MultiCraft

International Journal of Engineering, Science and Technology
Vol. 1, No. 1, 2009, pp. 61-73

INTERNATIONAL JOURNAL OF ENGINEERING, SCIENCE AND TECHNOLOGY
www.ijest-ng.com

Single electron based binary multipliers with overflow detection

Souvik Sarkar[1], Anup Kumar Biswas[2], Ankush Ghosh[1], Subir Kumar Sarkar[1]*

[1] Department of Electronics and Telecommunication Engineering, Jadavpur University,Kolkata, India.*
[2] Department of Commuter Science and Engineering, Jalpaiguri Govt. Engineering College, India
[*]Corresponding Author: e-mail:sksarkar@jdvu.ac.in

Abstract

Low power consumption, high operating speed and high integration density equipment(s) are financially indispensable in modern Electronics. Single Electron Device (SED) is one such equipment. Single Electron Devices are capable of controlling the transport of only an electron. A single electron is sufficient to store information in the SED**.** This paper presents the approach for designing multipliers by using single-electron based device. Multipliers with overflow detection based on serial and parallel prefix computation algorithm are elaborately discussed analytically and designed. The overflow detection circuits works in parallel with a simplified multiplier to reduce the overall area and to increase the speed compared to the classical digital circuits. Power consumption in the single electron circuit is low irrespective of Bipolar junction transistor (BJT) or Complimentary Metal Oxide Semiconductor (CMOS) circuits. Power consumption can be drastically reduced by reducing the nodes. The processing speed of SED will be nearly close to electronic speed. Noise during processing becomes ultra-low when the mode built with SEDs is in operation.

Keywords: Single-electron, Binary multiplier, overflow detection, binary decision diagram.

1. Introduction

Single electron based logic gates have already been constructed with binary decision diagram (BDD) (Asahi *et al,* 1998; Biswas *et al,* 2003a) with clock pulses of 1ns each. The technique of tunneling of an electron is utilized for those gates. This technique may also be used for more complex logical circuit (Korotkov *et al,* 1998; Biswas *et al,* 2003b) like a case of a binary multipliers and overflow detection**.** For easily understanding the operation of some of the single electronic gates having different number of inputs have been depicted. Single electron tunneling devices exploit effects that arise due to the quantized nature of charge. These effects have been observed in systems of small metal structures (Asahi *et al,* 1995; Asahi *et al,* 1997), in semiconductors structures and in structures made from conducting polymers. Because these effects are omnipresent in small structures, they are likely to have an impact on any future nano-scale electronic circuits. These devices are able to use in low power circuits as only a few electron is needed for carrying information (Biswas *et al,* 2003a). The speed power product of single electronic device is predicted to lie close to the quantum limit set by the Heisenberg's Uncertainty Principle. The processing speed of such device will be close to the electronic speed (Biswas *et al,* 2003b).

When the number of bits that do not fit into space available then overflow occurs. For example, when an arithmetic logic operation creates a result outside of the range of the representable number, overflow occurs (Schulte, *et al 2000*). If overflow occurs, an error flag is generated to indicate "out of range". Well-known that, multiplication for two n-bit integers produces a n+n=2n product. But some electronic architecture only return n-least significant bit out of the 2n-bits and overflow sets in the product cannot be represented correctly with only n bits. For example, IBM's Power Microprocessor family supports a 32-bit by 32-bit two's complements multiply instruction which returns the least significant 32-bits of the 64-bit product and an overflow flag (IBM, 2000). Similarly Java Virtual Machine supports two integer multiplication instructions; a 32-bit by 32-bit *imul* (IBM, 2000) instruction which returns the 32 least significant bits of the product and 64 bit by 64-bit *imul* instruction which returns the 64 least significant bits of the product (Lindholm and Yelin, 1996). In the present work, signed and unsigned multipliers with overflow detection circuit are implemented using single electron tunneling phenomena.

2. Coulomb blockade and single electron transistor

A tunnel junction shown in Figure 1 is considered to be a thin insulating barrier between the two conducting electrodes. The electrodes may be a superconducting or semiconducting if they are supercononducting, Cooper pairs with a change of two elementary charges i.e., neither superconducting nor semiconducting, electrons with one elementary charge (1.6×10^{-19} C) carry the current.

In classical electrodynamics, no current can flow through an insulating barrier. But in quantum mechanics, there is a nonvanishing (i.e., greater than zero) probability for an electron one side of the barrier to reach the others side. If we apply bias voltage, there will be a current flow. Avoiding additional effects, according to first-order–approximation-tunneling current is proportional to the applied bias voltage. In electrical terms, a tunnel junction behaves like a resistor of a constant value depending experimentally upon the barrier thickness. If two conductors connected with an insulating layer in between has not only a resistance but also a capacitance. In this context the tunnel junction acts as a capacitor and the insulator is said to be dielectric. For the discrete nature of electric charge, current following through a tunnel junction is a series of events in which only one electron goes through the tunnel junction. As the electron tunnels the junction, the tunnel capacitance is charged with an elementary charge building up a voltage V=e/C; C=junction capacitance. If the capacitance of the tunnel junction is very small, the voltage developed in the tunnel junction may be adequate to prevent another electron to tunnel. The electrical current is suppressed then for the bias voltage lower than the voltage developed in the tunnel junction and the resistance of the device no longer remains constant. The increment of the differential resistance of the tuunel junction around zero bias is considered as the Coloumb blockade. So, we can define the Coloumb blockade as increased resistance at very low bias voltages of an electronic device which is having at least one low capacitance tunnel junction.

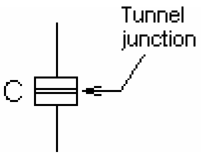


Figure 1. Tunnel Junction

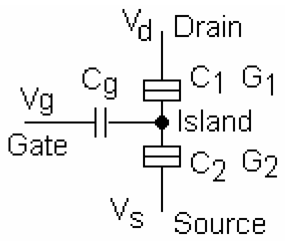


Figure 2. Single Electron Transistor (SET)

The fundamental principle of single-electronics is based on the Coulomb blockade. Single electron tunneling circuits seem to be a promising candidate for future VLSI for its ultra-low power consumption, ultra small size and rich functionality.

