A High Performance FIR Filter Dedicated to Digital Video Transmission

Shun Morikawa[†]   Keisuke Okada[†]   Sumitaka Takeuchi[‡]   Isao Shirakawa[†]

[†]Dept. Information Systems Engineering,
Faculty of Engineering, Osaka University
2–1 Yamada-Oka, Suita, Osaka 565 Japan
Tel: +81-6-879-7808
Fax: +81-6-875-5902
e-mail: {morikawa, okada, sirakawa}
@ise.eng.osaka-u.ac.jp

[‡] System LSI Lab.,
Mitsubishi Electric Corporation
4–1 Mizuhara, Itami, Hyogo, 664 Japan
Tel: +81-727-84-7343
Fax: +81-727-84-7439
e-mail {stakeuch}@lsi.melco.co.jp

Abstract— **A digital filter is one of the fundamental elements in the digital video transmission, and a multiplier acts as the key factor that determines the operation speed and silicon area of the filter. Even though the coefficients to the filter are desired to be programmable, it is possible to change coefficients in the vertical fly-back interval of television receivers. This allows the preloadability of coefficients to the filter such that each coefficient can be treated as a constant during the filtering operation.**

Motivated by such functionalities, a novel multiplier together with an FIR filter architecture is described, which has been designed by means of a 0.5μm double metal CMOS technology.

I. INTRODUCTION

A digital FIR filter is one of the basic components in a digital video transmission system, as illustrated in Fig. 1, for which high speed, low power consumption, and small silicon area are the most important design factors. A multiplier is the main functional block which affects the performance of the FIR filter, and a variety of high speed multipliers have been implemented on a single or a number of VLSI chips[1]-[3]. However, it is not easy to accomplish both the fast operation and the low power consumption in a small silicon area. To overcome these difficulties, it is of primary importance to develop an application specific multiplier and FIR filter architecture dedicated to the digital video transmission.

It should be noticed that the band width of the digital video transmission must be from 6MHz for the terrestrial NTSC to 54MHz for the broadcast satellite, and that the requirements for a multiplier used in an FIR filter are the operation speed of 20-100MHz and the data precision of 8-10 bits. Moreover, the following two features have to be taken into account from the viewpoint of functional operations.

1. Programmability and preloadability of coefficients: This means that each coefficient can be treated as a constant during the filtering operation.

2. FIR filtering with more than 10 taps: For the purpose of the high speed operation, the same number of multipliers should be integrated.

In this paper, first a novel architecture is proposed for the multiplier in Section II, aiming at implementing these two features. Design methodologies are described for functional blocks in Section III, together with the hardware evaluation. In Section IV, an FIR filter architecture is devised by means of the proposed multipliers, and a number of implementation results are also shown to demonstrate the practicable performance. Section V summarizes the concluding remarks.
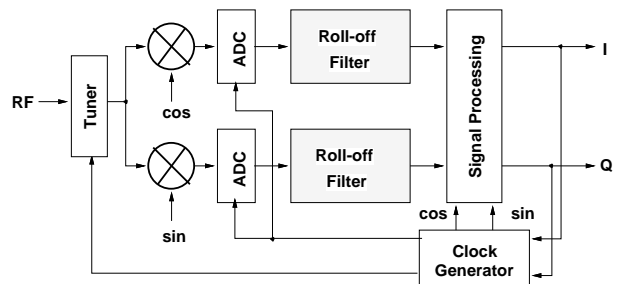

Fig. 1. Block diagram of demodulation part in video transmission system

II. MULTIPLIER ARCHITECTURE

The basic idea of high speed parallel multipliers reported so far is based on the improvement of Booth's
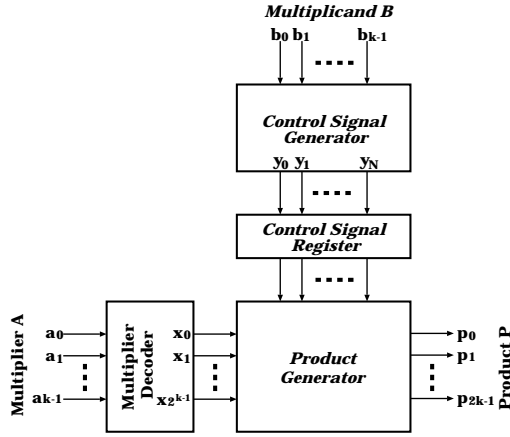
Fig. 2. Basic configuration of proposed multiplier

decoding methods, the reduction of partial products, and the development of fast adders to sum up partial products[4]-[5]. However, the proposed architecture is based on the idea that each bit of the product is to be determined by a simple logical combination of the bits of the multiplier and multiplicand[6]. To this end, our architecture includes a step of converting a multiplier represented by the binary code to the one represented by the l-out-of-N code, as adopted for the bit/word-line selection signal in memory devices. This idea seems to be realistic because the bit size of the multiplier of our interest is up to 10, instead of 32 or 64.

