International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 5, May - 2014

FPGA Implementation of Optimized Heterogeneous Adder for DSP Applications

Aishwarya T
Electronics and Communication dept,
Global Academy Of Technology,
Bangalore, India.

Arvind Prasad L
Electronics and Communication dept,
Global Academy Of Technology,
Bangalore, India.

Nikith G S
Electronics and Communication dept,
Global Academy Of Technology,
Bangalore, India.

Ahish S
Electronics and Communication dept,
Global Academy Of Technology,
Bangalore, India.

***Abstract*— Arithmetic operations are widely used and play important role in various digital systems such as computers and signal processors. An Adder is one of the significant hardware blocks in most digital systems such as digital signal processors and microprocessors etc. A lot of researches have been carried out in order to design an efficient adder circuits in terms of compactness, high speed and low power consumption, but area and speed are two conflict parameters. So, improving speed results always in larger area occupied by circuit on chip and vice-versa. In this paper, we propose, a new type of adder architecture known as heterogeneous adder that consists of concatenation of sub-adders (ling adder, carry look ahead adder and carry skip adder). The heterogeneous adder architecture provides better design tradeoffs in terms of area and delay characteristics.**

Keywords
Adder, Ripple carry adder, Look ahead carry adder, VHDL simulation, ling adder.

I. INTRODUCTION

The design of area-efficient and low-power VLSI architectures need efficient arithmetic processing units, which are optimized for the performance parameters, namely, area and power consumption. Adders are the key components in general purpose microprocessors and digital signal processors. They also find use in many other functions such as subtraction, multiplication and division. The evolution of portable or mobile communication devices such as laptops, cellular phones, video games, etc. is the most important factor driving the need for area efficient design. The main reason behind the development of area efficient and low power circuits is that, many portable devices must be smaller in size and their applications require low power dissipation and high throughput. In most cases, the cost and weight of batteries become a bottleneck that prevents the reduction of system cost and weight.

Moreover, for most portable systems, the IC (Integrated Circuit) components consume a significant portion of the total system power. Area efficient design also plays a significant role in high-performance integrated circuits such as microprocessors and other high-speed digital circuits. Due to the increase in clock frequency, there is a proportional increase in power dissipation. The power consumed by the integrated circuit is dissipated in the form of heat. This may lead to problems such as circuit degradation and operating failures. The power consumption in microprocessors is projected to grow linearly in proportion to their die size and clock frequency. Various cooling systems have been introduced to reduce the heat from power dissipation and keep the chip temperature at an admissible level. This in turn has increased the required area and packaging cost which results in large revenue. Efficient low power and area design techniques are required to avoid these problems. Therefore, the area efficient design makes the chip size smaller, reduces cost and weight making the devices handy. Reducing a circuit's average power consumption typically improves the circuit's reliability. This leads to a reduction in cooling requirements, which in turn reduces the packaging and cooling costs.

The adders presented in this paper are all modeled by using VHDL for 16-bit unsigned data. XILINX ISE 13.2i is used as synthesis tool and FPGA-Spartan III (XC3S250E) device is selected to get area report. Modelsim XE III 6.2g is used to get timing simulation.

II. RELATED WORK

A. 16 bit heterogeneous adder:

16-bit Heterogeneous adder proposed in [1], consists of four sub adders SA1 (Ripple carry adder), SA2 (Carry skip adder), SA3 (Carry select) and SA4 (Carry look ahead adder). Initially, equal bit-width for each sub-adder i.e. 4-bit as shown in figure 1 is chosen. All these four sub adders are concatenates to form a 16-bit heterogeneous adder.

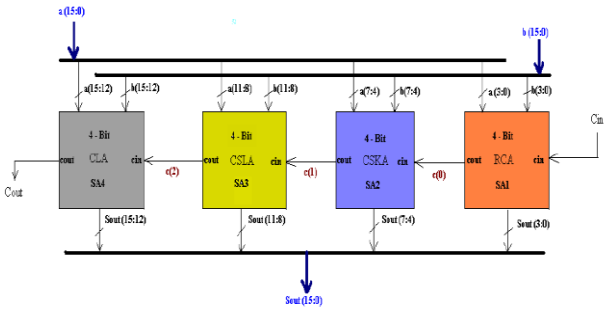

Fig. 1. 16-bit Heterogeneous adder.

B. *32 bit heterogeneous adder*[12]*:*

The architecture of a heterogeneous adder includes different types of adder implementations. 32-bit heterogeneous adder proposed in this work consists of four sub adders (SA), 8-bit carry look-ahead adder, 8-bit carry skip adder, 8-bit carry select adder and 8-bit ripple carry adder.

Bit size selection for each sub-adder can be done on the basis of requirements (i.e. Area, Speed and Power constraints) of particular application where the design is to be implemented. For example, ripple carry adder cover small area and less power consumption but at the cost of large operation delay whereas carry skip adder gives high speed of operation but at the cost of large area. Therefore in order to optimize adder design as per requirement, 32-bit heterogeneous adder is designed as shown in figure 2.

