ROMANIAN JOURNAL OF INFORMATION SCIENCE AND TECHNOLOGY
Volume **10**, Number 1, 2007, 67–83

An Analysis of Basic Structures for Effective Computation in Single Electron Tunneling Technology

C. MEENDERINCK, S. COŢOFANĂ

Computer Engineering Lab, Delft University of Technology
Delft, The Netherlands
E-mail: {Cor,Sorin}@ce.et.tudelft.nl

Abstract. Single Electron Tunneling (SET) technology appears to be a promising alternative for CMOS as it exhibits excellent power consumption and scalability features. Moreover, this new technology opens up avenues for new computational paradigms, which require building blocks with unconventional behavior. In this paper we discuss a number of basic building blocks that allow for effective implementations of computational structures in those new paradigms, and analyze them in terms of area, delay, and energy consumption.

Keywords: single electron tunneling, logic design, threshold logic, computer arithmetic, electron counting.

1. Introduction

Ever since decades we have seen an ongoing increase in integrated circuit performance mainly due to advances in fabrication technology and improvements in computational paradigms. However, fabrication technology is kind of stagnating and it is generally expected that current technology, i.e., CMOS, cannot be pushed beyond a certain limit. This limit is expected to arise in mainly two areas: power consumption and scalability. The International Technology Roadmap for Semiconductors (ITRS) [1] states that "we have reached the point where the horizon of the Roadmap challenges the most optimistic projections for continues scaling of CMOS." Consequently, the Roadmap included post-CMOS devices. Moreover, the amount of research in this field of emerging technologies has exploded.

A promising candidate to succeed CMOS is Single Electron Tunneling (SET) technology [2], as it does not suffer from the limitations CMOS faces (power consumption and scalability). SET technology allows the control of single or few electrons and therefore has potential to perform computation with ultra low power consumption. Downscaling feature sizes increases the quantum mechanical behavior, especially when reaching the nanometer region. For CMOS this causes problems whereas for SET, which is based on quantum mechanical principles, this improves device behavior. Consequently, SET technology is scalable to the nanometer region and beyond.

SET technology is fundamentally different from CMOS as it is based on tunneling of electrons. This difference opens up avenues for new computational paradigms [3, 4, 5], which try to effectively use the basic SET properties. Theoretical results on the complexity of logic and arithmetic operations using those new paradigms indicate great potential. However, the actual practical results depend on the capabilities of the utilized building blocks. In previous research we already identified a number of basic building blocks [3, 6, 7]. In this paper we analyze these building blocks with respect to limitations, area, delay, and energy consumption.

This paper is organized as follows. In Section 2 we briefly present some background on SET technology. In Section 3 we present five basic building blocks in SET technology and analyze them. In Section 4 we provide an example of how the presented building blocks can be used for computational purposes. Finally, Section 5 concludes the paper.

2. Background

In the classic physics theory electrons are viewed of as particles and the theory does not allow electrons to cross a barrier like a piece of insulator. In 1923 L. de Broglie [8] suggested that particles may also behave like waves. Three years later this hypothesis was formally described by Schrödinger (see for example [9] for the Schrödinger theory), which became the basis for the quantum mechanics theory of today. According to the Schrödinger wave equation, there is a probability that an electron tunnels through a barrier and enters a classically forbidden region and this phenomenon is called tunneling.

Tunneling

The tunnel junction, the basic circuit element of SET technology, is based on this tunneling phenomenon. The tunnel junction is created by separating two conductors with a thin insulator (see Fig. 1) and therefore it behaves in principle like a capacitor. However, given that the insulator is thin enough quantum tunneling may occur.

For an electron to tunnel through the junction, the Coulomb energy $E_C = \frac{q_e^2}{2C}$, where C is the capacitance of the tunnel junction and q_e is the charge of an electron (1.60217×10^{-19} C), is at least needed. If the Coulomb energy is not available a tunnel event cannot happen. This phenomenon is known as Coulomb blockade. A voltage source can provide the energy needed for an electron to tunnel.

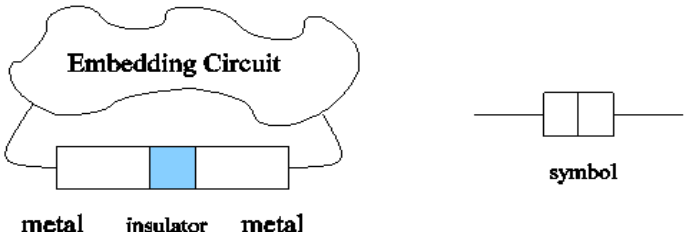

Fig. 1. Tunnel junction schematic representation.

There is another condition for observing the tunneling phenomenon. In classical theories an electron was assumed to be well localized. However, in the quantum mechanics theory electrons are described by wave functions, indicating the probability of the presence of an electron. If a tunnel barrier is insufficiently opaque the electron wave function extends through the barrier and the electron is not clearly localized on either site of the tunnel junction. The opaqueness of a tunnel barrier is described by the tunnel resistance R_t. A sufficient condition [10, 11] for observing SET charging effects is:

$$R_t \gg \frac{h}{q_e^2} = 25.6 \text{ k}\Omega, \tag{1}$$

where h is Planck's constant (6.62607×10^{-34} J·s). For further explanation of the tunnel resistance and the derivation of this equation we refer the reader to [12, 13]. In this research we assume a tunnel resistance R_t of 100 kΩ, which is a commonly used value.

Analyzing SET circuits

When designing circuits with tunnel junctions, one needs to be able to calculate the conditions for observing electron tunneling. This could be done by calculating all free energy in the circuit, but even for modest sized circuits this method results in very complex computation [14]. Therefore, the method of the critical voltage is generally employed [15, 16, 17]. This method states that an electron may only tunnel if the voltage across the tunnel junction V_j is greater than a critical voltage V_c. The critical voltage of a tunnel junction can be expressed as:

$$V_C = \frac{q_e}{2\,(C_j + C_e)}, \tag{2}$$

where C_j is the capacitance of the tunnel junction and C_e is the capacitance of the remainder of the circuit as seen from the junction's perspective. In other words, tunneling can occur if and only if $|V_j| \geq V_c$.

