SMPTE 292M Scrambler/Descrambler IP Core

Beta Release

Cypress Semiconductor Corporation • 3901 North First Street • San Jose, CA 95134 • 408-943-2600
Revised July 21, 2004

1.0 Introduction

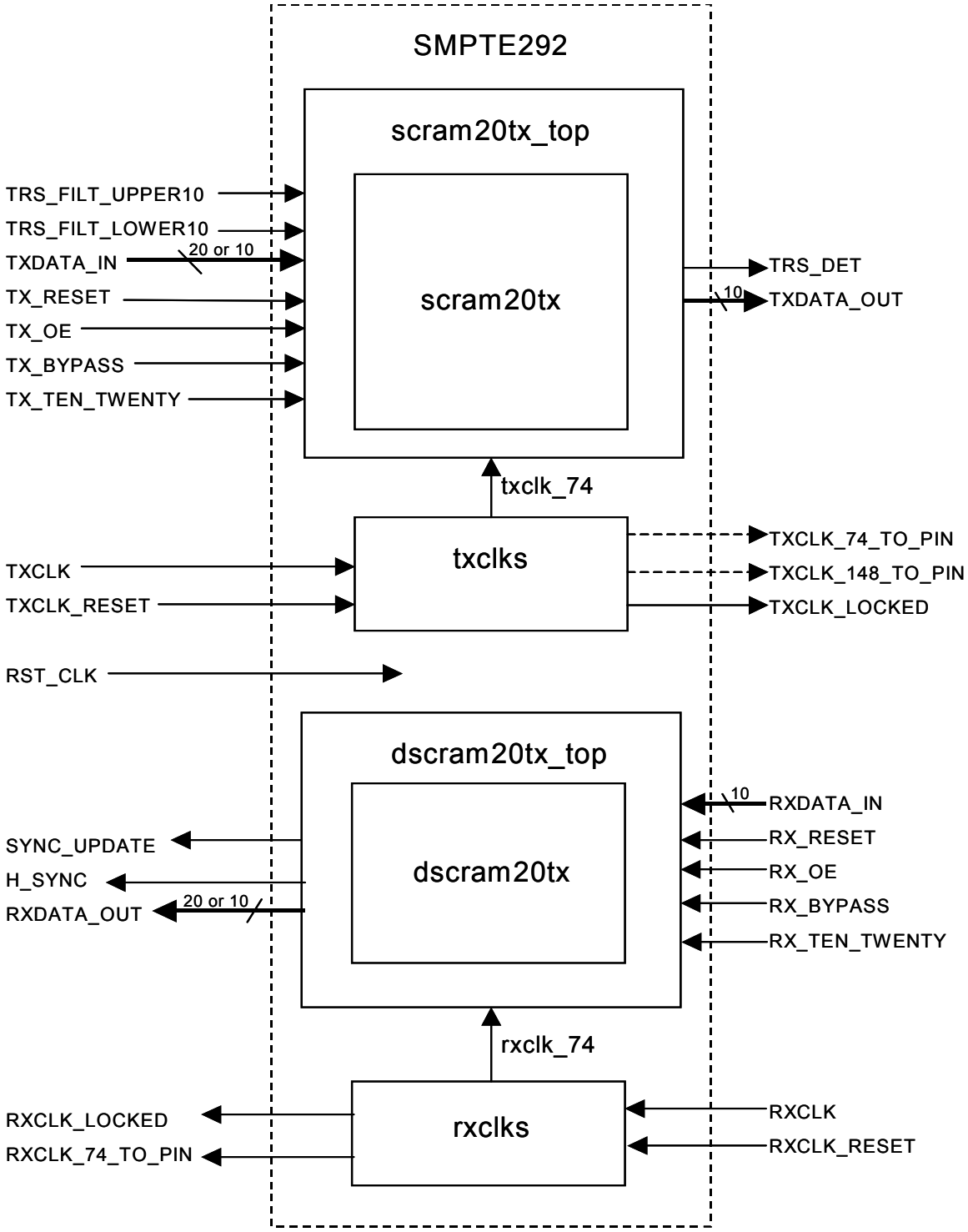
The SMPTE 292M Scrambler/Descrambler core is an HDL core that enables a designer to encode and decode SMPTE 260M and ANSI/SMPTE 274M data according to the SMPTE 292M standard. The designer can connect either a 10-bit or 20-bit wide data path video coprocessor to the Scrambler/ Descrambler and Cypress's HOTLink II transceiver device on the other side. The deliverables include Verilog-2001 and VHDL-2002 IP cores targeted towards Xilinx Spartan-III or Altera Cyclone FPGAs. A thorough test bench and all related design files are provided.

