Periodic Symmetric Functions and Addition Related Arithmetic Operations in Single Electron Tunneling Technology

Cor Meenderinck Sorin Cotofana
Computer Engineering Lab, Delft University of Technology, Delft, The Netherlands
{Cor,Sorin}@ce.et.tudelft.nl

Abstract— **This paper investigates the implementation of Periodic Symmetric Functions (PSF) in single electron tunneling technology. First, we discuss the procedure of expressing arithmetic and logic operations as generalized PSFs. Second, we propose a building block that performs a multiple input PSF. The block we propose can be used for the computation of any function that is or can be expressed as a PSF, thus it can be utilized for the implementation of a large number of arithmetic operations, e.g., parity, addition, multi-operand addition, as they belong to the class of generalized PSFs. To demonstrate the capabilities of the PSF scheme we present the design and simulation results of a PSF based 3-bit adder and a PSF based block save adder. Finally, we explain how these PSF based schemes can be used in a Single Electron Encoded Logic (SEEL) environment.**

Keywords— **single electron tunneling, periodic symmetric function.**

I. Introduction

It is generally expected that current semiconductor technologies, i.e., CMOS, cannot be pushed beyond a certain limit because of problems arising in the area of power consumption and scalability. A promising alternative is Single Electron Tunneling (SET) technology [1], which has the potential of performing computation with lower power consumption than CMOS and is scalable to the nanometer region and beyond [2].

Several proposals have been made to implement computational operations using SET technology and these implementations are mainly categorized in two types (see for example [1], [3]). The first type of implementation represents logic values by voltage (see [3] for an overview) while the second type of implementation represents bits by single electrons. Single Electron Encoded Logic (SEEL) [4] is an example of the latter.

Thus far most implementations focussed on designing logic gates to perform operations in the digital domain. SET technology however, possesses properties, e.g. Coulomb oscillations, that open new avenues for the implementation of logic and arithmetic functions. In this line of reasoning we assume in this paper a basic SET structure, the electron trap, that exhibits a periodic behavior and use it as a basis for the implementation of Periodic Symmetric Functions (PSFs). As a large number of arithmetic operations, including addition and parity check, can be expressed as PSFs, the electron trap provides a natural base for nonstandard SET based implementations of logic and arithmetic functional units. In this line of reasoning we investigate in this paper PSF based implementations of addition related arithmetic functions.

The remainder of this paper is organized as follows. Section II briefly describes the SET phenomenon. In Section III some preliminaries on periodic symmetric functions are presented and it is explained how arithmetic and logic operations can be expresed as PSFs. In Section IV a building block is proposed that performs a multiple input generalized periodic symmetric function. Using this building block in Section V a PSF based 3-bit adder is proposed and in Section VI a PSF based block save adder is proposed. In Section VII it is described how this scheme can be adjusted to operate in a SEEL environment by augmenting it with static inverting buffers. In Section VIII some practical considerations are discussed and Section IX concludes the paper.

II. Background

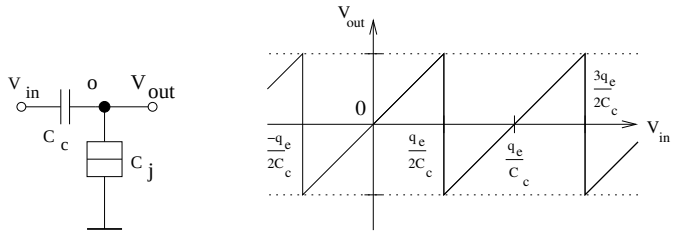
SET circuits are based on tunnel junctions which consist of an ultra-thin insulating layer in a conducting material. In classical physics no charge transport is possible through an insulator. However, when the insulating layer is thin enough the transport or tunneling of charge can occur in a discrete and accurate manner, i.e., one electron at a time, if it reduces the amount of energy in the system. Tunneling through a junction becomes possible when the junction's current voltage V_j exceeds the junction's critical voltage $V_c = \frac{q_e}{2(C_e+C_j)}$ [5], where $q_e = 1.602 \cdot 10^{-19}C$, C_j is the capacitance of the tunnel junction, and C_e is the equivalent capacitive value of the remainder of the circuit as seen from the junction. In other words, tunneling can occur if and only if $|V_j| \geq V_c$, in which case the junction is called unstable. Electron tunneling is stochastic in nature and as such the delay cannot be analyzed in the traditional sense. Instead, for each transported electron one can describe the switching delay as

