Code length Reduction using CRFS Decoding for Frame Synchronization

Anil Patel[1] Snehal Patel[2]
[1, 2]PG Scholar
[1, 2]Electronics & Communication System Department
[1] Sagar Institute of Research and Technology, Bhopal (SIRT, Bhopal) [2] G. E. C. , Surat

Abstract---The process of the mobile station searching for a cell and achieving code and time synchronization to its downlink scrambling code is referred to as cell search. Cell search is complete in two modes: Initial mode cell search when mobile is switched on, and Target cell search in which idle mode search when mobile is inactive and active mode search during a call or activities. The process of achieving code and time synchronization in cell search algorithm divided in to five stages which are slot synchronization, frame synchronization and scrambling code group identification, scrambling code identification, frequency acquisition, and cell identification. Initial cell search is need all five stages while in that last two stages do not need in target cell search. The slot synchronization is achieved by the primary synchronization code in cell search procedure to use in slot boundary detection. During 2nd stage secondary synchronization code is use to achieve frame synchronization and scrambling code group identification in cell search procedure. A pipelined process of the first three stages that minimizes the average code and time acquisition time, during initial cell search procedure much large which is reduced by maximum likehood based frequency acquisition method after first three stage completion

Keywords: W-CDMA, CFRS

I. Introduction

The S-SCH is used to identify the frame boundary and scrambling code group identity. Unlike the P-SCH sequence, the S-SCH sequences vary from slot to slot. There are 16 S-SCH sequences, mapped correspondingly to 16 S-SCH symbols, labeled from 1–16. A frame (15 slots) of 15 such S-SCH symbols forms a codeword taken from a codebook of 64 codewords. The same codeword is repeated every frame in a cell. These 64 codewords correspond to the 64 code groups used throughout the system; thus a code group can be detected by identifying the codeword transmitted in every S-SCH frame. Furthermore, the 64 codewords are all chosen to have distinct code phase shifts, and any phase shift of a codeword is different from all phase shifts of all other codewords. With these properties, the frame boundary can be detected by identifying the correct starting phase of the S-SCH symbol sequence. To maximize the minimum symbol distance of the codebook, between different cyclic shifts of the same codeword or between any cyclic shifts of different codewords, the use of a Comma-Free Reed–Solomon (RS) code was proposed.

A CFRS code is a combination of Comma-Free (CF) and Reed-Solomon (RS) codes. A CF code is one in which any cyclic-shift version of its codeword is not a valid codeword, and therefore can be used for frame synchronization. A RS code, on the other hand, has been well known for its powerful error-correcting capability and widespread use in various applications. For 15 slots per frame, a (15, 3) Reed–Solomon (RS) code over GF(16) is used. The RS code has a minimum distance of 13. Moreover, to minimize cross-channel interference, the 16 S-SCH sequences and the P-SCH are mutually orthogonal.

During stage 2 of the cell search procedure, the MS uses the SCHs Secondary Synchronization Code (SSC) to achieve frame synchronization and identify the code group of the cell found in stage 1. This is done by correlating the received signal with all possible SSC sequences and identifying the maximum correlation value.

II. Frame Synchronization and Code Group Identification

After slot synchronization, code-group and frame synchronization can be accomplished in the second stage, as shown in Fig. 1. First, 16 (complex-valued) correlators are used for the 16 SSC correlations in a slot. Coherent accumulation is possible in this stage by using the channel estimation coming out from the first stage. Second, after making hard-decision on which SSC code is used in slots, a set of 15 hard-decided symbols is collected as x for CFRS decoding, where x is correlated with the 64 CFRS code words. Since each codeword has 15 possible cyclic-shift positions, it results in 960 (64x15) correlation values. Lastly, the code- group and cyclic-shift index associated with the maximum value are identified as the desired code group and frame boundary, respectively. The more detailed operation of the CFRS decoding is given in the next section.

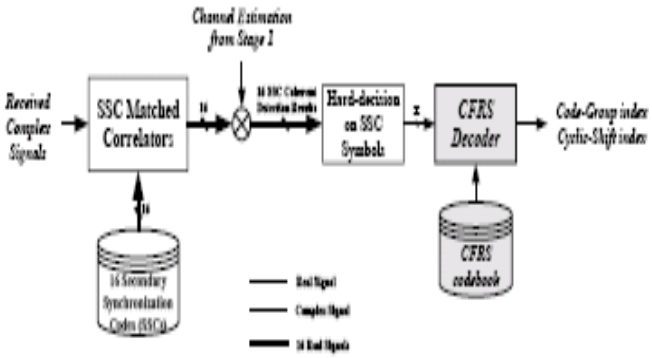

Fig. 1: Detections of the second stage [6].

