International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 10, October 2026)

Basic Logic Module (BLM) –SET Based Threshold Logic Multiplexer

Anup Kumar Biswas
Assistant Professor, Computer Science and Engineering, Kalyani Govt. Engineering College, Kalyani, Nadia-741235, West Bengal, India

***Abstract*-- Modern electronics demand low cost, ultra-low power consumption, high processing speeds, and maximum integration density. These attributes are economically essential across engineering, technology, business, and manufacturing sectors. This work demonstrates that these requirements can be effectively met by designing circuits with Single Electron Tunneling Transistors (SETTs) and Threshold Logic Gates (TLGs). Because a single electron possesses sufficient charge to store bits of information, Single Electron Devices (SEDs) operate at a fraction of the energy required by conventional complementary metal-oxide-semiconductor (CMOS) circuits. For complex circuit architectures, TLGs provide an ideal implementation framework by combining ultra-low noise thresholds with processing delays that approach fundamental electronic speeds. This paper demonstrates the implementation of fundamental logic gatesand a 4×1 multiplexer, using this hybrid approach. The corresponding performance metrics—including operational speed, power dissipation, component count, and fan-out constraints—are detailed in the subsequent sections.**

***Keywords*--Tunneling, Coulomb blockade, threshold, processing delay.**

I. Introduction

Single Electron Tunneling (SET) technology offers a powerful framework for implementing the full spectrum of digital electronics, ranging from basic logic gates to complex combinational and sequential processing circuits. By leveraging nanoscale electronic control, SED-based designs drastically reduce active component counts compared to traditional CMOS architectures. Operationally, single-electron tunneling events are governed by an applied bias voltage alongside multiple input voltages, which are coupled to isolated nanoscale islands via small capacitances. To validate the efficacy of this framework for scalable digital systems, this paper presents the design and architectural synthesis of fundamental logic gates and a 4×1 multiplexer,. The comprehensive architectural topologies and performance evaluations for these components are systematically detailed from Section 5 onward.

II. Literature Review

Progress in advanced computing, multimedia, aerospace communication, medicine, and smart agriculture relies heavily on the development of high-speed processing devices. Accelerating these modern systems requires a fundamental transition toward ultra-fast basic electronic components. Single-Electron Tunneling (SETun) circuits have emerged as a highly promising alternative capable of meeting these strict performance demands. Over the past few decades, this research domain has evolved from Binary Decision Diagrams (BDD) [6,7] to single-electron transistors (SETs)[6–10, 15,16], and eventually to modern threshold logic gates [17]. Researchers in tunneling circuitry and threshold logic anticipate that SET-based devices will deliver substantial improvements, including microscopic dimensions, reduced production costs, minimal power consumption, high integration density, negligible operational delay, and ultra-low noise levels. Building upon these advantages, several studies have successfully demonstrated the implementation of highly complex SET-based architectures.

III. Tunnel Junction With A True Capacitor

Figure 1 illustrates a basic SET configurations consisting of a true capacitor $C_c$connected in series with a tunnel junction capacitor C_j. The intersection between these two components forms a nanoscale conducting island (represented by the black shaded circle), which serves as a circuit node containing k excess electrons. In this setup, electron tunneling occurs exclusively across the tunnel junction when external forces compel electrons to enter or exit this island node.

The electrical state of the node is governed by the value of k:

- **k = 0:** Indicates the island is in absolute **electrical equilibrium**, meaning the number of electrons perfectly balances the number of protons, leaving zero net charge.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 10, October 2026)

- **k< 0:** Represents a state where k electrons have been removed from the island. Because an individual electron carries a negative charge $q_e = -1.602 \times 10^{-19} C$, a negative k value yields a net **positive charge** on the island.
- **k> 0:** Shows that surplus electrons have been added to the node, resulting in a net **negative charge** accumulation on the island.

Tunnel Junction Electrical Characteristics and Stability Analysis

The **critical voltage** V_C of a tunnel junction is defined by the following expression[8, 9]:

$$V_C = \frac{e}{2(C_C + C_j)} \quad \text{..... (1)}$$

Following the sign convention illustrated in Fig. 1, the total junction voltage V_j across the tunnel junction is given by:

$$V_j = \frac{C_c V_{in}}{C_C + C_j} + \frac{ke}{C_C + C_j} \quad \text{..... (2)}$$

The components of Equation (2) are defined as follows:

- *First term:* Represents the capacitive division of the input voltage V_{in} across the two series-connected capacitances, C_C and C_j.
- *Second term:* Represents the voltage contribution from k excess electrons present on the island, divided by the total capacitance $(C_j + C_C)$ between the input and ground.

The resulting output voltage V_{out} of the circuit in Fig. 1 is expressed as:

$$V_{out} = \frac{C_j V_{in}}{C_C + C_j} - \frac{ke}{C_C + C_j} \quad \text{..... (3)}$$

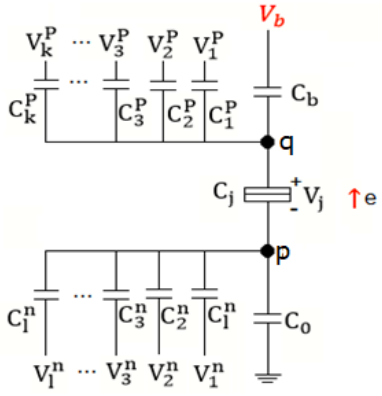


Fig. 3 Multiple input single output threshold logic gate

Stability Condition

The circuit remains in a stable state as long as it satisfies the condition $|V_j| < V_C$. By substituting Equations (1) and (2) into this condition, the stability range for the input voltage V_{in} is established as:

$$\frac{-e-2ke}{2C_c} < V_{in} < \frac{e-2ke}{2C_c} \quad \text{..... (4)}$$

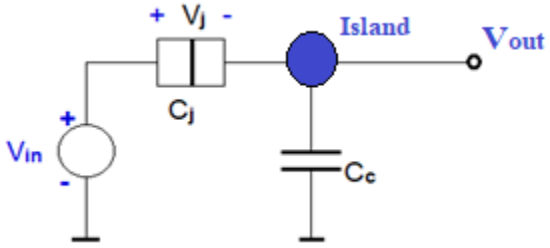


Fig.1 the tunnel junction and capacitance schematicFig.

