ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

FPGA Implementation of FFT Algorithm for IEEE 802.16e (Mobile WiMAX)

G.Thrinath Reddy
BE (Osmania University),
Methodist College of Engineering & Technology,
Hyderabad.

P.Babu Khan
BE (Osmania University),
Methodist College of Engineering & Technology,
Hyderabad.

Abstract:

Mobile WiMAX (Worldwide Interoperability for Microwave Access) or 802.16e standard was ratified by theIEEE in late 2005 as a potential to emerge as a real viable competitor to existing 3G technologies. Mobile WiMAX uses an OFDMA™ technology called 1K-FFT. Orthogonal Frequency-Division Multiple Access (OFDMA) is a multi-user version of the popular Orthogonal frequency -division multiplexing (OFDM) digital modulation scheme. In the widely used OFDM systems, the FFT and IFFT pairs are used to modulate and demodulate the data constellation on the sub-carriers. This paper presents a high level implementation of a high performance FFT for OFDM Modulator and Demodulator. The design has been coded in Verilog and targeted into Xilinx Spartan3 FPGAs. Radix-22 Algorithm is proposed and used for the OFDM communication system. This algorithm has the same multiplicative complexity as the radix-4 algorithm, but retains the butterfly structure of radix-2 algorithm.

Index Terms:

Radix 22algorithm, Fast Fourier Transform,Orthogonal Frequency Division Multiplexing, Mobile WiMAX.

I. INTRODUCTION:

The true Mobile WiMAX standard of 802.16e is divergent from Fixed WiMAX. While clearly based on the same OFDM base technology adopted in 802.16-2004, the 802.16e version is designed to deliver service across many more sub-channels than the OFDM 256-FFT. It is important to note that both standards support single carrier, OFDM 256-FFT and at least OFDMA 1K-FFT. OFDM technology is used for many communication systems such as Asymmetric digital subscriber line (ADSL), Wireless Local Area Network (WLAN) or Multimedia Communication Services [1]. One of the key components in OFDM system is the Fast Fourier Transform (FFT).

There are more and more communication systems require higher points FFT and higher symbol rates. The requirement establishes challenges for low power and high speed FFT design with large points.The FFT algorithm eliminates the redundant calculation which is needed in computing Discrete Fourier Transform (DFT) and is thus very suitable for efficient hardware implementation [2]. In addition to computing efficient DFT, the FFT also finds applications in linear filtering, digital spectral analysis and correlation analysis, Ultra Wide Band (UWB) applications, etc. A hardware oriented radix-22 algorithm [3] is developed by integrating a twiddle factor decomposition technique in divide and conquer approach to form a spatially regular Signal Flow Graph (SFG). Mapping the algorithm to the cascading delay feedback structure leads to the proposed architecture [4].The next section describes architecture & design methodology, followed by its implementation in VERILOG Hardware Description Language (VERILOG) code and utilization, performance and implementation in OFDM systems. Finally we conclude with a comparison of hardware requirement of R22SDF and several other popular pipeline architectures.

II.ARCHITECTURE AND DESIGN METHODOLOGY:

A. Radix-22 Decimation in Frequency FFT Algorithm :

A useful state-of-the-art review of hardware architectures for FFTs was given by He et al. [5] and different approaches were put into functional blocks with unified terminology. From the definition of DFT of size N [6]:

$$X(k)=\sum_{n=0}^{N-1}x(n)W_N^{nk},0\leq k<N \qquad (1)$$

whereWN denotes the primitive Nth root of unity, with its exponent evaluated modulo N, x(n) is the input sequence and X(k) is the DFT. He [5] applied a 3-dimensional linear index map,

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

$$n=<\frac{N}{2}n_1+\frac{N}{4}n_2+n_3>_N$$
$$K=<K_1+2K_2+4K_3>_N \qquad (2)$$

and common factor algorithm (CFA) to derive a set of 4 DFTs of length $\frac{N}{4}$ as,

$$X(K_1+2K_2+4K_3)=$$
$$\sum_{n_3=0}^{\frac{N}{4}-1}[H(k_1,k_2,n_3)W_N^{n_3(k_1+2k_2)}]W_{\frac{N}{4}}^{n_3k_3} \qquad (3)$$

where n 1,n2,n3 are the index terms of the input sample n and k1,k2,k 3 are the index terms of the output sample k and where H(k1,k2,k3) is expressed in eqn (4).

$$H(k_1,k_2,n_3)=[x(n_3)+(-1)^{k_1}x(n_3+\frac{N}{2})]+$$
$$(-j)^{(k_1+2k_2)}[x(n_3+\frac{N}{4})+(-1)^{k_1}x(n_3+\frac{3N}{4})] \qquad (4)$$

Eqn (4) represents the first two stages of butterflies with only trivial multiplications in the SFG, as BFI and BFII. Full multipliers are required after the two butterflies in order to compute the product of the decomposed twiddlefactor$WN^{n3(k12k2)}$ in eqn (3). Note the order of the twiddlefactors is different from that of radix-4 algorithm. Applying this CFA procedure recursively to theremaining DFTs of length N/4 in eqn (3), the complete radix-22 Decimation-in-frequency (DIF FFT) algorithm is obtained. The corresponding FFT flow graph for N=16 is shown in Fig. 1 where small diamonds represent trivialmultiplication by $WN^{N/4}$–j, which involves only real-imaginary swapping and sign inversion [5].

