Area and Energy Efficient QCA Based Compact Serial Concatenated Convolutional Code Encoder

Premananda B.S.\textsuperscript{1}, Dhanush T.N.\textsuperscript{2}, Vaishnavi S. Parashar\textsuperscript{3}
\textsuperscript{1,2,3} Department of Electronics and Telecommunication Engineering, RV College of Engineering, Bengaluru, India

premanandabs@gmail.com, dneel31051998@gmail.com, vaishnavisparashar@gmail.com

Abstract. Quantum-dot Cellular Automata (QCA) is a transistor-less technology known for its low power consumption and higher clock rate. Serial Concatenated Convolutional Coding (SCCC) encoder is a class of forward error correction. This paper picturizes the implementation of the outer encoder as a (7, 4, 1) Bose Chaudhary Hocquenghem encoder that serves the purpose of burst error correction, a pseudo-random inter-leaver used for permuting of systematic code words and finally the inner encoder which is used for the correction of random errors in QCA. Two different architectures of the SCCC encoder have been proposed and discussed in this study. In the proposed two architectures, the first based on external clock signals whereas the second based on internal clock generation. The sub-blocks outer encoder, pseudo-random inter-leaver and inner encoder of the SCCC encoder are optimized, implemented and simulated using QCADesigner and then integrated to design a compact SCCC encoder. The energy dissipation is computed using QCADesigner-E. The proposed SCCC encoder reduced the total area by 46% and energy dissipation by 50% when compared to the reference SCCC encoder. The proposed encoders are more efficient in terms of cell count, energy dissipation and area occupancy respectively.

Keywords: Energy dissipation, FEC, QCA, SCCC

1. Introduction

The bandwidth, error tolerance and latency play an important role in a communication system. There is an increase in the requirement of bandwidth, and the margin for error and latency has reduced, thereby needing new techniques to increase the bandwidth and improve the quality of transmission [9]. One such solution for the aforementioned problem is to use the forward error correction (FEC) technique. FEC is an error correction technique in which the redundant bits appended with the data transmitted. The redundant bits added to the data used at the receiver to check if the received bits are error-free. The error can be detected and corrected in the receiver, need for retransmission is eliminated, thereby improving the efficiency of the usage of bandwidth [6].

In the FEC coding scheme, trellis coding is applied twice, once on the direct data sequence which is being transmitted and once, on the interleaved data. Trellis coding used twice, the error correcting capacity increased in SCCC. As the signal-to-noise ratio (SNR) increases, the Bit Error Rate (BER) of parallel-concatenated convolutional codes might not continue to decrease, reaching the so-called error floor. To achieve superior error-free performance, SCCCs were preferred. Due to the negligible error floor, SCCCs used in deep space communication applications [7].

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
Published under licence by IOP Publishing Ltd
SCCC encoder play important role in the communication domain where data security has become important criteria to be considered for efficient transmission of the data between the source and destination. The SCCC encoder used in channel encoding phase of the transmitter in the communication block. The increase in the frequency of operation and data rate enables an increase of the switching power, thereby increasing the power consumption of the system [8]. To overcome the problems of high-power consumption, density beyond the submicron level and low speed Quantum cellular dot automata (QCA) used for high power applications [1]. Adiabatic logic, [5] can be used but requires more area. When compared to the implementation of the same circuit using CMOS technology, QCA has better performance in terms of device density and energy dissipation [12]. All the basic cells required for designing the SCCC encoders implemented first in QCA, integrated to obtain the required design. An expandable parallel-to-serial converter (PSC) and a scalable serial-to-parallel converter (SPC) designed using the 4-clock mechanism of QCA.

A scalable pseudo-random inter-leaver (PRI) is implemented which can also be used as a last-in-first-out converter. The overall reduction in the area and energy dissipation achieved because of reanalysis of the existing cells and logic and increasing the reusability of the cells available.

This paper mainly focuses on designing compact SCCC encoders using the QCA technology. The flow of the work is as follows: Following the introduction in section 1, the literature review discussed in section 2 with the inferences related to the designing of the SCCC encoder. QCA concepts such as QCA cell, inverter, and majority voter and simulation engine along with the simulation setups discussed in section 3. Section 4 discussed the implementation of the SCCC encoder using the bottom-up approach where the sub-components designed first, and integrated to implement the SCCC encoder. The results obtained from every component discussed in section 5 with the conclusion and future work discussed in the last part of the paper.

