Hardware User Guide
UG-214

One Technology Way • P.O. Box 9106 • Norwood, MA 02062-9106, U.S.A. • Tel: 781.329.4700 • Fax: 781.461.3113 • www.analog.com

ADV7842 Fast Switching 2:1 HDMI 1.4 Receiver with 3D-Comb Decoder and Digitizer

This hardware user guide documents the functionality and features of the ADV7842. The ADV7842 is a dual HDMI® fast switching receiver with a 12-bit, 170 MHz video and graphics digitizer and a 3D comb filter decoder.

This user guide consists of sixteen chapters and a set of appendices.

TABLE OF CONTENTS

REVISION HISTORY

1/11—Revision 0: Initial Version

1. INTRODUCTION TO THE ADV7842 HARDWARE MANUAL

1.1 DESCRIPTION OF THE HARDWARE MANUAL

This manual provides a detailed description of the functionality and features supported by the ADV7842.

1.2 DISCLAIMER

The information contained in this document is proprietary of Analog Devices Inc. (ADI). This document must not be made available to anybody other than the intended recipient without the written permission of ADI. The content of this document is believed to be correct. If any errors are found within this document or if clarification is needed, contact the authors at ATV_VIDEORX_Apps@analog.com.

1.3 NUMBER NOTATIONS

Notation	Description
bit N	Bits are numbered in little endian format, that is, the least significant bit of a number is referred to as bit 0
V[X:Y]	Bit field representation covering bit X to Y of a value or a field V
0xNN	Hexadecimal (base-16) numbers are preceded by the prefix '0x'
0bNN	Binary (base-2) numbers are preceded by the prefix '0b'
NN	Decimal (base-10) are represented using no additional prefixes or suffixes

1.4 REGISTER ACCESS CONVENTIONS

Mode	Description
R/W	Memory location has read and write access.
R	Memory location is read access only. A read always returns 0 unless specified otherwise.
W	Memory location is write access only.

1.5 ACRONYMS AND ABBREVIATIONS

Acronym/Abbreviation	Description
ACP	Audio Content Protection
ADC	Analog to Digital Converter
AFE	Analog Front End
AGC	Automatic Gain Control
Ainfo	HDCP register. Refer to HDCP documentation.
AKSV	HDCP Transmitter Key Selection Vector. Refer to HDCP documentation.
An	64-bit pseudo-random value generated by HDCP Cipher function of Device A
AP	Audio Output Pin
AVI	Auxiliary Video Information
Aux	Auxiliary
Bcaps	HDCP register. Refer to HDCP documentation.
BGA	Ball Grid Array
BKSV	HDCP Receiver Key Selection Vector. Refer to HDCP documentation.
CP	Component Processor
CSC	Color Space Converter/Conversion
CSync	Composite Synchronization
CTI	Chroma Transient Improvement
DCM	Decimation
DDR	Double Data Rate

Acronym/Abbreviation	Description
DDFS	Direct Digital Frequency Synthesizer
DE	Data Enable
DID	Data Identification Word
DLL	Delay Locked Loop
DNR	Digital Noise Reduction
DPP	Data Preprocessor
DUT	Device Under Test (designate the ADV7842 unless stated otherwise)
DVI	Digital Visual Interface
EAV	End of Active Video
ED	Enhanced Definition
EMC	Electromagnetic Compatibility
EQ	Equalizer
HD	High Definition
HDCP	High Bandwidth Digital Content Protection
HDMI	High Bandwidth Multimedia Interface
HDTV	High Definition Television
HPA	Hot Plug Assert
HPD	Hot Plug Detect
HQI	High Quality Input
HSync	Horizontal Synchronization
IC	Integrated Circuit
ISRC	International Standard Recording Code
I^2S	Inter IC Sound
I^2C	Inter Integrated Circuit
KSV	Key Selection Vector
LLC	Line Locked Clock
LSB	Least Significant Bit
L-PCM	Linear Pulse Coded Modulated
Mbps	Megabit per Second
MPEG	Moving Picture Expert Group
Ms	Millisecond
MSB	Most Significant Bit
NC	No Connect
OTP	One Time Programmable
PAR	Parallel
Pj'	HDCP Enhanced Link Verification Response. Refer to HDCP documentation.
Ri'	HDCP Link verification response. Refer to HDCP documentation.
Rx	Receiver
SA	Slave Address
SAV	Start of Active Video
SD	Standard Definition
SDP	Standard Definition Processor
SDR	Single Data Rate
SHA-1	Refer to HDCP documentation.
SMPTE	Society of Motion Picture and Television Engineers
SNR	Signal to Noise Ratio
SOG	Sync on Green
SOY	Sync on Y
SPA	Source Physical Address
SPD	Source Production Descriptor
SSPD	Synchronization Source Polarity Detector
STDI	Standard Identification
TBC	Timebase Correction

Acronym/Abbreviation	Description
TMDS	Transition Minimized Differential Signaling
Tx	Transmitter
US	Up Sampling
VBI	Video Blanking Interval
VDP	VBI Data Processor
VSync	Vertical Synchronization
XTAL	Crystal Oscillator

1.6 FIELD FUNCTION DESCRIPTION

The function of a field is described in a table preceded by the bit name, a short function description, the I²C map, the register location within the I²C map, and a detailed description of the field.

