ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 8, Issue 2, February 2020

A Survey on Serial parallel Multiplication Techniques

A.Rishikesh[1], S.Asvika[2], M.Hariharan[3], G.Pasupathi[4], L.Arulleo Felix[5]

U.G. Student, Department of Electronics and Communication Engineering, Trichy Engineering College, Sivagnanam Nagar, Konalai, Trichy, Tamil nadu, India[1, 2, 3 & 4]

Assistant Professor, Department of Electronics and Communication Engineering, Trichy Engineering College, Sivagnanam Nagar, Konalai, Trichy, Tamil nadu, India[5]

ABSTRACT: Multiplication is arguably the most important primitive for digital signal processing (DSP) and machine learning(ML) applications, dictating the area, delay and overall performance of parallel implementation. The work on the optimization of multiplication circuits has been extensive. And presents a survey of Booth multiplier on various techniques in order to achieve low power circuits. It defines the complications met by the engineers at the physical design abstraction and reviews some of the techniques which are proposed to overcome these difficulties.

KEYWORDS: VLSI, FPGA, ASIC, MSD, LSB, ML, DNN, GPU.

I. INTRODUCTION

A Multiplier plays an important role in today's digital signal processing and various other applications. With advances in technology, many researchers have tried and are trying to design multipliers which offer either of the following design targets. They are high speed, low power consumption, regularity of layout and hence less area or even combination of them in one multiplier thus making them suitable for various high speed, low power and compact VLSI implementation. The common multiplication method is "add and shift" algorithm. In parallel multipliers number of partial products to be added is the main parameter that determines the performance of the multiplier. To reduce the number of partial products to be added, Modified Booth algorithm is one of the most popular algorithms. Booth algorithm is a multiplication algorithm that multiplies two signed binary numbers in two's complement notation. The algorithm was invented by Andrew Donald Booth in 1950. Booth used desk calculators that were faster at shifting than adding and created the algorithm to increase their speed. Booth's algorithm is of interest in the study of computer architecture. It is a powerful algorithm for Signed number multiplication, which treats both positive and negative numbers uniformly. For the standard add-shift operation, each multiplier bit generates one multiple of the multiplicand to be added to the partial product. If the multiplier is very large, then a large number of multiplicands have to be added. In this case the delay of multiplier is determined mainly by the number of additions to be performed. Booth algorithm multiplier will reduce the number of multiplicand multiples.

II. TECHNIQUES

A.PARALLEL ARITHMETIC OPERATORS

Online arithmetic has been widely studied for ASIC implementation. Online components were originally designed to perform computations in digit serial with most significant digit (MSD) first, resulting in the ability to chain arithmetic operators together for low latency. More recently, research has shown that digit parallel online operators can fail more gracefully when operating beyond the deterministic clocking region in comparison to operators with conventional arithmetic. Unfortunately, the utilization of online arithmetic operators in the past has required a large

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 8, Issue 2, February 2020

area overhead for FPGA implementation. In this paper, we propose novel approaches to implement the key primitives of online arithmetic, adders and multipliers, efficiently on modern Xilinx FPGAs with 6-input LUTs and carry resources.

B. BIT-SERIAL MULTIPLIER

Generally serial multipliers are of two types they are serial-serial multipliers and serial-parallel multipliers. In serial-serial multipliers both the operands are loaded in bit serial fashion, in order to reduce data input pads two. Where as in serial-parallel multiplier loads one operand in a bit serial fashion and the other in always available in parallel fashion. The existing architectures in serial-serial multipliers have so many disadvantages. The architecture proposed by Manas use a csa architecture in which the area occupied by the reduction block is very high because of large number of full adders used. It also has large delay due to propagation through adders.

C. SHIFT AND ADD MULTIPLIER

For standard add shift algorithm, every multiplier bit gives one multiple of the multiplicand which is added to partial products. A more number of multiplicands are added, if multiplier is very large. In this situation the number of additions to be performed determines the delay of multipliers. The performance will get better, if the number of additions are minimum. The value of multiplicand to be accumulated and added depends on the value of multiplier LSB. At each clock cycle the multiplier is shifted one bit to the right and its value is tested. If it is a 0, then only a shift operation is performed. If the value is a 1, then the multiplicand is added to the accumulator and is shifted by one bit to the right. After all the multiplier bits have been tested the product is in the accumulator. The accumulator is 2Q (P+Q) in size and initially the Q, LSBs contains the Multiplier. The delay is Q cycles maximum. Serial multipliers consume more power. So power is an important criterion there we should prefer parallel multipliers like booth multipliers to serial multipliers. The parallel multipliers like booth multiplier perform the computations using very few adders and very few iterative steps. As a result of which they cover minimum space as compared to the serial multiplier.

