FPGA implementation of digital filters synthesized using the frequency-response masking technique

Zheng, H. Q.; Lim, Yong Ching; Yu, Ya Jun; Foo, Say Wei

2001

Lim, Y. C., Yu, Y. J., Zheng, H.Q., & Foo, S.W. (2001). FPGA implementation of digital filters synthesized using the frequency-response masking technique. IEEE International Symposium on Circuits and Systems 2001: (pp. 173-176). Singapore: National University of Singapore.

<https://hdl.handle.net/10356/90768>

© 2001 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE. This material is presented to ensure timely dissemination of scholarly and technical work. Copyright and all rights therein are retained by authors or by other copyright holders. All persons copying this information are expected to adhere to the terms and constraints invoked by each author's copyright. In most cases, these works may not be reposted without the explicit permission of the copyright holder. <http://www.ieee.org/portal/site> This material is presented to ensure timely dissemination of scholarly and technical work. Copyright and all rights therein are retained by authors or by other copyright holders. All persons copying this information are expected to adhere to the terms and constraints invoked by each author's copyright. In most cases, these works may not be reposted without the explicit permission of the copyright holder.

FPGA IMPLEMENTATION OF DIGITAL FILTERS SYNTHESIZED USING THE FREQUENCY-RESPONSE MASKING TECHNIQUE

Y. C. Lim¹, Y. J. Yu¹, H.Q. Zheng² and S.W. Foo¹

¹Department of Electrical and Computer Engineering
National University of Singapore, Singapore 119260

² School of Engineering, Temasek Polytechnic
Singapore 529757

ABSTRACT

The effective length of a filter designed using the frequency-response masking technique is very high and requires a very large number of delay elements. In this paper, we present some useful techniques for reducing the data transfer between the FPGA and external memory when the random logic are implemented using FPGA and the delay elements are implemented using external memory such as DRAM.

1. INTRODUCTION

The frequency-response masking technique [1-14] uses a system of sub-filters as shown in Fig.1 to synthesize a very sharp filter. The sub-filter determining the transition band of the filter (called the band-edge shaping filter) is obtained by replacing each delay element of a prototype filter by M delay elements and so has very long filter length although it has very low arithmetic complexity. When an FPGA is used in the implementation, in order to reduce hardware cost, only the arithmetic unit, random logic functions, and a limited amount of storage locations are implemented on the FPGA. The tap delay line of the band-edge shaping filter and those of the masking filters are usually implemented in low cost external memory such as DRAM. The coefficient values of the sub-filters are stored on external ROM. Accessing external memory is usually very much slower than internal on chip data transfer within the FPGA. Thus, fetching data from external memory contribute a significant overhead to the operating speed of the filter. In order to improve the operating speed of the filter, it is obviously advantageous to reduce the amount of data transfer between external memory and the FPGA.

Fig.1 A filter synthesized using the frequency-response masking technique.

2. EXTERNAL MEMORY ACCESS

Let the lengths of $H_a(z)$, $H_{Ma}(z)$, and $H_{Mc}(z)$ be N_a , N_{Ma} and N_{Mc} , respectively. Thus, in every sampling interval, $N_a + N_{Ma} + N_{Mc}$ samples of signal data must be fetched from memory. Taking the symmetry of the coefficients into consideration there are $(N_a + N_{Ma} + N_{Mc})/2$ distinct coefficients. If two address pointers (one points to each of the two data to be multiplied by the same coefficient) for each sub-filter are maintained inside the FPGA, then there will be $(N_a + N_{Ma} + N_{Mc})/2$ ROM fetches per sampling interval to fetch the coefficients; otherwise, there will be $(N_a + N_{Ma} + N_{Mc})$ ROM fetches per sampling interval.

