Research Article

Low-Cost Design of an FIR Filter by Using a Coefficient Mapping Method

Ming-Chih Chen and Hong-Yi Wu

Department of Electronic Engineering, National Kaohsiung First University of Science and Technology, Kaohsiung City 811, Taiwan

Correspondence should be addressed to Ming-Chih Chen; mjchen@nkfust.edu.tw

Received 14 September 2013; Accepted 9 October 2013

Academic Editor: Teen-Hang Meen

Copyright © 2013 M.-C. Chen and H.-Y. Wu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This work presents a novel coefficient mapping method to reduce the area cost of the finite impulse response (FIR) filter design, especially for optimizing its coefficients. Being capable of reducing the area cost and improving the filter performance, the proposed mapping method consists of four steps: quantization of coefficients, import of parameters, constitution of prime coefficients with parameters, and constitution of residual coefficients with prime coefficients. Effectiveness of the proposed coefficient mapping method is verified by selecting the 48-tap filter of IS-95 code division multiple access (CDMA) standard as the benchmark. Experimental results indicate that the proposed design with canonical signed digit (CSD) coefficients can operate at 86 MHz with an area of 241,813 μm^2 , leading to a throughput rate of 1,382 Mbps. Its ratio of throughput/area is 5,715 Kbps/ μm^2 , yielding a higher performance than that of previous designs. In summary, the proposed design reduces 5.7% of the total filter area, shortens 25.7% of the critical path delay, and improves 14.8% of the throughput/area by a value over that of the best design reported before.

1. Introduction

Digital signal processing applications are common in home entertainment systems, television sets, high-fidelity audio equipment, and information systems. The digital filter is an important component in mathematical operations on a sampled, discrete-time signal to enhance the certainty of a signal. The digital filter is characterized by its transfer function. Two digital filters are infinite impulse response (IIR) and finite impulse response (FIR) filters. The IIR filter consists of a transfer function with feedback mode, and the FIR filter consists of the function with nonfeedback mode. Commonly found in image processing, audio processing, and wireless communications, FIR filter applications are characterized by a linear phase, arbitrary magnitude, and relatively easy implementation. The filter hardware consists of adders, subtractors, shifters, and registers. Many related works [1–9] attempt to reduce the number of these required components in filter implementation, especially for the optimization of coefficients' realization. Experimental results demonstrate that the proposed coefficient mapping method performs better than previous designs in terms of area ratio.

The rest of this paper is organized as follows. Section 2 briefly describes previous researches for filter optimization. Section 3 then describes the coefficient mapping method. Next, Section 4 summarizes the experimental results and compares them with those of other previous designs. Conclusions are finally drawn in Section 5, along with recommendations for future research.

2. Background

2.1. Digital FIR Filter. Digital filters generally vary in coefficients, based on their specifications. The design of coefficients in a filter can be divided into four portions: coefficient selection, coefficient identification, searching algorithm, and coefficient quantization.

(1) Coefficient Selection. Typically determined by a set of filter specifications, coefficient selection must consider the number of taps, bit width, and filter complexity. According to the different complexities of coefficients, different algorithms are used to find the common subexpressions (CSs) and eliminate them for obtaining the best area reduction.

FIGURE 1: Filter architecture of a direct form.

(2) *Coefficient Identification.* Coefficients must be encoded to determine the area cost of a filter and the frequency of extracting common subexpressions. In coefficient encoding, the common expression is binary encoding. However, this encoding method causes more 1's signals in data expression and more calculations in hardware implementation. Hence, optimizing more coefficients [10, 11] involves using the canonic signed digit (CSD) expression to eliminate many 1's signals and using less common subexpressions.

(3) *Searching Algorithm.* The searching algorithm can find more common subexpressions to reduce the area cost of filter. Although many works [1–9, 12, 13] have attempted to find as many common subexpressions as possible, a more complex algorithm may not yield a higher performance, especially in coefficients with a low complexity.

(4) *Coefficient Quantization.* Coefficient quantization is an effective means of reducing the number of logic gates while implementing a filter. When the coefficients are quantized for implementation, the commonly used rounding method causes a deviation in the time and frequency responses of the implemented filter from the ideal response. Sensitivity of the filter response is of priority concern when quantizing the coefficients.

