



A Novel n -Decimal Reversible Radix Binary-Coded Decimal Multiplier Using Radix Encoding Scheme

K. Saranya¹ · K. N. Vijeyakumar²

Received: 26 February 2019 / Revised: 10 September 2020 / Accepted: 13 September 2020 /

Published online: 25 September 2020

© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

Reversible logic has been emerging as a replacement for conventional digital computers and has been found to be a promising technology for the future that can improve the quality of circuits in terms of power, speed, heat dissipation, life span, and input traceability. The new generation of quantum processors demands the efficient implementation of decimal data path elements in many commercial, financial, and Internet-based applications. Reversible binary-coded decimal multipliers are among the important circuits in quantum computers. In this approach, we propose a low-power reversible radix binary-coded decimal multiplier. The proposed methodology uses a novel 4221 reversible recorder gate to select precomputed constant multiples and binary to excess-six conversion reversible decimal adder for partial product compression. The proposed multiplier is designed with 180-nm application-specific integrated circuit technology using the Cadence EDA tool and is compared with state-of-the-art BCD algorithms designed using reversible gates. Experimental evaluations revealed that the proposed design demonstrates power and power-delay product reductions of 28% and 37%, respectively, compared to those of the best BCD algorithms implemented in reversible logic.

Keywords Radix recoder · Reversible gates · Quantum application · Decimal multiplier

✉ K. Saranya
miles2gosaran@gmail.com

K. N. Vijeyakumar
vijey.tn@gmail.com

¹ Department of Electrical and Electronics Engineering, Dr. Mahalingam College of Engineering and Technology, Pollachi, Coimbatore, India

² Department of Electronics and Communication Engineering, Dr. Mahalingam College of Engineering and Technology, Pollachi, Coimbatore, India

1 Introduction

In conventional computers, information loss leads to more power and heat dissipation problems. In the 1960s, Landauer [12] stated that conventional computation, irrespective of the realization methodology, would result in high energy and information loss. He also proved that each bit of information lost is accompanied by $KT\ln 2$ joules of energy loss. Bennett [2] showed that a circuit must be constructed using a reversible logic gate to avoid $KT\ln 2$ joules of energy dissipation. In recent years, reversible logic has been used in various areas of quantum computation [14, 16], DNA computation, etc. The multiplier is a very important component in high-performance computing systems such as digital filters, microprocessors, and digital signal processors. A system's performance is mostly dependent on the multiplier because the multiplier is generally the time-consuming component among the various data path elements. A number of approaches to designing full-width [11], fixed-width [21–23], and approximate multipliers [15] have been proposed. However, binary multipliers are not suitable for monetary transactions, as the small round-off errors in binary to BCD conversion cause large errors in the final results. The best choice for systems involving monetary transactions is a BCD multiplier, in which the binary inputs are recoded to BCD and arithmetic operation is performed on the BCD operands [9, 10, 20, 24]. Additionally, high integration in deep submicron technology increases the power density and poses heat dissipation problems.

Conventional multipliers use Baugh Wooley, Pezaris, Vedic, Booth, etc., algorithms to improve the speed at the expense of the hardware cost. Though these concepts are efficient in normal binary multiplication, the radix method is found to be more efficient for BCD multiplication [5, 8, 21, 23].

A number of algorithms and approaches are proposed for the design of arithmetic units for decimal operations [4]. Erle et al. [5] proposed a fixed-point BCD multiplier that utilizes a simple recoder unit to produce intermediate PPs in signed form. As PPs are generated sequentially, this multiplier achieves good improvement in speed. Jaberipur et al. [8] proposed a delay and area-optimized BCD digit multiplier to pre-compute all the possible 10 multiples (0-9X) of the multiplicand at the outset of the multiplication process. A 10:1 selector is used to select the required product. Lang et al. [13] proposed a decimal multiply unit constructed by parallel combinational logic to reach the desired speed. Vazquez et al. [22] proposed BCD architectures employing 4221 and 5211 binary codes. A shift operation is performed on the basic multiplicand multiple to generate other multiples. Vazquez et al. [21] improved the algorithm in [22] using excess-3 code. The self-complementing property of excess-3 code is used to speed up the computation, but the conversion from binary to excess-3 and back occupies more hardware area. Additionally, the design of BCD multipliers in conventional logic consumes considerable power and poses heat dissipation problems [21–23]. To date, many reversible binary multipliers have been proposed and implemented using various techniques [3, 18]. In this approach, a novel design for reversible BCD multiplication using the radix recoder algorithm is proposed and implemented using basic reversible gates and is compared with BCD designs in [13, 21].

The rest of the paper is organized as follows. Section 2 describes the various reversible gates used in the proposed decimal multiplier design. Section 3 provides

insight into the design of the proposed BCD multiplier. Section 4 compares the performance of the proposed decimal multiplier with that of state-of-the-art BCD multipliers designed in reversible logic. Finally, Sect. 5 provides a brief conclusion of the work done.

2 Reversible Logic Gates

Reversible gates are elements that consist of an equal number of inputs and outputs. The NOT gate, controlled NOT (CNOT) gate, controlled-V gate, and controlled-V + gate [19] are basic reversible gates in quantum processing similar to the AND, OR, and NOT gates in conventional computers. The total count of basic quantum gates contributes to the overall quantum cost (QC) of the newly developed reversible gates. The unique direct mapping between the inputs and outputs supports the traceability of the output from the input and vice versa in reversible gates. Constant inputs (CIs) and garbage outputs (GOs) are included in the $m \times k$ gate to make it reversible. The following symbols are used to represent logic functions in the following subsections: \wedge - EX-OR, $|$ - AND, and $+$ - OR.

