

Pipeline Architecture for $N=K*2L$ Bit Modular ALU: Case study between current generation computing and Vedic computing

Chiranjeevi G.N

Department of Electronics and Communication Engineering
PESIT-Bangalore South Campus, Bangalore 560100
Affiliated to Visvesvaraya Technological University, Belagavi,
Karnataka, India
chiranjeevign @pes.edu

Subhash Kulkarni

Department of Electronics and Communication Engineering
PESIT-Bangalore South Campus, Bangalore 560100
Affiliated to Visvesvaraya Technological University, Belagavi,
Karnataka, India
sskul@pes.edu

Abstract: This paper describes a design architecture that performs mathematical operations using Vedic sutra for upward compatibility in pipeline manner. In spite of increasing the area, performance and reducing power, Vedic architecture have observed to be inherently compatible with higher efficiency for pipeline architecture. However Vedic architecture leads to additional flexibility starting from 2-bit base modules to 8-bit modules ($L=1,2,3$) and pipeline can be compactable to any base modules for any given length.

Many researchers proposed arithmetic algorithms at simulation level using vedic sutra. These algorithm have been evaluated with better performance, area and speed. The literature has been widely found to be towards individual arithmetical operators like multiplier, square and cube and so on. A consolidated computing architecture, especially N-bit ALU is yet be realized Generalized N-bit ALU, can always be realized using pipeline modular architecture. The proposition is on realizing N-bit using ' 2^L ' as base modules using 'K' modules in pipelining. The authors have extensively verified modular architecture for 4-Bit modules for 16 bit and 32 bit pipelined operations. Individually multiplication using Urdhva Tiryakbhyam, division using Dhvajanka sutra, square using Dwandwayoga sutra. MAC unit which involves multiplication algorithms used in FFT and IFFT using sutras of Vedic mathematics and it is possible to achieve reduce version interms of speed and delay, compared to different generations of ALU

The authors are now exploring N-bit ALU architecture FPGA implementation using Vedic sutras with flexible modular pipeline architecture and mainly targeted for Digital Signal processing applications.

Keywords: modular architecture, Vedic computing, arithmetic algorithms of ALU, pipeline computing, digital signal processing

I. INTRODUCTION

In this paper an attempt is made to indicate pipeline architecture felicitation, which is equally flexible with Vedic sutras. on the contrary it is not been used in conventional ALU approach, Here N-bit CPU with flexibility of Vedic architecture components in pipeline manner with data handling capability of any lower values of N especially divided by 2 number has been suggested and extendible in cascade.

There is wide scope for exploiting the unique

characteristics exhibited by Vedic sutras especially insensitivity to direction while computing the result in arithmetic operation.

II. REVIEW WORK

Vedic sutras have been influencing computing architecture researchers in seeking alternatives for rapid computation since last 2 decades. Majority of the research literature indicates the use of Urdhva Tiryagbhyam sutra for efficient multiplication. Though literature survey indicates wide usage of the sutras in hardware implementation of architecture, it has been found exceptionally for individual arithmetic operation. Attempts have been made to use sutras on application based eg. MAC for Signal Processing, custom architecture for specific domain applications. A generic architecture for application development has been a void area in literature and has been an area of interest in the current era. This requires exploration towards an integrated Arithmetic Unit implementation. In this work we have proposed a modular and pipeline arithmetic unit that has following unique features. The architecture works on N-bit size, where N can be 8 – 64 bit with modular cascade facility having $N = fn(K, L)$, with $K=2, 4$ or 8 and $L=1, 2, 3$ or 4.

Higher value of N – bit architecture exhibits downward compatibility without any additional hardware. The Arithmetic unit proposed works for all operations of $\{+, -, \times, /\}$. The core architecture is implemented with multiplication and addition, whereas subtraction and division uses the same addition and multiplication with complement data representation.

The pipeline architecture ($N= 2L *K$) has been successfully implemented by the authors in their earlier work on 32 bit MAC unit used for DSP applications [7]. Hardware implementation of the Arithmetic unit has been verified on Spartan 3E Board/Basys3(artix_7) family. For the Xilinx Spartan and Artix family, it is found that the gate delay significantly reduces by using Vedic math sutras. The present work proposes a flexible, modular and pipeline facilitation for a generic arithmetic unit to be used for any application.