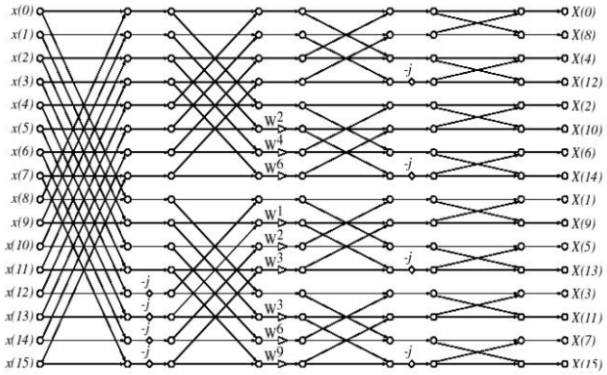

Fig 1. Radix-22 DIF FFT flow graph for N=16.

B. Radix-22 FFT Architecture:

Mapping radix-22 DIF FFT algorithm derived to the radix-2 SDF architecture, a new architecture of R22SDF approach is obtained [3].

Fig 2 outlines an implementation of the R22SDF architecture for N=1024, note the similarity of the data-path to R2SDF and the reduced number of multipliers. The implementation uses two types of butterflies; one identical to that in R2SDF, the other contains also the logic to implement the trivial twiddle factor multiplication, as shown in Fig 3 (i), (ii) respectively [3].Due to the spatial regularity of Radix-22 algorithm, the synchronization control of the processor is very simple. A (log2N)-bit binary counter serves two purposes: synchronization controller and address counter for twiddle factor reading in each stage. With the help of the butterfly structures shown in Fig 3, the scheduled operation of the R22SDF processor in Fig 2 is as follows.On first N/2 cycles, the 2-to-1 multiplexers in the first butterfly module switch to position “0”, and the butterfly is idle. The input data from left is directed to the shift registers until they are filled. On next N/2 cycles, the multiplexers turn to position “1”, the butterfly computes a 2-point DFT with incoming data and the data stored in the shift registers.

$$Z1(n)=x(n)+x(n+\frac{N}{2})$$
$$Z1(n+\frac{N}{2})=x(n)-x\left(n+\frac{N}{2}\right),0\le n<\frac{N}{2} \qquad (5)$$

The butterfly outputs Z1(n) and Z1(n + N/2) are computed according to the equations given in eqn (5). Z1(n) is sent to apply the twiddle factors, and Z1(n + N/2) is sent back to the shift registers to be “multiplied” in still next N/2 cycles when the first half of the next frame of time sequence is loaded in. The operation of the second butterfly is similar to that of the first one, except the “distance” of butterfly input sequence are just N/4 and the trivial twiddle factor multiplication has been implemented by real-imaginary swapping with a commutator and controlled add/subtract operations, as in Fig 3(i) (ii), which requires two bit control signal from the synchronizing counter. The data then goes through a full complex multiplier, working at 75% utility, accomplishes the result of first level of radix-4 DFT word by word. Further processing repeats this pattern with the distance of the input data decreases by half at each consecutive butterfly stages. After N-1 clock cycles, the result of the complete DFT transform streams out to the right, in bit-reversed order. The next frame of transform can be computed without pausing due to the pipelined processing of each stage.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

