Reed-Solomon FEC Demonstration

Features

- Implements a Reed-Solomon core with digital video broadcast (DVB) specifications
- Windows 98/NT/2000 application provides a graphical user interface (GUI) for setting options
- Three error channels are available
- An optional interleaver handles burst errors
- Results are displayed as bitmap pictures
- Supports the system-on-a-programmable-chip (SOPC) board, the APEX™ DSP development board, and the Nios™ development board

General Description

The Altera® Reed-Solomon demonstration shows the benefits of using Altera's forward error correction (FEC) solutions. The demonstration consists of a Windows application in which you specify a picture to transmit over a channel using the Reed-Solomon core. You can use the application to alter the type and intensity of errors inserted into the data as it passes through the channel. The Altera board implements the hardware portion of the design, which includes the encoder, decoder, and channel.

☞ You can download the demonstration from the Reed-Solomon MegaCore® function web page on the Altera web site. Point your web browser at http://www.altera.com/IPmegastore and search for Reed-Solomon.

After you select the picture and error type, the bytes representing the colors of the pixels within the picture are transmitted to the board via an RS-232 port. The data is loaded into a Nios™ embedded processor, which stores it in a FIFO buffer. The data is read out of the FIFO buffer and processed in two ways:

- One data path goes through the selected channel only and is displayed in the GUI with the errors visible.

and

- The other data path goes through the Reed-Solomon encoder, through the channel, through the Reed-Solomon decoder, and then is displayed in the GUI.

☞ The demonstration provides an option to include an interleaver/deinterleaver in the data path. If you turn on the **Add Interleaver/Deinterleaver** option (it is only available with the burst channel), the demonstration places an interleaver just after the Reed-Solomon encoder and a deinterleaver just before the decoder.

The MegaCore functions in this demonstration are configured with the following parameters:

- Half, standard, streaming Reed-Solomon decoder (204, 188)
  – 8-bit symbol width
  – 204 symbols per codeword (N = 204)
  – 16 check symbols (R = 16)
- Block, discrete interleaver (232 × 5)

The demonstration runs at 33 MHz. Figure 1 shows an overview of the demonstration setup using the SOPC board.

