Introduction

Computer arithmetic has become wide spread in many applications such as AI/ML, automotive computing, graphics computing, HPC, signal processing applications and many more applications. In these applications, basic arithmetic cells are half adder (HA) and full adder (FA) and are responsible for all complicated mathematical computations. These adder cells require high performance and drains lot of power. Moreover, they are also responsible increasing dependency between dependent instructions such as multiply accumulate, loop instructions, object movements in graphics, signal processing computations, neural network training in AI/ML, linear algebra, climate modeling computations in HPC and many more areas.

Literature review

In order to overcome above many adders are invented such as ripple carry adder (RCA), carry save adder (CSA), carry look-ahead adder (CLA), carry skip adder (CSkA) and carry select adder (CSelA) [1]-[7].

The RCA is constructed by cascading FA blocks in series. One full adder is responsible for the addition of two binary digits at any stage of the ripple carry. The carryout of one stage is fed directly to the carry-in of the next stage. Even though this is a simple adder and can be used to add unrestricted bit length numbers, it is however not very efficient when large bit numbers are used. One of the most serious drawbacks of this adder is that the delay increases linearly with the bit length.

The CSA reduces the addition of 3 numbers to the addition of 2 numbers. The propagation delay is 3 gates regardless of the number of bits. The carry-save unit consists of n FAs, each of which computes a single sum and carries bit based solely on the corresponding bits of the three input numbers. The entire sum can then be computed by shifting the carry sequence left by one place and appending a 0 to the front (most significant bit) of the partial sum sequence and adding this sequence with RCA produces the resulting n + 1-bit value. This process can be continued indefinitely, adding an input for each stage of FAs, without any intermediate carry propagation. These stages can be arranged in a binary tree structure, with cumulative delay logarithmic in the number of inputs to be added, and invariant of the number of bits per input. The main application of carry save algorithm is, well known for multiplier architecture is used for efficient CMOS implementation of much wider variety of algorithms for high speed digital signal processing. CSA applied in the partial product line of array multipliers will speed up the carry propagation in the array but in final stage of partial product one has to use one of above adders to propagate carry.

CLA adder is designed to overcome the latency introduced by the rippling effect of the carry bits. The propagation delay occurred in the parallel adders can be eliminated by carry look ahead adder. This adder is based on the principle of looking at the lower order bits of the augends and addend if a higher order carry is generated. This adder reduces the carry delay by reducing the number of gates through which a carry signal must propagate. Carry look ahead depends on two things: Calculating for each digit position, whether that position is going to propagate a carry if one comes in from the right and combining these calculated values to be able to deduce quickly whether, for each group of digits, that group is going to propagate a carry that comes in from the right. The net effect is that the carries start by propagating slowly through each 4-bit group, just as in a ripple-carry system, but then moves 4 times faster, leaping from one look ahead carry unit to the next. Finally, within each group that receives a carry, the carry propagates slowly within the digits in that group. CLA increases silicon area thus drains lot of power.

A CSkA consists of a simple ripple carry-adder with a special speed up carry chain called a skip chain. CSkA is a fast adder compared to ripple carry adder when addition of large number of bits take place; carry skip adder has $O(\sqrt{n})$ delay provides a good compromise in terms of delay, along with a simple and regular layout. This chain defines the distribution of ripple carry blocks, which compose the skip adder.

A CSKa is designed to speed up a wide adder by aiding the propagation of a carry bit around a portion of the entire adder. Actually the ripple carry adder is faster for small values of N. However the industrial demands these days, which most desktop and laptop computers use word lengths of 64 bits like multimedia processors, makes the carry skip structure more interesting. CSKa is a variation of RCA though it is bit faster than RCA wiring congestion leads to denser silicon area.

A CSelA is divided into sectors, each of which – except for the least-significant –performs two additions in parallel, one assuming a carry-in of zero, the other a carry-in of one. A four bit carry select adder generally consists of two RCAs and a multiplexer. The CSelA is simple but rather fast, having a gate level depth of $O(\sqrt{n})$. Adding two n-bit numbers with a carry select adder is done with two adders (two RCAs) in order to perform the calculation twice, one time with the assumption of the carry being zero and the other assuming one. After the two results are calculated, the correct sum, as well as the correct carry, is then selected with the multiplexer once the correct carry is known. CSelA requires twice silicon area compared to RCA, making chip expensive and power greedy.