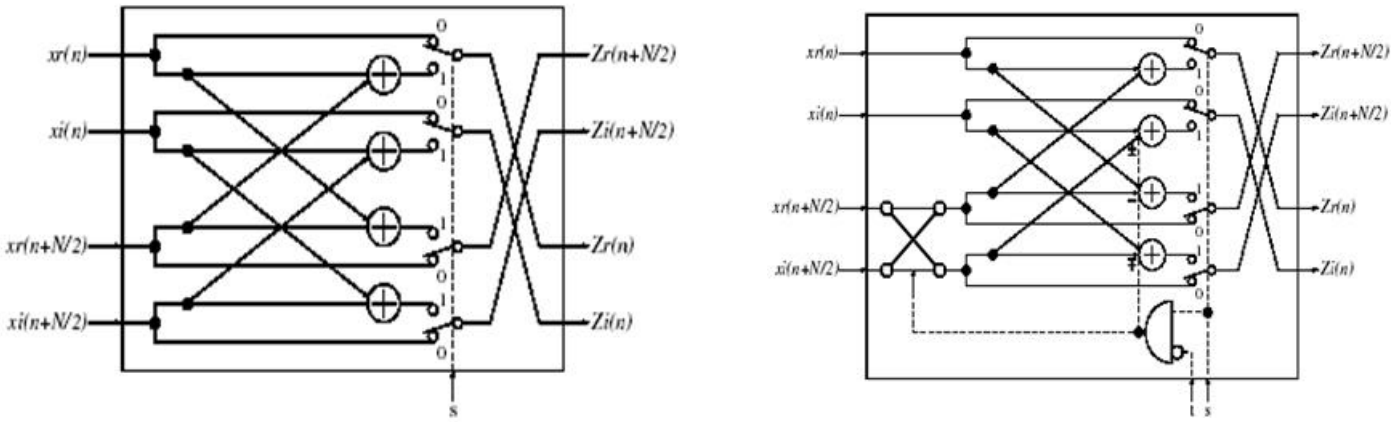

Fig 2. Butterfly structure for R22SDF FFT

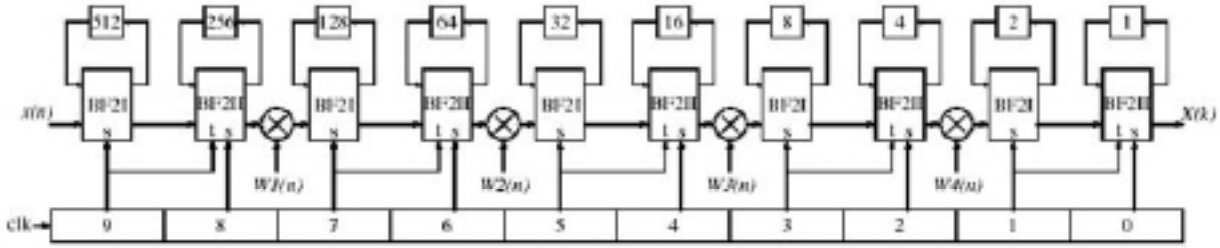

Fig 3. R22SDF pipeline FFT architecture for N=1024

III. INTRODUCTION TO VERILOG:

VERILOG Hardware Description Language (VERILOG) was introduced by Gateway Design Automation in 1984 as a proprietary hardware description and simulation language [7]. The introduction of Verilog-based synthesis tools in1988 by then-fledging Synopsys and the 1989 acquisition of Gateway by Candence Design Systems were important events that led to wide-spread use of the language [7].VERILOG synthesis tools can create logic-circuit structures directly from VERILOG behavioral descriptions, and target them to a selected technology for realization. Using VERILOG, you can design, simulate, and synthesize anything form a simple combinational circuit to a complete microprocessor system on a chip.VERILOG started out with and still has the following features [7]:

1)Designs may be decomposed hierarchically.
2)Each design element has both a well-defined interface and a precise functional specification.
3)Functional specifications can use either a behavioral algorithm or an actual hardware structure to define initially by an algorithm, to allow design verification of higher level elements that use it; later, the algorithmic definition can be replaced by a preferred hardware structure.
4)Concurrency, timing, and clocking can all be modeled. VERILOG handles asynchronous as well as synchronous sequential-circuit synthesis.
5)The logical operation and timing behavior of a design can be simulated.

Thus, VERILOG started out as a documentation and modeling language, allowing the behavior of digital-system designs to be precisely specified and simulated. The VERILOG language specification allows multiple modules to be stored in a single text file. When one VERILOG module instantiates another, the compiler finds the other by searching the current workspace, as well as predefined libraries, for a module with the instantiated name. Thus, when using VERILOG-1995, there should be only one definition of each module, usually in a file with the same name as the module. However, VERILOG-2001 actually allows you to define multiple versions of each module, and it provides a separate configuration management facility that allows you to specify which one to use for each different instantiation during a particular compilation or synthesis run. This lets you try out different approaches without throwing away orrenaming your other efforts. All these features of VERILOG will help better in simulation and synthesis of our proposed architecture.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

IV. IMPLEMENTATION IN VERILOG:

The R22SDF presented above has been fully coded in VERILOG Hardware Description Language (VERILOG). Once the design is coded in VERILOG, the Modelsim XEIII 6.2c compiler [8] and the Xilinx Foundation ISA Environment 9.1i [9] generate a net-list for FPGA configuration. The net-list can then be downloaded into the FPGA using the same Xilinx tools and Texas Instruments prototyping board.From the architecture of R22SDF in Fig 2, the butterfly blocks BF2I and BF2II are described as building blocks in VERILOG code. Booth multiplication algorithm for signed binary numbers is used for complex multipliers.

