OFDM Synchronization Techniques Analysis for IEEE 802.16d Review

Poonam Kiran Rana
PG Student
ME-ECE (ME & VLSI)
Chandigarh University, India

Deepika Setia
Asst. Professor
ECE Department
Chandigarh University, India

ABSTRACT

Orthogonal frequency division multiplexing (OFDM) is a reliable technology for high-speed data transmission by virtue of its spectral efficiency and robustness to multi-path fading. These advantages can be achieved only with good synchronization both in time and frequency. This paper introduces efficient synchronization methods for an OFDM based system, IEEE 802.16d. A symbol timing synchronization scheme is critical in the design of an OFDM receiver. Large timing errors can result in loss of orthogonality between subcarrier, ISI and severe bit degradation. To minimize this degradation two kinds of synchronization algorithms are provided: Auto-correlation and cross correlation. This brief discusses these techniques used in synchronization of OFDM and which one is preferred.

Keywords

Auto Correlation, Cross Correlation, IEEE 802.16d standards, orthogonal frequency division multiplexing (OFDM), Synchronization**.**

1. INTRODUCTION

Orthogonal Frequency-Division Multiplexing (OFDM) is a method of encoding digital data on multiple carrier frequencies. OFDM is a popular for wideband digital communication and is known as the special case of frequency division multiplexing. It is an effective modulation technique used in both wired and wireless communication systems. It is a multicarrier modulation technique that has used for wireless communication systems due to the high spectral efficiency, immunity to multipath distortion, and being flexible to integrate with other techniques. However, the high-peak-to-average power ratio and sensitivity to synchronization errors are the major drawbacks for OFDM systems. Due to its spectral efficiency and robustness to multipath fading it is considered in high bit rate data transmissions in many networks such as IEEE 802.11 and 802.16d. But OFDM performance is seen to sensitive to receiver synchronization. The errors in synchronization leads to Inter symbol interference and offsets in frequency causes inter carrier interference. Hence, synchronization is important for efficient performance in OFDM systems [1]. Here in this paper is the review of why synchronization of OFDM is done and different techniques that were used to achieve that. Autocorrelation and Cross Correlation were the techniques that were being used to obtain low power synchronization and further research is also being done in order to reduce complexity and other drawbacks of the synchronization process. Here in this paper synchronization and its errors are discussed first and what research has been done till now on this subject. Then the correlation techniques are discussed and what work has been done about it till now. And in the end what is the future scope in this research and the conclusion of all that is been done. Now here is the review of the OFDM synchronization and the techniques used.

2. SYNCHRONIZATION OF OFDM

In OFDM overlapping of sub channels causes fairly well spectral efficiency but the performance is sensitive to sub channel interference caused by frequency offset [2]. Also, the timing errors caused leads to inter symbol interference, which lowers the performance of OFDM. Therefore in order to achieve a good performance synchronization in both time and frequency is required. Much research has focused on improving OFDM synchronization performance and accuracy. Cyclic prefix (CP)-based methods were introduced to determine frequency offset and symbol timing. Cyclic Prefix means cyclically extended the guard interval where by each symbol periodic extension of the sequence itself. To support this, all OFDM frames begin with preamble symbols which can also be used to estimate the frequency offset. This relies on the characteristic of a preamble signal where autocorrelation of the received signal is done.

Researchers focused on improving OFDM synchronization by focusing on its performance and accuracy. One of the methods introduced was cyclic prefix method to compute frequency offset and symbol timing errors but it was difficult to find the start of the frame. So all the OFDM symbols begin with preamble symbols and are used in calculating the frequency offset [3]-[4]. However there is some uncertainty in determining the start of the frame. So modified timing metrics in the autocorrelation of the symbols was done so as to determine the start of the frame but this phenomenon was sensitive to Additive White Gaussian Noise (AWGN) and causes error in frequency selection.

Synchronization scheme widely used in burst mode OFDM systems such as IEEE 802.16d is to calculate the auto-correlation of the received samples. This scheme is robust to noise and carrier frequency offset and is computed at low hardware cost. But it is low on complexity and there is seen to be little uncertainty in the results. And the result of autocorrelation technique is sensitive to additive white gaussian noise (AWGN) and frequency selectivity. T.H.Kim proposed synchronization method based upon the IEEE 802.16d preamble symbol with two different computation processes [5]. In first method to obtain more synchronized frequency and to reduce the hardware cost autocorrelation is done for coarse symbol time offset and fractional carrier frequency offset. And in the second method fine symbol time offset and integer carrier frequency offset are calculated by doing cross correlation between the received samples and known preamble.