The detailed description consists of:

- The values the field can take for a readable field
- The values the field can be set to for a writable field
-

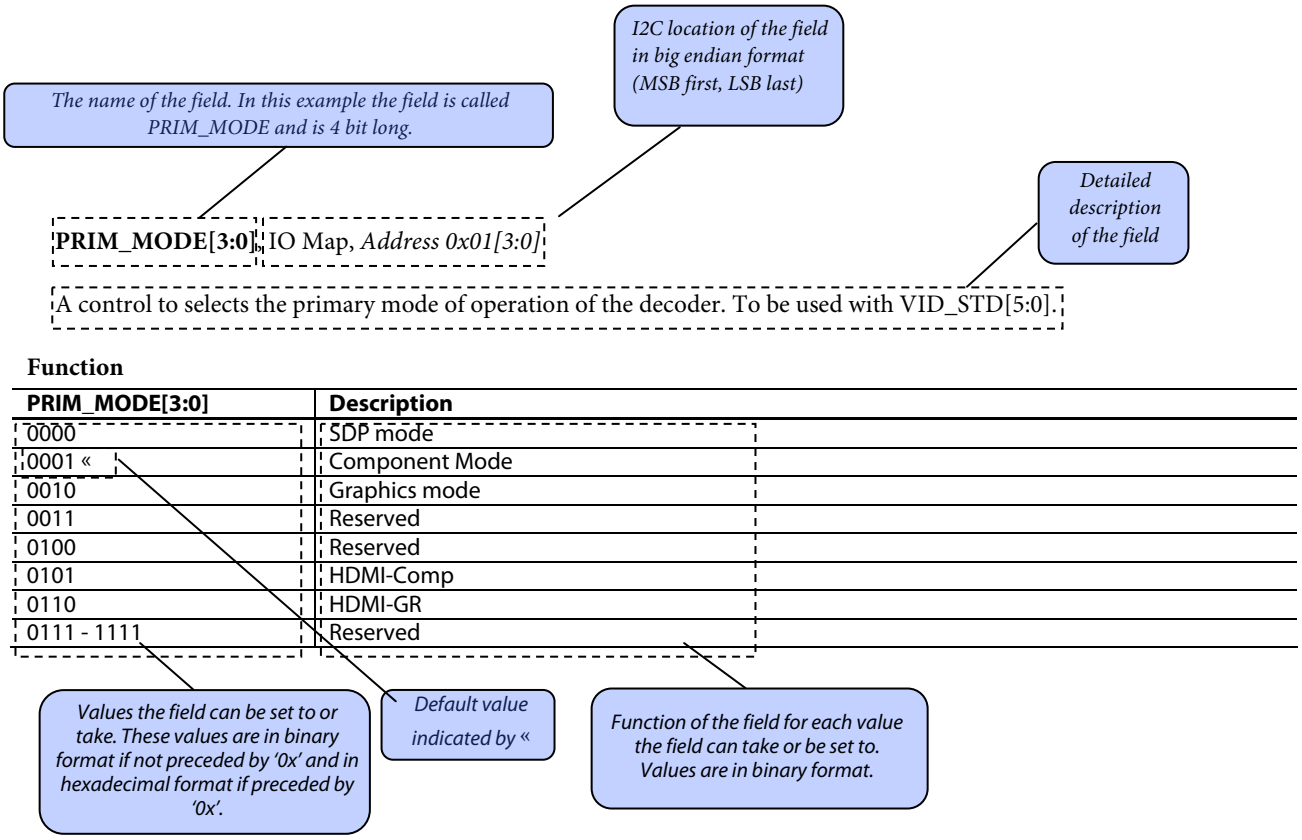


Figure 1: Field Description Format

1.7 REFERENCES

HDMI Licensing and LLC, High-Definition Multimedia Interface, Revision 1.4, June 5, 2009

Digital Content Protection (DCP) LLC, High-bandwidth Digital Content Protection System, Revision 1.3, December 21, 2006