2.1 Feynman Gate (FG)

The fan-out is always one, as reversible gates maintain traceability of the input from the output. If $A = 0$ in Eq. (1), the FG functions as a copying circuit, and when $A = 1$, the FG performs the inversion operation. Fan-out of more than one circuit requirement is accomplished mainly by a Feynman (2*2) gate with a QC of 1 [6].

$$\text{Input}\{P; Q; S\} = \text{Output}\{A; A \wedge B\} \quad (1)$$

2.2 Nines Complement Gate (NCG)

A NCG is a 5×5 reversible gate used to generate the decimal complement. The Boolean logic in Eq. (2) realizes a gate with a QC of 9.

$$\text{Input}\{P; Q; R; S; T\} = \text{Output}\{P; Q; R; (P + Q)R \wedge S \wedge R|T; (P + Q) \wedge R \wedge T\} \quad (2)$$

2.3 Peres Gate (PG)

A 3*3 Peres gate is a combination of an FG and a Toffoli gate, and this gate can perform the operations shown in Table 1. In our proposed design, a PG is used for performing the majority of operations with a QC of 4.

$$\text{Input}\{P; Q; R\} = \text{Output}\{A; A \wedge B; A|B \wedge C\} \quad (3)$$

Table 1 Operations performed by a PG

<i>A</i>	<i>B</i>	<i>C</i>	<i>P</i>	<i>Q</i>	<i>R</i>	Operation
0	<i>B</i>	<i>C</i>	<i>A</i>	<i>B</i>	<i>C</i>	Copying
1	<i>B</i>	<i>C</i>	<i>A</i>	<i>B</i> '	<i>B</i> ' <i>C</i>	EX-OR
<i>A</i>	0	<i>C</i>	<i>A</i>	<i>A</i>	<i>C</i>	Copying
<i>A</i>	1	<i>C</i>	<i>A</i>	<i>A</i> '	<i>A</i> ' <i>C</i>	NOT
<i>A</i>	<i>B</i>	0	<i>A</i>	<i>A</i> ' <i>B</i>	<i>A</i> <i>B</i>	AND
<i>A</i>	<i>B</i>	1	<i>A</i>	<i>A</i> ' <i>B</i>	(<i>A</i> <i>B</i>)'	NAND

2.4 Hybrid New Gate (HNG)

A HNG gate is used for performing addition, and the input–output relationship of a HNG is given by Eq. (4). *U* and *Q* in Eq. (4) represent the carry output and sum signal, respectively. The QC of HNG is 6 and is significantly less than that of other gates used for reversible addition.

$$\text{Input } \{P; Q; R; S\} = \text{Output } \{A; B; A \wedge B \wedge C; (A \wedge B)C \wedge A | B \wedge D\} \quad (4)$$

3 Proposed Methodology

The block-level structure of the proposed RRBCDM is shown in Fig. 1. It consists of a 4221 radix recoder unit developed using an RRG, a multiplicand multiple generator (MMG), and a PP compression unit employing gate-level BESC-RDA circuitry for BCD recognition and correction and is shown in Fig. 2. The BESC-RDA is designed using PGs, HNGs, NCGs, and FGs. The following subsections detail the function of various blocks of the proposed RRBCD multiplier.

3.1 Radix Recoder Unit

Various binary codes, viz. 2421, 4221, Gray code, 5211, 8421, and excess-3 codes, exist. However, 4221 is chosen as an appropriate code for the design of the recoding unit in the proposed multiplier, as the maximum BCD representation with weight 4221 is 9. The Boolean expression of the recoder unit is given by Eq. (5), and the corresponding logic is shown in Table 2. Figure 3 shows the block-level structure of an RRG that combines an FG and a PG to implement recoder logic. The BCD input represented by *Y* [3:0] has passed through the 4221 RRG gate. The 4221 recoder output is directly given as the select signal to the second input of PG in MMG to transmit the data. If the recoder output is one, the MMG output is passed else; all zeros are passed out. Figure 4 shows the block diagram of the radix recoder unit implemented using reversible gates. Note from Fig. 4 that the recoder output is given as a second input to

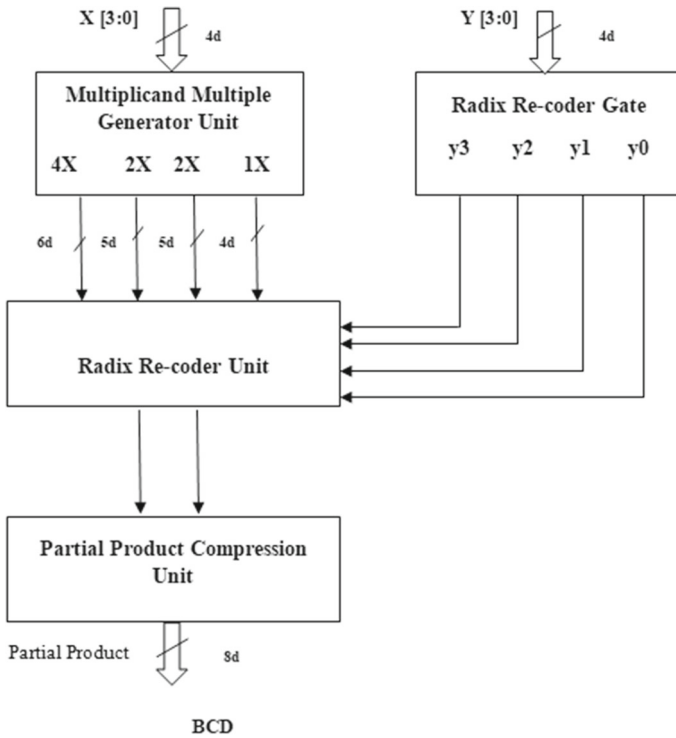


Fig. 1 Block diagram of radix binary-coded decimal multiplier

PG in the MMG unit to transmit the data or zeros based on the selected signal to the PPG unit.

$$\{y_3, y_2, y_1, y_0\} = \{Y_3 + Y_2, Y_3, Y_3 + Y_1, Y_0\} \tag{5}$$

Table 2 shows the input–output configuration of the 4221 recoder unit. The quantum representation of the RRG shown in Fig. 5 shows that the QC of RRG is 11 and that the GO count is 6 [GO1 to GO6]. Though the GO of the RRG is relatively high, it is utilized in making the gate reversible and faster and allowing it to dissipate less heat. Additionally, Fig. 5 shows that the CI count of the RRG is 5.

