INTRODUCTION

1.1 General

The Fast Fourier Transform (FFT) is one of the most used algorithms in digital signal processing. The FFT, which facilitates the efficient transformation between the time domain and the frequency domain for a sampled signal, is used in many applications, e.g., radar, communication, sonar, speech signal processing.

In the last decade, the interest for high speed wireless and on cable communication has increased. Orthogonal Frequency Division Multiplexing (OFDM) technique, which is a special Multicarrier Modulation (MCM) method, has been demonstrated to be an efficient and reliable approach for high-speed data transmission. The immunity to multipath fading channel and the capability for parallel signal processing make it a promising candidate for the next generation wide-band communication systems. The modulation and demodulation of OFDM based communication systems can be efficiently implemented with an FFT, which has made the FFT valuable for those communication systems. The OFDM based communication systems have high performance requirement in both throughput and power consumption. This performance requirement necessitates an application-specific integrated circuit (ASIC) solution for FFT implementation. This thesis addresses the problem of designing efficient application-specific FFT processors for OFDM based wide-band communication systems.

The requirement for a high capacity, fast, reliable and secure media in wireless domain particularly for mobile communication systems have to be identified to support multimedia transmission and are considered to very essential in recent years. Further, a suitable modulation scheme in support of the above specified
requirement is another important factor to be investigated in depth. Out of various modulation schemes the Orthogonal Frequency Division Multiplexing (OFDM) is identified as an effective modulation scheme in order to meet out the requirements as specified. In OFDM the data are normally grouped into many streams of data in parallel and each stream is assigned with a carrier.

OFDM has got many applications particularly for wireless communication such as television, audio, wireless networks and broadband internet access. The characteristic of OFDM such as flexibility to channel conditions makes it as widely popular. The core key in the OFDM is Fast Fourier Transform (FFT)/Inverse Fast Fourier Transform (IFFT). The modulation and demodulation of OFDM based mobile communication systems can be efficiently implemented with an FFT and IFFT, which has made these algorithms valuable for those communication systems. The functionality of an OFDM is represented in the form of a block diagram by illustrating receiver and transmitter of an OFDM system in Figure 1.1. The figure shown above indicates the digital implementation of an OFDM sub carrier modulator / demodulators with respect to Discrete Fourier Transform (DFT). In this the number of sub carrier can be changed according to the requirement.

![Figure 1.1 Basic block diagram of MIMO OFDM system](image-url)
One of the most used and popular algorithms in digital signal processing system is the Fast Fourier Transform (FFT). The FFT are applied in radar, communication, sonar and speech signal processing. These applications require efficient transformation for a sampled signal between the time domain and the frequency domain. From the statistics it can be seen that the highest demand for efficient wireless and mobile communication. The performance of OFDM with regard to Area occupancy in FPGA and power consumption give rise to an Application-Specific Integrated Circuit (ASIC) solution for FFT implementation.

1.2 DFT and FFT
In order to get an efficient implementation from FFT processor, the mathematical properties of FFT must be exploited to its maximum level and hence the selection of FFT algorithm [31] results with the better speed, hardware complexity, power consumption etc.

The N-point discrete Fast Fourier Transform (DFT) is defined as:

\[ X[k] = \sum_{n=0}^{N-1} x[n] W_N^{nk}, \quad k = 0,1,\ldots,N-1 \]  

(1.1)

where \( W_N^{nk} = e^{-j\frac{2\pi nk}{N}} \) \( 0 \leq k \leq N-1 \) is the DFT coefficient.

\( X(k) \) is the \( k^{th} \) harmonic and \( x(n) \) is the \( n^{th} \) input sample. Direct DFT calculation requires a computational complexity of \( O(N^2) \). By using The Cooley–Tukey FFT algorithm, the complexity can be reduced to \( O(N \log_r N) \). The Cooley-Tukey FFT is the most universal of all FFT algorithms, because of any factorization of \( N \) is possible. The most popular Cooley-Tukey FFTs are those were the transform length is a power of a basis \( r \), i.e., \( N = r^S \). These algorithms are referred to as radix-\( r \) algorithms. The most commonly used are those of basis \( r = 2 \) and \( r = 4 \). For \( r = 2 \) and \( S \) stages, for instance, the following index mapping of the Cooley–Tukey algorithm gives:
\[ n = 2^{s-1} n_1 + 2^{s-2} n_2 + \ldots + 2 n_{s-1} + n_s \]  
(1.2)

\[ k = 2^{s-1} k_s + 2^{s-2} k_{s-1} + \ldots + 2 k_2 + k_1 \]  
(1.3)

\[ n_1, n_2, \ldots, n_{s-1}, n_s = 0, 1 \text{ and } k_s, k_{s-1}, \ldots, k_2, k_1 = 0, 1 \]

The Cooley–Tukey algorithm is based on a divide-and-conquers approach in the frequency domain and therefore is referred to as decimation-in-frequency (DIF) FFT. The DFT formula is split into two summations:

\[
X[k] = \sum_{n=0}^{N-1} x[n] W_N^{nk} + \sum_{n=N/2}^{N-1} x[n] W_N^{(n+N/2)k} \quad \text{and} \quad W_N^{(N/2)^k} = (-1)^k
\]
(1.4)

\[
X[2k] = \sum_{n=0}^{N/2-1} \left( x(n) + \frac{N}{2} \right) W_N^{2nk} = \sum_{n=0}^{N/2-1} \left( x(n) + \frac{N}{2} \right) W_N^{Nk/2}
\]
(1.5)

\[
X[2k+1] = \sum_{n=0}^{N/2-1} \left( x(n) - \frac{N}{2} \right) W_N^{2nk} = \sum_{n=0}^{N/2-1} \left( x(n) - \frac{N}{2} \right) W_N^{Nk/2}
\]
(1.6)

