On Computing Addition Related Arithmetic Operations via Controlled Transport of Charge

Sorin Cotofana, Casper Lageweg and Stamatis Vassiliadis
Computer Engineering Laboratory,
Delft University of Technology,
Delft, The Netherlands
{Sorin,Casper,Stamatis}@CE.ET.TUDelft.NL

Abstract

In this paper we investigate the implementation of basic arithmetic functions, such as addition and multiplication, in Single Electron Tunneling (SET) technology. First, we describe the SET equivalents of Boolean CMOS gates and Threshold logic gates. Second, we propose a set of building blocks, which can be utilized for a novel design style, namely arithmetic operations performed by direct manipulation of the location of individual electrons within the system. Using this new set of building blocks, we propose several novel approaches for computing addition related arithmetic operations via the controlled transport of charge (individual electrons). In particular, we prove the following: n-bit addition can be implemented with a depth-2 network built with $O(n)$ circuit elements; n-input parity can be computed with a depth-2 network constructed with $O(n)$ circuit elements and the same applies for $n|\log n$ counters; multiple operand addition of m n-bit operands can be implemented with a depth-2 network using $O(mn)$ circuit elements; and finally n-bit multiplication can be implemented with a depth-3 network built with $O(n)$ circuit elements.

1 Introduction

Feature size reduction in microelectronic circuits has been an important contributing factor to the dramatic increase in the processing power of computer arithmetic circuits. However, it is generally accepted that sooner or later MOS based circuits cannot be reduced further in (feature) size due to fundamental physical restrictions [15]. Therefore, several emerging technologies are currently being investigated [13]. Single Electron Tunneling (SET) [8] is one such technology candidate and offers greater scaling potential than MOS as well as ultra-low power consumption. Additionally, recent advances in silicon based fabrication technology (see for example [14]) show potential for room temperature operation. However SET devices display a switching behavior that differs from traditional MOS devices. This provides new possibilities and challenges for implementing digital circuits.

SET technology introduces the quantum tunnel junction as a new circuit element for (logic) circuits. The tunnel junction can be thought of as a "leaky" capacitor, such that the "leaking" can be controlled by the voltage across the tunnel junction. Although this behavior at first glance appears similar to that of a diode, the difference stands in the scale at which switching occurs. Charge transport though a tunnel junction can only occur in quantities of a single electron at a time. Additionally, given the feature sizes anticipated for such circuits, the transport of a single electron can have a significant effect on the voltage across a tunnel junction. This implies that transporting a few electrons through a tunnel junction will inhibit further charge transport, making it possible to control the transport of charge in discrete and accurate quantities.

The ability to control the transport of individual electrons in SET technology introduces a broad range of new possibilities and challenges for implementing computer arithmetic circuits. In this paper we investigate the computation of addition related arithmetic operations in SET technology by controlling the transport of individual electrons. First, we briefly present the SET equivalent of two conventional design styles, namely the equivalents of CMOS and threshold logic gates. Second, we propose a set of building blocks, which can be utilized for charge controlled computations. Third, using the new set of building blocks, we propose several novel approaches for computing addition related arithmetic functions, e.g., addition, parity, counting, multiplication, via the controlled transport of charge. Related to these new schemes we prove that the following holds true:

- The addition/subtraction of two n-bit operands can be computed with a depth-2 network composed out of

$3n + 1$ circuit elements[1].

- The n-parity function can be computed with a depth-2 network constructed with $n + 1$ circuit elements.
- The $n|\log n$ counter can be implemented with a depth-2 network constructed with $n + \log n$ circuit elements.
- The m n-bit multiple operand addition can be implemented using at most $n(m + 1) + [\log m] + 1$ circuit elements.
- The multiplication of two n-bit operands can be computed with a depth-3 network with $4n - 1$ circuit elements.

The remainder of this paper is organized as follows: Section 2 briefly presents some SET background theory, explaining the basic switching behavior appearing in SET circuits. Section 3 presents the SET equivalent of the CMOS design style. Section 4 presents the implementation of threshold gates in SET technology. In Section 5 we propose a set of new building blocks for controlled charge transport. Section 6 proposes new schemes for the calculation of addition and multiplication via the controlled transport of single electrons. Finally, Section 7 concludes the paper with some final remarks.

2 Background

Single Electron Tunneling technology introduces the quantum tunnel junction as a new circuit element. A tunnel junction consist of two conductors separated by an extremely thin insulating layer. The insulating layer acts as an energy barrier which inhibits charge transport under normal (classical) physics laws. However, according to quantum physics theory, charge transport of individual electrons through this insulating layer can occur if this results in a reduction of the total energy present in the circuit. The transport of charge through a tunnel junction is referred to as *tunneling*, while the transport of a single electron is referred to as a *tunnel event*. Electrons are considered to tunnel through a tunnel junction strictly one after another.

