Research Paper

Volume : 3 | Issue : 12 | December 2014 • ISSN No 2277 - 8179

Optimization of Logarithmic Convertor Using Evolutionary Algorithm

Mathematics

KEYWORDS : leading one detector, leading one position detector,logarithmic convertor, genetic algorithm.

S.Asha Research Scholar, St.Peter's University

Dr. R. Rani Hemamalini Professor, St.Peter's University

ABSTRACT

The logarithmic approach of arithmetic operation shows the ease method of computation. This papers deals with the design of leading one detector, which forms the basic building block for this logarithmic computational approach. First, the design of leading one detector using basic logic gates without multiplexers results in the increased computational speed with significant reduction in gate count and power dissipation. Second, leading one position detector design reports the position of leading one bit without encoder or ROM as in the case of leading one detector. The leading one position detector obtained by this method is efficient in terms of frequency and gate count when compared to the conventional approach. Also the logarithmic convertors designed by using the evolved blocks results in increased performance. The evolutionary implementation of these blocks is carried out using genetic algorithm approach.

I.INTRODUCTION

Logarithm is used for analysing the quantity which varies over a wide range. Also to simplify the complex arithmetic operations (multiplication, division, square root, square etc.,) logarithm is used. For example, multiplication and division operation is reduced to simple additions and subtractions. The binary logarithmic convertor in Mitchell's method consists of leading one detector and a logarithmic shifter [1]. Logarithmic shifter alone is used in antilog convertors. Due to approximations, the results obtained have some amount of error and these types of circuits are used in real time applications such as digital filtering, fast Fourier transforms and other image processing applications which need faster results with acceptable error level and the error can be reduced using the eight sub regions approach[6].

In this paper, binary logarithmic convertors are designed by using the evolved leading one detectors and the delay in the logarithmic shifting operation is also minimized. Section II shed some light on genetic algorithm and section III presents the review of earlier design and the proposed method of leading one detector. Section IV concludes the discussion of the proposed design of leading one position detector, with results comparison. Section V deals with the logarithmic shifter design. Section VI analyses the results and section VII includes conclusion.

Fig 1. Explains the process flow of Mitchell's binary logarithmic convertor.

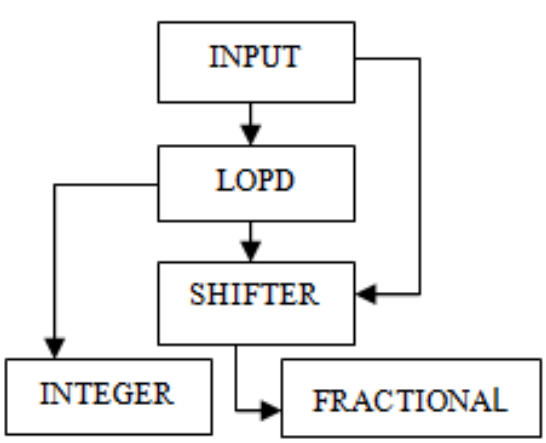

Fig 1: Approximate Mitchell's logarithmic convertor

II.GENETIC ALGORITHM

Genetic algorithm is an adaptive search technique which has been proved effective in solving combinatorial optimisation problem. This search samples the large solution space to find the global maxima or the minimum position based on the fitness measure. An initial random population which represent the part of the solution is chosen. This search continues to visit the various parts of the fitness surface by generating offspring using the convergence and the mutation operations. This kind of genetic evolution is based on the concept that two fittest individual will generate offspring of more fitness. Finally leads to a better optimum solution after finite number of generations [2].

A. Chromosome encoding:

The chromosome pattern defines the interconnections of the circuit. Each chromosomes give different types of circuit. The chromosome is the array of variable size depending upon reconfigurable array [2]. This array is a string of the integers which defines the connections between the elements i.e. the internal connections to gates. The evolution array of programmable nodes is done at both interconnection level and gate level.

The result can be easily converged by using the coarse grained architectures instead of using the fine grained architectures. So for the evolution of the large circuits with coarse grained architecture is easy compared to the fine grained type. In this type both the coarse and fine grained structures are used.

