A Modified FFT Algorithm for OFDM Based Wireless System

G. Harish Kumar[1], Mahesh kusuma[2]

[1,2] Asst.Prof

Abstract-The limited available spectrum and the inefficiency in the spectrum usage in a fixed spectrum assignment policy, demands a new communication prototype to exploit the existing wireless spectrum opportunistically. This new networking paradigm is referred to as next generation networks as well as Dynamic Spectrum Access (DSA) and cognitive radio networks. The Fast Fourier Transform (FFT) and its inverse (IFFT) are very important algorithms in signal processing, software-defined radio, and the most promising modulation technique i.e. Orthogonal Frequency Division Multiplexing (OFDM). From the standard structure of OFDM we can find that IFFT/FFT modules play the vital role for any OFDM based transceiver. So when zero valued inputs/outputs outnumber nonzero inputs/outputs, then general IFFT/FFT algorithm for OFDM is no longer efficient in term of execution time. It is possible to reduce the execution time by "pruning" the FFT. In this paper we have implemented a novel and efficient input zero traced FFT pruning (IZTFFTP) algorithm based on DIF radix-2 technique. Compare to other algorithms, the results of IZFFTP shows that it is independent of the position of the zero valued input and also maintaining a good trade-off between time and space complexity of any system by not only reducing the number of complex multiplication as well as complex additions also. The proposed algorithm is implemented in high level computer program i.e. in C++and this is similar to the Cooley-Tukey radix-2 FFT algorithm, retaining all the key features such as simplicity and regularity, by making some alternation and programming modification .

Index Terms— Cognitive radio, OFDM, FFT, Pruning Techniques, Execution time.

I. INTRODUCTION

FFT is an efficient tool in the fields of signal processing and linear system analysis. DFT isn't generalized and utilized widely until FFT was proposed. But the inherent contradiction between FFT's spectrum resolution and computational time consumption limits its application. To match with the order or requirement of a system, the common method is to extend the input data sequence x(n) by padding number of zeros at the end of it and which is responsible for a increased value of computational time. But calculation on undesired frequency is unnecessary. As the OFDM based cognitive radio [1] has the capability to nullify individual sub carriers to avoid interference with the licensed user. So, that there could be a large number of zero valued inputs/outputs compare to non-zero terms. This is the most important thing in the orthogonal frequency division multiplexing (OFDM), which is used as baseband transmission in spectrum pooling technique. Though large bandwidth supports high data rates but practically it is impossible to find contiguous empty bandwidth. So much efficient data rates are achieved by using non-contiguous vacant subcarriers of a targeted spectrum pool. This type of OFDM is known as non-contiguous OFDM or NC-OFDM [2-3], which helps to avoid the harmful interference by deactivating those subcarriers, which are acquired by different licensed users. That means the input values of the IFFT‟s of those particular subcarriers is zero. As NC-OFDM consists of large number of de-activated or null subcarrier i.e. numbers of zero valued inputs/outputs outnumber non-zero inputs/outputs. So the conventional FFT is no longer efficient in terms of complexity, execution time and hardware architecture. Several researchers have proposed different way to make FFT faster by “pruning” the conventional one [4].

In this paper we have proposed an input zero traced FFT pruning (IZTFFTP) algorithm, suitable for NC-OFDM based transceiver. It is based on normal Cooley-Tukey radix-2 DIF algorithm using matrix factorising process. Result shows IZTFFTP is more efficient than ordinary FFT. This paper is organised in the following subsections: section-II discusses related researches in this topic, section-III contains structure of NC-OFDM, section-IV describes about general FFT algorithm (using matrix factorisation technique), section-V generalizes the proposed technique for pruning (including flowchart), results and concluding remarks are presented in section-VI&VII respectively.

