DESIGN OF HIGH SPEED AND LOW POWER CARRY SKIP ADDER USING SPECULATIVE TECHNIQUE

Priya.V[2], *PG Student, Dept. of ECE, Venkateshwara Hi-Tech Engineering College, Gobi, India.*

Kamalakannan.R.S[1], *Asst. Professor Dept. of ECE, Venkateshwara Hi-Tech Engineering College, Gobi, India.*

Abstract--- *The speed enhancement is achieved by applying concatenation and instrumentation schemes to improve the good organization of the conventional CSKA (Conv-CSKA) structure. In addition, instead of utilizing multiplexer logic, the arrangement make use of AND-OR-Invert (AOI) and OR-AND-Invert (OAI) compound gates for the skip logic. Which lowers the power utilization without considerably impacting the speed, is presented. This extension utilizes a modified parallel structure for increasing the loose time, and hence, enable a more voltage reduction. The proposed structures are assessed by compare their velocity power, and energy parameters with those of other adders using a 45-NM static CMOS technology for a wide range of supply voltages. A carry skip adder (CSKA) structure that has a higher pace yet lower get-up-and-go consumption compared with the conventional one. A variable latency adder employs speculation the exact arithmetic function is replace with an approximated one that is faster and gives the correct result most of the time, but not always. The approximated adder is augmented with an error detection network that asserts a mistake signal when speculation fails. A story variable latency speculative adder based on Han-Carlson parallel- prefix topology that resulted more helpful than variable latency Kogge-Stone topology. The paper describes the stages in which variable latency speculative prefix adders can be subdivided and presents a tale error detection network that reduce error probability compared to previous approaches. Several variable latency speculative adders, for a variety of operand lengths by means of both Han-Carlson and Kogge-Stone topology.*

Keywords -- *Carry skips adder (CSKA), energy efficient, high Performance, hybrid variable latency adders, voltage scaling Addition, digital arithmetic, parallel-prefix adders, speculative adders, speculative functional units, variable latency adders.*

1. INTRODUCTION

ADDERS are a key building block of maths and logic units (ALUs) [1] and hence increasing their speed and reducing their power/energy expenditure strongly affect the speed and power consumption of processors. There are countless works on the subject of optimizing the speed and power of these units, which have been report in [2]–[9].Obviously, it is highly desirable to achieve higher speeds at low power/energy consumptions, which is a challenge for the designers of general point processors. One of the effective techniques to lesser the power consumption of digital circuits is to reduce the supply voltage due to the quadratic dependence of the switching energy on the voltage. Moreover, the subthreshold current, which is the main leakage component in OFF campaign has an exponential dependence on the supply voltage level through the drain-induced barrier lowering effect [10]. Depending on the amount of the supply voltage reduction, the operation of ON devices may reside in the group hold, near-threshold, or subthreshold regions. Working in the superthreshold region provides us with lower delay and higher switching and leakage powers compared with the near/subthreshold regions. In the subthreshold region, the logic gate delay and leakage power exhibit exponential dependences on the supply and threshold voltages. Moreover, these voltages are (potentially) subject to process and environmental variations in the nanoscale technologies. The variations increase uncertainties in the aforesaid performance parameters. In addition, the small subthreshold current causes a large delay for the circuits operating in the subthreshold region [10]. **A**DDERS ARE basic functional units in computer sums Binary adders are used in microprocessor for addition and subtraction operations as well as for floating point multiplication and division. Therefore adders are fundamental components and improving their performance is one of the major challenges in digital designs. Theoretical do research [1] has established lower bounds on the area and delay of -bit adders: the former varies linearly with udder size, the latter has a behavior. High swiftness adders are based on well established similar prefix architectures [1], [2], including Brent-Kung [3], Kogge-Stone [4], Sklansky [5], Han-Carlson [6], Ladner-Fischer [7], Knowles [8]. These standard architectures operate with undying latency. Better average performances can be achieved by using patchy latency adders, that have been recently proposed in literature[9]. A variable latency adder employs speculation: the exact

arithmetic utility is replaced with an approximated one that is faster and gives the correct result most of the time, but not always. The approximated adder is augmented with an error detection network that asserts an output gesture when speculation fails.

2. PREVIOUSLY PROPOSED ARCHITUCTURE

2.1. CSKA Structure

The structure is based on top of combining the concatenation and the incrementation with the Conv-CSKA structure, and hence, is denoted by CI-CSKA. It provides us with the ability to use simpler carry bounce logics. The logic replaces 2:1 multiplexers by AOI/OAI compound gates. The gates, which consist of less transistors, have lower delay, area, and smaller power consumption compared with those of the 2:1 multiplexer. Note that, in this structure, as the carry propagates through the hop logics, it becomes complemented[3].

