HIGH THROUGHPUT FFT/IFFT ARCHITECTURE FOR MIMO OFDM: A REVIEW

M.V.Nageswara Rao  
Department of ECE  
GMR Institute of Technology  
Rajam, Andhra Pradesh, India

Pogiri Revathi  
Department of ECE  
SVCET  
Etchela  
Andhra Pradesh, India

L.Govinda Rao  
Department of ECE  
GMR Institute of Technology  
Rajam, Andhra Pradesh, India

Abstract— Digital signal processing (DSP) is widely involved in the technology of everyday use. This prevalence drives the research interest for developing DSP algorithms for the VLSI implementations with better performance. Moreover, such implementations should offer lower power, higher speed and lower area. This leads the scope for novel algorithms, architectures for wireless communications applications like MIMO, OFDM, advanced channel coding. The Fast Fourier transform (FFT) processor is the key component in an OFDM system. In this paper, the hardware complexity, performance of various FFT architectures reported in the literature for OFDM applications are reviewed and compared. In this review, the architectures aimed for various wireless technologies such as IEEE 802.11ad, IEEE 802.11n, IEEE 802.11n etc.

Keywords— FFT/IFFT, OFDM, MIMO, Wi-Fi, WIMAX

I. INTRODUCTION

IEEE 802.11ac standard which was recently recommended is the fastest Wireless Fidelity (Wi-Fi) technology, offers faster data rates compared to the earlier technologies like IEEE 802.11ac. Association of industries and IEEE are working to enable the technology offers the data rates up to 7 Gbps for applications like high definition video streaming on smart phones. The high throughput processing can be achieved with multiple input multiple output (MIMO) systems. In the physical layer of MIMO system, orthogonal frequency division multiplexing (OFDM) technique is being widely adopted. A simplified block diagram of the OFDM communication system is shown in Fig.1.

In the Orthogonal Frequency Division Multiplexing, the Fast Fourier transform (FFT) processor is a key component. Here, data to be communicated is treated as a frequency-domain representation of the signal. IDFT (IFFT) is used to derive the time domain signal from data before transmission. At the receiver, FFT is used to get back the data (frequency domain representation of the signal) from the received signal.

Fig.1. A Complete OFDM communication system

II. LITERATURE REVIEW

OFDM technique is widely adopted in various applications like digital subscriber lines [1], [2], [3] for modems in wired-communication, and IEEE802.11 [4] Wi-Fi, IEEE802.16 [5], [6] 3GPP long term evolution (LTE) for wireless-communication modems used in base band data processing. To convert the modulated information from frequency domain to time domain Inverse fast Fourier transform (IFFT) is employed at the transmitter. In contrast, FFT collects samples from the time domain and are restored in the frequency domain at receiver. On other hand, Multiple Input Multiple
Output (MIMO) technique dramatically improves the data throughput. Therefore, MIMO-OFDM system can offer high data throughput for wireless communications [7].

During the last few decades, many researchers reported the FFT/IFFT architectures in literature for various OFDM applications. A mixed radix FFT with continuous flow data output is reported in [8], [9]. Here, seamless data processing is achieved by employing two memory blocks of size-N to generate output stream seamlessly. In this architecture, one of the memory block is used in the processing of current FFT/IFFT symbols, and the other memory block stores the earlier FFT/IFFT outcomes. Various pipeline FFT/IFFT architectures presented in the literature can be classified as single or multi path delay feedback (SDF or MDF) and single or multipath delay commutator (SDFC or MDFC) architectures [10]. In order to generate seamless output SDF schemes possess feedback paths to handle intermediate results in each pipe [11-20]. In contrast, various feed forward architectures based on radix-2k are reported in [1], [21], [22], [47], [54] [56]. The proposed pipelined architecture in [30] for RFFT is designed on the radix-2 algorithm. However, this technique cannot be generalized for higher radix algorithms. This issue is resolved by the technique proposed in [31]. In order to reduce the computational complexity numerous algorithms for FFT computation are reported in the literature, out of which the Cooley-Tukey radix-2 FFT [32] is very prevalent. Some standard FFT architectures in the literature [14], [35-39] are based on radix-4 [33], split-radix [34], radix-2k [21] algorithms, which are based on radix-2 algorithm. Radix-2 multi-path delay commutator (R2MDC) is proposed in [35], which is one of the most classical approaches for pipelined implementation of radix-2 FFT. By utilizing the R2MDC storage buffer effectively, the R2SDF architecture is achieved with reduced memory [35]. R4SDF [36] and R4MDC [37], [38] have been reported as radix-4 R2SDF and R4MDC. In [39], modified radix-4 algorithm is used to reduce the complexity of the R4MDC architecture, radix-4 single-path delay commutator (R4SDC) is proposed. Various parallel architectures for FFT are reported in [11], [22], [27], [40], [41]. These hardware architectures have high hardware complexity with poor utilization.