Most of the discussions below assume the reader has prior knowledge of the functionality of the Cypress HOTLink II™ PHYs. The testbench used for simulating the SMPTE 292M core is referred to as the TB.

2.0 Features

- Fully compatible with SMPTE 292M.
- Communicates with Cypress HOTLink II Video PHYs.
- Uses DDR I/Os and a 20-bit internal bus so the FPGA can operate with a 74.25-MHz clock, while still allowing for 1.485 Gbps serial data transmission by the Cypress transceivers.
- $X^9 + X^4 + 1$ scrambler and/or descrambler NRZI-to-NRZ decoder may be bypassed for raw data output.
- Compatible with 10-bit and 20-bit wide data-path video coprocessors.
- Scrambler and descrambler are completely independent of each other.

3.0 Block Diagram



4.0 Signal Descriptions

System Inputs	Width	Active:	Description
TXCLK	1	n/a	Clock Write. This clock controls all synchronous operations of the scrambler. It operates at the character rate which is equivalent to one 20th of the serialized bit-rate.
TXCLK_RESET	1	high	When active resets the scrambler's DLL or PLL
TX_RESET	1	high	When active resets the scrambler
TX_OE	1	high	When inactive, will tri-state the outputs of the scrambler
TX_BYPASS	1	high	Bypass SMPTE Encoding. If BYPASS is high at the rising edge of TXCLK, the data latched into the input register is routed around both the SMPTE scrambler and the NRZI encoder and presented to the output register. If BYPASS is low at the rising edge of TXCLK, the data present in the input register is routed through the SMPTE scrambler and NRZI encoder.
TX_TEN_TWENTY	1	n/a	When high, the video coprocessor-to-scrambler datapath interface will be 10 bits wide and clocked at a DDR. TXDATA_IN(10:19) and TRS_FILT_UPPER10 will be ignored. When low, the video processor to scrambler datapath interface will be 20 bits wide and clocked effectively at a single data rate. All bits of TXDATA_IN will be used. This input is ignored when the "ten_twenty_param" parameter is set to 0, which would permanently create a 20-bit coprocessor-to-scrambler interface.
RXCLK	1	n/a	Recovered Clock Read. This clock controls all synchronous operations of the descrambler. It operates at the character rate which is equivalent to one 20th the serialized bit-rate.
RXCLK_RESET	1	high	When active resets the descrambler DLL(s) or PLL
RX_RESET	1	high	When active resets the descrambler
RX_OE	1	high	When inactive, will tri-state the outputs of the descrambler
RX_BYPASS	1	high	Bypass SMPTE decoding. If BYPASS is high at the rising edge of RXCLK, the data latched into the input register is routed around both the NRZI decoder and the SMPTE descrambler and presented to the output register. If BYPASS is low at the rising edge of RXCLK, the data present in the input register is routed through the NRZI decoder and SMPTE descrambler.
RX_TEN_TWENTY	1	n/a	When high, the video coprocessor-to-descrambler datapath interface will be 10 bits wide and clocked at a DDR. RXDATA_OUT(10:19) will be ignored. When low, the video processor to descrambler datapath interface will be 20 bits wide and clocked effectively at a single data rate. All the bits of RXDATA_OUT will be used. This input is ignored when the "ten_twenty_param" parameter is set to 0, which would permanently create a 20-bit descrambler-to-coprocessor interface.
RST_CLK	1	n/a	Mandatory clock input for internally registering TXCLK_RESET, RXCLK_RESET, TX_RESET, and RX_RESET. The clock input should be equal to or less than 74.25 MHz. For any proper reset to occur, the respective reset line should be held active for at least 5 RST_CLK cycles.