2. Related Works
This section provides a summary of the work carried out as a process of designing required circuits to meet the application criteria.

A quantum-dot cellular automaton is a transistor-less technology meant to be the replacement for the existing CMOS technology. QCA designer provides circuits with high processing speed, low energy dissipated and a major advantage of having the circuit perform at the nanoscale level. Three input majority gates are the most elementary circuit, which used for designing any QCA based circuits. The majority voter and inverter gate plays a vital role in designing the required digital circuits. The importance of these structures discussed in [8].

An efficient SPC and PSC designed and implemented in the SCCC encoder [7]. The paper provided an idea for the implementation of parity bit generation with the help of code generator polynomials. The disadvantages of the designs in [7] are that the structures were huge, dissipated more energy and occupied a larger area. PSC and SPC play an important role in the implementation of both the outer encoder (OE) and the PRI. An efficient design of the PSC and SPC discussed in [3].

The evaluation of applications of the D flip-flops led to the design of shift registers, application of the flip-flops [4]. These shift registers used in the implementation of the SPC and PSC. The previous architectures of D flip-flops profusely helped in designing the required shift register circuits that found to be a vital part of designing the OE and inter-leaver circuit of the SCCC encoder. The resulting shift register circuits are more efficient than the ones mentioned in [8].

In the conventional circuits, Bose-Chaudhuri-Hocquenghem (BCH) codes are generated using linear feedback shift registers [9]. The BCH encoders implemented by applying coplanar XOR gates. The convolution code encoders for various constraint lengths are implemented using XOR gates. The energy dissipation, cell count and area for each of the components of SCCC encoder tabulated. The SCCC encoder divided into three fragments namely, OE, inter-leaver and inner encoder (IE). The designing of these fragments consists of SPC and PSC designed using the shift registers in [10].
The proposed architectures in this paper are efficient when compared to the existing architectures. A thorough analysis of existing designs, reanalysis of the logic of these designs and reusability of the cells resulted in compact design.

3. QCA Background
QCA, a nanoscale technology that known for its higher switching speeds and lower energy dissipation used to design sequential and combinational digital circuits. This technology used for achieving higher speed and lesser energy dissipative designs to use for several purposes in the field of digital electronics and communication. QCA consists of QCA cell, majority voter and inverter, simulation engines and simulation type setup.

3.1 QCA Cell
The QCA cell is the basic element of QCA. The QCA cell used here has four dots, each dot representing the slots for electron movement. If the electron is similar to that of the electron in the atom of a cell, the holes are similar to the dots in the QCA cell. These QCA cells when arranged in the required manner would represent the operation of the desired circuits and arranged in such a way that they make way for the electron movement essential for the proper operation of the circuit [11]. The representation of the fundamental block of QCA i.e., the QCA cell shown in Figure 1. The usage of these cells in the designing of the majority voter and inverter discussed in the next section. The QCA cell also possesses dimensions, which are very essential in area calculation. These cells are 18nm in height and 18nm in width. Each cell separated from its adjacent cell by a distance of 2nm. The area of the entire circuit calculated using these dimensions based on the cell occupancy.

![Figure 1. QCA cell with electron and dot representation](image)

3.2 QCA Majority Voter and Inverter
The basic building blocks of the QCA are its cells. Similarly, the majority voter and inverter are the backbone of designing circuits in QCA. The majority voter is a gate with five cells, which gives a combinational output and is in two cases. In the first case, it consists of three inputs, a single output where the output will be the result of sum of the product of all the inputs. In the second case, when the polarity is given to either of the inputs ‘-1’ or ‘1’ it would result in the AND and OR and operation respectively [4]. The inverter gate also plays an important role in designing the digital circuits, digital building blocks such as XOR gates, which in turn used for designing adder circuits. The inverter returns the complement of the given input. Figure 2 represents the QCA inverter gate of 180° cross wire representation. The literals A represent the input and Z represents the output.

