The Design of High Speed FIR Filter using Improved DA Algorithm and it's FPGA Implementation

Magatha Nayak Bhukya[1], K. Anjaiah[2], G. Sravya[3], P. Nagaraju[4]

[1]M.Tech(VLSI), NOVA College of Engineering & Technology, Jangareddygudem, A.P., India.

[2]M.Tech, VLSI, NOVA College of Engineering & Technology, Jangareddygudem, A.P., India.

[3]M.Tech, Associate Professor, NOVA College of Engineering & Technology, Jangareddygudem, A.P., India.

[4]M.Tech, Associate Professor & HOD of ECE, NOVA College of Engineering & Technology, Jangareddygudem, A.P., India.

***Abstract* - when the DA (distributed arithmetic) algorithm is directly applied in FPGA (field programmable gate array) to realize FIR (finite impulse response) filter, it is difficult to achieve the best configuration in the coefficient of FIR filter, the storage resource and the computing speed. According to this problem, the paper provides the detailed analysis and discussion in the algorithm, the memory size and the look-up table speed. Also, the corresponding optimization and improvement measures are discussed and the concrete Hardware realization of the circuit is presented. The design based on Altera EP2C5T144C8 chips is synthesized under the integrated environment of QUARTUS II 7.1. The results of Simulation and test show that this method greatly reduces the FPGA hardware resource and the high speed filtering is achieved. The design has a big breakthrough compared to the traditional FPGA realization.**

***Keywords*- FIR filter; DA algorithm; FPGA.**

I. INTRODUCTION

Micro Electro Mechanical Systems (MEMS) is the process of producing and combining miniaturized mechanical elements, sensors, actuators and electronics. MEMS make technology possible by developing smart products, increasing the computational ability of micro electronics with the quality and control capabilities of micro sensors and micro actuators and increase space of importance in design and application. MEMS are called "Micro machines" in Japan and "Microsystems Technology" in Europe. In the Inertial Navigation system the Accelerometer has shown in Figure 1.1 measures acceleration of a moving object and also detects the tilt. Accelerometer [1] is a device that detects acceleration and tilt. Accelerometers detect impact and deploy automobile airbags as well as retract the hard disk's read/write heads when a laptop is dropped.

The advantages of the FPGA approach to digital filter implementation include higher sampling rates than are available from traditional DSP chips, lower costs than an ASIC for moderate volume applications, and more flexibility than the alternate approaches. In literature, several multiplier-less schemes had been proposed. These methods can be classified in two categories according to how they manipulate the filter coefficients for the multiply operation. The first type of multiplier-less technique is the conversion-based approach, in which the coefficients are transformed to other numeric representations whose hardware implementation or manipulation is more efficient than the traditional binary representation. Example of such techniques is the Canonic Sign Digit (CSD) method, in which coefficients are represented by a combination of powers of two in such a way that multiplication can be simply implemented with adder/subtractors and shifters [2], and the Dempster-Mcleod method, which similarly involves the representation of filter coefficients with powers of two but in this case arranging partial results in cascade to introduce further savings in the usage of adders [3].

The second type of multiplier-less method involves use of memories (RAMs, ROMs) or Look-Up Tables (LUTs) to store pre-computed values of coefficient operations. These memory-based methods involve Constant Coefficient Multiplier method and the very-well known Distributed Arithmetic method [4] as examples. Distributed Arithmetic (DA) algorithm appeared as a very efficient solution especially suited for LUT-based FPGA architectures. Croisier et al [5] had proposed the multiplierless architecture of DA algorithm and it is based on an efficient partition of the function in partial terms

using 2's complement binary representation of data. The partial terms can be pre-computed and stored in LUTs. Yoo et al. [6] observed that the requirement of memory/LUT capacity increases exponentially with the order of the filter, given that DA implementations need 2K – words, *K* being the number of taps of the filter.

II. METHODOLOGY

The principle of DA algorithm is as follows. The output of linear time-invariant system is shown as Eq. (1).

$$Y = \sum_{m=1}^{M} A_m X_m \quad \text{---------- (1)}$$

Where A_m is a fixed factor, X_m is the input data (X_m <1). X_m can be expressed as Eq. (2) using the binary complement.

$$X_m = -x_{mo} + \sum_{n=1}^{N-1} x_{mn} 2^{-n} \quad \text{---------- (2)}$$

Where x_{mn} **is 0 or 1,** x_{m0} is sign bit, $x_{m,N-1}$ is the least significant bit. Then Y can be expressed as Eq. (3).

$$\mathbf{Y = \sum_{n=1}^{M-1} \sum_{m=1}^{M} A_m x_{mn} 2^{-n} + \sum_{m=1}^{M} A_m (-x_{mo}) \text{ ----- (3)}}$$

In Eq. (3), as the value of x_{mn} is 0 or 1, there are 2^M kinds of different results of

$$\sum_{m=1}^{M} A_m x_{mn}$$

If we construct a LUT which can store al l the possible combination of values [7], we can calculate the value of 2^M in advance and store them in the LUT. Using x_{mn} as the LUT address signal, the shifting (2^{-1} operation) and adding operation are carried out on the output of the LUT.

