XAPP683 (v1.0.1) March 19, 2004

Multi-Rate HD/SD-SDI Transmitter Using Virtex-II Pro RocketIO Multi-Gigabit Transceivers

Author: John F. Snow

Summary

The SD-SDI standard is widely used in broadcast studios and video production centers to transport standard definition (SD) digital video serially over video coax cable. The HD-SDI standard is similar, but transports high-definition (HD) digital video. The SD-SDI and HD-SDI standards are similar enough that it is possible to implement interfaces for video equipment that support both standards through the same connector.

This application note describes how to use the RocketIO™ multi-gigabit transceivers available in the Virtex-II Pro™ family of FPGA devices to implement a transmitter that can support both SD-SDI and HD-SDI. The flexibility of the RocketIO transceivers, combined with the programmable logic of the Virtex-II Pro devices, makes it possible to implement multi-rate SDI interfaces.

Since all Virtex-II Pro devices have four or more RocketIO transceivers, it is possible to implement multiple HD-SDI and SDI-SDI interfaces in a single FPGA.

Introduction

The SD-SDI standard, traditionally know simply as SDI, is defined by the SMPTE 259M standard [Ref 1] and also by the equivalent ITU-R BT.656 standard [Ref 2]. HD-SDI is defined by the SMPTE 292M standard. The SD-SDI standard is widely used in broadcast studios and video production centers today. Use of HD-SDI is increasing rapidly as the broadcast industry ramps up support for HDTV broadcasting.

Throughout this document, the term SD-SDI is used to refer to the SD version of the interface standard. SDI is used to refer generically to both SD-SDI and HD-SDI. And, multi-rate SDI is used to refer to support of both standards.

This application note focuses specifically on the implementation of a multi-rate SDI transmitter using the RocketIO transceivers available in the Virtex-II Pro FPGA family. A companion application note, XAPP684, describes the implementation of multi-rate SDI receivers, also using RocketIO transceivers. [Ref 3]

This application note refers to the Xilinx SD-SDI application note set [Ref 4] and the Xilinx HD-SDI transmitter application note, XAPP680 [Ref 5]. Please refer to these application notes for more information.

The RocketIO transceivers are designed to support serial bit rates from 622 Mbps to 3.125 Gbps. HD-SDI, with bit rates of approximately 1.5 Gbps, falls within this range. However, all of the current SD-SDI bit rates, ranging from 143 Mbps to 540 Mbps, are well below the range supported by the RocketIO transceivers. Therefore, the main focus of this application note is on a technique that allows the RocketIO transmitter to support these slower bit rates without violating any of the specifications of the RocketIO transceivers. XAPP680 describes how to use the RocketIO transceiver to build an HD-SDI transmitter. The combination of the SD-SDI transmission technique described here and the HD-SDI transmitter from XAPP680 results in a multi-rate SDI transmitter design. SD-SDI-only transmitters can also be built using the RocketIO transceivers, or they can be implemented in the fabric of the FPGA as described in the SD-SDI application note set.

Two reference designs are included with this application note. One is a multi-rate SDI transmitter and the other is an SD-SDI only transmitter. These reference designs have been tested and verified on the Xilinx SDV demo board [Ref 6].

SD-SDI and HD-SDI Similarities and Differences

The basic electrical specifications of HD-SDI and SD-SDI are virtually identical, making it possible to design multi-rate SDI interfaces that support both standards. Both standards have the same basic electrical interface: a singled-ended signal with an 800 mV peak-to-peak swing centered around 0.0V. Both use 75Ω coaxial cable and BNC connectors. And, both use the same encoding algorithm.

The most obvious difference between HD-SDI and SD-SDI is the much higher bit rate required to support HD video. HD-SDI has two bit rates: 1.485 Gbps and 1.485/1.001 Gbps. SD-SDI bit rates range from 143 Mbps to 540 Mbps.

To support the higher bit rates, HD-SDI transmitters must have faster rise and fall times on their outputs. In fact, the slowest rise and fall times permitted by the HD-SDI standard are too fast to be legal for SD-SDI.

Another difference is that HD video is 20 bits wide with two 10-bit channels, one for chroma and one for luma. An HD-SDI transmitter encodes and transmits 20 bits for every video clock cycle, whereas SD-SDI only encodes and transmits 10 bits per video clock cycle.

Due to the higher bit rate of HD-SDI, the maximum coax cable length supported by HD-SDI is 100 meters versus the 300-meter maximum supported by SD-SDI. However, HD-SDI also permits the use of an optical interface to allow longer transmission distances.

Generating SD-SDI Bitstreams with the RocketIO Transmitter

The original SD-SDI standard supports four bit rates:

- 143 Mbps for NTSC digital composite video
- 177.3 Mbps for PAL digital composite video
- 270 Mbps for NTSC and PAL digital component video
- 360 Mbps for NTSC and PAL 16:9 aspect ratio digital component video

In addition, a more recent document (SMPTE 344M) adds a 540 Mbps bit rate compatible with SD-SDI.

The digital composite video rates of 143 Mbps and 177.3 Mbps are rarely used. The most commonly used bit rate, by far, is 270 Mbps. It is quite common for a piece of video equipment to support only the 270 Mbps bit rate through its SDI interface. It is also common for equipment to support both 270 Mbps and 360 Mbps. The 540 Mbps bit rate is relatively new and not widely supported, yet.

All of these bit rates are well below the 622 Mbps minimum bit rate supported by RocketIO transceivers in Virtex-II Pro devices. Therefore, it is necessary to work around this minimum bit rate limitation in order to support SD-SDI with the RocketIO transceivers.

