HB0832

CoreSDIRX v2.2

Handbook

Microsemi

a Microchip company

1 Revision History

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the most current publication.

1.1 Revision 3.0

Updated for CoreSDIRX v2.2.

1.2 Revision 2.0

Updated for CoreSDIRX v2.1.

1.3 Revision 1.0

Revision 1.0 was the first publication of this document. Created for CoreSDIRX v2.0.

2 References

SMPTE ST 259 -SDTV Digital Signal/Data - Serial Digital Interface

SMPTE ST 292-1 - 1.5 Gb/s Signal/Data Serial Interface

SMPTE ST 424 - 3 Gb/s Signal/Data Serial Interface

AMBA 3 APB Protocol Specification

UG0667 - Microsemi PolarFire FPGA User Guide

Contents

List of Figures

List of Tables

3 Introduction

3.1 Overview

CoreSDIRX DirectCore IP is a Serial Digital Interface (SDI) De-framer. CoreSDIRX supports Standard Definition SDI (SD-SDI) Level C, High Definition SDI (HD-SDI) and 3 Gigabits per second SDI (3G-SDI) Level A SDI standards defined by the Society of Motion Picture and Television Engineers (SMPTE).

3.2 Features

CoreSDIRX supports the following features:

- Compliant with SMPTE 259 (SD-SDI) standard
- Compliant with SMPTE 292 (HD-SDI) standard.
- Compliant with SMPTE 424 (3G-SDI) standard.
- Supports data rates 270 Mb/s and 270/1.001 Mb/s for SD-SDI Level C mode.
- Supports data rates 1.485 Gb/s and 1.485/1.001 Gb/s for HD-SDI mode.
- Supports data rates 2.97 Gb/s and 2.97/1.001 Gb/s for 3G-SDI Level A mode.
- Performs Non-Return-to-Zero Inverted (NRZI) decoding and descrambling.
- Performs word alignment of SDI data stream.
- Extracts timing reference information from Start of Active Video (SAV) and End of Active Video (EAV) packets.
- Extracts line number information from Line Number (LN) packets (in HD-SDI or 3G-SDI mode only).
- Performs Cyclic Redundancy Check (CRC) on SDI data stream (in HD-SDI or 3G-SDI mode only).
- Identifies and extracts video data and ancillary data.

3.3 Core Version

This handbook is for CoreSDIRX version 2.2.

3.4 Supported Families

- PolarFire®

3.5 Device Utilization and Performance

Device utilization and performance data is provided in Table 1 for the supported device families. The data listed in this table is indicative only. The overall device utilization and performance of the core is system dependent.

Table 1 Device Utilization

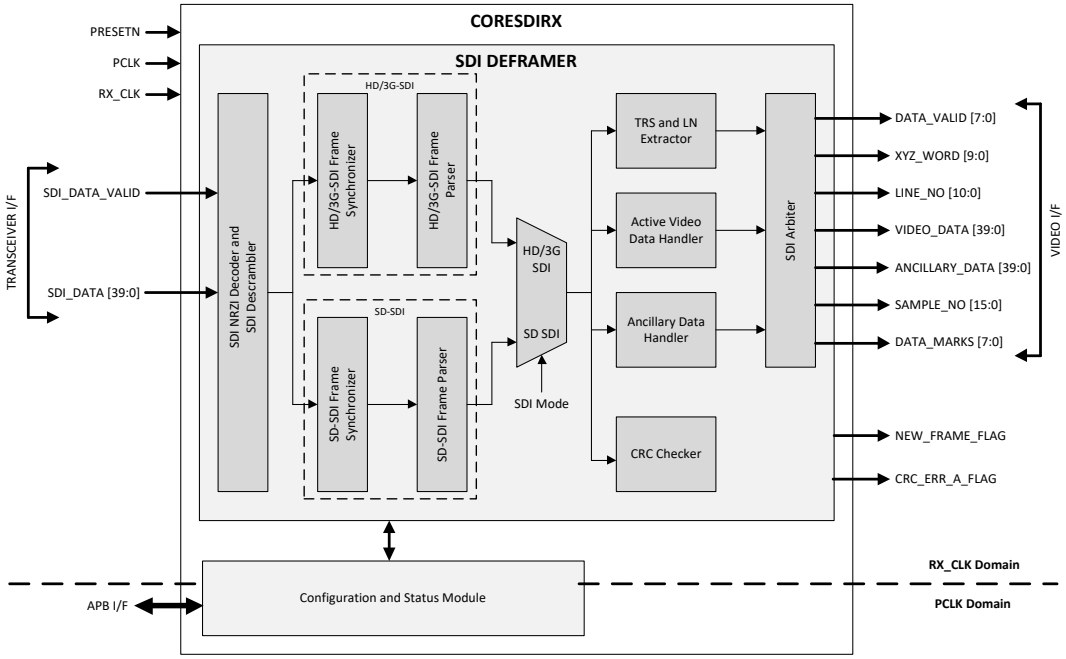
Family (Device)	Logic Elements				Performance (in MHz)	
	Sequential (DFF)	Combinatorial (4LUT)	Total	Percentage	RX_CLK Frequency	PCLK Frequency
PolarFire (MPF300T_ES)	3026	6279	9305	1.55	166	313

Note: The data in the Table 1 is achieved using typical synthesis and layout settings. Frequency (in MHz) was set to 100 and speed grade was Standard.

4 Functional Description

CoreSDIRX is a SDI Deframer. CoreSDIRX extracts the video data from the SDI data stream. Figure 1 shows the Functional block diagram of CoreSDIRX. The major blocks are SDI Descrambler and NRZI Decoder, SDI Frame Synchronizer, SDI Frame Parser, SDI CRC Checker, Active Video and Ancillary data handler, TRS and LN Extractor. The core has three interfaces, which are: Uncompressed Video Interface, Transceiver Interface, and APB Slave Interface. For more information on the interfaces refer Interface section.

