Efficient Bit-Level Systolic Arrays for QMF Banks
Jia-Wen Lin, Yung-Chang Chen, and Chin-Liang Wang
Institute of Electrical Engineering, National Tsing Hua University, Hsin-Chu, Taiwan, R.O.C.

ABSTRACT

In this paper, various systolic arrays are proposed for the application to quadrature mirror filter (QMF) banks. A word-level systolic array is firstly presented to realize QMF banks. It is subsequently refined to bit-level array with bit-parallel arithmetic via the well-known two-level pipelining techniques and is then converted to bit-serial form by using the bit-serial inner product array proposed by Wang et al. By applying the polyphase representation as well as fully utilizing the special relations among QMFs, aside from the memory cost, the whole filter bank can be constructed by using only about one half of the hardware expense of a prototype filter. In comparison with the direct realization using polyphase representation, the number of the systolic multiplier-accumulators (SMAs) required for our architecture is halved. Thus, both the chip area and transistor-count are reduced. As a result, with today's commercial CMOS technology, the whole filter bank can be implemented within a single-chip for various video applications.

1. INTRODUCTION

Since it was introduced, subband coding (SBC) is so far one of the most effective coding approaches for video applications. Fig. 1 shows a configuration for typical 1-D two-band subband coding systems. This coding approach decomposes the source signal into separable encoding bands. Because the human visual system is less sensitive to high-band errors and most energy is concentrated in low frequency bands, it is possible to encode these bands in various significance, thereby achieving coding gains and providing hierarchical video services. This makes it especially promising for the future broadband ISDN environment.

Since SBC needs filter banks to split the input signal into many subbands, aside from the coding mechanism, the filter bank is the kernel of this coding scheme. In order to avoid errors arising from the resampling and imperfect filtering processes, a number of filter banks possessing the perfect reconstruction (PR) property have been widely investigated. Among them, the QMF banks are the most famous. Since these filter banks usually deal with large amount of data, high-speed computing hardwares are necessary to meet the real time requirements. Recently, the development of VLSI technology has made high-speed, parallel processing of large volume of data practical and cost-effective. Therefore, it is an attractive approach for implementation of real time multirate systems using VLSI circuits. Since the introduction of systolic arrays by Kung and Leiserson, they have been widely used to design special-purpose signal processing systems. A systolic architecture possesses modularity, regularity, local connection, massive parallelism, linear-rate pipelinability, and is thus suitable for VLSI implementation and has a very high throughput rate.

Many systolic arrays have been proposed for high-speed digital filtering, of which only few are dedicated to subband filtering. It is not efficient to directly apply a general-purpose filtering array to subband coding systems, since they never utilize the special properties of such systems. Recently, Pestel et al. proposed a dedicated VLSI architecture for subband filtering by using the polyphase representations of QMF banks. In their architectures, however, the special relations among the coefficients of QMF banks are not fully utilized. In addition, the highest pipelinability is not achieved in their design. In this paper, we will focus on the design of systolic arrays for QMF banks.

2. QUADRATURE MIRROR FILTER (QMF) BANKS

The most popular and widely used technique for the analysis/synthesis filter banks of the subband coders is the well-known quadrature mirror filters (QMF's). By using QMF banks, the aliasing effect due to the imperfect filters and critical downsampling can be exactly eliminated, so an alias-free reconstruction of the input signal is possible. As well, the phase
distortion can be canceled. The amplitude distortion can not be exactly eliminated but can be minimized. Though QMF banks can not exactly achieve perfect reconstruction, it will be shown that the implementation of QMF banks is very efficient. Typically, the coefficients of QMF banks satisfy the following conditions: \(^1\)

\[
h_1(n) = (-1)^n h_0(n) \tag{1}
\]

\[
g_0(n) = h_0(n) \tag{2}
\]

\[
g_1(n) = -h_1(n) \tag{3}
\]

\[
h_0(n) = h_0(L-1-n), 0 \leq n \leq \frac{L}{2} - 1 \tag{4}
\]

It is important to note that, if 2-D QMF banks are separable (i.e., \(h_{ik}(n_1, n_2) = h_1(n_1) h_k(n_2)\)), they can be implemented by using the tree-structured configuration which comprises of several stages of 1-D QMF banks shown in Fig. 2 (which omits the synthesis part). This configuration leads to a simpler and more hardware-economical realization of the separable 2-D QMF banks, since it confines the filter design problems of 2-D QMF to 1-D's.

