Design of a Parallel Self Timed Adder Circuit Using Recursive Approach

Author1

M. MOUNIKA
Designation: STUDENT
MANDAVA INSTITUTE
OF ENGINEERING AND
TECHNOLOGY'S
JAGGAYYAPET

Author2

K. BHASKAR
Designation: guide
Mandava Institute of
Engineering and
Technology's
JAGGAYYAPET

Author3

B. SUBHAKARA RAO
Designation: HOD
Mandava Institute of
Engineering and
Technology's
JAGGAYYAPET

Abstract-

This brief proposes the design of a parallel self timed adder. It is based on an iterative formulation to perform multi bit addition. The operation will be parallel for the bits that does not need any chains or carry propagation. Hence, the design achieves more performance over random operand condition with no addition of any look ahead scheme and speed-up circuitry. A practical implementation is provided in this paper along with an extension of 4 block size when it is compared to the existing adder. Hence the implemented parallel self timed adder is of block size 8. The implementation is not at all complex and will not have any limits of high fan-outs. But a high fan-in gate is required as it is unavoidable for asynchronous logic and is achieved by the parallel connection of transistors. Simulations have been verified using Modelsim 6.5b tool that shows the practicality of design and Simulation of output waveforms are verified for the operation of addition.

INTRODUCTION

Binary addition is the single most important operation that a processor performs. Most of the adders have been designed for synchronous circuits even though there is a strong interest in clockless/ asynchronous processors/circuits [1]. Asynchronous circuits do not assume any quantization of time. Therefore, they hold great potential for logic design as they are free from several problems of clocked (synchronous) circuits. In principle, logic flow in asynchronous circuits is controlled by a request-acknowledgment handshaking protocol to establish a pipeline in the absence of clocks. Explicit handshaking blocks for small elements, such as bit adders, are expensive. Therefore, it is implicitly and efficiently managed using dual-rail carry propagation in adders. A valid dual-rail carry output also provides acknowledgment from a single-bit adder block. Thus, asynchronous adders are either based on full dual-rail encoding of all signals (more formally using null convention logic [2] that uses symbolically correct logic instead of Boolean logic) or pipelined operation using single-rail data encoding and dual-rail carry representation for acknowledgments. While these constructs add robustness to circuit designs, they also introduce significant overhead to the average case performance benefits of asynchronous adders. Therefore, a more efficient alternative approach is worthy of consideration that can address these problems.

This brief presents an asynchronous parallel self-timed adder (PASTA) using the algorithm originally proposed in [3]. The design of PASTA is regular and uses half-adders (HAs) along with multiplexers requiring minimal interconnections. Thus, it is suitable for VLSI implementation. The design works in a parallel manner for independent carry chain blocks. The implementation in this brief is unique as it employs feedback through XOR logic gates to constitute a single-rail cyclic asynchronous sequential adder [4]. Cyclic circuits can be more resource efficient

than their acyclic counterparts [5], [6]. On the other hand, wave pipelining (or maximal rate pipelining) is a technique that can apply pipelined inputs before the outputs are stabilized [7]. The proposed circuit manages automatic single-rail pipelining of the carry inputs separated by propagation and inertial delays of the gates in the circuit path. Thus, it is effectively a singlerail wave-pipelined approach and quite different from conventional pipelined adders using dual-rail encoding to implicitly represent the pipelining of carry signals.

LITERATURE REVIEW

This brief presents a parallel single-rail self-timed adder. It is based on a recursive formulation for performing multi bit binary addition. The operation is parallel for those bits that do not need any carry chain propagation. Thus, the design attains logarithmic performance over random operand conditions without any special speedup circuitry or look-ahead schema. A practical implementation is provided along with a completion detection unit. The implementation is regular and does not have any practical limitations of high fanouts. A high fan-in gate is required though but this is unavoidable for asynchronous logic and is managed by connecting the transistors in parallel. Simulations have been performed using an industry standard toolkit that verify the practicality and superiority of the proposed approach over existing asynchronous adders.

An adder or summer is a digital circuit that performs addition of numbers. In many computers and other kinds of processors, adders are used not only in the arithmetic logic unit(s), but also in other parts of the processor, where they are used to calculate addresses, table indices, and similar operations. Although adders can be constructed for many numerical representations, such as binary-coded

decimal or excess-3, the most common adders operate on binary numbers. In cases where two's complement or ones' complement is being used to represent negative numbers, it is trivial to modify an adder into an adder–subtractor. Other signed number representations require a more complex adder.

As technology scales down into the lower nanometer values power, delay, area and frequency becomes important parameters for the analysis and design of any circuits. This brief presents a parallel single-rail selftimedadder. It is based on a recursive formulation for performing multibit binary addition. The operation is parallel for those bits that do not need any carry chain propagation. Thus, the design attains logarithmic performance over random operand conditions without any special speedup circuitry or look-ahead schema. A practical implementation is provided along with a completion detection unit. The implementation is regular and does not have any practical limitations of high fanouts. A high fan-in gate is required though but this is unavoidable for asynchronous logic and is managed by connecting the transistors in parallel. Simulations have been performed using anindustry standard toolkit that verify the practicality and superiority of the proposed approach over existing asynchronous adders.