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:

$$t_d = \frac{-\ln\left(P_{error}\right) q_e R_t}{|V_j| - V_c}, \tag{3}$$

where P_{error} is the chance that the desired charge transport has not occurred after t_d seconds [4]. In this paper we assume P_{error}=10^{-8}. Each transported electron reduces the system energy by $\Delta E = q_e\left(|V_j| - V_c\right)$ from which the consumed energy can be calculated.

Note that the SET technology can physically be implemented in various ways, e.g., classical semiconductor lithography 18] and by carbon nanotubes [19]. Therefore, for the blocks we discuss in this paper, the circuit area is evaluated in terms of the total number of circuit elements (capacitors and junctions).

3. Basic building blocks

SET technology enables accurate control of the transportation of discrete electrons. Moreover, SET allows the representation of values by the number of electrons, i.e., Boolean values are represented by the presence or absence of one electron, while integer values are represented by the corresponding number of electrons. To effectively utilize these encodings in arithmetic and logic operations, building blocks are required that perform basic signal operations on Boolean and multi-value signals.

Previous investigations suggested that Boolean operations can be implemented using threshold logic gates and inverting buffers [4]. Those two building blocks operate in principle like normal Boolean gates. In order to perform arithmetic operations via direct charge manipulation [3] the following set of building blocks is required: MVke (Move k electrons) block, MCke (Move Conditionally k electrons) block, and PSF (periodic symmetric function) block. The first two are mainly used to create and manipulate multi-value signals while the latter is mainly used to read multi-value signals and perform their conversion to Boolean signals. This section presents the implementation and analysis of those basic building blocks. The results presented in here are based on calculations and SIMON [20] simulations.

3.1. Threshold gate

An n-input linear threshold logic gate is a device which is able to evaluate any linearly separable Boolean function given by:

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

$$f\left(X\right) = \sum_{i=1}^{n} w_i x_i - \varphi, \tag{5}$$

where $x_i s$ are the n inputs and $w_i s$ are the corresponding n integer weights. The linear threshold gate performs a comparison between the weighted sum of the inputs $\sum_{i=1}^{n} w_i x_i$ and the threshold value ϕ. If the weighted sum of its inputs is greater than or equal to the threshold, the gate produces logic '1'. Otherwise the output is logic '0'. The threshold logic gate can operate on Boolean signals as well as on multi-valued digital or analog signals.

Figure 2 depicts an implementation of the threshold gate in SET technology. The critical voltage V_c, needed to enable tunneling, acts as the intrinsic threshold of the circuit. If the voltage across the junction is larger then V_c an electron tunnels through the junction from node $n2$ to node $n1$, resulting in a 'high' output that is associated with logic '1'. The supply voltage V_s is used as the biasing voltage, weighted by the capacitor C_b, to adjust the gate threshold to the desired value ϕ. The input signals $V^p = \{V_1^p, V_2^p, ..., V_r^p\}$ are weighted by their corresponding capacitors $C^p = \{C_1^p, C_2^p, ..., C_r^p\}$ and added to the voltage across the junction. The input signals $V^n = \{V_1^n, V_2^n, ..., V_s^n\}$ are weighted by their corresponding capacitors $C^n = \{C_1^n, C_2^n, ..., C_s^n\}$ and subtracted from the voltage across the junction. The resulting threshold function for the circuit is:

$$f(X) = C_\Sigma^n \sum_{k=1}^{r} C_k^p V_k^p - C_\Sigma^p \sum_{l=1}^{s} C_l^n V_l^n - \varphi, \tag{6}$$

$$\varphi = \frac{1}{2}\left(C_\Sigma^p + C_\Sigma^n\right) q_e - C_\Sigma^n C_b V_b, \tag{7}$$

where $C_\Sigma^p = C_b + \sum_{k=1}^{r} C_k^p$ and $C_\Sigma^n = C_o + \sum_{l=1}^{s} C_l^n$.

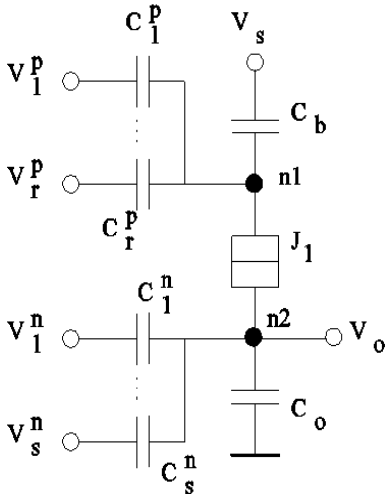

Fig. 2. Threshold logic gate.

Using the implementation presented above, we can create all kinds of Boolean gate, e.g., AND, NAND, OR, and NOR. Theoretically speaking, we can create Boolean gates with an arbitrary number of positive and negative weighted inputs. However, for practical circuits the size of the capacitors C_i^p and C_i^n as well as the desired output voltage restrict the possibilities. The exact limitations are dependent on the actual implementation.

The area cost of the circuit is five elements. As the energy consumption and the delay are dependent on the voltage across the junction at the time of a tunnel event, they are thus implementation dependent. For a typical implementation of a 2-input Boolean gate the energy consumption is approximately 0.8 meV and the delay

is approximately 0.8 ns. In this paper a typical implementation assumes a supply voltage of 16 mV and a representation of logic '1' of 16 mV as well.

We also assessed the dependence between delay, energy, and the number of inputs for different values of P_{error}, of which the results are presented in Fig. 3. The energy consumption is linear with the number of inputs. However, the delay is independent on the number of inputs, in contrast to CMOS logic gates where we see such a dependency.

