# Constellation Mapper and Demapper for WiMAX

May 2007, version 1.1 Application Note 439

## Introduction

Altera provides building blocks that can be used to accelerate the development of an *IEEE 802.16e-2005* (WiMAX) compliant basestation. This application note describes a reference design that demonstrates the suitability of the Altera® tools and devices for implementing the constellation mapping and demapping functions, which can also be easily adapted for compatibility with other wireless standards.

WiMAX is an emerging broadband wireless technology that promises high-speed data services. The *IEEE 802.16e-2005* standard enables mobility. There is significant market potential for this technology and it is currently being deployed by equipment manufacturers. Altera devices are the ideal platform for high throughput DSP designs such as those found on a WiMAX basestation channel card. The devices' dedicated multiplier blocks and inherent parallel structure gives a significant cost and performance advantage over general purpose processors for this type of design.

For more information on *IEEE 802.16e-2005*, refer to the *IEEE Standard for Local and Metropolitan Area Networks, Part 16: Air Interface for Fixed Broadband Wireless Access Systems, IEEE P802.16e-2005, February 2006.*

The reference design has the following features:

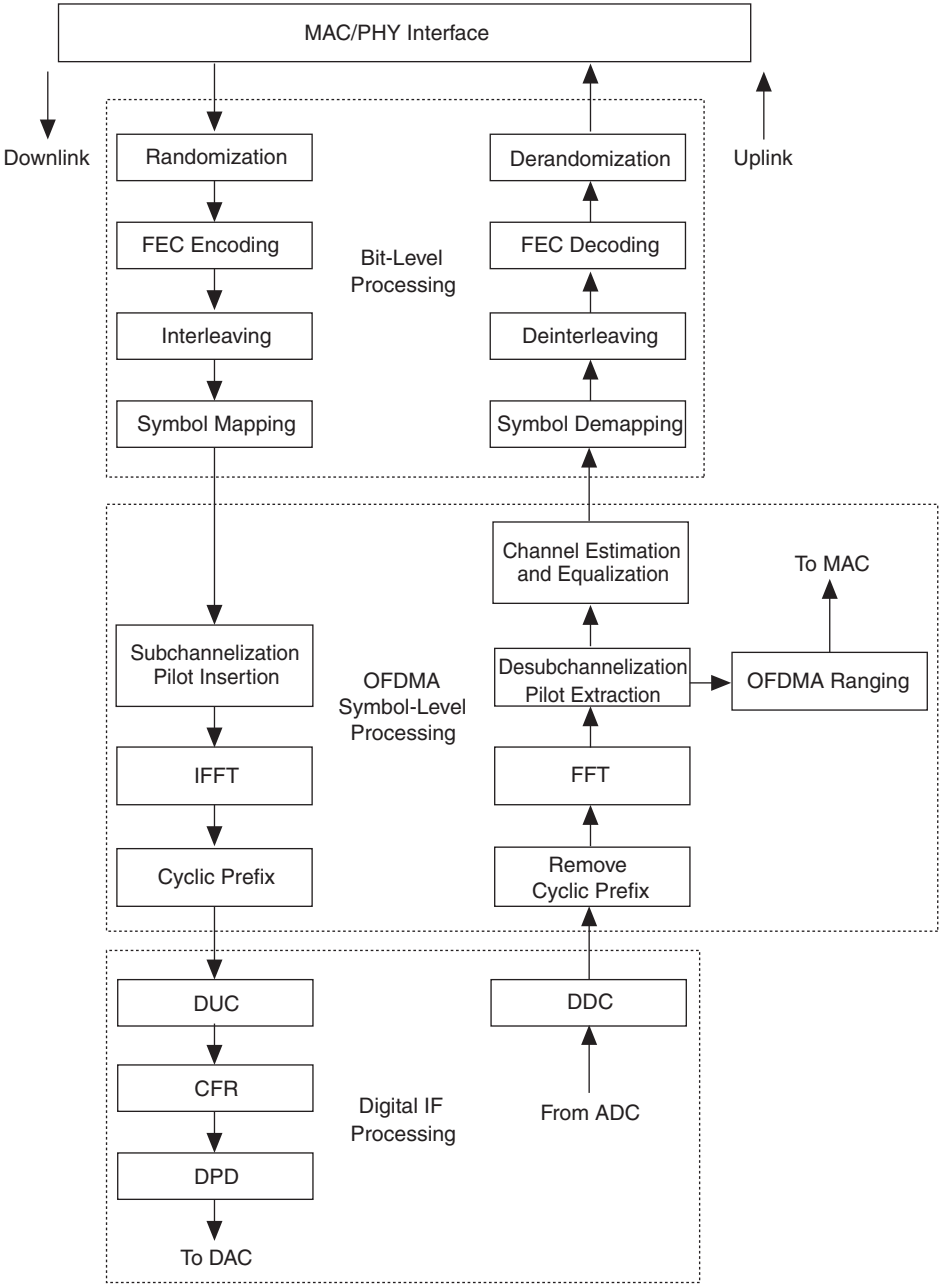
- DSP Builder-based design
- Parameterizable design optimized for efficient use of Cyclone® III and Stratix® III FPGA resources
- Soft-decision demapping for optimal BER performance when used with the Altera Viterbi Compiler