$$t_d = \frac{-ln(P_{error})q_eR_t}{|V_j| - V_c}, \quad (1)$$

where R_t is the junction's resistance and P_{error} is the chance that the desired charge transport has not occurred after t_d seconds. In this paper we assume $R_t = 10^5\Omega$ and $P_{error} = 10^{-8}$.

Note that the implementations discussed in here are technology independent. SET tunnel junctions can for example be implemented by classical semiconductor lithography and by carbon nanotubes [6]. Therefore, circuit area is evaluated in terms the total number of circuit elements (capacitors and junctions).

A well know SET structure is the electron trap depicted in Figure 1 which has a periodic transfer function. The SET electron trap functions as follows. If the input voltage rises, the output voltage follows due to capacitance division. At



(a) Circuit. (b) Transfer function.

Fig. 1. SET electron trap.

some point, though, the voltage across the tunnel junction exceeds the critical voltage and an electron tunnels to the output node. The output voltage therefore drops. As the input voltage continues to rise, the output voltage rises again until it reaches the critical voltage.

The relation between the input voltage V_{in} and the output voltage V_{out} of the electron trap can be derived as

$$V_{in} = \frac{C_{\Sigma o}}{C_i} V_{out} + \frac{q_o}{C_i}, \tag{2}$$

where $C_{\Sigma o}$ is the sum of all capacitances connected to node o and q_o is the net charge in node o. The critical voltage of the tunnel junction is expressed as $V_c = \frac{q_e}{2C_{\Sigma o}}$. We know from the description of the electron trap, that the output voltage reaches its maximum when this voltage reaches the critical voltage of the tunnel junction. Thus, by substituting the expression of the critical voltage into Equation (2), the input voltage for which the output voltage reaches its maximum can be expressed as:

$$V_{i,peak} = \frac{q_e}{2C_i} + \frac{kq_e}{C_i} \quad \text{for } k = 0, 1, 2, ... \tag{3}$$

This equation suggests that the period of the electron trap transfer function is dependent only on the magnitude of capacitance C_i, while the capacitance of the tunnel junction has no influence. The periodic nature of the electron trap transfer function provides the fundamentals for effective implementation of Periodic Symmetric Functions (PSFs) in SET technology, which is explained in Section IV.

III. PSF based arithmetic

A function on n variables is symmetric if and only if for any permutation σ of $< 1, 2, ..., n >$, $F_s(x_1, x_2, ..., x_n) = F_s(x_{\sigma(1)}, x_{\sigma(2)}, ..., x_{\sigma(n)})$. In other words, a symmetric function is independent on the order of the operands. Addition and multiplication, for example, are symmetric functions since $x_1 + x_2 = x_2 + x_1$ and $x_1 * x_2 = x_2 * x_1$.

As Boolean functions have operands that are either '0' or '1' Boolean symmetric functions depend on the number of 'ones' in the input. Thus, a Boolean symmetric function entirely depends on the sum of its input values: $F_s(x_1, x_2, ..., x_n) = F_s(\sum_{i=1}^{n} x_i)$. This allows for a more compact representation of the function as it can be described by a vector $v = v_0 v_1 ... v_n$ where v_i is the output of F_s when the sum of the inputs is i. This representation is linear in the number of inputs while the traditional truth table has a size of 2^n entries.

