

DESIGN OF EFFICIENT LOW POWER MULTIPLIER BY ANOMALY RETRENCHMENT EXPLOITATION

DIVYA SATHI BALAGAI¹, Y.RAJYA LAKSHMI²

Department Of Electronics & Communication Engineering
Sri Sivani College Of Engineering,
ChilakaPalem Jn, Srikakulam, India

Abstract-- This paper presents the design exploration and applications of combinational VLSI designs for multimedia/DSP purposes. This paper proposes an original glitch-diminishing technique to filter out useless switching power by asserting the data signals after the data transient period. The design of an efficient integrated circuit in terms of power, area, and speed simultaneously, has become a very challenging problem. Power dissipation is recognized as a critical parameter in modern. The objective of a good multiplier is to provide a physically compact, good speed and low power-consuming chip. To save significant power consumption of a VLSI design, it is a good direction to reduce its dynamic power that is the major part of total power dissipation. In this paper, we propose a high speed low-power multiplier adopting a modified Booth encoder which is controlled by a detection unit using an AND gate. The modified booth encoder will reduce the number of partial products generated by a factor of 2.

In this project we used Modelsim for logical verification, and further synthesizing it on Xilinx-ISE tool using target technology and performing placing & routing operation for system verification on targeted FPGA.

Keywords-- Modified Booth encoder, low power multiplier, detection logic unit.

I. INTRODUCTION

This paper presents the design exploration and applications of power suppression technique on multipliers for high-speed and low-power purposes. To filter out the useless switching power, there are two approaches, i.e., using registers and using AND gates, to assert the data signals of multipliers after the data transition. The power suppression technique has been applied on both the modified Booth decoder and the compression tree of multipliers to enlarge the power reduction. The simulation results show that the implementation with AND gates owns an extremely high flexibility on adjusting the data asserting time which not only facilitates the robustness but also leads to a 40% speed improvement. Adopting a 0.18- μ m CMOS technology, the

proposed multiplier dissipates only 0.0121 mW per MHz in H.264 texture coding applications, and obtains a 40% power reduction. To satisfy MOORE'S law and to produce consumer electronics goods with more backup and less weight, low power VLSI design is necessary.

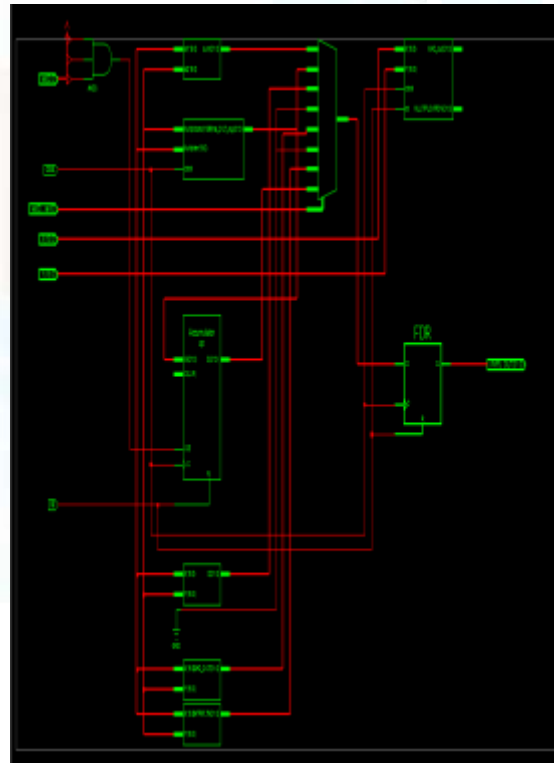


Fig.1 Block diagram Of Proposed Technique

II. Booth's Algorithm:

Booth's Algorithm is simple but powerful. Speed of VMFU is dependent on the number of partial products and

speed of accumulate partial product. Booth's Algorithm provide us to reduced partial products. We choose radix-4 algorithm.