![Figure 2. QCA inverter gate](image)

3.3 Simulation Engine and Setup Type
The QCA circuits are simulated using QCADesigner 2.0.3 and QCADesigner-E for the calculation of the energy dissipation of the circuits [14]. The simulations of the QCA circuits performed in two ways, one bi-stable approximation and another coherence vector with energy (used in QCADesigner-E). The Bi-stable approximation deals with the simulation of the circuits without any time or period
dependencies. Thus, it is the fastest when compared to the coherence vector. On the other hand, the coherence vector with energy approximation deals with the simulation of circuits with time dependencies. This may be slower when compared to bi-stable but increases accuracy with the results and calculation of the energy dissipation. The bi-stable approximation and coherence vector simulation engine has default parametric values that define their operational ability.

Table 1 represents such parametric values. The simulation engines supported by the simulation type setup. The type setups are of two types, exhaustive and vector table. The inputs fed with involuntarily generated values in the exhaustive type, whereas the Vector Table used to feed inputs with the manually generated values. When designing larger circuits, the user prefer using the vector table since circuits needs testing in all possible cases. The dimensions of the QCA cells are also important in the QCA along with other attributes of QCA. The cell dimension is as follows with its height and width both 18 nm each with the QCA dot diameter being 5nm. Multiple layers can be created along with the main cell layer, which is default that is coplanar [2]. All the cells are easily scalable and scaled in terms of nanometre. Runge-kutta and euler method are some of the numerical analyses implemented in the simulation engine.

Table 1. Parametric values of simulation engines

| Parametric Values                  | Bi-stable Approximation | Coherence Vector |
|------------------------------------|-------------------------|------------------|
| Number of samples                  | 12800                   | 12800            |
| Convergence Tolerance              | 0.001                   | NA               |
| Radius of Effect                   | 65nm                    | 80nm             |
| Relative Permittivity              | 12.90                   | NA               |
| Clock High                         | 9.8 e-22 J              | 9.8 e-22 J       |
| Clock Low                          | 3.80 e-23 J             | 3.80 e-23 J      |
| Layer Separation                   | 11.50 nm                | 11.50 nm         |
| Clock Amplitude Factor             | 2.00                    | 2.00             |
| Maximum Iteration per sample       | 100                     | NA               |
| Temperature                        | NA                      | 1.000K           |

4. Implementation of SCCC Encoder

The implementation of the SCCC encoder analysed in QCA using the bottom-up approach. The primary blocks of the SCCC encoder are OE, inter-leaver and inner encoder. The Inter-leaver constructed here is pseudo-random in nature. The block diagram of the SCCC encoder [15] is shown in Figure 3. The components of SCCC are an OE, a linking PRI and an IE. The message bit D serially fed to the OE. Original message bit D is encoded using OE, parity bits are added to D and a systematic code word C is obtained for OE. The code word C is then given to PRI where it is permuted to obtain I. Finally, the permuted bits I provided to IE where it is coded using a convolution code encoder to get the encoded bits Y0 and Y1. The OE is employed using BCH code encoder.

4.1 Outer Encoder

The OE is the primary stage of the SCCC encoder which is a (7, 4, 1) BCH encoder that is used for burst error correction. The OE comprises a 4-bit SPC along with a parity bit generator and a 3-bit PSC. Figure 4 represents the block diagram of an OE.

![Figure 3. Block diagram of SCCC encoder](image-url)
The 4-bit SPC is the first stage to be designed in an OE that parallelizes the message bits (D) which is shown of the polynomial form $A(x)$ into message bits as $[A_0, A_1, A_2, A_3]$. The parallel bits are represented in the form of $[\alpha, \beta, \gamma, \delta]$ where $\alpha$ represents $A_0$ and $\beta, \gamma$ and $\delta$ represent $A_1, A_2$ and $A_3$ respectively. The outputs are represented in the form of polynomials shown in equation (1) where $D(x)$ represents the message bits as a function of $x$.

$$D(x) = \alpha + \beta x + \gamma x^2 + \delta x^3$$

The proposed 4-bit SPC consists of a different majority gate approach to previous designs and is deficient in the presence of a separate clock signal along with the absence of multiplexers in their implementation. Figure 5 shows the proposed design of a 4-bit SPC where $D$ is the message bits and $A_3-A_0$ is the combination of the parallelized set of message bits. The 4-bit SPC followed by the parity bit generator, which is the second stage of the OE.

