3	T_4	T_5	T_6		
0 0	ON	OFF	ON	OFF	OFF	ON	0	1
0 1	OFF	ON	OFF	ON	ON	OF	1	0
1 0	ON	OFF	OFF	ON	ON	OFF	1	0
1 1	OFF	ON	ON	OFF	OFF	ON	0	1

3.1.1.2.4 Proposed Structure-2 for XOR/XNOR Version-4

In Fig.3.2(d), the TG_2 turns ON and TG_1 turns OFF when B is logic '0' and A is passed to the XOR-output. However, A is also connected to the inverter I_1 and the logic A^1 is this passed on to the XNOR-output. The opposite happens when B is logic '1' and the XNOR and XOR output gets logic A and A^1 respectively. The circuit gives full output swing for every input pattern. Table 3.8 gives the operation of the Fig.3.2(d) by defining the active states of all the transistors for complete set of inputs.

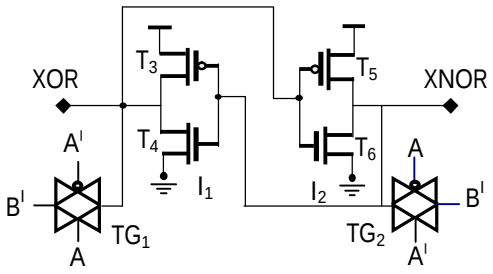


Fig.3.2(a) Proposed Structure-2 XOR/XNORVersion-1

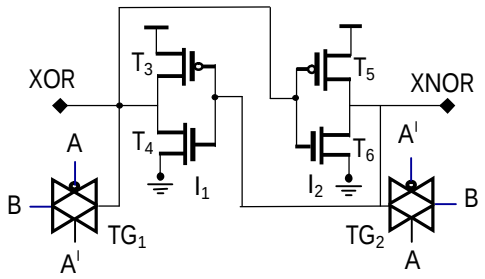


Fig.3.2(b) Proposed Structure-2 XOR/XNORVersion-2

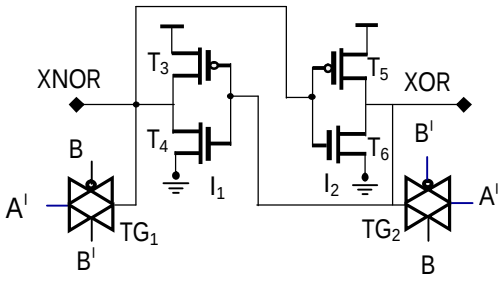


Fig.3.2(c) Proposed Structure-2 XOR/XNORVersion-3

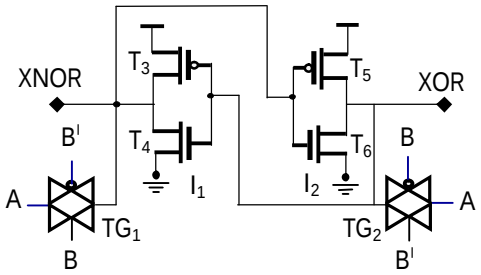


Fig.3.2(d) Proposed Structure-2 XOR/XNORVersion-4

Table 3.8 State of all the transistors in Fig.3.2(d)

A B	TG_1	TG_2	I_1		I_2		XOR	XNOR
			T_3	T_4	T_5	T_6		
0 0	OFF	ON	ON	OFF	OFF	ON	0	1
0 1	ON	OFF	OFF	ON	ON	OFF	1	0
1 0	OFF	ON	OFF	ON	ON	OFF	1	0
1 1	ON	OFF	ON	OFF	OFF	ON	0	1

3.1.2 Three input differential XOR/XNOR circuits

In this section three new structures are proposed for direct three input XOR/XNOR logic gate circuits ([9] from list of publications). Advanced logic function can be implemented using the feature of FinFET transistor to work as a simple switch that can avail the advantage of being simple and fast. A switch can be implemented with a single transistor (n-FinFET or p-FinFET) called pass transistor logic (PTL) or with a combination of two transistors (n-FinFET and p-FinFET) as transmission gate logic (TGL). TGL switch provides better noise margin and is very efficient in implementing complex logic gates. The area overhead is the basic disadvantage of TGL designs. So, in this part PTL logic is used to construct the XOR/XNOR circuits. Alternately, all the proposed circuits can be implemented with TGL logic, but area required is doubled.

The proposed gate circuits have used the concept of pass transistor logic and back to back connection of inverters. The pass transistor design permits the usage of less number of transistors, very low-power operation and produces high-performance. In this pass transistor logic, n-FinFET device passes a strong logic ‘0’, but a weak logic ‘1’ while p-FinFET device passes a strong logic ‘1’, but a weak logic ‘0’, so the complementary FinFET inverters are used to pass a strong output logic level for all input combinations of logic ‘1’ and logic ‘0’.

Although there are many XOR/XNOR designs existing by using pass transistor logic, the improvements are needed in the number of transistors, the power consumption and speed. Also, they are producing weak output signal swing for certain input combinations. Due to a threshold voltage loss problem, they are not reliable for very low supply voltages even though they are proposed to be noise tolerant. The circuits can be constructed with PTL technology for getting less area, less delay and less power. But the drop of driving capability is the main drawback in PTL technology. To overcome this hurdle, one solution is that adding inverters at output paths. Fig.3.3 to

Fig.3.5 shows the circuit schematics of the proposed structure-1 to structure-3 three input XOR/XNOR logic gate using PTL and back to back connection of two inverters. These circuits can give full logic swing for both outputs with only 14, 12 and 10 transistors even at +0.4 to +0.6 voltage also.

All these circuits produce a perfect full rail to rail swing operation at every input combination. The basic logic involved in the proposed structure is that the inputs C or C^{I} (in Fig 3.4 and Fig 3.5) are passed to the output terminals XOR or XNOR depending upon the inputs A, B, A^{I} and B^{I} which are used as the control signals to the pass transistors. And the basic logic provided for the proposed structure-1 is that its inputs such as A or A^{I} or B or B^{I} or C or C^{I} (in Fig 3.3) are passed to the output terminals XOR or XNOR depending upon the inputs A , A^{I}, B, B^{I}, C and C^{I} which are used as the control terminals to the pass transistors. Hence, based on the input signals, the pass transistors acting as switches will be in either ON or OFF state, thus allowing required logic to the output. But logic implemented using PTL fails to give full swing due to loss of threshold, and to overcome this back, to back connection of inverters are used at the output. Another advantage of using an inverter pair is that it improves the driving capability of the circuits.

A back to back connection of two inverters is used in all the proposed circuits to achieve one of the outputs either XOR or XNOR depending on the input pattern and full swing logic for two XOR and XNOR outputs. Thus, the designs are carefully implemented to obtain the truth tables of three input XOR and XNOR logics. The advantages derived from the proposed circuits are that only one conduction path exists for any combination of inputs. The sizes of transistors are carefully selected for optimal power delay performance under various operational conditions. As per the operation, the critical path consists of only two switches and one loop, hence the proposed structures are supposed to be having high performance. Another advantage of all the circuits is that they have low power consumption because of less number of transistors in ON condition in every input pattern and of minimum size individually. The working of all the proposed three input XOR/XNOR structures can be analyzed by studying the operation of the circuits for the whole set of input patterns ABC = 000, 001, 010, 011, 100, 101, 110, 111.

3.1.2.1 Proposed Structure-1 for 3 input XOR/XNOR circuit

The proposed structure-1 circuit is shown in Fig.3.3. It is constructed using PTL with 10 pass transistors and 2 inverters. In Fig.3.3, the logic is implemented with n-FinFET

transistors only and it requires complementary input signals. For example, in Fig 3.3, when ABC = '101', then the path of transistors T_3 and T_5 are switched ON and input B i.e full logic '0' is passed on to the output of XOR and full logic '1' is generated at the output XNOR through an inverter (transistors I_{p1} and I_{n1}). For '001' combination the path of transistors T_6 and T_7 is switched ON so that the full logic '0' i.e. input C^l is passed to XNOR output and full logic '0' is produced at the XOR output through the inverter (transistors Ip_2 and In_2).

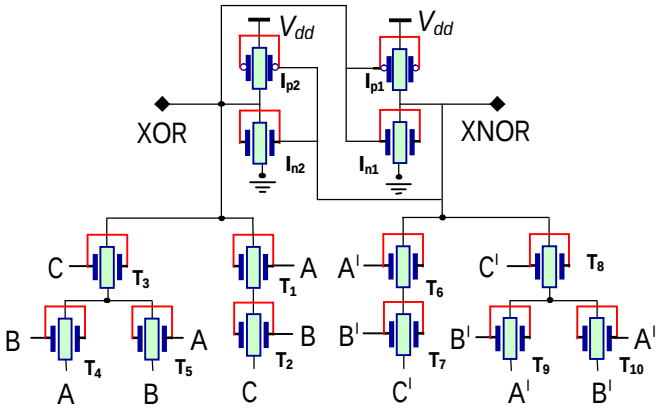


Fig.3.3 Three input XOR/XNOR Structure-1

A similar operation applies in all the rest of the cases. For the input '000' and '111' cases the passed signals at XOR or XNOR are considered degraded logics because of the threshold voltage loss problem are pass transistors. When an input pattern is '000', then the transistors in the inverters I_{n2} and I_{p1} forms, feedback loop to produce full logics at the both XOR and XNOR outputs. For the '111' pattern, the transistors I_{n1} and I_{p2} forms loop to get full logics at the both XOR and XNOR outputs. Similar operation applies to all the remaining combinations of inputs. Table 3.9 gives the truth table of Fig.3.3 by including conduction states of all the transistors.

