An Energy-aware Model for the Logic Synthesis of Quantum-Dot Cellular Automata

Frank Sill Torres, *Member, IEEE,* Robert Wille, *Senior Member, IEEE,*
Philipp Niemann, *Member, IEEE,* and Rolf Drechsler, *Fellow, IEEE*

***Abstract*—*Quantum-Dot Cellular Automata* (QCA) are an emerging field-coupled nanotechnology with remarkable performance and energy efficiency. In order to enable the exploration of this technology, we propose a model for the logic synthesis of QCA circuits that, for the first time, considers and abstracts all main physical aspects—in particular, energy dissipation. To this end, we review in detail how energy is dissipated in QCA cells and present a corresponding environment that allows for the estimation of the energy dissipation with respect to any specific set of technology parameters. Based on that, we derive a model for logic synthesis. A case study confirms the accuracy of the proposed model and reveals that interconnections have a significant impact in this technology—motivating a more rigorous consideration. These findings eventually provide the basis for a new generation of synthesis approaches at the logic level that are explicitly dedicated to QCA systems.**

***Index Terms*—Quantum-Dot Cellular Automata, Logic Synthesis, Energy Dissipation, Field-Coupled Nanocomputing**

I. Introduction

Quantum-Dot Cellular Automata (QCA) [1] provide a circuit technology based on *Field-Coupled Nanocomputing* (FCN). Here, structures which are able to confine electric charges, so called *quantum dots*, are arranged in a square-like fashion so that free mobile electrons can move between them. Since these electrons impose mutual repulsion due to Coulomb interaction [2], they tend to locate themselves at opposite corners of the square (i. e. at the top-left and bottom-right position or the top-right and bottom-left position). By this, these electrons assume stable states which are interpreted as binary 0 and 1. Composing several of such squares to a grid and additionally applying a dedicated clocking scheme to control the mobility of the electrons allows to eventually realize arbitrary Boolean functions. At the same time, this way of representing and processing Boolean values and functions, respectively, allows for systems with highest processing performance and remarkably low energy dissipation [3], [4]—making QCA a promising alternative to conventional integrated circuit technologies.

Manuscript received October XY, 2017. The work of F. Sill Torres was supported by the Brazilian agencies CNPq (grant numbers 310326/2016-8 and 202512/2017-7) and FAPEMIG (grant number PPM-00624-16).

F. Sill Torres is with the Federal University of Minas Gerais, 31270-901 Belo Horizonte, Brazil, and also with the Group for Computer Architecture, University of Bremen, 28359 Bremen, Germany (e-mail: franksill@ufmg.br).

P. Niemann is with the Cyber-Physical Systems Department, DFKI GmbH, 28359 Bremen, Germany.

R. Wille is with the Institute for Integrated Circuits, Johannes Kepler University Linz, 4040 Linz, Austria and also with the Cyber-Physical Systems Department, DFKI GmbH, 28359 Bremen, Germany.

R. Drechsler is with the Group for Computer Architecture, University of Bremen, 28359 Bremen, Germany, and also with the Cyber-Physical Systems Department, DFKI GmbH, 28359 Bremen, Germany.

As a consequence, how to realize QCA circuits received significant attention in the past: This includes numerous contributions on their physical realization (see e. g. [5]–[8]), but also first works on logic synthesis of QCA circuits have been presented (see e. g. [9]–[13]). The latter contributions are essential since, as for conventional circuitry, complex systems can eventually only be realized with the help of efficient design methods. While logic synthesis represents a key step of the design, corresponding methods for logic synthesis of QCA systems are still in an early stage. Although initial QCA designs realizing arithmetic circuits, processors, and FPGAs have been proposed in [14], [15], and [16], respectively, these designs have been derived manually.

At the same time, automatic logic synthesis for QCA circuits is a non-trivial task as it requires a comprehensive consideration of different aspects, namely:

- The *functional level*, i. e. models and methods respectively describing and generating a netlist of the circuit to be realized. Here, solutions for conventional circuitry can be re-used, such as the description of logic functions in terms of (conventional) gate libraries or approaches for automatic logic synthesis [17].
- The *physical level*, i. e. models and methods respectively describing and evaluating the costs of a circuit (or components thereof) in terms of area, delay, and energy dissipation. Here, rather advanced solutions already exist for area (e. g. defined by the grid size of the QCA circuit) and delay (e. g. resulting from the applied clocking scheme) [11], [18]. However, for energy dissipation no proper model applicable for the automatic design of QCA circuits exists yet—although energy efficiency is one of the major benefits of this technology.
- The *geometric level*, i. e. models and methods respectively describing and generating the layout of the QCA circuit (e. g. its pin locations, gate placement, or interconnection routing). Here, solutions as proposed in [13], [19] have already been applied.

Most logic synthesis methods for QCA circuits (e .g. as proposed in [9]–[13], [18], [20]) indeed follow this separation of concerns and consider these levels separately. More precisely, they follow a two-stage design flow in which the desired function is synthesized first in terms of a conventional circuit (utilizing the accomplishments in automatic logic design developed for conventional circuits in the past decades such as [17], [21]). Afterwards, the resulting (conventional) circuit is mapped into a proper QCA circuit using corresponding building blocks.

This flow explicitly decouples the two stages, i. e. the actual synthesis step is basically technology-independent and does not consider any QCA-related objectives, while the mapping step usually does not change the netlist generated by the first step, e. g. in order to better satisfy geometric constraints or improve the physical cost. While issues such as area and delay (for the physical level) are already covered to some extent by given implementations of the logic functions, particularly energy dissipation is hardly considered in the first step.

As a consequence, conventional circuit descriptions are frequently mapped into QCA circuits which are optimized with respect to conventional cost metrics such as the number of gates, delay, etc., but may not be very suited for QCA systems and their energy dissipation. Obviously, this is crucial for a technology that is mainly motivated by its energy efficiency.

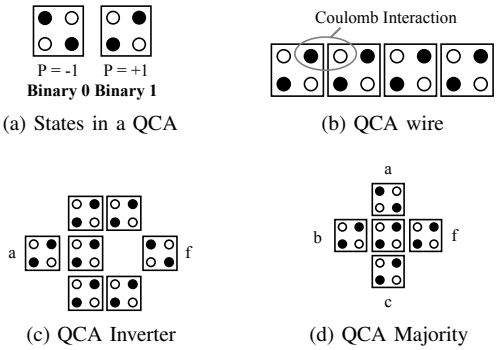
In this work, we are addressing this issue by lifting the consideration of energy dissipation in QCA designs from the physical level to the functional level. To this end, we first review in detail how energy is dissipated in QCA circuits and what technology parameters have an effect on energy dissipation. Based on this, we provide a methodology as well as a corresponding tool that allows for the estimation of the energy dissipation with respect to any specific set of technology parameters. A case study shows that the determined energy dissipation for important building blocks is suitable as a cost function, i. e. the values can be used to approximate the energy dissipation of larger circuits that are made up from these blocks.

For the first time, this leads to an energy-aware model for the logic synthesis of QCA designs which comprehensibly does not only cover a gate library of elementary building blocks to be used in order to realize arbitrary complex functionality but also provides proper, technology-specific cost functions for area, delay, and, in particular, energy dissipation which can be adapted to the parameters of the target technology. Moreover, as a consequence of the resulting model we unveil that, in contrast to logic synthesis for conventional circuitry, interconnections have significant physical costs in QCA and all synthesis approaches generating QCA circuits should consider them. These findings eventually provide the basis for more dedicated automatic logic synthesis methods for this promising technology.

In the remainder of this work, the proposed model is introduced as follows: First, the basics on QCA are reviewed in the next section. Afterwards, the energy dissipation of QCA cells is investigated in detail in Section III. This includes a precise description of the methodology and the tool that are used to compute the actual energy dissipation of QCA circuits. These findings are eventually utilized to formulate the proposed model in Section IV—constituting the main result of this work. Afterwards, the accuracy of the model and the impact of interconnections are discussed in Section V. Finally, the work is concluded in Section VI.

II. Quantum-Dot Cellular Automata

Quantum-Dot Cellular Automata (QCA) are a field-coupled nanotechnology that conducts computations fundamentally differently from today's technologies. Information is stored in terms of the polarity of small cells and can be propagated to adjacent cells using electrostatic force (Coulomb interaction).