You can use the reference design can be used as a starting point to accelerate designs based on WiMAX or 3GPP LTE (third generation partner project, long-term evolution) protocol

## WiMAX Physical Layer

Figure 1 on page 2 gives an overview of Altera's reference design blocks for implementing the *IEEE 802.16e-2005* scalable orthogonal frequency-division multiple-access physical layer in WiMAX basestations. This application note illustrates the functionality and implementation of the symbol mapping and symbol demapping blocks. These blocks represent the interfaces between bit level and symbol level processing.

*Figure 1. WiMAX Physical Layer*

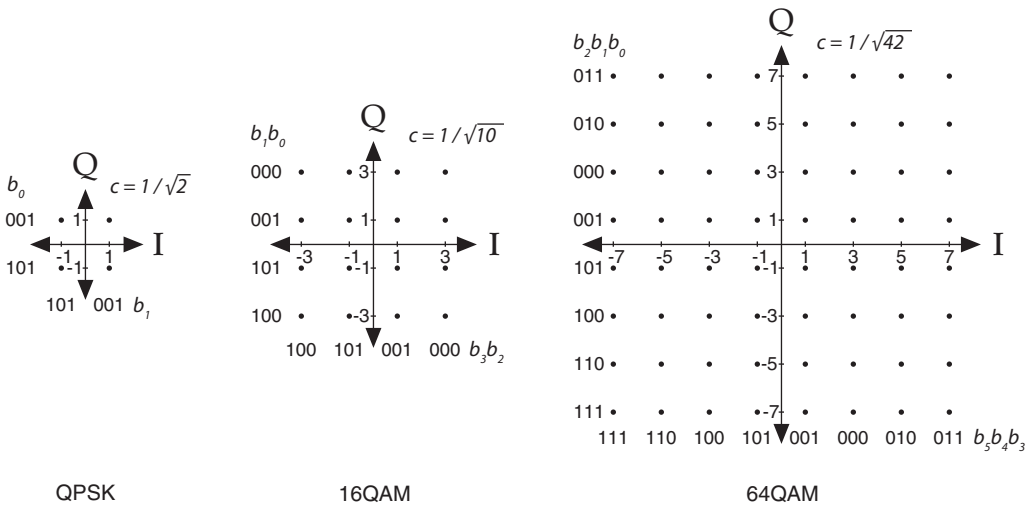


## WiMAX Modulation Specifications

This reference design provides a constellation mapper and demapper to use with the *IEEE 802.16e-2005* specification. The modes supported are gray-coded QPSK, 16QAM, and 64QAM modulation schemes. You can configure the hardware modules at run time, so you can use them in a multi-user system where each user or subchannel may be operating with a different modulation scheme.

