

## Introduction

The Altera® orthogonal frequency division multiplexing (OFDM) kernel can be used to accelerate the development of wireless OFDM transceivers such as those required for the deployment of mobile broadband wireless networks based on the *IEEE 802.16* standard.



For more information on *IEEE 802.16*, refer to the *IEEE Standard for Local and Metropolitan Area Networks, Part 16: Air Interface for Fixed Broadband Wireless Access Systems*, IEEE P802.16-REVd/D5-2004, May 2004.

OFDM is one of the key physical layer components associated with mobile worldwide interoperability for microwave access (WiMAX) and is widely regarded as an enabling technology for future broadband wireless protocols including the 3GPP and 3GPP2 long term evolution standards.

This reference design demonstrates the suitability of Cyclone® II, Stratix® II, and Stratix III, FPGAs for implementing OFDM symbol-level modulation.

## Key Features of the Reference Design

The OFDM kernel has the following key features:

- Support for 128, 512, 1K, and 2K FFT sizes to address variable bandwidths from 1.25 to 20 MHz
- Parameterizable design
- Optimized for efficient use of Cyclone II, Stratix II, and Stratix III device resources

You can use the reference design as a starting point to accelerate designs based on OFDM modulation such as the WiMAX or 3GPP long term evolution protocol. Altera supplies the reference design as clear-text register transfer level (RTL) HDL.

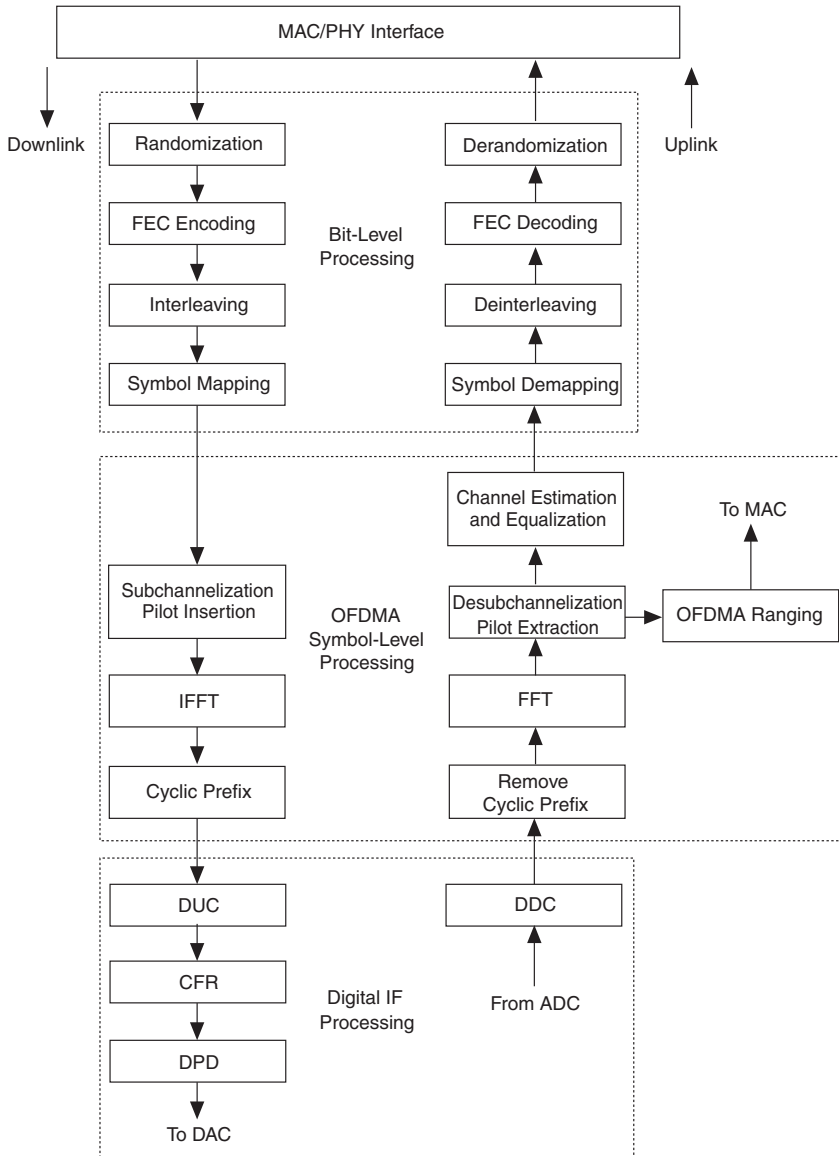


Please contact your local **Altera sales representative** for a copy of the reference design.