Figure 1 CoreSDIRX Functional Block Diagram



4.1 SDI Descrambler and NRZI Decoder

SDI Descrambler and NRZI Decoder block performs the NRZI decoding and SDI Descrambling on the incoming SDI data stream. NRZI decoding is performed as per the polynomial specified in the SMPTE SD/HD/3G-SDI specification

$$G_2(X) = X + 1$$

The descrambling is performed as per the polynomial specified in the SMPTE SD/HD/3G-SDI specification

$$G_1(X) = X^9 + X^4 + 1$$

4.2 SDI Frame Synchronizer

The deserialized data from the Transceiver may not be aligned to the word boundary. This block detects the number of bit shift in the data word by checking for the header ($3FF_h$, 000_h, 000_h) sequence

present in the SDI stream. Once the number of bit shift is detected, the data words are aligned for the required word boundary.

HD/3G-SDI Frame Synchronizer performs frame synchronization when core is configured in HD-SDI or 3G-SDI mode. SD-SDI Frame Synchronizer performs frame synchronization when core is configured in SD-SDI mode.

4.3 SDI Frame Parser

This block parses the aligned SDI data stream and separates out the active video data, ancillary data, EAV / SAV packets, line number packets and CRC packets. The line number packets and CRC packets are extracted in HD-SDI or 3G-SDI mode only. This block indicates whenever a new video frame is received on the SDI data stream by asserting NEW_FRAME_FLAG for one cycle of receive clock.

HD/3G-SDI Frame Parser performs frame parsing when core is configured in HD-SDI or 3G-SDI mode. SD-SDI Frame Parser performs frame parsing when core is configured in SD-SDI mode.

4.4 SDI CRC Checker

SDI CRC checker block performs the CRC check on the SDI data stream. The CRC check is performed for every line of the video data. The 18-bit CRC computed by CRC checker is compared with the18-bit CRC data recovered from the CRC packets available in the SDI data stream. Core does CRC check for both Color-difference channel and Luma channel data. This block indicates whenever CRC error is detected by asserting CRC_A_ERR_FLAG for one cycle of receive clock. CRC checking is performed in HD-SDI or 3G-SDI mode only. CRC checking is as per CRC polynomial specified in the SMPTE HD/3G-SDI specification

$$CRC(X) = X^{18} + X^{5} + X^{4} + 1$$

4.5 Active Video and Ancillary Data Handler

This block handles the active video data and ancillary data. This block counts the number of samples received in each video line. This block generates the V-synch, H-synch and other data marks. Refer Ports description for more information on sample number and data marks.

4.6 TRS and LN Extractor

This block decodes the line number data from the LN packets extracted out from the SDI data stream by frame parser module. This block extracts the XYZ word from the EAV / SAV packets separated out from the SDI data stream by the frame parser.

4.7 SDI Arbiter

This module outputs the video data, ancillary data, line number, XYZ word, data marks and sample number data received from TRS and LN extractor block and data handler block on the uncompressed video interface aligned with the respective data valid signals.

4.8 Configuration and Status Module

This block configures the core for selected configuration based on the data written into configuration register. This block allows reading the status of the core. The configuration register and status registers are accessible through the APB interface. For more information on the registers refer Register Map section.

5 Register Map

The table lists registers available in CoreSDIRX. These registers can be accessed through APB Interface.

Table 2 Register Map

Address	Register Name	Type	Width	Reset value	Description
0x00	CFG_SDI	R/W	32	0x00000082	SDI Configuration Register.
0x10	STAT_SDI	R	32	0x00000000	SDI Status Register. Reports back the actual SDI configuration for which the core is currently working.
0x14	SDI_FRM_CNT	R	32	0x00000000	SDI frame count register. This register counts the number of SDI frames received. This is a roll-over count.
0x18	CRC_A_ERR_CNT	R	32	0x00000000	CRC error count register. This register counts the number of times CRC error(s) encountered in the received SDI frame(s). This is a roll-over count.

5.1 CFG_SDI

Register to configure the core for required mode color and Chroma subsampling.

Table 3 SDI Configuration Register

Address	Register Name	Type	Width	Reset value	Description
0x00	CFG_SDI	R/W	32	0x00000082	SDI Configuration Register

Table 4 SDI Configuration Register Bit Field Description

Bit	Name	Type	Reset Value	Description
31:25	Reserved	-	-	-
24	VALID	R/W	1	Valid bit. Active High. Indicates the configuration data is valid
23:10	Reserved	-	-	-
9:6	MODE	R/W	0010	**SDI Mode Type** 0010: HD-SDI 0000: SD-SDI - Level C 0101: 3G-SDI - Level A Other values are reserved.
5:3	Reserved	-	-	-
2:0	COLOR	R/W	010	**SDI Color Type** Value / Color Coding / Chroma Subsampling 000 / RGB / 4:4:4 001 / YCbCr / 4:4:4 010 / YCbCr / 4:2:2 Other values are reserved.

5.2 STAT_SDI

Register to get status of the core configuration in which it is currently working.

Table 5 SDI Status Register

Address	Register Name	Type	Width	Reset Value	Description
0x10	STAT_SDI	R	32	0x00000000	SDI Status Register. Provides the status of the core configuration in which it is currently working.

Table 6 SDI Status Register Bit Field Description

Bit	Name	Type	Reset Value	Description
31:10	Reserved	R	-	-
9:6	MODE	R	0000	**SDI Mode Type** 0010: HD-SDI 0000: SD-SDI - Level C 0101: 3G-SDI - Level A Other values are reserved.
5:3	Reserved	-	-	-
2:0	COLOR	R	000	**SDI Color Type** Value / Color Coding / Chroma Subsampling 000 / RGB / 4:4:4 001 / YCbCr / 4:4:4 010 / YCbCr / 4:2:2 Other values are reserved.