Therefore, at the output of the skip logic of even stages, the complement of the carry is generated. The structure has a considerable minor propagation delay with a somewhat smaller area compared with those of the conventional one. Note that while the power consumptions of the AOI (or OAI) gate are slighter 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).

In this structure, when the first block computes the summation of its corresponding input bits (i.e., SM1, S1), and C1, the other blocks simultaneously compute the intermediate results [i.e., {ZK j+Mj , ZK j+2, ZK j+1} for K j =_j−1 r=1 Mr (j = 2, . . . , Q)], and also Cj signals.
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 internal structure of the CI-CSKA shown in Figure.3.1. 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, Q).the carry effort of all the RCA block, except for the first block which is Ci, is zero (concatenation of the RCA blocks). Therefore, all the blocks execute their jobs simultaneously[3].

The incrementation block use the intermediate results generated by the RCA block and the carry output of the previous stage to calculate the final abridgment of the stage.

2.2. The internal structure

The internal configuration of the incrementation block, which contains a chain of half-adders (HAs).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[1].

The skip logic determines the carry output of the j th stage (CO, j) based on the in-between results of the j th stage and the carry output of the previous stage ($CO, j−1$) as well as the carry output of the corresponding RCA block (Cj). When determining $CO, j,$ these cases may be encountered. When Cj is equal to one, CO, j will be one. On the other hand, when Cj is equal to zero, if the product of the intermediate results is one (zero), the value of CO, j will be the same as $CO, j−1$ (zero).

The reason for using together AOI and OAI compound gates as the skip logics is the inverting functions of these gates in standard cell libraries. This way the need for an inverter gate, which increases the power consumption and delay, is eliminated. if an AOI is used as the skip logic, the next skip logic should use OAI gate. In addition, another point to talk about is that the use of the proposed skipping structure in the Conv-CSKA structure increases the holdup of the critical path considerably[7].

This originates from the fact that, in the Conv-CSKA, the skip logic (AOI or OAI compound gates) is not able to bypass the nothing carry input awaiting the zero carry input propagates from the corresponding RCA block. To solve this problem, in the proposed structure, we have used an RCA lump 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 carry of the blocks are calculated in parallel.

2.3. Area and delay

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. In totaling except for the first RCA block, the carry input for all other blocks is zero, and hence, for these blocks, the first adder cell in the RCA chain is a HA[8].

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. In addition, note that the proposed structure utilizes incrementation blocks that do not exist in the conventional one. These blocks, however, may be implemented with about the same logic gates (XOR and gates) as those used for generating the

select signal of the multiplexer in the conventional structure.

Therefore, the area usage of the proposed CI-CSKA structure is decreased compare with that of the conventional one. The critical path of the proposed CI-CSKA structure, which contains three parts, is shown in Figure.3.2. These parts include the string of the FAs of the first stage, the path of the skip logics, and the incrementation block in the last period

To reduce the delay very much for computing the carry output of the stage, the carry output of the incrementation block is not used. the skip logic determine the carry output of the j th stage (CO, j) based on the intermediate products of the j th stage and the take output of the previous phase($CO, j-1$) as well as the carry output of the corresponding RCA block (Cj)[5].
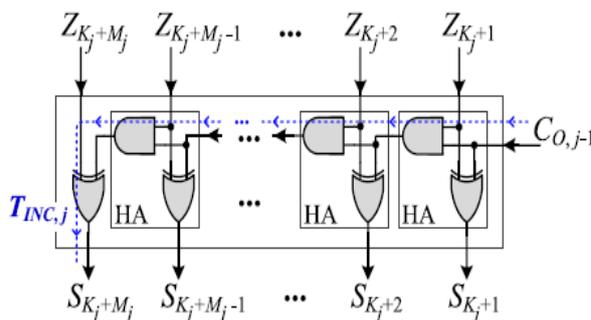
FIG.1.internal structure of the j th incrementation block

When determining $CO, j,$ these cases may be encountered. When Cj is equal to one, CO, j will be one. On the other hand, when Cj is equal to zero, if the product of the intermediate results is one (zero), the value of CO, j will be the same as $CO, j-1$ (zero).The grounds for using together AOI and OAI compound gates as the skip logics is the inverting functions of these gates in regular cell libraries[2].

