Partially reversible pipelined QCA circuits: combining power and throughput

Marco Ottavi *Member, IEEE,* Sarah Frost-Murphy *Student Member, IEEE,* Erik DeBenedictis *Member, IEEE,* and Peter Kogge *Fellow, IEEE,*

***Abstract*— This paper introduces an architecture for quantum-dot cellular automata (QCA) systems with the potential for high throughput and low power dissipation. The combination of regions with Bennett clocking and of regions used for memory storage combines the low power advantages of reversible computing and the high throughput advantages of pipelining. A simple case study is employed to evaluate the proposed pipelined architecture in terms of throughput and power consumption due to information dissipation.**

I. Introduction

Among the innovative technologies that have been proposed to overcome the limitations of "end of the roadmap" CMOS, Quantum-dot Cellular Automata (QCA) shows features that are very promising to achieve both high computational throughput and low power dissipation. The QCA computational paradigm [1] [2] [3] on one hand introduces highly pipelined architectures with extremely high speeds (in the order of THz) while on the other hand radically departs from switch based CMOS, avoiding the movement of charge from V_{dd} to Ground and the consequent energy dissipation. An operating single cell [4] and a functional logic gate have been demonstrated [5] using metal dot implementations at cryogenic temperatures. Moreover, recent advances in fabrication of molecular scale QCA cells suggest the realizability of QCA cells a few nanometers on a side that would allow room temperature operation.

In addition to having great promise for being small, high speed devices, it has been shown that QCA has great potential for low power operation. The reversible computation paradigm is particularly well suited to QCA since Timler showed that in a clocked, information preserving system, the energy dissipation of the QCA circuit can be much lower than k_BTln2 [6].

Reversible computation is drawing increasing interest as a very low power computation paradigm since it is able to overcome the fundamental power limitation of the current irreversible approach. In fact, it is foreseen [7] that in few decades, the main obstacle to further integration of computing will be thermal factor as bit energies approach the absolute thermodynamic lower bound of $kTln2$, [8], when only the energy associated to the physical information of a bit (physical entropy) will be used to encode each logical bit. This assumption is based on the fact that when destroying the information of a bit, the energy associated with it needs to be irreversibly "thermalized", or converted to thermal energy $Q = T\Delta S = Tk_Bln2$ where T is the temperature, k_B is the Boltzmann's constant and ΔS is the increase in entropy related to the loss of free energy of one bit.

However, information does not need to be destroyed in the course of computation. This was shown in Bennett's seminal work [9]. In short, Bennett's idea was to store intermediate results of computation rather than destroying them. Once the output is computed and saved, the direction of computation can be reversed to decompute the intermediate results. In other words, instead of destroying the intermediate results, they are transformed back into the original input. This approach allows the power consumption to be reduced to arbitrarily low levels but incurs a cost in either computation time or space [10].

The Bennett approach has inspired the introduction of the "Bennett clocking" scheme for QCA [11]. The Bennett clocking scheme will be discussed in detail in section III-A. In brief, the intermediate values of the computation are saved "in place" by leaving the QCA cells that computed the intermediate results locked. The Bennett scheme has very low power consumption and no space redundancy as it does not require any modification at the QCA circuit but introduces a significant time redundancy compared to Landauer clocking, the traditional QCA clocking scheme with inherent fine-grained pipelining.

A QCA circuit using only the Bennett scheme could require an unacceptably long time to produce its outputs. A hybrid solution would be useful to reduce the power consumption without sacrificing high throughput. Such hybrid solution should combine the advantages of pipelining and reversibility to achieve high throughput and low power consumption. This paper proposes such a hybrid design approach for QCA circuits that combines regions of Bennett clocked logic with memory stages to facilitate pipelining. The approach is evaluated in terms of throughput and power consumption due to bit erasures for different pipeline granularities. The power consumption due to the clocking layer is beyond the scope of this work and will be discussed elsewhere.

This paper is structured as follows: section 1 introduces molecular QCA, section 2 presents the proposed approach discussing the details of Computation Stages and Memory stages. Section 3 discusses the parameters used for the performance evaluation while section 4 applies the evaluation to

M. Ottavi, S. Frost-Murphy, E. DeBenedictis are with Sandia National Laboratories, Albuquerque NM.

P. Kogge is with University of Notre Dame, South Bend IN

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energys National Nuclear Security Administration under Contract DE-AC04-94AL85000.

R&A approved SAND Number: 2006-5905J

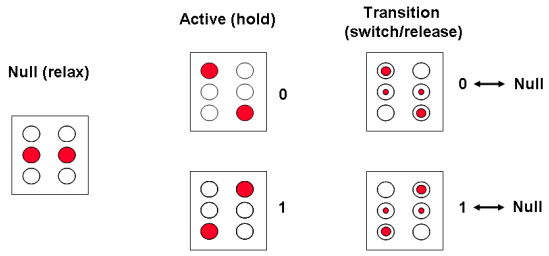

Fig. 1. QCA cell bistable feature

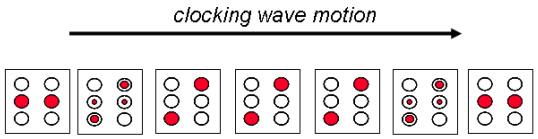

Fig. 2. QCA binary wire

a case studies of a binary tree composed of XOR gates (parity checker) and provides closed form formulas used for the performance evaluation together with some parametric plots to compare Landauer and Bennett clocking schemes. Finally, in section 5 the conclusions are drawn.