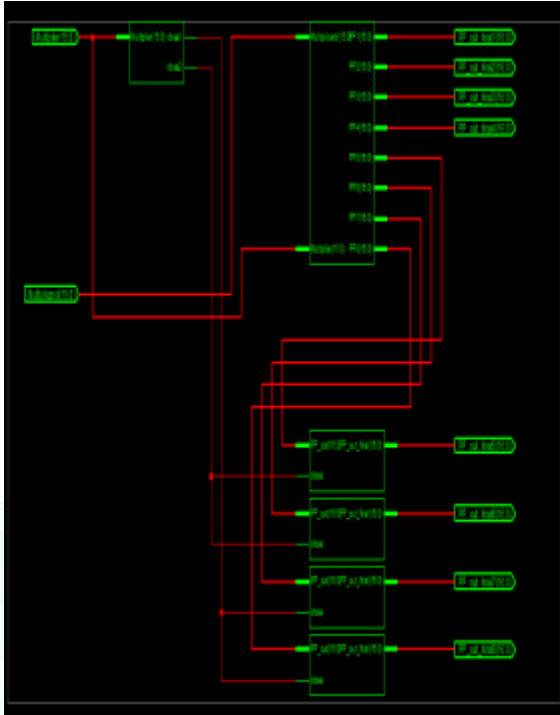


Fig. 2 Booth's Encoder Unit

A. A radix-2 modified Booth's algorithm:

- Original Booth's algorithm has an inefficient case. The 17 partial products are generated in 16bit x 16bit signed or unsigned multiplication.
- Modified Booth's radix-4 algorithm has fatal encoding time in 16bit x 16bit multiplication.

B. Modified Booth Encoder:

Multiplication consists of three steps: 1) the first step to generate the partial products; 2) the second step to add the generated partial products until the last two rows are remained; 3) the third step to compute the final multiplication results by adding the last two rows. The modified Booth algorithm reduces the number of partial products by half in the first step. We used the modified Booth encoding (MBE) scheme proposed in. It is known as the most efficient Booth encoding and decoding scheme. To multiply X by Y using the modified Booth algorithm starts from grouping Y by three bits and encoding into one of {-2, -1, 0, 1, 2}. Table I shows the rules to generate the encoded signals by MBE scheme and Fig. 1 (a) shows the corresponding logic diagram. The Booth decoder generates the partial products using the encoded signals as shown in

Fig.

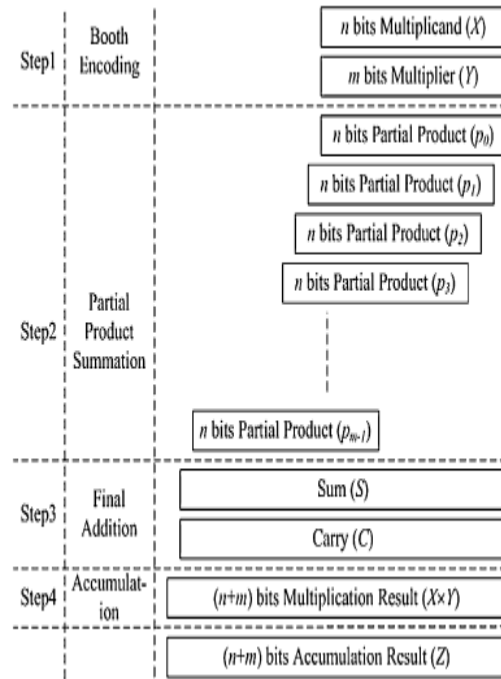


Fig.3 Steps in Booth's Encoder

The above Fig. shows the generated partial products and sign extension scheme of the 8-bit modified Booth multiplier. The partial products generated by the modified Booth algorithm are added in parallel using the Wallace tree until the last two rows are remained. The final multiplication results are generated by adding the last two rows. The carry propagation adder is usually used in this step.

