# Low-Power Reconfigurable Adders for Variable Accuracy Hybrid Applications

[1]R.Jothin, [2]C.Vasanthanayaki
[1]Associate Professor, [2]Professor
[1, 2] Department of Electronics and Communication Engineering
[1]KGiSL Institute of Technology, Coimbatore 641035, India, [2] Government College of Technology, Coimbatore 641013, India

***Abstract:*** Nowadays in modern technology, reconfigurable modules are essentially needed to meet a variable accuracy requirement up to 100% in hybrid applications. Hence in this paper, 16-bit carry select adder structure based Variable Accuracy Reconfigurable Adders (VARA1-7) are proposed by using the single bit Reconfigurable Full Adder (RCFA). The RCFA module is based on a structural hierarchy of AND-OR logic and multiplexer logic to reduce power consumption. Further, the performance of the proposed 16-bit adders is evaluated by comparing its speed, area, power, number of variable accuracy output for input with those of the existing carry select adder. Based on the simulation results, the proposed VARA7 shows a high speed, lesser power delay product. But the proposed VARA4 structure consumes 5.67% lesser power consumption and 0.23% lesser power delay product with variable accuracy output up to 100% for an input operand pair at the cost of 5.76% area and 5.5% speed than the existing conventional 16-bit CSLA.

***Index Terms* – RCFA, VARA, Real Time, CSLA, Hybrid Application, Low Power.**

## I. INTRODUCTION

Minimizing the power consumption of circuits is important for a wide variety of high quantity digital data computing applications, because of the increasing levels of integration and the desire for portability. Power reduction has to be addressed at every design level, like gate and transistor-level technology where most of the power can be saved at the high level of abstraction. At the gate level of high performance architectures, an optimized compact design is desired to achieve energy efficiency and to be reliable for high quantity digital data computing hybrid applications. The hybrid application produces different outputs from the worst case to best case accuracy. Several optimization techniques have been proposed to minimize the area of the design while maintaining the performance.They are path based optimization and global optimization in the design. In path-based optimization, gates in the critical paths are upsized to achieve the desired performance, while the gates counts in the off critical paths are reduced to achieve low power consumption. In the global optimization, all gates counts in the architecture are globally optimized for a given delay. The architectures are mostly designed for the highest performance to satisfy the overall system cycle time requirements. They are composed of large and highly parallel architectures with logic regularity. As such, the static power consumption is substantial for such architectures. However, every application does not require a fast circuit to operate at the highest performance level all the time. Different circuit techniques have been proposed to reduce power and energy consumption utilizing the timing slack without impacting performance. These techniques can be categorized based on when and how they utilize the available timing slack. The logic gates optimization and switching activity in the critical path basically influences the area, speed, power dissipation, and the wiring complexity of a chip level VLSI system. In a VLSI system, application-specific digital signal processing architecture has been implemented for a high quantity digital data computing applications [1]. The performance of the digital signal processor core is arisen, which depend on the configuration, design parameters and effective utilization of the data path and on-chip memory architectures. The performance of the most critical functional units in the data path unit is totally dependent on being adders. While considering the elementary structure of an image processing application, it is a combination of the multipliers and delays, which in turn are the combination of the adders in the data path unit. If adders are too slow or consume more energy, the overall performance of the processor will be degraded. Initially, a conventional adder was used to operate at high accuracy rates for image processing applications [2].
The functional behavior of the existing conventional full adder (CFA) to form a carry and sum output signals are given in equations (1) and (2).

$$CARRY = (A \oplus B) \cdot C + A \cdot B \qquad (1)$$

$$SUM = (A \oplus B) \oplus C \qquad (2)$$

The gate level logic implementation of CFA using half adder has thirteen basic logic gates (AND, NOT and OR gates), six logic delays in the sum output and five logic delays in the carry propagation output. Normally, the existing conventional full adder has been employed in the simplest Ripple Carry Adder (RCA) and the Carry Select Adder (CSLA) structures in the data path unit [3]. After that in VLSI design, approximation techniques were developed that can be found in parallel or block processing either to increase the effective throughput or to reduce the power consumption at moderate accuracy rates for the human perception [4]. Vision and hearing are the two means by which humans perceive the outside world. It is estimated that 75% of the information received by human is visual. As the human visual system identifies thousands of images based on color, it is one of the most dominant and distinguishable visual features in an image [5]. The visual perception of color starts with achromatic light source, capable of emitting electromagnetic radiation with wavelengths approximately between 400 and 700

nm. Traditionally, the error tolerant application system involves a number of approximation adder modules for parallel processing implementation [6]. In recent years, an Error-Tolerant Adder (ETA) is employed to achieve low power, high speed and area efficiency with moderate accuracy rates for image processing applications. Approximate full adder integrated with conventional full adder plays a critical role to implement an error-tolerant adder at the cost of accuracy [7]. Logic complexity reduction at the transistor level six Mirror Adder (MA) cell was proposed to implement low power approximate adders for error-tolerant applications [8]. Three approximate adders (AXAs) based on an XOR/XNOR was proposed using complementary pass transistor logic adders to obtain area and energy efficiency [9]. In a bit-width-aware constant-delay run-time Accuracy Programmable Adder (APA), the probability of input combinations exhibiting accurate results is programmable and adaptively controlled by the number of iterations [10].Quantum-dot cellular automata (QCA) error tolerant adder has emerged as an attractive alternative to CMOS technology based adder in nanoscale era [11]. A novel approximate adder that exploits the generate signals for carry speculation. Furthermore, a low-area overhead module is employed to reduce the relative error and a sign correction module to fix the sign error [12]. The ETAI reduces the maximum carry path by dividing the total number of bits into inaccurate and accurate parts. The normal addition is provided at least significant bit to most significant bit for the accurate part and special addition from most significant part to the least significant path is used for the inaccurate part. ETAI exhibited low accuracy when the input operand values fall in the inaccurate part. In order to avoid this limitation, ETAII was proposed by the group the bits for non-overlapping sub-adders to reduce the maximum carry propagation. ETAII has low accuracy for large word size of input operands. A modified version of ETAII (ETAIIM) was proposed to overcome the above-said limitation of ETAII, by incorporating sub-adders in the most significant portion which in turn enhance the accuracy [13]. Energy efficient, low power error tolerant adder (ELAETA-I or ELAETA-II) has error sensitive circuit in the most significant bit position of the inaccurate part, which computes the carry and appropriate addition of carry to the least significant bit position of the accurate part. It increases the accuracy by 20% when normal OR operation is performed instead of XOR operation on the inaccurate part of area efficiency over the existing ETA-I or ETA-II, respectively [14]. Error detection and correction mechanism were introduced in the approximate adders ACA-I, ACA-II [15]. In ACA-I, multiple sub-adders with one resultant bit per sub-adder was used in the critical path carry chain to get more accurate output with area overhead. To reduce the area overhead, ACA-II was proposed by introducing multiple overlapping sub-adders with half of sub adder length resultant bits. Gracefully-degrading adder (GDA) has multiple non-overlapping sub-adder and carry look ahead structure for carry prediction block. The multiplexer also used to select a carry from the previous sub-adder or from the carry prediction block [16]. Generic accuracy configurable adder (GeAr) has K "L-bit" (where L = half of the word size and K=number of sub-adders) overlapping sub-adders to perform the ACA-I, ACA-II and ETAII function by configuring the parameters of a number of resultant bits and number of previous bit for carry prediction in sub-adder [17].

