DESIGN AND SIMULATION OF A MEMORY BASED FFT USING BUTTERFLY RADIX-4 ALGORITHM

D. Aparna
Dept of Electronics and Communications,
NARAYANA ENGINEERING COLLEGE, Nellore, Andhra Pradesh, India.

G. Sindhura Bhargavi
Assistant Professor, Dept of Electronics and Communications,
NARAYANA ENGINEERING COLLEGE, Nellore, Andhra Pradesh, India.

Abstract—This concise presents a novel 4096-point radix-4 memory based fast Fourier transform (FFT). The proposed design takes after a contention free procedure that requires an aggregate memory of size N and a couple of extra multiplexers. The control is likewise straightforward, as it is created specifically from the bits of a counter. Aside from the low unpredictability, the FFT has been executed on a Virtex-5 field programmable gate array (FPGA) utilizing DSP cuts. The objective has been to decrease the utilization of circulated rationale, which is rare in the objective FPGA. With this reason, the majority of the equipment has been executed in DSP48E. Thus, the proposed FPGA is productive as far as equipment assets, as is appeared by the exploratory outcomes.

Index Terms—Fast Fourier transform (FFT), field programmable gate array (FPGA), memory based architecture, radix-4, VLSI.

I. INTRODUCTION

The fast Fourier transform (FFT) is a standout amongst the most critical calculations in the field of computerized flag handling, used to ascertain the discrete Fourier transform effectively. The FFT is a piece of various frameworks in an extensive assortment of uses. Some of the time the framework requests the calculation of the FFT at a high rate. For this reason, pipelined FFTs are mostly utilized [1], [2]. Memory-based FFTs comprises of a memory or bank of recollections that store the information.

These information are perused from memory, prepared by butterflies and rotators, and put away again in memory. This procedure rehashes iteratively until every one of the phases of the FFT calculation are ascertained. The upside of memory-based FFTs is the decrease in the quantity of butterflies and rotators, as they are reused for various phases of the FFT.

This concise presents a novel radix-4 memory-based FFT. The proposed configuration has a few points of interest. As for the past radix-4 approaches, it utilizes the base memory of N tests and a couple of extra multiplexers. Besides, the proposed approach has been actualized utilizing DSP48E cuts on a field programmable gate array (FPGA). The usage permits to incorporate the segments of the engineering in the DSP48E, which lessens the equipment, particularly the measure of circulated rationale. Therefore, the proposed approach is a reduced answer for the FPGA that takes leeway of the utilization of DSP48E cuts, leaving room in the FPGA for other complex and zone requesting components. This brief is organized as follows. Section II describes the proposed memory-based FFT. Section III explains the implementation using DSP slices. Section IV compares the proposed FFT to the previous memory-based FFTs. Section V presents an application where the proposed FFT has been used. Section VI shows the experimental results on the FPGA. Finally, Section VII summarizes the main conclusions of this brief.
II. PROPOSED MEMORY-BASED FFT

A. Basic Architecture

The fundamental design of the proposed 4096-point FFT is appeared in Fig. 1. The engineering utilizes radix-4 and processes the FFT calculation iteratively in six emphases, which originates from the way that in a radix-r memory-based FFT, the quantity of cycles is

\[ I_t = \frac{\log_2 N}{\log_2 r} = \frac{n}{\log_2 r}. \]  

The proposed configuration incorporates four recollections of N/4 tests in parallel rather than a solitary memory of N tests. This permits to peruse and compose information all the while in every one of the recollections, which lessens the inactivity and builds the throughput of the circuit. Subsequently, at each clock cycle, the PE gets and gives four examples in parallel, one from and to every memory.