III. VEDIC SUTRAS FOR ALU

Conventional available architecture on ALU uses arithmetic operations especially addition and subtraction using serial adder. For N which is evolved from 16 bit in 1980's to 128bit in current era. Generally subtraction is implemented with complementary addition. Multiplication and division is commonly implemented using successive



addition operation and successive subtraction respectively

Scope exists largely for multiplication and division using vedic sutras, in this work we have targeted multiplication and division feasibility in FPGA that uses urdhva tiryakbhyam sutra for multiplication similarly division is implemented using successive complement addition

A. Square using Dwandwayoga Sutra

Here in order to compute the square of number, we use Duplex property of Urdhva Triyakbhyam. For square operation a dedicated hardware can improve its performance compare to multiplier architecture

1) Algorithm for 4 x 4 bit Square Using Urdhva Tiryakbhyam D – Duplex

For example:

$X_3 X_2 X_1 X_0$ Multiplicand

$Y_3 Y_2 Y_1 Y_0$ Multiplier

H G F E D C B A-- final product

2) Parallel Computation

1. $D = X_0 * Y_0 = A$
2. $D = 2 * X_1 * Y_0 = B$
3. $D = 2 * X_2 * Y_0 + X_1 * Y_1 = C$
4. $D = 2 * X_3 * Y_0 + 2 * X_2 * Y_1 = D$
5. $D = 2 * X_3 * Y_1 + X_2 * Y_2 = E$
6. $D = 2 * X_3 * Y_2 = F$
7. $D = X_3 * Y_3 = G$

B. Algorithm for Multiplication using Urdhva tiryakbhyam

Here the multiplier is implemented using vertical and crosswise method. This is a common method used for all the possible cases of multiplication. We can achieve parallelism in generation of partial products and its final results .

IV. DESIGNIMPLEMENTATION

The proposed Arithmetic Module has first been split into three smaller modules (shown in fig 1), that is

1. Multiplier
2. MAC unit
3. Arithmetic module,

As a whole. These modules have been made using Verilog HDL and simulated for different case study and synthesized in Xilinx Vivado 2016.1 with Spartan 3 family and XC3S400 device and Zynq700 Zedboard as target.

Arithmetic block is considered as unique and important functional block. It handles all the Arithmetic and logic operations that are required for user constraints.in this proposed method arithmetic block is implemented using Vedic algorithms and priority is given for multiplication. To implement addition and subtraction conventional method is used.

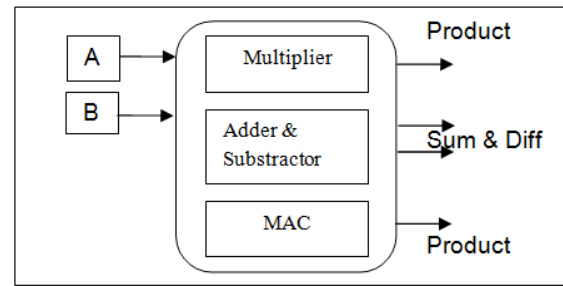


Fig. 1. Basic block diagram of Arithmetic unit

Design starts with the implementation of multiplier design of size $2 \times 2(2^1 * 1)$, where it is equivalent to $2L * K$ bit multiplier

A. 2×2 bit Multiplier($2^1 * 1$) where $K=1$ stage

In 2×2 bit multiplier, the multiplicand has 2 bits each and the result of multiplication is of 4 bits. So in input ranging from (00) to (11) and output lies in the set of (0000, 0001, 0010, 0011, 0100, 0110, 1001).