This method the require for an inverter gate, which increases the power consumption and delay, is eliminated. if an AOI is used as the skip logic, the after that skip logic be imaginary to use OAI gate.

In addition, another point to mention is that the use of the anticipated skipping structure in the Conv-CSKA structure increases the delay of the grave path considerably. This originates from the fact that, 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 propagate on or after the corresponding RCA block[4].

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 comparison of indicates that the delay of the proposed structure is smaller than that of the conventional one.

The First reason is that the holdup of the skip logic is considerably smaller than that of the conventional structure while the number of the stages is concerning the same in both structures.

Second, since TAND and TXOR are smaller than TCARRY and TSUM, the third additive term becomes smaller than the third phrase. It should be noted that the delay reduction of the skip logic has the chief impact on the delay decline of the whole structure.

2.4. STAGE SIZES CONSIDERATION

comparable to the Conv-CSKA structure, the proposed CI-CSKA structure may be implement with either the procedure for determining the stage sizes is demonstrated for the 32-bit adder. It includes together the conventional and the CI-CSKA structures. The number of stages and the corresponding size for each stage has been determined based on a 45-nm static CMOS technology[8].

The dashed and dotted lines in the plot indicate the rates of size increase and decrease. While the increase and decrease rates in the conformist structure are balanced, the decrease speed is more than the increase one in the case of the proposed structure. It originates from the fact that, in the Conv-CSKA structure, both of the stage size increase and decrease are determined based on the RCA block delay the increase is determined based on the RCA block delay and the decrease is determined based on the incrementation block delay .The imbalanced rates may yield a larger nucleus stage and smaller figure of stages leading to a sSmaller propagation delay.

3. PROPOSED STAGE SIZES CONSIDERATION

Variable latency speculative prefix adders can be subdivided in five stages: Pre-processing, speculative prefix-processing, post-processing, error detection and error correction. The error correction stage is off the critical path, as it has two clocks Cycles to obtain the exact sum when speculation fails[8].

3.1. PRE-PROCESSING AND SPECULATIVE PREFIX-PROCESSING

In the pre-processing stage is generate and spread signals are computed. The speculative prefix-processing

stage is one of the foremost differences compared with the standard prefix adders recalled in previous section. in its place of computing all the required to acquire the exact carry values, only a subset of block generate and propagate signals is calculated.

In the post giving out stage approximate carry values are obtained from this subset. The production of the speculative prefix-processing period will also be used in the error detection and in the error correction stages discussed in the following. The basic assumption behind speculative prefix-processing stage is that carry signal propagate for no more than bits, This postulation is corroborated by the analyses that demonstrate that have a proliferate chain longer that is a very rare event.

3.2. KOGGE-STONE TOPOLOGY AND HAN-CARLSON TOPOLOGY

The Kogge-Stone exploratory prefix-processing stage has been proposed in can be obtained by pruning the last levels of a established Kogge-Stone adder. Kogge-Stone shown in Figure.1. The last level of a bit Kogge-Stone adder is pruned. As it can be observed, for the length of circulate chains extends for 8 bits, resulting in a speculative prefix-processing stage with Han-Carlson adder constitutes a good trade-off flanked by fan out, number of logic levels and number of black cells[6].
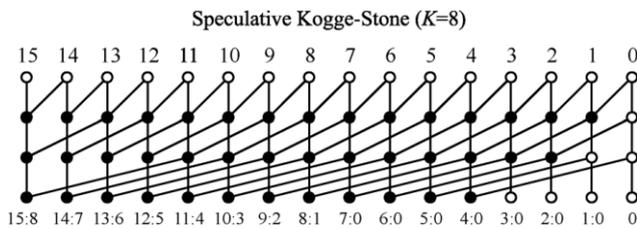
FIG.2. Kogge-stone speculative prefix-processing stage

Because of this, Han-Carlson adder can achieve equal speed performance respect to Kogge-Stone adder, at lower power consumption and area. Therefore it is interesting to implement a speculative Han-Carlson adder. Moved by these reasons, we have generated a Han-Carlson speculative prefix-processing stage by deleting the last rows of the Kogge-Stone part of the adder.

The two Brent- Kung rows at the beginning and at the end of the graph are unchanged, while the last Kogge-Stone row is pruned. This yields a speculative stage. In general, one has where the number of pruned levels.