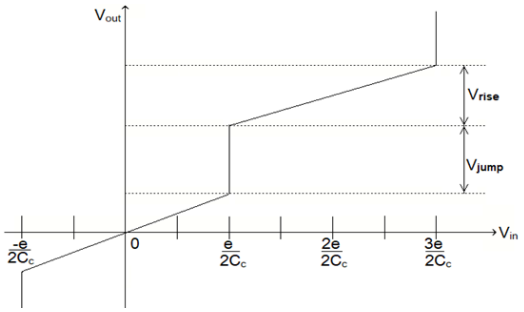


Fig. 2 corresponding junction characteristics.

For any analog input voltage V_{in}, the circuit stabilizes at an integer value k that satisfies equation (4). Gradual changes to V_{in} initially cause a proportional output shift due to the capacitive division shown in Fig. 1. This linear behavior continues until the absolute junction voltage $|V_j|$ exceeds the critical threshold V_C. Once crossed, an electron tunnels through the junction toward the higher voltage side. Specifically, a positive V_{in} forces an electron to tunnel away from the island node, triggering a sudden spike in the island's voltage. Conversely, a negative V_{in} causes an electron to tunnel onto the island, resulting in a sudden drop in node voltage.

The resulting input-output transfer characteristics are illustrated in Fig. 2. The abrupt shift in output voltage is defined as $V_{jump} = \frac{e}{C_C + C_j}$. Meanwhile, the linear voltage rise across the stable range—driven by capacitive division over C_C—is expressed as $V_{rise} = \frac{eC_j}{(C_C + C_j)C_C}$.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 10, October 2026)

The ratio of these two values yields the circuit's transfer function: $\frac{V_{jump}}{V_{rise}} = \frac{\frac{e}{C_C+C_j}}{\frac{eC_j}{(C_C+C_j)C_C}} = \frac{C_c}{C_j}$. If $V_{jump} >> V_{rise}$ (which occurs when $C_c >> C_j$), this relationship forms the distinct staircase pattern shown in Fig. 2.

IV. Multiple-Input Threshold Logic Gate (TLG)

A multiple-input threshold logic gate (TLG) consists of a tunnel junction characterized by a capacitance C_j and an internal resistance R_j. As illustrated in Fig. 3, two distinct sets of multiple input-signal levels, V_k^Ps and V_l^ns, are connected at two separate nodes, designated as 'q' and 'p'.

- **Node 'q' (Top Inputs):** Each positive-side input voltage V_k^P is capacitively coupled to node 'q' via corresponding capacitances C_k^P. Additionally, a supply/bias voltage V_b is connected to node 'q' through a biasing capacitor C_b.
- **Node 'p' (Bottom Inputs):** Each negative-side input voltage V_l^n is capacitively coupled to node 'p' through capacitors C_l^n. The tunnel junction capacitor C_j directly connects to node 'p', which is further coupled to the ground through a dedicated coupling capacitor C_o.

Mathematical Functionality

A TLG implements a switching operation governed by the signum function sgn{S(x)}. The output characteristics are defined by the following dependent equations:

$$g(x) = \text{sgn}\{S(x)\} = \begin{cases} 0, & if\ h(x) < 0 \\ 1, & if\ h(x) \geq 0 \end{cases} \quad \text{(5a)}$$

$$S(x) = \sum_{k=1}^{n}(w_k \times x_k) - \theta \quad \text{(5b)}$$

Where:

- x_k represents the n-Boolean inputs.
- w_k represents the corresponding n integer weights assigned to those inputs.
- θ represents the threshold value of the gate.

Linear Threshold Gate (LTG) Operation:

In a Linear Threshold Gate (LTG), the weighted sum of the inputs, $\sum_{k=1}^{n}(w_k \times x_k)$, is compared directly against a predefined threshold value, θ. The output logic state is determined by the following condition:

- Logic High ("1"): When the weighted sum is greater than or equal to the threshold $\sum_{k=1}^{n}(w_k \times x_k) \geq \theta$.
- Logic Low ("0"): When the weighted sum falls below the threshold.

Junction Voltage and Input Vector Dynamics:

The total junction voltage V_j is modulated by two distinct groups of weighted input signals:

- Positive Weighted Inputs (Top): The input voltage signals $V_1^P, V_2^P, V_3^P, \ldots, V_k^P$ are scaled by their corresponding vector capacitances $C_1^P, C_2^P, C_3^P, \ldots, C_k^P$, and added to the junction voltage.
- Negative Weighted Inputs (Bottom): The input voltage signals $V_1^n, V_2^n, V_3^n, \ldots, V_l^n$ are scaled by their corresponding vector capacitances $C_1^n, C_2^n, C_3^n, \ldots, C_l^n$ and subtracted from the junction voltage.

Tunneling Conditions and Threshold Adjustment:

To enable single-electron tunneling, the circuit evaluates the system voltage against the critical voltage V_c, which functions as the intrinsic threshold of the tunnel junction. The gate threshold value θ can be tuned externally using the supply/bias voltage V_b, which is coupled to the tunnel junction via a bias capacitance C_b.

A discrete tunneling event triggers if and only if an electron successfully passes through the junction from node 'p' to node 'q', following the direction indicated by the red arrow in **Fig. 3**.

To determine the capacitance values and the critical voltage across the tunnel junction, we define the following total capacitance notations:

$$C_\Sigma^P = C_b + \sum_{k=1}^{g} C_k^P \quad \text{(6)}$$

$$C_\Sigma^n = C_0 + \sum_{l=1}^{h} C_l^n \quad \text{(7)}$$

$$C_T = C_\Sigma^P C_j + C_\Sigma^P C_\Sigma^n + C_j C_\Sigma^n \quad \text{(8)}$$

When all voltage sources in Fig. 3 are directly coupled to the ground, the system simplifies into a series network of three core capacitances: C_Σ^P, C_Σ^n and C_j. Within this framework, C_T in Equation (8) represents the sum of all two-term products of these three foundational capacitances.

