International Journal of Electronics and
Communication Engineering (IJECE)
ISSN 2278-9901
Vol. 2, Issue 2, May 2013, 27-32
© IASET

International Academy of Science,
Engineering and Technology
Connecting Researchers; Nurturing Innovations
IASET

DESIGN OF A HIGH PERFORMANCE SERIAL SERIAL MULTIPLIER

NAJINI NOORBHASHA[1] & P. RAMAKRISHNA[2]

[1]M.Tech Student, CVSR College of Engineering, Hyderabad, India

[2]Associate Professor, ECE Department, CVSR College of Engineering, Hyderabad, India

ABSTRACT

In the past few years, multipliers have been a tremendous increase in the demand for better quality electronic equipment because of increasing standards of living of people. This has led to the development of new technologies like System On Chip (S.O.C)s and scaling down the transistor sizes so that we can integrate a greater amount of functionality into a smaller area of silicon.

The densely packed components will result in higher power consumption per unit area of silicon and higher power dissipation too. This could result in increasing the temperature of the chip which could slow down the chip or could damage it. The higher power consumption also leads to quick draining of battery operated devices. In many palm-hold devices like smart phones, i-pods etc there are many applications like voice interface, high definition audio and video, video calling, 3G and 4G services, etc. which needs a very good quality DSP circuit.FIR Filters are a major component of DSP circuits. FIR Filters mainly comprise of multipliers and adders as the power consuming elements. For a good quality DSP circuit we need to implement an FIR Filter with wide multipliers that give good precision and range for good multiplication. And also some times to reduce the wiring cost, it is common practice to transmit the data through a high speed serial link. In some ICs the designers try to reduce the number of IO pads in order to reduce the power consumption and silicon area. Therefore efforts are made to design high speed serial interface in order to facilitate on-chip buffering and parallel processing. In this project we design seral-serial multiplier to reduce the complexity of SOC's, and power dissipation. The RTL coding for this project has been done in verilog HDL. Modelsim Simulator has been used to perform functional verification through simulation. XILINX ISE is used to perform synthesis and power analysis.

KEYWORDS: Binary Multiplication, Parallel Multipliers, Serial Multipliers, On-Chip Serial Link Bus Architecture

INTRODUCTION

Multipliers are the fundamental and essential building blocks of VLSI systems. The design and implementation approaches of multipliers contribute substantially to the area, speed and power consumption of computational intensive VLSI systems. Often the delay of multipliers dominates the critical path of this System and due to issues reliability and portability. Power consumption is a critical criterion for applications that demand low-power as its primary metric. While low power and high speed multiplier circuits are highly demanded, it is not always possible to achieve both criteria simultaneously. Therefore, a good multiplier design requires some tradeoffs between speed and power consumption. Bit-serial multipliers provide the lowest possible area complexity.

Parallel multipliers are multiplier which has parallel operands i.e., the entire operand bits is available at a time. Whereas, in serial multipliers the two input operands are given serially i.e., here only single bit is received at the multiplier at a time from each operand. Typically, multiplier in hardware is implemented in three stages. They are partial product generation (PPs), reduction of partial product terms and the final carry propagation adder. The partial products can be generated either in parallel or serially depending on the target application and availability of i/p data. The partial products are generally reduced by CSAs using an array or a tree structure. Carry propagation addition is inevitable when the partial products are reduced to two rows. The addition process is carried out by RCA for low power or CLA adder for high speed. The height of the partial product tree increases with the wordlenth of i/p data. Therefore the area, delay and power dissipation increases. Therefore it is highly desirable to reduce the no of partial products before the CSA stage [3]. This can be achieved by modified BOOTH algorithm [6] to reduce the height of pp matrix. But it leads more area and delay compared to the simple pp generation process. Another approach is to use high order column compressors instead of FAs. But it is slower and consumes more power than the FAs.

And also some times to reduce the wiring cost, it is common practice to transmit the data through a high speed serial link [7], [8]. In some ICs the designers try to reduce the number of IO pads in order to reduce the power consumption and silicon area. Therefore efforts are made to design high speed serial interface in order to facilitate on-chip buffering and parallel processing. Parallel multipliers are popular for their high speed operation but if the word lengths go higher they are often constrained by the hardware cost and power consumption of the applications.

In general applications like dsp, image processing, encryption and decryption techniques the word lengths are of hundreds of bits. Therefore, low-cost serial multipliers are widely used. Serial multipliers are also found in applications like system-on-chip (SoC) design. In present technology more intellectual property cores are integrated on the Soc, which results in large interconnect area and high power consumption. And also due to more interconnections the routing between one module to another module will be very difficult and also congested. To overcome this problem we are evolved recently to on-chip bus and alternative on-chip serial-link bus structure [9].