(a) States in a QCA (b) QCA wire (c) QCA Inverter (d) QCA Majority

Fig. 1: QCA states and operations

More precisely, a QCA *cell* is typically composed of four quantum dots which are able to confine an electric charge [22], [23]. These quantum dots are arranged at the corners of a square, such as depicted in Fig. 1. Further, each cell contains two free and mobile electrons (illustrated by black dots in Fig. 1) which are able to tunnel between adjacent dots. Tunneling to the outside of the cell is prevented by a potential barrier and also tunneling within the cell can temporarily be prevented—leading to a stable state. As the electrons within a cell experience mutual repulsion due to Coulomb interaction, they tend to locate at opposite corners of the square in these cases. Consequently, an isolated cell may be in one of two stable energy states (termed as *cell polarizations*) $P = -1$ and $P = +1$ as shown in Fig. 1a. This allows for an encoding of binary information by identifying $P = -1$ with a binary 0 and $P = +1$ with a binary 1.

Moreover, when placed close to each other, the polarization of one cell influences the polarization of the other—again by Coulomb interaction. This causes electrons to avoid to be located in neighboring quantum dots. Exploiting this effect allows for the realization of wires as well as logic gates.

Example 1. *Based on the concepts introduced above, the following circuit elements can easily be realized:*

- *A* wire *as shown in Fig. 1b, where e. g. a 1-state is propagated through several cells by Coulomb interaction (here, from left to right).*
- *An* inverter *as shown in Fig. 1c, where e. g. a 1-state is copied to two paths, which then are combined diagonally, such that the 1-state is inverted to a 0-state (again, from left to right).*
- *A* majority gate *as shown in Fig. 1d, where e. g. a single 0-state from input a competes with two 1-states coming from inputs b and c. The output follows the majority of the input states and, thus, becomes a 1-state in this case.*

Note that further classical logic operations such as OR and AND gates can easily be derived from the majority gate by locking one of its inputs to a 1-state (yielding an OR) or 0-state (yielding an AND).

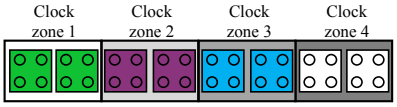


Fig. 2: QCA wire with cells in four clock zones

In order to execute these and more complex logic operations, a dedicated *clocking* is required which, starting with the initialization of the QCA cells, properly propagates the data from cell to cell and avoids metastable states [24]. To this end, an external clock is employed which consists of four phases and regulates the interdot barriers within a QCA cell such that the cell can be polarized or not. In the so-called *relax* phase, the cell is depolarized. During the following *switch* phase, the interdot barriers are raised while a new input is being applied. Consequently, the cell polarizes into one of the two antipodal states. In the following *hold* phase, the cell keeps its polarization and acts as input for adjacent cells which are in the *switch* phase. During the final *release* phase, the interdot barriers are lowered thereby removing the previous polarization of the cell.

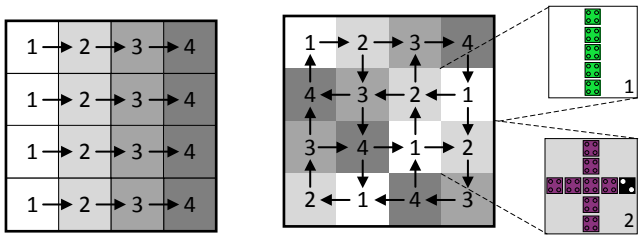
Usually, four different clock signals shifted by one/two/three phases are provided for this purpose [25]. Further, cells are grouped in clock zones, whereby each zone is controlled by one of the four clock signals. The data flow is controlled by placing adjacent clock zones such that the cells which shall pass their data are in *hold* phase when the cells that shall receive the data are in *switch* phase. An example illustrates the concepts.

Example 2. *Consider Fig. 2 showing a QCA wire with cells in four different clock zones. When clock zone 2 is in the* switch *phase, then clock zone 1 is in the* hold *phase. Thus, in this clock phase, cells in clock zone 2 polarize according to the polarization of the adjacent cells in clock zone 1. During the next clock phase, clock zone 2 changes to* hold*, while clock zone 3 is in* switch*. Consequently, data is passed from zone 2 to 3, similar to a pipeline structure.*

Because of this clocking scheme, logic operations such as reviewed in Example 1 and Fig. 1 must be arranged in a fashion that respects the corresponding timing, i. e. such that data is properly passed from one block to another. To this end, usually a fix arrangement of clock zones is imposed on a QCA layout where each clock zone contains 5×5 QCA cells (following the proposal from [26]–[28]).

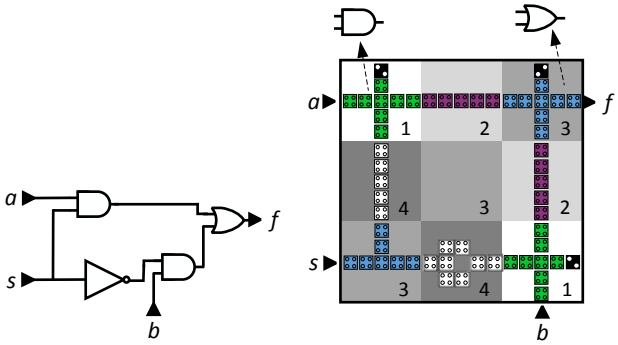
Example 3. *Consider the two clocking schemes depicted in Fig. 3. Both schemes define a grid of clock zones, or* tiles*, which are consecutively numbered from 1 to 4, whereby the clock signals of consecutive zones differ by a phase-shift of one phase. The grids are organized such that each clock zone has at least one neighboring zone that can provide data and one that can receive data. The possible data flows between adjacent clock zones are indicated by arrows.*

Given the introduced concepts, complex designs can be realized by mapping conventional circuits onto QCA.



(a) Wave clocking based on [1] (b) USE clocking scheme proposed in [28]

Fig. 3: Exemplary clocking schemes



(a) Conventional circuit (b) Mapped to QCA grid

Fig. 4: Multiplexer circuit mapped to a QCA grid

Example 4. *Consider the multiplexer (*MUX*) function $f = as + b\bar{s}$ to be realized. Using conventional design tools, a gate level representation of this function as shown in Fig. 4a results. Mapping this netlist to QCA logic operations (see Fig. 1) and a QCA clocking scheme as shown in Fig. 4b properly satisfies all logic and timing constraints, i. e. the desired function is implemented and the output of one clock zone is always propagated to the input of a clock zone containing the next operation or wire.*

III. Energy Dissipation of Quantum-Dot Cellular Automata

This sections discusses in detail the energy behavior of QCA cells and presents a tool that enables the estimation of the actual energy dissipation of QCA circuits. In order to faithfully model the physical behavior of a QCA cell (including its energy dissipation), it is necessary to consider not only the cell's polarization but its complete quantum-level state. Therefore, we begin this section with a review of the state-of-the-art quantum-level modeling of QCA cell behavior. Next, the resulting description of the energy behavior is detailed followed by the introduction of the developed tool for the estimation of energy dissipation in QCA circuits.

A. Quantum-Level Behavior of QCA Cells

Each QCA cell has two associated states: one describes the cell's actual energy state in terms of its *density matrix* $\hat{\rho}$. The other one is called *steady-state* and describes which state the cell should "ideally" have and will try to assume (as it has the

lowest possible energy) [29]. The *steady-state* depends on the polarizations of neighboring cells as well as the cell's current tunneling behavior. Once these parameters are changed, the cell tries to follow the (accordingly changed) steady-state as closely as possible.

Assuming QCA cells with four dots and two completely polarized states $P = \pm 1$, the density matrix $\hat{\rho}$ of a QCA cell at any given point in time, i. e. its actual quantum (energy) state, has dimension 2×2. For convenience, it is represented in terms of the associated *coherence vector* $\vec{\lambda} = (\lambda_x, \lambda_y, \lambda_z)$ which represents $\hat{\rho}$ in the coordinates of the Pauli operators σ_x, σ_y, σ_z, i. e. $\lambda_i = \mathrm{Tr}(\hat{\rho}\sigma_i)$ [30]. By this, the polarization P_i of a cell i can be identified with the expectation value of the Pauli spin operator or, equivalently, with the third component of the coherence vector [29], [31]:

$$P_i = -\langle \hat{\sigma}_z \rangle_\rho := -\,\mathrm{Tr}(\hat{\rho}\sigma_z) = -\lambda_z. \tag{1}$$

The factors that affect the steady-state of a cell i, i. e. the interplay with neighbored cells as well as modifications to the tunneling energy (controlled by the clock), are described by the Hamiltonian [29], [32]

$$\hat{H}_i = \begin{bmatrix} -\frac{1}{2}\Phi & -\gamma \\ -\gamma & \frac{1}{2}\Phi \end{bmatrix}, \tag{2}$$

where γ is the tunneling energy between two neighboring dots in the same cell and $\Phi := \sum_{j \in N(i)} E_{kink}^{i,j} P_j$ models the Coulombic interactions with cells from the neighborhood $N(i)$ of the cell. More precisely, this interaction depends on the other cells' polarization P_j as well as the so-called "kink energy" $E_{kink}^{i,j}$ (between two cells i and j) which quantifies the energy cost of both cells having opposite polarizations[1].