To transmit SD-SDI, the RocketIO transceiver is configured to run at some integer multiple of the SD-SDI bit rate and each bit is sent multiple times consecutively. For example, given a reference clock of 54 MHz, a RocketIO transceiver multiplies this reference clock by 20 resulting in a 1.08 Gbps bitstream—exactly four times 270 Mbps. If each encoded bit is transmitted by the RocketIO transmitter four times consecutively, the RocketIO transmitter produces a bitstream that is, in every way, equivalent to a normal 270 Mbps SD-SDI bitstream. Figure 1 shows this in more detail.

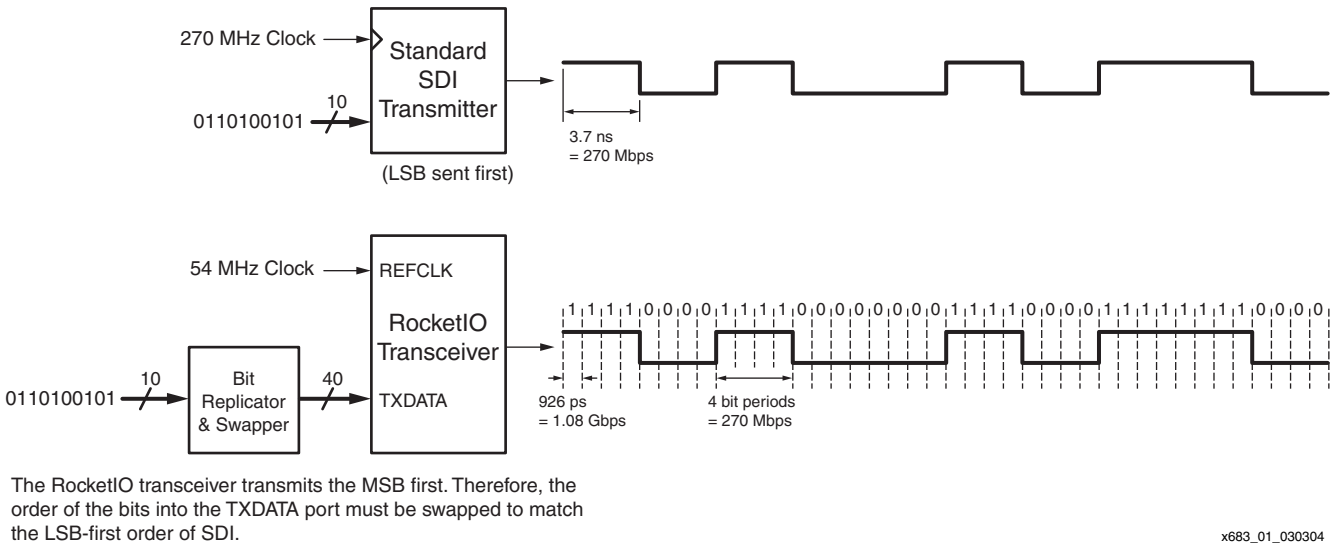

Figure 1: **RocketIO Transmitter Producing a 270 Mbps SD-SDI Bitstream**

Any bitstream frequency that is any integer multiple of the SD-SDI bit rate can be used, so long as it is fast enough to meet the minimum frequency requirements for the RocketIO transceiver. The "Clocks" section of this application note discusses clock requirements in detail.

Cable Driver

Both SD-SDI and HD-SDI use singled-ended (unbalanced) signaling over 75Ω coaxial cable. As mentioned in XAPP680, Xilinx does not recommend driving this single-ended interface directly using the RocketIO transmitter because the transceiver's CML outputs are designed to drive differentially. Instead, an SDI cable driver with differential inputs should be used.

When implementing a multi-rate SDI transmitter, use of a multi-rate SDI cable driver is essential. SD-SDI and HD-SDI have the same basic electrical specifications for the transmitter, but they do differ on one important specification—rise and fall time. SMPTE 259M requires that the rise and fall times of the SD-SDI signal must be no less than 400 ps and no more than 1.50 ns.(1) On the other hand, SMPTE 292M requires that the HD-SDI signal rise and fall time must be no more than 270 ps. In order to meet these differing requirements, a cable driver with adjustable slew rates must be used.

Figure 2 shows how to interface a GS1528 multi-rate SDI cable driver to the RocketIO transmitter. The GS1528 cable driver has 3.3V LVPECL differential inputs. The transmitter output of the RocketIO transceiver is a differential 2.5V CML pair not directly compatible with the cable driver's LVPECL inputs. AC coupling is used between the RocketIO transceiver's output and the input of the GS1528 to shift the signal levels from the 2.5V CML levels to the LVPECL levels. Large AC coupling capacitor values, in the range of 1 µF to 4.7 µF, must be used in order to successfully pass the pathological waveforms that can be generated by the HD-SDI and SD-SDI encoders.(2) The rate control input pin of the GS1528 sets the slew rate of the driver. This input must be High for SD-SDI and low for HD-SDI.

1. SMPTE 344M defines a new SD-SDI bit rate of 540 Mbps. To support this faster bit rate, SMPTE 344M requires compatible SDI transmitter to have rise and fall times of no less than 400 ps and no more than 800 ps. This is a subset of the 400 ps to 1.50 ns range allowed by SMPTE 259M.
2. The pathological waveforms for HD-SDI are documented in SMPTE RP 198. SMPTE RP 178 describes the similar set of pathological waveforms for SD-SDI. These waveforms are worst-case waveforms that can be generated by the SDI encoder and include a low frequency square wave pattern and a poorly DC balanced pattern.