Our primary objective is to derive the expression for the critical voltage V_c across the tunnel junction. To achieve this, the network's capacitances are divided into two distinct categories:

a) The tunnel junction capacitance C_j.
b) The equivalent capacitance C_e of the remaining circuit, as viewed from the terminals of the tunnel junction.

Following the established frameworks in literature [1–4, 10], the threshold or critical voltage for the tunnel junction can be evaluated as:

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 10, October 2026)

$$V_c = \frac{e}{2(C_j + C_e)} \quad \text{..... (9)}$$

$$= \frac{e}{2[C_j + (C_\Sigma^P || C_\Sigma^n)]}$$

$$= \frac{e}{2[C_j + \frac{(C_\Sigma^P) * (C_\Sigma^n)}{(C_\Sigma^P + C_\Sigma^n)}]}$$

$$= \frac{e(C_\Sigma^P + C_\Sigma^n)}{2[C_j * (C_\Sigma^P + C_\Sigma^n) + (C_\Sigma^P) * (C_\Sigma^n)]}$$

$$= \frac{e(C_\Sigma^P + C_\Sigma^n)}{2C_T} \quad \text{..... (10)}$$

Tunneling occurs across the junction whenever the magnitude of the junction voltage $|V_j|$ satisfies the critical threshold cor
:

$$|V_j| \geq V_c \quad \text{..... (11)}$$

Conversely, when the junction voltage remains below this critical threshold $|V_j| < V_c$, no electron tunneling takes place. In this regime, the circuit is suppressed from transferring charge and maintains a fixed, stable state.We can also write for the function S(x) as [9, 10]

$$C_T S(x) = C_\Sigma^n \sum_{k=1}^{g} C_k^P V_k^P - C_\Sigma^p \sum_{l=1}^{h} C_l^n V_l^n - 0.5(C_\Sigma^P + C_\Sigma^n)e + C_\Sigma^n C_b V_b \quad \text{.....(12)}$$

This can be simplified into the standard form:

$$S(x) = C_\Sigma^n \sum_{k=1}^{g} C_k^P V_k^P - C_\Sigma^p \sum_{l=1}^{h} C_l^n V_l^n - \theta \quad \text{.....(13)}$$

where the threshold parameter θ is defined as:

$$\theta = 0.5(C_\Sigma^P + C_\Sigma^n)e - C_\Sigma^n C_b V_b \quad \text{.....(14)}$$

V. NOT Gate

A NOT gate (inverter) complements its input signal, producing an inverted output as illustrated in Fig. 4(d)-(e). The inverter consists of two single-electron transistors (SET1 and SET2) connected in series. Two identical input voltages, V_{in}, are coupled to the islands of SET1 and SET2 through two equal-valued capacitors, C_1 and C_2, respectively.

Each SET island is composed of gold with a 10 nm diameter, maintaining an individual capacitance of less than 10^{-17} F. The intermediate node between SET1 and SET2 serves as the output terminal V_0 and is tied to the ground via a load capacitor $C_L$to mitigate undesirable charging effects.

For the inverter simulation, the baseline capacitance is defined as 1aF, using the following designated parameter values:

- Gate Voltages: :V_{g1}=0, V_{g2}=0.1×$\frac{q_e}{C}$
- Capacitances: $C_L = 9C$, , $C_1 = \frac{1}{2}C$,$C_2 = \frac{1}{2}C$,$C_{g1} = \frac{17}{4}C$ and $C_{g2} = \frac{17}{4}C$
- Tunneling Coefficients / Parameters:$t_4 = \frac{1}{10}C$, $t_3 = \frac{1}{2}C$, $t_2 = \frac{1}{2}C$, $t_1 = \frac{1}{10}C$
- Resistances: R1 =R2=50KΩ

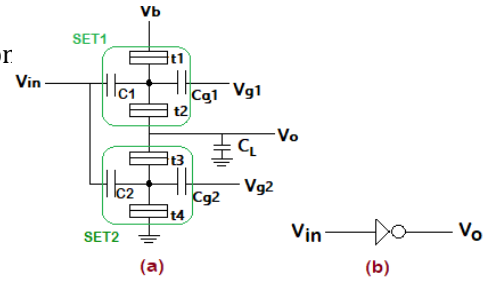
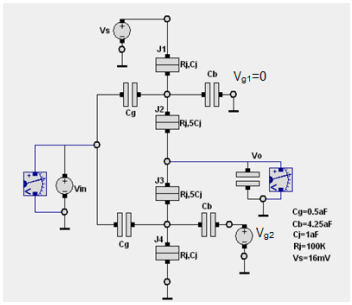
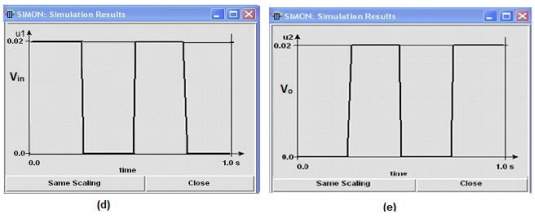


Fig. 4(a) Buffer **(b) symbol**



4(c) Simulation set of Buffer



4 (d), (e) result of Buffer input and output

The proposed Single-Electron Transistor (SET) inverter circuit—illustrated conceptually along with its simulation setup and results in Figs. 4(a) through 4(e)—operates by producing a complementary output logic state relative to its input.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 10, October 2026)

Specifically, the output voltage V_0switches to a logic high when the input voltage V_{in} is logic low (0 V), and conversely drops to a logic low when $V_{in}$is logic high.