Single electron Transistor (SET) is shown in Figure 2. A SET has two tunnel junctions having capacitances and conductances C_1, C2 and G_1, G2 respectively, and shares one common electrode with a low capacitance known as island**.** The electric potential of the island can be tuned by a third electrode, called gate, which is capacitively coupled (gate capacitance $C_{g)}$ to the island. The drain, source and gate voltages are V_d, V_s and V_g respectively. For proper operations of SET both of the conductances G1 and G2, of course, are to be smaller than 1/Rq; where Rq = $h/e^2 \approx 25.8\ K\Omega$ and charging energy $E_C=e^2/(2C)$ [where $C = C_1+C_2+C_g$] has to be greater than thermal fluctuations kT i.e., $E_C>kT$.

3. About root node of single-electron Binary Decision Diagram (BDD)

The path selector depicted in Figure 3 is driven by a signal X_i and a clock pulse Φ_i, i =1, 2, 3, 4. This path selector consists of two SETs and one capacitance C_1. When an electron comes at point A and pulse Φ_i>5mV is applied then the electron can cross tunnel junctions (J_1 and J_2) to B (or C) depending on whether the Coulomb energy [$E_c=e^2/(2C)$] + applied energy is greater than the potential height of the barrier energy of junction(s) J_1 (or J_2). Following this principle, the electron follows the path ABD (or ACE) provided the signal X_i>5mV(or $\overline{X}_i$) and the corresponding total energy Coulomb energy + applied energy is greater than static potential junction energy of J_3 (or J_4).This path ABD(or ACE) is said to be 1-arm (or 0-arm). Figure 4 is the symbol of this path selector.

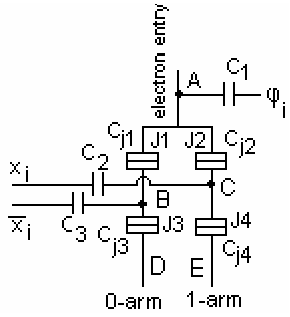


Figure 3. Path selector of an electron

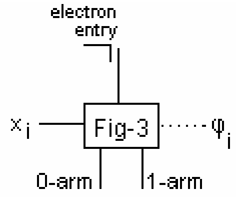


Figure 4. Symbol of path selector

4. Configuration of Electron to Voltage Converter

Figure 5 acts as a buffer. The configuration of the input electron to output voltage has been depicted in Figure 6 and its operation has been explained below.

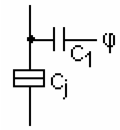


Figure 5. A buffer

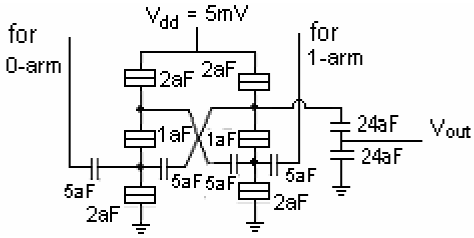


Figure 6. Electron to voltage converter

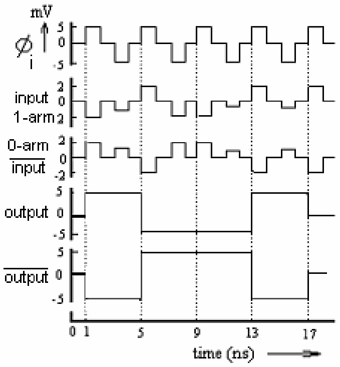


Figure 7. Input output waveforms

It contains six tunnel junction capacitances and six capacitances (their values are depicted in the Figure 7). It has two input terminals: *input(for 1-arm)* and $\overline{input}$ *(for 0-arm)*and one output terminal V_{out} . The 1-arm and 0-arm of a single electron circuit are connected, respectively, to the *input(for 1-arm)* and $\overline{input}$ *(for 0-arm)* of the Figure 6. If an electron reaches at the 1-arm, the output voltage V_{out} becomes approximately 5mV. If the electron reaches at 0-arm then V_{out} will be approximately 0 (zero). The input and output waveforms of Figure 6 are shown in Figure 7.

5. Some Single-Electronic Gates

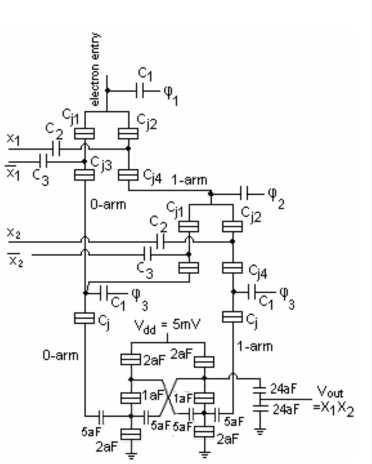


Figure 8. Two-input AND gate

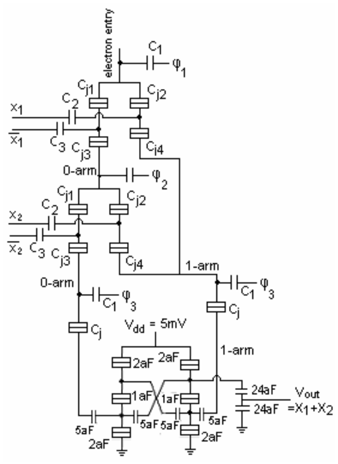


Figure 9. Two-input OR gate

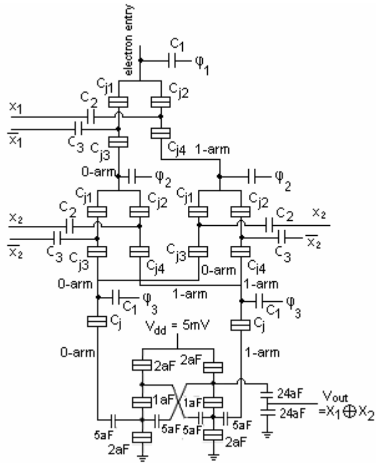


Figure 10. Two-input XOR gate

6. Theorem related to overflow

6.1 Theorem-1:

For two unsigned binary codewords having l and m significant bits respectively, no overflow for the product (multiplication) of these two codewords will happen if total number of significant bit of the product is $n \geq l+m$.

