Quantum Dot Cellular Automata Check Node Implementation for LDPC Decoders

Quantum dot Cellular Automata Check Node Implementation for LDPC Decoders

M. Awais, M. Vacca, M. Graziano, M. Ruo Roch and G. Masera

***Abstract*—Quantum dot Cellular Automata (QCA) is an emerging nanotechnology that has gained significant research interest in recent years. Extremely small feature sizes, ultra low power consumption and high clock frequency make QCA a potentially attractive solution for implementing computing architectures at the nano-scale. To be considered as a suitable CMOS substitute, the QCA technology must be able to implement complex real time applications with affordable complexity. Low Density Parity Check (LDPC) decoding is one of such applications. The core of LDPC decoding lies in the check node (CN) processing element which executes actual decoding algorithm and contributes towards overall performance and complexity of LDPC decoder.**

This work presents a novel QCA architecture for partial parallel, layered LDPC check node. The check node executes Normalized Min Sum decoding algorithm and is flexible to support check node degree dc up to 20. The check node is constructed using a VHDL behavioral model of QCA elementary circuits which provides a hierarchical bottom up approach to evaluate the logical behavior, area and power dissipation of whole design. Performance evaluations are reported for the two main implementations of QCA i.e. molecular and magnetic.

***Index Terms*—QCA, Low Density Parity Check codes, Channel Decoder, VHDL, check node.**

I. Introduction

Fundamental physical limits of current CMOS technology have made further miniaturization of CMOS devices problematic. This difficulty has led researchers to probe into possible alternatives to CMOS as summarized in ITRS report [1]. One potential approach currently being investigated is the Quantum dot Cellular Automata (QCA) [2] which is an application to microelectronics of the original cellular automata principle [3]. At present, two are the appealing implementations of QCA: molecular QCA [4] built using complex molecules with many oxide-reduction centers and magnetic QCA [5] based on single domain nanomagnets with only two stable magnetization states. Both implementations are promising in terms of high density, fast switching speed (1 THz for molecular QCA [6]) and low power [7]. Unlike conventional logic circuits which rely on the conduction principle, QCA operates by the Coulomb (molecular QCA) or magnetostatic (magnetic QCA) interaction that connects the state of two neighbor cells.

Many QCA circuits have been proposed in literature e.g. simple blocks like multiplexers [8], more complex arithmetic circuits like adders and multipliers [9] or dividers [10], sequential circuits like latches [11] and memories [12]. However, only facing the design of complex circuits we can pinpoint the real positive and negative aspects of this technology as a possible CMOS substitute. One application from forward error correction domain is the decoding of Low Density Parity Check (LDPC) codes. LDPC decoder implementation using CMOS VLSI technology is a challenging task mainly because of the computationally intensive nature of decoding algorithm and the severe requirements imposed by applications in terms of throughput, power, error rate, decoding efficiency and run time flexibility to support multiple codes.

Most of the circuits that constitute state of the art for QCA are made using QCAdesigner [13], which allows physical placement of individual cells. However many of the proposed circuits do not take into account the physical feasibility of the clock system (see section II for clock description). An alternative approach is the use of a QCA model written using VHDL language [14]. This model not only provides an easy description of complex circuits but also allows power and area estimation keeping in view the actual implementations of QCA i.e. magnetic and molecular QCA [15] [16].

In this work we propose a QCA architecture of the check node (CN), the processing core of an LDPC decoder (see section III for background on LDPC decoders). We adapt the decoding architecture to the specific characteristics of QCA technology, by exploiting majority voting circuits and inherent delaying and pipelining behavior of wires (see Section II for introduction to QCA). We are able to realize a fully pipelined architecture of partial parallel CN, reconfigurable to support up to check node degree $dc = 20$ (section IV). The circuit is described using a realistic layout aware VHDL model (section V) which allows the circuit simulation and performance estimation for both magnetic and molecular QCA. Simulation results show that remarkable area saving and high throughput could be achieved for molecular QCA implementation, while magnetic QCA is attractive for achieving low power. For both cases, the proposed design has an area fairly smaller and clock speed comparable or much larger than its implementation on up to date CMOS technology.

II. QCA background

A. Basic Structures

QCA is based upon the encoding of binary information in the charge configuration within quantum dot cells. An ideal QCA cell can be viewed as a square, in which a charge container or "quantum dot" is placed at each vertex. At ground state (equilibrium), the cell contains a bunch of extra electrons that are confined in the cell but can quantum mechanically tunnel between the dots. The electrons occupy the two dots on the diagonal due to Coulomb repulsion. As there are two possible diagonals (polarizations), two ground states are possible and represent logic '0' and '1' as shown in Fig. 1.A.

M. Awais, M. Vacca, M. Graziano, M. Ruo Roch and G. Masera are with the Electronics and Telecommunications Department of Politecnico di Torino, Italy. E-mail: mariagrazia.graziano@polito.it.

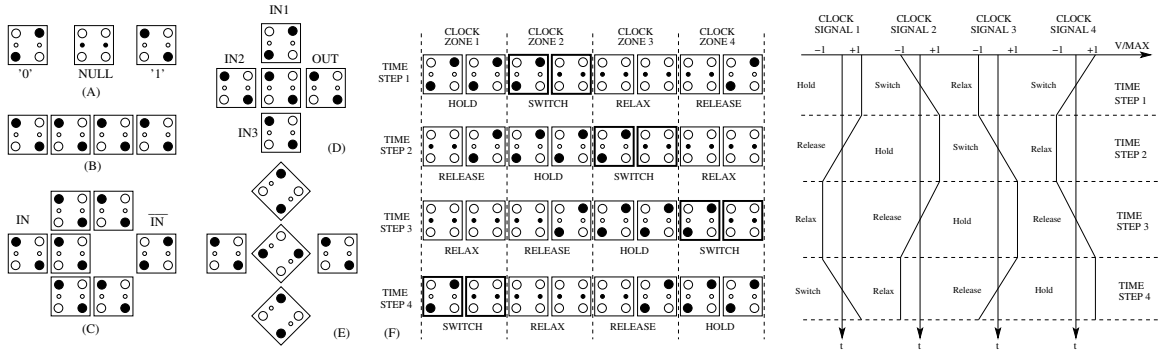

Figure 1: Quantum dot Cellular Automata. A) 6 dot QCA cells representing logic value '0' ,'1' and an intermediate 'Null' state B) QCA wire. C) Inverter. D) Majority voter. E) Crosswire. F) Four phase clocking scheme.

The basic data propagation structure in QCA is the binary wire made by placing side by side, in a horizontal or vertical way, the base cells as shown in Fig. 1.B. One of the primitive logic gates in QCA is the majority voter [17]. The majority voter with logic function *MAJ (A,B,C)=AB+AC+BC* can be implemented with only five QCA cells as shown in Fig. 1.D. Logical AND and OR functions can be realized by setting an input of the majority voter, permanently to a '0' and '1' value respectively. The inverter is the other basic gate shown in Fig. 1.C. This implementation exploits the diagonal coupling of the cell that forces the opposite cell state. Because the diagonal coupling entails a weaker electrostatic interaction than the horizontal or vertical one, two arms are used in this case. In each arm the input data is doubled before being inverted in order to obtain a stronger electrostatic interaction, hence a more reliable structure. Coplanar wire crossing is one of the most elegant features of QCA technology. As shown in Fig. 1.E, QCA cross wire is obtained by overlapping two wires, where the cells of the vertical wire are rotated by 45 degrees.

B. Clocking in QCA

Regardless of implementation, a system made up of QCA devices will realistically require some clock mechanism providing in addition to timing and synchronization a true power gain [18]. Here we briefly discuss how a clock structure facilitates computation of a system of QCA devices.

