

M. Tech (VLSI) DIVYA CHOUDHARY 2019

NEW REVERSE CARRY PROPAGATE ADDER USING MODIFIED GDI TECHNIQUE

A DISSERTATION REPORT

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE
OF

MASTER OF TECHNOLOGY
IN
VLSI AND EMBEDDED SYSTEMS

Submitted by:

DIVYA CHOUDHARY

2K17/VLS/09

Under the supervision of

PROF. D. R. BHASKAR



**ELECTRONICS AND COMMUNICATION
ENGINEERING**

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

JULY, 2019

NEW REVERSE CARRY PROPAGATE ADDER USING MODIFIED GDI TECHNIQUE

A MAJOR PROJECT REPORT

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE
OF

MASTER OF TECHNOLOGY
IN
VLSI AND EMBEDDED SYSTEMS

Submitted by:

DIVYA CHOUDHARY

2K17/VLS/09

Under the supervision of

PROF. D. R. BHASKAR



**ELECTRONICS AND COMMUNICATION
ENGINEERING**

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

JULY, 2019

ECE DEPARTMENT
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CANDIDATE’S DECLARATION

I, **DIVYA CHOUDHARY**, **2K17/VLS/09** student of M.Tech (VLSI), hereby declare that the Dissertation titled “**NEW REVERSE CARRY PROPAGATE ADDER USING MODIFIED GDI TECHNIQUE**” which is submitted by me to the Department of Electronics and Communication, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi

DIVYA CHOUDHARY

Date: 3rd JULY, 2019

ECE DEPARTMENT
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

CERTIFICATE

I hereby certify that the Dissertation titled “**NEW REVERSE CARRY PROPAGATE ADDER USING MODIFIED GDI TECHNIQUE**” which is submitted by **DIVYA CHOUDHARY, 2K17/VLS/09** [ECE Department], Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: Delhi

Date: 3rd JULY, 2019

D.R.BHASKAR

SUPERVISOR

PROFESSOR

Department of Electronics and Communication

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

ABSTRACT

Addition is the most important function in arithmetic and logical operations. Approximate Computing can be used to reduce the number of transistors, delay and power constraints in VLSI design, which makes the use of approximate adders possible in error-tolerant applications. Existing Approximate Reverse Carry Propagate Adder designs [1] have proved to be advantageous in improving these constraints. A new design of Reverse Carry Propagate Adder has been proposed using Modified-Gate Diffusion Input (GDI) technique [7]. A 4-bit Multiplier has also been designed using this RCPFA and results verified with Xilinx Tool. Proposed circuit design simulations have been carried out in 45-nm process technology using Cadence Virtuoso. The results indicate 57% and 44% reduction in Power and Delay respectively.

ACKNOWLEDGEMENT

A successful project can never be prepared by the efforts of the person to whom the project is assigned, but it also demands the help and guardianship of people who helped in the completion of the project.

I would like to thank all those people who have helped me in this research and inspired me during my study.

With profound sense of gratitude, I thank Prof. D. R. Bhaskar, my Research Supervisor, for his encouragement, support, patience and his guidance in this research work.

Furthermore I would also like to thank Prof. Rajeshwari Pandey, who gave me permission to use all required equipment and the necessary format to complete the report.

I take immense delight in extending my acknowledgement to my family and friends who have helped me throughout this research work.

DIVYA CHOUDHARY

CONTENTS

Candidate's Declaration	i
Certificate	ii
Abstract	iii
Acknowledgement	iv
Contents	v
List of Tables	vi
List of Figures	vii
List of Symbols, abbreviations	viii
CHAPTER 1 INTRODUCTION	1
CHAPTER 2 LITERATURE REVIEW	3
2.1 Approximate Mirror Adder	3
2.2 Lower Part OR Adder	6
2.3 Approximate XOR based Adder	7
2.3.1 AXA-1	7
2.3.2 AXA-2	8
2.3.3 AXA-3	9
2.4 Transmission Gate Approximate Adder	10
2.4.1 TGA-1	10
2.4.2 TGA-2	11
CHAPTER 3 RCPFA APPROXIMATE ADDER	13
3.1 Existing Reverse Carry Propagate Adder	14
3.1.1 Internal Structure of R.C.P.F.A.	16
3.1.2 RCPFA-1	17
3.1.3 RCPFA-2	20
3.1.4 RCPFA-3	23

CHAPTER 4 PROPOSED RCPFAs	27
4.1 Modified GDI Technique	27
4.1.1 Miscellaneous Functions of Modified GDI Cell	28
4.2 Proposed RCPFA-1 Circuit	28
4.3 Proposed RCPFA-2 Circuit	30
4.4 Proposed RCPFA-3 Circuit	32
4.5 COMPARISON WITH RCPFAs[1]	34
4.6 32-bit Hybrid Adder	35
CHAPTER 5 MULTIPLIER	37
5.1 Ripple-Carry Adder using RCPFA	37
5.2 Multiplier using RCPFA	38
CHAPTER 6 VERILOG CODES AND SIMULATION RESULTS	40
6.1 RCPFA-2	40
6.1.1 Verilog code for RCPFA-2	40
6.1.2 Test bench for RCPFA-2	41
6.1.3 Simulation Result	41
6.2 RCPA	41
6.2.1 Verilog code for RCPA	41
6.2.2 Test bench for RCPA	41
6.2.3 Simulation Result	43
6.3 4-bit Multiplier	43
6.3.1 Verilog code for 4×4 Multiplier	44
6.3.2 Test bench	46
6.3.3 Simulation Result	46
CONCLUSION	47
REFERENCES	48
RESEARCH PUBLICATION	49

LIST OF TABLES

TABLE I	Truth Table of RCPFA-1
TABLE II	Truth Table of RCPFA-2
TABLE III	Truth Table of RCPFA-3
TABLE IV	Miscellaneous Functions of a Modified GDI Cell
TABLE V	Comparison Between Proposed and Conventional RCPFAs

LIST OF FIGURES

Fig. 2.1	Approximate mirror adder (AMA-1)
Fig. 2.2	Approximate mirror adder (AMA-2)
Fig. 2.3	Approximate mirror adder (AMA-3)
Fig. 2.4	Lower Part OR Adder
Fig. 2.5	Approximate XOR based Adder type-1
Fig. 2.6	Approximate XOR based Adder type-2
Fig. 2.7	Approximate XOR based Adder type-3
Fig. 2.8	Approximate TGA-1 Adder.
Fig. 2.9	Approximate TGA-2 Adder.
Fig. 3.1	RCPFA cell
Fig. 3.2	n-bit RCPA.
Fig. 3.3	Karnaugh maps for Sum (S_i) and carry (C_i) of general form of RCPFA
Fig. 3.4	RCPFA-1
Fig. 3.5	Schematic for RCPFA-1
Fig. 3.6	Transient response for RCPFA-1
Fig. 3.7	Power plot for RCPFA-1
Fig. 3.8	RCPFA-2
Fig. 3.9	Schematic for RCPFA-2
Fig. 3.10	Transient response for RCPFA-2
Fig. 3.11	RCPFA-3
Fig. 3.12	Schematic for RCPFA-3
Fig. 3.13	Transient response for RCPFA-3
Fig. 3.14	Normalized MED of the approximate adders for different bit lengths.
Fig. 3.15	Normalized MED of the approximate adders for different bit lengths.
Fig. 3.16	Normalized MED of the approximate adders for different bit lengths.
Fig. 4.1	Modified GDI cell

Fig. 4.2	Schematic of Proposed RCPFA-1
Fig. 4.3	Transient response of proposed RCPFA-1.
Fig. 4.4	Power plot of Proposed RCPFA-1
Fig. 4.5	Schematic of Proposed RCPFA-2
Fig. 4.6	Transient response of proposed RCPFA-2.
Fig. 4.7	Power plot of Proposed RCPFA-2
Fig. 4.8	Schematic of Proposed RCPFA-3
Fig. 4.9	Transient response of proposed RCPFA-3
Fig. 4.10	Power plot of Proposed RCPFA-3
Fig. 4.11	Schematic of 32-bit hybrid adder using Proposed RCPFA
Fig. 4.12	n-bit hybrid adder with k-bit RCPFA part
Fig. 4.13	Delay of the 32bit hybrid adder Vs the approximate part width (k)
Fig. 5.1	Ripple-Carry adder using RCPFA Cell
Fig. 5.2	4-bit binary Multiplier
Fig. 6.1	Simulation Result for RCPFA
Fig. 6.2	Simulation Result for RCPA
Fig. 6.3	Simulation Result for 4-bit Multiplier

LIST OF SYMBOLS, ABBREVIATIONS

RCPFA	Reverse Carry Propagate Full Adder
TG	Transmission Gate
ER	Error Rate
AMA	Approximate Mirror Adder
AXA	Approximate XOR/XNOR Adder
TGA	Transmission Gate Adder
CMOS	Complementary Metal Oxide Semiconductor

CHAPTER 1

INTRODUCTION

With the advancement in technology, the number of transistors in an integrated circuit is increasing and devices are getting more compact day by day. Due to which, the demand for more compact implementation of circuits is increasing. Since adder is the main logical block of arithmetic and logic unit, many researchers have tried to obtain a circuit which can perform the same functionality with fewer number of transistors. One of the computation techniques that can be employed in this area of interest, to implement circuits with lesser hardware is Approximate Computing[2].

The increase in the number of transistors mounted on a chip is accompanied by the demand of Power reduction. Over the years, Approximate computing has managed to be a promising method to decrease power, area and delay constraints in Physical design. But it also results in a loss of computational accuracy. Power can be spared by upgrading the module into an approximate version. It is required to examine the different logic functionalities of the circuit when designing the inexact version of the module. As one of the elementary, but crucial components of arithmetic circuits, adders have pulled in a broad enthusiasm for upgrading and implementing approximate modules. Many Approximate adder circuits have been proposed by utilizing less number of transistors. In order to do so, the carry propagation chain has also been truncated for operations based on speculation. The approximate designs have proved to be better in achieving a superior performance in terms of area, power and delay compared to traditional (exact) adders.

Since adder is the main digital circuit of arithmetic and logic unit, many researchers have presented their research works [3]-[6] on optimizing speed and power of adder by using Approximate Computing.

Approximate adders have gained importance for utilization in circuits where some of error is tolerable. In [3] the approximate mirror adders (AMAs) have been proposed which save power by utilizing less no. of transistors connected in a mirror adder design. Another approximate adder lower part OR Adder (LOA) consists of two sub-adder sections performing different functions and calculating the approximate results[4]. In [5] XOR/XNOR gates based adders have been proposed implemented by pass transistors which provide a significant decrease in power with better performance.[6] utilizes transmission gates in place of pass transistors which proves to be a better replacement as unlike pass transistors , they do not suffer from signal degradation.

The use of the Approximate-Reverse-Carry-Propagate-Adder (R.C.P.A) is presented in [1], which propagates the carry in reverse direction, thereby decreasing the carry weight as it propagates making the adder less sensitive to delay variations as compared to traditional adders.

Hence, the main purpose of this dissertation is to present a new circuit for Reverse-Carry-Propagate-Adder using Modified GDI Technique.

CHAPTER 2

LITERATURE REVIEW

In the approximate adder modules; the basic idea is to divide the multiple bit adder into two sub sections : the (precise) upper-section for manipulating higher significant bits and the (inexact) lower-section for lower significant bits. For every bit in the lower sub-module, one-bit approximate full adder implements an inexact function, thus a modification of the addition operation. This is generally achieved by modifying the Full-Adder circuit design either at device level or at circuit level, which is comparable to changing some of the output values in the full adder truth table at the functional level.

Some of the common approximate adders presented by numerous researchers are as follows:

2.1 Approximate Mirror Adder

One of the widely used accurate implementation of full adder is mirror adder. The Approximate-Mirror-Adder (AMA) can be obtained from this design by reduction of transistors. By removing some transistors, three approximate mirror adder designs can be obtained which have been presented in [3]. The circuits shown in Fig.2.1 to 2.3 are capable of achieving lower power dissipation and reduced circuit complexity. Due

to less transistors, the node capacitances can get charged and discharged at a faster rate, thus experiences a shorter delay. A step-by-step procedure is being followed in coming up with several inexact mirror adder cells with less transistors. One by one the transistors are removed from the conventional schematic. However, these can not be removed in an arbitrary fashion. It is to be made sure that not even a single combination of inputs A, B and Ci results in short-circuit or open-circuit in the modified schematic of the adder. Another significant point to be kept in mind is that the resulting simplified schematic should introduce a minimal amount of error in the truth table of full adder.

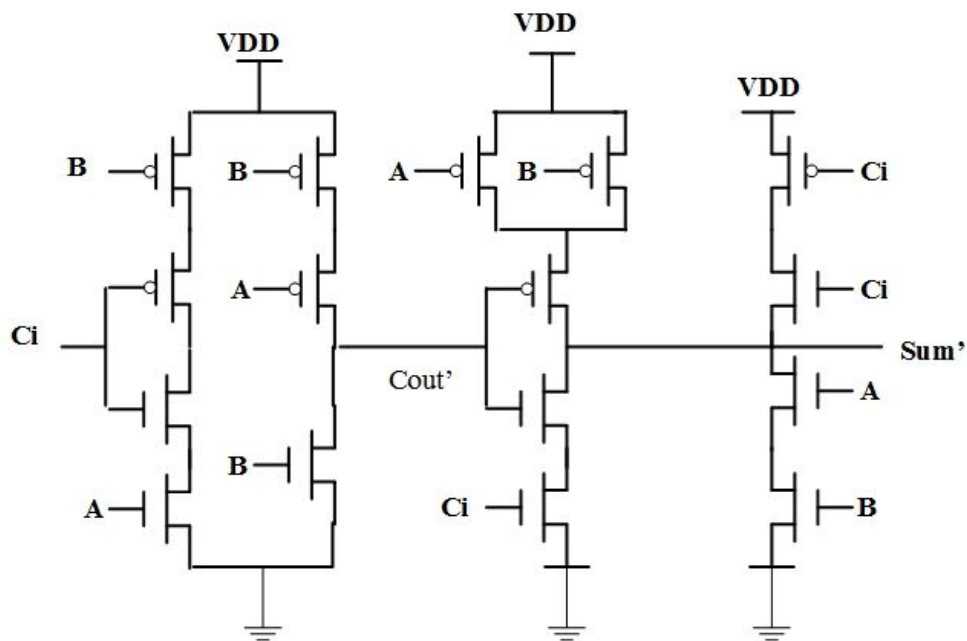


Fig. 2.1 Approximate mirror adder (AMA-1)

If some of the series connected transistors are removed, it will encourage quick charging and discharging of the node capacitances. In addition, the removal of transistors not only decreases circuit complexity but also helps in reducing the αC term which exists due to switched capacitances in the expression for dynamic power

$$P_{\text{dynamic}} = \alpha C V_{DD}^2 f \quad (2.1)$$

where α is the switching activity or it can be defined as the probability of the output node having a transition from zero to one and C is the output load capacitance.

Reduction in this term decreases the dynamic power dissipation. This also reduces the total area on the chip.

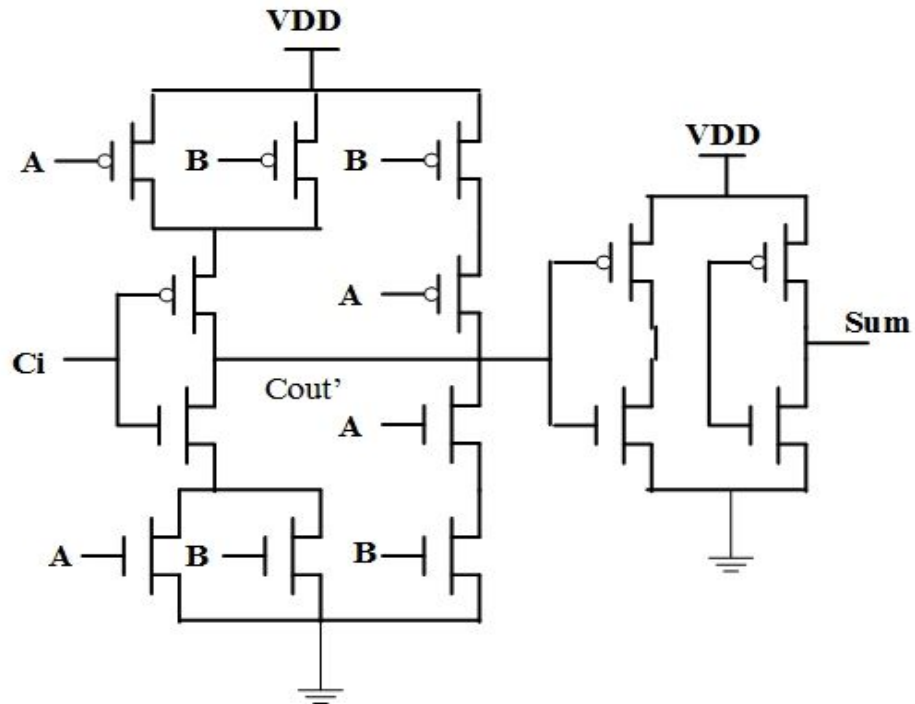


Fig. 2.2 Approximate mirror adder (AMA-2)

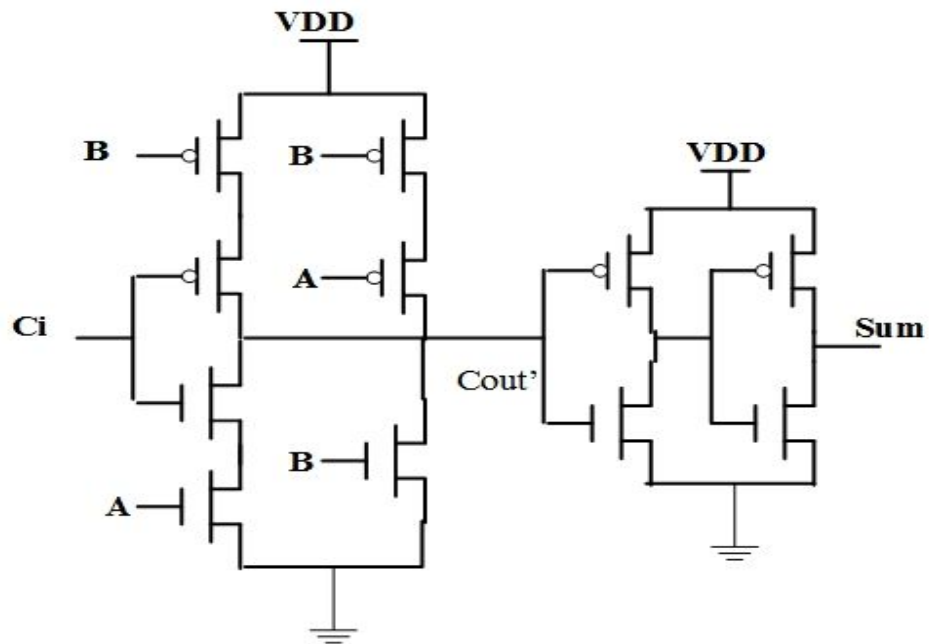


Fig. 2.3 Approximate mirror adder (AMA-3)

2.2 Lower Part OR Adder

The second type of approximate adder is Lower-Part-OR-Adder(LOA)[4]. In LOA, OR gates (also referred to as the lower section) have been utilized to approximately manipulate the less significant bits.

“Lower-Part-OR Adder” Structure: Addition is a fundamental operation which can significantly impact the overall achievable performances. The Lower part OR Adder(LOA) breaks a q-bit number into two smaller r and s bit numbers (where, $r+s=q$).

