International Journal of Scientific Research in Science, Engineering and Technology
Print ISSN: 2395-1990 | Online ISSN : 2394-4099 (www.ijsrset.com)
doi : https://doi.org/10.32628/IJSRSET218512

An Accumulator using Electron Tunneling Through Tunnel Junction

Dr. Anup Kumar Biswas
Department of Computer Science and Engineering, Kalyani Govt. Engineering College, Kalyani, Nadia, West Bengal, India

Article Info
Volume 8, Issue 5
Page Number : 163-183

Publication Issue :
September-October-2021

Article History
Accepted : 10 Sep 2021
Published: 19 Sep 2021

ABSTRACT

Instead of an existing logical Technology, by using an emerging technology we will be able to make an electronic circuit with high speed, low cost, high concentration density, light in weight, reduced gate numbers and low power consumption. This technology is based on the linear threshold logic condition and electron-tunneling event. At the time of implementing a circuit, a multi-inputs but one-output based logic-node will be brought in our consideration. In this work, we have designed a 1-bit accumulator and then implemented it. To develop an accumulator, some small components like 2-input AND, 3-input AND, 3-input OR, 8-input OR, 9-input OR gate and above all a JK Flip-flop (for 1-bit) are to be collected and connected them in logical order to obtain the proper circuit. After verifying all their characteristics with the results obtained from the simulator, we have built a 1-bit accumulator. All the small components are provided in due places. They are analyzed, detected their threshold logic equations, shown their threshold logic gates (TLGs), tabulated their truth tables, drawn their input-output waveforms, given their respective circuits with exact parameter values. In the accumulator, there are nine control variables S1 through S9 in view of performing the operations (i) Addition, (ii) clear, (iii) complement, (iv) AND, (v) OR, (vi) XOR, (vii) Right-shift, (viii) Left-shift and (ix) increment with positive triggering clock pulses. Whether our present work's circuits are faster or slower with respect to the similar circuits of CMOS based- and Single electron transistor (SET) based circuits are compared and observed that our TLG based circuits are faster than the CMOS and SET based circuits. The power consumed for tunneling event for a circuit is measured and sensed that it would remain in the range of 10meV to 250meV which is low. All the circuits we have presented in this work are of 'generic multiple input threshold logic gate' which is elaborately discussed.

Keywords : Electron-tunneling, Threshold logic, *JK* Flip-flop, Accumulator, high-speed.

I. INTRODUCTION

TERMINPLOGIES

A. *Threshold Logic Gate*

A **threshold logic gate** (TLG) is capable of implementing a Boolean function in a different way away from the traditional one, as follows. Every input variable has its determined weight, and the gate must have a threshold/intrinsic value. When the sum of the weights with their ON inputs is equal to or greater than the threshold value then the output is '1'. Otherwise, the output is '0'. This operating principle can be written as below.

$$f = \begin{cases} 1, & \sum_{i=1}^{n} w_i x_i \geq \theta \\ 0, & otherwise \end{cases}$$

where x_i represents input, w_i represents the weight of each input, and θ is the threshold value.

Threshold logic function (TLF) which represents a Boolean function implemented by a single TLG. The TLF is called linearly separable[1-7,14]. A TLF is realized by a vector $[w_1,w_2,\ldots,w_n; \theta]$. A TLF may be a more complex function, for example, $f = x_1 x_2 + x_1 x_3 + x_2 x_3 x_4 + x_2 x_3 x_5$ which corresponds to the TLG $f = [4,3,3,1,1; 7]$ and this five variables logical expression can be represented by a single node TLG as below.

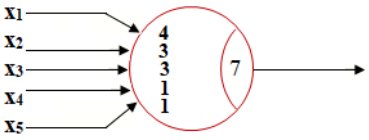


Fig.1 TLG of $f = [4,3,3,1,1; 7]$

The important advantage of a threshold logic is that number of gates can be reduced to a minimum number of nodes (gates) used in the circuit, decreasing area or element numbers.

Threshold Logic Properties

A TLF will be unate function, but the reverse is not always true, that is, all the unate functions are not TLF [1]. Hence, if a function bears binate variables, the function cannot be TLF. The function $f = x1x2 + x3x4$ is an example of an unate function but not of a TLF. When the logic function has complemented variables, these variables can be manipulated in the same way as the functions having all non-complemented variables, to identify if it is a TLF. For instance, If $f(x_1, x_2,\ldots, x_n)$ is a threshold function defined by $[w_1,w_2,\ldots,w_n; \theta]$ then its complement $\bar{f}(x_1,x_2,\ldots, x_n)$ is also a threshold function defined by $[-w_1,-w_2,\ldots,-w_n;1-\theta]$. When a function is a TLF, then by selectively complementing all the inputs, we can obtain a realization by a TLG with only positive coupling weights.

B. Majority Decision Functions (MDF)

This operating principle can also be expressed in another way as Majority Decision Functions.

A "majority decision function" of n variables for a logical element having n-inputs, $x_1, x_2, \ldots, x_n$ and one output.

The output value of the element is

$$\begin{cases} 1, & for \sum_{i=1}^{n} w_i x_i \geq \theta \\ 0, & for \sum_{i=1}^{n} w_i x_i \leq \theta - 1 \end{cases}$$

where x_i represents input, w_i represents the coupling weight of each input x_i, and θ is the threshold value.

For instance, the structure (2, 1, 1; 2) can be represented by a MDF as $x_1 + x_2x_3$; so, this function is a MDF. Whereas, the function $f(x_1x_2x_3x_4) = x_1x_2 + x_3x_4$ is not a MDF because of that it cannot be cognized by any single majority decision element.

II. DESIGN OF AN ACCUMULATOR

An accumulator consists of n-flip-flops and n-stages. Every stage consists of a flip-flop, one data input terminal B_i, and a combinational logic along with the flip-flop.

For the designing purpose, we will consider only typical stage i, whereas the n-bit accumulator has n stages for i =1, 2, 3,…, n. Every stage A_i is connected with the adjacent stage A_{i-1} on its right and stage A_{i+1} staying on its left. The last stage (leftmost stage) and first stage have no adjacent neighbors on the left and right side respectively. In this work, the register is considered to be made up of JK flop-flops.

There are nine control/selection lines with selection variables S_i, i =1, 2…, 9 initiates the micro-operation. We have to ensure that a single selection variable is enabled at a particular moment. As the selection variables are mutually exclusive, we can separate the combinational circuit regarding a stage into smaller components or circuits, one for each micro-operation. The list of micro-operation is shown in Table-1.

Table-1. Micro-operation of an accumulator

Selection variable	Name of operation	μ-operation
S_1	Addition	$A \leftarrow A + B$
S_2	Clear	A←0
S_3	Complement	A←$\bar{A}$
S_4	AND	A←A∧B
S_5	OR	A←A∨B
S_6	XOR	A←A⊕B
S_7	Shift-right	A←shr$A$
S_8	Shift-left	A←shl$A$
S_9	Increment	A←A+1
	Check for zero	If A=0 then Z=1

Now, all the operations marked in the Table-1 are discussed in the sub-sections for determining the logical expressions.

2A. Selection variable S_1 for Addition B to A:

If the selection variable S_1 is 1 then the micro-operation for addition is initiated. In the present state A_i of each JK flip-flop of each stage i is taken as the first input for a full adder, B_i is the data input (2nd input) and the previous carry input C_i. The output of full adder is sent to the flip-flop A_i and the carry output C_{i+1} in passed to the carry input of the next stage. The input-output relationship for the case of a full adder of a stage is given in Table-2 below.

Table-2

Present state	Inputs		Next state	JK Flip-flop inputs		Output
A_i	B_i	C_i	A_i	JA_i	KA_i	C_{i+1}
0	0	0	0	0	X	0
0	0	1	1	1	X	0
0	1	0	1	1	X	0
0	1	1	0	0	X	1
1	0	0	1	X	0	0
1	0	1	0	X	1	1
1	1	0	0	X	1	1
1	1	1	1	X	0	1

From this table, we can easily find out the Boolean functions for the Addition micro-operation. We have designated the J input and K input of flip-flop A_i by JA_i and KA_i respectively. Noted that the equations of JA_i and KA_i must affect the flip-flop when S_1 is enabled, so they are to be ANDed with S_1. The micro-operation (Addition) associated with the combinational circuit can be written by the following Boolean functions.

$JA_i = B_i\bar{C}_iS_1 + \bar{B}_iC_iS_1$

$KA_i = B_i\bar{C}_iS_1 + \bar{B}_iC_iS_1$

$C_{i+1}= A_iB_i + B_iC_i + C_iA_i$

The first two functions are identical, which indicates a condition of complementing A_i. The third equation provides the carry for the next higher stage.

2B. Selection variable S_2 for Clear:

Selection variable S_2 makes all the flip-flops in register A clear. To happen this we must apply S_2 =1 to K input of JK flip-flop. J input value will be kept as 0. So the equations for micro-operation in the case of clear will be as.

$JA_i = 0$
$KA_i= S_2$

2C. Selection variable S_3 for Complement:

We know when both of the inputs of the JK flip-flop are 1s, the output would be the complement of the previous one. So. For the purpose of complement we apply $\boldsymbol{S_3}$ as below.

$JA_i = S_3$
$KA_i= S_3$

2D. Selection variable S_4 for AND Micro-operation:

AND operation is initiated when the variable S_4 is applied. This logic operation happens over A_i and B_i and the result is transferred to A_i. The execution table is shown in Table-3 for this AND micro-operation.

Table-3

Present state	input	Next state	Flip-flop inputs	
A_i	B_i	A_i	JA_i	KA_i
0	0	0	0	X
0	1	0	0	X
1	0	0	X	1
1	1	1	X	0

From this table we will be able to get two expressions as.

