1536-Point FFT for 3GPP Long Term Evolution

October 2007, ver. 1.0 Application Note 480

Introduction

3GPP Long Term Evolution (LTE) is an ongoing project to improve the universal mobile telecommunication system (UMTS) standard to handle future requirements of mobile phones. The main targets include higher data rates, improved spectrum efficiency, improved coverage, and reduced latency.

In an LTE project, the system must support variable transmission bandwidths, including 1.25 MHz, 2.5 MHz, 5 MHz, 10 MHz, 15 MHz, and 20 MHz. Each transmission bandwidth corresponds to a fast Fourier transform (FFT) size of 128, 256, 512, 1024, 1536, and 2048 points, respectively.

This reference design implements a 1536-point FFT as a stand-alone core. This core satisfies the FFT size requirement of 1536 points for a bandwidth of 15 MHz in an LTE project. To meet the other FFT size requirement of two's/four's exponent for the rest of the bandwidth, refer to the Altera® FFT IP MegaCore® function, which is a parameterizable core that supports sizes from 32 to 16K points with radix-2 and radix-4 computations. Among these FFT sizes, most of the calculations are done with a radix-2/4 engine. The 1536-point discrete Fourier transform (DFT) is the only one using radix-3 in addition to radix-2 and radix-4 computation. In other words, Altera offers complete solutions to meet all FFT sizes required in the LTE project.

For more information about the LTE project, refer to the 3GPP project for Technical Specification Group Radio Access Network, Physical layer aspects for evolved Universal Terrestrial Radio Access (UTRA), 3GPP TR 25.814 v7.1.0 (2006-09).

This reference design has the following features:

- Verilog HDL-based design
- MATLAB bit-accurate simulation models
- Verilog testbench and Tcl script provided for the ModelSim® simulator
- Target clock rate:
  - @184.32MHz for Stratix® III device
  - @122.88MHz for Cyclone® III device
- Streaming 1536-point FFT with natural input and natural output order
- Avalon® Streaming (ST) compliant interfaces without backpressure

- Fixed point I/O representation to maintain precision
  - 16-bit-wide input data, 33-bit-wide output data
- 16-bit-wide twiddle factor precision
- Transform direction (FFT/IFFT) specifiable on a per-block basis

You can use the reference design to accelerate designs based on the 3GPP LTE project.

Background

This reference design implements a 1536-point FFT using the Altera FFT IP MegaCore function with a 512-point configuration and the variable streaming architecture. The FFT is a DFT algorithm that reduces the number of computations needed from $O(N^2)$ to O(NlogN) by decomposition. The DFT of a sequence *x(n)* is shown in Equation 1:

(1)
$$X(k) = \sum_{n=0}^{N-1} x(n) \bullet e^{(-j2\pi nk)/N}$$

In this case, $k = 0, 1, \ldots N-1$ and N is the transform length.

In this reference design, the transform length, *N*, is 1536. Using the decimation in time (DIT) method, the first step is to divide the input sequence into three sequences. Each sequence goes through the 512-point FFT computation. The last step applies a twiddle factor to the outputs of the second and third FFT computation and then combines the three sequences to get a final output result.

The mathematical equation of 1536-point FFT decomposition is shown in Equation 2:

$$(2)\quad X(k) = \sum_{n=0}^{1535} x(n) \bullet W_{1536}^{kn}$$

$$= \sum_{m=0}^{511} x(3m) \bullet W_{1536}^{k(3m)} + \sum_{m=0}^{511} x(3m+1) \bullet W_{1536}^{k(3m+1)} + \sum_{m=0}^{511} x(3m+2) \bullet W_{1536}^{k(3m+2)}$$

$$= \sum_{m=0}^{511} x(3m) \bullet W_{512}^{km} + W_{1536}^{k} \sum_{m=0}^{511} x(3m+1) \bullet W_{512}^{km} + W_{1536}^{2k} \sum_{m=0}^{511} x(3m+2) \bullet W_{512}^{km}$$