Then $\sum_{m=1}^{M} A_m x_{mn}$ can be realized through **N-1** cycles and the result of Multiplication accumulation can be achieved directly. So the complicated multiplication accumulation operation is converted to the shifting and adding operation. The parallel computing is adopted to improve the speed of calculation. The complicated multiplication - accumulation operation is converted to the shifting and adding operation when the DA algorithm is directly applied to realize linear time invariant system. However, the scale of the LUT will Increase exponentially with the coefficient. If the coefficient is small, it is very convenient to realize through the rich structure of FPGA LUT; while the coefficient is large, it will take up a lot of storage resources of FPGA and reduce the calculation speed. Meanwhile, the **N-1** cycles also result in the too long LUT time and the low computing speed. The paper presents the improvement and optimization of the DA algorithm aiming at the problems of the configuration in the coefficient of FIR filter, the storage resource and the calculating speed, which make The memory size smaller and the operation speed faster to improve the Computational performance.

Improved design of the DA Algorithm from eq.(2),**Xm** can be expressed as Eq.(4).

$$\mathbf{X_m = 1/2[X_m - (-X_m)]} \quad \text{---------- (4)}$$

Where the **–Xm** can be expressed as Eq.(5) according to the binary complement operation.

$$\mathbf{-X_m = -x_{mo} + \sum_{n=1}^{N-1} x_{mn} 2^{-n} + 2^{-(N-1)}} \quad \text{----------(5)}$$

Put eq. (2) into Eq. (4), eq (6) can be achieved.

$$\mathbf{X_m = 1/2[-(x_{mo} - x_{mo}) + \sum_{n=1}^{N-1} (\bar{x}_{mn} - x_{mn}) 2^{-n} - 2^{-(N-1)}]} \quad \text{----(6)}$$

For convenience two variables are define as follows

$$\Phi_{mo} = -(x_{mo} - x_{mo}) \qquad \Phi_{mn} = (x_{mn} - x_{mn})$$

In which, as the value of x_{mn} is 0 or 1, so the value of ϕ_{mn} and $m0\ \phi$ is ±1. Then Eq. (6) can be expressed as Eq. (7).

$$\mathbf{Xm = 1/2 [\sum_{n=1}^{N-1} \Phi_{mn} 2^{-n} - 2^{-(N-1)}]} \quad \text{---------- (7)}$$

Put Eq. (7) into Eq. (1), Eq. (8) can be achieved.

$$\mathbf{Y = 1/2 \sum_{K=1}^{K} A_k \sum_{n=1}^{N-1} \Phi_{mn} 2^{-n} - 2^{-(N-1)}}$$

$$\mathbf{Y = 1/2 [\sum_{n=1}^{N-1} \sum_{m=1}^{M} A_m \Phi_{mn} 2^{-n} - \sum_{m=1}^{M} A_m 2^{-(N-1)}]} \quad \text{---------- (8)}$$

III. THE CIRCUIT DESIGN OF FIR FILTER

A. The design index and parameters extraction:

A 16th-order FIR filter is designed. Its Parameters are as follows: the sampling frequency is 2.25 MHz; the pass band cutoff frequency is 100 kHz; the width of the input data, the output data and the filter coefficient is 8, 16 and 20 bits respectively. It adopts Hamming window to design and Mat Lab simulation to calculate its unit-sampling response h(k) and amplify it 216 times. The h (k) is as follows.

h(0)=h(15)=298D h(1)=h(14)=578D

h(2)=h(13)=1364D h(3)=h(12)=2718D

h(4)=h(11)=4503D h(5)=h(10)=6400D

h(6)=h(09)=7996D h(7)=h(8)=8908D

B. The hardware circuit unit: The hardware circuit is shown in Fig. 2.

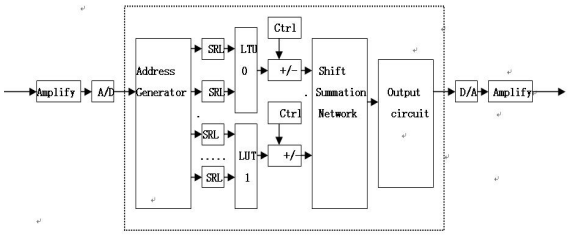


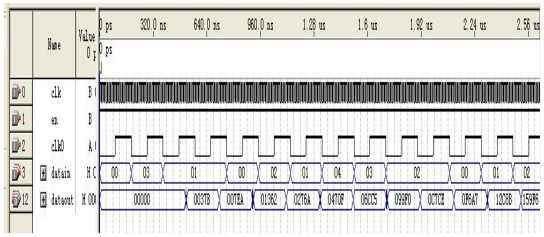
Fig. 2.

