### Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.0 IMPLEMENTATION OF 128 POINT FFT PROCESSOR USING R42SDF ARCHITECTURE</td>
<td>54</td>
</tr>
<tr>
<td>5.1 Introduction</td>
<td>54</td>
</tr>
<tr>
<td>5.2 Implementation of OFDM</td>
<td>55</td>
</tr>
<tr>
<td>5.3 Trade-offs for VLSI Design</td>
<td>56</td>
</tr>
<tr>
<td>5.3.1 Choice of Radix number</td>
<td>56</td>
</tr>
<tr>
<td>5.3.2 Parallel and Pipelining Operation</td>
<td>57</td>
</tr>
<tr>
<td>5.3.3 Low Power</td>
<td>57</td>
</tr>
<tr>
<td>5.4 CSD Multipliers</td>
<td>58</td>
</tr>
<tr>
<td>5.5 The FFT processor using R42SDF</td>
<td>58</td>
</tr>
<tr>
<td>5.6 The Mixed Radix 4/2 Butterfly</td>
<td>60</td>
</tr>
<tr>
<td>5.7 Results and Discussion</td>
<td>63</td>
</tr>
<tr>
<td>5.8 Summary</td>
<td>65</td>
</tr>
</tbody>
</table>
CHAPTER - 5
IMPLEMENTATION OF 128 POINT FFT PROCESSOR
USING R42SDF ARCHITECTURE

5.1 INTRODUCTION

This chapter illustrates that various types of FFT algorithms can be adopted for OFDM systems using Hardware Description Language (HDL) thereby decreasing the complexity of multiplication greater than 30% by employing CSD constant multipliers in the place of conventional programmable multipliers with 0.35 μm CMOS technology. Generally the sizes of FFT are modified in various applications of OFDM based upon the requirement. The proposed architecture for the FFT processor is Mixed Radix 4/2 single delay feedback (R42SDF) architecture along with the concept of bit reversal in output which can be obtained by index decomposition method. It is shown that this design architecture can reduce the cost of hardware, consumption of power and increase the speed of operation if compared to conventional multipliers.

The Fast Fourier Transform is an important component in the systems of OFDM. Recently, there is a huge requirement for a power and speed efficient FFT processor of longer length in various applications of OFDM [7]. The architectures used for adopting the FFT processor are - Single-memory architecture, Dual-memory architecture and Pipeline architecture. The first two types of architectures occupy less area than the third one. But their throughput is low with a requirement of higher clock frequency when compared to the third architecture. Generally the pipeline style is employed for applications of high throughput [6]. It needs \((\log_2 N)\) number of butterfly elements and manipulates faster than the architecture having single memory.
But consumption of power or area is larger in this type of architecture. Hence an appropriate area and power efficient FFT algorithm must be adopted to implement the pipeline style for various applications of OFDM. From the comparison of various types of algorithms of FFT, the Mixed Radix 4/2 algorithm with SDF style has been adopted for the design of the 128 point FFT processor for OFDM systems.

5.2 IMPLEMENTATION OF OFDM

In OFDM, the frequencies of sub-carrier are orthogonal to each other. By this, the total interference between the sub-channels is removed and there is no need to go for inter-carrier guard bands between the channels [5]. Due to this arrangement, the design of the transmitter and the receiver is simplified. The guard bands meant for demodulation of subcarriers in the system will not be required any more. The principle of orthogonality must be understood from the concept of stochastic processes. Two random processes are said to be orthogonal if they are not correlated to each other. This concept in communication system gives a clear picture of impact of orthogonality in OFDM.

Figure 5.1 shows the implementation of OFDM using FFT. In general, the FFT sinusoids form a group based on orthogonality and a signal can be expressed as an integration of those sinusoids in its vector space. The basic idea of FFT is that transformation relates its corresponding input signal with all of its sine functions in an essential manner. This is adopted at the transmitting and receiving ends to extend the input signal into a group of subcarriers and process them respectively. The subcarrier signals are then integrated to form the baseband signal from the transmitter. The concept of orthogonality and uncorrelation with respect to subcarriers is completely utilized in OFDM and good results are obtained. The correlation in FFT for a particular subcarrier observes energy only for that subcarrier due to the uncorrelated
nature of FFT functions. The energy from other subcarriers is not available due to uncorrelation. Due to this concept, the spectrums of subcarrier signals overlap without creating interference.

Fig. 5.1 Block Diagram for Implementation of OFDM

5.3 TRADE-OFFS FOR VLSI DESIGN

5.3.1. Choice of Radix Number

The time needed to calculate an N point FFT with a single butterfly is given by-

\[ T_{\text{FFT}} = \frac{N}{r} \cdot \log_r N \cdot T_r \]  \hspace{1cm} (5.1)

Where

- \( r \) - Radix number
- \( N / r \) - No of butterfly stages
- \( \log_r N \) - No of stages
- \( T_r \) - Time to calculate one butterfly element

This requirement for time can be reduced appropriately by the number of butterfly elements. It is understood from equation (5.1), the time required for computation
decreases with the increase in radix number and complexity of hardware. Moreover the number of manipulations is less for higher radix especially with large values of N.