VividSparks CFA Technology

With 10 years of R&D efforts we were finally able to crack carry propagation problem puzzle. VividSparks CFA technology detects carry bits in advance by just looking at input bit patterns. Carry Detection Unit has *only* logic depth of 6 CMOS transistors. Following figure 1 shows over all CFA architecture. Final sum is calculated using carry detected unit. CFA technology is better explained in [8].
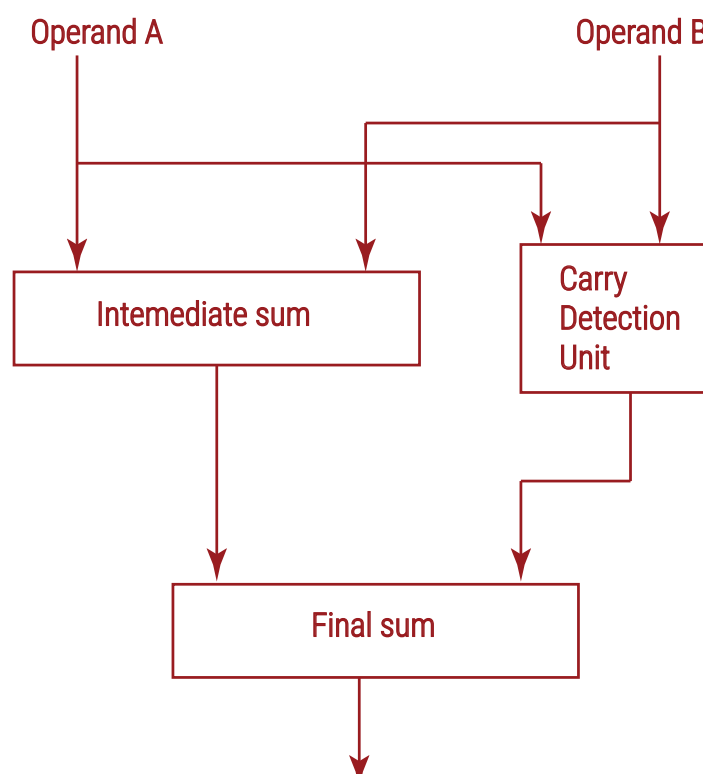
Figure 1: CFA Architecture

Further Applications of VividSparks CFA Technology

- Leading 0's and 1's detection in arithmetic circuits

- Booth multiplier CSA accumulation

- Booth decoding of multiplier

- Odd multiple generation in Booth multipliers for 3X, 5X, 7X and so on

- Comparators

- Branch prediction operations

- ALU operations

- Quire operations in POSIT and

- many more applications!

References

[1] M. J. Flynn, S. Oberman, "*Advanced Computer Arithmetic Design*", pp.01-22.

[2] K.Hwang, "*Computer Arithmetic Principles, Architecture and Design*", pp. 69-128.

[3] T. Sultana. R. Bardhan, T. F. Bithee, Z. Tabassum; N. J Lisa "*A compact design of n-bit ripple carry adder circuit using QCA architecture* ", 2015 IEEE/ACIS 14th International Conference on Computer and Information Science (ICIS).

[4] R. Mahalakshmi; T. Sasilatha, et al, "*A power efficient carry save adder and modified carry save adder using CMOS technology* ", 2013 IEEE International Conference on Computational Intelligence and Computing Research.

[5] J. Miao, S. Li, et al, "*A novel implementation of 4-bit carry look-ahead adder* ", 2017 International Conference on Electron Devices and Solid-State Circuits (EDSSC).

[6] A. Arora, V. Niranjan, "*A new 16-bit high speed and variable stage carry skip adder*", 2017 3rd International Conference on Computational Intelligence & Communication Technology (CICT).

[7] P. Sexena, et. al, "Design of low power and high speed Carry Select Adder using Brent Kung adder", 2015 International Conference on VLSI Systems, Architecture, Technology and Applications (VLSI-SATA).

[8] https://www.youtube.com/watch?v=LbaA0vEa1Uc&ab_channel=VividSparks

VividSparks IT Solutions Pvt. Ltd.
License no: U72200K2014OPC077975
#38 BSK Layout, Hubli-580031, India.
www.vivid-sparks.com
inquiry@vivid-sparks.com