3. REDUCING MEMORY FETCH FOR $H_{Ma}(z)$ AND $H_{Mc}(z)$

Let $H_{Ma}(z)$ and $H_{Mc}(z)$ be given by

$$H_{Ma}(z) = \sum_{n=0}^{N_{Ma}-1} h_{Ma}(n)z^{-n} \quad (1)$$

and

$$H_{Mc}(z) = \sum_{n=0}^{N_{Mc}-1} h_{Mc}(n)z^{-n} \quad (2)$$

$H_{Ma}(z)$ and $H_{Mc}(z)$ may be expressed in their R polyphase components as

$$H_{Ma}(z) = \sum_{r=0}^{R-1} z^{-r} \sum_{n=0}^{N_{Ma}/R-1} h_{Ma}(nR+r) z^{-nR} \quad (3)$$

and

$$H_{Mc}(z) = \sum_{r=0}^{R-1} z^{-r} \sum_{n=0}^{N_{Mc}/R-1} h_{Mc}(nR+r) z^{-nR} \quad (4)$$

In (3) and (4), we have assumed that N_{Ma} and N_{Mc} are divisible by R . If N_{Ma} and N_{Mc} are not divisible by R , then zero valued terms are added to $H_{Ma}(z)$ and $H_{Mc}(z)$ until they are divisible by R . The synthesis structure for $R = 3$ is shown in Fig.2. It can be seen from Fig.2 that the number of memory accesses is reduced by a factor of R but $R - 1$ additional storage locations are needed to store the outputs of the polyphase filters. Furthermore, the coefficients of the polyphase filters may not be symmetrical.

Fig.2 Polyphase implementation of $H_{Ma}(z)$ and $H_{Mc}(z)$ for $R = 3$. Note that the number of memory accesses is reduced by a factor of R .

4. $H_{Ma}(z)$ AND $H_{Mc}(z)$ MAY SHARE THE SAME SET OF DELAY ELEMENTS

If $H_{Ma}(z)$ and $H_{Mc}(z)$ are implemented as shown in Fig.3, obviously they may share the same delay elements as shown in Fig.4. The implementation structure of Fig.4 is useful if the delay elements of $H_{Ma}(z)$ and $H_{Mc}(z)$ are implemented in FPGA where the total number of delay elements must be reduced to the minimum. The corresponding polyphase structure for $R = 2$ is shown in Fig.5.

Fig.3 An implementation for $H_{Ma}(z)$ and $H_{Mc}(z)$.

Fig.4 An implementation for $H_{Ma}(z)$ and $H_{Mc}(z)$ where $H_{Ma}(z)$ and $H_{Mc}(z)$ share the same delay elements assuming that $H_{Ma}(z)$ and $H_{Mc}(z)$ have the same order.

Fig.5 Polyphase implementation of the circuit shown in Fig.4 for $R = 2$.

5 REDUCING MEMORY FETCH FOR $H_a(z^M)$

Let

$$X(z) = \sum_n x(n)z^{-n} \quad (5)$$

$X(z)$ may be expressed in its polyphase components as

$$X(z) = \sum_{m=0}^{M-1} z^{-m} X_m(z^M) \quad (6)$$

where

$$X_m(z^M) = \sum_{n=0}^{\infty} x(nM + m)z^{-nM} \quad (7)$$

Since $U(z) = X(z)H_a(z^M)$, we have

$$U(z) = \sum_{m=0}^{M-1} z^{-m} X_m(z^M) H_a(z^M) \quad (8)$$

Let

$$U_m(z^M) = X_m(z^M) H_a(z^M) \quad (9)$$

From (8) and (9), $U(z)$ may be written as

$$U(z) = \sum_{m=0}^{M-1} z^{-m} U_m(z^M) \quad (10)$$

It can be seen from (10) that $U_m(z)$ is the m^{th} polyphase component of $U(z)$. From (9), it is obvious that the polyphase components of $U(z)$ are the polyphase components of $X(z)$ filtered by the same filter $H_a(z^M)$. Because the same filter is used to filter all the polyphase components of $U(z)$, the number of ROM fetch operations to fetch the coefficients of $H_a(z^M)$ can be reduced if a further delay of M sampling interval in the computation of $U(z)$ can be tolerated. Let

$$U'(z) = z^{-M} U(z) \quad (11)$$

$U'(z)$ may be obtained using the configuration shown in Fig.6. In Fig.6, the down sampling and up sampling processes do not cause any aliasing problem. Let $h_a(n)$ be the n^{th} coefficient of $H_a(z)$. In the implementation of Fig.6, in order to minimize the number of ROM fetch operations to fetch the coefficients of $H_a(z)$, when $h_a(n)$ for a particular value of n is fetched, it should be used to multiply with the corresponding elements of $X_m(z)$ for all m before it is discarded. This will reduce the number of ROM fetch operations to fetch the coefficients of $H_a(z)$ by a factor of M in the expense of an additional $2M$ storage locations on the FPGA.