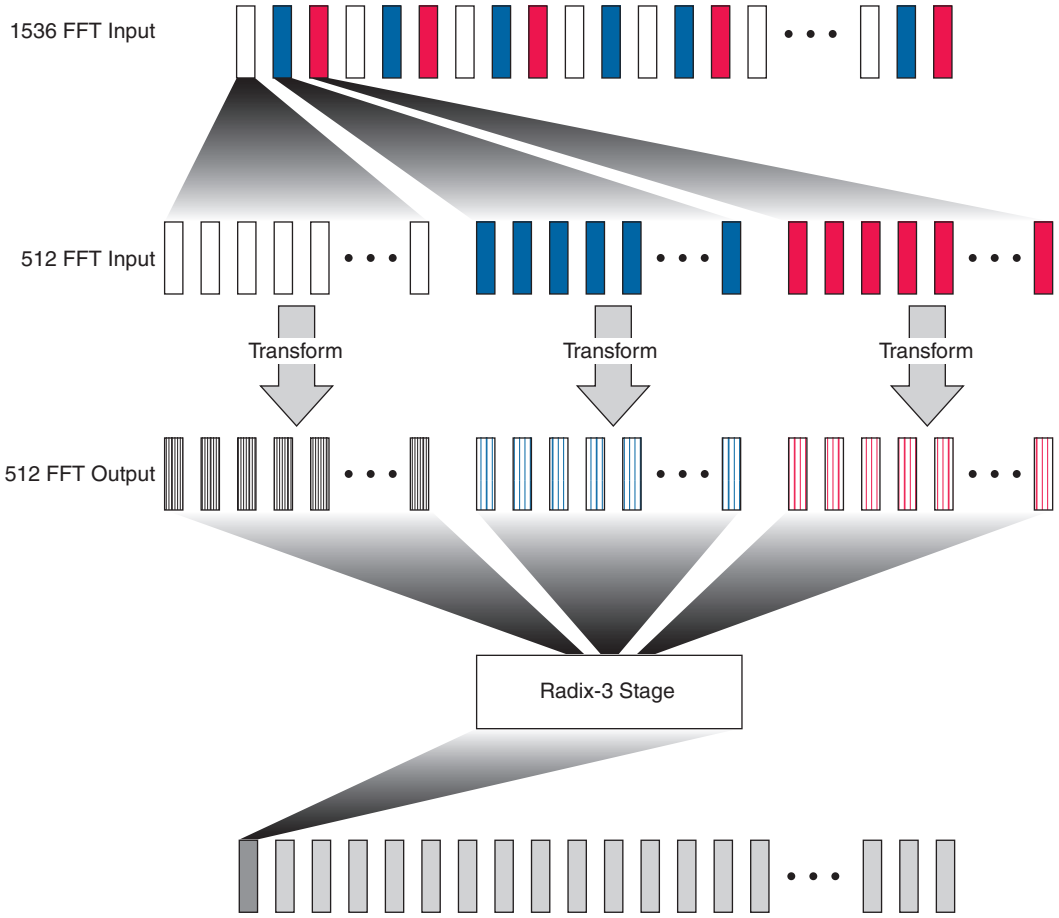
In this case, $W_N^{kn} = e^{(-j2\pi kn)/N}$

A 1536-point FFT can be decomposed into three 512-point FFTs followed by a single radix-3 combinational stage. Using the DIT method, the design breaks down the input sequence into top, middle, and bottom samples that feed into one 512-point FFT block consecutively using the FFT IP MegaCore function. Twiddle factors (*W*) are applied to the outputs of the second and third 512-point FFTs. The results from the FFT IP MegaCore function are recombined and reordered to obtain the final FFT output. This process is shown in Figure 1. Similar to the FFT IP MegaCore function, the reference design uses an Avalon Streaming compliant data transfer without backpressure.

For more information about Avalon Streaming interfaces, refer to the *Avalon Streaming Interface Specification*.

