CHAPTER 4

PERFORMANCE ANALYSIS OF MODIFIED BIT PARALLEL MULTIPLIER BASED PIPELINED RADIX-4 SINGLE-PATH DELAY COMMUTATOR

Fourier Transform is the analysis of the frequency domain signal. It is the basis of many signal processing and communication applications. The Fourier transform has many applications, in the field of physical science that uses sine wave signals such as engineering, physics, chemistry and applied mathematics, will make use of digital and discrete data. 4G communications and other wireless system based communication are presently hot topics of research and development in the wireless communication and networking field. Digital signal processing FFT plays an important role in various applications like spectral analysis, filter simulation, auto-correlation and pattern recognition for the computation of DFT. While FFT reduces the computation time and improves the performance largely over the direct estimation of DFT, the FFT algorithm is one of the methods having high computational complexity in the physical layer of these communication systems. Hence, FFT architecture is presented for implementing the FFT computation efficiently.

4.1 Radix-4 FFT

The butterfly block implementation in DSP processor requires selection of radix. First, several FFT algorithms have been proposed such as Radix-2, Radix-4, Radix-8 and several other higher order radixes FFT. A Radix-4 FFT has log_4N stages. So, the 16 point Radix-4 FFT consists of two stages. Each stage has several butterfly operations. Each butterfly consists four inputs and four outputs. The Radix-4
decimation in frequency (DIF) algorithm divides an N-point DFT into four N/4-point DFTs. Each of these is separated into N/16-point DFTs giving sixteen N/16 DFTs and so on. The final stage produces a 4-point DFT which is simply a butterfly calculation for a Radix-4 FFT. The equation of Radix-4 FFT is represented as,

\[ Y_k = \sum_{n=0}^{\frac{N-1}{4}} y_n W_n^{nk}, 0 \leq k \leq N - 1 \]

\[ Y[k] = \sum_{n=0}^{\frac{N}{4}-1} y(n)W_n^{nk} + \sum_{n=\frac{N}{4}}^{\frac{N}{2}-1} y(n)W_n^{nk} + \sum_{n=\frac{N}{2}}^{\frac{3N}{4}-1} y(n)W_n^{nk} + \sum_{n=\frac{3N}{4}}^{N-1} y(n)W_n^{nk} \]

\[ Y[k] = \sum_{n=0}^{\frac{N}{4}-1} \left\{ y(n) + W_{n/4}^{nk}(y(n+N/4)) + W_{3n/4}^{Nk}(y(n+3N/4)) \right\} W_n^{nk} \quad (4.1) \]

Compared to Radix-2 FFT, computational stages can be reduced in Radix-4 FFT. Therefore, Radix-4 FFT has been used to boost the performances of OFDM system. The Radix-4 DIF butterfly is consists of three complex multipliers, as well as the other needed complex adder and subtractions. The reduction of the multipliers is obtained at the cost of an increase of three adders. Radix-4 FFT is an advanced technique for implementing Fourier transformation of discrete input signals. Radix-4 FFT algorithms have used in both DIT FFT and DIF FFT. In Radix-4 FFT, four numbers of input points can be processed at the same time, rather than processing two input points at the same time in radix-2 FFT.

For hardware realization of FFT, multi-bank memory and "in place" addressing approach are repeatedly used to speed up the memory access time and reduce the hardware consumption (Swapnil et al.,). For Radix-r FFT, r banks of memory are required to store data and each memory bank could be a two-port memory. Number of clock cycles needed for the Radix-r algorithm is given by \( (N/r) / (\log N) \), so for
calculating 64 point FFT Radix-4 algorithm needs 48 clock cycles where to perform the same 64 point FFT Radix-2 algorithm needs 192 clock cycles, which means Radix-2 algorithm is 4 times slower in comparison with Radix-4 algorithm. The structure of 4-point Radix-4 DIF FFT is shown in Figure 4.1

![Figure 4.1 Structure of 4 Point Radix-4 DIF FFT](image)

Normally, more than one hundred operations are necessary to estimate this Radix-4 butterfly. Due to high amount parallelism in a parallel processor, the Radix-4 DIF butterfly can be completed with 22 machine cycle. Compared to other devices, it is three or four times faster. The data flow structure of 16 point Radix-4 FFT is shown in Figure 4.2.
Figure 4.2 Data Flow Structure of 16 point Radix-4 FFT