A more efficient representation scheme for QMF banks is to use the polyphase structure. Fig. 3 shows the polyphase representation of two-band 1-D QMF analysis/synthesis banks. The coefficients of filter partition 1, \(h_{p0}(n)\), comprise the odd indices of the prototype half-band lowpass filter, i.e.,

\[
h_{p0}(n) = h_0(2n), 0 \leq n \leq \frac{N}{2} - 1. \tag{5}
\]

In contrast, the coefficients of filter partition 2, \(h_{p1}(n)\), comprise the even indices of the prototype filter, and they are in the reverse order of partition 1's because of the linear phase property of the prototype filter. That is,

\[
h_{p1}(n) = h_0(2n + 1) = h_{p0}\left(\frac{N}{2} - 1 - n\right), 0 \leq n \leq \frac{N}{2} - 1. \tag{6}
\]

Note that, this structure takes only about one half of the hardware expense of the original filter banks.

### 3. A WORD-LEVEL SYSTOLIC ARRAY FOR QMF BANKS

According to the polyphase structure shown in Fig. 3, we get

\[
y_{LP} = \sum_{m=0}^{\frac{N}{2}-1} a_0(m) x_{p0}(n-m) + \sum_{m=0}^{\frac{N}{2}-1} a_1(m) x_{p1}(n-m), \tag{7a}
\]

\[
y_{HP} = \sum_{m=0}^{\frac{N}{2}-1} a_0(m) x_{p0}(n-m) - \sum_{m=0}^{\frac{N}{2}-1} a_1(m) x_{p1}(n-m), \tag{7b}
\]

\(a_0(m) = h_{p0}(m); a_1(m) = h_{p1}(m)\) for \(m = 0, 1, \ldots, \frac{N}{2} - 1\). Substituting Eq. (6) into Eqs. (7a) and (7b), the above equations can be rewritten as:

\[
y_{LP} = \sum_{m=0}^{\frac{N}{2}-1} a_0(m) x_{p0}(n-m) + \sum_{m=0}^{\frac{N}{2}-1} a_0\left(\frac{N}{2} - 1 - m\right) x_{p1}(n-m), \tag{8a}
\]
\[ y_{hr} = \sum_{m=0}^{N-1} a_0 \left( m \right) x_{p1} \left( n - m \right) - \sum_{m=0}^{N-1} a_0 \left( \frac{N}{2} - 1 - m \right) x_{p2} \left( n - m \right). \]  

(8b)

It can be seen that the overall operation of this system is to perform two convolution operations respectively. The two convolution operations are to compute the two inner products of the two input data vectors, namely, \( x_{p0} = [x_{n-1} x_{n-2} \cdots x_{n-N+2}] \) and \( x_{p1} = [x_{n-1} x_{n-3} \cdots x_{n-N+1}] \), with their corresponding coefficient vectors \( a_0 = [a_0 a_1 \cdots a_{N-2}] \) and \( a_1 = [a_{N-1} a_{N-2} \cdots a_0] \). In short, the two corresponding inner products are \( x_{p0}a_0^t \) and \( x_{p1}a_1^t \), respectively. Consequently, the low-pass and high-pass filtered versions of input data stream \( x \) via a 1-D two-band QMF analysis bank are \( y_{lp} = x_{p0}a_0^t + x_{p1}a_1^t \) and \( y_{hp} = x_{p0}a_0^t - x_{p1}a_1^t \), respectively. By using the fact that the coefficients of the filter partition 2 are in the reverse order of the filter partition 1’s, they can be rewritten as

\[ y_{lp} = \tilde{x}_{p0}a^t + \tilde{x}_{p1}a^t, \]

(9)

\[ y_{hp} = \tilde{x}_{p0}a^t - \tilde{x}_{p1}a^t, \]

(10)

where \( \tilde{x}_{p0} = x_{p0} \) and \( \tilde{x}_{p1} \) stands for the vector with entries in the reverse order of those in \( x_{p1} \), i.e., \( \tilde{x}_{p1} = [x_{n-N+1} x_{n-N+3} \cdots x_1] \). Accordingly, a word-level systolic architecture for 1-D two-band QMF banks is derived in Fig. 4.