A-WP-RSFECDEMO-2.1

Figure 1. Reed-Solomon Demonstration Setup with the SOPC Board

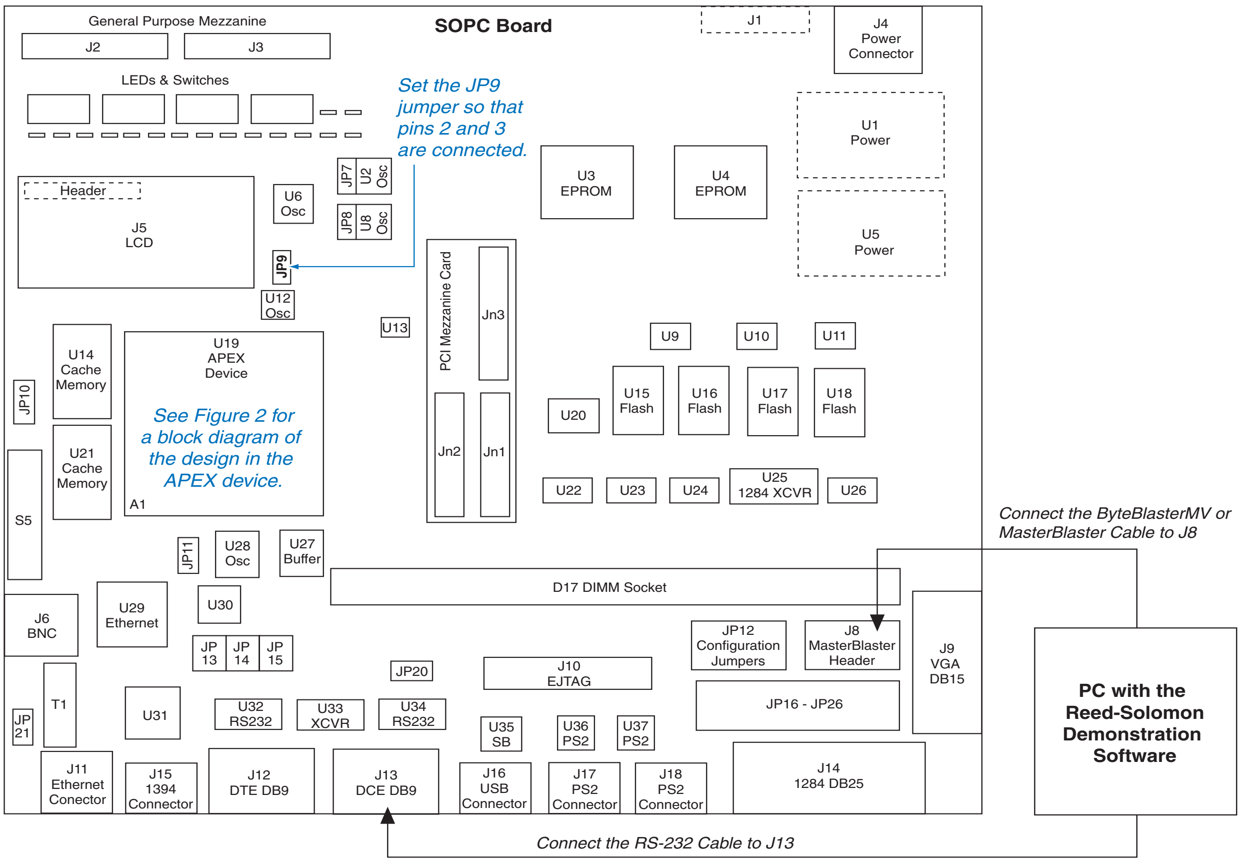


Figure 2 shows an overview of the demonstration setup using the APEX DSP development board.

Figure 2. Reed-Solomon Demonstration Setup with the APEX DSP Development Board

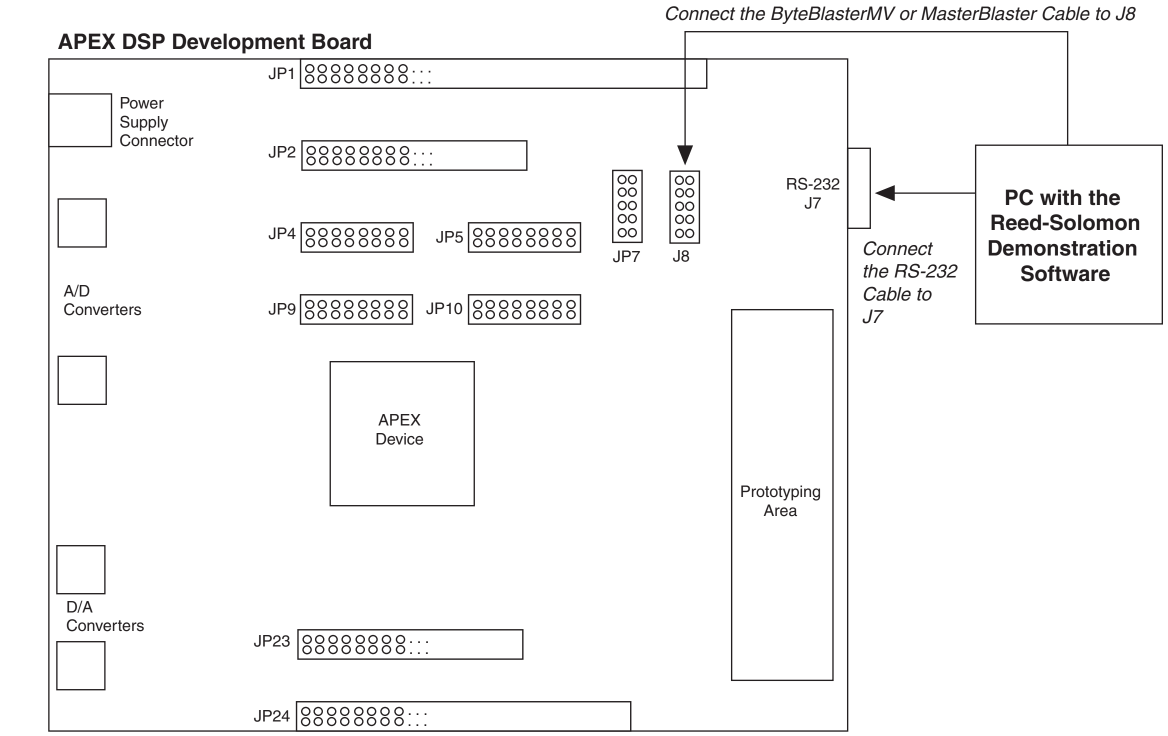


Figure 3 shows an overview of the demonstration setup using the Nios development board.