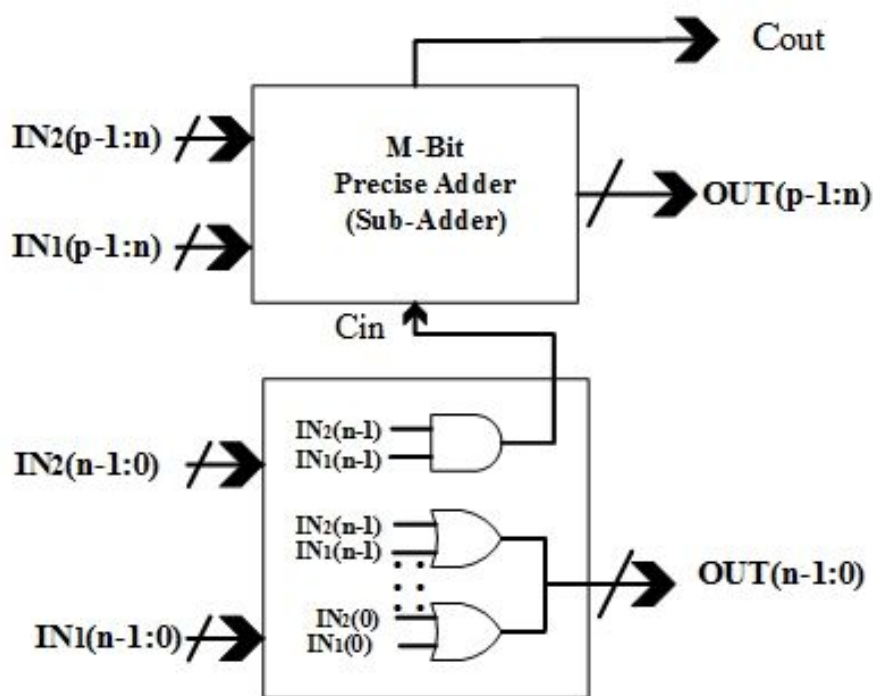


Fig. 2.4 Lower Part OR Adder

As shown in the Fig.2.4, a q-bit LOA makes use of regular precise adders (called sub-adder) that computes the accurate values of the ‘r’ most significant bits

(upper-section) of the result along with the OR gates that provide the approximate values of the 's' least significant bits (lower-section) by applying bitwise OR operation to the respective bits of the input. An extra AND gate is required for the generation of input carry for more significant bits. This AND gate produces a carry when both the inputs to the most significant full adder in the lower section of the module are '1'. This is the main reason why the module lacks some degree of precision since the input carry is calculated to be zero for most of the cases and thus discarded in the lower section of the adder module. Only the trivial carries are taken into account from the lower section to the upper section of the LOA adder, which results in its imprecision.

2.3 Approximate XOR based Adder

2.3.1 AXA-1

Fig.2.5 shows Approximate XOR based Adder type 1. In the type 1 schematic, the XOR operation is carried out by two pass transistors and an inverter connected to inputs X and Y respectively.

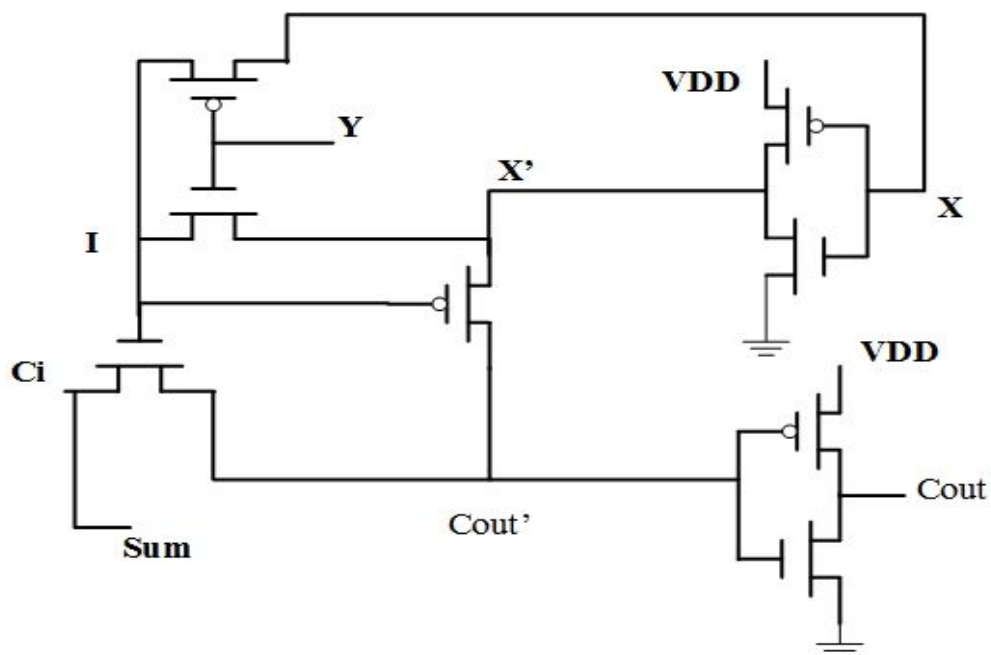


Fig. 2.5 Approximate XOR based Adder type-1

When the input Y equals '1', I becomes equal to complement of input X and when Y is '0', I equals to X i.e., $I=X\oplus Y$. The Sum and Cout can be calculated precisely only for 4 out of 8 different combinations of the inputs. Thus the total error distance that can be achieved with this circuit design is 4. The total transistor count is 8. The expression for Sum and Cout are given by:

$$\text{Sum} = \text{Cin}; \quad (2.2)$$

$$\text{Cout} = (X\oplus Y)\text{Cin} + \overline{Y}X \quad (2.3)$$

2.3.2 AXA-2

The circuit design in Fig.2.6 shows the implementation of the Approximate XOR based adder Type-2.

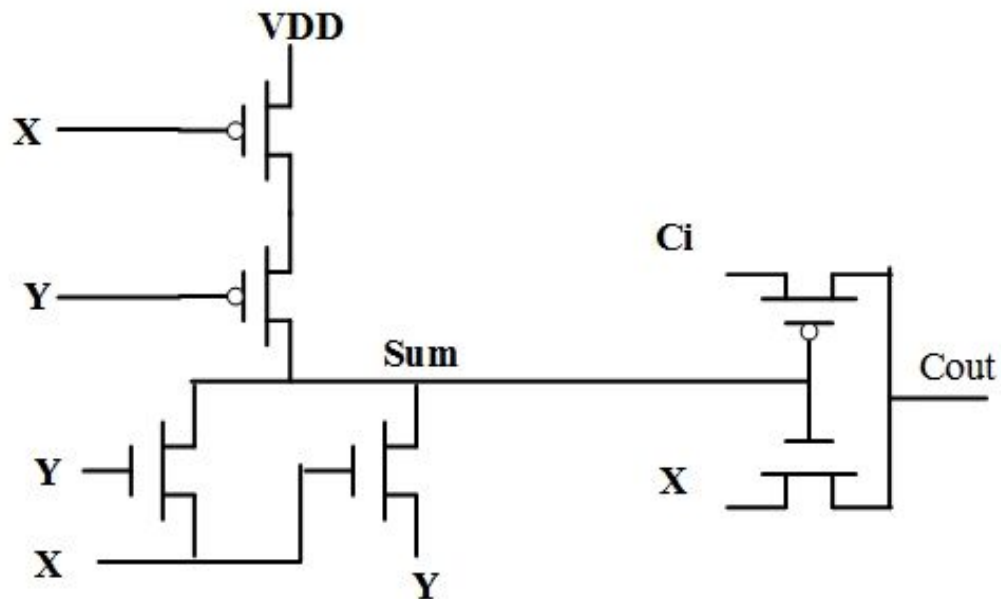


Fig. 2.6 Approximate XOR based Adder type-2

AXA-2 can be implemented with the help of only 6 transistors, which utilizes one pass transistor block and 4 transistors for the XNOR operation. Sum is accurate for only 4 input combinations out of the 8, while Cout can be calculated accurately for all input combinations. For this circuit design also, the total error distance that can be achieved is 4. The expressions for Sum and Cout are:

$$\text{Sum} = \overline{(X \oplus Y)} \quad (2.4)$$

$$\text{Cout} = (X \oplus Y) \text{Cin} + XY \quad (2.5)$$

As the pass transistors suffer from the problem of signal degradation due to threshold voltage drop, the output signals for Sum and Cout do not achieve a full swing for some transitions.

2.3.3 AXA-3

The circuit design in Fig. 2.7 is an extension of the circuit used in AXA-2. In order to achieve better accuracy for the Sum calculations two more transistors are incorporated in pass transistor configuration. Total number of transistors required for the circuit design is 8, out of which 4 are used for performing the XNOR operation. As a result of extra two gates used, the Sum can be computed correctly for 6 input combinations out of 8 and Cout was already accurate for all possible combinations. The circuit of AXA-3 provides a total error distance of 2.

The expressions for Sum and Cout are:

$$\text{Sum} = \overline{(X \oplus Y)} \text{Cin} \quad (2.6)$$

$$\text{Cout} = (X \oplus Y) \text{Cin} + XY \quad (2.7)$$

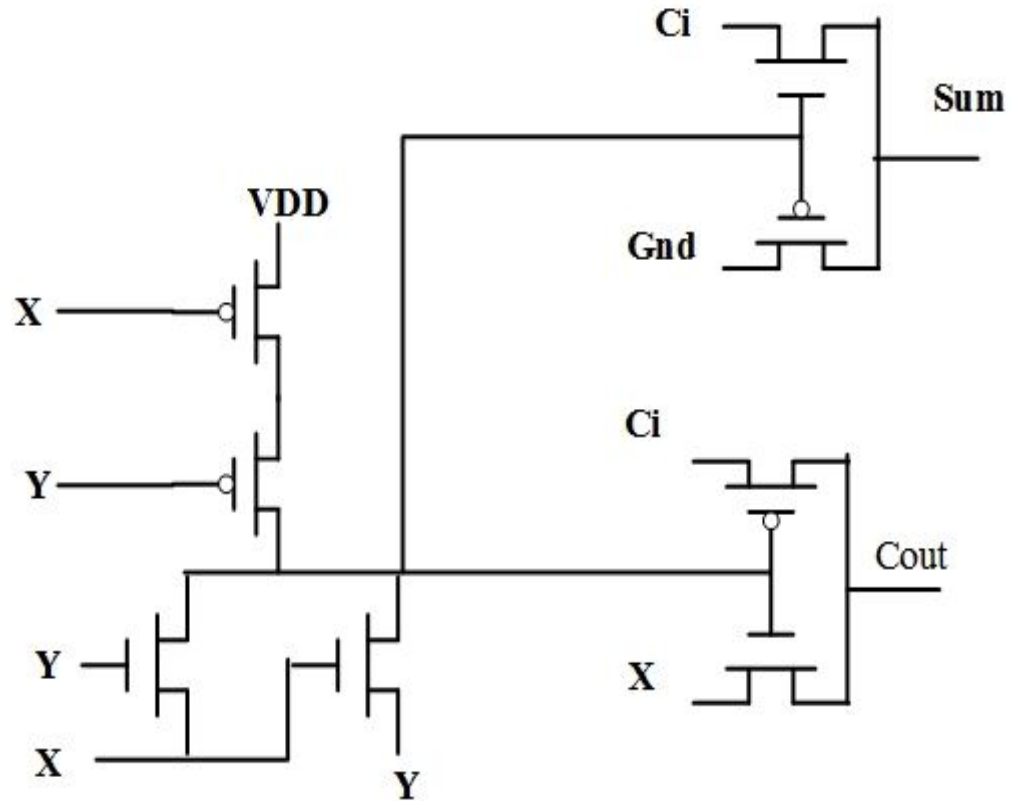


Fig. 2.7 Approximate XOR based Adder type-3

2.4 Transmission Gate Approximate Adders

In these approximate adders the transmission gates have been used in place of pass transistors as they suffer from the problem of signal distortion or degradation. Two new multiplexer based designs have been presented.

2.4.1 Approximate Adder (TGA-1)

It consists of transmission gate based multiplexer which in total uses 16 transistors. This generates two incorrect results out of 8, therefore the ER is two.

Fig.2.8 shows TGA-1 Adder. The logical expressions for Sum and Carry are:

$$\text{Sum} = \overline{(X \oplus Y)} C_{in} + X \bar{Y} \quad (2.8)$$

$$C_{out} = Y \quad (2.9)$$

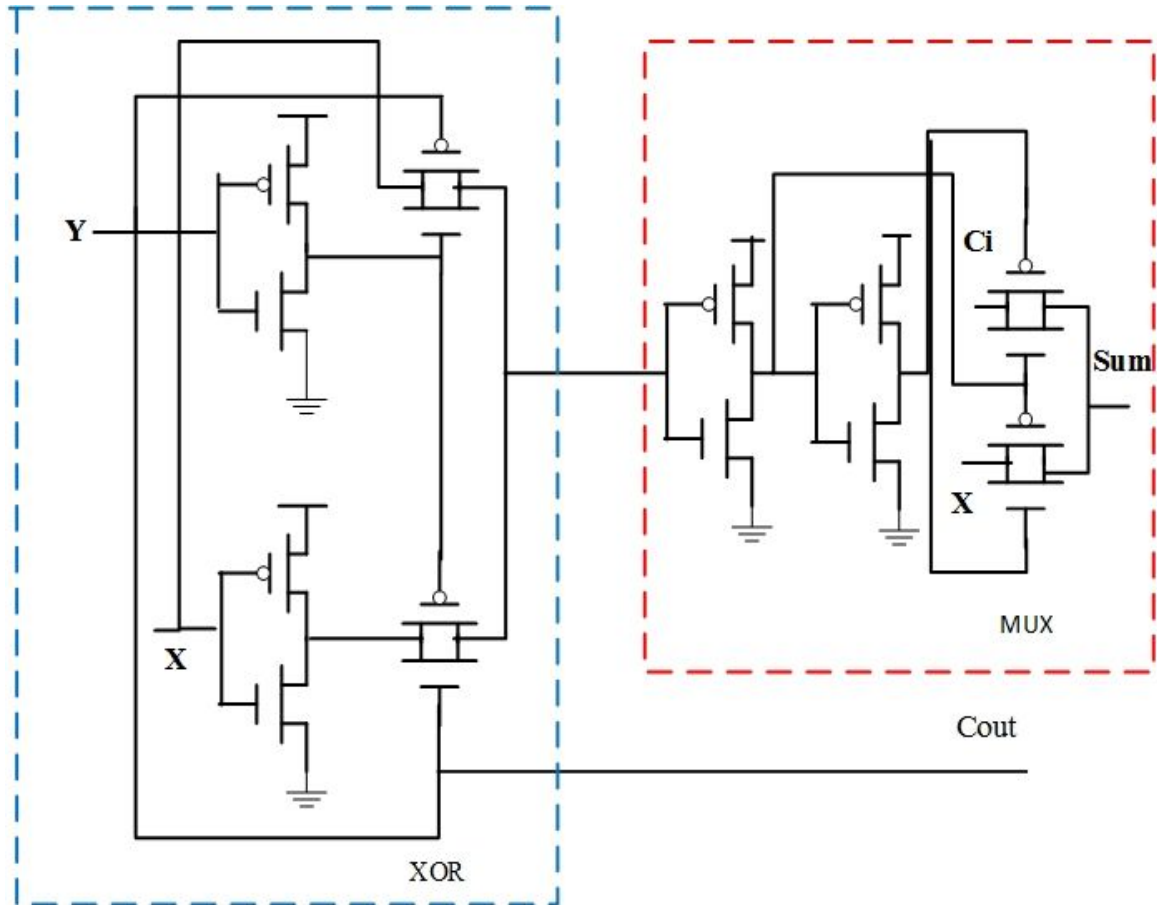


Fig. 2.8 Approximate TGA-1 Adder.

2.4.2 Approximate Adder (TGA-2)

As shown in Fig.2.9, the structure of adder comprises of 22 transistors and the logic functions for the sum and carry out are :

$$\text{Sum} = \overline{(X \oplus Y)} C_{in} \quad (2.10)$$

$$C_{out} = X + Y \quad (2.11)$$

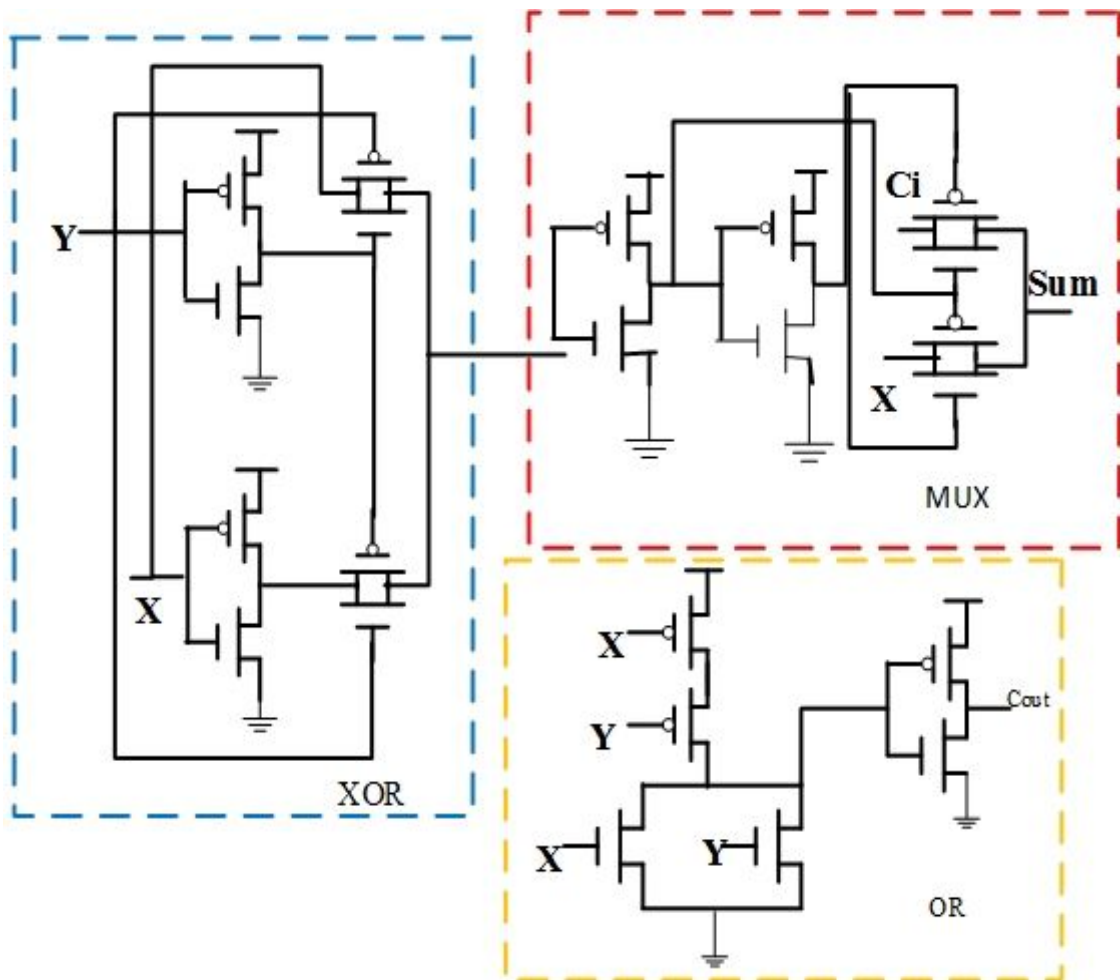


Fig. 2.9 Approximate TGA-2 Adder.

CHAPTER 3

RCPFA APPROXIMATE ADDERS

The conventional full adder has three input bits (namely the two input bits and an input carry) and two output bits (namely the sum-bit and carry-out bit). The sum-bit having the same weight as the input bits and carry-out bit with double the weight of the inputs. The carry-propagation delay is an important timing parameter in the full adder design owing to the fact that it is utilized in determining the critical path delay of multiple bit adders and multipliers.