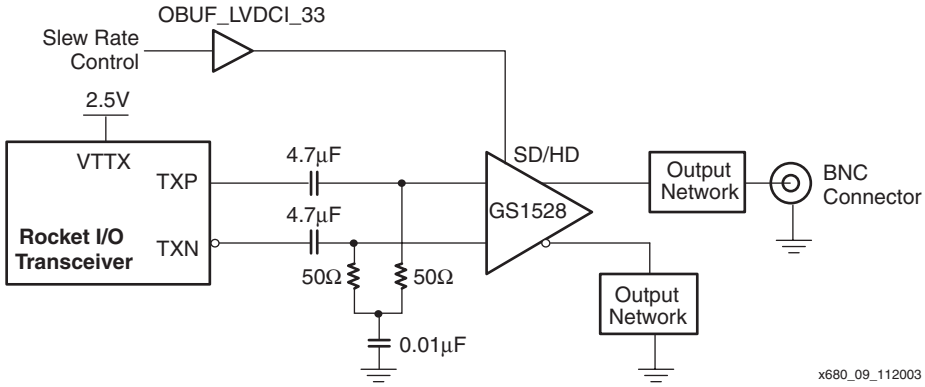

Figure 2: **Interfacing the GS1528 Cable Driver to the RocketIO Transmitter**

Clocks

One of the most important aspects of implementing an SDI transmitter using RocketIO transceivers is providing the right clocks to the transceivers. RocketIO transceivers requires two types of clocks, reference clocks and user clocks. The reference clocks are used to generate the bit-rate clock for the serializer. The user clocks are used to clock data from the fabric of the FPGA into the RocketIO transceiver. More details about the clocking requirements of the RocketIO transceivers can be found in the RocketIO Transceiver User Guide [Ref 7] and in XAPP680.

Reference Clocks

The reference clocks provide low-jitter frequency references for the RocketIO transceiver. The RocketIO transceiver multiplies the selected reference clock by 20 to obtain a bit-rate clock for the transmitter's serializer. Jitter present on the reference clock shows up as jitter on the transmitter output, so it is important that a low jitter reference clock be used.

When implementing an HD-SDI transmitter, the reference clock must be either 74.25 MHz or 74.25/1.001 MHz, depending on which HD-SDI bit rate (1.485 Gbps or 1.485/1.001 Gbps) is being transmitted. The 1.485 Gbps bit rate supports 60 Hz video update rate (and derivatives of 60 Hz such as 30 Hz, 25 Hz and 24 Hz). The 1.485/1.001 Gbps rate supports the 59.94 Hz video update rate (and derivatives) that are used primarily in North America.

For all of the SD-SDI bit rates, except 540 Mbps, the reference clock must be some multiple of the SD-SDI word-rate clock since the SD video clocks are too slow to be used directly as reference clocks to the RocketIO transceiver.

The source of the reference clocks is application specific. In most cases parallel digital video is supplied to the transmitter with an associated word-rate clock from some external source. For HD-SDI, the word-rate clock would be exactly the correct frequency needed for the reference clock to the RocketIO transmitter. However, for SD-SDI, the word-rate clock would typically need to be multiplied by at least two to get a clock fast enough for the RocketIO transceiver. In either case, keep in mind that an external PLL may be required to reduce the jitter on video clocks so that they are suitable for the RocketIO transceiver reference clocks. XAPP680 discusses jitter reduction requirements in more detail.

User Clocks

The user clocks load data into the RocketIO transceiver from the fabric of the FPGA. Each RocketIO transceiver requires two transmitter user clocks called TXUSRCLK and TXUSRCLK2.

TXUSRCLK must always be frequency locked to the selected reference clock. There is no required phase relationship between TXUSRCLK and the selected reference clock, but they must be exactly the same frequency.

The frequency and phase relationships between TXUSRCLK and TXUSRCLK2 depend on the width of the TXDATA input port of the RocketIO transceiver. For HD-SDI, it is usually most convenient to use a 20-bit wide TXDATA port as this matches the data word width of HD-SDI (10 bits of Y and 10 bits of C). When using a 20-bit TXDATA port, TXUSRCLK2 must have the same frequency and phase as TXUSRCLK (simply connect TXUSRCLK and TXUSRCLK2 to the same clock source). Consult the RocketIO Transceiver User Guide for TXUSRCLK2 requirements when other TXDATA port widths are used.

When implementing SD-SDI running at 270 Mbps, it would be slightly more convenient to use a 40-bit wide TXDATA port. This is because each encoded bit must be replicated four times if the transceiver's bit rate is four times the SD-SDI bit rate. Since each encoded SD-SDI word is 10 bits long, the resulting vector to the TXDATA port is 40 bits after bit replication. However, because the width of the TXDATA port is fixed at FPGA configuration time, a TXDATA port width that is suitable for both HD-SDI and SD-SDI must be chosen when implementing a multi-rate SDI interface. Because a 40-bit TXDATA port requires the use of a DCM to produce proper frequency and phase relationships between TXUSRCLK and TXUSRCLK2, it can be more desirable to use a 20-bit TXDATA port. So, the 40-bit SD-SDI bit vector resulting from bit replication must be sent to the TXDATA port as two 20-bit vectors, least significant half first.