The principle idea of the architecture shown in Fig. 4 is to reduce the two coefficient vectors to only one via a tricky arrangement of the input data flow as shown. Such an arrangement halves the requirement for the storage of the coefficients. Referring to Fig. 4, the input data are first demultiplexed into two polyphase components. One component is fed into the array of multiplexer cells (for notation simplicity, we refer to it as “the MUX array”) from top to bottom via a linear chain of one-word shift registers. The other enters the MUX array from bottom to top. With the above arrangement, the two input data streams are interleaved word-by-word via the MUX array. Note that, the clock rate for the shift registers is one half of the overall clock rate. Thus, the two data words entering each multiplexer cell can be selected interleavingly without any loss. This also reduces the overall power consumption, especially for 2-D applications. In order to perform pipelining operations properly, appropriate number of delays are inserted between horizontally adjacent main array cell and multiplexer cell. After emerging from the array of delays, the interleaved data words enter the main array in a skew fashion. The linear chain of the main array cells, which performs the inner product operation, is composed of \( \frac{N}{2} \) systolic multiplier-accumulator (SMA) cells. It is used to compute the two inner products, namely, \( \tilde{x}_{p0}a^t \) and \( \tilde{x}_{p1}a^t \). Then, the two split-band outputs are obtained at the bottom cell by evaluating the sum and the difference of the two inner products mentioned above.

Separable 2-D QMF banks can be realized by using the tree structure proposed by Pestel et al.\(^4\) shown in Fig. 5. Each block of filter banks can be replaced with our architectures to reduce the hardware expense.

### 4. BIT-LEVEL SYSTOLIC ARRAYS FOR QMF BANKS

In this section we offer two bit-level implementations of the aforementioned word-level architecture. These two bit-level arrays are with bit-parallel and bit-serial arithmetic, respectively. They are obtained by employing the concept of two-level pipelining\(^1\). That is, each component of the word-level architecture is replaced with a proper bit-level pipelined circuit. From the viewpoint of data processing, bit-level systolic arrays with bit-serial arithmetic can be regarded as the time-sharing versions of those with bit-parallel arithmetic. In comparison with bit-parallel ones, bit-serial systolic arrays are much more efficient in hardware and with less I/O bandwidth but can only process data of lower bit-rate. Therefore, there are always some trade-off’s between the selections of bit-parallel and bit-serial arithmetic when hardware complexity and throughput are both taken into consideration. Actually, they are application-dependent.

As discussed above, the fundamental work for a QMF analysis bank is the computation of the two inner products \{\( y_m, m = 0, 1 \)\}, of two \( \frac{N}{2} \)-tuple input vectors \( \{\tilde{x}_{pm}\} = \{\tilde{x}_{pm,0} \tilde{x}_{pm,1} \cdots \tilde{x}_{pm,\frac{N}{2}-1}\} \) and \( m = 0, 1 \)}, and the \( \frac{N}{2} \)-tuple coefficient vector \( a = [a_0 a_1 \cdots a_{N-1}] \), where

\[ y_m = a^t\tilde{x}_{pm} = a_0\tilde{x}_{pm,0} + a_1\tilde{x}_{pm,1} + \cdots + a_{\frac{N}{2}-1}\tilde{x}_{pm,\frac{N}{2}-1}. \]  

(11)
\[ a_i = (a_i^{W-1} a_i^{W-2} \ldots a_i^1 a_i^0) , \]  

\[ \bar{x}_{pm,i} = (\bar{x}_{pm,i}^{W-1} \bar{x}_{pm,i}^{W-2} \ldots \bar{x}_{pm,i}^1 \bar{x}_{pm,i}^0) , \]  

\[ y_m = (y_m^{W+G-1} y_m^{W+G-2} \ldots y_m^1 y_m^0) , \]  

\[ G = \left\lfloor \log_2 \frac{N}{2} \right\rfloor . \]

The number \( G \) is the wordlength growth required for inner product computations. For notation simplicity and without loss of generality, we have assumed that each input data word and each coefficient word are both of \( W \) bits.