6 Interface

CoreSDIRX has following three interfaces:

Uncompressed Video Interface

A simple custom video data interface that outputs the uncompressed video data extracted from the incoming SDI data stream. CoreSDIRX outputs the uncompressed video as described in the Timing Diagram section.

Transceiver Interface

This interface accepts the 40-bit parallel data recovered by the Transceiver from the SDI data stream. The data from Transceiver is received at the rate at which the Transceiver is configured to operate in selected SDI mode SD/HD/3G-SDI.

APB Slave Interface

The APB interface provides access to the Configuration and Status registers of CoreSDIRX.

6.1 Configuration Parameters

There are no configurable parameters / generics.

6.2 Ports

The port signals for CoreSDIRX are described in the following table.

<table>
<tr><th>Port Name</th><th>Width</th><th>Type</th><th>Description</th></tr>
<tr><td colspan="4">Reset</td></tr>
<tr><td>PRESETN</td><td>1</td><td>Input</td><td>Active LOW asynchronous reset. Asynchronous assertion and synchronous de-assertion. This reset is synchronized and used inside the core with respect to each of the clock inputs.</td></tr>
<tr><td colspan="4">Clocks</td></tr>
<tr><td>PCLK</td><td>1</td><td>Input</td><td>APB system clock. All APB interface signals are clocked on the rising edge of this signal.</td></tr>
<tr><td>RX_CLK</td><td>1</td><td>Input</td><td>Receive clock from Transceiver. All the Transceiver interface signals and uncompressed video interface signals are clocked on rising edge of this clock.
Recommended to connect LANEx_RX_CLK of Transceiver.
<table>
<tr><th>Mode</th><th>Data rate</th><th>TX_CLK frequency</th></tr>
<tr><td rowspan="2">3G-SDI Level A</td><td>2.97 Gb/s</td><td>74.25 MHz</td></tr>
<tr><td>2.97/1.001 Gb/s</td><td>74.25/1.001 MHz</td></tr>
<tr><td rowspan="2">HD-SDI</td><td>1.485 Gb/s</td><td>37.125 MHz</td></tr>
<tr><td>1.485/1.001 Gb/s</td><td>37.125/1.001 MHz</td></tr>
<tr><td rowspan="2">SD-SDI</td><td>270 Mb/s</td><td>6.75 MHz</td></tr>
<tr><td>270/1.001 Mb/s</td><td>6.75/1.001 MHz</td></tr>
</table></td></tr>
</table>

Port Name	Width	Type	Description
Transceiver Interface			
SDI_DATA_VALID	1	Input	Receive data valid from Transceiver. Recommended to connect LANEx_RX_VAL of Transceiver.
SDI_DATA	40	Input	Receive data from Transceiver. Recommended to connect LANEx_RX_DATA of Transceiver.
Uncompressed Video Interface			
DATA_VALID	8	Output	Data Valid signal. This signal indicates which of the uncompressed video interface signal is valid. The bits are defined as follows and indicates: [7] Video data is valid. Accompanies: VIDEO_DATA DATA_MARKS SAMPLE_NO [6] Reserved [5] Ancillary data is valid. Accompanies: ANCILLARY_DATA SAMPLE_NO [4] Reserved [3] Reserved [2] Reserved [1] TRS (EAV or SAV) packet is valid. Accompanies: XYZ_WORD [0] Line number data is valid. Accompanies LINE_NO **Note:** DATA_VALID [0] bit is tied to zero when core is configured in SD-SDI mode.
XYZ_WORD	10	Output	The EAV or SAV word recovered from the TRS packet of the incoming SDI data stream. XYZ_WORD updates with the DATA_VALID [1] bit and remains static until the next TRS (SAV or EAV) packet.
LINE_NO	11	Output	Current line number of the video frame recovered from LN0 and LN1 packets of the incoming SDI data stream. LINE_NO updates aligned with DATA_VALID [0] bit and remains static until the next line number data is recovered. LINE_NO is tied to zero when core is configured in SD-SDI mode.
ANCILLARY_DATA	40	Output	ANCILLARY_DATA output recovered from the SDI frame. This port updates aligned with the DATA_VALID [5] bit. ANCILLARY_DATA [39:30] is the first 10 bit data word out of the 40-bit aligned word recovered from the SDI data stream. ANCILLARY_DATA [29:20] is the second 10 bit data word out of the 40-bit word recovered from the SDI data stream. ANCILLARY_DATA [19:10] is the third 10 bit data word out of the 40-bit word recovered from the SDI data stream. ANCILLARY_DATA [9:0] is the last 10 bit data word out of the 40-bit word recovered from the SDI data stream.
VIDEO_DATA	40	Output	Active video data recovered from incoming SDI data stream.