II. Molecular QCA

Quantum dot Cellular Automata (QCA) is a computation paradigm based on a cell composed of six quantum dots and two extra charges that can tunnel between the dots of the cell but cannot tunnel outside the cell. The Coulombic repulsion between the extra charges leads to two stable states in which the charges are in antipodal locations along one of the two the diagonals (figure 1). A logical zero corresponds to the configuration where a line through the extra charges has a negative slope. A logical one corresponds to a positive slope. The center dots are used to facilitate clocking of the QCA cell by means of an electric field perpendicular to the plane of the QCA cells. By means of the clocking field, the extra charges can be drawn into the center dots rendering the cell neutral or pushed onto the corner dots forcing the cell to assume a particular value.

Through Coulombic interaction, the information contained in a single cell can be propagated to other cells to form a binary wire (figure 2). The basic logic gate, the majority voter, works in a similar fashion where the output cell assumes the configuration of the majority of the inputs (figure 3). Along with the inverter, this forms a functionally complete logic set.

Figures 2 and 3 also show that the propagation of the signal is carried out through a sequence of the four clock states switch, hold, release, and relax. These clock phases are generated by a traveling wave of E field perpendicular to the QCA plane. In the switch phase, a cell is assuming a new configuration when the charges are moving from the center dots to the corners. In the hold phase, a cell has a definite configuration and can drive the value of neighboring cells. In the release phase the cell is losing its configuration as the extra charges are drawn into the center dots. Finally, when a cell

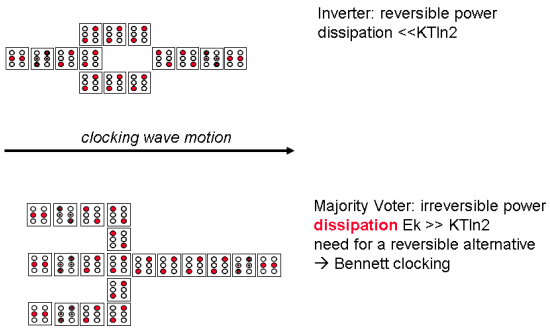

Fig. 3. QCA Gates

is in the relax phase, it cannot influence the configuration of neighboring cells. In the six dot cell, this corresponds to when the extra charges are in the center dots. In figure 2, the cells on the far left and far right of the wire segment shown are in relax. The second cell (from the left) is releasing its value. The third, fourth, and fifth cells have a definite value and are in the hold phase. These cells are driving the sixth cell which is in the switch phase, assuming a new value. Since the seventh cell, the far right cell, is in the relax phase, it has no value and cannot influence the new configuration being assumed by the sixth cell.

The clocking scheme in which this pattern ripples down the wire in one direction is the traditional QCA clocking scheme, Landauer clocking. Its advantage is that can be pipelined at a very fine-grained level. However, while the wire and the inverter are logically reversible functions characterized by low power consumption ($\ll k_B T ln2$) since they do not destroy information, the majority voter is logically irreversible since the minority input is destroyed. The minority input's energy is thermalized, increasing the entropy by $\Delta S = k_B ln2$ and dissipating at least $k_B T ln2$ (approximately the kink energy E_k).

To take advantage of the low power computation potential of QCA, information cannot be destroyed in this manner. Bennett provides a solution to the problem. If the intermediate results, in this case the inputs to the majority gate, are saved, the majority function can be decomputed after its output has been latched. In the context of QCA, this can be done with no space overhead by Bennett clocking the circuit [11].

This approach will form the foundation of the hybrid solution presented here. While Bennett clocking does not incur space overhead, it does entail a performance hit in terms of throughput. The hybrid approach discussed below leverages the strengths of Bennett as well as the strengths of pipelining.

III. Proposed approach

This section reports the proposed approach to obtain high performance in power consumption and throughput. The design is divided into computational and memory stages, the computational stages are clocked with the Bennett scheme and do not dissipate power. The memory stages are used to introduce multiple stages in the circuit and increase throughput. We consider a circuit as being partitioned into M stages where

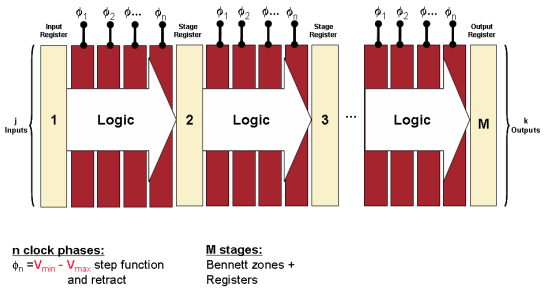

Fig. 4. Proposed pipelined approach: Top view

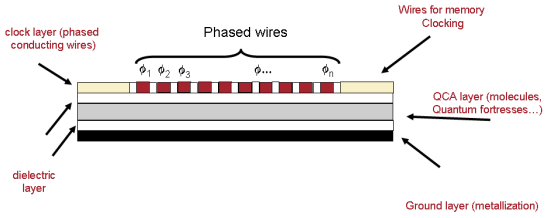

Fig. 5. Proposed pipelined approach: Cross Section

each stage has i_j inputs and o_j outputs. Obviously $i_j = o_{(j-1)}$ since a stage's input is the output from the previous stage.