II. RELATED RESEARCH

As FFT have some major application in the field of signal processing, many researches are still going on to make it more efficient and flexible in terms of system requirement. In order to decrease the computation time of general FFT algorithm, several pruning algorithm have been proposed. In 1971 J.D Markel proposes first FFT pruning algorithm [4].It was based on DIF FFT model. Following the similar way Skinner [5] has proposed an algorithm based on DIT FFT model using the inputs points only. Then ,comprising both the input & output pruning Rao & Srinivas have proposed another algorithm[6].In 2000 R.G.Alves gave an idea for general FFT pruning algorithm[7].A different view of FFT algorithm, Transform decomposition method have been proposed by Sorensen & Burrus [8].TD method generally a modified version of CTFFT algorithm, there DFT is decomposed into two smaller DFTs. Though it is complex but in terms of hardware implementation TD is more efficient and flexible rather than pruning.

From all of these papers we get some idea about the needs of a suitable pruning technique and their approach to prune a FFT. Most of them is based on the formation of a matrix (except TD), consists of N rows &log $_2$N columns. But this generation of an assistant matrix itself is a complicated and time consuming process and hard to understand also. To get rid-off from these Inefficiencies we have proposed IZTFFTP algorithm, which is the modified version of general Coole-Tukey algorithm (using matrix factorization) [9], by simple changing in the programming approach.

III. OFDM STRUCTURE

OFDM is a digital multi carrier modulation technique. OFDM uses a large number of closely spaced orthogonal subcarrier. In This subsection we will discuss about the OFDM framework and its generation technique using the general schematic in the figure 1-(A, B).Basically in DSA network, it is not practically possible to find a contiguous block of spectrum, which are being utilised fully. So by employing the dynamic spectrum sensing and channel estimation technique, the subcarriers that are accessed by the licensed user are turned off. The OFDM based transceivers those are capable to deactivate the used subcarriers are known as NC-OFDM based transceivers [3].Using Fig.1 we will discuss about the internal architecture of this complex system step by step.

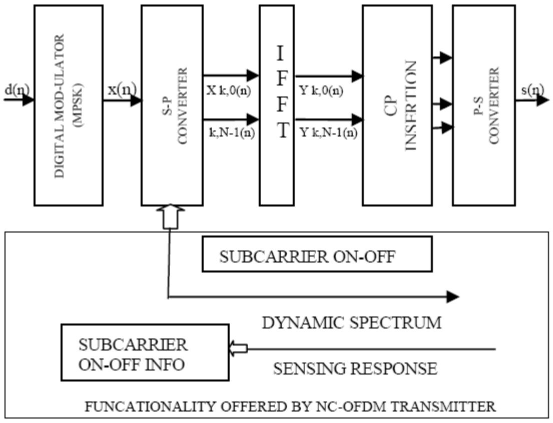

Fig.1-A: General NC-OFDM transmitter

Let x(n) is the modulated version of the input data stream d(n).Modulation has been done by using any digital modulation scheme. Then the modulated data stream is divided into N separate data streams by using S-P converter. Each of the streams is transmitted through those orthogonal subcarriers and after summing them we will get the composite OFDM signal. As we discussed earlier that NC-OFDM transceiver contains many deactivated subcarriers that are used by the licensed users. Those signals are also transmitted to the receiver but have no significant contribution in the IFFT/FFT computation. To reduce the inter symbol interference one cyclic prefix (CP) block is added to the symbol and this CP works as guard interval between two different symbol. Then it is passed through a P-S converter. Finally the signal is upsampled and passed through a D/A converter for converting it into analog signal s(n) and transmitted through the RF link.