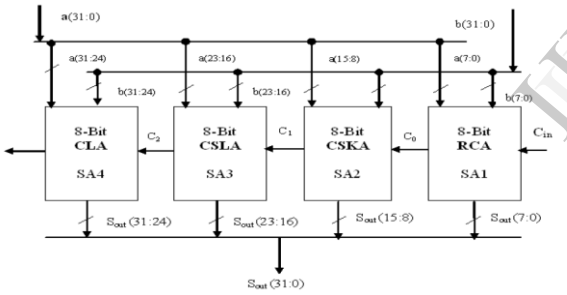

Fig. 2. 32-bit heterogeneous adder

III. PROPOSED MODEL

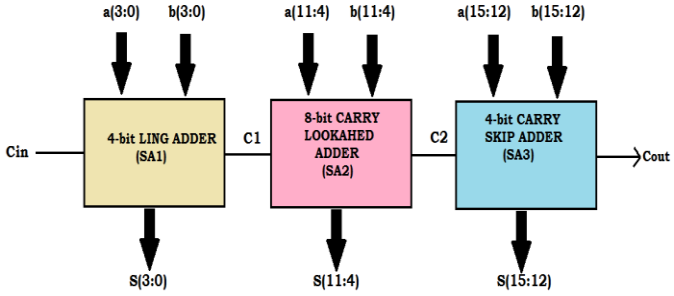

Fig 3:Proposed 16 bit Heterogeneous adder.

The 16-bit Heterogeneous adder proposed in this paper consists of three sub adders SA1,SA2 and SA3. Sub adder (SA1) consists of 4-bit Ling architecture, Sub adder (SA2) consists of 8-bit carry look ahead adder and Sub adder (SA3) consists of 4-bit Carry skip adder architecture. These sub adders concatenates to form a heterogeneous adder. The order of sub adder has an impact on the performance of a heterogeneous adder. It is observed that Ling adder requires less area but at the cost of large delay and vice-versa in case of hybrid carry skip adder, whereas carry lookahead adder has reduced delay. Therefore, in order to get optimized result in terms of area utilization and speed of operation we combine three adders to form a single adder with different bit size

A. *Ling Adder Design:*

The Ling Adder is a type of Look-Ahead Adder with a slight modification that results in significant hardware saving. Ling's modification consists of propagating hi = ci + ci-1 instead of ci. This result in reduction of number of gates required for implementation. Therefore, the Boolean expression for calculating next carry and sum are:

$$Ci = hi\ (Gi\text{-}1 + Pi\text{-}1)\ldots\ldots\ldots\ldots\ldots(1)$$
$$Si = Pi\ xor\ hi\ (Gi\text{-}1 + Pi\text{-}1)\ldots\ldots\ (2)$$

Where Pi is Carry propagation and Gi is Carry generation.

B. *Carry skip adder:*

The carry-skip adder (CSK) is a very simple but ingenious adder, with a minimum of additional logic. The n-bit adder is divided into 'k' (n/k-bit) ripple-carry adder blocks, where 'k' is greater than, or equal to 4. Each adder block has a group propagate signal, which means that when this propagate signal is 1, an incoming carry cannot be absorbed and will propagate through the adder block instead. If that is the case, we could as well skip this adder segment. This is exactly the idea behind the skip adder: when it can be determined, a priori, that a block cannot absorb a carry, it skips the block via the skip logic.

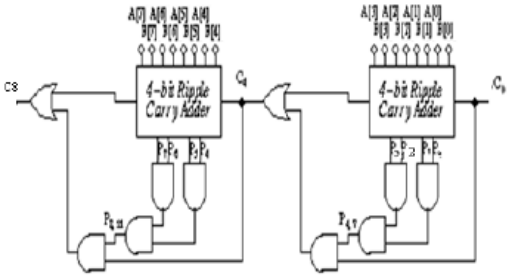

Fig. 4. Carry skip adder

C. *Carry lookahead adder:*

A Ripple Carry Adder (RCA) is an optimized area-efficient adder design. The layout of a ripple-carry adder is simple, which allows for fast design time but, it is relatively slow, since each full adder must wait for the carry bit to be calculated from the previous full adder. The maximum delay in RCA is computed from the carry-in input to the carry out, passing through each full adder along the way. By making tradeoffs between area and performance delay in adder circuit, faster but larger designs than RCA, can be construct that is Carry look ahead Adder (CLA).