Another algorithm was proposed by C.N.Kishore which computes cross correlation between the known and received preamble which overcame the problem of determination of start of the frame and this algorithm required the knowing of time domain preamble [6]. This algorithm provided with low signal to noise ratio while being high on complex computations.

Further research was done in order to find an appropriate algorithm for the synchronization of OFDM which was low on hardware cost and complex computations. Apart from auto correlation another technique was considered. Meanwhile double autocorrelation technique was used in implementation of OFDM-WLAN synchronizer so that hardware cost was reduced and this technique was quite successful in doing so [7]. This technique used double autocorrelation of the known preamble symbols which was efficient in providing timing synchronization.

In order to compare the performance and complexity of these two techniques A.Fort discussed the implementation of both auto correlation and cross correlation methods [8]. Auto correlation technique where provides with low hardware cost but gives less accuracy than the cross correlation algorithm. Cross correlation technique however comes with the high cost factor still it is preferred over auto correlation despite of its high complex computations. Cross correlation technique provides with accurate results compared to auto correlation techniques.

3. INTRODUCTION TO CORRELATORS

Generally, correlators are also called multipliers. It is widely used in OFDM. In the fourth generation technology there are multiple person passing multiple data or single person passing several information at the same time. So synchronization problem occurs causing frequency offsets. In order to overcome this problem correlators were designed. These correlators are mainly used under wireless communication. There are two main kinds of correlation: Autocorrelation and Cross Correlation. Under here is the review of what these correlation mean and how were being used[8].

3.1 Autocorrelation

Autocorrelation means using the same signals but with different parameters. Auto correlation majorly uses multiplier based correlators.. Autocorrelation-based techniques are preferred for implementation on FPGA because of their lower hardware costs. Dick and Harris reported on the FPGA implementation of an OFDM transceiver. They showed that FPGAs, with their highly parallel architecture are suitable for the implementation of OFDM transceivers also presented an FPGA implementation an OFDM-WLAN synchronizer. In this brief, the timing synchronization is obtained by double autocorrelation based on short training symbol that allows a reduction in the hardware cost on FPGA.The performance and complexity of FPGA implementation of autocorrelation and cross correlation algorithms. Their results show that the accuracy of cross-correlation algorithms is better than that of autocorrelation algorithms. However, the accuracy of cross correlation comes at significant hardware cost. Despite proposing a new cross correlator implementation presented in to reduce hardware cost compared to a classic cross-correlation approach, it is still at least five times more complex to implement that autocorrelation because of the fact that several multipliers are required.

3.2 Cross Correlation

Cross correlation is known as multiplying two different signals with two different parameters. This technique mainly uses multiplierless computations. One of its main features is that high speed data could be transmitted and received. Cross-correlation between received samples and a known preamble can achieve highly accurate timing synchronization. However, computation of this requires significant resources. Multiplierless correlators for timing synchronization were introduced in designs for IEEE 802.11a OFDM frames, based on expressing the correlator coefficients as sums of powers of 2 that only require shift and add operations. Also implementation of correlators on FPGAs is a highly efficient job to do.

3.3 Design of Correlators

Desgning of correlators is done in order to deal with the timing synchronization, power consumption and resource utilization. Correlators are also called filters as they mainly clear the output. Correlators are designed using preamble symbols and here we are discussing about the IEEE 802.16d symbol which consists of two consecutive OFDM symbols as shown in Fig.1.

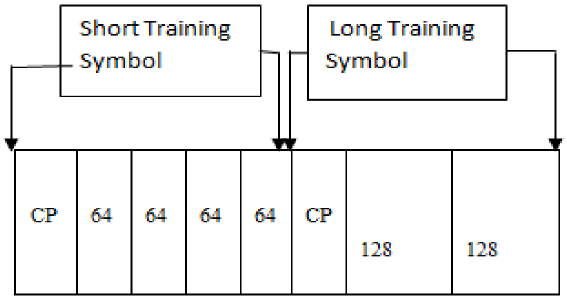

Fig 1: Downlink preamble symbols for IEEE 802.16 [9].