X	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
s_0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
s_1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1
s_2	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1
s_3	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1

TABLE I
Symmetric functions $F_s(X)$ for sum bits of 3-bit addition.

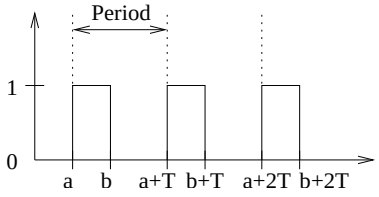


Fig. 2. Periodic Symmetric Function.

A generalized symmetric function $F_g(X)$ is a function that depends on the weighted sum of its inputs $X = (\sum_{i=1}^{n} x_i w_i)$, where w_i is the weight of input x_i.

A Periodic Symmetric Function (PSF) is a symmetric function for which there exists a period T such that $F_s(X) = F_s(X + T)$. A PSF is completely defined by the constants a, b, and T, where a is the first positive transition and b is the first negative transition (see Figure 2).

Many arithmetic operations can be described as generalized periodic symmetric functions. Assume, for example, the binary addition of operands $\{a_2, a_1, a_0\}$ and $\{b_2, b_1, b_0\}$ resulting in a sum $\{s_3, s_2, s_1, s_0\}$. A weight $w_k = 2^k$ can be assigned to every a_k, b_k such that $X = \sum_{k=0}^{2} 2^k (a_k + b_k)$. Each sum bit s_i can be described as a symmetric function of X as shown by Table I.

From the table a periodicity can be observed for all four symmetric functions s_i, $i = 0, 1, 2, 3$. More precisely each sum bit s_i can be calculated with a periodic symmetric function $F_{s,i}(X)$ that has a period of $T = 2^{i+1}$ and the first positive transition at $2^{(i+1)}$. Given that each input bit a_k, b_k has a weight $w_k = 2^k$, each input bit with $k > i$ has no effect on the value of $F_{s,i}(X)$. Consequently each sum bit s_i can be calculated with a periodic symmetric function $F_{s,i}$ as:

$$s_i = F_{s,i}(\sum_{k=0}^{i} 2^k (a_k + b_k)). \tag{4}$$

In the same way many other arithmetic functions can be described by generalized periodic symmetric functions. In the next section a building block is proposed that can evaluate generalized periodic symmetric functions.

IV. Multiple Input *PSF* Building Block

The SET electron trap can be used as a basis for a building block that performs a PSF, though two extensions are

needed. First, the number of inputs of the electron trap needs to be increased. Second, the triangular transfer function of the electron trap needs to be converted into a rectangular one.

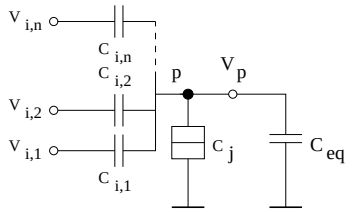


Fig. 3. Multiple input SET electron trap.

Figure 3 presents a modified electron trap, which allows it to be driven by multiple inputs in the same time. The capacitance C_{eq} represents the equivalent capacitance of the circuit connected to the output (V_p) of the electron trap. Every input $V_{i,x}$, $x = 1, ..., n$ contributes to the output voltage according to the following expression

$$V_{p,x} = \frac{C_{i,x}}{C_{\Sigma p}} V_{i,x}, \tag{5}$$

where $C_{\Sigma p} = C_j + C_{eq} + \sum_{x=1}^{n} C_{i,x}$ is the total capacitance connected to node p. The total output voltage is the sum of the contributions of all inputs $V_p = \sum_{x=0}^{n} V_{p,x}$.