The parity bit generator generates the parity bits that constitute the error correction part of the encoder. These parity bits decide whether the message bits received are error-free or not. The parity bits can be odd or even parity. These parity bits are special cases of cyclic redundancy codes. These play a vital role in error detection at the receiver end. The 3-bit parity bit generator constructed using modulo-2 adders. The construction of the parity generators using the modulo-2 adders shown in Figure 6, depicts the use of the modulo-2 adders in the implementation of the parity generators of outer (7, 4, 1) BCH encoder [7]. The parallel bits are the outputs of the 4-bit SPC used to obtain three parity bits from the generator. The bits $g_0, g_1$ and $g_2$ are code generator polynomials used in the OE. The QCA based modulo-2 adder represented in Figure 7. The inputs are represented by $A$ and $B$ whereas the $Z$ is the output of the modulo-2 adders. The parity bit generator constructed to generate a 3-bit parity code using the 4-bit output of the SPC. The output $R$ of the parity bit generator as given in equation (4), in the polynomial form. In parity bit generator A3, A2, A1 and A0 are the inputs and R2, R1 and R0 are required parity bits. The calculation of the Parity bits can be explained using equation (2) where $P(x)$ represents the parity bits as a function of $x$ and $A(x)$ represents the parallel form of the message bits.

$$P(x) = (A(x) \times x^3) mod[x^3 + x + 1] = R0 + R1x + R2x^2$$  

$$\text{Figure 4. Block diagram of outer encoder}$$

$$\text{Figure 5. QCA based design of 4-bit SPC}$$
The final stage of the OE is the 3-bit PSC that serializes the previously generated parity bits R0, R1 and R2 into serial bits of the form R. These bits R when combined with the message bits (D) or A(x) results in the systematic code-word (C). The 3-bit PSC depicted in Figure 8. The parity bits R2, R1 and R0 act as inputs to the PSC and R is the output of the PSC that is in serial form, connected to the message bits, which results in the systematic code word. Figure 9 represents the QCA based design of an OE where D is the initial input to the OE and A3, A2, A1 and A0 as the outputs of SPC [15]. The R2, R1 and R0 are the outputs of the parity bit generator. R is the output of the PSC and C is the final output of the OE i.e., the systematic codeword.
4.1.1 Pseudo-Random Inter-leaver

The PRI is an intermediate stage of the SCCC encoder, which disorganizes the systematic codeword into a newly generated set of codeword known as permuted code word (I). The PRI comprises two stages namely 7-bit SPC and 7-bit PSC. The systematic code word (C) first parallelizes it to seven different bits represented as C0 to C6 and then interchanges their position in reverse and then the 7-bit PSC serializes those swapped bits to form the permuted codeword (I). Since the permuted codeword found to be of random order in comparison to the systematic codeword by the inter-leaver, this intermediate stage is PRI. The flow of 7-bit PRI illustrated in Figure 10. The bit position is interchanged and the resulting permuted code word (I) is the swapped form of the systematic code word (C). The C from the OE is the input to the SPC and parallelized and then the parallelized set of 7-bits fed into the 7-bit PSC that serializes the 7-bits to form the permuted code word (I) [15]. In addition to the 7-bit PRI [15], another PRI implemented with the help of the previously discussed designs of proposed SPC and PSC. The QCA based realization of the 7-bit PRI shown in Figure 11.

![Figure 9. QCA based circuit of outer encoder](image)

**Figure 9.** QCA based circuit of outer encoder

![Figure 10. The flow of bits in the PRI](image)

**Figure 10.** The flow of bits in the PRI
Figure 1. QCA based design of modified 7-bit PRI

4.1.2 Inner Encoder

The final stage of the SCCC encoder is the IE [13]. The IE used is (2, 1, 2) classical convolutional (CC) encoder which is a (n, k, N) where n denotes the output count, k the input count and N the constraint length of the encoder and the code rate is denoted by k/n [7]. The QCA based realization of an IE illustrated in Figure 12 where the permuted code word is the input to the inter-leaver and Y1 and Y0 are the expected outputs [15]. Figure 9 in [15] represents the proposed-I QCA based SCCC encoder. The SCCC encoder (proposed-II) with different approach is illustrated in Figure 13. The design of PSC and SPC modified based on internal clock generation whereas the designs of the proposed parity bit generator and the IE retained.