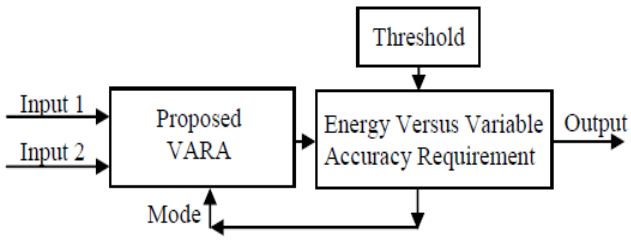


**Figure 1. Energy versus Variable Accuracy Hybrid Application**

After the chip level implementation, the inaccurate and accurate parts of the existing error tolerant adder family and the conventional adder family are statically fixed [20,21]. Hence these adders fail to produce variable accuracies multiple outputs from low to high (100%) for an input operand pair in the implementation of hybrid applications. To overcome these drawbacks, variable accuracy low power reconfigurable adders (VARA 1-7) are proposed. The inaccurate and accurate parts of the proposed reconfigurable adders are dynamically varying to exhibit variable accuracies multiple outputs for an input operand pair at the cost of area and speed for hybrid applications. The reliability of the design structure is an important criterion for energy versus variable accuracy application, which could be utilized to produce accurate and variable accuracies outputs for the hybrid (exact and inexact) application as shown in Figure 1. The accuracy of the output is lesser or greater than the threshold value, the inaccurate or accurate part of the proposed VARA is dynamically varying to achieve the threshold value in the energy versus variable accuracies requirement. Improving the performance of the full adder logic would greatly forward the single bit binary operations involved in multiple bit adders. Therefore initially, a Reconfigurable Full Adder (RCFA) module is proposed. Further, the RCFA cells are incorporated in the proposed 16-bit Variable Accuracy Reconfigurable Adders (VARA 1-7) topology to exhibit variable accuracy outputs for an input operand pair. The internal structure of the proposed design can be configured externally by the simple mode configuration and these VARA adders are more suitable to meet all range of variable and fixed accuracy requirement for the hybrid applications.

The remainder of this paper is organized as follows. Section II describes the proposed Reconfigurable Full Adder (RCFA); Comparison of simulation result of the single bit full adders is given in Section III; Design and analysis of the proposed 16-bit Variable Accuracy Reconfigurable Adders (VARA1-7) are demonstrated in Section IV; Section V concludes this paper.

## II. PROPOSED RE-CONFIGURABLE FULL ADDER (RCFA)

Logic Level optimization of the proposed reconfigurable full adder using simplified AND-OR logic and multiplexer structures using eleven basic logic gates, six logic delays in the sum output and three logic delays in the carry propagation output to achieve power reduction is shown in Figure 2.

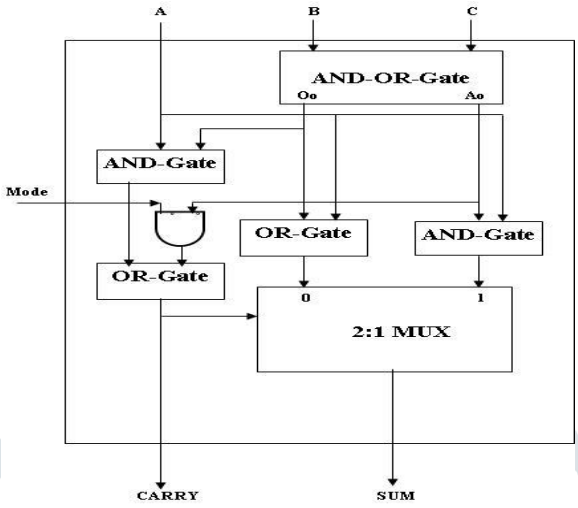


**Figure 2. Architecture of proposed RCFA**

The 2:1 multiplexer structure has four basic logic gates and three basic logic gate delays. The general architecture of AND-OR gate block which has simplified one AND and OR gate function as given in equations (3) and (4).

$$Oo = B+C \quad (3)$$

$$Ao = B \cdot C \quad (4)$$

In RCFA, the XOR gate is replaced with basic logic gates in the carry propagation output as given in equation (5) and the multiplexer based pre-computation selection is utilized to reduce power consumption.

$$CARRY = A \cdot (B+C) + (Mode) \cdot (B \cdot C) \quad (5)$$

The accurate carry signal is generated using a structural hierarchy of AND-OR gate blocks. The OR output of all inputs is applied in one of the multiplexer inputs (M_0 =A+B+C). The product of all inputs is applied in another one of the multiplexer inputs (M_1 = A·B·C). If the carry output is equal to zero, the OR output of all inputs is selected otherwise the product of all inputs is selected in the RCFA sum output. If the Mode is equal to one, accurate carry is generated for the conventional full adder operation otherwise the approximate carry is generated for an approximate full adder function which has one error in the sum and carry output out of eight cases as shown in Table 1.

**Table 1. Truth Table of Proposed Carry Select Approximate Full Adder**

| INPUTS | | | Existing CFA OUTPUTS | | MUX INPUTS OF RCFA | | Mode = 0 Proposed RCFA OUTPUTS | | Mode = 1 Proposed RCFA OUTPUTS | |
|---|---|---|---|---|---|---|---|---|---|---|
| A | B | C | C | S | M_0 | M_1 | C | S | C | S |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Highlighted box indicates the error output*

## III. Results and Discussion of Full Adders

### 3.1 Logical gate count and delay analysis

The proposed RCFA has two logical gate count savings with same critical path sum delay and two carry propagation gate delay savings when compared to the existing CFA is shown in Figure 3.

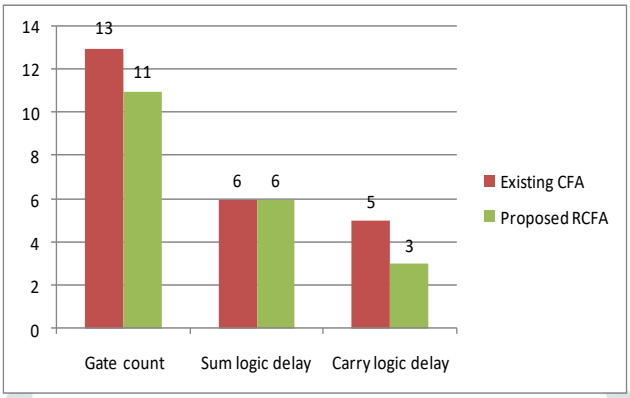


**Figure 3. Logical comparison of full adders**

The combination of input value (011) for which there is an error in the inaccurate mode due to the mode input value is equal to zero of the logic function in the carry output, if there is another error in the multiplexer sum output that reduces the error distance as one by causing a compensating deviation in the output value. Error distance is the arithmetic difference between error and exact outputs. The pass rate is represented by the number of exact outputs over than the total number of outputs. The proposed RCFA has variable pass rates and it has 87.5% pass rate for inaccurate output or 100% pass rate for precision output when compared to the CFA.

