DESIGN OF A SYNCHRONIZATION SCHEME FOR A BANDWIDTH-ON-DEMAND MULTIPLEXER-DEMULTIPLEXER PAIR BASED ON WAVELET PACKET TREE FILTER BANKS

M. Sablatash and J. Lodge
Communications Research Centre
3701 Carling Avenue
P.O. Box 11490, Station H, Ottawa, ON K2H 8S2, Canada

ABSTRACT
For a spectrum-efficient, bandwidth-on-demand multiplexer-demultiplexer for a multiuser, multicarrier communication system based on wavelet packet trees for downstream transmission described in recent publications by the authors, an innovative scheme for synchronization is proposed which uses a unique sync word at the root of the tree. An example filterbank tree with 8 leaves is used throughout the paper. For this example the creation of the unique 32-bit sync word is described. The multirate signal processing and construction and properties of 25 32x32 matrices required in the multiplexer to find locations of sync words at input ports to cancel the effects of data and insert the 32-bit sync word in a window in the correct location at the root of the tree is described. The ratio of maximum to minimum absolute values of eigenvalues of the matrices is used to ensure well-conditioning, since they must be inverted in the processing.

2. FINDING THE UNIQUE 32-BIT SYNC WORD
The 32-bit sync word is

\[ m_{32} = [-1 -1 -1 -1 -1 -1 -1 -1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1] \]  (1)

This was obtained from the standard 31-bit m-sequence

\[ m_{31} = [-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1] \]  (2)

by shifting the sequence by one to the left, cyclically, so that a \(-1\) appears at the end of the sequence, and then simply adding a \(-1\) to the end of the resulting shifted sequence to obtain the 32-bit word of (1). The sequence \(m_{32}\) has good autocorrelation and random spectral properties. The reason for choosing a length of 32 for the sync word is that sync words at input ports to effect the insertion of \(m_{32}\) were upsampled by 2, 4 or 8, depending which level of the tree they were at, which resulted in an requirement that \(m_{32}\) be even.

3. FINDING ALL INDEPENDENT SETS OF INPUT PORTS
The problem of selecting all possible subtrees with independent ports is the same as the problem of selecting the possible sets of wavelet packet functions which can form wavelet packet bases. This problem has been solved and the solution given in [11] and [4] as follows for a tree of \(L\) levels. For every partition \(P\) of \(N_L = \{1, 2, ..., 2^L - 1\}\) into sets of the form

\[ I_{k,n} = \left\{ k, n, ..., 2^k (n + 1) - 1 \right\} \]  the collection of

so that inputs (dependent on the data) can zero out the locations of the unique sync word at the root of the tree. The main purpose of this paper is to describe how a 32-bit sync word is chosen, how to identify all the possible independent sets of input ports for a tree such as that shown in Fig. 1 and, mostly, to show how to find the sync words at any chosen set of input ports, and to ensure that this is feasible.
Figure 1. Example tree with 8 leaves, different designs of the quadrature mirror filter pairs with non-linear phase at each level, and Gray code ordering [9, 10] used to illustrate the proposed synchronization scheme method and computations.

The number of subtrees with independent inputs can be found from the recursive equation

\[ T_n = T_{n-1}^2 + 1. \]  

The recursion begins with depth \( n=1 \), and \( L=0 \) levels, for which \( T_{n,1}=0 \), so \( T_n=T_{1}=1 \), which is just the degenerate case of the root of the tree. Thus, for trees of depths 1 to 7, corresponding to levels 0 to 6, and to numbers of leaves from \( 2^6=1 \) (just the root) to \( 2^8=64 \), the numbers of subtrees with independent inputs are 1, 2, 5, 26, 677, 458,330, and 210,066,388,901, respectively.
Clearly, the number of possible sets of subtrees with independent inputs increases very rapidly with number of leaves (or levels) in a tree, so it would not be practical to determine and list these by inspection as has been done for the tree with 8 leaves of Fig. 1.