In order to construct the steady-state coherence vector $\vec{\lambda}_{ss}$ from the cell's Hamiltonian, we project it onto the Pauli basis in a similar fashion as this was done above for the density matrix. More precisely, we construct the real-valued three-dimensional energy vector $\vec{\Gamma}$ [29], [33] whose components are given by $\Gamma_i = \frac{\mathrm{Tr}(\hat{H}\sigma_i)}{\hbar}$, where $\hbar$ means the reduced Planck constant, i. e.

$$\vec{\Gamma} = \frac{1}{\hbar}\left[-2\gamma, 0, \Phi\right]. \tag{3}$$

Following [29], [33], the steady-state coherence vector $\vec{\lambda}_{ss}$ is then given by

$$\vec{\lambda}_{ss} = -\frac{\vec{\Gamma}}{|\vec{\Gamma}|}\tanh \eta_{th} \tag{4}$$

with $|\vec{\Gamma}| = \sqrt{4\gamma^2 + (\Phi)^2}$, also known as the Rabi frequency [34], and η_{th} refers to the thermal ratio with

$$\eta_{th} = \frac{\hbar|\vec{\Gamma}|}{2k_B T}, \tag{5}$$

where k_B is the Boltzmann's constant and T refers to the temperature.

Using this notation, the dynamic behavior of a QCA cell, i. e. the quantum state evolution towards the steady-state, is then modeled by the following equation [29], [32]:

$$\frac{d}{dt}\vec{\lambda} = \vec{\Gamma} \times \vec{\lambda} - \frac{1}{\tau}\left(\vec{\lambda} - \vec{\lambda}_{ss}\right) \tag{6}$$

Here, the first part $\frac{d}{dt}\vec{\lambda} = \vec{\Gamma} \times \vec{\lambda}$ (where $\times$ denotes cross/vector product) describes the ideal evolution according to the Liouville equation—without dissipative effects. In practice, however, the change of the quantum state (relaxation) is damped and some portion of the energy is dissipated to the environment. To this end, the dissipative component $-\frac{1}{\tau}(\vec{\lambda} - \vec{\lambda}_{ss})$ models the interaction of the QCA cell with the thermal bath in a relaxation-time approximation where τ denotes the technology-dependent relaxation time [29], [33].

Overall, Eq. (6) describes how a cell's state changes towards its steady-state w.r.t. the time constant τ [31]. The lower the value of this technology-dependent parameter, the higher the energy dissipation of the QCA cell [31], [33].

B. Energy Behavior of QCA Cells

Using the quantum-level modeling of QCA cell behavior presented above, we can now turn to the actual energy analysis. This analysis has to be conducted over complete clock cycles. In the beginning of a clock cycle, each QCA cell is depolarized. Energy is taken from the clock as well as from neighboring cells in order to achieve the polarized state induced by the polarization of the neighboring cells. Most of this energy is restored to the clock as well as distributed to the neighboring cells until the cell becomes depolarized again at the end of the clock cycle. However, as discussed above, some portion of the energy dissipates to the environment.

In the following, we investigate how the different energies can be measured. By definition, the current energy $\mathbb{E}$ of a QCA cell at any time is given via the expectation value of its Hamiltonian $\hat{H}$ w.r.t. the current density matrix $\hat{\rho}$ [29]:

$$\mathbb{E} = \langle \hat{H} \rangle_{\hat{\rho}} := \mathrm{Tr}(\hat{\rho}\hat{H}) \tag{7}$$

Noting that $\hat{H} = -\gamma\sigma_x + \frac{1}{2}\Phi\sigma_z$ and exploiting linearity of the Tr operator, this can be re-written and simplified as

$$\begin{aligned} \mathbb{E} &= \mathrm{Tr}(\hat{\rho}(-\gamma\sigma_x + \frac{1}{2}\Phi\sigma_z)) \\ &= -\gamma\,\mathrm{Tr}(\hat{\rho}\sigma_x) + \frac{1}{2}\Phi\,\mathrm{Tr}\left(\hat{\rho}\sigma_z\right) \\ &= -\gamma\lambda_x + \frac{1}{2}\Phi\lambda_z \\ &= \frac{\hbar}{2}\left[\frac{1}{\hbar}(-2\gamma\lambda_x + 0 \cdot \lambda_y + \Phi\lambda_z)\right] \\ &= \frac{\hbar}{2}\vec{\Gamma} \cdot \vec{\lambda}, \end{aligned} \tag{8}$$

[1]Note that the above Hamiltonian solely considers Coulombic interactions amongst nearby cells and ignores intercellular entanglement effects between cells. The authors in [32] and [31] discuss this simplification and conclude that this so-called intercellular Hartree approximation is adequate for quasi-adiabatic time evolution. This observation is also partially validated by experimental work on QCA-like systems [5], [7]. However, it has been pointed out the weakness of this approximation to model depolarization effects, especially for long wires, i. e. wires with more than five QCA cells, and long clock phases [31].

where the $\cdot$ in the last row (as well as in the following equations) denotes scalar product.

The instantaneous power P can then be described by

$$P = \frac{d}{dt}\mathbb{E} = \frac{d}{dt}\left(\frac{\hbar}{2}\vec{\Gamma}\cdot\vec{\lambda}\right) \tag{9}$$

and the total energy dissipation E_{total} of a QCA cell during a complete clock cycle with period T_{clk} eventually is given as

$$\begin{aligned} E_{total} &= \int_{t_0}^{t_0+T_{clk}} P dt' \\ &= \frac{\hbar}{2}\int_{t_0}^{t_0+T_{clk}} \left(\frac{d}{dt}\vec{\Gamma}\cdot\vec{\lambda} + \frac{d}{dt}\vec{\lambda}\cdot\vec{\Gamma}\right) dt'. \end{aligned} \tag{10}$$

The first summand in the integrand of Eq. (10) is the product of the derivate of the energy vector of the cell, given in Eq. (3), and the coherence vector. This term refers to the energy transfer with the clock (E_{clk}) as well as neighboring cells (E_{IO}) during a clock cycle [29], [33], i. e.

$$E_{clk} + E_{IO} = \frac{\hbar}{2}\int_{t_0}^{t_0+T_{clk}} \left(\frac{d}{dt}\vec{\Gamma}\cdot\vec{\lambda}\right) dt' \tag{11}$$

with

$$E_{clk} = \frac{1}{2}\int_{t_0}^{t_0+T_{clk}} \left(\frac{d}{dt}(-2\gamma)\cdot\lambda_x\right) dt' \tag{12}$$

and

$$E_{IO} = \frac{1}{2}\int_{t_0}^{t_0+T_{clk}} \left(\frac{d}{dt}\Phi\cdot\lambda_z\right) dt', \tag{13}$$

where λ_x and λ_z refer to the x and z components of the coherence vector $\vec{\lambda}$.

The second summand in the integrand of Eq. (10) is the product of the derivate of the coherence vector and the energy vector and captures the energy transfer E_{env} with the environment during a clock cycle. E_{env} is the actually dissipated energy of a QCA cell during a clock cycle and can be determined via integrating Eqs. (3), (4) and (6) into the second term of Eq. (10), i. e.

$$\begin{aligned} E_{env} &= \frac{\hbar}{2}\int_{t_0}^{t_0+T_{clk}} \left(\frac{d}{dt}\vec{\lambda}\cdot\vec{\Gamma}\right) dt' \\ &= \frac{\hbar}{2}\int_{t_0}^{t_0+T_{clk}} \Bigg[\vec{\Gamma}\cdot\left(\vec{\Gamma}\times\vec{\lambda}\right) \\ &\qquad - \frac{1}{\tau}\left(\vec{\Gamma}\cdot\vec{\lambda} + \vec{\Gamma}\cdot\frac{\vec{\Gamma}}{|\vec{\Gamma}|}\tanh\eta_{th}\right)\Bigg] dt' \\ &= -\frac{\hbar}{2\tau}\int_{t_0}^{t_0+T_{clk}} \left[\left(\vec{\Gamma}\cdot\vec{\lambda} + |\vec{\Gamma}|\tanh\eta_{th}\right)\right] dt' \\ &= \frac{\hbar}{2\tau}\int_{t_0}^{t_0+T_{clk}} \Big[(2\gamma)\cdot\lambda_x - \Phi\cdot\lambda_z \\ &\qquad - \sqrt{4\gamma^2+\Phi^2}\tanh\eta_{th}\Big] dt', \end{aligned} \tag{14}$$

where we use that $\vec{\Gamma}\cdot\vec{\Gamma} = |\vec{\Gamma}|^2$ as well as that the term $\vec{\Gamma}\cdot(\vec{\Gamma}\times\vec{\lambda})$ vanishes, since $\vec{\Gamma}\times\vec{\lambda}$ is always orthogonal to $\vec{\Gamma}$ and, thus, the scalar product yields 0.