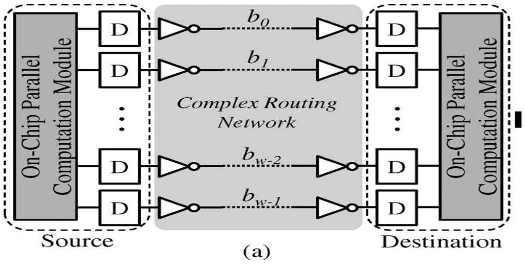

Figure 1: Conventional Bus Structure

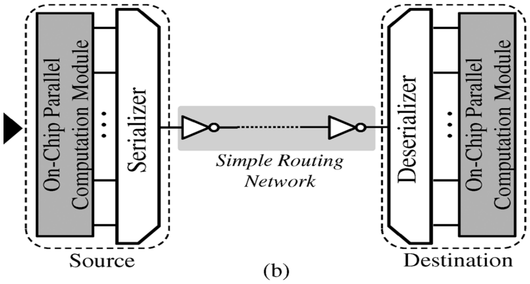
On-Chip Parallel Computation Module
Serializer
Simple Routing Network
Deserializer
On-Chip Parallel Computation Module
Source
(b)
Destination

Figure 2: Serial-Link Bus Structure

In the above figures (1) and (2) shows the difference between a conventional parallel on-chip bus and an on-chip serial bus, respectively. In figure (2) serializer is used to convert the parallel data stream to a serial bit stream in order to pass through simply routed serial link. At the destination end the deserializer is used to convert the data stream from a serial link again back to parallel data. So to make computations with normal speed we have to transfer the data through serial link at high speed in order that it should not wait for any data. We can do this by performing some partial computations on the incoming data stream at high speed while data is being buffered.

The rest of the paper is organized as follows section II revises the existing serial-serial multipliers, section III deals with the proposed serial-serial multiplier, section IV gives implementation results and section V will concludes the paper followed by reference.

EXISTING SERIAL MULTIPLIERS

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 [1]-[2] 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.

PROPOSED SERILA MULTIPLIER

Here in this section we will propose a technique for the generation of individual rows of partial products by considering two serial input operands. In which one operand is starting from LSB and another starts from MSB. This multiplier will generate all partial product rows and their accumulation will be done in only 2n cycles for an n×n multiplication.

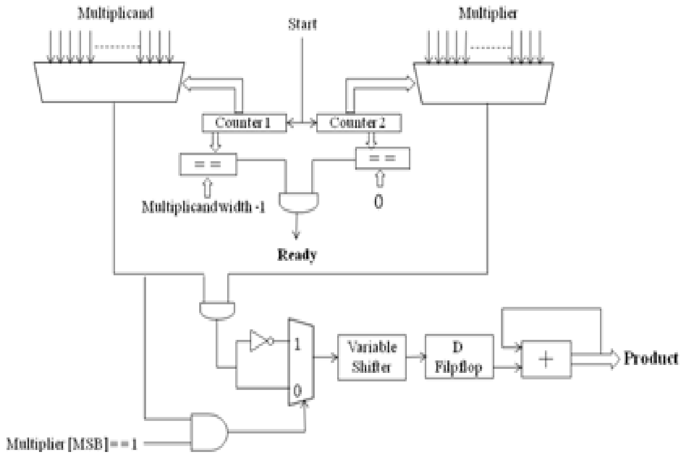

Figure 3: Block Diagram of Serial Multiplier

Serial multiplier uses a single bit multiplication i.e. an and gate to perform multiple bit multiplication. Each bit in the multiplier is multiplied with each bit of the multiplicand in a systematic fashion and the result is accumulated accordingly. Since this multiplier is a signed multiplier the result of gate is complemented in the event of MSB bit of the multiplier being 1. Once the multiplication of all the bits in the multiplicand are multiplied with one bit of the multiplicand the accumulated result is shifted before adding it to other partial products. Figure 3 shows the block level diagram of the serial multiplier.

Serial Multiplier is a multiplier where in its computes the end result by multiplying each bit of the multiplier with each bit of the multiplicand in a serial order i.e only one bit at a time. The intermediate products are accumulated and shifted for addition with the next partial product.

For example, if A_2 A_1 A_0 and B_2 B_1 B_0 are two signed numbers, The sign of the output depends on the sign bits B_2 and A_2. If both bits are either 1 or both bits are 0, the output is positive or negative. Therefore, only one multiplier, one shifter and one adder is required in Serial Multiplier. It can be concluded that the hardware of serial multiplier is greatly reduced and the power consumption as well.