When each signal line represents the value of a given multiplier, the multiplication can be executed by connecting that line to the final output which represents the required product, where the connection is determined in accordance with a given multiplicand. In Fig. 2, a block diagram of the proposed multiplier is shown, which is constructed of the following four functional blocks; (1) Multiplier Decoder, (2) Control Signal Generator, (3) Control Signal Register, and (4) Product Generator.

In this section, functions of these blocks are exemplified by using such a simple model, in which both multipliers and multiplicands are represented by the 2-bit binary code.

A. Multiplier Decoder

Suppose that an input to this block is a binary coded multiplier $A(=a_0 \times 2^0 + a_1 \times 2^1)$, and an output is a 1-out-of-N code $X(=x_1 \sim x_3)$, where each x_i represents the value of multiplier A. The relation between A and X is summarized in Table I.

TABLE I
CONVERSION FROM BINARY CODE OF MULTIPLIER A TO 1-OUT-OF-N CODE OF X

multiplier A			code X		
	a_1	a_0	x_1	x_2	x_3
0	0	0	0	0	0
1	0	1	1	0	0
2	1	0	0	1	0
3	1	1	0	0	1

TABLE II
RELATION BETWEEN 1-OUT-OF-N CODE X AND MULTIPLICAND B FOR EACH BIT p_i OF PRODUCT $P = p_3 p_2 p_1 p_0$

code X (multiplier A)	product P			
	p_0	p_1	p_2	p_3
x_1 (1)	B=1,3	B=2,3		
x_2 (2)		B=1,3	B=2,3	
x_3 (3)	B=1,3	B=1,2	B=2	B=3

B. Control Signal Generator

Given a code $X(=x_1 \sim x_3)$ representing a multiplier A and a multiplicand B($=0 \sim 3$), each bit of the product $P(=p_0 \times 2^0 + p_1 \times 2^1 + p_2 \times 2^2 + p_3 \times 2^3)$ is obtained as shown in Table II. It can be verified from Table II that p_i ($i=0 \sim 3$) is set to logic "1" by the following combinations of values of B;

$$
\begin{aligned}
\text{(i)} \quad & B = \text{"1"} \quad or \quad \text{"3"} \\
\text{(ii)} \quad & B = \text{"2"} \\
\text{(iii)} \quad & B = \text{"2"} \quad or \quad \text{"3"} \\
\text{(iv)} \quad & B = \text{"1"} \quad or \quad \text{"2"} \\
\text{(v)} \quad & B = \text{"3"}
\end{aligned}
$$

Define a control signal $Y(y_0 \sim y_4)$ such that y_0 corresponds to combination (i), y_1 to (ii), y_2 to (iii), y_3 to (iv), y_4 to (v). Table III shows the relation between B and Y. We can see from this table that, for example, when multiplicand B is equal to "3", y_0, y_2, and y_4 of control signal Y are set to "1".

TABLE III
RELATION BETWEEN MULTIPLICAND B AND CONTROL SIGNAL Y

multiplicand B			control signal Y				
	b_1	b_0	y_0	y_1	y_2	y_3	y_4
1	0	1	1	0	0	1	0
2	1	0	0	1	1	1	0
3	1	1	1	0	1	0	1

TABLE IV

RELATION BETWEEN 1-OUT-OF-N CODE X AND CONTROL SIGNAL Y FOR EACH BIT p_i OF PRODUCT P

code X (multiplier A)	product P			
	p_0	p_1	p_2	p_3
x_1 (1)	y_0	y_2		
x_2 (2)		y_0	y_2	
x_3 (3)	y_0	y_3	y_1	y_4

C. Product Generator

Table IV indicates that each bit p_i of product P can be derived from a logical combination of x_i and y_j, which can be written as

$$p_0 = x_1 y_0 + x_3 y_0,$$
$$p_1 = x_1 y_2 + x_2 y_0 + x_3 y_3,$$
$$p_2 = x_2 y_2 + x_3 y_1,$$
$$p_3 = x_3 y_4.$$

Each p_i (i=0∼3) is of the sum-of-product form of x_i and y_j, which are independent of the bit widths of multiplier A and multiplicand B. Thus each operation in this block is expected to be very fast, since the critical path is composed only of an AND-gate and an OR-gate.