From Equation (5) it can be observed that every input $V_{i,x}$ is contributing to the voltage on node p according to the size of the capacitor $C_{i,x}$. Thus by choosing different values for $C_{i,x}$, inputs can be given different weights. Let $V_{i,high}$ be the input voltage representing logic '1' and let w_x be the weight of input x, the corresponding capacitance of input x can be evaluated as:

$$C_{i,x} = \frac{q_e}{2 w_x V_{i,high}}. \tag{6}$$

Equation (6) allows us to compute the capacitances of the inputs and associate to each input its appropriate weight. Now as the number of inputs was extended, the next step is to change the shark tooth transfer function into a rectangular one.

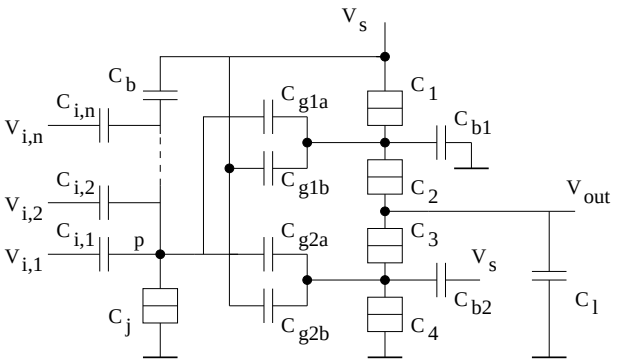
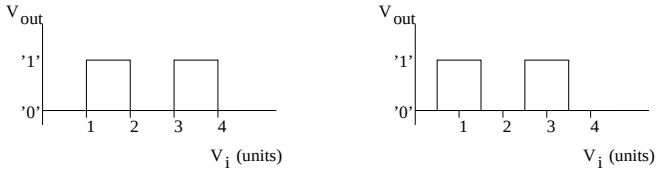


Fig. 4. The multiple input PSF implementation.

To obtain a rectangular shape transfer function we connect to the output of the electron trap a static inverting buffer [4], which then acts as a literal gate. The resulting topology, called the multiple input PSF block is depicted in Figure 4.

We notice here that in the multiple input PSF block, one of the inputs of the electron trap is connected to the supply voltage through the capacitor C_b. This input causes a bias on node p of the electron trap, which is added for the following reason. Assuming an electron trap with only one input and no bias, the transfer function would be as depicted in Figure 5(a). From the transfer function it is seen that the first positive transition is located at exactly an input voltage corresponding to one unit. If, due to various effects (cross-talking, impurities, parameter deviation, etc.) the input voltage is a little less than the voltage corresponding to one unit, the output of the PSF block would be logic '0' instead of the expected logic '1'.



(a) without bias. (b) with bias.

Fig. 5. PSF block transfer function.

The addition of the bias, with a magnitude of half a unit, causes all transitions to move to the left by a half unit (see Figure 5(b)). Consequently, the first positive transition has moved from an input of one unit to a half unit and in this way the structure is less sensitive to parameter variation, etc. Note that the input, though analog, is discrete in nature and only takes values of whole units. Therefore, adding a bias of a half unit results in maximal robustness of the implementation.

V. Example: 3-bit PSF based Adder

When adding two binary numbers $A = \{a_{n-1}, ..., a_1, a_0\}$ and $B = \{b_{n-1}, ..., b_1, b_0\}$ the result is a sum $S = \{s_n, ..., s_1, s_0\}$. As explained in Section III each sum bit s_i can be calculated with a generalized periodic symmetric function. Thus a PSF based addition scheme can be build by utilizing a multiple input PSF block for each output bit and connecting to it all the necessary inputs using the proper weights according to Equation 4.

A 3-bit PSF based adder was build and simulated of which the schematic is depicted in Figure 6. We assumed a supply voltage $V_s = 16mV$ and that logic '1' is represented as $16mV$. Further we assumed that the inputs are driven by ideal voltage sources.