Thus, the overall latency of the real implementation varies as the processing word length changes [3]. Look-up-table (LUT) based Random Access Memories (RAMs) and Flip-Flops are used to implement feedback memory of the very last stages where are the RAM blocks in the FPGA are used for the rest of the stages. Similarly, LUT-based Read Only Memories (ROMs) are used to implement twiddle ROMs of the very last stages whereas Block RAMs are used for the rest of stages [5]. The FFT is heavily pipelined to achieve as highest clock frequency as possible.

Twiddle factors are generated by an external program and embedded to the VHDL code.The implementation results after implementing in Xilinx Spartan3 FPGA (fig 4) [10] are listed in Table 1(a), 1(b). Table 1(a) shows the implementation results where as table 1(b) shows the timing summary.The resulting figures show that our implementation outperforms the other implementations of that kind. Its speed nearly matches that of the Xilinx core but its throughput is more than 3 times higher due to its pipeline nature.

TABLE 1(a) IMPLEMENTATION RESULT:

Logic Utilization	Used	Available	Utilization
No. of Slices	3155	3584	88%
No. of Slice Flip flops	1514	7168	21%
No. of 4 input LUTs	5916	7168	82%
No. of bonded IOBs	32	97	32%
No. of Mult18x18s	16	16	100%
No. of GCLKs	1	8	12%

TABLE 1(b) TIMING SUMMARY:

Minimum period	10.827ns (Maximum Frequency: 92.366MHz)
Minimum input arrival time before clock	5.406ns
Maximum output required time after clock	6.216ns

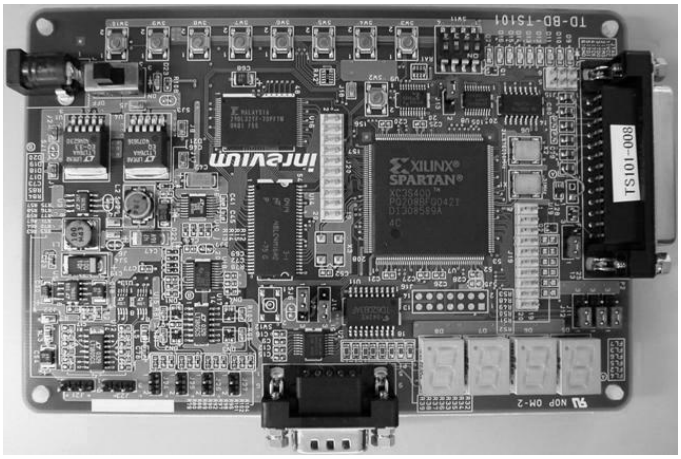

Fig 4. Xilinx Spartan 3 FPGA Kit

V. APPLICATIONS OF PROPOSED FFT IN IEEE 802.16E COMMUNICATION STANDARD:

Worldwide Interoperability for Microwave Access (WiMAX) [11, 12], is a telecommunications technology thatprovides wireless transmission of data using a variety of transmission modes, from point-to-multipoint links to portable and fully mobile internet access. The technology provides up to 10 Mbps broadband speed without the need for cables. The technology is based on the IEEE 802.16 standard (also called Broadband Wireless Access). The 802.16e standard [12] uses OFDMA with 2K-FFT, 1K-FFT, 512-FFT, and 256-FFT capability. The fundamental principle of the OFDM system is to decompose the high rate data stream (bandwidth = W) into N lower rate data streams and then to transmit themsimultaneously over a large number of subcarriers [13]. The IFFT and the FFT are used for, respectively, modulating and demodulating the data constellations on the orthogonal subcarriers [14]. In an OFDM system, the transmitter and receiver blocks contain the FFT modules as shown in Fig 5(a) & (b). The FFT processor must finish the transform within 312.5ns to serve the purpose in the OFDM system. Our FFT architecture effectively fits into the system since it has a minimum required time period of 10.827ns (table 1(b)).

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

An OFDM carrier signal is the sum of a number of orthogonal sub-carriers, with baseband data on each sub-carrier being independently modulated commonly using some type of quadrature amplitude modulation (QAM) or phase-shift keying (PSK) [15]. This composite baseband signal is typically used to modulate a main RF carrier. s[n] is a serial stream of binary digits. By inverse multiplexing, these are first demultiplexed into N parallel streams, and each one mapped to a (possibly complex) symbol stream using some modulation constellation (QAM, PSK, etc.). Note that the constellations may be different, so some streams may carry a higher bit-rate than others.The receiver picks up the signal r(t), which is then quadrature-mixed down to baseband using cosine and sine waves at the carrier frequency. This also creates signals centered on 2fc, so low-pass filters are used to reject these. The baseband signals are then sampled and digitized using analogue-to-digital converters (ADCs), and a forward FFT is used to convert back to the frequency domain [15].