System Outputs	Width	Active:	Description
TXCLK_74_TO_PIN	1	n/a	Output version of the input TXCLK. Optional for clocking the transceiver. In Altera designs this output will not be active if the 148.5 MHz clock output option is chosen.
TXCLK_148_TO_PIN	1	n/a	Output version of the input TXCLK multiplied by 2. Optional for clocking the transceiver. In Altera designs this output will not be active if the 74.5 MHz clock output option is chosen.
TXCLK_LOCKED	1	high	When active indicates the scrambler's DDL or PLL has locked
RXCLK_LOCKED	1	high	When active indicates the descrambler's DDL(s) or PLL has locked
RXCLK_74_TO_PIN	1	n/a	Output version of the input RXCLK. Optional for clocking the receiving video co-processor.

Signal Inputs	Width	Active:	Description
TRS_FILT_UPPER10	1	low	TRS Character Filter. When TX_TEN_TWENTY is low, this signal will control TXDATA_IN(10:19) that is clocked in on the rising edge of TXCLK. When TX_TEN_TWENTY is high, this signal will be ignored. It controls an internal filter that converts the low-order two bits of all TRS characters to the same state as the upper eight bits. This allows a proper 60-bit TRS ID to be generated when the scrambler is operated with 8-bit or non-standard video streams. When this signal is low, all characters from 000–003 are converted to 000, and all characters from 3FC–3FF are converted to 3FF. When TRS_FILT_UPPER10 is disabled high (and TX_TEN_TWENTY is low), all characters from TXDATA_IN(10:19) are passed to the scrambler without modification. Bit 10 is the LSB and bit 19 is the MSB of TXDATA_IN(10:19).
TRS_FILT_LOWER10	1	low	TRS Character Filter. When TX_TEN_TWENTY is high, this signal will control TXDATA_IN(0:9) that is clocked in on the rising and falling edge of TXCLK. When TX_TEN_TWENTY is low, this signal will control TXDATA_IN(0:9) that is clocked in on the rising edge of TXCLK. It controls an internal filter that converts the low-order two bits of all TRS characters to the same state as the upper eight bits. This allows a proper 60-bit TRS ID to be generated when the scrambler is operated with 8-bit or non-standard video streams. When this signal is low, all characters from 000–003 are converted to 000, and all characters from 3FC–3FF are converted to 3FF. When TRS_FILT is disabled high, all characters from TXDATA_IN(0:9) are passed to the scrambler without modification. Bit 0 is the LSB and bit 9 is the MSB of TXDATA_IN(0:9).
TXDATA_IN	10 or 20	n/a	Data input to the scrambler from the video coprocesseor. When TX_TEN_TWENTY is high, inputs 0-9 should be used at DDR while inputs 10-19 should be ignored. When TX_TEN_TWENTY is low, all inputs should be used at a single data rate. Bit 0 is the LSB and bit 9 or 19 will be the MSB.
RXDATA_IN	10	n/a	Data input to the descrambler from the Cypress Hotlink II transceiver. It is clocked in on both the rising and falling edge of RXCLK. Bit 0 is the LSB and bit 9 is the MSB.

Signal Outputs	Width	Active:	Description
TRS_DET	1	low	TRS Character Detected by the scrambler. This output indicates when a character used in the TRS sequence is detected in the input register. If the data contains any of the reserved characters of 000–003 or 3FC–3FF in 10-bit hex, the output will be low for one clock period. If the character in the input register is any other pattern this output will remain high
TXDATA_OUT	10	n/a	Data output from the scrambler to the Cypress Hotlink II transceiver. It is clocked out on both the rising and falling edge of TXCLK. Bit 0 is the LSB and bit 9 is the MSB.
SYNC_UPDATE	1	high	Sync update, output of the descrambler. This output pulses high when a TRS sequence is detected that is offset from its previous 10-bit character offset. If the descrambler-to-coprocessor interface is a 10-bit datapath, the pulse will be one-half cycle of RXCLK. If the descrambler-to-coprocessor interface is a 20-bit datapath, the pulse will be a full cycle of RXCLK. This output pulses at the same time as the H_SYNC signal toggles.
H_SYNC	1	n/a	Horizontal Sync, output of the descrambler. This output toggles once every time the TRS field is recognized. For 10-bit configurations, it changes state one 74.25 MHz clock cycle prior to the first character of the TRS field (3FF in 10-bit hex) appearing at the RXDATA_OUT(0:9) outputs. For 20-bit configurations, it changes state one 74.25 MHz clock cycle prior to the first character of the TRS field (FFFFF in 20-bit hex) appearing at the RXDATA_OUT(0:19) outputs. This output toggles to indicate detection of a TRS sequence even when the TRS characters are at a different offset from the present offset.
RXDATA_OUT	10 or 20	n/a	Data output from the descrambler to the video coprocessor. When RX_TEN_TWENTY is high, bits 0-9 are clocked out on both the rising and falling edge of RXCLK and bits 10-19 are unused. When RX_TEN_TWENTY is low, all bits are clocked out on the rising edge of RXCLK. Bit 0 is the LSB and bit 9 or 19 will be the MSB.