B. Conflict-Free Access

As each of the four recollections are gotten to all the while, it must be guaranteed that the four examples handled in the PE each clock cycle originate from various recollections. This requests a contention free memory get to technique as appeared straightaway. The documentation in this brief is a similar one utilized as a part of past works [12], [13]. At first, examples are put away in normal request in the recollections. Tests 0 to N/4 − 1 are put away in MEM0, N/4 to N/2 − 1 in MEM1, etc. On the off chance that we number the examples from 0 to N − 1 with a record I ≡ bn−1bn−2 ... b0, the situation in memory of each example I is

\[ p_1 = b_{n-3}, b_{n-4} \ldots, b_0 | b_{n-1}, b_{n-2}. \]  

Fig. 2. Circuits for the permutations in the multiplexers. They only consist of multiplexers controlled by the bits of the counter. (a) σ1. (b) σ3.

Bits bn−1 and bn−2 show in which of the four recollections the example is put away, while bits bn−3,..., b0 are the address. In a radix-2 FFT, the butterfly at organize s works on tests whose files contrast in the bit bn−s [2]. For the principal organize/cycle, bits bn−1bn−2 are now in various recollections as per (2). For the second emphasis, tests that contrast in bits bn−3bn−4 need to touch base in parallel to the PE.

This is accomplished by ascertaining the change

\[ \sigma_1(u_{n-1}, \ldots, u_0) = u_{n-1}, \ldots, u_2 | u_{n-1} \oplus u_1, u_{n-2} \oplus u_0 \]  

on the information in position P1, which prompts the

\[ p_2 = b_{n-3}, b_{n-4}, \ldots, b_0, b_{n-1}, b_{n-2} | b_{n-3}, b_{n-4}. \]  

This is accomplished by ascertaining the change

\[ \sigma_2(u_{n-1}, \ldots, u_0) = u_{n-1}, \ldots, u_2 | u_{n-1} \oplus u_1, u_{n-2} \oplus u_0 \]  

on the information in position P1. The change σ1 is controlled by the bits cn−3 and cn−4. These are the two MSBs of the control counter with bits cn−3,..., c0. Along these lines, the control is straightforward, as it is taken specifically from the bits of the counter.

The second permutation

\[ \sigma_2(u_{n-1}, \ldots, u_0) = u_{n-1}, \ldots, u_2 | u_{n-1} \oplus u_1, u_{n-2} \oplus u_0 \]  

is the change of the multiplexers previously the recollections, where (⊕) speaks to the rationale XOR work. The circuit that computes this stage is appeared in Fig. 2(a). This stage decides the memory in which information will be put away. The change σ1 is controlled by the bits cn−3 and cn−4. These are the two MSBs of the control counter with bits cn−3,..., c0. Along these lines, the control is straightforward, as it is taken specifically from the bits of the counter.
Fig. 3. Generation of the memory address for MEM2. The counter is replicated twice and the corresponding bits are selected by the multiplexer. Note that the circuit only consists of NOT gates and the multiplexer.

is carried out by the memories. It only affects the content of the memories. The memory address is obtained as

\[ W_{i+1} = \overline{R_i} = c_2 \oplus m_1, c_2 \oplus m_0, \ldots, c_1 \oplus m_1, c_0 \oplus m_0 \]  
\[ c_{n-3}, \ldots, c_i \]  

where \( m_1, m_0 \) values are the bits that show the memory, \( c_i \) esteem are the bits of the control counter, \( Wi+1 \) is the written work address at cycle \( i + 1 \), and \( Ri \) is the perusing address at emphasis \( i \). Note that \( Wi+1 = Ri \). This implies at every cycle, information are composed in the addresses that are discharged in the past emphasis. This improvement permits to just utilize an aggregate memory of \( N \). For the primary cycle, \( W_1 = R_0 \) is equivalent to the control counter. The age of the memory address is appeared in Fig. 3 for MEM2, for which \( m_1m_0 = 10 \).

The third permutation

\[ \sigma_3(y_{k-1}, \ldots, y_0) = y_{k-1}, \ldots, y_3 \oplus y_1, y_2 \oplus y_0 \]  

is the permutation of the multiplexers after the recollections, appeared in Fig. 2(b). As \( \sigma_1 \), it just decides the memory in which information are put away.

Fig. 4 demonstrates the information administration for a 16-point FFT. The best piece of Fig. 4 demonstrates the information orders at the diverse phases of the circuit underneath. In the first place, information are put away in memory.