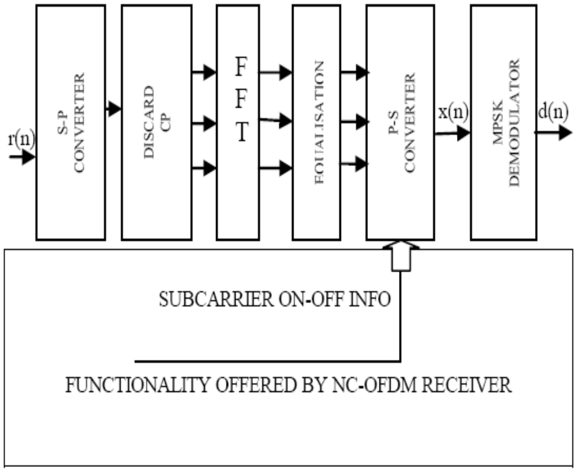

Fig.1-B: General NC-OFDM receiver

At the receiver s(n) is converted to digital signal r(n) by passing through an A/D converter. Then the signal is passed through an S-P converter to make parallel data stream .After this CP is discarded and fed that signal to the FFT block to transform the time domain data into frequency domain. Finally to obtain the original signal, multiplexing using P-S converter and demodulation has been done.

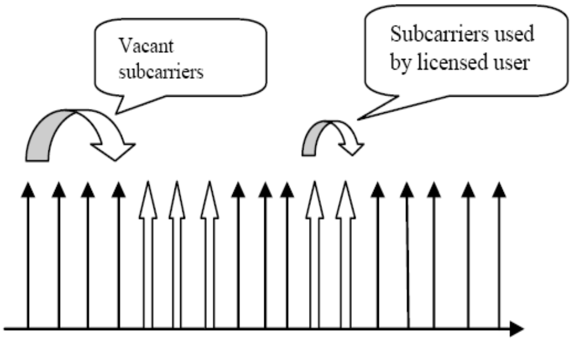

Fig.2: subcarrier distribution in NC-OFDM system

IV. GENERAL FFT ALGORITHM

The FFT/IFFT is the most critical part of any signal processing system. Here also in

OFDM based transceiver those are the most computational intensive blocks of the total system. So an inefficient IFFT/FFT may decrease the overall system response. This paper is based on radix-2 DIF FFT algorithm, which have a divide and conquer approach, where N-points DFT is decomposed into successively small DFTs(with odd and even part separately).

The basic formula for DFT is-

$$X(n) = \sum_{K=0}^{N-1} Xo(K)\, e^{-\frac{2j\pi nk}{N}} \qquad n=0, 1, 2 \ldots N\text{-}1 \text{ ----------- } (1)$$

N is the order,for example if N= 4 then we can write the eqn-1 in a matrix form as-

$$\begin{pmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{pmatrix} = \begin{pmatrix} Wo & Wo & Wo & Wo \\ Wo & W1 & W2 & W3 \\ Wo & W2 & W4 & W6 \\ Wo & W3 & W6 & W9 \end{pmatrix} \begin{pmatrix} xo(0) \\ xo(1) \\ xo(2) \\ xo(3) \end{pmatrix} \text{ --------------- } (2)$$

More compactly as $X(n) = W^{nk}\, Xo(K)$ -------------------- (3)

As W (represent twiddle factor) and Xo(K) are complex, so to solve the eqn (3) , complex multiplication and N(N-1) complex addition is required. But FFT algorithm reduces the computational time by reducing the total number of complex multiplications and additions.

To illustrate FFT algorithm, conveniently I take N=, where
□ is an integer and rewriting
the eqn (2) as

$$\begin{pmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & W1 & W2 & W3 \\ 1 & W2 & Wo & W2 \\ 1 & W3 & W2 & W1 \end{pmatrix} \begin{pmatrix} xo(0) \\ xo(1) \\ xo(2) \\ xo(3) \end{pmatrix} \text{ ---------------- } (4)$$

So now by factoring the matrix (4) and changing the order of the rows(separate odd and even part) :

$$\begin{pmatrix} X(0) \\ X(2) \\ X(1) \\ X(3) \end{pmatrix} = \begin{pmatrix} 1 & Wo & 0 & 0 \\ 1 & W2 & 0 & 0 \\ 0 & 0 & 1 & W1 \\ 0 & 0 & 1 & W3 \end{pmatrix} \begin{pmatrix} 1 & 0 & Wo & 0 \\ 0 & 1 & 0 & Wo \\ 1 & 0 & W2 & 0 \\ 0 & 1 & 0 & W2 \end{pmatrix} \begin{pmatrix} xo(0) \\ xo(1) \\ xo(2) \\ xo(3) \end{pmatrix} \text{ ---- } (5)$$

Though it is a complex process to understand but it is easy to implement in a computer program. For this reason we preferred this method of FFT computation.So,FFT algorithm is a simple factoring procedure of N×N matrix into ▱ matrices(each N×N). Fot larger value of N,we need to interpret this matrix factorization process into a graphical manner which will help to build up a flowchart for an efficient computer program. We have shown a graphical view in fig-(3) where N=8.