The symbols are normalized so that each constellation has an equal average power, as described by the specification. The mapping process also multiplies each constellation point by a scaling factor. Figure 2 shows the gray mapping for the three constellation modes and the associated scaling factor. The demapper performs the complementary operation of the mapper by extracting the bitstream from the received complex stream.

***Figure 2. Gray-Coded Modulation Schemes***




The constellation mapper takes groups of bits and maps them to specific constellation points. A specific magnitude and phase represents a certain combination of bits. The operating mode for each user or subchannel is configured at run time and is determined by assessing the channel quality. The MAC layer adjusts the modulation scheme of each user, to maximize throughput while meeting an acceptable error rate.

At the receiver, the phase and magnitude of each carrier is extracted, and a decision must be made about what combination of bits the transmitter sent. Because each of the carriers has been subjected to distortion by the

wireless channel, there is an error in the position of each constellation point. To decide on the combination of bits the design divides the complex plane into equally sized regions that correspond to each constellation point and outputs the bit combination of the region that the received signal appears in—known as hard decision decoding.

You can achieve significantly better coding gain by making soft decisions. For soft decision decoding each bit is assigned a confidence that it will be a 0 or a 1. Although this decoding leads to greater hardware complexity, it is essential to achieve satisfactory bit error performance, especially for higher order modulation schemes such as 64QAM.

For QPSK, the soft information associated with each bit is calculated by evaluating the values of the in-phase and quadrature components. The calculation of soft information is more difficult for higher order modulation schemes, because the real and imaginary axes carry more than one bit of information.

This reference design implements a highly efficient, simplified log likelihood ratio method for calculating the soft information.

For more information on the theory and performance of the algorithm, refer to *Simplified Soft-Output Demapper for Binary Interleaved COFDM with Application to HIPERLAN/2, Filippo Tosator & Paola Bisaglia, HPL-2001-246, October 2001.*

### *Soft Information Calculations*

The following equations summarize the calculation of the soft information for the bits on the *I* plane. You can calculate the corresponding soft information associated with the imaginary plane using the same equations and substituting the imaginary part of the received symbol into the equations.

**QPSK**

$$D_{I,1} \approx -y_I[i]$$

**16QAM**

$$D_{I,1} \approx -y_I[i]$$

$$D_{I,2} \approx |-y_I[i]| - 2$$

**64QAM**

$$D_{I,1} \approx -y_I[i]$$

$$D_{I,2} \approx |-y_I[i]| - 2$$

$$D_{I,3} \approx |-y_I[i]| - 4| - 2$$

### Viterbi MegaCore Function Design Considerations

The soft decision metrics are ultimately passed on to the Viterbi decoder MegaCore® function. The precision of the soft decision metrics must be reduced, to reduce the complexity of the Viterbi decoder. Six bits of precision exhibits the best balance between performance and complexity.

To make forward error correction (FEC) possible, the WiMAX specification requires the transmitter to apply redundancy to the data, by encoding each block of data using a tail-biting convolutional encoder. This special type of convolutional encoder has the memory initialized with the last data bits of the FEC block that are encoded, which the Viterbi decoder exploits.

For more information on the Altera Viterbi Compiler, refer to the *Viterbi Compiler User Guide* and *Viterbi Tail-Biting Double-Pass Decoding (P = TB)* on the Altera website at www.altera.com/support/examples/dsp-builder/exm-viterbi-tail.html.

The Altera Viterbi Compiler also supports soft decision decoding. Table 1 gives an example of the format of a three-bit soft decision input. This format is used in this reference design, but with six bits of precision.