Y_{i+1}	Y_i	Y_{i-1}	Value	$X1_b$	$X2_b$	Neg	Z
0	0	0	0	1	0	0	1
0	0	1	1	0	1	0	1
0	1	0	1	0	1	0	0
0	1	1	2	1	0	0	0
1	0	0	-2	1	0	1	0
1	0	1	-1	0	1	1	0
1	1	0	-1	0	1	1	1
1	1	1	0	1	0	1	1

Fig. 4 Truth table for MBE Scheme

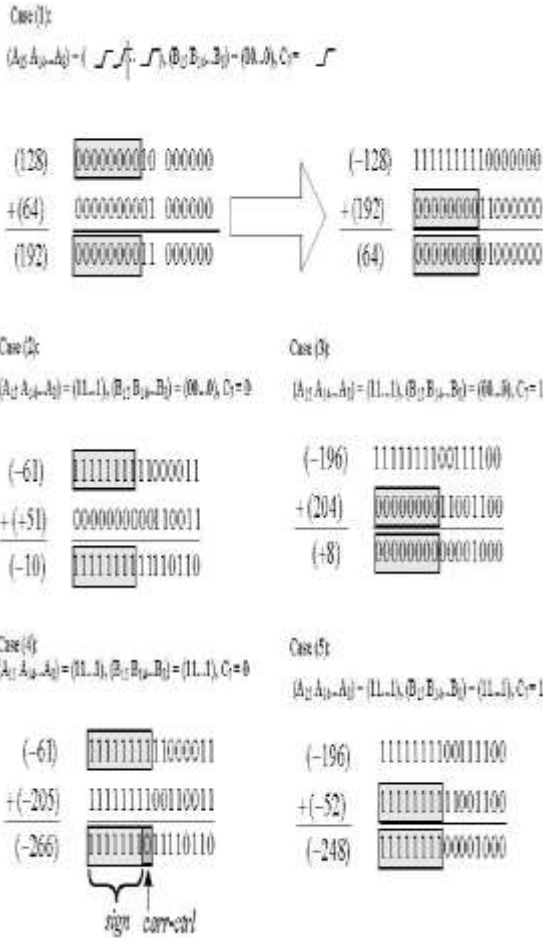


Fig. 5 internal logic

To know whether the MSP affects the computation results or not. We need a detection logic unit to detect the effective ranges of the inputs. The Boolean logical equations shown below express the behavioral principles of the detection logic unit in the MSP circuits of the SPST-based adder/subtractor.

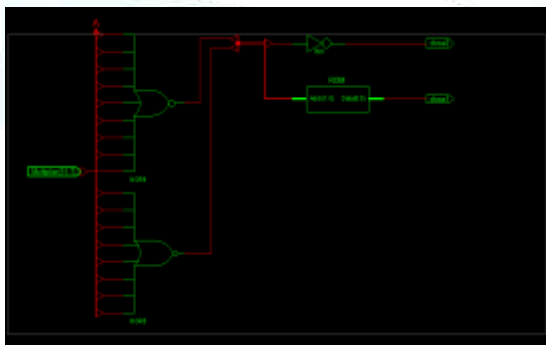


Fig. 6 Detection Logic Unit

where $A[m]$ and $B[n]$ respectively denote the m th bit of the operands A and the n th bit of the operand B , and $AMSP$ and $BMSP$ respectively denote the MSP parts, i.e. the 9th bit to the 16th bit, of the operands A and B . When the bits in $AMSP$ and/or those in $BMSP$ are all ones, the value of A_{and} and/or that of B_{and} respectively become one, while the bits in $AMSP$ and/or those in $BMSP$ are all zeros, the value of A_{nor} and/or that of B_{nor} respectively turn into one. Being one of the three outputs of the detection logic unit, close denotes whether the MSP circuits can be neglected or not. When the two input operand can be classified into one of the five classes as shown in figure 7.

