International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 02 Issue: 09 | Dec-2015 www.irjet.net p-ISSN: 2395-0072

DESIGN AND IMPLEMENTATION OF HYBRID VARIABLE LATENCY CARRY SKIP ADDER

Mr.V.Govindaraj,M.E.,(Ph.D)[1], R.Biruntha[2], S.Suganya[3]

[1] *Professor, Department of ECE, KPRIET, Tamilnadu, India*
[2] *PG Scholar, Department of ECE, KPRIET, Tamilnadu, India*
[3] *PG Scholar, Department of ECE, KPRIET, Tamilnadu, India*

Abstract - ***This paper present a carry skip adder (CSKA) structure that has a higher speed yet lower power consumption compared with the conventional one. The speed enhancement is achieved by applying concatenation and incrementation schemes to improve the efficiency of the conventional CSKA (Conv-CSKA) structure. In addition, instead of utilizing multiplexer logic, the proposed structure makes use of AND-OR-Invert (AOI) and OR-AND-Invert (OAI) compound gates for the skip logic. The structure may be realized with both fixed stage size and variable stage size styles. Finally, a hybrid variable latency extension of the proposed structure, which lowers the power consumption without considerably impacting the speed, is presented. This extension utilizes a modified parallel structure for increasing the slack time. Simulations on the proposed hybrid variable latency CSKA reveal reduction in the power consumption compared with the latest works in this field while having a reasonably high speed.***

Keywords
Carry skip adder (CSKA), Power efficient, High performance, Hybrid variable latency adders.

1. INTRODUCTION

Adders are a key building block in arithmetic and logic units (ALUs) and hence increasing their speed and reducing their power/energy consumption strongly affect the speed and power consumption of processors. Obviously, it is highly desirable to achieve higher speeds at low-power/energy consumptions, which is a challenge for the designers of general purpose processors.

one may choose between different adder structures/families for optimizing power and speed. There are many adder families with different delays, power consumptions, and area usages. Examples include ripple carry adder (RCA), carry increment adder (CIA), carry skip adder (CSKA), carry select adder (CSLA), and parallel prefix adders (PPAs). The RCA has the simplest structure with the smallest area and power consumption but with the worst critical path delay. In the CSLA, the speed, power consumption, and area usages are considerably larger than those of the RCA. The PPAs, which are also called carry look-ahead adders, exploit direct parallel prefix structures to generate the carry as fast as possible. There are different types of the parallel prefix algorithms that lead to different PPA structures with different performances. As an example, the Kogge–Stone adder (KSA) is one of the fastest structures but results in large power consumption and area usage. It should be noted that the structure complexities of PPAs are more than those of other adder schemes.

The CSKA is an efficient adder in terms of power consumption and area usage. The critical path delay of the CSKA is much smaller than the one in the RCA, whereas its area and power consumption are similar to those of the RCA. In addition, the power-delay product (PDP) of the CSKA is smaller than those of the CSLA and PPA structures. In addition, due to the small number of transistors, the CSKA benefits from relatively short wiring lengths as well as a regular and simple layout. The comparatively lower speed of this adder structure, however, limits its use for high-speed applications.

In this paper, given the attractive features of the CSKA structure, we have focused on reducing its delay by modifying its implementation based on the static CMOS logic. The proposed modification increases the speed considerably while maintaining the low area and power consumption features of the CSKA. In addition, an adjustment of the structure, based on the variable latency technique, which in turn lowers the power consumption without considerably impacting the CSKA speed, is also presented. The design of (hybrid) variable latency CSKA structures have been reported in the literature. Hence, the contributions of this paper can be summarized as follows.
1) Proposing a modified CSKA structure by combining the concatenation and the incrementation schemes to the conventional CSKA (Conv-CSKA) structure for enhancing the speed and power efficiency of the adder. The modification provides us with the ability to use simpler carry skip logics based on the AOI/OAI compound gates instead of the multiplexer.

2) Providing a design strategy for constructing an efficient CSKA structure based on analytically expressions presented for the critical path delay.
3) Proposing a hybrid variable latency CSKA structure based on the extension of the suggested CSKA, by replacing some of the middle stages in its structure with a PPA, which is modified in this paper.