Suppose two unsigned binary word A and B. Significant bits of A include the leftmost 1 and all the remaining bits right to that. If A=00100010, then A has six significant bits. Similarly if B=10010001, then it has eight significant bits. Now $M=A\times B$, then there will be no overflow if the significant bit of the product "M" is $l+m > n$ i.e., we have to prove for non-overflow $l+m>n$.

Limit of the value of A is

$$2^{l-1} \leq A \leq 2^{l}-1 \qquad (1)$$

Similarly limit of B is

$$2^{m-1} \leq A \leq 2^{m}-1 \qquad (2)$$

Limit of the value of $M=A\times B$ is

$$2^{l+m-2} \leq A \leq 2^{l+m}-(2^{l}+2^{m})+1 \qquad (3)$$

from the left hand side of the equation if $2^{n} \leq 2^{l+m-2}$ then overflow occurs.

$$\text{i.e., if } l+m \geq n+2 \qquad (4)$$

overflow occurs, overflow does not occur if $l+m<n+2$ (5)

From the right hand side of the equation (iii) there will be no overflow if

$$2^{l+m}-(2^{l}+2^{m})+1 \leq 2^{n}-1 \qquad (6)$$

but any positive integer $-(2^{l}+2^{m}) \leq -2$ or $(2^{m}+2^{m})+1 \leq -1$ (7)

combining equation(6) and (7) we get $2^{l+m} \leq 2^{n}$

$$\text{or } l+m \leq n \qquad (8)$$

from equations (v) and (viii) we can conclude that no overflow occurs if $l+m \leq n$ or if $n \geq l+m$

6.2 Detection of Overflow:

From the equation (4) it is obvious that if $l+m \geq n+2$ then overflow must happen. This condition is satisfied with the equation (24-26) given below.

$$O_{v1}=a_{n-1}.b_1+(a_{n-1}+a_{n-2}).b_2+(a_{n-1}+a_{n-2}+a_{n-3}).b_3+\ldots\ldots+(a_{n-1}+a_{n-2}+\ldots+a_1).b_{n-1} \qquad (9)$$

For verification we take first term $a_{n-1}.b_1$, if its value is 1 i.e., $a_{n-1}.b_1=1$ then the total number of significant bits is at least n+2 as A has at least n significant bits and B has at least 2 significant bits. In the same way, if the second term $(a_{n-1}+a_{n-2}).b_2$ is 1 then A has at least (n-1) significant bits and B has at least 3 significant bits. Similarly, if the last term $(a_{n-1}+a_{n-2}+\ldots\ldots+a_1).b_{n-1}$ is 1, then A has at least 2 significant bits and B has at least n significant bits so the total number of significant bits will be (n+2). The equation (9) is written as

$O_{v1}=\sum_{i=1}^{n-1} \sum_{j=1}^{n-1} a_{n-j}.b_i$, where bit-dot-product and bit-summation indicate logical AND and OR respectively. Comparing equation (4) and (8), we get that when $l+m=n+1$ the multiplied value may overflow but the value is less than 2^{n+1}. In this situation the product bits p_0 through p_n are sufficient to indicate whether the product value will overflow or not. If m_n is equal to 1 then only overflow occurs. So we conclude that if $l+m \geq n+2$ or $m_n=1$ then overflow must occur.

But from equation (9), the essential number of AND and OR gates are respectively

$$1+2+3+\ldots+(n-1)=(n-1).n/2 \qquad (10)$$

$$\text{and } (n-2)+(1+2+3+4+\ldots+n-2)=\{(n-1).n/2\}-1 \qquad (11)$$

If the recursive (iterative) technique is applied then the number of AND or OR gates can be reduced.

6.3. Iterative technique:

The hardware reduction can be used here by using the iterative process. We take

$$r_{k+1}=r_k+o_{k+1}.b_k \text{ and } \quad o_{k+1} = o_k + a_{n-k} \quad \text{where } 2 \leq k \leq n-1.$$

Initially we take, $o_{k=2}=a_{n-1}$ and $r_{k=2}=a_{n-1}.b_1$

Then $o_3 = o_2 + a_{n-2} = a_{n-1} + a_{n-2}$ $\quad r_3=r_2+o_3.b_2 = a_{n-1}.b_1+(a_{n-1}+ a_{n-2}).b_2$

$$o_4 = a_{n-1}+ a_{n-2} + a_{n-3} \qquad r_4= r_3+o_4.b_3$$
$$= a_{n-1}.b_1+(a_{n-1}+ a_{n-2}).b_2+(a_{n-1}+ a_{n-2} + a_{n-3}).b_3$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$r_n=O_{vl}=a_{n-1}.b_1+(a_{n-1}+a_{n-2}).b_2+(a_{n-1}+a_{n-2}+a_{n-3}).b_3+\ldots+(a_{n-1}+a_{n-2}+\ldots+a_1).b_{n-1} \tag{12}$$

After (n-1) iterations we obtain the result of the equation (12) which indicates the overflow. if the significant bits of the product M is $\geq n+1$ then also overflow occurs As overflow when $P>2^n$. So we can draw the block diagram for overflow using multiplier and iterative circuit like as