IMPLEMENTATION RESULTS

The whole multiplier is implemented using verilog in Xilinx ISE 13.4 and it is targeting Xilinx Virtex-3 xc5vlx20t-2ff323.A testbench is used to generate the stimulus and applies it to the implemented serial-serial multiplier. The serial-serial multiplier code was also checked using Xilinx [10]. The design was synthesized using Xilinx synthesis XST tool [10] . Post synthesis and place and route simulations were made to ensure the design functionality after synthesis and place and route. Table 1 shows the resources utilized by it while serial-serial multiplier implemented.

Table 1: Results Obtained Proposed Serial-Serial Multiplier

NO OF SLICES	35
DYNAMIC POWER (mw)	3.9
CLOCK DELAY (ns)	6.53
SPEED (Mhz)	165.14

Synthesis Results

The Figure 4. shows the top level module of a 4 bit serial multiplier. The '**clk**' signal is used to time all the operations while the '**rst**' signal is used to initialize the outputs and the internal registers. The '**data_in**' is used to feed the input signal values and the output of the filter can be observed on '**data_out**'. The '**start**' is used to start the operation. The Figure 5.shows the RTL schematic view of serial multiplier.

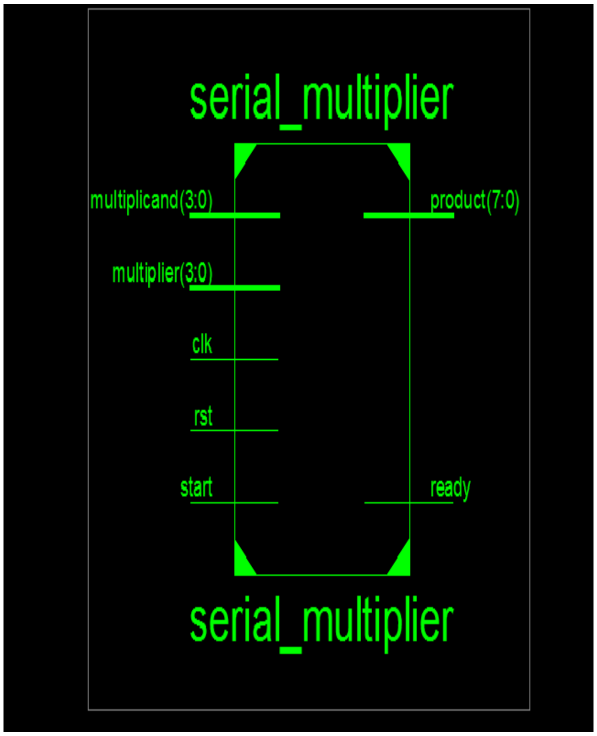

Figure 4: Top Level Module of a 4 Bit Serial Multiplier

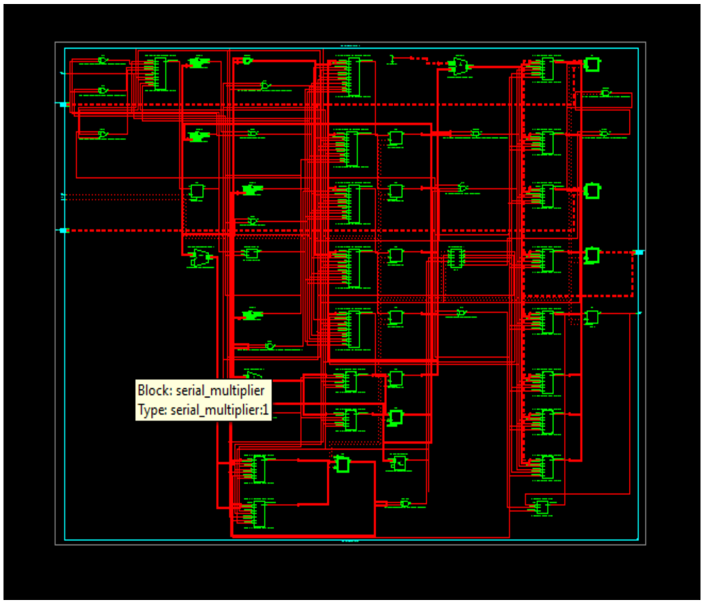

Figure 5: RTL Schematic View of Serial Multiplier

Simulation Results

The Figure 6, shows the working of a 4 bit serial multiplier with different combinations of multiplier and multiplicand. We see that the output is 'ready' after 2n cycles after the 'start' signal goes high. This shows the latency in this multiplier.