The 3-bit PSF adder has 6 inputs $\{a_2, a_1, a_0, b_2, b_1, b_0\}$ which have weights $\{4, 2, 1, 4, 2, 1\}$ and four outputs $\{s_3, s_2, s_1, s_0\}$. Using these values of the weights and Equa-

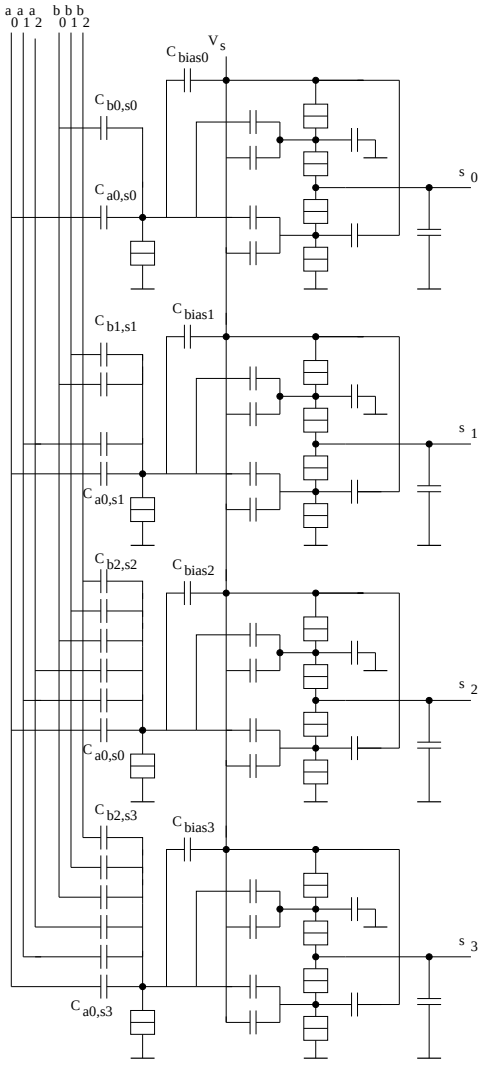


Fig. 6. 3-bit PSF based addition scheme.

tion (6), the values for the input capacitances of the MPSF block were calculated.

The simulation results of the 3-bit PSF based adder are presented in Figure 7 and they indicate that the PSF based adder functions correctly. The adder requires 70 circuit elements and, assuming an error probability $P_{error} = 10^{-8}$, has a delay of 16.0 ns.

VI. Example: PSF based Block Save Adder

Multi-operand addition is often necessary for the realization of some arithmetic operations like multiplication, especially when delay is of great importance. Generally speaking multi-operand addition based multiplication follows the scheme presented in Figure 8, for the particular case of a 4-bit multiplication. First the operands are multiplied bit wise, which results in four rows of bits. To find the final result of the multiplication, a multi operand addition is required. This multi operand addition is traditionally performed by the following two steps. In the first step the number of rows is reduced to two rows using counters (see Figure 9(a,b)), which avoid carry propagation and thus

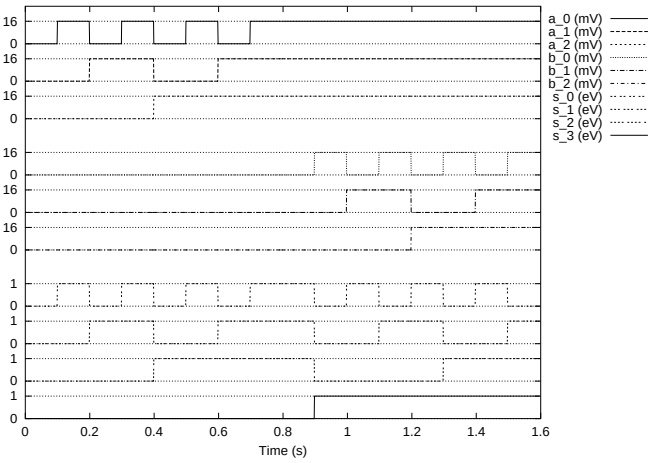


Fig. 7. Simulation results for 3-bit PSF based adder.

minimizes the delay. In the case in Figure 8, one reduction step is sufficient to end up with two rows, but in general more reduction steps might be required. In the second step the two rows of bits are added using a fast adder structure, based for example on a carry look-ahead adder.