III. CFRS Decoding

As mentioned above, a set of 64 CFRS code words (of length 15 symbols) is used in the 3GPP W-CDMA/FDD system for frame and code-group identification. Before frame is synchronized, each set of consecutive 15 symbols coming out of the hard-decision device in Fig. 2.1 represents a cyclic-shift version of a CFRS codeword if no error occurs

on the hard decision. Since there are 15 cyclic-shift positions for each codeword, the decoder has to determine from 960 "hypotheses" to obtain the true code group and frame boundary.

The CFRS decoding is given as the following.

$$(i, j) = \arg\{\max\{Y_{i,j} = X \square H_{i,j}\}_{\substack{i=0...63,\\ j=0...14}}\},$$

$$X = \{x_1, x_2, x_3..., x_{15}\},\ x_k \in GF(16), 1 \le k \le 15,$$

$$H_{i,j} = \{h_{i,j+1}, h_{i,j+2}, h_{i,j+3}..., h_{i,15}, h_{i,1}, ...h_{i,j-1}\},\ h_{i,j} \in GF(16), 1 \le j \le 15, \quad (1)$$

Where i is the code group index,
j is the cyclic-shift index,
x is the received SSC code vector,
Hi,j is the i-th hypothetical CFRS codeword with j cyclic-left-shift, and

$Y_{i,j} = X \square H_{i,j}$ is the correlation value of x and Hi,j.

That is,

$$X \square Hi, j = \sum_{k=1}^{15} X_k \oplus h_{i,j+k \bmod (15)}, \quad (2)$$

where $\{X_k\}$ are the received SSC symbols at the output of hard decision device,

$\{H_{i,j}\}$ are the code symbols of the i-th codeword, and $\oplus$ denotes the XOR (Exclusive OR) operation.

A. Systolic CFRS decoder architecture

From figure 3.2.2(a) observe, the decoder consists of an input pattern generator (IPG), a 16x15 processing element (PE) systolic array, a 16x1 boundary processing element (BPE) array, a refreshing module, and a 64x15 CFRS ROM. The IPG is responsible for cyclically shifting the received vector **x**. Note that in hardware implementation instead of cyclically shifting 64 **H**i, j as given in (6), only **x** is shifted in the correlation operation so as to reduce circuit complexity and power consumption. Also, the IPG has to transform all 15 cyclic-shift versions of **x** into a skewed form for pipelining process. These cyclic-shift versions of **x** will be used as the input patterns to the 64x15 PE systolic array. Figure3.2.2 (b) details the IPG architecture design. It consists of a cyclic-shift register to rotate **x**, a Johnson counter to generate the mask for skewing cyclic-shift versions of **x**, and some combinational circuits to perform masking operations.

The 16x15 PE systolic array is dedicated to correlate 15 cyclic-shift versions of **x** with possible 64 codewords, **H**i, j for i = 0~63, j = 0. The code symbols {hi,j} are pre-stored into 16x15 PEs of the systolic array with one row for one codeword. The code symbols will be replaced after them are fully correlated with the patterns. Each row of PEs is responsible for correlating the skewed input patterns with its codeword. Fig. 3.2.2(c) with the shows that a PE consists of correlating circuit, a 4-bit adder (recall that there are 15 symbols in a codeword, 15 < 24), and three registers (Reg. X, H, and Y). After summing the correlation output with the partial results $Y_{i,j}$ propagated from the left-hand PE, the updated $Y_{i,j}$ is passed forward to the right-hand PE, and at the same time, the input symbol X_k is passed down to the next PE below. It is clear that both the vertical and horizontal data processing flows of the array are fully parallel and pipelined.

Tail Bits-
All-zero bits to indicate the start and the end of the burst.

Data Bits :-
Speech data is to carry after

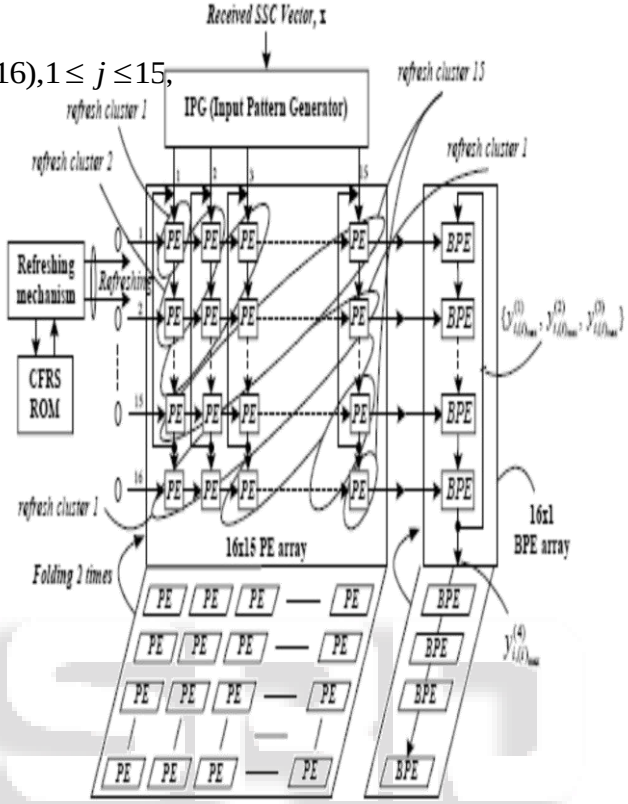

Fig. 2: Systolic architecture for CFRS decoding[6].