While the above Eqs. (12)-(14) explicitly describe how to calculate the energy terms of interest, there is one further equation that becomes useful for estimating the actual precision/error of the calculations. In fact, assuming that the system is always going towards the thermal steady-state, as indicated in [31] and [29], the energy terms must sum to zero, i. e.

$$E_{total} = E_{env} + E_{clk} + E_{IO} = 0. \tag{15}$$

C. Tool for Energy Analysis

The considerations provided above now allow for a precise evaluation of the energy dissipation in QCA circuits. Thus far, the approach presented in [33] is considered state-of-the-art for this purpose. However, this method comes with major shortcomings as it e. g. solely focuses on determining upper bounds for the energy dissipation. Therefore, the authors reduced the formula for estimation of the energy transfer between QCA cell and environment (see Eq. (14)) under the assumption of non-adiabatic clocking, i. e. abrupt switching of the clock signal [1]. As a consequence, the modeling of adiabatic clocking, a fundamental requirement for a low energy dissipation, is not supported. Further, the authors assume that a QCA cell can solely possess the polarizations +1 and -1 (ignoring any intermediate values) and support only very limited manipulation of the technology parameters.

In order to overcome these drawbacks, we chose to integrate the above model for energy analysis into the *QCADesigner* [35], [36] which is widely applied for the design and simulation of QCA circuits. This tool is well-suited as it permits the definition of material, design and simulation parameters (which are listed with their standard values in Table I), but even more as the built-in *Coherence Vector Simulation Engine* (CVSE) already implements the quantum state model reviewed in Section III-A.

The CVSE is a fixed time step transient analysis. In each iteration step new values for the components of the coherence vector $\vec{\lambda}$ and the tunneling energy γ are calculated. While the latter is directly related to the clock signal, which is defined externally, the components of $\vec{\lambda}$ are estimated by numerically solving the differential equation shown in Eq. (6). As mentioned above, the polarization corresponds to the third component of $\vec{\lambda}$, while the kink energies E_{kink} are precomputed values following from the design architecture[2].

In order to incorporate energy analysis, we added to each iteration step the estimation of the differential values of all energy components defined in equations (12)-(14). In the resulting tool, which we term *QCADesigner-E*[3], we further enabled the configuration of a more realistic clock signal with Gaussian shaped slopes (rather than the standard ramp slopes) and also changed the characteristics of stimulated input signals

[2]The kink energy of a pair of cells is a static measure and can be estimated via the electrostatic interaction between all the charges of both cells with: $E_{kink}^{i,j} = \frac{1}{4\pi\epsilon_0\epsilon_r}\left(\sum_{L=1}^{4}\sum_{M=1}^{4}\frac{q_M^{i,1}q_L^{j,1} - q_M^{i,1}q_L^{j,-1}}{|d_{M,L}^{i,j}|}\right)$, where ϵ_0 refers to the permittivity in vacuum, ϵ_r is the relative permittivity of the material used to fabricate the QCA cell, $|d_{M,L}^{i,j}|$ means the distance between the dots M and L of cells i and j, and $q_D^{c,pol}$ is the value of the charge in dot D in cell c if c has polarization $P_c = pol$.

[3]The tool has been made publicly available as open-source at [38].

TABLE I: Adjustable Technology Parameters

(a) Parameters of standard *QCADesigner*

Parameter	Description	Standard Value
Size	Size of a quantum dot	5 nm
Cell area	Dimensions of each cell	18 nm x 18 nm
Cell distance	Distance between two cells	20 nm
Layer distance	Distance between QCA layers in case of multi-layer crossing [37]	11.5 nm
τ	Relaxation time	1E-15 s
γ_H	Max. saturation energy of clock signal	9.8E-22 J
γ_L	Min. saturation energy of clock signal	3.8E-23 J
ϵ_r	Relative permittivity of material for QCA system	12.9*
Temp	Operating temperature	1 K
r_{effect}	Maximum distance between cells whose interaction is considered	80 nm†

* Relative permittivity of GaAs and AlGaAs

† Interaction effects between two cells decays inversely with the fifth power of its distance

(b) Additional parameters in *QCADesigner-E*

Parameter	Description	Standard Value
T_γ	Period of the clock signal	10E-12 s
γ_{slope}	Rise and fall time of the clock signal slopes	1E-12 s
γ_{shape}	Shape of clock signal slopes [RAMP/GAUSSIAN]	GAUSSIAN
T_{in}	Period of the input signals	10E-12 s
T_{sim}	Total simulation time	80E-12 s
T_{step}	Time interval of each iteration step	1E-17 s

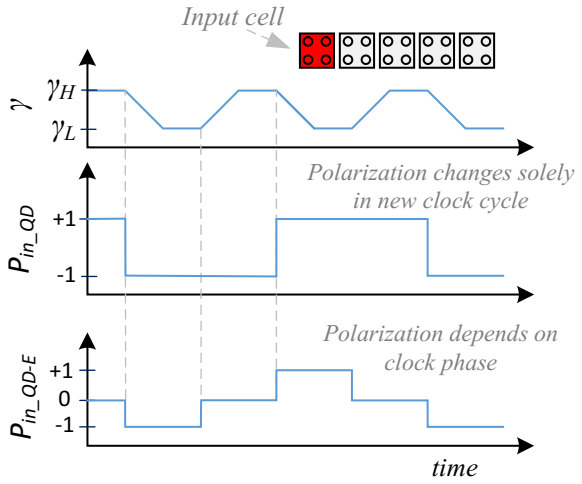


Fig. 5: Polarizations of QCA input cells in the standard QCADesigner (P_{in_QD}) and the enhanced QCADesigner-E simulator (P_{in_QD-E}) in dependence of the clock signal γ.

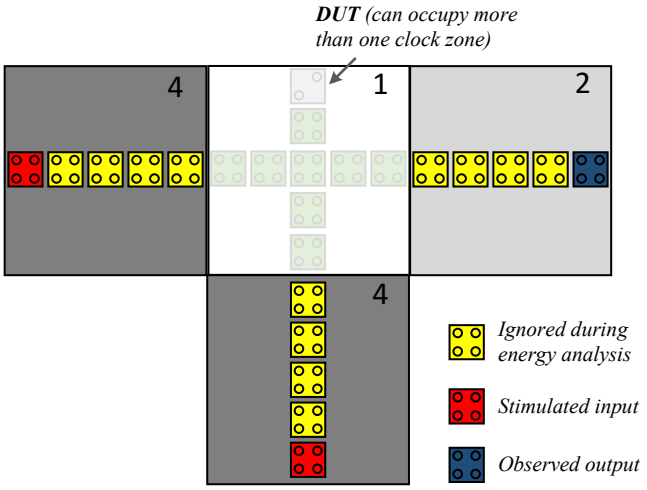


Fig. 6: Test bench for characterization of QCA gates.

such that their behavior is closer to a clocked QCA cell. More precisely, the polarization of input cells follows the phase of the related clock signal and is no longer constant throughout a complete clock cycle, as shown in Fig. 5. Nonetheless, in order to allow for a realistic energy analysis we additionally cascade this still somehow artificial signal and rather emulate an interaction with neighboring QCA structures by placing a few buffer cells between the stimulated inputs and the actual inputs of the considered design (the resulting test bench is shown in Fig. 6). Clearly, these cells have to be excluded from the overall energy analysis.

The energy dissipation of the whole circuit is eventually calculated via the summation of the E_{env} values for all QCA cells (with the exception of the cells excluded from energy analysis). As the integration interval starts with the rising slope of the clock, the dissipation of each clock zone is calculated individually before the values of all zones are added accordingly.

D. Validation of the Tool

In the following, we present the investigations that were performed to validate the results of the proposed QCADesigner-E tool with respect to the energy modeling from Section III-B.

To this end, recall that the energy dissipation of a QCA cell, i. e. its energy transfer with the environment, is calculated over a complete clock cycle, as indicated in Eq. (14). The related error ϵ_{env} of this value, caused by internal rounding errors and inadequate iteration step lengths (see Table I), follows from Eq. (15) as

$$\epsilon_{env} = E_{env} - (E_{clk} + E_{IO}). \tag{16}$$

In order to keep this error small, we determined the appropriate value of the time interval T_{step} of each iteration step for the standard parameter configuration listed in Table Ia. For this purpose, we implemented several test circuits and varied T_{step} depending on the relaxation time τ from 0.1τ to 0.001τ, i. e. from 1E-16 s to 1E-18 s. Table II lists the obtained results. It can be concluded that an iteration step

TABLE II: Error of estimated energy dissipation for different iteration step intervals. Wx_y means a wire with x cells and y cells in each clock zone.