Figure 12. QCA based design of inner encoder

5. Results and Discussion

The implementation of SCCC encoder discussed in the previous section. This section deals with the performance analysis, comparison and simulation results of the SCCC encoder. The performance analysis of the SCCC encoder such as 4-bit SPC, parity bit generator, OE, 7-bit SPC, inter-leaver and the IE dealt with in this section. The D denotes the message bits, C the systematic codeword, I, the permuted codeword and Y0 and Y1, the required outputs of the SCCC encoder.

Cell Count: Figure 14 represents the comparison of the cell count of the parity generator, OE, inter-leaver etc. of the proposed design, [15] and the architecture in [7]. The proposed 4-bit SPC has a cell count of 48 whereas the one in [7] is 184. The proposed OE has 519 cells while the ones in [7] have 732 cells. The 7-bit SPC has 98 cells in comparison to that of 1086 cells in [7]. The cell count of Inter-leaver of the proposed is 199 whereas the existing inter-leaver is 1432 cells. Results infer that there is reduction in the cell count in the proposed designs.
Area: The area comparison of the components of the proposed SCCC encoder with the existing architectures carried out in this section. Figure 15 represents the comparison of the area occupancy of the components of the SCCC encoder such as parity generator, OE, inter-leaver etc. of proposed design, [15] and in [7]. The proposed 4-bit SPC has an area of 0.06 µm$^2$, the one in [7] occupies an area of 0.3192 µm$^2$. The proposed OE occupies an area of 1.51 µm$^2$ and OE in [7] occupied 1.89 µm$^2$. A similar analysis was performed for parity generator, inter-leaver, 7-bit SPC and 7-PSC occupies results inferred that proposed architectures occupied less area.
Energy Dissipation: The energy dissipation of components of the SCCC encoders of proposed, [15] and existing [7] compared and results discussed in this section. Figure 16 represents the comparison of energy dissipation of some of the components of the SCCC encoder. The energy dissipation is less compared to reference architecture. The SCCC encoders (proposed-I [15] and proposed-II) and architecture in [7] compared in terms of cell count, energy dissipation and area occupancy and listed in Table 2. The SCCC encoder in [7] has 2300 cells occupying an area of 4.65 µm² with 0.38 eV of energy dissipation. The proposed SCCC encoder contains 822 cells occupying an area of 2.16 µm² while the energy dissipation is 0.192 eV, which infer that there is reduction of cell count by 1478 compared to [7]. The total area reduced by 46% and energy dissipation minimized by 50% when compared to the SCCC encoder in [7]. The overall reduction in the cell count, area and energy dissipation achieved because of reanalysis of the existing logic and increasing the reusability of the cells available. Hence, the proposed encoder is more compact, dissipates less energy and the cost of implementation is minimized.

![Area occupancy (µm²) of components of SCCC encoder](image1)

**Figure 15.** Area analysis of components of SCCC encoders

![Energy dissipation (eV) of components of SCCC encoder](image2)

**Figure 16.** Energy dissipation of components of SCCC encoders

| SCCC Encoder | Cell Count | Area Occupied (µm²) | Energy Dissipation (eV) | Cost |
|--------------|------------|---------------------|-------------------------|------|
| [7]          | 2300       | 4.56                | 0.38                    | 18.4 |
| Proposed-I   | 1325       | 2.78                | 0.245                   | 11.2 |
| Proposed-II  | 822        | 2.16                | 0.192                   | 8.64 |
6. Conclusions
This study proposes a design for the serial concatenated convolution codes encoders and its realization in the QCA. The SCCC is the integration of the outer encoder, inner encoder and pseudo-random inter-leaver. The OE is a (7, 4, 1) BCH encoder where (7, 4, 1) is in the form of (n, k, N). The OE is the combination of 4-bit SPC, parity bit generator and 3-bit PSC. Two sets of SCCC encoders proposed. The difference between the proposed - I and proposed - II is that the former design contains a separate cell assigned for clock signal while the latter design is implemented on the basic idea of cell patterns themselves acting as clock signals. Thus through the results it has been proved that proposed SCCC encoders are efficient in terms of area, cell count and energy dissipation. The future scope is to design an SCCC encoder with a higher range of BCH encoder and Inter-leaver. The BCH encoders can also be implemented using other higher forms such as (15, 11, 1), (31, 16, 3) etc. These BCH encoders supported by CC encoders of higher bits implemented in inner encoder.