5.0 Functional Description

5.1 Scrambler

The Scrambler potion of the IP core is designed to encode SMPTE 260M and ANSI/SMPTE 274M bit-parallel digital characters (or other data formats) using the SMPTE 292M encoding rules. Following encoding, the characters are output as bit-parallel characters ready for serialization. The encoded outputs are designed to be directly mated to a HOTLink II Video transceiver, which then converts the bit-parallel characters into a SMPTE 292M compatible high-speed serial data stream.

This device performs both TRS (sync) detection and filtering, data scrambling with the SMPTE 259M $X^9 + X^4 + 1$ algorithm, and NRZ-to-NRZI encoding. These functions typically operate at a 74.25 MHz character rate, while utilizing a 20-bit-internal bus and DDR I/Os. For those systems operating with non-SMPTE 292M compliant video streams (or for diagnostic purposes), the scrambler and NRZI encoding functions can be disabled.

5.1.1 Xilinx DLL and Altera PLL

The scrambler expects a SMPTE 292 half-rate input clock of 74.25 MHz. The DLL or PLL will internally deskew this clock and use it to clock the entire scrambler data path logic. For optionally clocking the transceiver, the DLL or PLL can output a 74.25 MHz and/or a 148.5 MHz clock, depending upon which HDL module is chosen. HDL module options will be discussed in a later section.

5.1.2 Input Register

If the TX_TEN_TWENTY input signal is high, the input register will be clocked by the rising edge and falling edge of TXCLK. It will capture the TXDATA_IN(0-9) inputs and the TRS_FILT_LOWER10 input twice for every clock cycle and ignore the TXDATA_IN(10-19) inputs and TRS_FILT_UPPER10 input. If the TX_TEN_TWENTY input is low, the input register will be effectively clocked by only the rising edge of TXCLK. It will capture the TXDATA_IN(0-19) inputs and both the TRS_FILT_LOWER10 and TRS_FILT_UPPER10 every clock cycle.

In addition to the data and TRS filter inputs, all control inputs except TX_OE are also captured at each rising edge of TXCLK. This includes TX_RESET, BYPASS, and TX_TEN_TWENTY.

5.1.3 TRS Filter

The TRS Filters are used to convert all 8-bit TRS characters (000–003 and 3FC–3FF in 10-bit hex) to their full 10-bit value. If the TX_TEN_TWENTY input signal is low, if the TRS_FILT_LOWER10 is active (low), and any of these values are detected in the input register, the lower two bits of the data(0-9) are forced to either 0s or 1s respectively. If the TX_TEN_TWENTY input signal is low, if the TRS_FILT_UPPER10 is active (low) and any of these values are detected in the input register, then the lower two bits of the data(10-19) are forced to either zeros or ones respectively. If the TX_TEN_TWENTY input signal is high, the TRS_FILT_UPPER10 is ignored. But if TX_TEN_TWENTY is high, if TRS_FILT_LOWER10 is active (low), and any of these values are detected in the input register, the lower two bits of the data(0-9) are forced to either 0s or 1s respectively. This allows the encoder to be used with both 8-bit and 10-bit SMPTE character streams.