The short symbol consists of four identical 64-sample fragments in time, preceded by a CP. This is followed by the long symbol which contains two repetitions of a128-sample fragment and a CP. The 64 samples in the short symbol are used to perform cross correlation with the received samples for timing synchronization. Therefore, the correlators are designed to compute cross correlation with 64 constant coefficients. In this brief, we explore two approaches to implement such correlators[9]. The first is based on Xilinx Virtex-6 FPGA DSP48E1 Slices which is the standard approach to such an implementation. The second uses multiplierless correlation implemented on both a Xilinx Virtex-6 and a low-power Xilinx Spartan-6 device. Both designs are designed to receive real and imaginary 16bit samples in this fixed-point format. The output is the sum of 64 coefficient products, with each smaller than unity. So, the complex output words are in 21-b fixed-point. If such a design were implemented blindly, with no consideration for the FPGA architecture, the synthesis tools would infer the use of embedded DSP blocks for multiplication, but would likely achieve poor timing because of the inability to optimize the use of the DSP block and external logic elements.

The DSP48E1 primitive’s additional circuitry within them that enables the design of optimized data path. However, this must be done manually through writing the code in a particular style. Otherwise, the synthesis tools cannot always infer the most efficient structure.

4. METHODOLOGY

In the design of correlators two techniques so far has been designed one is multiplier based correlator and other is multiplierless based correlator. Both of these techniques are implemented in their own unique style based on their own computation techniques and both of them have their own improvements and drawbacks.

First would be the multiplier based correlator[9] which was implemented DSP48E1 Slice inside the Virtex-6 contains a multiplier followed by a configurable arithmetic unit to provide many independent functions, e.g., multiply, multiply-accumulate, multiply-add, three-input add etc. It also allows the data path to be configured for various input combinations and register stages; a three stage pipeline offers maximum performance. The first design uses non-pipelined DSP48E1 Slices in transpose direct form, as shown in Fig. 2 with 64 coefficients, Pr corresponding to the 64 complex conjugated values of samples in the preamble.

The output of the FIR filter is transpose direct form correlation in it. The coefficients are pre computed according to the IEEE 802.16d standard. The second design spreads the complex multiply-adds in a five-stage pipeline, shown in Fig. 2, consisting of DSP48E1 Slices configured for three-stage internal pipelining. Ri _ Re and Ri_ Image the real and imaginary parts of received sample, respectively. Pr_ Re and Pr_ Im similarly represent the complex conjugation of known preamble.

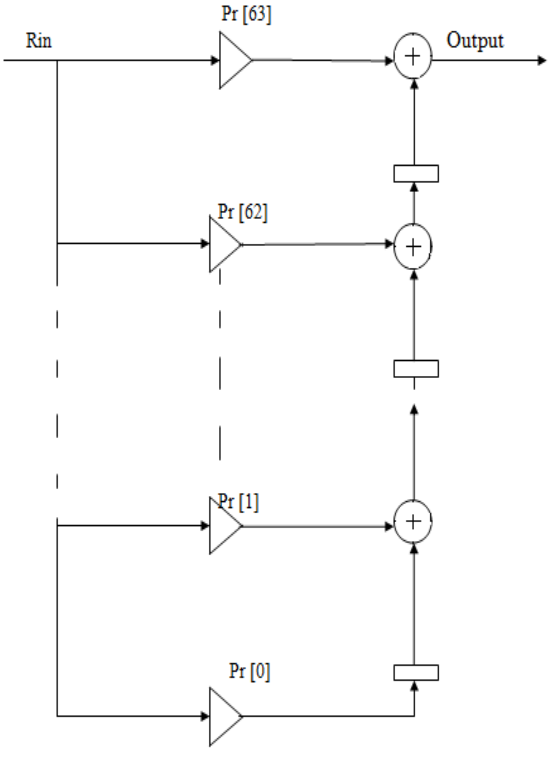

Fig 2: Transpose direct form of the multiplier correlator [9].

The coefficients are pre computed according to the IEEE 802.16dstandard. The second design spreads the complex multiply-adds in a five-stage pipeline, shown in Fig. 3, consisting of DSP48E1 Slices configured for three-stage internal pipelining. Ri_Re and Ri _Im are the real and imaginary parts of received sample, respectively.Pr_Re and Pr _Im similarly represent the complex conjugation o*f* known preamble. The pipeline registers for the Pr _Re, Pr _Im are eliminated because they are considered to be of constant value Re and Im are the real and imaginary parts of the previous multiply-add, MAn−1. The output of these complex multiply-add can be passing the direct form of the correlator.