For decoding efficiency, each PE is refreshed upon the patterns flowed through it. A refreshing module is needed in order to update $h_{i,j}$ in PEs. As a result of the skewed patterns in the systolic array, the refreshing operation is also in a skew form. In Figure 2.2, PEs required to be refreshed simultaneously are clustered together for clarity. There are total 15 refreshing clusters, and each cluster includes 16 PEs. Each cluster's refreshing period is 15 clocks. Figure 3.2.2(d) details the refreshing module that consists of 15 cluster counters, a ROM reader, a refreshing register, and a cluster dispatcher. When a certain refreshing counter times out, the ROM reader will read out the corresponding $h_{i,j}$ from the CFRS ROM, and then the cluster dispatcher will update the $h_{i,j}$ to their destination PEs.Furthermore, the input patterns are externally feed-backed from the PEs in the 15th row to the first row as new input patterns for the next set of codeword correlations. As a result, IPG only needs to generate the skewed patterns once.

After completing all the 960 correlations, the maximum of the correlation values $\{Y_{i,j}\}$ will be selected. As shown in Figure 2.2, an array of 16x1 BPEs is employed to find out the maximum since the correlation values $\{Y_{i,j}\}$ from the PE systolic array are outputted in a skewed form. Figure 3.2.2(e) shows that a BPE comprises two comparators (Cmp. I and J) and two register (Reg. I and J).

For the i-th row, Cmp. J determines the maximum correlation value $y_{i,(i)max}$, and Reg. J stores $y_{i,(i)max}$ and the corresponding cyclic-shift index max (i), respectively. On the other hand, Cmp. I determine the larger between $y_{i,(i)max}$ and $y_{i-1,(i-1)max}$ along with the corresponding cyclic-shift index and code-group index are stored into Reg. I, and will be passed down to the BPE below in the next clock. The $y_{i-1,(i-1)max}$, code-group index i, and cyclic-shift index max (i) of the last (16-th) BPE should be passed on to the first BPE for subsequent comparisons.

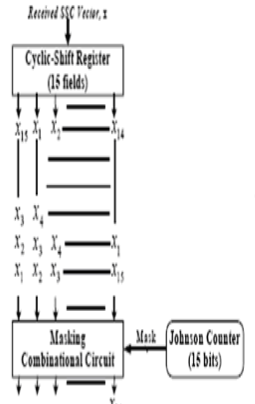

Figure 3.2.2(b): Architecture of Input Pattern Generator (IPG)[6].

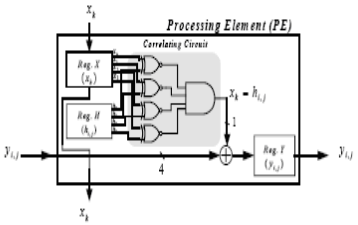

Figure 3.2.2(c): Architecture of Processing Element (PE)[6].

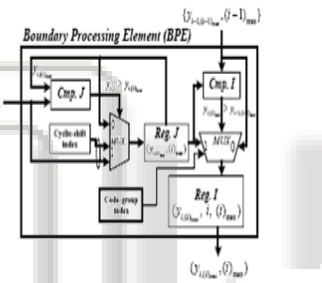

Figure 3.2.2(e): Architecture of boundary processing element (BPE) [6].

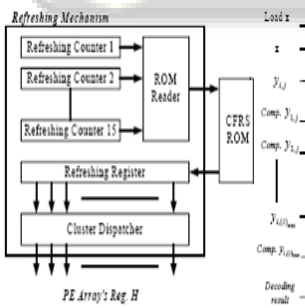

Figure 3.2.2(d): The refreshing mechanism architecture[6].

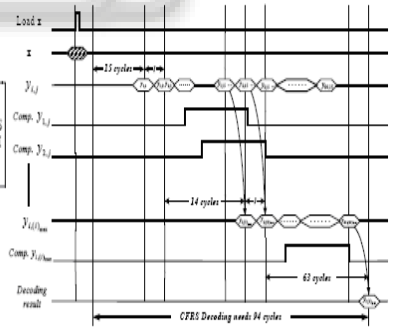

Figure 3.2.2(f): CFRS decoding signals and timing[6].