A. Computation Stages

In this section we describe the clocking scheme for the combinational parts of the circuit. Before describing the adopted clocking scheme we provide some background on the clock distribution techniques and clocking schemes proposed for QCA.

We use the distribution mechanism introduced in [12] where an E field generated on a layer of metallic wires above (or below) the QCA layer controls the tunneling within individual QCA cells. The cells are not directly connected to the clocking circuitry, a substantial advantage when working at the molecular scale. Moreover the continuous transition of the E field on the leading edge of the wave reduces the possibility of kink, or error on a wire.

The traveling E field is generated by providing each of the wires with a voltage phase shifted from the neighbor by $\frac{\pi}{2}$ and having a conducting ground layer on the other side of the QCA layer. Hennessy shows that the E field generated with such a circuit can assume a sinusoidal shape, allowing for Landauer style clocking. The z component of the vector $\vec{E}$ that acts as the clock signal can be described by the wave equation [13]:

$$E_z(x,t) = E_0 cos(\kappa x - \omega t).$$

Computation, the switching of cells, occurs only on the leading edge of the wave, giving the circuit directionality and virtually eliminating the probability of kink. This is a space continuous implementation of the classic four phases-four zones clocking scheme introduced in [2]. This clocking strategy has been called a "traveling wave", a "computational wave"[14], and "Landauer"[6] clocking. Here, in the context of reversibility, we choose to use "Landauer" clocking to describe this clocking approach.

The maximum performance in terms of speed is related to the maximum applicable clock speed and is a consequence of the physics of the tunneling between quantum dots. In order to maintain the adiabatic solution of the Schrödinger equation the switch time t^* of the E field on a cell must be greater than the tunneling speed between quantum dots [2]. Consequently, the fastest applicable clock period on a cell is

$$T_l = 2t^*$$

and therefore $\omega \leq \omega_0 = \frac{2\pi}{2t^*}$. The constraint on the maximum applicable period will be used in a later section to measure the throughput of Landauer clocking for the case studied, generally for o outputs $Tr = \frac{o}{2t^*}$.

The traveling E_z wave is also characterized by its phase velocity

$$v = \frac{\omega}{\kappa} = \frac{\lambda}{2t^*}.$$

Notice that v and λ are directly proportional. For a given t^*, then, the throughput will be constant, but the velocity of the wave will indirectly describe the depth of the pipeline. The wavelength λ is the distance between neighboring regions of active QCA cells. As the distance between active regions grows (i.e. the pipeline depth decreases) and the period remains constant, the phase velocity of the wave increases. In other words, the deeper the pipeline, the slower the wave's velocity.

In this paper we assume the clocking distribution mechanism of [12] is used to produce the Bennett clocking scheme in the computational stages of the pipeline.

The Bennett scheme has two steps: computation and decomputation. In the first step it performs the computation on the inputs and propagates to the outputs without deleting the intermediate results. In the second step the intermediate results are decomputed by the clock "backing off". In other words, the release of the cells starts from the outputs and retreats to the inputs, eventually releasing the whole circuit. This process does not delete any information because every cell that is released can "copy" its contents to the still locked cell that produced the information in the cell being released. This process prevents the information from being thermalized [6].

This discussion leads to several points worth mentioning, some of which have been mentioned explicitly and some only implied: circuits implemented with Bennett clocking do not dissipate energy over the course of a computation/decomputation cycle; at the end of a Bennett region's computation/de-computation cycle, both the original input and the output are stored; and finally, the speed of computation is a consequence of the time required for the clocking signals to propagate back and forth across the region. Further, it is not necessary to make any modifications to an irreversible QCA circuit in order to make it reversible. In this case, reversibility is caused by the clock rather than by the circuit, avoiding the circuit bloating required for reversibility with Landauer clocking.

Moreover circuits clocked with the Bennett scheme have also an important advantage that the circuit does not require any modification to the layout to avoid deleting the information

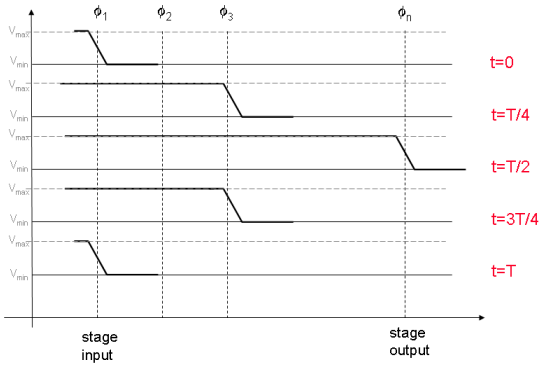

Fig. 6. Clocking wave for the Bennett scheme. T represents the time period of the pipeline stage.