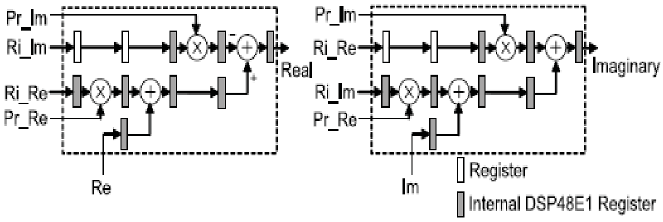

Fig 3: Pipeline structure of the complex number multiply-add [9]

Fig. 3 presents the pipeline structure of the correlator. The additional pipeline registers are required for handling the received sample. Adding pipeline registers should improve the performance significantly. The coefficients are pre computed according to the IEEE 802.16dstandard. The second design spreads the complex multiply-adds in a five-stage pipeline, shown in Fig. 3, consisting of DSP48E1 Slices configured for three-stage internal pipelining. Ri_Re and Ri_Im are the real and imaginary parts of received sample, respectively.

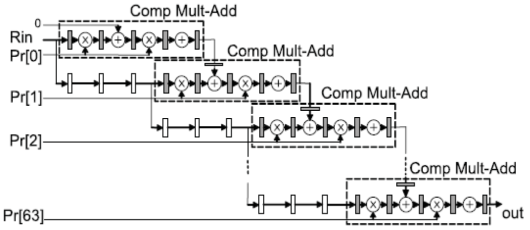

Fig 4: Pipeline structure of correlator using DSP48E1 Slices [9]

In above figure of pipeline structure using DSP Slices. Adaptive systems are set to become more main stream, as numerous practical applications in the communications domain emerge. FPGAs offer an ideal implementation platform, combining high performance with flexibility. While significant research has been undertaken in the area of FPGA partial reconfiguration, it has focused primarily on low-level architecture-specific implementations.

Drawbacks of this implementation were that this multiplierless correlator has high resource utilization. Then multiplying two different signals caused signal to noise ratio (SNR). In this technique sometimes signals were interfered. At the same time signal was not synchronized. So synchronization problem still occurred though on a small level.

Second method that was used was multiplierless correlation. The principle of multiplierless correlators is to represent coefficients and round them in the form of summed powers of 2. Hence, a shift-and-add is performed instead of multiplying by coefficients. It is expected that multiplierless correlation is more efficient, but with embedded hard multipliers in modern FPGAs, it is unclear whether they should still be considered favourable. Furthermore, synchronization accuracy must be considered. To explore this, four alternative multiplierless correlators are implemented using four coefficient sets with increasing degrees of rounding, to compare the cost and performance and evaluate against multiplier-based correlators. The coefficient sets are found by quantizing the 64 normalized preamble samples with quantization of 1, 0.5, 0.25, and 0.125. Structure for multiplierless correlators is shown in figure 5. It shows that timing synchronization for IEEE 802.16a WLANs requires using a correlator to correlate the received signal with a known waveform. Straightforward implementation of this correlator results in the need to perform 320 million complex multiplications per second. This significant requirement can be eliminated by using multiplierless correlators. multiplierless correlators are designed based on constraining the real and imaginarly parts of correlator coeficients to be sums of powers of two. Sets of coefficients that yield good synchronization performance for simple AWGN channels are first identified; then their goodness for indoor communication environments is verified by simulation for multipath fading channels. Several multiplierless correlators are implemented for the timing synchronization and resource utilization.

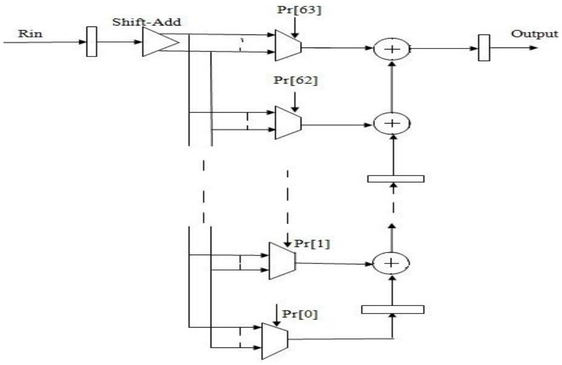

Fig 5: Multiplierless correlator [9]