The computational procedure can be repeated through decimation of the N/2-point DFTs \(X(2k)\) and \(X(2K+1)\). The entire algorithm involves \(\log_N 2\) stages, where each stage involves \(N/2\) operation units (butterflies). The computation of the \(N\) point DFT via the decimation-in-frequency FFT, as in the decimation-in-time algorithm requires \((N/2) \cdot \log_N 2\) complex multiplication and \(N \cdot \log_N 2\) complex addition.
1.3 Basic Concepts of FFT Algorithms

For reduced number of computations and multiplications are done through two schemes in the FFT algorithms:

1.3.1 Divide-and-Conquer

By decomposing the computation of the Discrete Fourier Transfer (DFT) into a sequence of length N into successively smaller DFT this is achieved. There are different varieties of algorithms to obtain such principle, all with comparable improvements in computational speed. The three major steps involved in the divide-and-conquer paradigm are

Step 1. Divide the sequence of data into two or more number of smaller subsequent sizes.
Step 2. Recursively solve each subsequent size with the same algorithm. Apply the boundary condition to terminate the recursion when the sizes are small enough.
Step 3. Obtain the solution for the original problem by combining the solutions to the subsequences.

1.3.2 Properties of the Coefficient $W_N^{nk}$

In Equation (1) the main control parameter is $W_N^{nk}$ that is the DFT coefficient also called twiddle factor. The partial twiddle factor of an $N$- point DFT [32] ($N$ is the power of two) shows in Figure 1.2.
Most approaches to improve the efficiency of the computation of the DFT exploit the properties of twiddle factor $W_{nk}$ as described below:

1. $W_{N}^{0} = -W_{N}^{N/2} = 1$

   
   $W_{N}^{N/4} = -W_{N}^{3N/4} = -j$

2. $W_{N}^{n(k+N/2)} = -W_{N}^{nk}$

3. $W_{N}^{nk} = W_{N}^{(n+N)k} = W_{N}^{n(k+N)}$ Periodicity in $n$ and $k$

For certain values of the product $kn$, the sine and cosine functions take on the value 1 or 0 (property 1), eliminating the multiplications, as shown below,

$$A \times W_{N}^{nk} + B \times W_{N}^{nk+N/2} = (A + B \times W_{N}^{N/2}) \times W_{N}^{nk} = (A - B) \times W_{N}^{nk} \quad (1.7)$$

$$A \times W_{N}^{nk} + B \times W_{N}^{nk+N/4} = (A + B \times W_{N}^{N/4}) \times W_{N}^{nk} = (A - jB) \times W_{N}^{nk} \quad (1.8)$$

However, reductions of this type still leave with an amount of computation that is proportional to $N^2$. Fortunately, the second property of the periodicity of the
complex sequence $W_{nk}^n$ can be arrived in achieving significantly greater reduction of the computation.

$$A \times W_{nk}^n + B \times W_{nk}^{n(k+N)} = (A + B \times W_{nk}^{nN}) \times W_{nk}^n = (A + B) \times W_{nk}^n$$  \hspace{1cm} (1.9)

where $W_{nk}^n = 1$ \hspace{0.5cm} $n = 0,1,2,3,\ldots,N-1$. We can further solve the symmetric property of the twiddle factor to reduce some complex multiplications.

$$W_{nk+N/8}^n = -W_{nk+5N/8}^n = \frac{\sqrt{2}}{2} (1 - j) \times W_{nk}^n$$  \hspace{1cm} and

$$W_{nk+3N/8}^n = -W_{nk+7N/8}^n = \frac{\sqrt{2}}{2} (1 + j) \times W_{nk}^n$$  \hspace{1cm} (1.10)

The multiplication of a complex number by twiddle factor in above equations involves two real multiplication and two real additions. Thus, the symmetry utilization of phase difference $\pm 45^\circ$ only requires two real multiplications.

Many implementation approaches for the FFT have been proposed since the discovery of FFT algorithms. Due to the high computation workload and intensive memory access, the implementation of FFT algorithms is still a challenging task.

1.4 OFDM Basics

OFDM is a special MCM technique. The idea for MCM is to divide transmission bandwidth into many narrow sub channels (subcarriers), which transmit data in parallel [33].

The principle for MCM is shown in Fig. 1.3. The high rate data stream at $M f_{sym}$ bits/s is grouped into blocks with $M$ bits per block at a rate of $f_{sym}$. A block is called a symbol. A symbol allocates $m_k$ bits of M bits for modulation of a carrier k
at $f_c$, $k$ and totally $M$ bits for modulation of $N$ carriers. This results in $N$ sub channels, which send symbols at a rate of $fsym$.

In the conventional MCM, the $N$ sub channels are non-overlapping. Each sub channel has its own modulator and demodulator. This leads to inefficient usage of spectrum and excessive hardware requirement.

The OFDM technique can overcome those drawbacks. With OFDM, the spectrum can be used more efficient since overlapping of sub channels is allowed. The overlapping does not cause interference of sub channels due to the orthogonal modulation.