A clocked QCA cell is a six dot cell (Fig. 1.A) that besides logic '0' and '1' also has a 'Null' state. The 'Null' state is the halfway unstable state useful in so called 'adiabatic switching'. Before every transition, the cell is moved to the intermediate unstable 'Null' state by means of an external potential (magnetic or electric) applied to central dots; in this way the potential barrier is reduced and even a little field (like the one produced by adjacent cell) is sufficient to force a switching. This is accomplished by using four distinct and periodic phases $(0, \frac{\pi}{2}, \pi, \frac{3\pi}{2})$ of a reference clock signal as shown in Fig. 1.F. A QCA circuit is partitioned into a number of clock zones where adjacent clock zones have a $\pi/2$ phase shift between them and every fourth clock zone will have the same applied signal (Fig. 1.F). The four phases are Relax, Switch, Hold and Release. During the Relax phase, there is no inter dot barrier and cell remains unpolarized. During Switch phase, the inter dot barrier is slowly raised and cell attains a polarization under the influence of its neighbors. In the Hold phase, barriers are high and cell retains its polarity acting as an input to the neighboring cells. Finally, in the Release phase, barriers are lowered and cell looses its polarity. In Fig. 1.F, cells of clock zone 1 are in the Hold state, at time step 1, and they act as drivers for cells in clock zone 2. At the same time, cells in clock zone 3 are in the Relax state and have no driving capability. So the cells of clock zone 2 switch under the influence of cells at clock zone 1. At the next time step the situation is repeated but the switching zone is the next in the chain. In this way signal flow takes place along the circuit at the rate of one clock zone per cycle. This clocking mechanism is responsible for inherent pipelined behavior of QCA and multi bit information transfer through signal latching. A signal is effectively "latched" when one clock zone goes into Hold phase and acts as an input to the subsequent zone. Aside from original proposal [2], two are the appealing implementations of QCA i.e. magnetic QCA (MQCA) and molecular QCA.

C. Magnetic QCA

In the magnetic QCA (MQCA) [5] or NanoMagnet Logic (NML), the base cell is a single domain nanomagnet, asymmetric in shape with high aspect ratio. The nanomagnet can be magnetized only in two possible ways encoded as two possible logic states i.e. '0' and '1' as shown in Fig. 2.A. The magnetization vector is parallel to the long side, the so called easy axis. The two stable magnetization states are separated by a high potential barrier. Therefore cell switching from one state to the other one requires a clock which in MQCA case is a strong magnetic field, applied along the short side of the nanomagnets, the so-called hard axis. When this field is applied nanomagnets are forced into an unstable state, with the magnetization directed along the hard axis. As soon as the magnetic field is removed, nanomagnets reorganize themselves in an anti ferromagnetic order. The magnetic field is normally generated by a current flowing through a wire placed under the magnets plane (Fig. 2.E).

Basic NML devices are similar to the generic versions described above, with a few differences. For example, in the horizontal MQCA wire adjacent cells alternate their polarization state as shown in Fig. 2.B. Therefore, the NML

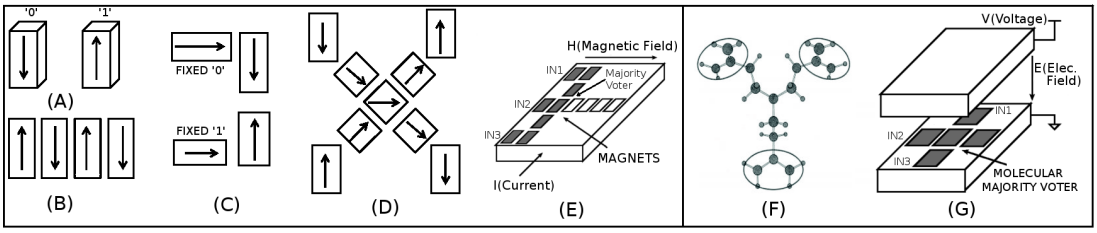

Figure 2: Magnetic QCA (from A to E) and Molecular QCA (F-G). A) Single domain nanomagnets with two stable magnetization states encoded as logic '0' and '1'. B) MQCA wire. C) Fixed magnetization cells. D) Crosswire. E) MQCA majority voter gate : input and output cells must be in the same clock zone. Clock is a strong magnetic field produced by current flowing through a nanowire placed under the magnets plane. F) Example of a possible molecule for QCA application [19]. G) Molecular QCA majority voter, clock is a strong electric field perpendicular to molecular plane.

horizontal wire is equivalent to a chain of inverters. This peculiarity introduces a constraint when sizing the clock zone because the number of cells along the horizontal direction inside the clock zone must be even. Figure 2.D shows a possible implementation of MQCA cross wire [5]. It is made by smaller square cells with the side length between 50 nm and 100 nm. These cells have more stable states with respect to the asymmetric shape cells in particular, the central block of the cross-wire has four stable states. For this reason it can receive simultaneously the two inputs coming from the two wires and propagate them correctly. Fixed polarization cells (see Fig. 2.C) can be implemented with nanomagnets placed in horizontal way and are used as fixed input to a '0' or '1' logic values. During the initial reset they are magnetized along the longest side with the same direction as the clock magnetic field (conventionally with right direction) and because it is a stable state it remains in this state even if the clock field is no longer present. Figure 2.E shows the magnetic implementation of the MAJ and the generation of the magnetic field using a current flowing through a wire placed under the magnets plane.

D. Molecular QCA

A proposed but not yet implemented method is building QCA devices from single molecules. Such an implementation offers some exciting features e.g. remarkably low dimensions (1-2 nm [20]), very high switching speeds (1THz [20]), operation at room temperature, highly symmetric QCA cell structure [21] and possibility of mass-producing devices by means of self-assembly. A single-molecule implementation of a QCA cell requires a molecule with peculiar features: in this kind of molecule, charge is localized in a few oxide-reduction (redox) sites; moreover tunneling is possible among those sites because of bridging ligands. Therefore, redox sites act as quantum dots, able to encode and propagate information. Recent experiments suggest to use mixed valence complexes to construct QCA cells where non bonding orbitals (π or d) act as dot sites for a QCA molecule. Several works have been reported in literature to synthesize molecules acting as QCA cells. In [19], the authors present an analysis of a simple molecular system *Molecule 1*, a three dot molecule composed of three allyl groups connected by alkyl bridges in a V shape as shown in Fig. 2.F. The work in [22] presents a two redox center molecule attached on a Silicon substrate. The quantum dots in the molecule are ferrocene and $Ru(dppm)_2$ groups, while the tunneling junction for the mobile electron is provided by the Carbon-Carbon triple bond. Two molecules form a four-dot QCA cell. [22] also presents four redox center molecule in which two mobile electrons can tunnel through the $(\eta 4{-}C_4)Co(\eta 5{-}C_5H_5)$ group. Recently few more molecules have been proposed [23] [24] that can act as clocked QCA and experimental steps have been discussed [23] [25].

Also for molecular QCA a clock is necessary. This is done by applying an electrical field perpendicular the molecular plane. The electrical field can be generated applying a variable voltage to two electrodes, placed over and under the plane (Fig. 2.G shows an example of clocked majority voter). Wiring requirements for the generation of electrical field impose an upper limit to density of QCA systems. Metallic single walled carbon nano tubes (SWNTs) are expected to be excellent conductors and can be used to generate a clocking field that smoothly propagates the QCA signals [26], once fabrication process is reliable enough.

III. Introduction to LDPC codes and decoding

Low Density Parity check codes [27] are a forward error correction (FEC) technique adopted in a number of advanced applications ranging from wireless communications (e.g. WiMax [28] and DVB-S2 [29]) to multi-level flash memories. Hardware implementation of a generic LDPC decoder consists of two types of processing elements i.e. Variable (or bit) Nodes (VNs) and Check Nodes (CNs). The interconnection between CNs and VNs is determined by a parity check matrix H which is specified in the standards. The LDPC decoding is an iterative process which involves two way exchange of soft information messages (often expressed as Logarithmic Likelihood Ratio LLR) between CNs and VNs. Check node is the most important part of the whole LDPC decoder since it executes the actual decoding algorithm. The state of the art for LDPC decoding consists of standard Belief Propagation algorithm [27] which is optimal yet very complex and its several sub optimal approximations. One such approximation to BP is Normalized Min Sum (NMS) [30] algorithm which provides a good trade off between complexity and error correction capability of LDPC decoder. The order in which CNs and VNs are updated is called schedule [31]. The standard schedule is the flooding schedule [27] which involves CNs

update followed by VNs update. Layered message passing (LMP) [32] is another schedule which merges the CN and VN updates into a single CN update and as a consequence, requires 50% less iterations to meet a certain bit error rate (BER). This work considers NMS algorithm with LMP schedule. The update rule for proposed CN is described in Algorithm 1.