### 3.2 Design metrics analysis using Cadence Compiler

VHDL hardware description language is used to design an existing and the proposed 16-bit adders in Xilinx14.2 software. All these designs are simulated using an Isim simulator and implemented in a Field Programmable Gate Array (FPGA - xc3s500e, pq208, -4) [18] for functional verification. In addition to this, the design metrics of these designs are investigated using Cadence Encounter(R) software, which is based on a gate level Application Specific Integrated Circuits (ASIC) TSMC 90nm technology [19] is presented in Table 2. Hardware co-simulation is performed by the integration of the Xilinx system generator, predefined modules and user-defined modules (Black Box) in a MATLAB Simulink for an image blending application.
From the Table II, it is found that the proposed RCFA design has 4.35% more critical path delay and 5.88% area overhead than the existing CFA. But the proposed RCFA consumes 2.10% less power than the existing CFA. In addition to this, the RCFA provides 100% or 87.5% pass rate by the mode input configuration. Therefore, the proposed RCFA design is considered to design a 16-bit low power reconfigurable adder for variable accuracy hybrid applications.

**Table 2. Design metrics comparison of adders using cadence compiler**

| Adder Type | Cell Area ($\mu m^2$) | Total Power (nW) | Delay (ps) |
|---|---|---|---|
| Existing CFA | 17 | 365.414 | 276 |
| Proposed RCFA | 18 | 357.725 | 288 |

## IV. Design and Analysis of Proposed 16-bit variable accuracy reconfigurable adders (VARA1-7)

After the chip level fabrication, the error tolerant adder and the conventional adder family, failed to produce multiple output for variable accuracy hybrid applications. To overcome this limitation, seven numbers of 16-bit CSLA structured Variable Accuracy Reconfigurable Adder (VARA1-7) designs are proposed to exhibit variable accuracy outputs by the mode (M) selection for an input operand pair (A and B).

### 4.1 16-BIT PROPOSED VARIABLE ACCURACY RECONFIGURABLE ADDERS (VARA1-4) USING RCFA

VARA1 has a static part in the most significant 12-bits for an accurate addition and has a dynamic part in the least significant 4-bits for the accurate or inaccurate addition. In the dynamic part, the reconfigurable conventional full adder (RCFA) based 4-bit RCA structure is implemented to perform either accurate or inaccurate addition. Whereas in the static part, the existing conventional full adder (CFA) based CSLA structure is used for the accurate addition only is shown in Figure 4.

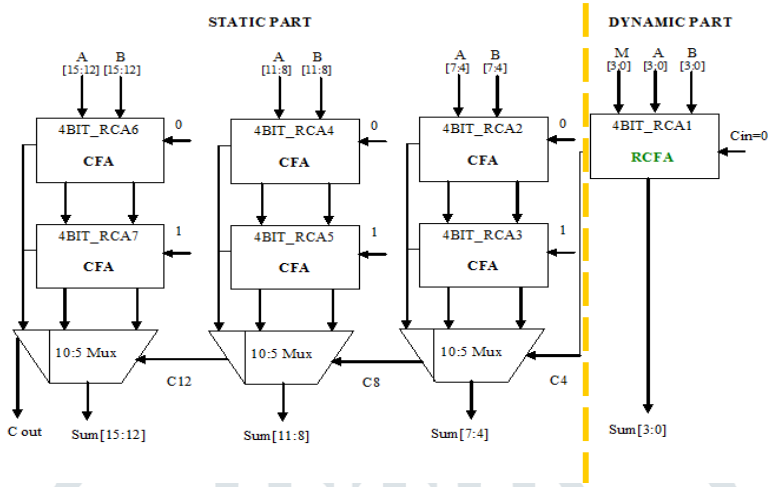


**Figure 4. Simplified block diagram of proposed 16-bit VARA1**

The four least significant bits of the input operands (A and B) are lying in the dynamic part for the accurate or inaccurate addition and these bits could be configured individually by the sixteen-level of four modes (M) configuration bits from "0000 to "1111. If the mode is equal to "1111" the VARA1 works as a conventional CSLA otherwise it exhibits variable accuracy outputs. A total number of variable accuracy outputs = $2^M$ ($2^M$ -1 inaccurate outputs and one accurate output), where M = number of mode bits in the dynamic part.

Table 3 presents the simplified design features of the VARA1-7. Similarly, VARA2 and VARA3 have the last 8 and 12 bits in the dynamic part for mode configuration to exhibit 256 and 4096 variable outputs for an input combination respectively.

**Table 3. Design features of proposed 16-bit VARA1-7**

| Adder Type | 15-12 | 11-8 | 7-4 | 3-0 |
|---|---|---|---|---|
| VARA1 | STATIC | | | DYNAMIC |
| VARA2 | STATIC | | DYNAMIC | |
| VARA3 | STATIC | DYNAMIC | | |
| VARA4 | DYNAMIC | | | |
| VARA5 | DYNAMIC | | | STATIC |
| VARA6 | DYNAMIC | | STATIC | |
| VARA7 | DYNAMIC | STATIC | | |

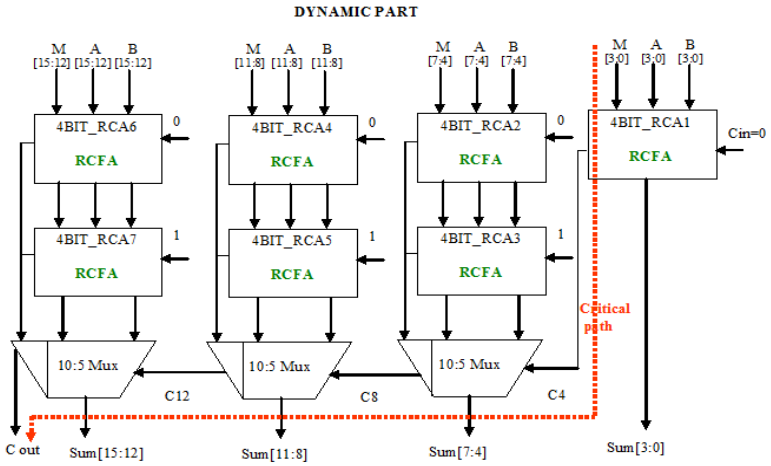


**Figure 5. Simplified block diagram of proposed 16-bit VARA4**

In the VARA4 design, 16-bits of input operands are lying in the dynamic part is shown in Figure 5. If the number of bit increases in the dynamic part, the number of possible variable accuracy outputs will be increased. The dynamic part of this design has 16-bits in the mode configuration, which are utilized to configure the bitwise accurate or inaccurate function. Therefore the design provides $2^{16}$ = 65536 number of variable accuracy outputs for an operand pair. If all the mode bits are equal to one, it acts as a conventional CSLA to exhibit one 100% accuracy output otherwise it exhibits (65535) multiple possible variable accuracy outputs for an input pair. The critical path delay depends upon the maximum delay of the carry chain. The critical path delay of the VARA4 design is the total delay of 4-bit RCA1, Mux1, Mux2, and Mux3 to carry out. Hence the overall critical path delay of the existing CSLA and the VARA4 design is compared for a speed improvement in Table 4.

