International Journal of Research in Engineering and Applied Sciences(IJREAS)
Available online at http://euroasiapub.org/journals.php
Vol. 7 Issue 9, September-2017, pp. 1~9
ISSN (O): 2249-3905, ISSN(P): 2349-6525 | Impact Factor: 7.196

A Class of Fault Tolerant Ling Parallel Prefix Adders with Low Overhead

Asma Iqbal[1],
Deccan College of Engineering & Technology
Hyderabad, Telangana, India

Dr. K. Manjunathachari[2],
GITAM School of Technology, GITAM University
Hyderabad, Telangana, India

Abstract

Prefix Adders are often susceptible to defects or failures. Therefore, incorporating redundancy into the design is important to make them more robust. While conventional schemes apply redundancy, more recently fault localization and selective spatial redundancy have been tried. But they have their fair share of hardware overhead. In this work, a new approach to fault tolerance in parallel prefix structures which takes advantage of certain features of Ling adders is proposed. Results prove that the proposed scheme works 15% faster when compared to the existing prefix adder schemes

Keywords: Fault tolerance, fault localization, redundancy, Parallel Prefix adders, Ling Adders

INTRODUCTION

It was predicted by Moore that the number of transistors per unit area would approximately double every ten years [1]. This trend has been observed and the result is very densely fabricated ICs. Due to this, occurrence of faults and failures also becomes more probable in the arithmetic circuits thereby reducing the yield [2]. The common reasons for occurrence of faults include process variations and manufacturing defects [3]. Process variations that effect arithmetic circuits such as prefix adders can be addressed to a large extent by incorporating fault tolerance in the design. Early fault tolerant schemes worked on simple redundancy, such as Triple Modular Redundancy (TMR) [4] and Time Shared Triple Modular Redundancy (TSTMR) [5]. However, fault tolerance while offering robustness also results in area and delay overhead. In view of issues such as area and/or delay overhead that are inherent to redundancy, new schemes that take advantage of inherent fault localization and spatial redundancy have been proposed. Efficient use of the above schemes individually [4], [5] or as a combination has been proposed [3], [6]. In [3] the inherent regular structure of a Kogge-Stone (KS) carry lookahead adder (CLA) is utilized to obtain

fault tolerance with minimum area overhead. However, CLA based Parallel Prefix Adders (PPA) like Brent-Kung (BK), Ladner-Fischer (LF) and Sklansky (SS) do not exhibit a regular structure as Kogge-Stone (KS) and consequently implementing fault tolerance in these structures requires large die-area. This paper presents a technique for incorporating fault tolerance in adder structures using the inherent even-odd independence which the parallel prefix Ling adder exhibits. The proposed scheme with its implementation is explained in the next section followed by Mathematical analysis in terms of area overhead and time optimization. Finally the results and discussions are summarized.

PREVIOUS WORK

Fault tolerance is achieved by anticipating faults and duplicating critical hardware for corrective action. Earlier techniques used for implementing fault tolerance in adder schemes are based on hardware redundancy and time redundancy. In [4] the authors suggest a simple scheme to obtain fault tolerance by triplicating the adder module. As illustrated in Fig.1 the output of these individual adder blocks are given to the voter circuit which generates the majority signal. This scheme referred to as Triple Modular Redundancy (TMR), requires a large area as three adder modules are used.