5.3.2 Parallel and Pipelining Operation

Parallel processing is an important design parameter by which more number of processing elements can be employed to build large hardware. This concept can be used to compromise the speed of computation and power at the cost of additional hardware. Pipelining is a method to improve the speed or obtain low power in an efficient manner. In this method, a group of registers is placed in the forward view of a circuit in order to reduce the critical path. Due to this, the increase in clock speed and reduction in supply voltage can be obtained at the expense of system latency.

5.3.3 Low Power

Both parallel processing and pipelining can be employed to reduce the supply voltage which leads to low dissipation of power according to the equation (5.2).

\[ P = k \cdot \frac{C_L}{V_{DD}} \]  

(5.2)

Where

- \( k \) – Switching capacity
- \( C_L \) – Total capacitance of the circuit
- \( F \) – Frequency of operation
- \( V_{DD} \) – Supply voltage

However the circuit delay increases in a drastic manner when the supply voltage becomes equal to sum of the threshold voltages of respective transistors in CMOS process. The static power dissipation increases with the reduction in threshold voltage.
5.4 CSD MULTIPLIERS

The canonic sign digit (CSD) system is a method used to multiply one fixed-point number by another [1]. The scaler of CSD can be transformed to a multiplier of four-quadrants by means of control logic. They find use in signal processing filters, digital up-converters and down-converters. The circuit exploits the advantage of architecture of FPGA to contribute multipliers of higher efficiency. They are employed in Fourier transformations and other applications of DSP where a four-quadrant multiplier is needed. The block diagram of a CSD multiplier is shown in Figure 5.2.

![Block Diagram of a CSD Constant Multiplier](image)

**Fig 5.2 Block Diagram of a CSD Constant Multiplier**

5.5 THE FFT PROCESSOR USING R42SDF

A FFT is used to calculate the DFT and gives the same result as that of computing DFT in a direct way. The only disparity between them is that an FFT computes faster than that of DFT. The Discrete Fourier Transform (DFT) has a key position in several
fields of DSP such as analysis of correlation, analysis of spectrum, linear filtering etc. The DFT is given by equation (5.1).

$$Z[n] = \sum_{m=0}^{M-1} z[m]W^m_M$$

(5.1)

$$n = 0, 1, ..... M - 1$$

where $W^m_M = e^{\frac{2\pi j}{M}mk}$ is the coefficient of DFT. The computation of equation (5.1) needs (M-1) complex additions and ‘M’ complex multiplications for each value of DFT. The total calculation for all values of DFT needs M (M-1) complex additions and M^2 complex multiplications. But this will consume large amount of time for bigger values of ‘M’, because the time of computation is directly proportional to M^2. Therefore it is highly inevitable to decrease the count of additions and multiplications. This is a better approach to manipulate the DFT and the complexity of computation is reduced from $O(M^2)$ to $O(M \log_2 M)$.

Among the types of FFT architectures available, the pipeline architecture is comparatively preferred because higher performance with moderate cost can be achieved with this architecture. To have an area and power efficient FFT processor, the mixed radix 4/2 algorithm and the multiplier area reduction technique are employed. The typical pipeline architecture based on single delay commutator (SDF) and mixed radix 4/2 (R42SDF) for the suggested FFT processor is illustrated in Figure 5.3.
5.6 THE MIXED RADIX 4/2 BUTTERFLY

A mixed radix algorithm is an integration of different algorithms of radix number - different radices are used in different stages for computations of FFT. The manipulation of a 128-point FFT is done in two phases adopting one phase with processing elements of Radix-4, preceded by other phase of processing elements of Radix-2. Due to this, the mixed radix algorithm becomes somewhat complex when compared to radix-r, but gives a lot of choices in selecting the size of transformation. The Mixed-Radix depends on the integration of sub-transform modules and small length FFTs to form a large FFT. However the concept of ordering the output sequences is not offered by this algorithm.

The basic butterfly unit of mixed-radix 4/2 algorithm is shown in Figure 5.4. It employs both radix-2 algorithms and radix-2^2 algorithms which can obtain computations of FFT that are not power of four. The mixed-radix 4/2 unit computes four butterfly outputs with respect to X (0) ~ X (3). The butterfly unit has eight complex adders and three complex multipliers. To choose either Radix 4 algorithm or Radix 2 algorithm, four multiplexers are used which are indicated by the solid box shown in Figure 5.4.
A 16-point FFT processor using Mixed-Radix 4-2 butterfly with bit reversing is illustrated to verify the operation of design architecture. In Figure 5.4, the 16-point FFT comprises a total of four Mixed-Radix 4-2 butterfly elements. In the initial stage, the 16 point sequence of input is decomposed by two ‘8 bit’ groups corresponding to \( n3 = 0, 1, 2, 3, 4, 5, 6, 7 \) respectively. Each ‘8’ bit group becomes the corresponding input sequence for two Mixed-Radix 4-2 butterfly elements out of four elements available. Once the sequences of input enters the first Mixed-Radix 4-2 butterfly stage, the order of output sequences is indicated with small number besides each output line of the butterfly as shown in Figure 5.4.