Table 1: Gate utilisation for the design

TYPE	INPUTS	STRUCTURE
1	A	NOT
2	A,B	AND
3	A,B	OR
4	A,B	NAND
5	A,B	NOR
6	A,B	AND(NOT(A),B
7	A,B,C	OR(AND(A,B),C)

B.FITNESS FUNCTION

The fitness function is the required design to be obtained. For each chromosome the fitness function is calculated. The fitness measure is a measure between the fitness of the each chromosome with the required fitness. The chromosome with the the better fitness is chosen for the next generation and the process is repeated until the desired chromosome is obtained with the required fitness.

C. GENETIC OPERATORS

The mutation and crossover operators are the genetic operators. The mutation operator is used in this paper instead of crossover

operator because the convergence of the result is faster in mutation process and it is easier when compared to the crossover operator. At sometimes the convergence of the result also depends upon the mutation rate.

III. LEADING ONE DETECTOR

In binary logarithmic circuits, the primary operation is to determine the integer and the fractional parts. The position of the leading one bit represents the integer part of the logarithm. The fractional part is obtained by shifting the input using the output from the LOD. So it is necessary to design a circuit which detects the leading one bit position with less hardware and consumes low power with fast speed. The LOD produces bit one at the output, while keeping its position, and circuit sets all other output bits to zero. The LOD is evolved using the genetic algorithm.

The leading one detector proposed in paper [4] uses the multiplexers as the major component. In case of using the multiplexers, one input is always grounded and the other input only connected. For example mux (A, 0) is equivalent to AND operation of input A with complement of the control word. In proposed method the additional AND gate and OR gate gets reduced and the critical path time also gets reduced.Therefore the use of logic gates in the leading one detector is efficient in terms of speed, is and power consumption.

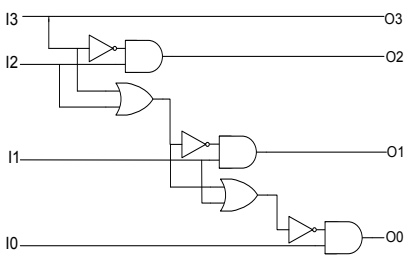

Fig 2: Evolved four bit LOD

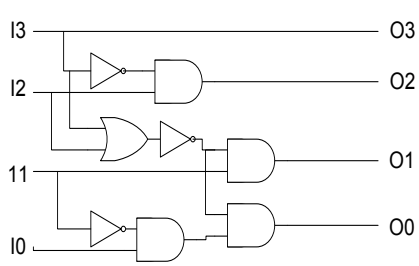

Fig 3: Evolved four bit LOD

It is necessary to decode the position of the leading one detector to obtain the integer part. So the ROM is needed or the encoder is used to encode the position of the leading one bit. To avoid this complexity the leading one position detector is designed which gives the position of the leading one bit and having the advantage of ROM free operation [5].

Table 2: Gate count and frequency

LOD TYPE	GATES USED	GATE COUNT	FREQUENCY
4 BIT	1.MUX	18	787Mhz
	2.AND,NOT	18	2.85Ghz
	3.AND,OR,NOT	18	2.85Ghz
8 BIT	MUX	69	584Mhz
16 BIT	4 BIT LOD(1)	120	336Mhz
	4 BIT LOD(2)	120	480Mhz
	4 BIT LOD(3)	117	502Mhz
	8 BIT LOD	96	478Mhz
32 BIT	16 BIT LOD	183	325Mhz

The higher order LODs are designed using the four bit LODs and final output is combined by using the AND gates. First stage is grouping of input bits into four and gives it to the LOD. Then group the two LOD outputs and this process continues till the final output is reached.

The results obtained for the LODs of different size are shown in the table. The result shows that the gate count and critical path time also decreases in the proposed method compared to the earlier method. Thus the frequency of operation also increases in the proposed method.

IV LEADING ONE POSITION DETECTOR:

To obtain the integer part of the logarithmic value, the position of the leading one bit must de detected.The position of the leading one bit gives the integer part of the logarithmic convertor. So it is necessary to design a leading one position detector [5] with high speed and reduced gate structure because this integer part serves as a control word to shift the input to get the fractional part. The flag bit is generated by doing OR operation of all the input bits. If the flag bit is zero then the input values contain no zeros in it otherwise it contains one and the position of the leading one id detected by the LOPD.

The leading one position detector is evolved by using the genetic algorithm and many number of different structures are obtained for the same function and the circuit with less number of gates and less critical path time are chosen for the logarithmic circuits.

The leading one position detector gives the position of the leading one bit and the circuits obtained by the process of evolution are shown below.