Fig. 4 demonstrates the substance of the distinctive locations in \( M_0 - M_3 \). These information are perused by \( R_0 = c_1c_0 \), i.e., information from the recollections are perused all together.

The information side step the multiplexer after the memory, which is crippled in this emphasis, and sources of info the butterfly. The composition address is \( W_1 = c_1c_0 \), which is equivalent to \( R_0 \) to maintain a strategic distance from memory get to clashes, as appeared in (8). The recollections are perused by \( R_1 = c_1 \oplus m_1, c_0 \oplus m_0 \).

C. Rotations

The rotations of the FFT are performed by three complex multipliers. Each of them is connected to a rotation memory, which is an

<table>
<thead>
<tr>
<th>Control bits</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>( c_1c_0 )</td>
<td>Output 0</td>
</tr>
<tr>
<td>00</td>
<td>A=M0+M2</td>
</tr>
<tr>
<td>01</td>
<td>A=M1+M3</td>
</tr>
<tr>
<td>10</td>
<td>A=M2+M0</td>
</tr>
<tr>
<td>11</td>
<td>A=M3+M1</td>
</tr>
</tbody>
</table>

ROM of \( N/4 \) tends to that store the sine and cosine segments of the pivot edge \( \phi = -m(2\pi/N) \), where \( m \in \{1, 2, 3\} \) is the memory and \( I \) is the memory address.

The memory address of the turn recollections is produced essentially from the control counter, as appeared in Fig. 5. The address is the same for each of the three recollections and is gotten by empowering the bits of the counter contingent upon the emphasis.

II. IMPLEMENTATION USING DSP SLICES
The proposed design has been actualized on a Virtex-5 XC5VSX95T FPGA. The VSX family is described by including an extensive number of DSP48E and a little measure of appropriated rationale. Consequently, we have sought after to boost the utilization of DSP48E and limit the utilization of conveyed rationale by actualizing on DSP48E every one of the components of the design with the exception of the recollections. Leeway of utilizing DSP cuts is that they can be timed at high clock frequencies.

Moreover, the execution on DSP cuts takes into consideration substantial word lengths without decreasing the clock recurrence, contrasted and outlines actualized in the appropriated rationale, where the clock recurrence might be diminished while expanding the word length.

The recollections MEM0 to MEM3 are actualized utilizing piece RAM (BRAM) recollections. Every memory has 1024 locations and each address stores an example of 24 + 24 bits for the genuine and fanciful parts, individually.

The module BTF0 comprises of two DSP48E in which the utilization of the multiplier has been handicapped and permits to work with four contributions of 24 + 24 bits. The Arithmetic rationale unit inside the DSP48E is designed in mode SIMD = TWO24. Along these lines the genuine and fanciful segments of the information are worked together.

The two arrangements of multiplexers in Fig. 2 would speak to a critical cost in the event that they were BTF0 figures the main crossed terms of the radix-4 butterfly and fuses the multiplexers in Fig. 2(b). In this manner, the BTF0 changes the tasks of the DSP48E contingent upon the two LSBs of the control counter , as per table1. The module BTF1 is undifferentiated from BTF0. It comprises of two DSP48E and ascertains the second piece of the radix _4 butterfly. The activities that are executed rely upon the bits cn_3cn_4 of the control counter , as appeared in Table II.

The module TWD in Fig.6 ascertains the augmentations by the twiddle factors. The twiddle factors are put away in the ROM memory, which is utilized as a part of the considerable number of emphasis of the FFT.

<table>
<thead>
<tr>
<th>Control bits</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>A+B</td>
</tr>
<tr>
<td>01</td>
<td>A-B</td>
</tr>
<tr>
<td>10</td>
<td>C+jD</td>
</tr>
<tr>
<td>11</td>
<td>C-jD</td>
</tr>
</tbody>
</table>

Fig. 6 demonstrates the design.

The recollections MEM0 to MEM3 are actualized utilizing piece RAM (BRAM) recollections. Every memory has 1024 locations and each address stores an example of 24 + 24 bits for the genuine and fanciful parts, individually.