CEA, CEA-861-D, A DTV Profile for Uncompressed High Speed Digital Interfaces, Revision D, July 18, 2006

ITU, ITU-R BT.656-4, Interface for Digital Component Video Signals in 525-Line and 625-Line Television Systems Operating at the 4:2:2 Level of Recommendation ITU-R BT.601, February 1998

ITU, ITU-R BT.601-5 Studio encoding parameters of digital television for standard 4:3 and widescreen 16:9 aspect ratios, December 1995

ITU, ITU-R BT.709-5 Parameter values for the HDTV standards for production and international programme exchange, April 2002

CENELEC, EN 50157, Part 1, Domestic and similar electronic equipment interconnection requirements: AV.link

CENELEC, EN 50157, Part 2-1, Domestic and similar electronic equipment interconnection requirements: AV.link

CENELEC, EN 50157, Part 2-2, Domestic and similar electronic equipment interconnection requirements: AV.link

CENELEC, EN 50157, Part 2-3, Domestic and similar electronic equipment interconnection requirements: AV.link

2 INTRODUCTION

The ADV7842 is a high quality, single chip, multiformat video decoder graphics digitizer with an integrated 2:1 multiplexed High-Definition Multimedia Interface (HDMI™) receiver. The multiformat 3D comb filter decoder supports the conversion of PAL, NTSC, and SECAM standards in the form of a composite or an S-Video input signal into a digital ITU-R BT.656 format. SCART and overlay functionality are enabled by the ability of the ADV7842 to process simultaneously CVBS and standard definition RGB signals.

The ADV7842 contains one main component processor (CP), which processes YPrPb and RGB component formats, including RGB graphics. The CP also processes the video signals from the HDMI receiver. The ADV7842 can operate in dual HDMI and analog input mode, thus allowing for fast switching between the ADCs and the HDMI.

The ADV7842 supports the decoding of a component RGB/YPrPb video signal into a digital YCrCb or RGB pixel output stream. The support for component video includes 525i, 625i, 525p, 625p, 720p, 1080i, 1080p, and 1250i standards, as well as many other SMPTE and HD standards. The ADV7842 supports graphics digitization. The ADV7842 is capable of digitizing RGB graphics signals from VGA to UXGA rates and converting them into a digital RGB or YCrCb pixel output stream. Internal EDID is available for one graphic port.

The ADV7842 incorporates a dual input HDMI 1.4-compatible receiver that supports all HDTV formats up to 1080p and display resolutions up to UXGA (1600 × 1200 at 60 Hz).

The ADV7842 supports full HDCP de-encryption with internal key storage. The ADV7842 features HDCP authentication, sync measurement and status monitoring is available for all non selected HDMI ports. These features allow for extremely fast switching between ports. Each HDMI port has dedicated +5V Detect and Hot Plug Assert pin. The HDMI receiver also includes an integrated equalizer that ensures robust operation of the interface. The HDMI receiver has advanced audio functionality, such as a mute controller, that prevents audible extraneous noise in the audio output. In addition the HDMI receiver incorporates a CEC controller and internal EDID support. Both features can also be made available in power saving modes.

Fabricated in an advanced CMOS process, the ADV7842 is provided in a 17 mm × 17 mm, 256-ball, BGA, surface-mount, RoHS-compliant package and is specified over the -10°C to +70°C temperature range.

2.1 ANALOG FRONT END

The ADV7842 analog front end comprises of four 170 MHz, 12-bit ADCs that digitize the analog video signal before applying it to the SDP or CP. The analog front end uses differential channels to each ADC to ensure high performance in a mixed-signal application.

The front end also includes a 12-channel input mux that enables multiple video signals to be applied to the ADV7842 without the requirement of an external mux. Current and voltage clamp control loops ensure that any dc offsets are removed from the video signal. The clamps are positioned in front of each ADC to ensure that the video signal remains within the range of the converter.

The ADCs are configured to run up to 8× oversampling mode when decoding composite or S-Video inputs. For component 525i, 625i, 525p, and 625p sources, 2× oversampling is performed. All other video standards are 1× oversampled. Oversampling the video signals reduces the cost and complexity of external anti aliasing filters with the benefit of an increased signal to noise ratio (SNR).

Optional internal anti aliasing filters with programmable bandwidth are positioned in front of each ADC. These filters can be used to band-limit video signals, removing spurious and out-of-band noise.

The ADV7842 can support the simultaneous processing of CVBS and RGB standard definition signals to enable SCART compatibility and overlay functionality. A combination of CVBS and RGB inputs can be mixed and the output is under the control of I^2C registers and the fast blank pin.

