International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878, Volume-8 Issue-3, September 2019

Hamming Code Converter using Reversible Toffoli Netlist

Diganta Sengupta, Mahamuda Sultana, Atal Chaudhuri

Abstract*: Technological advancements in the last two decades have witnessed a plethora of emerging technologies gaining global research attention. One of them is reversible logic due to its capability to arrest tremendous amount of heat generated in modern digital conversions at the architecture level. With CMOS reaching near capping of threshold parameters, researchers are imperatively exploring the reversible domain for possible alternatives. Telecommunication is one area which expects appreciable share of digital resources. This communication proposes the reversible version of Hamming Code converter, decoder, and Corrector unit. Hamming codes form one of the perfect codes attaining the Hamming Bound. The designs have been implemented using Toffoli Netlist. A state-of-the-art four variable reversible gate in literature has been used to design the high level schematic for part of the proposals. The authors provide the results in terms of primitive Quantum Metrics.

***Keywords*: Reversible Hamming Code, Toffoli Netlist, Hamming Bound, Reversible Logic, Perfect Codes, Quantum Metrics, Four variable reversible gate.**

I. INTRODUCTION

Technological growth has witnessed decline of Moore's law in recent years with computer scientists proposing non-adherence by the next couple of years. The major reason behind this development can be attributed to the dramatic demand for higher clocking speeds in digital architectures. In addition to enhanced clocking, every bit of information change contributes $KTLn2$ Joules of energy dissipation [1]. Bennett [2] has argued that this heat dissipation can be arrested if the computation is reversible. Thus research in Reversible Logic gained global importance in the previous two decades with major thrust observed in the last decade. A number of reversible architectures have been proposed till date for combinational as well as sequential counterparts of existing digital designs. Reversible computing has presented two domains of implementations; one using reversible gates and the other using Quantum Dot Cellular Automata (QCA) cells [3] [4]. In this communication, the authors concentrate on the reversible gate architectures. Four reversible gates form the primitive domain; Toffoli Gate [5], Feynman Gate [6], Fredkin Gate [7] and Peres Gate [8]. The work in this paper concentrates on Toffoli Netlist (cascade of Toffoli gates) designs. A state-of-the-art four variable reversible gate [9] has been used to design the high level schematic for the error signal generator, as discussed later. Multiple four variable reversible gates have been proposed till date [10] [11] [12] with varied realizations. Apart from reversible gate proposals, circuit optimization [13] and online testability [14] of reversible circuits have been two major research thrust areas. This communication proposes three designs for Hamming Code generation, decoding and correction using reversible Toffoli gates. Hamming Code is one of the forward error-correction codes widely accepted for digital telecommunication. Conversion of existing digital architecture into the reversible domain has been heavily researched and this motivated the authors to design one of the most widely used perfect codes for telecommunication in the reversible domain. Hamming code is a perfect code attaining the Hamming bound. Three designs have been proposed. The first design is the Hamming Code generator. The second design is the Hamming code checker with the third one correcting the error. The proposed designs can check and correct single bit errors. The correction unit uses 4:16 reversible decoder.

The rest of the paper is organized as follows. The next section details the related work. Section 3 outlines the proposed designs with the subsequent section providing the analysis. Section 5 concludes the work.

II. RELATED WORK

Although there have been numerous proposals for implementing reversible circuits, only two literary proposals have been for Hamming Code [15] [16]. Both the implementations have been made using FPGAs realized using either Tanner EDA or Xilinx. As far as our knowledge goes, this is the very first implementation for Hamming Code Converters using primitive reversible gates; i.e. Toffoli gates. In [15], the authors proposed architectures for 16nm, 22nm, 32nm, and 45nm technologies. Authors in [16] provide Xilinx simulations for their design and provide theoretical results for the quantum cost, garbage output and delay calculations. Reversible 4: 16 decoder designs have been proposed in [17] [18] [19]. In this paper, the authors use the implementation given in Fig. 9 of [19].