# WiMAX Physical Layer

Figure 1 shows an overview of the IEEE 802.16e-2005 scalable orthogonal frequency-division multiple access (OFDMA) physical layer (PHY) for WiMAX basestations.

Figure 1. WiMAX Physical Layer Implementation



The OFDM Kernel refers to the inverse fast Fourier transform (IFFT) and cyclic prefix insertion blocks in the downlink flow and the FFT and cyclic prefix removal blocks in the uplink flow.

To support orthogonal frequency-division multiple access (OFDMA) an extension to the OFDM kernel is required that allows each user to be allocated with a portion of the available carriers. This process is referred to as subchannelization.

Altera's WiMAX building blocks include bit-level, OFDMA symbol-level, and digital intermediate frequency (IF) processing blocks. For bit-level processing, Altera provides symbol mapping/demapping reference designs and support for forward error correction (FEC) using the Reed-Solomon and Viterbi MegaCore® functions.

The OFDMA symbol-level processing blocks include reference designs that demonstrate subchannelization and desubchannelization with cyclic prefix insertion supported by the fast Fourier transform (FFT), and inverse fast Fourier transform (IFFT) MegaCore functions. Other OFDMA symbol-level reference designs illustrate ranging, channel estimation, and channel equalization.

The digital IF processing blocks include single antenna and multi-antenna digital up converter (DUC) and digital down converter (DDC) reference designs, and advanced crest-factor reduction (CFR) and digital predistortion (DPD).

This application note illustrates the functionality and implementation of the OFDM kernel.



For more information on the multiple access aspect associated with the OFDMA modulation scheme, refer to the following application notes:

- *AN-412 A Scalable OFDMA Engine for WiMAX*
- *AN-450 Uplink Desubchannelization for WiMAX*
- *AN-451 Downlink Subchannelization for WiMAX*



For more information on related Altera WiMAX solutions, refer to the following application notes:

- *AN 421: Accelerating WiMAX DUC & DDC System Designs*
- *AN 430: WiMAX OFDMA Ranging*
- *AN 434: Channel Estimation & Equalization for Mobile WiMAX Basestations*
- *AN 439 Constellation Mapper and Demapper for WiMAX*

## OFDM Kernel Functionality

The physical layer is based around OFDM modulation. Data is mapped in the frequency domain onto the available carriers. For this data to be conveyed across a radio channel, it is transformed into the time domain using an inverse fast Fourier transform (IFFT) operation. To provide multipath immunity and tolerance for synchronization errors, a cyclic prefix is added to the time domain representation of the data.

Multiple modes are supported to accommodate variable channel bandwidths. This scalable architecture is achieved by using different FFT/IFFT sizes. This reference design supports transform sizes of 128, 512, 1,024, and 2,048.

## Implementation

FPGAs are well suited to FFT and IFFT processing because they are capable of high speed complex multiplications.

DSP devices typically have up to eight dedicated multipliers, whereas the Stratix III EP3SE110 FPGA has 112 DSP blocks that offer a throughput of nearly 500 GMACs and can support up to 896 18x18 multipliers, which is an order of magnitude higher than current DSP devices.

[Figure 2 on page 5](#) shows the embedded digital signal processing (DSP) blocks in an Altera Stratix III device.