$JA_i = 0$
$KA_i= \bar{B}_i$

In these equations the variable S_4 is not included. While including this S_4, we must modify the above equations as

$JA_i = 0$
$KA_i= \bar{B}_i S_4$

2E. Selection variable S_5 for OR Microoperation:

An OR operation is initiated when the variable S_5 is applied. This logic operation happens over A_i and B_i and the result is transferred to A_i. The execution table is shown in Table-4 for this OR micro-operation.

Table-4

Present state	input	Next state	Flip-flop inputs	
A_i	B_i	A_i	JA_i	KA_i
0	0	0	0	X
0	1	1	1	X
1	0	1	X	0
1	1	1	X	0

From this table we will be able to get two expressions as.

$JA_i = B_i$
$KA_i= 0$

In these equations the variable S_5 is not included. While including this S_5, we must modify the above equations as

$JA_i = B_i S_5$
$KA_i= 0$

2F. Selection variable S_6 for X-OR Microoperation:

An OR operation is initiated when the variable S_5 is applied. This logic operation happens over A_i and B_i

and the result is transferred to A_i. The execution table is shown in Table-5 for this OR micro-operation.

Table-5

Present state	input	Next state	Flip-flop inputs	
A_i	B_i	A_i	JA_i	KA_i
0	0	0	0	X
0	1	1	1	X
1	0	1	X	0
1	1	0	X	1

From this table we will be able to get two expressions by using Karnaugh map as

$JA_i = \mathrm{B}_i$

$KA_i = \mathrm{B}_i$.

In this Left shift operation, the contents of the register A is shifted one position towards left. That is the value of the flip-flop A_{i-1} is transferred to the

$JA_i = \mathrm{A}_{i-1}\, S_8$

$KA_i = \bar{\mathrm{A}}_{i-1}\, S_8$

2H. Selection variable S_9 for increment:

In this operation, the content of register A is increased by1. This operation is similar to a binary counter shown in Fig. 2. For the case of JK flip-flop when both the inputs are 1, the output is the complement of the previous state and when both the inputs are 0 the outputs of the flip-flops remain the same with the positive clock pulses. From the Fig. 2, it is seen that each stage will be complemented when the corresponding input carry E_i =1. Here every stage creates an output carry E_{i+1} toward the next left stage. The first stage is complemented when $S_9 = 1$. The logical expression for a typical stage is expressed as below.

$JA_i = E_i$ $\quad KA_i = E_i$

$E_{i+1} = A_i E_i$ $\quad E_1 = S_9$

In these equations the variable S_6 is not included. While including this S_6, we must modify the above equations as

$JA_i = \mathrm{B}_i\, S_6$

$KA_i = \mathrm{B}_i\, S_6$

2G. Selection variable S_7 for Right shift Microoperation:

In this right shift operation, the contents of the register A is shifted one position towards right. That is the value of the flip-flop A_{i+1} is transferred to the flip-flop A_i. So, this transfer operation can be expressed as

$JA_i = \mathrm{A}_{i+1}\, S_7$

$KA_i = \bar{\mathrm{A}}_{i+1}\, S_7$

2J. Selection variable S_8 for Left shift Microoperation:

flip-flop A_i . So, this transfer operation can be expressed as

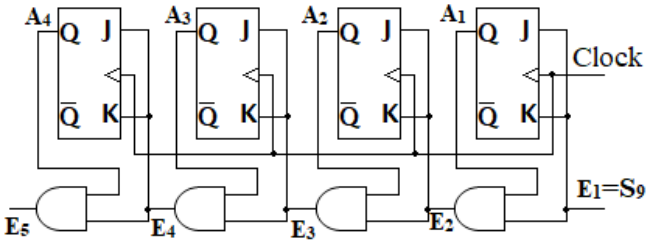


Fig.2 A 4-bit binary counter

2I. Check for Zero (Z) for the Accumulator

Z is an output variable emerging from the accumulator. It is used to indicate whether the content of the register A is zero or not. The value of Z =1 whenever all the JK flip-flops in the accumulator are cleared. If a flip-flop is cleared, the output of its complement ($\bar{Q}$) equals to 1. From the Fig. 3, every stage creates a variable Z_{i+1} while ANDing an input variable Z_i and complement output of A_i. If all the values of Z_i are 1, then we can say the accumulator is in state Zero. Hence we can express the logical equation for a particular stage as below. Z=1, if the output bit signal from the final stage equals to 1.

$Z_{i+1} = Z_i \bar{\mathrm{A}}_i$ $\quad$ i=1, 2, … n

Z_1=1

Z_{n+1}= Z

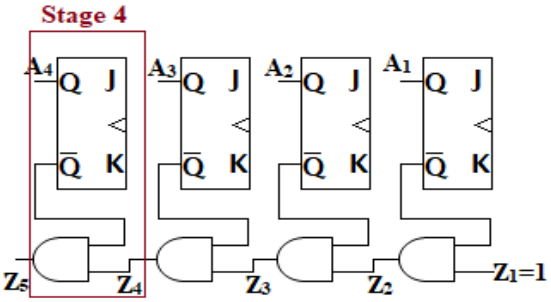


Fig.3 checking the zero content of a register A

2K. Single Stage of Accumulator

Now we can make a typical stage of an accumulator with the help of the expressions of micro-operations derived above. As the variables from S_1 to S_9 are mutually exclusive i.e., they are not activated at the same time, so the respective circuits are joined/connected with an OR operation. By joining all the input functions for the sake of J and K inputs of the flip-flops A_i, we can create a composite set of input logical expression for a particular stage.

$JA_i = B_i\bar{C}_iS_1 + \bar{B}_iC_iS_1 + S_3 + B_iS_5 + B_iS_6 + A_{i+1}S_7 + A_{i-1}S_8 + E_i$

$KA_i = B_i\bar{C}_iS_1 + \bar{B}_iC_iS_1 + S_2 + S_3 + \bar{B}_iS_4 + B_iS_6 + \bar{A}_{i+1}S_7 + \bar{A}_{i-1}S_8 + E_i$

For the case of the accumulator, each stage, of course, produces the carries for the stage towards left. These carries are as follows.

$C_{i+1} = A_iB_i + B_iC_i + C_iA_i$

$E_{i+1} = A_iE_i$

$Z_{i+1} = Z_i\bar{A}_i$

Based on these logical expressions, a logic diagram for a typical stage as to the accumulator is drawn in Fig.4. This is basically the direct implementation of the logical expressions listed above.

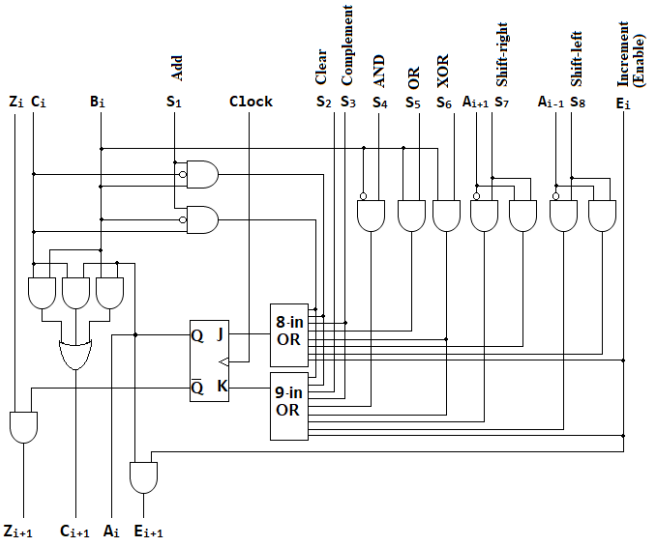


Fig. 4 Logic diagram of a single stage Accumulator

1. Generic Multiple input threshold logic gate

A multiple threshold logic gate [1-7 ,12-14] which may also be called a generic Multiple input threshold logic gate is shown in Fig. 5.

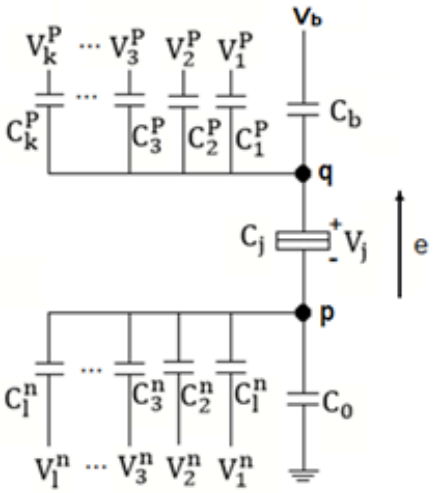


Fig.5 TLG with multi-input

It consists of a tunnel junction having internal capacitance C_j and resistance R_j, two multi-input-signals (i) V_k^Ps and (ii) V_l^ns connected at two points 'q' and 'p' respectively. Each input voltage V_k^P (upper side) is connected to the point "q" through the capacitance C_k^P ; and each input *voltage* V_l^n (lower side), is joined at the point "p" through the *capacitance* C_l^n. The bias *voltage* V_b , for the

assistance of the tunneling, is connected to the point "q" through the true capacitor C_b. Tunnel Junction capacitor C_j is linked between "p" and "q". Point "p" is grounded through C0. C_j and C0 play the important role in the circuit. For a LTG, the operation of it can be expressed by a function called *signun function*. *The signun function* of h(x) expressed by two equations (1a) and (1b).

$$g(x) = \text{sgn}\{h(x)\} = \begin{cases} 0, & if\ h(x) < 0 \\ 1, & if\ h(x) \geq 0 \end{cases} \quad \cdots\cdots\cdots (1a)$$

$$h(x) = \sum_{k=1}^{n}(w_k \times x_k) - \theta \quad \cdots\cdots\cdots (1b)$$

where x_k represents n-Boolean inputs, and w_k are their corresponding integer weights.