Circuit	T_{step} = 1E-16 s			T_{step} = 1E-17 s			T_{step} = 1E-18 s		
	E_{env} [eV]	$\lvert\epsilon_{env}\rvert$ [eV]	$\lvert\epsilon_{env}\rvert$ [%]	E_{env} [eV]	$\lvert\epsilon_{env}\rvert$ [eV]	$\lvert\epsilon_{env}\rvert$ [%]	E_{env} [eV]	$\lvert\epsilon_{env}\rvert$ [eV]	$\lvert\epsilon_{env}\rvert$ [%]
W4_1	2.00E-4	1.98E-5	9.90 %	2.20E-4	2.07E-6	0.94 %	2.20E-4	2.70E-7	0.12 %
W8_8	2.60E-4	2.40E-5	9.23 %	2.90E-4	2.53E-6	0.87 %	2.90E-4	3.60E-7	0.12 %
W8_2	2.30E-4	2.00E-5	8.70 %	2.60E-4	2.10E-6	0.81 %	2.60E-4	2.60E-7	0.10 %
W16_2	2.90E-4	2.20E-5	7.59 %	3.30E-4	2.33E-6	0.71 %	3.30E-4	2.70E-7	0.08 %

TABLE III: Comparison of analytical and simulation results for selected test cases

Test Description	**Analytical result [meV]**	**Simulation result [meV]**
Cells c0 and c1, P_{c0} changes from 0 to 1 to 0 as Dirac pulse, both cells in same clock zone, full clock cycle	$E_{IO}^{c0,c1} = 0.67$	$E_{IO}^{c0,c1} = 0.68$
Cells and c0 and c1, P_{c0} changes from -1 to 1 as Dirac pulse, both cells in same clock zone, full clock cycle	$E_{IO}^{c0,c1} = 0.17$	$E_{IO}^{c0,c1} = 0.18$
Cells c0 and c1, P_{c0} changes from -1 to 1 as Dirac pulse, c0 in clock zone 1 and c1 in zone 3, full clock cycle	$E_{IO}^{c0,c1} = 1.47$	$E_{IO}^{c0,c1} = 1.48$
Cells c0 and c1, P_{c0} changes with γ_{slope} from -1 to 1, both cells in same clock zone, full clock cycle	$E_{IO}^{c0,c1}$ =0.34	$E_{IO}^{c0,c1} = 0.36$
Cells c0 and c1, P_{c0} changes with γ_{slope} from -1 to 1, both cells in same clock zone, halve clock cycle	$E_{clk}^{c1} = 5.7$	$E_{clk}^{c1} = 5.6$

time of 1E-17 s = 0.01τ offers a good trade-off between simulation time and calculation error, having in mind that the simulation time increases linearly with the reduction of the iteration step length. Consequently, all following analyses have been executed with $T_{step} = 0.01\tau$.

Next, we constructed several test cases for which precise values for the energy flow could be derived analytically and compared those with the results of the tool. To this end, we focused solely on the estimation of the energy transfer between cells (E_{IO}) and with the clock signal (E_{clk}). Both require similar calculation steps as E_{env}, but the analytic calculations are much shorter and simpler. The following example shall demonstrate the nature of this study.

Example 5. *This test case relates to the estimation of the energy transfer $E_{IO}^{c0,c1}$ between two cells c_0 and c_1, during the change of polarization of the input cell c_0 as depicted in Fig. 7 using the parameters listed in Table Ia. For simplification, we assume that the change of the polarization P_{c0} of c_0 can be modeled with steep pulse slopes such that the differential $\frac{d\Phi}{dt}$ is only non-zero at the time the input changes. Further, we can reduce the term $\tanh\eta_{th}$ to 1 due to the low ambient temperature of $T = 1\ K$ (see also Eq. (5)). Given the parameters in Table Ia, the kink energy between c_0 and c_1 results to $E_{kink}^{c0,c1} = 1.48E\text{-}3$ eV. Hence, the minimal and maximal energies of the clock signal can be expressed as $\gamma_H = 4.33E_{kink}^{c0,c1}$ and $\gamma_L = 0.16E_{kink}^{c0,c1}$. Further, we assume that at t_0 and $t_1 = t_0 + T_{clk}/2$, both cells are in steady-state.*

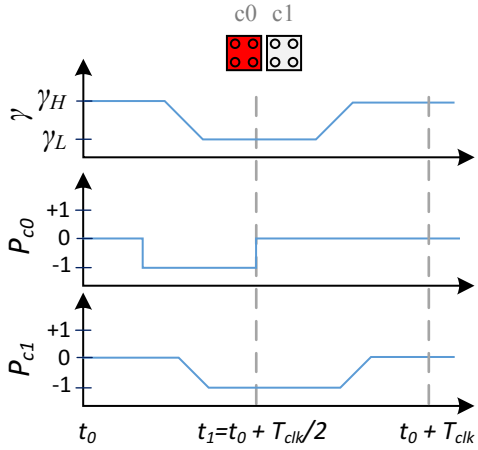


Fig. 7: Polarization P_{c0} and P_{c1} of input cell c_0 and cell c_1 and related clock signal γ.

Using Eq. (13) the expected energy results to:

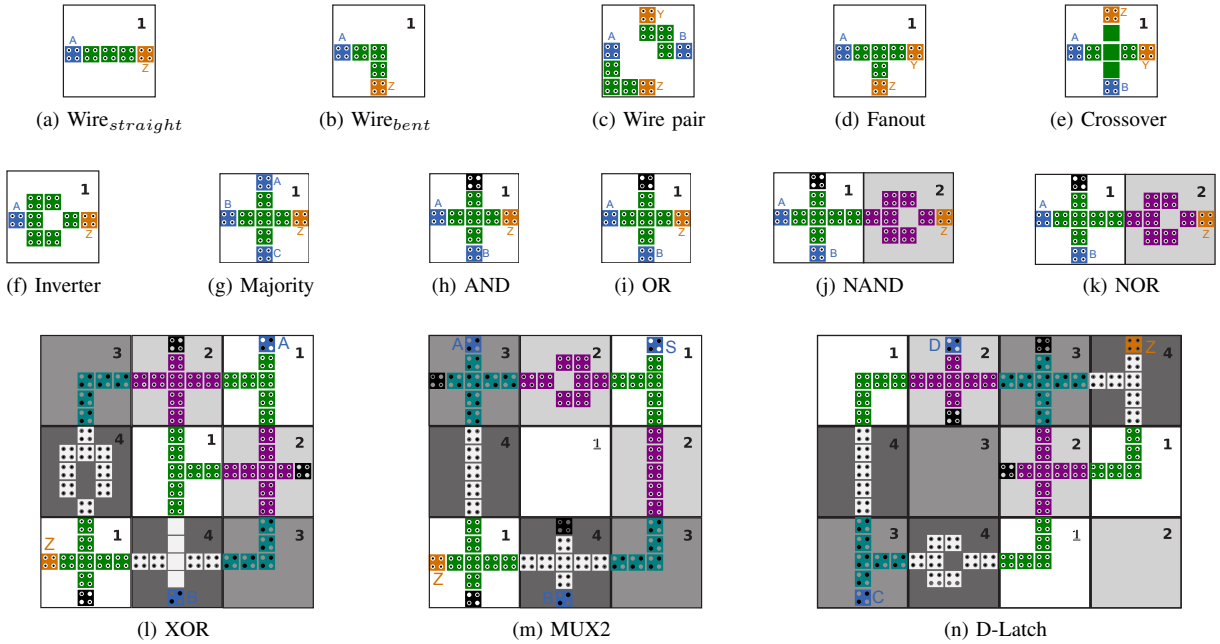
$$\begin{aligned} E_{IO}^{c0,c1} &= \frac{1}{2}\int_{t_0}^{t_1}\frac{d\Phi}{dt}dt' \cdot \vec{\lambda}_{ss,z}^{c1} + \frac{1}{2}\int_{t_1}^{t_0+T_{clk}}\frac{d\Phi}{dt}dt' \cdot \vec{\lambda}_{ss,z}^{c1} \\ &= \frac{\Delta P_{c0}\cdot E_{kink}^{c0,c1}}{2}\frac{\Phi(t_0)\cdot\tanh\eta_{th}}{\sqrt{\Phi(t_0)^2 + (2\gamma(t_0))^2}} \\ &+ \frac{\Delta P_{c0}\cdot E_{kink}^{c0,c1}}{2}\frac{\Phi(t_1)\cdot\tanh\eta_{th}}{\sqrt{\Phi(t_1)^2 + (2\gamma(t_1))^2}} \\ &= 0 + \frac{1\cdot E_{kink}^{c0,c1}}{2}\frac{-1\cdot E_{kink}^{c0,c1}}{\sqrt{\left(E_{kink}^{c0,c1}\right)^2 + \left(0.32E_{kink}^{c0,c1}\right)^2}} \\ &= -0.67E\text{-}3\ eV \end{aligned} \tag{17}$$

This value is very similar to the simulated result -0.68E-3 eV of the QCADesigner-E tool.