References
[1] M. Goswami, B. Kumar, H. Tibrewal, and S. Mazumdar, “Efficient Realization of Digital Logic Circuit Using QCA Multiplexer,” Proceeding of 2nd International Conference on Business and Information Management, pp. 1-7, 2014.
[2] Premananda B.S., Bhargav U.K., and K.S. Vineeth, “Design and Analysis of Compact QCA Based 4-Bit Serial-Parallel Multiplier,” Proceedings of IEEE International Conference on Electrical, Electronics, Communication, Computer Technologies and Optimization Techniques, pp. 1014-1018, 2018.
[3] D.S. Silva, L.H.B. Sardinha, M. A.M. Vieira, L.F.M. Vieira, and O.P.V. Neto, “Robust Serial Nano Communication with QCA,” IEEE Transactions on Nanotechnology, vol. 14, no. 3, pp. 464-472, 2015.
[4] Harshitha S., Dhanush T.N., and Premananda B.S., “A Novel QCA based Compact Scan Flip-flop for Digital Design Testing,” International Journal of Engineering and Advanced Technology, vol. 9, pp. 6681-6686, 2019.
[5] M.G. Ganavi and B.S. Premananda, “Design of Low-Power Square Root Carry Select Adder and Wallace Tree Multiplier using Adiabatic Logic” Lecture Notes in Electrical Engineering, LNEE vol. 545, Springer, pp. 767-781, 2019.
[6] M. Zhang, L. Cai, X. Yang, H. Cui, and C. Feng, “Design and Simulation of Turbo Encoder in Quantum-dot Cellular Automata”, IEEE Transactions on Nanotechnology, vol. 14(5), pp. 820-828, 2015.
[7] Y. Zhang, G. Xie, and J. Han, “Serial Concatenated Convolutional Code Encoder in Quantum-dot Cellular Automata”, Journal of Nano Communication Networks, vol. 22, pp. 365-373, 2019.
[8] Lee Ai Lim, A. Ghazali, S.C.T. Yan, and Chau Chien Fat, “Sequential Circuit Design using Quantum-dot Cellular Automata,” Proceedings of International Conference on Circuits and System, pp. 162-167, 2012.
[9] N. Bahar, S. Waheed, N. Hossain, and M. Asaduzzaman, “A Novel 3-Input xor Function Implementation in QCA with Energy Dissipation Analysis,” Alexandria Engineering Journal, vol. 57(2), pp. 729-738, 2018.
[10] I. Gassoumi, L. Touil, B. Ounti, and A. Mitbaa, “An Ultra-Low Power Parity Generator Circuit Based on QCA Technology,” Journal of Electrical and Computer Engineering, vol. 20, pp. 1-8, 2019.
[11] R. Karrhik, Sankit R. Kassa, Susmitha V.G., B. Naresh, “Energy Efficient Relicue (2N X 1) Multiplier Design Using QCA Approach,” Institute of Integrative Omics and Applied Biotechnology, vol. 9, pp. 113-117, 2018.
[12] Dhanush T.N., Harshitha S., and Premananda B.S., “Design of QCA based Three-Stage Pseudo Random Number Generator”, Proceedings of International Conference on Advances in Computing, Communication and Materials, ICACCM-2020, pp. 233-238, 2020.
[13] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, “Serial Concatenation of Interleaved Codes: Performance Analysis, Design, and Iterative Decoding,” IEEE Transactions on Information Theory, vol. 44(3), pp. 909–926, 1998.
[14] R. Jayalakshmi and M. Senthil Kumaran, “Reliability Analysis and Power Dissipation of QCA Circuits for next Generation Computing,” International Journal of Nano-electronics and Materials, vol. 12, pp. 485-494, 2019.
[15] Dhanush T.N., Vaishanavi S. Parashar, and Premananda B.S. “Analysis of QCA-Based Serial Concatenated Convolution Coding Encoder for Error Correction”, Recent Trends in Electronics and Communication, VCAS 2020, Lecture Notes in Electrical Engineering, vol. 777, Springer, 2021 (in press).