The LTG compares the threshold value θ with the weighted sum of the *inputs* $\sum_{k=1}^{n}(w_k \times x_k)$. When $\sum_{k=1}^{n}(w_k \times x_k)$ is more than or equal to the threshold value θ of the circuit, then output of the LTG shows high (logical "1"), otherwise low (logical "0").

III. THRESHOLD LOGIC GATES BASED CIRCUITS

A typical stage of an accumulator has been designed and presented in a pictorial view by using the direct implementation of the logical expressions discussed. In is transparent for us that the small elements required for implementing the typical stage are (i) 2-input AND gate, (ii) 3-input AND gate (iii) 3-input OR gate, (iv) 8-input OR gate, (v) 9-input OR gate and (vi) JK flip-flop with clock. Now, we will present all these components based on linear threshold gates (LTGs) and they will be verified by means of a software called SIMON.

4A. 2-input AND gate (linear threshold logic)

By using the Generic Multiple input threshold logic gate, a 2-input AND gate with positive logic will be obtained easily. Consider the threshold logic expression of 2-input AND gate to be

$$AND(A,B) = sgn\{w_A A + w_B B - \theta\} \ldots\ldots (2)$$

To obtain a 2-input threshold logic AND gate, we can draw the truth Table-6 of a 2-input AND gate and compare the coupling weights w_A and w_B of two variables A and B respectively with the threshold value θ [1-5,7,8].

Table-6

A	B	F(A,B)	θ	inequality
0	0	0	$0<\theta$	(1)
0	1	0	$w_B < \theta$	(2)
1	0	0	$w_A < \theta$	(3)
1	1	1	$w_B + w_A \geq \theta$	(4)

As 2-input AND gate is a positive logic, we can assume the weights of A and B are 1 each. Then from the four inequalities in the Table-6, if we take w_B=1, w_A=1 and θ=2, then the four inequalities in 4th column in Table-6 are satisfied. Hence the linear threshold logic equation for the cited AND gate becomes

$$AND(A,B) = sgn\{A + B - 2\} \cdots\cdots\cdots (3)$$

Based upon the equation (5), we can place two input capacitancesC_1^p *and* C_2^p with two positive logic inputs in the Generic Multiple input threshold logic gate. For this purpose the parameter values we have measured are as follows: logic input "0"=0V, logic "1" = 16mV, C=1aF, $C_1^p = C_2^p = \frac{1}{2}C$ =0.5aF, $C_b = 9aF$, $C_j = 0.25aF$, $C_L = 9aF$, $C_0 = 10aF$, $R_j = 10^5$ Ω, . V_b $= V_s$ =16mV. The corresponding 2-input threshold logic gate is given in Fig. 6(a) with its simulated results in Fig. 6(b).

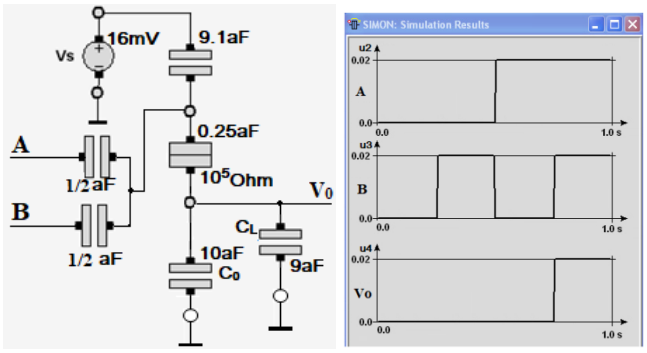


Fig. 6(a) 2-input AND Gate (b) Simulation result

4B. 2-input AND gate with one complement input

We are to explore the construction of the Fig. 6(c) and find out the threshold logic symbol for the same. If we think of the similarity of logic equations AND(A,B)=AB and AND(A,$\bar{B}$)=A$\bar{B}$, a threshold logic equation of AND(A,$\bar{B}$)=A$\bar{B}$ can be written as

AND(A,$\bar{B}$)= $sgn\{A + \bar{B} - 2\}$ ..... (4)

But we know, $\bar{B}$+B=1 or $\bar{B}$=−B+1,

Putting this value into equation (4), we obtain equation (5)

F(A,B)= AND(A,$\bar{B}$)= $sgn\{A - B - 1\}$ ..... (5)

Based upon the equation (5), we can place two input capacitances C_1^p *and* C_1^n with one positive logic input and one negative input in the Generic Multiple input threshold logic gate. For this purpose the parameter values we have measured are as follows: logic input "0"=0V, logic "1" = 16mV, C=1aF, $C_1^p = C_1^n = \frac{1}{2}C$ =0.5aF, $C_b = 9.5aF$, $C_j = 0.25aF$, $C_L = 9aF$, $C_0 = 9.5aF$, $R_j = 10^5$ Ω, . $V_b = V_s = 16$mV. The corresponding 2-input threshold logic gate is given in Fig. 6(e) with its simulated results in Fig. 6(f).

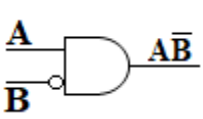
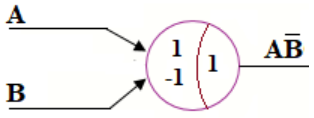


Fig.6(c) symbol of A$\bar{B}$



6(d) TLG based node of A$\bar{B}$

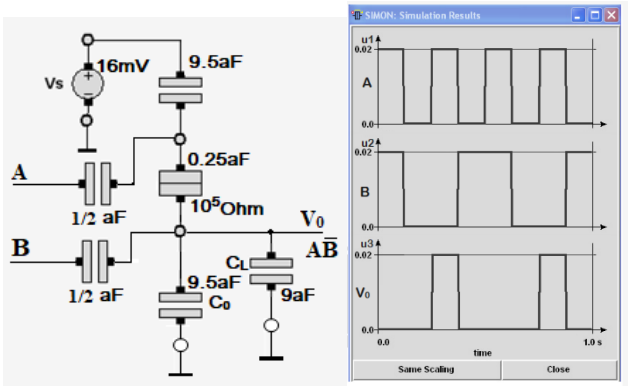


Fig. 6(e) 2-input TLG of A$\bar{B}$ Fig. 6(f) simulated result of Fig 6(e)

4 C. 3-input AND gate (linear threshold logic)

By using the Generic Multiple input threshold logic gate, a 3-input AND gate with positive logic will be obtained easily. Consider the threshold logic expression of 3-input AND gate to be

$AND(A,B,C) = sgn\{w_A A + w_B B + w_C C - \theta\}$..... (6)

To obtain a 3-input threshold logic AND gate, we can draw the truth Table-7 of a 3-input AND gate and compare the coupling weights w_A , w_B and w_C of three variables A, B and C respectively with the threshold value θ [3-7,13-15].

Table-7

A	B	C	F= ABC	θ	Eqn. No.
0	0	0	0	$0<\theta$	(1)
0	0	1	0	$w_C < \theta$	(2)
0	1	0	0	$w_B < \theta$	(3)
0	1	1	0	$w_B + w_C < \theta$	(4)
1	0	0	0	$w_A < \theta$	(5)
1	0	1	0	$w_A + w_C < \theta$	(6)
1	1	0	0	$w_A + w_B < \theta$	(7)
1	1	1	1	$w_A + w_B + w_C \geq \theta$	(8)

As 3-input AND gate is a positive logic, we can assume the weights of A, B and C are 1 each. Then from the eight inequalities in the Table-7, if we take w_B=1, w_A=1, w_C=1 and θ=3, then the eight inequalities in 5th column in Table-7 are satisfied. Hence the linear threshold logic equation for the 3-input AND gate becomes Equation (7). The symbolic representation of

linear threshold logic for the 3-input AND gate is given in Fig. 7(a)

$$AND(A,B,C) = sgn\{A+B+C-3\} \quad \text{.....(7)}$$

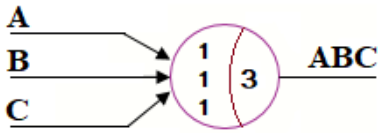


Fig. 7(a) symbolic representation 3-input AND

Based upon the equation (7), we can place three input capacitances C_1^p , C_2^p *and* C_3^P with three positive logic inputs in the Generic Multiple input threshold logic gate. For this purpose the parameter values we have measured are as follows logic input "0"=0V, logic "1" = 16mV, C=1aF, $C_1^p = C_2^P = C_3^P = \frac{1}{2}C$ =0.5aF, $C_b = 8.5aF$, $C_j = 0.25aF$, $C_L = 9aF$, $C_0 = 10aF$, R_j = 10^5 Ω, . V_b = V_s =16mV. The corresponding 3-input threshold logic gate is given in Fig. 6(b) with its simulated results in Fig. 6(c).

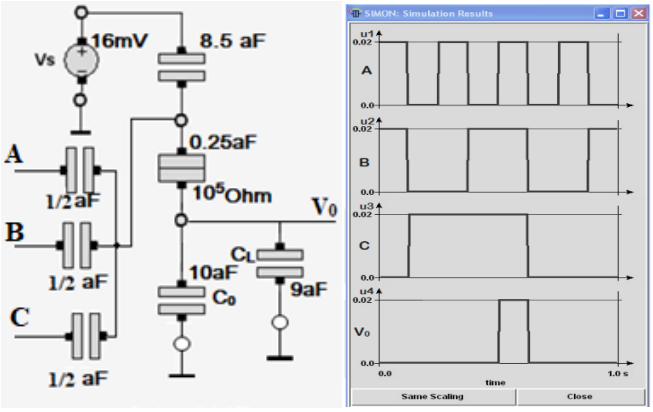


Fig. 6(b) 3-input TLG of ABC | Fig.6(c) simulated result

4D. 3-input AND gate with one complement input

When making an Accumulator, we require an expression $A\bar{B}C$ to be implemented by a three input gate as shown in Fig. 7(a). Keeping the similarity of equation (7), we can write the threshold logic equation (TLE) for the logic expression $A\bar{B}C$ as in equation (8).