Algorithm 1 Update rule for NMS-LMP check node

$\forall\ (CN_c, VN_i)\ :\ c \epsilon \{1, \cdots, M\}$ and $i \epsilon \{1, \cdots, N\}$ do

$$M^n_{i,c} = SO^{n-1}_i - M^{n-1}_{c,i} \tag{1}$$

$$sgn(M^n_{c,i}) = \prod_{i' \epsilon V_c/i} sgn(M^n_{i',c}) \tag{2}$$

$$|M^n_{c,i}| = \alpha \min_{i' \epsilon V_c/i} M^n_{i',c} \tag{3}$$

$$SO^n_i = M^n_{i,c} + M^n_{c,i} \tag{4}$$

M and N are the numbers of CNs and VNs respectively, index i $(i = 1, 2, ..., d_c)$ runs over incident CN edges, where d_c is the total number of incident edges i.e. total number of VNs connected to a CN, $M^n_{i,c}$ and $M^n_{c,i}$ are the VN-to-CN and CN-to-VN messages respectively for an edge i at iteration n. SO^n_i is the posterior LLR value of an edge i at iteration n, $sgn(x)$ denotes the sign of x, $\min$ denotes the minimum, Vc/i denotes the set of all edges except the edge i and α is the positive normalization constant less than 1.

CMOS implementations of most LDPC decoders feature serial CNs where one out of d_c incoming messages is received per clock cycle and its magnitude is compared with the previous minimum value given as a feed back to the comparator, then the message is updated based on the comparator decision. This approach is not feasible for two reasons. Firstly, the read and write latency of check node is directly proportional to d_c making its use prohibitive for large values of d_c. Secondly, due to the inherent pipelined behavior of QCA technology, the presence of feedback signals in the circuit leads to relevant performance drops. This work proposes as a novel contribution a partial parallel check node which processes 4 VN-to-CN messages per clock cycle and provides flexibility to support up to $d_c = 20$. This design also avoids feedback path which results in significant reduction in CN latency.

IV. Proposed QCA Check Node Architecture

The proposed CN architecture is shown in Fig.3. The CN is able to process four incident edges at a time. The subtractors on the left execute equation (1) while the adders on the right execute equation (4). The central part consists of several functional units that execute the NMS algorithm (3). These functional units are described below.

A. Two Min. Extractor (TME)

In the NMS algorithm, out of all incoming LLRs of a check node, only two values are of interest, i.e. the overall minimum $Min1$ and the second minimum $Min2$ which are then normalized by a positive constant α. Therefore, the first step is to obtain $Min1$ and $Min2$ which is done in TME block. Figure 3.A shows the detailed architecture of TME block. The simplest functional unit of TME is the CMP, which is an n-bit binary comparator that receives two inputs A and B and returns a logic one if $A > B$. In CMOS technology, a comparison function is easily implemented with the help of ripple carry adder (RCA) which has $O(n)$ delay. However RCA based comparison is not efficient for QCA implementation because of the inherent pipelined behavior of QCA gates and even wires, which directly impacts the overall delay of boolean function. In addition, direct implementation of RCA adders in QCA leads to poor utilization of majority voters and hence significant area overhead.

In order to achieve an optimal delay with efficient majority voter utilization, we adopted a recursive comparison technique as described in [33]. Suppose two n-bit operands A and B are partitioned as $A = [A_1|A_0]$ and $B = [B_1|B_0]$ where A_1 and B_1 are the most significant parts of A and B respectively, while A_0 and B_0 are the least significant parts. The condition $A \geq B$ is true if $A_1 > B_1$ or simultaneously $A_1 = B_1$ and $A_0 \geq B_0$. Similarly $A > B$ is true if $A_1 > B_1$ or simultaneously $A_1 \geq B_1$ and $A_0 > B_0$. Defining intermediate logical variables $p_i = (A_i > B_i)$, $q_i = (A_i \geq B_i)$ and $(A_i = B_i) = q_i\overline{p}_i$ where $i = 0, 1$, we obtain $A \geq B = p_1 + (q_1\overline{p}_1)q_0$ and $A > B = p_1 + q_1p_0$. It has been shown in [33] that $Q = (A \geq B) = MAJ(p_1, q_1, q_0)$ and $P = (A > B) = MAJ(p_1, q_1, p_0)$. The partitioning procedure is recursively repeated up to single bit partitions. In this way, an 8-bit comparator (CMP) is implemented with inputs $a_{[7:0]}$ and $b_{[7:0]}$ as shown in Fig. 3.D. The architecture can be described by the following set of equations.

$$\begin{aligned}
p_{[7:0]} = (a_{[7:0]} > b_{[7:0]}) &= MAJ(p_{[7:4]}, q_{[7:4]}, p_{[3:0]}) \\
p_{[3:0]} &= MAJ(p_{[3:2]}, q_{[3:2]}, p_{[1:0]}) \\
q_{[7:4]} &= MAJ(p_{[7:6]}, q_{[7:6]}, q_{[5:4]}) \\
p_{[7:4]} &= MAJ(p_{[7:6]}, q_{[7:6]}, p_{[5:4]}) \\
p_{[1:0]} = MAJ(p_1, q_1, p_0); p_{[3:2]} &= MAJ(p_3, q_3, p_2) \\
q_{[3:2]} = MAJ(p_3, q_3, q_2); q_{[5:4]} &= MAJ(p_5, q_5, q_4) \\
q_{[7:6]} = MAJ(p_7, q_7, q_6); p_{[7:6]} &= MAJ(p_7, q_7, p_6) \\
p_{[5:4]} &= MAJ(p_5, q_5, p_4) \\
p_0 = a_0\overline{b}_0; p_1 = a_1\overline{b}_1; p_2 &= a_2\overline{b}_2; p_3 = a_3\overline{b}_3; \\
p_4 = a_4\overline{b}_4; p_5 = a_5\overline{b}_5; p_6 &= a_6\overline{b}_6; p_7 = a_7\overline{b}_7; \\
q_0 = a_0 + \overline{b}_0; q_1 = a_1 + \overline{b}_1; q_2 &= a_2 + \overline{b}_2; q_3 = a_3 + \overline{b}_3; \\
q_4 = a_4 + \overline{b}_4; q_5 = a_5 + \overline{b}_5; q_6 &= a_6 + \overline{b}_6; q_7 = a_7 + \overline{b}_7;
\end{aligned}$$

The CMP2 block is a simple compare-select circuit where the output flag of CMP is used to select from the two inputs the first minimum and the second minimum [34] as shown in Fig. 3.C. In order to synchronize the signal flow according to actual pipelined behavior of QCA wires and gates, several delay blocks are introduced (see section V for more details). Each delay block consists of a number of pipelined registers associated with clock zones between the input and output. The registers are clocked according to four phase clocking scheme. The four input comparator CMP4 (Fig. 3.B) includes five CMP2 units organized into three stages. The whole TME unit consists of a five stage chain of CMP4 units. Due to inherent pipelined behavior of QCA circuits, each stage is