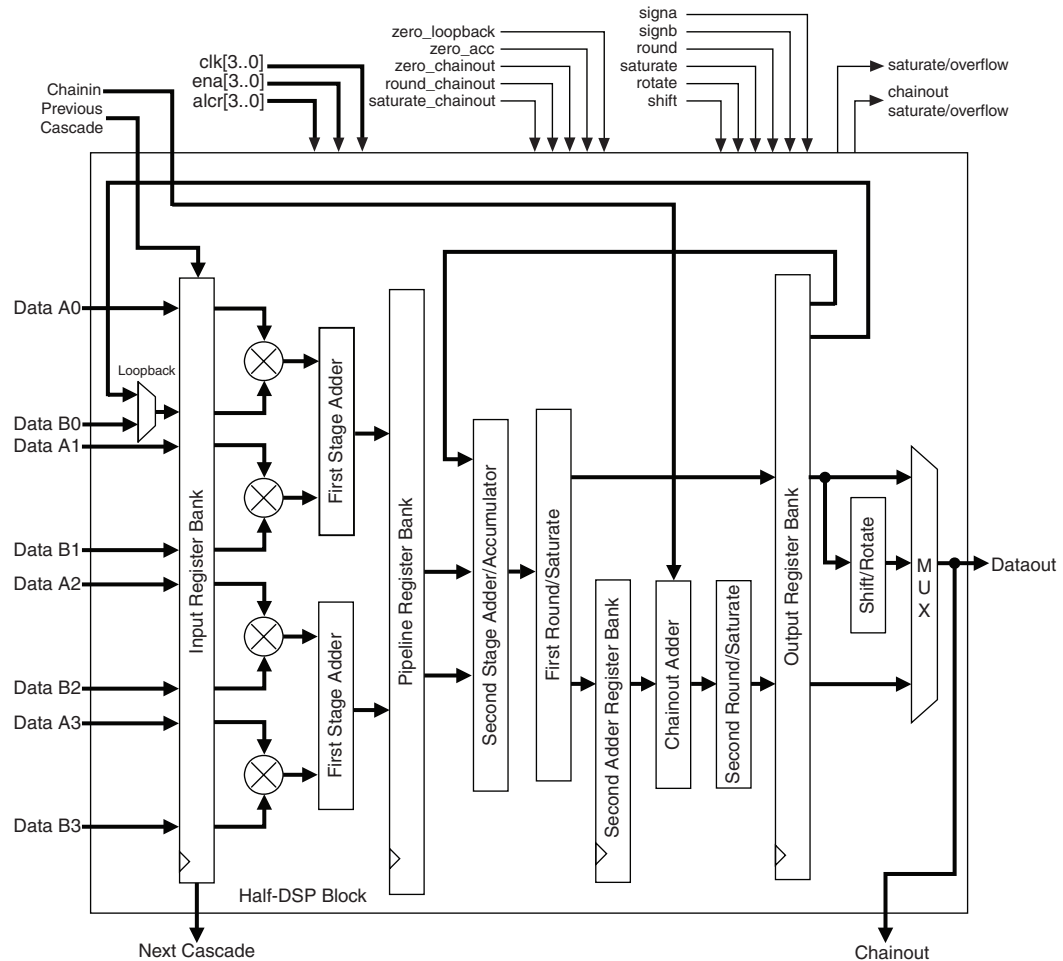
Such a massive difference in signal processing capability between FPGAs and DSP devices is further accentuated when dealing with basestations that employ advanced, multiple antenna techniques such as space time codes (STC), beam forming, and multiple-input multiple-output (MIMO) schemes.

The combination of OFDMA and MIMO is widely regarded as a key enabler of higher data rates in current and future WiMAX and 3GPP long term evolution (LTE) wireless systems. When multiple transmit and receive antennas are employed at a basestation, the OFDMA symbol processing functions have to be implemented for each antenna stream separately before MIMO decoding is performed.

The symbol-level complexity grows linearly with the number of antennas implemented on DSPs that perform serial operations. For example, for two transmit and two receive antennas the FFT and IFFT functions for WiMAX take up approximately 60% of a 1-GHz DSP core when the transform size is 2,048 points.

In contrast, a multiple antenna-based implementation scales very efficiently when implemented with FPGAs. Using Altera devices, you can exploit parallel processing and time-multiplexing between the data from multiple antennas. The same 2x2 antenna FFT/IFFT configuration uses less than 10% of a Stratix II 2S60 device.

**Figure 2. Embedded DSP Blocks Architecture in Stratix III Devices**



## Design Methodology

Altera provides the reference design as clear text VHDL. The reference design also demonstrates the use of the FFT MegaCore function.