Manuscript published on 30 September 2019
* Correspondence Author
Diganta Sengupta*, Dept. of Computer Science and Engineering, Techno International Batanagar, Kolkata, West Bengal 700141, India, sg.diganta@gmail.com
Mahamuda Sultana, Dept. of Information Technology, Techno International New Town, Kolkata, West Bengal 700150, India, sg.mahamuda@gmail.com
Atal Chaudhuri, Vice-Chancellor, Veer Surendra Sai University of Technology, Sambalpur, Odisha 768018, India, atalc23@gmail.com

The choice reflects the uniformity of Toffoli design. Authors of [19] have proposed design for 4:16 decoder using Toffoli, Feynman and Peres Gate. It can be observed in [12] that the three mentioned gates are enhanced versions of Toffoli gates. Since, the designs have been realized used Toffoli Gates only, hence the authors have selected Fig. 9 of [19].

The authors in [19] have proposed 2:4 decoder, 3:8 decoder using 2:4 decoder and 4:16 decoder encasing 3:8 decoder. The complete 4:16 decoder consumes twelve Feynman Gates, four Toffoli (CNOT, NOT) Gates and one Peres Gate. The quantum cost for Feynman, CNOT, and Peres Gate is 1, 1, and 4 respectively. Hence, the total Quantum Cost for their design account for $12 + 4 + 4 = 20$. This value has been used for the Quantum Cost calculation for the design proposed in this paper. The design proposal in [17] is similar to the design in Fig. 9 of [19].

A state-of-the-art four variable reversible gate has been used to design the proposals in this communication. Figure 1 presents the high level diagrams for Binary-to-Gray converter (BG4) gate and Gray-to-Binary converter (GB4) Gates [9]. Figure 2 provides the Toffoli Gate implementations for the two gates.

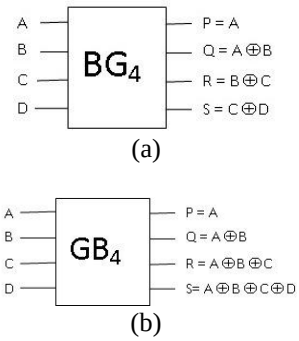

Fig. 1.High level diagram for (a) BG4 (Four variable Binary-to-Gray converter), and (b) GB4 (Four variable Gray-to-Binary converter)

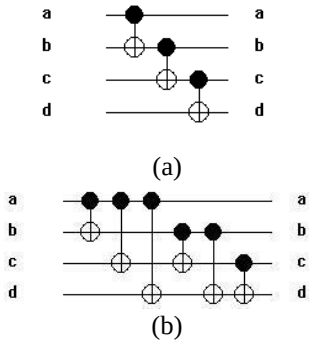

Fig. 2.Toffoli implementations for (a) BG4 Gate, and (b) GB4 Gate

The next section presents the Hamming Code Converter design.

III. PROPOSED HAMMING CODE GENERATOR, CHECKER, AND CORRECTOR

Hamming codes are part of perfect codes having full attainment of Hamming Bound. This section provides the equations for hamming code generation and correction. Let, the message to be encrypted is given by Equation 1.

$$Message\ M = m_7m_6m_5m_4m_3m_2m_1m_0 \quad (1)$$

$$Coded\ Message\ C = m_7m_6m_5m_4h_8m_3m_2m_1h_4m_0h_2h_1 \quad (2)$$

Where $h_1, h_2, h_4, and\ h_8$ are the hamming codes for *M*. Let, D denote the coded message *C*. Therefore,

$$D = \sum_{i=0}^{11} d_{i+1}2^i \quad (3)$$

$$\Rightarrow D = d_{12}d_{11}d_{10}d_9d_8d_7d_6d_5d_4d_3d_2d_1 \quad (3)$$

Where

$d_1 = h_1\ ;\ d_2 = h_2\ ;\ d_3 = m_0\ ;\ d_4 = h_4\ ;\ d_5 = m_1$

$d_6 = m_2\ ;\ d_7 = m_3\ ;\ d_8 = h_8\ ;\ d_9 = m_4\ ;\ d_{10} = m_5$

$d_{11} = m_6\ ;\ d_{12} = m_7$

The Hamming codes are generated according to the following equations.

$$h_1 = d_1 \oplus d_3 \oplus d_5 \oplus d_7 \oplus d_9 \oplus d_{11} \quad (4)$$

$$h_2 = d_2 \oplus d_3 \oplus d_6 \oplus d_7 \oplus d_{10} \oplus d_{11} \quad (5)$$

$$h_4 = d_4 \oplus d_5 \oplus d_6 \oplus d_7 \oplus d_{12} \quad (6)$$

$$h_8 = d_8 \oplus d_9 \oplus d_{10} \oplus d_{11} \oplus d_{12} \quad (7)$$

Where, $\oplus$ denotes the Binary XOR operation.