The total flow graph is based on two parent equations, that are-

$$x_l = x_{l-1}(K) + W^P x_{l-1}\left(K + \frac{N}{2^l}\right) \text{ ------------- } (6)$$

$$x_l\left(K + \frac{N}{2^l}\right) = x_{l-1}(K) - W^P x_{l-1}\left(K + \frac{N}{2^l}\right) \text{ -------------- } (7)$$

Where l = no. Of array in the flow graph, and W^P is the twiddle factor.

Those two points x_l & $x_l\left(K + \frac{N}{2^l}\right)$, spaced by N/,are known as dual node to each other. Every dual node needs single calculation as both the equations have same input stream and the value of twiddle factor is also same. The only difference is the polarity W^P of the as $W^P = -W^{P+n/2}$ .Using this process if we compute FFT for that particular example (N=4),

then

- ⦿ = „0‟ operation [full pruning]
- ● = single operation [partial pruning]
- ○ = full butterfly operation [no pruning]

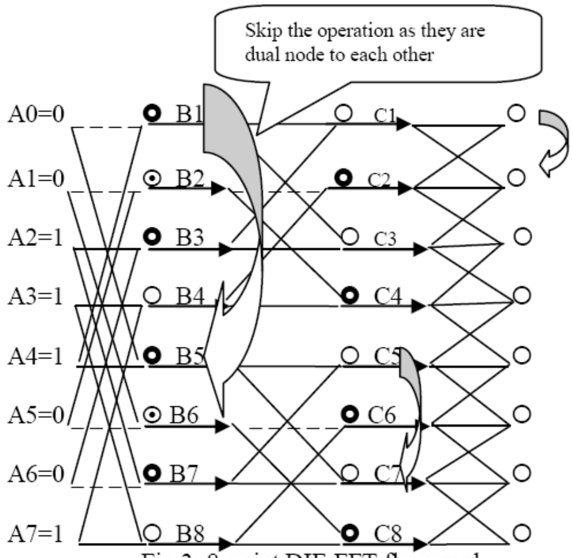

Fig.3: 8-point DIF-FFT flowgraph

Only N □/2=4 complex multiplication, and N□=8 complex addition is needed, which is remarkably less than direct DFT method. But still conventional FFT will work inefficiently when a large number of zero valued inputs are present compare to non-zero inputs. To alleviate this inefficiency in this paper I have proposed some techniques and modify the standard flowchart [9]. Also I have proposed a new flow graph [shown in the Fig-8] and implement that into a computer program.

V. PROPOSED PRUNING TECHNIQUES

To increase the efficiency of the FFT technique several pruning and different other techniques have been proposed by many researchers. In this paper, we have implemented a new pruning technique i.e. IZTFFTP by simple modification and some changes in the conventional flowchart of FFT [9] and also includes some tricky mathematical techniques to reduce the total execution time.

□ ***Zero tracing-*** as in wide band communication system a large portion of frequency channel may be unoccupied by the licensed user, so no. of zero valued inputs are much greater than the non-zero valued inputs in a FFT/IFFT operation at the transceiver. Then this algorithm will give best response in terms of reduced execution time by

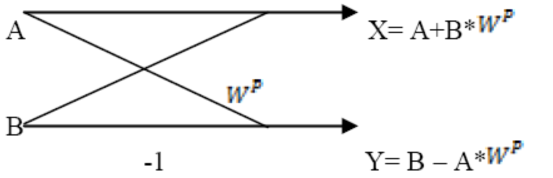

Fig .4: small butterfly unit of a dual node pair

reducing the no. of complex computation required for twiddle factor calculation. IZTFFTP have a strong searching condition, which have a 2-D array for storing the input & output values after every iteration of butterfly calculation. In a input searching result whenever it found „zero" at any input, simply omit that calculation by considering following two useful condition:

□ ***Half Butterfly computation or partial pruning***-The basic computation part of FFT is the butterfly calculation. From fig.3 we pick a single part of a standard butterfly unit. In the fig.4 A & B two complex inputs are dual node to each other .A full butterfly calculation requires 4 complex multiplications and 6 complex additions/subtractions [4].