Figure 3. Reed-Solomon Demonstration Setup with the Nios Development Board

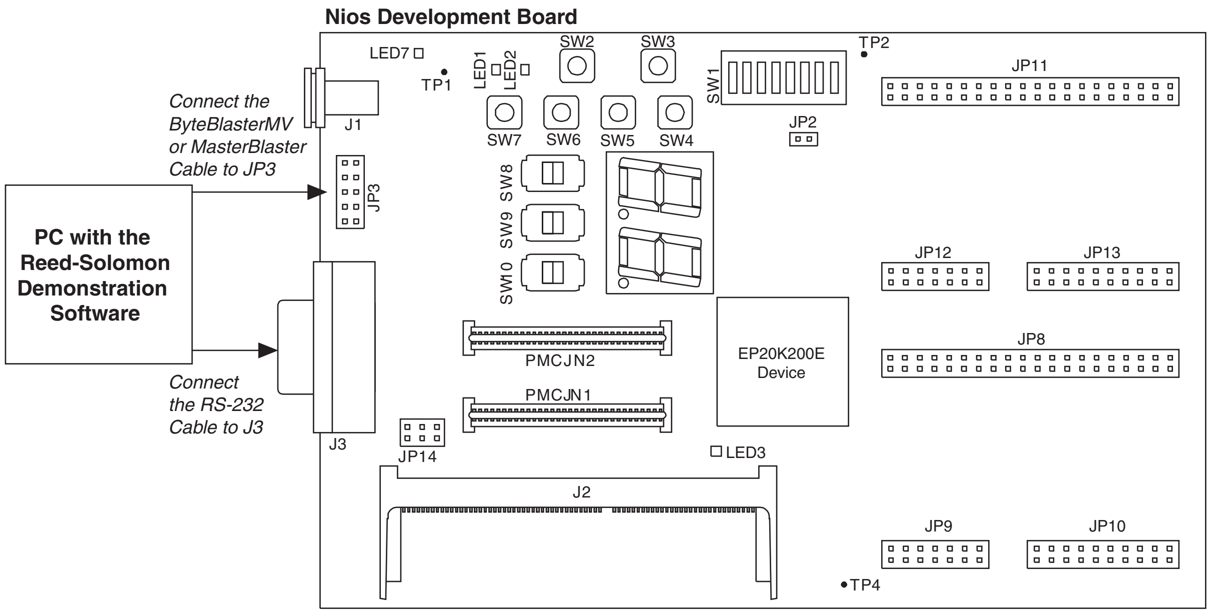
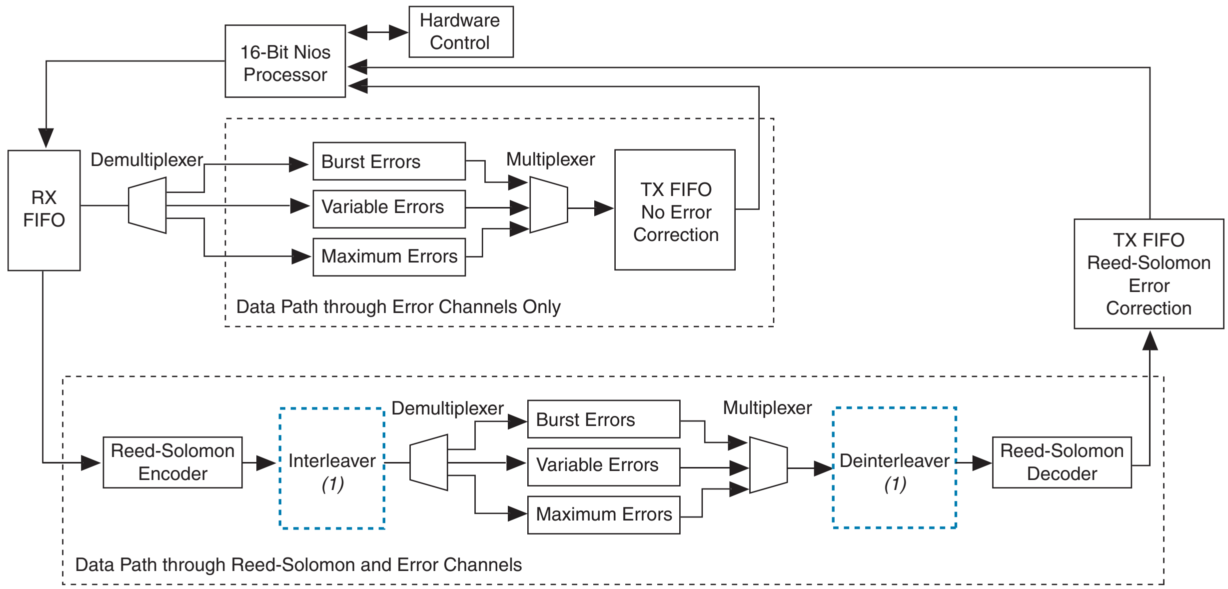


