Efficient Implementation of Parallel Correlators for Code Acquisition in DS/CDMA Systems

Seung Hyuk Ahn*, Joon Tae Kim** and Yong Hoon Lee*

*Dept. of Electrical Engineering Korea Advanced Institute of Science and Technology

**Digital TV Research Lab., LG Electronics Co., Ltd.

E-MAIL: yohlee@eekaist.kaist.ac.kr

Abstract

In this paper, we propose a method for reducing the computational complexity of a bank of passive/active correlators, which are used for rapid code acquisition of direct-sequence code division multiple access (DS-CDMA) systems. The proposed algorithm extracts common sub-expressions (CSEs) from the correlation coefficients and reduces the number of operations by sharing the additions for evaluating the CSEs. It is shown that the proposed algorithm is applied to the implementation of a bank of passive/active correlators and the computational complexity is substantially reduced.

I. Introduction

In direct sequence spread-spectrum (DS-SS) systems, the goal of code acquisition is to achieve a coarse time alignment between the received pseudo-noise (PN) code and the locally generated code to an accuracy of a fraction of one PN sequence chip. A popular approach to code acquisition is the *serial* search techniques [1]-[3] which correlate the received and locally generated code sequences and test the synchronization based on either the crossing of a threshold or the maximum correlation. These techniques can be thought of as a serial realization of a maximum-likelihood (ML) search method. Code acquisition may be achieved using *parallel* search techniques that simultaneously employ multiple correlators [4]-[6]. The parallel methods reduce the mean acquisition time at the expense of additional computation. In [6], it is shown that computational load of the parallel acquisition can be decreased when the number of correlators, say K, is equal to the correlation length, L, and L is equal to a period of the PN sequences.

In this paper, we propose a design method that can reduce the complexity of parallel correlators in practical situations where $K<<L$ and L is an arbitrary positive integer. The proposed method is based on the algorithm in [7]-[8], which is developed for reducing the complexity of digital filters, and can be applicable to both passive and active correlators.

The organization of this paper is as follows. In Section II, a method for reducing computational complexity of the parallel passive correlators is proposed. We extend this method to the case of parallel active correlators in Section III and conclusions are drawn in Section IV.

II. Implementation of a bank of passive correlators

Suppose that K correlators with binary coefficients are employed. An efficient structure for realizing these correlators is derived by extracting and sharing *common sub expressions* (CSEs) from the correlation coefficients. For example, consider two correlators with coefficients [1 1 −1 −1] and [1 −1 1 −1]. If we denote the outputs of the correlators by r_1(n) and r_2(n), then $r_1(n)=x(n)+x(n-1)-x(n-2)-x(n-3)=\{x(n)-x(n-3)\}+\{x(n-1)-x(n-2)\}$ and $r_2(n)=x(n)-x(n-1)+x(n-2)-x(n-3)=\{x(n)-x(n-3)\}-\{x(n-1)-x(n-2)\}$ where $x(n)$ is the input sequence. These expressions have $\{x(n)-x(n-3)\}$ and $\{x(n-1)-x(n-2)\}$ in common and thereby 2 additions can be reduced if the CSEs are shared. When K correlators are used, the outputs may be expressed as $\mathbf{r}(n)=\mathbf{C}\mathbf{x}(n)$, where $\mathbf{r}(n)=[r_1(n)\ r_2(n)\ \ldots\ r_K(n)]^T$ and $\mathbf{x}(n)=[x(n)\ x(n-1)\ \ldots\ x(n-L+1)]^T$; the PN code matrix $\mathbf{C}=[\mathbf{c}_1\ \mathbf{c}_2\ \ldots\ \mathbf{c}_L]$ is a matrix K by L and $\mathbf{c}_i$ is a K-dimensional column vector. The PN sequence of the j-th correlator is the j-th row of matrix $\mathbf{C}$. Referring to Fig.1, which illustrates an example with K=4 and L=5, the procedure for extracting CSEs is stated as follows.

Step 1: Given $\mathbf{C}$, we find out common columns and group them. Here two columns $\mathbf{c}_i$ and $\mathbf{c}_j$ are said to be common if $\mathbf{c}_i=\pm\mathbf{c}_j$. Then CSEs are extracted by examining the common columns. In Fig. 1(a), $\mathbf{c}_1$ and $\mathbf{c}_2$ are common, and we can see that the term $(x(n)+x(n-1))$ is a CSE for all correlators.