2.2. Optimization of the FIR Filter. The equation of FIR filter can be expressed as (1), and its transferred function can also be expressed as (2):

$$y(n) = \sum_{k=0}^{N-1} h(k) x(n-k), \quad (1)$$

$$H(z) = \sum_{n=0}^{N-1} h(n) z^{-n}. \quad (2)$$

Parameter n in (1) and (2) is expressed as the number of taps. This parameter is related to the output of a filter system and its frequency responses. Two implementation methods used for a filter are direct and cascade architectures. Here, the direct architecture is of priority concern, especially for enhancing the frequency responses of a filter with all zeros. The direct architecture can also be divided into direct and transposed forms. The direct form consists mainly of multipliers, adders, and registers. Figure 1 shows the architecture of the direct form. An n -tap filter with a direct form requires n copies of a multiplier, $n-1$ copies of an adder, and n copies of a register.

Coefficients $h(0)$ – $h(k)$ can be expressed as binary numbers. The binary expressions for the coefficients can be found with the common subexpressions (CSs) between them. In previous researches [1–9], there are three common subexpression elimination (CSE) methods including horizontal, vertical, and mixed searching methods. The method in [1] uses CSD expressions [2] for the coefficients and uses horizontal searching algorithm to find the CSs including $(1, 0, 1)$, $(1, 0, -1)$, $(1, 0, 0, 1)$, and $(1, 0, 0, -1)$. This method first optimizes the CS with the highest appearance until no CS can be extracted. In [3, 7], this method also uses CSD expressions, yet uses vertical search to find the CSs, including $(1, -1)$ and $(-1, 1)$. By modifying the vertical searching method in [8], the method in [4] first uses horizontal search to extract the CSs including $(1, 0, 1)$, $(1, 0, -1)$, $(1, 0, 0, 1)$ and $(1, 0, 0, -1)$, which is expressed in CSD format. Thereafter, this method performs a vertical search to extract the CSs, including $(1, 0, 1)$ and $(-1, 0, 1)$.

In addition to using two horizontal methods and one vertical searching method to extract the CSs, the method in [5] also uses a multiplier-adder block (MAB) and structure adder (SA) to construct the CSs and their residues. Besides finding two CSs with the same appearances, the method in [6] extracts the CS with a smaller bit width. A previous work [8] developed two methods for extracting the CSs in CSD format. The first method analyzes the CSs with 3-, 4-, and 5-bit by performing the statistics of their appearances. The second method searches the coefficients up to down and extracts the CSs between them by using vertical search. The method in [9] proposes a rule in which the depth of logic gates cannot be increased by performing the horizontal search in the same way as in [4] and the vertical search in the same way as in [8].

Following implementation of the above searching methods, the CSs can be extracted and the same CSs can be used for calculation only once. Calculation times of the filter are reduced due to the extractions. These searching methods can also reduce the required number of adders and subtractors. To verify the different searching methods, the 48-tap filter of IS-95 CDMA is selected as the benchmark.

3. Proposed Coefficient Mapping Method

The mapping method divides the coefficients into two parts: primary coefficients and remaining coefficients. The parameters that are set up in the algorithm are Q_h and Q_p . The constitution of coefficients has two steps: constitution of primary coefficients and calculation of multiple relations between the primary and remaining coefficients. The operation steps of the mapping method are described as follows.

Step 1. Normalize the n -tap coefficients c_i by multiplying 2^m . The normalized results are $h_i = c_i \times 2^m$, for $0 \leq i \leq n-1$.

Step 2. Separate normalized coefficients h_i into two parts, H_1 and H_2 :

$$\begin{aligned} H_1 &= \{h_j\}, & 0 \leq j \leq \frac{n}{2} - 1, \\ H_2 &= \{h_k\}, & \frac{n}{2} \leq k \leq n-1. \end{aligned} \quad (3)$$

(a)

(b)

FIGURE 2: (a) Frequency responses, poles and zeros distributions of the original filter. (b) Frequency responses, poles and zeros distributions of the modified filter.