3.1.2.2 Proposed Structure-2 for 3 input XOR/XNOR circuit

The proposed structure-2 circuit is shown in Fig.3.4. It consists of 8 pass transistors of n-FinFET and 2 static inverters. Complementary control input signals are needed for Fig.3.4, since the logic is implemented with n-FinFET transistors only. For example, in Fig.3.4, when ABC = '101', then the path of transistors T_5 and T_6 is switched ON and input C i.e, logic '1' is passed on to the output of XNOR, and full logic '0' is generated at the output XOR through inverter I_2 and full logic '1' is restored at XNOR output through inverter I_1. For '001' combination the path of the transistors T_3 and T_4 is switched ON so that degraded logic '1' i.e input C is passed to XOR output and full

logic ‘0’ is produced at the XNOR output through the inverter I_1 and full logic ‘1’ is restored at the XOR output through inverter I_2. A similar operation applies to the rest of all the cases. For some of the input cases the passed signals at XOR or XNOR are degraded logics because of the threshold voltage loss problem of pass transistors. In such cases, the transistors in the inverters I_{n2} and I_{p1} forms, feedback loop and the transistors $I_{n1}$and I_{p2} forms another feedback loop to produce full logics at both XOR and XNOR outputs. Similar operation applies to all the remaining combinations of inputs and that can be observed from Table 3.10.

Table 3.9 State of all the transistors of Fig.3.3

A B C	**T_1**	**T_2**	**T_3**	**T_4**	**T_5**	**T_6**	**T_7**	**T_8**	**T_9**	**T_{10}**	**I_1**		**I_2**		**XOR**	**XNOR**
											I_{n1}	**I_{p1}**	**I_{n2}**	**I_{p2}**		
0 0 0	0	0	0	0	0	1	1	1	1	1	0	1	1	0	0	1
0 0 1	0	0	1	0	0	1	1	0	1	1	1	0	0	1	1	0
0 1 0	0	1	0	1	0	1	0	1	0	1	1	0	0	1	1	0
0 1 1	0	1	1	1	0	1	0	0	0	1	0	1	1	0	0	1
1 0 0	1	0	0	0	1	0	1	1	1	0	1	0	0	1	1	0
1 0 1	1	0	1	0	1	0	1	0	1	0	0	1	1	0	0	1
1 1 0	1	1	0	1	1	0	0	1	0	0	0	1	1	0	0	1
1 1 1	1	1	1	1	1	0	0	0	0	0	1	0	0	1	1	0

Note: 0 indicates OFF and 1 indicates ON for the transistors from T_1 to T_{10}, I_1 and I_2 (I_{p1}, I_{p2}, I_{n1} and I_{n2})

3.1.2.3 Proposed Structure-3 for 3 input XOR/XNOR circuit

Fig.3.5 shows the proposed structure-3 circuit using PTL logic. Fig.3.5 consists of 6 pass transistors and two inverters. The minimum number of transistors is used in this structure. Complementary control input signals are needed for Fig.3.5, since the logic is implemented with n-FinFET transistors only. For example, in Fig.3.5, when ABC = ‘101’, then the path of transistors T_4 and T_5 is switched ON and input C i.e, logic ‘1’ is passed on to the output of XNOR, and full logic ‘0’ is generated at the output XOR through inverter I_2 and full logic ‘1’ is restored at XNOR output through inverter I_1. For ‘001’ combination the path of the transistors T_1 and T_2 is switched ON so that degraded logic ‘1’ i.e input C is passed to XOR output and full logic ‘0’ is produced at

the XNOR output through the inverter I_1 and full logic '1' is restored at the XOR output through inverter I_2.

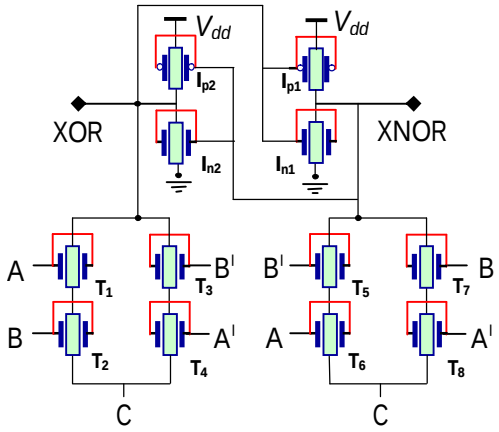


Fig.3.4 Proposed three input XOR/XNOR Structure-2

Table 3.10 Status of all the transistors of Fig.3.4

A B C	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	I_1		I_2		XOR	XNOR
									I_{n1}	I_{p1}	I_{n2}	I_{p2}		
0 0 0	0	0	1	1	1	0	0	1	0	1	1	0	0	1
0 0 1	0	0	1	1	1	0	0	1	1	0	0	1	1	0
0 1 0	0	1	0	1	0	0	1	1	1	0	0	1	1	0
0 1 1	0	1	0	1	0	0	1	1	0	1	1	0	0	1
1 0 0	1	0	1	0	1	1	0	0	1	0	0	1	1	0
1 0 1	1	0	1	0	1	1	0	0	0	1	1	0	0	1
1 1 0	1	1	0	0	0	1	1	0	0	1	1	0	0	1
1 1 1	1	1	0	0	0	1	1	0	1	0	0	1	1	0

Note: 0 indicates OFF and 1 indicates ON for the transistors from T_1 to T_8, I_1 and I_2 (I_{p1}, I_{p2}, I_{n1} and I_{n2})

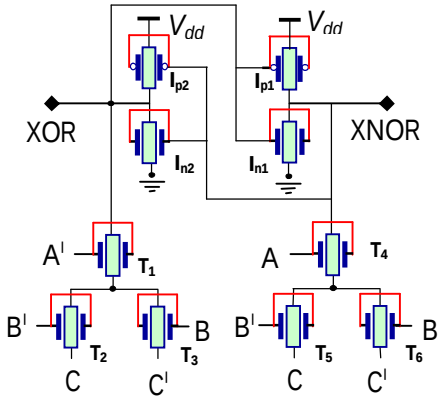


Fig.3.5 Proposed three input XOR/XNOR Structure-3

A similar operation applies to the rest of all the cases. For some of the input cases the passed signals at XOR or XNOR are degraded logics because of the threshold voltage loss problem of pass transistors. In such cases, the transistors in the inverters I_{n2} and I_{p1} forms, feedback loop and the transistors $I_{n1}$and I_{p2} forms another feedback loop to produce full logics at both XOR and XNOR outputs. Similar operation applies to all the remaining combinations of inputs and that can be explained from Table 3.11.

3.1.3 Four input differential XOR/XNOR circuits

In futuristic portable devices, the supply voltage is decreased to bring down the power dissipation. As the supply voltage is scaled down below the threshold voltage, the normal MOSFET device cannot be used due to lower I_{ON}/I_{OFF} ratio which may however reduce the static power dissipation. The power consumption becomes a major setback for further scaling. The continued reduction of MOSFET size is contributing to increased leakage current due to short channel effects. A promising alternative for MOSFET which does not suffer from these limitations is FinFET. It is used as an substitute for low power applications due to its higher sub threshold swing, extremely low off state current (I_{OFF}) and excellent sub-threshold characteristics. Hence, FinFET transistor has invited a lot of attention for analog and RF applications. Also, the dynamic power dissipation will get decreased in FinFET since the operating voltage is very low.

Table 3.11 Status of all the transistors of Fig.3.5

A B C	**T_1**	**T_2**	**T_3**	**T_4**	**T_5**	**T_6**	**I_1**		**I_2**		**XOR**	**XNOR**
							I_{n1}	**I_{p1}**	**I_{n2}**	**I_{p2}**		
0 0 0	1	1	0	0	1	0	0	1	1	0	0	1
0 0 1	1	1	0	0	1	0	1	0	0	1	1	0
0 1 0	1	0	1	0	0	1	1	0	0	1	1	0
0 1 1	1	0	1	0	0	1	0	1	1	0	0	1
1 0 0	0	1	0	1	1	0	1	0	0	1	1	0
1 0 1	0	1	0	1	1	0	0	1	1	0	0	1
1 1 0	0	0	1	1	0	1	0	1	1	0	0	1
1 1 1	0	0	1	1	0	1	1	0	0	1	1	0

Note: 0 indicates OFF and 1 indicates ON for the transistors from T_1 to T_6, I_1 and I_2 (I_{p1}, I_{p2}, I_{n1} and I_{n2})

The proposed structure of 4-bit input XOR/XNOR ([5] from list of publications) is shown in Fig.3.6. This structure is constructed using Pass Transistor Logic (PTL) and two inverters. In Fig.3.6, the logic is implemented with n-FinFET transistors and it requires complementary input signals. The proposed structure is symmetric and simple. All the inputs except the least significant input acts as control signals and the least significant input and/or the complementary of this least significant input is passed to the outputs. The back to back connection of these two inverters acts as restoration logic to get a full swing since the pass transistor logic has the threshold voltage loss problem.