4. LOCATING THE SYNC WORDS AT THE INPUT PORTS

To explain how to locate the input sync words in the input data blocks for one of the 25 non-trivial independent sets of input ports the set 13, 12, 5 and 6 in Fig. 1 is used to illustrate the method. A matrix A of unit pulse responses from the input ports to the 32-bit sync word at the root of the tree is constructed as follows. As will be better understood shortly, 16 consecutive unit pulses into port 13, 8 into port 12, and 4 each into ports 5 and 6 will yield 32 samples unit pulse responses in a 32-bit window at the root of the tree, from top to bottom of the matrix, in the same order as the set of input ports, 13, 12, 5 and 6, and in the successive order of the unit pulse inputs in the data block inputs at each of the ports. First, because the path through from port 6, through ports 11 and 14, to the root of the tree of Fig. 1 passes only through high-pass filters with maximum phase transfer functions, the unit pulse responses which are delayed the longest are those for unit pulse inputs into port 6. The maximum variations in the unit pulse responses for the 4 consecutive pulse inputs into port 6 was judged to occur in the window 100-131 at the root of the tree. Each of the pulse responses due to consecutive input unit pulses at ports 6 and 5 is shifted by 8 samples from the preceding one in the window 100-131 due to the total upsampling by 8. For the 8 unit pulse inputs into port 12 this shift is by 4, while for the 16 unit pulse inputs into port 13 the shift is by 2. To obtain the lowest possible ratio of maximum to minimum absolute values of eigenvalues of A, so that it will be well-conditioned for inversion will be the maximum variations of all the unit pulse responses must occur in the window 100-131. To effect this the unit input pulses at the ports 5, 12 and 13 must be appropriately delayed by amounts which were determined by trying 2 or 3 starting sample locations of the pulse responses with the input unit pulse sequence starting at 1 to map into into 100 in the window, and choosing those which resulted in the lowest eigenvalue ratio. This resulted in eigenvalue ratios between 1.1033 and 1.6744 and the locations in the input data blocks of input sync words at each of the input ports for all 25 non-trivial independent sets of input ports for Fig. 1.

It was observed that for ports 4, 5, 10, 11, 13 and 14 the starting position of the sync word in an input data block for these ports differed by one when the same port appeared in different sets of input ports. Thus the position in the data block for the sync word input at port 4 was 8-11 in 5 sets and 7-10 in 5 sets. At port 5 it was 4-7 in 9 sets and 3-6 in one set. At port 10 it was 15-22 in 8 sets and 16-23 in 2 sets. At port 11 it was 7-14 in 8 sets and 8-15 in 2 sets. At port 13 it was 50-65 in 3 sets, and 49-64 in two sets. At port 14 it was 32-47 at 4 ports and 31-46 at one port.

Since the ratio of maximum to minimum of the absolute values of the matrix A lie 1.1033 and 1.6744 the matrix A is well-conditioned for all 25 cases, so the method, which will be seen in the next section to depend on the inversion of A, is feasible.

It was observed that when the coefficients of the lowpass filters are all normalized so that their sum is unity, the absolute values of the eigenvalues of A are all less than one. It can be shown that [12] the eigenvalues of A are all less than unity in absolute value, as was observed for all 25 sets of input ports.

5. INSERTION OF THE UNIQUE SYNC WORD AT THE TREE ROOT

The inverse of the matrix A is used to calculate the sync words \( S_1 \) at the input ports which will cancel the samples, \( d/32 \), in the
window at the root of the tree due to the data in the input data block, from the relationship

\[ A S_1 + d 32 = 0, \quad (4) \]

thus creating zeros in the window of length 32. Then the unique sync word, \( m_{32} \), is inserted in the window. Thus, in this case, the \( S \) is calculated from (4) using the inverse of \( A \). Then the filtering of the input sync words \( S \) by \( A \) results in zero values for the 32 samples in the window, according to (4). The unique sync word is then inserted in the window at the root of the tree. This method has been implemented in recent simulation programs [12] to study the BER performance of the downstream communication system with the proposed synchronization scheme.

6. CONCLUDING DISCUSSION

For the example QMF bank with 8 leaves considered in this paper a technique is described for finding the locations and values of input sync words to zero out the samples in the unique sync word locations in a 32-bit long window at the root of the tree due to the data transmitted from the input data block. The 32x32 matrices of unit pulse responses from the positions of the input sync words to the unique sync word at the root of the tree were all found to be very well-conditioned so they can be inverted for finding the input sync words. For normalization such that the sum of lowpass filter coefficients are equal to one the eigenvalues of all the matrices in the example were all less than one in absolute value. This can be shown to be true in general.

From parallel studies [12] it has been found that the synchronization scheme works for linear phase filter designs in the QMF’s. Very similar techniques can be applied to filter banks such as the one studied here, although the linear phase case is easier to implement.

7. ACKNOWLEDGEMENTS

The authors are grateful to Dr. Paul Guinand for help in proving (3) and to W.F. McGee for many discussions and insights, and excellent results in simulations.

8. REFERENCES