3.2 Multiplicand Multiple Generator

The multiplicand multiple generations are performed in parallel using the MMG unit. The MMG unit uses 4 dual PGs to generate a $4 \times$ block and 12 PGs to generate $2X$, $2X$, and $1X$ blocks. Dual PGs are used instead of PGs because a $4X$ block cannot be directly passed to the PP compression unit in the next stage, as the proposed detection/correction logic works for a maximum value of 18. Therefore, the $4X$ block is generated as $2X + 2X$. Although this adds to an extra quantum cost count of 16

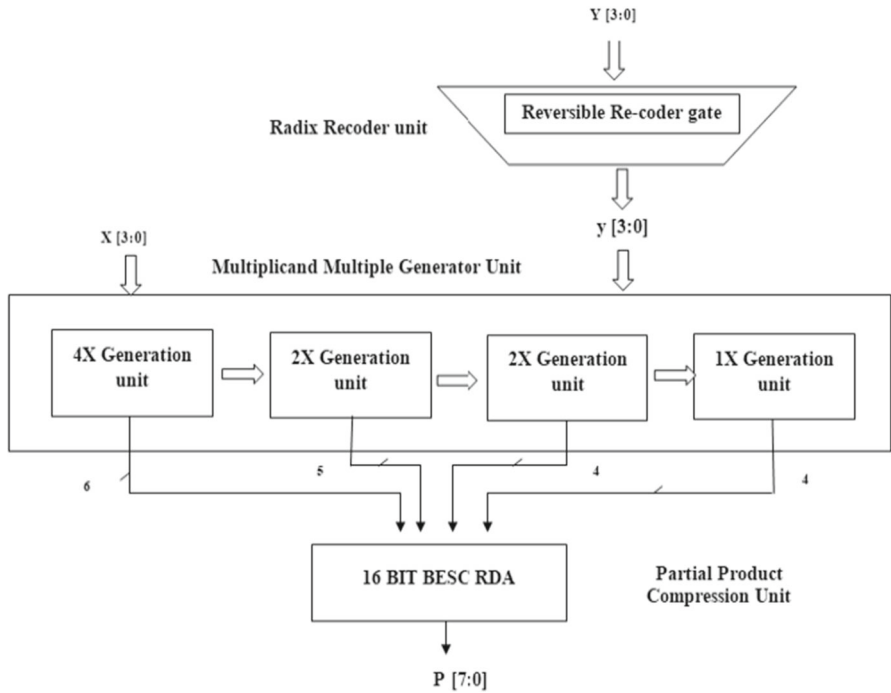


Fig. 2 One-digit reversible radix binary-coded decimal multiplier (1-d RRBCDM)

Table 2 Input–output configuration of 4221 radix recoder units

Decimal	Y3	Y2	Y1	Y0	y3 (Y4i)	y2 (Y2i)	y1 (Y2i)	y0 (Y1i)
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	1	0
3	0	0	1	1	0	0	1	1
4	0	1	0	0	1	0	0	0
5	0	1	0	1	1	0	0	1
6	0	1	1	0	1	0	1	0
7	0	1	1	1	1	0	1	1
8	1	0	0	0	1	1	1	0
9	1	0	0	1	1	1	1	1

and one extra BESC-RDA, the delay is reduced, and the output is maintained in BCD form. A 3*3 PG is utilized to function as a multiplexer to transfer data or zeros. The PG receives the input signal X [3:0] as the first input. The PG receives the select signal from the radix recoding unit as the second input. X [3:0] and Y [3:0] are the multiplicand and multiplier, respectively. Y [3:0] is fed to the 4221 recoding unit, and the recoded output y[3:0] obtained from the RRG unit is used as the second input of

Fig. 3 Block-level structure of the RRG

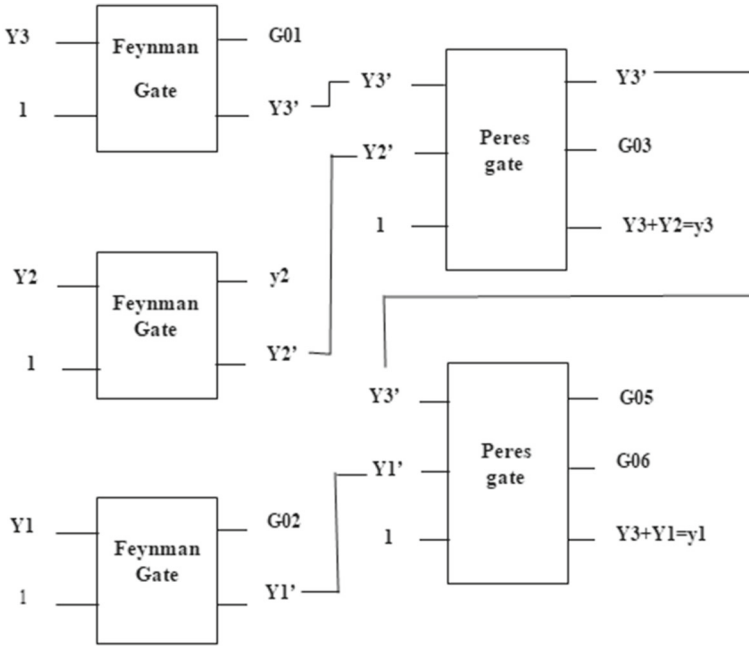
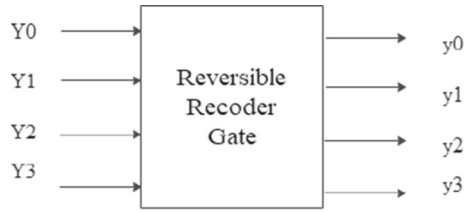


Fig. 4 Block-level implementation of the RRG

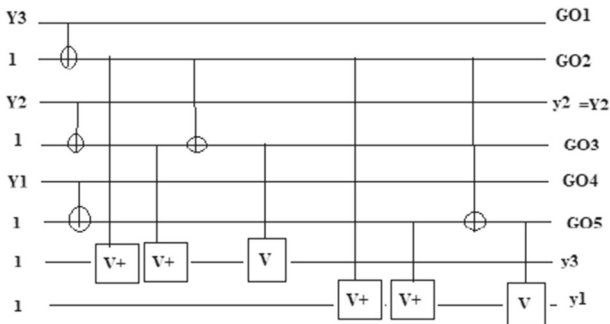


Fig. 5 Quantum representation of the RRG

the dual PGs and PGs of the 4X, 2X, 2X, and 1X blocks. The third input of the PG is fed with a constant logic low signal.