Figure 4 shows a block diagram of the design implemented in the APEX device on the board.

Figure 4. Block Diagram of Design Implemented in APEX Device on the Board



Note:
(1) The interleaver and deinterleaver are only added if you turn on the **Add Interleaver/Deinterleaver** option in the GUI.

Definitions

The following definitions apply to this demonstration only:

- *Symbol*—An 8-bit piece of data
- *Codeword*—204 bytes consisting of 188 data symbols and 16 Reed-Solomon check symbols
- *Check symbol*—A symbol that the Reed-Solomon encoder adds to the data so the decoder can correct any errors that occur during the data transmission
- *Data packet*—A 940-byte piece of data
- *BER*—Bit error rate

Reed-Solomon Demonstration Walkthrough

To perform the walkthrough, you will need the following items:

- An SOPC board, APEX DSP development board, or Nios development board
- A PC running the Windows 98/NT/2000 operating system with the MAX+PLUS® II, Quartus™, or Quartus II software installed
- A ByteBlasterMV™ or a MasterBlaster™ download cable
- An RS-232 cable

☞ These instructions assume that you have already installed the demonstration software. Refer to the **readme** file provided with the software for installation instructions.

Connect the Cables to the Board

Follow the instructions in the following sections to connect the MasterBlaster or ByteBlasterMV, serial, and power cables to the board you want to use.

SOPC Board

1. Connect power to the board.

2. Connect the ByteBlasterMV or MasterBlaster cable to the JTAG header on the board (J8) and to your PC.

3. Set the JP12 jumper to program the appropriate device (see page 41 of the *System-on-a-Programmable-Chip Development Board User Guide* for instructions on setting the jumper). You can configure the APEX device directly using an SRAM Object File (**.sof**) or you can program the EPC2 devices with the three Programmer Object Files (**.pof**). If you use the POFs, you must turn power to the board off and then on again for the configuration devices to configure the APEX device.

4. Set the JP9 jumper so that pins 2 and 3 are connected.

5. Using the MAX+PLUS II, Quartus, or Quartus II software, configure the APEX device with the SOF or program the configuration devices with the POFs provided with the demonstration. See *AN 116 (Configuring APEX 20K, FLEX 10K & FLEX 6000 Devices)* for more information on configuring Altera devices.

   ☞ When the device is configured successfully, an LED flashes once per second.

6. Attach the serial cable to one of the serial ports on your PC and to the RS-232 male connector at J13 on the SOPC board.

   ☞ You will need a female-to-female RS-232 cable to connect the PC and the SOPC board.

APEX DSP Development Board

1. Connect power to the board.

2. You can configure the APEX device directly using an SOF or you can program the EPC16 device with a POF. If you use the POF, you must turn power to the board off and then on again for the EPC16 device to configure the APEX device.

   To configure the APEX device directly, connect the ByteBlasterMV or MasterBlaster cable to the APEX device JTAG header on the board (J8) and to your PC.

   or

   To program the EPC16 device, connect the ByteBlasterMV or MasterBlaster cable to the EPC16 JTAG header on the board (JP7) and to your PC.