To achieve this complementary behavior, the gate bias voltages for the two transistors must be pre-configured to $V_{g1} = 0\,V$ andV_{g2}=16mV. A common input voltage $V_{in}$is tied directly to both SET1 and SET2:

- Logic Low Input V_{in}= 0V**:** When V_{in}low, SET1 enters conduction mode via electron tunneling. Simultaneously, SET2 is forced into a Coulomb blockade state, which suppresses steady current flow and prevents electron tunneling. This combined action pulls the output voltage V_0up to the bias voltage V_b, establishing a logic high state at the output.
- Logic High Input V_{in}≈16mV**:** When a logic high is applied, it shifts the induced charge on the island nodes of both SET1 and SET2 by a fraction of an elementary electron charge. This fractional shift pushes the upper transistor (SET1) into a Coulomb blockade state while rendering the lower transistor (SET2) conductive. As a result, the output voltage drops from high to low (logic 0).

Consequently, the dual-transistor configuration dynamically alternates between conduction and Coulomb blockade states to execute standard inverter logic functions.

Parameter Selection and Input Logic Levels

For simulation and validation purposes, the Boolean logic levels are defined using the elementary electron charge $q_e$and the island capacitance C:

- **Logic "0":**0V
- **Logic "1":** Defined by the expression $0.1\times\frac{q_e}{C}$

Assuming a baseline capacitance of C=1aF ($1\times 10^{-18}F$) and an elementary charge of $q_e = 1.602\times 10^{-19}C$ the exact voltage required for a logic "1" is calculated as:

Logic "1"= $0.1\times\frac{1.602\times10^{-19}}{1\times10^{-18}}$=$0.1\times 1.602\times 10^{-2}$=$16.02\times 10^{-3}$ =16.02mV ≅ 16 mV.

VI. OR Gate

To model a two-input OR gate using threshold logic, we define the gate's output using the signum function equation: OR(A,B)=sgn$\{w_A.A + w_B.B - \theta\}$, where w_A and $w_B$represent the weights of inputs A and B, and θ represents the threshold value.

The threshold logic gate is constructed by first establishing the standard OR gate truth table (Table-1) and then comparing the variable weights w_A and w_B against the threshold θ to satisfy all logic conditions.

Table-1

A	B	F(A,B)	θ	Eqn. no.
0	0	0	0<θ	(1)
0	1	1	$w_B \geq \theta$	(2)
1	0	1	$w_A \geq \theta$	(3)
1	1	1	$w_B + w_A \geq \theta$	(4)

By setting w_B=1, w_A=1 and θ=0.5, all four conditional equations in Table-1 are fully satisfied, yielding this specific solution set. Consequently, the threshold logic expression for an OR gate is defined by Equation (15), and its corresponding Threshold Logic Gate (TLG) architecture is illustrated in Fig. 5(a):

$$OR(A,B) = sgn\{A + B - 0.5\} \cdots\cdots (15)$$



Fig. 5(a) Threshold logic OR gate

To implement this OR gate configuration, the hardware parameters for Fig. 5(a) are set as follows:

$C_1^n = C_2^n$=0.5aF,$C_3 = 11.7aF, C_{b1} = C_{b2} = 4.25aF$, $C_{g1} = C_{g2} = 0.5aF, C_L = 9aF, C_0 = 8aF$, R_j=$10^5\Omega$, $V_s = 16mV$.

Following the execution of the simulator using these operational values, the resulting output waveforms are captured and displayed in Figures 5(c) through 5(e), which represent the transient simulation responses of the TLG-based OR gate.

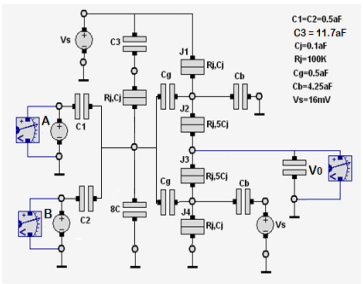
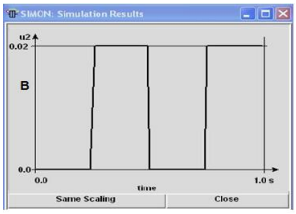
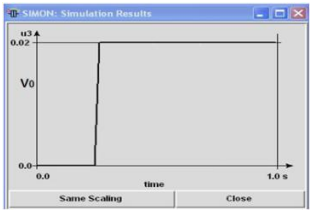


Fig. 5(b) OR gate **(c)**

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 10, October 2026)



(d)



(e)

Fig. 5(c)-(e) simulation responses of the TLG-based OR gate.

VII. 2-INPUT THRESHOLD AND GATE

A 2-input threshold logic AND gate can be designed by analyzing the truth table of a standard AND gate. By comparing the conditional weighted inputs w_A and $w_B$of the two variables A and B against the threshold valueθ, we can establish the system's governing constraints. Solving the four resulting inequalities yields a valid solution set of w_B=1, w_A=1 and θ=2. Based on these derived values, the threshold logic expression for the AND gate is defined in Equation (16), and its schematic representation is illustrated in Figure 6(a).

Table-2

A	B	F(A,B)	θ	Eqn. no.
0	0	0	0<θ	(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)

$$AND(A,B) = sgn\{A + B - 2\} \cdots\cdots (16)$$

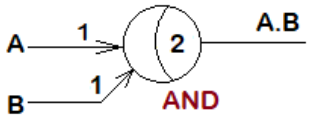


Fig. 6(a) Threshold logic AND gate

The parameters chosen for the **AND gate** implementation include $C_1^n = C_2^n = 0.5\text{aF}, C_{b1} = C_{b2} = 4.25\text{aF}, C_{g1} = C_{g2} = 0.5aF, C_L = 9aF, C_0 = 8aF$, and$R_j = 10^5\Omega$. While the schematic/configuration environment is depicted in **Fig. 6(b)**, the final simulation trajectories are captured in **Figs. 6(c)–(e)**.