Port Name	Width	Type	Description
			VIDEO_DATA updates aligned with DATA_VALID [7] bit. Each 10 bit of the video data corresponds to C or Y channel data VIDEO_DATA [39:30] - **Cb** data (Blue difference chroma sample) VIDEO_DATA [29:20] - **Y0** data (Luma sample 0) VIDEO_DATA [19:10] - **Cr** data (Red difference chroma sample) VIDEO_DATA [9: 0] - **Y1** data (Luma sample 1)
DATA_MARKS	8	Output	DATA_MARKS recovered from incoming SDI frame. This port updates aligned with DATA_VALID [7] bit. This port indicates which 10-bit of the VIDEO_DATA output is the first and the last sample of an active video frame and an active video line. DATA_MARKS is valid for only one clock cycle of RX_CLK. The bits are defined as follows and indicates [7] 1st 10-bit word in VIDEO_DATA is 1st sample of the active video frame ("V-Synch") [6] 1st 10-bit word in VIDEO_DATA is 1st sample of the active video line ("H-Synch") [5] This is the last sample of the active video for this line of video. [4] This is the last of the active video samples for this frame. [3:0] Reserved
SAMPLE_NO	16	Output	Sample number of the video data or ancillary data. SAMPLE_NO indicates the current sample number of the active video data when DATA_VALID [7] bit is asserted. The count begins with sample 0 as the first sample after the SAV packet and counts up from there until EAV packet. SAMPLE_NO indicates the current sample number of the ancillary data when DATA_VALID [5] bit is asserted. The count begins with sample 0 as the first sample after the CRC packet and counts up from there until SAV packet in the horizontal ancillary data space (HANC). The count begins with sample 0 as the first sample after the SAV packet and counts up from there until EAV packet in the vertical ancillary data space (VANC). For 4:2:2 chroma subsampling, sample number data increments by 2 with every new data output. For 4:4:4 chroma subsampling sample number increments by 1 with every new data output.
FLAGS			
NEW_FRAME_FLAG	1	Output	SDI new frame flag. This flag goes high for one clock cycle of receive clock indicating a new video frame is received.
CRC_A_ERR_FLAG	1	Output	CRC error indication flag for HD-SDI and 3G-SDI Level A. This flag goes high for one clock cycle of receive clock indicating CRC error in the received SDI data. This flag is tied to zero when core is configured in SD-SDI mode.
APB Slave Interface			
PADDR	8	Input	APB address bus.
PSEL	1	Input	APB select signal. A HIGH indicates the slave device is selected and data transfer is required.

Port Name	Width	Type	Description
PENABLE	1	Input	APB enable signal. A HIGH indicates second and subsequent cycles of APB transfer.
PWRITE	1	Input	APB direction signal. Indicates APB write access when HIGH and APB read access when LOW.
PWDATA	32	Input	APB write data bus.
PREADY	1	Output	APB ready signal. A HIGH indicates slave is ready for APB data transfers.
PRDATA	32	Output	APB read data bus.
PSLVERR	1	Output	APB slave error signal. A HIGH indicates APB data transfer failure.

Note: x in LANEx can be 0, 1, 2, or 3

7 Timing Diagrams

7.1 APB Interface

Refer AMBA APB document for APB I/F timing.

7.2 Uncompressed Video Interface

The uncompressed video output from the CoreSDIRX is output as per this protocol:

- The timing reference signal (TRS) is output on XYZ_WORD port with DATA_VALID [1] signal asserted. This TRS can indicate either an EAV or a SAV data. The XYZ_WORD word remains static and updates when next EAV or SAV is recovered from SDI data stream.
- The line number data is output on LINE_NO port with DATA_VALID [0] asserted. Line number data is updated in the next clock cycle after EAV data is output on the XYZ_WORD port. The LINE_NO data remains static until next line number data is recovered from SDI data stream. The line number output is set to zero in SD-SDI mode.
- The ancillary data is output on ANCILLARY_DATA port with DATA_VALID [5] signal asserted. The first 40-bit ancillary data recovered from Horizontal ancillary data space (HANC) is output in next clock cycle after EAV data is output on XYZ_WORD port. The first 40-bit ancillary data recovered from the Vertical ancillary data space (VANC) is output in next clock cycle after SAV data is output on XYZ_WORD port.

  Note: In case of SD-SDI mode the ancillary data recovered from HANC is output in next cycle after EAV data is output on XYZ_WORD port.
- The active video data is output on VIDEO_DATA port with DATA_VALID [7] signal asserted. The first 40-bit recovered active video sample data is updated after SAV data is output on the XYZ_WORD port. Active video data updates for every clock cycle until EAV word is detected.
- DATA_MARKS is output in alignment with video data output. DATA_MARKS updates for the first sample and the last sample of every active video line. Data marks remains valid only for one clock cycle. For more information about DATA_MARKS refer Ports description table.
- The SAMPLE_NO is output in alignment with video data or ancillary data output. Sample number starts with 0 for the first sample data and increments until last sample of ancillary or active video data. For more information about SAMPLE_NO refer Ports description table.

Figure 2 and Figure 3 provides the uncompressed video interface timing diagrams during the active video region and vertical blanking region in HD/3G-SDI mode. These timing diagrams are for a 4:2:2 YCbCr video with the Raster resolution: 2200 X 1125 and active video resolution: 1920 X 1080.

Figure 4 and Figure 5 provides the uncompressed video interface timing during the active video region and vertical blanking region in SD-SDI mode. The timing diagrams is for a 4:2:2 YCbCr video with the Raster resolution: 1728 X 625 and active video resolution: 1440 X 576.

Note: A0, A2 …represent 40-bit ancillary data in VANC. H0, H2… represent 40-bit ancillary data in HANC. V0, V2… represent 40-bit active video data.

Figure 2 Uncompressed Video Interface Timing Diagram during Active Video Region in HD/3G-SDI mode

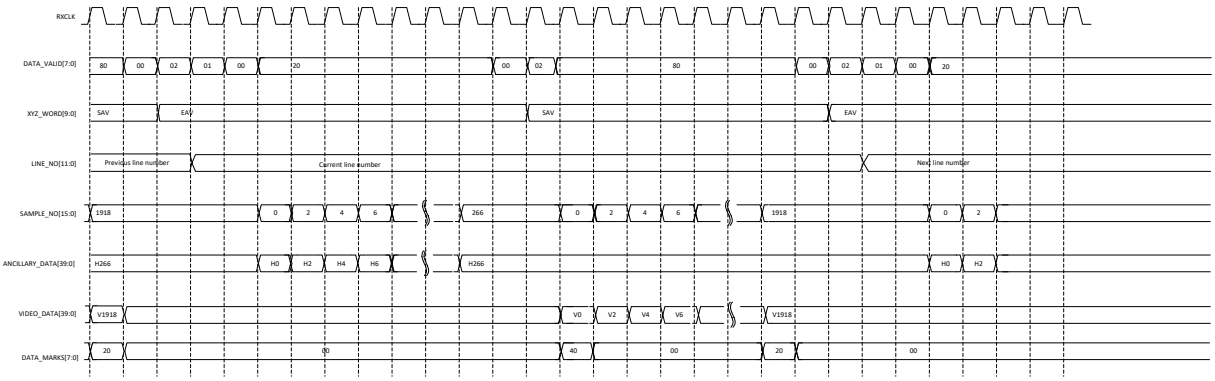


Figure 3 Uncompressed Video Interface Timing Diagram during Vertical Blanking Region in HD/3G-SDI mode