3. Arrange the jumpers at JP25 so that pins 1 and 2 are connected and pins 3 and 4 are connected.

4. Using the MAX+PLUS II, Quartus, or Quartus II software, configure the APEX device with the SOF or program the EPC16 device with the POF provided with the demonstration. See *AN 116 (Configuring APEX 20K, FLEX 10K & FLEX 6000 Devices)* for more information on configuring Altera devices.

   ☞ When the device is configured successfully, an LED flashes once per second.

5. Attach the serial cable to one of the serial ports on your PC and to the RS-232 male connector at J7 on the board.

Nios Development Board

1. Connect power to the board.

2. Connect the ByteBlasterMV or MasterBlaster cable to the JTAG header on the board (JP3) and to your PC.

3. Move the SW8 two-position switch towards the RS-232 connector (connect position) and move the SW9 and SW10 two-position switches away from the RS-232 connector (bypass position).

4. Using the MAX+PLUS II, Quartus, or Quartus II software, configure the APEX device with the SOF provided with the demonstration. See *AN 116 (Configuring APEX 20K, FLEX 10K & FLEX 6000 Devices)* for more information on configuring Altera devices.

   ☞ When the device is configured successfully, an LED flashes once per second.

5. Attach the serial cable to one of the serial ports on your PC and to the RS-232 male connector at J3 on the board.

Run the Demonstration

This section describes how to run the demonstration. The application does not display properly on systems that use large fonts. To change your font settings, perform the following steps. If you are already using small fonts, skip to step 7.

1. Choose **Settings > Control Panel** (Windows Start menu).

2. Double-click the **Display** icon.

3. Click the **Settings** tab.

4. Choose **Small Fonts** under **Font Size**.

5. Click **OK**.

6. Reboot your system for the changes to take effect.

7. Execute the **rs_demo_v2.exe** file to start the demonstration software.

8. Click the **PC Options** button, indicate which COM port you are using, and click **OK**.

9. Select the picture to transmit in the **Picture Selection** box.

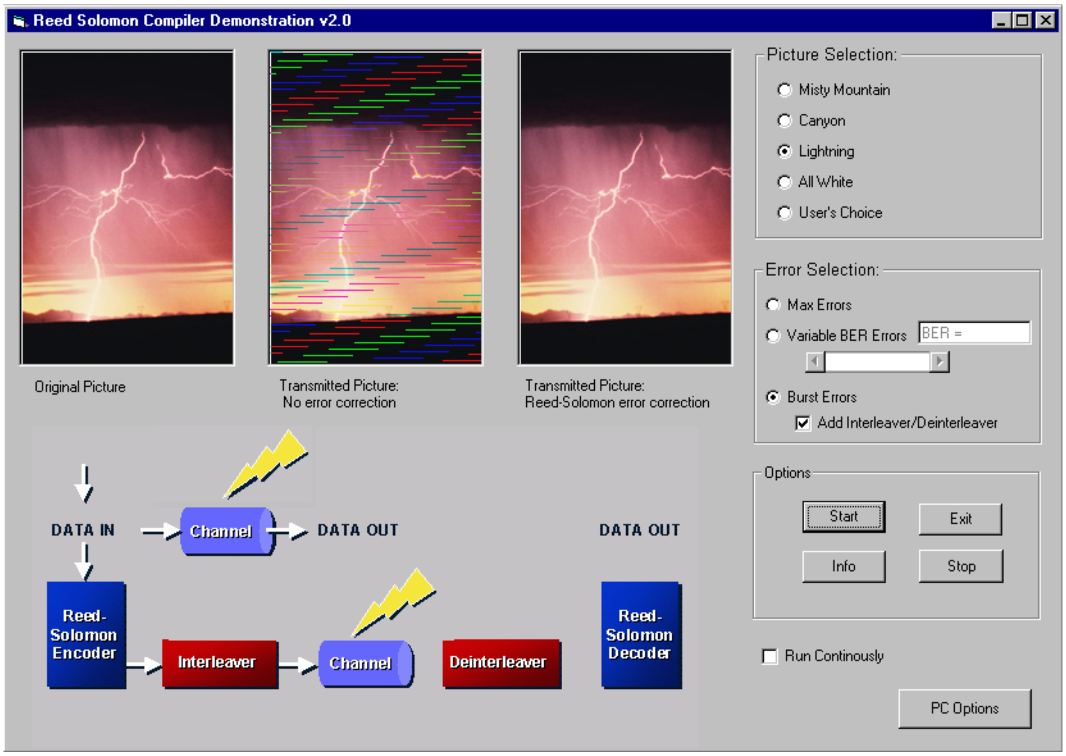
   ☞ If you prefer, you can use a picture of your choice instead of the ones provided with the demonstration. See “Adding a Custom Picture” on page 8.