If TRS_FILT_LOWER_10 is high, the filter function for data(0-9) is disabled and all characters are passed from the input register to the SMPTE scrambler unmodified. If TRS_FILT_UPPER_10 is high as TX_TEN_TWENTY is low, the filter function for data(10-19) is disabled and all characters are passed from the input register to the SMPTE scrambler unmodified.

5.1.4 TRS Detector

TRS detector looks for the most significant eight bits of the input register (0-9 or 10-19) to be either all 1s or all 0s. If either of these values is detected, the TRS_DET output will go low following the rising edge of TXCLK, and remain low until a character is detected in the input register that is not all 0s or 1s.

5.1.5 SMPTE Scrambler

The SMPTE scrambler implements a parallel encoded version of a linear-feedback shift register. It encodes the data present in the input register using the X9 + X4 + 1 polynomial to increase the transition density of the serial data stream and to decrease the DC-content of the transmitted serial bit stream.

5.1.6 NRZI Encoder

The scrambled data is also fed through an NRZ-to-NRZI encoder. This also increases the transition density of the serial data stream, decreases the DC-content of the transmitted serial bit stream, and makes the serial stream insensitive to polarity inversions.

5.1.7 Permanent 20-bit Data Path Configuration

If it is decided to build the coprocessor-to-scrambler interface as a permanent 20-bit data path, then the “ten_twenty_param” parameter within the scramtx10_top HDL module can be set to 0. During synthesis and place & route (or fitting), this would optimize out the 10-bit data path framing logic. Leaving “ten_twenty_param” as 1 will allow the ten_twenty input signal to control whether or not the coprocessor-to-scrambler interface is a 10 or 20-bit data path.

5.1.8 Scrambler Latency—Xilinx Implementation

If the Xilinx scrambler is configured to communicate to the video coprocessor using a 10-bit data path, the latency it introduces to the data path will be 9 cycles of a 74.25-MHz clock. If the scrambler is configured to communicate to the video coprocessor using a 20-bit data path, the latency it introduces to the data path will be 4.5 cycles of a 74.25-MHz clock.

5.1.9 Scrambler Latency—Altera Implementation

If the Altera scrambler is configured to communicate to the video coprocessor using a 10-bit data path, the latency it introduces to the data path will be 9.5 cycles of a 74.25-MHz clock. If the scrambler is configured to communicate to the video coprocessor using a 20-bit data path, the latency it introduces to the data path will be 6 cycles of a 74.25-MHz clock.

5.2 Descrambler

The inputs of the descrambler core are designed to be directly mated to a CYV15G0*0*DXB HOTLink II transceiver, which converts the SMPTE 292M compatible high-speed serial data stream into 10-bit parallel characters for internal processing. The descrambler output is optionally 10 or 20 bits parallel data.

This descrambler core performs both TRS (sync) detection and framing, data descrambling with the SMPTE 292M $X^9 + X^4 + 1$ algorithm, and NRZI-to-NRZ decoding. These functions operate at a 74.25 MHz character rate, while utilizing a 20-bit-internal bus and DDR I/Os. For those systems operating with non-SMPTE 292M-compliant video streams (or for diagnostic purposes), the descrambler and NRZI decoding functions can be disabled.

5.2.1 Xilinx DLL(s) and Altera PLL

The descrambler expects a SMPTE 292 half-rate input clock of 74.25 MHz or a SMPTE 292 full-rate input clock of 148.5 MHz. If the 74.25 MHz input option is chosen, the DLL or PLL will internally deskew this clock and use it to clock the entire descrambler data path logic. If the 148.5 MHz input option is chosen, the DLL or PLL will internally deskew this clock, divide it by 2, and use it to clock the entire descrambler data path logic. For optionally clocking the transceiver, the DLL or PLL will output a 74.25 MHz clock. HDL module options will be discussed in a later section.

5.2.2 Input Register

The input register is clocked by the rising and falling edge of RXCLK. This register captures the RXDATA_IN (0-9) inputs twice for every clock cycle. In addition to the data inputs, all control inputs except OE, are captured at only the rising edge of RXCLK. This includes RX_BYPASS, RX_RESET, and RX_TEN_TWENTY.