The orthogonality can be explained in frequency domain. The symbol rate is $fsym$, e.g., each symbol is sent during a symbol time $T$ (which is equal to $1/fsym$). The frequency spacing between adjacent sub channels is set to be $1/T$ Hz, the carrier signals can be expressed as following:
\begin{equation}
f_k = f_0 + \frac{k}{T} \quad 0 \leq k \leq N-1 \tag{1.11}
\end{equation}

\begin{equation}
g_k(t) = \begin{cases} e^{j2\pi jk t} & 0 \leq t \leq T \\ 0 & \text{otherwise} \end{cases} \tag{1.12}
\end{equation}

where \( f_0 \) is the system base frequency and is the signal for carrier \( k \) at frequency \( f_k \). If the frequency of subcarrier \( k \) and the base function are chosen according to Eq. (1.1) and Eq. (1.2), its spectrum is a sinc function with zero points \( f_0 + l/T \) at \( (l \text{ is integer}) \) except \( l = k \) or \( f_k \). It means that there is no interference to other subchannels with the selected functions.

This orthogonality can also be found in the time domain. For two carrier signals, \( g_k \), and \( g_l \), the integral over a symbol time is

\begin{equation}
\int_0^T g_k(t)g_l^*(t)dt = \begin{cases} T & k=1 \\ 0 & \text{otherwise} \end{cases} \tag{1.13}
\end{equation}

which shows that two carriers are orthogonal.

![Figure 1.4 Spectrum overlapping of subcarriers for OFDM.](image)

OFDM overcomes the inefficient implementation of the modulator and demodulator for conventional MCM. From Fig. 1.3, the sending signal \( x(t) \) is the summation of symbol transmission in all subchannels, e.g.,
\[ X[t] = \sum_{k=0}^{N-1} S_k g_k(t) = e^{j2\pi f_0 t} \sum_{k=0}^{N-1} S_k e^{j2\pi k t / T} \]  

(1.14)

where \( S_k \) is the modulated signal of \( mk \) bit, which should be transmitted by subchannel \( k \). This is an \( N \)-point Inverse Discrete Fourier Transform (IDFT) and baseband modulation (with \( e^{j2\pi f t_0} \)).

The IDFT can be computed efficiently by Inverse Fast Fourier Transform (IFFT) algorithm. Hence the OFDM modulator can be implemented with one IFFT processor and baseband modulator for \( N \) subcarriers instead of \( N \) modulators for conventional MCM. In similar way, the OFDM demodulator can be implemented more efficient than that of conventional MCM. The simplified OFDM system based on the FFT is shown in Fig. 1.5.

\[ \text{Figure 1.5 OFDM system based on FFT.} \]

In reality, the interference between sub channels exists due to the non-ideal channel characteristics and frequency offset in transmitters and receivers. This interference effects the performance of the OFDM system. The frequency offset can, in most case, be compensated.
The other issues, for instance, inter-symbol interference, can be reduced by techniques like cyclic prefix.

1.5 Literature Survey

Shousheng. He and Mats Torkelson,(1998) who have designed a new form of FFT, the radix-2 algorithm. By exploiting the spatial regularity of the new algorithm, minimal requirement for both dominant components in VLSI implementation has been achieved: only 4 complex multipliers and 1024 complex-word data memory for the pipelined 1K FFT processor. it can compute $2^n$, $n=0, 1, ..., 10$ complex point forward and inverse FFT in real time with up to 30 MHz sampling frequency. The SQNR is above 50 dB for white noise input [1].

Shousheng. He and Mats Torkelson,(1998) proposed the FFT processor is one of the key components in the implementation of wideband OFDM systems. Architectures with a structured pipeline have been used to meet the fast, real-time processing demand and low-power consumption requirement in a mobile environment. Architectures based on new forms of FFT, the radix-2i algorithm derived by cascade decomposition, is proposed. By exploiting the spatial regularity of the new algorithm, the requirement for both dominant elements in VLSI implementation, the memory size and the number of complex multipliers, have been minimized. Progressive wordlength adjustment has been introduced to optimize the total memory size with a given signal-to-quantization-noise-ratio (SQNR) requirement in fixed-point processing. A new complex multiplier based on distributed arithmetic further enhanced the area/power efficiency of the design. A single-chip processor for 1 K complex point FFT transform is used to demonstrate the design issues under consideration[2].

Shousheng. He and Mats Torkelson,(1996) investigated a new VLSI architecture for a real-time pipeline FFT processor is proposed. A hardware-oriented radix-22
algorithm is derived by integrating a twiddle factor decomposition technique in the divide-and-conquer approach. The radix-22 algorithm has the same multiplicative complexity as the radix-4 algorithm, but retains the butterfly structure of the radix-2 algorithm. The single-path delay-feedback architecture is used to exploit the spatial regularity in the signal flow graph of the algorithm. For length-N DFT computation, the hardware requirement of the proposed architecture is minimal on both dominant components: log4N-1 complexity multipliers and N-1 complexity data memory. The validity and efficiency of the architecture have been verified by simulation in the hardware description language VHDL [3].

C. Burrus (1977) proposed the mapping of one-dimensional arrays into two- or higher dimensional arrays is the basis of the fast Fourier transforms (FFT) algorithms and certain fast convolution schemes. This paper gives the general conditions for these mappings to be unique and cyclic, and then considers the application to discrete Fourier transform (DFT) and convolution evaluation[4].