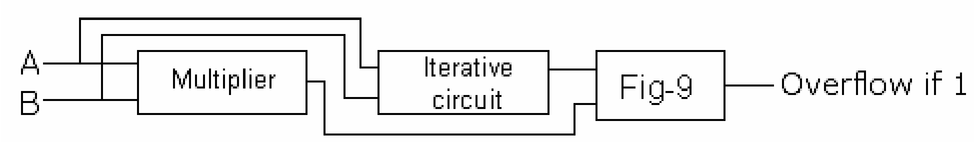


Figure 11. Block diagram for overflow

6.4. Highest order of input AND and OR gates for r_n *(Big oh of r_n) :*

From the equation (12), number of AND gates and OR gate are (n-1).n /2 and {(n-1).n /2}-1 respectively. Now, $(n^2/2-n/2) \leq n^2/2$ when $n \geq 2$

$$\leq n^2 \quad \text{when } n \geq 2$$

So big oh of the number of AND and OR gates are $O(n^2)$ and $O(n^2)$ respectively.

6.5. Implementation of multiplier:

From Theorem-1 and from the iteration equation (12), We can implement an unsigned multiplier and a overflow circuit. First eight bit unsigned digital multiplier has been implemented in Figure 12.

6.6. Reduction of delay:

The unsigned digital overflow can be reduced by using the prefix algorithm (Gok *et al*, 2000). The equation (12) are rewritten as

$$O_{vl}=A_{n-1}.b1+A_{n-2}.b_2+A_{n-3}.b_3+\ldots\ldots+A_1.b_{n-1} \tag{13}$$

$$\text{where } \sum_{k=1}^{n-1} a_k \tag{14}$$

6.7. Serial Prefix:

For a size n, assume that the inputs are $d_1, d_2, d_3, d_4, \ldots$, dn, and the related operation used is "*". After applying "*" each output y_i is written as

$$y_i = d_1 * d_2 * d_3 * d_4 * \ldots * d_{i-1} * d_i \tag{15}$$

where $1 \leq i \leq n$ **.**

Equation (15) can be utilized for sequentially computing y_i by using (n-1) times of "*" operations. Then this process is called serial prefix computation.

6.8. Parallel prefix:

From equation (15), it is clear that we can apply the operation "∗"in any order. For example, we can determine ($d_1 * d_2$), ($d_3 * d_4$),……or ,($d_{i-1} * d_i$) in parallel. This technique is called parallel-prefix (Cha *et al,* 2000; Ladver *et al,* 1980) technique. For reduction of time delay parallel prefix technique can be applied.

$A_k = \sum_{j=1}^{k} a_j$, $1 \le k \le n-1$, can be implemented using parallel prefix technique for an n = 8bit using the operator "OR". Parallel prefix technique is given in Figure 12 and Figure 13.

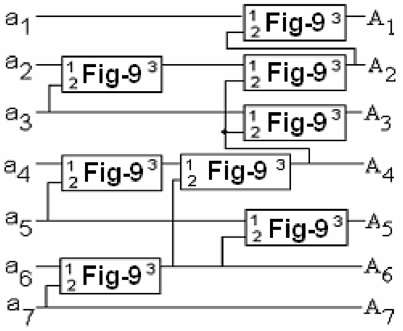


Figure 12. Parallel prefixcircuit for $A_k = \Sigma a_j$

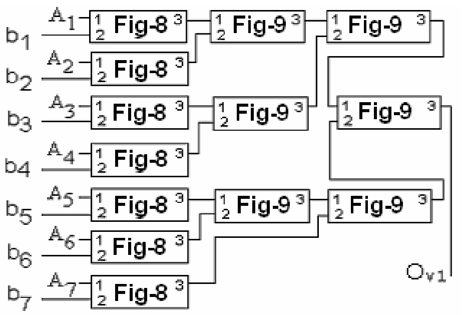


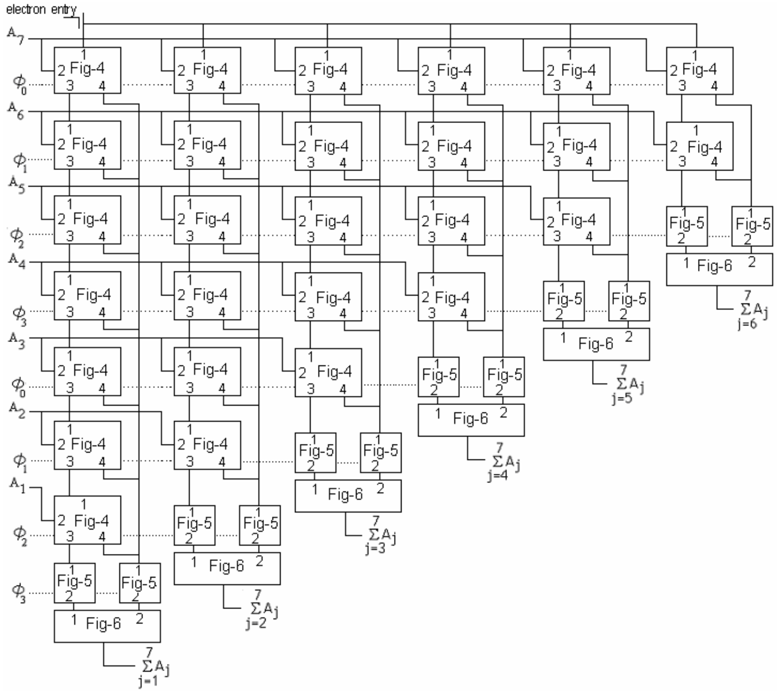
Figure 13. Parallel Prefix technique for overflow (O_{v1})determination.

6.9. Delay Optimization:

If we analyze the parallel prefix based implemented circuit, it is possible to reduce the node-number and the processing delay time. After analysis, the Figure 12 and Figure 13 are implemented another way so that the processing delays are minimized/optimized. The optimized figures are given in Figure 14 and Figure 15 respectively. The processing delay based on SED and analyzed SED for parallel prefix circuit is shown in Table 1.