3.3. ERROR DETECTION

The conditions in which at least one of the approximate carries is wrong (misprediction) are signaled by the error detection stage. In case of misprediction, an error signal is asserted by error detection stage and the output of the post-processing stage is discarded. The error correction stage will give the correct sum in the next clock period.

it can easily be seen that the number of terms to be OR-ed to obtain the error signal is halved in the Han-Carlson topology, compared to Kogge-Stone "checking nodes" the nodes of the prefix-processing stage, whose outputs are needed to compute the error signal[11].

The checking nodes for both the Kogge-Stone and the Han-Carlson are highlighted as big hatched dots as it can be observed, in Kogge-Stone some of the checking cells are at the last level of the graph; their output signals are available after three black cells delay.

Han-Carlson speculative of prefix-processing stage shown in figure.4.2. The critical checking cells are in the second last level of the graph and are also available after three black cells delay, in spite of the larger number of levels of the Han-Carlson prefix-processing stage. From the above observations, it can be concluded that error detection is sensibly simplified and potentially faster in Han- Carlson,
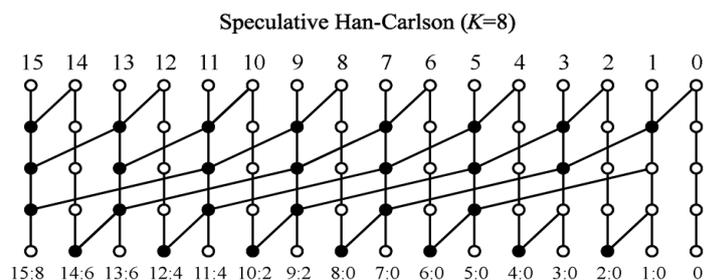
FIG.3. Han-carlson speculative prefix-processing stage

As an additional note, the need of driving the gates of the error detection stage increases the fanout of the checking cells, slowing the speculative prefix-processing stage.

3.4. ERROR CORRECTION

The error correction stage computes the exact carry signals to be used in case of mis prediction. The error correction stage is composed by the levels of the prefix processing stage pruned to obtain the speculative adder.

The error correction stage of the proposed speculative Han-Carlson adder; the error correction for Kogge-Stone topology can be obtained similarly. It can be observed that the inclusion of the error correction stage increases the fan out of some of the cells of the speculative prefix-processing stage, with adverse effect on adder speed. Post-Processing the approximate carries is already available at the output of
The prefix-processing stage. The post-processing, according to is equal to the one of a non-speculative adder and consists of xor gates[10].

Comparison between variable latency adder and the non-speculative Han-Carlson topology reveal that variable latency adders allow reducing the minimum achievable delay. For instance, in the 64 bit case, the minimum achievable delay is about 280 ps for the non-speculative adder and reduces up to 225 ps in the variable latency architecture.

The analysis of Area Occupation and Power Dissipation shows that speculative adders are not effective for large average delay. As the timing constraint imposed during synthesis is made tighter speculative adders become advantageous.

It is not easy to compare performances (in terms of power, speed, and area) of different designs, since they strongly depend on timing constraint used during synthesis. The results reported in the following have been obtained by performing several syntheses of the circuits under investigation, by varying the timing constraint[12].

In this way we can compare the various topologies and find the most effective ones depending on the required speed. The dynamic power dissipation has been evaluated after synthesis by extracting the nodes activities from a back-annotated simulation broadcasts.

4. RESULT AND DISSCUTIONS

In this paper, reduce the delay and power and area, Compare to the existing approach. It is used for high speed devices.
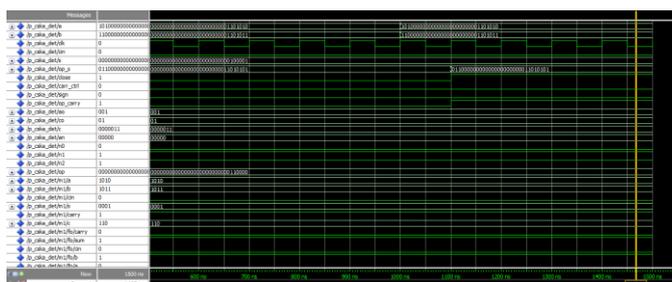
5. CONCLUSION