Step 3. Quantize h_j and h_k based on Q_h . Equations (4) and (5) show the relations between Q_h , h_j , and h_k

$$h_j = Q_h(S_{h,j}) \times \Delta h_j, \quad (4)$$

$$h_k = Q_h(S_{h,k}) \times \Delta h_k. \quad (5)$$

Parameters Δh_j and Δh_k are the variances after performing the quantization. Let $h'_j = Q_h(S_{h,j})$ and let $h'_k = Q_h(S_{h,k})$. Equations (4) and (5) thus become the following equations separately:

$$h_j = h'_j \times \Delta h_j, \quad (6)$$

$$h_k = h'_k \times \Delta h_k. \quad (7)$$

Step 4. Find a multiple value $p_{k,j}$ that is between h_k and h_j , and find a multiple value $p'_{k,j}$ that is between h'_k and h'_j .

The relations are shown as

$$h_k = p_{k,j} \times h_j \implies p_{k,j} = \frac{h_k}{h_j}, \quad (8)$$

$$h'_k = p'_{k,j} \times h'_j \implies p'_{k,j} = \frac{h'_k}{h'_j}. \quad (9)$$

Step 5. Quantize $p'_{k,j}$ based on Q_p . Equation (10) shows the relation between Q_p and $p'_{k,j}$. Parameter $\Delta p_{k,j}$ denotes the variance after performing the quantization:

$$p'_{k,j} = Q_p(S_{p,k,j}) \times \Delta p_{k,j}. \quad (10)$$

Let $p''_{k,j} = Q_p(S_{p,k,j})$, and (11) shows the relation between $p'_{k,j}$ and $p''_{k,j}$:

$$p'_{k,j} = p''_{k,j} \times \Delta p_{k,j}. \quad (11)$$

Step 6. Constitute h'_k and substitute (11) with (9). The relation between h'_k , h'_j , and $p'_{k,j}$ becomes

$$h'_k = p''_{k,j} \times h'_j \times \Delta p_{k,j}. \quad (12)$$

TABLE 1: New coefficients after performing the mapping method ($Q_h = 32, Q_p = 0.25$).

Group	Sign	Coefficient		Error rate (%)
		h_j, h_k (original coefficient)	h'_j, h'_k (new coefficient)	
H_1				
c_0	—	829	832 (32×26)	0.36
c_1	—	1120	1120 (32×35)	0.00
c_2	—	1172	1184 (32×37)	1.01
c_3	—	548	544 (32×17)	0.74
c_4	+	708	704 (32×22)	0.57
c_5	+	2128	2144 (32×67)	0.75
c_6	+	2982	2976 (32×93)	0.20
c_7	+	2684	2688 (32×84)	0.15
c_8	+	1215	1216 (32×38)	0.08
H_2				
c_9	—	721	832 (832×1)	13.3
c_{10}	—	1990	1960 (1120×1.75)	1.53
c_{11}	—	1677	1776 (1184×1.5)	5.57
c_{12}	+	258	272 (544×0.5)	5.15
c_{13}	+	2765	2816 (704×4)	1.81
c_{14}	+	4157	4288 (2144×2)	3.06
c_{15}	+	3098	2976 (2976×1)	4.10
c_{16}	—	421	672 (2688×0.25)	37.4
c_{17}	—	4560	4560 (1216×3.75)	3.09

Step 7. Revert the parameters h_j and h_k by substituting (9) and (12) with (7). Equations (13) and (14) show the relations as follows:

$$h_j = Q_h(S_{h,j}) \times \Delta h_j \implies h_j = h'_j \times \Delta h_j, \quad (13)$$

$$\begin{aligned} h_k &= Q_h(S_{h,j}) \times Q_p(S_{p,k,j}) \times \Delta p_{k,j} \times \Delta h_k \\ \implies h_k &= h'_j \times p''_{k,j} \times \Delta p_{k,j} \times \Delta h_k. \end{aligned} \quad (14)$$

Let $h''_k = h'_j \times p''_{k,j}$, and (14) is then modified to

$$h_k = h''_k \times \Delta p_{k,j} \times \Delta h_k. \quad (15)$$

According to the variance, the calculation error and correct rates can be expressed as

$$e_i = \begin{cases} 1 - \Delta & \text{if } \Delta \leq 1, \\ \Delta - 1 & \text{if } \Delta > 1, \end{cases} \quad (16)$$

$$a_i = 1 - e_i. \quad (17)$$

Substitute the variances of h_k and h_j into (16). The error rate e_i can be obtained, for $0 \leq i \leq n - 1$.