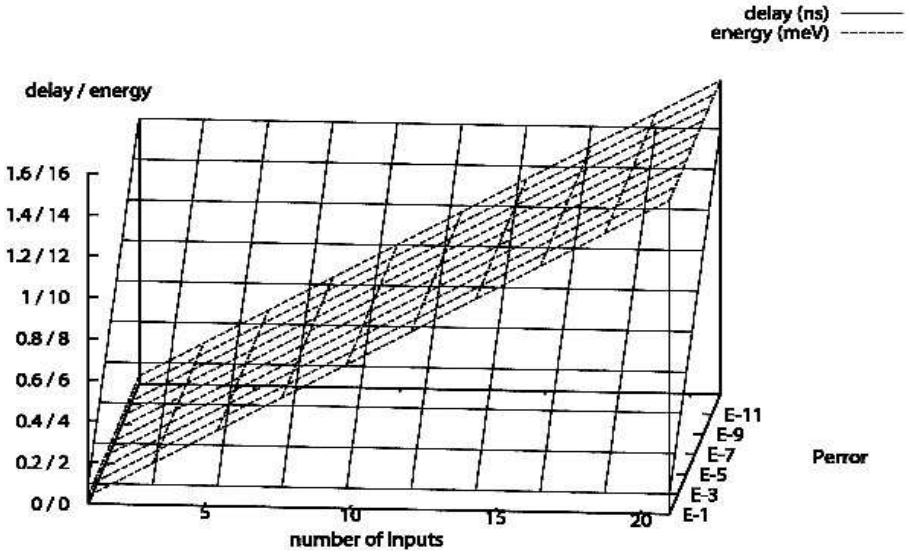

Fig. 3. Delay and energy consumption of an n-input AND gate implemented by a threshold gate.

3.2. The inverting buffer

To improve the fan-out capabilities and to reduce feedback effects, the threshold logic gate has to be augmented with a buffer. Figure 4 depicts a possible implementation of such an inverting buffer. Other implementations are possible, but this one is preferred as it has the lowest output-to-input feedback ratio, which is fundamental for a buffer.

The junctions J_1, J_2 and the capacitor C_{g1} in the figure form a SET transistor [21]. When combined with the bias capacitor C_{b1} the switching behavior of this SET transistor becomes similar to that of a MOS p-type transistor. Likewise, junctions J_3, J_4, and capacitors C_{g2}, C_{b2} displays a switching behavior similar to an n-type MOS transistor. The circuit as a whole operates as follows. Assuming all circuit nodes contain a net charge of 0 electrons and V_i is 'low', a tunnel event can occur in junction J_1 only, which in turn results in a tunnel event in J_2 and a 'high' output. If then V_i becomes 'high', a tunnel event can occur in junction J_4 only, which in turn results in a tunnel event in J_3 and a 'low' output.

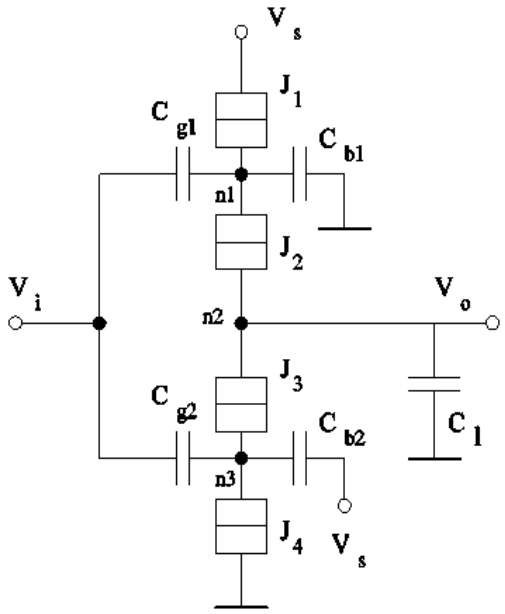

Fig. 4. Inverting buffer.

The area cost of the inverting buffer is 9 elements. When designing such a buffer, there are a number of trade offs that can be made, e.g., delay, energy consumption, feedback ratio, and sensitivity to imperfections. A number of different designs have been implemented and simulated. The delay ranges from 0.4 ns to 1.9 ns, while the energy consumption ranges from 7 meV to 11 meV, assuming zero fan-out. Figure 5 depicts the dependency of the delay and energy consumption on the fan-out of the buffer. For this experiment we assumed that all gates on the output of the buffer have an input capacitance equal to that of the buffer itself, i.e., 0.5 aF. From the graph we can see that the delay of the buffer doubles for every 11 extra gates on the output. The maximum number of gates that the buffer can drive is dependent on the amount of feedback that the gates cause on the output node of the buffer. Our experiments indicate that for a typical implementation of the buffer it can drive up to 19 threshold logic gates.

3.3. MVke

The Move k electron (*MVke*) block controls the transport of an adjustable number of electrons to/from a charge reservoir. An *MVke* block has inputs V_e (enable), V_r (reset), and V_v (V) and has a build in constant k such that the circuit transports Vk electrons when enabled. When a reset is applied all the transported electrons return to their original position and the circuit becomes charge neutral.

Figure 6 presents an *MVke* implementation, which removes electrons from a charge reservoir (implemented by capacitance C_{cr}) and which operates as follows. While V_r (reset) and V_v are zero and input V$_e$ (enable) is set to 'high', the voltage across

junction J_1 gets very close to its critical voltage. If at the same time V_v is set to 'high' the critical voltage is exceeded and one electron tunnels from node $n2$ to node $n1$. As a result of this event a positive charge is present on node $n2$, which causes the voltage of junction J_2 to exceed its critical voltage. So, one electron tunnels from node $n3$, which actually is the charge reservoir, to node $n2$. This process of two tunnel events continues until the voltage across junction J_1 has dropped below its critical voltage again. The number of electrons that are removed from the charge reservoir is proportional to the magnitude of both V_v and C_v. Therefore in this implementation the value of k can be set by choosing the corresponding value of C_v.