The Y2 represents the MSB of the output and is given by

Y2=X7+X6+X5+X5.

The evolved structures for Y1 are,

Y11=(X7+X6) + (X5+X4)'(X3+X2)

Y12=(X7+X6+X5+X4)(X7+X6)

+ (X7+X6+X5+X4)'(X3+X2)

Y1P=(X7+X6+X3+X2) + ((X7+X6+X5') (X7+X6+X4))

where Y11, Y12 are the evolved structures and Y1P is the conventional structure.

The evolved structures for Y0 are,

Y01=(X7+X6+X5+X4)(X7+X5(X7+X6)')
+(X7+X6+X5+X4)' (X3+X1(X3+X2)')

Y02=X7+X6'(X5+X4'(X2'X1+X3))

Y03=X7+(X7+X6)'(X5+X4'(X3+X1(X2+X3)'))

Y0P=X7+(((X5+X4)'(X3+X2)'+(X1+X3))(X5)

(X7'+X6))

Volume : 3 | Issue : 12 | December 2014 • ISSN No 2277 - 8179

where Y21, Y22, Y23 are the evolved structures and Y2P is the conventional structure.

The results obtained from the evolved equation shows that the performance of the evolved structure increases as the number of input bit increases and it shows better performance when compared to the conventional methods. The results of the structures are listed in the table and this shows that in higher order LOPDs, the evolved design is more efficient when compared to the conventional method. In the table for each structure, the first row represents the critical path time or the maximum combinational delay and the second represents the gate count for each structure.

For example, the table shows the results of different structures for the 8 bit LOPD and there is mild difference between the conventional and the proposed method. But in table, the results are for 16 bit, 32 bit, 64 bit for different structures are there. These results reveal that the performance of higher order LOPDs is better compared to the conventional method.

Table 3: 8 bit and 16 bit LOPD

LOPD	8 BIT		16 BIT	
	gates	Time(ns)	gates	Time(ns)
Y11,Y21	60	1.18	131	2.64
Y11,Y22	61	1.21	154	2.84
Y11,Y23	60	1.05	148	2.29
Y11,Y2P	57	1.05	135	1.83
Y12,Y21	48	1.78	129	3.42
Y12,Y22	48	1.64	122	1.83
Y12,Y23	48	1.64	125	2.44
Y12,Y2P	42	1.64	119	2.43
Y1P,Y21	62	1.13	139	2.13
Y1P,Y22	62	0.98	145	3.25
Y1P,Y23	61	0.98	148	1.67
Y1P,Y2P	58	0.98	143	2.23

Table 4: 32 bit and 64 bit LOPD

LOPD	32 BIT		64 BIT	
	gates	Time(ns)	gates	Time(ns)
Y11,Y21	311	2.58	673	4.42
Y11,Y22	341	3.04	733	4.42
Y11Y23	348	2.73	745	4.42
Y11,Y2P	328	2.83	707	3.59
Y12,Y21	322	3.71	693	4.42
Y12,Y22	308	2.45	666	4.42
Y12,Y23	321	3.21	692	4.42
Y12,Y2P	297	3.42	642	4.42
Y1P,Y21	341	2.52	733	4.42
Y1P,Y22	346	3.76	741	4.42
Y1P,Y23	354	2.43	758	4.42
Y1P,Y2P	351	2.81	753	4.42

The use of 4 to 1 multiplexers results in more number of gates and the critical path delay is also more when compared to using the 2 to 1 multiplexers. Thus the use of 2 to 1 multiplexers is more efficient compared to the 4 to 1 multiplexers. For example the 32 bit LOPD by using 2 to 1 multiplexers is better than the 4 to 1 multiplexers. The results are shown in the table for 32 bit in table and table.