Table III lists executed tests and its analytical and simulation results which show only negligible differences. Although this study does not allow for a quantitative validation, its results can be understood as relevant indicators for the correctness of the implemented analysis tool.

IV. Resulting Model for Logic Synthesis

Based on the findings from the previous section, now the desired model for logic synthesis is derived which should replace conventional metrics used thus far and, by this, allow for a more dedicated logic synthesis of QCA circuits. To



(a) Wire$_{straight}$ (b) Wire$_{bent}$ (c) Wire pair (d) Fanout (e) Crossover

(f) Inverter (g) Majority (h) AND (i) OR (j) NAND (k) NOR

(l) XOR (m) MUX2 (n) D-Latch

Fig. 8: Layout of interconnection elements and standard gates of characterized library (adapted from [39]).

this end, we still assume the established design paradigms that (1) the elementary building blocks considered for synthesis are given by the standard gates introduced in [39] and depicted in Fig. 8 as well as that (2) a tile-based clocking scheme [26]–[28], [40] is employed. Finally, we follow the geometric model as proposed in [39] and also depicted in Fig. 8. Within this context and as motivated in Section I, the remaining issue is how to additionally incorporate a proper abstraction of the physical constraints into the model, i. e. area, delay, and energy dissipation. For area and delay, we utilized previously proposed solutions, namely:

- Area is defined by the amount of tiles a gate is occupying, whereas surrounded but not applied tiles are counted as well. That means, simple gates like an Inverter or Majority, depicted in Fig. 8f and 8g, have an area of 1, while the multiplexer as shown in Fig. 8m has area costs of 9. It should be noted that the area of a gate is not always defined in a rectangular fashion. For example, the area cost of the D-Latch shown in Fig. 8n is 11 (i. e. the right-bottom tile does not contribute to the area as it could be used by another element or wire).
- The gate delay is the time a signal requires to pass from a gate input to an output, whereas each input/output pair is called *timing arc*. The QCA gate delay is independent of the input signals, because there is no difference in the timing behavior for polarizations $P = -1$ and $P = +1$ [1]. Also, input slope and output load, known from logic gates in CMOS technologies, can be ignored in QCA circuits. This can be explained by the deterministic behavior of a clock zone, as by definition, the polarizations of all cells in a clock zone are stabilized during a clock phase [1]. Consequently, the gate delay follows from the clock frequency and the amount of clock zones, i. e. tiles, between gate input and output. That means, the delays for all *timing arcs* of gates that occupy just one tile, like Majority or AND, are 1. In case of more complex gates like the D-Latch, the delays for the *timing arcs* might differ.

This leaves the question how energy dissipation is abstracted. As stated above, the findings from Section III are utilized for this purpose. Recall that the energy dissipation of a gate depends on the particular inputs as well as the applied clocking scheme (as well as other technology parameters). As a consequence, the energy dissipation of each gate can be estimated for fast clock signals with steep slopes as well as slow signals with smooth slopes that explore the adiabaticity of the system (see also Section III-B). Hence, we provide corresponding energy dissipation values for applications with short delays but high energy dissipation as well as for designs with longer delays but reduced energy dissipation. As a mixture of both modes can only be coordinated on higher abstraction layers, it is not considered here.

Eventually, this results in a model as shown in Table IV. Here, for each interconnection element such as wires and fanout as well as for each gate type, the respectively needed area (with respect to the number of tiles), the delay (distinguished for each timing arc between the input signals A, B, C and the output signal Z), as well as the energy consumption (with respect to all possible input assignments) are listed. As discussed above, we provide numbers for two different modes (a regular mode with $f_{clk} = 25$ GHz and a fast mode with $f_{clk} = 100$ GHz). Note that, in case other clock frequencies (or any other differing technology parameters) are supposed to be applied, these numbers will change and the developed tool QCADesigner-E (presented in Section III) can be used to determine the respective values based on the findings from

TABLE IV: Energy-aware model for the logic synthesis of Quantum-Dot Cellular Automata[a]

		Area [tiles][c]	Delay[d] [clk zones]			Energy Dissipation [meV]													
						Regular mode (f_{clk} = 25 GHz) with respect to the following input assignments								Fast mode (f_{clk} = 100 GHz) with respect to the following input assignments							
			$A \rightarrow Z$	$B \rightarrow Z$	$C \rightarrow Z$	000	001	010	011	100	101	110	111	000	001	010	011	100	101	110	111
Interconn. Element	$\text{Wire}_{straight}$	1	1			0.09	0.09							0.82	0.82						
	Wire_{bent}	1	1			0.10	0.10							0.84	0.84						
	Wire pair	1	1			0.17	0.17	0.17	0. 17					1.60	1.61	1.61	1.60				
	Fanout	1	1			0.12	0.12							1.15	1.15						
	Crossover	1	1			0.28	0.28	0.28	0.28					2.57	2.58	2.58	2.57				
Logic Gate[b]	Inverter	1	1			0.13	0.13							1.19	1.19						
	Majority	1	1	1	1	0.15	0.82	0.82	0.82	0.82	0.82	0.82	0.15	1.41	1.77	1.78	1.77	1.77	1.78	1.77	1.41
	OR	1	1	1		0.18	0.79	0.79	0.12					1.30	1.52	1.54	1.19				
	AND	1	1	1		0.12	0.79	0.79	0.18					1.19	1.54	1.53	1.30				
	NOR	2	2	2		0.31	0.92	0.92	0.25					2.49	2.72	2.73	2.38				
	NAND	2	2	2		0.25	0.92	0.92	0.31					2.38	2.73	2.72	2.49				
	XOR	9	5	6		1.99	2.70	2.70	1.99					12.08	12.48	12.47	12.09				
	MUX2[e]	9	3	2	5	2.31	2.31	2.31	1.65	1.64	1.64	2.31	1.64	9.23	9.21	9.23	8.88	8.89	8.88	9.24	8.89
	D-Latch[f]	11	3	6		2.56	2.56	2.56	1.84	1.84	2.56	1.84	1.84	11.20	11.15	11.15	10.80	10.80	11.15	10.80	10.80

[a] Technology parameters taken from Table Ia, simulation parameters are: γ_{shape} = GAUSSIAN, T_{step} = 1E-17 s, γ_{slope} = 1E-12 s for f_{clk} = 25 GHz, and γ_{slope} = 1E-13 s for f_{clk} = 100 GHz, all characterizations executed with test bench depicted in Fig. 6.
[b] Related layouts are depicted in Fig. 8.
[c] Each tile contains 5x5 QCA cells, see also [28], [40].
[d] Delay is measured in numbers of clock zones a signal must pass from input to output pin. Having in mind that the clocks of the clock zones are shifted by quarters of the clock period, the actual delay follows from: $\frac{1}{4*f_{clk}} \times$ Delay [clk zones].
[e] Input order is A-B-S.
[f] Input order is D-C, input assignment order is /Z-D-C with /Z being the former state of Z.

above. To this end, the layout data for the gates and structures, together with the tool, have been made publicly available at [38] so that designers can generate the respectively needed model.

The resulting model is obviously much more accurate for QCA designs as conventional metrics applied thus far. However, also compared to more related endeavors such as reported in [41], much more precision is obtained. In fact, the model proposed in [41] simply combines the physical aspects area and delay in a simple single formula. By this, they provide a cost function which artificially considers two complementary cost issues as a single optimization criterion although they would require a multi-objective optimization approach. Besides that, this model completely ignores energy dissipation which is crucial for a technology such as QCA that is mainly motivated by its energy efficiency.[4]

Overall, Table IV provides the first model for QCA circuits which is applicable for logic synthesis of QCA circuits and, at the same time, comprehensively considers all important physical aspects of this emerging technology.

V. Accuracy of the Model and Effect of Interconnections

In order to estimate the accuracy of the proposed model, we compared the values obtained by the model proposed in Table IV with actual physical values obtained by simulations with the tool QCADesigner-E presented in Section III. More precisely, we considered five representative QCA circuits and explicitly implemented as well as analyzed them (as a whole) in the tool QCADesigner-E—yielding simulation results of their physical costs. Afterwards, we considered the logical structure of those circuits and extracted elementary components in terms of minimal sized units, i. e. gates and interconnection elements with size one. Using the extracted components and the proposed model from Table IV, we determined the corresponding model values for area, delay, and energy dissipation for all possible assignments of the circuit's primary inputs—yielding the model results of their physical costs.