To accelerate integration with Altera intellectual property (IP) or other subsystems, the interfaces support the Altera Avalon® Streaming (Avalon-ST) interface specification.



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

Altera has verified the RTL behavior against a fixed point model of the algorithms. The reference design includes RTL testbenches that stimulate the designs using golden reference data from the fixed point model and check for correct functionality.

## Functional Description

The OFDM kernel handles the FFT operations and cyclic prefix addition and removal. The FFT size is a parameter that you must specify at synthesis time, but you can change the guard interval at run time.

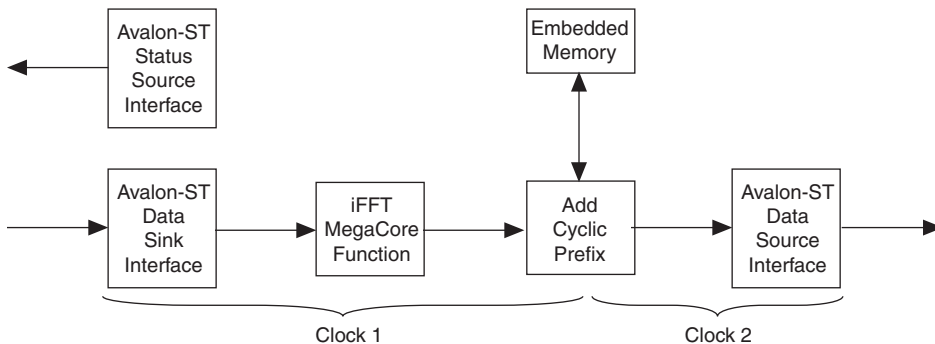
### Downlink Transmit

The downlink OFDM kernel module performs an inverse Fourier transform of the frequency domain input data and adds a cyclic prefix to the resulting time domain data.

The cyclic prefix addition block contains a controller that buffers the output packets from the FFT, and adds the appropriate proportion of the end of the output packet to the beginning of the output packet. As this requires a fairly significant memory resource, the hardware architecture has been designed so that the embedded memory may be shared with the uplink OFDM kernel if the modem is operating in time division duplex (TDD) mode.

Figure 3 shows a block diagram of the downlink OFDM kernel.

**Figure 3. Downlink OFDM Kernel Block Diagram**

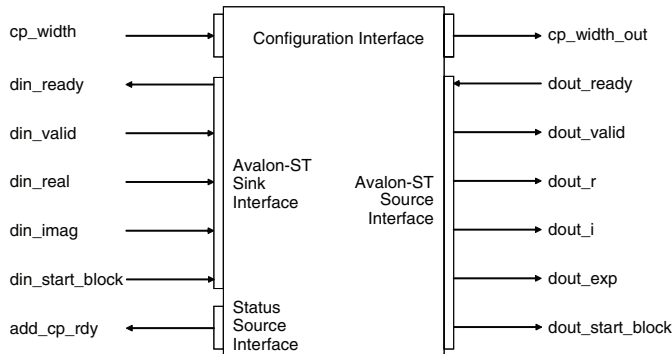


### Interface Specifications

The block has two clock domains. In addition, there are two reset ports; one for each clock domain. The reset ports are active low.

Figure 4 shows the downlink OFDM kernel interfaces.

**Figure 4. Downlink OFDM Kernel Interfaces**



The input interface has the following features:

- Avalon-ST data sink and status source
- Ready signal latency of one cycle—the earliest time valid data may be presented to the block after ready has been signaled is one clock cycle

The output interface has the following features:

- Avalon-ST data source
- Ready signal latency of four cycles—the block responds to new data or stops delivering data four cycles after an event on the ready signal
- Support for back pressure
- Dynamically changeable cyclic prefix

Figure 5 shows the downlink kernel input interface timing diagram.