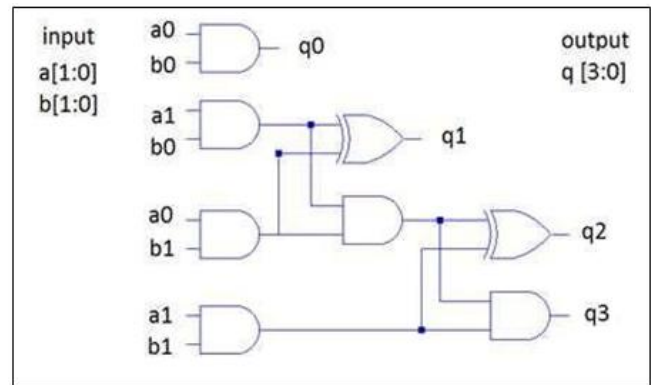


Fig. 2. Hardware Implementation of $2^1 * 1$ (where $L=1, K=1$)

B. 4x4 bit Multiplier

The 4x4 Multiplier is made by using $K=4$, 2×2 multiplier blocks. Here, the multiplicands are of bit size ($n=4$) where as the result is of 8 bit size. The input is broken into smaller blocks of size of $n/2 = 2$, for both inputs, that is a and b. These newly formed block of 2 bits are given as input to 2×2 multiplier block and the result produced 4 bits, which are the output produced from 2×2 multiplier block are sent for addition to an addition tree.

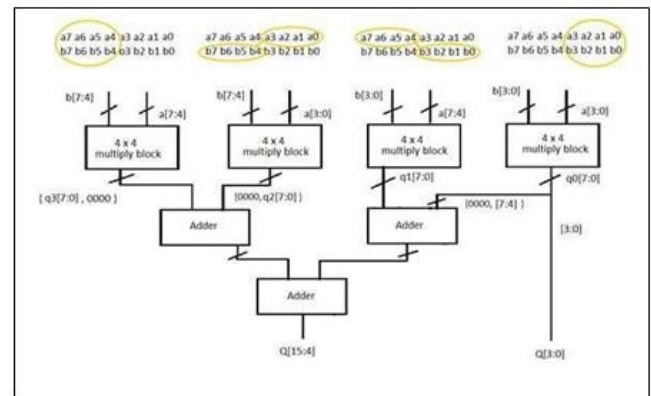


Fig. 3. Hardware Implementation of $2^2 * 4$ (where $L=2, K=4$)

C. 8x8 bit Multiplier ($2^3 \times 4$) where $K=4$ or 8 stages

The 8x8 Multiplier is made by using 4, 4x4 multiplier blocks or using 8, 2x2 multiplier Here, the multiplicands are of bit size ($n=8$) where as the result is of 16 bit size. The input is broken into smaller size of $n/2 = 4$, for both inputs, that is a and b, just like as in case of 4x4 multiply block. These newly formed block of 4 bits are given as input to 4x4 multiplier block, where again these new chunks are broken into even smaller of size $n/4 = 2$ and fed to 2x2 multiply block. The result produced, from output of 4x4 bit multiply block which is of 8 bits, are sent for addition to an addition tree Fig 4: Hardware Implementation of $2^3 \times 4$ (where $L=3$, $K=4$)

The Arithmetic module designed in this work, makes use of 4 components that are, Adder, Subtractor, and Multiplier. As a subtraction, and multiplication on n -bit data. The arithmetic unit uses conventional adder and subtractor, while the multiplier unit are made using Vedic Mathematics Algorithm. The control signals which guide the Arithmetic unit to perform a particular operation, i.e. Addition, subtraction, multiplication operation are s_0 and s_1 , which are provided by the control circuit. The status of control lines s_0 and s_1 and the corresponding arithmetic operation being performed.

TABLE I. CONTROL SIGNALS FOR DIFFERENT CHOICE FROM USER
END

S1	S0	Operations
0	0	addition
0	1	subtraction
1	0	multiplication
1	1	others

V. PERFORMANCE EVALUATION AND ARCHITECTURE ASSESSMENT

Based on the simulation and synthesis of computing architecture the performance evaluation over speed and area (in terms of LUT's) is presented under the following testbench specifications.

In the test bench FPGA family is chosen over two generation of Spartan and vertex family. Observation and discussions are presented for performance of area and speed for $N=K \times 2L$, where K is 2,4,8 and L is 1,2,3,4 to frame architecture of $N=32$ bit and $N=64$ bit and higher level based on user requirements and constraints.

The greatest advantage over other multiplier is time delay is reduced as number of bits increases from 8×8 to 16×16

And the area required or utilized is very less compared to other conventional methods. As per area utilization report vedic square requires 259 count and array multiplier requires 590 for 16×16 . The results are extracted for target device as Zynq7000.