**Table 4. Critical path carry chain analysis of the existing CSLA and proposed VARA4**

| ADDER | Total delay (ps) | Mux3 (ps) | Mux2 (ps) | Mux1 (ps) | 4-bit RCA1 (ps) |
|---|---|---|---|---|---|
| Con.CSLA | 1428 | 138 | 254 | 240 | 796 |
| VARA4 | 1507 | 138 | 254 | 289 | 826 |

### 4.2 16-BIT PROPOSED VARIABLE ACCURACY RECONFIGURABLE ADDERS (VARA5-VARA7) USING RCFA

To reduce the critical path delay and area, the CFA based static part is incorporated in the least significant position and the dynamic part is implemented in the most significant bit position in the VARA5-7 designs. Normally most significant bits (MSB) play an important role to represent the quality of the information than the least significant bits (LSB). Therefore the dynamic part provides a variable accuracy outputs from worst case to best level and a Static part is utilized to neglect a possible variation in the LSB part of the VARA5-7 designs. In the VARA5 adder, the static part has the last four least significant bits of the input operands. The other twelve MSB bits are used to configure the accurate or inaccurate addition by the 4096 level of mode configuration is shown in Figure 6. Similarly, in the VARA6 and VARA7 architectures, the first 8-bits and 4-bits of the dynamic part for mode configuration to exhibit 256 and 16 variable outputs from the worst case accuracy respectively are shown in Figure 7 and 8.

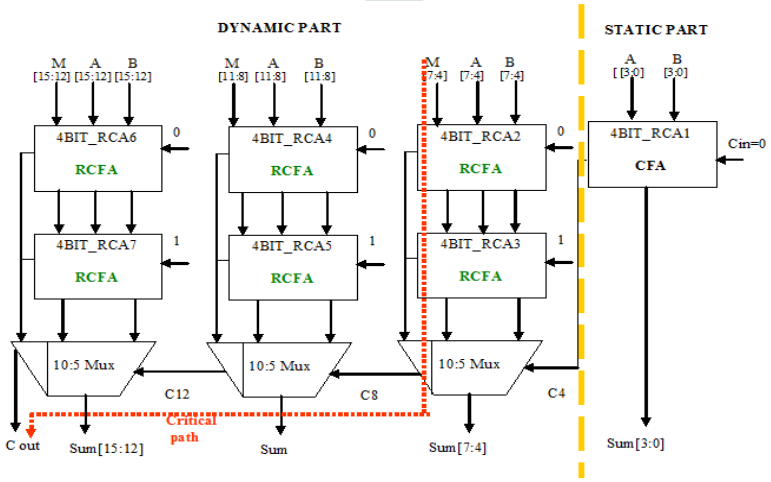


**Figure 6. Simplified block diagram of proposed 16-bit VARA5**

**Table 5. Critical path carry analysis of the proposed VARA5-7**

| ADDER | Total delay (ps) | Mux3 (ps) | Mux2 (ps) | 4-bit RCA4 (ps) | Mux1 (ps) | 4-bit RCA2 (ps) | 4-bit RCA1 (ps) |
|---|---|---|---|---|---|---|---|
| VARA5 | 1535 | 138 | 255 | - | 233 | 909 | |
| VARA6 | 1433 | 138 | 234 | 1060 | - | - | - |
| VARA7 | 1425 | 138 | 254 | - | 239 | - | 794 |

The critical path of the VARA5 or VARA6 design is changed from 4-bit RCA1of the static part to RCA2 or RCA4 of the dynamic part because the static part CFA_RCA has a lesser delay than the dynamic part RCFA_RCA is shown in Table 5.

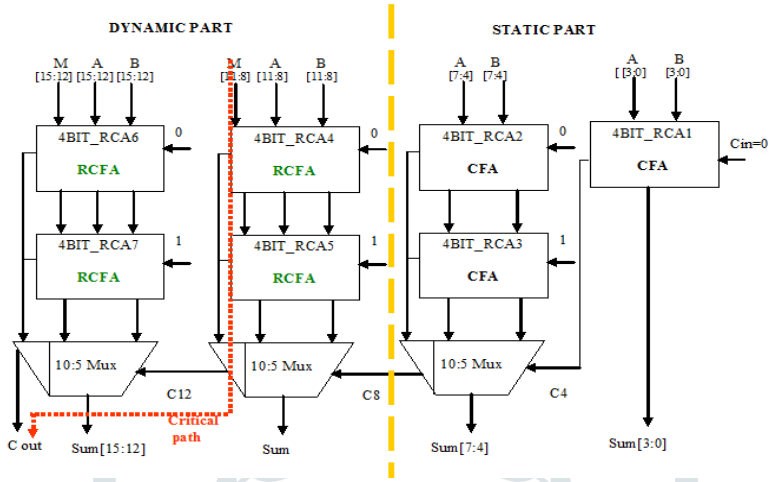


**Figure 7. Simplified block diagram of proposed 16-bit VARA6**

In the VARA7 the dynamic part delay is lesser than the static part delay. Hence the critical path delay is the sum of 4-bit RCA1, Mux1, Mux2, and Mux3 to carry out delays is shown Figure 8. The VARA7 has less delay when compared with other proposed VARA's, and existing conventional CSLA. The VARA7 provides fifteen levels of low accuracy inaccurate outputs and a 100% accuracy output. The design has lesser power consumption and lesser power delay product than the existing conventional CSLA. Hence the design is also suitable for conventional operation and low accuracy outputs.