Rather then calculating for each tunnel junction if a hypothetical tunnel event results in a reduction of the circuit's energy, we can calculate the critical voltage V_c, which is the voltage threshold needed across the tunnel junction to make a tunnel event through this tunnel junction possible. For calculating the critical voltage of a junction, we assume a tunnel junction with a capacitance of C_j. The remainder of the circuit, as viewed from the tunnel junction's perspective, has an equivalent capacitance of C_e. Given the approach presented in [5], we calculate V_c for the junction as

$$V_c = \frac{e}{2(C_e + C_j)}. \quad (1)$$

In the equation above, as well as in the remainder of this discussion, we refer to the charge of the electron as $e = 1.602 \cdot 10^{-19}\ C$. Strictly speaking this is incorrect, as the charge of the electron is of course negative. However, it is more intuitive to consider the electron as a positive constant for the formulas which determine if a tunnel event will take place or not. We will of course correct for this when we discuss the direction in which the tunnel event takes place. Generally speaking, if we define the voltage across a junction as V_j, a tunnel event will occur through this tunnel junction if and only if $|V_j| \geq V_c$. If tunnel events cannot occur in any of the circuit's tunnel junctions, i.e., $|V_j| < V_c$ for all junctions in the circuit, the circuit is in a *stable state*. For our research we focus on circuits where a limited number of tunnel events may occur, resulting in a stable state. Each stable state determines a new output value resulting from the distribution of charge throughout the circuit.

Assuming that a tunnel event is possible, the orthodox theory for single electron tunneling (see for example [5] for a more extensive introduction) states that tunneling is a stochastic process, in which the rate at which tunnel events occur at $0K$ temperature is

$$\Gamma = \frac{|V_j - V_c|}{eR_t} \quad (2)$$

where R_t is the tunnel resistance (usually $\approx 10^5\Omega$). Note that a non-$0K$ temperature implies a lower event rate. Assuming that an individual tunnel event can be described as a Poisson process, we can calculate the required delay t for a single tunnel event to occur for a given error chance P_{err} as:

$$t = \frac{ln(P_{err})eR_t}{|V_j - V_c|}. \quad (3)$$

Given that the minimum amount of transportable charge consists of a single electron, there exists a minimum energy threshold, called the Coulomb energy, which must be present in the circuit so that the transport of a single electron reduces the total amount of energy in the system. Resulting, in order to utilize the electron tunneling phenomenon, all other types of energy must be much smaller then the Coulomb energy. For thermal energy, this implies that, if we intend to add or remove charge to a circuit node by means of tunnel events, the total capacitance attached to such circuit nodes must be less then $900aF$ for $1K$ temperature operation, or less then $3aF$ for $300K$ (room temperature) operation [10]. This represents a major SET fabrication technology hurdle as even for cryostat temperature

[1] By circuit element we mean in this context any of the building block presented in Section 5.

operation very small circuit features are required to implement such small capacitors. Another major technology challenge comes from the fact that thus far all experimental circuits have displayed a random offset charge (random charge present on circuit nodes), which is assumed to be the result of trapped charge particles in the tunnel junctions themselves or in the substrate. This random charge results in a random additional voltage across tunnel junctions, which can cause errors in their switching behavior. At the same time there are indications [8] that the offset charge problem may reduce or even disappear entire for the nanometer-scale feature size circuits required for room temperature operations. Given this and the fact that in our investigation we focus on the efficient utilization of the SET behavioral properties we ignore the aspects related to offset charge and its potential influence on SET based computational structures.

3 CMOS Like SET Gates

One of the first SET circuits examined in literature is the SET transistor (see [7] for an early review paper). The SET transistor consists of two tunnel junctions in series, with a capacitor attached to the interlaying circuit node, as is displayed in Figure 1. The resulting 3-terminal structure can be seen as being similar to a MOS transistor, such that the gate voltage V_g can control the transport of charge through the tunnel junctions (current I_d).

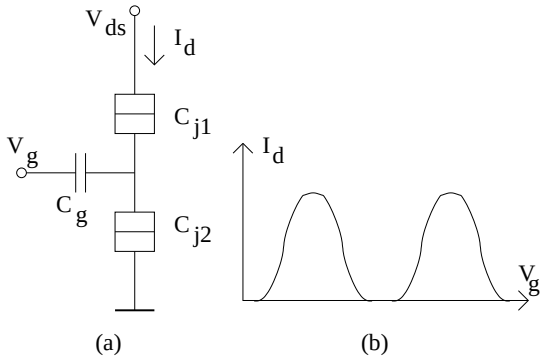

Figure 1. The SET transistor (a) circuit and (b) transfer function.