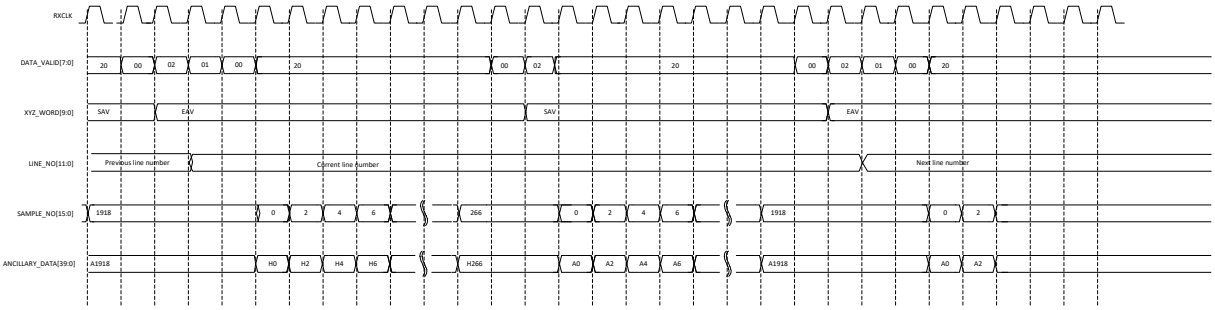


Figure 4 Uncompressed Video Interface Timing Diagram during Active Video Region in SD-SDI mode

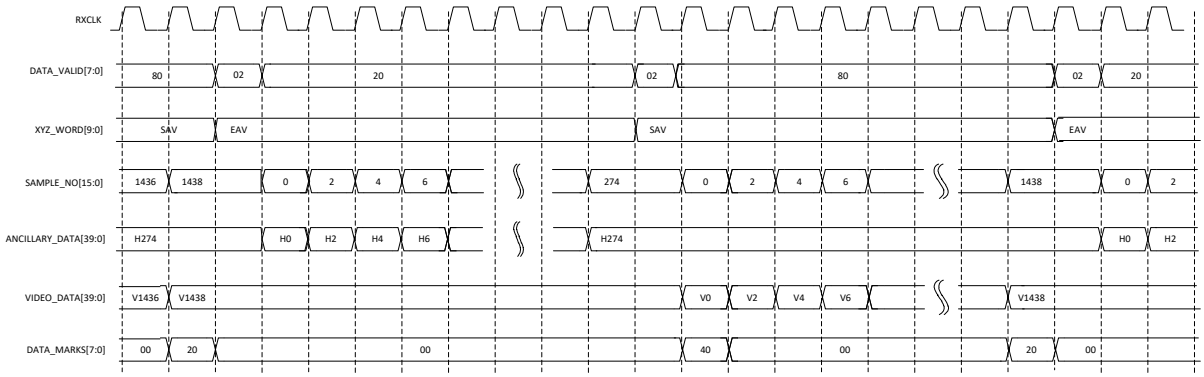
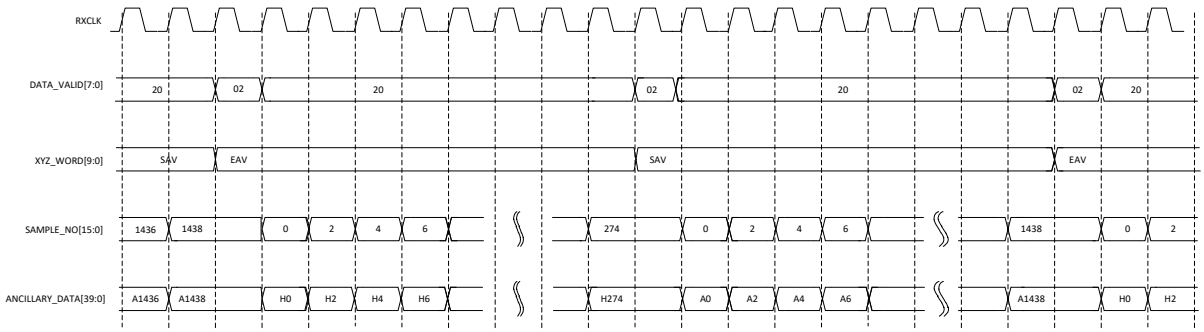


Figure 5 Uncompressed Video Interface Timing Diagram during Vertical Blanking Region in SD-SDI mode



8 Tool Flow

8.1 License

CoreSDIRX is available in two versions:

- Obfuscated
- Evaluation

The Evaluation version is freely available and supports four hours of functionality on silicon.