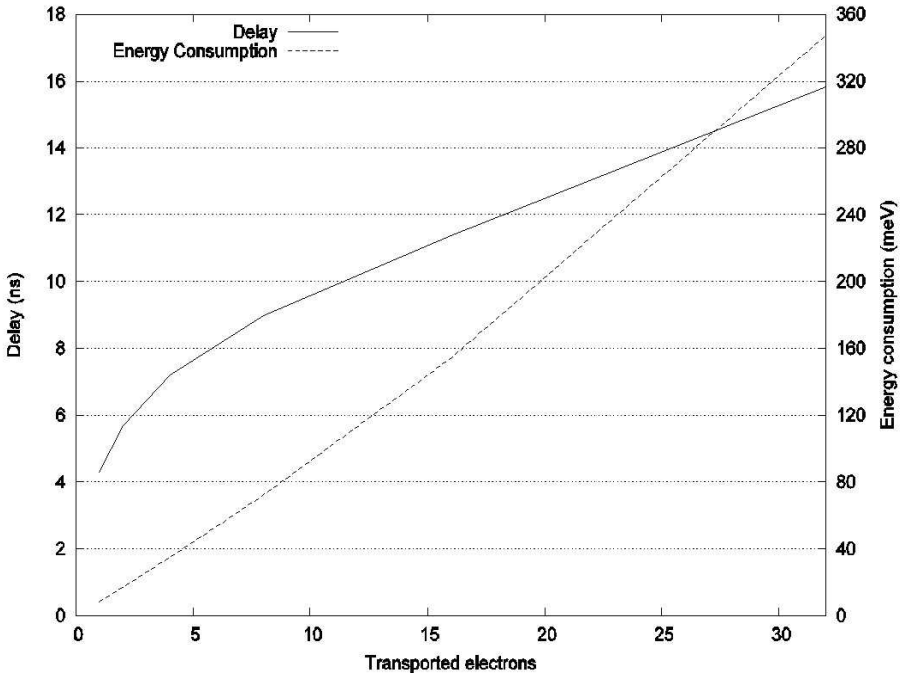

Fig. 5. Delay and energy consumption of the inverting buffer with respect to fan-out.

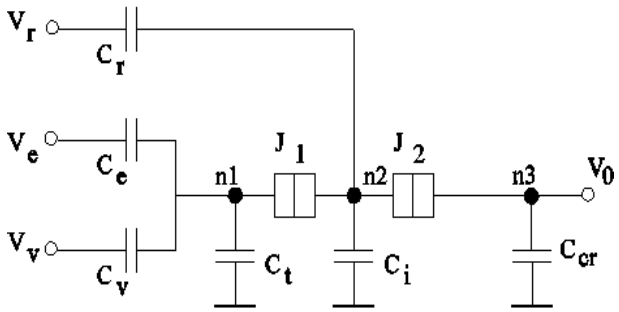

Fig. 6. *MVke* block.

The circuit can be reset to its initial state by setting V_r to 'high' while the other inputs are zero. The positive voltage on V_r causes a negative voltage across junction J_1 which exceeds the critical voltage and thus one electron tunnels from node $n1$ to node $n2$. This in turn places junction J_2 into an unstable state, resulting in an electron tunneling to the reservoir. This process continues until all excess electrons present on node $n1$ are returned to node $n3$ and the charge reservoir is back into the neutral state.

The number of transportable electrons is limited by the voltage they produce on the output reservoir. Implementing the charge reservoir with a large capacitance results in a low voltage on the output of the *MVke* block and thus on a high limit to the number of transportable electrons. On the other hand, the low voltage on the charge reservoir requires a higher accuracy in the next stage, which operates on the value in the reservoir. For a typical implementation of the *MVke* block we calculated an upper limit of the number of transportable electrons of 529.

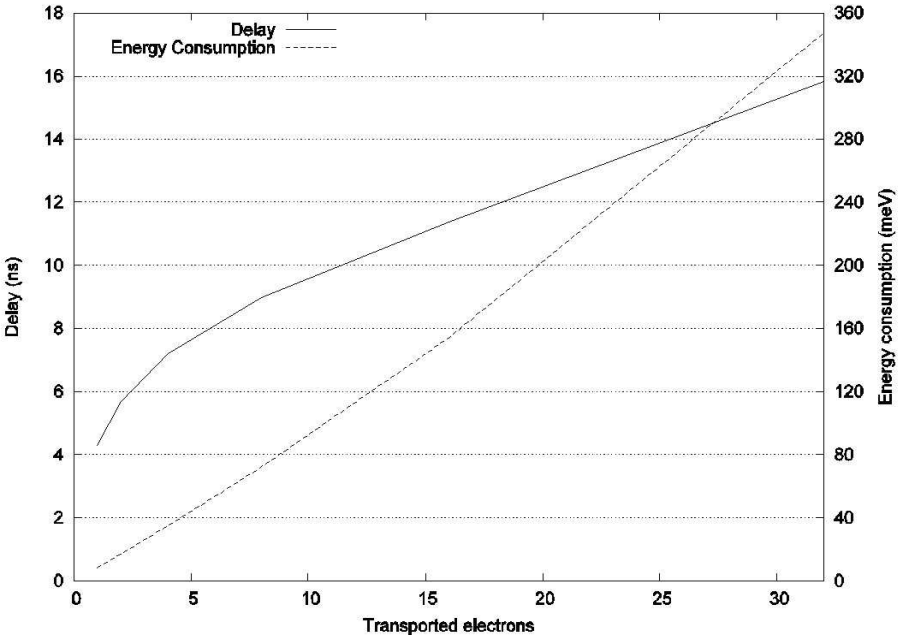

Fig. 7. Delay and energy consumption of the *MVke* block.

The area cost of the *MVke* block is 8 elements. The delay and the energy consumption are dependent on the actual parameters of the circuit and on the actual number of transported electrons. Figure 7 depicts the delay and energy consumption for different values of transported electrons k, assuming a typical implementation. As expected, the energy consumption is linear on the number of transported electrons. The delay is logarithmic on the number of transported electrons, because for large k the first electrons experience a larger 'force' and therefore tunnel a lot faster.

3.4. MCke

The Move Conditional k electrons (*MCke*) block transports a fixed amount of electrons k to/from a charge reservoir if and only if the input V_v exceeds a certain threshold. The *MCke* block also has an enable (V_e) input and a reset (V_r), thus it is capable of returning the transported electrons.