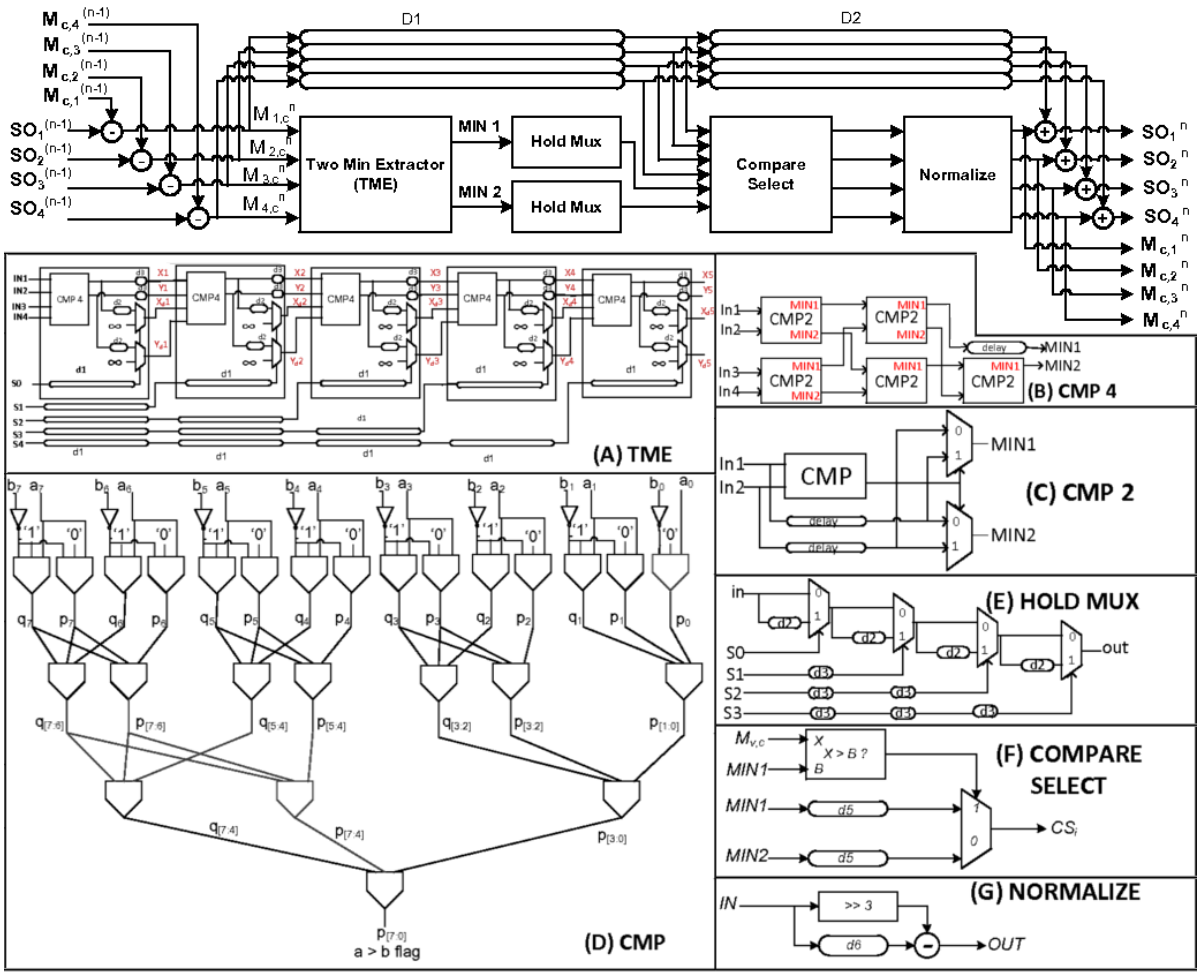

Figure 3: Proposed Check Node (Top Level). A) TME. Main component of the check node (CN). B) CMP4: TME sub block. C) CMP2: CMP4 sub block. D) CMP: CMP2 basic block. E) HOLD. F) COMPARE SELECT. G) NORMALIZE. E, F and G are among the main components of CN.

able to receive (generate) a set of four inputs (outputs) per macro clock cycle (4 phases). For each stage i, the outputs are X_i, Y_i, X_di, Y_di, where $i = 1, \cdots, 5$. X_i and Y_i are the direct outputs of CMP4 block and represent the two minimum values extracted from the current set of inputs, while X_di and Y_di are the outputs of CMP4 delayed by one macro clock cycle and represent the two minimum values derived from the previous set of inputs. $S0 - S4$ are the select lines of multiplexers which control the propagation of signals (X_di and Y_di) from one CMP4 unit to the following one. The notation ∞ is used to denote the maximum value that could be represented by using n-bits. The complete operation of TME is described in Table I where rows represent the values of intermediate variables with respect to time and d_c. Following notations are adopted, let s denotes a transaction i.e. sequence of complete inputs/outputs, $\{m, n\}_s$ denotes the set containing inputs m to n of transaction s, $M\{m, n\}_s$ denotes the set containing $Min1$ and $Min2$ of set $\{m, n\}_s$, $\{Xi, Yi\}$ and $\{X_di, Y_di\}$ represent the sets containing intermediate signals Xi, Yi and X_di, Y_di respectively. For example, with $d_c = 8$, each input transaction consists of 8 values which are applied in two sets i.e. sets $\{1, 4\}$ and $\{5, 8\}$; the first set contains inputs 1 to 4 and is applied at time step n, while the second set contains inputs 5 to 8 and is applied at time step $n+1$. The output for each transaction consists of two sets $M\{1, 4\}_s$ and $M\{1, 8\}_s$ where the latter represents the valid output i.e. $Min1$ and $Min2$ of inputs 1 to 8.

B. Hold Mux

As shown in Table I, for all values of d_c, an input transaction 's' takes $\frac{dc}{4}$ clock cycles whereas the corresponding valid output remains stable for only one clock cycle. In order to produce correct results, the valid $Min1$ and $Min2$ values of a transaction must remain stable for $\frac{dc}{4}$ clock cycles in order to be synchronized with the corresponding inputs to the compare select processing unit. This is achieved by using a multistage multiplexer "Hold Mux" (Fig. 3.E). Each stage consists of a 2x1 multiplexer which receives a direct and a delayed value of same input, where delay is equal to 1 macro clock cycle (4 zones). Table II shows the sequence of values for control word S[3:0] as a function of time and d_c.