### 4.1 Bit-parallel systolic array

Fig. 6 depicts the bit-parallel SMA with two's complement arithmetic. It is the improved version of the design proposed by McCanny and McWhirter\(^{13}\). The circuit shown in Fig. 6 is composed of \( W^2 \) TYPE-P1 cells and \( \frac{1}{2} W (W+1) + GW \) TYPE-P2 cells. The TYPE-P1 cell performs the bit-level multiply-and-add operation, while the TYPE-P2 cell performs only bit-level addition operation. The logic functions of these two types of cells are depicted in Fig. 7. It takes \( W+i \) cycles for the bit \( s'(n) \) of the result \( s(n)=a(n)x(n)+s'(n) \) to appear at the output after \( a_0(n) \) is fed into the system. It is noteworthy that, in order to perform two's complement arithmetic properly, each TYPE-P1 cell comprises a built-in control bit \( d \) to manipulate the polarity of each bit of the operand \( x(n) \) as shown in Fig. 7. Performing multiplications with two's complement arithmetic, instead of the sign-extension approach used in some literatures\(^{5,9}\), we can reduce the hardware expense by using the correction-term-compensation approach based on the two's complement arithmetic derived from Baugh-Wooly algorithm\(^{14}\). The correction term, can be properly included in the arithmetic results by adding it to the operand \( s'(n) \). That is, we replace the operand \( s'(n) \) with \( s'(n)+ct \), where \( ct \) stands for the correction term. This approach is easily achieved by sending \( ct \) as \( s'(n) \) at the first SMA in the filter arrays, since, in general, the accumulation term \( s'(n) \) in the first SMA of a filtering array is initially zero. In comparison with McCanny's array, this improved array simplifies \( \frac{1}{2} W (W+1) + GW \) TYPE-P1 cells to TYPE-P2 cells, thus reduces the hardware expense, since the TYPE-P2 cell is with less complexity in hardware.

By employing this SMA, a bit-level systolic array with bit-parallel arithmetic for QMF banks is obtained as shown in Fig. 8. In this array, the coefficient vector \( a = [a_0 \ a_1 \ \cdots \ a_{2^N-1}] \) is stored in the latches, denoted by the solid circles, in a sequential order as shown. As discussed above, the input data are first decomposed into two polyphase components via a demultiplexer. Subsequently, these two polyphase components flow along two opposite directions and enter the main array via a linear chain of bit-level multiplexer cells. The arrangement of the shift registers, which are above the multiplexer cells, is organized according to the word-level realization. The main array is composed of \( \frac{N}{2} \) bit-parallel bit-level SMAs, where \( N \) is the filter length. It acts as a dual-input inner product array with a single set of coefficients. The two resulting inner products enter the linear chain of TYPE-P3 cells interleavingly. As mentioned previously, the lowpass-filtered output is obtained by summing the two inner products, while the highpass-filtered version is their difference. The linear chain of TYPE-P3 cells is used to evaluate the sum and the difference of the two inner products, simultaneously. As shown in Fig. 9, each TYPE-P3 cell consists of two full adders and several one-bit latches. In this linear chain, the overall operation of the upper full adders is to evaluate the sum of two consecutive input bit-parallel data. A "0" is sent to the right input line above the top cell as the initial carry for the upper full adders. In contrast, the overall operation of the lower full adders is to evaluate their difference. Thus, one of the two data bits entering each TYPE-P3 cell is inverted, then sent to the lower full adder. A "1" is fed as the initial carry for the lower full adders. The purpose of such arrangement is to perform subtraction with two's complement arithmetic. It is noteworthy that, to accommodate the wordlength growth due to the two arithmetic operations, the one-bit sign extension is appended at the bottom of the leftmost array as shown. The output results are obtained at a rate of two outputs per two cycles. Since the resulting data are significant only at one out of two cycles, one must select the timing for grabbing data carefully.
4.2 Bit-serial systolic array

A bit-serial systolic array which can perform two distinct inner product computations was proposed by Wang et al.\textsuperscript{2}. Based on this inner product array (IPA), we can construct our bit-serial realization for QMF banks. For this IPA, an improvement is made by appending additional $W+1$ ones at the rightmost of the linear chains of TYPE-S2 cells to obtain a regular skewed output format, thereby facilitating the succeeding processing. Recall that, since the coefficients of the filter partitions 2 are in the reverse order of the filter partition 1's, we can use only one of the even indices of the half-band prototype filter’s coefficients to perform lowpass and highpass filtering processes, simultaneously. These coefficients are distributed over the internal latches of the TYPE-Pi cells as shown.