However, unlike the MOS transistor, the current I_d through the SET transistor has a periodic response to the input voltage V_g. By adding a capacitively coupled bias voltage to the interlaying node, the transfer function of the SET transistor can be translated over the V_g axis.

When two SET transistors with different biasing voltages are combined in a single circuit, we arrive at the CMOS-type inverter structure proposed in [6], as displayed in Figure 2. In the Figure, the upper and lower SET transistor behave as a p-type and an n-type MOS transistor, respectively. Additionally it was suggested in [6] that using p-type and n-type SET transistors as a basis, one can now convert existing CMOS cell libraries to SET technology.

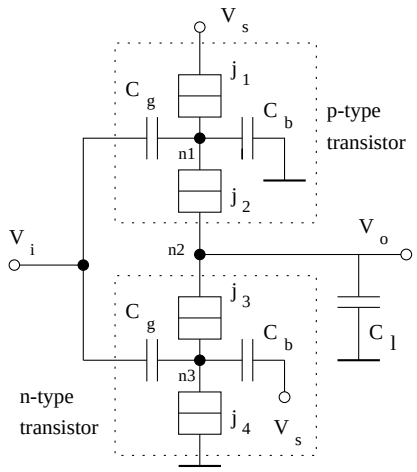

Figure 2. CMOS-type SET inverter.

The main disadvantage of the approach described above is that the current transport though an "open" transistor still consists of a large number of individual electrons "dripping" through the tunnel junctions. This is obviously a far slower process then the transport of just one single electron through the same junction, and consequently does not use the SET technology to its full potential. Therefore, the next step would be to limit the charge transport through an open transistor to just 1 electron. This results in the principle of Single Electron Encoded Logic (SEEL), in which the Boolean logic values 0 and 1 are encoded as a net charge of 0 and $1e$ on the circuit's output node.

However, when the SEEL approach is applied to converted CMOS cells with multiple p-type or n-type transistors in series, the circuits will no longer operate correctly, as clarified by the following example. Assume a series of 2 p-type transistors, of which the one bordering the load capacitor is open while the other one is closed. This situation will result in the removal of 1 electron from the load capacitor, resulting in an incorrect "high" output. Only the inverter circuit itself will operate correctly under a single electron encoded logic regime. This implies that CMOS type SET logic must encode the Boolean logic values 0 and 1 as "few" and "many" electron charges. We can therefore conclude that CMOS-type SET logic cannot efficiently utilize the SET features. To circumvent this problem we introduce in the next section SET based threshold logic gates, which can operate according to the SEEL paradigm.

4 Threshold Logic Gates

Threshold logic gates are devices able to compute any linearly separable Boolean function given by:

$$F(X) \quad = \quad sgn\{\mathcal{F}(X)\} = \begin{cases} 0 & \text{if } \mathcal{F}(X) < 0 \\ 1 & \text{if } \mathcal{F}(X) \geq 0 \end{cases} \quad (4)$$

where $\mathcal{F}(X) = \sum_{i=1}^{n} \omega_i x_i - \psi$, x_i are the n Boolean inputs, and w_i are the corresponding n integer weights. The linear threshold gate performs a comparison between the weighted sum of the inputs $\Sigma_{i=1}^{n} \omega_i x_i$ and the threshold value ψ. If the weighted sum of inputs is *greater than or equal to* the threshold, the gate produces a logic 1. Otherwise the output is a logic 0. Threshold logic gates are inherently more powerful then standard Boolean gates [12].

As stated in Section 2, a SET tunnel junction requires a minimum voltage $|V_j| \geq V_c$ in order for a tunnel event to occur. This critical voltage V_c acts as a naturally occurring threshold ψ with which the junction voltage V_j is compared. If we add capacitively coupled inputs to the circuit nodes on either side of the tunnel junction, the inputs will make a positively or negatively weighted contribution to the voltage across this junction (depending on the sign definition of V_j). Similarly, we can add a capacitively coupled biasing voltage in order to adjust the threshold to the desired value. This approach resulted in a generic threshold gate implementation [2] as displayed in Figure 3.

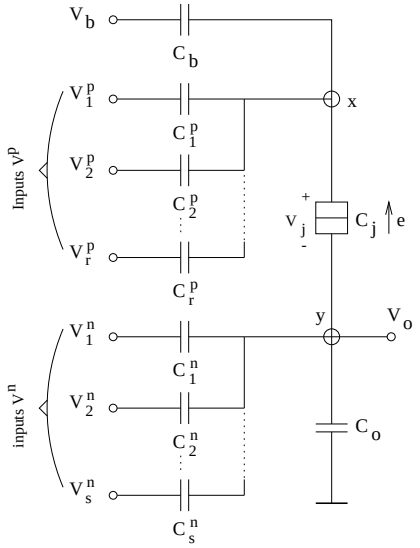

Figure 3. The n-input linear threshold gate.