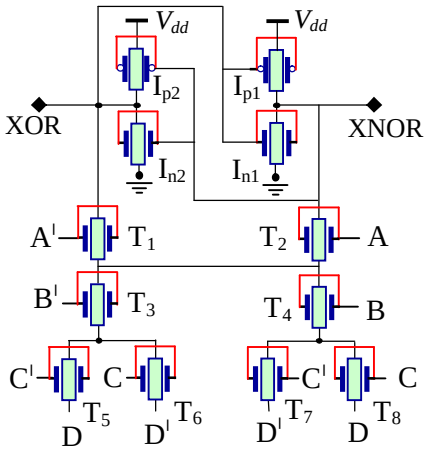


Fig.3.6 Proposed structure for 4-bit input XOR/XNOR circuit

In Fig.3.6, logic is constructed using eight n-FinFET transistors and two inverters. The working of the proposed four input XOR/XNOR circuit can be better examined by looking at the complete set of input signals ABCD = 0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111. For example, in Fig.3.6, for '0000' combination of inputs the transistors T_1, T_3 and T_5 are switched ON so that the full logic '0' i.e. input D is passed to XOR output and full logic '1' is produced at the XNOR output through the inverter I_1. When an input pattern is '0001', the transistors T_1, T_3 and T_5 are switched ON so that degraded logic '1' is generated at XOR output, then the transistors in the inverters I_{n1} and I_{p2} forms, feedback loop to produce full logics at the both XOR and XNOR outputs. For ABCD = '1101', then the transistors T_2, T_4 and T_7 are switched ON and input D^I i.e. full logic '0' is passed to the output of XNOR and full logic '1' is generated at the output XOR through inverter I_2. For ABCD = '1100', then the transistors T_2, T_4 and T_7 are switched ON and input D^I i.e the degraded logic '1' is passed to the output of XNOR then the transistors in the inverters I_{n2} and I_{p1} forms a feedback loop to produce full logics at both XOR and

XNOR outputs. A similar operation applies to the rest of the cases. It can be seen from the Table 3.12.

Table 3.12 States of transistors in the Fig.3.6 for all input combinations

A B C D	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	I_1		I_2		XOR	XNOR
									I_{n1}	I_{p1}	I_{n2}	I_{p2}		
0 0 0 0	1	0	1	0	1	0	1	0	0	1	1	0	0	1
0 0 0 1	1	0	1	0	1	0	1	0	1	0	0	1	1	0
0 0 1 0	1	0	1	0	0	1	0	1	1	0	0	1	1	0
0 0 1 1	1	0	1	0	0	1	0	1	0	1	1	0	0	1
0 1 0 0	1	0	0	1	1	0	1	0	1	0	0	1	1	0
0 1 0 1	1	0	0	1	1	0	1	0	0	1	1	0	0	1
0 1 1 0	1	0	0	1	0	1	0	1	0	1	1	0	0	1
0 1 1 1	1	0	0	1	0	1	0	1	1	0	0	1	1	0
1 0 0 0	0	1	1	0	1	0	1	0	1	0	0	1	1	0
1 0 0 1	0	1	1	0	1	0	1	0	0	1	1	0	0	1
1 0 1 0	0	1	1	0	0	1	0	1	0	1	1	0	0	1
1 0 1 1	0	1	1	0	0	1	0	1	1	0	0	1	1	0
1 1 0 0	0	1	0	1	1	0	1	0	0	1	1	0	0	1
1 1 0 1	0	1	0	1	1	0	1	0	1	0	0	1	1	0
1 1 1 0	0	1	0	1	0	1	0	1	1	0	0	1	1	0
1 1 1 1	0	1	0	1	0	1	0	1	0	1	1	0	0	1

Note: 0 indicates OFF and 1 indicates ON for the transistors from T_1 to T_8, I_1 and I_2 (I_{p1}, I_{p2}, I_{n1} and I_{n2})

3.1.4 Multi Input XOR/XNOR circuits

The proposed structure is presented in Fig.3.7 ([4] from list of publications). It is constructed using Pass Transistor Logic (PTL) and two inverters. In Fig.3.7, the logic is implemented with FinFET transistors and it requires complementary input signals. The proposed structure is symmetric and simple. All the inputs except least significant input acts as control signals and least significant input and/or complementary of least significant input is passed to the outputs. The back to back connection of two inverters acts as restoration logic to get a full swing since the PTL have the threshold voltage loss problem.

Fig.3.8 shows the structure of 8-bit input XOR/XNOR logic circuit. Fig.3.8 operation is similar to the structure in Fig.3.6 which is already presented, so that they are not included here to avoid repetition.

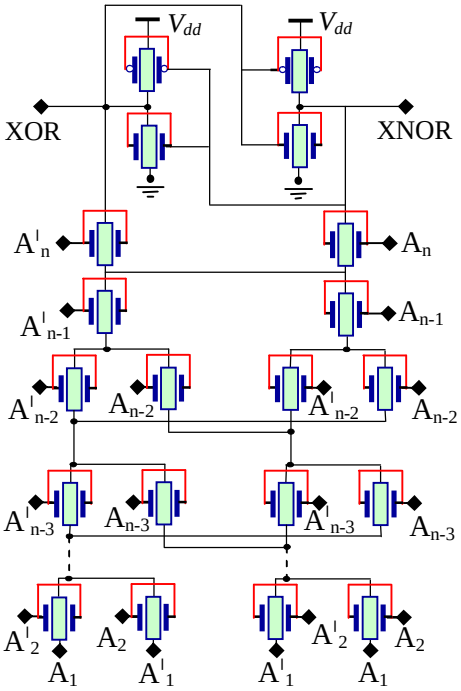


Fig.3.7 General Structure for multi input XOR/XNOR Circuit

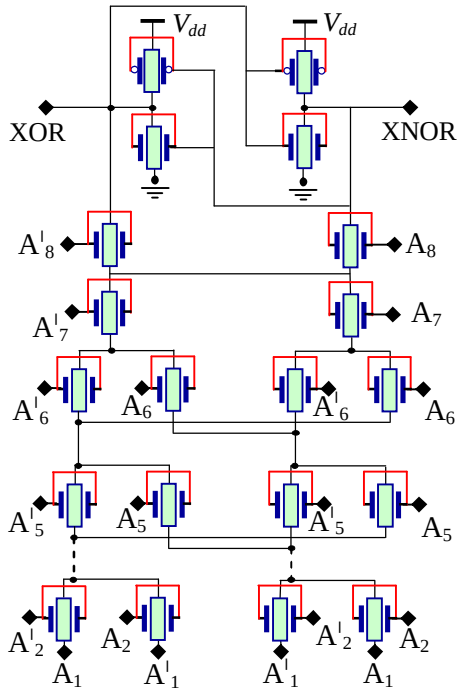


Fig.3.8 Structure for 8-input XOR/XNOR circuit

3.1.5 Direct three input XOR circuits

Advanced logic function can be implemented using the feature of MOS transistor to work as a simple switch that can avail the advantage of being simple and fast. Complex logic gates can be implemented with minimum numbers of transistors thus being simple and with reduced parasitic capacitance. A switch can be implemented

with a single transistor called pass transistor logic (PTL) or with two transistors as transmission gate logic (TGL). Here three circuits are proposed using TGL and PTL logics ([10] from list of publications).

3.1.5.1 Proposed-1 three input XOR logic gate

In the first proposed three input XOR logic gate the logic is completely based on the transmission gate logic and inverters are used for obtaining the complementary of the inputs. TGL switch provides better noise margin and is very efficient in implementing complex logic gates. The area overhead and speed are the basic disadvantages of full complementary static CMOS designs that can be overcome by TGL designs.

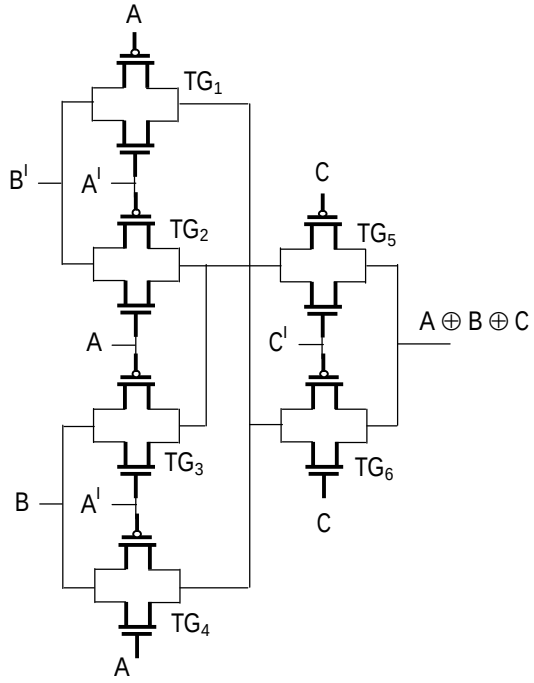


Fig.3.9 Proposed-1 three input XOR logic gate

Fig.3.9 shows the circuit schematic of the proposed-1 three input XOR logic gate using TGL. It carries out a perfect full swing operation for every input pattern. The basic logic at the back of the proposed gate is that the inputs B and B^{l} are passed to the output terminal depending upon the inputs A and C which are used as the control terminals to the transmission gates. Hence, based on the input signals A and C the transmission gates acting as switches will be in either ON or OFF state, thus allowing B or B^{l} to the output. So, the design is carefully implemented to obtain the truth table of three input XOR logic.

