

Floorplan-aware Low-complexity Digital Filter Synthesis for Low-power & High-speed *

Dongku Kang, Hunsoo Choo and Kaushik Roy
Purdue University
Electrical and Computer Engineering
West Lafayette, IN USA
{dkang,chooh,kaushik}@purdue.edu

Abstract

In this paper, we propose a floorplan-aware complexity reduction methodology for digital filters. The proposed scheme integrates high-level synthesis and floorplan to obtain improvement in both computational complexity and interconnect delay. Physical information of floorplan is incorporated into architecture synthesis. By considering interconnects while synthesizing reduced-complexity filters, the layout-centric architecture achieves better performance in the evolving scaled technologies. In our experiments, we achieved 15% improvement in critical-path delay over conventional design methodology.

1. Introduction

In digital signal processing systems, digital filters are the most widely used computation block. In many applications, which require high speed and low power, fixed coefficient finite impulse response (FIR) filters are commonly used. These filters are often implemented in application specific integrated circuits (ASIC) instead of the general purpose digital signal processor (DSP) for high speed and low power. To improve processing speed with lower power dissipation in ASIC design, complexity reduction schemes have been proposed by many researchers.

As a method for complexity reduction, graph theoretical approaches were proposed in [1], [2]. The *differential coefficients for multiplierless implementation* (DCMI) and *optimal differential coefficients for multiplierless implementation* (ODCMI) were proposed using computation reordering and computation reuse to reduce complexity. *Shift inclusive differential coefficients* (SIDC) optimization was also proposed in [3]. By considering coefficient shifting as well as differential coefficients, SIDC identifies the unique non-redundent computations of multiple constant multiplications (MCM)

[4], exploiting the arithmetic equivalence property. In the proposed work, we will consider SIDC filters for low-power operation.

Conventional logic-centric methodologies focus on the reduction of the total number of adders. However, in recent nanometer technology, interconnects significantly contribute to delay. For instance, the latency of a 1.0 mm interconnect can be 100ps in 0.1 um technology, while intrinsic delay of the MOSFET is reduced to 1ps [5].

In SIDC implementation, a product of a filter coefficient and input vector is reused for the computation of other products. This computation sharing scheme can have serious impact on the interconnect. Depending on the shared product that is reused in other computations, there may exist considerably long interconnects to distribute the products. To prevent interconnect overhead, computation blocks in the critical path have to be placed closer to one another. However, wirelengths of the interconnects are not available before floorplan. Therefore, to enhance the speed and power of the fabricated ASIC's in the advanced technologies, it is highly desirable that filter synthesis procedure is incorporated with floorplan. To the best of our knowledge, there is no research that addresses this issue. In our work, we will address the interconnect problem in the framework of filter design.

In this paper, we propose a novel floorplan-aware SIDC scheme that considers physical information. The proposed scheme simultaneously performs both SIDC complexity reduction and floorplan. By considering critical-path delay (including interconnects) in filter synthesis, low-complexity filter with better performance can be obtained.

2. Background

The input-output relationship of an M-tap linear time-invariant (LTI) FIR filter can be expressed by Eq.

* This research was supported in part by Semiconductor Research Corporation.

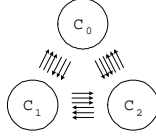


Figure 1. Graph using SIDC when the wordlength $W = 2$.

1. The c_i and $x(n-i)$ represent the i th coefficient and the input scalar at the time $n-i$, respectively. $P_i^{(n)}$ denotes the product of $c_i \cdot x(n-i)$ for $i = 0, 1, \dots, M-1$ at the time instance n . Eq. 1 can be implemented in direct form or transposed direct form. In general, transposed direct form provides faster implementation because it requires one multiply-and-add operation in the critical path. When filtering operation is implemented in the transposed direct form, the inner products of Eq. 1 is recasted into products of scalars and an input vector.