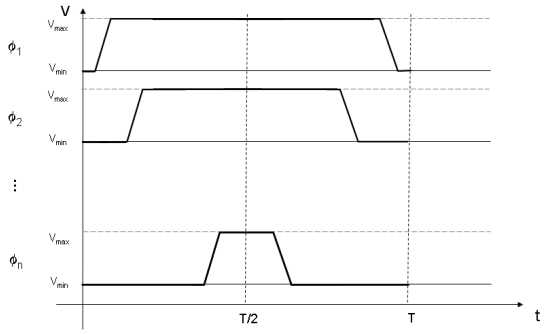

Fig. 7. Clock signal to the buried wires

included in the inputs as it would happen if the inputs needed to be propagated to the outputs. The power dissipated when losing a bit of information is almost equal to the kink energy $E_{diss} \simeq E_k \gg KTln2$. The value of the dissipated energy is obtained from the non equilibrium equation i.e. a set of first-order differential equations for the coherence vector of QCA cells in contact with the thermal environment [13].

The Bennett scheme can be implemented using Hennessy's clocking implementation strategy by applying suitable signals, $\Phi_1 ... \Phi_n$, (figure 7) to the buried wires. The signals needed to produce the Bennett style clock are very different from the signals needed to produce a Landauer style clock. For the Bennett clock, once the QCA cells have been locked, they must remain locked throughout the rest of the computation phase and be released in the decomputation phase as described earlier (figure 6).

The pattern of waveforms present on each wire required to produce this effect can be seen in figure 7. Notice that Φ_i remains high at V_{max} until Φ_{i+1} goes to V_{min}.

With this clocking scheme, data are output from the stage at every period $t = T$, which is the time required for the clock to sweep forward and latch the output and then retract back decomputing all intermediate solutions. An approximate analytical expression describing the clocking wave $E_z(x,t)$ during one period T_b is the following

$$E_z(x,t) = E_0 \left(1 - u_0 \left(\frac{x}{\lambda_c} - tri_{T_b}(t - \frac{T_b}{2})\right)\right)$$

where u_0 is the Heaviside step function tri_T is the triangular function of width T and λ_c is the width of the Bennett-clocked region. The use of u_0 represents an approximation on the use of smooth transitions of duration t^*.

As discussed above, to preserve adiabaticity, the switch time on a cell must be at least t^*. Therefore, considering d the lateral size of a QCA cell, and $N = \lambda_c/d$ the width of the Bennett-clocked region in number of cells, the period can be defined by:

$$T_b = \frac{2\lambda}{d} t^* = 2Nt^*.$$

B. Memory Stages

The memory stages are a single buffer register used to separate the different stages of the pipeline. The memory stages provide the inputs to the Bennett clocked zones and latch their outputs. Their implementation is straightforward as they could in principle be implemented with a single vertical row of QCA cells, or the minimum number of cells related to the achievable pitch of the clocking wires.

In the simplest design, the contents of the latch would be overwritten on each cycle when the new input was stored. This would result in the dissipation of the number of bits stored in each latch multiplied by the number of latches. However, the properties of QCA cells and the clock can be exploited to minimize this dissipation.

As shown in Figure 8 the clocking signal is sinusoidal with the same period T of the Bennett clocking scheme. Rather than using the traditional QCA circuit design methodology where completely locked cells are used to drive the value of its neighboring switching cells that have fully relaxed cells on their other side, an asymmetric interaction is used. In this case, the cells that would normally be in the relax phase (having no value) are instead in the process of releasing their values while the latch is assuming its values. The directionality of the circuit is preserved because the signal from the driver cell is still much stronger than that of the releasing cell. However, if the data being released is the same as the new data being latched, that bit will not be dissipated. Instead, it will be "copied" into the new bit being stored.

The signals applied to each buried clocking wire to achieve this asymmetric interaction are shown in figure 7. Phase Φ_1 of stage $j+1$ releases the information contained in it while the memory stage is switching and the the phase Φ_n of stage j is in hold phase. This should allow the new value to propagate appropriately to the stage j while avoiding the deletion of the information in stage $j+1$ when the value is the same.

A sketch of the propagation in the two opposite directions is shown in figure 9 where two opposite values are interacting on the memory cells in the middle. Since the cell on the left locks its value (attains the hold phase) earlier than the one on the right, the Coulombic interaction (quadripole moment) on the memory cell is stronger and therefore it should cause the memory cell to assume its value.