The working of the proposed three inputs XOR logic gate can be better examined by looking at the total set of input signals ABC= 000, 001, 010, 011,100, 101,110, 111. Firstly, look at the case when A=B=C=0, then the transmission gates TG_1, TG_3, TG_5 are ON hence input B is passed from TG_3 and TG_5 to the end product. Next, consider

the case when A=0, B=0 and C=1, then the transmission gates TG_1, TG_3, TG_6 are ON hence input B^I is passed from TG_1 and TG_6 to the output. A similar argument applies to the rest of the cases whose output conforms to the three input Exclusive-OR logic operation. All the possible states of the inputs are summarized in Table 3.13.

Table 3.13 Active state of the proposed Fig.3.9 circuit in all cases

ABC	TG_1	TG_2	TG_3	TG_4	TG_5	TG_6	A xor B xor C
000	ON	OFF	ON	OFF	ON	OFF	0
001	ON	OFF	ON	OFF	OFF	ON	1
010	ON	OFF	ON	OFF	ON	OFF	1
011	ON	OFF	ON	OFF	OFF	ON	0
100	OFF	ON	OFF	ON	ON	OFF	1
101	OFF	ON	OFF	ON	OFF	ON	0
110	OFF	ON	OFF	ON	ON	OFF	0
111	OFF	ON	OFF	ON	OFF	ON	1

3.1.5.2 Proposed-2 three input XOR logic gate

In the proposed-2 PTL is used to implement the circuit. As a pass transistor switch requires only one transistor unlike TGL which requires two, hence the transistor count reduces much by implementing the proposed logic by using pass transistor switches. But logic implemented using PTL fails to give full swing due to loss of threshold, to overcome this loss buffer is used at the output. Another advantage of using an inverter is it improves the driving capability of the cell. This implementation is shown in Fig. 3.10. The working of this proposed-2 is also similar to the proposed-1 circuit that the inputs A and C will act as control signals to the pass transistors that act as switches allowing B or B^I to the output depending on the input state.

Table 3.14 Active state of the proposed Fig.3.10 circuit in all cases

ABC	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	A xor B xor C
000	OFF	OFF	OFF	OFF	ON	ON	ON	ON	0
001	OFF	ON	ON	OFF	ON	OFF	ON	OFF	1
010	OFF	OFF	OFF	OFF	ON	ON	ON	ON	1
011	OFF	ON	ON	OFF	ON	OFF	ON	OFF	0
100	ON	OFF	OFF	ON	OFF	ON	OFF	ON	1
101	ON	ON	ON	ON	OFF	OFF	OFF	OFF	0
110	ON	OFF	OFF	ON	OFF	ON	OFF	ON	0
111	ON	ON	ON	ON	OFF	OFF	OFF	OFF	1

As in this proposed-2 circuit, we are using an additional inverter at the output, hence speed reduces and power consumption increases compared to the proposed-1 circuit. All the possible states of the inputs are summarized in Table 3.14 and the operation of Fig.3.10 can be examined from Table 3.14.

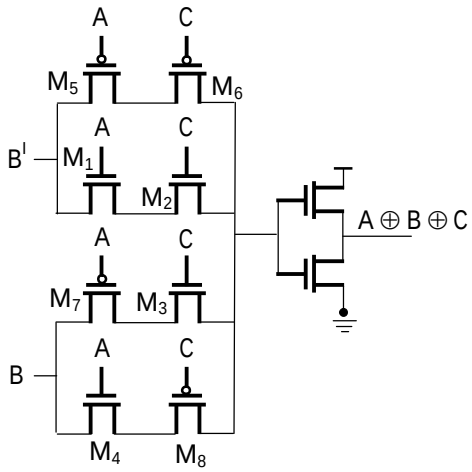


Fig.3.10 Proposed-2 three input XOR logic gate

3.1.5.2 Proposed-3 three input XOR logic gate

From the proposed-2 circuit, we can eliminate two assisting transistors M_3 and M_8 and reconnect the transistors M_7 and M_4 to M_2 and M_6 respectively. Without making any change in the logic the proposed circuit now requires only 10 transistors to represent the three input XOR logic truth table. This forms the proposed-3 circuit in Fig.3.11.

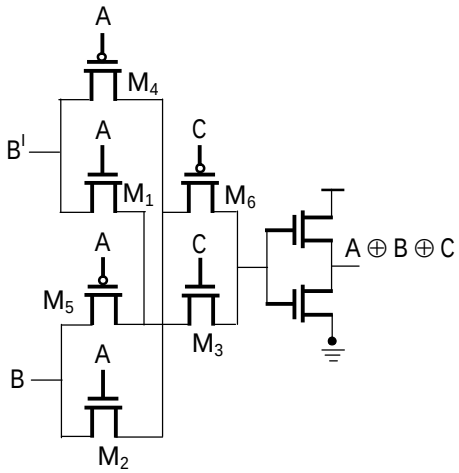


Fig.3.11 Proposed-3 three input XOR logic gate

The circuit still delivers full swing at the output. It saves area as well as power. It also has the good driving capability as its output is also driven through an inverter. The operation is summarized in Table 3.15 by including active states of all the transistors for complete set of inputs.

Table 3.15 Active state of the proposed Fig.3.11 circuit in all cases

ABC	M_1	M_2	M_3	M_4	M_5	M_6	A xor B xor C
000	OFF	OFF	OFF	ON	ON	ON	0
001	OFF	OFF	ON	ON	ON	OFF	1
010	OFF	OFF	OFF	ON	ON	ON	1
011	OFF	OFF	ON	ON	ON	OFF	0
100	ON	ON	OFF	OFF	OFF	ON	1
101	ON	ON	ON	OFF	OFF	OFF	0
110	ON	ON	OFF	OFF	OFF	ON	0
111	ON	ON	ON	OFF	OFF	OFF	1

3.1.6 Two input XNOR circuits

Here we proposed two low-power XNOR gates using six transistors ([7] from list of publications). The proposed gate circuits used pass transistor logic (PTL) concept and current source inverter and current sink inverter for complementary input. The pass transistor design uses small transistor count and operates with very low-power and results high-performance.

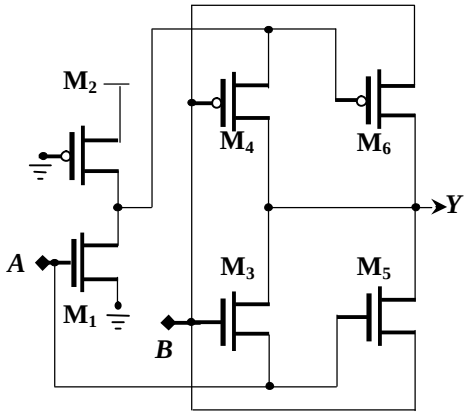


Fig.3.12 XNOR Circuit with Current Source Inverter

Fig.3.12 and Fig.3.13 show the circuit schematics of the proposed XNOR with current source inverter and current sink inverter. In both of these circuits, the output Y generates logic ‘1’ when A and B are logic ‘0’. For this condition, transistor M_4 ON, and it will pass a strong logic ‘1’ to the output Y that is the complemented version of A. And when A is logic ‘0’ and B is logic ‘1’, only transistor M_3 is ON and a strong logic ‘0’ is passed to the output Y. And for A logic ‘1’ and B logic ‘0’, transistors M_4, M_5 and M_6 are ON but the transistor M_5 will pass a strong logic ‘0’ to the output

Y. And when both inputs are logic ‘1’, the transistors M_3, M_5 and M_6 are ON but M_6 passes a strong logic ‘1’ to the output Y. The complemented signal is generated from the current source inverter in Fig.3.12. The gate of transistors M_1 is connected to Gnd, so it is always ON. If A is logic ‘0’, M_1 OFF so V_{dd} is passed or if A is logic ‘1’, M_1 ON so Gnd is passed to source of M_4 and gate of M_6. In Fig.3.13, the complemented signal is generated from the current sink inverter. Here V_{dd} is connected to the gate of M_1 so it is always ON. Depending on the A value either V_{dd} or Gnd reached to M_4 and M_6. These two circuits perform a perfect full swing operation for every input pattern.

