Symbol Interleaver/Deinterleaver
MegaCore Function

Target Applications:
Digital Communications

Family:
APEX™ 20K & FLEX® 10K

Ordering Code:
PLSM-INLV

Vendor:
Altera Corporation
101 Innovation Drive
San Jose, CA 95134
http://www.altera.com
Tel. (408) 544-7000

Features
- Offers high-speed data rates: 120 million samples per second (MSPS)
- Supports convolutional interleaving algorithm
- Supports block interleaving algorithm
- Contains a burst error distribution analyzer

General Description
Interleaving is a standard digital signal processing (DSP) function used in many communications systems. Applications that store or transmit data use symbol interleaver/deinterleavers to reduce spurious noise that can corrupt data. The Altera® symbol interleaver/deinterleaver MegaCore® function supports both a continuous mode and a discrete mode, making it compatible with any type of Reed-Solomon function. Also, the symbol interleaver/deinterleaver improves the efficiency of Reed-Solomon encoders/decoders by spreading the errors across several codewords.

The symbol interleaver/deinterleaver MegaCore function has been optimized for Altera APEX™ 20K and FLEX® 10K devices. The MegaWizard® Plug-In reduces design and simulation cycles from weeks to minutes, optimizing memory configurations based on user-defined parameters. For block interleaving, the MegaWizard Plug-In uses single-port RAM; for convolutional interleaving, the interleaver/deinterleaver function utilizes embedded array block (EAB) and embedded system block (ESB) dual-port RAM capability. Flexible programmable logic combined with the symbol interleaver/deinterleaver MegaWizard Plug-In allows you to build real-time systems that analyze and improve custom parameter values.

Functional Description
Interleavers arrange the symbols into a group of transmitted codewords so that adjacent symbols in the data stream are not from the same codeword. Deinterleavers reassemble the codewords when the deinterleaver processes the data stream. This process of interleaving and deinterleaving data reduces the effects of burst errors (multiple errors in a row) as it rearranges codeword symbols and spreads the errors among multiple codewords.

The interleaving/deinterleaving process introduces a constant delay between the interleaver input data and the deinterleaver output data. When a symbol enters a branch of the interleaver or deinterleaver, it exits the branch via the output commutator after a variable delay that depends on the branch index. The symbol interleaver/deinterleaver supports two algorithms: convolutional and block.

Convolutional Interleavers/Deinterleavers
Convolutional interleavers/deinterleavers process data in a continuous stream, making them ideal for high-speed applications that require burst error correction (e.g., digital video broadcasting). Compared to block interleavers/deinterleavers, convolutional interleavers/deinterleavers provide reduced delay and lower memory usage for the same distribution of errors.
Figure 1 shows the structure of the convolutional interleaver/deinterleaver.

**Figure 1. Convolutional Interleaver/Deinterleaver Structure**

Block Interleaver/Deinterleaver

Block interleavers/deinterleavers use single-port RAM memory configured as a matrix of $n$ rows by $m$ columns to interleave data. Symbol transmission consists of an alternating sequence of write and read cycles. During the write cycle, input symbols are written column-by-column; during the read cycle, output symbols are read row-by-row. Each cycle delay is equal to the buffer size (block length $\times$ by span delay).

Figure 2 shows the block structure for a six-symbol codeword during each cycle.

**Figure 2. Block Interleaver/Deinterleaver Structure for a Six-Symbol Codeword**

**MegaWizard Plug-In**

The MegaWizard Plug-In Manager, which you can use within the Quartus™ or MAX+PLUS® II software or as a standalone application, lets you create or modify design files for the interleaver/deinterleaver MegaCore function. When you run the symbol interleaver/deinterleaver, the MegaWizard Plug-In Manager creates AHDL (.tdf), VHDL (.vhd), and Verilog (.v) files that you can instantiate in your system design, as well as MAX+PLUS II vector files (.vec) and symbol files (.sym) that allow you to view the design waveform.
Table 1 shows the MegaWizard options available for block and convolutional functions.

<table>
<thead>
<tr>
<th>Option</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of columns</td>
<td>Block</td>
<td>Specifies the total length of the codeword (data and checksum symbols)</td>
</tr>
<tr>
<td>Number of branches</td>
<td>Convolutional</td>
<td>Specifies the number of branches used by the interleaver</td>
</tr>
<tr>
<td>Direction</td>
<td>Block or convolutional</td>
<td>Specifies whether you wish to create an interleaver or a deinterleaver</td>
</tr>
<tr>
<td>Memory type</td>
<td>Block or convolutional</td>
<td>Indicates whether you wish to use internal or external memory</td>
</tr>
<tr>
<td>Number of rows</td>
<td>Block</td>
<td>Specifies the maximum number of codewords in the block interleaver/deinterleaver memory</td>
</tr>
<tr>
<td>Unit delay element</td>
<td>Convolutional</td>
<td>Specifies the unit delay for each branch of the interleaver/deinterleaver</td>
</tr>
<tr>
<td>Symbol width</td>
<td>Block or convolutional</td>
<td>Specifies the width of the input symbol</td>
</tr>
<tr>
<td>Type</td>
<td>Block or convolutional</td>
<td>Indicates whether you wish to create a block or convolutional</td>
</tr>
</tbody>
</table>

Performance & Implementation

This section provides an overview of the steps that create and implement a custom interleaver/deinterleaver function. The MegaWizard Plug-In Manager lets you create or modify design files that contain custom megafunction variations. You can then simulate your design with the Quartus or MAX+PLUS II Simulator to verify compatibility and instantiate the custom megafunction in your design file.

Design Entry

Start the MegaWizard Plug-In Manager by choosing the MegaWizard Plug-In Manager command (File menu) in the Quartus or MAX+PLUS II software, or type the command megawiz at a command or UNIX prompt. Choose the type of file you wish to create from the Available Megafunctions dialog box on the second page of the MegaWizard Plug-In Manager. Select the type of algorithm (convolutional or block) and the direction (interleaver or deinterleaver). Once you specify the characteristic options of the function, you can use the Manager to view and analyze burst error distribution.

Simulation

You can simulate your design after you have compiled it with the Quartus or MAX+PLUS II software. The vector file created by the MegaWizard Plug-In Manager is loaded automatically when you begin the simulation. Once the simulation is complete, you can choose the OPEN SCF button on the MegaWizard Plug-In Manager screen to view the design waveform. Verify the function compatibility with your design requirements, then insert the function into your system-level design.
Table 2 shows the interleaver/deinterleaver function’s performance as calculated using the MAX+PLUS II version 9.2 software.

<table>
<thead>
<tr>
<th>Function</th>
<th>Parameters</th>
<th>Device</th>
<th>Speed Grade</th>
<th>Logic Elements Used</th>
<th>EABs Used</th>
<th>fMAX (MHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Convolutional interleaver using FLEX® 10KE EABs</td>
<td>Depth = 12, Unit Delay = 17, Symbol width = 8 bits (digital video broadcast settings)</td>
<td>FLEX 10KE</td>
<td>1</td>
<td>392</td>
<td>8</td>
<td>110</td>
</tr>
<tr>
<td>Block interleaver using single-port RAM</td>
<td>Block length = 36, Span delay = 20, Data Width = 8 (UTRA) ITU-R RTT</td>
<td>FLEX 10KE</td>
<td>-1</td>
<td>40</td>
<td>4</td>
<td>120</td>
</tr>
</tbody>
</table>