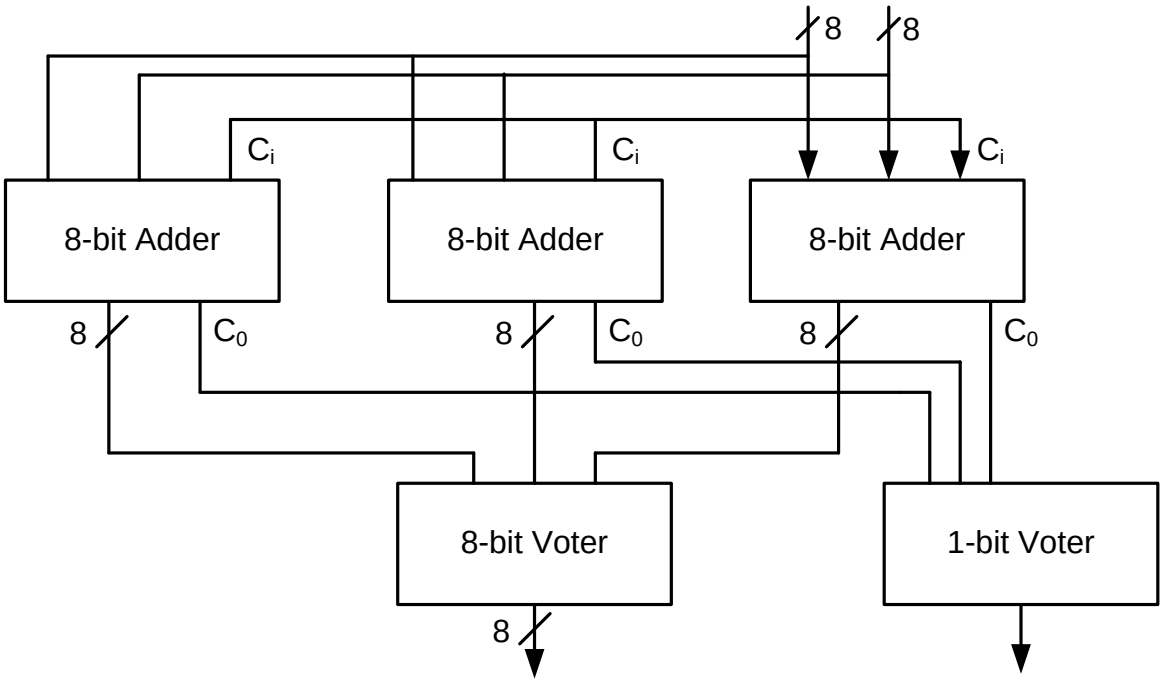

Fig. 1. 8-bit Adder with Triple Modular Redundancy (TMR)

Figure 2 illustrates Time Shared Triple Modular Redundancy (TSTMR) scheme proposed by [5] wherein the operands as well as the adder is divided into three parts. Each part of the operand is sent to the adder via a multiplexer with the least significant part being sent first and the most significant part last. After three cycles the entire sum is computed. A 1-bit cell is used to store the

intermediate carry. Although, TSTMR scheme takes up less area, the delay overhead is significant due to three cycle operation and multiplexer/voter delays. Both these methods mask the effects of faults and thereby increase the overall system reliability, however at the cost of either hardware or delay. This was overcome through fault localization and spatial redundancy techniques. Fault localization due to the mutually exclusive even-odd structure of the Kogge-Stone adder is utilized in [3] to obtain fault tolerance within a minimum area but with delay overhead. This technique is not viable in the case of the other CLA based Parallel Prefix Adders like Brent-Kung (BK), Sklansky(SS) and Ladner- Fischer(LF) as they do not exhibit the mutually exclusive even–odd structure. Fault tolerance was obtained in CLA based prefix structures by a generalized technique as mentioned in [6] using selective redundancy and adaptive clocking. In this method, critical nodes were identified and duplicated along with additional hardware. In the event of fault, the addition operation requires more than one cycle to complete the task. However, the redundant circuitry adds to the hardware complexity and also increases the critical path delay. Thus, it is evident that the CLA based prefix adder schemes have area overhead. However, the proposed approach uses the Ling based prefix adders that exhibit even-odd independence across all architectures. The same is given in detail in the section mentioned below.