In this figure, the input signals $V^p = \{V_1^p, V_2^p, \ldots, V_r^p\}$ are weighted by their corresponding capacitors $C^p = \{C_1^p, C_2^p, \ldots, C_r^p\}$ and added to the voltage across the tunnel junction. The input signals $V^n = \{V_1^n, V_2^n, \ldots, V_r^n\}$ are weighted by their corresponding capacitors $C^n = \{C_1^n, C_2^n, \ldots, C_r^n\}$ and subtracted from the voltage across the tunnel junction. The biasing voltage V_b, weighted by the capacitor C_b, is used to adjust the gate threshold to the desired value ψ. If $sgn\{V_j - V_c\}$, a single electron is transported from node y to node x, which results in a high output.

The generic threshold gate described above can be used to implement any threshold function (including the standard Boolean logic gates). However, due to the passive nature of this circuit we must apply sufficient buffering between different threshold gates in order to alleviate feedback effects as well as to maintain correct logic levels [3]. The CMOS style inverter described in Section 3 can act as a SEEL buffer [1]. A similar non-inverting buffer can also be derived from the inverter by removing both bias capacitors C_b and choosing a different set of capacitor values, resulting in an even smaller buffer.

Thus by utilizing the SET based threshold gate approach all the Boolean and/or Threshold logic schemes for the computation of arithmetic functions can be potentially implemented with no major change in the paradigm. Moreover by encoding Boolean values in a net charge of 0 or $1e$, the SET threshold logic makes efficient use of the SET technology and potentially provides the premises for ultra-low power consumption computations.

However, given that in SET technology it is possible to control the transport of individual electrons, we can further improve efficiency if we can encode n-bit operands as a number of electrons stored at a specific circuit location and perform arithmetic operations via the controlled transport of single electrons. Before exploring this novel concept further we propose in the next section a set of new building blocks to constitute the fundament for computing arithmetic operations via controlled transport of charge.

5 Building Blocks for Electron Counting

In this section we propose two basic blocks: one can be utilized to move electrons within a SET circuit and the other one can be utilized to implement periodic symmetric functions. The novel circuit blocks introduced in the sequel are then utilized in Section 6 as a basis for constructing larger structures for addition related arithmetic operations, e.g., addition, parity, counting, and multiplication.

5.1 The MVke Block

The $MVke$ block displayed in Figure 4 is a basic block with which a variable number of electrons can be added to or removed from a charge reservoir. Typically, a charge reservoir is a circuit node that is capacitively coupled to ground. A charge reservoir with a capacitance C_r containing a charge of $V \times e$ is therefore equivalent to a voltage source $U = \frac{V \times e}{C_r}$. The $MVke$ block operates as follows: if

the Boolean control signal $E = 1$, a charge of $V \times k \times e$ is moved to the electron reservoir when the block is triggered by a clock pulse (CLK), where k is a positive integer constant and V is an integer (variable) value. Note that V could either be another charge reservoir containing $V \times e$ electrons or an equivalent voltage source. For positive V values the $MVke$ block is in "add" mode (adding charge to the reservoir) while for negative V values the $MVke$ block is in "remove" mode (removing charge from the reservoir).

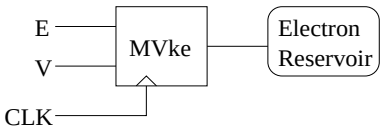

Figure 4. The MVke block.

Earlier experiments have demonstrated [4, 11] that the SET turnstile circuit, originally proposed by [9], can be modified such that it can control the transport of charge to and from a charge reservoir. However, the (modified) turnstile can only move one electron per clock pulse fact that precludes its direct utilization as an $MVke$ block. Given that the SET transistor operates as a controlled switch it can be used to extend the (modified) turnstile circuit capabilities such as it can let tunnel a larger amount of electrons per clock signal. A possible implementation of the $MVke$ block, based on the SET transistor and the operating principle of the turnstile circuit, is displayed in Figure 5.

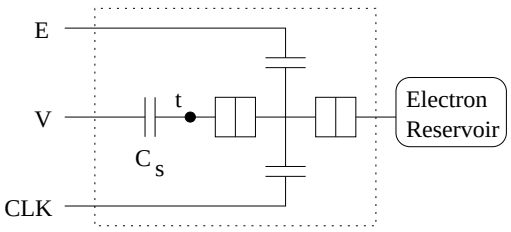

Figure 5. Possible $MVke$ block implementation.