Lihong Jia et al (1998) presents a new VLSI-oriented fast Fourier transform (FFT) algorithm-radix-2/4/8, which can effectively minimize the number of complex multiplications. This algorithm can be implemented efficiently using a pipelined architecture. Based on this pipelined architecture, an 8 K FFT ASIC is designed for use in the DVB (Digital Video Broadcasting) application in 0.6 μm-3.3 V triple-metal CMOS process [5].

Daisuke Takahashi (2001) proposed an extended split-radix fast Fourier transform (FFT) algorithm is proposed. The extended split-radix FFT algorithm has the same asymptotic arithmetic complexity as the conventional split-radix FFT algorithm. Moreover, this algorithm has the advantage of fewer loads and stores than either the conventional split-radix FFT algorithm or the radix-4 FFT algorithm [6].

Lin, Y.-T (2005) present the Fast Fourier transform (FFT) processing is one of the key procedures in the popular orthogonal frequency division multiplexing (OFDM)
communication systems. Structured pipeline architectures and low power consumption are the main concerns for its VLSI implementation. In the paper, the authors report a variable-length FFT processor design that is based on a radix-2/4/8 algorithm and a single-path delay feedback architecture. The processor can be used in various OFDM-based communication systems, such as digital audio broadcasting (DAB), digital video broadcasting-terrestrial (DVB-T), asymmetric digital subscriber loop (ADSL) and very-high-speed digital subscriber loop (VDSL). To reduce power consumption and chip area, special current-mode SRAMs are adopted to replace shift registers in the delay lines. In addition, techniques including complex multipliers containing three real multiplications, and reduced sine/cosine tables are adopted. The chip is fabricated using a 0.35 μm CMOS process and it measures 3900 μm × 5500 μm. According to the measured results, the 2048-point FFT operation can function correctly up to 45 MHz with a 3.3 V supply voltage and power consumption of 640 mW. In low-power operation, when the supply voltage is scaled down to 2.3 V, the processor consumes 176 mW when it runs at 17.8 MHz [7].

Alamouti, S.M. (1998) presents a simple two-branch transmit diversity scheme. Using two transmit antennas and one receive antenna the scheme provides the same diversity order as maximal-ratio receiver combining (MRRC) with one transmit antenna, and two receive antennas. It is also shown that the scheme may easily be generalized to two transmit antennas and M receive antennas to provide a diversity order of 2M. The new scheme does not require any bandwidth expansion or any feedback from the receiver to the transmitter and its computation complexity is similar to MRRC [8].

Becker, J. (2002) proposed a Systems-on-chip (SoCs) has become reality now, driven by fast development of CMOS VLSI technologies. Complex system integration onto one single die introduces a set of various challenges and perspectives for industrial and academic institutions. Important issues to be addressed here are cost-effective technologies, efficient and application-tailored
hardware/software architectures, and corresponding IP-based EDA methods. Due to exponentially increasing CMOS mask costs, essential aspects for the industry are now adaptivity of SoCs, which can be realized by integrating reconfigurable reusable hardware parts on different granularities into configurable systems-on-chip (CSoCs) [9].

Blum R. S. (2001) investigate the improved space-time coding for multiple-input multiple-output orthogonal frequency division multiplexing is studied for wireless systems using QPSK modulation for four transmit and four receive antennas. A 256-state code is shown to perform within 3 dB of outage capacity (and within 2 dB with perfect channel estimation), which is better than any other published result without using iterative decoding [10].

P. Coulton and D. Carline (2004) analyzed the software Defined Radio (SDR) technologies and concepts have become one of the most important topics of research particularly in communications as it aims to deliver greater platform flexibility when compared with conventional radio technology. Field Programmable Gate Arrays (FPGAs) have been suggested as an enabling technology for the hardware platform of an SDR system although the programmability offered was relatively course grained in that designs could only be altered through a complex set of software tools with whole device having to be reprogrammed to change its operation. The SelectMAP interface if the Xilinx VirtexPGA introduced a means where the configuration data applied represents only a fraction of a full configuration file. However, the flexibility is restricted and in order to maximize flexibility the hardware platform functional objects must relative themselves to both the architecture of the device and the requirements of other objects implemented within the device, In this paper they define SDR functional objects suitable for implementation on an FPGA and provide an example of applying these techniques to the part of the 802.11a base band definition that offers variable coding and modulation[11].
C. Dick and F. Harris, (2003) investigates the Orthogonal frequency division multiplexing (OFDM) based communication is increasingly being used in environments that exhibit severe multipath. While there are ASSP solutions for many common (e.g. 802.11a) and emerging standards, many communication systems, for example a military software radio, demand flexibility. The arithmetic requirements of an OFDM system can be very demanding. Even the ubiquitous 802.11a WLAN system has arithmetic requirements in the billions-of-operations per second region and cannot be satisfied even by high-end DSP microprocessors. This paper reports on the FPGA implementation of an OFDM transceiver. In addition to the FFT based modulator and demodulator, receiver synchronization and channel estimation is discussed. The FPGA resource requirements of the various sub-systems are reported and the design methodology employed for system design, verification and FPGA implementation is described [12].