Step 2: The matrix $\mathbf{C}$ is partitioned into two submatrices $\mathbf{C}_{2,0}$ and $\mathbf{C}_{2,1}$, which contain the upper and the lower $K/2$ rows of $\mathbf{C}$, respectively. If $\mathbf{C}$ has odd number of rows, each submatrices will contain the upper $(K+1)/2$ rows and the lower $(K-1)/2$ rows. Then for each submatrix, common columns are grouped and CSEs are extracted. In this case, the CSEs from a submatrix are valid only

This work was performed as a part of ASIC Development Project supported by MOCIE, MOST and MIC of Korea.

0-7803-5471-0/99/$10.00©1999 IEEE

for those correlators associated with the submatrix. In Fig. 1(b), the first four columns of $\mathbf{C}_{2,0}$ are common and $\{(x(n)+x(n-1))-x(n-2)+x(n-3)\}$ is a CSE of the two correlators $r_1(n)$ and $r_2(n)$. Similarly, from $\mathbf{C}_{2,1}$ we obtain $\{-(x(n)+x(n-1))+x(n-3)\}$ and $\{x(n-2)-x(n-4)\}$ as CSEs for evaluating $r_3(n)$ and $r_4(n)$.

Step 3: Step 2 is repeated for each submatrix. Matrix partitioning and CSE extraction is repeated until no further partitioning is possible.

In Step 1, common columns exist if $L>2^{K-1}$. For a fixed L, the number of common columns increases as K decreases. The required number of additions for implementing a bank of passive correlators can be reduced by sharing CSEs. As an example, consider Fig. 2 which shows the implementation of the four correlators in Fig.1. The conventional direct implementation in Fig. 2(a) requires 16 additions. By sharing the CSE obtained from Fig. 1(a), the required number of additions reduces to 13 as shown in Fig. 2(b); use of the CSEs from Fig. 1(b) decreases the number of additions to 9 (Fig. 2(c)). The *expected* number of additions required by the proposed algorithm can be obtained as follows. Let $\mathbf{\Omega}$ denote the set of all possible binary (±1) vectors of dimension K whose first element is 1. Note that the number of elements in $\mathbf{\Omega}$ as 2^{K-1}. Suppose that the columns of the PN code matrix $\mathbf{c}_j$, $j=1,\ldots,L$, are selected from $\mathbf{\Omega}$ through L independent trials, and that both K and L are powers of two satisfying $L\geq 2^{K-1}$. Then the expected number of CSEs in Step 1 of the algorithm is 2^{K-1} and the expected number of terms in each CSE is $L/2^{K-1}$. If we denote the expected number of additions for evaluating the CSEs in Step j by N_j, then N_1 is given by $N_1=2^{K-1}/(L/2^{K-1}-1)$. For Step 2, the expected number of CSEs for each submatrix is $2^{K/2-1}$ and the expected number of terms in each CSE is $L/2^{K/2-1}$. Assuming that the CSEs of Step 1 are already evaluated, the expected number of additions for calculating a CSE in Step 2 is $(L/2^{K/2-1})/(L/2^{K-1})-1=2^{K-1}/2^{K/2-1}-1$. Therefore, considering two submatrices in Step 2, the number of required additions is given by $N_2=2^{K/2-1}\cdot(2^{K-1}/2^{K/2-1}-1)\cdot 2$. In this manner, we get the following total number of additions:

$$\sum_{j=1}^{1+\log_2 K} N_j = 2^{K-1}\cdot\left(L/2^{K-1}-1\right)+\sum_{j=2}^{1+\log_2 K}\left\{2^{K/2^{j-1}-1}\cdot\left(2^{K/2^{j-1}}-1\right)\cdot 2^{j-1}\right\}. \quad (1)$$

Table 1 lists the total numbers obtained from (1) for various (K, L) values.

To examine both the performance of the proposed algorithm and the validity of the expression in (1), we applied the algorithm to the design of parallel passive correlators evaluating partial correlation for IS-95 CDMA mobile communication systems [9]. Table 2 lists the number of required additions for some values of (K, L), when a portion of the short PN sequence for IS-95 is used. It is seen that we can substantially reduce the complexity by using the proposed method. Comparing the columns for K=1 and K=2 indicates that two correlators can be implemented by increasing only one addition. Similarly, the proposed method increases only a few additions when K=3 and K=4. Comparing Tables 1 and 2, we can see that the analytical and the experimental results match well.

Although the proposed algorithm can dramatically reduce the required number of additions, it leads to a hardware architecture which is not modular. As expected from Fig. 2(c), the connection between adders and registers becomes very complex for larger values of L. Therefore, when L is large, it is recommended to perform column-wise partitioning of the K-by-L PN code matrix $\mathbf{C}$ before applying the proposed algorithm.