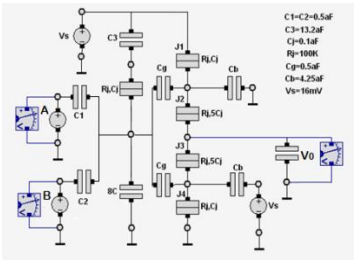
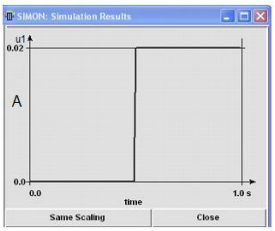
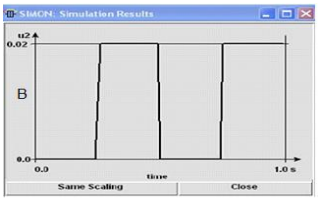


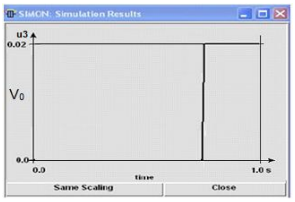
Fig.6 (b) AND Gate



(c)



(d)



(e)

Fig.6 (d), (e) simulation trajectories of 2-input AND gate

7.1 Threshold logic equation for 3-input AND gate

The threshold logic equation for a 3-input AND gate is derived by assuming positive logic, meaning all weights $w_A, w_B, w_C$and the threshold θ must be positive.

By assigning the weights$w_A = 1, w_B = 1$ andw_C=1, and selecting a threshold of $\theta = 2.5$ (or any value within the range 2 <$\theta \leq$ 3), all conditional equations in the fifth column of Table 3 are satisfied. Consequently, the threshold logic equation is defined as:

$$AND(A,B,C) = sgn\{w_A.A + w_B.B + w_C.C - \theta\}$$
$$= sgn\{A + B + C - 2.5\} \cdots\cdots (17)$$

Table-3 (Truth table of 3-in AND gate)

A	B	C	F(A,B,C)= ABC	θ
0	0	0	0	0<θ
0	0	1	0	$\theta > w_C$
0	1	0	0	$\theta > w_B$
0	1	1	0	$\theta > w_B + w_C$
1	0	0	0	$\theta > w_A$
1	0	1	0	$\theta > w_A + w_C$
1	1	0	0	$\theta > w_A + w_B$
1	1	1	1	$w_A + w_B + w_C \geq \theta$

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 10, October 2026)

The corresponding simulation setup and input-output results are illustrated in Figures 7(a) and 7(b), respectively.

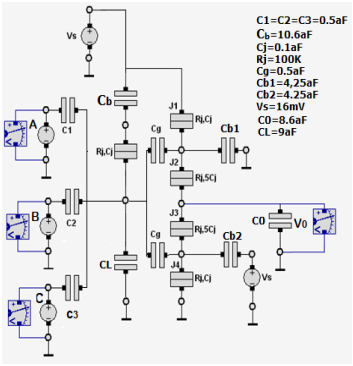


Fig. 7(a) set of 3-input AND gate

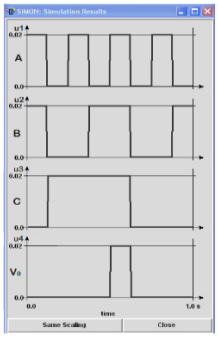


Fig. 7(b) result

VIII. Linear Separability And Threshold Logic Realization Of The XOR Gate

The logical expression for a two-input XOR gate is defined as Y=A.$\bar{B}$ +$\bar{A}$.B, where A and B represent the input variables. When this two-variable XOR function is plotted in a 2D space, the resulting spatial diagram demonstrates that the true outputs (red bubbles) and false outputs (colorless bubbles) In Fig. 8 cannot be divided by a single straight line. Because no linear boundary exists to partition these distinct regions, the XOR Boolean logic function is not linearly separable. Consequently, a single, direct Threshold Logic Gate (TLG) cannot be constructed to represent this function.

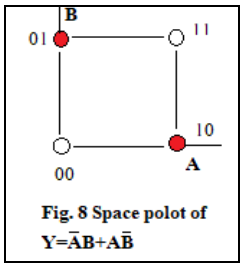


Fig. 8 Space polot of Y=$\bar{A}$B+A$\bar{B}$

To implement the XOR function using threshold logic, the expression must be broken down into sub-components. For the component P = (A.$\bar{B}$), the corresponding threshold gate equation is expressed as:

$$P = sgn\{A + \bar{B} - 2\} \quad \cdots\cdots (18)$$

By applying the Boolean identity B + $\bar{B}$ =1 or $\bar{B}$= − B + 1, Equation (18) can be rewritten in terms of the primary inputs as:

$$P = sgn\{A - B - (1)\} \quad \cdots\cdots (19)$$

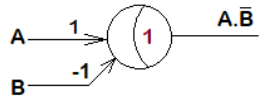


Fig. 8(a) Threshold gate of P = A.$\bar{B}$

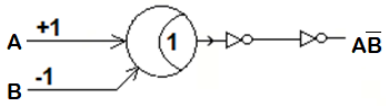


Fig. 8(b) Threshold gate of A.$\bar{B}$ utilising two buffers

This transformation allows the sub-function to be realised using standard threshold weights and thresholds, as illustrated in the gate configurations of Figure 8(a) (Threshold gate of P = A.$\bar{B}$ and Figure 8(b) (Threshold gate of A.$\bar{B}$ buffers).

Table-4
Truth table of equation (19)

A	B	P
0	0	0
0	1	0
1	0	1
1	1	0

Table-5 Truth table of $\bar{P}$

A	B	$\bar{P}$	θ
0	0	1	$0 \geq \theta$
0	1	1	$w_b \geq \theta$
1	0	0	$w_a < \theta$
1	1	1	$w_a + w_b \geq \theta$

We aim to determine the inverting expression of P that yields $\bar{P}$=$\bar{A}$+B. By evaluating the conditional equations in the fourth column of Table 5, this expression can be formulated as $\bar{P}$=sgn $\{-A+B-(-0.5)\}$. The Threshold logic gate of $\bar{P}$ and of P are depicted in Fig. 9(a) and 9(b) respectively.

To implement the threshold logic gate for $\bar{P}$, we define the input logic levels as "0"=0V, and "1" = 16mV. The remaining circuit parameters are specified as: C=1aF, $C_1^n = C_1^P = \frac{1}{2}C$ =0.5aF, $C_b = 10.24aF, C_j = 0.25aF$, $C_L = 9aF, C_0 = 9.5aF$, *and* $R_j = 10^5\Omega$,. Given that $e/C = 160mV$, the bias voltage is calculated as $V_b = 0.95\, e/C = 15.2mV$. Since this value is approximately equal to 16mV, we set $V_b = V_s$=16mV.