The user clocks are usually global FPGA clocks that are also used to clock the parts of the SDI transmitter implemented in the FPGA fabric.

Clocking Example

There are many different ways to generate all the necessary reference clocks and user clocks needed for a multi-rate SDI transmitter. The details are application specific. The following example illustrates a common configuration used for a multi-rate SDI transmitter design.

In this example, shown in Figure 3, the parallel HD video and the parallel SD video enter the FPGA separately with separate video clocks. The HD video clock will be either 74.25 MHz or 74.1758 MHz, depending on the video sample rate. In this example, the SD video clock is always 27 MHz and the SD-SDI bit rate is always 270 Mbps.

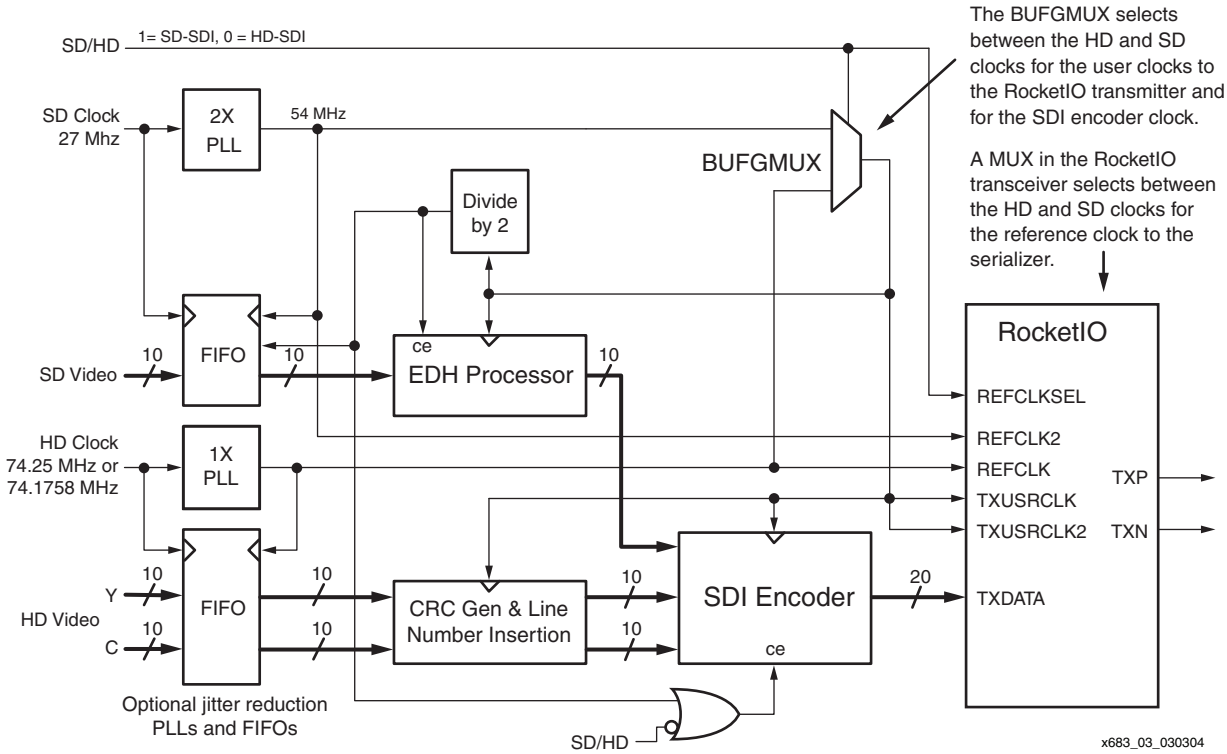

Figure 3: **Clocking Example**

If the HD and SD video clocks have too much jitter, then jitter reduction PLLs, external to the FPGA, must be used to reduce jitter on video clocks. See XAPP680 and XAPP247 for more details about jitter reduction. In addition, the 27 MHz SD clock must be doubled to 54 MHz before it can be used as a reference clock for the RocketIO transceiver. In this example, the external jitter reduction PLL for the SD-SDI clock also serves to double the frequency of this clock to 54 MHz.

The low-jitter HD video clock and the 54 MHz SD reference clock from the clock doubler are connected to the two REFCLK inputs of the RocketIO transceiver. The RocketIO transceiver contains a reference clock MUX that can be used to select between these reference clock sources.

The TXUSRCLK and TXUSRCLK2 inputs to the RocketIO transceiver must be the same frequency as the selected reference clock. The HD and SD reference clocks are multiplexed using a BUFGMUX to provide a global transmitter clock connected to the SDI encoder and the TXUSRCLK and TXUSRCLK2 inputs of the RocketIO transceiver. Note that much of the SD-SDI transmitter logic actually needs to run at 27 MHz, not 54 MHz. However, the global transmitter clock in the FPGA when running at SD-SDI rates is 54 MHz. So, a clock enable signal is generated from the 54 MHz clock that enables the clock of the SD-SDI transmitter logic every other clock cycle, effectively reducing the clock rate of the SD-SDI transmitter to 27 MHz. The SDI encoder module needs to have its clock enable asserted every other clock cycle for SD-SDI and every clock cycle for HD-SDI.

Since the CLK2X output of Virtex-II Pro's DCMs doubles the input clock, it is tempting to think about using a DCM to generate the 54 MHz reference clock from the 27 MHz input clock. Normally, Xilinx does not recommend using a DCM to generate a reference clock for the RocketIO transceivers, due to the output jitter specs of the DCM. However, in this particular case, more jitter can be tolerated on the RocketIO transceiver's reference clock input because the amount of jitter allowed at 270 Mbps is quite large.