AND(A,$\bar{B}$,C)= $sgn\{A+\bar{B}+C-3\}$ .....(8)

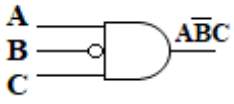
But we know, $\bar{B}$+B=1 or $\bar{B}$=−B+1,

Putting this value into equation (8), we obtain equation (9)

F(A,B,C)= AND(A,$\bar{B}$,C)

$= sgn\{A-B+C-2\}$ .....(9)

Based upon the equation (9), we can place two input capacitances C_1^p , C_2^p *and* C_1^n with two positive logic input and one negative input in the Generic Multiple input threshold logic gate. For this purpose the parameter values we have measured are as follows: logic input "0"=0V, logic "1" = 16mV, C=1aF, $C_1^p = C_2^p = C_1^n = \frac{1}{2}C$ =0.5aF, $C_b = 9aF$, $C_j = 0.25aF$, $C_L = 9aF$, $C_0 = 9.5aF$, R_j = 10^5 Ω, . V_b = V_s =16mV. *The* corresponding 3-input threshold logic gate is given in Fig. 7(b) with its simulated results in Fig. 7(c).



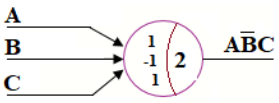


Fig. 7(a) logic gate of $A\bar{B}C$ 7(b) Symbol of TLG of

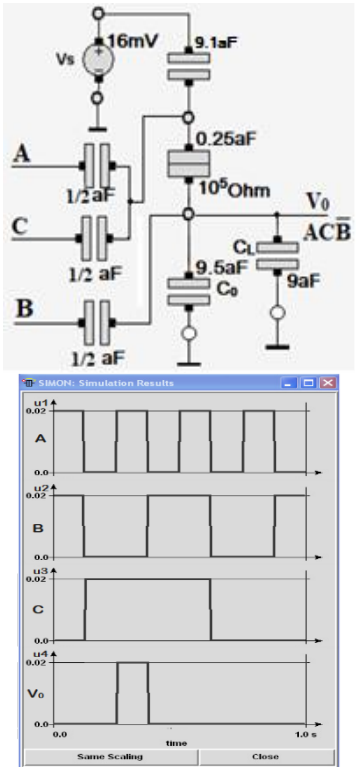
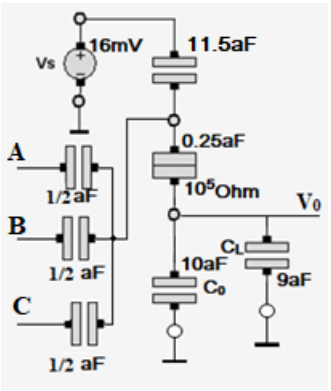


Fig. 7(c) 3-input TLG of $A\bar{B}C$ | Fig.6(c) simulated result

4E. 3-input OR gate

We can find out the threshold logic gate of 3-input OR gate as the procedure discussed above. The Threshold logic equation is of 3-input OR is given in equation (10).

$$OR(A,B,C) = sgn\{A+B+C-1\} \cdots\cdots (10)$$



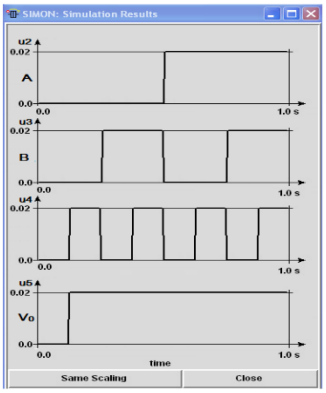


Fig. 8(a) 3-in TLG of A+B+C Fig.8(b) simulated result

4F. 8-input OR and 9-input Threshold Logic OR gate

For space limitation, we have not elaborately discussed regarding the process to find out the threshold logic equations of 8-input and 9-input OR gates. The threshold logic(TL) equations of them are given below.

$$OR(X_1X_2X_3X_4X_5X_6X_7X_8) = sgn\{\bigvee_{i=1}^{8} X_i - 1\} \cdots\cdots (11)$$

$$OR(X_1X_2X_3X_4X_5X_6X_7X_8X_9) = sgn\{\bigvee_{i=1}^{9} X_i - 1\} \cdots\cdots (12)$$

The corresponding TL 8-input OR and 9-input OR gates are given in Fig. 9(a) and Fig. 9(b) respectively.

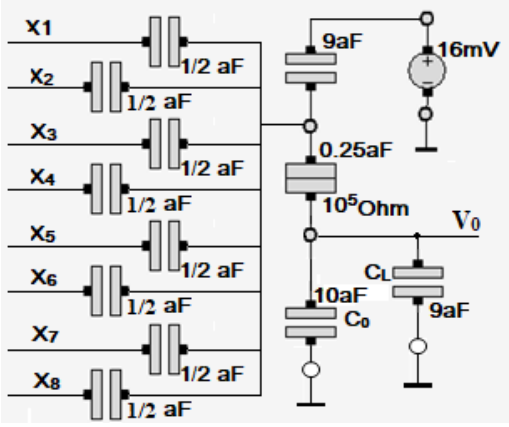


Fig. 9 (a) 8-input TL OR gate

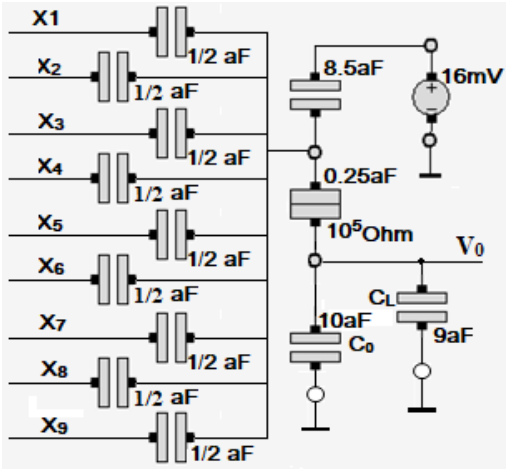


Fig. 9(b)-input TL OR gate

4G. Implementation of JK Flip-flop using Threshold Logic Gate

First we can draw the truth table for a JK Flip-flop which is given in Table-8.

Table-8

Q^t	J	K	Q^{t+1}
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

The logic function of JK Flip-flop on the basis of Boolean gate can be written as

$$Q^{t+1}=JQ^{\bar{t}}+\bar{K}Q^{t} \quad \cdots\cdots (13)$$

and $$Q^{\bar{t}+1}=\bar{J}Q^{t}+KQ^{\bar{t}} \quad \cdots\cdots (14)$$

We know that $Q^{\bar{t}+1}$=NOT (Q^{t+1}). We are familiar with the expression $Y=A.\bar{B}+\bar{A}.B$ and it is similar to the equation (13). Now for representing the Boolean function $Q^{t+1}=J.\bar{Q}^{t}+\bar{K}.Q^{t}$ using threshold logic gate, first we take $P=(J.\bar{Q}^{t})$; as it is an AND function, so using equation (5) we can write for P as.

$$P = sgn\{J- Q^{t} - (1)\} \quad \cdots\cdots (15)$$

The threshold Logic gate of equation (15) is drawn in Fig. 10. The truth table of equation (15) is given in Table-9.

Table-9. Truth table of equation (15)

J	Q^{t}	P
0	0	0
0	1	0
1	0	1
1	1	0

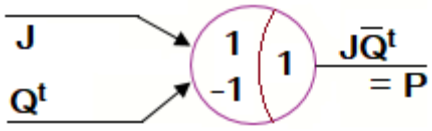


Fig. 10 threshold Logic gate of equation (15)

Now is the time for us to express the Boolean expression $Q^{t+1}= J.\bar{Q}^{t}+\bar{K}.Q^{t}$. We put $P= J.\bar{Q}^{t}$

So, $Q^{t+1}= P + \bar{K}Q^{t}$

$$\Rightarrow Q^{t+1} = P + \bar{K}Q^{t} \quad \cdots\cdots (16)$$

For having the threshold gate logic of equation (16), we draw the truth Table-8 with the help of Table-9 and from which we solve the minimum value solution to get the coupling weight values.

Table-10

CP	$\bar{K}$	P	Q^{t}	Q^{t+1}	θ
⌐	0	0	0	0	$0<\theta$
⌐	0	0	1	0	$W_{Qt} < \theta$
⌐	0	1	0	1	$W_P \geq \theta$
⌐	0	0	1	0	$W_{Qt}< \theta$
⌐	1	0	0	0	$W_{\bar{k}}<\theta$
⌐	1	0	1	1	$W_{\bar{k}} + W_{Qt} \geq \theta$
⌐	1	1	0	1	$W_{\bar{k}} + W_P \geq \theta$
⌐	1	0	1	1	$W_{\bar{k}} + W_{Qt} \geq \theta$

After solving the conditional equations in the 6th column in Table-10 we obtain a solution set which is $\{W_{\bar{K}}, W_P, W_{Qt}; \theta\} = \{1, 2, 1; 2\}$

Hence the Threshold equation for the T Flip-flop is

$Q^{t+1}(\bar{K},P,Q^{t}) = sgn\{\bar{K}+2P+Q^{t}-(2)\}$ as $\bar{K}= -K+1$, so we get

$$Q^{t+1}(K,P,Q^{t}) = sgn\{-K+2P+Q^{t}-(1)\} \quad \cdots\cdots (17)$$

As the Accumulator will require positive clock pulse to execute a particular logic, we must involve a clock in the JK flip-flop. Including the clock, the JK flip-flop of equation (17) using TLG has been given in Fig.11.