$$y(n) = \sum_{i=0}^{M-1} c_i x(n-i) = \sum_{i=0}^{M-1} P_i^{(n)} \quad (1)$$

A low-complexity digital filter design methodology (SIDC) was proposed in [3]. SIDC utilizes arithmetic equivalence of filter operation. For instance, let us consider a simple multiplication of Eq. 2. Note that a product of $2^L \cdot c_i \cdot x(n)$ can be easily obtained by L-bit shift of $c_i \cdot x(n)$ because shift operation can be implemented by hard-wiring. Using the shifting property, the product $P_j^{(n)}$ is represented as follows:

$$P_j^{(n)} = (c_j - c_i)x(n) + c_i x(n) \quad (2)$$

$$= (c_j \pm 2^L \cdot c_i)x(n) \mp 2^L \cdot c_i x(n) \quad (3)$$

Low-complexity implementation is obtained by selecting pairs of $\{c_i, c_j\}$ and corresponding shift operation, which maximally simplifies the SIDC's $\Delta_{i,j}^L = (c_j - 2^L c_i)$ for $i, j = 0, 1, \dots, M-1$. $\Delta_{i,j}^L$ is called *color*. This problem is mapped to a directed multigraph (Fig. 1) by considering different values for $L = 0, 1, \dots, W$ in Eq. 3, where W denotes the wordlength [3]. The solution can be obtained by using a well-known graph theoretical algorithm.

3. Floorplan-aware SIDC

SIDC algorithm proposed in [3] determines subgraphs of the complete multigraph which lead to a unique implementation of a digital FIR filter. To determine acyclic subgraphs which can be directly translated into low-complexity filter, root vertices and required colors have to be chosen. Root vertex denotes

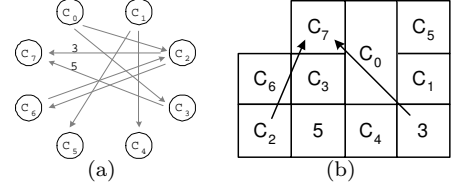


Figure 2. Multigraph representation of SIDC FIR filter and floorplan

a product which has to be computed first and reused to compute other products. In our formulation, we adapted the algorithm in [3] to select root vertices and required colors. After choosing root vertices and required colors, the vertices in the SIDC graph is connected by edges with selected colors. As an example, in Fig. 2(a), c_0 and c_1 are chosen as root vertices. Also, all edges are colored with 3 or 5 assuming that they are chosen as required colors to compute the given products for filtering.

Traditionally, floorplanning is performed after filter synthesis is completed. In the multigraph of Fig. 2(a), only one incoming edge for a vertex has to be chosen in the filter synthesis stage. In Fig. 2(a), c_7 can be computed by reusing either c_2 or c_3 . Note that the selection of edge does not change the implementation complexity. In this example, let us assume that the incoming edge from c_2 is selected for c_7 . The corresponding floorplan example is shown in Fig. 2(b).

However, if we carefully inspect the floorplan, we can further reduce the interconnect by choosing a different incoming edge. An example is shown in Fig. 3. From the initial floorplan in Fig. 2(b), we can choose different incoming edge for c_7 . By selecting incoming edge from c_3 instead of c_2 , interconnect lengths are reduced in Fig. 3(a). After this movement, by moving the computation block of color 5 closer to c_7 in Fig. 3(b), we can further improve the design in terms of delay and length of interconnects. Therefore, it is highly desirable to consider physical information during the synthesis stage. We propose a novel floorplan-aware filter synthesis methodology that considers interconnect.

When one incoming edge is chosen for the non-root vertices, we can implement FIR filter in transposed direct form as shown in Fig. 4. In *primary computation network*, multiplications of root vertices and required colors ($\Delta_{i,j}^L$) are computed. Using the results from the *primary computation network*, products involving other coefficients are computed in the *secondary adder network*. Finally, the output is computed in *delay-and-add network*, using the products computed in the primary and secondary networks.

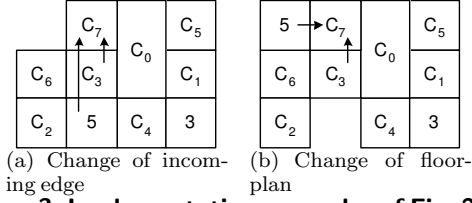


Figure 3. Implementation examples of Fig. 2(b).