There are some issues with the DCMs to keep in mind, however. First, the DCM does not reduce jitter. All jitter present on the DCM's input clock will be passed through the output clocks with some additional jitter added by the DCM. Thus, a low jitter input clock to the DCM is a requirement. Second, this should only be attempted with the CLK2X output of the DCM and never the CLKFX output since the CLKFX output produces more jitter. Third, use of the DCM as a clock doubler to produce a reference clock to the RocketIO transceiver has not been characterized in any way, and Xilinx cannot guarantee the RocketIO transmitter output jitter results obtained using this configuration.

More Thoughts on Clocks

In some applications more than two reference clock sources may be needed for a multi-rate SDI transmitter. If, for example, the FPGA itself is the source of the video, such as from a video test pattern generator as described in XAPP682 [Ref 8], the FPGA may need both 74.25 MHz and 74.25/1.001 MHz reference clocks for HD-SDI and also 54 MHz for SD-SDI. The RocketIO transceiver actually has four reference clock inputs (two REFCLK and two BREFCLK). However, the RocketIO transceiver is configured during FPGA configuration to use either the two REFCLK or the two BREFCLK inputs. Thus, only two reference clock inputs can be used without reconfiguring the RocketIO transceiver. There are a several solutions to this problem and these are discussed in detail in XAPP684.

For those applications where the FPGA is the source of video, a single reference clock frequency can support SD-SDI bit rates of 270 Mbps, 360 Mbps, and 540 Mbps. The 1.08 Gbps bitstream produced by the RocketIO transmitter when provided with a reference clock of 54 MHz, is exactly 4 * 270 Mbps, 3 * 360 Mbps, and 2 * 540 Mbps. With a transmitter designed to replicate each encoded SD-SDI bit two times when transmitting 540 Mbps, three times when transmitting 360 Mbps, and four times when transmitting 270 Mbps, all three bit rates can be derived from this one reference clock frequency.

Reference Designs

The reference design for this application note may be downloaded from the Xilinx website at http://www.xilinx.com/bvdocs/appnotes/xapp683.zip. The reference design is described in the following paragraphs.

The reference design consists of several modules that are used to transmit SD-SDI using a RocketIO transceiver. These modules can be combined with the HD-SDI transmitter design described in XAPP680 to implement a multi-rate SDI transmitter, or they can be used by themselves to implement an SD-SDI transmitter using a RocketIO transceiver.

An SDI transmitter that uses these modules will usually also require an error detection and handling (EDH) packet generator. EDH packets carry cyclic redundancy checksums (CRC) for each field allowing the receiver to check for transmission errors. EDH packets are used only for SD-SDI. HD-SDI uses a more robust error detection scheme with a CRC for each video line. An EDH processor that can generate and insert SD-SDI EDH packets is described in Xilinx application note XAPP299 [Ref 4].

Multi-Rate Encoder

Figure 4 shows the encoding algorithm used for both HD-SDI and SD-SDI. In this diagram, the encoding algorithm is shown as if implemented serially, one bit at a time. However, in the reference design encoder, encoding is done in parallel, processing 20 bits per clock cycle for HD-SDI and 10 bits per clock cycle for SD-SDI.

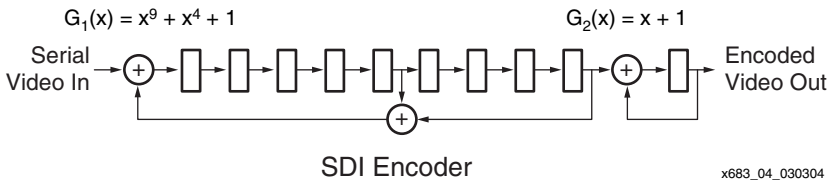

Figure 4: **SDI Encoding Algorithm**

XAPP680 includes a 10-bit SDI encoder module called smpte_encoder. A single smpte_encoder module is used to encode SD-SDI. Two of these encoders are combined to implement an HD-SDI encoder as in the hdsdi_encoder module from XAPP680.

One way to implement a multi-rate SDI encoder is to simply instantiate the hdsdi_encoder module from XAPP680 and an additional smpte_encoder module for SD-SDI and then multiplex the outputs of the encoders into the RocketIO transceiver's TXDATA port (after bit replication is done on the SD-SDI encoder output). The smpte_encoder module is small, so this method results in a fairly small multi-rate SDI encoder design.

However, a slightly more efficient implementation is found in the multi_sdi_encoder module. As shown in Figure 5, this module uses two smpte_encoder modules to encode the HD-SDI data, just like the hdsdi_encoder module. The Y channel smpte_encoder module is also used to encode the SD-SDI data. In order to switch between HD and SD modes of operation, two MUXes are added to the encoder, controlled by a signal that indicates whether the module is running in HD mode or SD mode. These MUXes select the feedback source for the Y encoder so that the feedback comes from the C encoder when encoding HD and from the Y encoder when encoding SD.