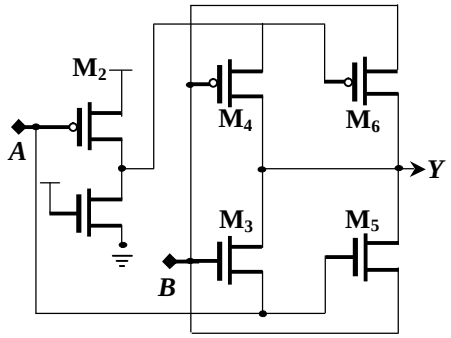


Fig 3.13 XNOR Circuit with Current Sink Inverter

3.2 Fault Secure, Full Rail to Rail Swing, High Performance and Better PDP XOR/XNOR Circuit: Self -checking XOR/XNOR Circuits

Fault or error can be determined by calculating the output of the circuits in self-checking circuits. In several operations, an error checking procedures use Totally Self-Checking (TSC) circuits. TSC circuits are used to detect errors simultaneously with normal operation. The concept of TSC circuits was first proposed in (D. A. Anderson et al, 1973) and then generalized in (A. P. Chandrakasan et al, 1992), as follows.

Fault secure definition: For a set of faults F, a circuit is fault-secure, for any valid code word of input and for any single fault, the circuit either gives an invalid output code word, or doesn’t make the error in the output.

Self testing definition: For a set of faults F, a circuit is self-testing, for every fault in F, the circuit gives invalid code word at output for any one valid input code word. It means any single mistake can be detectable by some valid code word as input.

Totally self checking definition: If a circuit satisfies fault secure and self-testing properties, then it is called as totally self-checking.

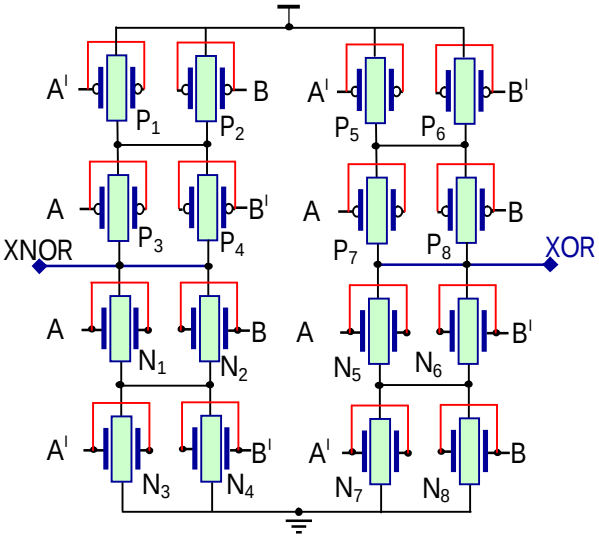


Fig.3.14 Self checking XOR/XNOR circuit

Table 3.16 State of Transistors in the proposed Fig.3.14

AA^IBB^I	N1	N2	N3	N4	N5	N6	N7	N8	P1	P2	P3	P4	P5	P6	P7	P8	XOR	XNOR
0 0 0 0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
0 0 0 1	0	0	0	1	0	1	0	0	1	1	1	0	1	0	1	1	1	1
0 0 1 0	0	1	0	0	0	0	0	1	1	0	1	1	1	1	1	0	1	1
0 1 0 0	0	0	1	0	0	0	1	0	0	1	1	1	0	1	1	1	1	1
0 1 0 1	0	0	1	1	0	1	1	0	0	1	1	0	0	0	1	1	0	1
0 1 1 0	0	1	1	0	0	0	1	1	0	0	1	1	0	1	1	0	1	0
0 1 1 1	0	1	1	1	0	1	1	1	0	0	1	0	0	0	1	0	0	0
1 0 0 0	1	0	0	0	1	0	0	0	1	1	0	1	1	1	0	1	1	1
1 0 0 1	1	0	0	1	1	1	0	0	1	1	0	0	1	0	0	1	1	0
1 0 1 0	1	1	0	0	1	0	0	1	1	0	0	1	1	1	0	0	0	1
1 0 1 1	1	1	0	1	1	1	0	1	1	0	0	0	1	0	0	0	0	0
1 1 0 0	1	0	1	0	1	0	1	0	0	1	0	1	0	1	0	1	1	1
1 1 0 1	1	0	1	1	1	1	1	0	0	1	0	0	0	0	0	1	0	0
1 1 1 0	1	1	1	0	1	0	1	1	0	0	0	1	0	1	0	0	0	0
1 1 1 1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0

Note: 0 indicates OFF and 1 indicates ON for the transistors from N_1 to N_8 & P_1 to P_8

Code disjoint definition: If a circuit gives valid output code words for valid input code words and invalid output code words for invalid input code words.

Totally self-checking checker definition: If a circuit satisfies self-testing and code-disjoint properties, then it is called as a totally self checking checker.

Fig.3.14 shows the proposed self-checking XOR/XNOR circuit ([7] from list of publications). It has both differential inputs and differential outputs. The operation of the proposed low power self-checking XOR/XNOR structure can be studied by

considering the total set of input signals AA^1BB^1 from 0000 to 1111 which is shown in Table 3.16.

3.3 Low Voltage, Low Power, Full Rail to Rail Swing and Better PDP XOR/XNOR Circuits: Sub Threshold Operated DCVSL XOR/XNOR Circuits

The growing need of energy efficient circuits that work at low voltages with low power consumption has led to an idea of low voltage circuits. MOS circuits are found to be the perfect solution to fit into this idea. But as the threshold voltage cannot be scaled down proportionally with respect to the supply voltage of the MOS circuits, the MOS circuits face difficulties in the sub-threshold region of operation. Several methods were proposed to make the operation of MOS circuits possible at sub-threshold region. One among such methods is voltage-boost logic where the internal voltage of some nodes is boosted to a voltage level higher than the supply voltage. Also, different other technologies like FinFET, TFET, HTFET are introduced and are at research level these days to overcome the aforesaid problems with FinFETs and MOSFETs (H. Lui et al, 2013, X. Lin et al, 2014). But as these new technologies are at research level, it is worth to choose to work with MOSFETs. The circuit in (J. W. Kim et al, 2012) is a best example of this voltage boosted logic circuit. In this circuit for the given supply voltage the source-body voltage is slightly forward biased, allowing the transistor threshold voltage to be reduced, by boosting the source voltages of those transistors below the ground. In this the logic used is a CMOS differential logic, which is compared to be better than the logic styles used in (L. G. Heller et al, 1984, R. H. Krambeck, et al, 1982, A. Wang et al, 2004, J. W. Kim et al, 2008) under several issues like speed performance, reduced overdrive voltage (V_{gs}-V_{th}), the inefficiency of the logic functioning embedded etc.

Differential cascade voltage switch logic (DCVSL) gates shown in Fig.3.15 operate at all times with both true and complementary signals i.e., it has logic flexibility (K. M. Chu et al, 1987). DCVSL gates have the potential of high fan-in thus reducing logic path, has high speed and it has the capability of producing completion signals for asynchronous operations. The logic used in this proposed circuit is similar to the boosted CMOS differential logic (BCDL) shown in Fig.3.16.

In the proposed structures ([3] from list of publications) voltage boosting blocks are negative voltage generators. It generates a negative voltage approximately equal to the applied V_{dd} thus providing the required gate-source voltage of the transistors in the logic gate to make them to conduct at voltages lower than their threshold voltage. In

the proposed circuit-1, it is a two transistor circuit with a capacitor which provides sufficient negative voltage to the logic attached to it. In the proposed circuit-2, it is a three transistor and one capacitor circuit providing the necessary functionality.

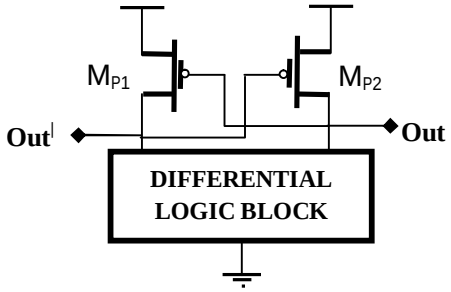


Fig.3.15 Conventional DCVSL (K. M. Chu et al, 1987)

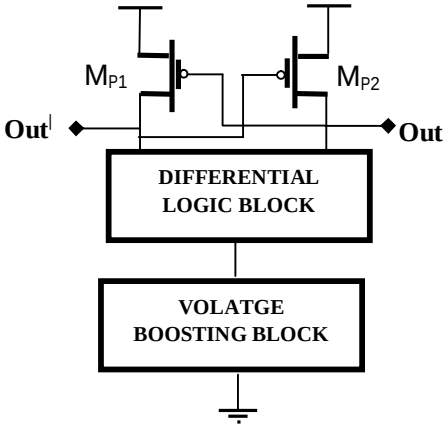
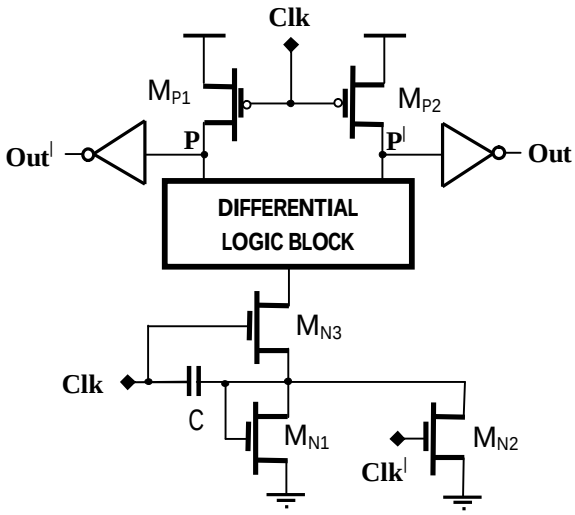