IV. Performance Evaluation

The performances of the proposed solution are evaluated both in terms of throughput and power consumption. These

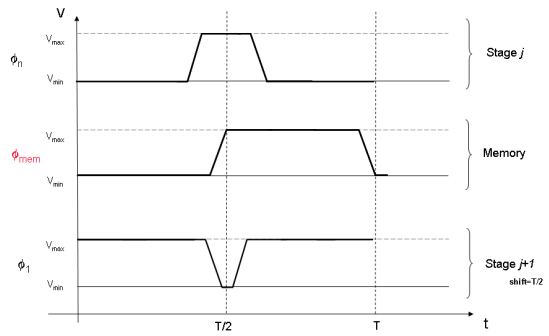

Fig. 8. Clocking signal for the memory zones

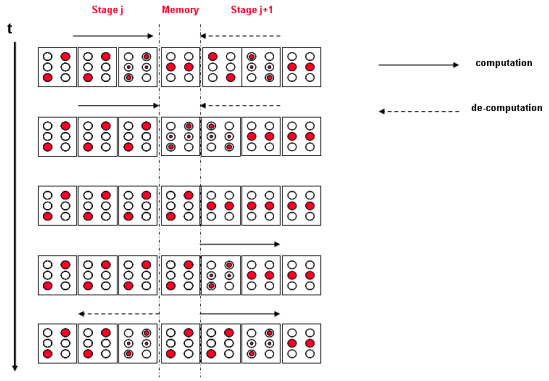

Fig. 9. Asymmetric interaction on the memory cell

two factors are in conflict with each other here. Increasing the number of pipeline stages leads to higher throughput. However, increasing the number of pipeline stages also increases the amount of discarded information, leading to higher power consumption. Therefore the number of stages chosen is the outcome of a trade off between computing performances and power consumption.

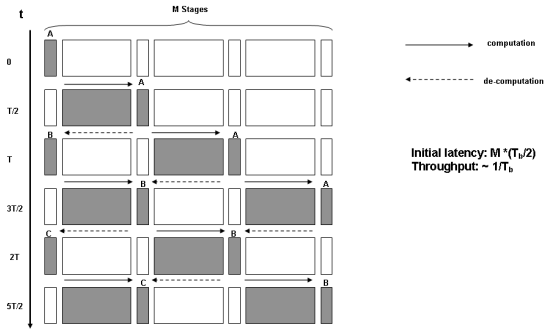

Fig. 10. Pipelined stages with Bennett clocking

Consider a pipelined circuit with no feedback loops composed of M stages and o outputs. Recall that the period of a Bennett stage is $T_b = 2Nt^*$. The throughput is described by:

$$T_r = \frac{o}{T_b} = \frac{o}{2Nt^*}.$$

The initial latency L_b is proportional to $T_b/2$ (figure 10):

$$L_b = \frac{MT_b}{2}$$

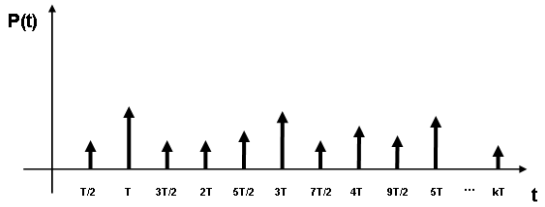

Fig. 11. Possible shape of $P(t)$

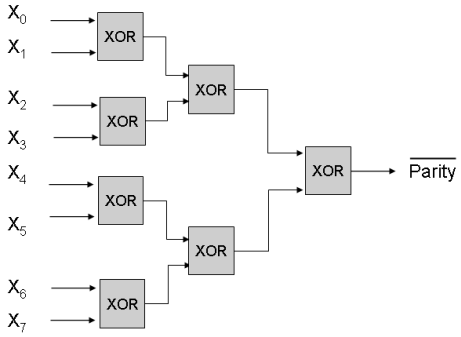

Fig. 12. Case Study: XOR tree parity checker

For the same M staged pipeline the power consumption $P(t)$ can be described as a function of time as follows:

$$P(t) = E_{diss} \cdot \sum_{j=0}^{\infty} \sum_{i=0}^{M-1} K_i(t)\delta(t - \frac{jT_b}{2}) \qquad (1)$$

where $K_i(t)$ is the number of inputs on stage i that change value at time t, E_{diss} is the energy dissipated (thermalized) when a bit is deleted on the stage registers.

The time varying value of $K_i(t)$ accounts for the random time variability of the data in the pipeline on the memory stage i. On average, and it can be thought of as being equal to half of the bits stored in the memory. The power dissipation of a circuit is therefore spatially localized on the memory stages and is a time varying function composed of a train of pulses accounting for the dissipation occurring at the discrete time instants $t = jT/2$, where j is an integer, on the memory stages. Figure 11 shows a possible shape of the P(t) not related to a specific circuit implementation. Note also that, as can be seen in Figure 10 at each $t = jT/2$ the power dissipation occurs only on $\lfloor M/2 \rfloor$ i.e. at the same the deletion of data occurs only in that half of the memory stages where the computing and the decomputing waves meet. Consequently, at a given time $t = nT/2$ the actual number of coefficients $K_i(t) \neq 0$ is $\lfloor M/2 \rfloor$.