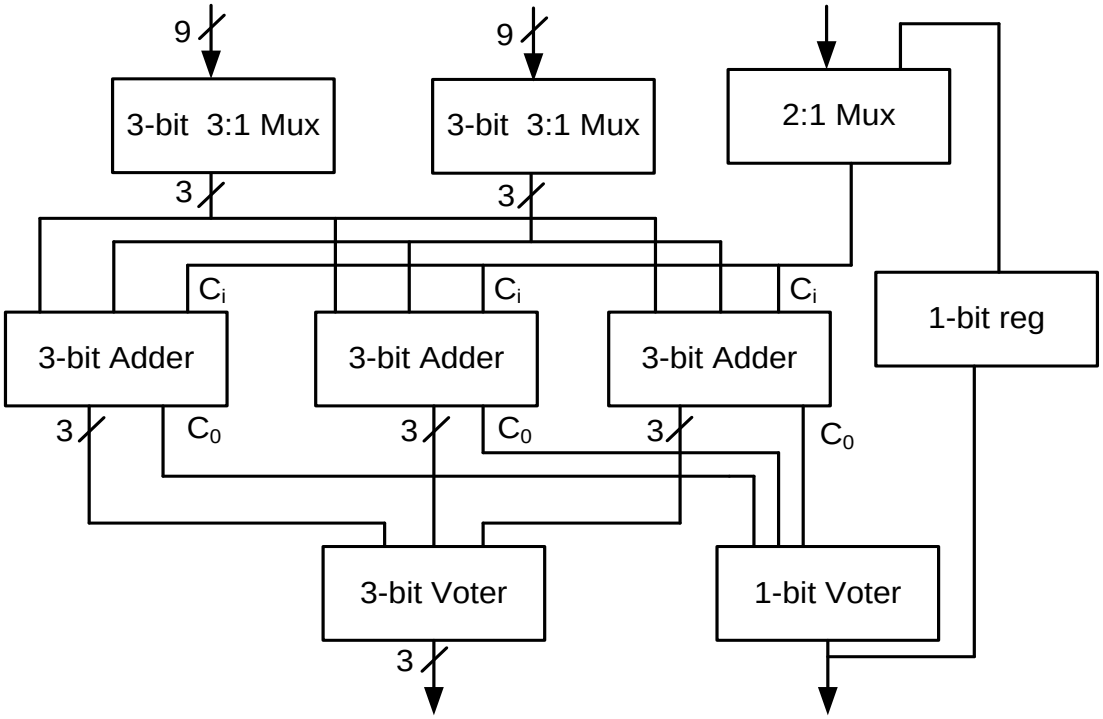

Fig. 2. 8-bit Adder using Time Shared TMR (TSTMR)

PROPOSED WORK

This section presents the mathematical equations of the Ling parallel prefix adder [7] showing inherent even-odd independence. Then we discuss how this independence can be exploited to

obtain fault tolerance with minimum area overhead and delay optimization.

a- Even odd independence in Ling adders

Ling adders [7] are multi level carry look-ahead adders (CLA) with modifications to save hardware. The modifications suggested by Ling were to propagate Ling carry, hi that are formed by considering adjacent carry signals ci and ci−1 instead of the output carry generated ci .

$h_i = c_i + c_{i-1}$ (1)

The i th Ling Carry can be expressed as:

$H_i = g_i + g_{i-1} + p_{i-1}.g_{i-2} + \ldots + p_{i-1}.p_{i-2} \ldots p_1.g_0$ (2)

$G^*_i = g_i + g_{i-1}$ (3)

$P^*_i = p_i .p_{i-1}$ (4)

Carry computation can be transferred into a prefix structure using the 'o ' operator which associates pairs of generate and propagate bits as

$(g, p)o(g', p') = g + p.g', p.p'$

$H_{2i} = (G^*_{2i}, P^*_{2i-1})\ o\ (G^*_{2i-2}, P^*_{2i-3})o \ldots o(G^*_0, P^*_{-1})$ (5)

$H_{2i+1} = (G^*_{2i+1}, P^*_{2i})o(G^*_{2i-1}, P^*_{2i-2})o...o(G^*_1, P^*_0)$ (6)

Where $G^*_k = P^*_k = 0$ for all $k < 0$

The equations (5) and (6) indicate that even and odd set of bits are mutually independent, hence confining a fault to either of these sets. To illustrate the even-odd independence, for example, an 8-bit Ladner-Fischer Ling prefix structure is shown in Fig.3. It consists of black computation cells to produce generate and propagate terms. The structure has 8 columns with the i^{th} column computing the ith Ling carry. It can be observed that there are no nodes in the even columns feeding nodes of odd columns and vice-versa.