Fig.3.16 BCDL Structure (J. W. Kim et al, 2012)

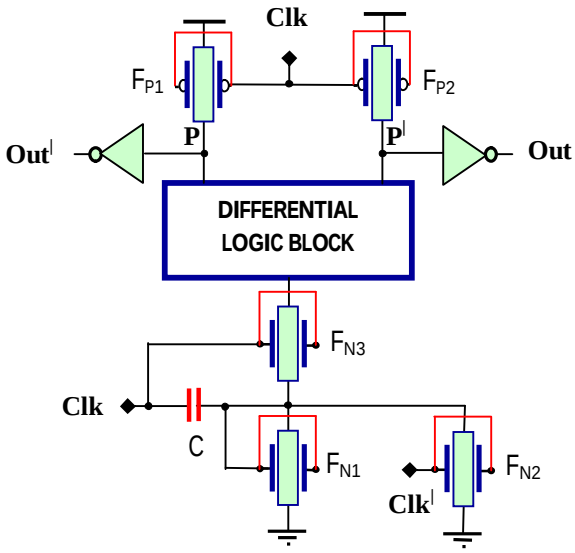
3.3.1 Sub-threshold voltage operated DCVSL proposed circuit-1

Figure 3.17(a) shows the structure of the proposed circuit-1 with MOSFETs. It has a pre-charged differential block followed by differential block and a negative voltage generator block. The pre-charged differential block has transistors M_{P1}, M_{P2} and two inverters in the output section. The negative voltage generator block has transistors M_{N1}, M_{N2} and a capacitor, a transistor M_{N3} bridges between these two blocks. Another inverter may be used in the output section to decrease the delay of the transient.

The circuit operates in two phases, initially when the clk signal goes low the pre-charged differential block operates and draws V_{dd} to the nodes, P and P', as the PMOS transistors M_{P1} and M_{P2} starts conducting. When the clk signal goes high the capacitor charges and the transistor M_{N1} in the negative voltage generator block conducts and then the transistor M_{N3} passes the negative voltage generated to the transistors in the logic gate, increasing their gate-source voltage and thus enabling those transistors to conduct at voltages, near to the transistor threshold voltage.



(a) MOSFETs version



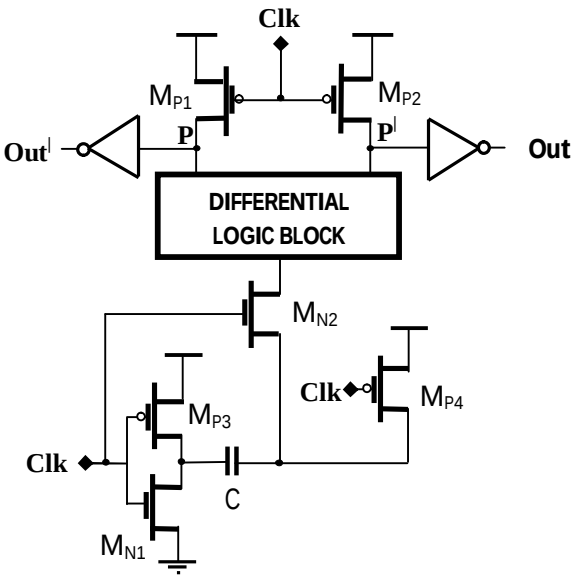
(b) FinFETs version

Fig.3.17 Sub-threshold voltage operated DCVSL proposed circuit-1

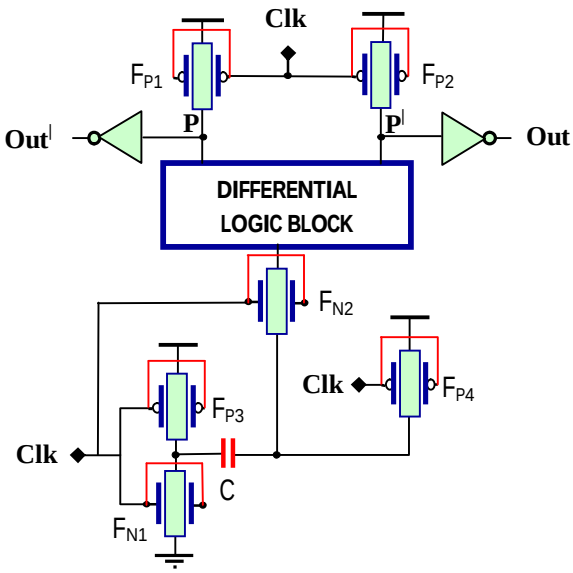
The transistors in the logic gate, then evaluate the required logic. Now, when the clock (Clk) goes low the transistor M_{N2} conducts thus providing a discharge path for the capacitor without disturbing its further operation. Thus, the circuit operates to give complementary outputs at low voltages. Likewise, FinFETs based proposed circuit-1 shown in Fig.3.16(b) has similar operations.

3.3.2 Sub-threshold voltage operated DCVSL proposed circuit-2

The proposed circuit-2 as shown in Fig.3.18(a) also consists of a pre-charged differential logic block followed by a differential logic block and a negative voltage generator block. Pre-charged block has transistors M_{P1}, M_{P2} and two inverters.



(a) MOSFETs version



(b) FinFETs version

Fig.3.18 Sub-threshold voltage operated DCVSL proposed circuit-2

The negative voltage generator has transistors M_{N1}, M_{P3}, M_{P4} and a capacitor. A transistor M_{N2} connects these two blocks. When the clk signal is low the transistor M_{P3} is ON and it charges the capacitor up to V_{dd}, at the same time transistor M_{P4} is ON which draws the same $+V_{dd}$ to the capacitor. Now when the clk goes high transistor M_{N1} is ON thus pulling ground to the capacitor providing negative voltage to the transistor M_{N2} which in turn passes the same voltage to the logic gate. The logic gate evaluates the proper logic. Likewise, FinFETs based proposed circuit-2 shown in Fig.3.18(b) had similar operation. The XOR and XNOR gates have been got tested

using MOSFETs and FinFETs version at low voltages and the results compared. The results obtained had a trade-off between delay and power.

3.4 Multi Valued Logic, Full Rail to Rail Swing and Better PDP XOR/XNOR Circuits: Ternary XOR and XNOR Gates

Many multi-valued logic, arithmetic circuits using CNTFET have been demonstrated in the literature. However, the multi-valued logic circuits could be of more interest in the CNTFET nanotechnology. The multiple-threshold voltage design technique is the most suitable method for designing voltage-mode MVL circuits. The CNTFET devices can obtain the desired threshold voltage by choosing specific diameters for their nanotubes. Correspondingly, in MVL many logical and arithmetic operations could be accomplished with higher speed and smaller number of computation stages.

The critical problems of the binary logic in designing complex, large and compact chips are the interconnections and pin-out difficulties that limit the number of connections inner side and outer side of the circuits. By employing MVL, wires carry information to a greater extent, which expedite savings in the count of interconnections and in the shield between them, and also pins convey more information that consequences a reduction in the number of pins. Moreover, MVL storage allows storing more information per memory cell. In addition, many practical applications like robotics, decision systems and process control can be realized more efficiently by choosing MVL systems. Hence, this MVL can be applied to work out the binary logic challenges more efficiently.

3.4.1 A brief of MVL design

Let us consider an n-valued function F(X) with k variables, where X = {x_1, x_2, x_3, …, x_k} and each x_i (i = 1, 2, …k) can adopt the values from M = {0, 1, 2, …, n − 1}. Therefore, the function F(X) is a mapping *f*: $M_k \rightarrow M$ and consequently there are n^{M_k} different functions possible in the set f. The ternary logic is a common MVL, which comprises three significant logic levels. The three logic levels can be considered as ‘0’, ‘1’ and ‘2’ symbols, which are counterpart to 0, $V_{dd}/2$ and V_{dd} voltage levels. Negative, standard and positive are the three different types of logics defined for the ternary logic. In this static complementary and pseudo types of negative, standard and positive ternary XOR and XNOR circuits are proposed ([2] from list of publications).

3.4.2 Proposed normal Ternary XOR

3.4.2.1 Standard Ternary XOR (STXOR)

Fig. 3.19 presents the CNTFET based proposed STXOR logic gate circuit. It used 18 CNTFETs. The logic of Fig.3.19 is expressed in (3.1). Table 3.17 shows the operation and the truth table of Fig. 3.19.