The comparison of both costs is summarized in Table V—providing the minimal, maximal, and average difference of the model values to the values obtained from QCADesigner-E. In fact, area and delay can be abstracted with basically perfect precision to the logical level. But also with respect to energy dissipation, the proposed model turns out to be rather accurate. In fact, the abstracted values from the model are off from the actual physical values by at most 5 %. This is more than sufficient for a consideration at the logical level and confirms the accuracy of the model.

Besides that, we also used the model to evaluate the effect of interconnection elements such as wires or fanouts in the costs considerations of QCA circuits. To this end, recall that existing synthesis approaches for QCA circuits usually ignore these elements and solely focus on the logic components, i. e. the gates [9]–[12], [41]. This might be motivated by the fact that interconnections are also frequently ignored by logic synthesis approaches proposed for conventional circuitry. However, as shown by our findings from above and reflected in the proposed model from Table IV, interconnection elements may have a significant impact to the overall costs.

In order to evaluate that, we conducted the same comparison as discussed above and summarized in Table V—just with

[4]Provided the already discussed drawbacks, it is almost negligible that the model from [41] does not consider interconnections, although they have a significant impact to the overall energy dissipation (as shown in the next section).

TABLE V: Error of proposed model

	Area	Delay	Energy Dissipation					
			Slow			Fast		
			Min	Max	Avg	Min	Max	Avg
XOR	0 %	0 %	3 %	4 %	3 %	3 %	3 %	3 %
MUX2	0 %	0 %	2 %	2 %	2 %	0 %	1 %	1 %
D-Latch	0 %	0 %	1 %	2 %	1 %	0 %	1 %	0 %
10INV chain	0 %	0 %	0 %	0 %	0 %	1 %	1 %	1 %
Full-Adder	0 %	0 %	3 %	3 %	3 %	4 %	4 %	4 %

TABLE VI: Error without interconnection modeling

	Area	Delay	Energy Dissipation					
			Slow			Fast		
			Min	Max	Avg	Min	Max	Avg
XOR	56 %	63 %	20 %	24 %	22 %	35 %	36 %	35 %
MUX2	56 %	80 %	18 %	26 %	22 %	40 %	42 %	41 %
D-Latch	64 %	50 %	26 %	34 %	30 %	50 %	52 %	51 %
10INV chain	47 %	47 %	38 %	38 %	38 %	39 %	39 %	39 %
Full-Adder	88 %	79 %	83 %	84 %	84 %	83 %	83 %	83 %

the difference that we completely ignored all interconnections when determining the model costs (i. e. we basically ignored all interconnection elements in Table IV and assumed all costs of these elements to be zero). The obtained results are summarized in Table VI. This clearly confirms the importance of interconnections in QCA circuits. In fact, not considering interconnections yields model values which are off by at least 20 %—in most of the cases much higher errors can be observed. Obviously, this is unacceptable for a proper abstraction and, hence, synthesis approaches have to explicitly take the interconnections of the circuit into consideration if they are aiming for producing QCA circuits that are accurately optimized with respect to physical costs in terms of area, delay, and energy dissipation.

VI. Conclusions

In this work, we proposed a model for the logic synthesis of QCA circuits that, for the first time, considers and abstracts all main physical aspects of this emerging technology—in particular, energy dissipation which is essential for a technology that is mainly motivated by its energy efficiency. To this end, we reviewed in detail how energy flows in QCA systems can be precisely modeled as well as analyzed and provided a corresponding analysis methodology as well as a tool (QCADesigner-E) that implements this methodology. The tool as well as layout data allowing the consideration of further QCA technology parameters have been made publicly available at [38].

Based on that, a model for logic synthesis was derived which basically is summarized in Table IV. Comparisons with actual physical values determined by simulations within the QCADesigner-E tool confirmed the accuracy of the proposed model.

The findings and the resulting model allow to re-think how logic synthesis of QCA circuits shall be conducted in the future. The main lessons learned are summarized by (1) the new cost functions which should be considered (i. e. Table IV) as well as (2) the fact that interconnections matter in this technology and should not be ignored (as frequently done in logic synthesis for conventional circuitry but also all proposed cost metrics proposed for QCA circuits thus far). This eventually provides the basis for a new generation of synthesis approaches at the logic level that are explicitly dedicated to the QCA technology.

References

[1] C. S. Lent and P. D. Tougaw, "A device architecture for computing with quantum dots," *Proceedings of the IEEE*, vol. 85, no. 4, pp. 541–557, Apr 1997.

[2] C. Lent, P. Tougaw, W. Porod, and G. Bernstein, "Quantum cellular automata," *Nanotechnology*, vol. 4, no. 1, p. 49, 1993.

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

[4] J. Pitters, L. Livadaru, M. Haider, and R. Wolkow, "Tunnel coupled dangling bond structures on hydrogen terminated silicon surfaces," *Journal of Chemical Physics*, vol. 134, no. 6, 2011.

[5] A. O. Orlov, I. Amlani, R. K. Kummamuru, R. Ramasubramaniam, G. Toth, C. S. Lent, G. H. Bernstein, and G. L. Snider, "Experimental demonstration of clocked single-electron switching in quantum-dot cellular automata," *Applied Physics Letters*, vol. 77, no. 2, pp. 295–297, 2000.

[6] 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.

[7] A. O. Orlov, R. Kummamuru, R. Ramasubramaniam, C. S. Lent, G. H. Bernstein, and G. L. Snider, "Clocked quantum-dot cellular automata shift register," *Surface Science*, vol. 532, no. C, pp. 1193 – 1198, 2003.

[8] M. B. Haider, J. L. Pitters, G. A. DiLabio, L. Livadaru, J. Y. Mutus, and R. A. Wolkow, "Controlled coupling and occupation of silicon atomic quantum dots at room temperature," *Phys. Rev. Lett.*, vol. 102, p. 046805, Jan 2009.

[9] M. Momenzadeh, J. Huang, M. B. Tahoori, and F. Lombardi, "Characterization, test, and logic synthesis of and-or-inverter (aoi) gate design for qca implementation," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 12, pp. 1881–1893, Dec 2005.

[10] R. Zhang, K. Walus, W. Wang, and G. A. Jullien, "A method of majority logic reduction for quantum cellular automata," *IEEE Transactions on Nanotechnology*, vol. 3, no. 4, pp. 443–450, Dec 2004.

[11] K. Kong, Y. Shang, and R. Lu, "An optimized majority logic synthesis methodology for quantum-dot cellular automata," *IEEE Transactions on Nanotechnology*, vol. 9, no. 2, pp. 170–183, March 2010.

[12] M. G. A. Martins, V. Callegaro, F. S. Marranghello, R. P. Ribas, and A. I. Reis, "Majority-based logic synthesis for nanometric technologies," in *14th IEEE International Conference on Nanotechnology*, Aug 2014, pp. 256–261.

[13] M. Walter, R. Wille, D. Grosse, F. S. Torres, and R. Drechsler, "An exact method for design exploration of quantum-dot cellular automata," in *Design, Automation and Test in Europe (DATE)*, 2018.

[14] S. Perri and P. Corsonello, "New methodology for the design of efficient binary addition circuits in qca," *IEEE Transactions on Nanotechnology*, vol. 11, no. 6, pp. 1192–1200, 2012.

[15] E. Fazzion, O. Fonseca, J. Nacif, O. Neto, A. Fernandes, and D. Silva, "A quantum-dot cellular automata processor design," in *27th Symposium on Integrated Circuits and Systems Design (SBCCI)*, Sept 2014, pp. 1–7.

[16] M. Kianpour and R. Sabbaghi-Nadooshan, "A novel quantum-dot cellular automata clb of fpga," *Journal of Computational Electronics*, vol. 13, no. 3, pp. 709–725, 2014.

[17] R. Brayton and A. Mishchenko, "ABC: An academic industrial-strength verification tool," in *CAV*, 2010, pp. 24–40.

[18] R. K. Nath, B. Sen, and B. K. Sikdar, "Optimal synthesis of qca logic circuit eliminating wire-crossings," *IET Circuits, Devices Systems*, vol. 11, no. 3, pp. 201–208, 2017.

[19] A. Trindade, R. Ferreira, J. Nacif, D. Sales, and O. Neto, "A placement and routing algorithm for quantum-dot cellular automata," in *29th Symposium on Integrated Circuits and Systems Design (SBCCI)*, Sept 2016, pp. 1–6.

[20] P. Wang, M. Niamat, and S. Vemuru, *Majority Logic Synthesis Based on Nauty Algorithm*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 111–132.

[21] R. K. Brayton, R. Rudell, A. Sangiovanni-Vincentelli, and A. R. Wang, "Mis: A multiple-level logic optimization system," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 6, no. 6, pp. 1062–1081, November 1987.