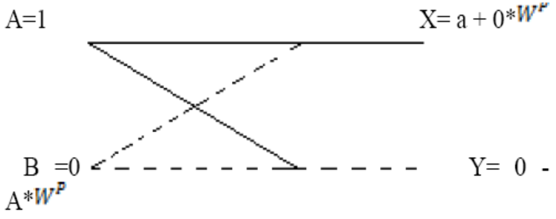

Fig.5: partial pruning structure

□ ***"0" operation or complete pruning-*** Fig.6 shows the part where both the input of the dual node pair is zero. then outputs obtained from the mathematical calculation of equation 6-7 is also zero.

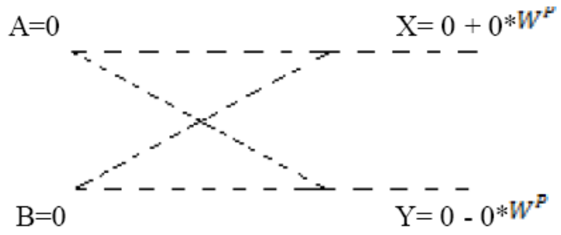

Fig.6: complete pruning structure

In such cases where number of zero is remarkably very large IZTFFTP works most effectively and program will automatically goes to the next node for required computation omitting these unnecessary complex calculations.

In this way the total no of complex multiplications and additions are reduced remarkably as well as the execution time also. In the next result section we will find how the new technique gives better response with respect to general FFT algorithm.

VI. RESULTS

In general most of the FFT or pruning algorithms are literally proposed. Some of them are implemented either in MATLAB or in FORTRAN. The main constraints of those techniques are that, they are not dynamically efficient enough for any type of input dataset. It is very rare to find an algorithm implemented in high level computer program, which is able to show the required actual execution time for an FFT operation. Here we have implemented a FFT pruning technique in high level computer language i.e. in C++ and execute it in Linux platform .To check the FFT results, before pruning operation we have simulated the core FFT code based on matrix factorization process, in a DSP environment i.e. VDSP++, which is the programming evaluation window of ADSP BF533 EZ-KITLITE DSP starter kit. The flowchart of the program has been shown in figure-8.

Table-1 contains the assumed input data set, stored in a 2-D array, considering only real values and as this is only for an NC-OFDM system so the number of possible unlicensed user is less than the total no of subcarriers.

Input serial	Real part	Imaginary part
a [0]	1	0
a[1]	1	0
a[2]	0	0
a[3]	0	0
a[4]	0	0
a[5]	0	0
a[6]	0	0
a[7]	1	0

Table-1: 2-D array's input in tabular form

Assuming a[0], a[1] & a[7], these three inputs needs to be computed in a single FFT operation of order 8.

In table-2, the comparison of the outputs have been shown assuming 1024 no of subcarriers are there in that OFDM system.

Order of FFT	No. Of CM*		No. Of CA*		execution time(mili seconds)	
	Ordinary FFT	Pruned FFT	Ordinary FFT	pruned FFT	ordinary FFT	pruned FFT
8	12	11	24	7	0.55	0.47
16	32	25	88	23	0.76	0.68
32	80	62	248	83	2.51	0.80
64	192	132	632	207	9.41	1.80
128	448	284	1528	511	21.02	2.65
256	1024	584	3576	1127	77.02	3.91
512	2304	1208	8185	2471	306.88	6.44
1024	5120	2448	18424	5175	1204.32	13.97

Table-2: outputs comparison of two different algorithms

Output shows the significant reduction of computational complexity by reducing the total no. of complex operation i.e. both the multiplications and additions compare to the ordinary FFT operation. Also we have shown the compared value of the reduced execution time for two different algorithms (with pruning and without pruning) using a same configured (2.10GHz, Intel core2 duo CPU) computer system.