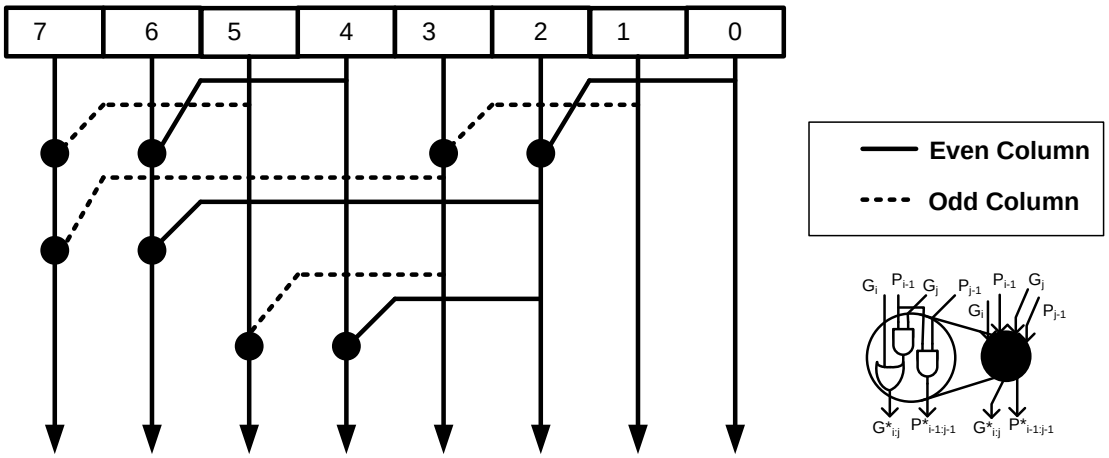

Fig. 3. 8-bit Ladner-Fisher Ling Prefix Structure

b-Fault tolerance in Ling adders

In Fig.4 (a) the block diagram of a general self correcting CLA based PPA [6] is shown. It consists of input shift logic, pre-computation block, CLA prefix structure, post computation block and output shift logic. The CLA prefix structure includes duplicated important nodes (computation cells) and multiplexers to select the fault free node. This is the protected part of the prefix structure meaning a fault in this part can be masked and a fault free output is obtained. The other nodes which are not duplicated form the unprotected part of the prefix structure meaning a fault in this part may result in erroneous results. In case a fault is detected in the protected part the computation is done in multiple clock cycles. In the first cycle, one set of operands are applied and outputs are computed while in subsequent cycles the other outputs are calculated by shifting the inputs and using the fault free part of the PPA. To achieve this additional column(s) (for computation of shifted bits) are added. Depending on the architecture one or two additional columns are added. This has its fair share of drawbacks such as delay overhead (as computation in the faulty adder may require two or three clock cycles) and vulnerability to faults in the unprotected part of the prefix structure. It is proposed in this work to overcome these weaknesses in CLA based prefix adders to the extent possible by means of a fault tolerant self-correcting structure using Ling parallel prefix adders.

Figure. 4(b) gives the block diagram of the proposed self-correcting Ling adder consisting of input shift logic, pre-computation block, Ling prefix structure with no unprotected part, post computation block and output shift logic. An additional column is added for computation of the shifted bit. Mutual independence of even and odd bits can be extended to fault localization in Ling architecture. Hence a fault in the odd part effects only the odd carries and the even part computes the sum correctly. The same holds true for a fault in the even carries of the Ling PPA. Additional hardware is incorporated at design time to utilize this property of Ling adders so as to obtain a class of fault tolerant selfcorrecting adder structure by the introduction of bit shift scheme [3]. Shifting is facilitated by a single redundant column to account for the MSB while registers store partial sums with multiplexers added to make sure the partial sums are stored correctly. A single clock cycle would be sufficient for the adder to carry out computations if the adder is found to be faultfree after manufacturing tests are carried out. In case a fault is detected, a shift enable signal is asserted by a designated circuitry and the operation carried out in two cycles. In cycle- 1 the fault-free bits are computed (even or odd) and stored in the output register. In cycle-2, a shift operation is initiated and then the other set of bits (odd or even) are computed. This technique additionally uses multiplexers at the input and output for shifting operands and storing the correct output [3].