5.2.3 NRZI-to-NRZ Decoder

The data in the input register is routed through an NRZI-to-NRZ decoder prior to being fed to the SMPTE descrambler. This removes the extra transitions added to the data stream by the NRZI encoder at the transmit end of the interface.

5.2.4 SMPTE Descrambler

Once the data has been converted back to NRZ, it is then routed through a linear feed-forward descrambler. It decodes the data present in the NRZ decode register using the $X^9 + X^4 + 1$ polynomial to remove the extra transitions added to the data stream at the transmit end of the interface.

5.2.5 TRS Framer

The TRS Framer is used to detect all 60-bit TRS sequences (FFFFF, 00000, 00000 in 20-bit hex) in the character stream. Anytime this sequence is detected, the H_SYNC output toggles.

This sequence is also used to frame the received characters so that the characters delivered to the output register are on their correct 20-bit boundaries. If the TRS sequence is detected in the decoded data stream, the character offset register is set to match the offset of the TRS sequence, and both the TRS sequence and the following characters are output on their proper 20-bit boundaries.

If a TRS sequence is detected whose character offset does not match that in the offset register, the SYNC_UPDATE signal will pulse high for one clock cycle, coincident with H_SYNC toggling.

5.2.6 Permanent 20-bit Data Path Configuration

If it is decided to build the descrambler-to-coprocessor interface as a permanent 20-bit data path, then the "ten_twenty_param" parameter within the dscramtx10_top HDL module can be set to 0. During synthesis and place and route (or fitting), this would optimize out the logic related to the 10-bit implementation. Leaving "ten_twenty_param" as 1 will allow the ten_twenty input signal to control whether or not the descrambler-to-coprocessor interface is a 10 or 20-bit data path.

5.2.7 Descrambler Latency—Xilinx Implementation

The Xilinx descrambler introduces a latency of 9 cycles of a 74.25-MHz clock regardless of 10 or 20 bit descrambler-to-coprocessor configuration.

5.2.8 Descrambler Latency—Altera Implementation

The Altera descrambler introduces a latency of 10 cycles of a 74.25-MHz clock regardless of 10 or 20 bit descrambler-to-coprocessor configuration.

6.0 Data Path Format

6.1 Coprocessor to Scrambler, 10-bit Data Path Only

The scrambler expects to receive 10-bit parallel data following a TRS in the exact order and format of 1^{st}-1111111111b, 2^{nd}-1111111111b, 3^{rd}-0000000000b, 4^{th}-0000000000b, 5^{th}-0000000000b, and 6^{th}-0000000000b. If the TRS_filt_lower10 input is active low, the 2 LSBs of each word can be 00b, 01b, 10b, or 11b and the scrambler TRS filter will translate them into either 1111111111b or 0000000000b as necessary.

6.2 Coprocessor to Scrambler, 20-bit Data Path Only

The scrambler expects to receive 20-bit parallel data following a TRS in the exact order and format of 1^{st}-11111111111111111111b, 2^{nd}-00000000000000000000b, and 3^{rd}-00000000000000000000b. If the TRS_filt_lower10 input is active low, the 2 LSBs of the lower 10 bits can be 00b, 01b, 10b, or 11b and the scrambler TRS filter will translate them into either 1111111111b or 0000000000b as necessary. If the TRS_filt_upper10 input is active low, the 2 LSBs of the upper 10 bits can be 00b, 01b, 10b, or 11b and the scrambler TRS filter will translate them into either 1111111111b or 0000000000b as necessary.

6.3 Descrambler to Coprocessor, 10-bit Data Path Only

The descrambler will send 10-bit parallel data to the coprocessor following a TRS in the exact order and format of 1^{st}-1111111111b, 2^{nd}-1111111111b, 3^{rd}-0000000000b, 4^{th}-0000000000b, 5^{th}-0000000000b, and 6^{th}-0000000000b.

6.4 Descrambler to Coprocessor, 20-bit Data Path Only

The descrambler will send 20-bit parallel data to the coprocessor following a TRS in the exact order and format of 1st-11111111111111111111b, 2nd-00000000000000000000b, and 3rd-00000000000000000000b.