Table 5: 32 bit LOPD using 2 to 1 and 4 to 1 multiplexers

LOPD	32 BIT		64 BIT	
	gates	Time(ns)	gates	Time(ns)
Y11,Y21	311	2.58	300	4.72
Y11,Y22	341	3.04	348	4.23
Y11Y23	348	2.73	318	4.18
Y11,Y2P	328	2.83	295	3.96
Y12,Y21	322	3.71	316	3.95
Y12,Y22	308	2.45	339	4.43
Y12,Y23	321	3.21	335	3.79
Y12,Y2P	297	3.42	309	4.69
Y1P,Y21	341	2.52	323	4.16
Y1P,Y22	346	3.76	347	4.40
Y1P,Y23	354	2.43	334	3.80
Y1P,Y2P	351	2.81	330	4.26

Fig.evolved four bit LOPD

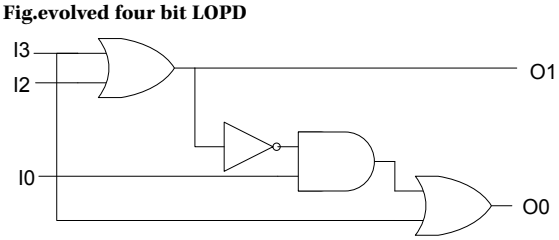

Fig.evolved four bit LOPD

The higher order LOPDs is obtained by using the lower order LOPDS. The first step is grouping of eight bit and finding the leading one position for each group. Next step is converging of two groups by using the control bit. If the control bit is one, the first group is selected otherwise the next and the same process is repeated until all the groups are converged. The control bit is the one which determines the selection of the divided input pattern. The control bit is generated by doing OR operation for the first half of the input.

V.LOGARITHMIC SHIFTER

The logarithmic shifter used in the logarithmic convertor consists of the multiplexer based structure. There is no shift operation when the control bit is '1' and the shift operation takes place when the control bit is '0'. The amount of shifting depends upon the weighting position of the bit. The shift operation is divided into the number of stages and the output from the present stage is given as the input to the next stage. For the each stage the control bit is given from MSB to LSB of the integer part. Instead of using the shifter, the input is directly connected to the output by appending the zeros at the lower bits. For example the eight bit number is shifted by four times if the control word is zero then the last four bits are directly connected to the MSB and the LSB positions are filled with zeros.

VI.RESULTS AND DISCUSSION

The leading one detectors proposed in the paper is having high speed of operation with the reduction in the gate count com-

pared to the previous methods. The leading one position detectors shows that the evolved structures are more efficient in terms of area and speed when compared with the conventional method. The gates get reduced in the higher order structures by using this evolved leading one position detectors. The logarithmic convertors designed using these evolved structure are efficient than compared to the previous methods and the results are shown in the table.

Table 6: Logarithmic convertors using evolved structures

Log convertors	Gate count	Time(ns)
8 bit	203	2.21
16 bit	487	4.43
32 bit	1297	5.25
64 bit	2973	8.05

VI.CONCLUSION

The proposed method of leading one detector and leading one position detectors are very efficient in terms of area, power and speed. The various structures evolved using these approaches are synthesised using ASIC library SCL05. The error statistics of this proposed method comply with that of Mitchell's method and this concept can be further extended by using the 8 regions approach. The logarithmic shifter does the shifting operation at a higher speed. Logarithmic convertors designed using these blocks has reduction in the gate count and the combinational delay.The antilogarithmic convertors are obtained only by shifting operations. Thus the evolved logarithmic blocks used in the logarithmic convertors show better performance compared to the conventional methods.

REFERENCE

[1] K. H. Abed, R. E. Siferd," CMOS VLSI implementation of a low-power logarithmic converter", IEEETrans on Computers, , Vol.52, No. 11, pp. 1421 – 1433,2003 | [2]. Vesselin K. Vassilev, Julian F. Miller, Terence C. Fogarty, "Towards the automatic Design of More Efficient Digital Circuits," proceedings of second NASA/DOD workshop on Evolvable Hardware, 2000. | [3]. Zhijun Li, Jianping An, Miao Yang, Jing Yang,"FPGA Design and Implementation of an Improved 32-bit Binary Logarithm Converter", IEEE conference, 2006. | [4]. Khalid H. Abed Raymond E. Siferd,"VLSI Implementations of Low-Power Leading-One Detector Circuits",IEEE conferences, 2006. | [5] Tso-Bing Juang, Sheng-Hung Chen, Huang-Jia Cheng" A Lower Error and ROM-Free Logarithmic Converter for Digital Signal Processing Applications"IEEE transactions on circuits and systems—ii: express briefs, vol. 56, no. 12, December 2009. | [6]Davide De Caro, Nicola Petra, , andAntonio G. M. Strollo,"Efficient Logarithmic Converters for DigitalSignal Processing Applications" IEEE transactions on circuits and systems—ii: express briefs, vol. 58, no. 10, October 2011. |