**Figure 5. Downlink Kernel Input Interface Timing Diagram**

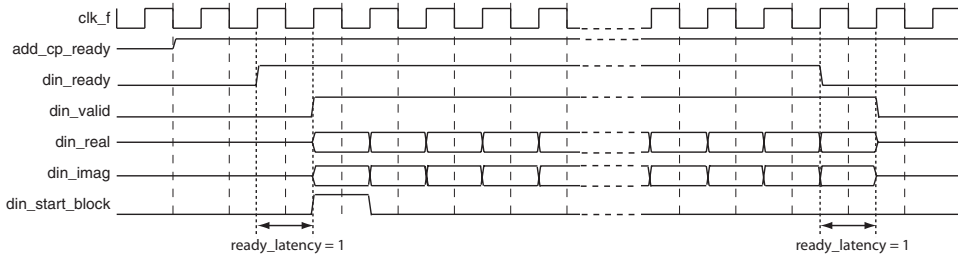
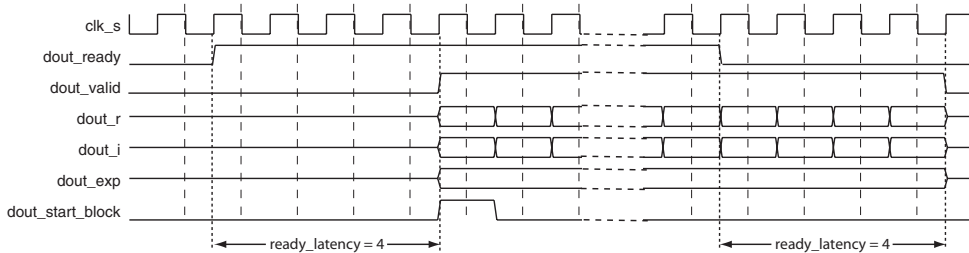


Figure 6 shows the downlink kernel output interface timing diagram.

**Figure 6. Downlink Kernel Output Interface Timing Diagram**



The FFT size can be configured using the LOG2\_N\_FFT parameter. This parameter should be set to the log of base two of the desired FFT size. The MADDR\_WIDTH parameter should be set to LOG2\_N\_FFT + 1. The real and imaginary data buses are signed fixed point format and the width is configurable using the WIDTH generic. The exponent of the output data is fixed at 6 bits. Table 1 shows the guard interval is configured using the cp\_width bus, and the settings:

<i>Table 1. Guard Interval Configuration</i>	
Guard Interval Length	cp_width[1..0]
1/4	00
1/8	01
1/16	10
1/32	11



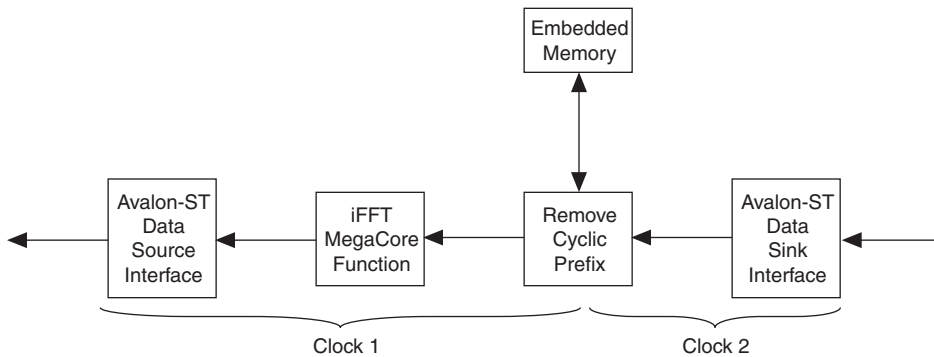
The source should only commence feeding a new symbol of data into the kernel, if both `din_ready` and `add_cp_rdy` are high. The source should continue to feed in the remaining symbol of data if `din_ready` is high. (In reality, once the kernel has accepted the first subcarrier for a symbol, it accepts the remaining subcarriers on subsequent clock cycles without deasserting `din_ready`.)

### *Uplink Receive*

The uplink OFDM kernel module performs an FFT of the time domain input data and removes the cyclic prefix. The Avalon-ST start of packet pulse should specify the start of the cyclic prefix. The remove cyclic prefix block ignores the data during the cyclic prefix and writes the remaining samples to the FFT input buffer.

Figure 7 shows a block diagram of the uplink OFDM kernel.