V. Case Study: Parity Checker XOR Tree

The size of the zones clocked with the Bennett scheme can vary from a minimum of two QCA cells size (the one cell case would be degenerated into a Landauer scheme and the clock would be a traveling wave) to the size of the whole circuit (thus becoming a purely Bennett clocked circuit). As

stated previously, we expect that by increasing the size of the zones the throughput and the power consumption would both decrease, degrading throughput while improving power consumption.

It should be noticed that the contribution to power dissipation strongly depends on the circuit layout: a circuit composed of only wires and inverters and thus composed of only reversible building blocks, would have the best performances with Landauer clocking as no information would be deleted apart from the I/O whereas a circuit comprising Majority voters would require the introduction of a Bennett scheme to reduce the dissipation due to the deletion of information. To have some advantage in power dissipation a Bennett clocked stage should have a number of MVs big enough such that the number of bits of information that would be deleted in that stage using Landauer clocking is significantly higher than the number of inputs deleted in the Bennett stage. Note that the number of bits deleted in a stage is not necessarily equal to the MVs as shown in the next example.

We show a simple example of the proposed approach. An M stage binary tree composed of XOR gates generates the parity bit for $w = 2^M$ inputs. We report a worst-case analysis of throughput and power dissipation of the XOR based parity bit generator by using the previously introduced formulas.

The same XOR tree is considered with different clocking schemes. Landauer clocking is used to provide an irreversible base case for comparison. For the Landauer clocked case, the throughput is

$$Tr_l = \frac{1}{T_l},$$

where T_l is the period of the Landauer clocking wave, and one result is output on each cycle after the pipeline has been filled.

With the Bennett scheme, throughput and power consumption depend on the period of the Bennett clocked regions. The period depends on the width of those regions. Here, we consider $T_b = 2Nt^*$, where N is the number of XOR gates-wide the region is. Since the same circuit is being compared, there is again one output per clock period. In other words,

$$Tr_b = \frac{1}{T_b} = \frac{1}{2Nt^*}.$$

For both the Landauer and Bennett clocked cases, the worst case dissipation for an XOR gate is $2E_{diss}$. This is arrived at in different ways, though. For the Landauer case, consider the internals of the XOR function. One implementation is

$$XOR(A, B) = (NOT(A)ANDB)OR(AANDNOT(B)).$$

This implementation uses two AND gates and one OR gate (figure 13). At most, the combination of inputs leads to a dissipation of $2E_{diss}$. The Bennett case is simpler in that there are two inputs to each XOR gate. No dissipation will occur within the XOR gate, but the inputs may be written over on the next cycle. This, then, also leads to a worst case dissipation of $2E_{diss}$.

To compare the power performances of a M stages binary XOR tree clocked with the Landauer and Bennett schemes also the following assumptions and definitions are used:

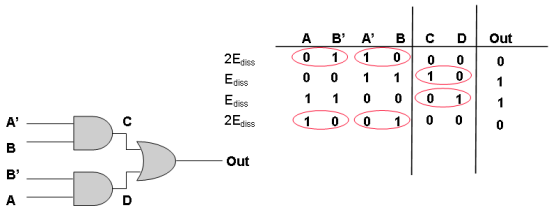

Fig. 13. Dissipation in the XOR gate

1) the dissipated energy of a thermalized bit of information is considered equal to the kink energy, i.e. $E_{diss} \simeq E_k$
2) the kink energy value is $E_k = 3.14577 \cdot 10^{-}20$ Joule obtained for a molecular squared cell of lateral size $l =$ 1.5nm [15] and relative permittivity $\epsilon_r = 1$ (no dielectric material between cells)
3) the number of stages of the XOR tree is k;
4) the number of stages of the pipeline is M;
5) the number of stages of the XOR tree per pipeline stage is $c = \frac{k}{M}$
6) the values of dissipated energy are calculated over the respective period of computation for each scheme, the corresponding power values are considered averaged on the same period
7) since we are considering the worst case scenario, the value of $K_i(t)$ from equation 1 is non time dependent therefore the deleted information is always equal to the number of inputs of stage i

From the previous assumptions and from equation 1 the energy dissipated in a period T_b for Bennett clocked scheme in the XOR binary tree is calculated as follows:

$$\begin{aligned} E_B &= \int_0^{T_b} P_B(t)dt \\ &= 2E_k \int_0^{T_b} \sum_{k=0}^{\infty} \sum_{i=0}^{M-1} K_i(t)\delta(t - \frac{kT_b}{2})dt \\ &= 2E_k \sum_{i=0}^{M-1} K_i(T_b/2) + K_i(T_b) \\ &= 2E_k \sum_{i=0}^{M-1} 2^{ic} \\ &= 2E_k \frac{2^{cM} - 1}{2^c - 1} \end{aligned}$$

where the formula for the sum of a geometric progression of ratio 2^c has been used.