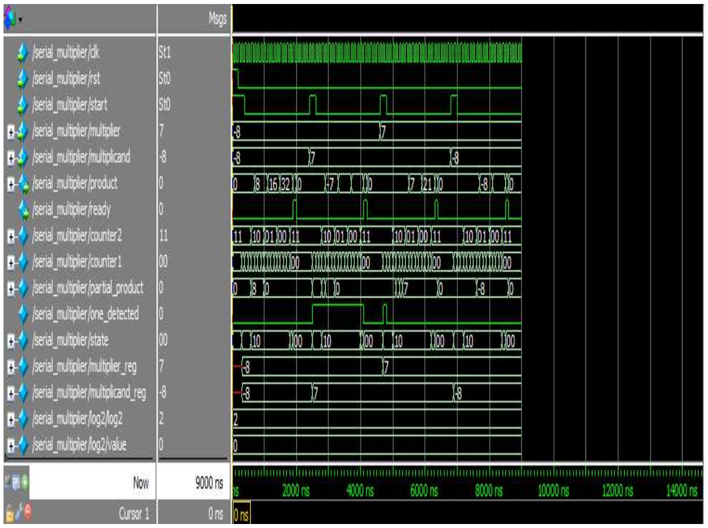
Msgs
/serial_multiplier/clk St1
/serial_multiplier/rst St0
/serial_multiplier/start St0
/serial_multiplier/multiplier 7
/serial_multiplier/multiplicand -8
/serial_multiplier/product 0
/serial_multiplier/ready 0
/serial_multiplier/counter2 11
/serial_multiplier/counter1 00
/serial_multiplier/partial_product 0
/serial_multiplier/one_detected 0
/serial_multiplier/state 00
/serial_multiplier/multiplier_reg 7
/serial_multiplier/multiplicand_reg -8
/serial_multiplier/log2/log2 2
/serial_multiplier/log2/value 0
Now 9000 ns
Cursor 1 0 ns
2000 ns
4000 ns
6000 ns
8000 ns
10000 ns
12000 ns
14000 ns
0 ns

Figure 6: Simulation Results of Serial Multiplier

CONCLUSIONS

In this paper, a different approach of computing serial-serial multiplier is proposed. By using this multiplier we can easily generate all partial rows in just 2n cycles for n×n multiplication. This proposed serial multiplier requires only one multiplier, one shifter and one adder . It can be concluded that the hardware of serial multiplier is greatly reduced and the power consumption as well. This serial multiplier approach has advantage of low I/O requirement and hence is most suitable for complex Soc and advance FPGAs.

REFERENCES

1. Manas Ranjan Meher, Ching Chuen Jong, and Chip-Hong Chang,A High Bit Rate Serial-Serial Multiplier With On-the-Fly Accumulation by Asynchronous,DigitalObjectIdentifier10.1109/TVLSI.2010.2060374.
2. M. R. Meher, C. C. Jong, and C. H. Chang, "High-speed and lowpowerserial accumulator for serial/parallel multiplier," in Proc. IEEE Asia-Pacific Conf. Circuits Syst. (APCCAS), Macau, China, 2008, pp. 176–179.
3. R.Menon "High performance 5:2 compressor architecture " IEE proc-Circuits Devices Syst., Vol. 153, no.5, pp. 447-452, oct. 2006.
4. Ron S. Waters, Earl E.Swartzlander "A Reduced Complexity Wallace multiplier reduction" in IEEE transactions on Computers, Aug.
5. Whytney J. Townsend, Earl E. Swartz, "A Comparison of Dadda and Wallace multiplier delays". Computer Engineering Research Center, The University of Texas.
6. A. D. Booth, "A signed binary multiplication technique," Quarterly J. Mechan. Appl. Math., vol. 4, no. 2, pp. 236–240, Aug. 1951.
7. M. Ghoneima, Y. Ismail, M. Khellah, J. Tschanz, and V. De, "Seriallink bus: A low-power on-chip bus architecture," IEEE Trans. Circuit.Syst. I, Reg. Papers, vol. 56, no. 9, pp. 2020–2032, Sep. 2009.
8. R. Dobkin, M. Moyal, A. Kolodny, and R. Ginosar, "Asynchronous current mode serial communication," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 18, no. 7, pp. 1107–1117, Jul. 2010.
9. M.Burzio and P,.pellegriono," seriailizer-paralleliing circuit for high speed digital signals",U.S Patent,Aug.4,1998.
10. "Xilinx13.4, Synthesis and Simulation Design Guide", UG626 (v13.4) January 19, 2012.