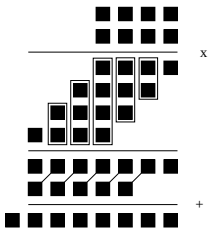


Fig. 8. Schematic of 4 bit multiplication process.

Another approach to implement multi operand addition is to partition the input bits in blocks and to use Block Save Adders (BSAs). Figure 9(c) depicts the schematic of a BSA(4,2,4), a block save addition of four rows and two columns producing a four bit output. Using this approach of partitioning results in a very small depth network, assuming the right block size was chosen [7].

Block save addition can easily be expressed as a generalized periodic symmetric function. Assuming a block of k columns and l rows containing the elements $a_{k,l}$, the sum

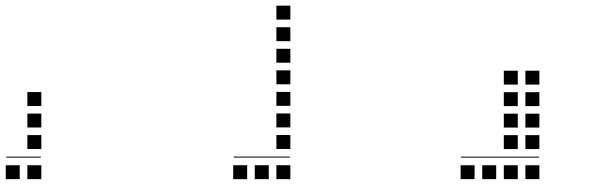


Fig. 9. Schematic of counters and block save adder.

bits can be described as a PSF of

$$X = \sum_{i=0}^{k} 2^i \sum_{j=0}^{l} a_{i,j}, \tag{7}$$

where $a_{0,0}$ is the bit in the right bottom corner. Thus a block save adder can be implemented using multiple input PSF building blocks.

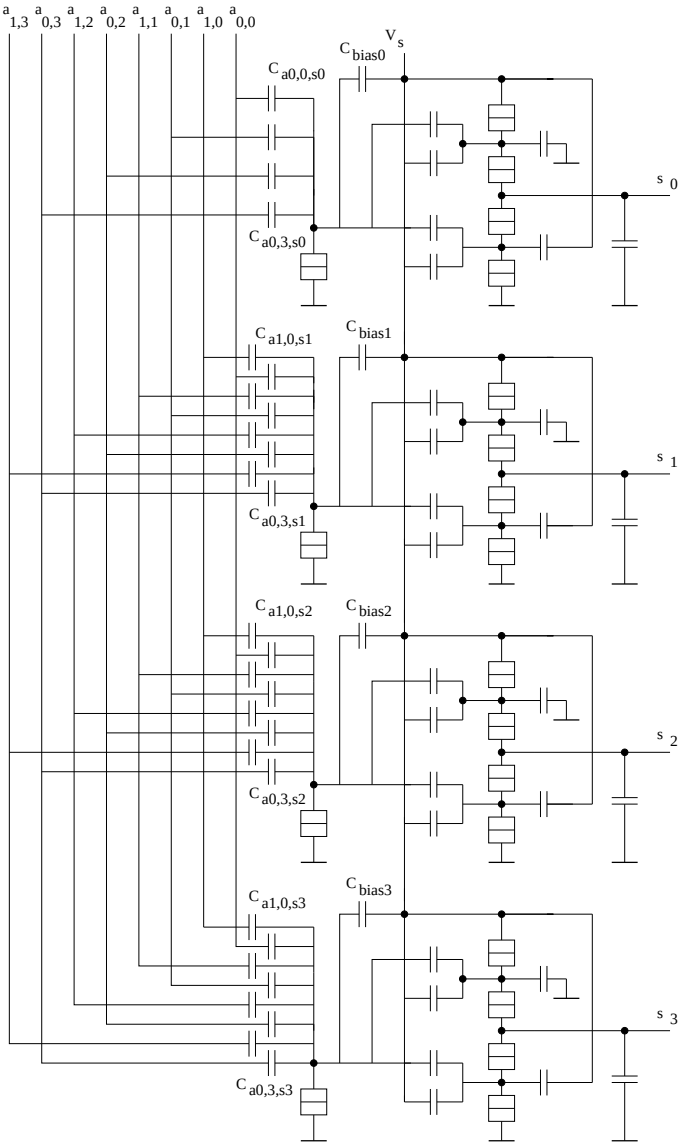


Fig. 10. PSF based BSA(4,2,4) scheme.