Jongren, G. et al. (2002) Multiple transmit and receive antennas can be used in wireless systems to achieve high data rate communication. Efficient space-time codes have been developed that utilize a large portion of the available capacity. These codes are designed under the assumption that the transmitter has no knowledge about the channel. In this work, on the other hand, we consider the case when the transmitter has partial, but not perfect, knowledge about the channel and how to improve a predetermined code so that this fact is taken into account. A performance criterion is derived for a frequency-nonselective fading channel and then utilized to optimize a linear transformation of the predetermined code. The resulting optimization problem turns out to be convex and can thus be efficiently solved using standard methods. In addition, a particularly efficient solution method is developed for the special case of independently fading channel coefficients. The proposed transmission scheme combines the benefits of conventional beamforming with those given by orthogonal space-time block coding. Simulation results for a narrow-band system with multiple transmit antennas and one or more receive antennas demonstrate significant gains over conventional methods in a scenario with nonperfect channel knowledge [13].
Bolcskei H. et al (2002) present this paper deals with the capacity behavior of wireless orthogonal frequency-division multiplexing (OFDM)-based spatial multiplexing systems in broad-band fading environments for the case where the channel is unknown at the transmitter and perfectly known at the receiver. Introducing a physically motivated multiple-input multiple-output (MIMO) broad-band fading channel model, the authors studied the influence of physical parameters such as the amount of delay spread, cluster angle spread, and total angle spread, and system parameters such as the number of antennas and antenna spacing on ergodic capacity and outage capacity. They found that, in the MIMO case, unlike the single-input single-output (SISO) case, delay spread channels may provide advantages over flat fading channels not only in terms of outage capacity but also in terms of ergodic capacity. Therefore, MIMO delay spread channels will in general provide both higher diversity gain and higher multiplexing gain than MIMO flat fading channels [14].

Han, W. et al (2005) proposes two novel parallel-pipelined FFT architectures, based on multiplier-less implementation, targeting wireless communication applications, such as IEEE 802.11 wireless baseband chip and MC-CDMA receiver. The proposed parallel-pipelined architectures have the advantages of high throughput and high power efficiency. The multiplier-less architecture uses shift and addition operations to realize complex multiplications. By combining a new commutator architecture, and a low power butterfly with this approach, the resulting power and area savings are up to 31% and 20% respectively, for 64-point and 16-point FFTs, as compared to parallel-pipelined FFTs based on Booth coded Wallace tree multipliers [16].

Jungnickel V. et al present a concept for implementing a real-time MIMO-OFDM system on a hybrid FPGA/DSP software-radio platform. The basic system parameters are chosen similar to the HiperLAN/2 standard using a fixed frame structure. A new C preamble is defined, in order to identify the individual transmit antennas at the receiver. We propose a code-multiplex approach in order minimize the channel estimation error. With the new preamble, the correlation circuits at the
receiver can be reused for all carriers and no multipliers must be used. Three methods to increase the channel tracking rate at the receiver are discussed and we argue that a combination of them may be suitable for a wide range of mobility scenarios [17].

Manavi and Y. R. Shayan (2004) have designed a prototype, which is based on the orthogonal frequency division multiplexing (OFDM) technique, is presented for the physical layer of the IEEE 802.11a standard. Implementation aspects of an OFDM modem on a Xilinx field programmable gate array (FPGA) are addressed. The system includes synchronization circuitry used for packet detection and time synchronization. The demonstrated design flow shows an approach to implementing and prototyping the architecture of a real-time base-band OFDM modem [19].

Masselos K. et al (2003) describe the wireless multimedia communication systems become increasingly more computational intensive and demand for higher flexibility. The realization of these systems on reconfigurable hardware offers a good balance for these requirements. In this paper the suitability of commercially available reconfigurable hardware platforms for the target application domain is evaluated. Based on this evaluation a heterogeneous partly reconfigurable system-on-chip platform is identified as ideal implementation platform for the targeted systems. Systems from different target domains are analyzed and different cases where the inclusion of reconfigurable hardware in their realizations would lead to improved quality in terms of implementation efficiency and flexibility are identified. Design methodology requirements for the realization of systems from the target application domain on the targeted platform are analysed and issues not covered by existing methodologies are identified. The principles of a methodology handling these open issues are described. Results from the prototyping of different systems are also presented and show the potentials of a reconfigurable hardware platform, which in the future will lead to reduced costs and increased flexibility of the wireless multimedia communication systems [20].
Meeuwsen M. J. et al (2004) presents a software based IEEE 802.11a digital baseband transmitter has been implemented on a highly parallel single-chip DSP. The processing platform is a programmable and reconfigurable asynchronous array of simple processors (AsAP) that is well matched to complex system workloads such as 802.11a. The transmitter is the first fully-compliant 802.11a software implementation, and is the first full-rate software implementation. The transmitter also complies with the high-rate portions of the 802.11g standard. It operates over all 8 data rates, includes additional upsampling and filtering functions, and sustains transmissions at 54 Mb/s on a 22-processor array; it is expected to occupy less than 20 mm2 in 0.18 μm CMOS [21].