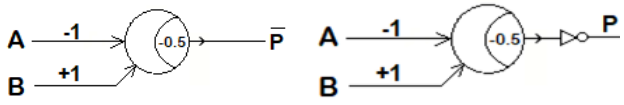


Fig. 9 (a) Threshold logic gate of $\bar{P}$ 9(b) of P

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 10, October 2026)

Next, we convert the Boolean expression Y=A.$\bar{B}$ +$\bar{A}$.Binto its threshold logic form. By assuming P =$\bar{A}$.B, the equation can be rewritten as

$$Y = P + \bar{A}.B \cdots\cdots (20)$$

To determine the threshold logic for the expression in Equation (20), Truth Table-6 is constructed in conjunction with Table 4. Solving the resulting inequalities yields the weight values presented in the fifth column of Table- 6.

Table-6

1st	2nd	3rd	4th	5th
A	**B**	**P**	**Y**	$\boldsymbol{\theta}$
0	0	0	0	$0<\theta$
0	1	0	1	$W_2 \geq \theta$
1	0	1	1	$W_1 + W_3 \geq \theta$
1	1	0	0	$W_1 + W_2<\theta$

After solving these conditional equations, the solution set is obtained as $\{ W_1, W_2, W_3; \theta\}$ = {-1, 1, 2 ; 1}. Consequently, the threshold equation for the XOR function Y is expressed as:

$$Y = \text{sgn}\{ -A + B + 2P - (1)\} \cdots\cdots (21)$$

The corresponding threshold logic gate implementation is depicted in Figure 11(a). For proper operation, an inverter is connected in series to produce the XNOR gate shown in Figure 10. Adding a second inverter in series restores the circuit to an XOR configuration, which is also illustrated in Figure 10. The simulation input-output result is given in Fig. 11(b).

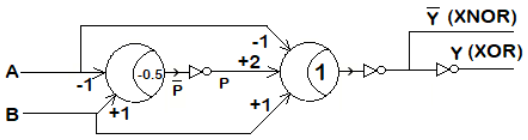


Fig. 10 XOR/ XNOR gate of two inputs

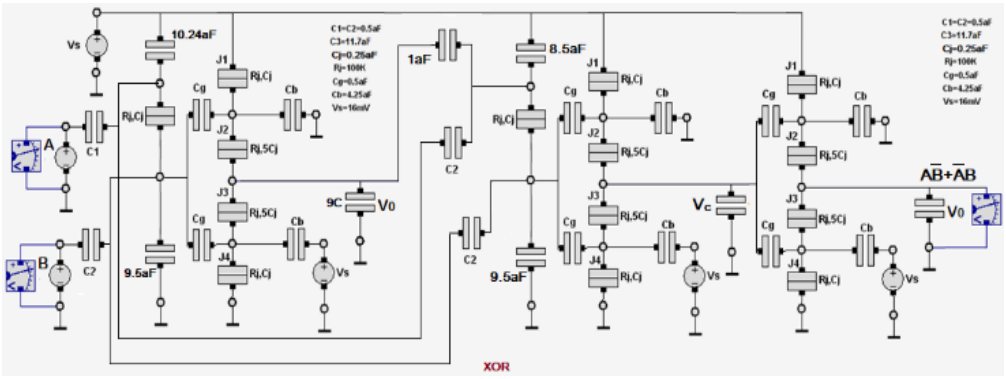


Fig. 11(a) Simulation set of XOR(A,B)

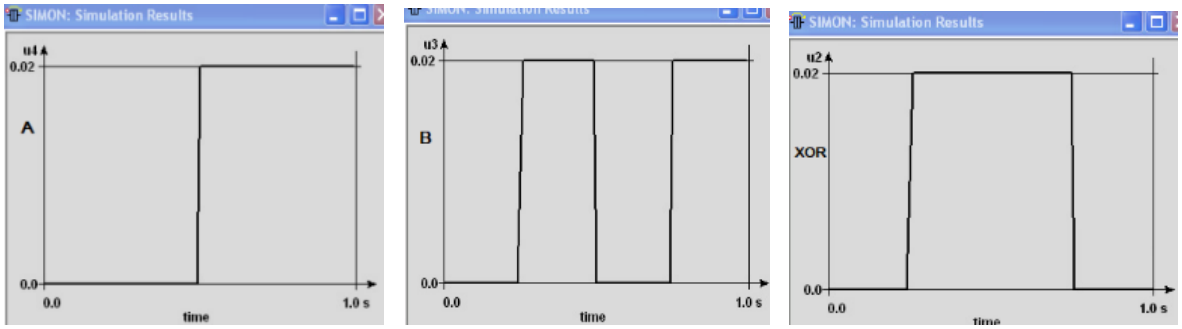


Fig.11(b) Simulation result of XOR(A,B)

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 10, October 2026)

IX. Design And Implementation Of A Threshold Logic Gate (TLG) Based 4:1 Multiplexer

A standard 4:1 Multiplexer (MUX) routes one of four data inputs to a single output terminal based on the binary state of two control lines. The structural layout of the proposed Threshold Logic Gate (TLG) based 4:1 Multiplexer is illustrated via a linear threshold diagram in Fig. 12(a). By modulating the logic levels of the selection lines $C_0$and C_1, the circuit selectively switches input A, B, C, or D directly to the output. The corresponding functional behavior and input-output relationships are detailed in Fig. 12(b).

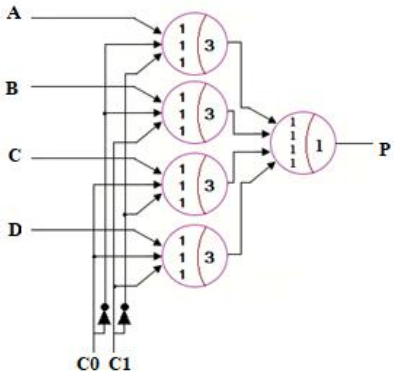


Fig. 12(a) TLG based 4:1 Multiplexer

C_1	C_0	P
0	0	D
0	1	C
1	0	B
1	1	A

Fig. 12(b) Functional Diagram

To realize this architecture using threshold logic, three core components are required: four 3-input AND gates, one 4-input OR gate and two inverters (two NOT gates)

By precisely tuning and calibrating the threshold parameters (weights and thresholds) for the TLG-based 3-input AND and 4-input OR gates, the complete multiplexer circuit was successfully integrated. The final simulation setup for this implemented 4:1 Multiplexer is depicted in Fig. 13.