III. Implementation of a bank of active correlators

Fig.3 illustrates an active correlator for code acquisition. Each input sample $x(j)$ is multiplied with the code value $c(j)$ and accumulated for L samples where L is the correlation length. The result $r(n)=\sum_{j=(n-1)L}^{nL-1} x(j)c(j)$, which is a partial correlation value, is produced with rate $1/LT_c$ where T_c is the chip period. The accumulator is reset after each LT_c period. The active correlator evaluates the partial correlation in a blockwise manner, employing different code sequences for different blocks. Therefore, the output vector of a bank of correlators is written as $\mathbf{r}(n)=\mathbf{C}(n)\mathbf{x}(n)$ where $n=kL$ (k=1,2,...) and $\mathbf{C}(n)$ is time-varying. In this case, the proposed algorithm for realizing passive correlators cannot be directly applied, because the PN sequence matrix $\mathbf{C}(n)$ is time varying. Our approach to efficient implementation of parallel active correlators is based on the observation that the matrix $\mathbf{C}(n)$, having K rows and L columns, can be decomposed into

$$\mathbf{C}(n)=\mathbf{C}_F\mathbf{C}_V(n) \quad (2)$$

where $\mathbf{C}_F$ is a fixed matrix with dimension (K, 2^{K-1}) and $\mathbf{C}_V(n)$ is a time-varying matrix with dimension (2^{K-1}, L). The columns of $\mathbf{C}_F$ are composed of all possible binary vectors of dimension K whose first element is 1 (i.e., the columns of $\mathbf{C}_F$ constitutes $\mathbf{\Omega}$). Let $c_V(n,i,j)$ denotes the (i,j)-th element of $\mathbf{C}_V(n)$, where $1\leq i\leq 2^{K-1}$ and $1\leq j\leq L$. These values are determined as follows: $c_V(n,i,j)=1$ if the j-th column of $\mathbf{C}_F$ is equal to the i-th column of $\mathbf{C}(n)$; $c_V(n,i,j)=-1$ if the j-th column of $\mathbf{C}_F$ multiplied by -1 is equal to the i-th column of $\mathbf{C}(n)$; $c_V(n,i,j)=0$, otherwise. This indicates that each column of $\mathbf{C}_V(n)$ has only one nonzero element which is either 1 or -1. In essence, $\mathbf{C}_V(n)$ selects proper K dimensional vectors from the columns of $\mathbf{C}_F$ to form the original matrix $\mathbf{C}(n)$. For example,

$$\mathbf{C}(n)=\begin{bmatrix}1&1&-1&1&-1\\1&1&-1&1&1\\-1&-1&1&1&-1\\-1&-1&-1&1&1\end{bmatrix}=\begin{bmatrix}1&1&1&1&1&1&1&1&1&1\\1&1&1&1&-1&-1&-1&-1&1&1\\1&1&-1&-1&1&1&-1&-1&0&0\\1&-1&1&-1&1&-1&1&-1&0&0\end{bmatrix}\begin{bmatrix}0&0&0&1&0\\0&0&0&0&0\\0&0&-1&0&0\\1&1&0&0&0\\0&0&0&0&0\\0&0&0&0&-1\\0&0&0&0&0\\0&0&0&0&0\end{bmatrix}=\mathbf{C}_F\mathbf{C}_V(n). \quad (3)$$

The output correlation vector may be rewritten as

$$\mathbf{r}(n)=\mathbf{C}_F\mathbf{C}_V(n)\mathbf{x}(n)=\mathbf{C}_F\mathbf{y}(n). \quad (4)$$

Therefore, once $\mathbf{y}(n)$ is evaluated, $\mathbf{r}(n)$ can be efficiently obtained following the procedure developed for passive correlators.

Now we explain the process of evaluating $\mathbf{y}(n)$ with the aid of the example in (5), which multiplies the input $\mathbf{x}(n)$ with $\mathbf{C}_V(n)$ in (3).

$$\mathbf{y}(n)=\begin{bmatrix} y_1(n)\\ y_2(n)\\ y_3(n)\\ y_4(n)\\ y_5(n)\\ y_6(n)\\ y_7(n)\\ y_8(n)\end{bmatrix}=\begin{bmatrix} 0&0&0&1&0\\ 0&0&0&0&0\\ 0&0&-1&0&0\\ 1&1&0&0&0\\ 0&0&0&0&0\\ 0&0&0&0&-1\\ 0&0&0&0&0\\ 0&0&0&0&0\end{bmatrix}\begin{bmatrix} x(0)\\ x(1)\\ x(2)\\ x(3)\\ x(4)\end{bmatrix}=\begin{bmatrix} x(3)\\ 0\\ -x(2)\\ x(0)+x(1)\\ 0\\ x(4)\\ 0\\ 0\end{bmatrix} \tag{5}$$