The circuit structure of FIR system when using the DA algorithm to implement the linear time-invariant system, the algorithm is optimized according the method of section 2.

The pre storing value corresponding to the upper half of the memory address of LUT storage will be the negative of the lower half and then the LUT reduces by half using symmetry. The address maker circuit generates the LUT address. The upper half of the address looks up its corresponding pre storing value. Meanwhile, the address is used as Ctrl control-adding-decrease implements to complete the positive and negative conversion between the pre storing value corresponding to the upper and lower half of it. According to result of the improvement and optimization, the LUT is divided into two 4-input LUTs and the address maker circuit divides the input signals into four segments in accordance with the 4-input LUT. The speed of signal sampling under the control of the FPGA can be adjusted. The data buffer can be established according to the order of the filter. As the designed filter is a 16th-order one, so the sampled serial data can be sent to the 20 bits serial-in parallel-out shift register, and then the data is divided and sent to the LUT in turn. As the coefficient is amplified 216times, the obtained result is reduced by the output circuit accordingly.

C. Circuit simulation and test

The design based on Althea's EP2C5T144C8 chips, the series of Cyclone□, is synthesized under the integrated environment of QUARTUS □ 7.1. The input sequence is x (n) = [0, 3, 1, 1, 0, 2, 1, 4, 3, 2, 2, 0, 1, 2, 2, 3] and the simulation waveform is shown in Fig. 3. The filter input/output in the waveform uses hexadecimal representation. Compared to Mat Lab simulation, the error of hardware simulation is □ ±1%.The designed results are consistent with what we desired. The circuit structure of FIR filter works well.



The implementation of filter based on FPGA is realized on EP2C5T144C8 chips by using of IP core, the DA algorithm and the improved DA algorithm separately. The results of compilation and test show that the needed LE is1522, 1269 and 776 respectively when IP core, the DA algorithm and the improved DA algorithm is used to implement filter. The improved algorithm can greatly reduce the hardware resource and improve the throughput efficiency. It meets the design requirements entirely.

IV. CONCLUSION

The complicated multiplication-accumulation operation is converted to the shifting and adding operation when the DA algorithm is directly applied to realize FIR filter .Aiming at the problems of the best configuration in the coefficient of FIR filter, the storage resource and the calculating speed, the DA algorithm is optimized and improved in the algorithm structure, the memory size and the look-up table speed.

The implementation of highly efficient serial and parallel DA algorithm was presented in this work. The results were analyzed for 3-tap and 16-tap FIR filter using partitioned input based LUT on Xilinx 10.1i as a target of SPARTAN-3E FPGA device .The speed performance of the Parallel DA FIR Filter was superior in comparison to all other techniques . For small tap filter less area, high speed and low power consumption is achieved after applying the Serial and Parallel DA technique. In large-tap FIR filter, speed of parallel DA FIR design technique become 3 times faster than that of conventional FIR filter. The proposed algorithm for FIR filters is also area efficient since approximately 50% of the area is saved with this technique as compared to conventional FIR filter design. Area efficiency and high speed is achieved with parallel DA technique at very slight cost of power consumption for large tap FIR filter. Since, distributed arithmetic FIR filters are area efficient and contained less delay, so these filters can be used in various applications such as pulse shaping FIR filter in WCDMA system, software design radio and signal processing system for high speed. In future the work to reduce the power consumption in large tap DA FIR filters could be performed.

REFERENCES

[1] Stanley A. White "Applications of Distributed arithmetic to digital signal processing tutorial review", IEEE ASSP Magazine 1989.

[2] M. Yamada, and A. Nishihara, "High-Speed FIR Digital Filter with CSD Coefficients Implemented on FPGA", in Proceedings of IEEE Design Automation Conference, 2001, pp. 7-8.

[3] M.A. Soderstrand, L.G. Johnson, H. Arichanthiran, M. Hoque, and R. Elangovan, "Reducing Hardware Requirement in FIR Filter Design", in Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing 2000, Vol. 6, pp. 3275 – 3278.

[4] Martinez-Peiro, J. Valls, T. Sansaloni, A.P. Pascual, and E.I. Boemo, "A Comparison between Lattice, Cascade and Direct Form FIR Filter Structures by using a FPGA Bit-Serial DA Implementation", in Proceedings of IEEE International Conference on Electronics, Circuits and Systems, 1999, Vol. 1,pp. 241 – 244.

[5] A. Croisier, D. J. Esteban, M. E. Levilion, and V. Rizo, "Digital Filter for PCM Encoded Signals", U.S. Patent No. 3,777,130, issued April, 1973.

[6] H. Yoo, and D. Anderson, "Hardware-Efficient Distributed Arithmetic Architecture for High-Order Digital Filters", in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005, Vol. 5, pp. 125 – 128.

[7] H. Chen, C. H. Xiong, S. N. Zhong, "FPGA-based efficient programmable poly phase FIR filter," Journal of Beijing institute of Technology, 2005, vol. 14, pp. 4-8.