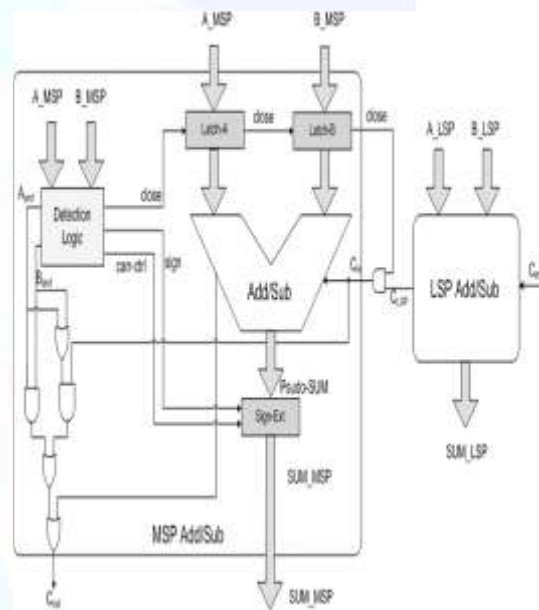


Fig. 7 Internal functioning unit

III. Device Programming

Now the design must be loaded on the FPGA. But the design must be converted to a format so that the FPGA can accept it. BITGEN program deals with the conversion. The routed NCD file is then given to the BITGEN program to generate a bit stream (a .BIT file) which can be used to configure the target FPGA device. This can be done using a cable. Selection of cable depends on the design.

A. Behavioral Simulation (RTL Simulation):

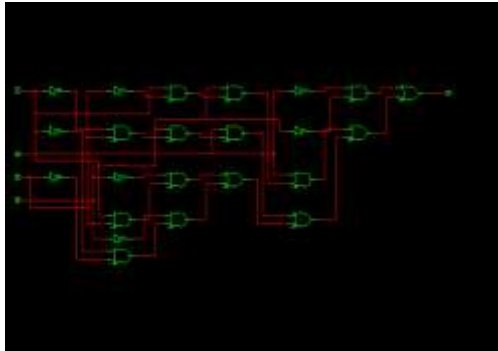


Fig. 8 LUT of Booth Encoder

This is first of all simulation steps; those are encountered throughout the hierarchy of the design flow. This simulation is performed before synthesis process to verify RTL (behavioral) code and to confirm that the design is functioning as intended. Behavioral simulation can be performed on either VHDL or Verilog designs. In this process, signals and variables are observed, procedures and functions are traced and breakpoints are set. This is a very fast simulation and so allows the designer to change the HDL code if the required functionality is not met with in a short time period. Since the design is not yet synthesized to gate level, timing and resource usage properties are still unknown.

B. Synthesis Result

The developed MAC design is simulated and verified their functionality. Once the functional verification is done, the RTL model is taken to the synthesis process using the Xilinx ISE tool. In synthesis process, the RTL model will be converted to the gate level netlist mapped to a specific technology library. This MAC design can be synthesized on the family of Spartan 3E.

Here in this Spartan 3E family, many different devices were available in the Xilinx ISE tool. In order to synthesis this design the device named as “XC3S500E” has been chosen and the package as “FG320” with the device speed such as “-4”. The design of MAC is synthesized and its results were analyzed as follows.

IV. Device utilization summary:

This device utilization includes the following.

- Logic Utilization
- Logic Distribution
- Total Gate count for the Design

The device utilization summery is shown above in which its gives the details of number of devices used from the available devices and also represented in %. Hence as the

result of the synthesis process, the device utilization in the used device and package is shown below.

Device utilisation summary:

Selected Device : 3a200ft256-4

Number of Slices:	689 out of 1920	35%
Number of Slice Flip Flops:	64 out of 3840	1%
Number of 4 input LUTs:	1285 out of 3840	33%
Number of IOs:	69	
Number of bonded IOBs:	69 out of 173	39%
Number of GCLBs:	1 out of 8	12%

Fig. 9 Device summary

A. RTL Schematic:

The RTL (Register Transfer Logic) can be viewed as black box after synthesise of design is made. It shows the inputs and outputs of the system. By double-clicking on the diagram we can see gates, flip-flops and MUX.