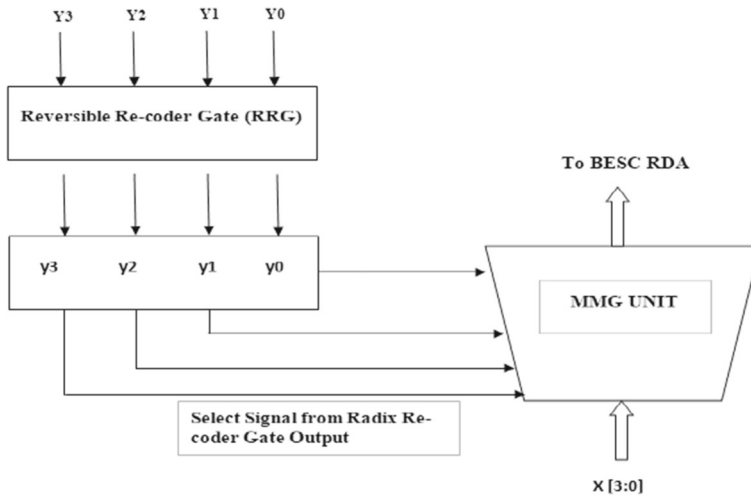


Fig. 6 Block diagram of the radix recoder unit in 1-RRBCDM

The first output of the dual PGs in the 4X block is $X [3:0]$. This is passed as the first input of the 2X block and vice versa to maintain the fan-out at 1. The second output of the dual PGs in the 4X and PGs in the 2X and 1X contributes to the garbage output. The total garbage output of the MMG unit is 20. If $y [3:0]$ is logic high, it transmits $X [3:0]$; otherwise, it transmits zeroes to the PP compression unit. However, 2X is obtained just by shifting the MMG output by appending zeroes as constant inputs. For example, $9*7$ in decimal indicates $X [3:0]$ is [1001] and $Y [3:0]$ is [0111]. Hence, $y [3:0]$ from the RRG is [1011]. $Y [3]$ activates the 4X unit and will generate two 2X units, i.e., 10010(18) and 10010 (18). The last 0 is appended as constant input in the PP compression unit, and addition is performed to produce 4X (36). Similarly, 2X is generated, as $y [1]$ is high. Again, a zero is appended to make it 18 in the PP compression unit shown in Fig. 6. The appending of zeroes is made possible due to the parallel performance of reversible gates and forms the background behind the selection of the 4221 recoding in the proposed system. As $y [0]$ is active high, nine is produced from the 1X unit, and 63 ($36 + 18 + 9$) is obtained as an output of 1-RRBCDM. The block-level implementation of the MMG unit is shown in Fig. 7.

3.3 Partial Product Compression

The reversible addition employs a BESC-RDA designed using HNGs, FGs, NCGs, and PGs. Based on the data transmitted from the MMG, an optimized PP compression unit is proposed. The 1-digit RDA has a QC of approximately 43. Higher-digit adders can be realized by cascading the 1-digit RDA. If the QC of the 1-digit RDA is M , then the QC of the N -digit n -bit adder is that given in Eq. (6). The QC of 4-digit (16-bit) BESC-RDA is $(4*43) + (6*12) = 244$. BESC-RDA involves overflow detection logic to check for overflow in the sum output of the adder. The BESC unit expression for the overflow detection signal is modified as $C_{out} = C_4 \hat{S}_3 (S_2 + S_1)$ and is shown in

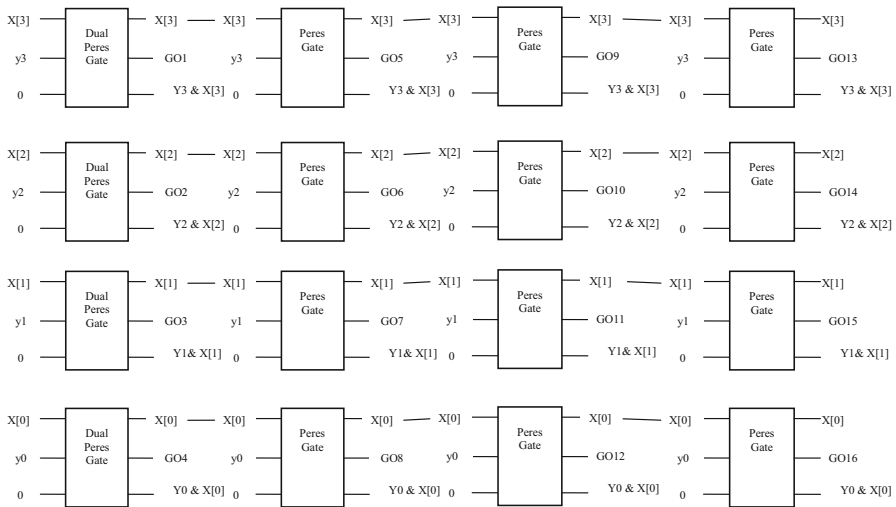


Fig. 7 Block diagram of the MMG unit

Fig. 8. For 4X generation, 1-digit RRBCDM employs a 16-bit BESC-RDA along with an 8-bit BESC-RDA in PP compression.

$$QC_nbit\ RDA = N * M + 6(n - 4) \tag{6}$$

3.4 Implementation of the 8*8 Bit Multiplier

In this part, the Urdhva-Tiryagbhyam-based Vedic multiplication technique is used for n -bit decimal multiplication. High-frequency operation increases processing power and leads to high power dissipation at high operating temperatures. This technique mainly generates partial products in parallel to simultaneously perform the addition operation, making the multiplier independent of the clock frequency operation. The Vedic concept is used along with the proposed reversible radix concept for $n \times n$ bit multiplications. In the 8×8 bit multiplier, the partial products are calculated in parallel, and the delay is decreased. Figure 9 shows the sequence of steps followed in the proposed multiplication algorithm. Note from the illustration block diagrams of the 8×8 multiplier shown in Fig. 9 that a total of four 4×4 RRBCDM modules, three 8-bit BESC-RDA units, and two 4-bit BESC-RDA units are required to design a 2-digit multiplier. The literature states that only three 8-bit adders are used for PP compression in the binary multiplier. However, in the proposed decimal system, two extra 4-bit BESC-RDAs are needed to reduce the computation time and preserve the output in decimal form.

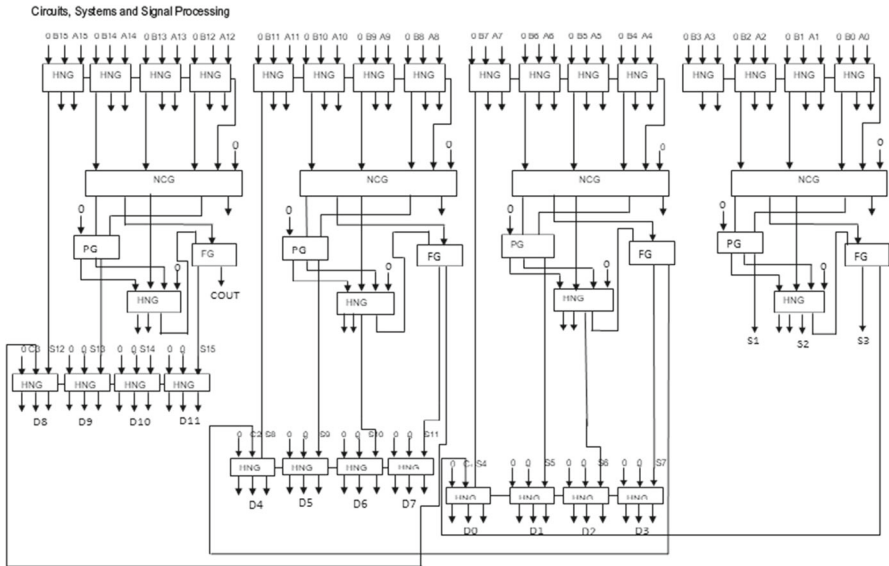


Fig. 8 The 16-bit binary to excess-six corrected reversible decimal adder (BESC-RDA)