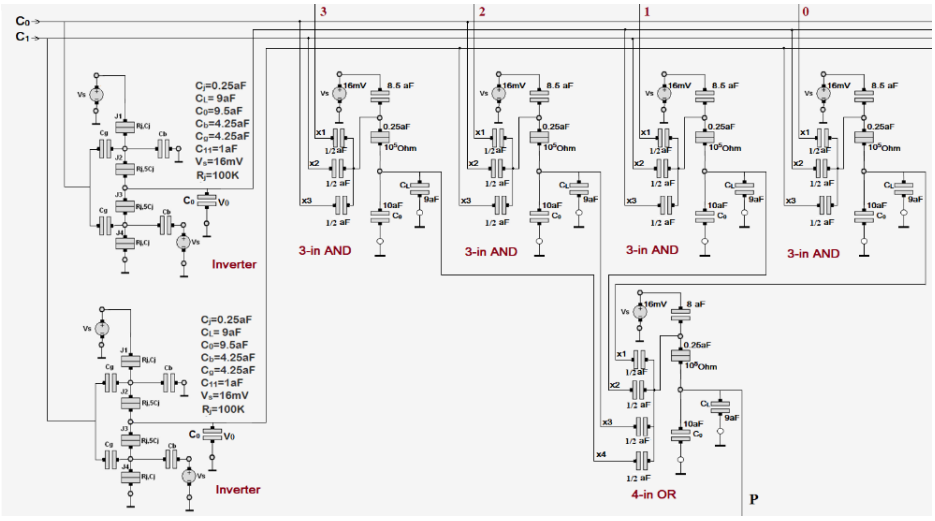


Fig.13 Simulation set of a 4:1 Multiplexer

9.1 Use of 4:1 Multiplexer

Because a4:1 Multiplexer acts as a data selector or controllable switch, it is used across a variety of hardware designs and computing architectures:

(i). Data Selection and Routing

The primary use of a 4:1 multiplexer is to control which data source connects to a shared destination line.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 10, October 2026)

- System Bus Sharing: In computers, multiple sub-components (like different registers or memory units) often need to send data over a single shared bus line. A multiplexer dictates which component gets access to the bus at any given millisecond.
- Processor Register Selection: Inside a Central Processing Unit (CPU), a 4:1 MUX can be used in the control unit to select which of four available registers will feed data into the Arithmetic Logic Unit (ALU).

(ii). Parallel-to-Serial Data Conversion

Computers process data in parallel (multiple bits at the same time), but digital communication networks often transmit data serially (one bit at a time over a single wire) to save space and cost.

- By connecting 4 parallel data bits to the MUX inputs (I_0 to I_3) and sequencing the select lines (C_1C_0) from 00 through 11, the parallel data is loaded onto the single output wire bit by bit.

(iii). Implementing Combinational Logic Functions

Multiplexers act as universal combinational circuits. Instead of wiring complex networks of individual AND, OR, and NOT gates, an engineer can use a single 4:1 MUX chip to build logic functions:

- Two-Variable Logic Functions: A 4:1 MUX can natively implement *any* two-variable Boolean function. By mapping variables A and B to the select pins, you simply tie the data inputs to either Logic 1 (VCC) or Logic 0 (Ground) according to the function's truth table.
- Three or Four-Variable Functions: Using simple techniques like an "implementation table," a 4:1 MUX can also handle functions with more than two variables by passing the extra variables directly into the data input lines.
- Building Logic Gates: You can easily configure a 4:1 MUX to function identically to standard logic gates, such as an AND gate, OR gate, or XOR gate.

(iv). Designing Larger Multiplexers (Cascading)

When an application requires a larger multiplexer (like an 8:1 MUX or a 16:1 MUX) but the designer only has smaller chips on hand, 4:1 multiplexers can be chained together in trees.

- For example, a common structural method allows engineers to construct a 16:1 MUX using four 4:1 MUXes for the first stage and an additional MUX or OR gate framework for the output stage.

(v). Telecommunications and Transmitter Circuits

In communication systems, a 4:1 MUX allows up to four separate audio, video, or data channels to take turns sending signals over a single transmission medium (like a telephone line or fiber optic cable). This application maximizes line efficiency and prevents the costly requirement of running dedicated wires for every individual signal source.

X. Discussion

In this work, we implemented several threshold logic gates—including Inverter, OR, AND, and XOR—alongside and a 4×1 Multiplexer circuit. The respective circuit diagrams and simulated waveforms are presented in their corresponding sections. Furthermore, we depicted these threshold logic gates by analyzing their distinct linearly separable logic equations.

To evaluate the operational speed of the proposed circuits, their processing delays must be analyzed. Determining this delay requires accounting for both the critical voltage V_c and the junction capacitance C_j. For our experimental setup, the operational temperature is set to T = 0K. Following established methodologies [8–10], the switching and processing delay of a given logic circuit can be calculated and investigated using the following expression:

$$\text{Delay} = -(e|\ln(P_{error})|R_t) / (\,|V_j| - V_c) \quad \cdots\cdots (22)$$

where V_j represents the junction voltage, V_c is the critical voltage, and R_t denotes the junction resistance.