In the most pessimistic scenario, the carry-propagation delay of the n-bit adder can be $n \times t_{cp}$, where n is the width of the adder. Now, if the clock period is smaller than $n \times t_{cp}$, then it can result in a set-up time violation which is a potential error. Even a very small violation in the timing can result in large amount of error if the error occurs in the most significant bits of the summation. This is due to the fact that the input carry for the most significant adder has to propagate through the less significant bit full adders. Therefore, if the carry is propagated in reverse direction, the total error introduced due to set-up and hold time violations will be reduced. This has motivated the researchers to design approximate adders where the carry can be propagated in reverse direction (counter-flow manner).

3.1 EXISTING REVERSE CARRY PROPAGATE FULL ADDER

The Exact Full Adder generates its Sum and Carry-out signal using equation (3.1)

$$A_i + B_i + C_i = S_i + 2 \times C_{i+1} \quad (3.1)$$

Where A_i and B_i are the i^{th} bits of the inputs A and B respectively, C_i and C_{i+1} are the carry-in and carry-out and S_i is the i^{th} bit of sum S. This equation shows how the output bits in the i^{th} position are related to the i^{th} bits of inputs A and B and to the carry-out of the previous adder (C_i). By adjusting the terms, i.e. moving C_i to the right hand side and C_{i+1} to the left hand side of the equation, one may write:

$$A_i + B_i - 2 \times C_{i+1} = S_i - C_i \quad (3.1)$$

Considering equation (3.1), the full adder can be thought as a structure whose operation depends on the output carry of the $(i+1)^{\text{th}}$ bit (C_{i+1}) and the input bits of the operands (A_i and B_i). The outputs for this modified structure will now be the sum and carry signals having same weights. One should notice that the input carry to the full adder in i^{th} bit position will be generated by full adder in $(i+1)^{\text{th}}$ bit position.

Depending on the input bits (A_i and B_i), the range for left hand side of equation (3.1) i.e. $S_i - C_i$ will be from set $\{-2, -1, 0, 1, 2\}$. While depending on the weights of output signals, the right hand side of the equation (3.1) can have values from set $\{-1, 0, 1\}$. This shows that the output will not be correct if left hand side resulted in set $\{-2, 2\}$. When right hand side of equation (3.1) becomes zero, (S, C) can take either (0, 0) or (1, 1) set as values. The forecast signal (F_i) is used to select one of the two sets. Forecast signal depends on the $(i-1)^{\text{th}}$ bit of the inputs. Based on the discussion given above, a family of RCPFAs has been suggested in [1]. Fig.3.1 shows a symbolic notation for RCPFA cell with four inputs (A_i , B_i , C_{i+1} and F_i) and three outputs (S_i , C_i , F_{i+1}).

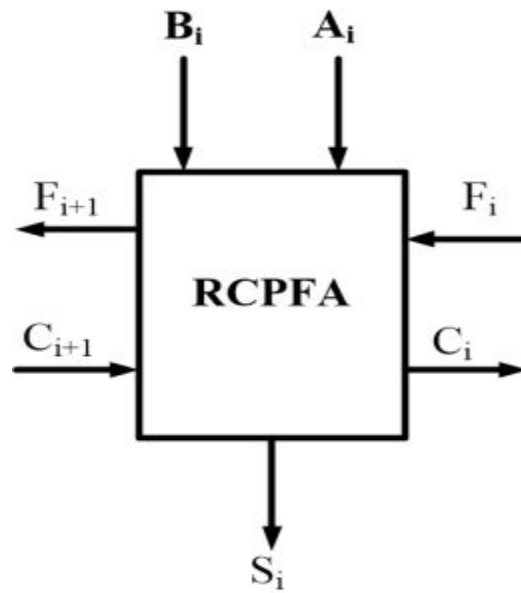


Fig.3.1. RCPFA cell

Fig.3.1 shows an n-bit RPCFA, where most significant carry C_n is taken equal to the forecast signal F_i of the most significant bit (MSB) RCPFA. Also, the input carry C_0 is used as forecast signal of least significant bit (LSB) RCPFA. The critical path for the n-bit RPCFA is also shown in the Fig.3.2.

Since, the error value increases in the direction of decrease of bit significance, the cumulative impact due to error (because of delay variation) is lower for higher significant bits.

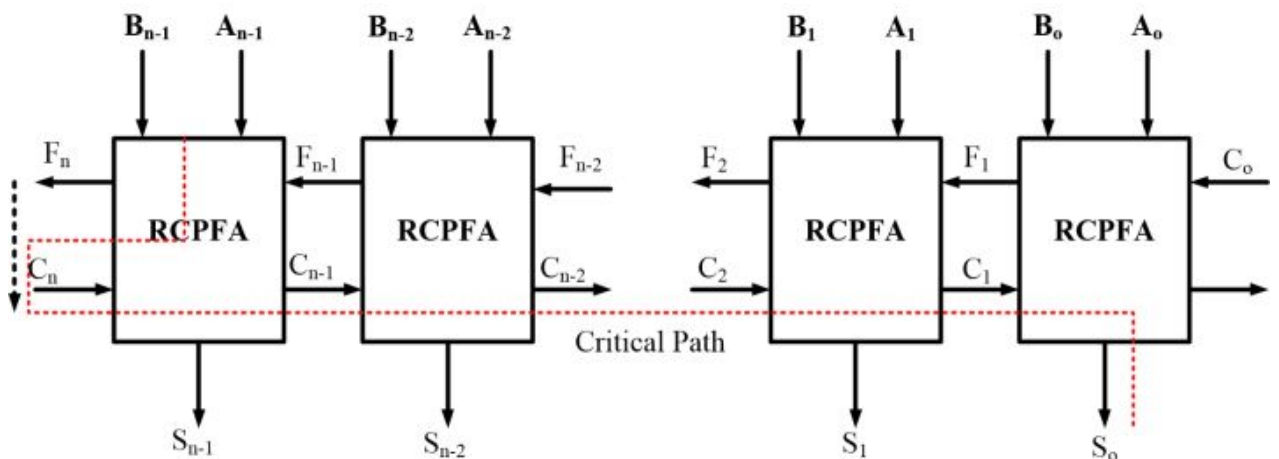


Figure.3.2 n-bit RPCFA.

3.1.1 Internal Structure of R.C.P.F.A.

To determine R.C.P.F.A. structure, Karnaugh maps for the Sum and carry are drawn on the basis of equation (3.2) derived above and considering forecast signal as one of the inputs to the circuit.

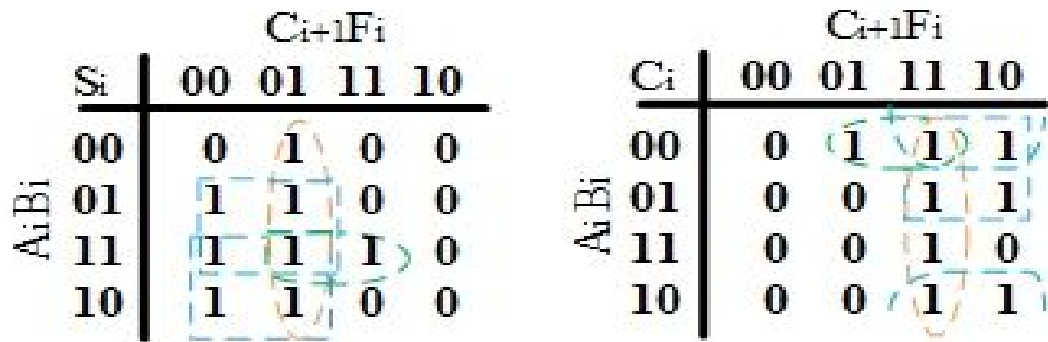


Fig.3.3 Karnaugh maps for Sum (S_i) and carry (C_i) of general form of R.C.P.F.A.

The Boolean relation between the various inputs for obtaining S_i and C_i are calculated as:

$$S_i = \overline{C_{i+1}} F_i + \overline{C_{i+1}} A_i + \overline{C_{i+1}} B_i + A_i B_i F_i \quad (3.3)$$

$$C_i = C_{i+1} F_i + C_{i+1} \overline{A_i} + C_{i+1} \overline{B_i} + \overline{A_i B_i} F_i. \quad (3.4)$$

An optimized gate level schematic can be obtained for implementing R.C.P.F.A. by simplifying equation (3) and (4) as

$$\begin{aligned} S_i &= F_i(\overline{C_{i+1}} + A_i B_i) + \overline{C_{i+1}}(A_i + B_i) \\ &= F_i \overline{X_i} + \overline{Y_i} \end{aligned} \quad (3.5)$$

$$\begin{aligned} C_i &= F_i(\overline{C_{i+1}}(A_i + B_i)) + (\overline{C_{i+1}} + A_i B_i) \\ &= F_i Y_i + X_i. \end{aligned} \quad (3.6)$$

For this approximate structure of the adder, the accuracy and the functioning of this schematic depends on the forecast signal F_i whose generation can lead to some overheads. So, in order to gain an optimized structure for R.C.P.F.A., the generation of this signal F_i should be simplified. There can be three different mechanisms for the generation of the forecast signal F_i based on which There are three different generation mechanisms for signal F based on which three types of Approximate-Reverse-Carry-Propagate-Adder are given :-

3.1.2 RCPFA-1

In the first R.C.P.F.A. structure (R.C.P.F.A-1) shown in Fig.3.4, which is directly obtained from equations (3.5) and (3.6), the forecast signal is taken equal to one of the inputs (A_i or B_i) e.g., F_{i+1} is taken equal to A_i . This schematic uses 26 transistors in total which is less than the number of transistors used in conventional full adder. The schematic, transient response and D.C. Power plot is shown in Fig.3.5 to 3.7. Table I shows the truth table for this adder circuit.

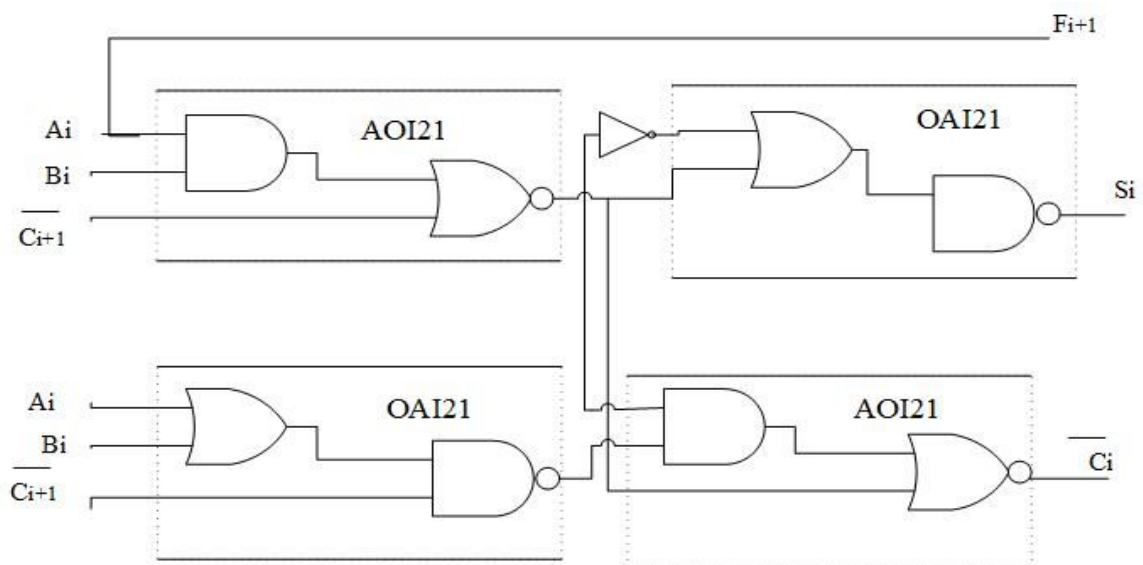


Fig.3.4. RCPFA-1.

TABLE I Truth Table of RCPFA-1[1]

ITEM	A _i	B _i	C _{i+1}	F _i	S _i	C _i	F _{i+1}	X _i	Y _i
1	0	0	0	0	0	0	0	0	1
2	0	0	0	1	1	1	0	0	1
3	0	0	1	x	0	1	0	1	1
4	0	1	0	x	1	0	0	0	0
5	0	1	1	x	0	1	0	1	1
6	1	0	0	x	1	0	1	0	0
7	1	0	1	x	0	1	1	1	1
8	1	1	0	x	1	0	1	0	0
9	1	1	1	0	0	0	1	0	0
10	1	1	1	1	1	1	1	0	1

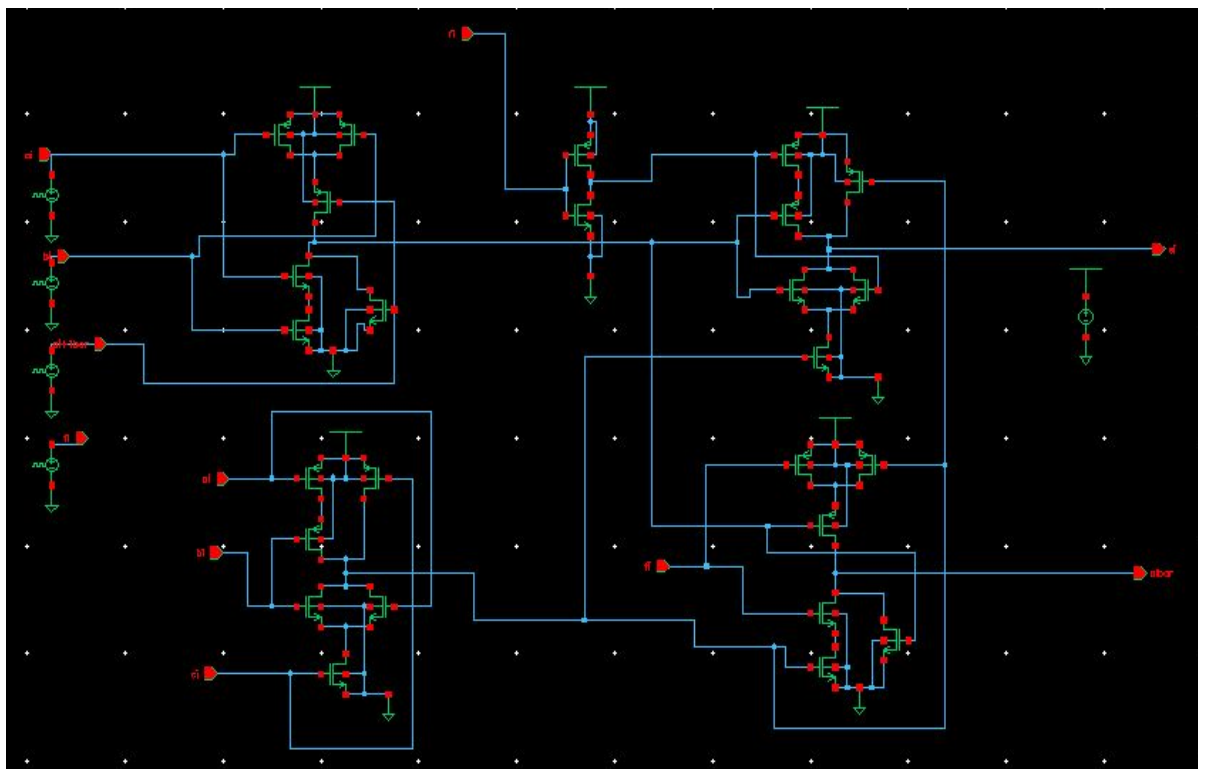


Fig.3.5. Schematic for RCPFA-1.

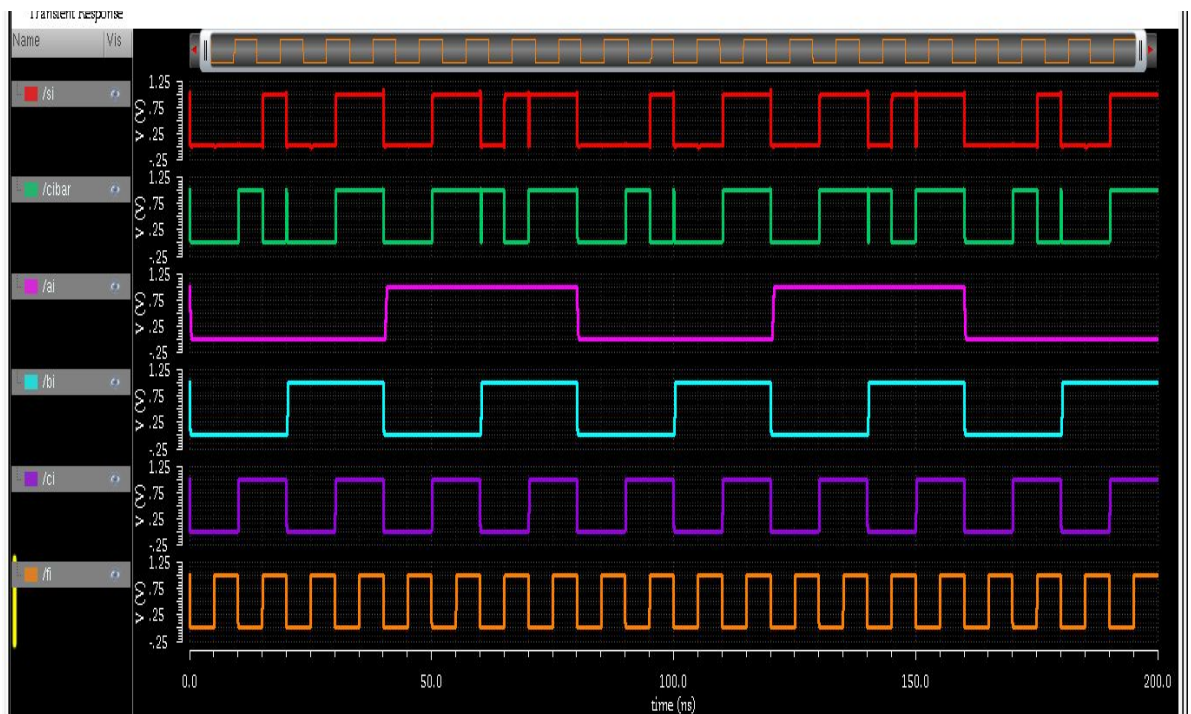


Fig.3.6. Transient response for RCPFA-1.

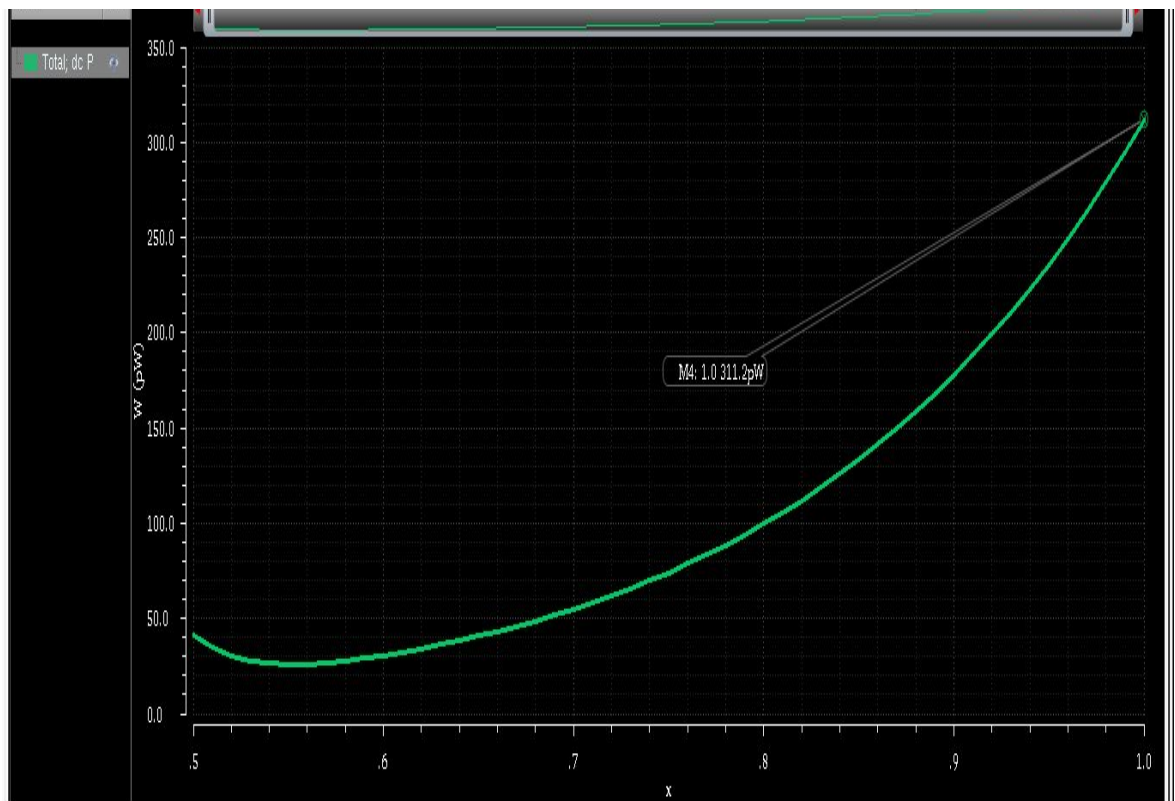


Fig.3.7. Power plot for RCPFA-1.