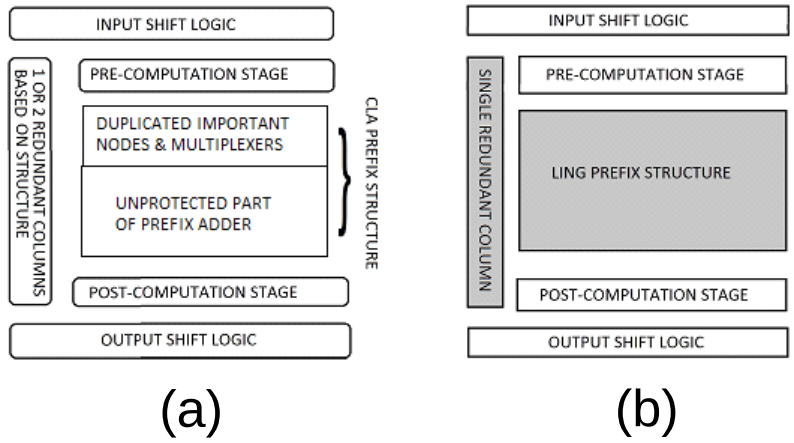

Fig. 4. (a) An existing structure the self-correcting CLA network (b) Overall structure of self-correcting Ling adder network

To understand the hardware requirements of fault tolerant Ling adders we begin with the 8-bit Ladner-Fischer PPA as shown in Fig.5. The prefix structure of a Ling PPA comprises of black cells only, which compute group generate and propagate. The column 8 highlighted with grey color is the redundant column. Apart from the prefix stage, the Ling structure also saves in terms of the logic required for the redundant column. Due to the symmetry of bits, one redundant column will suffice for all the adder structures unlike in [6] which use two columns in most of the CLA structures. All such considerations, pertaining to area and delay are consolidated in the next section.

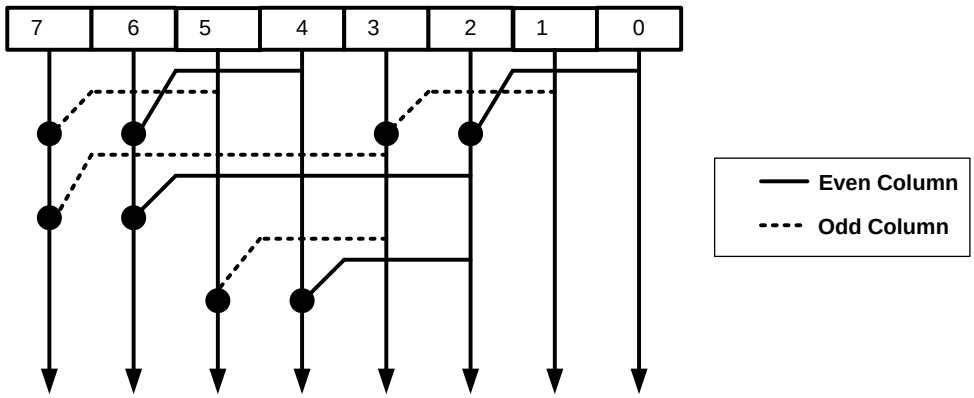

Fig. 5. An 8-bit fault tolerant Ling PPA (Ladner-Fischer)

RESULTS

In this paper the TMR [4], TSTMR [5], CLA based PPA [6] and Ling based PPAs have been compared. All the designs have been analyzed using unit gate modeling as this approach provides

a decent model for estimating the required design metrics of each component and does not depend strongly on any synthesis tool. Assumptions made are each two-input gate (AND, OR, NAND, NOR) is counted as one gate while Multiplexer, EX-OR and EX-NOR are counted as two gates, for both area and delay. In the proposed design the extra logic that comes in the precomputation stage has been assumed to be fault free [6], since it occupies significantly less area in comparison to redundant cells. The unit gate modeling of 16- bit fault tolerant adder using TMR, TSTMR, PPA and Ling PPA are compared. Design metrics area (A), delay (D), and area-delay product (A*D) have been considered and compared for all the designs.