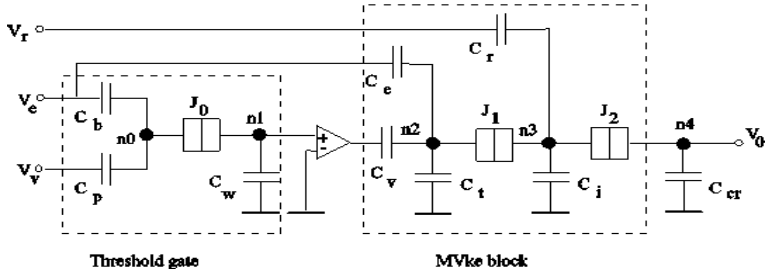

Fig. 8. The *MCke* block.

If we split the functionality of the *MCke* block in two parts, i.e., detecting the threshold condition and moving electrons, it is clear that the *MCke* block can be implemented by using a SET threshold gate in combination with an *MVke* block. The implementation depicted in Figure 8 uses this approach. The output of the threshold gate is buffered by an OpAmp, which is required to guaranty the correct functionality of both building blocks. We note here that OpAmps can potentially be implemented using a hybrid FET-SET technology [22, 23].

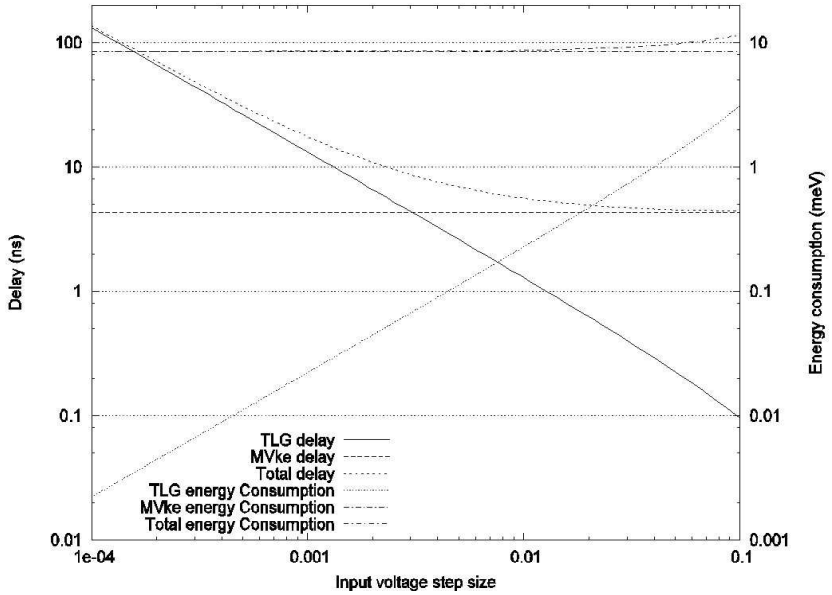

Fig. 9. Delay and energy consumption of the *MCke* block with respect to the input step size.

The area cost of the *MCke* block is 12 elements. Assuming a typical implementation and a value for k of 1, the delay is 1.55 ns and the energy consumption is 10.3 meV. These values are excluding the area, delay, and energy consumption of the required OpAmp. The delay of the threshold logic gate is very much depending on the voltage difference it has to detect. For large steps on the input voltage the delay is very small. However, for small steps on the input the delay can be very large. Figure 9 depicts the delay and energy consumption of the threshold logic gate (TLG) and the *MVke* block as well as of the entire *MCke* block. For this experiment we assumed for the *MVke* block a value for k of 1.

3.5. PSF

A *PSF* building block implements a Periodic Symmetric Function (PSF) on one or multiple inputs. A Boolean function of n variables F_s, is symmetric if and only if for any permutation σ of <1,2,...,n >, $F_s\left(x_1, x_2, ..., x_n\right) = F_s\left(x_{\sigma(1)}, x_{\sigma(2)}, ..., x_{\sigma(n)}\right)$. In other words, a Boolean symmetric function entirely depends on the sum of its input values $F_s\left(x_1, x_2, ..., x_n\right) = F_s\left(\sum_{i=1}^{n} x_i\right)$. A periodic symmetric function is a symmetric function for which there exists a period T such that $F_s\left(X\right) = F_s\left(X + T\right)$. 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 10).

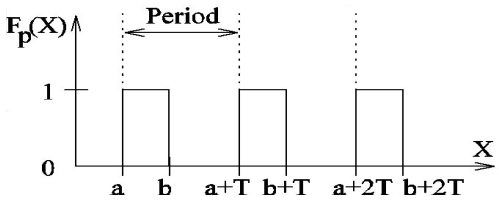

Fig. 10. Periodic symmetric function.

As a basis for the implementation of a *PSF* block a simple SET structure, known as the electron trap, can be used. The electron trap, depicted in Figure 11, has a periodic transfer function and functions as follows. If the input voltage rises, the output voltage follows due to capacitance division. At 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 again and the entire process continues until enough electrons tunneled.

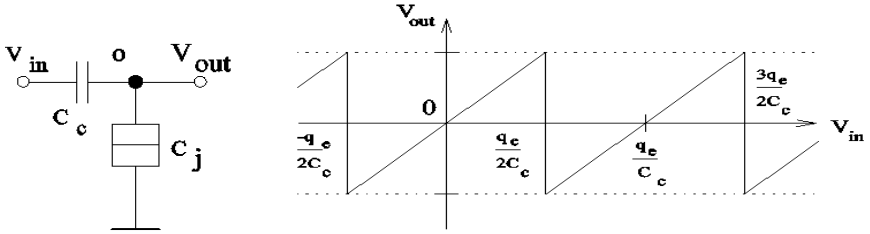

Fig. 11. Electron trap circuit and transfer function.

A *PSF* block can be implemented using a SET electron trap in combination with a SET inverter (see Figure 12). The electron trap has a triangular periodic transfer function. The inverter acts in this case as a literal gate and it transforms the saw tooth shaped transfer function of the electron trap into a rectangular shape transfer function.