**Figure 7. Uplink OFDM Kernel Block Diagram**



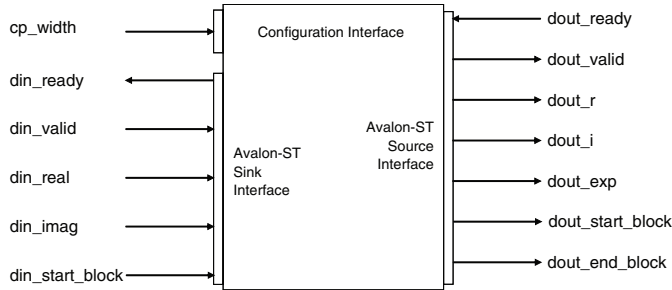
Because the channel characteristics can change, it is possible that the start of the packet pulse is not always after the start of the cyclic prefix time. The hardware has been designed to deal with this scenario but with the constraints that the variation of the pulse must be within the cyclic prefix time and that the start pulse will not be before the preceding symbol has been fully clocked in.

### **Uplink Interface Specifications**

The block has two clock domains. In addition, there are two reset ports; one for each clock domain. The reset ports are active low.

Figure 8 shows the uplink OFDM kernel interfaces.

**Figure 8. Uplink OFDM Kernel Interfaces**



The input interface has the following features:

- Avalon-ST data sink
- Ready signal latency of one cycle
- Does not apply back pressure since data is continuous from RF card

The output interface has the following features:

- Avalon-ST data source
- Ready signal latency of one cycle
- Does not accept back pressure from downstream sink
- Dynamically changeable cyclic prefix

Figure 9 shows the uplink kernel input interface timing diagram.

**Figure 9. Uplink Kernel Input Interface Timing Diagram**

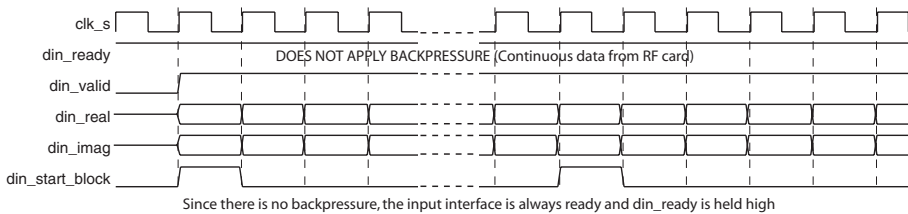
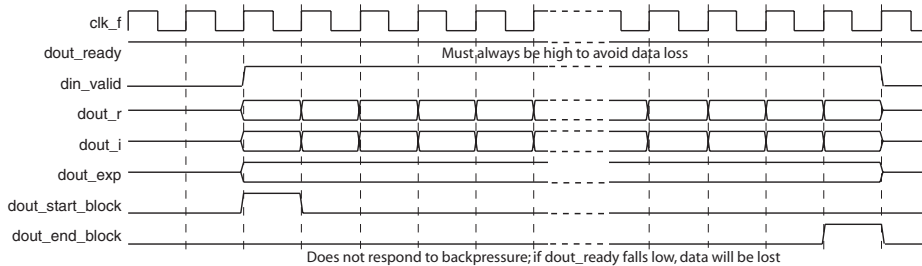


Figure 10 shows the uplink kernel output interface timing diagram.

**Figure 10. Uplink Kernel Output Interface Timing Diagram**



The FFT size can be configured using the LOG2\_N\_FFT parameter which should be  $\log_2$  of the desired FFT size. The MADDR\_WIDTH parameter should be the same as LOG2\_N\_FFT. The real and imaginary data buses are signed fixed point format and the width is configurable using the WIDTH generic. The exponent of the output data is fixed at 6 bits.

The guard interval is configured in the same way as the downlink kernel.

### FFT MegaCore Function

The FFT MegaCore function is capable of performing both the forward and inverse transform. The hardware architecture is chosen to minimize the resource usage and has the following parameters:

- Burst mode
- Single output engine
- Single instance of engine
- 16-bit internal and data input/output precision widths

In addition, this reference design implements two clock domains so that it is possible to exploit time sharing and minimize resource utilization in the FFT MegaCore function by running Clock 1 faster than Clock 2.