3.1.3 RCPFA-2

In the second approximate type adder (R.C.P.F.A-2), the carry-generate signal (A_i and B_i) is taken as the forecast input F_i . When F_i is the carry-generate signal, some of the states of Table I, where $X_i=1$ does not occur. Therefore by replacing X_i with zero, the general schematic of the Adder is simplified. This is shown in Fig.3.8 [4 gates from Fig.3.4 have been removed]. Thus by simplifying the structure, the number of transistors have been reduced from 26 to 20 . Out of 20 transistors, four are used in generating the forecast signal. The schematic, transient response and D.C. Power plot is shown in Fig.3.9 to 3.11. Table II shows the truth table for this adder circuit.

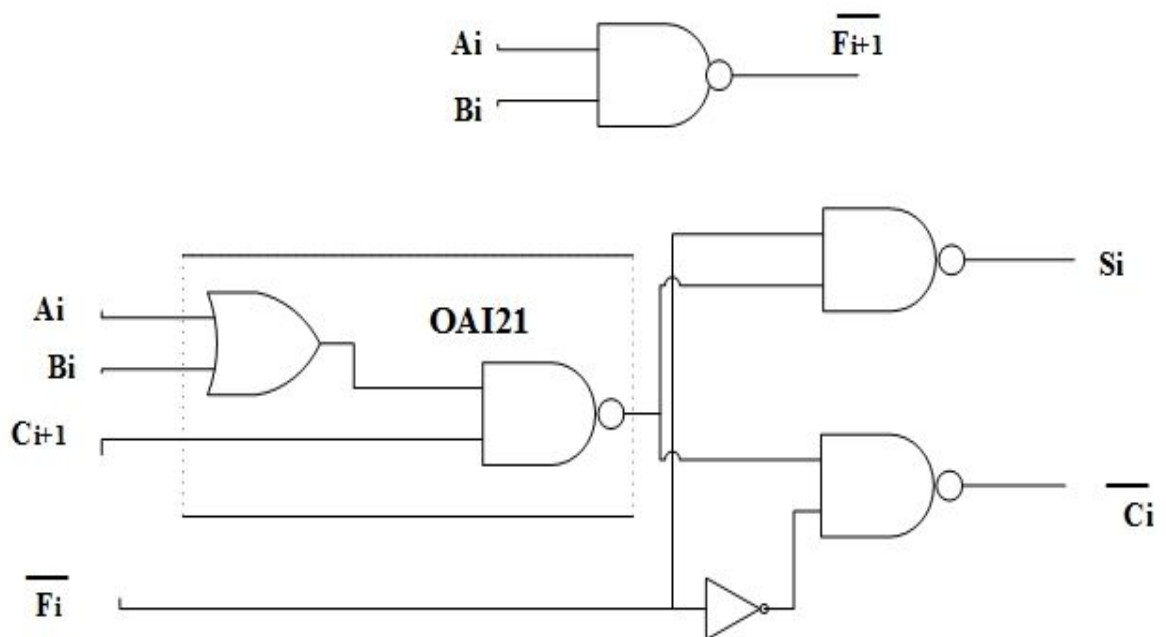


Fig.3.8 RCPFA-2.

TABLE II Truth Table of RCPFA-2[1]

ITEM	A _i	B _i	C _{i+1}	F _i	S _i	C _i	F _{i+1}
1	0	0	x	0	0	0	0
2	0	0	x	1	1	1	0
3	0	1	0	x	1	0	0
4	0	1	1	0	0	0	0
5	0	1	1	1	1	1	0
6	1	0	0	x	1	0	0
7	1	0	1	0	0	0	0
8	1	0	1	1	1	1	0
9	1	1	0	x	1	0	1
10	1	1	1	0	0	0	1
11	1	1	1	1	1	1	1

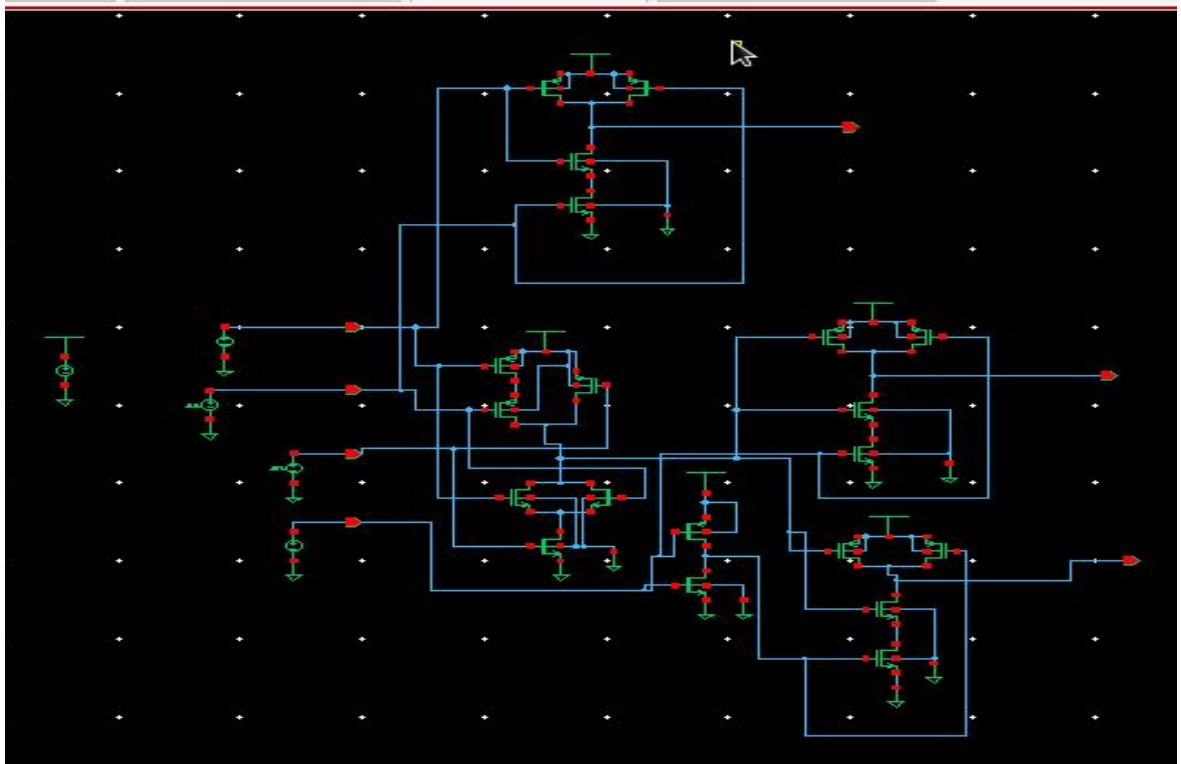


Fig.3.9. Schematic of RCPFA-2.

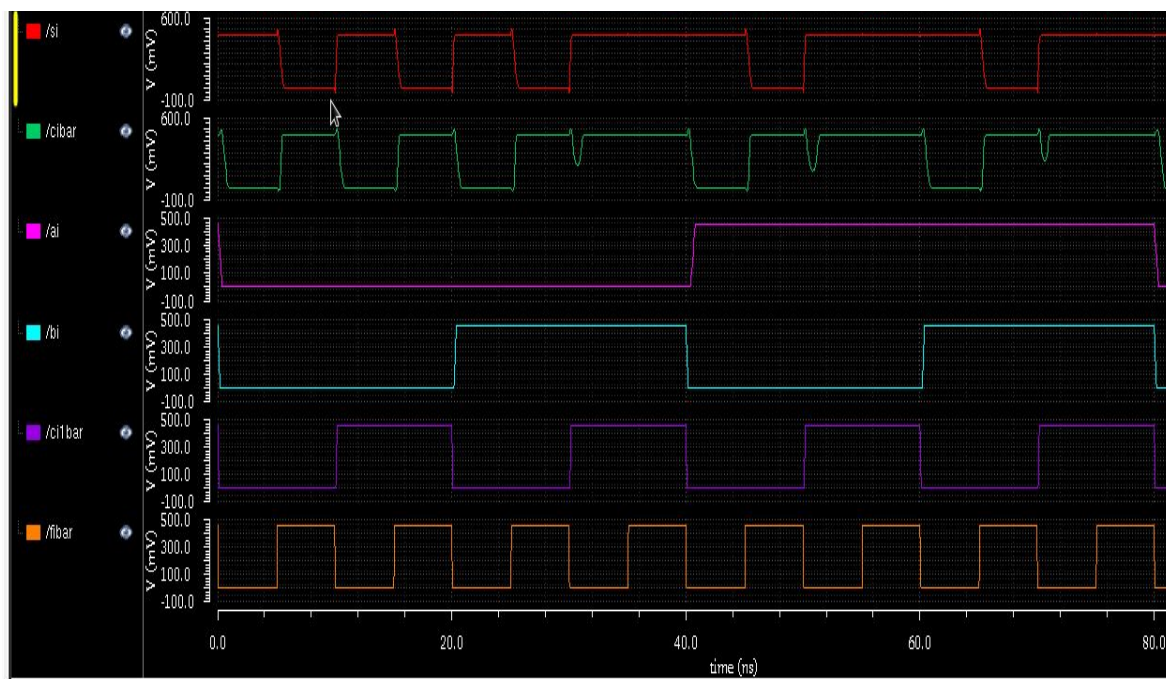


Fig.3.10 Transient response of RCPFA-2



Fig.3.12 Power Plot of RCPFA-2.

3.1.4 RCPFA-3

In the third approximate type adder (R.C.P.F.A-3), the carry-alive signal (A_i OR B_i) is taken as the forecast input F_i . When F_i is the carry-alive signal, some of the states of Table I, where $Y_i=0$ does not occur. Therefore by replacing Y_i with one, the general schematic of the Adder is simplified. This is shown in Fig.3.13 [4 gates from Fig.3.5 have been removed]. Thus by simplifying the structure, the number of transistors have been reduced from 26 to 20 . Out of 20 transistors, four are used in generating the forecast signal. The schematic, transient response and D.C. Power plot is shown in Fig.3.14 to 3.16. Table III shows the truth table for this adder circuit.

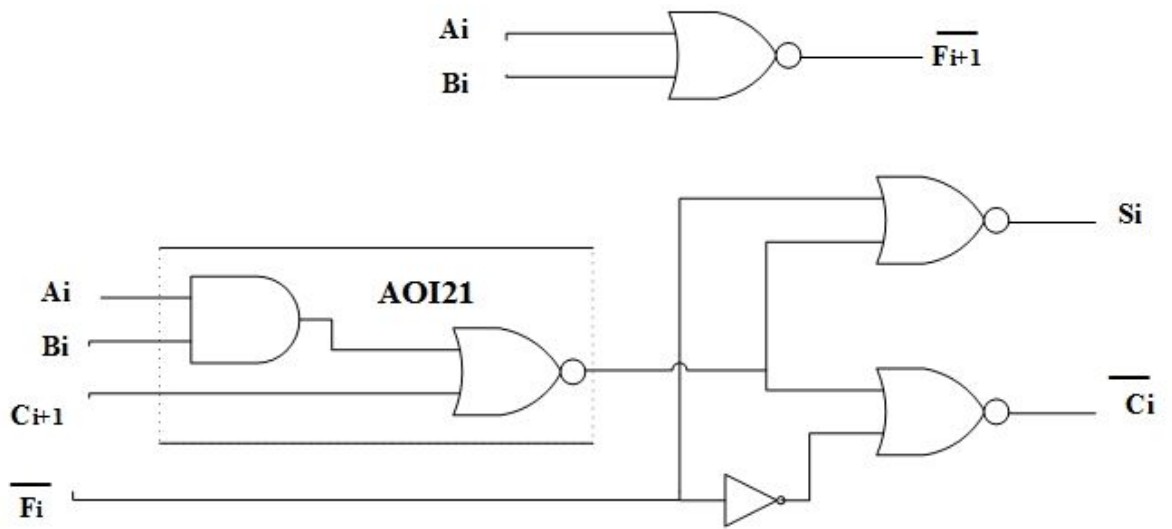


Fig.3.13. RCPFA-3.

TABLE III Truth Table of RCPFA-3[1]

ITEM	Ai	Bi	Ci+1	Fi	Si	Ci	Fi+1
1	0	0	0	0	0	0	0
2	0	0	0	1	1	1	0
3	0	0	1	x	0	1	0
4	0	1	0	0	0	0	1
5	0	1	0	1	1	1	1
6	0	1	1	x	0	1	1
7	1	0	0	0	0	0	1
8	1	0	0	1	1	1	1
9	1	0	1	x	0	1	1
10	1	1	x	0	0	0	1
11	1	1	x	1	1	1	1

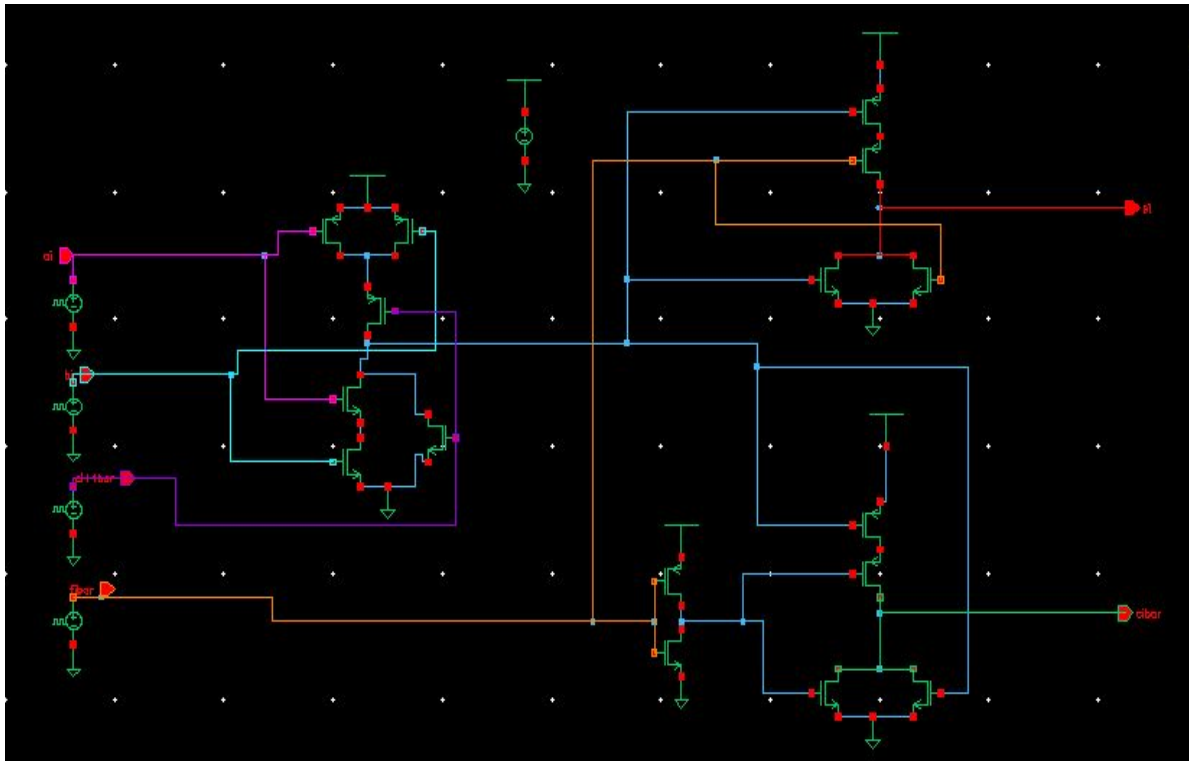


Fig.3.14. Schematic of RCPFA-3.

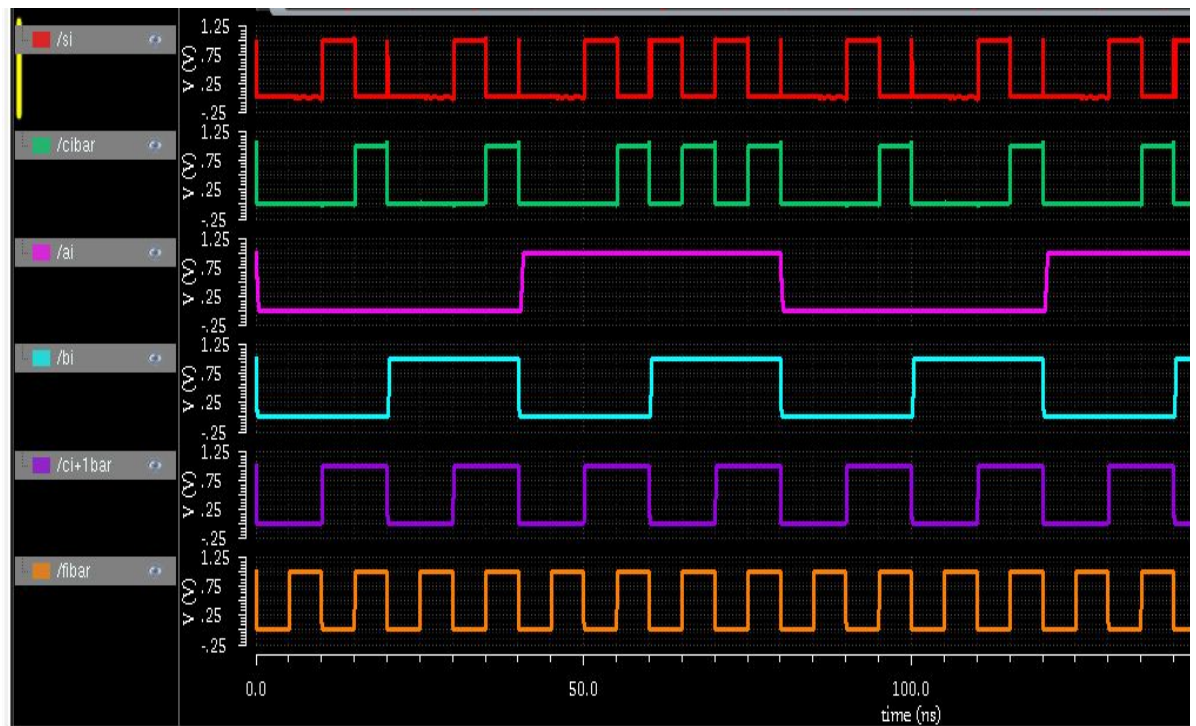


Fig.3.15. Transient response of RCPFA-3.