Table 1. Processing delay based on SED and analyzed SED for parallel prefix circuit

Figure No.	No. of Nodes required	Delay (ns)	Node Combination	Total delay time(ns)
Figure-12	32	12	84	28
Figure-13	52	16		
Figure-14	39	8	55	16
Figure-15	16	8		



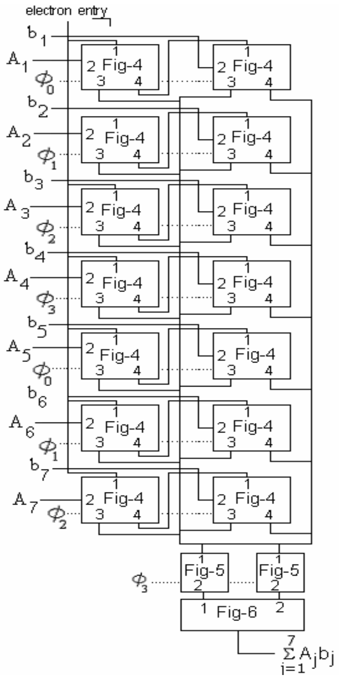
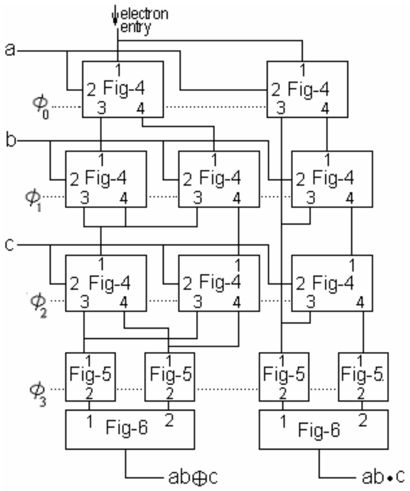


Figure 14. Analyzed Parallel prefix circuit of Figure 12

Figure 15. Analyzed Parallel prefix of Figure 13



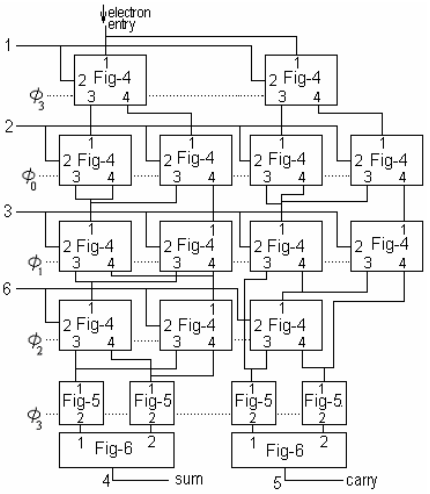


Figure 16. Modified Half Adder

Figure 17. Modified Full Adder

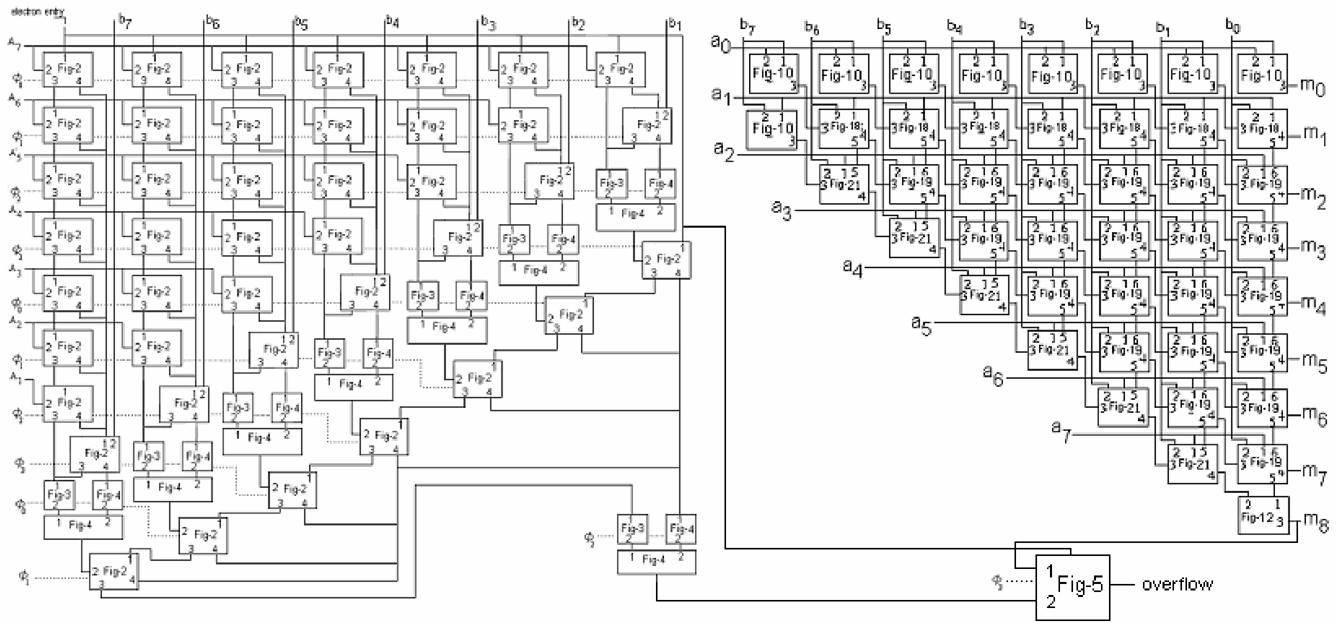


Figure 18. Multiplication and Overflow for unsigned numbers

7. Multiplier of sign numbers

To multiply the sign numbers we have to find the two's complements of the sign numbers. Next step is to determine the number of significant bits of the two's complement number(s) as according to the Theorem 2 significant numbers are essential for indicating the overflow condition. The significant bit numbers of a two's complement number is determined as:

The number of bits for first different rightmost bit with the sign bit of A and the bits right to that different rightmost bit is called significant bit numbers. Suppose A=00010010 and B=11111000, the number of significant bit of A and B are in Table 2.