The FFT MegaCore function generates block floating point output data and the output dynamic range is maximized for the given input and output data widths.



For more information on block floating point data, refer to AN83: *Binary Number Systems*.

### Clock Requirements

The clocking requirements are as follows:

- The two clock domains must be synchronous
- The minimum Clock 2 frequency is the data sampling frequency given in [Table 2](#). This would lead to a constant output from the FFT MegaCore function

<b>Table 2. Minimum Clock 2 Rate</b>		
<b>FFT Points</b>	<b>Bandwidth (MHZ)</b>	<b>Clock 2 (MHz)</b>
128	1.25	1.429
256	2.5	2.857
512	5	5.714
1,024	10	11.429
2,048	20	22.857

- The Clock 2 frequency may equal or exceed the Clock 1 frequency
- The Clock 1 requirements are dictated by the FFT MegaCore function and are summarized in [Table 3](#)

<b>Table 3. Clock 1 Requirements</b>			
<b>FFT Points</b>	<b>Required Data Rate (MHZ)</b>	<b>FFT Throughput (Cycles/N Block)</b>	<b>Clock 1 Minimum Speed (MHz)</b>
128	1.429	858	9.579
256	2.857	1,626	18.146
512	5.714	3,693	41.214
1,024	11.429	7,277	81.220
2,048	22.857	16,512	184.285

- [Table 3](#) ignores the cyclic prefix effect, which reduces the Clock 1 speed requirement slightly
- The Clock 1 minimum speed = throughput/ $N$  × data rate

## Getting Started

This section describes the system requirements, installation and other information about using the reference design.

### System Requirements

The OFDM kernel requires the following hardware and software:

- A PC running the Windows 2000/XP operating system
- Quartus II version 6.1
- ModelSim SE 5.7d
- Altera FFT MegaCore function



You can download the FFT MegaCore function from [www.altera.com](http://www.altera.com).

### Install the OFDM Kernel

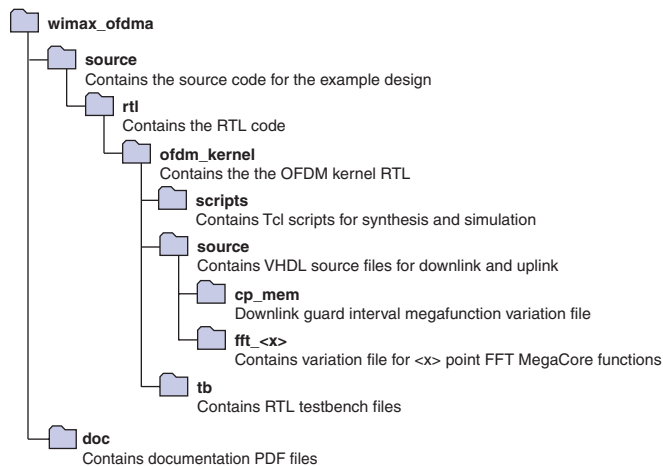
The OFDM kernel is part of the Scalable OFDMA Engine package. To install this package, run the executable file to launch Installshield, and follow the installation instructions.



The reference design is installed by default in the directory `c:\altera\reference_designs` but you can change the default directory during the installation.

Figure 11 illustrates the directory structure of the files associated with the OFDM Kernel.

**Figure 11. Directory Structure**





Additional **sim** and **build** directories are created in the directory structure when you simulate and synthesize the design.

1. To perform RTL Simulation, open the **dl\_ofdm\_kernel\_msim.tcl** script (for downlink) or the **ul\_ofdm\_kernel\_msim.tcl** script (for uplink) in ModelSim. Modify the installation path and desired FFT size. Then execute this macro from the ModelSim tools menu.
2. To perform synthesis, open the **dl\_ofdma\_kernel\_quartus.tcl** script (for downlink) or the **ul\_ofdm\_kernel\_quartus.tcl** script (for uplink) in the Quartus II software. Modify the installation path and desired FFT size. Source this script to create a Quartus project for synthesis.

## Performance

The tables in this section show the resource usage and maximum frequency of operation for Cyclone II, Stratix II, and Stratix III devices using the Quartus® II software with no special optimizations.