The circuit operates as follows. If a clock pulse arrives, the SET transistor is opened if and only if $E = 1$. When the transistor opens, $V \times k \times e$ charge will be added to or removed from the electron reservoir due to charge pulling or pushing effect of V. As a result of this charge transport, an opposite charge $-V \times k \times e$ will be stored on node 't'. The voltage resulting from this charge will cancel the effect of voltage source V, inhibiting further charge transport. Given that the capacitor C_s acts as a weight factor for V, the desired constant value k can be adjusted by changing the value of C_s.

5.2 The PSF Block

A Boolean symmetric function $F_s(x_0, x_1, \ldots, x_{n-1})$ is a Boolean function for which the output depends on the sum of the inputs $X = \sum_{i=0}^{n-1} x_i$. A Periodic Symmetric Function (PSF) $F_p(X)$ is a symmetric function for which $F_p(X) = F_p(X + T)$, where T is the period. Any PSF can be completely characterized by T, the value of its period, and a,b, the values of X corresponding with the first positive transition and the first negative transition, as displayed in Figure 6. Efficient implementation of periodic symmetric functions is quite important as many functions involved in computer arithmetic computations, e.g., parity, belong to this class of functions.

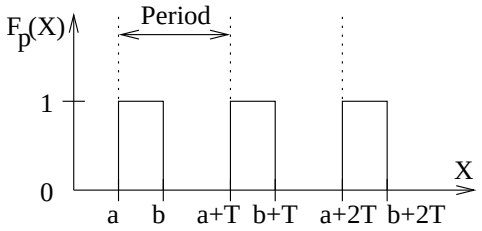

Figure 6. Period symmetric function $F_p(X)$.

Given the periodic transfer function of the SET transistor, as displayed in Figure 1, we can design a PSF block that can compute any PSF using a single SET transistor as a basis. The period of the SET transistor's transfer function can be adjusted to T by varying the value of the gate capacitor C_g. Likewise, the drain-source voltage V_{ds} determines the part of the function period in which $I_d > 0$, i.e., the length of the $[a, b]$ interval. Finally, a capacitively coupled bias voltage similar to that used for the CMOS-type inverter can translate the transfer function over the X axis in order to place the $[a, b]$ intervals in the required positions.

6 Electron Counting Based Arithmetic

In Sections 3, 4 we have demonstrated the feasibility of implementing classical Boolean and Threshold logic gates in SET technology. Although we can achieve the encoding of Boolean variables as a net charge of 0 and $1e$, this still does not use the full potential of SET. Given that we can control the transport of individual electrons, we have the possibility of encoding integer values X directly as a net charge Xe. Once integer values have been encoded as a number of electrons, we can perform arithmetic operations directly in electron charges. This reveals a broad range of novel computational schemes, which we will generally refer to as electron counting.

Within the context of this novel electron counting paradigm we investigate in the sequel SET networks for ad-

dition related arithmetic operations. We are mainly concerned in establishing the limits of such SET based circuit designs, thus we are interested in establishing theoretical bounds for delay and size, measured in terms of circuit elements[2], of the proposed implementations.

6.1 Addition & Subtraction

In this subsection we assume binary encoded n-bit operands, $A = (a_0, a_1, \ldots, a_{n-1})$ and $B = (b_0, b_1, \ldots, b_{n-1})$, and propose an electron counting scheme to compute the result of their addition/subtraction. The basic idea behind the method is first to convert the operands from digital to charge representation, add/subtract them in charge format, and convert the result back to binary digital representation.

Assuming binary operands, the first step in any electron counting process will be to convert a binary integer value X to its discrete analog equivalent Xe using a Digital to Analog Converter (DAC) which follows the general organization of the one introduced in [4]. As described in Section 5.1 the $MVke$ circuit (depicted in Figure 4) can be utilized to add/remove a number of electrons to/from a charge reservoir. When multiple such $MVke$ blocks operate in parallel on the same charge reservoir, electrons can be added to the reservoir in parallel. More specific, to convert an operand $X = (x_0, x_1, \ldots, x_{n-1})$, each bit x_i, $i = 0, 1, \ldots, n-1$ is connected to the E input of an $MVke$ block that has the V input hardwired to a bias potential that induces a $V \times k$ value equal with 2^i. Therefore, the operand X can be encoded as $\sum_{i=0}^{n-1} x_i 2^i e$ at the cost of n $MVke$ blocks in "add" mode. Thus this new DAC scheme has an $O(n)$ asymptotic complexity in terms of circuit elements.