Table 2. Significant bit numbers computation

number	Sign bit	1st unlike bit position from LSB	Significant bit number
A	0	5th [10010]	5
B	1	3rd [000]	3

7.1. Theorem-2: If the significant bits of two's complement numbers of A and B are l and m then the overflow of the product (multiplication) of these numbers will happen if l+m<n-1, (n=number of significant bits)

Say, A and B have significant numbers l and m respectively then A and B are bounded by

$$2^{l-1} \leq |A| < 2^{l} \tag{16}$$

$$2^{m-1} \leq |B| < 2^{m} \tag{17}$$

and the product is also bounded by

$$2^{l+m-2} \leq M = |\ A \times B\ | < 2^{l+m} \tag{18}$$

We know that the overflow occurs if $M \geq 2^{n-1}$ (19)

when M is positive

or, $M < -2^{n-1}$ (20)

when M is negative

(As overflow happens when $M \geq 2^{n-1}$ or $M < -2^{n-1}$)

Comparing equations (16) and (17), the left hand limit implies that overflow occurs if

$$2^{l+m-2} \geq 2^{n-1}$$

$$\text{or } l+m-2 \geq n-1$$

$$\text{or } l+m \geq n+1 \tag{21}$$

So, overflow will not happen if $l+m \leq n$ (22)

For the right hand limit from equation (16), overflow will not hold if

$$2^{l+m} < 2^{n-1}$$

$$\text{or } l+m < n-1 \tag{23}$$

Comparing equation (16) and (22) overflow will not occur if $l+m< n-1$.

7.2. Condition-a: when M is positive

From the Theorem 2 we observe that (i) overflow happen if $l+m \geq n+1$ and
(ii) overflow will not happen if $l+m< n-1$

But what happens when $l+m=n$ or $n-1$?

We investigate these two situations step by step if $l+m=n-1$, we get from equation (17)

$$2^{l+m-2} \leq M < 2^{l+m}$$

$$\text{or } \ 2^{n-3} \leq M < 2^{n-1} \tag{24}$$

It is known that for overflow condition $M \geq 2^{n-1}$ (25)

So overflow may happen but it does not exceed the limit 2^{n-1}. When $l+m=n$, then $2^{n-2} \leq M < 2^n$ so overflow may occur but it does not exceed the limit 2^n.

7.3. Condition -b: when M is negative

When $l+m=n$ or $n-1$ and the sign of M is negative or positive

i) If both A and B are positive and $M< 2^n$ then it implies that $m_n=0$ always (as $M \geq 2^{n-1}$ for overflow) and overflow holds only if $m_{n-1}=1$

ii) If A and B both are negatives then $M \leq 2^n$ indicates that $p_n=0$ all times and $p_{n-1}=1$only when overflow occurs.

iii) When A and B has different signs, then $-2^{n-1}<M$ means $p_n=1$ always and $p_{n-1}=0$ only when overflow occurs.

Clearly from the above three situations it is clear that

$$\text{or, } \ m_n \oplus m_{n-1}=1 \tag{26}$$

7.4. Verification of Output (O/P) of the Figure 19

We are to show that the output value of the circuit and the logic value of the logic function F= ab⊕ C ⊕ D are the same. The result will be 0 or 1 if the messenger electron reaches at the $\boxed{0}$ or $\boxed{1}$ valued node. For example, when [a b C D]=0110 then the electron follows the "0" branch between the nodes (i) and (ii), 0 or 1 branch between the nodes (ii) and (iii), 1-branch between then nodes (iii) and (iv) and 0-branch between the node (iv) and terminal $\boxed{1}$ and the messenger reaches the node $\boxed{1}$ i.e., the O/P will be 1. From the Table- 3 the other binary words can be verified.

Table 3. Binary word verification

a	b	C	D	O/P	ab	ab	F	remark
0	0	0	0	0	0		0	Value of O/P and F are same
0	1	0	0	1	0	0	1	
1	0	0	1	1	0		1	
		0	1	0			0	
		1	0	0			0	
1	1	1	0	1	1	1	1	
		1	1	0			0	
		1	1	1			1	

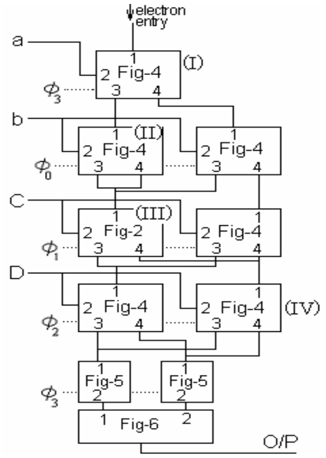


Figure 19. Overflow detection circuit

8. Comparison of time delay

The parallel prefix circuit given in Figure 14 represents the maximum processing time delay is 4×4=16ns whereas if the circuit is analyzed we shall be able to implement the circuit whose processing delay time must be lesser than the previous one i.e., the processing time can be optimized The same circuit has been implemented another way (Figure 16) which is a optimized form no doubt. In Figure-16, the processing delay time is 8ns, Similarly, the parallel circuit (Figure 15) consumes 16ns and its optimized form, Figure 17, requires 8ns. If these two parallel circuits are implemented by using CMOS/TTL gates then the minimum delay / processing time would have 48ns and 48 ns respectively. If the parallel prefix technique is not used then for ANDing of 7 bits will be for

(i) CMOS/TTL logic 12×6=72ns (Milman, 2000)
(ii) SED based 4×6=24ns
(iii) Analyzed based = 8ns.

In Table 4, time delays and fastness for CMOS/TTL gates, SED based circuits and analyzed based circuits of serial (sequential) and parallel prefix computations are given.