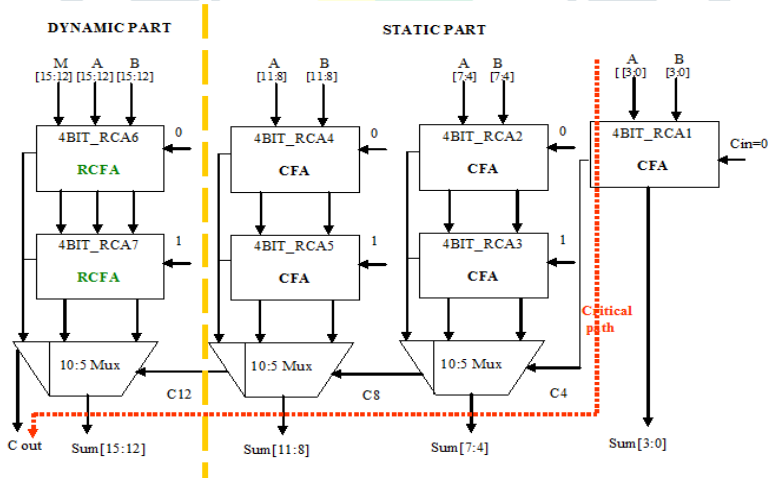


**Figure 8. Simplified block diagram of proposed 16-bit VARA7**

### 4.3 LOGICAL GATE COUNT ANALYSIS OF 16-BIT ADDERS

The existing 16-bit conventional CSLA has 424 basic logic gates (AND, OR, NOT gates). In 16-bit dual ripple carry adder it requires (7 numbers of RCA* 4 numbers of CFA in each RCA) = 7*4*13 = 364 gates and for 3 numbers in 10:5 multiplexer (5 numbers of 2:1 multiplexer in each 10:5 multiplexer), it requires (3*5*4) = 60 gates. The 16-bit VARA4 requires (7 numbers of 4-bit RCFA_RCA) = 7*4*11 = 308 logic gates and 3 numbers of 10:5 (3* 5*(2:1 mux)), it requires (3*5*4) =60 gates. Similarly the basic logic gate count of the proposed adders is calculated and it is shown in Figure 9. The VARA4 has 56, 48, 32, 16, 8, 24, 40 basic logic gate count savings than the existing CSLA and the other proposed VARA1-3, 5-7 respectively.

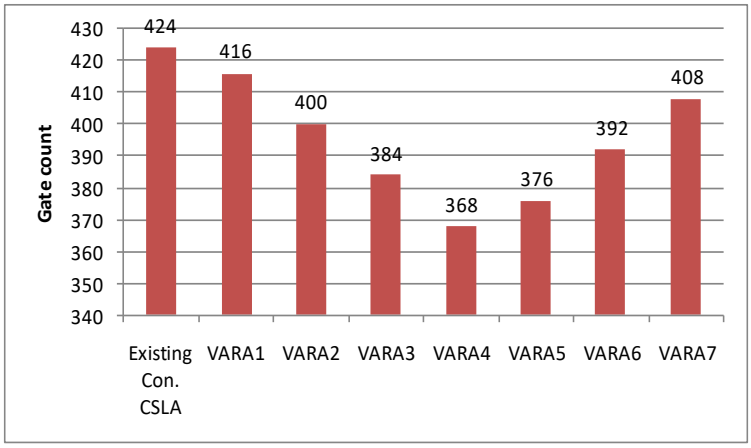


**Figure 9. Logical gate count comparison of 16-bit adders**

### 4.4 DESIGN METRICS COMPARISON OF 16-BIT ADDERS USING CADENCE RTL COMPILER

The design metrics of these existing and proposed designs are investigated using Cadence Encounter(R) software, which is based on a gate level Application Specific Integrated Circuits (ASIC) TSMC 90 nm technology. The investigated results are presented in Table 6. Based on the identical structures, the best cases of the design metrics architectures are compared. The existing conventional CSLA has 5.4% and 1.7% area savings when compared to the proposed VARA4 and VARA7. The advantage of the VARA4 design has 5.67% low power consumption and 0.23% less lesser power delay product with variable accuracy output up to 100% and it exhibits 5.5% higher delay than the existing conventional CSLA. The VARA7 has almost same delay (slightly low) and 0.6% lesser power delay product than the conventional CSLA. VARA1-3 and VARA5-6 consume low power and these proposed VARAs exhibit 16, 256, 4096, 4096, 256 variable outputs respectively for an input operand pair at the cost of area and speed when compared to the existing conventional CSLA.

**Table 6. Design metrics comparison of 16 bit conventional and variable accuracy reconfigurable adders**

| 16-bit Adder Type | Cell Area ($\mu m^2$) | Power (nW) | Delay (ps) | ADP ($\mu m^2$.ns) | PDP (fJ) | Energy (fJ)/ Cell Area ($\mu m^2$) | Function | MODE configurable bits | Number of Possible outputs for an operand Pair |
|---|---|---|---|---|---|---|---|---|---|
| Existing Conventional CSLA | 521 | 15346.663 | 1428 | 744 | 21.86 | 0.0420 | ACCURATE | - | ONE |
| Proposed VARA1 | 525 | 15013.752 | 1507 | 791 | 22.63 | 0.0430 | ACCURATE / INACCURATE | LSB 4 BITS M3-M0 | 16 |
| Proposed VARA2 | 534 | 15027.760 | 1535 | 819 | 23.07 | 0.0432 | ACCURATE / INACCURATE | LSB 8 BITS M7-M0 | 256 |
| Proposed VARA3 | 541 | 15084.228 | 1535 | 830 | 23.15 | 0.0428 | ACCURATE / INACCURATE | LSB 12 BITS M11-M0 | 4096 |
| Proposed VARA4 | 551 | 14472.797 | 1507 | 830 | 21.81 | 0.0396 | ACCURATE / INACCURATE | MSB-LSB 16 BITS M15-M0 | 65536 |
| Proposed VARA5 | 547 | 15206.798 | 1535 | 840 | 23.34 | 0.0427 | ACCURATE / INACCURATE | MSB 12 BITS M15-M4 | 4096 |
| Proposed VARA6 | 538 | 15071.779 | 1433 | 798 | 21.60 | 0.0401 | ACCURATE / INACCURATE | MSB 8 BITS M15-M8 | 256 |
| Proposed VARA7 | 530 | 14987.925 | 1425 | 756 | 21.36 | 0.0403 | ACCURATE / INACCURATE | MSB 4 BITS M15-M12 | 16 |

**Highlighted box indicates the best case*

### 4.5 QUALITY METRICS ANALYSIS USING FPGA BASED CO-SIMULATION

To analyze the overall performances, the proposed VARA1-7 and the existing designs are incorporated to implement a FPGA based Co-simulation system using MATLAB Simulink, Xilinx system generator and spartan6 FPGA for image blending application by setting or detecting target board information such as the board name (test), system clock (Clock Frequency 20Mhz, Clock Pin E10), JTAG options (Position 1, IR lengths 6) and configuration of targetable device family, part, speed and package (Spartan6, xc6slx16,-2 and ftg256) in the compilation of Xilinx system generator is shown in Figure 10 to investigate the variable quality metrics as given in equation (6) and (7) from worst case to best case accuracy.

$$PSNR = 20\log_{10}(V_{max}/\sqrt{MSE}) \qquad (6)$$

$V_{max}$, MSE, PSNR are the maximum dynamic range of the pixel value, Mean Square Error and Peak Signal to Noise Ratio, respectively for image processing applications.

$$MSE = \frac{1}{M\,X\,N}\sum_{i=1}^{M}\sum_{j=1}^{N}(X_{ij} - Y_{ij})^2 \qquad (7)$$