Fig. 5.4 Butterfly for Mixed-Radix 4/2 FFT
The block diagram of suggested Mixed-Radix 4-2 butterfly element is shown in Figure 5.5. It comprises two butterfly elements of Radix-4 and four butterfly elements of Radix-2. In the initial stage, the input of two butterfly elements of Radix-4 are indicated by the expression from ‘B4’ \((0, n3, kj)\) to ‘B4’ \((i, n3, kl)\). They are combined with \(x(n3), x(N/4 \pm n3), x(N/2 \pm n3), x(3N/4 \pm n3)\) and \(x(N/8 \pm n3), x(3N/8 \pm n3), x(5N/8 \pm n3), x(7N/8 \pm n3)\) respectively to form the output sequence. The output group of data is multiplied appropriately by the twiddle factors, once each input group of data enters the first Radix-4 butterfly elements. Then, this sequence of output data is given as input to the second stage where Radix-2 butterfly elements are available. Again, the output sequence of data is multiplied by the corresponding twiddle factors in the second stage. The twiddle factors \(W_Q \ (1+k)\) is the only multiplier element in the Mixed-Radix butterfly unit. The order of the output sequence is 0, 4, 2, 6, 1, 5, 3 and 7 which is same as that of bit reversing inside the butterfly element. Hence suggested butterfly structure comprises four butterfly elements of Radix-2, two butterfly elements of Radix-4, a single multiplier element and a shift unit for the purpose of twiddle factors.

![Diagram of Mixed-Radix 4-2 Butterfly element for 128 point FFT](image)
5.7 RESULTS AND DISCUSSIONS

The OFDM block is synthesized with Xilinx FPGA of Virtex-II for various available configurations using the parametric nature of its core. The synthesis results for 128 point FFT based OFDM adopting various types of algorithms are given in Table 5.1. The analysis of 128-point FFT is selected to make a comparison with respect to the number of Flip-flops and CLB slices for various algorithms of FFT. Table 5.2 shows the simulation results for the 128 point FFT processor using Single Delay Feedback (SDF) architecture with the CSD multiplier. Figure 5.6 indicates the impact of DFF, CLB slices and Function generators for different algorithms of FFT. Figure 5.7 shows the synthesis results with the implementation of proposed 128 point FFT.

Table 5.1 CLB Slices, DFF & Function Generators for different FFT algorithms

<table>
<thead>
<tr>
<th>OFDM with 128 point FFT</th>
<th>CLB Slices</th>
<th>Utilization</th>
<th>DFF</th>
<th>Function Generator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radix-2</td>
<td>2793</td>
<td>7.21%</td>
<td>5786</td>
<td>3285</td>
</tr>
<tr>
<td>Radix-4</td>
<td>2536</td>
<td>6.01%</td>
<td>3482</td>
<td>5672</td>
</tr>
<tr>
<td>Split Radix</td>
<td>2477</td>
<td>6.10%</td>
<td>5764</td>
<td>4178</td>
</tr>
<tr>
<td>Mixed Radix 4-2</td>
<td>2172</td>
<td>4.87%</td>
<td>4734</td>
<td>5242</td>
</tr>
</tbody>
</table>
Fig 5.6 CLB slices, DFF, Function generators for various FFT Algorithms

Table 5.2 Synthesis and Simulation Results for 128 point FFT processor

<table>
<thead>
<tr>
<th>Type (12 x 12)</th>
<th>Area (cell)</th>
<th>Power (mW)</th>
<th>Latency (ns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programmable multiplier</td>
<td>2771 (100%)</td>
<td>364.6 (100%)</td>
<td>8.4 (100%)</td>
</tr>
<tr>
<td>Proposed CSD multiplier</td>
<td>1249 (45%)</td>
<td>141.9 (39%)</td>
<td>9.91 (118%)</td>
</tr>
</tbody>
</table>
5.8 SUMMARY

In this chapter, the suggested Mixed Radix-4/2 algorithm and its architecture R42SDF is regular, pipelined and can be extended for any $2^n$-point of FFT. The SDF style using mixed radix 4/2 algorithm permits the usage of CSD multipliers in the place of programmable multipliers in order to reduce the complexity in multiplication. Moreover, the design of an OFDM module for various types of algorithms is adopted and verified. It is inferred that there is a requirement for complex adders or multipliers by various blocks during the design process and hence they should be optimized appropriately to increase their reusability. The resources of hardware are utilized and the simulation results are highlighted for the synthesized architecture. It is suggested that CSD multipliers can contribute a decrement of greater than 30% with respect to power and area in terms of complexity in multiplication. The 128-point FFT using SDF seems to have a better performance against its requirements of hardware.