Fig.6 Implementation of $H_a(z^M)$ where the input signal is split into its polyphase components.

The filters $H_a(z)$ in Fig.6 may also be split into its polyphase components to reduce the number of memory fetches for fetching $x(nM+m)$ in the same way as implementing $H_{Ma}(z)$ and $H_{Mc}(z)$ by splitting them into their polyphase components as discussed in Section 3. If each $H_a(z)$ in Fig.6 is split into R polyphase components, additional MR storage locations are needed.

6 CONCLUSION

Several techniques for reducing RAM and ROM fetches in FPGA implementation of filters synthesized using the frequency-response masking technique are presented. The number of signal data fetches from RAM when implementing an FIR filter can be reduced by splitting the filter into its polyphase components. The number of coefficient data fetches in the implementation of the band-edge shaping filter can be reduced by splitting the input signal into its polyphase components. A method for $H_{Ma}(z)$ and $H_{Mc}(z)$ to share the same delay elements are also presented.

7 REFERENCES

- [1] Y.C. Lim, "Frequency-response masking approach for the synthesis of sharp linear phase digital filters," *IEEE Transactions on Circuits and Systems*, vol. CAS-33, no. 4, pp. 357-364, April 1986.
- [2] Y.C. Lim, "A digital filter bank for digital audio systems," *IEEE Transactions on Circuits and Systems*, vol. CAS-33, no. 8, pp. 848-849, August 1986.
- [3] Y.C. Lim and Y. Lian, "The optimum design of one- and two-dimensional FIR filters using the frequency response masking technique," *IEEE Transactions on*

- Circuits and Systems II*, vol. CAS- II-40, no. 2, pp. 88-95, February 1993.
- [4] Y.C. Lim and Y. Lian, "Frequency-response masking approach for digital filter design: complexity reduction via masking filter factorization," *IEEE Transactions on Circuits and Systems II*, vol. CAS-II-41, no. 8, pp. 518-525, August 1994.
 - [5] Y.C. Lim and S.H. Low, "Frequency-response masking approach for the synthesis of sharp Two-dimensional diamond-shaped filters," *IEEE Transactions on Circuits and Systems II*, vol. CAS-II-45, no. 12, pp. 1573-1584, December 1998.
 - [6] R. Yang, B. Liu and Y.C. Lim, "A new structure of sharp transition FIR filters using frequency-response masking," *IEEE Transactions on Circuits and Systems*, vol. CAS-35, no. 8, pp. 955-966, August 1988.
 - [7] O. Gustafsson, H. Johansson and L. Wanhammar, "Design and efficient implementation of high-speed narrow-band recursive digital filters using single filter frequency masking techniques," *Proceedings of 2000 International Symposium on Circuits and Systems*, vol. III, pg.359-362, May 2000.
 - [8] H. Johansson, "New class of frequency-response masking FIR filters," *Proceedings of 2000 International Symposium on Circuits and Systems*, vol. III, pg.81-84, May 2000.
 - [9] H. Johansson and L. Wanhammar, "High-speed recursive digital filters based on frequency-response masking approach," *IEEE Transactions on Circuits and Systems II*, vol. CAS-II-47, pp. 48-61, January 2000.
 - [10] G. Rajan, Y. Neuvo and S.K. Mitra, "On the design of sharp cutoff wide-band FIR filters with reduced arithmetic complexity," *IEEE Transactions on Circuits and Systems*, vol. CAS-35, pp. 1447-1454, November 1988.
 - [11] G. Rajan, Y. Neuvo and S.K. Mitra, "Computationally efficient wide-band FIR filters with very narrow transition bands," *Proceedings of 1988 International Symposium on Circuits and Systems*, pg. 2013-2017.
 - [12] T. Saramäki, Y.C. Lim and R. Yang, "The synthesis of half-band filter using frequency-response masking technique," *IEEE Transactions on Circuits and Systems II*, vol. CAS-II-42, no. 1, pp. 58-60, January 1995.
 - [13] T. Saramäki, "Design of computationally efficient FIR filters using periodic subfilters as building blocks," in *The Circuits and Filters Handbook*, ed. W.K. Chen, CRC Press, Inc., pp.2578-2601, 1995.
 - [14] T. Saramäki and A.T. Fam, "Subfilter approach for designing efficient FIR filters," *Proceedings of 1988 International Symposium on Circuits and Systems*, pg. 2903-2915.