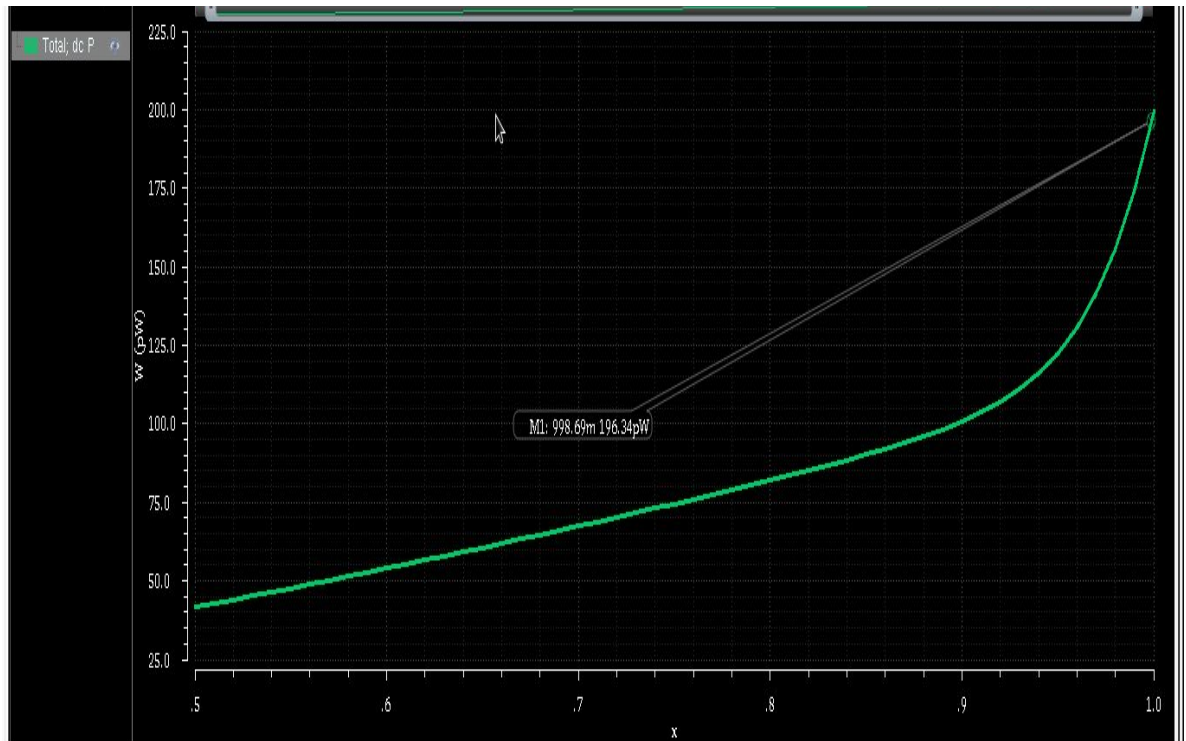


Fig.3.16. Power Plot of RCPFA-3.

CHAPTER 4

PROPOSED RCPFAs

4.1 Modified GDI Technique

The basic Modified cell is given in Fig.4.1. In this Modified-GDI cell, substrate terminal of PMOS is connected to V_{DD} and that of NMOS is connected to ground. In comparison to the static CMOS Process, this technique provides reduction in gate-leakage current and sub-threshold current[7].

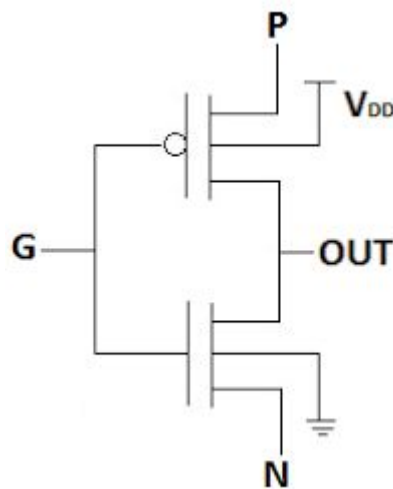


Fig.4.1 Modified GDI cell

Modified-GDI Technique is suitable for CMOS fabrication Process. Leakage currents can be measured in submicron region.

4.1.1 Miscellaneous Functions of Modified GDI Cell

By using an easy alteration to the input pattern of the basic Modified GDI cell, a large no of basic Boolean operations can be implemented as can be seen in Table IV.

TABLE IV Miscellaneous Functions of a Modified GDI Cell

N	P	G	OUT	FUNCTION	T.C. CMOS	T.C. M-GDI
0	B	A	A'B	F1	8	2
B	1	A	A'+B	F2	8	2
1	B	A	A+B	OR	6	2
B	0	A	AB	AND	6	2
C	B	A	AC+ A'B	MUX	6	2
0	1	A	A'	NOT	2	2

[T.C.-Transistor count]

4.2 PROPOSED RCPFA-1 CIRCUIT

The Proposed RCPFA circuit uses Modified GDI Technique to reduce no. of transistors to 22 ,while the transistor count was 26 in RCPFA-1 [1]. The transient response of the proposed RCPFA-1 is shown below in Fig.4.3. Total DC Power dissipated by the above proposed rcpfa is 311.2pW (Fig.4.4) which is very less as compared to the power dissipated by the existing RCPFA i.e 237.93pW. W/L Ratio for NMOS and PMOS transistors is 120nm/45nm and 300nm/45nm respectively. All simulation being carried out at 1V power supply.

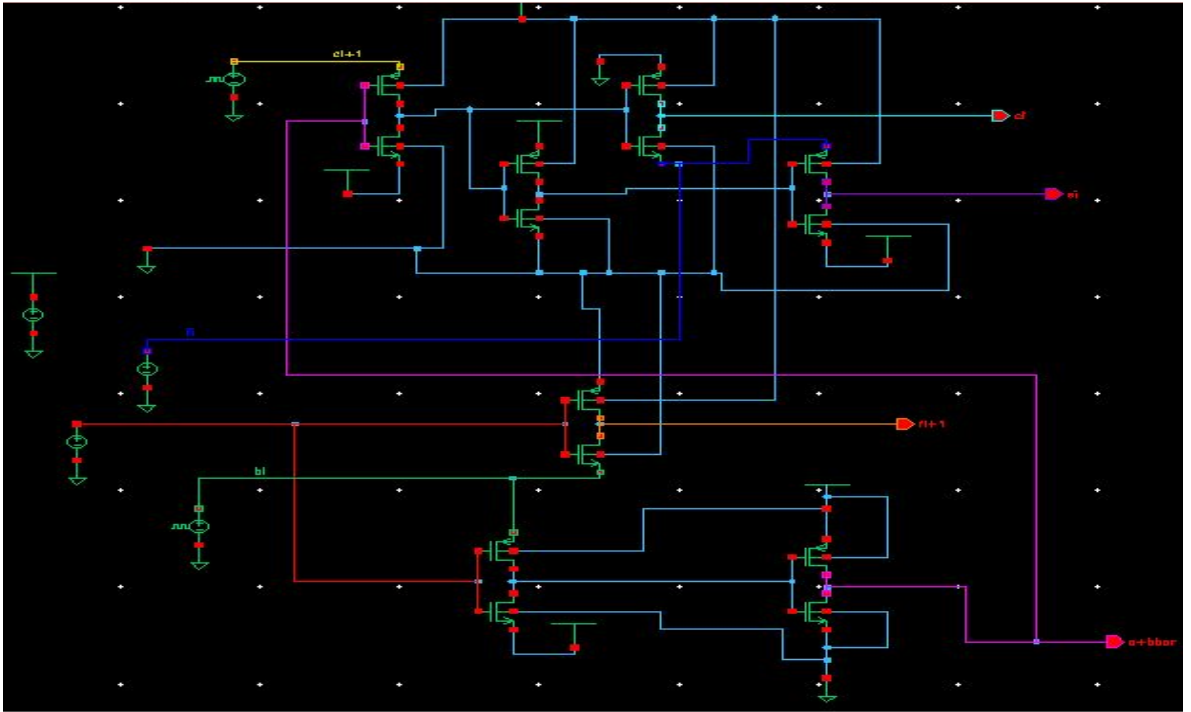


Fig.4.2 Schematic of Proposed RCPFA-1.

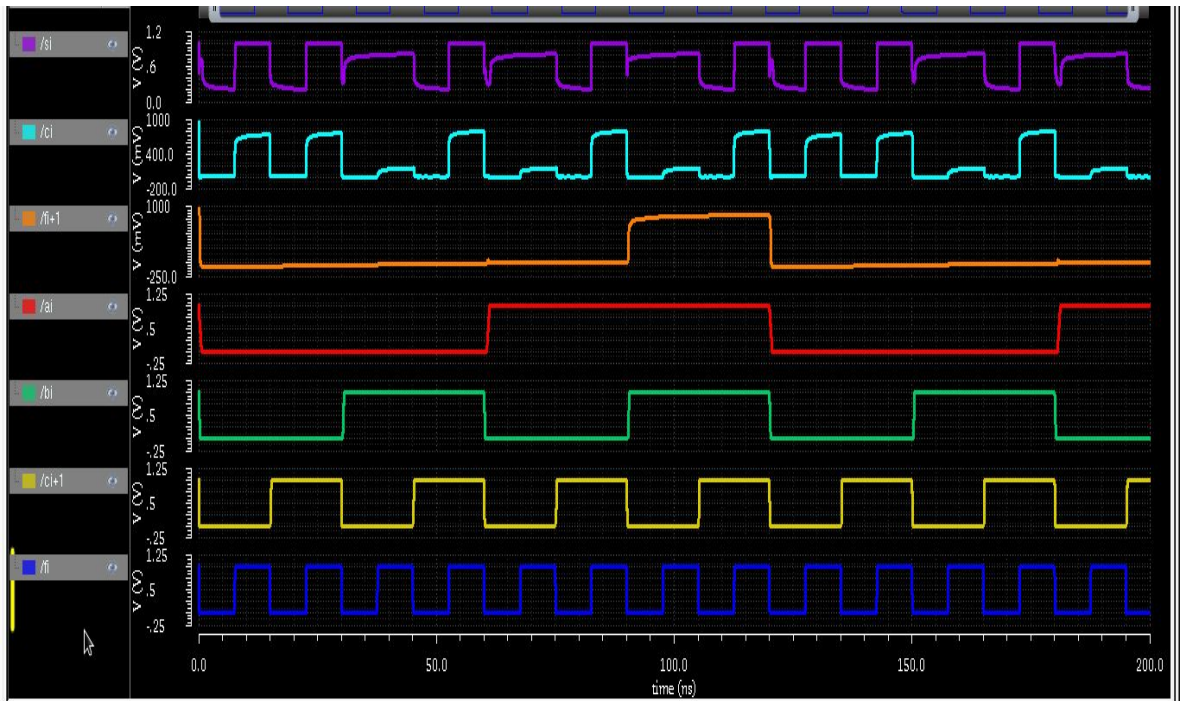


Fig.4.3 Transient response of proposed RCPFA-1.

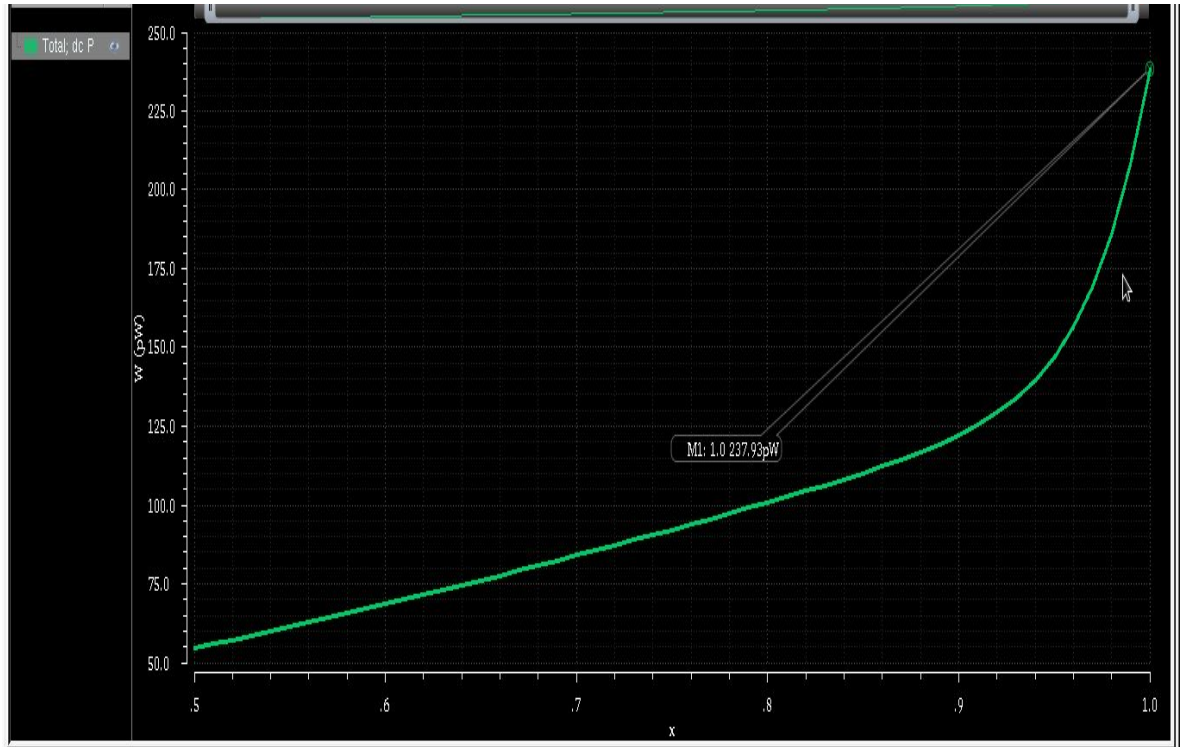


Fig.4.4. Power plot of Proposed RCPFA-1

4.3 PROPOSED RCPFA-2 CIRCUIT

The Proposed RCPFA-2 circuit (Fig.4.5) uses Modified GDI Technique to reduce no. of transistors to 14 ,while the transistor count was 20 in RCPFA-2[1].

By replacing X_i with 1 equation (3.5) and (3.6) reduces to

$$S_i = F_i + \overline{C_{i+1}} (A_i + B_i) \quad (4.1)$$

$$C_i = F_i (\overline{\overline{C_{i+1}} (A_i + B_i)}) \quad (4.2)$$

The transient response of the proposed RCPFA is shown in Fig.4.6.

Total DC Power (shown in Fig.4.7) dissipated by the above proposed rcpfa is 110.6pW which is very less as compared to the power dissipated by the existing RCPFA i.e 155.57pW. W/L Ratio for NMOS and PMOS transistors is 120nm/45nm and 300nm/45nm respectively. All simulation being carried out at 1V power supply.

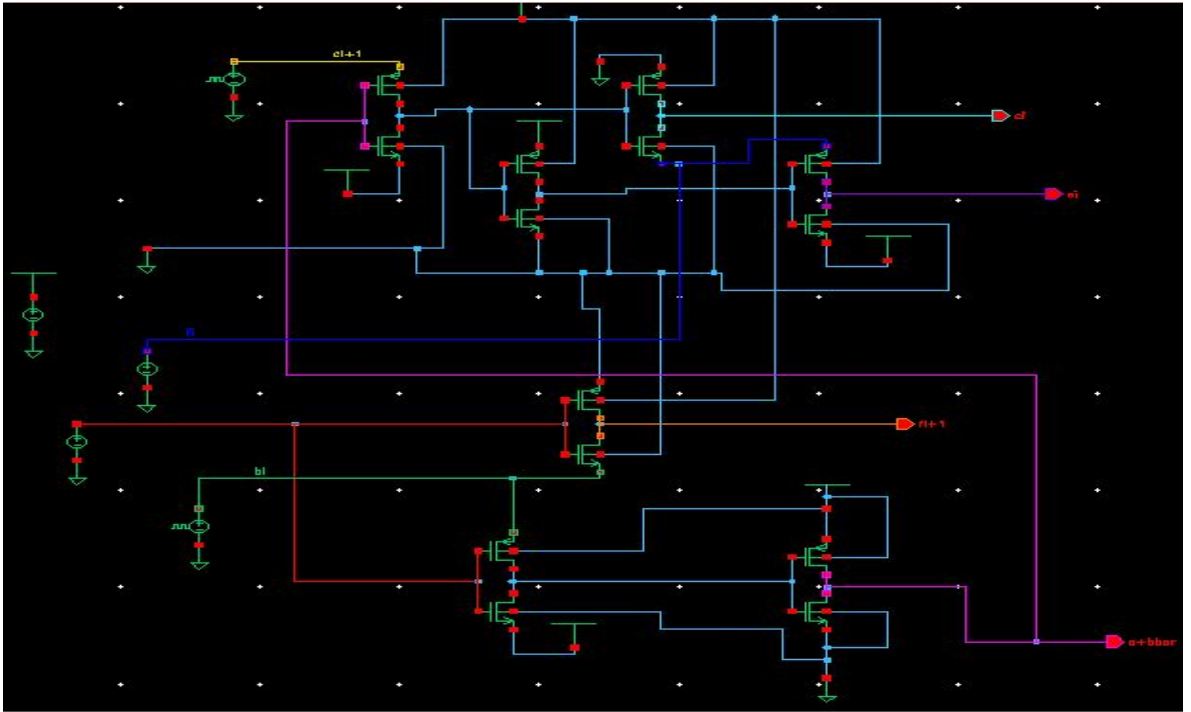


Fig.4.5. Schematic of Proposed RCPFA2.

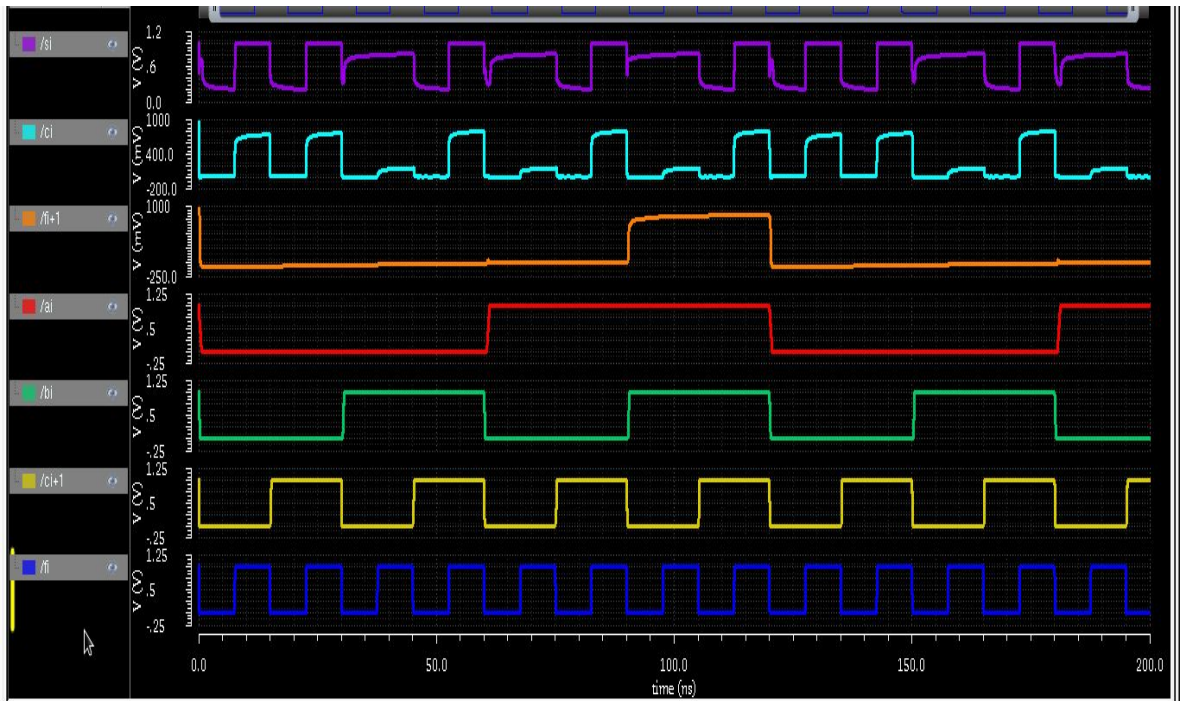


Fig.4.6. Transient response of proposed RCPFA2.