The speed enhancement was achieved by modifying the structure through the concatenation and incrementation techniques. In addition, AOI and OAI amalgam gates were exploited for the carry skip logics. The efficiency of the proposed structure for equally FSS and VSS was studied by comparing its power and delay with those of the Conv-CSKA, RCA, CIA, SQRT-CSLA, and KSA structures. The results exposed considerably lower PDP for the VSS implementation of the CI-CSKA structure over a wide range of voltage from super-threshold to in close proximity to threshold. The results also suggested the CI-CSKA structure as a very good adder for the applications where both the speed and energy consumption are critical. CI-CSKA was projected which exhibits a higher speed and lower energy consumption compared with those of the conventional one. Han-Carlson a variable latency adder outper forms beforehand developed variable latency Kogge-Stone architectures. Compared with traditional, Non -speculative, adders, our analysis demonstrate that variable latency Han-Carlson adders show sensible improvements when the highest rate is required; otherwise the burden imposed by error detection and error modification stages in excess ofwhelms any benefit.

REFERENCES

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

2. Chang. C.-H, Gu.J, and M. Zhang, "A review of 0.18 μm full adder performances for tree structured arithmetic circuits," (2005)IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 13, no. 6, pp. 686–695.

3. Chirca.K et al., "A static low-power, high-performance 32-bit carry skip adder,"(2004) in Proc. Euromicro Symp. Digit. Syst. Design (DSD), pp. 615–619.

4. Dreslinski .R. G, Wieckowski.M, Blaauw. D , Sylvester D., and Mudge. T, "Near-threshold computing: Reclaiming Moore's law through energy

efficient integrated circuits," (2010) Proc. IEEE, vol. 98, no. 2, pp. 253–266,.

5. Harris. T, taxonomy of parallel prefix networks," (2003) in Proc. IEEE Conf. Rec. 37th Asilomar Conf. Signals, Syst., Comput., vol. 2, pp. 2213–2217.

6. He .Y, and Chang. C.-H, "A power-delay efficient hybrid carrylookahead/ carry-select based redundant binary to two's complement converter,"(2008) IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 55, no. 1,pp. 336–346.

7. Jia.S et al., "Static CMOS implementation of logarithmic skip adder," (2003) in Proc. IEEE Conf. Electron Devices Solid-State Circuits, ,pp. 509–512.

8. Jain. S, et al., "A 280 mV-to-1.2 V wide-operating-range IA-32 processor in 32 nm CMOS," (2012)in IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers (ISSCC), pp. 66–68.

9. Koren.I, Computer Arithmetic Algorithms.(2002) Natick, MA, USA: A K Peters.

10. Markovic. D, Wang. C. C, Alarcon. L. P, Liu. T.-T, and Rabaey. J. M,"Ultralow-power design in near-threshold region,"(2010) Proc. IEEE, vol. 98, no. 2, pp. 237–252.

11. Mathew. S. K, Anders. M. A, Bloechel .B, Nguyen.T, Krishnamurthy. R. K, and Borkar. S, "A 4-GHz 300-mW 64-bit integer execution ALU with dual supply voltages in 90-nm CMOS,"(2005) IEEE J. Solid-State Circuits, vol. 40, no. 1, pp. 44–51.

12. Oklobdzija.V.G, Zeydel.B.R,. Dao.H.Q, Mathew.S, and Krishnamurthy.R, "Comparison of high-performance VLSI adders in the energy-delay space,"(2005) IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 13, no. 6, pp. 754–758.

13. Ramkumar.B and Kittur.H.M, "Low-power and area-efficient carry select adder,"(2012) IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 20,no. 2, pp. 371–375.

14. Suzuki.H, Jeong.W, and Roy.K, "Low power adder with adaptive supply voltage,"(2003) in Proc. 21st Int. Conf. Comput. Design, pp. 103–106.

15. Zlatanovici .R, Kao.S, and Nikolic.B, "Energy–delay optimization of 64-bit carry-lookahead adders with a 240 ps 90 nm CMOS design example," (2009) IEEE J. Solid-State Circuits, vol. 44, no. 2, pp. 569–583.

BIOGRAPHIES

Kamala Kannan R.S. received his B.E. degree in Electronics and communication engineering from Mookambigai College of Engineering , Trichy, Tamilnadu in 2007, the M.E. degree in VLSI from KSR College of technology , , Tamilnadu in 2011. He was an Assistant professor, Shree Venkateshwara Hi-Tech Engineering College, 2010-2016.

Priya. V received the B.E degree in electronic and communication engineering with first class from Shree Venkateshwara Hi-tech Engineering College, Gobichettipalayam , Tamilnadu in 2014, At present, She is engaged in M.E in Applied Electronics from Shree Venkateshwara Hi-Tech Engineering College, 2014-2016.