D. Constant Coefficient Multiplication

As described in Section I, we can assume that the coefficient data are given by multiplicand B, which can be fixed during the filtering operation, and hence control signal Y can be fixed as well. This means that before the filtering operation, it is possible to store in Control Signal Register the value of Y generated by Control Signal Generator. In this filtering operation, video data enter into the multiplier A, and then they are decoded through Multiplier Decoder. Thus the functional blocks of the multiplier can be divided into two; the *static* and *dynamic operational blocks*. Control Signal Generator and Control Signal Register are categorized in the static operational block, and Multiplier Decoder and Product Generator are in the dynamic operational block. This categorization is shown in Figs. 3 and 4.

E. FIR Filter

A basic FIR filter block diagram is shown in Fig. 5, where it should be pointed out that to realize the speed of 20-100MHz, a distinct multiplier is necessary at each tap, instead of using only one multiplier repeatedly at each tap. Apart from this diagram, the proposed FIR filter is outlined in Fig. 6, where D* denotes a pipeline register. Noting that there are only one Multiplier Decoder and
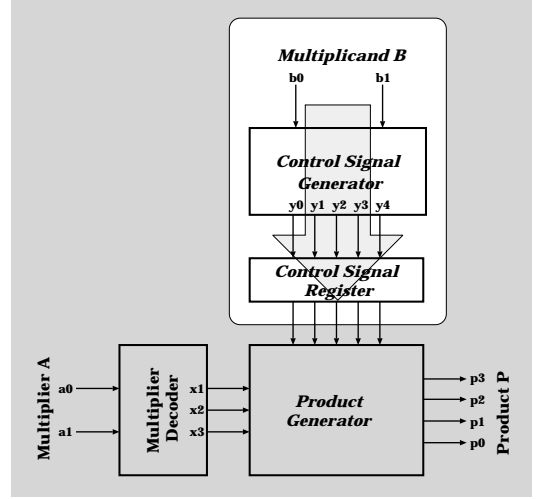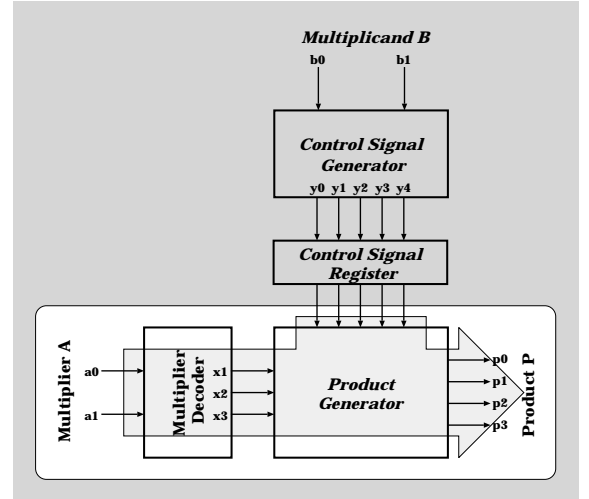

Fig. 3. Static operation block

Fig. 4. Dynamic operation block

only one Control Signal Generator, it can be easily verified that the integration of Product Generator in the minimum possible area is the most important issue for this filter implementation.

III. DESIGN METHODOLOGY

A. Multiplier Decoder

Since only one Multiplier Decoder is necessary for the FIR filter implementation, we have only to pay attention to its operation speed, but not to its circuit size. The function of this block is simple, with k inputs and 2^k-1 outputs. In the 8-bit case, the circuit is synthesized with
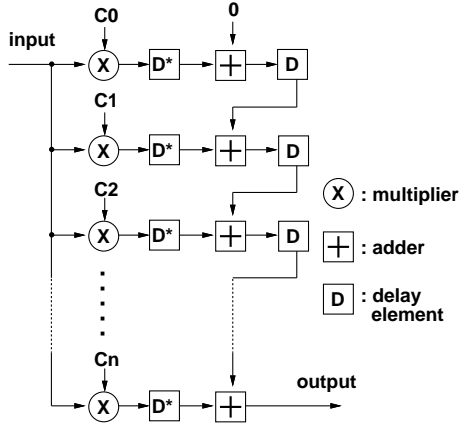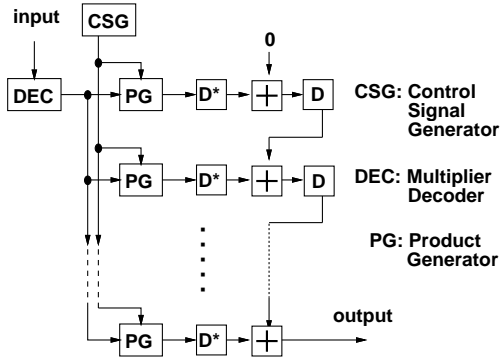
Fig. 5. Basic FIR filter block diagram