TABLE II. TOTAL DELAY FOR VEDIC AND CONVENTIONAL MULTIPLIER

N bit Total delay in ns	Vedic multiplier	Booth multiplier	Array multiplier
8*8	18ns	20ns	22ns
16*16	32ns	37ns	43ns

For square operation the total delay has been reduced from 25.380ns (as in normal multiplication) to 13.9ns in case of using Vedic mathematics sutras. Number of slices has also been reduced from 72 to 36 out of 4656.

For multiplication, total delay has been reduced from 25.380ns to 19.868ns with almost same number of slices.

Number of 8-bit and 16-bit adders/subtractors have also been reduced when design is implemented using Vedic Mathematics Sutras.

TABLE III. PERFORMANCE EVALUATION FOR AREA AND DELAY

Operation	Performance evaluation	Vedic Multiplication	Normal Multiplication
square	Delay	15.35ns	25.38ns
Multiplication		19.86ns	25.38ns
Multiplication	Area	72 out of 4656 LUT's	36 out of 4656 LUT's

VI. CONCLUSION

Above implementation and analysis it can lead to independent computing when the application word length is smaller than the architecture word length, N bit processor by product flexible and it posses potentially becoming parallel processor also. If 64 bit processor is will have control signals to enable and disable the lower components (32 bit or 16 bit) as per user requirements and can use for both pipeline and parallel architecture modes. As an alternate for downward compactable applications user will have choice of successive blocks.

Proposed architecture works well for multiplication in pipeline, since addition and subtraction happened to be redundant in pipeline it continues to be implemented in array form. Division can also be pipelined with multiplication, provided fractional dividend is replaced by integer $72/4$ is represented as $3 \times (24/4)$ where dividend $24/4$ is equivalent to 6×3 hence for division any initial pre-processing is required for implementation in pipelining and array division is well implemented by Dhvajanka sutra.

ACKNOWLEDGEMENT

This work is carried out under research Center of Electronics and Communications department in PESIT Bangalore South Campus, which is recognized by Visvesvaraya Technological University, Belgavi.

REFERENCES

- [1] Harpreet Singh Dhillon and Abhijit Mitra, "A Reduced- Bit Multiplication Algorithm for Digital Arithmetic's", International Journal of Computational and Mathematical Sciences, Spring 2008
- [2] Ruchi anchaliya, Chiranjeevi G N, Subhash Kulkarni, "Efficeint Computing Techniques using Vedic Mathematics Sutras", international journal of innovative research in electrical, Electronics, Instrumentation and Control engineering, volume 3, Issues 5, May 2015
- [3] Himanshu Thapliyal, Saurabh Kotiyal and M. B Srinivas, "Design and Analysis of A Novel Parallel Square and Cube Architecture Based On Ancient Indian Vedic Mathematics", Centre for VLSI and Embedded System Technologies, International Institute of Information Technology, Hyderabad, 500019, India, 2005 IEEE
- [4] Vaijyanath Kunchigi, linganagouda kulkarni and subhash kulkarni 32-bit MAC unit design using Vedic multiplier – published at: "International Journal of Scientific and Research Publications (IJSRP), Volume 3, Issue2, February 2013 Edition".

- [5] Sumita Vaidya and Deepak Dandekar, "Delay- Power Performance comparison of Multipliers in VLSI Circuit Design", International Journal of Computer Networks & Communications (IJCNC), Vol.2, No.4, July 2010.
- [6] Umesh Akare, T.V. More and R.S. Lonkar, "Performance Evaluation and Synthesis of Vedic Multiplier", National Conference on Innovative Paradigms in Engineering & Technology (NCIPET- 2012), proceedings published by International

BIOGRAPHY

Chiranjeevi G.N. is Assistant Professor at PES University EC Campus (PESIT Bangalore South Campus), Karnataka, India Areas of interest are in the field of VLSI, FPGA and Image Processing applications. Currently the research interest in low power VLSI, FPGA and Medical Image processing.

Dr. Subhash Kulkarni is Professor and Principal at PESIT South Campus, Bangalore. Dr. Subhash Kulkarni is having more than 30 years of Academic Teaching Experience. His area of interest fields are control systems and math models in DSP applications, field of Vedic mathematics (ALU operations) for fast processor.