Given the $MVke$-DAC encoding scheme described above, the addition and subtraction operations can be implemented in a straightforward manner. The addition of two n-bit operands A and B can be embedded in the conversion process if the operands are converted into charge format, via a total of $2n$ $MVke$ blocks in "add" mode that share a single charge reservoir. Similar, the subtraction operation, i.e., $A - B$, can also be embedded in the conversion process for the same cost. In this case, the $MVke$ blocks converting B operate in "remove" mode, encoding B as $-Be$, while still operating on the same electron reservoir.

Once the result corresponding to the addition/subtraction is available in the charge reservoir as a charge Ye, where $Y = A + B$ or $Y = A - B$, we need to convert this result back to a digital format in order to finalize the computation process. To achieve this an Analog to Digital Conversion (ADC) process is required. In the following we propose an ADC circuit that is taking advantage of the periodic transfer function of the SET transistor.

If N is the maximum number of extra electrons that can be present in the result electron reservoir, $m = 1 + [\log N]$ bits are required to represent this value in binary format. Then, following the base 2 counting rules, any ADC output bit s_i, $i = 0, 1, \ldots, [\log N]$ is equal to 1 inside an interval that includes 2^i consecutive integers, every 2^{i+1} integers, and 0 otherwise. Thus each bit s_i can be described by a periodic symmetric function with period 2^{i+1}. Then each output bit s_i can be computed by a PSF block that had been adjusted in order to have a transfer function that copies the periodic symmetric function required for the bit position i.

Thus we can implement an m-bit ADC using m PSF blocks (the PSF applied at bit position i is tuned to exhibits the periodic transfer function corresponding to that s_i bit) that operate in parallel on an electron reservoir. Given that we are addressing the particular case of n-bit operand addition, such that $m = n + 1$, then the cost of the required ADC circuit is in the order of $O(n)$.

Summarizing, the electron counting based addition/subtraction of two n-bit operands can be implemented with a depth-2 SET network built with $3n + 1$ circuit elements, therefor with an $O(n)$ asymptotic complexity measured in terms of circuit elements.

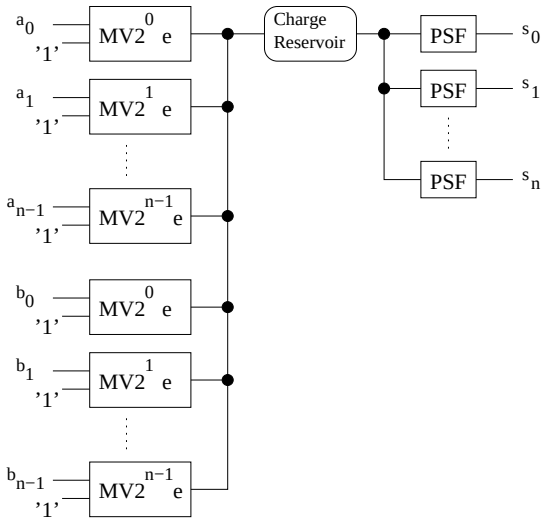

Figure 7. Organization of n-bit addition/subtraction circuit.

The overall organization of the circuit is depicted in Fig-

[2]By circuit element we mean in this context any of the building blocks presented in Section 5. We also assume that all the building blocks have the same cost, measured in terms of tunnel junctions and capacitors, and the same delay. More detailed computations can be also made in order to evaluate the proposed networks in terms of tunnel junctions and capacitors but such computations are beyond the scope of the paper. We preferred to use the generic concept of circuit element to keep the discussion implementation independent and to simplify the derivations.

ure 7. We note here that in the Figure, the value k of the $MVke$ blocks has been drawn inside the block to suggest that it was implemented by properly adjusting the C_S value, while all inputs V have been fixed to the equivalent of a charge reservoir with $1e$ charge.

Even though the proposed scheme was meant for addition/subtraction it has a broader scope and some of its alternative utilizations are discussed in the following:

- **n-bit parity function:** The scheme can be applied for $O(1)$ delay computation of the n-bit parity function as follows: The n inputs are connected via $MVke$ blocks (all the V inputs are hardwired to a bias potential that induces $V \times k = 1$) to a charge reservoir that provides input information to one PSF, molded to have the transfer function equal to 1 inside an interval that includes 2^0 consecutive integers, every 2^1 integers, and 0 otherwise, i.e., corresponding to parity. Thus the n-parity function can be computed with a depth-2 network constructed with $n+1$ circuit blocks. When compared to Boolean and Threshold logic based schemes (an AND-OR implementation of the n-input parity requires 2^{n-1} n-input AND gates and one 2^{m-1}-input OR gate) this is a substantial improvement.

- **$n|\log n$ counters:** To implement an $n|\log n$ counter we just have to augment the n-bit parity network with $\log n - 1$ additional PSF blocks. Thus the $n|\log n$ counter can be implemented with a depth-2 network constructed with $n + \log n$ circuit elements.