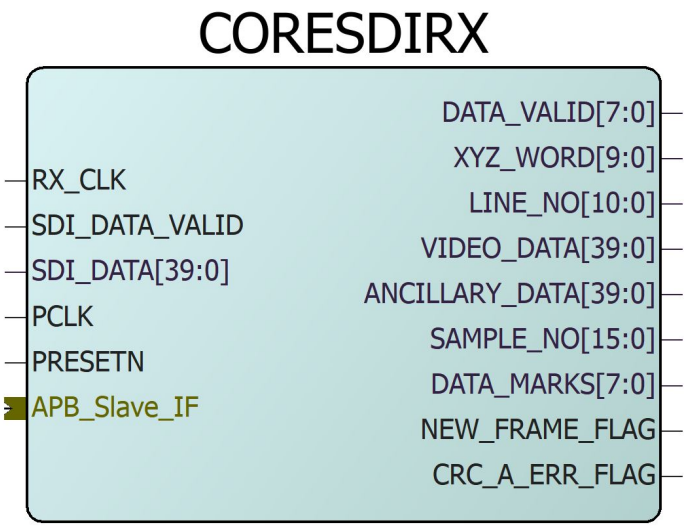
The Obfuscated version is license locked and will be available only with Libero Gold and Platinum Licenses.

8.2 SmartDesign

CoreSDIRX is pre-installed in the Libero SmartDesign IP deployment design environment or downloaded from the online repository. Figure 6 shows an example instantiated.

Note: Unless specified otherwise, this document uses the name Libero to identify Libero SoC PolarFire.

Figure 6 CoreSDIRX Instance View



The core can be configured using the configuration GUI within SmartDesign. An example of the GUI is shown in Figure 7.

Figure 7 Configuring CoreSDIRX in SmartDesign

Note: For information on using SmartDesign to instantiate and generate cores, refer to Libero User Guide.

8.3 Simulation in Libero

User testbench is provide along with CoreSDIRX.

To run user testbench simulations, select User Testbench flow in core configuration window. When SmartDesign generates the design files, it also generates the user testbench files. Set the design root to the CoreSDIRX instantiation in the Libero design hierarchy pane and click Simulation in the Libero Design Flow window. This invokes ModelSim and automatically runs the user testbench simulation.

8.4 Synthesis in Libero

To run synthesis on the CoreSDIRX, set the design root to the IP component instance and run the Synthesis tool from the Libero Design Flow pane.

8.5 Place-and-Route in Libero

After the design is synthesized, run the Place-And-Route tool from the Libero design flow pane.

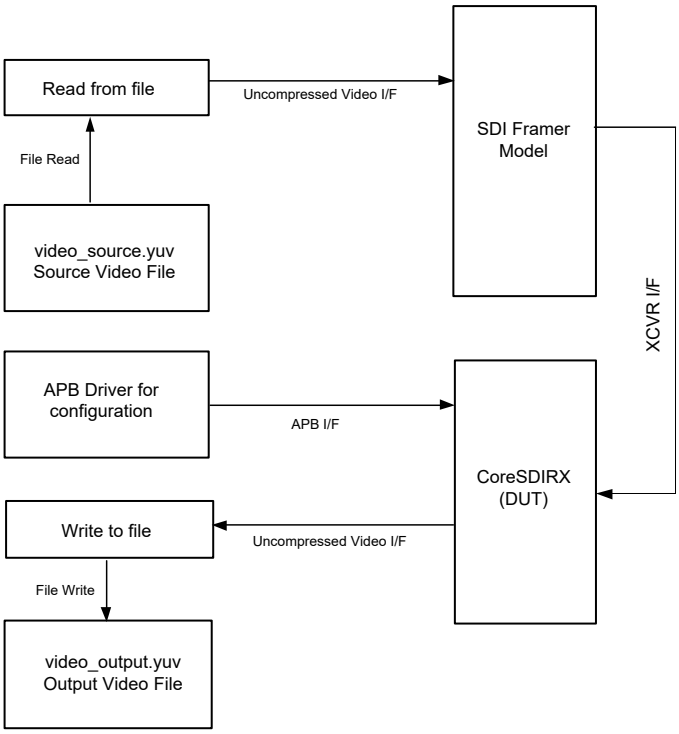
9 Testbench

A unified test-bench is used to verify and test CoreSDIRX called as user test-bench.

9.1 User Test-bench

A simplified block diagram of the user testbench is as shown in Figure 8.

Figure 8 CoreSDIRX User Test-bench



The source video file video_source.yuv is packaged in CoreSDIRX CPZ and is copied to simulation folder of Libero project when the core is instantiated and generated in Libero SmartDesign. The testbench has CoreSDIRX (DUT), SDI Framer model and video generator block.

The APB master model of the testbench writes to the Configuration registers of CoreSDIRX (DUT) and configures the core for required SDI mode. The SDI Framer model and the video generator are configured with the same configuration for which the DUT is configured. The video generator of the testbench reads the active video data from source video file and drives it on uncompressed video I/F of the SDI Framer Model along with other required timing reference and data valid signals. The SDI Framer Model generates the SDI frame from the uncompressed video. The SDI Framer Model drives the framed video on to the Transceiver interface.

The Transceiver I/F of SDI Framer Model is connected to the Transceiver I/F of DUT. DUT generates uncompressed video from the framed video. The uncompressed video is written to video_output.yuv file.

The source and the output video files are available in the simulation folder of the Libero project for comparison. The source and output video files with .yuv extension are in YUV422 8-bit colour format. These videos can be played on any commercial yuv video players.

In SD-SDI mode, the testbench generates color bar pattern and drives it on to the uncompressed video interface of the SDI Framer Model. The SDI output of SDI Framer Model is loop backed to SDI input of DUT. The output from DUT is written to video_output.yuv file. The testbench generated pattern is written to video_output_vg_sd.yuv file.

In SD-SDI mode, the video_output_vg_sd.yuv and the video_output.yuv files will be available in the simulation folder of the Libero project and are in UYVY 8-bit color format. These videos can be played on any commercial yuv video players.

10 System Integration

This section provides hints to ease the integration of CoreSDIRX.

10.1 CoreSDIRX System Integration for 3G/HD-SDI mode

Refer **Figure 9** for 3G/HD-SDI mode System Integration.