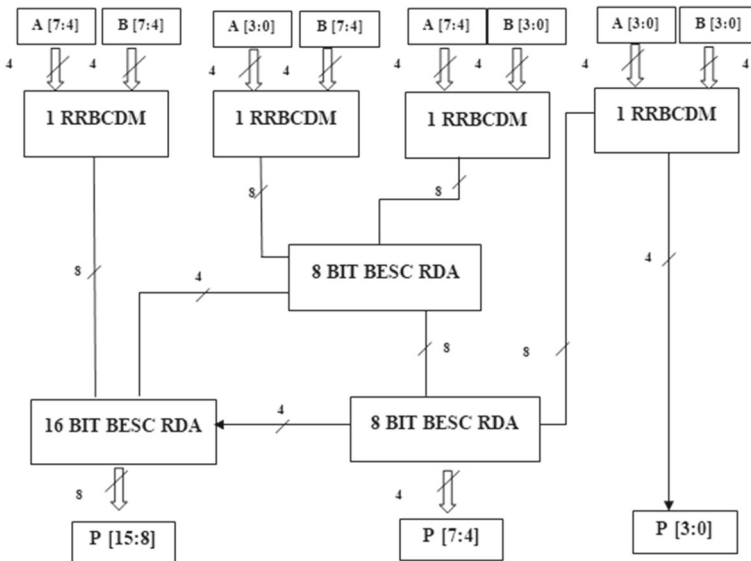


Fig. 9 Two-digit reversible radix binary-coded decimal multiplier (2-d RRBCDM)

3.5 Implementation of the 16 × 16 Multiplier

In this section, we will discuss the design of a 16 × 16 (4-d RRBCDM) multiplier using the proposed methodology. Figure 10 shows the block diagram of the 4-d RRBCDM based on the proposed methodology. The inputs are represented as $a [15:0]$ and b

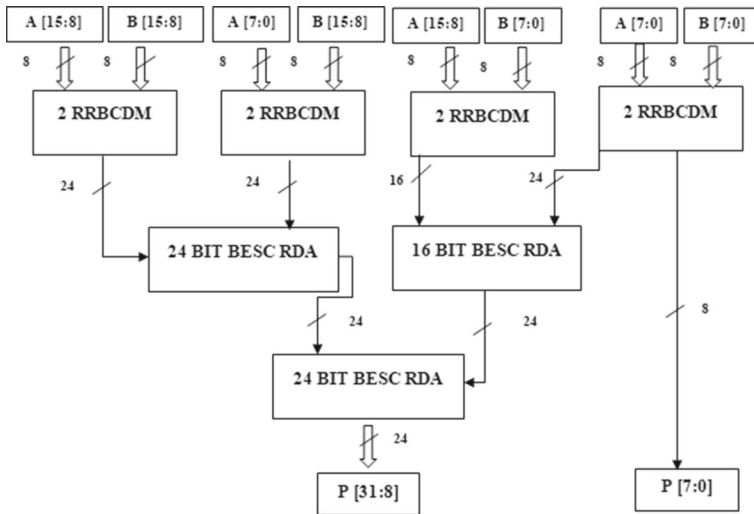


Fig. 10 Four-digit reversible radix binary-coded decimal multiplier (4-d RRBCDM)

[15:0], and the corresponding output of the multiplier is represented as $p[31:0]$. The PPs are generated in parallel, hence reducing the delay effectiveness. The multiplier is implemented using four 2-d RRBCDM, two 24-bit BESCRDA, and 16-bit BESCRDA. In total, two 32-bit BESCRDAs are used. Note from Fig. 10 that the 4-digit RRBCDM employs four 16-digit BESCRDAs. The novelty of the proposed methodology is that it is scalable, thus making the realization of N -digit RRBCDM feasible for application-specific designs. If $n = 32$, four 16-bit ($N = 4$ digits) multipliers are required, as shown in Fig. 10, and eight 16-bit BESCRDAs. Similarly, all higher size multipliers can easily be implemented using the proposed approach. The N -digit multiplier is obtained by cascading four $N/2 * N/2$ multipliers and N times 16-bit BESCRDA, as shown in Fig. 11.

4 Results and Discussion

4.1 Quantum Metrics

The various quantum parameters that are used to evaluate the performance of reversible circuits are the QC, CI, GO, and total quantum operating cost (TQOC). The QC represents the cost of the circuit in terms of the cost of a primitive gate. The QC is calculated by counting the number of primitive reversible logic gates ($1*1$ or $2*2$) required to realize the circuit. The QC of a $1*1$ gate is 0, and that of a $2*2$ gate is 1. The QC of the reversible gates used in the design is given in Sect. 2. The CI represents the number of inputs that have to be maintained constant at either 0 or 1 to synthesize the given logical function and maintain reversibility. The GO represents the extra outputs that are not used in the synthesis of a given circuit but are added to make an n -input,

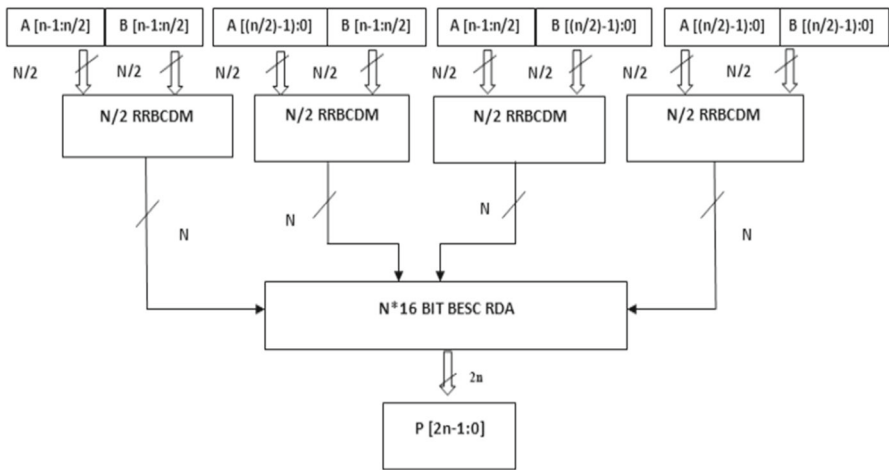


Fig. 11 *N*-digit reversible radix binary-coded decimal multiplier (*N*-RRBCDM)