The module BTF0 comprises of two DSP48E in which the utilization of the multiplier has been handicapped and permits to work with four contributions of 24 + 24 bits. The Arithmetic rationale unit inside the DSP48E is designed in mode SIMD = TWO24. Along these lines the genuine and fanciful segments of the information are worked together.

The two arrangements of multiplexers in Fig. 2 would speak to a critical cost in the event that they were BTF0 figures the main crossed terms of the radix-4 butterfly and fuses the multiplexers in Fig. 2(b). In this manner, the BTF0 changes the tasks of the DSP48E contingent upon the two LSBs of the control counter , as per table1. The module BTF1 is undifferentiated from BTF0. It comprises of two DSP48E and ascertains the second piece of the radix _4 butterfly. The activities that are executed rely upon the bits cn_3cn_4 of the control counter , as appeared in Table II.

The module TWD in Fig.6 ascertains the augmentations by the twiddle factors. The twiddle factors are put away in the ROM memory, which is utilized as a part of the considerable number of emphasis of the FFT.
While, in the primary cycle, the coefficients are perused one by one all together, in whatever is left of emphasis, the LSBs of the control counter are scratched off keeping in mind the end goal to decide the address [14], as appeared in Fig. 5. As the yields of BTF1 are rearranged, the twiddle factors are additionally given in this rearranged arrange. Amid the last cycle, the module TWD does not have to figure any pivot. Hence, the multipliers of this module are utilized to figure the squared extent of the mind boggling values, i.e., |C2 + S2|, with a specific end goal to decide the power at each yield recurrence.

IV. COMPARISON

Table III looks at different memory-based FFTs. In the memory-based FFTs, there is a tradeoff between the measure of assets of the engineering and the preparing time, TPROC. This relies upon the radix. The higher the radix, the bigger the PE.

A vast PE expands the zone, yet decreases TPROC.

\[ T_{PROC} = \frac{N}{r} \cdot T_{MEM} \] .......(9)

Where, TMEM is the entrance time to the memory.

In this manner, radix-2 FFTs in Table III need less assets [3]– [5], while the high-radix FFTs [8]– [10] accomplish higher throughput.

Radix-4 is in the center between radix-2 and high radices. In this manner, it displays a tradeoff amongst assets and execution. Among radix-4 plans [6], [7], [10], the proposed approach is described by the utilization of minimal measure of assets, while keeping an indistinguishable TPROC from other radix-4 plans. Specifically, it just needs an aggregate memory of N tests contrasted and 2N tests [6] or 2N(φ + 1) tests [10]. Note, notwithstanding, that the approach in [6] is proposed for nonstop stream while our approach isn't. Contrasted and [7] the proposed configuration lessens altogether the quantity of multiplexers to just 16 2-input multiplexers, contrasted and 16-input multiplexers and demultiplexers in [7], which is proportionate to 240 2-input multiplexers.

V. APPLICATION CASE

The proposed FFT has been utilized for range examination. Fig. 7 demonstrates the piece outline of the range analyzer.

The framework incorporates four channels. Each channel comprises of a limited drive reaction (FIR) channel, a devastation arrange (DEC), a 4096-point iterative FFT, and a periodogram examination.

The FIR channel is responsible for the data transmission adjustment. The separating is finished with suitable