Table I compares the area overhead and delay involved in the existing [4], [5], [6] and proposed 16–bit fault tolerant adder topologies. The area-delay product for TMR, TSTMR, CLA and proposed Ling based schemes is also included. It can be observed from the Table I that the fault tolerant Parallel Prefix adders based on CLA or Ling carries have a distinct advantage over TMR and TSTMR in terms of delay involved whereas TSTMR has an advantage in area overhead. If area delay product is considered then the Parallel Prefix Adders have a clear advantage. If we consider the CLA based PPA and the Ling based PPA the improvement in delay is 17%, 8% and 34% in SS, LF and BK structures respectively. The area overhead in the proposed design is 13%, 30% and 18% in SS, LF and BK respectively. Overall improvement in the area-delay product is 33%, 5% and 43% in SS, LF and BK respectively and can be observed from Table I.

CONCLUSIONS

This paper presents a scheme to obtain fault tolerance in adders by using the Ling Parallel Prefix structures by adding a minimum amount of hardware and time redundancy. The results above indicate a considerable improvement in the delay involved. There is an improvement in delay of 8% to 34% for various architectures when compared to the existing CLA based design.

TABLE – 1. A comparison of area, delay and area-delay product of various fault tolerant adders

Fault tolerant adder architecture		Delay (D)		Area (A)		Area-Delay product (A*D)	
		Units	Improvement(%)	Units	Overhead (%)	Units	Improvement(%)
TMR[3]		35	100	180	100	6300	100
TSTMR[4]		51	146	160	89	8160	130
Sklansky PPA	CLA[6]	18	52	296	165	5328	85
	Proposed Ling	12	35	273	152	3276	52
Ladner-Fischer PPA	CLA[6]	15	43	321	179	4815	77
	Proposed Ling	12	35	376	209	4512	72
Brent-Kung PPA	CLA[6]	24	69	258	144	6192	99
	Ling	12	35	291	162	3492	56

If the area-delay product is considered there is an overall improvement in all the architectures with the maximum improvement in the Brent-Kung PPA where it is 43% and minimum being 5%. It can be said a judicious selection of architecture can give a considerable improvement in the overall performance of adders.

REFERENCE

] R. R. Schaller, (1997): "Moore's law: past, present and future In: IEEE spectrum, vol. 34, no. 6, pp. 52–59

N. Campregher, P. Y. Cheung, G. A. Constantinides, and M. Vasilko, (2005): Analysis of yield loss due to random photolithographic defects in the interconnect structure of fpgas. In Proceedings of the 2005 ACM/SIGDA 13th international symposium on Field-programmable gate arrays. ACM, pp. 138–148

S. Ghosh, P. Ndai, and K. Roy, (2008): A novel low overhead fault tolerant kogge-stone adder using adaptive clocking," In Design, Automation and Test in Europe, DATE'08. IEEE, pp. 366–371.

J. Barry, (1989): Design and analysis of fault tolerant digital systems.

] Y.-M. Hsu, (1995): Concurrent error correcting arithmetic processors. Ph.D. dissertation, University of Texas at Austin.

S. Ghosh and K. Roy, (2011): Novel low overhead post-silicon self-correction technique for parallel prefix adders using selective redundancy and adaptive clocking. In IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 19, no. 8, pp. 1504–1507.

G. Dimitrakopoulos and D. Nikolos, (2005): High-speed parallel-prefix vlsi ling adders. In IEEE Transactions on Computers, vol. 54, no. 2, pp. 225–231.

Biography of Authors

	Mrs. Asma Iqbal did her B.Sc. Engg(Electronics) from Aligarh Muslim University and M.Tech(DSCE) from JNTU(H). She is currently working as an Associate Professor at DCET, Hyderabad. She has a teaching experience of more than 18 years.
	Dr. K. Manjunathachari did his B.E.(ECE) from Gulbarga University, M.Tech(DSCE) from JNTU(H) and then his Doctorate from JNTU(K). He is currently Professor and Head of ECE department at GITAM School of Technology, Hyderabad campus. He has more than 20 years of teaching and industrial experience. He has published 8 papers in national and international conferences and journals.