Table 3 Quantum parameters of the proposed 1-RRBCDM

Blocks	QC	CI	GI	RGC	TQOC
Recoding unit	11	5	6	5	21
Radix MMG	80	20	20	20	120
PP compression unit	306	50	100	56	412
Total	397	75	126	81	553

k-output function ($(n; k)$ function) reversible. Equation (7) gives the relation between the numbers of GOs and CIs to be maintained in the reversible gates

$$\text{Input} + \text{CI} = \text{Logical output} + \text{GO} \tag{7}$$

$$\text{TQOC} = \text{QC} + \text{CI} + \text{Reversible Gate Count.} \tag{8}$$

The quantum parameters of 1-RRBCDM are shown in Table 3. The main significance of the proposed methodology is the 4221-based reversible recoding unit, which achieves an 18% improvement in QC compared to those of the designs in [17, 13, [21]. The recoder and MMG perform parallel operations, thus providing a significant delay reduction. Additionally, the proposed design demonstrates a significant reduction in quantum parameters due to the BESC-RDA-based PP compression unit. Evaluations revealed that the RRG, MMG, and PPG units contribute 3%, 20%, and 77% of the QC; 7%, 26%, and 67% of the CI; 7%, 24%, and 68% of the reversible gate count (RGC); and 4.3%, 21.5%, and 74.2% of the TQOC. Additionally, note from Table 3 that the PP compression unit contributes to approximately 70% of the quantum metrics.

Table 4 compares the quantum parameters of the proposed design to those in Refs [13, 21, 17]. Note from Table 4 that the proposed design demonstrates significant

Table 4 Comparison of the quantum parameters of 1-RRBCDM

Designs	QC	CI	GI	RGC
Proposed	397	75	126	81
Rajmohan et al. [17]	506	91	123	96
Lang and Nannarelli (2006) Radix-5 multiplier [13]	595	157	134	122
Vazquez et al. (2014) Radix-10 multiplier [21]	647	136	232	142

QC reduction compared to those of prior designs due to the proposed recoding and MMG units. Figure 12 shows the TQOC and GC plots of the proposed and prior reversible designs. Note that proposed design demonstrates 20%, 36.7% and 40.2% TQOC reductions, respectively, compared to those of the designs in Refs [17, 13, 21] respectively.

The novelty of the proposed methodology is the ease of scaling for higher digits in the input operand. By using the proposed 4×4 binary multiplier and BESC-RDA block, higher digit ($n = 4, 8, 16, 32$) multipliers can be realized. As the PP compression unit plays a significant role in large multipliers, the quantum metrics of BESC-RDA are obtained for $n = 4, 8, 16$, which is shown in Table 5. Considering N to be the digit size, n to be the bit size, and the QC, CI, GO, and RGC of the 1-digit adder to be $QC_1 = 43$, $CI_1 = 7$, $GO_1 = 15$, and $RGC_1 = 8$, then the quantum parameters of the N -digit BESC-RDA can be obtained using Eqs. (9)–(12).

$$QC_N \text{ bit RDA} = N * QC_1 + 6(n - 4) \tag{9}$$

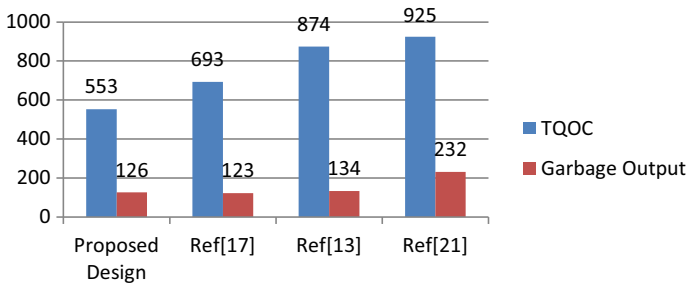


Fig. 12 Comparison of the quantum metrics of 1-RRBCDM

Table 5 Quantum metrics of BESC-RDA

Quantum metrics	1-d BESC-RDA (4-bit adder)	2-d BESC-RDA (8-bit adder)	4-d BESC-RDA (16-bit adder)
QC	43	102	224
CI	7	19	41
GO	15	34	68
RGC	8	22	44

Table 6 Quantum parameters of n -RRDM ($n = 1, 2, 4, 8$)

Multiplier size	Blocks	QC	CI	GI	RGC	TQOC
4*4	1-RRDM unit	397	75	126	81	553
8*8	4 *1-RRDM, 2*16 BESC-RDA	2036	382	640	412	2828
16*16	4 *2-RRDM, 4*16 BESC-RDA	9040	1692	2832	1824	12556
32*32	4 *4-RRDM, 4*16 BESC-RDA	37952	7096	11872	7648	52696

$$CI_N = N * CI_1 + (4N - 3) \quad (10)$$

$$GO_N = N * GO_1 + 2N \quad (11)$$

$$RGC_N = 2RGC_1 + 3N \quad (12)$$

The quantum metrics of various higher bit-width multipliers ($n = 4, 8, 16, 32$) are realized using 4×4 RRBCDM as the basic cell and the Urdhva Tiryagbhyam Vedic algorithm, which are shown in Table 6.

4.2 Performance Metrics of 1-d RRBCDM in the ASIC platform

The proposed 1-digit RRBCDM and state-of-the-art BCD designs are designed with Cadence Virtuoso Design Environment with 180-nm ASIC technology and simulated using Cadence Spectre. In simulations, parameters such as the supply voltage variation, frequency variation, and temperature variation are analyzed to determine the optimized working conditions for the multiplier.

4.3 Supply Voltage Analysis

PP compression is the critical stage in the multiplication operation, and the main functional block of the RRBCDM is the BESC-RDA unit. Hence, we constructed a plot of supply voltage against the PDP for the reversible adder, as shown in Fig. 13. Note that in Fig. 13, the PDP of the reversible adder is minimal at a supply voltage of 3 V, and hence, we used the same value for all further simulations with 180-nm ASIC technology.