- **Multiple operand addition:** The addition scheme can be easily extended to support multiple operand addition by connecting more than 2 DAC circuits to the same electron reservoir and adjusting the ADC converter in order to be able to convert values up to $m \times (2^n - 1)$, where m is the number of operands and n their bit length. Thus the m n-bit multiple operand addition can be implemented using $m \times n$ $MVke$ blocks and at most $n + \lceil \log m \rceil + 1$ PSF blocks.

6.2 Multiplication

In this section we propose an electron counting multiplication scheme that follows to some extent the paradigm we introduced for addition. Assume we have the input operands $A = (a_0, a_1, \ldots, a_{n-1})$ and $B = (b_0, b_1, \ldots, b_{n-1})$ and we want to compute $P = A \times B$.

One way to do the multiplication is to utilize the multiple operand addition scheme presented at the end of the previous section. To utilize that scheme we have to calculate first all the partial products $a_i b_j$, $i = 0, 1, \ldots, n-1$, $j = 0, 1, \ldots, n-1$ with n^2 2-input AND gates. Subsequently, each row of partial products is connected to a DAC structure with the $MVke$ block input V hardwired to a potential that reflects the correct weight for the partial product it processes. This implies that n DAC circuits are now connected to the charge reservoir and that the ADC converter is adjusted in order to be able to convert values up to $n \times (2^n - 1)$. Thus the multiplication can be implemented with a depth-3 network constructed with n^2 2-input AND gates, n^2 Mke blocks, and $2n - 1$ PSF blocks. This implies that the overall asymptotic complexity of the multiplication circuit is in the order of $O(n^2)$ circuit elements.

In the sequel we introduce a different technique that makes use of the ability to transport a variable number of electrons to/from a charge reservoir exhibited by the $MVke$ structure discussed in Section 5.1 and depicted in Figure 4. Such a block can transport $V \times k$ electrons when k is a built-in constant (can be changed via a circuit parameter, i.e., C_S value) and V is a variable specified by the content of a charge reservoir.

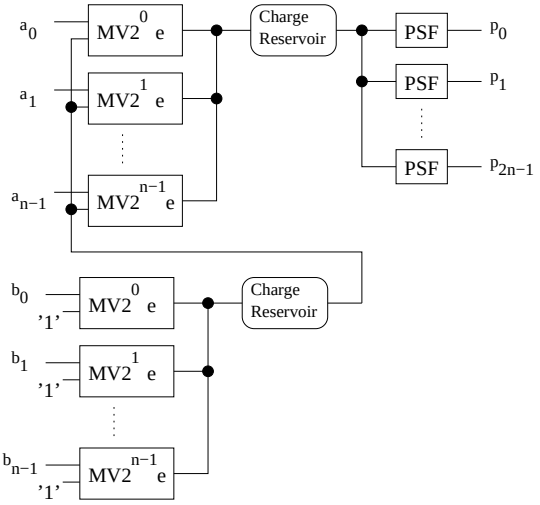

Figure 8. Organization of n-bit multiplication circuit.

The basic idea behind the scheme is again to add a charge Pe to a charge reservoir and to utilize an ADC structure to obtain the binary representation of the product P. The general organization of the proposed multiplication circuit is depicted in Figure 8. Again, the value k of the $MVke$ blocks has been drawn inside the blocks themselves to suggest that that k value was implemented inside the block by properly adjusting the C_S value. The scheme is utilizing a clock for synchronization purposes[3] and the computation process can be described as follows: First, on the positive clock value, a number of electrons corresponding to the value of the B operand, i.e., $\sum_{i=0}^{n-1} b_i 2^i$, are added to

[3]We assume here a level triggered behavior but the scheme can work with edge triggered policy as well.

the corresponding charge reservoir. This is achieved with n $MVke$ blocks each of them assuming as inputs the b_i bit and having the V input hardwired to the equivalent of a charge reservoir with $1e$ charge, such that $V \times k = 2^i$. Second, on the negative clock value, a charge of $A \times Be$ is added to the other charge reservoir. This is achieved with n $MVke$ blocks assuming as inputs the a_i bits and the analog value present on the charge reservoir processed in the previous computation step. As each $MVke$ block in this stage contributes $a_i \times 2^i \times B$ electrons, a final charge of $\sum_{i=0}^{n-1} a_i 2^i \times Be$, i.e, $A \times Be$ is present in the output charge reservoir when the second step is completed. Last, the value on the output charge reservoir is converted to a binary encoded value representing the product with $2n-1$ PSF blocks.