A parallel self-timed adder (PASTA) is disclosed. It is based on recursive formulation and uses only half adders for performing multi-bit binary addition. Theoretically the operation is parallel for those bits that do not need any carry chain propagation. Thus the new approach attains logarithmic performance without any special speed-up circuitry or look-ahead schema. The corresponding CMOS implementation of the design along with completion detection unit is also

presented. The design is regular and does not have any practical limitations of fan-ins or fan-outs or complex interconnections. Thus it is more suitable for adoption in fast adder implementation in high-performance processors. The performance of the implementation is tested using SPICE circuit simulation tool by linear technology. Simulation results show its superiority over cascaded circuit adders. A constant time carry propagation is also achieved using the proposed implementation by tuning the CMOS parameters.

PROPOSED WORK

In this section, the architecture and theory behind PASTA is presented. The adder first accepts two input operands to perform halfadditions for each bit. Subsequently, it iterates using earlier generated carry and sums to perform half-additions repeatedly until all carry bits are consumed and settled at zero level.

Architecture of PASTA:

The general architecture of the adder is shown in Fig. 1. The selection input for two-input multiplexers corresponds to the Req handshake signal and will be a single 0 to 1 transition denoted by SEL. It will initially select the actual operands during SEL = 0 and will switch to feedback/carry paths for subsequent iterations using SEL = 1. The feedback path from the HAs enables the multiple iterations to continue until the completion when all carry signals will assume zero values.

State Diagrams:

Each state is represented by (Ci+1 Si) pair where Ci+1, Si represent carry out and sum values, respectively, from the ith bit adder block. During the initial phase, the circuit merely works as a combinational HA operating in fundamental mode. It is apparent that due to the use of HAs instead of FAs, state (11) cannot appear.

During the iterative phase (SEL = 1), the feedback path through multiplexer block is

activated. The carry transitions (Ci) are allowed as many times as needed to complete the recursion.

From the definition of fundamental mode circuits, the present design cannot be considered as a fundamental mode circuit as the input–outputs will go through several transitions before producing the final output. It is not a Muller circuit working outside the fundamental mode either as internally, several transitions will take place, as shown in the state diagram. This is analogous to cyclic sequential circuits where gate delays are utilized to separate individual states [4].
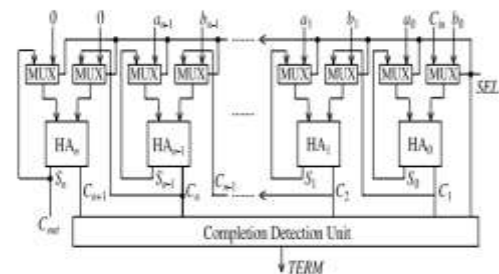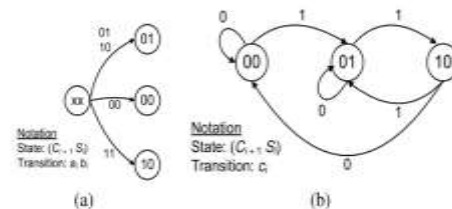

Fig. 1. General block diagram of PASTA.

Fig. 2. State diagrams for PASTA. (a) Initial phase. (b) Iterative phase.

Recursive Formula for Binary Addition:

Let S_{j}^{i} and C_{j}^{i+1} denote the sum and carry, respectively, for ith bit at the jth iteration. The initial condition (j = 0) for addition is formulated as follows:

$$S0_i = a_i \oplus b_i$$

$$C0_{i+1} = a_i b_i .$$

The jth iteration for the recursive addition is formulated by

$$S_{j}^{i} = S_{j-1}^{i} \oplus C_{j-1}^{i} , \quad 0 \leq i < n$$

$$C_{j}^{i+1} = S_{j-1}^{i} C_{j-1}^{i} , \quad 0 \leq i \leq n.$$

The recursion is terminated at kth iteration when the following condition is met:

$$Ck\ n + Ck\ n-1 + \cdots + Ck\ 1 = 0,\ 0 \le k \le n.$$

Now, the correctness of the recursive formulation is inductively proved as follows.

We now consider the $(l + 1)$th bit state ($Ck\ l+2$, $Sk\ l+1$) for kth iteration. It could also be in any of $(0, 0)$, $(0, 1)$, or $(1, 0)$ states. In $(k+1)$th iteration, the $(0, 0)$ and $(1, 0)$ states from the kth iteration will correctly produce output of $(0, 1)$ following (2) and (3). For $(0, 1)$ state, the carry successfully propagates through this bit level Thus, all the single-bit adders will successfully kill or propagate the carries until all carries are zero fulfilling the terminating condition.

The mathematical form presented above is valid under the condition that the iterations progress synchronously for all bit levels and the required input and outputs for a specific iteration will also be in synchrony with the progress of one iteration. In the next section, we present an implementation of the proposed architecture which is subsequently verified using simulations.

