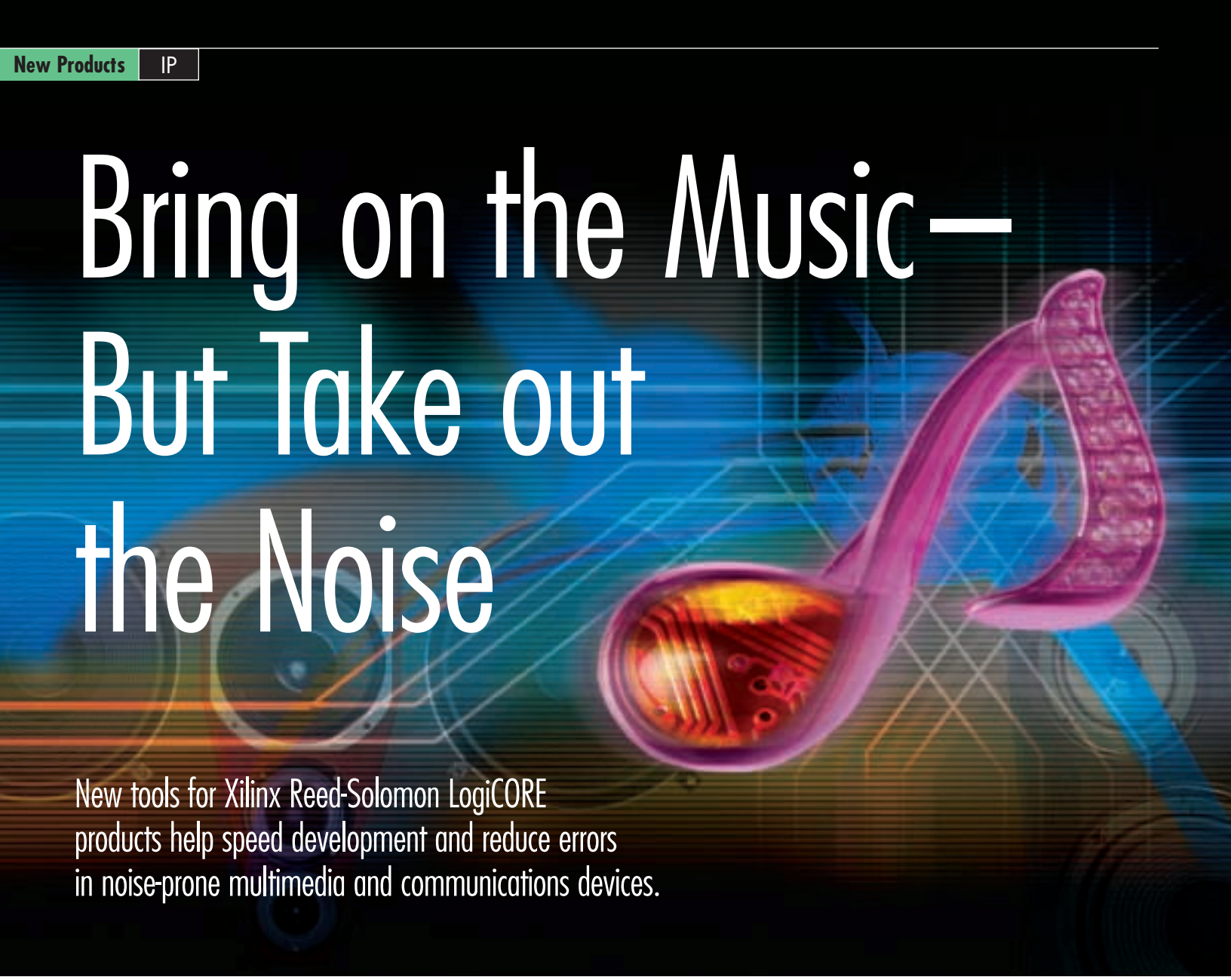
Bring on the Music—But Take out the Noise



New tools for Xilinx Reed-Solomon LogiCORE products help speed development and reduce errors in noise-prone multimedia and communications devices.

by Warren Miller
VP of Marketing
Avnet Design Services
Warren.Miller@avnet.com

Because Reed-Solomon codes are very good at correcting burst errors, they are used in applications where noise or defects can take out a series of information bits. Products like CD and DVD players, as well as wireless and hard-wired communications systems, use Reed-Solomon codes.

Successful communications in noisy environments using Reed-Solomon codes are possible through their means of detecting and correcting multiple errors without needing to retransmit the data. This increases bandwidth and improves data integrity in error-inducing communications channels.

Avnet Design Services has created a hardware reference design to demonstrate the functionality of two Xilinx LogiCORE™ Reed-Solomon IP cores in an error-prone system. The reference design uses standard Avnet-developed evaluation kits with the Raptor/ISD (in-system developer) IP environment from Experience First Inc.

Strength from the Core

The strength of Reed-Solomon codes in handling multiple bit errors makes them one of the most efficient and common error correction codes used in communications applications. Other codes (such as Hamming codes) are better suited for the more random error patterns found in memory systems, where only one bit at a time is affected.

Reed-Solomon codes append a series of symbols (check words) to a series of information symbols (data words). The check words are constructed (similar to the familiar parity check) in such a way that if errors are introduced anywhere in the data, the correct data can be reconstructed. The number of errors that can be corrected is one half the number of check symbols (rounded down). For 8-bit-wide data words, a common code has a block size of 207 words with 20 check symbols and 187 data symbols. This is the example code we will use in the reference design.