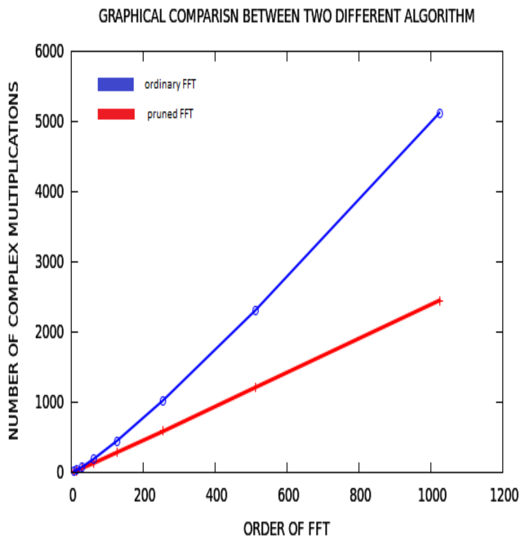

Fig.-7: graphical comparison between two different algorithms.

Using Linux GNU plot we have shown the graphical comparison of the output responses in Fig.-7, where X-axis represent the order of FFT and Y-axis for the number of complex multiplications.

VII. CONCLUSIONS

In summary through this paper we have proposed and implemented a novel technique of FFT pruning computation in high level computer program by modifying the in place computation technique in the conventional FFT algorithm [9].Though most of the Researchers did not give so much importance in the number of additions but it also have a significant effect in case of hardware implementation (FPGA). Results shows IZTFFTP is much efficient than ordinary FFT algorithm as it takes very less time to compute where number of Zero valued inputs/outputs are greater than the total number of non Zero terms, with maintaining a good trade-off between time and space complexity, and it is also independent to any input data sets.