4.4 Temperature Analysis

Another essential metric that designers and manufacturers have taken into consideration in ASIC implementation is the design sensitivity to temperature. Contemporarily, circuits must be able to perform as expected under different temperatures. In this

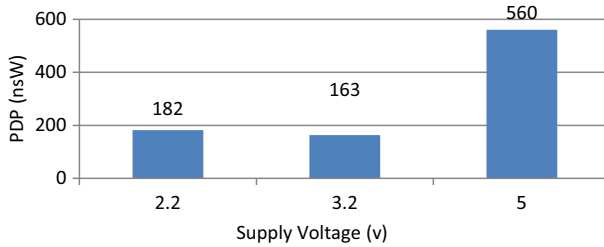


Fig. 13 Supply voltage vs PDP of BESC-RDA

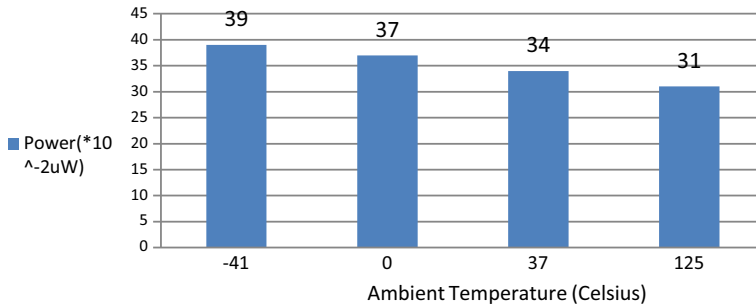


Fig. 14 Temperature versus power analysis

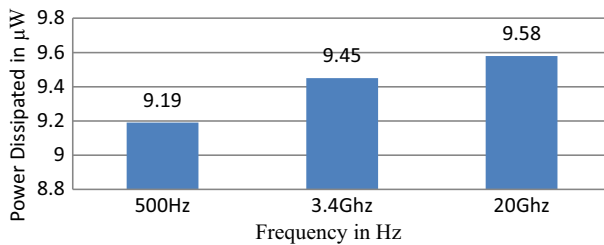


Fig. 15 Power versus frequency analysis

regard, the proposed RRBCDM was investigated under various temperatures, ranging from $-40\text{ }^{\circ}\text{C}$ up to $125\text{ }^{\circ}\text{C}$, to measure their sensitivity. Figure 14 shows the plot of the average power consumption against the temperature. Figure 14 shows that the proposed RRBCDM displays a constant behavior and high stability in terms of average power consumption versus temperature variation.

4.5 Frequency Analysis

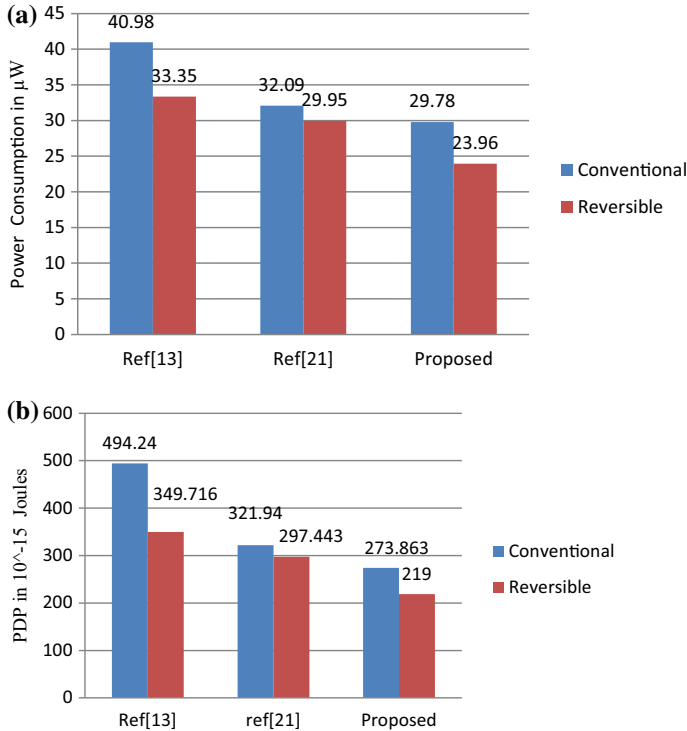
To evaluate the suitability of the proposed BCD design for high-frequency applications, we estimated the power dissipation for various frequencies, which is shown in Fig. 15. Figure 15 shows that power dissipation increases by $2\%/Hz$ and $1\%/Hz$ for operating frequencies below and above 3.4 GHz , respectively.

Table 7 Power, area, and delay comparison of the proposed RRBCD multiplier and prior algorithms in 90-nm and 180-nm ASIC Technology ($n = 1$)

Architecture	Area (μm^2)		Power (μW)		Delay(ps)		PDP (10–15 joules)	
	90 nm	180 nm	90 nm	180 nm	90 nm	180 nm	90 nm	180 nm
Lang and Nannarelli Radix-5 multiplier [13]	1302	4297	6.35	33.354	8065	10485	51.21	349.71
Vazquez et al. Radix-10 multiplier [21]	1202	3963	5.71	29.95	7639	9931	43.61	297.44
Gao et al. [7]	1191	2765	4.98	26.35	9756	10,08	48.58	265.76
Proposed radix design	1035	3413	4.56	23.96	7054	9171	32.16	219.73

Table 8 Power, area, and delay comparison of the proposed RRBCDM for varying n in 180-nm ASIC Technology

No. of input digits (n)	Area (μm^2)	Power (μW)	Delay (ps)	PDP (10^{-15} joules)	ADP ($\mu\text{m}^2\text{-us}$)
1	3413	23.96	9171	219.737	31.3
2	6232	52.68	16725	881.073	104.230
4	10302	99.68	31784	3168.229	327.407

**Fig. 16** **a** Power comparison of the reversible BCD design with the conventional counterparts. **b** PDP comparison of the reversible BCD design with the conventional counterparts

4.6 Power, Area and Delay Metrics

Table 7 compares the performance parameters, viz. the area, power dissipation, worst-case propagation delay, area-delay product (ADP), and PDP of the 1-d RRBCDM and prior designs in 180-nm ASIC technology. The simulation environment adjusts to 3.2 V as the power supply and 32 mega-hertz (MHz) as the operating frequency. Note from Table 7 that the proposed design exhibits 28% and 25% power reductions and 20.5% and 13.8% area reductions compared to those of the designs of Lang and Nannarelli (2006) [13] and Vazquez et al. (2014) [21], respectively, due to the proposed BESC-RDA logic and reversible gate used that realize BCD correction and constant multiple