Look ahead carry algorithm speed up the operation to perform addition, because in this algorithm carry for the next

stages is calculated in advance based on input signals. 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

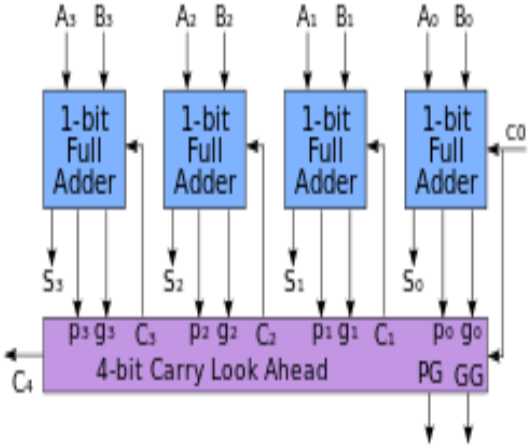

Fig. 5. Carry lookahead adder.

IV.SIMULATION RESULTS

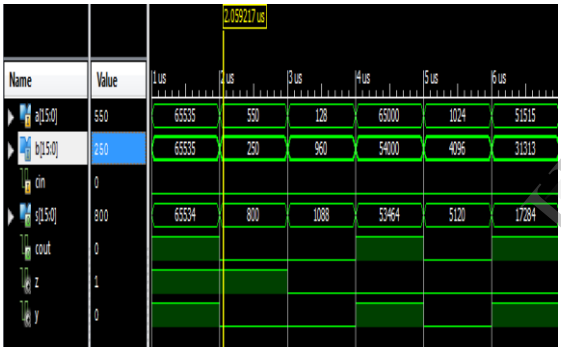

Fig. 6. Simulation Waveform of Proposed Heterogeneous adder.

TABLE 1 COMPARISON OF AREA AND DELAY

PARAMETERS	Carry skip adder 16-bit	Heterogeneous adder (CSLA, CLA, CSK)[1]	Heterogeneous adder (Ling, CLA, CSK)
No. Of SLICES	26 out of 3584	18 out of 2448	19 out of 2448
No. Of LUT's	45 out of 7168	32 out of 4896	33 out of 4896
Delay	29.211ns	22.022ns	17.727ns

The fig 6 shows the simulation waveform of the proposed full adder. The table1 shows the performance comparison between existing heterogeneous adder [1]and proposed heterogeneous adder. It is observed that delay is reduced in proposed method comparing with the existing.

IV. CONCLUSION

The path delay of proposed heterogeneous adder is 40% lesser when compared to carry skip adder and 20% lesser when compared to heterogeneous adder implemented using CSLA, CLA and CSKA. The area performance of the heterogeneous adders is better the performance of the carry skip adder. In future, the combination of adders can be improved for better results.

REFERENCES

[1] Raminder Preet Pal Singh, Ashish Chaturvedi and Onkar Singh, "Trade-offs in Designing High-Performance Digital Adder based on Heterogeneous Architecture", International Journal of Computer Applications (0975 – 8887) Volume 56– No.13, October 2012

[2] B. Parhami, 2000, Computer Arithmetic, Algorithm and Hardware Design, Oxford University Press, New York, pp. 91-119.

[3] Pong P. Chu "RTL Hardware Design Using VHDL: coding for Efficiency, Portability and Scalability" Wiley-IEEE Press, New Jercy, 2006

[4] Hasan Krad and Aws Yousif Al-Taie, "Performance Analysis of a 32-Bit Multiplier with a Carry-Look-Ahead Adder and a 32-bit Multiplier with a Ripple Adder using VHDL", Journal of Computer Science 4 (2008): 305-308.

[5] Asadi, P. and K. Navi "A novel high-speed 54-54-bit multiplier", Am. J. Applied Sci., 4 (9) (2007): 666-672.

[6] Z. Abid, H. El-Razouk and D.A. El-Dib, "Low power multipliers based on new hybrid full adders", Microelectronics Journal, Volume 39, Issue 12, 2008, 1509-1515.

[7] Nagendra, C.; Irwin, M.J.; Owens, R.M.,"Area-time-power tradeoffs in parallel adders", Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on Volume 43, Issue 10, (1996): 689 – 702.

[8] Wang, Y.; Pai, C.; Song, X., "The design of hybrid carry look-ahead/carry-select adders, Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on Volume 49, Issue 1, (Jan 2002) : 16-24.

[9] May Phyo Thwal, Khin Htay Kyi, and Kyaw Swar Soe, "Implementation of Adder-Subtracter design with VerilogHDL", International Journal of Electronics, Circuits and Systems Volume 2 number 3, 2008.

[10] Min Cha and Earl E. Swartzlander, Jr, "Modified Carry Skip Adder for reducing first block delay", Proc. 43rd IEEE Midwest Symp. on Circuits and Systems, Lansing MI, 2000: 346-348.

[11] Yu Shen Lin, "Delay Efficient 32-bit Carry-Skip Adder", Electronics, Circuits and Systems, 2006, ICECS '06, pp. 506-509.

[12] Neha Agarwal and Satyajit Anand, " Logical Effort to Study the Performance of 32-bit Heterogeneous Adder", International Journal of Computer Applications (0975 – 8887) Volume 65– No.16, March 2013.