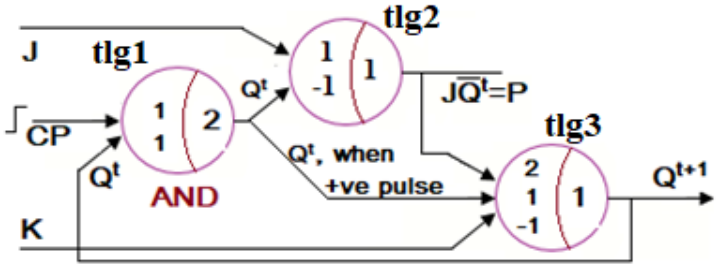


Fig. 11 Threshold logic gate based *JK* Flip-flop

For the simulation purpose of the T Flip-flop, we have chosen related parameters. For Threshold logic gate1 (tlg1): $C_0 = 9.5aF$, $R_j = 10^5\Omega$, $V_s = V_b = 16mV$, $C_1^p = 0.5aF, C_2^p = 0.5aF$, $C_b = 9aF$ $C_j = 0.25aF$ For the Threshold Logic gate2 (tlg2) : $C_1^p = 0.5aF, C_1^n = 0.5aF$, $C_b = 9.5aF$, $C_L = 9aF, C_0 = 9.5aF, C_j = 0.25aF$, $R_j = 10^5 \Omega, V_s = V_b = 16mV$. For the Threshold Logic gate2

(tlg3): $C_0 = 9.5aF$, $R_j = 10^5\Omega$, $V_s = V_b = 16mV$, $C_1^p = 0.5aF$, $C_2^p = 0.5\text{aF}$, $C_1^n = 0.5\text{aF}$, $C_b = 19aF$ $C_j = 0.25aF$

3H. logical expression of AB+BC+CA

The logic diagram given in Fig. 12(a) indicates a logic diagram of logic expression AB+BC+CA. This figure has four gates. If we replace the diagram with the help of the TLGs, only one TLG node (gate) is sufficient. Reduction property of TLG is applicable here. We can draw the truth Table-11 with the threshold inequalities.

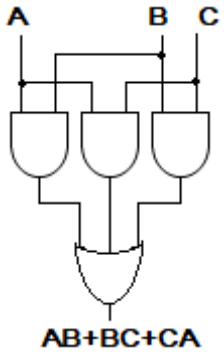


Fig. 12(a) Logical circuit of AB+BC+CA

Table-11

A	B	C	AB+BC+CA	θ	Inequality
0	0	0	0	$0<\theta$	
0	0	1	0	$w_C < \theta$	
0	1	0	0	$w_B < \theta$	
0	1	1	1	$w_B + w_C \geq \theta$	
1	0	0	0	$w_A < \theta$	
1	0	1	1	$w_A + w_C \geq \theta$	
1	1	0	1	$w_A + w_B \geq \theta$	
1	1	1	1	$w_A + w_B + w_C \geq \theta$	

After solving the eight inequalities in the Table-11, we get a solution set $\{w_A, w_B, w_C; \theta\} = \{1,1,1;\ 2\}$. So the threshold logic equation for AB+BC+CA is given in equation (18).

$F(A,B,C) = sgn\{A+B+C-2\}$ ·····(18)

According to the classical logic, to implement the Boolean logic AB+BC+CA we require four gates three AND gates and one OR gate. On the other hand, only one gate is sufficient to implement this logic expression. The symbol of the threshold logic gate representing the expression AB+BC+CA is given in Fig. 12(b) and the TLG based circuit is shown in Fig. 12(c).

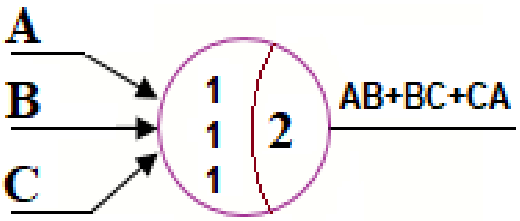


Fig. 12(b) Symbol of AB+BC+CA

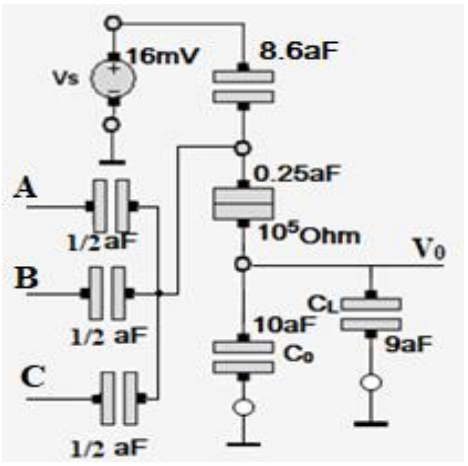


Fig. 12(c) TLG circuit of AB+BC+CA

2. Single stage Accumulator

All the TLG based small circuits are drawn above for the purpose of constructing a single stage accumulator in Fig.13 and by using n- number of such particular stage we have been able to draw a complete accumulator as shown in Fig. 15. A threshold logic gate based (TLG) single stage accumulator is depicted in Fig. 13. For the verification of the circuit, the simulation of the single stage accumulator is

performed and observed that the simulation results for different control signal conditions are matched with the theoretical aspects. Table-12 is provided for different input-output relationships. The simulation results for the typical stage in Fig. 13 is given in Fig. 14.

Table-12

	Input A_i B_i C_i	Clock pulse	Next A_i	result
S_1=1	0 1 1	⌐	0(sum)	1 (carry)
S_2=1	0	⌐	0	clear
S_3=1	0/1	⌐	1/0	complement
S_4=1	0 0 0 1 1 0 1 1	⌐	0 0 0 1	AND
S_5=1	0 0 0 1 1 0	⌐	0 1 1	OR
	1 1		1	
S_6=1	0 0 0 1 1 0 1 1	⌐	0 1 1 0	XOR
S_7=1	A_{i+1}=1 A_i=0/1	⌐	1	Right-shift
S_8=1	A_{i-1}=1 A_i=0/1	⌐	1	Left-shift
S_9=1	Assume $A_4A_3A_2A_1$ =1001	⌐	$A_4A_3A_2A_1$ =1010	increment

For different small components, distinct parameters are essential. For the understanding, at a glance, the parameter values are listed in Table-13.