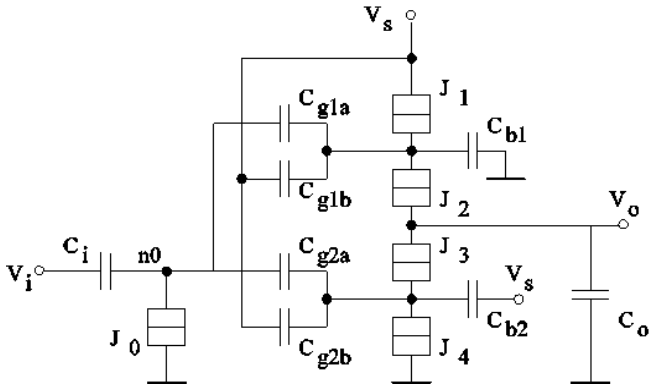

Fig. 12. The PSF block.

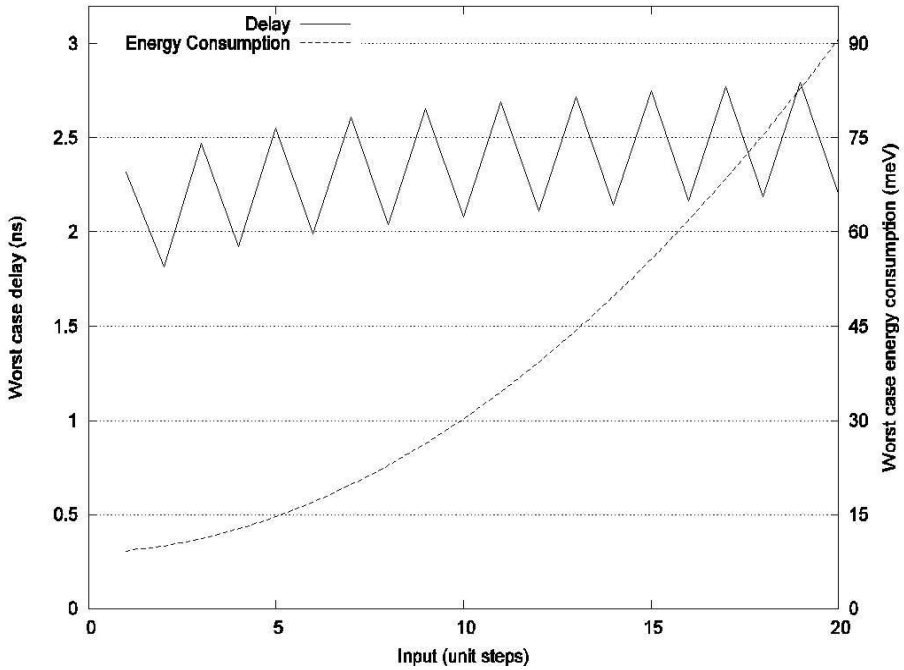

Fig. 13. Delay and energy consumption of the PSF block.

The *PSF* block as presented in Figure 12 can be extended with more inputs by simply connecting those via capacitances to node $n0$ and choosing a capacitance

value corresponding to the weight of the input. The number of inputs is limited by the accuracy of the inverter, as a lot of inputs imply small changes in the voltage on node $n0$. Furthermore, because of the analog nature of the device, for a large number of inputs, the circuit becomes more sensitive to feedback through the inputs.

The *PSF* block has an area cost 12 elements plus one for each input. The delay and energy consumption of the *PSF* block are dependent on the number and the weight of the inputs. For typical implementations the delay ranges from 1.5 ns to 10 ns, while the energy consumption ranges from 8 to 90 meV. Figure 13 presents the delay and energy consumption of a *PSF* block with respect to the input value. The *PSF* block implements a periodic symmetric function with a period of 2, which is also reflected in the delay. The energy consumption is exponential to the input value.

4. Example

In order to demonstrate the possibilities offered by the building blocks discussed in the previous section we present in this section an example circuit. Not only does this example, a 6-bit radix-8 adder, indicate the potential, it also shows the limitations of the building blocks and how this can be dealt with.

The 6-bit radix-8 adder is depicted in Fig. 14 and functions as follows. The inputs A and B are split into groups of three bits for reasons that will become clear soon. For now we focus on the first three bits of both inputs, which are connected each to the V input of an *MVke* block. The built-in constant k of each *MVke* block is set to the weight of the bit connected to its input. Now each $MV2^ie$ block adds $2^i\mathrm{q_e}$ charge to the charge reservoir CR^1 when the corresponding input (a_i or b_i) is logic '1'. Consequently, the charge reservoir CR^1 contains the sum of the values A and B, disregarding the upper three bits.

The three *PSF* blocks perform a conversion from the multi-valued signal in charge reservoir CR^1 to a binary representation using outputs s_0, s_1, and s_2, Each *PSF* block computes a periodic symmetric function on the sum of the addition with a period equal to the weight of the output bit. Therefore, each *PSF* block computes one bit of the output.

The scheme, as explained so far, forms a basic addition scheme that theoretically can add any two numbers. However, our analysis of the *MVke* block, as described in Section 2.3, prescribes a limitation to the number of electrons that the *MVke* block can move and thus limits the bit-size on the inputs. To overcome this problem, addition can be done in high radix.

In our example circuit we chose radix-8 for convenience, although a higher radix would have been possible. For this reason we split each 6-bit input into groups of three bits and added them separately. As Fig. 14 indicates, we added the upper three bits just in the same way we added the lower three bits and converted the sum back to a digital representation by a similar set of *PSF* blocks. To take care of a possible carry, we used an *MCke* (with k=1) block, which adds a charge of $\mathrm{q_e}$ to CR^2 if the value in CR^1 is larger then seven.