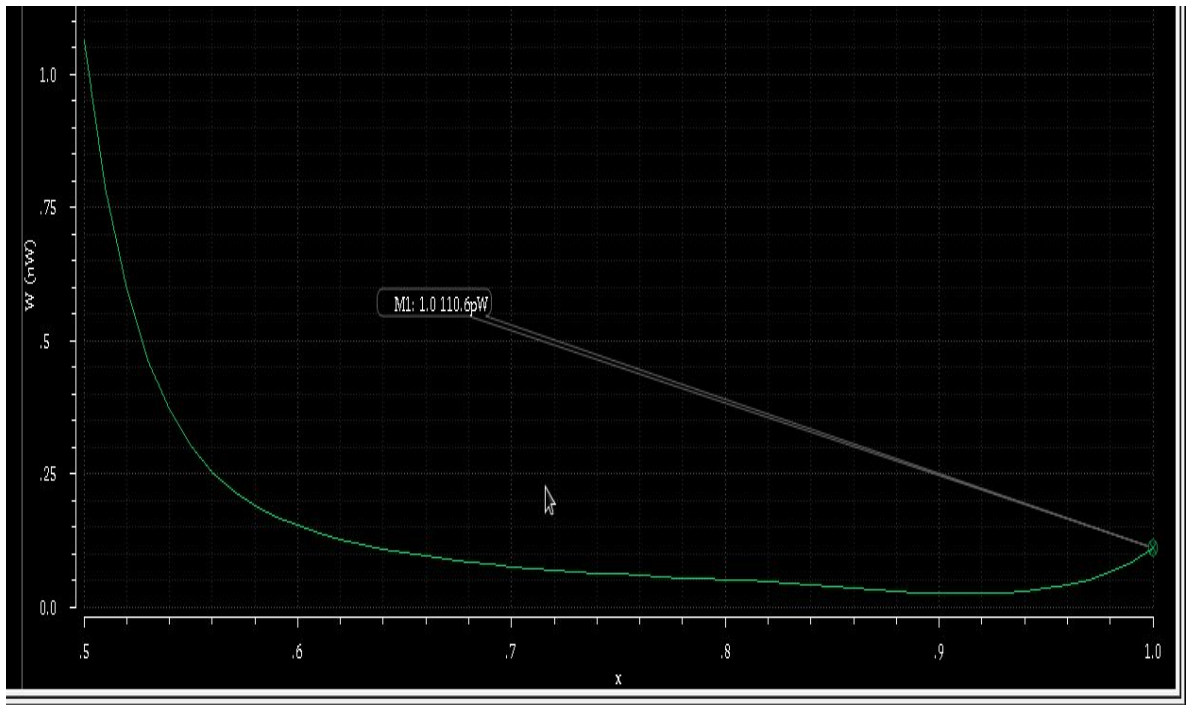


Fig.4.7. Power plot of Proposed RCPFA-2.

4.4 PROPOSED RCPFA-3 CIRCUIT

The Proposed RCPFA-3(Fig.4.8) circuit uses Modified GDI Technique to reduce no. of transistors to 14 ,while the transistor count was 20 in RCPFA-3[1].

By replacing Xi with 1 equation (3.5) and (3.6) reduces to

$$S_i = F_i (\overline{C_{i+1}} + A_i B_i) \quad (4.3)$$

$$C_i = F_i + (\overline{C_{i+1}} + (A_i B_i)) \quad (4.4)$$

The transient response of the proposed RCPFA-3 is shown in Fig.4.9.

Total DC Power (shown in Fig.4.10) dissipated by the above proposed RCPFA-3 is 153.95pW which is very less as compared to the power dissipated by the existing RCPFA-3 i.e 196.34pW. W/L Ratio for NMOS and PMOS transistors is 120nm/45nm and 300nm/45nm respectively. All simulation being carried out at 1V power supply.

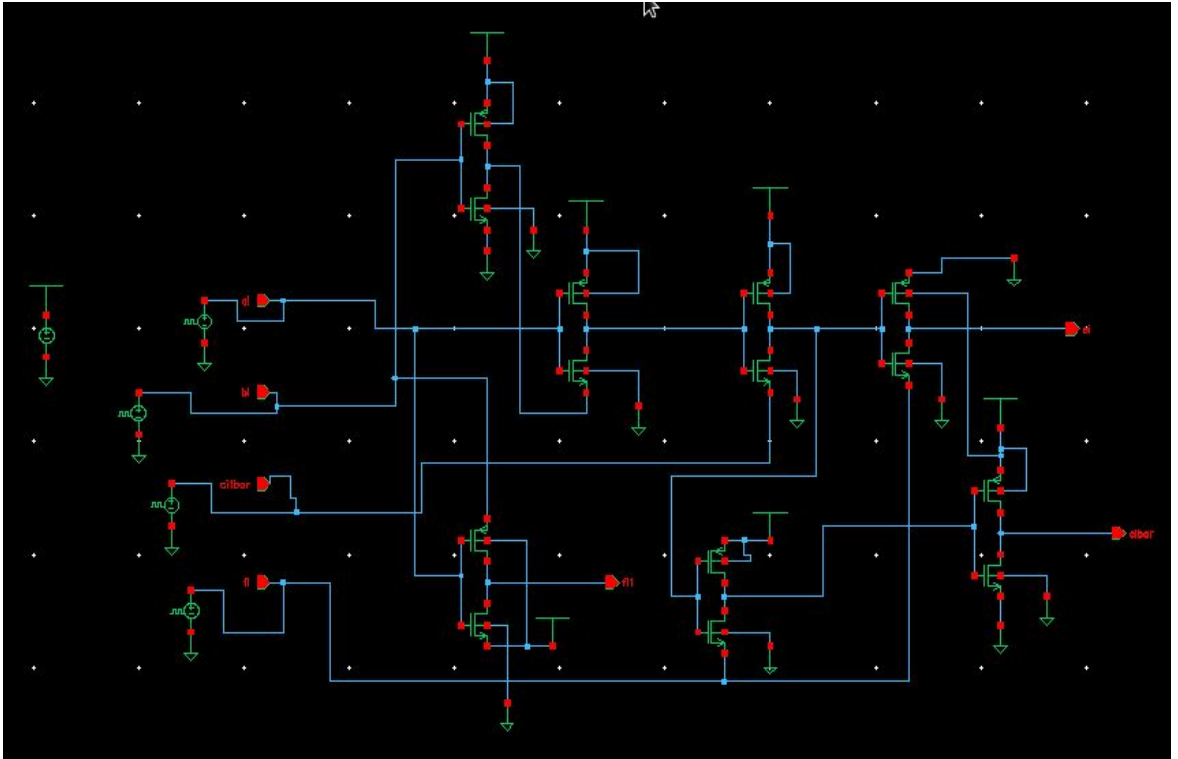


Fig.4.8 Schematic of Proposed RCPFA-3.

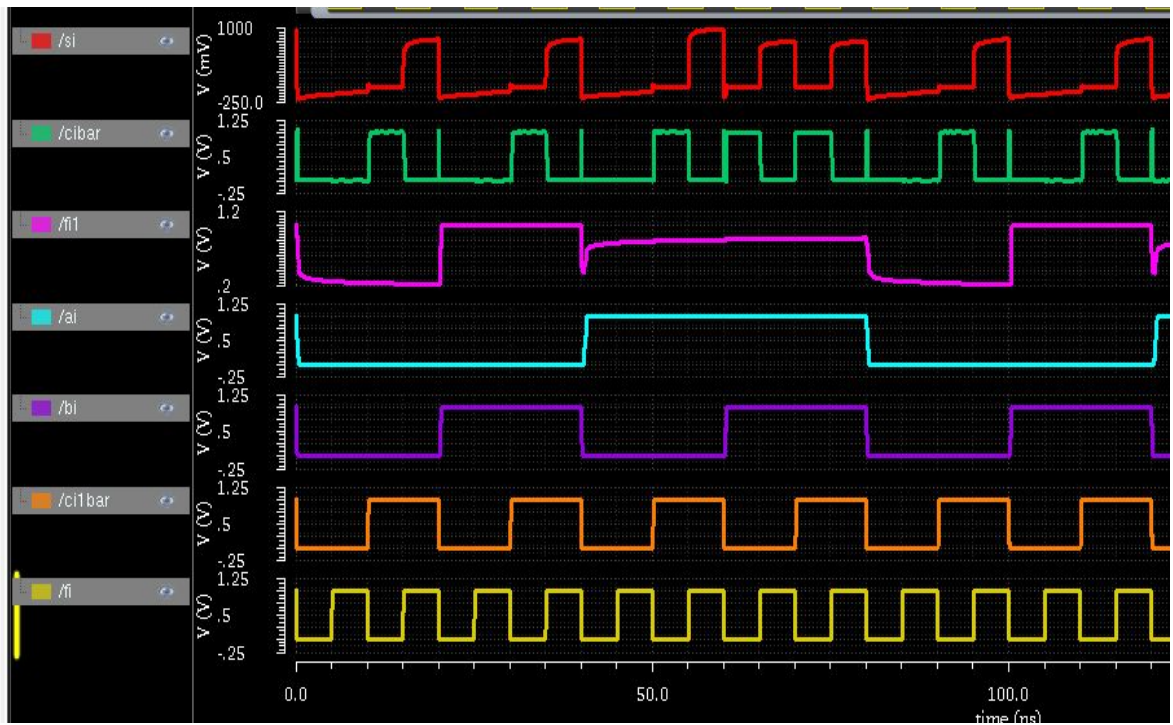


Fig.4.9. Transient response of proposed RCPFA-3.

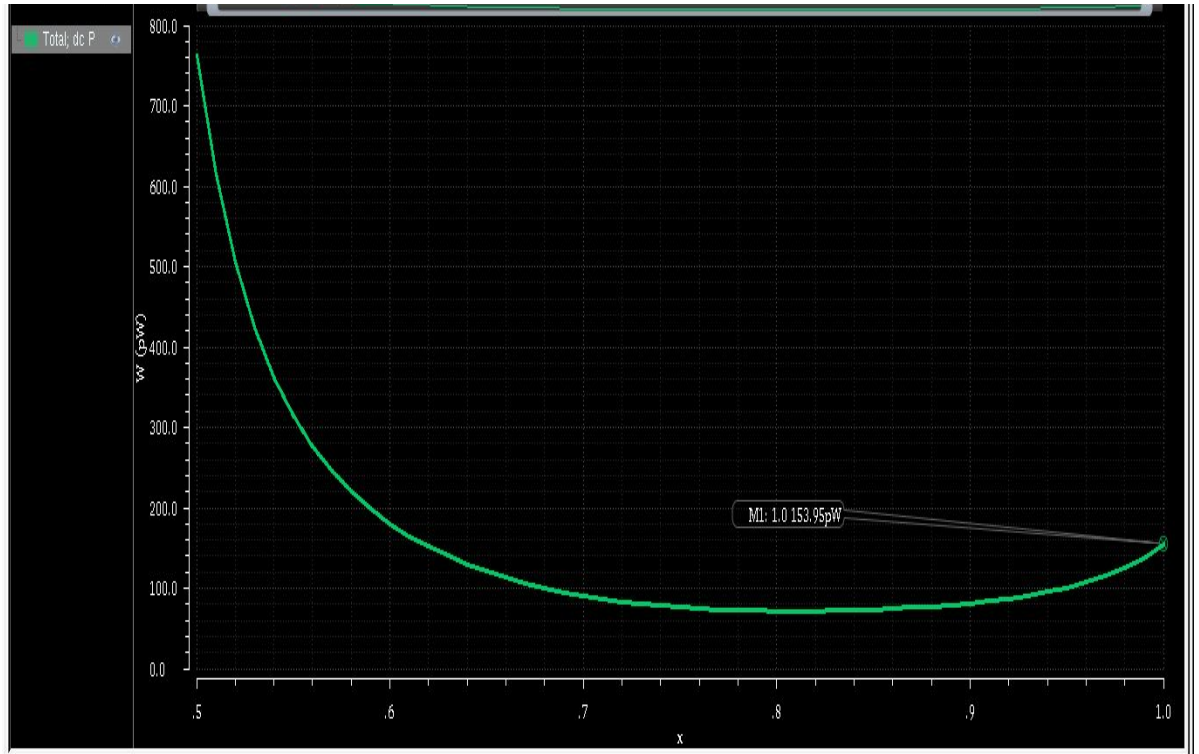


Fig.4.10. Power plot of Proposed RCPFA-3.

4.5 COMPARISON WITH RCPFAs[1]

As the no. of transistor has reduced from 20 to 14 , delay of the proposed circuit has also reduced considerably. Various delay including carry-propagation delay, carry-summation delay and forecast-signal delay values are shown in Table II.

TABLE V Comparison Between Proposed and Conventional RCPFAs.

<u>FA TYPE</u>	<u>POWER</u> <u>pW</u>	<u>T_{cp}</u> <u>ps</u>	<u>T_{cs}</u> <u>ps</u>	<u>T_f</u> <u>ps</u>	<u>no. of</u> <u>Transistors</u>
<u>RCPFA-1[1]</u>	311.2	178.2	203.7	n.a	26
<u>RCPFA-2[1]</u>	155.57	124.6	115.5	48.3	20
<u>RCPFA-3[1]</u>	196.34	146.8	181.4	53.5	20
<u>PROPOSED-1</u>	237.93	149.6	107.3	n.a	22
<u>PROPOSED-2</u>	110.6	69.56	55.84	20.87	14
<u>PROPOSED-3</u>	153.95	87.3	76.2	34.6	14

4.6 32-bit Hybrid Adder

The proposed RCPFA can be used in realizing hybrid adders. The general n -bit structure having k (width of the approximate part) RCPFA adders and remaining $(n-k)$ exact full adders is shown in Fig.4.11. Schematic diagram for the same is also shown in the Fig.4.12. The delay, power and the energy-delay product vary with the width of the approximate part of the hybrid adder. This is the reason why critical path delay is measured from the joining point of the two parts.

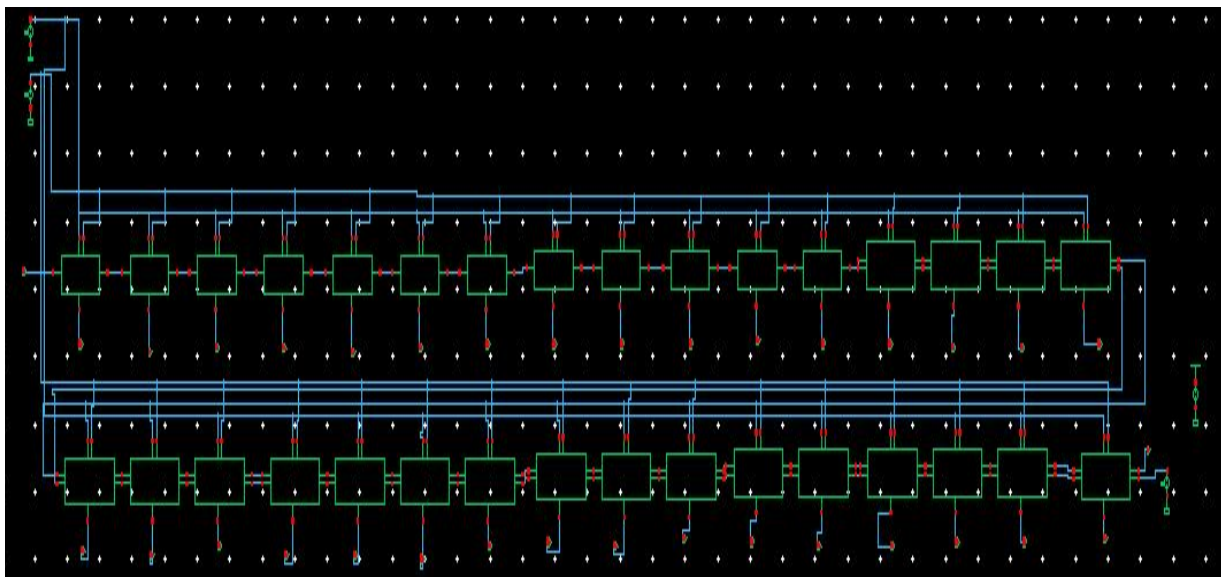


Fig.4.11. Schematic of 32-bit hybrid adder using Proposed RCPFA.

From this point, the carry is propagated to the MSB of the exact part and to the LSB of the inexact part. The graph of delay Vs width of approximate part for 32-bit hybrid adder using RCPFA is drawn in Fig.4.13. The value for k is varied between 0 and 32 and the critical path delay is measured. When the width of the approximate part (k) is 23, the critical path will show the minimum delay.

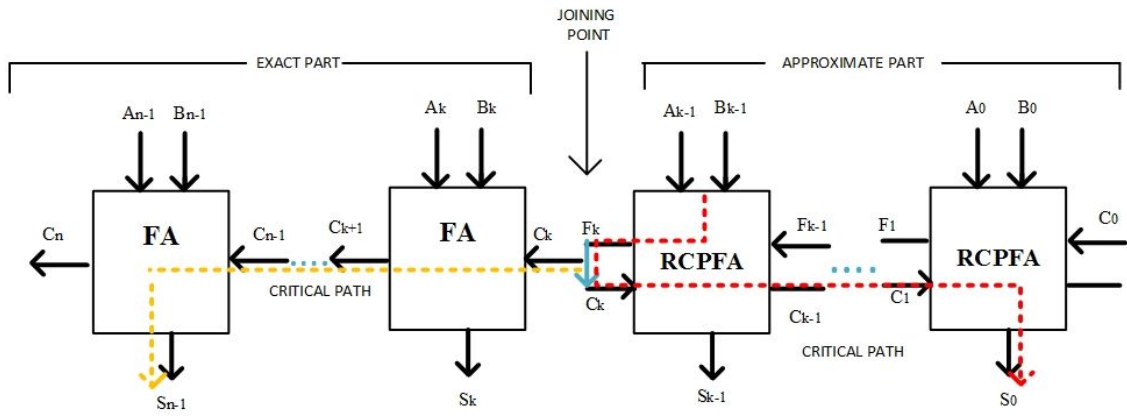


Fig.4.12 n-bit hybrid adder with k-bit RCPFA part.

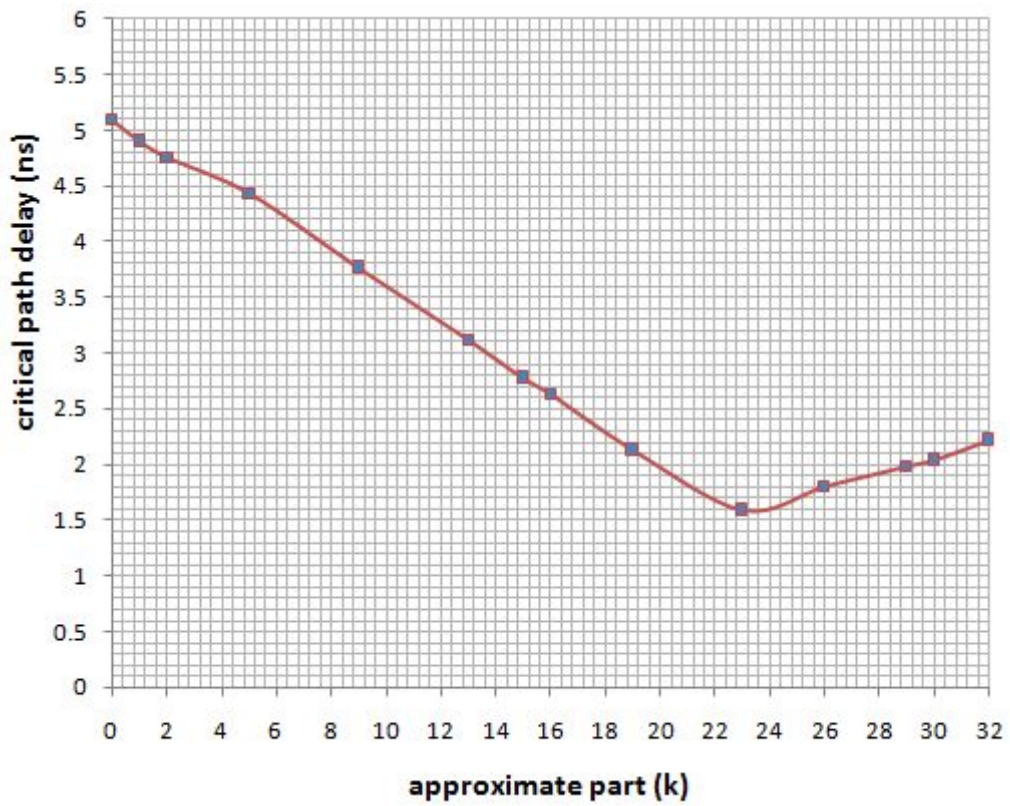


Fig.4.13. Delay of the 32bit hybrid adder Vs the approximate part width (k).