For example, this work selects 18 coefficients c_0 – c_{17} from 48-tap IS-95 CDMA to perform the mapping method. The coefficients are

$$\begin{aligned} c_0 &= -0.025299, & c_1 &= -0.034179, \\ c_2 &= -0.035766, & c_3 &= -0.016723, \\ c_4 &= 0.021606, & c_5 &= 0.064941, \\ c_6 &= 0.091003, & c_7 &= 0.081909, \\ c_8 &= 0.037078, & c_9 &= -0.022003, \\ c_{10} &= -0.060729, & c_{11} &= -0.051177, \\ c_{12} &= 0.007873, & c_{13} &= 0.084381, \\ c_{14} &= 0.126861, & c_{15} &= 0.094543, \\ c_{16} &= -0.012847, & c_{17} &= -0.143463. \end{aligned} \quad (18)$$

Before performing the mapping method, this work first sets up two parameters in which Q_h equals 32 and Q_p equals 0.25 to reduce the hardware design complexity. After Steps 1, 2, 3, 4, 5, 6, and 7 of the proposed method are performed, Table 1 lists the generated h_i values. Figure 2(a) shows the frequency responses, poles, and zeros distributions of the original filter. Figure 2(b) shows the behaviors of the filter with modified coefficients after applying the proposed mapping method. The behaviors in Figures 2(a) and 2(b) are approximately the same with each other.

TABLE 2: Performance comparison of various 48-tap filter designs for generating 16-bit outputs.

Methods	Architecture level	Area (μm^2)	Info.		
			Critical path delay (ns)	Gate level Throughput (Mbps)	Throughput/area ($\text{Kbps}/\mu\text{m}^2$)
Binary	(220, 0, 48)	300284	12.61	1268	4225
CSD	(88, 74, 48)	299622	12.08	1324	4421
Paško et al. [1]	(76, 22, 48)	271357	15.08	1061	3910
Jang and Yang [3]	(72, 70, 48)	295567	15.27	1047	3545
Vinod et al. [4]	(62, 34, 48)	274025	14.32	1117	4077
Choo et al. [5]	(67, 24, 48)	259104	14.18	1128	4355
Takahashi and Yokoyama [6]	(78, 22, 48)	273902	15.28	1047	3823
Mahesh and Vinod [7]	(88, 0, 48)	255450	15.58	1271	4979
Kato et al. [8]	(64, 26, 48)	264240	14.46	1106	4187
Vinod et al. [9]	(62, 34, 48)	260976	14.25	1122	4303
Our method with binary format	(66, 0, 48)	244547 (4.3%)	11.45 (26.5%)	1397 (9.9%)	5713 (14.7%)
Our method with CSD format	(23, 33, 48)	241813 (5.3%)	11.58 (25.7%)	1382 (8.7%)	5715 (14.8%)

4. Experimental Results

For comparison, Table 2 lists various 48-tap filter designs for IS-95 CDMA. These filters can generate 16-bit output data at one clock cycle. The table shows the architecture- and gate-level information of the filters. The area cost of the architecture-level includes how many adders, subtractors, and registers are used for implementation. Gate-level information includes the area cost, critical path delay, throughput, and throughput per area. In the architecture-level, analysis results indicate that the original filter with binary coefficients has the largest area cost among other designs. The design in [7] has the smallest summation for calculating the number of adders and subtractors. More than having the smallest area cost, the proposed design also achieves the highest throughput and ratio of throughput per area in the gate-level synthesis among other designs.