The Avnet Virtex Development Kit

The Virtex™ Development Kit developed by Avnet includes hardware tools on a single board for testing and debugging a refer-

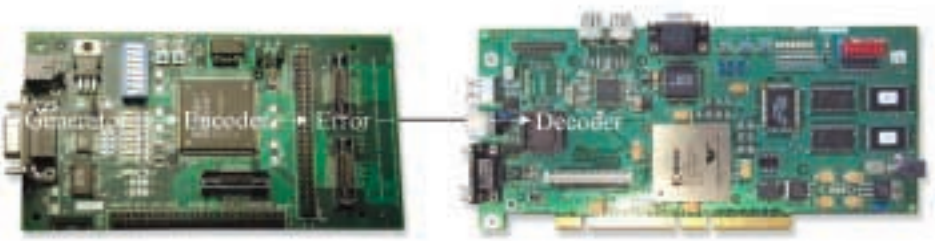


Figure 1 - The Avnet reference design hardware for Reed-Solomon code applications uses a Xilinx Spartan-II FPGA Encoder (left) and a Virtex-II Platform FPGA Decoder (right).

ence design implementing Reed Solomon codes on Xilinx Virtex and Virtex-II Platform FPGAs. The Avnet board includes the following features:

- Data Generator, which creates an input data stream
- Reed-Solomon Encoder, which takes blocks of input data and creates the check symbols
- Error Generator, which injects controlled pseudo-random error patterns into the communications channel
- Reed-Solomon Decoder, which detects and corrects errors in the transmitted data.

The Data Generator and Encoder reside on the Avnet-designed Xilinx Spartan™-II FPGA evaluation board, as shown in Figure 1 (left). The data block with appended check words is transferred to the Virtex evaluation board where the data is received and corrected by the Decoder, as shown in Figure 1 (right).

The Error Generator uses a pseudo-random number generator to select errors to inject into the information block. Adding errors to a reference design is necessary, because when there are no errors, it is not clear that anything is actually being done by the design. An IP development tool that works with the Avnet Virtex development board allows you to easily see the results of the decoder's processing in the reference design.

Raptor IP Development Environment

The Raptor IP Development Environment is an integrated software and firmware tool that works in conjunction with the Avnet Virtex Development Kit. The Raptor environment accelerates the development, integration, and test of IP or IP-based Xilinx FPGA designs. Raptor automatically inserts logic around any IP core to route inputs and outputs to real-time input signals, output signals, and to buffer memory available on the development board. This allows the core to be exercised at "hardware speeds."

Core output signals that are stored in the buffer memory can be read over the USB port to a host computer and displayed on the waveform viewer, just like software simulation results. This hardware speed approach to verification reduces the design/debug cycle time dramatically, making it easy to change the design or test set-up and see the results immediately. Used in conjunction with hardware-based input stimuli, verification of very complex and robust test suites are orders of magnitude faster than approaches based on pure software simulation.

The Raptor user interface, shown in Figure 2, allows you to specify the inputs and outputs to your IP. The necessary logic is automatically added around your IP and then compiled using the Xilinx development tools (Foundation™ Series software in this example). The IP core is then exercised by the hardware and the results are read out of memory and transferred to the PC over the USB port. The signals can be displayed on a waveform display (a sample output is shown in Figure 3). Because the signals are run at hardware speeds, hundreds of thousands of cycles can be exercised in the same time as a single cycle of software-only simulation.

Conclusion

Reed-Solomon codes are a basic building block of many digital communications systems. The Xilinx LogiCORE Encoder and Decoder were easily ported to the Avnet development boards, using the Raptor IP development system from Experience First. For more information on Avnet Development Boards visit www.ads.avnet.com or contact Warren Miller at Avnet Design Services (*warren.miller@avnet.com*). For more information on the Raptor ISD visit expfirst.com or contact Bob Read at Experience First (*bread@expfirst.com*).

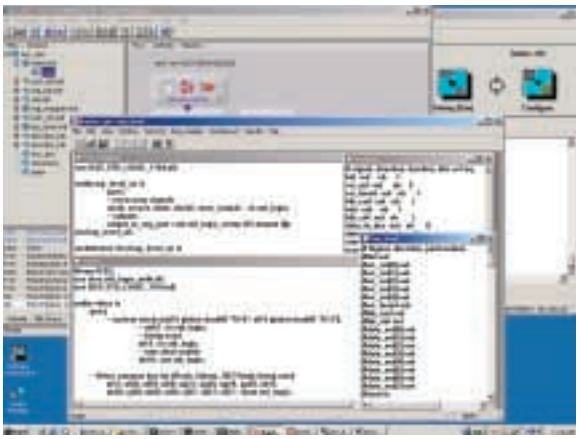


Figure 2 - Experience First's Raptor user interface lets you specify input and output signals to your IP reference design.

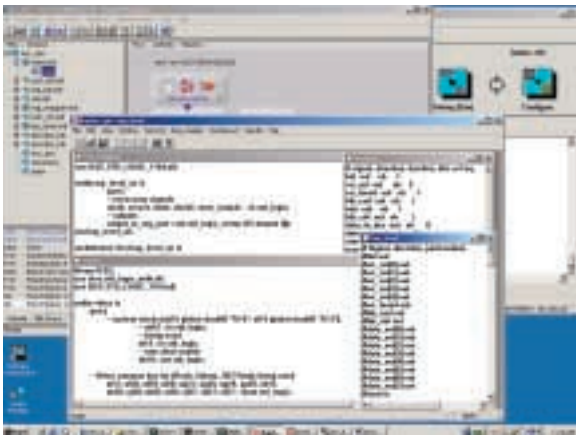


Figure 3 - Real-time Decoder waveform signals are input through the USB port.