After comparisons of four sets, the code-group index i and the associated cyclic-shift index max (i) coming out from the last BPE are the desired code-group index and cyclic-shift index, respectively. Figure 3.2.2(e) illustrates the decoding signals and timing. The total decoding latency of this design is 94 clock cycles. Among $y_{1,(1)max}$ them, 15 clocks are used for the generation of clock for 14 clocks for obtaining $y_{i,(i)max}$ by comparing $y_{1,1}$….$y_{1,15}$ clock for waiting $y_{2,(2)max}$ to come out, and finally 63 clocks for comparing $y_{1,(1)max}, y_{2,(2)max}, \ldots\ldots, y_{15,(15)max}$.

In summary, the decoder works as follows. After the IPG generates the skewed patterns of **x**, the 16x15 PE array begins to correlate the patterns with the first set of 16 codewords and produces 240 (15x16) correlation results, that is $Y_{i,j}$ for i = 0~15, j = 0~14. Upon those PEs in the same cluster completing correlations with all patterns, the refreshing module updates their hi,j by new ones, $h_{i,j}$ for i = 16~31, j = 0~14 correspondingly. Then, the correlations of another set of 16 codewords follow. The 16x1 BPE array determines the maximum value of $Y_{i,j}$ for i = 0~15, j = 0~14 first which is denoted as $y^1_{i,(i)max}$, and then it is feed-backed to the first BPE for comparisons with $y_{16,(16)max}, y_{17,(17)max}, \ldots\ldots, y_{31,(31)max}$.

After obtaining max $y^4_{i,(i)max}$, the associated i and max (i) are the desired code-group and frame-boundary indices, respectively.

IV. CONCLUSIONS

Cell search algorithm is necessary to achieve synchronization of channel of ms with bs for w-cdma system. cell search is mainly consisting of three stages: stage 1 slot synchronization, stage 2 frame synchronization and code group identification, stage 3 scrambling code identification. In stage 2 of the initial cell search algorithm can be implemented by many methods. Here, only we described using CFRS code decoding and second stage of pipelined process minimizes the average code

REFERENCES

[1] B.S. Atal and J.R. Remde. A new model of lpc excitation for producing natural sounding speech at low bit rates. Proc. IEEE-ICASSP, 1982.
[2] J.C. Bellamy. Digital Telephony. John Wiley and Sons, 2000.yg cv
[3] J. P. Campbell and T. E. Tremain. Voiced/unvoiced classification of speech with applications to the US government lpc-10e algorithm. IEEE ICASSP, page 9.11.1 to 9.11.4, 1986
[4] P.B. Denes and E.N. Pinson. The Speech Chain: The Physics and Biology of Spoken Language. Waverly Press, Baltimore, 1963.
[5] N.S. Jayant and P. Noll. Digital Coding of Waveforms. Englewood Cli_s, NJ: Prentice-Hall, 1984.
[6] E. F. Deprettere Kroon, P. and R. J. Sluyter. Regular-Pulse Excitation A Novel Approach to Effective and Enceinte Multipulse Coding of Speech, volume ASSP-34. IEEE Transactions on Acoustics, Speech, and Signal Processing, October 1986.
[7] Jing Su Ismail Lakkis and ShuZo Kato. A simple cohearentgmsk demodulator, volume 18. IEEE Communication Magazine, 2001
[8] J. Makhoul. Linear prediction: A tutorial review. Proceedings of the IEEE, page 561 to 580, 1975.
[9] J. D. Markel and A. H. Gray. Linear Prediction of Speech,. Springer-Verlag, NewYork, 1976.
[10] Bruno A Olshausen. Aliasing. October 10, 2000.

[11] J. G. Proakis. Digital Communications. McGraw Hill, third edition, 1995.
[12] L. R. Rabiner and R. W. Schafer. Digital Processing of Speech Signals. Prentice-Hall Signal Processing Series, 1978.
[13] L. R. Rabiner and R.W. Schafer. Digital Processing of Speech Signals,. Prentice-Hall, Englewood Clifis, NJ., 1986
[14] S. Singhal and B. S. Atal. Amplitude Optimization and Pitch Prediction in Multipulse Coders, volume 37. IEEE Transactions on Acoustics, Speech, and Signal Processing, 3 edition, March 1989
[15] Thierry Turletti. Towards the software realization of a gsm base station, volume 17. IEEE Communication Magazine, April 1999.
[16] Thierry Turletti. Complexity of a software gsm base station, volume 37. IEEE Communication Magazine, February 1999.
[17] Gina Upperman. Linear predictive coding in voice conversion. 2003.
[18] G. Davidson Yong, M. and A. Gersho. Encoding of lpc spectral parameters using switched adaptive inter-frame vector prediction. IEEE ICASSP, page 402 to 405., 1988.