The existing design of PASTA is of 4 block size and therefore it is capable of adding two 3bit binary numbers. The circuit of existing design is elaborated in the below figure.

The select line for 2x1 multiplexers corresponds to the Request handshake signal and will be a single low to high (0 to 1) transition denoted by SEL. It will first select the actual inputs during SEL = 0 and switches to feedback (or) carry paths using SEL = 1. The feedback path from the half adders enables the multiple iterations to continue until the completion when all carry signals will assume zero (0) values. During the recursive phase (SEL = 1), the feedback path through multiplexers block will be at mux output. The carry changes (ci) are allowed as many times as needed to complete the recursion.

SIMULATION RESULTS
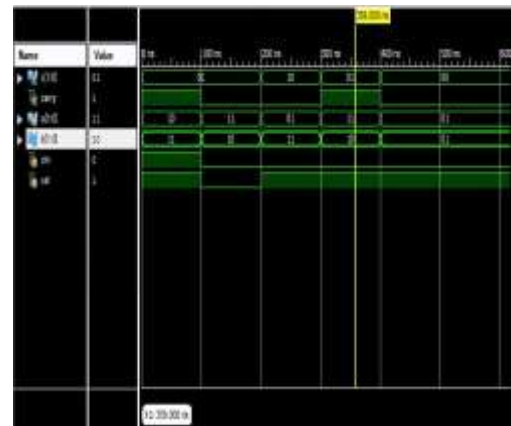
Simulation result of parallel self timed adder
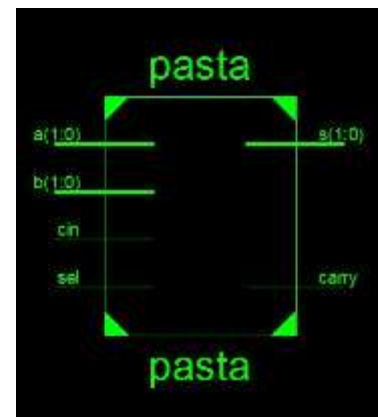
Block diagram of parallel self timed adder

CONCLUSION

This project presents an efficient design implementation of a parallel self timed adder of 4 block size (proposed) and whose operation is more efficient than traditional adders. The proposed design achieves a very simple n-bit adder i.e, RCA (Ripple Carry Adder) and similar to RCA in area wise as well as interconnection wise. The PASTA works in a parallel manner for independent carry chains and we also conclude that the implementation is very simple and the operation is very fast compared to the traditional carry chain adders. Simulation results of Modelsim tool verify the advantages of the proposed technique.

REFERENCES

[1] D. Geer, "Is it time for clockless chips? [Asynchronous processor chips]," *IEEE Comput.*, vol. 38, no. 3, pp. 18–19, Mar. 2005.

[2] J. Sparsø and S. Furber, *Principles of Asynchronous Circuit Design*. Boston, MA, USA: Kluwer Academic, 2001.

[3] P. Choudhury, S. Sahoo, and M. Chakraborty, "Implementation of basic arithmetic operations using cellular automaton," in *Proc. ICIT*, 2008, pp. 79–80.

[4] M. Z. Rahman and L. Kleeman, "A delay matched approach for the design of asynchronous sequential circuits," Dept. Comput. Syst. Technol., Univ. Malaya, Kuala Lumpur, Malaysia, Tech. Rep. 05042013, 2013.

[5] M. D. Riedel, "Cyclic combinational circuits," Ph.D. dissertation, Dept. Comput. Sci., California Inst. Technol., Pasadena, CA, USA, May 2004.

[6] R. F. Tinder, *Asynchronous Sequential Machine Design and Analysis: A Comprehensive Development of the Design and Analysis of Clock-Independent State Machines and Systems*. San Mateo, CA, USA: Morgan, 2009.

[7] W. Liu, C. T. Gray, D. Fan, and W. J. Farlow, "A 250-MHz wave pipelined adder in 2-μm CMOS," *IEEE J. Solid-State Circuits*, vol. 29, no. 9, pp. 1117–1128, Sep. 1994.

[8] F.-C. Cheng, S. H. Unger, and M. Theobald, "Self-timed carrylookahead adders," *IEEE Trans. Comput.*, vol. 49, no. 7, pp. 659–672, Jul. 2000.

[9] S. Nowick, "Design of a low-latency asynchronous adder using speculative completion," *IEE Proc. Comput. Digital Tech.*, vol. 143, no. 5, pp. 301–307, Sep. 1996.

[10] N. Weste and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective*. Reading, MA, USA: Addison-Wesley, 2005.

[11] C. Cornelius, S. Koppe, and D. Timmermann, "Dynamic circuit techniques in deep submicron technologies: Domino logic reconsidered," in *Proc. IEEE ICICDT*, Feb. 2006, pp. 1–4.

[12] M. Anis, S. Member, M. Allam, and M. Elmasry, "Impact of technology scaling on CMOS logic styles," *IEEE Trans. Circuits Syst., Analog Digital Signal Process.*, vol. 49, no. 8, pp. 577–588, Aug. 2002.