Image blending operation is performed based on image addition principle, but different weights are to be given for two test input images (f1 and f2). So that it gives a feeling of blending or transparency in the blended image (g). Images are added as per the equation (8) below:

$$g(x,y) = (1-\alpha)\cdot f1(x,y) + \alpha \cdot f2(x,y) \qquad (8)$$

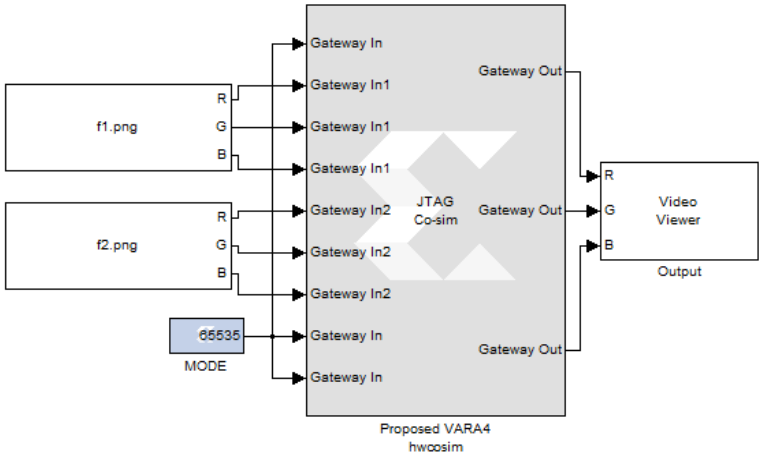


**Figure 10. Architecture of Proposed VARA4 based FPGA hardware co-simulation**

Where 'α' is the blending ratio which determines the influence of each input image in the output. By varying the values of 'α' from 0 to 1, a cool transition from one image to another is performed. This operator forms a blend of two input images of the same size. Similar to pixel addition, the value of each pixel in the output image is a linear combination of the corresponding pixel values in the input images. The coefficients of the linear combination are user-specified and they define the ratio by which to scale each image before combining them. These proportions are applied such that the output pixel values do not exceed the maximum pixel value.

**Table 7. PSNR value of VARA4 (case1, case9, case17) outputs**

| Blending factor | PSNR (dB) | | |
|---|---|---|---|
| | Case 1 | Case 9 | Case 17 |
| α = 0.25 | accurate output | 66.26 | 35.01 |
| α = 0.50 | accurate output | 64.69 | 30.49 |
| α = 0.75 | accurate output | 65.13 | 21.42 |

In some applications f2 can also be a constant, thus allowing a constant offset value to be added to a single image. 'α' can either be a constant factor for all pixels in the image or can be determined for each pixel separately using a mask. The size of the mask must then be identical with the size of the images. Table 7 shows the investigated PSNR value of the blended images for various blending factors. Table 8 presents the variable accuracy range and best case accuracy of the proposed and existing adders.
The main advantage of this VARA4 adder, which could give high range of multiple variable PSNR images from very low worst case to best case quality images are shown in Figure 11 and maintain to exhibit more or less the threshold value of PSNR (ex. More or less than 45dB) for different resolution input images after the chip-level implementation. Hence the PSNR versus

energy plays a critical role in the high quantity digital data communication. If the modes ($M_{15}$ –$M_0$) are equal to one in the mode configuration (19 cases configured instead of 65536 cases), the reconfigurable VARA4 works as a conventional CSLA to exhibit 100% accuracy otherwise it acts as a variable accuracy error tolerant CSLA as shown in Table 9. From the Table IX, the MSE value of the VARA4 increases with increase in the number of bits in the inaccurate function from the least to the most significant bit. In a 1-bit window method, each bit of input operands could be configured individually for either accurate or inaccurate addition. Hence it provides 65536 possible variable accuracy outputs from the 16-bit mode reconfiguration. Further, to reduce either the number of possible variable outputs or the number of mode reconfiguration from 65536 to 256, two adjacent bits of input operands are grouped together for an 8-bit mode reconfiguration in a 2-bit window method. Finally in the 4-bit window method, 4-bits are grouped together for the 4–bit mode reconfiguration to exhibit 16 levels of variable accuracy outputs is shown in Table 10 and also provides a best case output immediately for low (M=0011) or medium (M=0110) or high (M=1100) or equally distributed (M=1111) resolution input images.

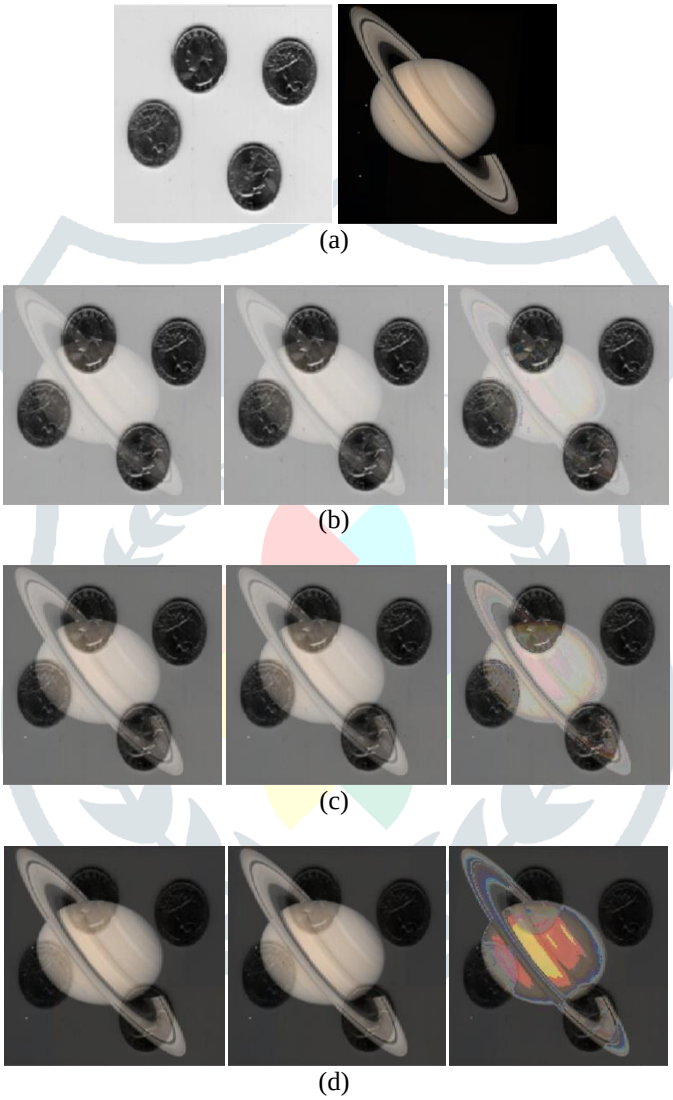

**Figure 11. (a) Input images f1 and f2, Blended images of VARA4 case 1, case 9, case 17 (b) α = 0.25 (c) α = 0.50 (d) α = 0.75**

### 4.6 ERROR METRICS ANALYSIS USING TEST VECTOR BASED SIMULATION

To analyze the overall Computational Accuracy (CA) of best case to worst case, $2^{2N}$ test vectors patterns are applied, where N=16. The simulations are carried out to examine the impact of the error on the approximate adder. For any approximate adder, the Error Distance (ED) is the difference between exact sum output (X) and the predicted approximate sum output (X'), ED=|X-X'|. The Relative Error Distance (RED) is the ratio of ED to the accurate output RED = ED/S = |X-X'|/X. The Error Rate (ER) is the ratio of incorrect outputs with respect to the total number of outputs. For any N-bit approximate adder, the Mean RED (MRED) is, $MRED = \frac{\sum_{i=0}^{2^{2N}-1} RED}{2^{2N}}$. Mean Error Distance (MED) is the average of ED, $MED = \frac{\sum ED}{2^{2N}}$. Normalized MED is, NMED = $MED/S_{max} = \frac{\frac{\sum_{i=0}^{2^{2N}-1} ED}{2^{2N}}}{V_{max}}$, Where $V_{max}$ is the maximum value of sum that can be obtained from an N-bit accurate adder, $V_{max} = (2^N-1) + (2^N-1)$.