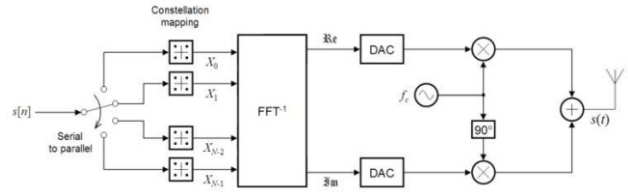

Fig 5(a) OFDM Transmitter

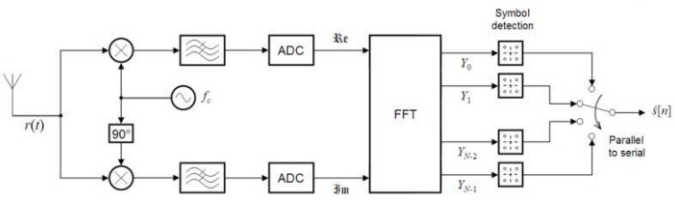

Fig 5(b) OFDM Receiver

Orthogonal Frequency-Division Multiple Access (OFDMA) is a multi-user version of the popular Orthogonal frequency-division multiplexing (OFDM) digital modulation scheme. Multiple access is achieved in OFDMA by assigning subsets of subcarriers to individual users as shown in fig 6. This allows simultaneous low data rate transmission from several users [15].

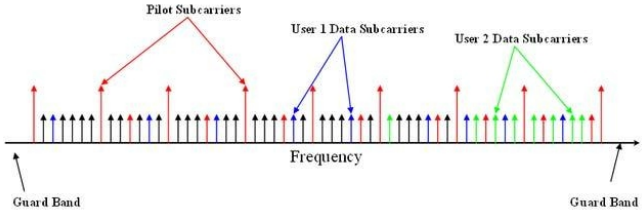

Fig 6. ODMA Subcarriers Pattern

OFDMA can be seen as an alternative to combining OFDM with time division multiple access (TDMA) or time-domain statistical multiplexing, i.e. packet mode communication. Low-data-rate users can send continuously with low transmission power instead of using a "pulsed" high-power carrier. Constant delay, and shorter delay, can be achieved. OFDMA can also be described as a combination of frequency domain and time domain multiple access, where the resources are partitioned in the time-frequency space, and slots are assigned along the OFDM symbol index as well as OFDM sub-carrier index. OFDMA is considered as highly suitable for broadband wireless networks, due to advantages including scalability and MIMO-friendliness, and ability to take advantage of channel frequency selectivity [15]. The trans-receiver structure of OFDMA is shown in fig 7.It shows that the proposed design has less memory access than the radix-4 FFT by 20 ~ 40%. Therefore, the proposed architecture consumes much lower power.

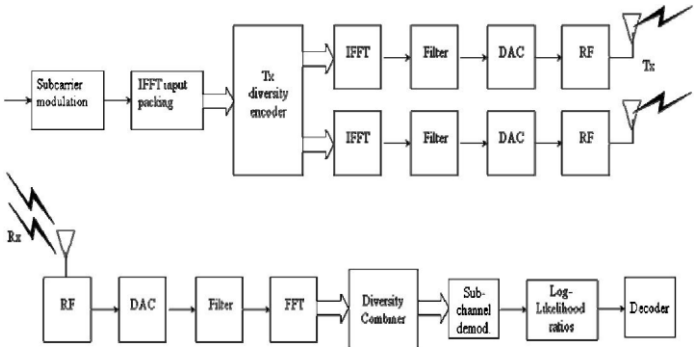

Fig 7. OFDMA Trans-receiver

VII. PERFORMANCE AND IMPLEMENTATION:

A. Hardware Requirement

The radix-4 butterfly needs 3 complex adders and 1 complex multiplier[16], while the proposed butterfly structure needs only 4 complex adders and 1 complex multiplier. This is because our design implements the constant multiplier by 4 reused complex adders. Fig. 8 shows memory addressing of SRAM0 and SRAM1 for N=64. All of the above-mentioned use separated single SRAM into 2 smaller SRAMs. This design can double SRAM throughput with inter-leaving access. In table 2, the hardware requirement of the proposed design is compared with various pipelined designs.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

TABLE 2:HARDWARE REQUIREMENT-COMPARISON:

	Multiplier #	Adder #	Memory size
R2MDC [17]	$2(\log_4 N-1)$	$4\log_4 N$	$3N/2-2$
R2SDF [18]	$2(\log_4 N-1)$	$4\log_4 N$	N-1
R4SDF [19]	$\log_4 N-1$	$8\log_4 N$	N-1
R4MDC [17]	$3(\log_4 N-1)$	$8\log_4 N$	$5N/2-4$
R4SDC [20]	$\log_4 N-1$	$3\log_4 N$	2N-2
Proposed	$\log_4 N-1$	$4\log_4 N$	N-1

B. Power Consumption:

The power consumption is measured by the number of times of data transition. The data transition times are proportional to the SRAM access times. Here we assume that the adders and multipliers are active at each clock cycle because of the pipelining architecture. The more the SRAM access times, the higher the power consumption Fig. 9 shows the SRAM access times versus N points FFT. The SRAM access times is linear to the number of the recursive iterations in FFT as described in Eq(6).The SRAM is accessed twice each clock cycle, so Eq(6) is multiplied by 2. It shows that the proposed design has less memory access than the radix-4 FFT by 20 ~ 40%. Therefore, the proposed architecture consumes much lower power.

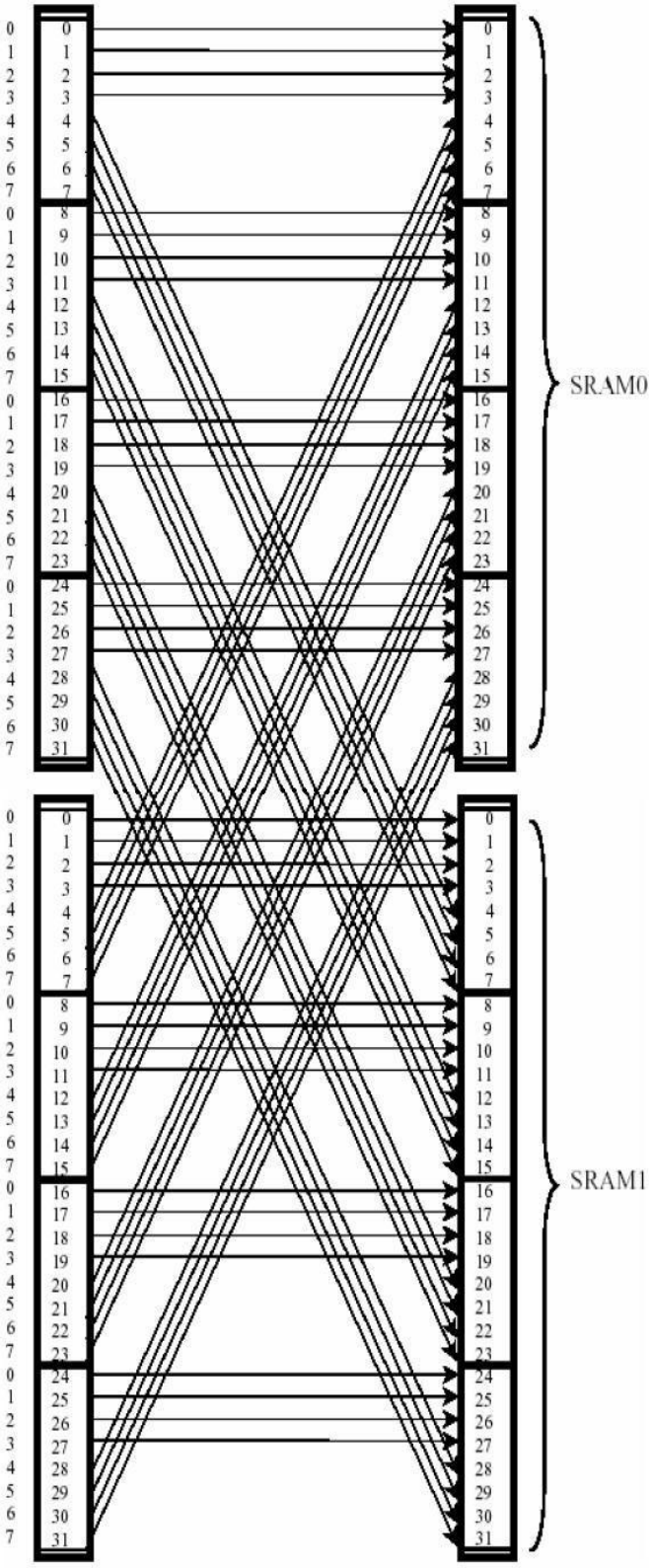

Fig 8. Memory addressing of SRAM0 and SRAM1 for N=64

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

SRAM access times = N * (iteration times) *2 (6)