Fig. 6. FIR filter block diagram using proposed multipliers

	the number of control signal	transistor count
4×4-b	27	371
6×6-b	162	7,850
8×8-b	891	*
10×10-b	4,563	*
2×8-b	16	266
3×10-b	40	2,274
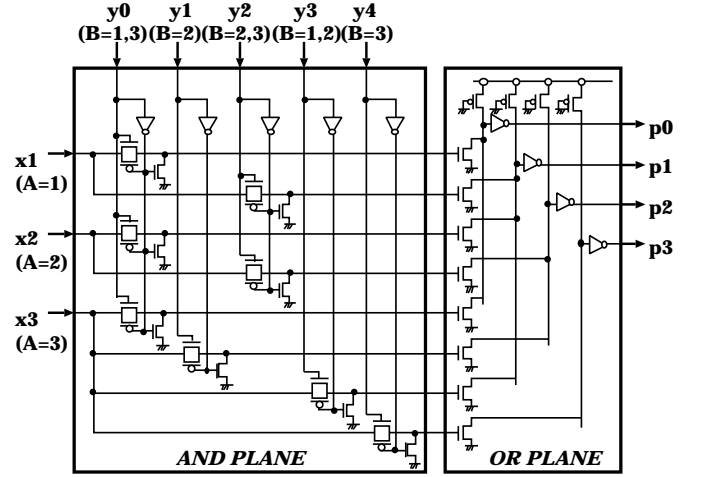
Fig. 7. Circuit configuration of Product Generator using TGs

the use of 619 gates by ASIC Synthesizer of COMPASS Design Navigator.

The layout data of Multiplier Decoder is generated by employing DAVINCI Macrocell Generator.

B. Control Signal Generator

In terms of implementing an FIR filter, only one Control Signal Generator is enough, since the control signal is generated before the filtering operation. As explained in Section II, given a bit p_i of product P it is essential to seek a set of all those multiplicands B that make p_i equal to logic "1" for an arbitrary multiplier A. A C-language program has been developed to generate a VHDL description of Control Signal Generator, so as to be synthesized by ASIC Synthesizer. As can be seen from the implementation result of Table V, in the case of the 8-bit by 8-bit and 10-bit by 10-bit multiplications, the numbers of control signals which are output from this block are 891 and 4563, respectively, but the synthesis of the both was failed because of hardware limitation.

From this result, the proposed idea for Control Signal Generator is not practical, and hence it is necessary to generate control signals of a smaller bit size.

C. Product Generator

Product Generator categorized in the dynamic operational block is expected to have a margin enough for the required speed, since the critical path is composed only of an AND-gate and an OR-gate. To achieve the integration in a small area, the design is to be carried out manually for this block by using transmission gates (TGs). The block diagram of this is exemplified in Fig. 7 for the case of the 2-bit by 2-bit multiplication. The logic function of each bit of product P is generated by a C-language program. As illustrated in this figure, N-ch transistors are added to avoid the high impedance output from the AND-Plane when the control signal y_i is "0". In the case of the 8-bit by 8-bit and 10-bit by 10-bit multiplications, the numbers of transistors are 16,142 and 82,882, respectively.

The layout of Product Generator has been designed manually by using SX-9000 VLSI Design System.

	transistor count		simulated critical path delay
	static operation block	dynamic operation block	
Fig. 8	3,670	3,599	5.7 nsec
basic	—	4,304	5.4 nsec

Fig. 8. Proposed FIR filter configuration

Fig. 9. Layout design of one tap block

IV. FIR FILTER CONFIGURATION AND PERFORMANCE EVALUATION

The operation speed of the proposed multiplier depends primarily on the dynamic operational block. We can apply plural multiply operations in a single operation cycle of 10-50nsec, since not only the critical path delay of Product Generator can be estimated to be small enough for the target speed, but also the automatic generation of Control Signal Generator is failed in Section III. The proposed FIR filter configuration shown in Fig. 8 is devised for employing the idea of executing plural operations in a single cycle. To achieve 10-bit by 10-bit multiplication, two Product Generators, which execute 2-bit by 10-bit and 3-bit by 10-bit multiplication, are equipped. In a single operation cycle, two Product Generators can be invoked two times respectively. At first, the lower 5-bit by 10-bit multiplication result is added to the output of the previous tap. Then the higher 5-bit by 10-bit multiplication result is accumulated to make up the complete output.