For more information about the Altera FFT IP MegaCore function, refer to the *FFT MegaCore Function User Guide*.

Figure 1. Decomposing and Recombining the 1536-Point FFT



I/O Data Flow Architecture

The variable streaming architecture allows continuous streaming of input data and produces a continuous stream of output data.

You can change direction on a block-by-block basis by asserting or deasserting the `inverse` signal and applying the `sink_sop` pulse (concurrent with the first input data sample of the block).

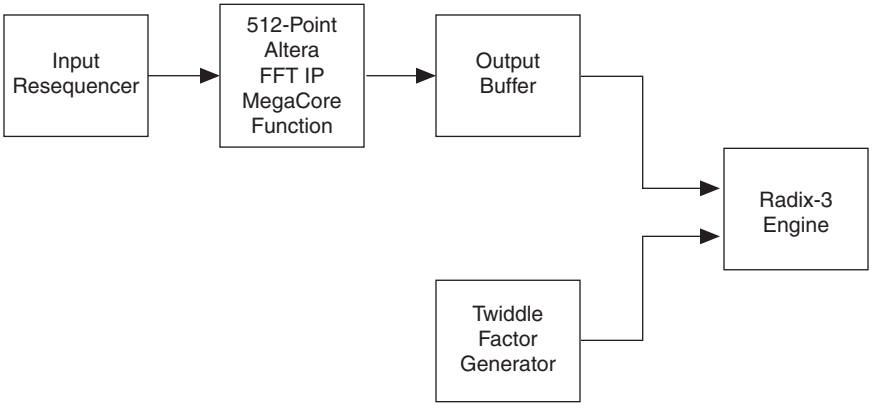
When the FFT has completed the transform of the input block, it asserts `source_valid` and outputs the complex transform domain data block. The FFT function asserts `source_sop` to indicate the first output sample. The input and output of the FFT are both in natural order, that is, 1..*N*.

For more information about the possible combinations of input and output order, refer to the "I/O Data Flow Architectures" section of the *FFT MegaCore Function User Guide*.

Functional Description

There are five major building blocks in the 1536-point FFT system (Figure 2).

Figure 2. 1536-Point FFT Block Diagram



- **Input Resequencer**

  The input resequencer parses the interleaved data sample into three sets of serial data and then bit-reverses each set of data before entering the FFT.

When the input resequencer receives a block of 1536 input samples, it bursts out the three sequential sets of 512 output samples and then waits for the next block of 1536 input samples. To maintain a streaming data flow, the resequencer uses a double buffer. Data is written into one buffer while the FFT reads from the other buffer in bit-reversed order. The depth of the buffer in the input resequencer is 3072 words (2 × 1536 words).

- **FFT IP MegaCore Function**

  To optimize resource usage, this reference design uses the same 512-point FFT three times consecutively. The FFT uses bit-reversed input and natural output in engine-only mode, which minimizes the memory usage and latency.

  For more information about the possible combinations of input and output order, refer to the *FFT MegaCore Function User Guide*.

- **Output Buffer**

  The three sets of FFT output are written into the output buffer sequentially while reading from it concurrently three times. The output buffer consists of three smaller buffers. Each of the smaller buffers also uses a double buffer to guarantee streaming throughput. The depth of the output buffer is 3072 words (2 × 3 × 512 words).

- **Twiddle Factor Generator**

  As described in Equation 1, twiddle factors are applied to the second and third FFT outputs in the radix-3 computation. Only the first quadrant of sine values is stored in the ROM to generate the whole cycle of twiddle factors, including both sine and cosine values. The timing of the twiddle factors is aligned with the real and imaginary data samples from the output buffer.

- **Radix-3 Engine**

  The radix-3 engine multiplies the three input data samples by their respective twiddle factors and sums the results to get the final result of the 1536-point FFT computations.