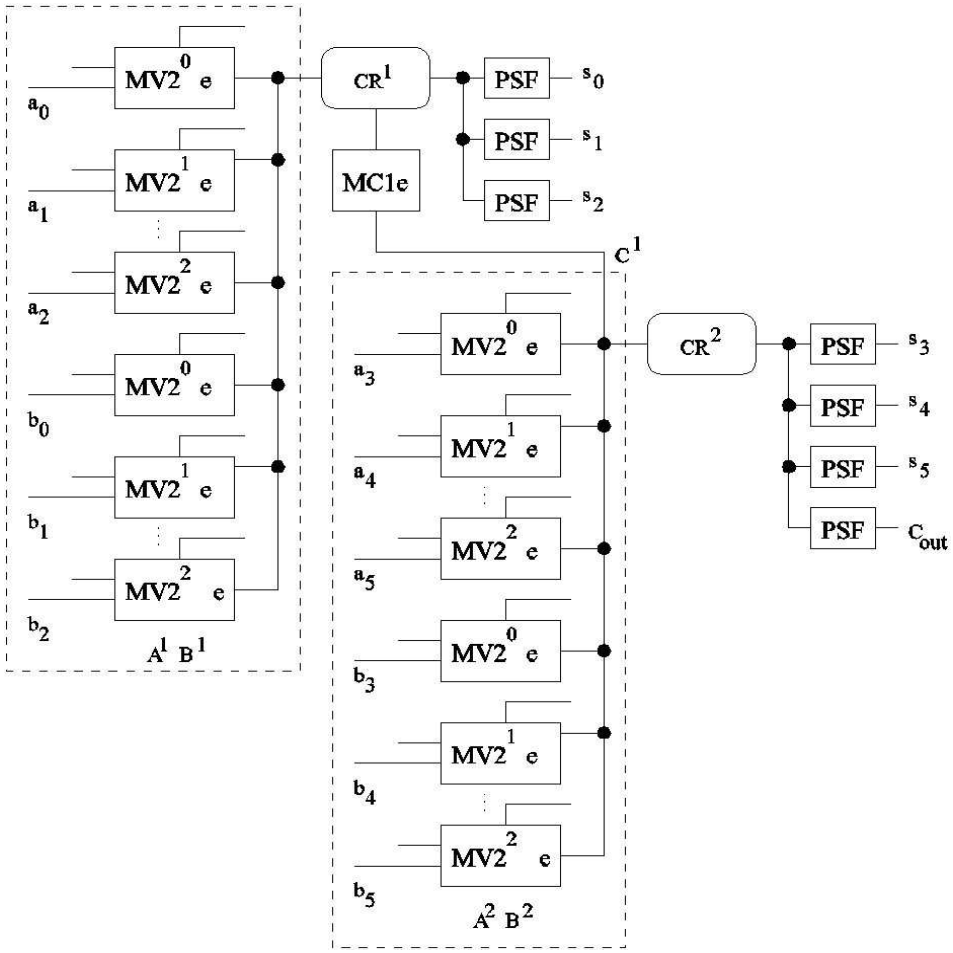

Fig. 14. Organization of the 6-bit radix-8 adder.

We simulated the 6-bit radix-8 addition scheme and the results are depicted in Fig. 15. The two signals in the top block represent reset and enable, respectively. The second two blocks, each containing six signals, represent the input vectors A (0,3,4,7,56,59,60) and B (0,4,4,7,56,60,60), respectively. These vectors were chosen such that each block is tested for its extreme conditions (min/max value, threshold value, etc.) The bottom block, containing seven signals represents the output vector of the adder. For each vector displayed in the graph, the top bar represents the least significant bit while the bottom bar represents the most significant bit.

The simulation indicates that the high radix adder functions correctly. The total area needed for the adder was calculated as 187 circuit elements, the delay as 4.15 ns, and the power consumption as 224 meV.

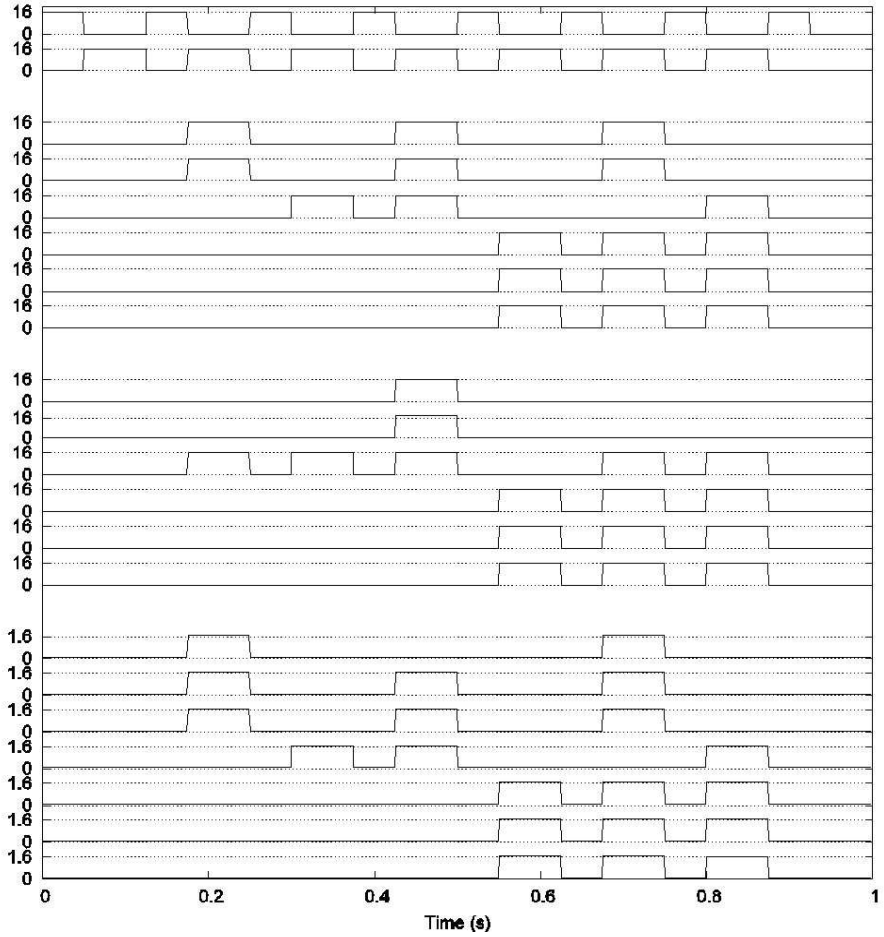

Fig. 15. Simulation results for the 6-bit radix-8 adder.