Similarly the energy dissipated in a Landauer clocked binary tree is also the sum on the energy dissipated on the whole

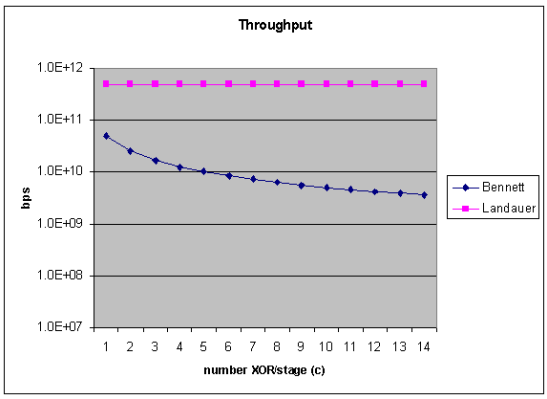

Fig. 14. Throughput comparison

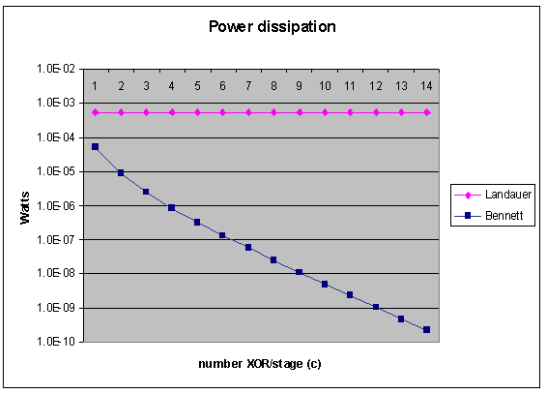

Fig. 16. Comparison of Power dissipation

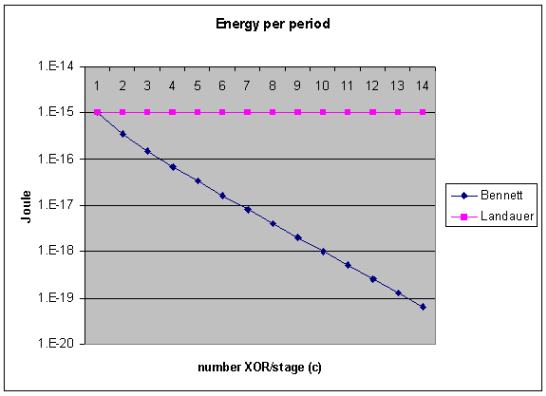

Fig. 15. Comparison of Energy dissipation per period

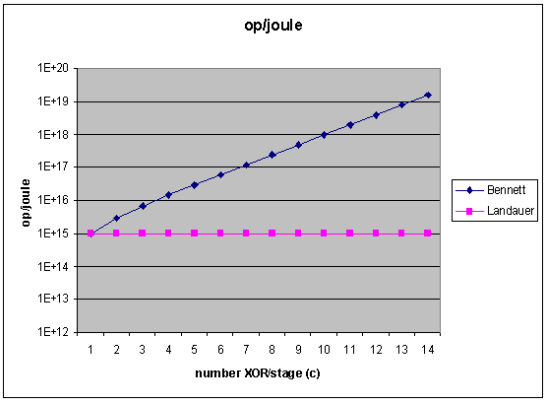

Fig. 17. Comparison of operations per joule

binary tree:

$$\begin{aligned} E_L &= \int_0^{T_l} P_L(t)dt \\ &= 2E_k \sum_{i=0}^{M-1} 2^i \\ &= 2E_k(2^M - 1) \end{aligned}$$

A brief note is called for on the result figures. Figures 14 through 18 show the effect of keeping the circuit size (k) constant and varying the number of pipeline stages (M) and consequently the number of XOR gates per pipeline stage (c). Although only the integer values of c are meaningful, the intermediate values are also shown to better show the trends. In addition, since the parameter c applies only to the Bennett clocked approach, the results for the Landauer clocked approach are constant.

Figure 14 shows a comparison of the throughput in the Bennett and Landauer schemes. As expected, the Landauer scheme shows higher throughput and the gap between the performances increases with the increase of c. In other words, as the pipeline stages get wider and the depth of the pipeline decreases, the throughput decreases.

Figure 15 shows the advantages of the Bennett scheme as a measure of the dissipated energy per period of computation. As the pipeline depth decreases (c increases), the power dissipated per period of operation improves since there are fewer latches whose contents are being dissipated. Notice that even when the entire circuit is in one Bennett stage, the dissipation does not drop to zero because the original inputs are still being deleted every T_b.

Figure 16 compares the power dissipation computed as the ratio between the energy dissipated per period and the time length of a computation period T_l and T_b. Again, the curves show that the power dissipation for Bennett clocking improves with the increase of c.

There is another important perspective from which one could ask questions about power consumption. That is, how much computing can be done given a unit of energy? Figure 17 shows the operations per joule obtained by both XOR tree approaches, considering the "amount of computing" to be the number of output bits obtained. The results again show that the shallow, Bennett clocked pipeline is more energy efficient than the Landauer clocking approach or deeper Bennett pipelines.