*Table 1. Soft Symbol Input Representation*

| Soft Symbol | Meaning |
|---|---|
| 011 | Strongest “0” |
| 010 | Strong “0” |
| 001 | Weak “0” |
| 000 | Weakest “0” |
| 111 | Weakest “1” |
| 110 | Weak “1” |
| 101 | Strong “1” |
| 100 | Strongest “1” |

## Implementation with DSP Builder

Digital signal processing (DSP) system design in Altera programmable logic devices (PLDs) requires both high-level algorithm and HDL development tools. The Altera DSP Builder integrates these tools by

combining the algorithm development, simulation, and verification capabilities of MATLAB and Simulink system-level design tools (from The MathWorks) with VHDL synthesis, simulation, and Altera development tools.

The DSP Builder shortens DSP design cycles by helping you create the hardware representation of a DSP design in an algorithm-friendly development environment. The existing MATLAB functions and Simulink blocks can be combined with Altera DSP Builder blocks and Altera intellectual property (IP) MegaCore functions to link system-level design and implementation with DSP algorithm development. DSP Builder allows system, algorithm, and hardware designers to share a common development platform.

Designers can use the blocks in DSP Builder to create a hardware implementation of a system modeled in Simulink in sampled time. DSP Builder contains bit- and cycle-accurate Simulink blocks, which cover basic operations such as arithmetic or storage functions. Complex functions can be integrated by using MegaCore functions in DSP Builder models.

You can verify the fixed point performance of the algorithms using DSP Builder and compare the results to a bit-accurate MATLAB simulation, which is useful when prototyping the physical layer using the MATLAB environment. You can integrate these features into a powerful testbench, to give an ideal development environment for wireless system engineers.

# Functional Description

This section describes the functionality of the constellation mapper and the constellation demapper.

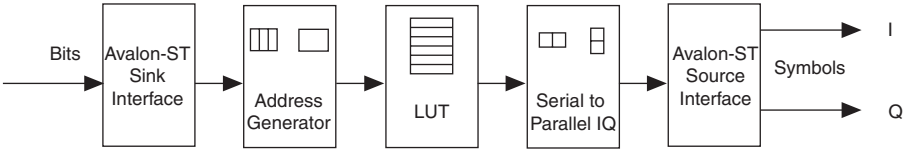
## Constellation Mapper

The constellation mapper takes a bitstream as an input and maps it onto appropriate constellation symbols, according to the modulation method that you specify.

### *Architecture*

Figure 3 shows the architecture of constellation mapper.

*Figure 3. Constellation Mapper*




The interleaver presents the input data as a serial bit stream. To generate the symbols, you must group together a number of bits and calculate which constellation point it corresponds to. The address generator groups $M/2$ bits together to formulate an address where $M$ is the constellation order. You can use this address to index a lookup table. The lookup table stores the constellation point corresponding to the group of bits.

To reduce the size of the lookup table, the design uses $M/2$ bits. The same lookup table is shared for $I$ and $Q$. You can generate an address location in the lookup table for each constellation point, but it is not memory efficient. The output of the lookup table is a serial stream of $I$ and $Q$ samples ($I$, $Q$, $I$, $Q$, and so on). This output is converted into two parallel output $I$ and $Q$ interfaces by the serial to parallel $IQ$ converter.

Table 2 shows how the modulation input is encoded.

*Table 2. Encoding for Modulation Input*

| Modulation Input | Associated Modulation Scheme |
|---|---|
| 00 | Invalid Mode |
| 01 | QPSK |
| 10 | 16QAM |
| 11 | 64QAM |

### *Parameters*

You can modify the amount of precision (bitwidth) and the slope (maximum constellation point) of the fixed point representation. The maximum constellation point must be greater than the maximum value of the three constellation schemes. It must also be large enough to accommodate additional headroom for noise and boosted BPSK pilots.

### *Interface Specifications*

The Avalon® Streaming (Avalon-ST) input and output interfaces make it easy to integrate this module with other IP from Altera and with any other proprietary interface.