[22] W. Liu, E. Swartzlander Jr, and M. O'Neill, *Design of semiconductor QCA systems*. Artech House, 2013.

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

[24] E. Blair and C. Lent, "An architecture for molecular computing using quantum-dot cellular automata," in *IEEE-NANO 2003*, vol. 1. IEEE, 2003, pp. 402–405.

[25] K. Hennessy and C. Lent, "Clocking of molecular quantum-dot cellular automata," *Journal of Vacuum Science & Technology B*, vol. 19, no. 5, pp. 1752–1755, 2001.

[26] V. Vankamamidi, M. Ottavi, and F. Lombardi, "Two-dimensional schemes for clocking/timing of qca circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 1, pp. 34–44, Jan 2008.

[27] M. Janez, P. Pecar, and M. Mraz, "Layout design of manufacturable quantum-dot cellular automata," *Microelectronics Journal*, vol. 43, no. 7, pp. 501–513, 2012.

[28] C. Campos, A. Marciano, O. Neto, and F. S. Torres, "USE: A universal, scalable, and efficient clocking scheme for QCA," *IEEE Trans. on CAD of Integrated Circuits and Systems*, vol. 35, no. 3, pp. 513–517, 2016.

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

[30] G. Mahler and V. Weberruß, *Quantum Networks: Dynamics of Open Nanostructures*. Springer Berlin Heidelberg, 1998.

[31] M. Taucer, F. Karim, K. Walus, and R. A. Wolkow, "Consequences of many-cell correlations in clocked quantum-dot cellular automata," *IEEE Transactions on Nanotechnology*, vol. 14, no. 4, pp. 638–647, July 2015.

[32] P. D. Tougaw and C. S. Lent, "Dynamic behavior of quantum cellular automata," *Journal of Applied Physics*, vol. 80, no. 8, pp. 4722–4736, 1996.

[33] S. Srivastava, S. Sarkar, and S. Bhanja, "Estimation of upper bound of power dissipation in qca circuits," *IEEE Transactions on Nanotechnology*, vol. 8, no. 1, pp. 116–127, Jan 2009.

[34] G. Chen, D. A. Church, B.-G. Englert, C. Henkel, B. Rohwedder, M. O. Scully, and M. S. Zubairy, *Quantum Computing Devices: Principles, Designs, and Analysis*, 1st ed., ser. Applied Mathematics and Nonlinear Science Series. Chapman and Hall/CRC, 2007.

[35] K. Walus, T. J. Dysart, G. A. Jullien, and R. A. Budiman, "Qcadesigner: A rapid design and simulation tool for quantum-dot cellular automata," *IEEE Trans. Nanotechnol.*, vol. 3, no. 1, pp. 26–31, Mar. 2004.

[36] K. Walus and G. A. Jullien, "Design tools for an emerging soc technology: Quantum-dot cellular automata," *Proceedings of the IEEE*, vol. 94, no. 6, pp. 1225–1244, June 2006.

[37] B. IL and P. P., "Two-layer synchronized ternary quantum-dot cellular automata wire crossings," *Nanoscale Research Letters*, vol. 7, no. 1, 2012.

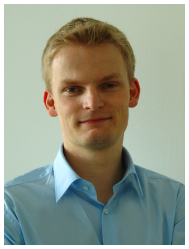
[38] Qcadesigner-e. [Online]. Available: https://github.com/FSillT/QCADesigner-E

[39] D. A. Reis, C. A. T. Campos, T. R. B. S. Soares, O. P. V. Neto, and F. S. Torres, "A methodology for standard cell design for qca," in *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2016, pp. 2114–2117.

[40] J. Huang, M. Momenzadeh, L. Schiano, M. Ottavi, and F. Lombardi, "Tile-based qca design using majority-like logic primitives," *J. Emerg. Technol. Comput. Syst.*, vol. 1, no. 3, pp. 163–185, Oct. 2005.

[41] W. Liu, L. Lu, M. O'Neill, and E. E. Swartzlander, "A first step toward cost functions for quantum-dot cellular automata designs," *IEEE Transactions on Nanotechnology*, vol. 13, no. 3, pp. 476–487, May 2014.

Frank Sill Torres (M'15) received the Diploma and Dr.-Ing. degrees in Electrical Engineering from the University of Rostock, Germany, in 2002 and 2007, respectively. He was with the Laboratory for Optronics and Microtechnologies, Federal University of Minas Gerais (UFMG), Brazil, from 2007 to 2009. Since 2010, he is Adjunct Professor at the Department of Electronic Engineering at the UFMG where he coordinates the ASIC Reliability Group. He is also permanent member of the Post-graduation programs in Electrical Engineering of the UFMG and the Federal University of Manaus, Brazil. In 2017, he was with the Group for Computer Architecture, Institute of Computer Science, University of Bremen, Germany. His current research interests include emerging technologies, design for reliability, and low-power integrated circuit design, and he is author of more than 80 publications in scientific journals, congresses and workshops. Frank Sill Torres was a member of several conference committees including e.g. ISCAS, LATS, MWSCAS, SBCCI and the chair of the SBCCI 2016. He is an Associate Researcher of the Brazilian National Research Council (CNPq).



Robert Wille (M'06-SM'09) is a Full Professor at the Johannes Kepler University Linz, Austria. He received the Diploma and Dr.-Ing. degrees in computer science from the University of Bremen, Germany, in 2006 and 2009, respectively. He was with the Group of Computer Architecture at the University of Bremen and with the German Research Center for Artificial Intelligence (DFKI). Additionally, he worked as Lecturer at the University of Applied Science in Bremen, Germany, and as visiting professor at the University of Potsdam, Germany, and the Technical University Dresden, Germany. His research interests include the design of circuits and systems for both conventional and emerging technologies. This includes work in the domain of verification and proof engines as well as the synthesis and optimization of quantum circuits, reversible circuits, optical circuits, and microfluidic biochips. Since 2007, he published more than 200 journal and conference papers in this area and was repeatedly awarded (e.g. with a Best Paper Award from ICCAD in 2013 and from FDL in 2010 as well as a Young Researchers Award from ISMVL in 2008). Besides that, he served Guest Editor for the ACM's JETC, Springer's LNCS, and the MVLSC. Additionally, he was General Chair for ISMVL and FDL, PC Chair for ISMVL, FDL, and others, as well as frequent PC member and track chair for conferences such as ASP-DAC, DAC, DATE, ICCAD, and more.

Philipp Niemann (M'15) Philipp Niemann received a Diploma degree in mathematics and a PhD degree in computer science from the University of Bremen, Germany, in 2012 and 2016, respectively. There, he worked in the Group for Computer Architecture under the supervision of Prof. Dr. Rolf Drechsler and Prof. Dr. Robert Wille. In 2017, he joined the Cyber-Physical Systems (CPS) department at the German Research Center for Artificial Intelligence (DFKI) in Bremen. His research interests are in the design of reversible and quantum circuits with a focus on decision diagrams as well as in the verification of formal models. Philipp Niemann published several papers on international conferences such as ASP-DAC, DATE, RC, MEMOCODE, and MoDELS.

Rolf Drechsler (M'94-SM'03-F'15) received the Diploma and Dr. Phil. Nat. degrees in computer science from J.W. Goethe University Frankfurt am Main, Frankfurt am Main, Germany, in 1992 and 1995, respectively. He was with the Institute of Computer Science, Albert-Ludwigs University, Freiburg im Breisgau, Germany, from 1995 to 2000, and with the Corporate Technology Department, Siemens AG, Munich, Germany, from 2000 to 2001. Since October 2001, he has been with the University of Bremen, Bremen, Germany, where he is currently a Full Professor and the Head of the Group for Computer Architecture, Institute of Computer Science. In 2011, he additionally became the Director of the Cyber-Physical Systems group at the German Research Center for Artificial Intelligence (DFKI) in Bremen. His current research interests include the development and design of data structures and algorithms with a focus on circuit and system design. He is an IEEE Fellow. Rolf Drechsler was a member of Program Committees of numerous conferences including e.g. DAC, ICCAD, DATE, ASP-DAC, FDL, MEMOCODE, FMCAD, Symposiums Chair ISMVL 1999 and 2014, Symposium Chair ETS 2018, and the Topic Chair for "Formal Verification" DATE 2004, DATE 2005, DAC 2010, as well as DAC 2011. He received best paper awards at the Haifa Verification Conference (HVC) in 2006, the Forum on specification & Design Languages (FDL) in 2007 and 2010, the IEEE Symposium on Design and Diagnostics of Electronic Circuits and Systems (DDECS) in 2010 and the IEEE/ACM International Conference on Computer-Aided Design (ICCAD) in 2013. He is an Associate Editor of IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, ACM Journal on Emerging Technologies in Computing Systems, and further journals.