<table>
<thead>
<tr>
<th>Approach</th>
<th>Radix</th>
<th>Parallel Process</th>
<th>Mem. Banks</th>
<th>Iterations</th>
<th>Cycles per It.</th>
<th>( T_{PROC} ) (Cycles)</th>
<th>Observations</th>
</tr>
</thead>
<tbody>
<tr>
<td>[3]</td>
<td>2</td>
<td>2</td>
<td>N</td>
<td>2</td>
<td>( \log_2 N )</td>
<td>N/2</td>
<td>( N(\log_2 N)/2 )</td>
</tr>
<tr>
<td>[4]</td>
<td>2</td>
<td>2</td>
<td>N</td>
<td>4</td>
<td>( \log_2 N )</td>
<td>N/2</td>
<td>( N(\log_2 N)/2 )</td>
</tr>
<tr>
<td>[5]</td>
<td>2</td>
<td>4</td>
<td>N</td>
<td>4</td>
<td>( \log_2 N - 1 )</td>
<td>N/4</td>
<td>( N(\log_2 N - 1)/4 + 1 )</td>
</tr>
<tr>
<td>[6]</td>
<td>4/2</td>
<td>4</td>
<td>2N</td>
<td>4</td>
<td>( \log_2 N/2 )</td>
<td>N/4</td>
<td>( N(\log_2 N)/8 )</td>
</tr>
<tr>
<td>[7]</td>
<td>4</td>
<td>4</td>
<td>N</td>
<td>4</td>
<td>( \log_2 N/2 )</td>
<td>N/4</td>
<td>( N(\log_2 N)/8 )</td>
</tr>
<tr>
<td>[8]</td>
<td>16/2</td>
<td>2</td>
<td>2N</td>
<td>4</td>
<td>( \log_2 N/3 )</td>
<td>N/4</td>
<td>( N(\log_2 N)/12 )</td>
</tr>
<tr>
<td>[9]</td>
<td>16</td>
<td>16</td>
<td>N</td>
<td>16</td>
<td>( \log_2 N/4 )</td>
<td>N/16</td>
<td>( N(\log_2 N)/64 )</td>
</tr>
<tr>
<td>[10]</td>
<td>r</td>
<td>r</td>
<td>2N(c + 1)</td>
<td>2r(c + 1)</td>
<td>( \log_2 N )</td>
<td>N/r</td>
<td>( N(\log_2 N)/r )</td>
</tr>
<tr>
<td>Proposed</td>
<td>4</td>
<td>4</td>
<td>N</td>
<td>4</td>
<td>( \log_2 N/2 )</td>
<td>N/4</td>
<td>( N(\log_2 N)/8 )</td>
</tr>
</tbody>
</table>

1 Parallel Process = Number of data that are processed in parallel. This means how many data are read in parallel from the memories in each clock cycle.

\( T_{PROC} \) = processing time, which is the product of the number of iterations and the number of cycles per iterations.
coefficients to abstain from associating, and to lessen the band obstructions and commotion.

After the channel, information are annihilated by a factor $L = 8$. The DEC piece gives one example each 20 ns to the FFT.

When every one of the examples are stacked into the FFT module, the estimation of the FFT begins. The FFT module applies a window to the information arrangement, and after that ascertains the FFT, lastly, the squared extent of the FFT (periodogram) is computed to get the energy of the signs.

VI. EXPERIMENTAL RESULTS
The design is simulated in Xilinx Software and developed in VERILOG language as shown in the figure.

![Figure 8. Proposed RTL Schematic](image1)

![Figure 9. RTL full schematic view](image2)

![Figure 10. Proposed Output Waveform](image3)

VII. CONCLUSION
The proposed 4096-point radix-4 memory-based FFT engineering presents a novel clash free access system. The new system requires the base measure of memory and a couple of multiplexers. This lessens the measure of equipment concerning the past radix-4 memory-based FFTs. Besides, the proposed FFT has been actualized effectively on a FPGA making utilization of the DSP cuts. The proposed configuration requires less dispersed rationale than the past outcomes on FPGA, while keeping a tant amount measure of DSP cuts and BRAM.

The delay, speed and design utilization summary (area) of proposed system is as shown in below table III.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed</td>
<td>-5</td>
</tr>
<tr>
<td>Delay</td>
<td>24.330nsec</td>
</tr>
<tr>
<td>Levels of Logic</td>
<td>23</td>
</tr>
<tr>
<td>Number of Slices</td>
<td>907 out of 960</td>
</tr>
<tr>
<td>Number of 4 Input LUTs</td>
<td>1631 out of 1920</td>
</tr>
<tr>
<td>Number of Bonded IOBs</td>
<td>478 out of 66</td>
</tr>
<tr>
<td>Number of MULT18X18SIOs</td>
<td>4 out of 4</td>
</tr>
</tbody>
</table>
REFERENCES