Suppose that we have 8 accumulators, denoted by $\{A_i, 1\le i\le 8\}$ for evaluating $\{y_i(n)\}$ where $\mathbf{y}(n)=[y_1(n),..., y_8(n)]^T$. When $x(0)$ is inputted, the PN code generators produce the first column of $\mathbf{C}(n)$ of (3). This vector is equal to the 4th column of $\mathbf{C}_F$ and thus the 4th element of the first column of $\mathbf{C}_V(n)$ becomes 1. This indicates that $x(0)$ is assigned to A_4. When the second input $x(1)$ is given, the PN code generators produce the second column of $\mathbf{C}(n)$ and this happens to be the 4th column of $\mathbf{C}_F$ as well. So $x(1)$ is also assigned to A_4. In this manner, each input value is assigned to one of $\{A_i\}$ depending on the code vector generated by the PN code generators. If the code vector is a sign-reversed version of the i-th column of $\mathbf{C}_F$, then the corresponding input is subtracted from the i-th accumulation: $x(n)$ is subtracted whenever the first element of the code vector is -1, since the first row of $\mathbf{C}_F$ consists of all 1s.

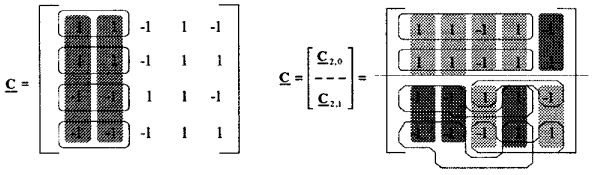
Fig. 3 illustrates the proposed structure for implementing four active correlators. The code generators produce the code vector $[c_1, c_2, c_3, c_4]$. The input $x(n)$ is first multiplied with c_1: it is subtracted whenever c_1=-1. To produce the sign reversed version of $[c_2, c_3, c_4]$ whenever c_1=-1, c_1 is also multiplied with $[c_2, c_3, c_4]$. The multiplexer (MUX) assigns $c_1x(n)$ to one of the accumulators depending on the vector $c_1\cdot[c_2\ c_3\ c_4]$. $\mathbf{y}(n)$ is obtained after accumulating L inputs.

The proposed structure for calculating $\mathbf{y}(n)$ requires 2^{K-1} accumulators and a multiplexer; thus its hardware implementation is more complex than the direct implementation of K active correlators. The number of additions for evaluating the outputs of a bank of active correlators, however, can be substantially reduced by using the proposed structure. Note that the L-by-1 vector $\mathbf{y}(n)$ is computed through L additions. This is because each input is assigned to only one accumulator. The number of additions for computing $\mathbf{C}_F\mathbf{y}(n)$ in (4) can be obtained as in the case of (1). The result is $\sum_{j=2}^{1+\log_2 K}\left\{2^{K/2^{j-1}-1}\cdot\left(2^{K/2^{j-1}}-1\right)\cdot 2^{j-1}\right\}$. It should be pointed out that this number is not an expected value but a fixed value. This holds since $\mathbf{C}_F$ is a fixed matrix regardless of the code sequences. The total number of additions N_t is given by

$$N_t = L+\sum_{j=2}^{1+\log_2 K}\left\{2^{K/2^{j-1}-1}\cdot\left(2^{K/2^{j-1}}-1\right)\cdot 2^{j-1}\right\} \tag{6}$$

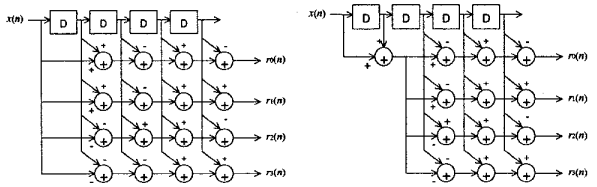
Table 3 lists N_t values for various N and K. The results indicate that the proposed method reduces the required additions when L is considerably larger than K.

The hardware architecture in Fig.4, which requires 2^{K-1} accumulators, and the results in Table 3 suggest the use of the proposed method only when K is reasonably small. When K is large, row-wise partitioning of the K-by-L code matrix $\mathbf{C}(n)$ is recommended.