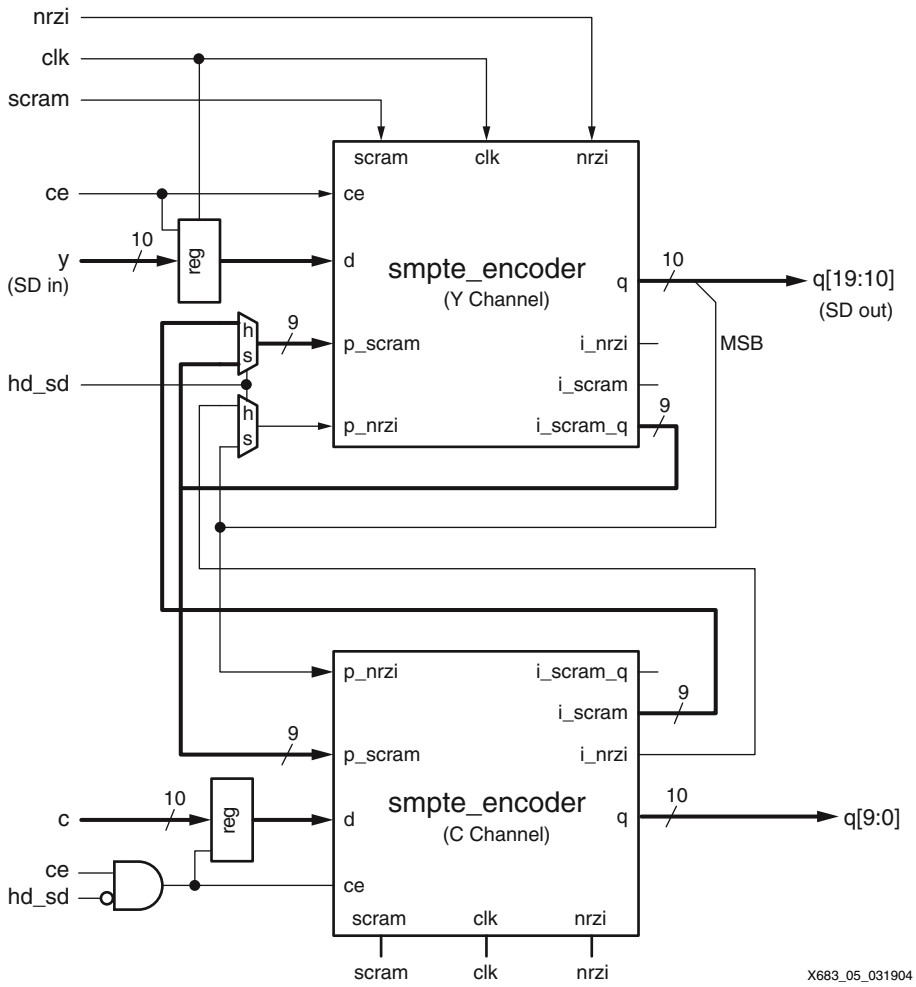

Figure 5: **Multi-Rate SDI Encoder Module**

SD-SDI Bit Replication

For SD-SDI, each bit from the encoder must be replicated some number of times depending on the reference clock frequency and the SD-SDI bitstream frequency. With a 54 MHz reference clock, each bit must be replicated four times to generate a 270 Mbps SD-SDI bitstream. This will produce 40 bits from each 10-bit input word. If the TXDATA port of the RocketIO transmitter is 20 bits wide, the 40 bits from the bit replicator must be multiplexed to first provide the least significant 20 bits during one TXUSRCLK cycle and the most significant 20 bits during the next TXUSRCLK cycle.

The multi_sdi_bitrep_4X module (Figure 6) is an implementation of a bit replicator designed for 4X bit replication. It includes an output MUX to produce a 20-bit vector for the RocketIO transceiver's TXDATA port. A control input indicates whether HD-SDI or SD-SDI is being encoded. If HD-SDI is encoded, the 20-bit input vector is simply passed through the module without any bit replication.

The bit replicator output vector must be bit-swapped to reverse the bit order before actually being connected to the TXDATA port of the RocketIO transceiver because the transceiver will transmit the MSB first. The hdsdi_rio_refclk module used in the reference design includes this bit swap function. The hdsdi_rio_refclk module is a wrapper around the RocketIO primitive GT_CUSTOM. It is identical to the hdsdi_rio module from XAPP680 except that the REFCLK inputs are active instead of the BREFCLK inputs. Due to the design of the SDV demo board, the REFCLK inputs must be used in this reference design instead of the BREFCLK inputs to the RocketIO transceiver.

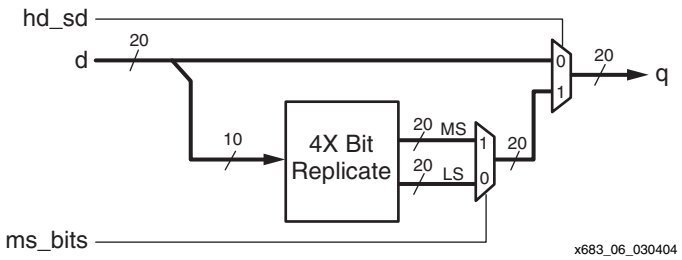

Figure 6: **SD-SDI Bit Replication**

Multi-Rate SDI Transmitter Example

The file sdv_multi_sdi_tx (Figure 7) contains an example of a multi-rate SDI transmitter. This file is designed to run on the Xilinx SDV demo board and the IOB constraints are specific to that board. The SDV demo board does not have the ability to accept parallel digital video, so in this example, video test pattern generators are used to produce the HD and SD video internally in the FPGA. Separate video pattern generators are used for HD-SDI (from XAPP682) and for SD-SDI (from XAPP248 [Ref 9]). In this example, the HD-SDI transmitter always runs at 74.25/1.001 MHz and the reference clock comes from a crystal oscillator on the SDV demo board. The 54 MHz reference clock for SD-SDI comes from another crystal oscillator.