Table 2 lists two filter designs: the coefficients expressed with binary and CSD formats. In the architecture-level, the proposed filter with CSD coefficients has the smallest summation for calculating the number of adders and subtractors. The proposed filter only requires a total of 56 adders and subtractors which are the smallest amounts among the previous designs. The best design is the method in [7], which has an area of $255,450 \mu\text{m}^2$ and achieves a throughput/area of $4,979 \text{ Kbps}/\mu\text{m}^2$. The proposed design with CSD coefficients can operate at 86 MHz with an area of $241,813 \mu\text{m}^2$, leading to a throughput rate of $1,382 \text{ Mbps}$. Its ratio of throughput/area is $5,715 \text{ Kbps}/\mu\text{m}^2$, which is the highest performance among the previous designs. In summary, the proposed design reduces 5.7% of the total filter area, shortens 25.7% of the critical path delay, and improves 14.8% of the throughput/area compared with the best design in [7].

5. Conclusions

This work has developed a novel filter design with coefficient mapping method. The proposed method can reduce the area

cost by finding the primary coefficients and using them to construct the remaining coefficients. The proposed method can also use several coefficients and construct all of the filter coefficients. Experimental results demonstrate that the proposed design with binary or CSD coefficients can more significantly reduce the area cost and improve the ratio of throughput/area compared with previous designs. Implementation results further demonstrate that the proposed design has the highest throughput with the lowest area cost.

Acknowledgment

This work is supported in part by the National Science Council of Taiwan under Grants NSC 99-2221-E-327-046.

References

- [1] R. Paško, P. Schaumont, V. Derudder, S. Vernalde, and D. Āuračková, "A new algorithm for elimination of common subexpressions," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 18, no. 1, pp. 58–68, 1999.
- [2] R. M. Hewlitt and E. S. Swartzlander Jr., "Canonical signed digit representation for FIR digital filters," in *Proceedings of the IEEE Workshop on Signal Processing Systems (SIPS '00)*, pp. 416–426, October 2000.
- [3] Y. Jang and S. Yang, "Low-power CSD linear phase FIR filter structure using vertical common sub-expression," *Electronics Letters*, vol. 38, no. 15, pp. 777–779, 2002.
- [4] A. P. Vinod, E. M.-K. Lai, A. B. Premkumar, and C. T. Lau, "FIR filter implementation by efficient sharing of horizontal and vertical common subexpressions," *Electronics Letters*, vol. 39, no. 2, pp. 251–253, 2003.
- [5] H. Choo, K. Muhammad, and K. Roy, "Complexity reduction of digital filters using shift inclusive differential coefficients," *IEEE Transactions on Signal Processing*, vol. 52, no. 6, pp. 1760–1772, 2004.
- [6] Y. Takahashi and M. Yokoyama, "New cost-effective VLSI implementation of multiplierless FIR filter using common

- subexpression elimination,” in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '05)*, pp. 1445–1448, May 2005.
- [7] R. Mahesh and A. P. Vinod, “A new common subexpression elimination algorithm for realizing low-complexity higher order digital filters,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 2, pp. 217–229, 2008.
 - [8] K. Kato, Y. Takahashi, and T. Sekine, “A new horizontal and vertical common subexpression elimination method for multiple constant multiplication,” in *Proceedings of the 16th IEEE International Conference on Electronics, Circuits and Systems (ICECS '09)*, pp. 124–127, December 2009.
 - [9] A. P. Vinod, E. Lai, D. L. Maskell, and P. K. Meher, “An improved common subexpression elimination method for reducing logic operators in FIR filter implementations without increasing logic depth,” *Integration, the VLSI Journal*, vol. 43, no. 1, pp. 124–135, 2010.
 - [10] C.-H. Chang and M. Faust, “On ‘a new common subexpression elimination algorithm for realizing low-complexity higher order digital filters,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 29, no. 5, pp. 844–848, 2010.
 - [11] D. Shi and Y. J. Yu, “Design of linear phase FIR filters with high probability of achieving minimum number of adders,” *IEEE Transactions on Circuits and Systems I*, vol. 58, no. 1, pp. 126–136, 2011.
 - [12] A. Shahein, Q. Zhang, N. Lotze, and Y. Manoli, “A novel hybrid monotonic local search algorithm for FIR filter coefficients optimization,” *IEEE Transactions on Circuits and Systems*, vol. 59, no. 3, pp. 616–627, 2011.
 - [13] B. Y. Kong and I. C. Park, “FIR filter synthesis based on interleaved processing of coefficient generation and multiplier-block synthesis,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 31, no. 8, pp. 1169–1179, 2012.