Time		1	2	3	4	5	6	7	8	9	10	Remarks
$d_c = 4$	inputs	$\{1,4\}_1$	$\{1,4\}_2$	$\{1,4\}_3$	$\{1,4\}_4$	$\{1,4\}_5$	$\{1,4\}_6$	$\{1,4\}_7$	$\{1,4\}_8$	$\{1,4\}_9$	$\{1,4\}_{10}$	$\{Xi, Yi\} = \{X1, Y1\}$, $\{X_di, Y_di\} = \{X_d1, Y_d1\}$ $i = 1, 2, 3$
	$\{X1, Y1\}$	$M\{1,4\}_1$	$M\{1,4\}_2$	$M\{1,4\}_3$	$M\{1,4\}_4$	$M\{1,4\}_5$	$M\{1,4\}_6$	$M\{1,4\}_7$	$M\{1,4\}_8$	$M\{1,4\}_9$	$M\{1,4\}_{10}$	
	$\{X_d1, Y_d1\}$	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	
	$\{X5, Y5\}$	$M\{1,4\}_1$	$M\{1,4\}_2$	$M\{1,4\}_3$	$M\{1,4\}_4$	$M\{1,4\}_5$	$M\{1,4\}_6$	$M\{1,4\}_7$	$M\{1,4\}_8$	$M\{1,4\}_9$	$M\{1,4\}_{10}$	
$d_c = 8$	inputs	$\{1,4\}_1$	$\{5,8\}_1$	$\{1,4\}_2$	$\{5,8\}_2$	$\{1,4\}_3$	$\{5,8\}_3$	$\{1,4\}_4$	$\{5,8\}_4$	$\{1,4\}_5$	$\{5,8\}_5$	$\{Xi, Yi\} = \{X2, Y2\}$ $\{X_di, Y_di\} = \{X_d2, Y_d2\}$ $i = 3, 4$
	$\{X1, Y1\}$	$M\{1,4\}_1$	$M\{5,8\}_1$	$M\{1,4\}_2$	$M\{5,8\}_2$	$M\{1,4\}_3$	$M\{5,8\}_3$	$M\{1,4\}_4$	$M\{5,8\}_4$	$M\{1,4\}_5$	$M\{5,8\}_5$	
	$\{X_d1, Y_d1\}$	∞	$M\{1,4\}_1$	∞	$M\{1,4\}_2$	∞	$M\{1,4\}_3$	∞	$M\{1,4\}_4$	∞	$M\{1,4\}_5$	
	$\{X2, Y2\}$	$M\{1,4\}_1$	$M\{1,8\}_1$	$M\{1,4\}_2$	$M\{1,8\}_2$	$M\{1,4\}_3$	$M\{1,8\}_3$	$M\{1,4\}_4$	$M\{1,8\}_4$	$M\{1,4\}_5$	$M\{1,8\}_5$	
	$\{X_d2, Y_d2\}$	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	
	$\{X5, Y5\}$	$M\{1,4\}_1$	$M\{1,8\}_1$	$M\{1,4\}_2$	$M\{1,8\}_2$	$M\{1,4\}_3$	$M\{1,8\}_3$	$M\{1,4\}_4$	$M\{1,8\}_4$	$M\{1,4\}_5$	$M\{1,8\}_5$	
$d_c = 20$	inputs	$\{1,4\}_1$	$\{5,8\}_1$	$\{9,12\}_1$	$\{13,16\}_1$	$\{17,20\}_1$	$\{1,4\}_2$	$\{5,8\}_2$	$\{9,12\}_2$	$\{13,16\}_2$	$\{17,20\}_2$	
	$\{X1, Y1\}$	$M\{1,4\}_1$	$M\{5,8\}_1$	$M\{9,12\}_1$	$M\{13,16\}_1$	$M\{17,20\}_1$	$M\{1,4\}_2$	$M\{5,8\}_2$	$M\{9,12\}_2$	$M\{13,16\}_2$	$M\{17,20\}_2$	
	$\{X_d1, Y_d1\}$	∞	$M\{1,4\}_1$	∞	∞	∞	∞	$M\{1,4\}_2$	∞	∞	∞	
	$\{X2, Y2\}$	$M\{1,4\}_1$	$M\{1,8\}_1$	$M\{9,12\}_1$	$M\{13,16\}_1$	$M\{17,20\}_1$	$M\{1,4\}_2$	$M\{1,8\}_2$	$M\{9,12\}_2$	$M\{13,16\}_2$	$M\{17,20\}_2$	
	$\{X_d2, Y_d2\}$	∞	∞	$M\{1,8\}_1$	∞	∞	∞	∞	$M\{1,8\}_2$	∞	∞	
	$\{X3, Y3\}$	$M\{1,4\}_1$	$M\{1,8\}_1$	$M\{1,12\}_1$	$M\{13,16\}_1$	$M\{17,20\}_1$	$M\{1,4\}_2$	$M\{1,8\}_2$	$M\{1,12\}_2$	$M\{13,16\}_2$	$M\{17,20\}_2$	
	$\{X_d3, Y_d3\}$	∞	∞	∞	$M\{1,12\}_1$	∞	∞	∞	∞	$\{1,12\}_2$	∞	
	$\{X4, Y4\}$	$M\{1,4\}_1$	$M\{1,8\}_1$	$M\{1,12\}_1$	$M\{1,16\}_1$	$M\{17,20\}_1$	$M\{1,4\}_2$	$M\{1,8\}_2$	$M\{1,12\}_2$	$M\{1,16\}_2$	$M\{17,20\}_2$	
	$\{X_d4, Y_d4\}$	∞	∞	∞	∞	$M\{1,16\}_1$	∞	∞	∞	∞	$M\{1,16\}_2$	
	$\{X5, Y5\}$	$M\{1,4\}_1$	$M\{1,8\}_1$	$M\{1,12\}_1$	$M\{1,16\}_1$	$M\{1,20\}_1$	$M\{1,4\}_2$	$M\{1,8\}_2$	$M\{1,12\}_2$	$M\{1,16\}_2$	$M\{1,20\}_2$	

Table I: Operation of TME: Entries represent the values of input, output and intermediate signals as a function of time (along columns) and d_c (along rows). ∞ is used to represent signals with maximum value for n-bit representation.

time	S[3:0]				
	$d_c = 4$	$d_c = 8$	$d_c = 12$	$d_c = 16$	$d_c = 20$
1	0x0	0x1	0xf	0xf	0xf
2	0x0	0x0	0xe	0xe	0xe
3	0x0	0x1	0x0	0xe	0xe
4	0x0	0x0	0xf	0x0	0xe
5	0x0	0x1	0xe	0xf	0x0

Table II: Control signals for Hold Mux

C. Compare Select and Normalize

In the top level CN architecture (Fig. 3), $M_{i,c}$ messages pass through D1 delay blocks, which cover clock zones occupied by both TME and Hold Mux. The delayed $M_{i,c}$ messages along with the $Min1$ and $Min2$ are received by Compare Select unit (Fig. 3.F) which performs the operation in equation (5), where $I = 1, \cdots, 4$, CS_I is the output of Compare Select block and $M_{i,c}$ is the corresponding input VN-CN message.

$$CS_I = \begin{cases} Min1 & \text{if } M_{i,c} > Min1 \\ Min2 & \text{else} \end{cases} \tag{5}$$

$$0.875 \times CS_I = CS_I - \frac{CS_I}{8} \tag{6}$$

Each output CS_I of Compare Select is multiplied by normalization factor $\alpha = 0.875$. The normalization rule (6) allows for very low cost implementation, with simple shift and subtract circuits (Fig. 3.G).

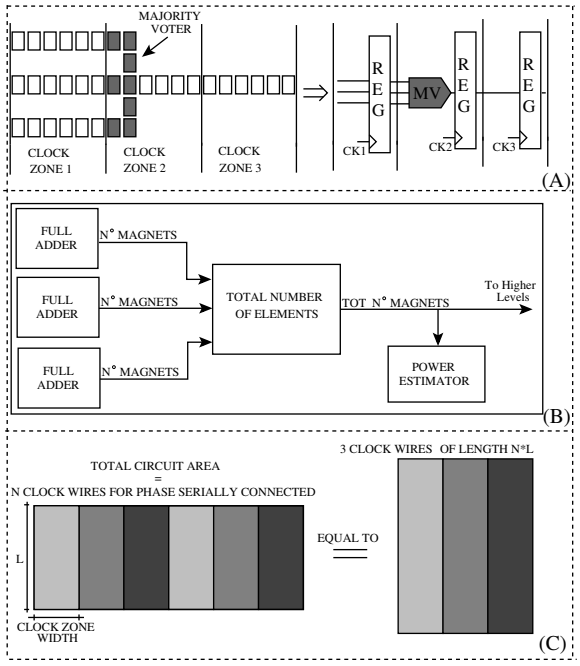

Figure 4: A) VHDL behavioral model. Registers are used to simulate the propagation delay. B) VHDL power estimation. The model is based on the hierarchical calculation of the number of elements for each component. C) Calculation of clock wires sizes. The layout is made by N wires of length L serially connected, equivalent to 3 single wires of length N*L.

V. VHDL MODEL

The key point in the VHDL modeling of QCA circuits is the behavior of clock zones. When the clock field (electric or magnetic) is applied to one clock zone, cells are forced to the reset state. When the field is removed cells switch according to the value of the cells of the previous clock zone. This is the same behavior as clocked registers, where a new data is accepted every clock cycle. As a consequence any QCA circuits can be modeled using registers to simulate the propagation delay of signals through the circuit. Normally a multiphase clock is used in QCA technology. Clock signals with different phases are applied to adjacent clock zones, in order to allow for correct signals propagation. In the literature the proposed clock schemes have 2, 3 or 4 phases. With this modeling different clock schemes can be simulated without changing the circuit description, simply applying the appropriate clock signals (depending on the scheme selected) to the registers.