Meng T. H. et al.(2003) investigate the tremendous growth in wireless LANs has generated interest in technologies that provide higher data rates and greater system capacities. The IEEE 802.11a standard [1], based on coded OFDM modulation [2], provides nearly five times the data rate and at least 20 times the overall system capacity compared to the incumbent 802.11b wireless LAN systems [3]. This article describes the design challenges and circuit implementation of a two-chip set that forms a complete 802.11a solution in 0.25 μm CMOS technology. Wherever possible, sophisticated digital signal processing techniques are used to compensate for possible analog impairments associated with integrating RF circuitry in a CMOS technology. The analog portion of the chip set implements a 5 GHz transceiver comprising all the necessary RF and analog circuits of the 802.11a standard integrated on a single chip. Some features of this IC include 22 dBm peak transmitted power, 8 dB overall receive chain noise figure, and –112 dBC/Hz synthesizer phase noise at 1 MHz frequency offset. The digital portion of the chip set, the baseband and MAC processor, contains dual ADCs/DACs and all the digital circuits for synchronization, detection, and 802.11 MAC layer data processing. This IC delivers up to 54 Mb/s in a 20 MHz channel according to the 802.11a standard, and includes proprietary modes supporting up to 108 Mb/s in a 40 MHz channel [22].
Jingzhao Ou and Prasanna, V.K. (2005) presents FPGA configured soft processors are an attractive choice for implementing many embedded systems. For application development using these soft processors, the users can execute portions of the applications as software programs and the other portions as customized hardware implementations. Being able to rapidly simulate various partitions of the applications on hardware and software is crucial to efficiently execute them on soft processors because (a) there are many possible configurations of soft processors, and (b) low-level simulation techniques are too time consuming for evaluating these different partitioning and configuration possibilities. While state-of-the-art design tools rely on low-level simulation and are unable to deliver such a fast simulation speed, we propose a high-level cycle-accurate hardware/software cosimulation environment based on MATLAB/Simulink for application development using soft processors. By utilizing the high-level cycle-accurate abstractions of the low-level hardware implementations and the arithmetic simulation capability provided by MATLAB/Simulink, our tool considerably accelerates the time for cycle-accurate functional simulation of both hardware and software portions of a given application running on soft processors. To illustrate our approach, we develop a CORDIC division application and a matrix multiplication application on a commercial soft processor. Up to 19.4x improvement in simulation time is achieved using our cosimulation environment compared with that of low-level simulation for various partitions of these applications and for various configurations of the soft processor [23].

Storn R. (1994) presents a radix-2 FFT-pipeline architecture has been developed which exhibits a two- to three-bit increase in accuracy for transform lengths N greater than $2^{10}$ if fixed-point arithmetic is utilised. The algorithm in use is a unification of the Cooley-Tukey radix-4 and radix-4+2 decompositions [24].

Rao, R.M et al (2004) investigate the wireless communication systems present unique challenges and trade-offs at various levels of the system design process. Since a variety of performance measures are important in wireless communications,
a family of testbeds becomes essential to validate the gains reported by the theory. Wireless testbeds also play a very important role in academia for training students and enabling research. In this article we discuss a classification scheme for wireless testbeds and present an example of the testbeds developed at UCLA for each of these cases. We present the unique capabilities of these testbeds, provide the results of the experiments, and discuss the role they play in an educational environment [25].

Stuber G. L et al. (2004) presents the orthogonal frequency division multiplexing (OFDM) is a popular method for high data rate wireless transmission. OFDM may be combined with antenna arrays at the transmitter and receiver to increase the diversity gain and/or to enhance the system capacity on time-varying and frequency-selective channels, resulting in a multiple-input multiple-output (MIMO) configuration. The paper explores various physical layer research challenges in MIMO-OFDM system design, including physical channel measurements and modeling, analog beam forming techniques using adaptive antenna arrays, space-time techniques for MIMO-OFDM, error control coding techniques, OFDM preamble and packet design, and signal processing algorithms used to perform time and frequency synchronization, channel estimation, and channel tracking in MIMO-OFDM systems. Finally, the paper considers a software radio implementation of MIMO-OFDM [26].

Tang, Y et al. (2005) investigate the explosive growth of 802.11-based wireless LANs has attracted interest in providing higher data rates and greater system capacities. Among the IEEE 802.11 standards, the 802.11a standard based on OFDM modulation scheme has been defined to address high-speed and large-system-capacity challenges. Hardware implementations are often used to meet the high-data-rate requirements of 802.11a standard. Although software based solutions are more attractive due to the lower cost, shorter development time, and higher flexibility, it is still a challenge to meet the high-data-rate requirements of 802.11a by software. In this paper, we implement a software-based 802.11a digital baseband
transmitter on the TI TMS320C64x DSP. The transmitter can operate over all data rates defined in the 802.11a standard and is compatible with the high-rate portions of the 802.11g standard. Two major optimizations have been used in the software implementation to achieve the high-data-rate: 1) parallelizing the scrambler function and 2) concatenating the FEC encoder, puncturing, and interleaver functions. Experimental results show that the optimized software implementation on a single C64x DSP with a clock frequency of 1.0 GHz can operate at the maximum of 136 Mbits/s, which is twice as fast as the previous software implementation at the same clock frequency [27].

A. Zelst. et. al. (2004) have discussed the combination of multiple-input multiple-output (MIMO) signal processing with orthogonal frequency division multiplexing (OFDM) is regarded as a promising solution for enhancing the data rates of next-generation wireless communication systems operating in frequency-selective fading environments. To realize this extension of OFDM with MIMO, a number of changes are required in the baseband signal processing. An overview is given of the necessary changes, including time and frequency synchronization, channel estimation, synchronization tracking, and MIMO detection. As a test case, the OFDM-based wireless local area network (WLAN) standard IEEE 802.11a is considered, but the results are applicable more generally. The complete MIMO OFDM processing is implemented in a system with three transmit and three receive antennas, and its performance is evaluated with both simulations and experimental test results. Results from measurements with this MIMO OFDM system in a typical office environment show, on average, a doubling of the system throughput, compared with a single antenna OFDM system. An average expected tripling of the throughput was most likely not achieved due to coupling between the transmitter and receiver branches [28].