Table 4. Comparison of time delays and fastness for CMOS/TTL gates, SED based circuits

Sl. No.	Circuit Name	Serial prefix Delay time (ns)	Faster (times) with respect to CMOS/TTL	Parallel prefix Delay time (ns)	Faster (times) with respect to CMOS/TTL
1	CMOS/TTL gates (Milman 2000)	72	1	48	1
2	SED gates (Asahi 1995; Asahi 1998)	24	3	16	3
3	Analyzed SED gates	8	9	8	6

9. Overflow completion

Combining the two conditions (i) n significant bits of the A and B, and (ii) the value of ($m_n \oplus m_{n-1}$), the resultant overflow is obtained for unsigned multiplication. The logical equation of the resultant overflow is:

$$\text{Overflow}=O_{v2} + (m_n \oplus m_{n-1}) \tag{27}$$

where O_{v2} is given in equation (28).For signed multiplication overflow must occur if $l+m \geq n+1$ [from equation (21)]. To satisfy this condition we can write below a logic equation [20-22], the value of which will be "1"when overflow happens,

$$O_{v2}= \{\tilde{b}_{n-2} \cdot \tilde{a}_1+ (\tilde{b}_{n-2}+ \tilde{b}_{n-3}) \cdot \tilde{a}_2+(\tilde{b}_{n-2}+ \tilde{b}_{n-3}+\tilde{b}_{n-4}) \cdot \tilde{a}_3+ \cdots+(\tilde{b}_{n-2}+ \tilde{b}_{n-3}+\cdots+\tilde{b}_1) \cdot \tilde{a}_1\} \tag{28}$$

that is O_{v2}= 1 for overflow, where $\tilde{a}_j = a_j \oplus a_{n-1}$ and $\tilde{b}_j = bj \oplus b_{n-1}$.

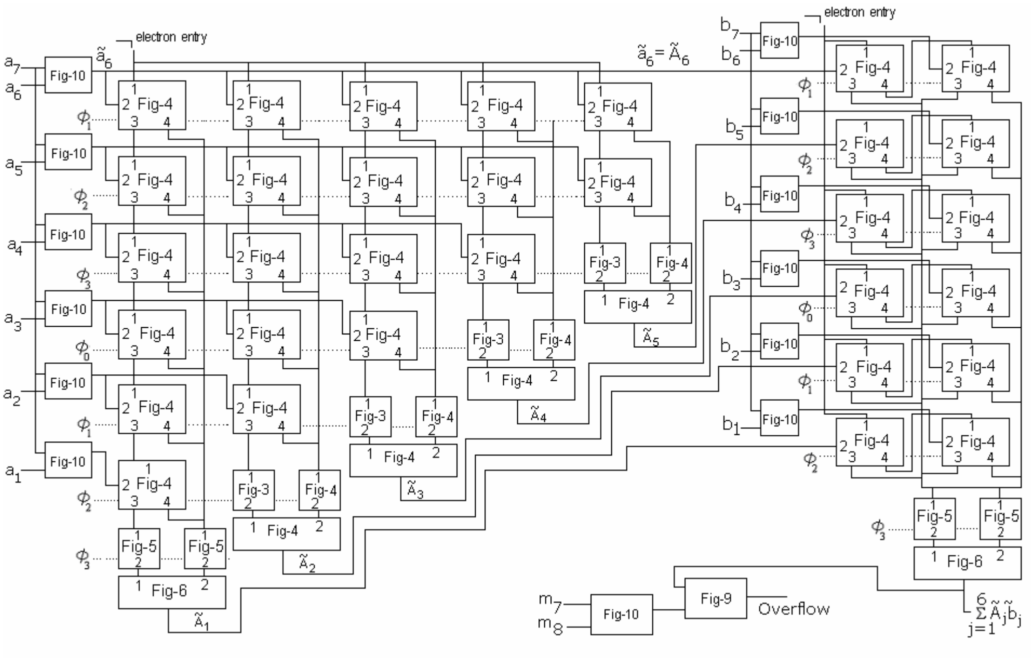


Figure 20. Overflow detection of sign-multiplier

10. Combination of signed and unsigned multiplication and overflow

We are able to combine the signed and unsigned multiplication with their overflows in a single multiplier for the similarities of their designed circuits without loss of the generality. To do this, an external control line is connected with the circuit. The input value of the control lone data (d) will be 0 or 1. We choose the general data input equation of a_j and b_j as

$$\tilde{a}_j = a_j \oplus (a_{n-1} \bullet d) \tag{29}$$

$$\tilde{b}_j = b_j \oplus (b_{n-1} \bullet d) \tag{30}$$

where $1 \leq j \leq n-1$.

When d=0 the equations (29) and (30) give a_j and b_j respectively and when d=1, give $\tilde{a}_j$ and $\tilde{b}_j$ avoiding $\tilde{a}_j = a_{n-1} \oplus a_{n-1}$ and $\tilde{b}_j = b_{n-1} \oplus b_{n-1}$ in the last case. Clearly, for d=0, the circuit will contribute the unsigned product and for d=1, the circuit will contributes the signed product.

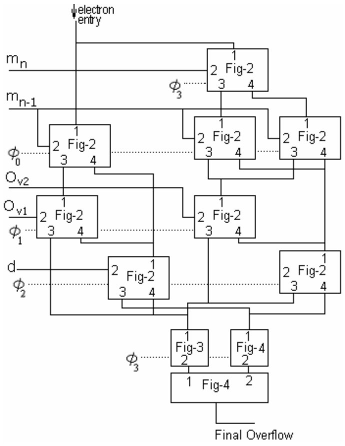


Figure 21. Final overflow circuit

In accordance with the Theorem 1 and Theorem-2 and with the derived equations of (9) and (27), we can create the multiplied value(s) and general cased overflow. Finally, the overflow is calculated from the overflow equation given as:

$$O_{flow}= (O_{v1}+m_n) \bullet \overline{d} + (O_{v2}+m_n \oplus m_{n-1}) \bullet d \qquad (31)$$

where "+", "⊕" and "•" indicates OR, X-OR and AND operations. The related Final overflow detection circuit is given in Figure 23. Where m_n and m_{n-1} are taken from the multiplication circuit in Figure 20 and O_{v1} and O_{v2} are taken from the overflow parts of the circuits of Figure 18 and Figure 20 respectively.