The construction of the VHDL model [35] is simple (Fig. 4.A): starting from the real layout of the circuit, for each clock phase a register is used to model the propagation delay, while ideal wire and gates (majority voters and inverters) with no delay are used to model the logic behavior of the circuit. The effectiveness of this model depends on the quality of the initial representation of the circuit. Knowing the exact layout of the circuit, i.e. the placement of each magnet or molecule allows the model to match the exact behavior of the circuit. However it is impossible to design "by hand" such a complex circuit and place every magnet or molecule. Moreover, up to

now, there are no place&route tools that allow to automatically generate circuits with these technologies. We have introduced this behavioral model to overcome these limitations. Simpler blocks are described following the exact circuit layout. Higher level block are not simply made interconnecting lower level blocks, but to simulate the propagation delay due to interconnections ad-hoc delay blocks are used. Using this approach it is possible to obtain the most accurate representation of the circuit without knowing its exact layout. This is a solution used also in CMOS logic synthesizer, and the obtained results are good approximations of the real performances of the circuit. The "size" of these delay blocks is chosen according to our experience of QCA circuit design to obtain a representation as realistic as possible. The advantage of such modeling approach is the possibility to easily describe complex circuits. Moreover the exploit of VHDL language grants the possibility to use powerful simulation tools, like Modelsim [36], which enable rapid simulation of very complex circuits.

Most of the circuits proposed in the literature are based on custom layout for the clock zones. This is a strong assumption that at the moment does not find any clue of realistic implementation in the near future. The circuit shown in Fig. 4.A is based on a different approach and uses parallel straight wires as clock zones. This layout is based on the solution proposed in [5] for magnetic circuits, which is currently the only solution for clocking which was experimentally demonstrated [37]. Since the number of elements that can be chained inside a clock zone is limited, the vertical signal propagation with this clock structure is bounded and leads to a "stairs-like" propagation (see Section VI and Figure 5.C for more details). This clock structure fits well also in case of molecular QCA, with the difference that two wires must be used (over and under the plane, see Figure 2.F). There are other structures proposed for magnetic QCA, like [38], that have lesser limitations. Different clock systems can also be used for molecular QCA, because in this case an electric field (i.e. a voltage) is required instead of a magnetic field (i.e. a current). At nanoscale the manipulation of voltages is easier than the manipulation of currents. However no simulation or experimental evidence of this clock structures has been proposed. Moreover, using the same clock structure for both QCA implementations allows to simulate both of them with one circuit description.

The proposed VHDL based approach allows to evaluate not only to evaluate the timing performance of the circuit, but also power dissipation and area. This is obtained with a hierarchical estimation of the total number of elements of the circuit. The total number of cells (magnets or molecules) of a particular block is calculated as the sum of the elements of its sub-blocks while the overhead due to interconnects is taken into account by means of constant coefficients. This process is recursively applied along the circuitry from the lowest to the highest logic level and gives as a result the estimated total number of cells in the circuit. An example is shown in Figure 4.B. The total number of cells of an adder is evaluated adding the number of cells of each full adder and multiplying it for a constant that keeps into account the interconnection wires. Constants are chosen considering the accurate layout of a certain number of circuits previously implemented.

The total number of magnets can be used to estimate the circuit area at each logic level for each component and for the whole circuit. The area is obtained multiplying the area of one element for the total number of elements. Several constants are introduced to keep into account the separation space among neighbor elements and the wasted area due to the constraints generated by the clock zones layout. The total number of magnets can be used to estimate the power consumption due to magnet switching. Due to intrinsic behavior of QCA circuits in each clock cycle every cells is forced to the reset state and then it switches into one of the logic states. To evaluate the power consumption due to cells switching, the average energy consumption of each cell is multiplied with total number of cells and the product is divided by the clock period.

A further improvement of the model was implemented. Since the length of clock zones (and therefore of wires) is related to the circuit area, it is possible to estimate the power losses due to clock system generation. With this specific clock system the circuit is composed of N parallel wires of length L (Fig. 4.C) for each clock phase. Each wire segment of a specific clock zone must be connected to the other segment of the same clock zones serially. For example all the segment of clock zone 1 are connected serially. The serial connection is adopted because it allows to use the lowest value of current, and this is important because power losses in clock wires depend on the square value of the current (I^2). As a consequence the whole circuit layout can be approximated as 3 clock wires, one for each clock phase, of length $N * L$. A constant is used to consider the extra wire length required for the serial connection. The width of the clock wires is chosen according to the maximum number of magnets that can be aligned in a row. Knowing the circuit area, and choosing the zone width it is possible to estimate the total length of the wires. Choosing a clock wire thickness is then possible to evaluate the resistance of each clock wire and to estimate the power losses due to the clock system. Further details on the model can be found in [35].

It is important to underline that the estimations obtained with this model must not be taken as absolute performance benchmark of the circuit. However this model is very useful to compare different architectures and/or technology for the same circuit [15] [16].

VI. Check Node Performance

Figure 5.A shows a schematic layout of the circuit. In Figure 5.B it is instead indicated the detailed layout (using NML) of the CMP block, which is the basic logic block of this architecture. Magnets have 60 nm width and 90 nm height, while molecules are chosen of 2x2 nm. For magnetic QCA, the clock wires are made up of Copper and are 500 nm wide and 600 nm thick. For molecular QCA the width and height of clock wires is 12 nm and 100 nm respectively. In general it is better to choose the highest possible value for wires section to reduce their resistance and therefore the power losses. The schematic shown in Figure 5.A gives a general idea of the circuit layout of the whole Check Node. The circuit has a very