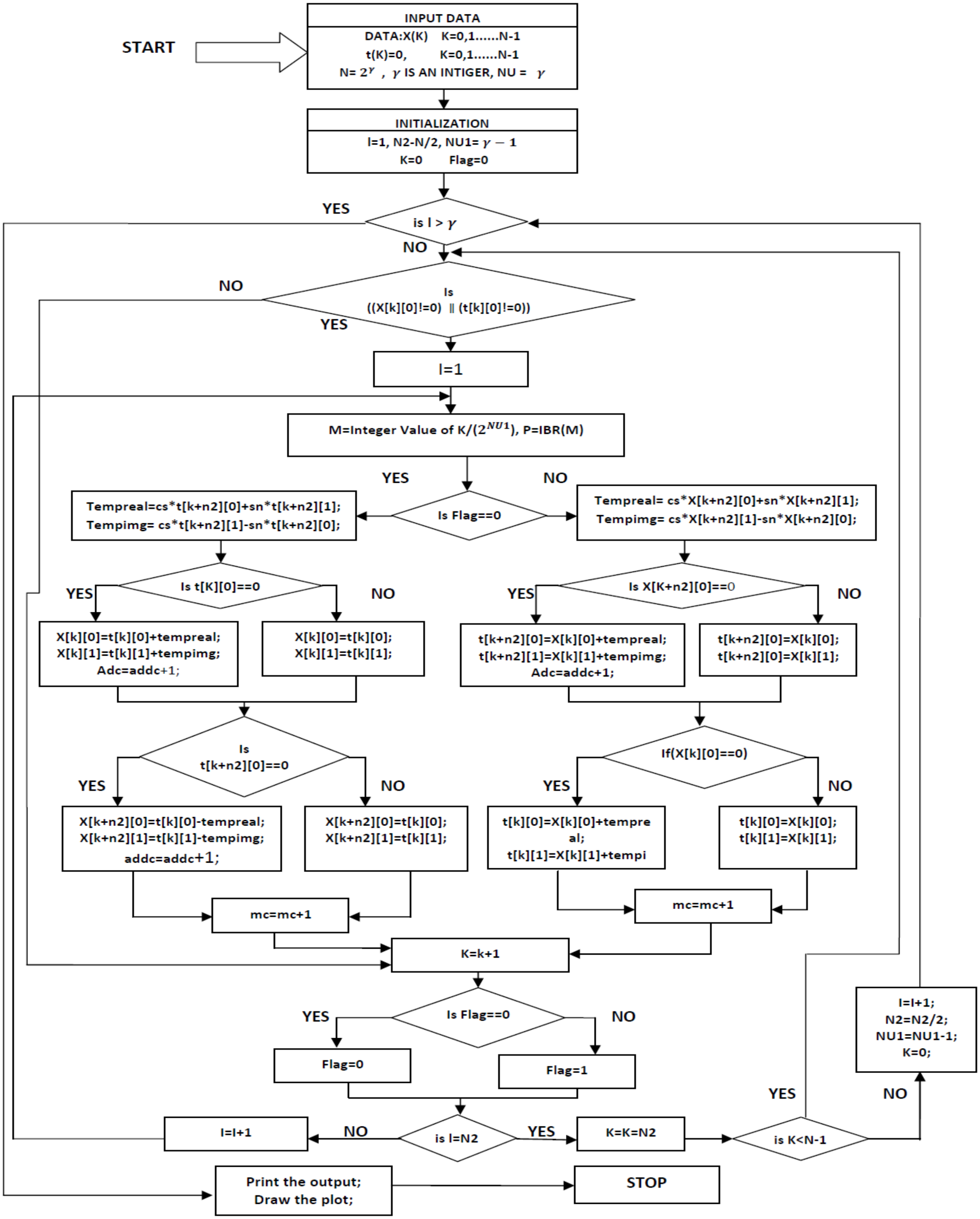

Fig.8- Programming Flowchart of the IZTFFTP algorithm

REFERENCES

[1] J. Mitola, III, "Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio," Thesis (PhD), Dept. of Teleinformatics, Royal Institute of Technology (KTH), Stockholm Sweden, May 2000.

[2] J. D. Poston and W. D. Horne, “Discontinuous OFDM considerations for dynamic spectrum access in idle TV channels,” inProc. IEEE Int. Symp. New Frontiers Dynamic Spectra. Access Networks, vol. 1, (Baltimore, MD, USA), pp. 607–610, Nov.

[3]R. Rajbanshi, A. M. Wyglinski, and G. J. Minden, Cognitive Radio Communication Networks, ch. 5. Springer-Verlag, 2007.

[4]J. D. Markel, “FFT Pruning,” IEEE Trans. Audio Electroacoust., vol. 19, pp. 305 – 311, Dec. 1971.

[5] D. P. Skinner, “Pruning the Decimation in time FFT algorithm,” in *Proc. IEEE Int. Conf.Acoust., Speech, Signal Process.*, vol. 24, Apr. 1976, pp.193 – 194

[6]T. V. Sreenivas and P. Rao, “FFT algorithm for both input and output Pruning,” in *Proc.IEEE Int. Conf. Acoust., Speech, Signal Process,* vol. 27, June 1979, pp. 291 – 292.

[7]R. G. Alves, P. L. Osorio, and M. N. S. Swamy, “General FFT Pruning Algorithm,” in *Proc.43rd IEEE Midwest Symp. Circuits and Systems*, vol. 3, Aug. 2000, pp. 1192 – 1195.

[8] H. V. Sorensen and C. S. Burrus, “Efficient computation of the DFT with only a subset of input or output points,” *IEEE Trans. Signal Processing*, vol. 41, pp. 1184 – 1200, Mar. 1993.

[9] E. Oran. Brigham, "The Fast Fourier Transform and Its Applications" Prentice Hall Publication, 1988, ISBN: 0133075052.