Table 4 shows the downlink performance for Cyclone II devices.

FFT Size	Combinational ALUTs	Logic Registers	Memory (Bits)	Memory M4K	18 × 18 Multipliers	f <sub>MAX</sub> (MHz)
128	1,446	1,602	12,800	4	3	190
512	1,501	1,642	51,200	13	3	190
1,024	1,450	1,583	102,400	25	3	196
2,048	1,581	1,682	204,800	50	3	190

Table 5 shows the downlink performance for Stratix II devices.

FFT Size	Combinational ALUTs	Logic Registers	Memory (Bits)	Memory Blocks			18 × 18 Multipliers	f <sub>MAX</sub> (MHz)
				M512	M4K	M-RAM		
128	1,252	1,508	12,800	0	4	0	3	286
512	1,297	1,549	51,200	0	13	0	3	296
1,024	1,255	1,487	102,400	0	25	0	3	269
2,048	1,364	1,587	204,800	0	50	0	3	284

Table 6 shows the downlink performance for Stratix III devices.

FFT Size	Combinational ALUTs	Logic Registers	Memory (Bits)	Memory Blocks			18 × 18 Multipliers	f <sub>MAX</sub> (MHz)
				M-ALUTs	M9K	M144K		
128	1,242	1,508	12,800	0	3	0	3	292
512	1,290	1,549	51,200	0	7	0	3	293
1,024	1,239	1,490	102,400	0	13	0	3	312
2,048	1,354	1,588	204,800	0	25	0	3	312

Table 7 shows the uplink performance for Cyclone II devices.

FFT Size	Combinational ALUTs	Logic Registers	Memory (Bits)	Memory M4K	18 × 18 Multipliers	f <sub>MAX</sub> (MHz)
128	1,414	1,570	8,704	3	3	195
512	1,478	1,614	34,816	9	3	194
1,024	1,428	1,557	69,632	17	3	198
2,048	1,564	1,658	139,264	34	3	195

Table 8 shows the uplink performance for Stratix II devices.

FFT Size	Combinational ALUTs	Logic Registers	Memory (Bits)	Memory Blocks			18 × 18 Multipliers	f <sub>MAX</sub> (MHz)
				M512	M4K	M-RAM		
128	1,221	1,474	8,704	0	3	0	3	282
512	1,271	1,519	34,816	0	9	0	3	286
1,024	1,227	1,464	69,632	0	17	0	3	282
2,048	1,344	1,562	139,264	0	34	0	3	280

Table 9 shows the uplink performance for Stratix III devices.

FFT Size	Combinational ALUTs	Logic Registers	Memory (Bits)	Memory Blocks			18 × 18 Multipliers	f <sub>MAX</sub> (MHz)
				M-ALUTs	M9K	M144K		
128	1,211	1,475	8,704	0	3	0	3	304
512	1,260	1,520	34,816	0	5	0	3	312
1,024	1,217	1,465	69,632	0	9	0	3	312
2,048	1,332	1,563	139,264	0	17	0	3	312

## Conclusion

This application note has outlined the advantages of using Altera FPGAs for implementing OFDM systems such as an IEEE 802.16e deployment.

A flexible, high-throughput DSP platform needs an FPGA-based implementation platform. In addition, this reference design demonstrates the implementation of a key function that may be exploited to facilitate rapid system deployment.

## Revision History

Table 10 shows the revision history for the AN-452: An OFDM FFT Kernel for WiMAX application note.

Version	Date	Errata Summary
1.0	February 2007	First release of this application note.



101 Innovation Drive  
San Jose, CA 95134  
(408) 544-7000  
www.altera.com  
Applications Hotline:  
(800) 800-EPLD  
Literature Services:  
literature@altera.com

Copyright © 2007 Altera Corporation. All rights reserved. Altera, The Programmable Solutions Company, the stylized Altera logo, specific device designations, and all other words and logos that are identified as trademarks and/or service marks are, unless noted otherwise, the trademarks and service marks of Altera Corporation in the U.S. and other countries. All other product or service names are the property of their respective holders. Altera products are protected under numerous U.S. and foreign patents and pending applications, maskwork rights, and copyrights. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera Corporation. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