10. Select the error type in the **Error Selection** box (the demonstration provides several types of errors that can be injected into the channel; see “Error Channels” on page 7 for more details), and click **Start**.

    ☞ The flashing LED flashes twice as fast while the data is being processed.

The data paths are shown in the block diagram in the demonstration software GUI. The transmitted pictures are displayed as the data bits are processed. See Figure 5.

Figure 5. Reed-Solomon Demonstration User Interface



Transmission Details

Each bitmap image transmitted in this demonstration has three data bytes representing each pixel. Data is transmitted in blocks of 5 Reed-Solomon codewords (188 bytes each for a total data packet of 940 bytes). After a block is transmitted to the board and processed, it is sent back to the demonstration software and displayed in the appropriate window before another data packet is sent. A 16-bit Nios embedded processor receives the data using a UART peripheral from the Nios peripheral library. The Nios processor writes the data into a memory-mapped FIFO buffer, where it is read out and passed to the appropriate data path. Then, the data is written to a pair of memory-mapped FIFO buffers from which the Nios processor reads the data and sends it back to the PC.

Error Channels

You can choose from three provided error channels. The **Max Errors** channel deterministically injects 8 errors into each Reed-Solomon codeword. The Reed-Solomon encoder used in this implementation has 16 check symbols, which supports the detection and correction of up to 8 errors per codeword. Therefore, the **Max Errors** channel option injects the maximum number of errors into the data path that the Reed-Solomon decoder can correct.

The **Variable BER** channel lets you select one of several different bit error rates (BERs). This channel injects errors routinely into each codeword, spreading the errors farther apart as lower BERs are selected. The highest BER setting adds 12 symbol errors to each codeword, which is more than this implementation of the Reed-Solomon core can correct. Therefore, this setting causes the Reed-Solomon decoder to fail, and both of the transmitted pictures return with errors.

The **Burst Error** channel injects an 40-symbol error into the first codeword of each data packet. If the **Add Interleaver/Deinterleaver** option is turned off, this channel causes the Reed-Solomon decoder to fail and both pictures return with errors. However, if the interleaver is added to the data path, the 40-symbol error is spread across 5 different codewords and the Reed-Solomon decoder can detect and correct all of the errors.

Adding a Custom Picture

All bitmaps of the same size have the same structure, therefore you can create a custom picture to transmit. Save your image as **my_file.bmp** in the <*installation directory*>**/bmps** directory. The image must be 183 pixels wide by 260 pixels tall.

For example, you can create an image using the Microsoft Paint software by performing the following steps:

1. View the image on your PC, for example in a web browser or other viewer.
2. Press the **Print Screen** button on your keyboard to copy the image to the clipboard.
3. Open Paint and paste the clipboard contents into a new file.
4. Choose the **Attributes** command (Image menu), set the height to 260 pixels and the width to 183 pixels, and click **OK**. If your current picture is bigger than the new size, it is cut from the right side and bottom to fit within the smaller area. If the current picture is smaller than the new size, the extra area is filled with the selected background color.

Technical Support

For technical support on this demonstration, send e-mail to **dsp@altera.com** or call Altera Applications at (800) 800-EPLD.

More Information

For more information about the Altera products used in this reference design, refer to the following documents:

- *System-on-a-Programmable-Chip Development Board User Guide*
- *Reed-Solomon Compiler MegaCore Function User Guide*
- *Symbol Interleaver/Deinterleaver MegaCore Function User Guide*
- *APEX 20K Programmable Logic Device Family Data Sheet*
- *AN 116: Configuring APEX 20K, FLEX 10K & FLEX 6000 Devices*

ALTERA®

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