11. Results and discussions

If both the operands are of same sign (i.e., either positive or negative) the result of the multiplication will be of positive sign. In such situation the Figure 19 is used for overflow detection. On the other hand, if the sign bit of the two operands are opposite signs i.e., the XOR of these two sign bit is 1 then Figure 20 is used for multiplication and overflow detection. These two signed and unsigned multiplication circuits are brought to a same platform for unifying them after a small modification of the input data. The Figure 21 delivers the final overflow indication. The overflow detection circuit does not depend upon the internal carries which is generated by the multiplier and does not require the partial product of the multiplier to be modified. So the circuit can be applied to any type of simplified multiplier which produces (n+1) product bits. The parallel and analyzed circuits have less area and fewer nodes than the serial prefix circuits. The designed multiplier circuit can be used efficiently for both signed and unsigned operands. The table 3 manifests the speed as well as the delay time is given for comparing the CMOS/TTL logic gates based circuit(s) and single electron based circuit. Obviously, Single electron device based circuit is at least 3 times faster than the classical logic based circuits.

12. Conclusion

A general technique for multiplication along with the overflow based on single electron tunneling phenomena has been implemented for positive (unsigned) and negative (signed) integer multiplication. This technique includes separately serial and parallel prefix algorithm. Parallel prefix algorithm is used for time saving and decreasing the gates or nodes. Analyzed circuit also optimizes the processing delay in comparison to the classical implemented techniques. Thermal agitation and stochastic errors caused by probabilistic fluctuations are excluded in our design.

Acknowledgment

Subir Kumar Sarkar thankfully acknowledges the financial support obtained from UGC vide F. No.36-100/2008(SR)

References

Asahi N., Akazawa, M. and Amemiya, Y. 1995. Binary decision diagram device. *IEEE Trans. ED.* 42 11, pp. 1999-2003.

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.

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.

Biswas A.K. Sarkar, S.K. 2002. Single electron logic circuits for the realization of a logic unit of a computer, *IJICS*, Vol.5, No.1, pp.53-58.

Biswas A.K., Sarkar S.K. 2003. Error detection and Debugging on Information in Communication system using single electron circuit based binary decision diagram. *Semiconductor Physics Quantum Electronics and opt electronics*. Vol. 6, pp. 1-8.

Biswas A.K. and Sarkar S.K. 2003. An Arithmetic logic unit of a computer based on single electron transport system. *Semiconductor Physics Quantum Electronics and opt electronics.* Vol. 6, No.1 pp.91-96.

Cha Y.H., Cho G.Y., Choi H.H. Song H.B., 2001. Bit result integer multiplier with overflow detector, *IEE Electronic Letters*. Vol. 37. pp. 940-942.

Gok M., Schulte M.J., Balzola P.I., and Brocato R.W., 2000. Combined unsigned and two's complement saturating multipliers, *Proc. SPIE: Advanced Signal Processing Algorithms, Architectures and Implementations.* pp.185-196.

IBM, 2000. Power PC Microprocessor Family. The Programming Environments for 32-bit Microprocessors, no. G522-0290-01.

Korotkov A.N., Likharev K.K. 1998. Single electron-parameter-based logic devices. *J. Appl. Phys*. Vol. 84, pp. 6114-6126.

Lindholm T. Yelin, F. 1996. *Java Virtual Machine Specification*. Addison-Wesley, Reading, MA.

Ladver R.E and Fiscer M.J. 1980. Parallel prefix computation. *J. ACM*, Vol. 27, pp. 831-938.

Milman J., Halkias C.C. 2000. *Integrated Electronics*, Edn. McGraw Hill.

Schulte M.J., Balzola P.I., Akkas A., Brocato R.W., 2000. Integer multiplication with overflow detection or saturation, *IEEE Trans. Computers,* Vol.49, No. 7, pp. 681-691.

Biographical notes

Mr. Souvik Sarkar is a Research Scholar in the Department Electronics and telecommunication Engineering, Jadavpur University. His areas of interest include nano, single electron and spintronic device based circuit modeling.

Dr. Anup Kumar Biswas is Lecturer in Commuter Science and Engineering in Jalpaiguri Govt. Engineering College (India). He has engaged in teaching and research activities since the last 10 years. His field of specialization is Single Electron Device. Dr. Biswas has published several papers in various national, international conferences and journals.

Mr. Ankush Ghosh is a Research Scholar in the Department Electronics and telecommunication Engineering, Jadavpur University under Prof. Subir Kumar Sarkar. His areas of interest include nano, single electron and spintronic device based circuit modeling.

Prof. Subir Kumar Sarkar is a Professor of Department of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata -700 032. He has published three Engineering text books (from CRC, USA) and more than 260 technical research papers in archival journals and peer reviewed conferences. Thirteen students have been awarded PhD (Engg) degree under his guidance and twelve more are presently pursuing PhD under his guidance. As Principal Investigator he has successfully completed four R&D projects and two more are running. He has visited several countries like France, UK, Switzerland and Japan for Technical reasons like training, presenting papers or visiting sophisticated laboratories. His most recent research focus is in the areas of simulations of nanodevice models, transport phenomenon, single electron & spintronics devices and their applications in VLSI circuits, low power VLSI design, ad hoc wireless networks, wireless mobile communication and watermarking. He is a life Fellow of the Institution of Engineers and Institution of Electronics and Telecommunication Engineers, life member of Indian Association for the cultivation of Science.

Received August 2009
Accepted September 2009
Final acceptance in revised form September 2009