5. Conclusions

In this paper we presented a set of building blocks, based on SET technology devices, that allows for the efficient implementation of logic and arithmetic structures in SET based new computational paradigms. The set of proposed building blocks can operate on Boolean signals (one electron encoded signals) as well as on discreet analog signals (multiple electron encoded signals) and can perform conversions among them. We analyzed the building blocks in terms of area, delay, and energy consumption and discussed a number of design trade offs. This study provides us the means to evaluate the actual expected performance of SET based schemes for a given fabrication technology and error probability.

References

[1] *International Technology Roadmap for Semiconductors*, 2003 Edition, Executive Summary, Downloadable from website `http://public.itrs.net/Home.htm`, 2003.

[2] WASER, R. (Ed.), *Nanoelectronics and Information Technology – Advanced Electronic Materials and Novel Devices*, 1^{st} ed. Wiley-VCH, Berlin, 2003.

[3] COŢOFANĂ, S., LAGEWEG, C., VASSILIADIS, S., *Addition Related Arithmetic Operations via Controlled Transport of Charge*, IEEE Transactions of Computers, vol. **54**, no. 3, pp. 243–256, March 2005.

[4] LAGEWEG, C., *Single electron tunneling based arithmetic computation*, Ph.D. dissertation, TU Delft, 2004.

[5] ASAHI, N., AKAZAWA, M., AMEMIYA, Y., *Single-Electron Logic Systems Based on the Binary Decision Diagram*, IEICE Transactions on Electronics, vol. **E81-C**, no. 1, pp. 49–56, January 1998.

[6] LAGEWEG, C. R., COŢOFANĂ, S. D., VASSILIADIS, S., *Static buffered SET based logic gates*, in *Proceedings of the 2002 2^{nd} IEEE Conference on Nanotechnology*, August 2002, pp. 491–494.

[7] MEENDERINCK, C. H., COŢOFANĂ, S. D., LAGEWEG, C. R., *High Radix Addition Via Conditional Charge Transport in Single Electron Tunneling Technology*, in *Proceedings of 16^{th} International Conference on Application-Specific Systems, Architectures and Processors*, pp. 294–299, Samos, Greece, July 2005.

[8] DINER, S., FARGUE, D., LOCHAK, G., SELLERI, F., *The wave-particle dualism – a tribute to Louis de Broglie on his 90^{th} birthday*, D. Reidel Publishing Company, 1984.

[9] GARCIA, N., DAMASK, A., *Physics for computer science students*, Springer-Verlag, 1991.

[10] LIKHAREV, K. K., ZORIN, A. B., *Theory of the bloch-wave escillations in small josephson junctions*, Journal of Low Temperature Physics, **59**, pp. 347–382, 1984.

[11] AVERIN, D. V., LIKHAREV, K. K., *Coulomb blockade of single-electron tunneling and coherent oscillations in small tunnel junctions*, Journal of low temperature physics, **62**, no. 3–4, pp. 345–373, 1986.

[12] GEERLIGS, L. J., ANDEREGG, V. F., VAN DER JEUGD, C. A., ROMIJN, J., MOOIJ, J. E., *Influence of dissipation on the coulomb blockade in small tunnel junctions*, Europhysics Letters, **10**(1), pp. 79–85, 1989.

[13] ZWERGER, W., SCHARPF, M., *Crossover from coulomb-blockade to ohmic conduction in small tunnel junctions*, Zeitschrift für Physik B – Condensed Matter, **85**, pp. 421–426, 1991.

[14] WASSHUBER, C., *About single-electron devices and circuits*, Ph.D. dissertation, TU Vienna, 1998.

[15] GEERLIGS, L. J., ANDEREGG, V. F., HOLWEG, P. A. M., MOOIJ, J. E., *Frequency-locked turnstile device for single electrons*, Physical Review Letters, **64**, no. 22, pp. 2691–2694, 1990.

[16] ESTEVE, D., *Transferring electrons one by one, Single Charge Tunneling – Coulomb Blockade Phenomena in Nanostructures* (H. Grabert and M.H. Devoret, eds.), NATO ASI series B, vol. **294**, Plenum Press and NATO Scientific Affairs Division, New York and London, 1992, pp. 109–137.

[17] GRABERT, H., INGOLD, G. L., DEVORET, M. H., ESTEVE, D., POTHIER, H., URBINA, C., *Single Electron Tunneling Rates in Multijunction Circuits*, Zeitschrift für Physics B, – Condensed Matter, **84**, pp. 143–155, 1991.

[18] HEIJ, C. P., HADLEY, P., MOOIJ, J. E., *Single-electron inverter*, Applied Physics Letters, vol. **78**, no. 8, pp. 1140–1142, February 2001.

[19] ISHIBASHI, K., TSUYA, D., SUZUKI, M., AOYAGI, Y., *Fabrication of a Single-Electron Inverter in Multiwall Carbon Nanotubes*, Applied Physics Letters, vol. **82**, no. 19, pp. 3307–3309, February 2001.

[20] `http://www.lybrary.com/simon/`.

[21] LIKHAREV, K., *Single-Electron Transistors: Electronic Analogs of the DC Squids*, *IEEE Transactions on Magnetics MG-23*, pp. 1142–1145, 1987.

[22] LIKHAREV, K., KOROTKOV, A., *Ultradense Hybrid SET/FET Dynamic RAM: Feasibility of Background Charge Independent Room Temperature Single Electron Digital Circuits*, in *Proceedings of the International Semiconductor Device Research Symposium*, Charlottesville, Virginia, October 1995, pp. 355–359.

[23] BANERJEE, S., HUANG, S., ODA, S., *Operation of Nanocrystalline-Silicon-Based Few-Electron Memory Devices in the Light of Electron Storage, Ejection and Lifetime Characteristics*, *IEEE Transactions on Nanotechnology*, vol. **2**, no. 2, pp. 88–92, 2003.