The Hamming Code checker equations are generated according to Equation 8 through Equation 11.

$$c_1 = d_1 \oplus d_3 \oplus d_5 \oplus d_7 \oplus d_9 \oplus d_{11} \oplus h_1 \quad (8)$$

$$c_2 = d_2 \oplus d_3 \oplus d_6 \oplus d_7 \oplus d_{10} \oplus d_{11} \oplus h_2 \quad (9)$$

$$c_4 = d_4 \oplus d_5 \oplus d_6 \oplus d_7 \oplus d_{12} \oplus h_4 \quad (10)$$

$$c_8 = d_8 \oplus d_9 \oplus d_{10} \oplus d_{11} \oplus d_{12} \oplus h_8 \quad (11)$$

Let

$H = h_8h_4h_2h_1\ and\ C' = c_8c_4c_2c_1$

Therefore, the Error message $'E'$ is obtained by bit-wise XORing of $H\ and\ C'$.

Therefore,

$$E = \sum_{i=0}^{3} e_i 2^i\ ;where\ e_i = h_j \oplus c_j \forall j \in \{8,4,2,1\} \quad (12)$$

The Error message *E* provides the bit position of error in the transmitted message *D*. Illustration 1 presents the process in details.

Illustration 1:

Let *M = 10011100*

Therefore,

$C = 1001h_8110h_40h_2h_1$ and

$D = 1001d_8110d_40d_2d_1$

Where

$h_1 = 0$, $h_2 = 0$, $h_4 = 0$, and $h_8 = 0$ respectively.

Therefore the transmitted message *D* is

D = 100101100000

Now, if the message bit

m_1 gets corrupted at bit position d_5 of *D*, then *D = 100101110000*. The bold and underlined bit represents the corruption.

Therefore, at the receiver end,

$c_1 = 1, c_2 = 0, c_4 = 1, and\ c_8 = 0$ respectively.

Therefore error E equals

E = 0101, where

$e_4 = h_8 \oplus c_8 = 0$

$e_3 = h_4 \oplus c_4 = 1$

$e_2 = h_2 \oplus c_2 = 0$

$e_1 = h_1 \oplus c_1 = 1$

Hence, bit position for corruption is given by *E*, i.e. 5th position (d_5) of the message has been corrupted in transmitted message *D*.

Figure 3 presents the Toffoli Netlist for the Hamming Code Generator. The Netlist generates and output represented by Equation 3, Equation 2 given as input.

Close observation of Figure 4 reveals that the architecture is same as that of Figure 3. Figure 3 generates the hamming bits $h_8h_4h_2h_1$ and places them at the respective positions in the transmitted data *D*.

For generation, initially these positions are treated as logic 0. The complete message *D* is then transmitted. At the receiver end, this received data *D* is fed to the same architecture with valid Hamming bits and thus the check bits ($c_8c_4c_2c_1$) are generated. Figure 2 reflects generation of the check bits $c_8c_4c_2c_1$.

For generation of the position of error, if any, the hamming bits and the check bits are bit-wise XORed. Therefore Equation 12 generates the error signal '*E*'. Figure 5 represents the Toffoli Netlist for generation of *E* signal. It is interesting to note that although four error bits are generated but the value of the signal lies between 0 and 12. The rest values are redundant.

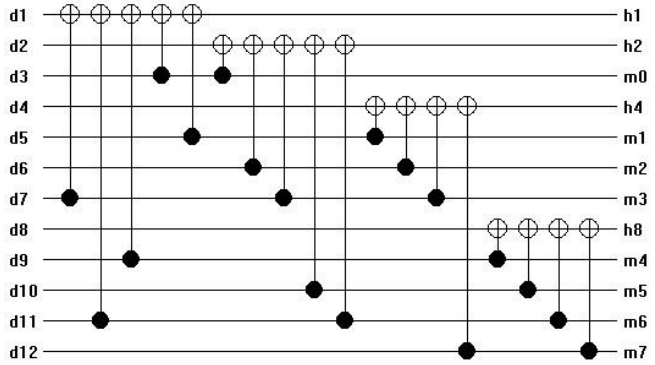

Fig. 3.Hamming Code Encoder using Toffoli Netlist

The Toffoli Netlist for the decoder at the receiving end is presented in Figure 2.