For more information on Avalon-ST interfaces, refer to the *Avalon Streaming Interface Specification*.

The Avalon-ST interface has the following parameters:

- Ready latency: 1
- Sink symbols per beat: 1
- Source symbols per beat: 2 (*I* and *Q* are conveyed synchronously)

The block supports packetized transfers using the start and end of packet signals. The start of a packet signal resets the state of the block so that the module can recover from a bad packet.

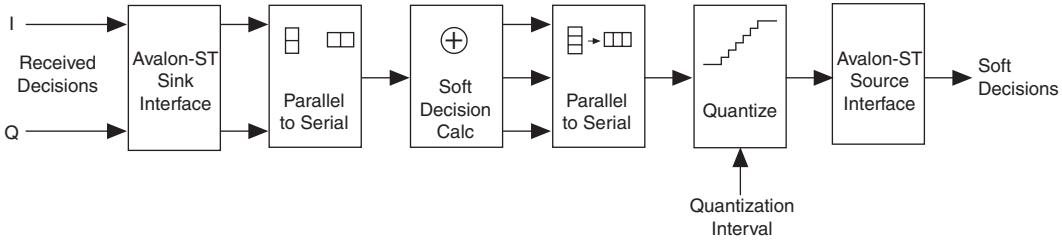
## Constellation Demapper

The constellation demapper takes packets of received constellation points as an input, and outputs the corresponding soft decision bitstream.

### *Architecture*

Figure 4 shows the architecture of the constellation demapper.

***Figure 4. Constellation Demapper***




To achieve maximum hardware efficiency, this constellation demapper is fully time division multiplexed. Backpressure is applied to the upstream module so that data is only provided at a maximum rate of one complex sample for every six clock cycles. If data is acquired every six clock cycles, the output bus fully conveys the soft decisions when operating in 64QAM modulation mode. In the 16QAM and QPSK modes, the output bus is not fully used.

The constellation demapper multiplexes the IQ data onto a single bus, and passes the serial data stream on to the soft decision calculator where the appropriate metrics for the modulation scheme are calculated (see "Soft Information Calculations" on page 4). These metrics determine the confidence in the polarity of each constellation bit. These metrics are time multiplexed onto a single bus, before they are passed on to a quantization module that reduces the bitwidth of the signal according to a quantization interval.

### *Parameters*

You can modify the amount of precision (bitwidth) and the slope (maximum constellation point) of the fixed point representation. The maximum constellation point must be greater than the maximum value of the three constellation schemes. It must also be large enough to accommodate additional headroom for noise.

### *Interface Specifications*

The Avalon Streaming (Avalon-ST) input and output interfaces make it easy to integrate this module with other IP from Altera and with any other proprietary interface.

The Avalon-ST interface has the following parameters:

- Ready latency: 1
- Sink symbols per beat: 1
- Source symbols per beat: 2 (*I* and *Q* are conveyed synchronously)

The block supports packetized transfers using the start and end of packet signals.

## Reference Design Throughput

For the 64QAM mode, one complex symbol of information is characterized by a group of six bits. The maximum symbol throughput is equal to one sixth of the maximum bitrate or soft decision throughput.

Table 3 shows the relationship between the FFT size and required baseband symbol throughput. Assuming a clock frequency of 182.784 MHz, you can show how many independent bitstreams can be mapped or demapped using a single constellation mapper/demapper module, which is useful for multiple-antenna or multiple-carrier basestations.

*Table 3. Constellation Mapper Processing Capability*

| FFT Size | Baseband Symbol Rate (MSPS) | Clock Frequency (MHz) | Oversampling Factor | Number of Streams |
|---|---|---|---|---|
| 128 | 2.428 | 182.784 | 128 | 21 |
| 512 | 5.712 | 182.784 | 32 | 5 |
| 1024 | 11.424 | 182.784 | 16 | 2 |
| 2048 | 22.848 | 182.784 | 8 | 1 |

# Getting Started

This section contains the following information:

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

## System Requirements

The scalable OFDM engine requires the following hardware and software:

- A PC running the Windows XP operating system
- Quartus II version 7.1
- DSP Builder version 7.1
- MATLAB version R2006B
- Simulink version R2006B

To take advantage of some of the testbench features, Altera recommends the MATLAB signal processing blockset.