Heejung Yu, et. al., (2005) presents Multiple input multiple output-orthogonal frequency division multiplexing (MIMO-OFDM) is regarded as a promising technology enabling higher data rate wireless communications in frequency
selective fading channels. This paper proposes the next generation wireless local area network (WLAN) physical (PHY) layer transmission technology using dual-band and MIMO-OFDM schemes. It describes a design of the higher data rate WLAN system targeting the maximum speed of 144 Mbps in 20 MHz bandwidth channels and 288 Mbps in 40 MHz. Additionally, the design suggests a new frame structure based on the 802.11a frame format for guaranteeing the compatibility with IEEE 802.11a legacy-OFDM systems and the MIMO channel estimation capability. For a cost-effective implementation and improved error performance, 2 transmit and 3 receive antennas are used in dual-band [29].

Chen, Y et al.,(2006) presents a low-power design of a two-stream MIMO FFT/IFFT processor for WiMAX applications. A novel block scaling method and a new ping-pong cache-memory architecture are proposed to reduce the power consumption and hardware cost. With these schemes, half the memory accesses and 64-Kbit memory can be saved. Furthermore, by proper scheduling of the two data streams, the proposed design achieves better hardware utilization and can process two 2048-point FFTs/IFFTs consecutively within 2052 cycles. A test chip of the proposed FFT/IFFT processor has been designed using UMC 0.13 mum 1P8M process with a core area of 1332times1590 mum². The SQNR performance of the 2048-point FFT/IFFT is over 48 dB for QPSK and 16/64-QAM modulations. Power dissipation of two 2048-point FFT computations is about 17.26 mW at 22.86 MHz which meets the maximum throughput rate of WiMAX applications [30].

1.6 VLSI Implementation

This section discusses the methodology of the research work and tools that involved in the process to complete the design and implementation of OFDM receiver in the FPGA hardware [34]. The work basically covers on the usage of the algorithm and some explanation on the Xilinx development board.
The methodology of the research is basically divided into four main phases. These phases are started with detailed study of the relevant topics followed by the design process, implementation, test and analysis. The phases are subdivided into several smaller stages and are explained this section.

The software used throughout the stages is shown in Figure 1.6. The functioning of software is discussed in this chapter. Xilinx Vertex is used for hardware part and some documentation regarding this hardware is also shown.

![Flow chart of the research methodology.](image)

1.6.1 Study Relevant Topics

Figure 1.6 illustrates the flow of the research work. As mention before, methodology of the research is divided into three main phases. The first phase will cover on study the relevant topics. On this stage, the works are subdivided into three main topics which is FFT and IFFT, VHDL programming [35, 36] and Xilinx
development board. These are the topics that need to cover before proceeding into the design phase. A study on FFT and IFFT is required to understand the computation process. This requirement is important especially during hardware development and software programming part. Bit representation in binary is also another issue which is required to be studied in this stage. Bit representation is crucial when the multiplication or addition process involved point values such as twiddle factor. In VHDL, there are two issues need to cover up which is Register Transfer Logic (RTL) and Behavioral Modeling and Synthesis. The last part in this phase is to study the Xilinx development board.

### 1.6.2 Design Phase

After a through investigation and deep analysis from the theoretical perspective, the works is further taken into the design phase. In this phase, the process is subdivided into several workable modules which are VHDL design, VHDL analyzer, Logic Synthesis, device fitting, and design verification [37]. These topics are actually the process involved in the completion of the hardware design. Each of the process requires different software to accomplish the design.

VHDL design is the first steps to perform in the design phase. Modelsim software is used as the design entry and programmed in VHDL language. Basically the process here is to generate the VHDL source code. After generating the code, FPGA Advantage software is used to verify the generated code. The software will perform two processes:

- VHDL analyzer and
- Logic synthesizer.

VHDL analyzer output is used for the logic synthesis and design verification. In logic synthesis, the net list file which is obtained from VHDL analyzer is synthesized based on the design constraints and technology library available in the
software. The software will produce *.edit as output file. This file is then used in technology mapping which is performed by Xilinx ISE software. In technology mapping, a process called device fitting is executed to partition, fit, place and route the design based on the targeted device.

Device fitting process will produces three main output file which is *.bit, *.mcs and *.snf file. There are two types of simulation at the design verification:

- Functional simulation and
- Timing simulation.

The functional simulation is to simulate the hardware function and this process is not carried out since the software used is not available. But the timing simulation is performed using Max + Plus II software. The timing simulation is to provide the timing function for the designed hardware.

### 1.6.3 Implementation

After the successful completion of the design, it is preceded towards implementation phase. There are two processes in this phase:

- Device programming and
- Software programming.

Device programming is the process to program FPGA board using Xilinx ISE software. This process basically will burn hardware design into FPGA board. Another task is to create test vector program in C. Creating this program is included in the software programming process.

### 1.6.4 Testing and Analysis

Final phase involved is the testing and analysis. During this phase, the output from hardware computation is compared with simulation output. This is to ensure that the design module works in line with output obtained from modelsim software.
1.6.5 VHDL and Simulation Software

VHDL is an acronym for VHSIC (Very high Speed Integrated Circuit) Hardware Description Language. It is a hardware description language that can be used to describe the structure and/or behavior of hardware designs. It is to model a digital systems at many levels of abstraction, ranging from the algorithmic level to gate level. The VHDL designs can be simulated and/or synthesized to create complex hardware designs.