2.2 STANDARD DEFINITION PROCESSOR

The standard definition processor (SDP) is capable of decoding a large selection of baseband video signals in composite and S-Video formats. The video standards supported by the SDP include PAL, PAL 60, PAL M, PAL N, PAL Nc, NTSC M/J, NTSC 4.43, and SECAM. The ADV7842 can automatically detect the video standard and process it accordingly.

The SDP has a 3D temporal comb filter and a 5-line adaptive 2D comb filter that gives superior chrominance and luminance separation when decoding a composite video signal. This highly adaptive filter automatically adjusts its processing mode according to the video standard and signal quality with no user intervention required. The SDP has an IF filter block that compensates for attenuation in the high frequency chroma spectrum due to a tuner saw filter. The SDP has specific luminance and chrominance parameter controls for brightness, contrast, saturation, and hue.

The ADV7842 implements a patented adaptive digital line length tracking (ADLLT) algorithm to track varying video line lengths from sources such as a VCR. ADLLT enables the ADV7842 to track and decode poor quality video sources (such as VCRs) and noisy sources (such as tuner outputs). Frame TBC ensures stable clock synchronization between the decoder and the downstream devices.

The SDP also contains both a luma transient improvement (LTI) and a chroma transient improvement (CTI) processor. This processor increases the edge rate on the luma and chroma transitions, resulting in a sharper video image. The SDP has a Rovi® detection circuit, which allows Type I, Type II, and Type III Rovi protection levels. The decoder is also fully robust to all Rovi signal inputs.

2.3 HDMI RECEIVER

The HDMI receiver on the ADV7842 incorporates a fast switching feature that allows inactive ports to be authenticated for seamless switching between encrypted HDMI sources. The ADV7842 HDMI receiver provides active equalization of the HDMI data signals. This equalization compensates for the high frequency losses inherent in HDMI and DVI cabling, especially at longer lengths and higher frequencies. The equalizer is capable of equalizing for cable lengths up to 30 meters to achieve robust receiver performance at high HDMI data rates.

With the inclusion of HDCP, displays can receive encrypted video content. The HDMI interface of the ADV7842 allows for authentication of a video receiver, decryption of encoded data at the receiver, and renewability of that authentication during transmission, as specified by the HDCP v1.3 protocol for active and background HDMI ports.

The ADV7842 supports also 3D Video – such as packing for all 3D formats up to a 225 MHz TMDS clock or up to a pixel clock to 148.5 MHz. The ADV7842 supports full colorimetry including SYCC601, Adobe RGB and Adobe YCC601.

The HDMI receiver offers advanced audio functionality. The receiver contains an audio mute controller, which can detect a variety of conditions that could result in audible extraneous noise in the audio output. Upon detection of these conditions, the audio data can be muted to prevent audio clicks or pops.

2.4 COMPONENT PROCESSOR

The CP section is capable of decoding a wide range of component video formats in any color space. Component video standards supported by the CP are 525i, 625i, 525p, 625p, 720p, 1080i, 1080p, 1250i, VGA up to UXGA at 60 Hz, and many other standards.

The output section of the CP is highly flexible. It can be configured in Single Data Rate (SDR) mode with one data packet per clock cycle or in a Double Data Rate (DDR) mode where data is presented on the rising and falling edge of the clock. In SDR mode, a 16/20/24-bit 4:2:2 or 24/30/36-bit 4:4:4 output is possible. In these modes, HS/CS, VS/FIELD, and FIELD/DE (where applicable) timing reference signals are provided. In DDR mode, the ADV7842 can be configured in an 8/10/12-bit 4:2:2 YCrCb or 12-bit 4:4:4 RGB/YCrCb pixel output interface with corresponding timing signals.

The CP section contains circuitry to enable the detection of Rovi encoded YPrPb signals for 525i, 625i, 525p, and 625p. It is designed to be fully robust when decoding these types of signals.

VBI extraction of CGMS data is performed by the VDP section of the ADV7842 for interlaced, progressive, and high definition scanning rates. The data extracted can be read back over the I^2C interface or ancillary data stream.