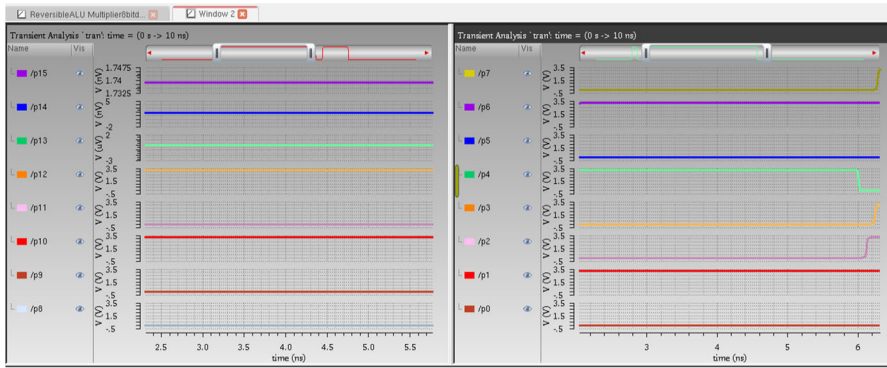


Fig. 17 Simulation waveforms of the 8-bit reversible multiplier

generations, respectively, with the minimum number of gates. Proportionately, the PDP of the proposed design is reduced by 37% and 26% compared to those of the designs of Lang and Nannarelli [13] and Vazquez et al. [21], respectively. Additionally, note that area, power, and PDP scale down occur when switching from 180- to 90-nm PDK technology.

In addition, we estimated the power, area, and delay of the proposed approach for different numbers of digits in the input (i.e., $n = 1, 2, 4$), which are shown in Table 8. Table 8 shows that the area and delay increase less proportionately for higher values of n in the input. The power and PDP metrics of the proposed BCD design increase for $2 \times$ increases in n .

Figure 16 compares the proposed RRBCDM and state-of-the-art BCD algorithms with the conventional counterparts in terms of power and PDP. Figure 16 shows that the power and PDP of the proposed and Ref [13] and Ref [21] reversible designs are reduced by 19.5% and 20%, 28% and 37% and 25% and 26%, respectively, compared to those of the conventional counterparts.

Figure 17 shows the simulation output waveforms of the proposed 2-d RRBCDM with inputs $X = 00100010$ and $Y = 01100110$ and the corresponding output 0001 0100 0101 0010.

5 Conclusion

In this brief, we proposed a novel methodology for BCD multiplication in reversible logic. The proposed design provides better performance in terms of area, power dissipation, and delay with efficient gate-level architecture BESC-RDA and simple reversible copying circuit for constant multiple generations. The functionality and driving capability of the proposed logic are evaluated with higher-digit implementation. Evaluations with higher digits in an input operand demonstrated the superior performance of the proposed BCD design by less than proportionate increases in area, power, and PDP for an increase in n . Comparison of the reversible designs with their quantum parameter counterparts revealed their preeminence in terms of not only power

and PDP but also in TQOC. The reversibility of the gates helps to trace back the input, and thus, the method is applicable in quantum computing.

Data Availability The authors declare that the data supporting the findings of this study are available within the article.

Compliance with Ethical Standards

Conflict of interest The authors declare that they have no conflicts of interest.

References

1. H.M.H. Babu, A.R. Chowdhury, in *18th International Conference on VLSI Design held jointly with 4th International Conference on Embedded Systems Design* (2005), pp. 255–260
2. C.H. Bennett, IBM J. Res. Dev. **17**(6), 525 (1973)
3. K. Bhardwaj, B.M. Deshpande, in *2013 26th International Conference on VLSI Design and 2013 12th International Conference on Embedded Systems* (2013), pp. 362–367
4. K.S.R. Baby Janagam, Int. J. Adv. Comput. Electron. Eng. **2**, 15 (2017)
5. M.A. Erle, E.M. Schwarz, M.J. Schulte, in *17th IEEE Symposium on Computer Arithmetic (ARITH'05)* (2005), pp. 21–28
6. R.P. Feynman, Found. Phys. **16**(6), 507 (1986)
7. S. Gao, D. Al-Khalili, J. Langlois, N. Chabini, *Int J Reconfig Comput*, 1 (2017)
8. S. Gorgin, G. Jaberipur, Microelectron. J. **40**(10), 1471 (2009)
9. C.E.M. Guardia, in *2012 VIII Southern Conference on Programmable Logic* (2012), pp. 1–6
10. H. Md Hasan Babu, Md. R. Islam, A. R. Chowdhury, S. Mostahed Ali Chowdhury, in *Euromicro symposium on digital system design, 2003. Proceedings* (2003), pp. 50–54
11. J.M. Jou, S.R. Kuang, R.D. Chen, IEEE Trans Circuits Syst II Analog Digit Signal Process **46**(6), 836 (1999)
12. R. Landauer, IBM J. Res. Dev. **5**(3), 183 (1961)
13. T. Lang, A. Nannarelli, in *2006 Fortieth Asilomar Conference on Signals, Systems and Computers* (2006), pp. 313–317
14. M. Mohammadi, M. Eshghi, Quantum Inf. Process. **8**(4), 297 (2009)
15. S. Narayanamoorthy, H.A. Moghaddam, Z. Liu, T. Park, N.S. Kim, in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* **23**(6), 1180 (2015)
16. M.A. Nielsen, I. Chuang. *Quantum computation and quantum information* (2002)
17. V.R. Rajmohan, M. Rajmohan, J. Comput. **2**, 112 (2010)
18. H. Thapliyal, M.B. Srinivas, in *IEEE International Conference on Computer Systems and Applications* (2006), pp. 100–103
19. H. Thapliyal, M. Srinivas, in *Asia-Pacific Conference on Advances in Computer Systems Architecture* (Springer, 2005), pp. 805–817
20. H. Thapliyal, S. Kotiyal, M.B. Srinivas, in *19th International Conference on VLSI Design held jointly with 5th International Conference on Embedded Systems Design (VLSID'06)* (2006), pp. 6
21. E.A. Vázquez, J.D. Bruguera, IEEE Trans. Comput. **63**(8), 1902 (2014)
22. E.A. Vazquez, P. Montuschi, IEEE Trans. Comput. **59**(5), 679 (2010)
23. Vazquez, E. Antelo, P. Montuschi, in *18th IEEE Symposium on Computer Arithmetic (ARITH'07)*, (2007), pp. 195–204
24. R. Wille, R. Drechsler, in *2009 46th ACM/IEEE Design Automation Conference* (2009), pp. 270–275

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.