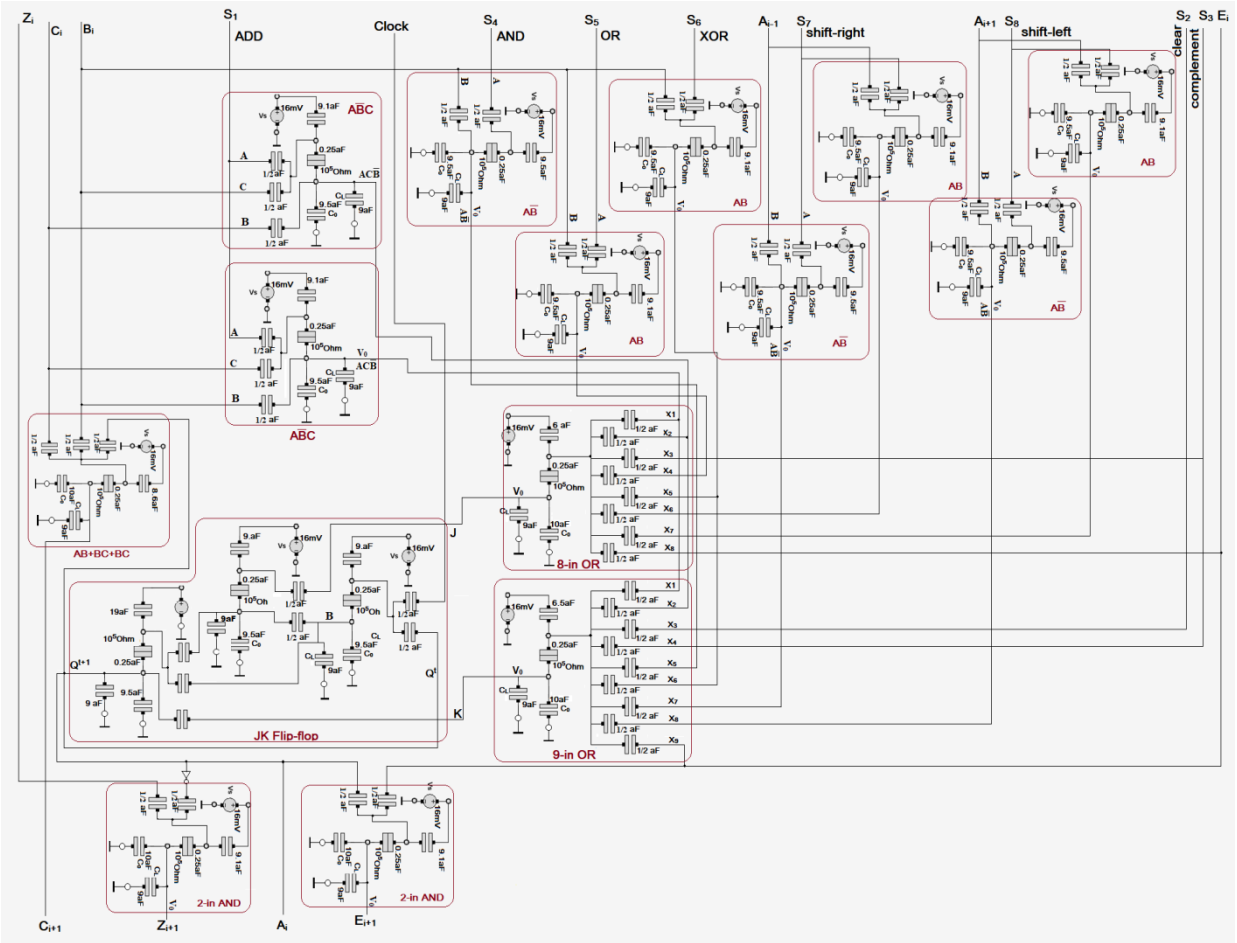


Fig. 13 A single stage of the accumulator

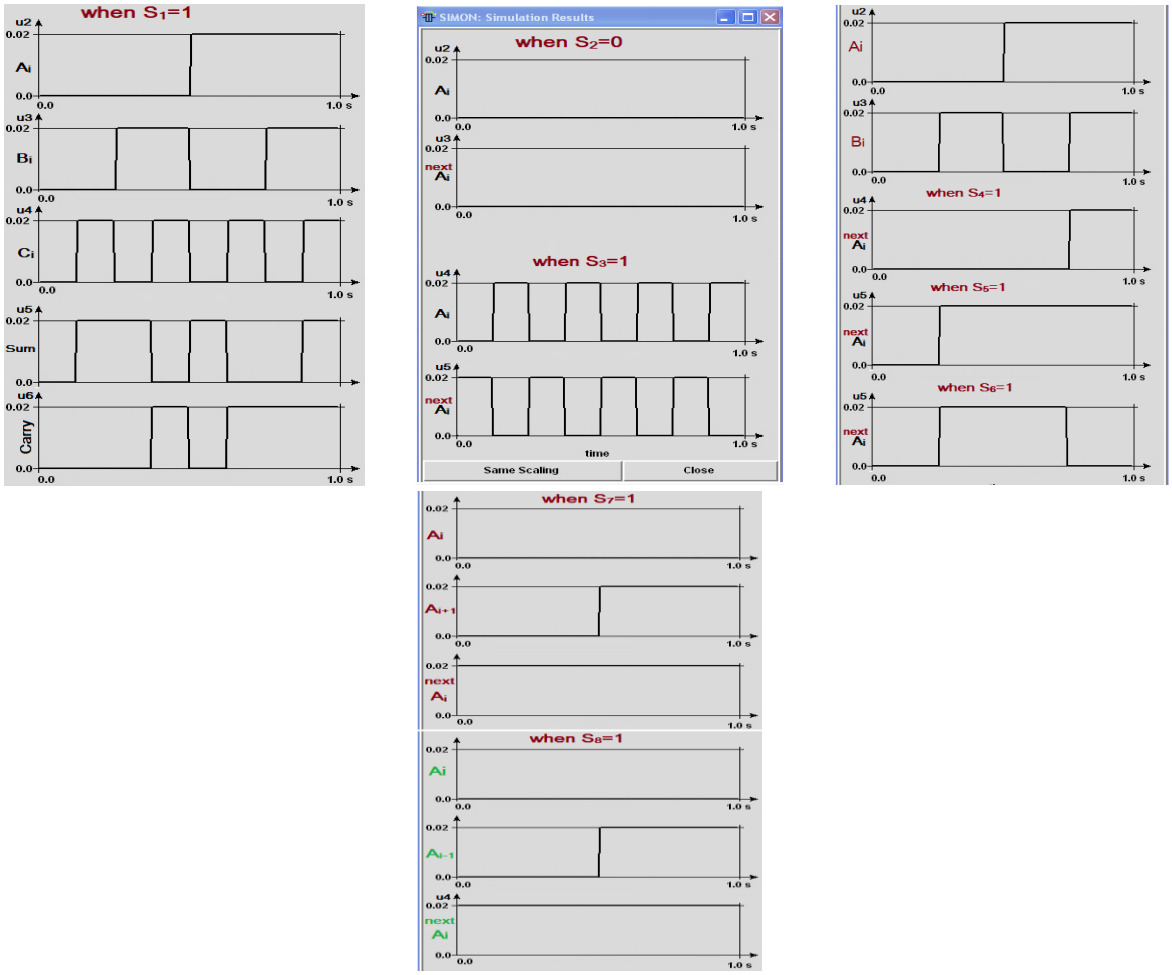
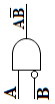
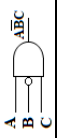


Fig. 14 Simulation result of a single state Accumulator

Table-13

Sl. No.	Expressions used & Name of Expression	parameters
1	$Q^{t+1}(K,P,Q^t) = sgn\{-K+2P+Q-(1)\}$ *JK* Flip-flop	For the simulation purpose of the T Flip-flop, we have chosen related parameters. For Threshold logic gate1 **(tlg1)**: $C_0 = 9.5aF$, $R_j = 10^5\Omega$, $V_s = V_b = 16mV$, $C_1^p = 0.5aF$, $C_2^p = 0.5\text{aF}$, $C_b = 9aF$ $C_j = 0.25aF$ For the Threshold Logic gate2 **(tlg2)** : $C_1^p = 0.5aF$, $C_1^n = 0.5\text{aF}$, $C_b = 9.5aF$, $C_L = 9aF$, $C_0 = 9.5aF$, $C_j = 0.25aF$, $R_j = 10^5\ \Omega$, $V_s = V_b = 16mV$. For the Threshold Logic gate2 **(tlg3)**: $C_0 = 9.5aF$, $R_j = 10^5\Omega$, $V_s = V_b = 16mV$, $C_1^p = 0.5aF$, $C_2^p = 0.5\text{aF}$, $C_1^n = 0.5\text{aF}$, $C_b = 19aF$ $C_j = 0.25aF$
2	$AND(A,B,C) = sgn\{A+B+C-3\}$ 3-input AND gate	logic input "0"=0V, logic "1" = 16mV, C=1aF, $C_1^p = C_2^p = C_3^p = \frac{1}{2}C$ =0.5aF, $C_b = 8.5aF$, $C_j = 0.25aF$, $C_L = 9aF$, $C_0 = 10aF$, $R_j = 10^5\ \Omega$, . $V_b = V_b = V_s = 16\text{mV}$

3	AND(A,$\bar{B}$)= $sgn\{A\text{-}B\text{-}1\}$ 	"0"=0V, logic "1" = 16mV, C=1aF, $C_1^p = C_1^n = \frac{1}{2}C$=0.5aF, $C_b = 9.5aF$, $C_j = 0.25aF$, $C_L = 9aF$, $C_0 = 9.5aF$, R_j=10^5 Ω, . $V_b = V_s$ =16mV
4	AND(A,$\bar{B}$,C)= $sgn\{A - B + C - 2\}$ 	logic input "0"=0V, logic "1" = 16mV, C=1aF, $C_1^p = C_2^p = C_1^n = \frac{1}{2}C$=0.5aF, $C_b = 9aF$, $C_j = 0.25aF$, $C_L = 9aF$, $C_0 = 9.5aF$, R_j=10^5 Ω, . $V_b = V_s$ =16mV.
5	$=sgn\{\sum_{i=1}^{8} X_i - 1\}$ 8-in OR gate	logic input "0"=0V, logic "1" = 16mV, C=1aF, $C_1^p = C_2^p = C_3^p = C_4^p = C_5^p = C_6^p = C_7^p = C_8^p$=0.5aF , $C_b = 9.1aF$, $C_j = 0.25aF$, $C_L = 9aF$, $C_0 = 9.8aF$, R_j=10^5 Ω, . $V_b = V_s$ =16mV.
6	$= sgn\{\sum_{i=1}^{9} X_i - 1\}$ 9-in OR	logic input "0"=0V, logic "1" = 16mV, C=1aF, $C_1^p = C_2^p = C_3^p = C_4^p = C_5^p = C_6^p = C_7^p = C_8^p = C_9^p = \frac{1}{2}C$=0.5aF , C_b8.5aF, $C_j = 0.25aF$, $C_L = 9aF$, $C_0 = 9.9aF$, R_j=10^5 Ω, . $V_b = V_s$ =16mV.
7	$Y \leftarrow \bar{Y}$ Inverter	For the inverter, the parameter values chosen are: V_{g1}=0, V_{g2}=0.1× $\frac{q_e}{C}$, $C_L = 9C$, $C_j = \frac{1}{10}C$, $5C_j = \frac{1}{2}C$, $C_g = \frac{1}{2}C$, $C_b = \frac{17}{4}C$, R_j =50KΩ. For the sake of simulating, the value of C =1aF is fixed.
8	AB+BC +CA	logic input "0"=0V, logic "1" = 16mV, C=1aF, $C_1^p = C_2^p = C_3^p = \frac{1}{2}C$ =0.5aF, $C_b = 8.6aF$, $C_j = 0.25aF$, $C_L = 9aF$, $C_0 = 10aF$, R_j=10^5 Ω, . $V_b = V_b$ =V_s=16mV

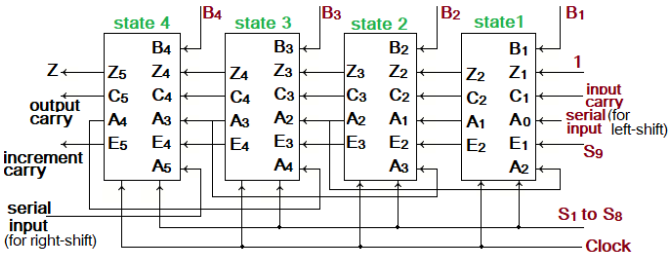


Fig. 15 A 4-bit Accumulator

An n-bit Accumulator needs n- stages which are connected in cascade, each stage bearing the circuit is shown in Fig. 13. All the control lines corresponding the variables, except S_9 have to be applied to every stage. The other inputs and outputs are contained by each stage will be connected to make a complete accumulator. Interconnections among the stages to make a desired Accumulator of 4-bit are shown in Fig. 15. In this diagram, each block corresponds to the circuit as Fig. 13. Each block takes control lines S_1 through S_8, and a clock pulse (Clock). All the other four outputs and six inputs in every block are similar to the Fig. 13 except that the subscript is changed by a number for each block. The circuit contains four B (B_1, B_2, B_3, B_4) inputs. The zero-detection chain as in Fig. 3 can be obtained by joining the Z_i variables in

cascade, where the first block takes the value 1(high). For the right-shift operation, the serial input passes through A_0, which corresponds to A_{i-1} for the stage 1.The carries are connected in cascade to full adder circuits. For the left-shift operation, the serial input passes through A_5, which corresponds to A_{i+1} for the stage 4. The carries are connected in cascade to full adder circuits. By the control variable S_9 connecting in stage 1, the operation of increment is enabled. The remaining blocks get the increment carry from the previous ones.