Fig. 9 shows the layout of one tap, which corresponds to circuit blocks enclosed by dotted line in Fig. 8 All blocks in this figure are categorized in the dynamic operation block, which have been manually designed by a 0.5μm
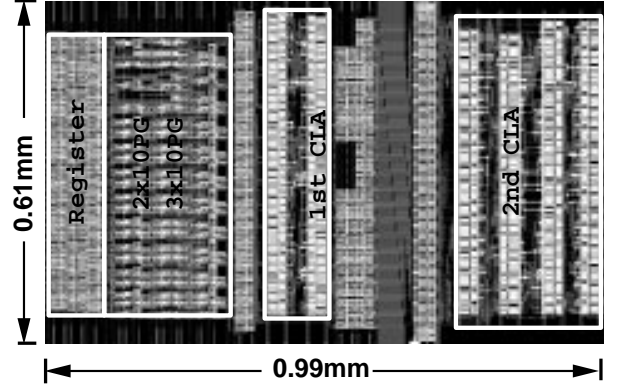
Process Technology	0.5μm double metal CMOS
Size	0.61mm × 0.99mm
Number of Transistors	3,599

double metal CMOS technology[1].

The transistor count and simulated critical path delay are summarized in Table VI, where 'basic' indicates the basic configuration of Fig. 5, which is composed of multipliers constructed by 2nd-order Booth's algorithm, Carry Save Adders, and Carry Look-Ahead Adders. In this basic configuration, a critical path exists in the multiplier.

On the other hand, in our model, a critical path exists in the Adder for generating the output of the tap. Although the simulation is carried out without considering the wiring delay, 100MHz operation can be sufficiently expected, since one tap has been integrated in such a small area of 0.99mm × 0.61mm. From Table VI, it is obvious that the transistor count can be reduced if more than 6 taps are integrated in a chip. Specifications of the implementation are summarized in Table VII.

---

[1] This chip has been fabricated in the chip fabrication program of VLSI Design and Education Center, University of Tokyo, with the collaboration by NTT Electronics Technology Corporation.

V. Conclusion

A novel multiplier architecture has been devised to be synthesized with the use of a high level synthesis tool, partly with the aid of the manual design for Product Generator. One FIR filter configurations have been attempted with the employment of the proposed multipliers.

As a result, we can see not only the reduction of the number of transistors when more than 6 taps are integrated, but also the achievement of the targetted operation speed of 100MHz.

Development is continuing further on the refinement of VLSI inplementation for a FIR filter integration and simulation with taking account of wiring delay.

References

[1] J. Mori, M. Nagamatsu, M. Hirano, S. Tanaka, M. Noda, Y. Toyoshima, K. Hashimoto, H. Hayashida, and K. Maeguchi, "A 10-NS 54×54-b parallel structure full array multiplier with 0.5μm CMOS technology," *IEEE J. Solid-State Circuits*, vol. 26, no. 4, pp. 600–605, Apr. 1991.

[2] M. Hatamian, and G. Cash, "A 7-MHz 8-bit $\times$ 8-bit parallel pipelined multiplier in 2.5μm CMOS," *IEEE J. Solid-State Circuits*, vol. SC-21, no. 4, pp. 505–513, Aug. 1986.

[3] M. Ercegovac, and T. Lang, "Fast multiplication without carry-propagate addition," *IEEE Trans. on Computer*, vol. 39, no. 11, pp. 1385–1390, Nov. 1990.

[4] A. D. Booth, "A signed binary multiplication technique," *Qt. J. Mech. Appl. Math*, vol. 4, part 2, 1951.

[5] C. S. Wallace, "A suggestion for fast multipliers," *IEEE Trans. Electronic Computers*, vol. EC-13, pp. 14–17, Feb. 1964.

[6] K. Okada, S. Morikawa, S. Takeuchi, and I. Shirakawa, "A design of high-performance multiplier for digital video transmission" in *Proc. ASP-DAC'95*, Chiba, Japan, pp. 429–434, Aug. 1995.