Figure 10 depicts the PSF based implementation of a BSA(4,2,4). The multiple input PSF block generating s_0 has a period of 2 and is therefore only connected to inputs with a weight of 1 ($a_{0,j}$). The simulation results for the BSA(4,2,4) based adder are presented in Figure 11 and they indicate that the block save adder functions correctly. The adder requires 80 circuit elements and, assuming an error probability $P_{error} = 10^{-8}$, has a delay of 16.0 ns.

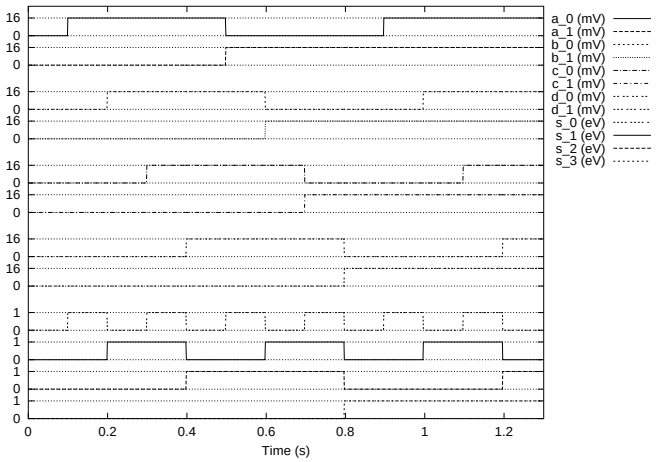


Fig. 11. Simulation results for BSA(4,2,4).

VII. Buffered PSF Scheme

The example circuits presented in Section V and Section VI assumed, among others, ideal input voltage sources. However, when using these schemes in a Single Electron Encoded Logic (SEEL) environment the inputs are not ideal. The output signal of a SEEL gate is generated by a static inverting buffer, which is depicted in Figure 12. The value logic '1' is represented as a net charge of one electron on node o, which results in an output voltage V_o of approximate $16mV$, assuming $C_l = 10aF$. For proper operation, when connecting a gate to the output of the buffer, it is assumed that its input capacitance C_i is much smaller than the load capacitor of the buffer, that is $C_i \ll C_l$. If that is not the case, the output voltage of the buffer would decrease and it might even cause the buffer to malfunction. This situation occurs when connecting a buffer to one input of the PSF addition scheme, or even to a single PSF block. Thus, a PSF based implementation cannot be driven by SEEL circuitry directly.

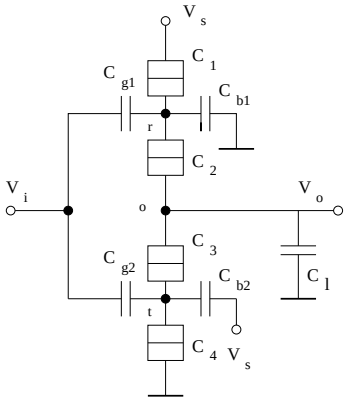


Fig. 12. Static inverting buffer.

The solution to this problem is an integral design of both the output buffers of the SEEL gates and the first stage of the PSF block, the electron trap. For this purpose a mathematical model of the connection between a buffer and an electron trap was derived. The model proved to be non-

linear and can only be solved using numerical methods. For a complete description of the model, the reader is referred to [8]. Calculations proved that it is not possible for a buffer to drive multiple electron traps in the same time and have all the latter function correct. Therefore buffers have to be added in front of each input of every PSF block. Moreover, to eliminate feedforward from the gates driving the inputs of the adder a buffer is also placed on every input of the PSF adder. A buffered 3-bit PSF based adder was designed and verified by means of simulation, which indicated the scheme to work correctly.