This example includes CRC generation and insertion and line number insertion for HD-SDI as described in XAPP680. It also includes the EDH processor design from XAPP299 for generating and inserting EDH packets into the SD-SDI video.

In this example, txusrclk runs at 54 MHz in SD mode, twice the SD word rate. A flip-flop divides txusrclk in half to generate a clock enable signal for the SD video path. This signal enables the SD video pattern generator, EDH processor, and the encoder every other cycle of the 54 MHz clock. When running in HD mode, the SD clock enable is always negated to reduce the power consumed by the idle SD modules. The SD clock enable signal also controls the MUX in the bit replicator module (ms_bits input) to alternate between outputting the MS and LS encoded and replicated bits when running in SD mode.

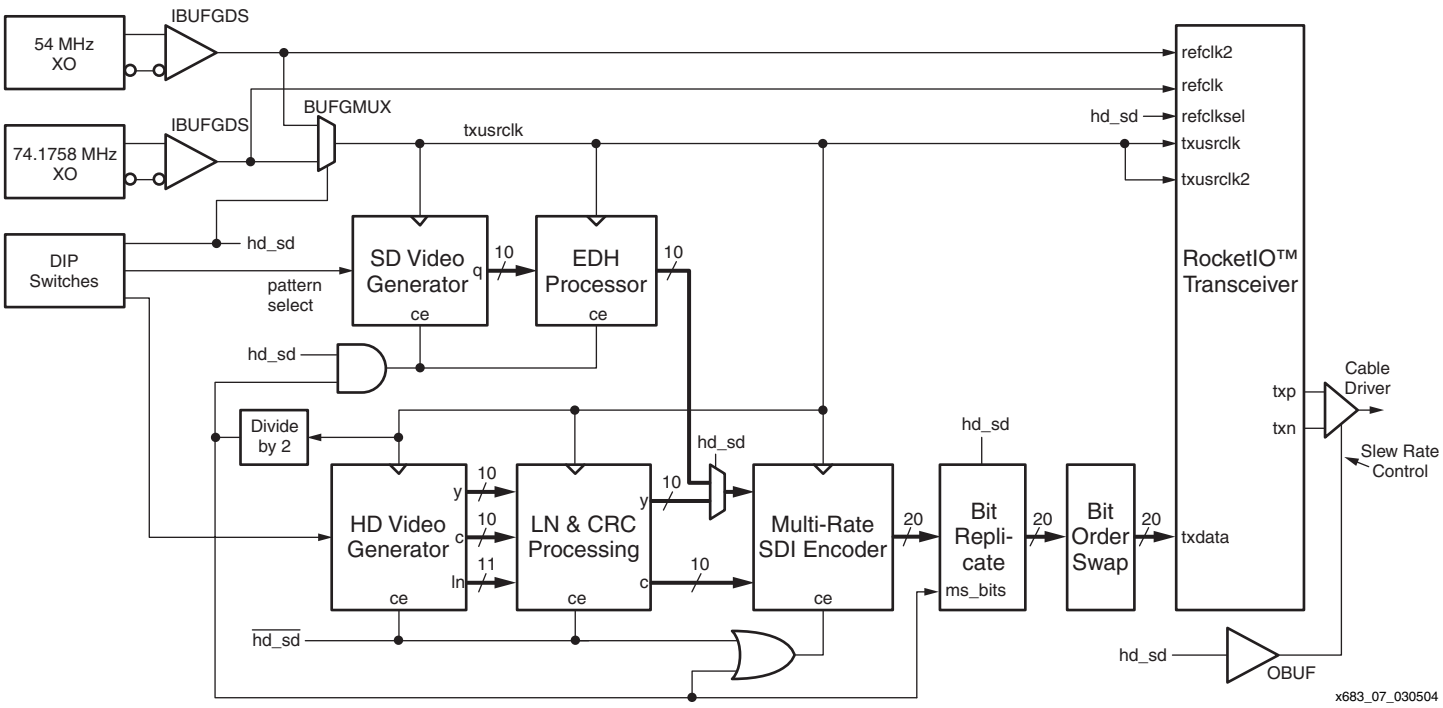

Figure 7: **Multi-Rate SDI Transmitter Example**

SD-SDI Transmitter Example Using the RocketIO Transceiver

A subset of the previous example illustrates how to implement an SD-SDI only transmitter using a RocketIO transceiver. This example is shown in Figure 8. The design includes the SD video test pattern generator and it also includes the EDH processor for EDH generation and insertion.