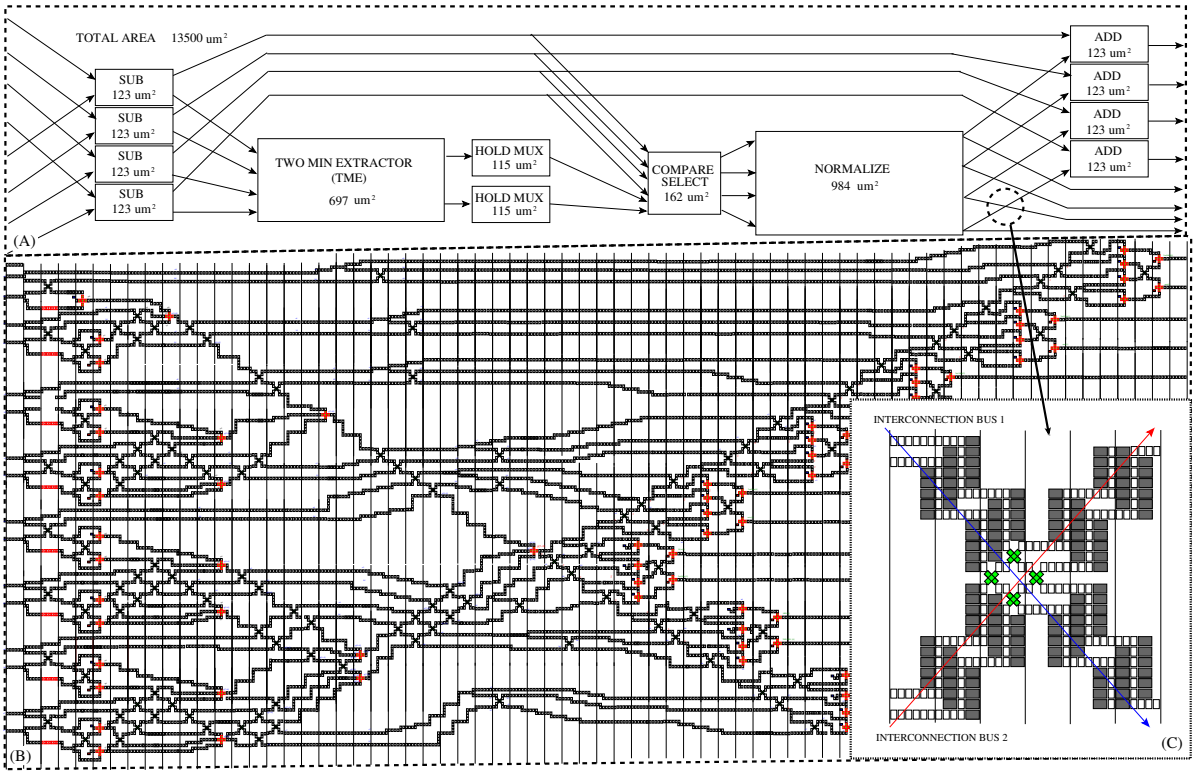

Figure 5: A) CMP block schematic layout. The layout is based on straight wires for the generation of the clock field. This layout was theoretically and experimentally demonstrated for Magnetic QCA [5], but due to its simplicity it can be adopted also for molecular QCA. For each block the estimated area is shown. Arrows show the routing of the interconnection wires. B) CMP block layout. The layout is based on straight wires for the generation of the clock field. This layout was theoretically and experimental demonstrated for Magnetic QCA [5], but due to its simplicity it can be adopted also for molecular QCA. C) A detail on vertical interconnection wires. Due to the layout limitations vertical signals follow a "stairs-like" propagation. Stabilizer blocks are used to improve the reliability in vertical signal propagation [39].

elongated shape. This is due to the chosen clock system, which favors the signal propagation in the horizontal direction and penalizes the vertical signal propagation. This can be clearly seen from Figure 5.C which shows simple crossing of two interconnection buses. The number of elements that can be cascaded per clock zone is limited. We have chosen to cascade a maximum of 6 elements in the horizontal direction whereas, the total number of elements cascaded in vertical direction is a little bit higher thanks to the use of helper blocks [39] which helps to improve the vertical signal propagation. However, also by using helper blocks the number of magnets that can be cascaded vertically cannot be too high. As a consequence if a long vertical interconnection is required a "stair-like" signal propagation must be used (Fig. 5.C), increasing the width of the circuit. The elongated layout of the circuit, with a balanced placement of the blocks, is therefore chosen to minimize the wasted area due to vertical signal propagation.

However also adopting a layout like this, the wasted area due to interconnections is quite high. The necessity to use cross wires (Fig. 5.B and Fig. 5.C) has also an impact on the wasted area. The total area of the components shown in Figure 5.A is around 3000 um^2 while the total area of the circuit is 13500 um^2. This situation is repeated inside each component, and it leads to the consequence that more than 95% of the area is dedicated to interconnections.

Table III: Comparison of the estimated performance of the proposed Check Node circuit with 4 different technologies: magnetic and molecular QCA, CMOS at 45nm and 21nm .

Technology	Mag. QCA	Mol. QCA	CMOS (45nm)	CMOS (21nm)
Clock (GHz)	0.1	1000	1.47	1.89
Latency (Clock cycles)	500	500	13	13
Area (mm^2)	0.0135	0.0000143	0.037	0.009
Power Dissipation (mW)				
Cells switching	0.003	88	–	–
Clock losses	1.6	23	–	–
Dynamic Power	–	–	37.96	17.49
Leakage Power	–	–	3.61	9.24
Total	1.603	111	41.57	26.73

This elongated layout has another consequence: the latency of the circuit is very high, 500 clock cycles as shown in Table III. This is not a particular problem in a pure feedforward circuit, since it is still possible to achieve the maximum throughput by means of pipeline. However this can be a problem in case of feedback circuits [16], so in that case the layout must be changed in order to minimize feedbacks length.

Table III shows power and area estimations of proposed check node and a comparison of the same circuit implemented

using CMOS technology[1]. The clock frequency used for magnetic QCA is 100MHz [40] which is the best theoretical frequency that can be obtained using adiabatic switching. It would be possible to obtain a clock frequency of about 1GHz [41], at the cost of greatly increasing the power consumption due to magnets switching. Since NML is interesting for its low power properties, a frequency of 100MHz was used. Molecular QCA can work at a frequency of 1THz [20]. This is an absolute best case scenario, that does not take into account the speed of the drive circuits and the technological issue due to molecular circuits fabrication. However we believe that it is important to show those technologies at the best of their possibility, to highlight the major advantages for which these technologies are studied (low power for magnetic QCA and high speed for molecular QCA). For QCA technology power is estimated using the VHDL model. For magnetic case the energy consumption is considered $30K_bT$ for each magnet, while a current of 10mA is used for clock losses estimation. For molecular case the energy consumption for each molecule is 2eV per switching [20] since the frequency chosen is 1THz, while an electric field of 1.5 V/nm [20] is used for clock losses estimation. CMOS data were obtained through Synopsys Design Compiler with a target technology of 45nm. A comparison with 21nm technology node is also shown obtained referring to predictions reported on the International technology Roadmap for Semiconductor [1].

Clearly NanoMagnet Logic is the winner from power consumption point of view, although the impact of clock system losses is quite remarkable. Molecular QCA are the best solution from the performance point of view, although this is clearly a best case scenario. The power consumption is higher than CMOS case but this is caused by the high operative frequency. Even if a notable improvement is achieved scaling from 45nm node to CMOS 21nm node, a remarkable advantage in terms of area and speed still holds in the molecular QCA case, and between one and two orders of magnitude is the total power reduction in the case of magnetic QCA. Moreover, even downgrading the CMOS 21nm node frequency to the Magnetic QCA value case, difference between the two technologies in terms of dissipated power would remain very large due to the leakage power. It is also worth underlining that the ITRS estimations only roughly consider the unavoidable overheads due to interconnects in a complex system. These overheads are expected to notably reduce the timing improvements predicted at device level [42]. Finally, although CMOS is not the best solution for speed and power, it has a remarkably smaller latency.

VII. Conclusions

As a relevant design case for QCA technology, this work outlined an innovative partial parallel Check Node architecture for LDPC decoders. Optimization was done both at architecture and layout levels. The whole architecture was designed using QCA basic logic gates. A realistic layout was presented taking into account a feasible clock structure.The design description was based on a VHDL model which allows both cycle true validation of the circuit, and area and power estimation for magnetic and molecular QCA technologies. Results prove that QCA technology can be adopted to implement applications characterized by very high processing complexity. Key in the design flow is adapting the architecture to the specific features of QCA circuits, which implies for example wise use of majority voters and pipelined wires. Comparing the performance offered by the QCA based decoder with an equivalent CMOS implementation, a considerable area saving was achieved both for magnetic and molecular QCA. Results further show that these kinds of decoders are perfectly suited to molecular QCA if high throughput is requested, while magnetic QCA is attractive for low power applications with low to moderate throughput requirements.

[1]Data for 21nm node are obtained multiplying values of 45nm node for coefficients obtained from the ITRS [1]. In the transition between 45nm and 21nm dynamic power decreases of 2.17 times, leakage power increases of 2.56 times, area decreases of 4 times and frequency increases of 1.29 times.

References

[1] "International technology roadmap for semiconductors (ITRS)," http://www.itrs.net, 2007., Tech. Rep.

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

[3] J. Schiff, *Cellular Automata: A Discrete View of the World*. Wiley & Sons, Inc., 2007.

[4] Y. Lu and C. Lent, "Theoretical study of molecular quantum dot cellular automata," in *Computational Electronics, 2004. IWCE-10 2004. Abstracts. 10th International Workshop on*, oct. 2004, pp. 118 – 119.

[5] M. Niemier, G. Bernstein, G. Csaba, A. Dingler, X. Hu, S. Kurtz, S. Liu, J. Nahas, W. Porod, M. Siddiq, and E. Varga, "Nanomagnet logic: progress toward system-level integration," *J. Phys.: Condens. Matter*, vol. 23, p. 34, Nov. 2011.

[6] J. Seminario, P. Derosa, L. Cordova, and B. Bozard, "A molecular device operating at terahertz frequencies: theoretical simulations," *Nanotechnology, IEEE Transactions on*, vol. 3, no. 1, pp. 215 – 218, march 2004.