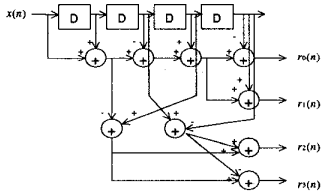


(a) Step 1 (b) Step 2

Fig. 1. Finding common columns from the correlation coefficient matrix.



(a) (b)



(c)

Fig.2 . Structures for implementing the four correlators in Fig. 1.
(a) Direct implementation
(b) Implementation after sharing the CSE from Fig. 1(a)
(c) Implementation after sharing the CSEs from Fig. 1(b).

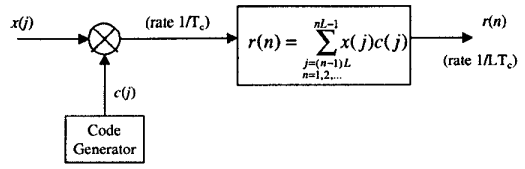


Fig.3. An active correlator, where T_c is the chip period.

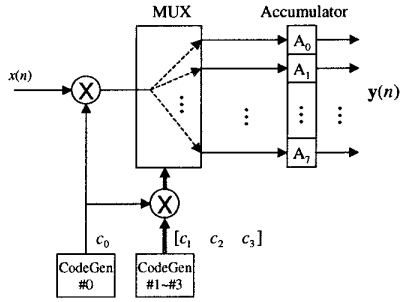


Fig. 4. The structure for implementing the four active correlators in (5).

L \ K	2	4	8
16	16	24	—
32	32	40	—
64	64	72	—
128	128	136	272
256	256	264	400

Table 1. Expected number of additions obtained from (1).

L \ K	1	2	3	4	6	8	10
64	63 (63)	64 (126)	67 (189)	72 (252)	102 (378)	144 (504)	166 (630)
128	127 (127)	128 (254)	131 (381)	136 (508)	166 (762)	272 (1016)	294 (1270)
256	255 (255)	256 (510)	259 (765)	264 (1020)	294 (1530)	400 (2040)	550 (2550)

Table 2. Number of additions required for implementing K correlators of span L by using the proposed method, when a portion of the short PN sequence for IS-95 is used. The numbers in the parentheses are those required by the direct implementation.

L \ K	2	4	8
16	17 (32)	32 (64)	288 (128)
32	33 (64)	48 (128)	304 (256)
64	65 (128)	80 (256)	336 (512)
128	129 256	144 (512)	400 (1024)
256	256 512	272 (1024)	528 (2048)

Table 3. Expected number of additions required for implementing K correlators of span L obtained from (6). The numbers in the parentheses are those required by the direct implementation.

IV. References

[1] A. Polydoros and C. L. Weber, "A unified approach to serial search spread-spectrum code acquisition-Part II: A matched-filter receiver," *IEEE Trans. on Comm.*, Vol. 32, No. 5, May 1984, pp. 550-560.

[2] Vladan M. Javanovic, "Analysis of strategies for serial-search spread-spectrum code acquisition – Direct approach," *IEEE Trans. Comm.*, Vol. 36, No. 11, Nov. 1988, pp. 1208-1220.

[3] Giovanni E. Corazza, "On the MAX/TC criterion for code acquisition and its application to DS-SSMA systems," *IEEE Trans. Comm.*, Vol. 44, No. 9, Sep. 1996, pp. 1173-1182.

[4] L. Milstein, J. Gevargis and P. K. Das, "Rapid acquisition for direct sequence spread spectrum communications using parallel SAW convolvers," *IEEE Trans. Comm.*, Vol. 33, No. 7, July 1985, pp.593-600.

[5] Weihua Zhuang, "Noncoherent hybrid parallel PN code acquisition for CDMA mobile communications," *IEEE Trans. Veh. Technol.*, Vol.45, No.4, Nov 1996, pp. 643-656.

[6] K.K.Chawla and D.V. Sarwate, "Parallel Acquisition of PN Sequences in DS/SS Systems," *IEEE Trans. Comm.*, Vol. 42, No. 5, May 1994, pp. 2155-2164.

[7] Richard Hartley, "Subexpression sharing in filters using canonical signed digit multipliers," *IEEE Trans. on Circuits Syst.*, Vol. 43, No. 10, Oct 1996, pp. 677-688.

[8] M. Yagyu, A. Nishihara, and N. Fujii, "Minimization of adders in fast FIR digital filters and its application to filter design," *Proc. IEEE Int. Symp. Circuits Syst.*, pp. 293-296, Atlanta, May 1996.

[9] TIA/EIA IS-95, "Mobile Station-Base Station Compatibility Standard for Dual Mode Wideband Spread Spectrum Cellular System," Telecommunications Industry Association, July 1993.