**Table 8. Worst case and best case PSNR values of 16-bit adders**

| | Input f1 and f2 ($\alpha$=0.25) | | | |
|---|---|---|---|---|
| **16-bit Adder Type** | **Range of MSE (e+004)** | **Range of variable PSNR(dB)** | **Best case PSNR(dB)** | **Number of Variable Outputs** |
| Proposed VARA1 | 0.00038 to 0 | 90.50 to Accurate | Accurate | 16 |
| Proposed VARA2 | 0.10162 to 0 | 66.26 to Accurate | Accurate | 256 |
| Proposed VARA3 | 32.80650 to 0 | 41.17 to Accurate | Accurate | 4096 |
| Proposed VARA4 | 134.26409 to 0 | 35.05 to Accurate | Accurate | 65536 |
| Proposed VARA5 | 134.26409 to 0.00038 | 35.05 to 90.50 | Accurate | 4096 |
| Proposed VARA6 | 134.26409 to 0.10162 | 35.05 to 66.26 | Accurate | 256 |
| Proposed VARA7 | 134.26409 to 32.80650 | 35.05 to 41.17 | Accurate | 16 |
| Existing Conventional CSLA | 0 | Not variable | Accurate | 1 |

**Table 9. PSNR values of 16-bit VARA4**

| | DYNAMIC PART | | | | 1 | ACCURATE | | | 0 | INACCURATE | | | | | | | ($\alpha$=0.25) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Case** | **$M_{15}$** | **$M_{14}$** | **$M_{13}$** | **$M_{12}$** | **$M_{11}$** | **$M_{10}$** | **$M_9$** | **$M_8$** | **$M_7$** | **$M_6$** | **$M_5$** | **$M_4$** | **$M_3$** | **$M_2$** | **$M_1$** | **$M_0$** | **MSE($e^{+004}$)** | **PSNR(dB)** |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.00000 | Accurate |
| 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0.00000 | Accurate |
| 3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0.00000 | Accurate |
| 4 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0.00010 | 96.34 |
| 5 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0.00038 | 90.50 |
| 6 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0.00120 | 85.54 |
| 7 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0.00462 | 79.68 |
| 8 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.02937 | 71.65 |
| 9 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.09056 | 66.76 |
| 10 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.68700 | 57.96 |
| 11 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1.37076 | 54.96 |
| 12 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3.76669 | 50.57 |
| 13 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 32.80651 | 41.17 |
| 14 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 49.19941 | 39.41 |
| 15 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 134.26519 | 35.01 |
| 16 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 134.26472 | 34.97 |
| 17 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 134.26409 | 34.84 |
| . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | . . . | | . . . |
| 65535 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 134.26409 | 34.84 |
| 65536 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 134.26409 | 34.84 |

**Table 10. Mode configuration of 16-bit VARA4**

| Window method | Number of Mode bits | | | | | | | | | | | | | | | | No. of Outputs |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 4-bit | M4 | | | | M2 | | | | M1 | | | | M0 | | | | 16 |
| 2-bit | M8 | | M7 | | M5 | | M4 | | M3 | | M2 | | M1 | | M0 | | 256 |
| 1-bit | M15 | M14 | M13 | M12 | M11 | M10 | M9 | M8 | M7 | M6 | M5 | M4 | M3 | M2 | M1 | M0 | 65536 |

The investigated quality and error metrics of the proposed VARA1-7 are shown in Table XI. Further, MED of the proposed VARAs increases with the increase in the number of bits in the inaccurate function from the least to the most significant bit and most to the least significant bit respectively. On the other hand, the overall computational accuracy of the VARA4 decreases rapidly from 100% to 91.7880% with the increase in the number of bits in the inaccurate function from the mode configuration MC1 to MC29. Similarly, CA of the VARA1 or VARA2 or VARA3 decreases from 100% to 99.9997% (MC1-5) or 99.9509% (MC1-9) or 99.3046% (MC1-13) with the increase in the number of bits in the inaccurate function from the least to the most significant bit respectively. Moreover, the CA of the VARA7 or VARA6 or VARA5 exhibits 100% (MC1) best case CA and the CA decreases from 94.5929% to 91.9340% (MC17-20) or 91.8487% (MC17-24) or 91.8000% (MC17-28) with the increase in the number of bits in the inaccurate function from the most to the least significant bit respectively.