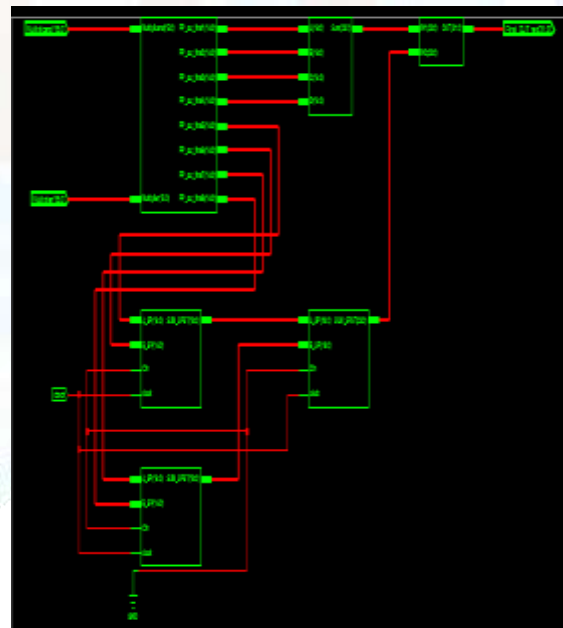


Fig. 10 Schematic of Booth Encoder with SPST Adder

B. Delay factor:


```

-----
Total                45.376ns (22.795ns logic, 22.581ns route)
                    (50.2% logic, 49.8% route)
-----
Timing constraint: Default OFFSET OUT AFTER for Clock 'clock'
Total number of paths / destination ports: 32 / 32
-----
Offset:              7.165ns (Levels of Logic = 1)
Source:              VMFU_OUT_31 (FF)
Destination:        VMFU_OUT<31> (PAD)
Source Clock:       clock rising

Data Path: VMFU_OUT_31 to VMFU_OUT<31>
-----
Cell:in->out      fanout  Delay  Delay  Logical Name (Net Name)
-----
FDRS:C->Q         1  0.720  0.801  VMFU_OUT_31 (VMFU_OUT_31)
OBUF:I->O         5.644                VMFU_OUT_31_OBUF (VMFU_OUT<31>)
-----
Total                7.165ns (6.364ns logic, 0.801ns route)
                    (88.8% logic, 11.2% route)
-----
CPU : 38.58 / 38.85 s | Elapsed : 39.00 / 39.00 s
-->
Total memory usage is 174528 kilobytes

```

```

-----
*                               Final Report                               *
-----
Final Results
RTL Top Level Output File Name : TOP_SPST.ngr
Top Level Output File Name    : TOP_SPST
Output Format                  : NCG
Optimization Goal              : Speed
Keep Hierarchy                 : NO

Design Statistics
# IOs                          : 69

Cell Usage :
# BELS                          : 2675
# GND                           : 1
# INV                           : 120
# LUT1                          : 9
# LUT2                          : 314
# LUT3                          : 455
# LUT4                          : 387
# MULT_AND                      : 25
# MUXCY                          : 618
# MUXF5                          : 123
# MUXF6                          : 17
# VCC                           : 1
# XORCY                          : 605
# FlipFlops/Latches             : 64
# FDR                            : 17
# FDRB                          : 32
# FDRS                          : 15
# Clock Buffers                 : 1
# BUFGP                          : 1
# IO Buffers                    : 68
# IBUF                          : 36
# OBUF                          : 32
-----

```

- The simulation results are discussed by considering different cases.
- The RTL model is synthesized using the Xilinx tool in Spartan 3E and their synthesis results were discussed with the help of generated reports.

V.CONCLUSION

An electronics glitch is an undesired transition that occurs before the signal settles to its intended value. In other words, glitch is an electrical pulse of short duration that is usually the result of a fault or design error, particularly in a digital circuit. For example, many electronic components, such as [flip-flops](#), are triggered by a pulse that must not be shorter than a specified minimum duration; otherwise, the component may malfunction. A pulse shorter than the specified minimum is called a glitch.