A method for decompose of discrete Fourier transform matrix reported in [13] enables the systematic implementation of the pipelined FFT processor. SDF schemes proposed in [11], [12] employ feedback paths to handle the intermediate results at each stage and to offer continuous processing. Here, FFT/IFFT processor generates the first output sample immediately after feeding the last input sample. Moreover, SDF schemes can process multiple data streams by employing one FFT/IFFT processor when the input data is properly scheduled. MDC topology is adopted in [1], which saves area compared to SDF FFT when there are multiple streams to process. However, MDC schemes require more memory for switch-box realization [23]. In MDC architecture, the radix-k butterflies are left unused till the arrival of kth input. Therefore, the butterfly hardware utility rate of the MDC FFT/IFFT computing core is 1/k, where it is 100% for SDF FFT/IFFT [1] [10]. Hence, (k−1)/k computing resources and memory are kept idle in MDC with single stream of input [1]. This utilization rate MDC can be made 100% by increasing the number of streams to r when the streams are properly scheduled and ordered to feed the pipelined FFT/IFFT. Hence, MDC FFT/IFFT is suitable in MIMO systems.

In MIMO system, to manage the multiple data streams, multiple functional blocks may be needed for the replicated processing of parallel input streams. However, this results in linearly increased FFT/IFFT complexity in MIMO systems. In earlier reported literature, the butterflies and memory are intuitively duplicated as per the number of data streams, and explored the techniques to exploit parallelism with diminished complexity. In MDC MIMO FFT/IFFT architecture, at each pipeline stage only one butterfly can be used [1]. However, proper scheduling and data reordering the input data of Ns streams is required. The radix of the butterfly employed in the FFT/IFFT architecture influence the hardware complexity. A butterfly with smaller radix simple in structure. On the other hand, to reduce the twiddle factor multiplications a high radix butterfly may be adopted. An MDC FFT/IFFT architecture is designed using radix-2k butterfly [24]. In [11], [15], [25-27] [29], the design of multi-path pipelined FFT processors provides a high throughput is reported. Nevertheless, for high throughput applications with a rate of over 2 GS/s throughput, the data-paths can be increased to eight or sixteen, results in increased hardware cost. To reduce the area and power consumption numerous FFT algorithms and schemes for dynamic scaling are presented in [15], [25-27]. Radix 2k FFT algorithm and architecture are proposed to reduce the number of complex multipliers. In [13], reduction of twiddle coefficient multiplications and to attain a simple butterfly concurrently, the radix k3 algorithms are suggested. Though the radix-2 based FFT processor is simple in structure, it involves many complex multipliers. The radix-4 based FFT
processor offers high throughput, but it results in high complexity [10].

Spectrum of real sample inputs viz. biomedical, speech, RADAR, audio signals and images is symmetric and nearly half the operations are redundant.

Table 1 Summary of various MIMO OFDM FFT/IFFT architectures reported in the literature

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Feature</th>
<th>Mandatory</th>
<th>Optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Channel Bandwidth</td>
<td>20 MHz, 40 MHz, 80 MHz</td>
<td>160 MHz, 80+80 MHz</td>
</tr>
<tr>
<td>2.</td>
<td>FFT Size</td>
<td>64,128,256</td>
<td>512</td>
</tr>
<tr>
<td>3.</td>
<td>Data subcarriers/Pilots</td>
<td>52/4, 108/6, 234/8</td>
<td>468/16</td>
</tr>
<tr>
<td>4.</td>
<td>Spatial streams and MIMO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.</td>
<td>Modulation types</td>
<td>BPSK, QPSK, 16QAM, 64QAM</td>
<td>256QAM</td>
</tr>
<tr>
<td>6.</td>
<td>Operating mode</td>
<td>Very high throughput</td>
<td>---</td>
</tr>
<tr>
<td>7.</td>
<td>MCS supported</td>
<td>0 to 7</td>
<td>8 and 9</td>
</tr>
</tbody>
</table>