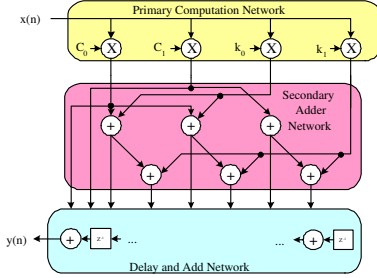


Figure 4. Implementation of FIR filter using transposed direct form and SIDC.

In the proposed framework, the initial run of conventional SIDC generates a multigraph consisting of all selected colors. Then, we need to choose only one incoming edge for non-root vertices. Also, when choosing the incoming edges, floorplan information has to be incorporated to estimate interconnect cost. We integrated floorplan and edge selection into a standard simulated annealing algorithm. By performing floorplan-based edge selection, the proposed algorithm transforms the SIDC graph that is more adapted to scaled technology.

The framework is shown in Fig. 5. First, an initial solution is generated from conventional SIDC. Then, simulated annealing is activated. Simulated annealing changes floorplan and selects different incoming edges. To effectively encode the floorplan in the algorithm, we used sequence pair [6].

The interconnect delay T is estimated as follows[7]:

$$T = R_{int}C_{int} + 2.3(R_{tr}C_{int} + R_{tr}C_L + R_{int}C_L) \quad (4)$$

where R_{int} and C_{int} are the total resistance and total capacitance of a wire segment. R_{tr} and C_L denotes resistance of transistor and load capacitance. The circuit parameters in Eq. 4 were obtained from the physical layout.

After each movement, cost is measured. To reduce the critical-path delay of a filter, wire lengths of the non-critical-path interconnect and area may increase.

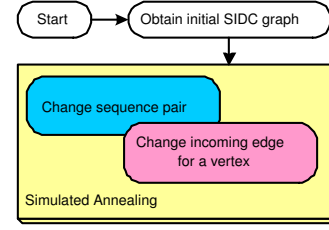


Figure 5. Flowplan-aware SIDC algorithm.

Example	EX1	EX2	EX3	EX4	EX5	EX6
Type	PM LP	BW LP	PM BS	LS BS	PM LP	LS BP
f_{p1}	0.15	0.5	0.2	0.2	0.4	0.3
f_{s1}	0.25	0.53	0.24	0.24	0.4175	0.315
f_{p2}			0.42	0.42		0.705
f_{s2}			0.38	0.38		0.69
R_p (dB)	2	3	3	3	3	1
R_s (dB)	60	65	50	50	70	70
Order	35	80	98	126	214	350

- BW: Butterworth, PM: Parks-McLellan, LS: Lease-square,
- LP: Low Pass, BP: Band Pass, BS: Band Stop

Table 1. Example Filters Description

To balance those factors, we formulated the cost function, which includes critical-path delay, wire and area. Each factor is weighted by α , β and γ .

$$cost = \alpha Delay + \beta Wire + \gamma Area \quad (5)$$

By changing incoming edges for non-root vertices, the proposed methodology generates different architecture subject to scaled technology. Also, it helps to avoid excessively long interconnects for better routability. Furthermore, by balancing the fanout, it relieves the extra buffer insertion problem.

4. Experimental Results

The filters used in our experiments are listed in Table 1. The Parks-McLellan (PM), Least-squares (LS), and Butterworth (BW) FIR filters are used as examples. The passband is described by ripple R_p and frequency f_{p1} , f_{p2} while the stopband is specified by ripple R_s and frequency f_{s1} , f_{s2} . For number representation, signed-magnitude scheme is used. Each filter example is optimized using the SIDC approach. The filters are assumed to have symmetric coefficients and implemented in transposed direct form as shown in Fig. 4. All coefficients are uniformly scaled to fit in a defined wordlength [1]. As discussed earlier, shift operations are implemented by hard-wiring. For wordlength, 12 bit coefficients are considered. *Brent-Kung* architecture is used for the adder component. TSMC 0.25 μ technology is used for implementation. Results are compared to the SIDC approach [3]. After reducing the

(mm)	EX1	EX2	EX3	EX4	EX5	EX6
FASIDC	15.7	23.8	52.2	53.4	76.1	128.0
SIDC	14.4	20.5	50.9	53.8	72.9	126.3

Table 2. Estimated total wirelength

(mm ²)	EX1	EX2	EX3	EX4	EX5	EX6
FASIDC	0.95	1.84	2.61	3.51	4.25	7.08
SIDC	1.10	1.76	2.99	3.23	4.48	6.69

Table 3. Estimated Area

complexity by *SIDC*, the same floorplan algorithm is applied. The results of the proposed approach is denoted as *FASIDC* (*Floorplan-aware SIDC*).