4.2 Radix-4 Single Path Delay Commutator (R4SDC) FFT

Pipelined architectures are used for FFTs that need high data throughput (Kumar et al.). The basic principle of pipelined architectures is to collapse the rows, in place of the stages like in column architectures. The architecture is developed from radix butterfly elements with commutator in between. The advantages of these architectures are high data throughput, small area and simple control unit. A commutator is a switch for data between the radix butterfly stages in the pipeline. It stores parts of the FFT computations temporarily to perform the switching properly and much different pipelined architecture is used. Pipelined architectures have different
memory requirements, different complexities, different utilization, etc. R4SDC FFT is a Radix-4 model of R2SDC. Radix-4 single path delay commutator (Mookherjee et al.,) consists of butterfly units, commutator and complex multiplier with shift registers to give delays. Radix$2^2$ algorithm (Laguri et al.,) has the multiplicative complexity as Radix-4 and keeping the butterfly architecture of Radix-2 FFT. The structure of Single-path delay commutator (SDC) FFT is illustrated in Figure 4.3.

Figure 4.3 Structure of Single-path Delay Commutator

R4SDC architecture has higher computational efficiency. By using different radices, the key to the algorithm is dividing the FFT into different stages. The R4SDC FFT, which creates a long critical path. In the R4SDC architecture, the multipliers are used up to 75% using the programmable butterfly elements. Using multiple delay-Commutators, the memory requirement can be reduced to $2N-2$. SDC architecture is seldom used to process the single input data stream because it uses more memory resources than SDF and has a more complicated control. The output orders of the pipelined architectures are bit-reversed and they need to be reversed to normal order.
A pipelined architecture which can reduce the required number of adders by half and generate the output sequence in normal order. The main drawback of this architecture is that it breaks the data integrity and increases the implementation complexity of the computational units as the input sequence is divided into two parallel half-word sequences.

4.3 Proposed Modified BPM based Radix-4 Pipelined Single-path Delay Commutator (R4SDC) FFT

In this research work, the architecture of proposed modified BPM based 16-point R4SDC FFT is used to reduce the computational path, hardware utilization and power consumption. R4SDC enhances the utilization of the butterfly units by varying the processing elements. However, the memory requirement is increased. R4SDC architecture gives high speed, high throughput rate and low hardware complexity. In the proposed method, R4SDC FFT architecture is proposed for enhancing the architectural performances regarding VLSI parameters.

Radix-4 Single-path Delay Commutator consists of the processing element, commutator, delay elements, twiddle factor and multiplexer. The operation of adder and subtractor has been done in processing element. In this architecture, Modified carry select adder circuit is used for addition operation to improve the speed of the processor. To convert the signal from one form into another form is called commutator. The architecture of 16 point Radix-4 SDC FFT is shown in Figure 4.4.

The single path delay commutator (SDC) architecture is depending on the modified multiple path delay commutator. Each stage requires one complex multiplier, a delay commutator to correct the order of the data and a butterfly element. There are six shift registers. The multiplexer control selects the required data. The switch control signal is to reorder data.
In the proposed architecture, 16 point real and imaginary input values are given to the first stage then the commutator structure is used to convert from one signal into another signal. After finishing the commutator process, the addition and subtraction operation has been done. In the addition operation, modified carry select adder has been used.

In the modified carry select adder, to reduce the gate count of the full adder structure. After that the values are multiplied by using twiddle factor multiplication. Finally, got the first stage output by using multiplexer. Similar operation has been done in the second stage. The multiplexer is used to control the signals. Twiddle factor, which is used to reduce the shifter and adder values. The twiddle factor values are stored in the form of LUTs. The complex multiplier circuit, Read Only Memory has been increased. To rectify this problem, modified structure of bit-parallel multiplier has been designed. MBPM does not need Read Only Memory to store the values of twiddle

Figure 4.4 Architecture of 16 point Radix-4 Single-path Delay Commutator (R4SDC) FFT
factor, instead of using accumulation and shifter based logical functions to perform the twiddle factor multiplication.

4.4 Results and Discussion

By using the Verilog HDL, the design of proposed modified BPM based Radix-4 pipelined Single Path Delay Commutator (R4SDC) FFT has been developed. The results of simulation waveform have been done by using ModelSim 6.3C and performances of synthesis results are calculated by Xilinx ISE10.1i design tool. The simulated output of Existing R4SDC FFT and proposed modified BPM based Radix-4 SDC FFT is shown in Figure 4.5 and Figure 4.6.