Finally figure 18 addresses the question "Given a second of time and a Joule of energy, what is the amount of operations (output bits) obtained?" thus introducing also a time factor to the evaluation of the approaches. In this case the results show an interesting intersection of the two curves introducing a watershed between the values of c for which Landauer clocking has better performances ($c \leq 6$) and the those for which Bennett clocking behaves better ($c > 6$). This result can

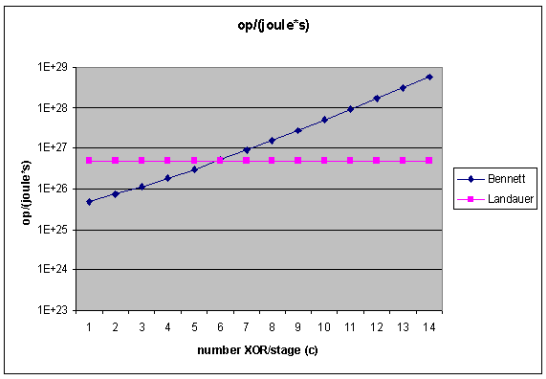

Fig. 18. Comparison of operations per joule per second

be explained as follows: for low values of c the throughput advantages of using a pipelined approach with the Bennett scheme are not sufficient to overcome the penalty in terms of power dissipation, with the increase of the size of the pipeline stages (higher c) the advantages in terms of power dissipation have a bigger impact with respect to the reduction in performances.

VI. Conclusions

This paper introduced a pipelined architecture for low power QCA circuits using the Bennett clocking scheme, a clocking scheme that allows intermediate results to be decomputed rather than erased which avoids power dissipation due to the destruction of information. This architecture allows designers to tune the reversibility of the system based on whether throughput or power consumption are the most important factors.

Combining the Bennett clocking scheme and pipelining introduces flexibility into the design space and allows the tradeoff between power and throughput to be meaningfully analyzed and made. This work provides the metrics by which to evaluate the throughput and power consumption of these two architectural approaches. With careful analysis of the proposed circuit, the Bennett clocked pipeline stages can provide substantial power savings over a Landauer clocked circuit, giving the designer the power to tune the clocking approach to balance constraints on throughput and power consumption.

References

[1] C.S. Lent, P.D. Tougaw, W. Porod, and G.H. Bernstein, "Quantum cellular automata.," *Nanotechnology*, vol. 4, no. 1, pp. 49–57, 1993.

[2] C.S. Lent and P.D. Tougaw, "A device architecture for computing with quantum dots," in *Proc. of the IEEE*, Mar 1997, vol. 85, pp. 541–557.

[3] P.D. Tougaw and C.S. Lent, "Logical devices implemented using quantum cellular automata," in *Journal of Applied Physics*, 1994, vol. 75(3), pp. 1818–1825.

[4] A.O. Orlov, I. Amlani, G.H. Bernstein, C.S. Lent, and G.L. Snider, "Realization of a functional cell for quantum-dot cellular automata," in *Science*, 1997, vol. 277, pp. 928–931.

[5] I. Amlani, A.O. Orlov, G. Toth, G.H. Bernstein, C.S. Lent, and G.L. Snider, "Digital Logic Gate Using Quantum-Dot Cellular Automata," *Science*, vol. 284, no. 5412, pp. 289–291, 1999.

[6] J. Timler and C. Lent, "Maxwell's demon and quantum-dot cellular automata," *Journal of Applied Physics*, vol. 94, no. 2, pp. 1050, 2003.

[7] Michael P. Frank, "Introduction to reversible computing: motivation, progress, and challenges," in *CF '05: Proceedings of the 2nd conference on Computing frontiers*, New York, NY, USA, 2005, pp. 385–390, ACM Press.

[8] MP Frank, "The physical limits of computing," *Computing in Science & Engineering [see also IEEE Computational Science and Engineering]*, vol. 4, no. 3, pp. 16–26, 2002.

[9] C.H. Bennett, "Logical reversibility of computation," *IBM J. Res. Develop*, vol. 17, no. 6, pp. 525–532, 1973.

[10] C.H. Bennett, "Time/space trade-offs for reversible computation," *SIAM Journal on Computing*, vol. 18, no. 4, pp. 766–776, 1989.

[11] C.S. Lent, S.E. Frost, and P.M. Kogge, "Reversible computation with quantum-dot cellular automata (QCA)," *Proceedings of the 2nd conference on Computing frontiers*, pp. 403–403, 2005.

[12] K. Hennessy and C.S. Lent, "Clocking of molecular quantum-dot cellular automata," in *JOURNAL OF VACUUM SCIENCE and TECHNOLOGY B*, 2001, vol. 19(5), pp. 1752–1755.

[13] CS Lent, Liu M., and Lu Y., "Bennett clocking of quantum-dot cellular automata and the limits to binary logic scaling," *Nanotechnology*, vol. 17, pp. 4240–4251, 2006.

[14] EP Blair and CS Lent, "An architecture for molecular computing using quantum-dot cellular automata," *Nanotechnology, 2003. IEEE-NANO 2003. 2003 Third IEEE Conference on*, vol. 1, 2003.

[15] SE Frost, TJ Dysart, PM Kogge, and CS Lent, "Carbon nanotubes for quantum-dot cellular automata clocking," *Nanotechnology, 2004. 4th IEEE Conference on*, pp. 171–173, 2004.