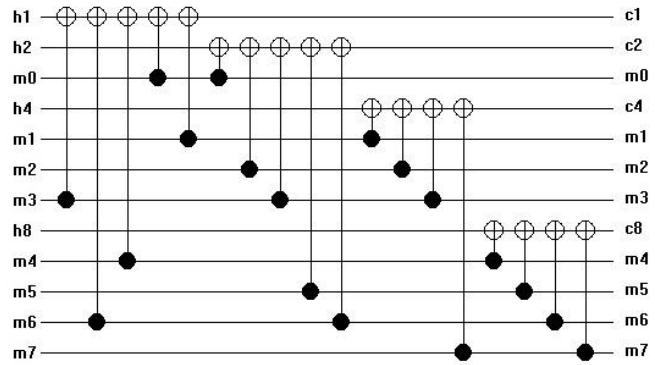

Fig. 4.Hamming Code Decoder using Toffoli Netlist

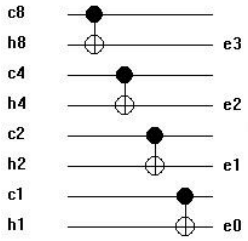

Fig. 5.Error signal (E) generator architecture

Figure 5 reflects that the two BG4 gates (Figure 1a) can be incorporated to generate the Error signal. Figure 6 reflects the situation.

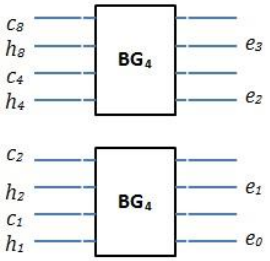

Fig. 6.Error signal generation using BG4 Gate

The Error signal thus generated is fed into a 4:16 decoder to trigger the exact output line according to the generated value of '*E*'. At the output of the 4:16 decoder, an array of Feynman gates is used to correct the signal. Feynman gate is a special type of Toffoli gate. Hence, the complete proposal falls within the purview of Toffoli gates as claimed by the author of this communication. Moreover, the 4:16 decoder has been selected from Fig. 9 of [19] as discussed earlier.

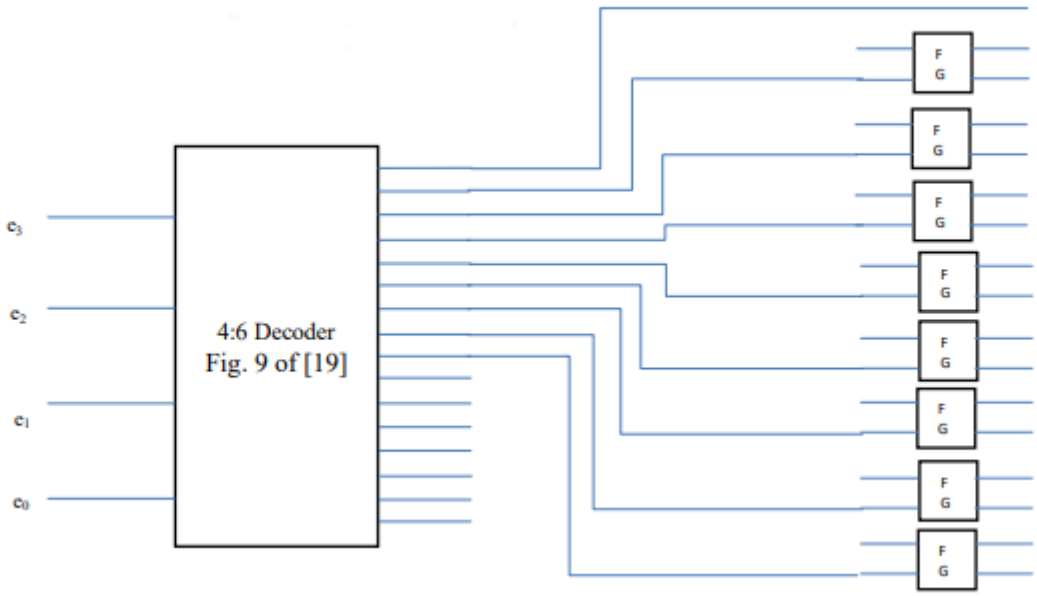

Fig. 7. Hamming Code Corrector circuit

IV. COMPARATIVE ANALYSIS

This section provides the analysis of the proposed designs with respect to the Quantum Cost, No. of Gates, and the Number of Two-Qubit Gates. Table I provides the results.