Getting Started

This section contains the following information:

- System Requirements
- Installing the Reference Design
- Running the Reference Design
- Synthesis Results

System Requirements

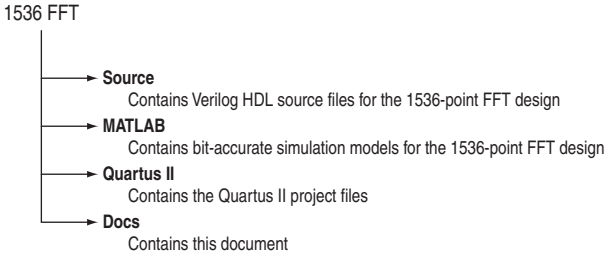
To run the reference design you must have the following software installed:

- Quartus® II Software version 7.1 and higher
- FFT MegaCore Function version 7.1 and higher
- MATLAB version R2007a
- ModelSim version 6.0b and higher

Installing the Reference Design

To install the reference design, extract the **1536FFT.zip** file to the specified directory. Figure 3 shows the directory structure after installation.

Figure 3. Directory Structure



Running the Reference Design

You can run the reference design using either ModelSim or MATLAB simulation.

ModelSim Simulation

The testbench and ModelSim Tcl script are provided for functional simulation. The testbench reads the real and imaginary input data samples from **real_input.txt** and **imag_input.txt** and writes the real and

imaginary output data samples to **real_output_ver.txt** and **imag_output_ver.txt**. These output results can be compared with the MATLAB simulation results.

To run the functional simulation for the 1536-point FFT design using the ModelSim tool, follow these steps:

1. Run the ModelSim software.
2. On the File menu, click **Change Directory** and select **..\1536FFT\source**.
3. Open the **test_fft1536.tcl** file. If necessary, update the paths of the libraries to point to the directory where the libraries are installed.
4. On the Tools menu, click **Execute Macro** and select the **test_fft1536.tcl** Tcl script.

MATLAB Simulation

To verify the RTL simulation results are correct, you can compare them with the results from the provided MATLAB models found in the **..\1536FFT\MATLAB** directory.

To run the MATLAB models, perform the following steps:

1. In MATLAB, change the directory to the **..\1536FFT\MATLAB** directory.
2. To generate the twiddle factors for the 1536-point FFT (the twiddle factor precision is set to 16 bits), type the following command in the MATLAB workspace:

```
>> [coshex, sinhex, twiddle] = gen_twiddle_factor(32768, 16); ↵
```

3. To run the MATLAB model, type the following command in the MATLAB workspace:

```
>> fft_1536_tb ↵
```

If the ModelSim RTL simulation results match the results from the MATLAB models, the following message is displayed:

Simulation passes – Results Match

If the results do not match, the following error message is displayed:

Simulation fails – Results Mismatch

The script **gen_input.m** generates a user-specified input stimulus and writes to **real_input.txt** and **imag_input.txt** for ModelSim RTL simulation.

Synthesis Results

The design was synthesized using the Altera Quartus II software version 7.1 targeting the EP3SL50F484C2 device. Table 1 shows the synthesis results (your results may vary slightly):

Table 1. Synthesis Results

Function	ALUTs	ALMs	DSP 36x36	DSP 18x18	M9Ks
Resequencer (double buffer)	44	30	0	0	16
512-point FFT	3078	3780	0	28	8
Output buffer (double buffer)	45	26	0	0	21
Twiddle factor generator	115	66	0	0	2
Radix-3 combination	435	280	8	0	0
Total	**3717**	**4182**	**8**	**28**	**47**

Conclusion

In the application of 3GPP RACH detection in an LTE project, other FFTs of size 3×2^n points may also be used. This reference design shows how you can easily and efficiently implement these radix-3 functions with the Altera FFT IP MegaCore function.

Document Revision History

Table 2 shows the revision history for this application note.

Table 2. Document Revision History

Date and Document Version	Changes Made	Summary of Changes
October 2007 v1.0	Initial release.	—

101 Innovation Drive
San Jose, CA 95134
www.altera.com
Technical Support:
www.altera.com/support/
Literature Services:
literature@altera.com