CHAPTER 5

MULTIPLIER

5.1 RIPPLE-CARRY ADDER USING RCPFA

Ripple-carry adder is a combinational circuit which calculates the arithmetic sum of two input binary numbers. It is constructed by connecting full adder circuits in cascade, where the carry-out of one full adder is connected to the carry-in of the next full adder.

As shown in the Fig.5.1 four RCPFA Adders are used together to form a 4 bit ripple carry adder. The carry is propagated in the reverse direction and carry(C_0) is connected to the forecast signal (F_0) of the least significant rcdfa as there is a lack of previous stage for producing the F_0 signal. In the case of most significant rcdfa the carry input (C_4) is assumed to be equal to forecast signal output (F_4).

In a ripple-carry adder, the outputs are known only after the carry has propagated from the least significant adder to most significant adder. In other words, output is valid only when the carry has rippled through all the stages. So, the sum and carry-out are available only after a considerable delay.

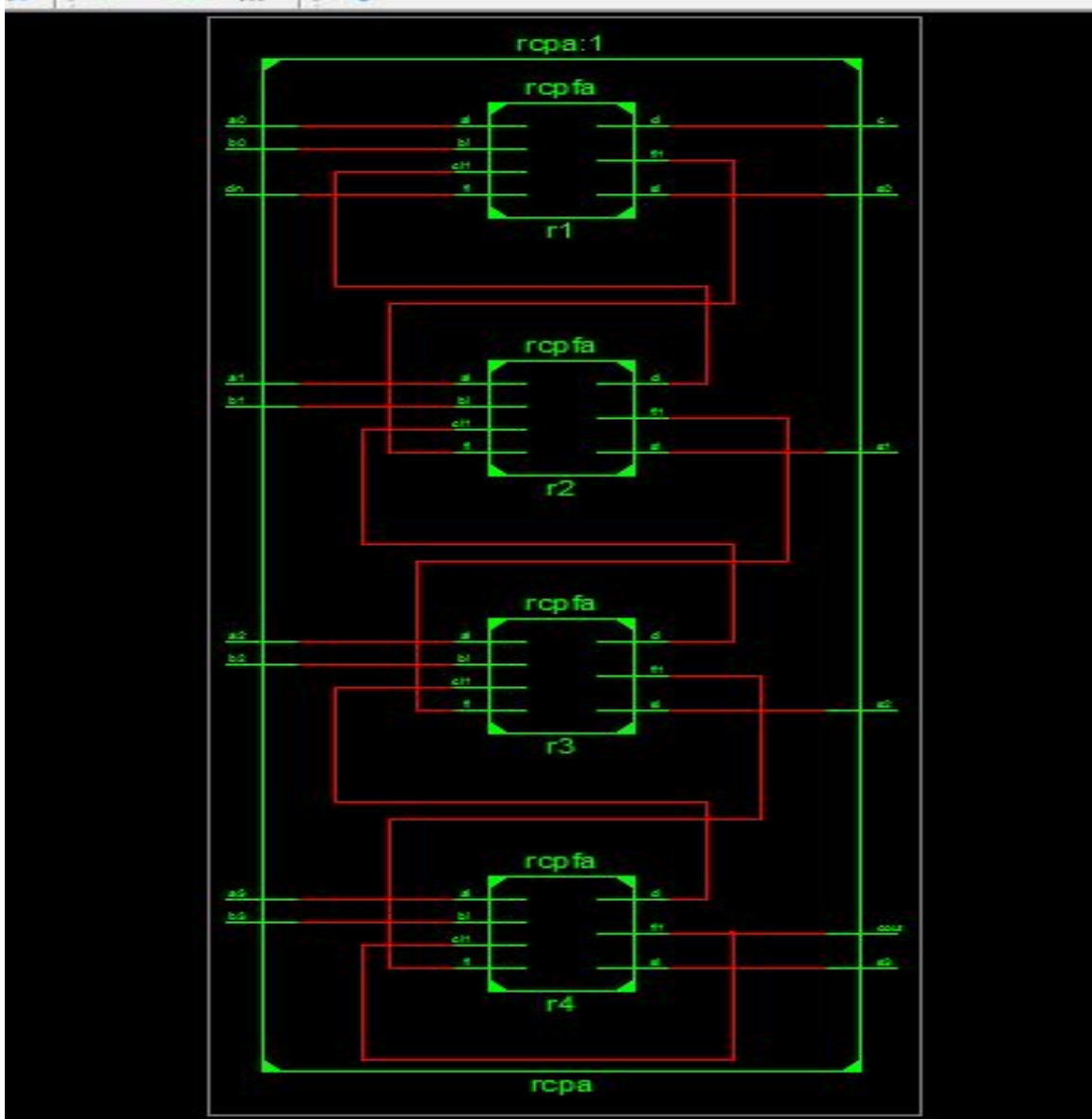


Fig.5.1. Ripple-Carry adder using RCPFA Cell.

5.2 Multiplier using RCPFA

In the proposed 4-bit multiplier circuit design, RCPFA cells are used to design a 4-bit ripple-carry adder and then these adders are utilized in the multiplier circuit. Normally, N-1 ripple carry adders are required to implement the multiplier module, where N is the number of input bits.

4-bit binary multiplier using above designed ripple carry adder is shown in Fig.5.2. by replacing full adders with 4 bit ripple carry adders using reverse carry propagate adder. It is evident that 3 number of RCA modules are necessary in order to obtain 4 bit multiplier circuit.

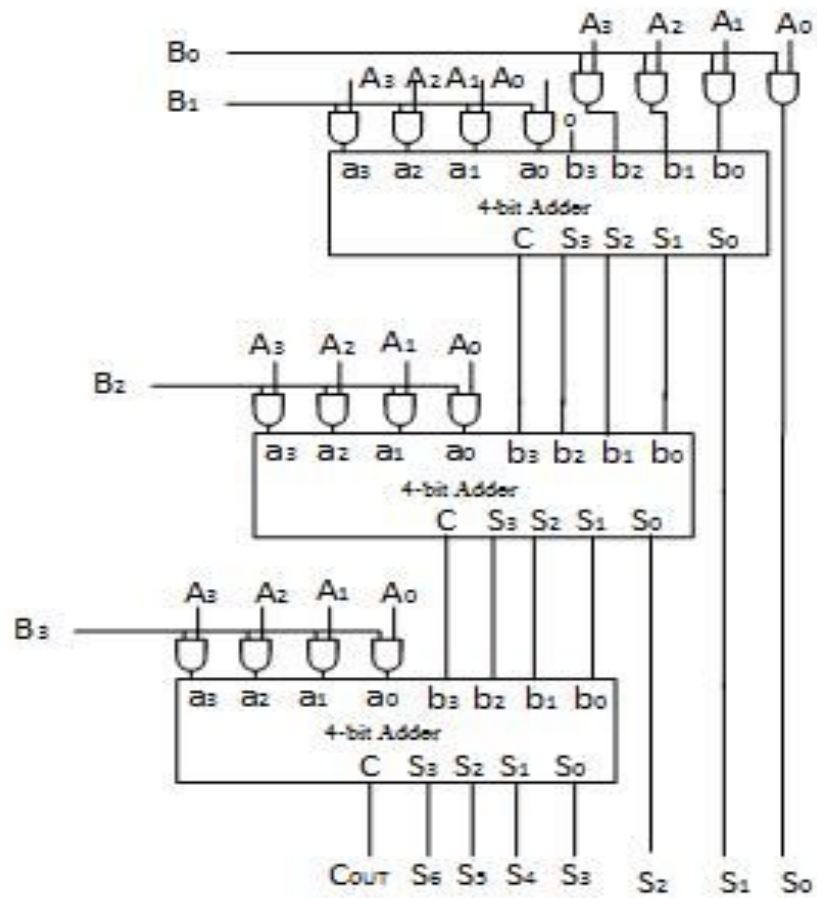


Fig.5.2. 4-bit binary Multiplier

CHAPTER 6

VERILOG CODES AND SIMULATION RESULTS

6.1 RCPFA-2

6.1.1 Verilog code for RCPFA-2

```
module rcpfa(input ai,bi,fi,ci1,
             output ci,fi1,si
);

    assign si= fi | ((~ci1)&(ai|bi));
    assign ci=fi & ( ci1 | ( ~ai) & (~bi));
    assign fi1=ai&bi;

endmodule
```

6.1.2 Test bench for RCPFA-2

```
module rcpfatest;
reg ai,bi,ci1,fi;
wire ci,si,fi1;

    rcpfa uut(.ai(ai),.bi(bi),.ci(ci),.fi(fi),.ci1(ci1),.fi1(fi1),.si(si));

initial
begin
ai=0;
bi=0;
ci1=0;
fi=0;
end
always
forever #80 ai=~ai;
always
forever #40 bi=~bi;
always
```

```

forever #20 ci1=~ci1;
always
forever #10 fi=~fi;

endmodule

```

6.1.3. Simulation Result

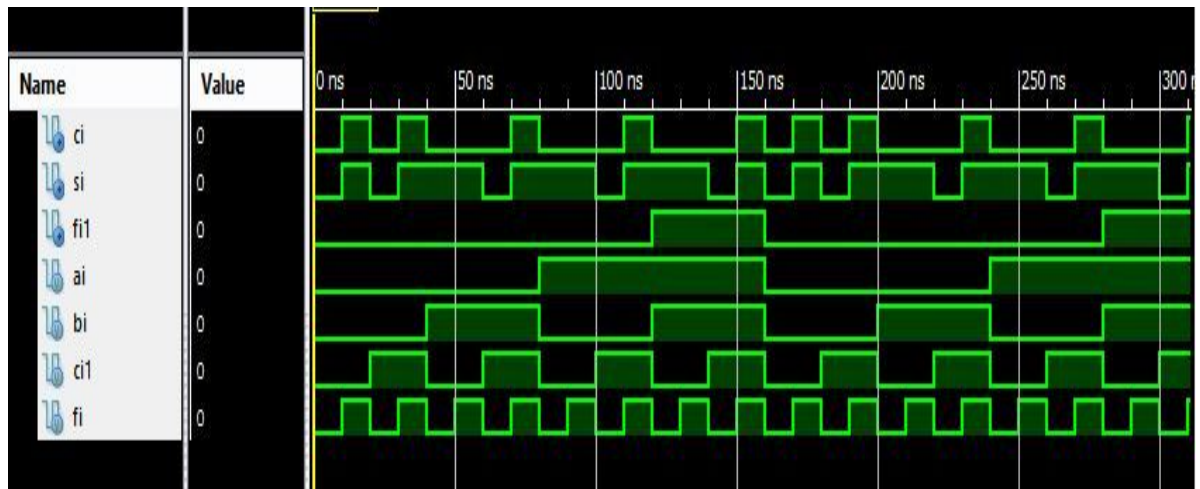


Fig.6.1 Simulation Result for RCPFA

6.2 RCPA

6.2.1 Verilog code for 4-bit RCPA

```

module rcpa(input cin,a0,a1,a2,a3,b0,b1,b2,b3,
            output c,
            output cout,s0,s1,s2,s3 );
    wire w1,w2,w3,w4,w5,w6,c4;
    rcpfa r1(.ai(a0),.bi(b0),.ci(c),.fi(cin),.ci1(w1),.fi1(w2),.si(s0));
    rcpfa r2(.ai(a1),.bi(b1),.ci(w1),.fi(w2),.ci1(w5),.fi1(w3),.si(s1));
    rcpfa r3(.ai(a2),.bi(b2),.ci(w5),.fi(w3),.ci1(w6),.fi1(w4),.si(s2));
    rcpfa r4(.ai(a3),.bi(b3),.ci(w6),.fi(w4),.ci1(c4),.fi1(c4),.si(s3));
    assign cout=c4;
endmodule

```

6.2.2. Test bench for RCPA

```

module testrcpa;

```

```

reg cin;
reg [3:0]at;
reg [3:0]bt;
wire cout,c;
wire s0,s1,s2,s3;
rcpa uut1 (.a0(at[0]),
           .a1(at[1]),
           .a2(at[2]),
           .a3(at[3]),
           .b0(bt[0]),
           .b1(bt[1]),
           .b2(bt[2]),
           .b3(bt[3]),
           .c(c),
           .cout(cout),
           .s0(s0),
           .s1(s1),
           .s2(s2),
           .s3(s3),
           .cin(cin));

```

```

initial
begin
at=4'b0000;
bt=4'b0001;
cin=0;
end
initial
begin
#10
at=4'b0110;
bt=4'b0010;
#10
at=4'b0011;
bt=4'b0111;
#10
at=4'b1110;
bt=4'b0010;
#10
at=4'b1111;
bt=4'b0011;
end

```

6.2.3. Simulation Result

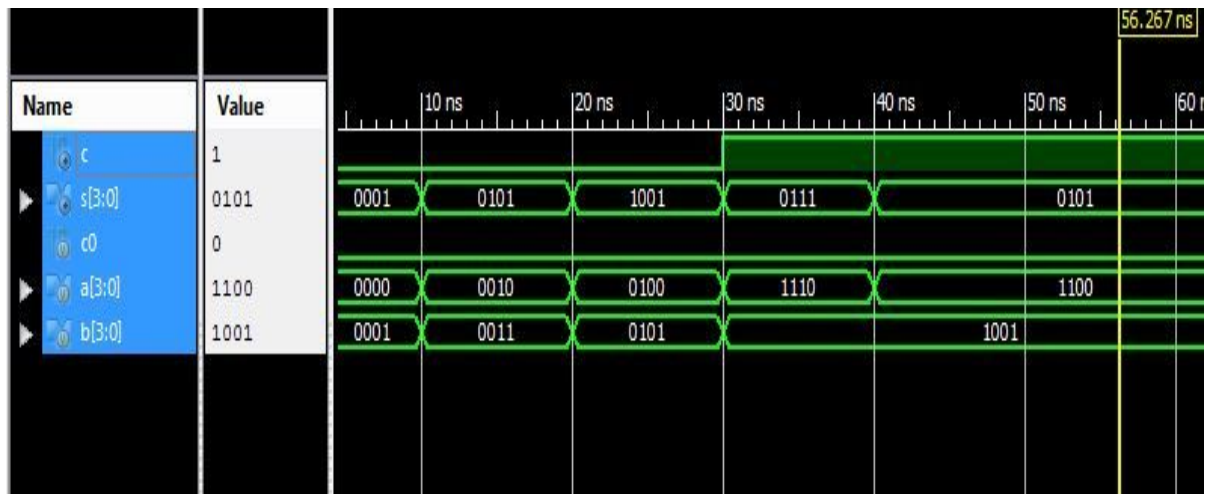


Fig.6.2. Simulation Result for RCPA

6.3 4-bit Multiplier

6.3.1 Verilog code for 4×4 Multiplier

```
module mul3( input cin,clk,[3:0] ai,bi,  
            output s0,s1,s2,s3,s4,s5,s6,cout  
            );
```

```
wire [15:0] n;  
wire y1, y2, y3, y4, y5, y6,y7,y8,y9;  
assign n[0]=ai[0]&bi[0];  
assign n[1]=ai[1]&bi[0];  
assign n[2]=ai[2]&bi[0];  
assign n[3]=ai[3]&bi[0];  
assign n[4]=ai[0]&bi[1];  
assign n[5]=ai[1]&bi[1];  
assign n[6]=ai[2]&bi[1];  
assign n[7]=ai[3]&bi[1];  
assign n[8]=ai[0]&bi[2];  
assign n[9]=ai[1]&bi[2];  
assign n[10]=ai[2]&bi[2];  
assign n[11]=ai[3]&bi[2];
```

```

assign n[12]=ai[0]&bi[3];
assign n[13]=ai[1]&bi[3];
assign n[14]=ai[2]&bi[3];
assign n[15]=ai[3]&bi[3];
assign y9=cin;
rcpa r1
(.a0(n[1]),.a1(n[2]),.a2(n[3]),.a3(1'b0),.b0(n[4]),.b1(n[5]),.b2(n[6]),.b3(n[7]),.s0(s1),.s
1(y1),.s2(y2),.s3(y3),.c(y9),.cout(y4),.cin(cin));
rcpa r2
(.a0(y1),.a1(y2),.a2(y3),.a3(y4),.b0(n[8]),.b1(n[9]),.b2(n[10]),.b3(n[11]),.s0(s2),.s1(y5
),.s2(y6),.s3(y7),.c(y9),.cout(y8),.cin(cin));
rcpa r3
(.a0(y5),.a1(y6),.a2(y7),.a3(y8),.b0(n[12]),.b1(n[13]),.b2(n[14]),.b3(n[15]),.s0(s3),.s1(
s4),.s2(s5),.s3(s6),.c(y9),.cout(cout),.cin(cin));
assign s0=n[0];
endmodule

```

6.3.2. Test bench

```

module testmul3;
reg clk;
reg cin;
reg [3:0]at;
reg [3:0]bt;
wire s0,s1,s2,s3,s4,s5,s6,cout;
    mul3 M(.clk(clk),.cin(cin),.ai(at),.bi(bt),
        .s0(s0),.s1(s1),.s2(s2),.s3(s3),.s4(s4),.s5(s5),.s6(s6),.cout(cout)
    );
    initial
        begin
            clk=0;
            cin=0;
            at=4'b0000;
            bt=4'b0000;
        end
    always #10 clk=~clk;

    initial
        begin
            #85
            at=4'b0010;

```



```
bt=4'b0010;  
    #85  
at=4'b0010;  
bt=4'b1010;  
    #85  
at=4'b0101;  
bt=4'b0110;  
    #85  
at=4'b0001;  
bt=4'b0001;  
    #85  
at=4'b0100;  
bt=4'b0100;  
    #85  
at=4'b1000;  
bt=4'b0100;  
    #85  
at=4'b1100;  
bt=4'b0100;  
    #85  
at=4'b0110;  
bt=4'b0101;  
    #85  
at=4'b0101;  
bt=4'b0101;  
    #85  
at=4'b1001;  
bt=4'b0001;  
end
```

endmodule

6.3.3. Simulation Result

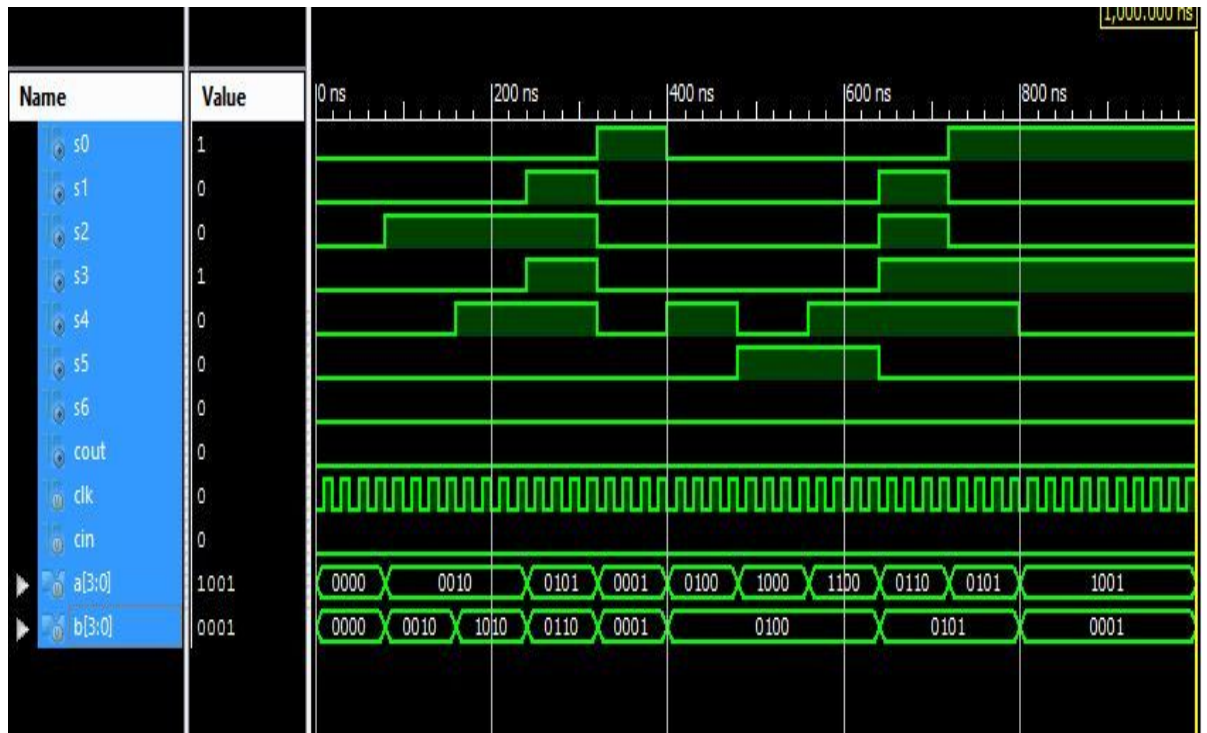


Fig.6.3. Simulation Result for 4-bit Multiplier

CONCLUSION

In this dissertation, new circuits have been proposed for Approximate Reverse Carry Propagate Full Adder based on Modified-GDI technique that propagates carry in counter flow direction from MSB to LSB. The propagation of carry in reverse direction insures higher stability in the delay variations as the weight of the carry decreases when the carry is propagated in a reverse direction. The efficacies of the proposed Reverse Carry Propagate Full Adder and the existing RCPFA[1] have been verified using 45-nm technology in Cadence Virtuoso. A 4-bit multiplier has also been designed using this RCPFA and results verified with Xilinx Tool. The results provided in Table V justify that the proposed RCPFA gives on average 57% and 44% improvement in power and delay respectively.