3. Delay, Switching energy and Fastness

Whenever a complex circuit is implemented, there must have some inevitable components which are connected in a proper way to get the circuit. In the present work, as an accumulator is considered, we require the inevitable components like 2-input AND, 3-input AND, 3-input OR, 8-input OR, 9-input OR, inverter [3-7] and JK Flip-flop with Clock pulse. For measuring the efficiency of an electronic substance/device, some factors are to be kept in brain like-time required for an execution, number of components, speed, power consumption, costs of materials, fan-outs, controlling charge, atmospheric temperature etc. Calculating the processing delays cum latency will be the first considered for this present work.

To measure the processing/switching delays, the critical voltage V_c and the tunnel junction capacitance C_j must be entangled. The processing delay for a circuit can be find out by using the expression given by the approach as written below [3-7, 14-15].

$$\text{Delay} = -(e|\ln(P_{error})|R_t) / (|V_j| - V_c) \cdots\cdots (19)$$

where, R_t is the internal resistance of the tunnel junction, V_c is the critical voltage and V_j is the tunnel junction voltage.

Execution is performed when an electron passes through the tunnel junction. When the tunnel junction voltage V_j which is a portion of the supply voltage is being greater than or equal to the internal critical voltage V_c , of the tunnel junction i.e., when $|V_j| \geq V_c$, the tunneling event occurs.

If conditional expression is satisfied by a 3-input AND gate, junction voltage of the 3-input AND gate becomes V_j =11.8mV, the internal critical voltage V_c of the tunnel junction is found to be 11.58mV. Given that the parameter values of tunnel resistance $R_t = 10^5\Omega$ and the probability of error $P_{error} = 10^{-12}$. Putting them into equation (19) we can measure the gate delay = 0.072|ln ($\boldsymbol{P_{error}}$)|ns = 1.98ns. In a similar manner we can calculate the delays of distinct circuits and they are listed in Table-14 and

Table-14

Gate/Device	n-elements	Delay (ns)	Switching Energy meV
inverter	9	0.022|ln($\boldsymbol{P_{error}}$) |	10.40
2-in NOR	6	0.060|ln($\boldsymbol{P_{error}}$) |	10.70
2-in OR	6	0.062|ln($\boldsymbol{P_{error}}$) |	10.80
2-in NAND	6	0.062|ln($\boldsymbol{P_{error}}$) |	10.80
2-in AND	6	0.062|ln($\boldsymbol{P_{error}}$) |	10.80
3-in AND	7	0.072|ln($\boldsymbol{P_{error}}$) |	11.58
3-in NAND	7	0.072|ln($\boldsymbol{P_{error}}$) |	11.58
2-in XOR	13	0.080|ln($\boldsymbol{P_{error}}$) |	21.20
3-in OR	7	0.073|ln($\boldsymbol{P_{error}}$) |	11.58
3-in NOR	7	0.072|ln($\boldsymbol{P_{error}}$) |	11.57
4-in OR	8	0.076|ln($\boldsymbol{P_{error}}$) |	14.22
8-in OR	12	0.077|ln($\boldsymbol{P_{error}}$) |	21.4
9-in OR	13	0.079|ln($\boldsymbol{P_{error}}$) |	23.18
JK flip-flop	19	0.197|ln($\boldsymbol{P_{error}}$) |	33.88
1 stage Accumulator	128	0.418|ln($\boldsymbol{P_{error}}$) |	224.7

While tunneling, an electron goes through the junction barrier, by changing the total energy in the circuit. The energy changes before and after a tunneling event can be measured by using the equation (20) given below.

$$\Delta E = E_{before\ tunnel} - E_{after\ tunnel}$$
$$= -e(V_c - |V_j|) \cdots\cdots\cdots\cdots\cdots\cdots (20)$$

This is the switching/tunneling energy $-e(V_c - |V_j|)$ consumed for a tunnel event in the tunneling circuit. We listed the energy consumptions for distinct TLG circuits in Table-14.

Two distinct curves have been traced in relation to the switching delay vs. switching error probability and the switching delay vs. the unit capacitance C (=1aF) in Fig. 16(a) and Fog. 16(b) respectively.

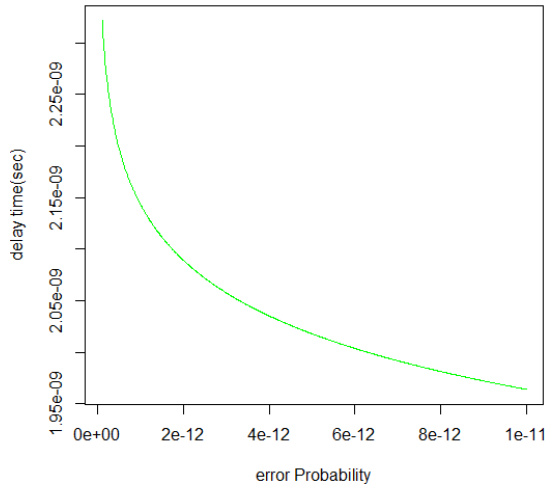


Fig. 16(a) Delay vs. Error Probability in TLGs

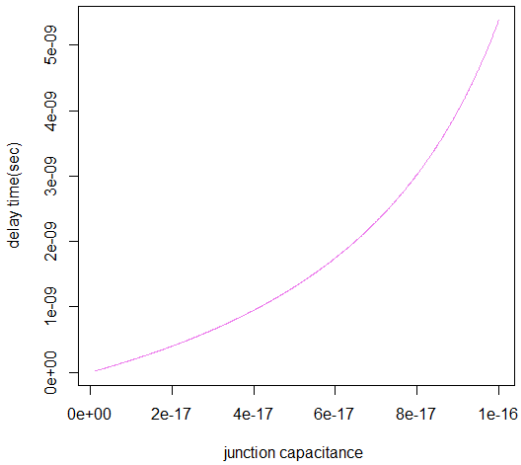


Fig. 16(b) Delay Vs. capacitance in circuits of TLGs.

How many number of fundamental elements like capacitors, resistors are needed for constructing a small elements like gate or circuit are counted. We have kept track of the data required from this work in respect of small element numbers, delays or late, and switching power etc. and all of them are tabulated in Table-15. Power consumption (meV) for each circuit vs. elements is shown in Fig. 17 below. The processing delays for single electron Transistor (SET) and TLG based circuits in reference the various elements are projected by using the bar diagram side by side in Fig. 18.

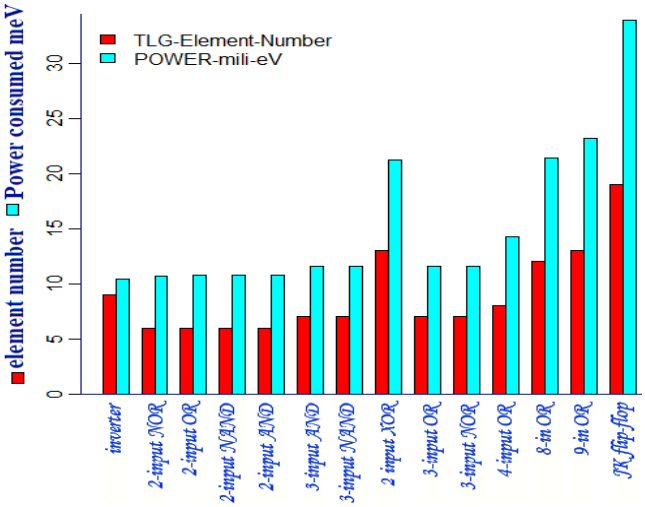


Fig. 17 Comparison of small Elements and switching energy

Table-15

Gate/Device	SET-based delay ns	LTG-based delay ns	Speed-up times
inverter	8	0.60	13.33
2-input NOR	4	1.65	2.42
2-input OR	4	1.71	2.33
2-input NAND	4	1.71	2.33
2-input AND	4	1.71	2.33
3-input AND	8	1.98	4.04
3-input NAND	8	1.98	4.04
2-input XOR	4	2.21	1.81
3-input OR	8	2.01	3.98
3-input NOR	8	1.99	4.02
4-input OR	8	2.15	3.72
8-in OR	28	2.12	13.2
9-in OR	32	2.18	14.67
JK flip-flop	16	5.44	2.94
1 stage Accumulator	44	11.54	3.81

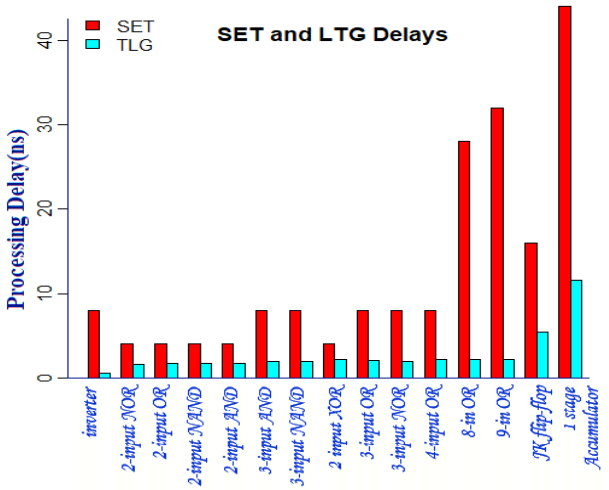


Fig. 18 Delays of SET/TLG Vs. gates

The execution delays for distinct TLG based circuits must not be similar, for instance, in a 3-input OR gate the latency becomes 2.01 ns, for the 3-input NAND gate the processing delay is 1.98 ns, and in a single stage Accumulator delay is 11.54ns.