Referring to Fig. 10, the input data are first demultiplexed into two polyphase components. One of the two components enters the first row in a bit-serial format and moves from top to bottom via a number of delays, while the other enters the $(\frac{N}{2} - 1)$th row and move in the opposite direction. Before entering the first row and the $(\frac{N}{2} - 1)$th row, the two input data bits are repeated twice and then interleaved bit by bit via a linear chain of multiplexer cells (TYPE-S3 cells) controlled by a MS stream. In order to prevent the increase of the amount of latches due to the repetition of the input data bits, the clock rate for both sending the input data bits and shifting the data bits in the two rightmost columns of delays (shift registers) is halved. To achieve this purpose, The CTRL signal can be used as the half-speed clock signal. The number of the delays are appropriately selected to ensure that the data bits enter each row in a skew parallelogram fashion for pipelining operations. They are $k_1 = k_3 = W$, $k_2 = W + 1$, $k_4 = W - 1$. The linear chain of the TYPE-S4 cells is used to evaluate the sum and difference of the two inner products. The logic functions of TYPE-S3 and TYPE-S4 cells are shown in Fig. 12. Each TYPE-S4 cell comprises a one-bit internal register and operates under the control of the MS stream. When $MS = 0$, the linear chain of TYPE-S4 cells stores the output data word of $2W+G$ bits emerging from the accumulator cells. When $MS = 1$, it evaluates the sums of the stored data word with the entering data word and its two’s complement (i.e., the one’s complement of the entering data word plus the value “1” at the carry input of the rightmost TYPE-S4 cell), simultaneously. Thus, two filtered outputs are obtained. Consequently, the resulting lowpass and highpass outputs come out from the bottom of this array in a bit parallel fashion at a rate of two outputs per $2 \times W$ cycles. To further reduce the pincount, a parallel to serial circuit\textsuperscript{2} can be used.

5. CONCLUSIONS

In this paper, we have proposed two bit-level systolic arrays with bit-parallel and bit-serial arithmetic for QMF banks. The former allows faster data rate, while the latter takes less hardware expense and pincount. Thus, they can be applied to various video applications depending on the trade-off’s of data rate, chip area, and pincount required. By applying the polyphase representation as well as fully utilizing the special relations among QMFs’, aside from the memory cost, the whole filter bank can be constructed by using only about one half of the hardware expense of a prototype filter. In comparison with the direct realization using polyphase representation, the number of SMAs required for our architecture is halved. Table. 1 illustrates the brief comparisons between the proposed realization and the direct realization\textsuperscript{10}. It shows that, aside from the memory cost, the hardware expense of our architecture is about one half of Pestel’s. For 2-D applications, by using the tree structure shown in Fig. 5, the data rate for SMAs is one half of the original’s for our architecture, and is one quarter for Pestel’s one. Since our designs achieve bit-level pipelinability, with today’s commercial 1.2μm CMOS technology allowing data rate of more than 30 MHz, they can meet the speed requirements of most video sources, even for HDTV applications. As a result, a single-chip realization for the whole filter bank is achievable.

REFERENCES


Fig. 1. A configuration for typical 1-D two-band subband coding systems

Fig. 2. A tree-structured configuration of separable 2-D QMF banks

Fig. 3. Polyphase structures of 1-D two-band QMF banks (a) analysis filter bank, (b) synthesis filter bank
Fig. 4. A word-level systolic array for QMF banks

Fig. 5. The tree-structured realization of 2-D separable QMF proposed by Pestel et al.
Fig. 6. An improved bit-parallel systolic multiplier-accumulator

TYPE-P1 cell : TYPE-P2 cell :

symbol :

symbol :

Fig. 7. The logical functions of TYPE-P1 and TYPE-P2 cells
Fig. 8. A bit-parallel bit-level systolic array for QMF banks, where $N = 6$

Fig. 9. The logic function of TYPE-P3 cell
Fig. 10. A bit-serial systolic array for QMF banks with \( N = 8 \)

\[
\begin{align*}
X' & \leftarrow X \\
Y' & \leftarrow Y \oplus C \oplus (aX)
\end{align*}
\]

\[
\begin{align*}
C' & \leftarrow YC + Y(aX) + C(aX)
\end{align*}
\]

Fig. 11. The logic functions of TYPE-S1 and TYPE-S2 cells
Fig. 12. The logic functions of TYPE-S3 and TYPE-S4 cells

Table 1. Comparison table of our proposed array and Pestel's for 1-D QMF banks

<table>
<thead>
<tr>
<th></th>
<th>Ours</th>
<th>Pestel's</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of SMAs</td>
<td>N/2</td>
<td>N</td>
</tr>
<tr>
<td>clock rate for SMA</td>
<td>input data rate</td>
<td>1/2 input data rate</td>
</tr>
<tr>
<td>clock rate for shift registers</td>
<td>1/2 input data rate</td>
<td>1/2 input data rate</td>
</tr>
<tr>
<td>throughput</td>
<td>2 outputs per two input clock cycles</td>
<td>2 outputs per two input clock cycles</td>
</tr>
</tbody>
</table>