VIII. Practical Considerations

In theory any arithmetic operation that can be expressed as a periodic symmetric function, can be build using multiple input PSF building blocks. However, practical considerations limit the number of inputs for the such schemes.

The first problem that arises when designing PSF based circuits with large operands, is the required accuracy. A PSF block with a large weight has a large number of unit steps in a period of its transfer function and thus a small step size. Small unit steps results in small margins for the threshold of the output buffer of the PSF block and therefore a high accuracy is needed. Thus the required accuracy is depending on the number of output bits.

The second problem is the delay of the PSF adder, which is exponential to the number of output bits. For addition this means that the delay is also exponential to the number of inputs, since the latter is linear to the number of output bits. But for multi operand addition the relation between the number of inputs and outputs is less than linear, resulting in a delay less than exponential. For parity check the delay is even independent on the number of inputs ($O(1)$).

To build an adder with large numbers of inputs, based on the PSF addition scheme, a hierarchical approach can be used. In this way the input operands are partitioned in $\frac{k}{2}$-bit blocks, where k is the maximum number of inputs a PSF adder can accommodate. For each block one PSF adder can be used and these PSF adders can be cascaded in a ripple-carry scheme or used in more efficient structures, e.g., carry look-ahead, carry-skip, etc.

IX. Conlcusion

This paper investigated the implementation of Periodic Symmetric Functions (PSF) in single electron tunneling technology. First, we discussed the procedure of expressing arithmetic and logic operations as generalized PSFs. Second, we proposed a generic building block that can evaluate multiple input PSFs. The block we proposed can be used for the computation of any function that is or can be expressed as a PSF, thus it can be utilized for the implementation of a large number of arithmetic operations, e.g., parity, addition, multi-operand addition, as they belong to the class of generalized PSFs. To demonstrate the capabilities of the PSF scheme we presented the design and simulation results of a PSF based 3-bit adder and a PSF based block save adder. Finally, we explained how these PSF based schemes can be embedded in a Single Electron Encoded Logic (SEEL) environment.

References

[1] R. Waser, Ed., *Nanoelectronics and Information Technology - Advanced Electronic Materials and Novel Devices*, 1st ed. Wiley-VCH, Berlin, 2003.

[2] "International Technology Roadmap for Semiconductors, 2003 Edition, Executive Summary," Downloadable from website http://public.itrs.net/home.htm, 2003, available from SEMATECH, ITRS department, 2706 Montoppolis Drive, Austin TX 78741, USA.

[3] K. Likharev, "Single-Electron Devices and Their Applications," *Proceeding of the IEEE*, vol. 87, no. 4, pp. 606–632, April 1999.

[4] C. Lageweg, S. Cotofana, and S. Vassiliadis, "Static buffered set based logic gates," in *2nd IEEE Conference on Nanotechnology (NANO)*, August 2002, pp. 491–494.

[5] C. Wasshuber, "About single-electron devices and circuits," Ph.D. dissertation, TU Vienna, 1998.

[6] K. Ishibashi, D. Tsuya, M. Suzuki, and Y. Aoyagi, "Fabrication of a Single-Electron Inverter in Multiwall Carbon Nanotubes," *Applied Physics Letters*, vol. 82, no. 19, pp. 3307–3309, February 2001.

[7] S. Vassiliadis, S. Cotofana, and J. Hoekstra, "Block save addition with threshold gates," Laboratory of Computer Archituecture and Digital Techniques, Delft University of Technology, Tech. Rep., 1995.

[8] C. Meenderinck, "Single electron technology based arithmetic operations," Master's thesis, CE-MS-2005-01, Delft University of Technology, 2005.