Considering that the value of P_{error} is 10^{-12}, the time after which the first output of the 4-bit Accumulator will fan out is 11.54×4= 46.16ns. Hence, we surely maintain the clock pulse time i.e., after every 46.16ns ns or more in the 4-bit Accumulator circuit. In this condition, frequency of fan-out for the Accumulator circuit will be $1/46.16\text{ns} = 21.66$ MHz.

We are intended to bring to light the circuit delays in connection with CMOS, SET-based and LTG-based. We have detected the processing delays as 12ns [5-11, 17-18] each for the case of a CMOS logic gate like AND, NAND, NOR and XOR. The tunneling time delay in the case of a single electron transistor (SET) [13-15] is approximately 4ns [4-10, 17-18]. The speed-up of LTG-based delays in connection to the corresponding delays of SET-based delays are tabulated in Table-15 and the three curves of (i) SET-

based delay, (ii) LTG-based delay and (iii) Speed-up times (ratio) versus the different logic gates / circuits are depicted in Fig.19.

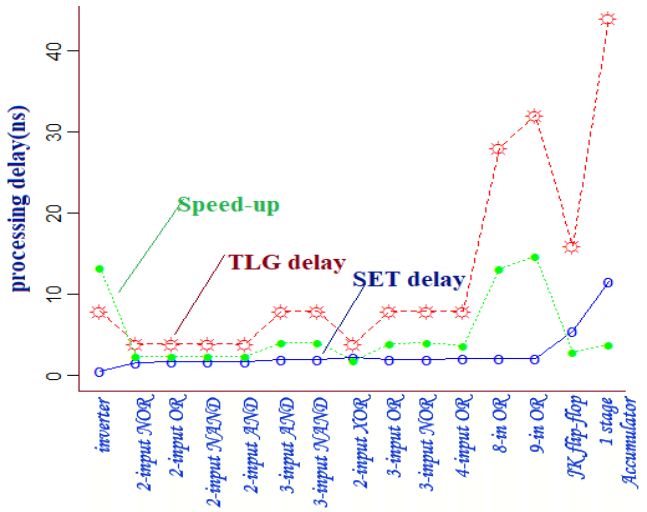


Fig. 19 Delays of SET- and TLG- based and their ratio

4. Switching delays of SET and LTG

Given that the probability of error is 10^{-12}, the processing delays in the case of different circuits we have tracked are given in Table-15. From this table, we can easily have that the LTG based circuit is at least 2 times faster than the SET based circuit. In view of CMOS, the LTG based circuit is 8 times faster than that.

IV. CONCLUSION

In the accumulator of the central processing unit (CPU), the intermediate results of the ALU operations are stored. In the absence of an accumulator in a CPU, the operating system writes the result of each operation (like addition, multiplication, shift, AND, OR etc.) to the main memory. The main memory is really slower to access than to access an accumulator. Since the technology, bringing to bear upon for the main memory, is really slower. Different operational results can be produced and stored in the accumulator by supplying the high logical value to the control variable lines S1 through S9. As the tunnel junction is very small in size (diameter 10nm), the concentration density of the TLG based circuits must be high. The maximum power consumed for the Accumulator is 224.7meV= $224.7 \times 10^{-3} \times 1.602 \times 10^{-19}$ =3.599 $\times$ 10^{-24} Joule which is, really, very low amount. From the perspective of the result of the processing delay, it is observed that the TLG-based circuit is at least 2-times faster in comparison with the SET based circuit and 8-times faster than the CMOS based circuit. In our observation, we have decided that after every 46.16ns one desired output is fanned out from output terminal of the accumulator. If we take the output of a complete 4-bit Accumulator then the fan-out will be $1 \div (46.16ns)$ =21.66MHz. Almost all the given circuits are verified by the simulator-SIMON and realized that the simulated results are matched with the theoretical data. The reality is that atmospheric temperature in single electron tunneling based linear threshold logic technology is to be maintained at very close to 0K and it is a very challenging job.

V. REFERENCES

[1]. Chin-Heng Liu, Chia-Chun Lin et al, "Threshold Function Identification by Redundancy Removal and Comprehensive Weight Assignments" IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 38, NO. 12, DECEMBER 2019

[2]. A.K. Biswas, "Measuring of an unknown voltage by using single electron transistor based voltmeter" , Semiconductor Physics, Quantum Electronics & Optoelectronics(SPQEO), 24 (3),

P. 277-287 (2021) August 2021; ISSN 1605-6582 (On-line)

[3]. Anup Kumar Biswas, "A High-Speed Bidirectional Register with Parallel Loading using single electron Threshold Logic Technology" ,International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Vol. 8 (4) : pp394-408; July-August-2021, Print ISSN: 2395-1990 | Online ISSN : 2394-4099 (www.ijsrset.com) doi : https://doi.org/10.32628/IJSRSET

[4]. Anup Kumar Biswas, "Integrated-Circuit Random Access Memory based on an emerging Technology—electron tunneling through Tunnel Junction" , International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Vol. 8 (4) pp.409-424 July-August-2021, Print ISSN: 2395-1990 | Online ISSN : 2394-4099

[5]. Anup Kumar Biswas, "A High-Speed Bidirectional Register with Parallel Loading using single electron Threshold Logic Technology", International Journal of Scientific Research in Science, Engineering and Technology Print ISSN: 2395-1990 | Online ISSN : 2394-4099 (www.ijsrset.com),July-August-2021, 8 (4) : pp394-408

[6]. Anup Kumar Biswas, "Application of single electron threshold logic gates and memory elements to an up-down Counter" International Journal of Creative Research Thoughts (IJCRT) | Volume 9, Issue 6 June 2021 | ISSN: 2320-2882

[7]. Anup Kumar Biswas, -"Implementation of A 4n-Bit Comparator based on IC Type 74L85 using Linear Threshold Gate Tunneling Technology" International Journal of Engineering Research & Technology ISSN: 2278-0181, Vol. 10 Issue 05, May-2021 pp.299-310,

[8]. Anup Kumar Biswas, "State Transition Diagram for A Pipeline Unit based on Single Electron Tunneling" International Journal of Engineering Research & Technology (IJERT) Vol. 10 Issue 04, April-2021pp.325-336, ISSN: 2278-0181

[9]. Anup Kumar Biswas, "Design of A Pipeline for A Fixed-Point Multiplication using Single Electron Tunneling Technology", International Journal of Engineering Research & Technology ISSN: 2278-0181, Vol. 10 Issue 04, April-2021 pp. 86-98,

[10]. A. K. Biswas and S. K. Sarkar: "An arithmetic logic unit of a computer based on single electron transport system" (SPQEO) Semiconductor Physics, Quantum Electronics & Opt-Electronics. 2003. Vol 6. pp 91-96 No.1,

[11]. A.K. Biswas and S. K. Sarkar: -"Error Detection and Debugging on Information in Communication System Using Single Electron Circuit Based Binary Decision Diagram." (SPQEO) Semiconductor Physics Quantum electronics and opt electronics, Vol. 6, pp.1-8, 2003

[12]. Alexander N. Korotkov, "Single-electron logic and memory devices" INT. ELECTRONICS, 1999, Vol. 86, No. 5, 511- 547

[13]. Casper Lageweg, Student Member, IEEE, Sorin Cot¸ofanˇa, Senior Member, IEEE, and Stamatis Vassiliadis, Fellow, IEEE "Single Electron Encoded Latches and Flip-Flops" IEEE TRANS. ON NANOTECHNOLOGY, VOL. 3, NO. 2, JUNE 2004

[14]. C. Lageweg, S. Cot¸fanˇa, and S. Vassiliadis, "A linear threshold gate implementation in single electron technology," IEEE Computer Society VLSI Workshop, Apr. 2001, pp. 93

[15]. K. Likharev, "Single-electron devices and their applications," Proc. IEEE, vol. 87, pp. 606–632, Apr. 1999.

[16]. J. R. Tucker,- "Complementary digital logic based on the Coulomb-blockade", Journal of Applied Physics., vol. 72, no. 9, pp. 4399–4413, November 1992.
[17]. Jacob. Millman and C. C. Halkias; "Integrated Electronics- Analog and Digital Circuits and Systems -second edition" McGraw Hill Education;
[18]. Millman's Electronic Devices & Circuits 4th Edition (English, Paperback, Jacob Millman)

BIOGRAPHY

ANUP KUMAR BISWAS, is an Assistant Professor in Computer Science and Engineering, Kalyani Govt. Engineering College,West Bengal, India. He defended his Ph.D.[Engg.] thesis in 2005 in the stream of Electronics and Telecommunication Endineering at Jadavpur University. He was a Senior Research Fellow(SRF) in Faculty of Engineering and Technology (*FET*) from 2002 to 2005 at the Department of Electronics &TC. Dr. Biswas has published 27 papers in national, international journals and conferences. The area of his scientific interests are computer arithmetic, parallel architectures, nanotechnology, and computer-aided design. He is engaged in research activities and teaching last 18 years.

Cite this article as :

Dr. Anup Kumar Biswas, "An Accumulator using Electron Tunneling Through Tunnel Junction", International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Online ISSN : 2394-4099, Print ISSN : 2395-1990, Volume 8 Issue 5, pp. 163-183, September-October 2021. Available at doi : https://doi.org/10.32628/IJSRSET218512
Journal URL : https://ijsrset.com/IJSRSET218512