**Table 11. Quality and Error metrics of 16-bit Vara1-7**

| MC | Mode (HEX) | MRED | MED | NMED | CA (%) |
|---|---|---|---|---|---|
| 1 | FFFF | 0.0000 | 0.0000 | 0.0000 | 100 |
| 2 | FFFE | 0.0000 | 0.0000 | 0.0000 | 100 |
| 3 | FFFC | $1.8829e^{-006}$ | $1.3400e^{-005}$ | $1.0224e^{-010}$ | 99.9998 |
| 4 | FFF8 | $1.9398e^{-006}$ | $4.6697e^{-005}$ | $3.5627e^{-010}$ | 99.9998 |
| 5 | FFF0 | $2.4413e^{-006}$ | $1.1608e^{-004}$ | $8.8563e^{-010}$ | 99.9997 |
| 6 | FFE0 | $5.4069e^{-005}$ | $2.5762e^{-004}$ | $1.9655e^{-009}$ | 99.9946 |
| 7 | FFC0 | $1.1109e^{-004}$ | $5.3690e^{-004}$ | $4.0963e^{-009}$ | 99.9888 |
| 8 | FF80 | $2.2557e^{-004}$ | 0.0011 | $8.3924e^{-009}$ | 99.9774 |
| 9 | FF00 | $4.9090e^{-004}$ | 0.0023 | $1.7547e^{-008}$ | 99.9509 |
| 10 | FE00 | $9.7289e^{-004}$ | 0.0046 | $3.5095e^{-008}$ | 99.9027 |
| 11 | FC00 | $1.9291e^{-003}$ | 0.0091 | $6.9428e^{-008}$ | 99.8071 |
| 12 | F800 | $3.7911e^{-003}$ | 0.0183 | $1.3962e^{-007}$ | 99.6208 |
| 13 | F000 | $6.954\ e^{-003}$ | 0.0364 | $2.7771e^{-007}$ | 99.3046 |
| 14 | E000 | 0.0134 | 0.0724 | $5.5237e^{-007}$ | 98.6581 |
| 15 | C000 | 0.0255 | 0.1451 | $1.1070e^{-006}$ | 97.4482 |
| 16 | 8000 | 0.0469 | 0.2918 | $2.2261e^{-006}$ | 95.3078 |
| 17 | 7FFF | 0.0541 | 0.4374 | $3.3373e^{-006}$ | 94.5929 |
| 18 | 3FFF | 0.0725 | 0.5467 | $4.1711e^{-006}$ | 92.7466 |
| 19 | 1FFF | 0.0787 | 0.5739 | $4.3788e^{-006}$ | 92.1329 |
| 20 | 0FFF | 0.0807 | 0.5808 | $4.4315e^{-006}$ | 91.9340 |
| 21 | 07FF | 0.0813 | 0.5825 | $4.4440e^{-006}$ | 91.8746 |
| 22 | 03FF | 0.0814 | 0.5830 | $4.4478e^{-006}$ | 91.8552 |
| 23 | 01FF | 0.0815 | 0.5831 | $4.4487e^{-006}$ | 91.8499 |
| 24 | 00FF | 0.0816 | 0.5832 | $4.4489e^{-006}$ | 91.8487 |
| 25 | 007F | 0.0817 | 0.5833 | $4.4490e^{-006}$ | 91.8099 |
| 26 | 003F | 0.0818 | 0.5834 | $4.4491e^{-006}$ | 91.8192 |
| 27 | 001F | 0.0819 | 0.5835 | $4.4492e^{-006}$ | 91.8998 |
| 28 | 000F | 0.0820 | 0.5836 | $4.4493e^{-006}$ | 91.8000 |
| 29 | 0000 | 0.0822 | 0.5837 | $4.4497e^{-006}$ | 91.7880 |

## 5. CONCLUSION

This paper has provided an analysis of 16-bit variable accuracy reconfigurable adders by using the multiplexer based RCFA. The proposed VARA4 shows better performance in the power consumption and it provides multiple quality variable accuracies outputs up to 100% with respect to the existing conventional CSLA approach and other proposed adders in the image blending application. The proposed VARA4 offers a savings of 5.67% power and 0.23% power delay product with variable accuracy outputs from worst-case to 100% best-case for an input operand pair at the cost of area and speed than the existing conventional 16-bit CSLA. Hence the 16-bit VARA4 is more suitable for variable accuracy hybrid applications. The proposed VARA7 has less power delay product and less delay than the existing conventional CSLA. Therefore the VARA7 design is also suitable for low accuracy and conventional addition based hybrid applications.

## REFERENCES


**[1]** Mohanty BK and Patel SK. "Area–delay–power efficient carry-select adder", IEEE Trans Circuits Syst II 2014, Vol.61, no.6, pp.418–22, June.

**[2]** Ramkumar B , Kittur HM . "Low-power and area-efficient carry select adder", IEEE Trans Very Large Scale Integr (VLSI) Syst 2012, Vol.20, no.2, pp.371–375, February.

**[3]** Bedrij OJ . "Carry-select adder", IRE Trans Electron Comput 1962, EC-Vol.11, no.3, pp.340–346, June .

**[4]** Saeed V. and Vaseghi. "Multimedia Signal Processing: Theory and Applications in Speech, Music, and Communications", 2007, John Wiley & Sons Ltd, England.

**[5]** Keshab K. Parhi. "VLSI Digital Signal Processing Systems: Design and Implementation", 1998, John Wiley & Sons Ltd, England.

**[6]** Yadav, S & Mehra, R "Analysis of FPGA Based Recursive Filter Using Optimization Techniques for High Throughput", International Journal of Engineering and Advanced Technology, 2014, Vol. 3, No. 4, pp. 341-343.

**[7]** Varatkar, G & Shanbhag, N. "Energy-efficient motion estimation using error-tolerance", in Proceedings of the IEEE/ACM International Symposium on Low Power Electronics and Design, 2006, pp. 113–118.

**[8]** V. Gupta, D. Mohapatra, A. Raghunathan and K. Roy, "Low-Power Digital Signal Processing Using Approximate Adders," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 32, no. 1, pp. 124-137, Jan. 2013.

**[9]** Z. Yang, A. Jain, J. Liang, J. Han, and F. Lombardi, "Approximate XOR/XNOR-based adders for inexact computing", in IEEE-NANO, 2013, pp. 690–693.

**[10]** Bharat Garg, Sunil Dutt, G.K. Sharma, "Bit-width-aware constant-delay run-time Accuracy Programmable Adder for error-resilient applications", Microelectronics Journal, Vol. 50, 2016, pp. 1-7.

**[11]** Dharmendra Kumar, Debasis Mitra, "Design of a practical fault-tolerant adder in QCA", Microelectronics Journal, Vol. 53, 2016, pp. 90-104.

**[12]** Junjun Hu, Zhijing Li, Meng Yang, Zixin Huang, Weikang Qian, "A high-accuracy approximate adder with correct sign calculation", Integration, the VLSI Journal, 2017, doi.org/10.1016/j.vlsi.2017.09.003.

**[13]** Zhu, N, Goh, WL & Yeo, KS. "An enhanced low-power highspeed adder for error-tolerant application," in Proceedings of the IEEE International Symposium on Integrated Circuits, 2009, pp. 69–72.

**[14]** Sakthivel, R and Kittur, HM. "Energy Efficient Low Area Error Tolerant Adder with Higher Accuracy", Circuits, Systems, and Signal Processing, 2014, vol. 33(8), pp. 2625-2641.

**[15]** A. B. Kahng, S. Kang, "Accuracy-configurable adder for approximate arithmetic designs", Design Automation Conference (DAC), 2012, pp. 820-825.

**[16]** R. Ye, T. Wang, F. Yuan, R. Kumar, Q. Xu, "On reconfiguration oriented approximate adder design and its application", International Conference on Computer-Aided Design (ICCAD), 2013, pp. 48-54.

**[17]** M. Shafique, W. Ahmad, R. Hafiz and J. Henkel, "A low latency generic accuracy configurable adder," 2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC), San Francisco, CA, 2015, pp. 1-6.

**[18]** https://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/device-models/bsdl-models/spartan-series-fpgas.html.

**[19]** Cadence SOC Encounter. [Online]. Available: http://www.cadence.com.

**[20]** Fazel Sharifi, Atiyeh Panahi, Mohammad Hossein Moaiyeri, Hojjat Sharifi & Keivan Navi, "High Performance CNFET-based Ternary Full Adders," IETE Journal of Research, 64:1, 108-115, 2018, DOI: 10.1080/03772063.2017.1338973

**[21]** R. Jothin and C. Vasanthanayaki, "High Performance Error Tolerant Adders for Image Processing Applications," IETE Journal of Research, 2018, DOI: 10.1080/03772063.2018.1535920