This work presents a versatile multimedia functional unit is designed with low-power technique, 16x16 multiplier-accumulator (MAC), with addition, subtraction, sum of absolute difference, interpolation. A Radix-2 Modified Booth multiplier circuit is used for MAC architecture. Compared to other circuits, the Booth multiplier has the highest operational speed and less hardware count. Power and delay is calculated for the blocks. MAC unit is designed with enable to reduce the total power consumption based on block enable technique. Using this block, the N-bit MAC unit is constructed and the total power consumption is calculated for the MAC unit.

The presented low-power technique called glitch diminishing power suppression technique and explores its applications in multimedia/DSP computations, where the theoretical analysis and the realization issues of the technique are fully discussed. The proposed technique can obviously decrease the switching (or dynamic) power dissipation,

which comprises a significant portion of the whole power dissipation in integrated circuits.

ACKNOWLEDGMENT

The author would like to thank Mrs.Y. RAJYALAKSHMI, Asst. Professor, Electronics & Communication Engineering (Specialization in VLSI Design), Sri Sivani College Of Engineering, JNTUK, for her continuous support and encouragement for this work. The authors would also like to acknowledge the support provided by the technical staff of Elect. & Comm. Engg., Sri Sivani College Of Engineering, JNTUK providing him with the ample amenities, ways & means through which he was capable to inclusive this task.



Divya Sathi Balaga – Pursuing M.Tech in VLSI in SSCE.



Y. Rajya Lakshmi – Asst. prof. in SSCE,

1. An international conference on “a novel method for watermarking using opaque & translucent methods in videos” by ICECE VIZAG, ASTAR, 17th June 2012.
2. An international conference on “the effect of high power amplifier non linearity on MC-CDMA systems using spreading sequences”, by ICCIT2012 with ISBN 978-93-82338-02-4.

[1] Swapna Enugala and Asha Bai. J, “A Spurious Power Suppression Technique For A Low Power Multiplier,” IJERT, ISSN: 2278-0181, Vol. 2, Issue 1, January – 2013.

[2] S.Surabhi and M.Jagadeeswari, “A Robust Power Downgrading Technique using Sparse Modulo $2 + 1$ Adder,” IJARCCCE ISSN (Print) : 2319-5940 ISSN (Online) : 2278-1021, Vol. 2, Issue 3, March – 2013.

[3] Julien Lamoureux, Guy G. F. Lemieux, and Steven J. E. Wilton, “GlitchLess: Dynamic Power Minimization in FPGAs Through Edge Alignment and Glitch Filtering,” IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 16, NO. 11, NOVEMBER 2008.

[4] M.PRAVEEN KUMAR and Prof. K ASHOK BABU, “A SPURIOUS-POWER SUPPRESSION TECHNIQUE FOR MULTIMEDIA/DSP APPLICATIONS,” IJAEST, ISSN: 2230-7818, Vol No. 11, Issue No. 1, 035 – 051.

[5] Kuan-Hung Chen, Kuo-Chuan Chao, Jinn-Shyan Wang, Yuan-Sun Chu and Jiun-In Guo, “An Efficient Spurious Power Suppression Technique (SPST) and its Applications on RAMPEG-4 AVC/H.264 Transform Coding Design,”

[6] K. H. Chen, K. C. Chao, J. I. Guo, J. S. Wang, and Y. S. Chu, “Design exploration of a spurious power suppression technique (SPST) and its applications,” in Proc. IEEE Asian Solid-State Circuits Conf., Hsinchu, Taiwan, Nov. 2005, pp. 341–344.

[7] A. Bellaouar and M. I. Elmasry, “Low-Power Digital VLSI Design” Circuits and Systems. Norwell, MA: Kluwer, 1995.

[8] A. P. Chandrakasan and R. W. Brodersen, “Minimizing power consumption in digital CMOS circuits,” Proc. IEEE, vol. 83, no. 4, pp. 498–523, Apr. 1995.

[9] K. K. Parhi, “Approaches to low-power implementations of DSP systems,” IEEE Trans. Circuits Syst. I, Fundam. Theory Appl., vol. 48, no. 10, pp. 1214–1224, Oct. 2001.

REFERENCES: