International Journal for Modern Trends in Science and Technology, 9(05): 897-900, 2023

ISSN: 2455-3778 online
DOI: https://doi.org/10.46501/IJMTST0905153
Available online at: http://www.ijmtst.com/vol9issue05.html

Design of multiplier with low latency Approximate 5-2 compressor

K. Santhi Sagar Reddy , P. Akanksha ,T.Asha, O.Pravallika

Department of Electronics and Communication Engineering, Narayana Engineering College, Nellore, Andhra Pradesh, India

To Cite this Article
A. Siva Sai Kumar, V. Madhu, Sk. Ahmed Kabeer, P. Yaswanth, M. Bhuvaneswar, P. Venkata. Design of multiplier with low latency Approximate 5-2 compressor. International Journal for Modern Trends in Science and Technology 2023, 9(05), pp. 897-900. https://doi.org/10.46501/IJMTST0905153

Article Info
Received: 21 April 2023; Accepted: 20 May 2023; Published: 24 May 2023.

ABSTRACT

Multipliers one of the fundamental units in computer arithmetic blocks. Also, the delay in the system's architecture is an important factor. In this project, it is proposed to design and analyze two novel approximate 5-2 compressors with less delay than the present exact compressors systems, while maintaining precision equivalent to the current systems. These delay efficient approximate compressors can be used in the parallel multipliers to accelerate the compression process. Approximate computing can be applied in some digital signal processing application that can tolerate errors in permissible range, where the approximate compressors come in handy. Compared with exact4-2 compressors, the proposed approximate 5-2 compressors achieve a significant reduction in delay.

KEYWORDS: Xilinx 14.7 Software

1. INTRODUCTION

Exact computing units aren't necessarily essential in applications like data mining and multimedia signal processing. They may be substituted with a similar item. Research into error-tolerant applications using approximation computation is on the increase. These applications rely heavily on adders and multipliers. In digital signal processing, approximate complete adders are suggested at the transistor level. Partial product accumulation in multipliers is handled by their suggested full-adder design.

The use of truncation in fixed-width multiplication designs is common to simplify the circuitry. In order to compensate again for quantization error induced by the reduced portion, a variable correction component is added. Accumulation of bits is critical in terms pf power usage when using approximation methods in multipliers. If the least relevant bits of the inputs can be trimmed, then partial products may be formed in order to decrease hardware complexity. In partial product accumulations, the suggested multiplier saves just a few adder circuits. A partial product reduction trees of four 8 x 8Dadda multiplier variations is given and applied with two types of roughly 4-2 compressors.

II. LITERATURE SURVEY

There has been a lot of study on approximate computing over the last decade, but much of it has focused on adders, which are hardware abstractions. Both traditional design measurements and approximation computing metrics are available for comparison. For computer arithmetic designs, inexact computing is

especially intriguing. In order to be used in a multiplier, the authors discuss the study and construction of two novel compressors with an approximate 4-2 compression ratio. A Dadda multiplier may be multiplied using four distinct techniques for using the suggested approximation compressors. There is substantial decrease in power consumption, delay and number of transistors compared to that of an exact design. There are three parts to this approximation multiplier: a Booth encoder, a 4-2 compressor, and an approximation tree structure. For 8x8, 16x16, & 32x32-bit signed multiplication schemes, the approximate design is built and tested. This project presents and discusses simulation findings for 45 nm technology. Image processing is used to test the suggested multiplier

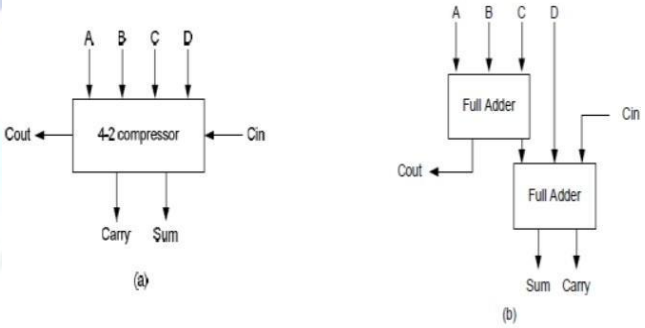
III. EXISTING METHOD

A Redundant Binary (RB) format may be utilized to create high-performance multipliers because of its flexibility and carry-free addition. The traditional RB multiplier adds an extra RB Partial Product (RBPP) row, since an Error-Correcting Word (ECW) is produced both by the radix-4 Modified Booth Encoding (MBE) or the RB encoding. For the MBE multiplier, this means an extra RBPP accumulation step. To save one RBPP accumulation step, a novel RB Altered Bits Generator (RBMPPG) has been developed in this study. RBMPPG is less wasteful than RB MBE multiplier because it creates fewer incomplete product rows. When the length of each multiplier operand is at least 32 bits, simulation findings demonstrate that the proposed RBMPPG-based designs greatly lower area and power consumption; these reductions over earlier NB multiplier designs result in a minor latency gain (approximately 5 percent). The suggested RB multipliers may minimize the power-delay product by up to 59 percent when compared to current RB multipliers.

IV. PROPOSED METHOD

Digital filters play a critical role in DSP. DSP's popularity is mostly due to its ability to perform at such a high level. Separation and restoration are the two primary functions ofafilter.Whenthesignalistaintedbyinterference,noise,orot hersignals,signalseparationisrequired.Consider,forexam ple,anEKGgadgetformonitoringababy'scardiacactivityw hilestillinthewomb.Themother'sbreathingandpulsewillli kelydistorttherawsignal.Thesesignalsmustbefilteredouts othattheymaybeevaluatedseparately.Whenasignalhas been distorted in some manner, signal restoration is utilized. It's possible to improve an audio recording that was recorded with low standards by filtering it. Using a lens that is not properly focused or a camera that is shaken is another example. Analog or digital filters may be used to comb these issues.

.



.1.1(a)adder compressor.
(b)4-2addercompressorimplementedwithfulladders.

Figures and Tables

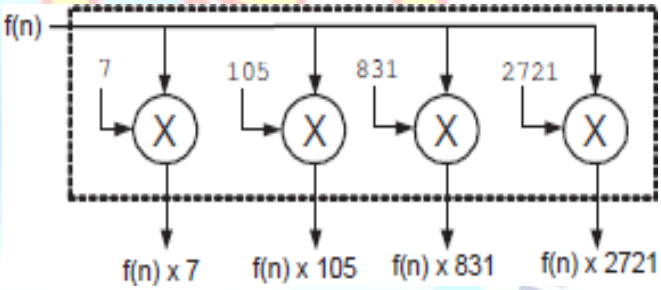


Fig:2.1Amultiplier-basedMCM example

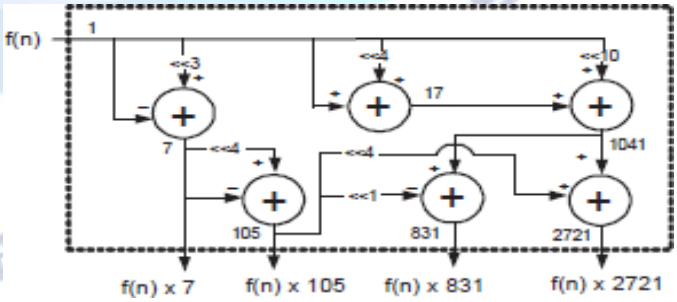


Fig:2.2 A multiplier less-based MCM exam**ple**

Equations

The straightforward method for decomposing the multiplication into adds and shifts translates 1's in

the binary representation of the constant t into shifts, and adds up the shifted inputs. For example ,fort= 71,

$$71x=1000111_2x=x\ll6+x\ll2+x\ll1+x$$

whichrequires3adds.Alternatively,the multiplication can be composed into subtracts and shifts by translating 0'sintoshifts,and subtracting from the closest constant consisting of 1's only (i.e., of theform2^n-1):

$$71x=1000111_2x=(x\ll7-x)-x\ll5-x\ll4-x\ll3 \quad (4.2)$$

Taking the best of the set wo methods yields in the worst and in the average case a solution with $\frac{b}{2}$+O(1) adds/subtracts, where b is the bit width of t.

A better digit-based method decomposes into both adds and subtracts by re-coding the number into the Canonical Signed Digit (CSD) representation [Aizen's 1961], which allows negative digits 1. Using CSD, the previous example can be improved to use only2add/subtract operations:

$$1000111_2x=1001001_{CSD}x=x\ll6+x\ll3-x$$

V. SOFTWARE IMPLEMENTATION USING XILINX

Xilinx Tools is a collection of digital circuit design software tools. Field-Programmable Gate Arrays(FPGA) as those from Xilinx or CPL devices are used(CPLD).For example, there are four stages in the design process: (a) the initial concept phase, (b) the actual design phase, and (c) functional simulation. A prototype entry tool, a Verilog Hardware Description (HDL) - VHDL or VHDL, or a mix of the two CAD tools may be used to input digital designs. Only the design flow using VHDL HDL will be used in this lab. As long as you start with VHDL HDL design requirements, you may use the CAD tools to create combinational and sequential circuits. The following are the stages in this design process:

a)Using a template-driven editor, create Verilog code design input files. Compile and design-simulation.

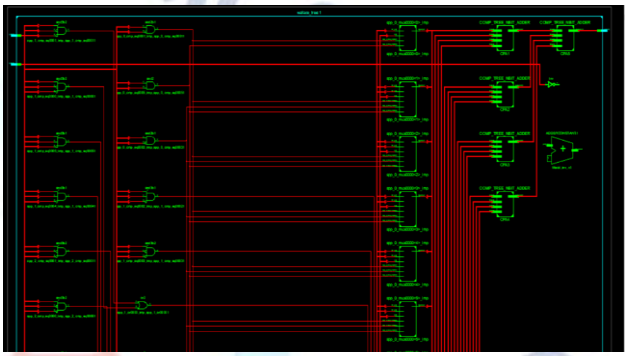
b)create the test vectors (functional simulation) instead of making use of a PLD (FPGA or CPLD).

c)Assign the design's input/output pins to a target device to implement it. Use an FPGA or CPLD to download the bit stream.

d) Interpretthe VHDL design file(s).

Testing on FPGA/CPLD device design.

VI.RESULTS



```
MUXCY:CI->O           1   0.013   0.000  CPA2/Madd_fsum_cy<59> (CPA2/Madd_fsum_cy<59>)
MUXCY:CI->O           1   0.013   0.000  CPA2/Madd_fsum_cy<60> (CPA2/Madd_fsum_cy<60>)
XORCY:CI->O           3   0.251   0.438  CPA2/Madd_fsum_xor<61> (n0538<61>)
LUT3:I0->O            2   0.043   0.347  CPA5/aa[61].b1/CCC1/Mmux_cout111 (CPA5/C<62>)
LUT5:I3->O            1   0.043   0.550  CPA5/aa[62].b1/CCC2/Mmux_sum111 (CPA5/sum<62>)
LUT6:I0->O            1   0.043   0.000  CPA5/Madd_fsum_lut<62> (CPA5/Madd_fsum_lut<62>)
MUXCY:S->O            0   0.230   0.000  CPA5/Madd_fsum_cy<62> (CPA5/Madd_fsum_cy<62>)
XORCY:CI->O           1   0.251   0.279  CPA5/Madd_fsum_xor<63> (p_63_OBUF)
OBUF:I->O                 0.000          p_63_OBUF (p<63>)
----------------------------------------
Total                     6.410ns (2.362ns logic, 4.048ns route)
                                  (36.8% logic, 63.2% route)

==========================================================================

oss Clock Domains Report:
-------------------------

==========================================================================

tal REAL time to Xst completion: 11.00 secs
tal CPU time to Xst completion: 10.95 secs

tal memory usage is 4620648 kilobytes

mber of errors   :    0 (   0 filtered)
mber of warnings :   10 (   0 filtered)
mber of infos    :    4 (   0 filtered)
```

DELAY

VI CONCLUSION

Using signals generated and propagated ,this project proposes efficient approximation multipliers. A basic OR gate is used to create changed partial products for approximation .Half-adder, full-adder, and 4-2 compressors are all recommended to minimize the left over partial products in the final product. In Multiplier1, approximations are applied to all n bits, but in Multiplier2, they are applied just to the n 1 least significant bit. The space and power consumption of Multiplier1 and Multiplier2 are significantly reduced compared to exact designs. Both Multiplier1 and Multiplier2 save 87 and 58 percent, respectively, from accurate multipliers in APP compared to previous approximation solutions.

ACKNOWLEDGEMENT

We are extremely grateful to Dr.P. NARAYANA, Ph.D. Founder, Narayana Educational Institutions, Andhra Pradesh for the kind blessings. We are extremely thankful to Mr. R Samba Siva Rao B. Tech, Registrar Narayana Engineering College, Nellore. We are much obliged to Dr. A.V.S Prasad, Ph.D. Director, Narayana Engineering & Pharmacy Colleges, for the continuous encouragement and support. We owe indebtedness to our Principal Dr.G. Srinivasulu Reddy, M.Tech., Ph.D., Narayana Engineering College, Nellore for providing us the required facilities. We express our deep sense of gratitude and sincere thanks to Dr. K. Murali, M.Tech, Ph.D. , Professor & HOD, Department of Electronics and Communication Engineering, Narayana Engineering College, Nellore for providing the necessary facilities and encouragement towards the project work. We thank our project guide, Dr.K. Santhi Sagar Reddy, MTech, Ph.D. Professor, Department of Electronics and Communication Engineering for his guidance, valuable suggestions and support in the completion of the project. We gratefully acknowledge and express our thanks to teaching and non-teaching staff of E.C.E Department. We would like to express our love and affection to our parents for their encouragement throughout this project

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, "Low-power -digital-signal processing using approximate adders, "IEEE Trans. Computer-Aided Design Integer. Circuits Syst., vol. 32, no. 1, pp. 124–137,Jan. 2013.

[2] E. J. King and E. E. Swartz lander, Jr., "Data-dependent truncation scheme for parallel multipliers,"in Proc.31stAsilomar Conf .SignalsCircuits Syst.,Nov.1998,pp.1178–1182.

[3] K.-J. Cho, K.-C. Lee, J.-G. Chung, and K. K. Pari, "Design low-error fixed-width modified both multiplier,"IEEETrans.VeryLarge-ScaleInteger.(VLSI)Syst.,vol.12, no.5, pp.522–531, May 2004.

[4] H. R. Mamdani, A. Ahmadi, S. M. Fakhri, and C. Lucas, "Bio-inspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications, "IEEE Trans. Circuits Syst. I,Reg. Papers, Vol. 57, no.4, pp. 850–862, Apr. 2010.

[5] A.Momani, J.Han ,P.Montecchio ,and F. Lombardi, "Design and analysis of approximate compressors for multiplication," IEEE Transcom., vol. 64, no. 4, pp. 984–994,Apr. 2015.

[6] S. Narayana Moorthy, H. A. Moghaddam, Z. Liu, T. Park, and N. SKim,"Energy-efficient approximate multiplication for digital signal processing and classification applications,"IEEETrans.VeryLarge-ScaleInteger.(VLSI)Syst.,vol.23,no.6,pp.1180–1184, Jun. 2015.

[7] G.Gervais,K.Tsunamis,S.Xydis,D.Souris,andK.Pelmets,"Design-efficientapproximate multiplication circuits through partial product perforation," IEEE Trans. VeryLarge-ScaleInteger. (VLSI) Syst.,vol. 24, no.10, pp. 3105–3117, Oct.2016.