*Modulation and Coding Schemes[58]

Pre- and post- transformed data reordering is crucial in a pipelined FFT/IFFT processor for multiple data streams. The pre- and post-transform data reordering methodologies for handling multiple data streams up to eight is presented in [57][58].

III. COMPARISON AND ANALYSIS

The parallel feedback architectures of radix-2^4 [16], [20] saved 50% of the adders and reduced the requirements of memory, while maintaining the same number of rotors when compared with the designs of radix-2^3 [24]. Among the 4-parallel architectures reported in the literature [24], the radix-2^3 feeds forward FFT [51] and the radix-4 feed forward FFT [37], [52], require the lower number of adders, rotators, and memory. Even though the architectures of radix-2^2 and radix-4 require the same quantity of hardware resources for 4-parallel samples, these resources possess different layouts. Radix-2^3 concedes the circuits for data management and rotators between radix-2 butterflies, where in radix-4, pairs of successive sets of radix-2 butterflies must essentially be together in order to form the radix-4 butterfly [24].

Comparing the architectures of 4-parallel radix-2^3 and radix-2^4 with the architecture of 4-parallel radix-4 feed forward FFT [18], [52], it is observed that radix-2^3 and radix-2^4 need the same number of adders and the same size of memory, but need less number of general rotators [24]. Furthermore, 4-parallel radix-2^3 feed forward FFT [24] saves 50% of the adders and 25% of the rotators as compared to radix 2^4 parallel feedback architectures [16], [20]. The 8-parallel radix-2^5 feed forward
FFT [24] have the capacity to save the number of rotators by 25% compared to radix-2 feed forward FFTs [53], adders by 50% and rotators by 25% with regard to feedback architectures [14].

Table- 3 Hardware resources for the given M-channel MIMO

<table>
<thead>
<tr>
<th>Topology</th>
<th>No. of Butterflies</th>
<th>Complex multipliers</th>
<th>Complex Adders</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radix-2 MDC</td>
<td>$M/2$</td>
<td>log$_2$N-2</td>
<td>2.log$_2$N</td>
</tr>
<tr>
<td>Radix-4 MDC</td>
<td>$M/4$</td>
<td>3.(log$_2$N-1)</td>
<td>8.log$_2$N</td>
</tr>
<tr>
<td>Radix-8 MDC</td>
<td>$M/8$</td>
<td>7.(log$_2$N-1)</td>
<td>(24+2TM$^*$).log$_2$N</td>
</tr>
<tr>
<td>Radix-2 SDF</td>
<td>M</td>
<td>log$_2$N-1</td>
<td>2.log$_2$N</td>
</tr>
<tr>
<td>Radix-4 SDF</td>
<td>M</td>
<td>log$_2$N-1</td>
<td>8.log$_2$N</td>
</tr>
</tbody>
</table>

*TM: Trial multipliers [10]*

The hardware complexity comparison of various pipeline structures [10] are compared in Table 3. Here, it can be observed that the radix-8 MDC is preferred structured for the FFT/IFFT for eight data stream applications. The hardware complexity of radix-R MDC is shared by R streams while the complexity of other structures is increased with the number of spatial streams [10].

IV. CONCLUSION

In this paper, the FFT/IFFT architectures aimed for various wireless technologies such as IEEE 802.11ac, IEEE 802.11n, WPAN, UWB, WiMAX reported in the literature for OFDM applications are reviewed and compared. The hardware complexity of various pipelined structured are compared. It is observed that the radix-R MDC FFT/IFFT for RxR MIMO-OFDM is more area efficient compared to other possible topologies. The radix-2$^8$ butterfly offers the structural simplicity and optimized hardware complexity.

V. REFERENCES