Figure 1.7 Basic Modeling structure for VHDL.

Figure 1.7 shown above describes the structure of VHDL programming, digital component’s behavior in terms of programming language. A circuit or sub circuit described with VHDL code is called a design entity. The General structure of VHDL has two main parts:

- Entity declaration and
- Architecture.

For entity declaration, it specifies the input and output signals for the entity. The architecture part basically gives the details of the circuit in terms of programming. Figure 1.8 shows the VHDL design entity.
1.6.6 FPGA Advantage Software

FPGA Express provides logic synthesis and optimization, so one can automatically convert a VHDL description to a gate-level implementation in a given technology. This methodology eliminates the earlier gate-level design bottleneck. It reduces circuit design time and errors introduced when hand translating a VHDL specification to gates.

Before synthesize process, the design entry from VHDL must be analyzed first. If the design entry is error-free, then the synthesize process can be one. But, if the design entry still has an error, it must be corrected and analyzed. There are two steps in implementation of this synthesize process:

- ✓ Create Implementation and
- ✓ Export Net list.

Create Implementation is to create a circuit by FPGA Advantage Software and Export Net list is for the synthesized circuit from FPGA Advantage which is sent to Xilinx ISE.

1.6.7 Xilinx ISE Implementation Tool

This is a testing stage using Xilinx ISE software for the implemented system. Here, the circuit design will be simulated and the output can be seen at the timing
diagram. Based on the output from timing diagram, the circuit design can be detected as to whether it is functional or not.

1.6.8 Low-Power Techniques

Until the early mid-90s, low-power electronics were considered to be useful only in a few applications largely comprising small personal battery-powered devices such as watches, calculators, etc.. The combination of two major developments made low power design a key objective is: speed and silicon area. The first development was the advancement of submicron CMOS technologies[39,40] which produced chips capable of much higher operating power levels, in-spite of a dramatic drop in the energy dissipated per operation. The second development was the dramatic increase in market demand and increased capabilities with sophisticated portable electronics such as laptop computers and cellular phones. Since then, market pressure for low-power [41] devices has come from both ends of the “performance spectrum”[42]. Portable electronics drive the need for lower power due to a limited energy budget set by a fixed maximum battery mass [42]. The high-performance electronics also require lower power dissipation, but for a different reason that is to keep packaging and cooling costs reasonable.

Here, it is considered that circuits fabricated only using Complementary Metal Oxide Semiconductor (CMOS) [39,40] technologies because of its superior combination of speed, cost, availability, energy-efficiency, and density. Other available semiconductor technologies[43] such as BiCMOS, GaAs, and SiGe generally have higher performance, but also have characteristics such as significant leakage or high minimum-Vdd requirements that make them less suitable for low-power[44,45] applications.
1.7 Research Objectives

The proposed work is design and investigates the different architectures of FFT and their performance evolution can be evaluated by using the Electronic Design Automation (EDA) tool set,

✓ To analyze and implement an OFDM system used in MIMO-OFDM.
✓ To propose a novel design for MIMO OFDM system with single point FFT.
✓ To design an area efficient and low power modified R2MDC FFT architecture for MIMO OFDM and to compare it with others for performing Analysis.

1.8 Organization of Thesis

Chapter 1 begins with an introduction to MIMO OFDM system. Then basic concepts of FFT algorithms are discussed in detail. Also discussed the design utilities required for the implementation of MIMO-OFDM in HDL design and the CMOS design and their software and hardware requirements. The detailed literature survey and research objectives are given.

Chapter 2 provides the most commonly used FFT algorithms, e.g., the Cooley-Tukey and Sande-Tukey algorithms. Each computation step was given in detail for the Cooley-Tukey algorithms. Other algorithms like prime factor algorithm, split-radix algorithm, and WFTA are also discussed.

Chapter 3 deals with some low power techniques that are applicable at different abstraction levels.

Chapter 4 provides the several FFT implementation classes and also various FFT architecture are discussed. The programmable DSP or FFT-specific processors cannot meet the requirements in both high throughput and low power applications.
Algorithm-specific implementations, especially with pipelined FFT architectures are better in this respect.

**Chapter 5** includes the detailed description about different FFT design methodology such as Radix-2, Radix-4, Radix-8, Mixed Radix 4-2, Split Radix, R2MDC and Modified R2MDC FFT.

**Chapter 6** explains about the FPGA implementation of various Fast Fourier Transform in Altera cyclone II DE-2 hardware, Altera cyclone II DE-2 is one of the High end Version of FPGA. This provides analysis of various FFT algorithms. The implementation snapshots are also given.

**Chapter 7** explains about the FPGA implementation of various Fast Fourier Transform such as Radix-2, Radix-4, Split radix2/4, Mixed radix 4/2, R2MDC and Modified R2MDC in Virtex-II FPGA hardware for MIMO-OFDM transceiver. The several implementation snapshots are given.

**Chapter 8** provides performance evaluation of various FFTs, conclusion and important future scope.

The Pin detail of Altera cyclone II DE-2 hardware and HDL coding of various building blocks of the MIMO OFDM developed in the laboratory are given in the **Appendix**.