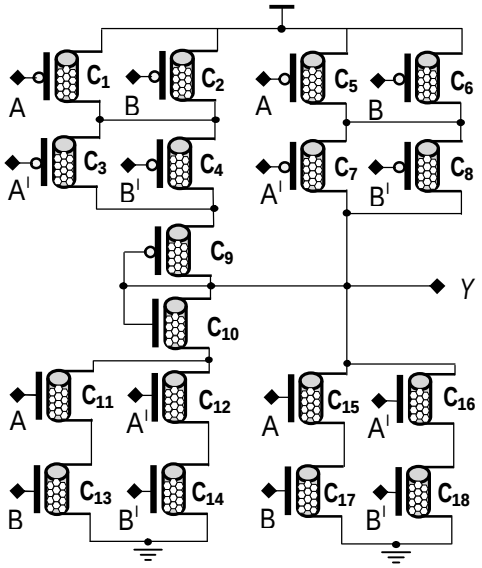


Fig.3.19 CNTFET based Standard Ternary XOR circuit

Table 3.17 Truth table for STXOR

I/P A, B	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**
CNTFET	0 0	0 1	0 2	10	1 1	1 2	2 0	2 1	2 2
C_1	ON	ON	ON	ON	ON	ON	OFF	OFF	OFF
C_2	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF
C_3	OFF	OFF	OFF	ON	ON	ON	ON	ON	ON
C_4	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON
C_5	ON	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF
C_6	ON	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF
C_7	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON	ON
C_8	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	ON
C_9	ON	ON	ON	ON	ON	ON	ON	ON	ON
C_{10}	ON	ON	ON	ON	ON	ON	ON	ON	ON
C_{11}	OFF	OFF	OFF	ON	ON	ON	ON	ON	ON
C_{12}	ON	ON	ON	ON	ON	ON	OFF	OFF	OFF
C_{13}	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON
C_{14}	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF
C_{15}	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON	ON
C_{16}	ON	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF
C_{17}	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	ON
C_{18}	ON	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF
o/p Y	0	1	2	1	1	1	2	1	0

$$Y = \begin{cases} 0; \text{ if } A = B \neq 1 \\ 2; \text{if } A = 0, B = 2 \text{ and } A = 2, B = 0 \\ 1; \text{OtherWise} \end{cases} \quad (3.1)$$

3.4.2.2 Positive Ternary XOR (PTXOR) and Negative Ternary XOR (NTXOR)

Fig.3.20 depicts the common structure for PTXOR and NTXOR circuits. This structure used 8 CNT transistors. PTXOR logic is expressed in (3.2). Table 3.18 shows the operation of Fig.3.20 for PTXOR.

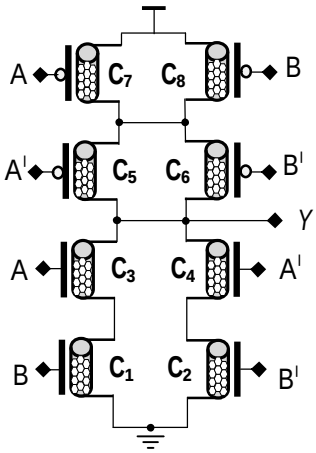


Fig.3.20 CNTFET based PTXOR and NTXOR structure

$$Y = \begin{cases} 2; \text{if } A = B \neq 1 \\ 0; \text{OtherWise} \end{cases} \quad (3.2)$$

Table 3.18 Truth table for PTXOR

I/P A, B	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**
CNTFET	0 0	0 1	0 2	10	1 1	1 2	2 0	2 1	2 2
C_1	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	ON
C_2	ON	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF
C_3	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON	ON
C_4	ON	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF
C_5	OFF	OFF	OFF	ON	ON	ON	ON	ON	ON
C_6	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON
C_7	ON	ON	ON	ON	ON	ON	OFF	OFF	OFF
C_8	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF
o/p Y	0	2	2	2	2	2	2	2	0

$$Y = \begin{cases} 0; \text{if } A = 0, B = 2 \text{ and } A = 2, B = 0 \\ 2; \text{OtherWise} \end{cases} \quad (3.3)$$

The operation of NTXOR is shown in (3.3) and Table 3.19 shows the operation of Fig.3.20 for NTXOR.

Table 3.19 Truth table for NTXOR

I/P A, B	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**
CNTFET	0 0	0 1	0 2	10	1 1	1 2	2 0	2 1	2 2
C_1	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON
C_2	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF
C_3	OFF	OFF	OFF	ON	ON	ON	ON	ON	ON
C_4	ON	ON	ON	ON	ON	ON	OFF	OFF	OFF
C_5	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON	ON
C_6	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	ON
C_7	ON	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF
C_8	ON	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF
o/p Y	0	0	2	0	0	0	2	0	0

3.4.3 Proposed normal Ternary XNOR

3.4.3.1 Standard Ternary XNOR (STXNOR)

Fig.3.21 displays the CNTFET based proposed STXNOR circuit and its logic is expressed in (3.4). It needed 18 CNTFETs and its operation is shown in Table 3.20.

$$Y = \begin{cases} 0; if \;\; A = 0, B = 2 \; and \;\; A = 2, B = 0 \\ 2; if \;\; \mathrm{A} = \mathrm{B} \neq 1 \\ 1; OtherWise \end{cases} \tag{3.4}$$

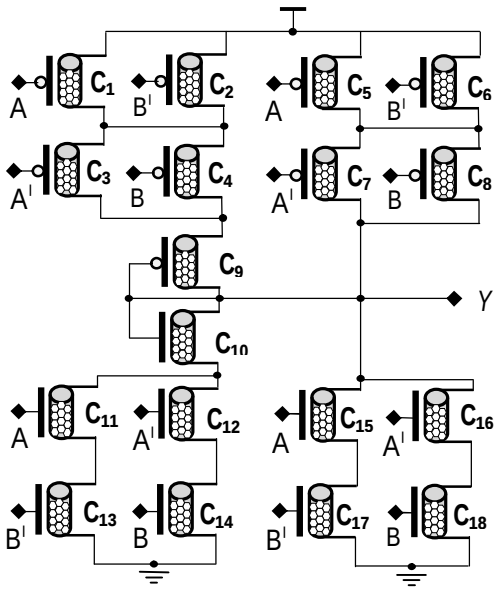


Fig.3.21 CNTFET based Standard ternary XNOR circuit

Table 3.20 Truth table for STXNOR

I/P A, B	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**
CNTFET	0 0	0 1	0 2	10	1 1	1 2	2 0	2 1	2 2
C_1	ON	ON	ON	ON	ON	ON	OFF	OFF	OFF
C_2	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON
C_3	OFF	OFF	OFF	ON	ON	ON	ON	ON	ON
C_4	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF
C_5	ON	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF
C_6	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	ON
C_7	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON	ON
C_8	ON	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF
C_9	ON	ON	ON	ON	ON	ON	ON	ON	ON
C_{10}	ON	ON	ON	ON	ON	ON	ON	ON	ON
C_{11}	OFF	OFF	OFF	ON	ON	ON	ON	ON	ON
C_{12}	ON	ON	ON	ON	ON	ON	OFF	OFF	OFF
C_{13}	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF
C_{14}	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON
C_{15}	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON	ON
C_{16}	ON	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF
C_{17}	ON	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF
C_{18}	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	ON
o/p Y	2	1	0	1	1	1	0	1	2

3.4.3.2 Positive Ternary XNOR (PTXNOR) and Negative Ternary XNOR (NTXNOR)

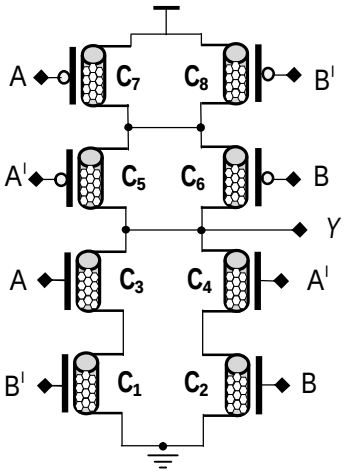


Fig.3.22 CNTFET based PTXNOR and NTXNOR structure

Fig.3.22 dipicts a common structure for PTXNOR and NTXNOR logic gates. This structure used 8 CNTFETs. PTXNOR logic is given in (3.5) and Table 3.21 shows the operation of PTXNOR. The logic of NTXNOR is given in (3.6) and its operation is shown in Table 3.22.

Table 3.21 Truth table for PTXNOR

I/P A, B	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**
CNTFET	0 0	0 1	0 2	10	1 1	1 2	2 0	2 1	2 2
C1	ON	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF
C2	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	ON
C3	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON	ON
C4	ON	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF
C5	OFF	OFF	OFF	ON	ON	ON	ON	ON	ON
C6	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF
C7	ON	ON	ON	ON	ON	ON	OFF	OFF	OFF
C8	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON
o/p Y	2	2	0	2	2	2	0	2	2

$$Y = \begin{cases} 0; if \ \ A = 0, B = 2 \ and \ A = 2, B = 0 \\ 2; OtherWise \end{cases} \tag{3.5}$$

$$Y = \begin{cases} 2; if \ \ A = B \neq 1 \\ 0; OtherWise \end{cases} \tag{3.6}$$

Table 3.22 Truth table for NTXNOR

I/P A, B	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**
CNTFET	0 0	0 1	0 2	10	1 1	1 2	2 0	2 1	2 2
C_1	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF
C_2	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON
C_3	OFF	OFF	OFF	ON	ON	ON	ON	ON	ON
C_4	ON	ON	ON	ON	ON	ON	OFF	OFF	OFF
C_5	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON	ON
C_6	ON	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF
C_7	ON	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF
C_8	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	ON
o/p Y	2	0	0	0	0	0	0	0	2

3.4.4 Proposed Pseudo Ternary XOR

3.4.4.1 PSEUDO STXOR

Fig.3.23 displays the CNTFET based proposed Pseudo STXOR circuit. This circuit took 10 CNTFETs. The logic of this circuit is given in (3.7). Table 3.23 shows its operation.