2.CONVENTIONAL CARRY SKIP ADDER

The structure of an *N*-bit Conv-CSKA, which is based on blocks of the RCA (RCA blocks), is shown in Fig. 1

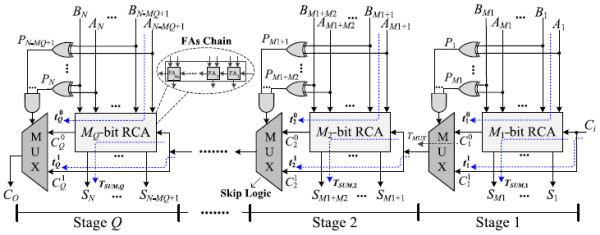

Fig -1: Conventional structure of the CSKA

In addition to the chain of FAs in each stage, there is a carry skip logic. For an RCA that contains *N* cascaded FAs, the worst propagation delay of the summation of two *N*-bit numbers, *A* and *B*, belongs to the case where all the FAs are in the propagation mode. It means that the worst case delay belongs to the case where
$Pi = Ai \oplus Bi = 1$ for $i = 1, \ldots, N$
where *Pi* is the propagation signal related to *Ai* and *Bi*. This shows that the delay of the RCA is linearly related to N [1].

In the case, where a group of cascaded FAs are in the propagate mode, the carry output of the chain is equal to the carry input. In the CSKA, the carry skip logic detects this situation, and makes the carry ready for the next stage without waiting for the operation of the FA chain to be completed. The skip operation is performed using the gates and the multiplexer shown in the figure. Based on this explanation, the *N* FAs of the CSKA are grouped in *Q* stages. Each stage contains an RCA block with *Mj* FAs ($j = 1, \ldots, Q$) and a skip logic. In each stage, the inputs of the multiplexer (skip logic) are the carry input of the stage and the carry output of its RCA block (FA chain). In addition, the product of the propagation signals (*P*) of the stage is used as the selector signal of the multiplexer. The CSKA may be implemented using FSS and VSS where the highest speed may be obtained for the VSS structure [2].

3. MODIFIED CSKA STRUCTURE

Based on the discussion presented in Section 2, it is concluded that by reducing the delay of the skip logic, one may lower the propagation delay of the CSKA significantly. Hence, in this paper, we present a modified CSKA structure that reduces this delay.

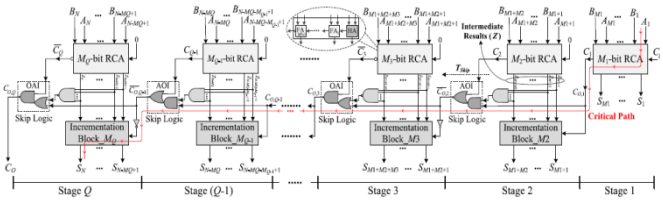

Fig -2: Proposed CI-CSKA Structure

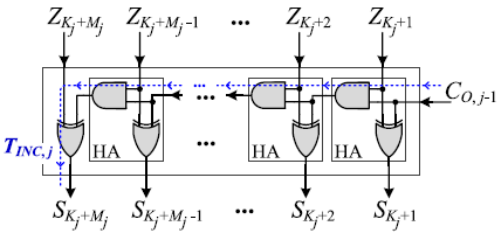

Fig-3: Internal structure of the *j* th incrementation block

3.1 General Description of the Proposed Structure

The structure is based on combining the concatenation and the incrementation schemes with the Conv-CSKA structure, and hence, is denoted by CI-CSKA. It provides us with the ability to use simpler carry skip logics. The logic replaces 2:1 multiplexers by AOI/OAI compound gates (Fig- 2). The gates, which consist of fewer transistors, have lower delay, area, and smaller power consumption compared with those of the 2:1 multiplexer [3]. The structure has a considerable lower propagation delay with a slightly smaller area compared with those of the conventional one. Note that while the power consumptions of the AOI (or OAI) gate are smaller than that of the multiplexer, the power consumption of the proposed CI-CSKA is a little more than that of the conventional one. This is due to the increase in the number of the gates, which imposes a higher wiring capacitance (in the noncritical paths).

Now, we describe the internal structure of the proposed CI-CSKA shown in Fig-2 in more detail. The adder contains two *N* bits inputs, *A* and *B*, and *Q* stages. Each stage consists of an RCA block with the size of *Mj* ($j = 1, \ldots, Q$). In this structure, the carry input of all the RCA blocks, except for the first block which is *Ci* , is zero (concatenation of the RCA blocks). Therefore, all the blocks execute their jobs simultaneously. In the proposed structure, the first stage has only one block, which is RCA. The stages 2 to *Q* consist of two blocks of RCA and incrementation. The incrementation block uses the intermediate results generated by the RCA block and the carry output of the previous stage to calculate the final summation of the stage. The internal structure of the

incrementation block, which contains a chain of half-adders (HAs), is shown in Fig-3. In addition, note that, to reduce the delay considerably, for computing the carry output of the stage, the carry output of the incrementation block is not used.