Figure 4.5 Simulation Result of Existing R4SDC FFT
Figure 4.6 Simulation Result of Proposed Modified BPM based R4SDC FFT

The synthesis result of existing R4SDC FFT and proposed modified BPM based R4SDC FFT is to determine the area consumption is illustrated in Figure 4.7 and Figure 4.8. The Figure 4.7 shows that the number of LUTs and Slices required for
existing R4SDC FFT. The number of Lookup Tables (LUTs) is required for implementing existing R4SDC FFT is 1820 and required number of occupied slices is 1078. However, in the case of proposed modified BPM based R4SDC FFT, the required number of LUTs is reduced to 1756 and required slices are reduced to 1037, which is shown in Figure 4.8. The numbers of gate counts are reduced, so the number of occupied slices and LUTs has been reduced.

Figure 4.7 Synthesis Result of Existing R4SDC FFT to determine the area consumption
Figure 4.8 Synthesis Result of proposed modified BPM based R4SDC FFT using modified carry select adder to determine the area consumption
Similarly, the synthesis result for both existing R4SDC FFT and proposed modified BPM based R4SDC FFT regarding delay is shown in Figure 4.9 and Figure 4.10.

Figure 4.9 Synthesis Result of Existing R4SDC FFT to determine the delay consumption
Figure 4.10 Synthesis Result of Proposed modified BPM based R4SDC FFT to determine the delay consumption

In the case of existing R4SDC FFT, the time required for performing operation 26.816ns, which is reduced to 26.184ns in the case of proposed modified BPM based R4SDC FFT.
Similarly, the synthesis result for both existing R4SDC FFT and proposed modified BPM based R4SDC FFT regarding power consumption is shown in Figure 4.11 and Figure 4.12.

![Synthesis Result](image)

**Figure 4.11 Synthesis Result of Existing R4SDC FFT to determine the power consumption**
Figure 4.12 Synthesis Result of Proposed Modified BPM based R4SDC FFT to determine the power consumption
In the case of existing R4SDC FFT, the power consumption is 0.807W, which is reduced to 0.775W in the case of proposed modified BPM based R4SDC FFT.

The comparison of both existing R4SDC FFT and proposed modified BPM based R4SDC FFT regarding LUTs, Slices, delay and power values are shown in Table 4.1.

**Table 4.1 Comparison of Existing R4SDC and Proposed Modified BPM based 16-point R4SDC FFT**

<table>
<thead>
<tr>
<th>Types / VLSI Concerns</th>
<th>Number of occupied Slices</th>
<th>Number of LUTs</th>
<th>Delay(ns)</th>
<th>Power(W)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Existing R4SDC FFT</td>
<td>1078</td>
<td>1820</td>
<td>26.816</td>
<td>0.807</td>
</tr>
<tr>
<td>Proposed Modified BPM based 16-point R4SDC FFT</td>
<td>1037</td>
<td>1756</td>
<td>26.184</td>
<td>0.775</td>
</tr>
<tr>
<td>Percentage Reduction</td>
<td>3.8%</td>
<td>3.51%</td>
<td>2.35%</td>
<td>3.96%</td>
</tr>
</tbody>
</table>

Table 4.1 shows that the proposed modified BPM based R4SDC FFT offers 3.8% reduction in occupied slices, 3.51% reduction in LUTs, 2.35% reduction in delay and 3.96% reduction in power consumption than the existing R4SDC FFT. The performance evaluation of both existing R4SDC FFT and proposed modified BPM based R4SDC FFT is pictorially represented in Figure 4.13.
Figure 4.13 Performance Evaluation of Existing R4SDC FFT and Proposed Modified BPM based R4SDC FFT

4.5 Summary

In this research work, the structure of modified BPM based 16-point Radix-4 SDC FFT has been developed through the Very Large Scale Integration (VLSI) design environment. The modified BPM based 16-point FFT with the R4SDC had a maximum frequency 38.191MHz and utilized 1037 slices on the Spartan-3. The conventional R4SDC ran at 37.291MHz and utilized 1078 slices on the Spartan-3. The main aim of this research work is to reduce the area, delay and power consumption of developed modified BPM based Radix-4 SDC FFT. Proposed modified bit parallel multiplier based Radix-4 SDC FFT offers 3.8% reduction in number of slice utilization, 3.51%
reduction in number of LUTs, 63.2% reduction in delay and 3.96% reduction in power consumption than the existing Radix-4 SDC FFT. Thus, the proposed design is mainly useful for low power applications such as WLAN etc. For future work, plan to optimize the complex multipliers for more power reduction and FFT will be extended to large number of FFT computations for OFDM and SDR applications. Compared to Radix-2 FFT, Radix-4 FFT has better performance. But in Radix-4 FFT, the area consumption has been increased. So, the combined Radix-2, 4 and 8 FFT will be discussed in the next chapter for reducing area and also computational stages.