This new scheme still implies a depth-3 network but requires $2n$ $MVke$ blocks and $2n-1$ PSF blocks. Thus the new scheme reduces the cost in terms of $MVke$ blocks from $O(n^2)$ to $O(n)$ and therefore, the overall asymptotic complexity is also reduced to $O(n)$ as no partial products have to be explicitly generated.

7 Conclusions

Single Electron Tunneling (SET) technology offers a potential for (sub)nanometer feature size scaling, room temperature operation, as well as ultra-low power consumption. However, it displays a switching behavior that differs from traditional MOS devices. This provides new possibilities and challenges for implementing computer arithmetic circuits. In this line of reasoning we investigated the implementation of basic arithmetic functions, such as addition and multiplication, in SET technology. First, we described the SET equivalent of two conventional design styles, namely the equivalents of CMOS and threshold logic gates. Second, we proposed a set of building blocks, which can be utilized for a novel design style, namely arithmetic operations performed by direct manipulation of the location of individual electrons within the system. Third, using the new set of building blocks, we proposed several novel approaches for computing arithmetic functions, e.g., addition, parity, counting, multiplication, via the controlled transport of individual electrons. Related to these new electron counting schemes we proved that the following holds true: the addition/subtraction of two n-bit operands can be computed with a depth-2 network composed out of $3n+1$ circuit elements; the n-parity function can be computed with a depth-2 network constructed with $n+1$ circuit elements; $n|\log n$ counter can be implemented with a depth-2 network constructed with $n + \log n$ circuit elements; m n-bit multiple operand addition can be implemented using at most $nm + n + [\log m] + 1$ circuit elements; and finally the multiplication of two n-bit operands can be computed with a depth-3 network with $4n-1$ circuit elements.

References

[1] C.Lageweg, S.Cotofana, and S.Vassiliadis. Static Buffered SET Based Logic Gates. In *2nd IEEE Conference on Nanotechnology (NANO)*, August 2002.

[2] C.Lageweg and S.Cotofana and S.Vassiliadis. A Linear Threshold Gate Implementation in Single Electron Technology. In *IEEE Computer Society Workshop on VLSI*, pages 93–98, April 2001.

[3] C.Lageweg and S.Cotofana and S.Vassiliadis. Achieving Fanout Capabilities in Single Electron Encoded Logic Networks. In *6th International Conference on Solid-State and IC Technology (ICSICT)*, October 2001.

[4] C.Lageweg and S.Cotofana and S.Vassiliadis. Digital to Analog Conversion Performed in Single Electron Technology. In *1st IEEE Conference on Nanotechnology (NANO)*, October 2001.

[5] C.Wasshuber. *About Single-Electron Devices and Circuits*. PhD thesis, TU Vienna, 1998.

[6] J.R.Tucker. Complementary Digital Logic based on the "Coulomb Blockade". *Journal of Applied Physics*, 72(9):4399–4413, November 1992.

[7] K.K.Likharev. Correlated Discrete Transfer of Single Electrons in Ultrasmall Tunnel Junctions. *IBM Journal of Research and Development*, 32(1):144–158, January 1988.

[8] K.K.Likharev. Single-Electron Devices and Their Applications. *Proceeding of the IEEE*, 87(4):606–632, April 1999.

[9] L.J.Geerligs, V.F.Anderegg, P.A.M.Holweg, J.E.Mooij, H.Poitier, D.Esteve, C.Urbina, and M.H.Devonet. Frequency-Locked Turnstile Device for Single Electrons. *Physical Review Letters*, 64(22):2691–2694, May 1990.

[10] M.Goossens. *Analog Neural Networks in Single-Electron Tunneling Technology*. PhD thesis, Delft University of Technology, 1998.

[11] K. M.Kirihara. Asymmetric Single Electron Turnstile and Its Electronic Circuit Applications. *IEICE Transactions on Electronics*, E81-C(1):57–61, January 1998.

[12] S.Muroga. *Threshold Logic and its Applications*. Wiley and Sons Inc., 1971.

[13] Technology roadmap for nanoelectronics. Downloadable from website http://www.cordis.lu/esprit/src/melna-rm.htm, 1999. Published by the Microelectronics Advanced Research Initiative (MELARI NANO), European Commission.

[14] Y.Ono, Y.Takahashi, K.Yamazaki, M.Nagase, H.Namatsu, K.Kurihara, and K.Murase. Fabrication Method for IC-Oriented Si Single-Electron Transistors. *IEEE Transactions on Electron Devices*, 49(3):193 –207, March 2000.

[15] Y.Taur, D.A.Buchanan, W.Chen, D.Frank, K.Ismail, S.Lo, G.Sai-Halasz, R.Viswanathan, H.Wann, S.Wind, and H.Wong. CMOS Scaling into the Nanometer Regime. *Proceeding of the IEEE*, 85(4):486–504, 1997.