Table- I: Name of the Table that justify the values

Architecture	QC	G#	TQG
Hamming Code Generator	18	18	18
Hamming Code Checker	18	18	18
Error Generator	4	4	4
4:16 Decoder	20	18	18
Array of Feynman Gates	8	8	8
Total	68	66	66

[a] QC: Quantum Cost

[b] G#: Gate Count

[c] TQG: Two-Qubit Gate Count

Thus, it can the stated that the Hamming Code Generator-Checker-Corrector circuit implemented using Toffoli Netlist amount to a Quantum Cost of 68.

The authors also tried to optimize the design using the algorithm provided in [13], but it is interesting to note that the proposed designs are best optimized. The designs are itself in terms of the primitive gates, hence, it is also not possible to decompose the designs any further. The designs are therefore the most primitive architecture themselves.

Since proposals in [15] and [16] are in terms of FPGAs and have been implemented in Tanner EDA and Xilinx, thus comparative analysis with respect to primitive quantum metric could not be done. To the best of knowledge and belief, this communication is the first ever proposal for Toffoli Netlist implementation of Hamming Code Converter. Table II states the facts.

Table- II: Comparative analysis of literary proposals

Proposals Platforms	Tanner EDA	Xilinx	Toffoli Netlist
[15]	√	x	x
[16]	x	√	x
Proposed	x	x	√

V. CONCLUSION

This communication proposes Toffoli Netlist implementation for Hamming Code Generator, Checker, and Corrector unit. This is the very first implementation of such reversible architectures in terms of primitive quantum gates. The proposed designs can form an integral part of future quantum architectures. Further, the designs can be implemented using QCA cells for a different paradigm of reversible logic concept. Other digital architectures dependent on Hamming Code can be explored for reversible counterparts.

REFERENCES

1. R Landauer, "Irreversibility and Heat Generation in the Computing Process," IBM Journal of Research and Development, vol. 5, no. 3, pp. 183 - 191, July 1961.
2. C H Bennett, "Logical Reversibility of Computation," IBM Journal of Research and Development, vol. 17, no. 6, pp. 525-532, Nov 1973.
3. J C Das and D De, "Circuit switching with Quantum-Dot Cellular Automata," Nano communication networks, vol. 14, pp. 16-28, December 2017.
4. J C Das and D De, "Optimized design of reversible gates in quantum dot-cellular automata: a review," Reviews in Theoretical Science, vol. 4, no. 3, pp. 279-286, September 2016.
5. T Toffoli, "Reversible computing," Lecture Notes in Computer Science (ICALP 1980), vol. 85, pp. 632-644, 1980.
6. Richard P Feynman, "Simulating Physics with Computers," International Journal of Theoretical Physics, vol. 21, no. 6, pp. 467-488, July 1982.
7. E Fredkin and T Toffoli, "Conservative Logic," International Journal of Theoretical Physics, vol. 21, no. 3-4, pp. 219-253, October 2006.
8. Asher Peres, "Reversible Logic and Quantum Computers," Physical Review A, vol. 32, no. 6, p. 3266, Dec 1985.