D.NEURAL NETWORK COMPUTING

Deep neural networks (DNNs) are the state-of-the-art technique in many recognition tasks such as object and speech recognition . DNNs comprise a feed-forward arrangement of layers, each exhibiting high computational demands andparallelism which are commonly exploited with the use of Graphics Processing Units (GPUs). However, the high computation demands of DNNs and the need for higher energy efficiency motivated special purpose architectures such as thestate-of-the-art DaDianNao (*DaDN*) which was reported to be up to 330x more energy efficient than a GPU . As power tends to be the limiting factor in modern high-performance designs, it is essential to achieve better energy efficiency inorder to improve performance further under the given power constraints. This work presents *Stripes* (*STR*), an implementation of a DNN performance improvement technique that 1) is complementary to existing techniques which exploit parallelism across computations, while 2) improving energy efficiency, and 3) enabling accuracy vs. performance trade offs. *STR* goes beyond parallelism across computations and exploitsthe data value representation requirements of DNNs. *STR* ismotivated by the observation that the precision required by DNNs varies significantly not only across networks but also across the layers of the same network**.**

E. ARRAY MULTIPLIER

Laxman S, Darshan Prabhu R, Mahesh S Shetty ,Mrs. Manjula BM,Dr. Chirag Sharma have presented this algorithm. The detailed study of different multipliers based on Array Multiplier, Constant coefficient multiplication (KCM) and multiplication based on vedic mathematics is introduced in this work. With an array multiplier two binary numbers will be multiplied by using of an array of half adders and full adders. Simultaneously addition of the different product terms is done in this array. By using an array of AND gates, the partial product terms are formed. Following this an array of AND gates, the adder array is used. The hardware structure for an p x q bit multiplier is described as (p

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 8, Issue 2, February 2020

x q) AND gates (p-1)q adders. Here q Half adders and (p-2). q Full adders. Array multiplier doing the multiplication process in traditional way. It looks like regular structure. Hence wiring and the layout are done in a much simplified manner. Add and shift algorithm is employed in an array multiplier. Implementation of this multiplier is simple but it requires larger area, with considerable delay.

III. CONCLUSION

It is concluded that the parallel multipliers are much option than the serial multiplier. In case of parallel multipliers, the total area is much less than that of serial multipliers. Hence the power consumption is also less. It has lesser delay and good noise immunity. Neural networkshas much more advanced techniques.particularly work well on some particular class of problems like image recognition. The neural network algorithms are very calculation intensive.To multiply two numbers by paper and pencil, the algorithm is to take the digits of the multiplier one at a time from right to left, multiplying the multiplicand by a single digit of the multiplier and placing the intermediate product in the appropriate positions to the left of the earlier results.

REFERENCES

1.P. Judd, J. Albericio, T. Hetherington, T. M. Aamodt, and A.Moshovos, "Stripes: Bit-serial deep neural network computing," in*Proc. 49th Annu. IEEE/ACM Int. Symp. Microarchitecture*, Oct. 2016,pp. 1–12.

2. B. Rashidi, S. M. Sayedi, and R. R. Farashahi, "Design ofa low-power and low-cost Booth-shift/add multiplexer-based multiplier,"in *Proc. Iranian Conf. Elect. Eng. (ICEE)*, May 2014,pp. 14–19.

3. P. Kalivas, K. Pekmestzi, P. Bougas, A. Tsirikos, and K. Gotsis, "Low latencyand high-efficiency bit serial-serial multipliers," in *Proc. 12th Eur. Signal Process. Conf.*, Sep. 2004, pp. 1345–1348.

4. B. Rashidi, "High performance and low-power finite impulse responsefilter based on ring topology with modified retiming serial multiplier onFPGA," *IET Signal Process.*, vol. 7, no. 8, pp. 743–753, Oct. 2013.

5. K. S. Trivedi and M. D. Ercegovac, "On-line algorithms for divisionand multiplication," *IEEE Trans. Comput.*, vol. C-26, no. 7,pp. 681–687, Jul. 1977.

6. K. Shi, D. Boland, and G. A. Constantinides, "Efficient FPGA implementationof digit parallel online arithmeticoperators," in *Proc. Int.Conf. Field-Program. Technol.*, Dec. 2014, pp. 115–122.