In this example design:

- Contains CoreSDIRX (SDI_Deframer_0) which is interfaced with MiV (MiV_Processor_0) soft processor and PF_XCVR (XCVR_SDI_0).
- Output pin “FABRIC_RESET_N” of CoreRESET_PF (RESET_PF_0) is used to drive MiV_Processor_0 reset pin “RESETN”. “FABRIC_RESET_N” and LANE0_TX_CLK_STABLE of XCVR_SDI_0 are used to drive PRESETN pin of both SDI_Framer_0 and SDI_Deframer_0.
- The SDI_Framer_0 has PCLK and TX_CLK clocks. SDI_Deframer_0 has PCLK and RX_CLK clocks.
- PCLK of both SDI_Framer_0 and SDI_Deframer_0 is an 80 MHz clock, driven from the output port “OUT0_FABCLK_0” of CCC_0.
- TX_CLK of SDI_Framer_0 is driven from LANE0_TX_CLK_R of XCVR_SDI_0 and RX_CLK of SDI_Deframer_0 is driven from LANE0_RX_CLK_R of XCVR_SDI_0.
- The LAN0_TX_CLK_R and LANE0_RX_CLK_R clock frequency depends on PF_XCVR configuration. 74.25 MHz for 3G mode and 37.125 MHz for HD mode. PF_XCVR can be configured through PFDRI_0.
- For 3G mode, PF_XCVR is configured with 2970 Mbps transceiver data rate, PMA mode with 40 bit @74.25 MHz. For HD mode, PF_XCVR is configured with 1485 Mbps transceiver data rate, PMA mode with 40 bit @37.125 MHz. In this example design, CDR reference clock frequency is 148.5 MHz.
- PF_XCVR_TXPLL_0 is configured with jitter cleaning mode.
- The SDI_Deframer_0 raw video data is looped back onto SDI_Framer_0 through CoreFIFO (FIFO_SDITX_RX_0).

10.2 CoreSDIRX System Integration for SD-SDI mode

Refer **Figure 10** for SD-SDI mode System Integration.

In this example design:

- Contains CoreSDIRX (SDI_Deframer_0) which is interfaced with MiV (MiV_Processor_0) soft processor and PF_XCVR (XCVR_top_0).
- Output pin “FABRIC_RESET_N” of CoreRESET_PF (RESET_PF_0) is used to drive MiV_Processor_0 reset pin “RESETN”.
- LANE0_RX_READY of XCVR_top_0, XCVR_INIT_DONE of DEV_INIT_0, PLL_LOCK_0 of CCC_Framer_0 and CCC_Deframer_0 are used to drive PRESETN pin of both SDI_Framer_0 and SDI_Deframer_0.
- The SDI_Framer_0 has PCLK and TX_CLK clocks. SDI_Deframer_0 has PCLK and RX_CLK clocks.
- PCLK of both SDI_Framer_0 and SDI_Deframer_0 is a 50 MHz clock, driven from the output port “OUT0_FABCLK_0” of CCC_Proc_0.
- RX_CLK of SDI_Deframer_0 is driven from “OUT0_FABCLK_0” of CCC_Deframer_0 which generates 6.75MHz and reference clock for the CCC_Deframer_0 is connected to LANE0_RX_CLK_R of XCVR_top_0.
- TX_CLK of SDI_Framer_0 is driven from “OUT0_FABCLK_0” of CCC_Framer_0 which generates 6.75MHz and reference clock for the CCC_Framer_0 is connected to LANE0_TX_CLK_R of XCVR_top_0.
- The LAN0_TX_CLK_R and LANE0_RX_CLK_R is working at 67.5 MHz for SD mode. Here the XCVR_top_0 is configured with 2700Mbps transceiver data rate, PMA mode with 40bit@ 67.5 MHz and the CDR mode is lock to reference. CDR reference clock frequency is 135 MHz. XCVR_top_0 is operated at 10 times oversampling rate.
- The TRANSITION_DETECTOR_0 and DOWNSAMPLER_0 modules are used to interface between XCVR_top_0 and SDI_Deframer_0. TRANSITION_DETECTOR_0 aligns the oversampled data from XCVR_top_0 to the bit boundary. DOWNSAMPLER_0 performs the 10x down sampling on the aligned data from TRANSITION_DETECTOR_0. The down-sampled data is provided to sdi input data of SDI_DEFRAMER_0.
- The UPSAMPLER_0 module is used to interface between SDI_Framer_0 and XCVR_top_0. UPSAMPLER_0 performs the 10x up-sampling on the sdi data output from SDI_Framer_0.
- The SDI_Deframer_0 raw video data is looped back onto SDI_Framer_0 through CoreFIFO (FIFO_SDI_RX_TO_TX_0).

Run the Libero flow with enabling the Timing Driven, High Effort Layout and Repair Minimum Delay Violations. The example design can be obtained from the Microsemi technical support team.