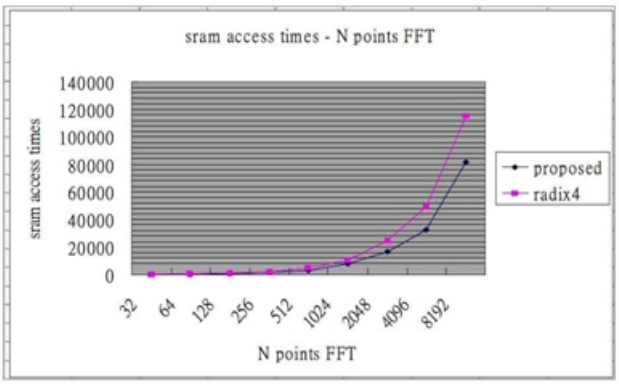

Fig 9. SRAM Access Times Vs N points FFT

C. Speed:

With fixed clock frequency, the processing OFDM symbol rate decreases as the FFT point N increases. A comparison with fixed clock frequency of 50 MHz is shown in Fig. 10 based on Eq. (7). It shows that the proposed architecture is better than radix-4 FFT by 25~ 66% .

$$\text{symbol rates} = \frac{(\text{clock frequency}) * N}{\text{Total cycles of each N points}}$$

(7) For fixed analog-to-digital converter sampling rate, the OFDM symbol rate in receiver is fixed too. It then requires higher chip clock frequency to process higher point FFT. We make a comparison with clock frequency and N-points FFT as shown in Eq. (8).

$$\text{Minimum clock frequency} = \frac{(\text{OFDM symbol rates}) * N}{\text{Total cycles of each N points}} \quad (8)$$

The proposed architecture is better than radix-4 [20] FFT by 20~40% .

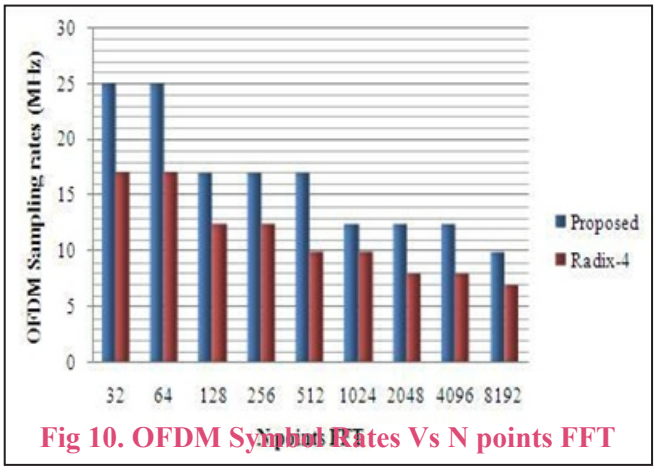

Fig 10. OFDM Symbol Rates Vs N points FFT

D. Implementation:

The Fujitsu Mobile WiMAX™ SoC, MB86K22, fully complies with the IEEE 802.16e standard using an OFDMA PHY [21]. This fully integrated baseband IC was built using generation. Power-gating technology shuts down the power supply in the unused blocks inside the device, so that the entire mobile WiMAX module consumes only 0.5mA, extending battery life. The chip supports multiple operating frequency and channel bandwidth profiles, as well as various subcarrier allocation schemes.The module would be able to support multiple channel bandwidths such as 3.5MHz, 5MHz, 7MHz, 10MHz and 20MHz; and all the popular frequency bands such as 2.3GHz, 2.5GHz and 3.5GHz. The Dual-core processor architecture best supports application-rich Mobile WiMAX operations.

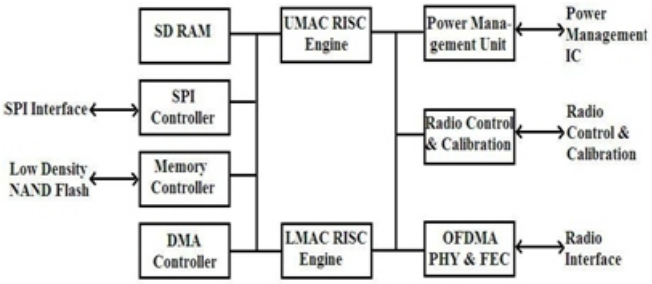

Fig 11. A simplified block diagram of the Fujitsu Mobile WiMAXSoC

Fig. 11 shows a simplified block diagram of the Fujitsu Mobile WiMAXSoC, MB86K22. It shows the dual processors and main hardware blocks that implement a complete PHY-to-MAC wireless MAN solution [21]. This SoC uses OFDMA PHY with FFT sizes of 512/1024 points.We have implemented the FFT, designed for efficient mobile WiMAX, by 16 bits word length and synthesized in Xilinx ISE 9.1 design compiler. Table 3 shows the processing time of IFFT/FFT in orientation with OFDMA and the projected values for various word lengths at 80MHz clock frequency.