9. S Banerjee, A K Pal, M Sultana, D Sengupta, and A Das, "Reversible Code Converters based on Application Specific Four Variable Reversible Gates," in International Conference on Emerging Technologies in Data Mining and Information Security (IEMIS 2018), Springer, Kolkata, 2018, pp. 465-477.
10. D Sengupta, M Sultana, and A Chaudhuri, "Realization of a Novel Reversible SCG Gate and its Application for Designing Parallel Adder/Subtractor and Match Logic," International Journal of Computer Applications, vol. 31, no. 9, pp. 30-35, October 2011.
11. M Sultana, A Chaudhuri, D Sengupta, and A Chaudhuri, "Logic Design and Quantum Mapping of a Novel Four Variable Reversible s2c2 Gate," in Annual Convention of Computer Society of India (CSI 2017), Springer, Kolkata, 2018, pp. 416-427.
12. M Sultana, A Chaudhuri, D Sengupta, and A Chaudhuri, "Toffoli Netlist and QCA Implementations for Existing Four Variable Reversible Gates – A Comparative Analysis," Microsystem Technologies, Springer, vol. 25, pp. 1987-2009, July 2018.
13. Kamalika Datta, Indranil Sengupta, and Hafizur Rahaman, "A Post-Synthesis Optimization Technique for Reversible Circuits Exploiting Negative Control Lines," IEEE Transactions On Computers, vol. 64, no. 4, pp. 1208 - 1214, April 2015.
14. Sk Noor Mahammad and Kamakoti Veezhinathan, "Constructing Online Testable Circuits Using Reversible Logic," IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT, vol. 59, no. 1, pp. 101-109, January 2010.
15. D R Choudhury and K Podder, "Design of Hamming Code Encoding and Decoding Circuit Using Transmission Gate Logic," International Research Journal of Engineering and Technology (IRJET), vol. 2, no. 7, pp. 1165-1169, October 2015.
16. C H Monika and D K Kavitha, "Design and analysis of Hamming Code Encoding, Decoding and Correcting Circuits usingReversible Logic," International Journal of Research, vol. 7, no. 12, pp. 36-41, December 2018.
17. S Chebiyyam and K B S Kumar, "Design of 4:16 decoder using reversible logic gates," Intl. Journal of Engineering Research and Applications, vol. 6, no. 4, pp. 43-45, April 2016.
18. A Kumar and S Gugnani, "Synthesis of 4 to 16 Reversible Decoder," Institute for Research and Development India (IRD India), vol. 1, no. 5, pp. 19-22, 2013.
19. G C Naguboina and K Anusudha, "Design And Synthesis Of Combinational Circuits Using Reversible Decoder In Xilinx," in IEEE International Conference on Computer, Communication, and Signal Processing (ICCCSP-2017), 2017.

AUTHORS PROFILE

Dr. Diganta Sengupta, B.Tech (2004) Electronics and Instrumentation Engineering from University of Kalyani, WB, IN. M.Tech (2010) CSE from Jadavpur University, WB, IN. Ph.D. (2016) from Jadavpur University, WB, IN. He is an IEEE member since 2016 and an ACM member in 2017. He is a life member of Computer Society of India (LM'16) and The Institution of Engineers (India) (M'16). He is also a member of IEEE Electron Device Society. Presently he is serving as the State Student Coordinator for West Bengal, India for Computer Society of India.
Dr. Diganta Sengupta is presently working in the capacity of Associate Professor in the Dept. of CSE, Techno International Batanagar, Kolkata, India. Formerly he was associated with the School of Computer Engineering (SCOPE), VIT University, Vellore, India.
His research interests include Decimal Numeric processors, Vedic Mathematics, Reversible Logic, Quantum Dot Cellular Automata, Taxonomy Generation Process and Hardware Accelerators for both classical as well as reversible computational engines. He has served as a reviewer for IEEE Access, JIKM (World Scientific), IJSAEM (Springer), IJBDCN (IGI-Global), ISSE (Springer) to name a few.

Mahamuda Sultana, B.Tech (2004) Computer Science and Engineering, University of Kalyani, WB, IN. M.Tech (2010), Computer Science and Engineering, Jadavpur University, WB, IN 700032. She is currently pursuing PhD in Computer Science and Engineering at Jadavpur University. Presently, she is associated in the capacity of Assistant Professor in the Dept. of Computer Science and Engineering, Techno International New Town, Kolkata, WB, IN 700156. She has a total of 12 years of teaching experience. Her research interests include Reversible Logic, Computer Architecture, Quantum Dot Cellular Automata and Vedic Mathematics.

Dr. Atal Chaudhuri received his Master of Electronics and Telecommunication Degree with Computer Science specialization in the year 1982 and Doctorate of Philosophy in Engineering from Jadavpur University, IN in 1989. Presently he is the Vice-Chancellor of Veer Surendra Si University of Technology (Burla University), Odisha, India. Previously he served as a Professor in the Dept. of Computer Science and Engineering at Jadavpur University. He has worked in the capacity of R&D Engineer and Project Engineer in various research projects in India and abroad. Formerly Dr. Atal Chaudhuri was a Senior Professor in the Department of Computer Science & Engineering of Jadavpur University, Kolkata, India. He is a Fellow at The Institution of Engineers (India) and a Life Member of Computer Society of India.