Figure 9 CoreSDIRX System Integration for 3G/HD-SDI mode

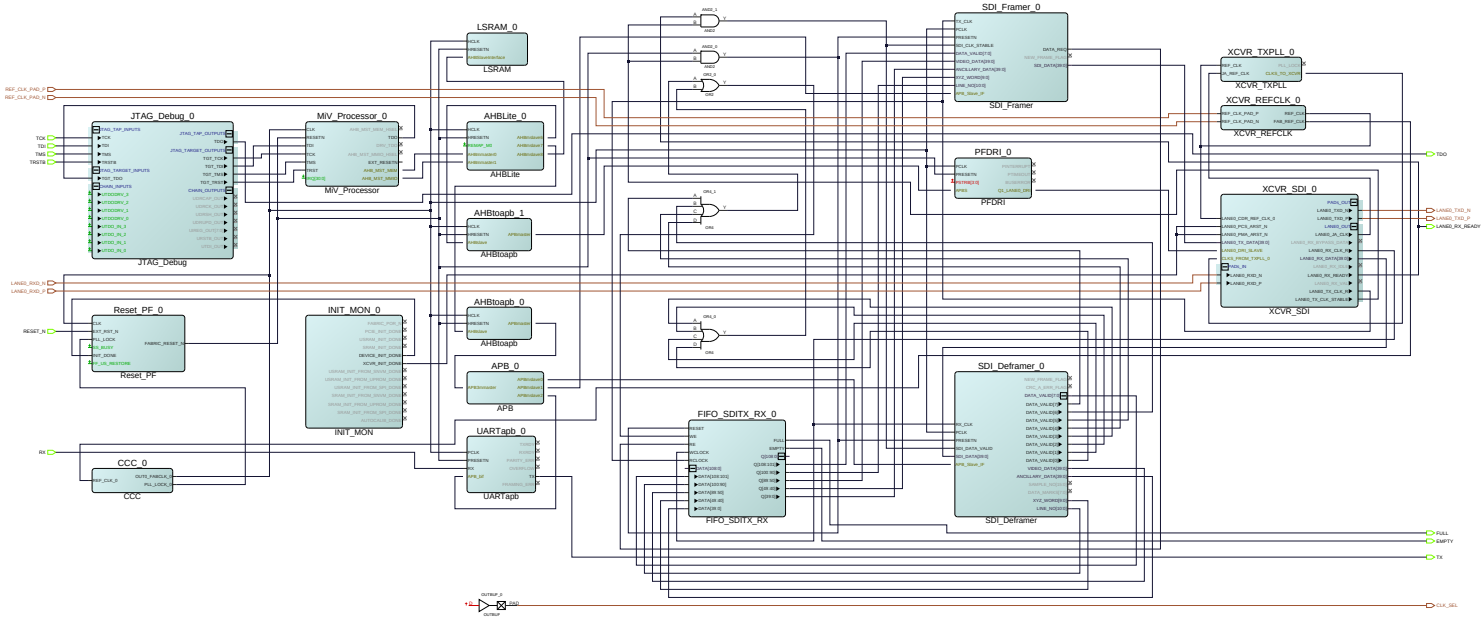
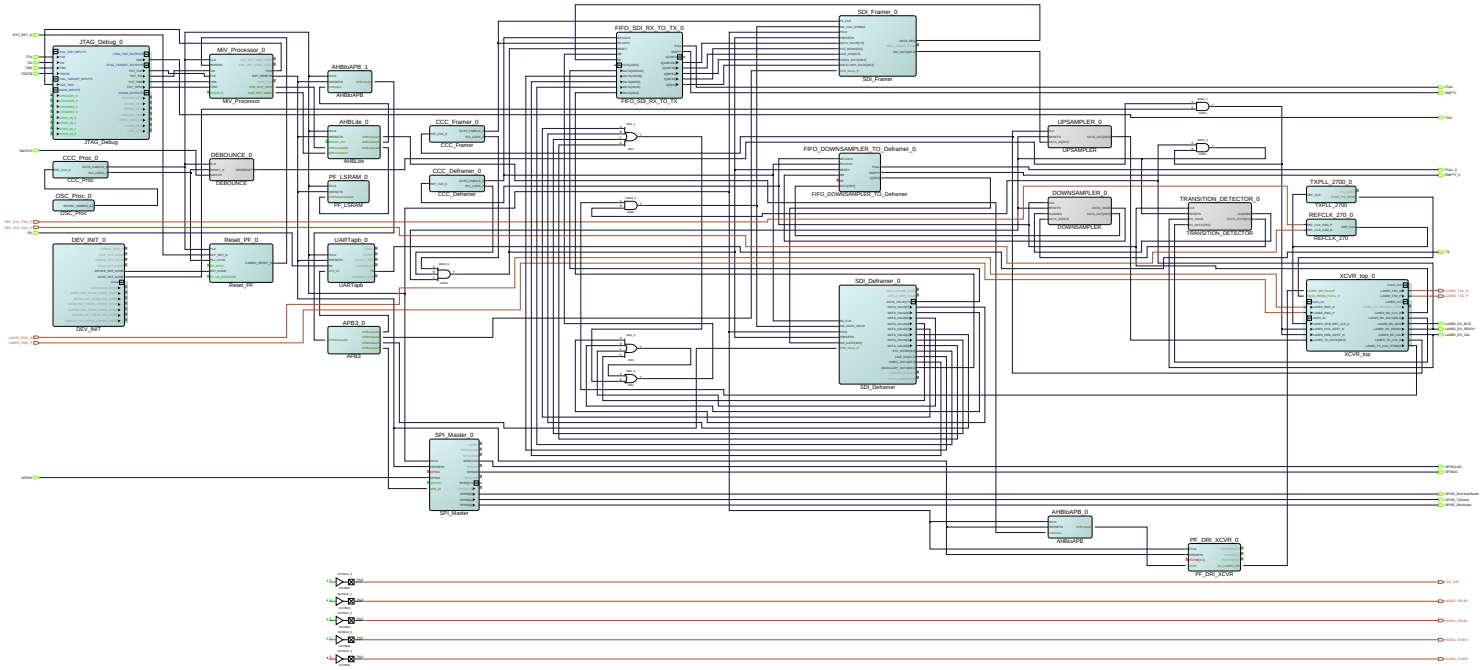


Figure 10 CoreSDIRX System Integration for SD-SDI mode



a MICROCHIP company

Microsemi Headquarters
One Enterprise, Aliso Viejo, CA 92656 USA
Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996
email: sales.support@microsemi.com
www.microsemi.com

Microsemi, a wholly owned subsidiary of Microchip Technology Inc. (Nasdaq: MCHP), offers a comprehensive portfolio of semiconductor and system solutions for aerospace & defense, communications, data center and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; enterprise storage and communication solutions, security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services.
Learn more at www.microsemi.com.