The OFDMA symbol period for mobile WiMAX 802.16e is 102.9µsec (including a guard time of 11.4µsec). The results clearly indicate that the processing time of the proposed FFT is below the OFDM symbol period of 102.9µsec, which ensures an appropriate operation for the communication standard 802.16e. The table also presents the normalized area occupied for 0.6µm technology.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

TABLE 2:MOBILE WIMAX APPLICATIONS FOR THE FFT:

Word Length (N)	Processing Time (µsec)	Normalized Area (mm^2)
16	1.55	9.76
64	2.38	14.51
256	6.21	20.75
512	31.54	51.37
1024	65.89	124.23

VIII. CONCLUSIONS:

the Fujitsu 65nm advanced CMOS low-leakage process technology. The operating power of the MB86K22 has been reduced by 36 percent from the previous We have proposed a memory based recursive FFT design which has much less gate counts, lower power consumption and higher speed. The proposed architecture has three main advantages (1) fewer butterfly iteration to reduce power consumption, (2) pipeline of radix-22 butterfly to speed up clock frequency, (3) even distribution of memory access to make utilization efficiency in SRAM ports. In summary, the speed performance of our design easily satisfies most application requirements of mobile WiMAX 802.16e, which uses OFDMA modulated wireless communication system. Our design also occupies lesser area, hence lower cost and power consumption.

REFERENCES:

[1]Lenart and Viktor Öwall. Architectures for dynamic data scaling in 2/4/8k pipeline FFT cores. IEEE transactions on Very Large Scale Integration (VLSI) systems, vol. 14, no. 11, Pages: 1286-1290, November 2006.
[2]MianSijjadMinallah, Gulistan Raja. Real Time FFT Processor Implementation. IEEE—ICET 2006 2nd International Conference on Emerging Technologies. Peshawar, Pakistan 13-14, Pages: 192-195, November 2006.
[3]S Sukhsawas, K Benkrid. A High-level Implementation of a High Performance Pipeline FFT on Virtex-E FPGAs. Proceedings of the IEEE Comp. Society Annual Symp. on VLSI Emerging Trends in Systems Design (ISVLSI'04). 0-7695-2097-9/2004.
[4]K. Harikrishna, T. Rama Rao and Vladimir A. Labay, "A RADIX-22 Pipeline FFT for Ultra Wide Band Technology", International Conference on Computer & Network Technology (ICCNT), conference proceedings published by World Scientific Press, Singapore. Chennai, India, Jul 24-28, 2009.
[5]S. He and M. Torkelson. A new approach to pipeline FFT processor, 10th Int. Parallel Processing Symp. (IPPS'96), p. 766–770, 1996.
[6]Digital Signal Processing: Principles, Algorithms and Applications (3rd Edition) by John G. Proakis and Dimitris K Manolakis - Prentice-Hall, Inc., - Oct 5, 1995.
[7]Digital Design Principles and Practices, Fourth Edition, John F. Wakerly, Pearson Education, Inc. 2006.
[8]Modelsim manual. Mentor Graphics Corporation. http://support.xilinx.com.
[9]Xilinx, Inc. http://www.xilinx.com/.
[10]Xilinx, Inc. High-Performance 1024-Point Complex FFT/IFFT V2.0. http://www.xilinx.com/ipcenter/.
[11]http://www.wimaxforum.org/.
[12]http://www.ieee802.org/16/tgd/.
[13]"OFDM Towards Fixed and Mobile Broadband Wireless Access", Uma ShankerJha, and Ramjee Prasad, Artech House Inc., London, 2007.
[14]"OFDM For Wireless Communications Systems", Ramjee Prasad, Artech House Inc., London, 2004.
[15]http://www.wikipedia.org/.
[16]Chi-Hong Su and Jen-Ming Wu, Member, IEEE, Reconfigurable FFT Design for Low Power OFDM Communication Systems, 2006, IEEE.
[17]L.R. Rabiner and B. Gold. Theory and Application of Digital SignalProcessing. Prentice-Hall, Inc., 1975.
[18]E.H. Wold and A.M. Despain. Pipeline and parallel-pipeline FFT processors for VLSI implementation. IEEE Trans. Comput., C-33(5):414–426, May 1984.
[19]A.M. Despain. Fourier transform computer using CORDIC iterations. IEEE Trans. Comput., C-23(10):993–1001, Oct.1974.
[20]G. Bi and E. V. Jones. A pipelined FFT processor for word sequential data. IEEE Trans. Acoust., Speech, Signal Processing, 37(12):1982– 1985, Dec. 1989.
[21]http://www.fujitsu.com/ .