7.0 HDL Module Options

The deliverable HDL modules are listed below by file name. A verilog testbench (TB) is used to test both the Verilog *and* VHDL configurations. Any file name that contains "xlx" is intended for a Xilinx Spartan-III FPGA only. Any module file that contains "atr" is intended for an Altera Cyclone FPGA only. Because of the numerous clocking options available, several clock files are provided for both Xilinx and Altera, both in Verilog and VHDL. However, as can be seen from the table, there are only 7 module names (no including the TB). Therefore exactly 7 files must be used for a complete design.

File name	Technology	HDL	HDL Module name	Function
smpte292.v	Independent	Verilog	smpte292	Top-level wrapper module. Wraps in modules: scramtx20_top, txclks, dscramrx20_top, and rxclks.
smpte292.vhd		VHDL		
scramtx20_xlx_top.v	Xilinx	Verilog	scramtx20_top	Scrambler top-level module. Contains the input and output DDR registers, performs framing according to the SMPTE 292 TRS, and wraps in the core scramtx20 logic.
scramtx20_xlx_top.vhd		VHDL		
scramtx20_atr_top.v	Altera	Verilog		
scramtx20_atr_top.vhd		VHDL		
scramtx20.v	Independent	Verilog	scramtx20	Scrambler core logic. Performs NRZI encoding and scrambling on a 20-bit-wide datastream.
scramtx20.vhd		VHDL		
txclks_xlx.v	Xilinx	Verilog	txclks	Scrambler clocking module. Deskews the 74.25 MHz input clock and outputs both a 74.25 MHz and 148.5 MHz clock for use with the transceiver.
txclks_xlx.vhd		VHDL		
txclks_74in_74out_atr.v	Altera	Verilog		Scrambler clocking module. Deskews the 74.25 MHz input clock and outputs a 74.25 MHz clock for use with the transceiver.
txclks_74in_74out_atr.vhd		VHDL		
txclks_74in_148out_atr.v		Verilog		Scrambler clocking module. Deskews the 74.25 MHz input clock and outputs a 148.5 MHz clock for use with the transceiver.
txclks_74in_148out_atr.vhd		VHDL		
dscramrx20_xlx_top.v	Xilinx	Verilog	dscramrx20_top	Descrambler top-level module. Contains the input and output DDR registers, performs framing according to the SMPTE 292 TRS, and wraps in the core dscramrx20 logic.
dscramrx20_xlx_top.vhd		VHDL		
dscramrx20_atr_top.v	Altera	Verilog		
dscramrx20_atr_top.vhd		VHDL		
dscramrx20.v	Independent	Verilog	dscramrx_top	Descrambler core logic. Performs NRZI decoding and descrambling on a 20-bit-wide datastream.
dscramrx20.vhd		VHDL		
rxclks_74in_74out_xlx.v	Xilinx	Verilog	rxclks	Descrambler clocking module. Deskews a 74.25 MHz input clock and outputs a 74.25 MHz clock for use with the video co-processor.
rxclks_74in_74out_xlx.vhd		VHDL		
rxclks_148in_74out_xlx.v		Verilog		Descrambler clocking module. Deskews a 148.5 MHz input clock, divides by 2, and outputs a 74.25 MHz clock for use with the video co-processor.
rxclks_148in_74out_xlx.vhd		VHDL		
rxclks_74in_74out_atr.v	Altera	Verilog		Descrambler clocking module. Deskews a 74.25 MHz input clock and outputs a 74.25 MHz clock for use with the video co-processor.
rxclks_74in_74out_atr.vhd		VHDL		
rxclks_148in_74out_atr.v		Verilog		Descrambler clocking module. Deskews a 148.5 MHz input clock, divides by 2, and outputs a 74.25 MHz clock for use with the video co-processor.
rxclks_148in_74out_atr.vhd		VHDL		
smpte292_tb.v	Independent	Verilog	smpte292_tb	Testbench which instantiates the smpte292 module