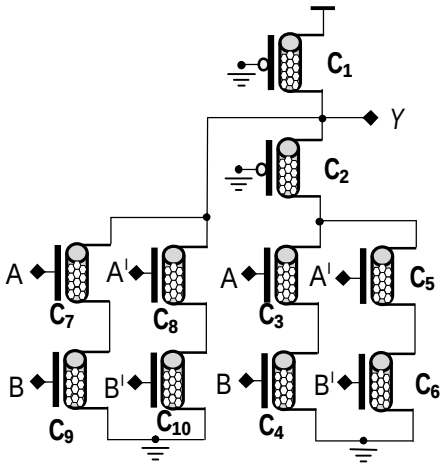


Fig.3.23 CNTFET based Pseudo STXOR

$$Y = \begin{cases} 0; \text{if } A = B \neq 1 \\ 2; if \ A = 0, B = 2 \text{ and } A = 2, B = 0 \\ 1; OtherWise \end{cases} \quad (3.7)$$

Table 3.23 Truth table for Pseudo STXOR

I/P A, B	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**
CNTFET	0 0	0 1	0 2	10	1 1	1 2	2 0	2 1	2 2
C_1	ON	ON	ON	ON	ON	ON	ON	ON	ON
C_2	ON	ON	ON	ON	ON	ON	ON	ON	ON
C_3	OFF	OFF	OFF	ON	ON	ON	ON	ON	ON
C_4	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON
C_5	ON	ON	ON	ON	ON	ON	OFF	OFF	OFF
C_6	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF
C_7	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON	ON
C_8	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	ON
C_9	ON	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF
C_{10}	ON	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF
o/p Y	0	1	2	1	1	1	2	1	0

3.4.4.2 PSEUDO PTXOR and NTXOR

Fig.3.24 shows a common structure for Pseudo PTXOR and NTXOR gates. It needed 5 CNTFET transistors. The logic of PTXOR is given in (3.8). Table 3.24 shows the operation of Pseudo PTXOR. The logic expression (3.9) and Table 3.25 shows the operation of Pseudo NTXOR.

$$Y = \begin{cases} 2; if \ A = B \neq 1 \\ 0; OtherWise \end{cases} \quad (3.8)$$

$$Y = \begin{cases} 0; if \ A = 0, B = 2 \ and \ A = 2, B = 0 \\ 2; OtherWise \end{cases} \quad (3.9)$$

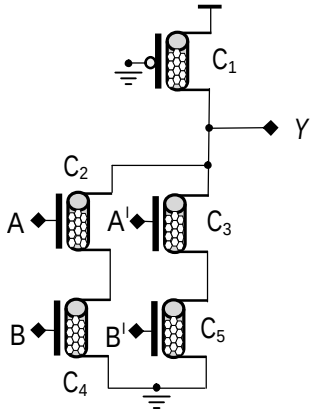


Fig.3.24 CNTFET based Pseudo PTXOR and NTXOR

Table 3.24 Truth table for Pseudo PTXOR

I/P A, B	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**
CNTFET	0 0	0 1	0 2	10	1 1	1 2	2 0	2 1	2 2
C1	ON	ON	ON	ON	ON	ON	ON	ON	ON
C2	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON	ON
C3	ON	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF
C4	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	ON
C5	ON	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF
o/p Y	0	2	2	2	2	2	2	2	0

Table 3.25 Truth table for Pseudo NTXOR

I/P A, B	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**
CNTFET	0 0	0 1	0 2	10	1 1	1 2	2 0	2 1	2 2
C_1	ON	ON	ON	ON	ON	ON	ON	ON	ON
C_2	OFF	OFF	OFF	ON	ON	ON	ON	ON	ON
C_3	ON	ON	ON	ON	ON	ON	OFF	OFF	OFF
C_4	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON
C_5	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF
o/p Y	0	0	2	0	0	0	2	0	0

3.4.5 Proposed Pseudo Ternary XNOR

3.4.5.1 PSEUDO STNXOR

Fig 3.25 shows the circuit of Pseudo ternary XNOR. It used 10 CNTFET transistors which are shown in Fig 3.25. Its operation is given in Table 3.26 and equation (3.10).

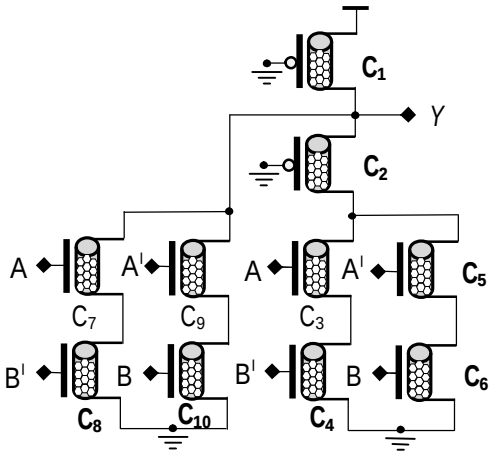


Fig.3.25 CNTFET based Pseudo ternary XNOR

Table 3.26 Truth table for Pseudo STXNOR

I/P A, B	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**
CNTFET	0 0	0 1	0 2	10	1 1	1 2	2 0	2 1	2 2
C_1	ON	ON	ON	ON	ON	ON	ON	ON	ON
C_2	ON	ON	ON	ON	ON	ON	ON	ON	ON
C_3	OFF	OFF	OFF	ON	ON	ON	ON	ON	ON
C_4	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF
C_5	ON	ON	ON	ON	ON	ON	OFF	OFF	OFF
C_6	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON
C_7	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON	ON
C_8	ON	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF
C_9	ON	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF
C_{10}	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	ON
o/p Y	2	1	0	1	1	1	0	1	2

$$Y = \begin{cases} 0; if\ \ A = 0, B = 2\ and\ A = 2, B = 0 \\ 2; if\ \ \mathrm{A} = \mathrm{B} \neq 1 \\ 1; OtherWise \end{cases} \qquad (3.10)$$

3.4.5.2 Pseudo PTXNOR and NTXNOR

Fig.3.26 gives the common structure for Pseudo PTXNOR and NTXNOR logic gates. This structure used 5 CNTFET transistors. Pseudo PTXNOR operation is expressed in (3.11) and in Table 3.27. Pseudo NTXNOR operation is shown in Table 3.28 which is expressed in (3.12).

$$Y = \begin{cases} 0; if\ \ A = 0, B = 2\ and\ A = 2, B = 0 \\ 2; OtherWise \end{cases} \qquad (3.11)$$

$$Y = \begin{cases} 2; if \ \ A = B \neq 1 \\ 0; OtherWise \end{cases} \tag{3.12}$$

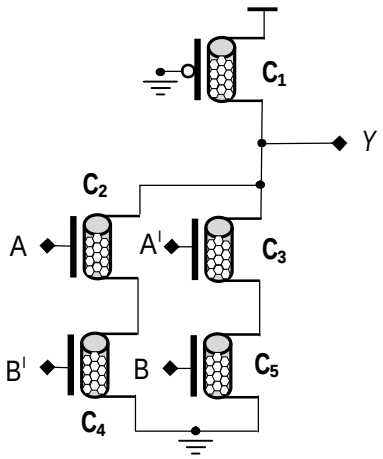


Fig.3.26 CNTFET based Pseudo PTXNOR and NTXNOR.

Table 3.27 Truth table for Pseudo PTXNOR

I/P A, B	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**
CNTFET	0 0	0 1	0 2	10	1 1	1 2	2 0	2 1	2 2
C_1	ON	ON	ON	ON	ON	ON	ON	ON	ON
C_2	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON	ON
C_3	ON	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF
C_4	ON	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF
C_5	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	ON
o/p Y	2	2	0	2	2	2	0	2	2

Table 3.28 Truth table for Pseudo NTXNOR

I/P A, B	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**	**A B**
CNTFET	0 0	0 1	0 2	10	1 1	1 2	2 0	2 1	2 2
C_1	ON	ON	ON	ON	ON	ON	ON	ON	ON
C_2	OFF	OFF	OFF	ON	ON	ON	ON	ON	ON
C_3	ON	ON	ON	ON	ON	ON	OFF	OFF	OFF
C_4	ON	ON	OFF	ON	ON	OFF	ON	ON	OFF
C_5	OFF	ON	ON	OFF	ON	ON	OFF	ON	ON
o/p Y	2	0	0	0	0	0	0	0	2