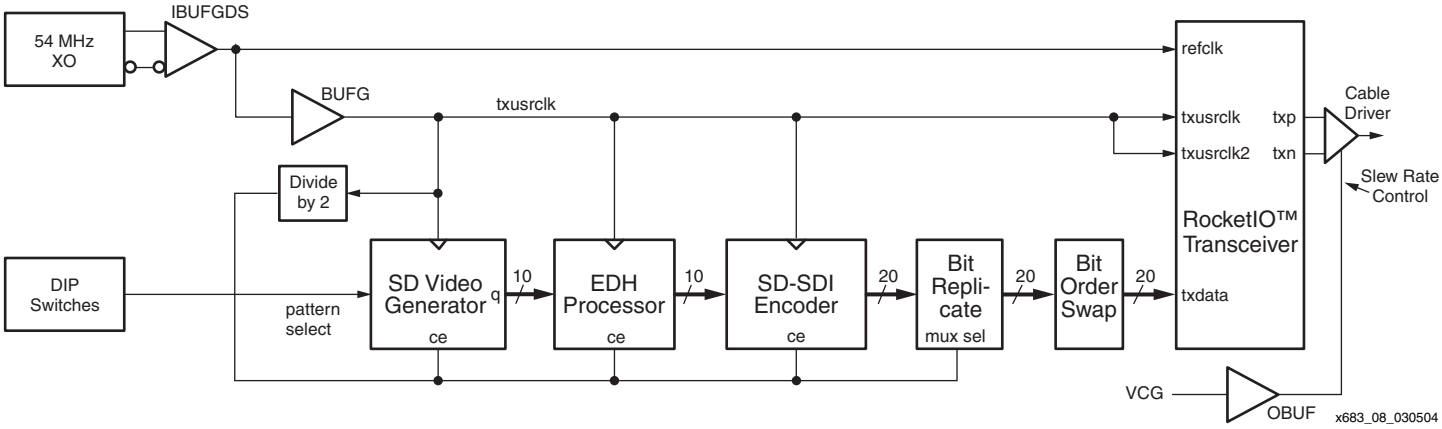

Figure 8: **SD-SDI Transmitter Example**

Jitter Performance

The output jitter of the RocketIO transmitter is dependent upon the amount of jitter present on the reference clocks, the intrinsic jitter of the RocketIO transceiver, jitter added by the cable driver, and any other jitter source.

The output jitter of an HD-SDI transmitter built with a RocketIO transceiver is documented in XAPP680. When generating SD-SDI bitstreams using the technique described here, the RocketIO transmitter can produce extremely low amount of output jitter as compared to standard SD-SDI transmitters available on the market. Table 1 shows the SD-SDI transmitter output jitter typically measured using the Xilinx SDV demo board. The low-jitter (less than 40 ps peak-to-peak) crystal oscillator was used as the reference clock to the RocketIO transceiver for these measurements. The jitter measurements include all sources of jitter on the board, including the cable driver. The measurements were taken with an SDI waveform analyzer connected to the SDV demo board with 1 meter of coax cable.

Table 1: **Typical SD-SDI Transmitter Output Jitter Values Using RocketIO Transceivers**

Timing Jitter	0.059 UI
Alignment Jitter	0.049 UI

Reference Design Size

Table 2 shows the FPGA resources used by the reference design. The size for just the multi-rate SDI encoder plus bit replicator function is shown. Also shown are the sizes of the two SDI transmitter examples, the multi-rate SDI transmitter and the RocketIO-based SD-SDI.

The largest piece of both of the transmitter examples is the SD-SDI EDH processor. In fact, XAPP299 shows that the implementation size of the EDH processor alone is actually larger than the entire SD-SDI transmitter example shown in Table 2 (and this example includes the EDH processor). In these examples, many of the receiver functions of the EDH processor module are not used and are optimized away by the implementation tools.

In all cases, the results were obtained using XST running under ISE 6.1. Area optimization was used. All designs were able to meet the necessary timing constraints using a Virtex-II Pro -5 speed grade device.

Table 2: **Reference Design Implementation Sizes**

Reference Design	FFs	LUTs	BRAMs
Multi-rate encoder and bit replicator	61	100	0
Multi-rate SDI Tx for SDV demo board	597	1013	6
SD-SDI only Tx for SDV demo board	420	725	3

Conclusions

This application note describes a technique that can be used to allow slow SD-SDI bitstreams to be transmitted using the RocketIO multi-gigabit transceivers in the Virtex-II Pro FPGA family. Combined with the HD-SDI transmitter design described in XAPP680, a multi-rate SDI transmitter can be implemented that supports both SD-SDI and HD-SDI.

All Virtex-II Pro devices contain multiple RocketIO transceivers, making it possible to implement four or more multi-rate SDI transmitters in a single Virtex-II Pro device. For applications that require multiple multi-rate SDI transmitters, this can provide a high level of integration, saving board space, power, and money as compared to discrete multi-rate SDI transmitter implementations.

References

1. All the SMPTE standards referenced in this application note are available from The Society of Motion Picture and Television Engineers. These standards can be purchased at the SMPTE web site: http://www.smpte.org.
2. The ITU-R BT.601-5 standard can be purchased from the International Telecommunication Union at http://www.itu.int/itudoc/itu-r/rec/bt/.
3. Xilinx application note XAPP684, "Multi-Rate HD/SD-SDI Receiver Using Virtex-II Pro RocketIO Multi-Gigabit Transceivers" by John F. Snow
4. The Xilinx SD-SDI applications notes are: XAPP247, "SDI Physical Layer"; XAPP288, "SDI Video Decoder"; XAPP298, "SDI Video Encoder"; XAPP299, "SDI Ancillary Data and EDH Processors"; and XAPP625, "Video Standard Detector and Flywheel Decoder".
5. XAPP680, "HD-SDI Transmitter Using Virtex-II Pro RocketIO Multi-Gigabit Transceivers" by John F. Snow.
6. The Xilinx SDV Demo board is available from Cook Technologies (part number CTXIL103). Further information is available at http://www.cook-tech.com.
7. UG024, *RocketIO Transceiver User Guide*, Xilinx, Inc.
8. XAPP682, "HDTV Video Pattern Generator" by John F. Snow.
9. XAPP248, "Digital Video Test Pattern Generators" by John F. Snow.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
03/15/04	1.0	Initial Xilinx release.
03/19/04	1.0.1	Minor edits throughout.