Among these correlators it only requires to perfom only 26 addition subtraction operations per correlator output while a similar synchronization performance can be maintained. Correlators that eliminate the need to perform multiplication are designed. Using this multiplierless correlator leads to considerable reduction in the implementation complexity of receivers. First using the data is eight bit data. In this eight bit data is mainly othe 0 to 7 bits. Then next in this signal will be passing the Shift_Add block. Shift_Add block mainly right shift or left shift in it. In this proposed method✓mainly deals with left shift option .Seven times left shift option will be used. Then next in this value will be passing multiplexer. Multiplexer normally will act as selected switch. Any value will be chosen at any times. In this section only multiplication of input coefficients value will happen. This common Shift_Add block calculates all possible values for 64 coefficients. The multiplexers are used to select the corresponding values from Shift_Add to accumulate in order to generate the correlator output. These are based on the expressed coefficients Pr[n] that are pre computed on the basis of quantizing the 64 preamble samples. Since the Pr[n] values are constants, after synthesizing the design, the multiplexer is optimized .as hard-wired logic, and the preamble cannot be changed. To support different OFDM preambles, the Pr[n] could be stored in a register, and a real multiplexer used instead of hard-wired logic. This results in increased resource utilization but provides a more flexible solution. Important things in this system we are using Shift Add register. In this register inside inbuilt of the clock in it. In this final output value there will be no interference and timing synchronization will be accurate.

Main advantages of using this technique are reduced timing synchronization and delay problem is solved. Resource utilization is also less as compared to multiplier based design as it uses less computations. And also power consumption is also less.

5. CONCLUSION

In conclusion, synchronization is an important factor to be solved in OFDM. In order to get a good performance an efficient synchronization should be implemented. For any technology to work various factors are needed to be considered which makes it usable, durable and something that is easily acquired. Implementation should be easy and accurate for any user. Same was discussed for OFDM synchronization which is done using two algorithms and here it was concluded which method was preferred. Lots of research was done in order to conclude which algorithm is preferred over the other and why. Here we discussed on the various research work that was done for this implementation of OFDM synchronizer. As cross correlation is accurate which is high on the demand list of synchronization is preferred over auto correlation. However autocorrelation based techniques were preferred only because of low hardware cost and low complex computations but was less accurate hence cross correlation was introduced. Further research is being done by implementing more correlators for the synchronization of OFDM where the complexity can be reduced along with the accuracy that cross correlator offered.

6. ACKNOWLEDGMENTS

The author would like to thank Nancy Ramanpreet, Gurinderpal Singh, Mohit Arora, Geetanjali Wasson of the ECE Department, Chandigarh University, Punjab, India for their contributions and support in her work

7. REFERENCES

[1] L. Hanzo and T. Keller , *OFDM and MC-CDMA: A Primer* (New York: Wiley, 2006).

[2] T. Fusco, M. Tanda, ML-based symbol timing and frequency offset estimation for OFDM systems with noncircular transmissions, *IEEE Trans. Signal Process.,vol. 54, no. 9, pp. 3527-3541, Sep. 2006.*

[3] J.J. Beek, M. sandell, M. Isaksson and P.O. Borjesson, Low complex frame synchronization in OFDM systems, *IEEE Int. Conf. Univ. Personal Commun., pp. 982-986, Nov. 1995.*

[4] N. Lashkarian, S. Kiaei, class of cyclic based estimators for frequency offset estimation of OFDM systems, *IEEE Trans. Commun., vol. 48, no. 12, pp. 2139-2149, Dec. 2000.*

[5] T.H. Kim, I.C. Park, Low power and high accurate synchronization for IEEE 802.16d systems, *IEEE Trans. Very Large Scale Integr. (VLSI)Syst., vol. 16, no. 12, pp. 1620-1630, Dec. 2008.*

[6] C.N. Kishore, V.U. Reddy, A Frame synchronization and frequency offset estimation algorithm for OFDM system and its analysis, *EURASIP J. Wirel. Commun. Netw., vol. 2006, pp. 1-16, Mar. 2006.*

[7] K. Wang, J. Singh, M. Faulkner, FPGA implementation of an OFDM-WLAN synchronizer, *IEEE Int. Workshop Electron. Des. Test appl., pp. 89-94, Jan. 2004.*

[8] A. Fort, J.W. Weijers, V. Derudder, W. Eberle, A. Bourdoux, A performance and complexity comparison of auto correlation and cross correlation for OFDM burst synchronization, *IEEE Int. Conf. Acoust. Speech Signal Process., vol. 2, pp. 341-344, 2003*

[9] T.H.Pham, S.A.Fahmy, Low power correlation for IEEE 802.16 OFDM Synchronization on FPGA, *IEEE Transactions On Very Large Scale Integration (VlSI) Systems, vol. 21, no. 8, Aug.2013.*