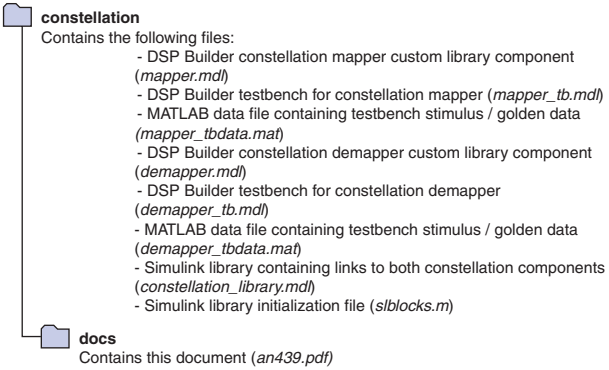
## Installing the Reference Design

To install the reference design, run the **an439-v1.0.exe** file to launch InstallShield and follow the installation instructions.

The default installation directory is **c:\altera\reference_designs\constellation**. If you have other WiMAX reference designs from Altera, install this reference design to the path **<wimax_ofdma\source\rtl>** and into a directory called **\constellation**.

Figure 5 shows the directory structure after installation.

***Figure 5. Directory Structure***




## Running the Reference Design

To run the reference design follow these steps:

1. Open MATLAB.
2. Change the MATLAB directory to the installation directory.
3. Open Simulink, and go to the Simulink library browser.
4. Verify that the **Altera WiMAX Constellation Map/Demap** reference design library exists.
5. Open the testbench associated with the desired reference design (**mapper_tb.mdl** or **demapper_tb.mdl**). The testbench automatically loads the necessary testbench stimulus from the file.
6. Simulate the design.

## Testbench Features

You can experiment with various input and output throughputs by modifying the output ready signal. Also, you can modify the input throughput, by providing a new piece of data only when the input interface is ready, and by providing another randomly generated binary number that is equal to one, which leads to a more exhaustive testbench.

The constellation mapper reference design is connected to an X/Y plot, so you can visualize the output constellation.

Both the constellation mapper and demapper testbenches verify the correct behavior when packets containing data associated with different modulation schemes are passed through the block.

### Synthesis Results

Table 4 and Table 5 show results using the Quartus II software, version 7.1.

*Table 4. Constellation Mapper Synthesis Results*

| Device | Combinational ALUTs/LUTs | Logic Registers | Memory | | 9×9 multipliers | $f_{MAX}$ (MHz) |
|---|---|---|---|---|---|---|
| | | | M512 | M9K | | |
| Stratix III EP3SE80F780C3 | 40 | 108 | 0 | 1 | 0 | 418 |
| Cyclone III EP3C80F780C6 | 59 | 108 | 0 | 1 | 0 | 400 |

*Table 5. Constellation Demapper Synthesis Results*

| Device | Combinational ALUTs/LUTs | Logic Registers | Memory | | 9×9 multipliers | $f_{MAX}$ (MHz) |
|---|---|---|---|---|---|---|
| | | | M512 | M9K | | |
| Stratix III EP3SE80F780C3 | 338 | 809 | 0 | 1 | 0 | 258 |
| Cyclone III EP3C80F780C6 | 378 | 741 | 0 | 1 | 0 | 224 |

## Conclusion

A number of wireless applications including GSM, wideband code division multiple access (W-CDMA), high-speed downlink packet access (HSDPA), WiMAX, and 3GPP LTE require constellation mapping and demapping. This reference design shows how you can easily and efficiently implement these functions on Altera FPGAs using DSP Builder-based design methodology.

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