As shown in Fig-2, if an AOI is used as the skip logic, the next skip logic should use OAI gate. In addition, in the Conv-CSKA, the skip logic (AOI or OAI compound gates) is not able to bypass the zero carry input until the zero carry input propagates from the corresponding RCA block. To solve this problem, in the proposed structure, we have used an RCA block with a carry input of zero (using the concatenation approach). This way, since the RCA block of the stage does not need to wait for the carry output of the previous stage, the output carries of the blocks are calculated in parallel.

3.2 Area and Delay of the Proposed Structure

As mentioned before, the use of the static AOI and OAI gates (six transistors) compared with the static 2:1 multiplexer (12 transistors), leads to decreases in the area usage and delay of the skip logic[3],[4]. In addition, except for the first RCA block, this means that $(Q - 1)$ FAs in the conventional structure are replaced with the same number of HAs in the suggested structure decreasing the area usage (Fig-2). In addition, note that the proposed structure utilizes incrementation blocks that do not exist in the conventional one. Therefore, the area usage of the proposed CI-CSKA structure is decreased compared with that of the conventional one.

3.3. Stage Sizes Consideration

Similar to the Conv-CSKA structure, the proposed CI-CSKA structure may be implemented with either
FSS or VSS. Here, the stage size is the same as the RCA and incrementation blocks size. In the case of the FSS (FSS-CI-CSKA), there are $Q = N/M$ stages with the size of M. The optimum value of M, which may be obtained using (1), is given by

$$M_{opt}=\sqrt{\frac{N(T_{AOI}+T_{OAI})}{2(T_{CARRY}+T_{AND}}} \qquad (1)$$

In the case of the VSS (VSS-CI-CSKA), the sizes of the stages obtained using some steps.

The size of the RCA block of the first stage is one. From the second stage to the nucleus stage, the size of stage is increased. The increase in the size is continued until the summation of all the sizes up to this stage becomes larger than $N/2$. The size of the last stage is one, and its RCA block contains a HA.

4. Proposed Hybrid Variable Latency CSKA Structure

The basic idea behind using VSS CSKA structures was based on almost balancing the delays of paths such that the delay of the critical path is minimized compared with that of the FSS structure. To provide the variable latency feature for the VSS CSKA structure, we replace some of the middle stages in our proposed structure with a PPA modified in this paper. The proposed hybrid variable latency CSKA structure is shown in Fig-4 wherean Mp-bit modified PPA is used for the pth stage (nucleus stage). Since the nucleus stage, which has the largest size (and delay) among the stages, replacing it by the PPA reduces the delay of the longest off-critical paths. Thus, the use of the fast PPA helps increasing the available slack time in the variable latency structure. It should be mentioned that since the input bits of the PPA block are used in the predictor block.

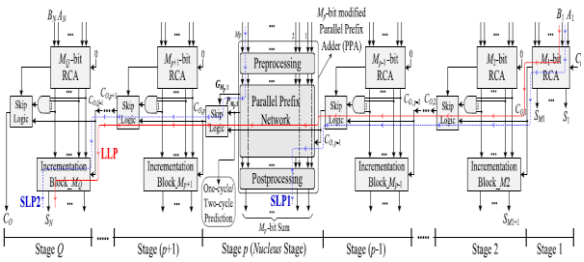

Fig-4: Structure of the proposed hybrid variable latency CSKA

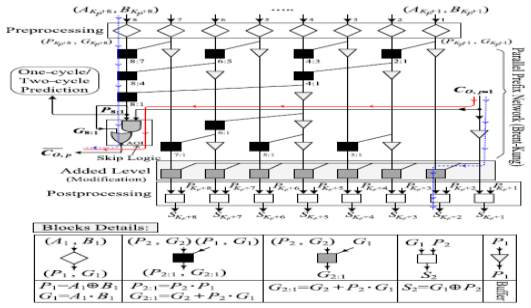

Fig-5: Internal structure of the *p*th stage of the proposed hybrid variable latency CSKA

In the proposed hybrid structure, the prefix network of the Kogge-Stone adder is used for constructing the nucleus stage (Fig-5). One the advantages of this adder compared with other prefix adders is less fan-out of adder, while the length of its wiring is smaller. Finally, it has a simple and regular layout. The internal structure of the stage p, including the modified PPA and skip logic, is shown in Fig-5. Note that, for this figure, the size of the PPA is assumed to be 8 (i.e., Mp = 8).

5. RESULT

5.1 Comparison of Different Parameters

Simulation was performed using the TANNER tool. Power and delay were directly obtained from software and area was measured in terms of number of transistors used in design.