8.0 HDL Testing

Modelsim 5.8c is used to test each configuration of the SMPTE 292 IP core. An incrementing count is used as an input to both the 20-bit and 10-bit video co-processor-to-scrambler configuration. The data path output of the scrambler, which is always 10 bits wide, is routed back to the descrambler data path input, which is also always 10 bits wide. The 20-bit descrambler-to-video co-processor configuration output matches the incrementing count 20-bit input to the scrambler. The 10-bit descrambler-to-video co-processor configuration output matches the incrementing count 10-bit input to the scrambler. All other functions are tested as well, such as 1) bypass, 2) output enable, 3) resets, 4) DLL/PLL resets, 5) DLL/PLL lock output, 6) clock outputs, 7) scrambler and descrambler framing, 8) TRS filtering, 9) TRS detection, 10) Sync offset error, 11) Sync toggle, and 12) ten_twenty_param configuration.

To repeat these tests, Modelsim 5.8c is recommended. The functional and timing simulation libraries can be obtained from their respective vendors. For Xilinx simulations, the “glbl.v” file is needed for use as a co-top-level entity with the testbench. Notice also that the rxclk input can be either 74.25 or 148.5 MHz. This can be set on lines 67 and 68 of the testbench.

9.0 Xilinx Results

Below are the typical results using Xilinx ISE 6.2.02i to target the Verilog code to a Xilinx Spartan-III XC3S50-4TQ144 FPGA. Actual user results may vary. This device is typically the lowest-cost solution for this configuration. The VHDL utilization results are almost identical.

9.1 Scrambler (ten_twenty_param = 1)

Number of I/O: .....40 out of 97 (41%)
Number of Slices:.....167 out of 768 (21%)
Internal max freq:..... greater than 74.25 MHz

9.2 Scrambler (ten_twenty_param = 0)

Number of I/O: .....39 out of 97 (40%)
Number of Slices:.....99 out of 768 (13%)
Internal max freq:..... greater than 74.25 MHz

9.3 Descrambler (ten_twenty_param = 1)

Number of I/O: .....39 out of 97 (40%)
Number of Slices:.....304 out of 768 (39%)
Internal max freq:..... greater than 74.25 MHz

9.4 Descrambler (ten_twenty_param = 0)

Number of I/O: .....38 out of 97 (39%)
Number of Slices:.....240 out of 768 (31%)
Internal max freq:..... greater than 74.25 MHz

9.5 Turnkey Scrambler/Descrambler with DLLs (ten_twenty_param = 1)

Number of I/O:.....83 out of 97 (86%)
Total DCMs:.....2 out of 2 (100%)
Number of BUFGMUXs:.....3 out of 8 (62%)
Number of Slices:.....430 out of 768 (55%)
Internal max freq: .....greater than 74.25 MHz

10.0 Altera Results

Below are the typical results using Altera Quartus II 4.0 sp1 to target the VHDL code to an Altera Cyclone EP1C4F324C8 FPGA. Actual user results may vary. This device is typically the lowest-cost solution for this configuration. The Quartus VHDL compilation is more efficient than its Verilog compilation, so the Verilog logic element utilization may be up to 25% greater.

10.1 Scrambler (ten_twenty_param = 1)

Number of I/O:.....39 out of 249 (16%)
Number logic elements:.....376 out of 4000 (9%)
Internal max freq: .....greater than 74.25 MHz

10.2 Scrambler (ten_twenty_param = 0)

Number of I/O:.....38 out of 249 (15%)
Number logic elements:.....227 out of 4000 (6%)
Internal max freq: .....greater than 74.25 MHz

10.3 Descrambler (ten_twenty_param = 1)

Number of I/O:.....38 out of 249 (15%)
Number logic elements:.....514 out of 4000 (13%)
Internal max freq: .....greater than 74.25 MHz

10.4 Descrambler (ten_twenty_param = 0)

Number of I/O:.....37 out of 249 (15%)
Number logic elements:.....469 out of 4000 (12%)
Internal max freq: .....greater than 74.25 MHz

10.5 Turnkey Scrambler and Descrambler with PLLs (ten_twenty_param = 1)

Number of I/O:.....83 out of 249 (33%)
Total PLLs:.....2 out of 2 (100%)
Number logic elements:.....885 out of 4000 (22%)
Internal max freq: .....greater than 74.25 MHz

Approved AN4052 7/29/2004 kkvtmp