A tunneling event occurs exclusively when the critical voltage V_c drops below the absolute junction voltage V_j, such that $V_c < |V_j|$. For the 2-input OR gate shown in Fig. 5(b), this condition is satisfied when input V_{in1}=A is at logic 1. This yields a junction voltage V_j =11.8mVagainst a critical voltage V_c=11.58mV. Given an acceptable error probability P_{error}= $10^{-12}$and a tunnel junction resistance$R_t = 10^5\Omega$, the calculated switching delay is $0.07281|\ln(P_{error})|$ns, which evaluates to 1.675 ns. Following this methodology, the circuit delays for various other configurations were determined and are compiled in Table 7.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435 (Online) Volume 15, Issue 10, October 2026)

Table-7

Gate/Device	elements	Delay	Switching Energy
inverter	09 elements	$0.022\|\ln(P_{error})\|$ ns	10.4 meV
2-input NOR	14 elements	$0.072\|\ln(P_{error})\|$ ns	10.7 meV
2-input OR	14 elements	$0.062\|\ln(P_{error})\|$ ns	10.8 meV
2-input NAND	14 elements	$0.080\|\ln(P_{error})\|$ ns	10.7 meV
2-input AND	14 elements	$0.062\|\ln(P_{error})\|$ ns	10.8 meV
3-input AND	15 elements	$0.104\|\ln(P_{error})\|$ ns	11.58 meV
3-input NAND	15 elements	$0.072\|\ln(P_{error})\|$ ns	11.58 meV
2-input XOR	29 elements	$0.102\|\ln(P_{error})\|$ ns	21.2 meV
3-input OR	15 elements	$0.104\|\ln(P_{error})\|$ ns	11.58 meV
3-input NOR	15 elements	$0.104\|\ln(P_{error})\|$ ns	11.55 meV
4-input OR	16 elements	$0.118\|\ln(P_{error})\|$ ns	12.35 meV
4x1-MUX	52 elements	$0.210\|\ln(P_{error})\|$ ns	39.74 meV

When an electron tunnels through a junction, the total energy of the circuit shifts. The net energy change ΔEbetween the states immediately before and after the tunneling event is governed by Equation (23):

$$\Delta E = E_{before\ tunnel} - E_{after\ tunnel} = -e(V_c - |V_j|) \quad \cdots\cdots (23)$$

This value represents the total switching energy dissipated during the tunneling process. The resulting performance characteristics are plotted in Fig. 14, which illustrates switching delay as a function of unit capacitance C [Fig. 14(a)] and switching delay as a function of switching error probability [Fig. 14(b)].

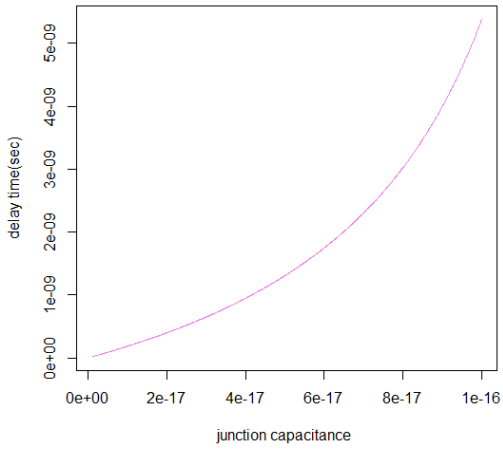


Fig. 14(a) Delay vs. Error Probability

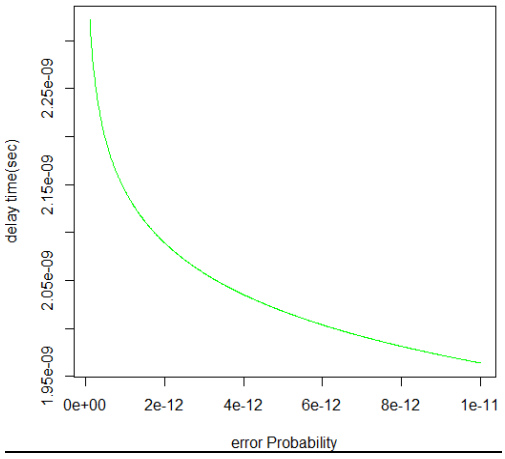


Fig. 14(b) Delay Vs. capacitance

XI. Conclusion

Driven by advancements in engineering, science, and the escalating demand for high-speed electronic devices, optimizing the operational velocity of fundamental logic gates has become crucial. Single-Electron Transistor (SET) and Threshold Logic Gate (TLG) paradigms offer significantly higher circuit speeds compared to conventional CMOS-based designs.

This work analyzes a specific tunnel junction and its characteristic curve, alongside the design of multi-input single-output threshold logic gates, including 2-input AND, OR, and XOR gates, a 3-input AND gate and a 4x1 multiplexer. Furthermore, a foundational Linear Threshold Gate (LTG) implementation is presented, establishing a framework to derive an entire family of logic gates.

The proposed circuits were validated through SIMON simulations, and their architectural metrics—specifically element counts, processing delays, and power dissipation—are systematically tabulated. The findings demonstrate that threshold logic gates operate at least twice as fast as their SET-based counterparts. However, a remaining challenge is their real-world operational requirement, which demands temperatures maintained strictly near 0K. Uses of MUX in some areas are cited.

REFERENCES

[1] 2021; ISSN 1605-6582 (On-line)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

[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

[3] 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,

[4] 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

[5] 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,

[6] 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,

[7] 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

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

[9] Casper Lageweg, Student Member, IEEE, SorinCot¸ofan˘a, 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

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

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

[12] J. R. Tucker,- "Complementary digital logic based on the Coulomb-blockade", Journal of Applied Physics., vol. 72, no. 9, pp. 4399–4413, November 1992.

[13] Jacob. Millman and C. C. Halkias; "Integrated Electronics- Analog and Digital Circuits and Systems -second edition" McGraw Hill Education;

[14] Millman's Electronic Devices & Circuits 4th Edition (English, Paperback, Jacob Millman)

[15] Asahi N., Akazawa, M. and Amemiya, Y.. 1997. Single electron logic device based on the binary decision diagram, *IEEE Trans on ED*. 44, 7. pp. 1109-1116.

[16] Asahi N. Akazawa, M. and Amemiya, Y. 1998. Single electron logic systems based on binary decision diagram. *IEICE Trans. Electron*. E-81-C. 1. pp. 49-56.

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

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 and Telecommunication Engineering. Dr. Biswas has published 30 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 21 years.