Table -1: Comparison among different structures

Different structure/ Parameter	Conv-CSKA	CI-CSKA	Hybrid Variable Latency CSKA
Power(mW)	0.50	0.49	0.48
Delay(ns)	30.5	29.3	27.3
Area(transistor count)	1624	1762	1752

From this table we can conclude that the proposed hybrid variable latency CSKA consumes less power when compared to the other structures. The analysis also shows that the delay is minimized.

5.2 Simulation Results

Using tanner software the output analysis for various structures of carry skip adder was obtained and it is shown in figure.

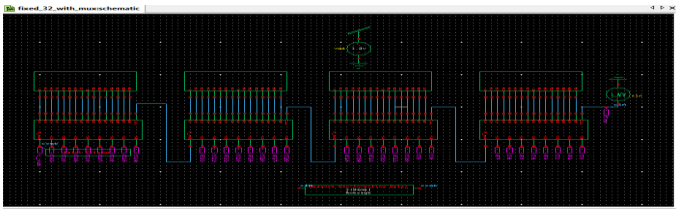

Fig -6: Conventional CSKA

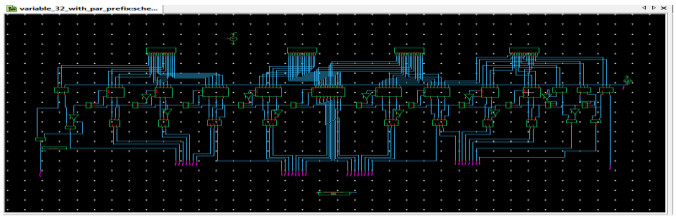

Fig -7: CI-CSKA. Stage size LSB to MSB {1,1,1,2,2,3,3,8,2,2,1}

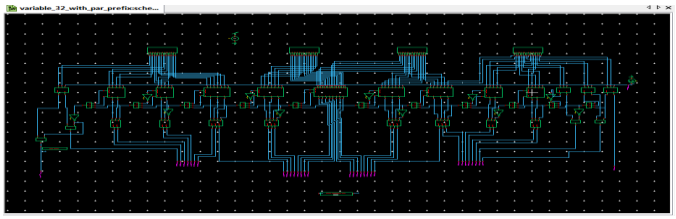

Fig -8: Proposed Hybrid Variable Latency CSKA

6. CONCLUTION

A hybrid variable latency extension of the structure CI-CSKA was proposed. It exploited a modified parallel adder structure at the middle stage for increasing the slack time, which provided us with the opportunity for lowering the power consumption. The suggested structure showed the lowest delay and PDP making itself as a better candidate for high-speed low-power applications. Here CI-CSKA is a static CMOS CSKA structure, which exhibits a higher speed and lower power consumption compared with those of the conventional one. The speed enhancement was achieved by modifying the structure through the concatenation and incrementation techniques. In addition, AOI and OAI compound gates were exploited for the carry skip logics. The results revealed considerably lower PDP. The results also suggested the CI-CSKA structure as a very good adder for the applications where both the speed and power consumption are critical.

REFERENCES

[1] Koren, *Computer Arithmetic Algorithms*, 2nd ed. Natick, MA, USA: A K Peters, Ltd., 2002.

[2] M. Alioto and G. Palumbo, "A simple strategy for optimized design of one-level carry-skip adders," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 50, no. 1, pp. 141–148, Jan. 2003.

[3] J. M. Rabaey, A. Chandrakasa, and B. Nikolic, *Digital Integrated Circuits: A Design Perspective*, 2nd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2003.

[4] NanGate 45 nm Open Cell Library. [Online]. Available: http://www.nangate.com/, accessed Dec. 2010.

[5] V. G. Oklobdzija, B. R. Zeydel, H. Dao, S. Mathew, and R. Krishnamurthy, "Energy-delay estimation technique for high- performance microprocessor VLSI adders," in *Proc. 16th IEEE Symp. Comput. Arithmetic*, Jun.2003, pp. 272–279

[6] P. M. Kogge and H. S. Stone, "A parallel algorithm for the efficient solution of a general class of recurrence equations," *IEEE Trans. Comput.*, vol. C-22, no. 8, pp. 786–793, Aug. 1973.

[7] S. Turrini, "Optimal group distribution in carry-skip adders," in *Proc. 9th IEEE Symp. Comput. Arithmetic*, Sep. 1989, pp. 96–103.

[8] C. Nagendra, M. J. Irwin, and R. M. Owens, "Area-time-power tradeoffs in parallel adders," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 43, no. 10, pp. 689–702, Oct. 1996.