Fig. 6 shows the estimated critical-path delay. Interconnect delay is based on Eq. 4. *FASIDC* reduced the interconnect delay and balanced the fanout for the adders in the critical-path. Estimated reduction for the critical-path delay was 21%.

Several factors affect the accuracy of the delay estimation (Eq. 4). First, each metal layer has different C_{int} values. Second, wire lengths are estimated using the half-perimeter wirelength. However, actual lengths depend on the port locations in the computation components. Taking these factors into consideration, we synthesized examples and obtained layouts using *Place&Routing* tools. After extracting parasitic components, the critical-path delays are measured by *Pathmill* [8]. In Fig. 7, critical-path delay of the synthesized circuits are shown. On average, *FASIDC* achieved 15% improvement over *SIDC*.

To reduce the interconnect lengths in the critical-path, other interconnects in the non-critical paths can be increased. Also, depending on the α , β and γ values in Eq. 4, there can be trade-off between delay, wirelength and area. In our experiment, wirelengths and areas of *FASIDC* were similar to the result of *SIDC*. Table 2 and Table 3 show estimated total wirelength and area.

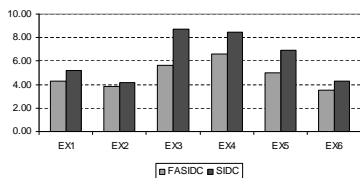


Figure 6. Estimated critical-path delay of adder network block

5. Conclusion

We proposed floorplan-aware complexity reduction methodology for digital filters. By reusing the computation from the nearest location in floorplan, we could reduce the interconnect delay of the critical-path. Also, by balancing the fanouts of the reused components, the speed of the circuit can be further improved. Floorplan and architecture synthesis were performed simultaneously to achieve this goal. Compared to the traditional *SIDC* approach, floorplan-aware *SIDC* achieved 15% reduction in critical-path delay.

References

- [1] K. Muhammad and K. Roy. A graph theoretic approach for synthesizing very low-complexity high-speed digital filters. *IEEE trans. Computer-Aided Design of Integrated Circuits and Systems*, 21(2), Feb 2002.
- [2] K. Muhammad and K. Roy. A graph theoretic approach for design and synthesis of multiplierless fir filters. In *ISSS*, 1999.
- [3] H. Choo, K. Muhammad, and K. Roy. MRPF: an architectural transformation for synthesis of high-performance and low-power digital filters. In *Design, Automation and Test in Europe Conference and Exhibition*, March 2003.
- [4] R. I. Hartley. Subexpression sharing in filters using canonic signed digital multipliers. *IEEE Trans. Circuits and Systems II*, 43(10):677–688, 1996.
- [5] J. Meindl. Interconnect limits on gigascale integration. In *Electrical Performance of Electronic Packaging*, pages 25–27, Oct 1999.
- [6] H. Murata et al. Vlsi module placement based on rectangle-packing by the sequence-pair. *IEEE Transactions on CAD Design of Integrated Circuits and Systems*, 15:1518–1524, Dec 1996.
- [7] Y. Fang and D. Wong. Simultaneous functional-unit binding and floorplanning. *IEEE/ACM International Conference on Computer-Aided Design I*, pages 317–321, Nov 1994.
- [8] Pathmill, version 2002.03, Synopsys inc.

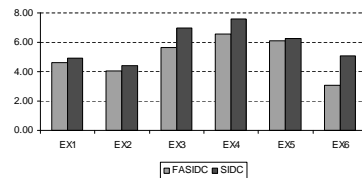


Figure 7. Critical-path delay of adder network measured by pathmill