[7] G. Csaba, P. Lugli, and W. Porod, "Power dissipation in nanomagnetic logic devices," in *Nanotechnology, 4th IEEE Conf. on*, 2004.

[8] S. Hashemi, M. Azghadi, and A. Zakerolhosseini, "A novel QCA multiplexer design," *Intl. Symp. on Telecommunications*, 2008.

[9] H. Cho and E. Swartzlander, "Adder and multiplier design in quantum-dot cellular automata," *IEEE Transaction On Computers*, vol. 58, no. 6, 2009.

[10] I. Kong, E. Swartzlander, and S. Kim, "Design of a Goldschmidt iterative divider for quantum-dot cellular automata," *Nanoarch*, Jul. 2009.

[11] X. Yang, L. Cai, and X. Zhao, "Low power dual-edge triggered flip-flop structure in quantum dot cellular automata," *IET Electronics Letters*, vol. 46, no. 12, 2010.

[12] B. Taskin and B. Hong, "Improving line-based QCA memory cell design through dual phase clocking," *Very Large Scale Integration (VLSI) Systems, IEEE Trans. on*, vol. 16, no. 12, pp. 1648 –1656, dec. 2008.

[13] K. Walus, T. Dysart, G. Jullien, and R. Budiman, "QCADesigner: A rapid design and simulation tool for quantum-dot cellular automata," *IEEE Transaction on Nanotechnology*, vol. 3, no. 1, March 2004.

[14] M. Graziano, M. Vacca, A. Chiolerio, and M. Zamboni, "A NCL-HDL snake-clock based magnetic QCA architecture," *IEEE Transaction on Nanotechnology*, no. 10, p. DOI:10.1109/TNANO.2011.2118229.

[15] M. Graziano, M. Vacca, D. Blua, and M. Zamboni, "Asynchrony in quantum-dot cellular automata nanocomputation: Elixir or poison?" *IEEE Design & Test of Computers*, 2011.

[16] M. Vacca, M. Graziano, and M. Zamboni, "Asynchronous solutions for nano-magnetic logic circuits," *ACM Journal on Emerging Technologies in Computing Systems*, vol. 7, no. 4, December 2011.

[17] P. D. Tougaw and C. S. Lent, "Logical devices implemented using quantum cellular automata," *Journal of Applied Physics*, vol. 75, no. 3, pp. 1818 –1825, feb 1994.

[18] J. Timler and C. Lent, "Power gain and dissipation in quantum dot cellular automata," *J. of App. Physics*, vol. 91, no. 2, pp. 823–831, 2002.

[19] C. Lent and B. Isaksen, "Clocked molecular quantum-dot cellular automata," *Electron Devices, IEEE Transactions on*, vol. 50, no. 9, pp. 1890 – 1896, sept. 2003.

[20] Y. Lu, M. Liu, and C. Lent, "Molecular electronics - from structure to circuit dynamics," in *Nanotechnology, 2006. IEEE-NANO 2006. Sixth IEEE Conference on*, vol. 1, june 2006, pp. 62 – 65.

[21] Y. Wang and M. Lieberman, "Thermodynamic behavior of molecular-scale quantum-dot cellular automata (QCA) wires and logic devices," *Nanotechnology, IEEE T. on*, vol. 3, no. 3, pp. 368 – 376, sept. 2004.

[22] Y. Lu and C. Lent, "Theoretical study of molecular quantum dot cellular automata," in *Computational Electronics, 2004. IWCE-10 2004. Abstracts. 10th International Workshop on*, oct. 2004, pp. 118 – 119.

[23] V. Arima, M. Iurlo, L. Zoli, S. Kumar, M. Piacenza, F. Della Sala, F. Matino, G. Maruccio, R. Rinaldi, F. Paolucci, M. Marcaccio, P. G. Cozzi, and A. P. Bramanti, "Toward quantum-dot cellular automata units: thiolated-carbazole linked bisferrocenes," *Nanoscale*, vol. 4, 2012.

[24] Y. Lu and C. Lent, "Self-doping of molecular quantum-dot cellular automata: mixed valence zwitterions." *Phys Chem Chem Phys : 13(33):14928-36*, 2011, sep 7.

[25] A. Pulimeno, M. Graziano, D. Demarchi, and G. Piccinini, "Towards a molecular qca wire: simulation of write-in and read-outsystems," *Solid-State Electronics*, vol. 77, November 2012.

[26] e. a. R.E. Smalley, *Carbon Nanotubes: Synthesis, Structure, Properties and Applications*. Springer-Verlag, 2001.

[27] R. Gallager, "Low-density parity-check codes," *Information Theory, IRE Transactions on*, vol. 8, no. 1, pp. 21 –28, 1962.

[28] M. Khan and S. Ghauri, "The WiMAX 802.16e physical layer model," in *Wireless, Mobile and Multimedia Networks, 2008. IET International Conference on*, 2008, pp. 117 –120.

[29] A. Morello and V. Mignone, "DVB-S2: The second generation standard for satellite broad-band services," *Proceedings of the IEEE*, vol. 94, no. 1, pp. 210 –227, 2006.

[30] J. Chen and M. Fossorier, "Near optimum universal belief propagation based decoding of LDPC codes and extension to turbo decoding," in *Information Theory, 2001. Proceedings. 2001 IEEE International Symposium on*, 2001, p. 189.

[31] M. Martina, G. Masera, S. Papaliralabos, P. Mathiopoulos, and F. Gioulekas, "On practical implementation and generalization of max* operation for Turbo and LDPC decoders," *Instrumentation Measurements, IEEE Transactions on*, to appear.

[32] E. Yeo, P. Pakzad, B. Nikolic, and V. Anantharam, "High throughput low density parity check decoder architectures," in *Global Telecomm. Conf., 2001. GLOBECOM '01. IEEE*, vol. 5, 2001, pp. 3019 –3024.

[33] M. Wagh, Y. Sun, and V. Annampedu, "Implementation of comparison function using quantum-dot cellular automata," *Proc. of NanoTech2008,Boston, MA,*, vol. 3, pp. 76–79, June 1-3 2008.

[34] L. Amarù, M. Martina, and G. Masera, "High speed architectures for finding the first two maximum/minimum values," *IEEE Transactions on VLSI systems*, vol. 20, no. 12, 2012.

[35] M. Vacca, M. Graziano, and M. Zamboni, "Nanomagnetic logic microprocessor: Hierarchical power model," *IEEE Tr. on VLSI Systems*, 2012.

[36] "Mentor graphics," http://www.modelsim.com.

[37] M. Alam, M. Siddiq, G. Bernstein, M. Niemier, W. Porod, and X. Hu, "On-chip clocking for nanomagnet logic devices," *IEEE Transaction on Nanotechnology*, 2009.

[38] J. Das, S. Alam, and S. Bhanja, "Low power magnetic quantum cellular automata realization using magnetic multi-layer structures," *J. Emerging and Sel. Topics in Cir. and Sys.*, vol. 1, no. 3, Sep. 267-276.

[39] D. Carlton, N. Emley, E. Tuchfeld, and J. Bokor, "Simulation studies of nanomagnet-based logic architecture," *Nanoletters*, vol. 8, no. 12, pp. 4173–4178, Nov. 2008.

[40] M. Niemier, X. Hu, M. Alam, G. Bernstein, M. P. W. Porod, and J. DeAngelis, "Clocking structures and power analysis for nanomagnet-based logic devices," in *International Symposium on Low Power Electronics and Design*. Portland-Oregon, USA: IEEE, 2007, pp. 26–31.

[41] M. Vacca, M. Graziano, and M. Zamboni, "Majority voter full characterization for nanomagnet logic circuits," *IEEE Transactions on Nanotechnology*, vol. 11, no. 5, 2012.

[42] A. Pulimeno, M. Graziano, and G. Piccinini, "Udsm trends comparison: From technology roadmap to ultrasparc niagara2," *IEEE Transactions on VLSI systems*, vol. 20, no. 7, 2012.