REFERENCES

- [1] M. Pashaeifar, M. Kamal, and A. Afzali-Kusha, "Approximate Reverse Carry Propagate Adder for Energy-Efficient DSP Application," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 11, pp. 2530–2541, Nov. 2018.
- [2] T. Moreau, A. Sampson, and L. Ceze, "Approximate computing: Making mobile systems more efficient," *IEEE Pervasive Comput.*, vol. 14, no. 2, pp. 9–13, Apr. 2015.
- [3] V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, "Low-power digital signal processing using approximate adders," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 1, pp. 124–137, Jan. 2013.
- [4] H. R. Mahdiani, A. Ahmadi, S. M. Fakhraie, and C. Lucas, "Bio-inspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 4, pp. 850–862, Apr. 2010.
- [5] Z. Yang, A. Jain, J. Liang, J. Han, and F. Lombardi, "Approximate XOR/XNOR-based adders for inexact computing," in *Proc. 13th IEEE Int. Conf. Nanotechnol. (NANO)*, pp. 690–693, Aug. 2013.
- [6] Z. Yang, J. Han, and F. Lombardi, "Transmission gate-based approximate adders for inexact computing," in *Proc. IEEE/ACM Int. Symp. Nanosc. Archit. (NANOARCH)*, Jul. 2015, pp. 145–150.
- [7] B. KeerthiPriya and R. Manoj Kumar "A New Low Power Area Efficient 2Bit Magnitude Comparator using Modified GDI technique in Cadence 45nm Technology" in *IEEE 2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)*, May 25-27, pp.30-34, 2016.

NEW REVERSE CARRY PROPAGATE ADDER USING MODIFIED GDI TECHNIQUE

DivyaChoudhary
Department of Electronics and Communication Engineering
Delhi Technological university
Delhi, India
divya.ddps@gmail.com

D. R. Bhaskar
Department of Electronics and Communication Engineering
Delhi Technological university
Delhi, India
drbhaskar@dtu.ac.in

Abstract— Addition is the most important function in arithmetic and logical operations. Approximate Computing can be used to reduce the number of transistors, delay and power constraints in VLSI design, which makes the use of approximate adders possible in error-tolerant applications. Existing Approximate Reverse Carry Propagate Adder designs [1] have proved to be advantageous in improving these constraints. A new design of Reverse Carry Propagate Adder has been proposed using Modified-Gate Diffusion Input (GDI) technique [2]. Proposed circuit design simulations have been carried out in 45-nm process technology using Cadence Virtuoso. The results indicate 57% and 44% reduction in Power and Delay respectively.

Keywords—Reverse Carry Propagate Adder, Modified-Gate Diffusion Input, carry propagation delay, carry to summation delay

I. INTRODUCTION

With the advancement in technology, the number of transistors in an integrated circuit is increasing and devices are getting more compact day by day. Due to which, the demand for more compact implementation of circuits is increasing. As adder is the main building block of arithmetic and logic unit, many researchers have presented their research works [3]-[6] on optimizing speed and power of adder by using Approximate Computing. Approximate adders have gained importance for utilization in circuits where some of error is tolerable. The use of the Approximate Reverse Carry Propagate Adder (RCPA) is presented in [1], which propagates the carry in reverse direction, thereby decreasing the carry weight as it propagates making the adder less vulnerable to delay variations as compared to conventional adders.

In this paper, we focus on reducing the number of transistors in RCPFA by utilizing the Modified-GDI technique.

In section-II, Reverse Carry Propagate Adder [1] is briefly reviewed. A new Reverse Carry Propagate Full Adder(RCPFA) circuit is proposed in section-III. The validity of proposed RCPFA circuit through simulation results is given in section-IV. Lastly, the concluding remarks have been presented in section V.

II. REVERSE CARRYPROPAGATE ADDER

The conventional full adder has three inputs (namely the two input bits and an input carry) and two outputs (sum and carry out). The sum having same weight as the inputs and carry out with double the weight of the inputs. Therefore, if the carry is propagated in reverse direction, the total error introduced due to set-up and hold time violations will be reduced.

RPCFA can be described with the help of following equation

$$S_i - C_i = A_i + B_i - 2C_{i+1} \quad (1)$$

Where $A_i(B_i)$ is the i^{th} bit of the input A(B), $C_i(C_{i+1})$ is the carry in (carry out) and S_i is the i^{th} bit of sum S.

When the right side of equation (1) becomes zero, (S, C) can take either (0, 0) or (1, 1) set as values. The forecast signal (F_i) is used to select one of the two sets. Forecast signal depends on the $(i-1)^{\text{th}}$ bit of the inputs.

Based on the discussion given above, a family of RCPFAs has been suggested in [1]. Fig. 1 shows a symbolic notation for RCPFA cell with four inputs (A_i , B_i , C_{i+1} and F_i) and three outputs (S_i , C_i , F_{i+1}).

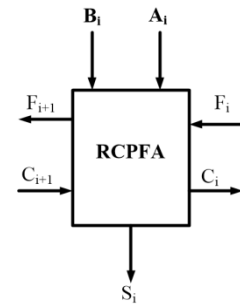


Fig. 1 RCPFA Cell

Fig. 2 shows an n-bit RPCFA, where most significant carry C_n is taken equal to the forecast signal F_1 of the most significant bit (MSB) RPCFA. Also, the input carry C_0 is used as forecast signal of least significant bit (LSB) RCPFA. The critical path for the n-bit RPCFA is also shown in the Fig. 2. Since, the error value increases in the direction of decrease of bit significance, the cumulative impact due to error (because of delay variation) is lower for higher significant bits.

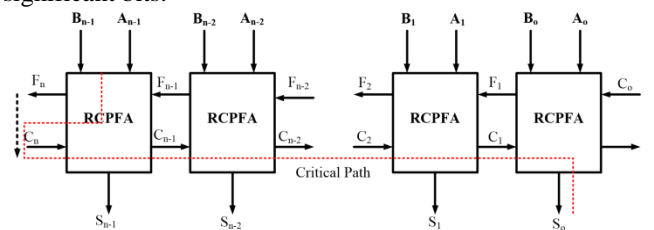


Fig. 2 n-bit RPCFA

Three types of RCPFAs are given in [1], RPCFA-1, RPCFA-2, RPCFA-3 with forecast signal equal to one of the input operands in RPCFA-1, the carry generate signal in RPCFA-2 and the carry alive signal in RPCFA-3. Standard CMOS gates have been used for the implementation of these RCPFAs which require 26, 20 and 20 transistors for RPCFA-1, RPCFA-2 and RPCFA-3 respectively.

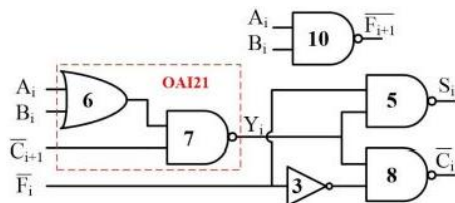


Fig. 3 RPCFA-2 circuit

The detailed study has been carried out on RPCFA-2 circuit as shown in Fig. 3 because it provides minimum delay and consumes least power and energy as compared to the other two RPCFAs. Fig. 4 shows the schematic for RCPFA-2.

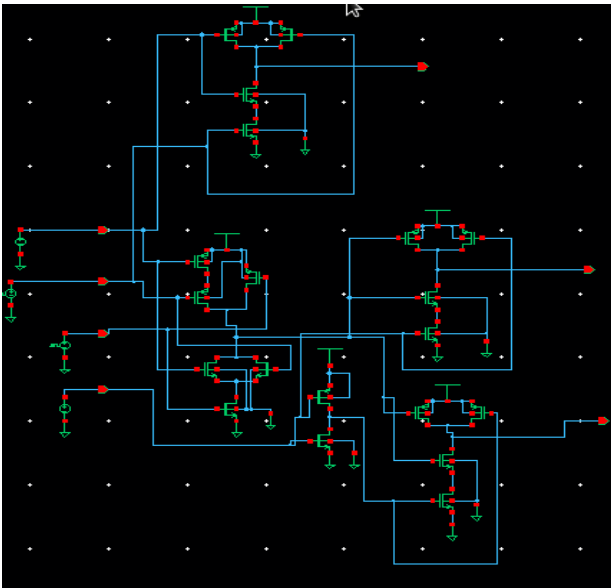


Fig. 4 Schematic for RCPFA-2[1]

III. MODIFIED-GDI TECHNIQUE AND PROPOSED RPCFA CIRCUIT

The basic Modified cell is given in Fig. 5. In this Modified-GDI cell, substrate terminals of NMOS and PMOS transistors are connected to ground and V_{DD} respectively.

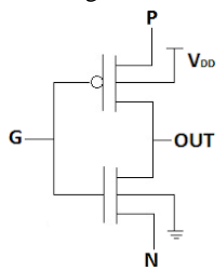


Fig. 5 Basic Modified-GDI cell

This Technique provides considerable reduction of both gate-leakage and sub-threshold currents compared to static CMOS Process.

Table-I shows basic functions that can be performed using Modified-GDI cell.

Table I Basic Functions performed using Modified-GDI Cell

N	P	G	OUT	FUNCTION	T.C. CMOS	T.C. M-GDI
0	B	A	$A'B$	F1	8	2
B	1	A	$A'+B$	F2	8	2
1	B	A	$A+B$	OR	6	2
B	0	A	AB	AND	6	2
C	B	A	$AC+A'B$	MUX	6	2
0	1	A	A'	NOT	2	2

[T.C.-Transistor count]

The RPCFA-2 [1] which was implemented using 20 transistors by standard CMOS Process can now be implemented using only 14 transistors with the help of this technique. While the gate level schematic for proposed RCPFA remains same as that of RCPFA-2[1] (Fig.3).The schematic for proposed RCPFA cell is shown in Fig.6.

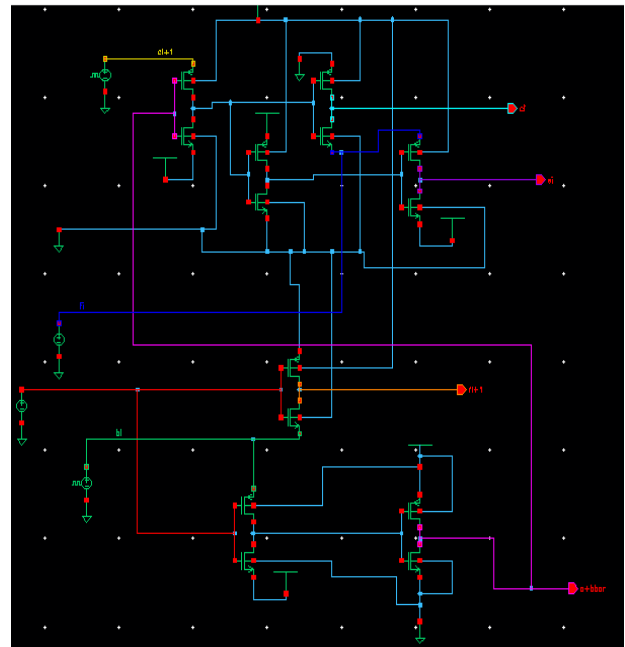


Fig. 6 Schematic for Proposed RCPFA Circuit

IV. SIMULATION RESULTS

By using Modified-GDI cell, same function can be implemented with less transistors as compared to the CMOS process (Table-I).Therefore the number of transistors has been reduced from 20 (in RCPFA-2[1]) to 14 (Proposed RCPFA), and so the power and delay of the proposed circuit have also been reduced considerably. Various delays including carry propagation delay (T_{cp}), carry to summation delay (T_{cs}), forecast signal delay (T_f) and power values are shown in Table II. Simulation results for RPCFA-2[1] and proposed RCPFA are shown in Fig.7 to Fig. 10. All the simulations have been carried out in 45-nm process technology using Cadence Virtuoso for 1V power supply.

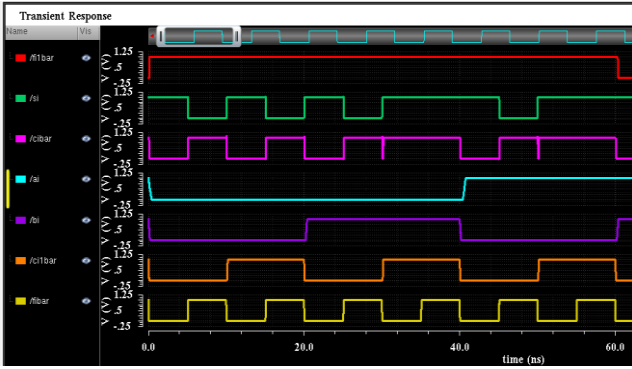


Fig.7 Transient response for RPCFA-2[1]

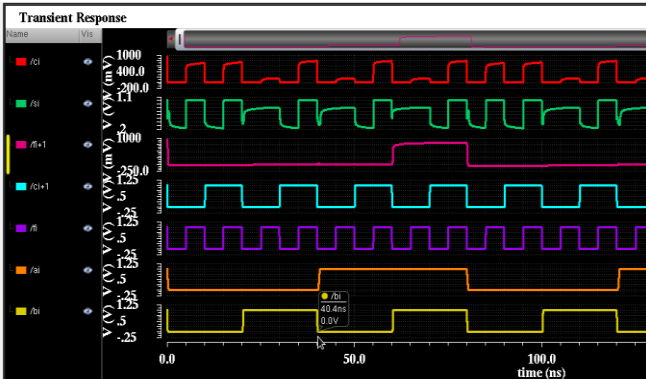


Fig.8 Transient response for Proposed RCPFA Circuit

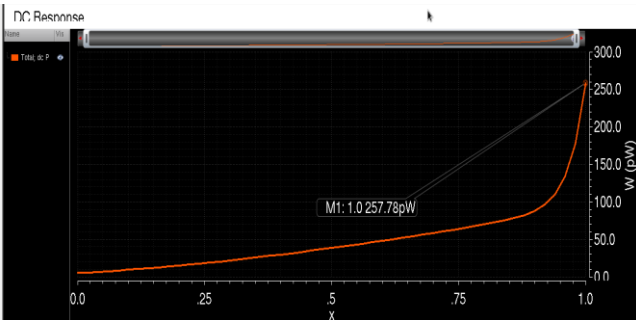


Fig.9 Power Plot for RPCFA-2[1]

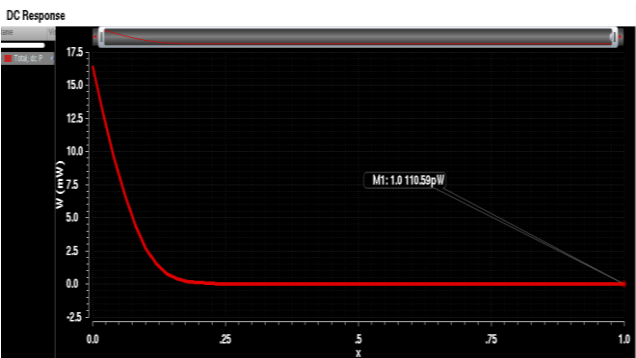


Fig.10 Power Plot for Proposed RCPFA

Table II Comparison between RCPFA-2[1] and Proposed Circuit in terms of delay and power.

FA TYPE	Power (pW)	T_{cp} (ps)	T_{cs} (ps)	T_r (ps)	Number of transistors
RPCFA-2[1]	257.78	124.6	115.5	48.3	20
PROPOSED	110.6	69.56	55.84	20.87	14

V. CONCLUSIONS

In this paper, a new circuit has been proposed for Approximate Reverse Carry Propagate Full Adder based on Modified-GDI technique that propagates carry in counter flow direction from MSB to LSB. The propagation of carry in reverse direction insures higher stability in the delay variations as the weight of the carry decreases when the carry is propagated in a reverse direction. The efficacies of the proposed Reverse Carry Propagate Full Adder and the existing RCPFA-2[1] have been verified using 45-nm technology in Cadence Virtuoso. The results provided in Table II justify that the proposed RCPFA gives on average 57% and 44% improvement in power and delay respectively.

ACKNOWLEDGMENT

Authors would like to thank Mr. Ajishek Raj for his help in making some of the drawings.

REFERENCES

- [1] M. Pashaeifar, M. Kamal, and A. Afzali-Kusha, "Approximate Reverse Carry Propagate Adder for Energy-Efficient DSP Application," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 26, no. 11, pp. 2530–2541, Nov. 2018.
- [2] B. KeerthiPriya and R. Manoj Kumar "A New Low Power Area Efficient 2Bit Magnitude Comparator using Modified GDI technique in Cadence 45nm Technology" in IEEE 2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCT), May 25-27, pp.30-34, 2016.
- [3] N. Zhu, W. L. Goh, W. Zhang, K. S. Yeo, and Z. H. Kong, "Design of low-power high-speed truncation-error-tolerant adder and its application in digital signal processing," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 18, no. 8, pp. 1225–1229, Aug. 2010.
- [4] H. R. Mahdiani, A. Ahmadi, S. M. Fakhraie, and C. Lucas, "Bio-inspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 57, no. 4, pp. 850–862, Apr. 2010.
- [5] V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, "Low-power digital signal processing using approximate adders," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 32, no. 1, pp. 124–137, Jan. 2013.
- [6] Z. Yang, A. Jain, J. Liang, J. Han, and F. Lombardi, "Approximate XOR/XNOR-based adders for inexact computing," in Proc. 13th IEEE Int. Conf. Nanotechnol. (NANO), pp. 690–693, Aug. 2013.