2.5 MAIN FEATURES OF ADV7842

2.5.1 Analog Front End

The analog front-end functionality includes:

- 170 MHz 12-bit ADCs enabling true 12-bit video decoding
- 12 analog input channel mux enabling multisource connection without the requirement of an external mux
- Voltage clamp control loops ensuring any DC offsets are removed from the video signal
- Digital PLL design delivering ultra low sampling jitter for digitizer

2.5.2 HDMI Receiver

- HDMI 1.4 compatible receiver
    - 3D Video Support including Frame packing for 3D formats up to a 225 MHz TMDS clock and/or up a pixel clock of 148.5 MHz
    - Full colorimetry support including SYCC601, Adobe RGB, Adobe YCC601
    - Advanced Audio Features
- HDCP v1.3 compliant receiver
- Fast switching between HDMI ports
- Supports Deep Color
- Supports all display resolutions up to UXGA (1600 x 1200 at 60 Hz)
- HBR, DSD, and PCM formats are supported with sampling frequency up to 192 kHz
- Programmable front end equalization for HDMI operation over cable lengths up to 30 meters
- Audio mute for removing extraneous noises
- Programmable interrupt generator to detect HDMI packets
- Internal EDID support

2.5.3 Composite and S-Video Processing

- Advanced adaptive 3D comb with concurrent Frame-based Timebase Correction (using either external DDR or SDR SDRAM memory)
- Adaptive 2D 5-line comb filters for NTSC and PAL that give superior chrominance and luminance separation for composite video
- Full automatic detection and autoswitching of all worldwide standards (PAL, NTSC, and SECAM)
- Automatic gain control with white peak mode that ensures the video is always processed without loss of the video processing range
- Proprietary architecture for locking to weak, noisy, and unstable sources from VCRs and tuners
- IF filter block that compensates for high frequency luma attenuation due to tuner saw filter
- LTI and CTI for PAL, NTSC and SECAM
- Simultaneous CVBS and HDMI Audio Processing
- Vertical and horizontal programmable luma peaking filters
- True full 12-bit deep color processing path from front to back end in 4:4:4/4:2:2 RGB/YCrCb formats

- 8× oversampling (108 MHz) for CVBS, and S-Video modes
- Line-locked clock output (LLC)
- Free run output mode that provides stable timing when no video input is present
- Internal color bar test pattern
- Advanced TBC with frame synchronization, which ensures nominal clock and data for nonstandard input
- Interlace-to-progressive conversion for 525i and 625i formats
- Color controls that include hue, brightness, saturation, and contrast

2.5.4 Component Video Processing

- Formats supported include 525i, 625i, 525p, 625p, 720p, 1080i, 1080p
- Automatic adjustments for gain (contrast) and offset (brightness); manual adjustment controls are also supported
- Support for analog component YPrPb/RGB video formats with embedded synchronization or with separate HSync, VS, or CS
- An any-to-any 3 × 3 CSC matrix support YCrCb to RGB and RGB to YCrCb.
- Provides color controls such as saturation, brightness, hue, and contrast
- Two standard identification (STDI) blocks that enable dual system component format detection
- Two synchronization source polarity detectors (SSPD) determine the source and polarity of the synchronization signals that accompany video inputs
- Certified Rovi copy protection detection on component formats (525i, 625i, 525p, and 625p)
- Free run output mode provides stable timing when no video input is present
- Arbitrary pixel sampling support for nonstandard video sources
- Autographic mode allows support for an extended selection of standards

2.5.5 RGB Graphics Processing

- 170 MHz conversion rate supports RGB input resolutions up to 1600 × 1200 at 60 Hz (UXGA)
- Automatic gain controls for graphics modes
- Contrast, brightness, saturation, and hue controls
- 64-phase Delay Locked Loop (DLL) allows optimum pixel clock sampling
- Automatic detection of synchronization source and polarity by the SSPD block
- Standard identification is enabled by either of the available STDI blocks
- RGB can be color space converted to YCrCb and decimated to a 4:2:2 format for video centric back-end IC interfacing
- Data enable (DE) output signal supplied for direct connection to HDMI/DVI transmitter IC
- Arbitrary pixel sampling support for nonstandard video sources
- Autographic mode allows support for an extended selection of standards

2.5.6 Video Output Formats

- Pseudo DDR (CCIR-656 type stream) 8/10/12-bit 4:2:2 YCrCb for 525i, 625i, 525P, and 625P
- SDR 16/20/24-bit 4:2:2 YCrCb for all standards
- SDR 24/30/36-bit 4:4:4 YCrCb/RGB for all CP and HDMI standards
- Double data rate (DDR) 8/10/12-bit 4:2:2 YCrCb for all standards up to 54MHz
- DDR 8/10/12-bit 4:4:4 YCrCb/RGB for all CP and HDMI standards up to 54MHz

2.5.7 Additional Features

- HS, VS, FIELD, CS and DE output signals with programmable position, polarity, and width
- Two interrupt request output pins, INT1, INT2
- Temperature range : -10°C to +70°C
- 17 mm x 17 mm, Pb-free BGA package

2.6 FUNCTIONAL BLOCK DIAGRAM

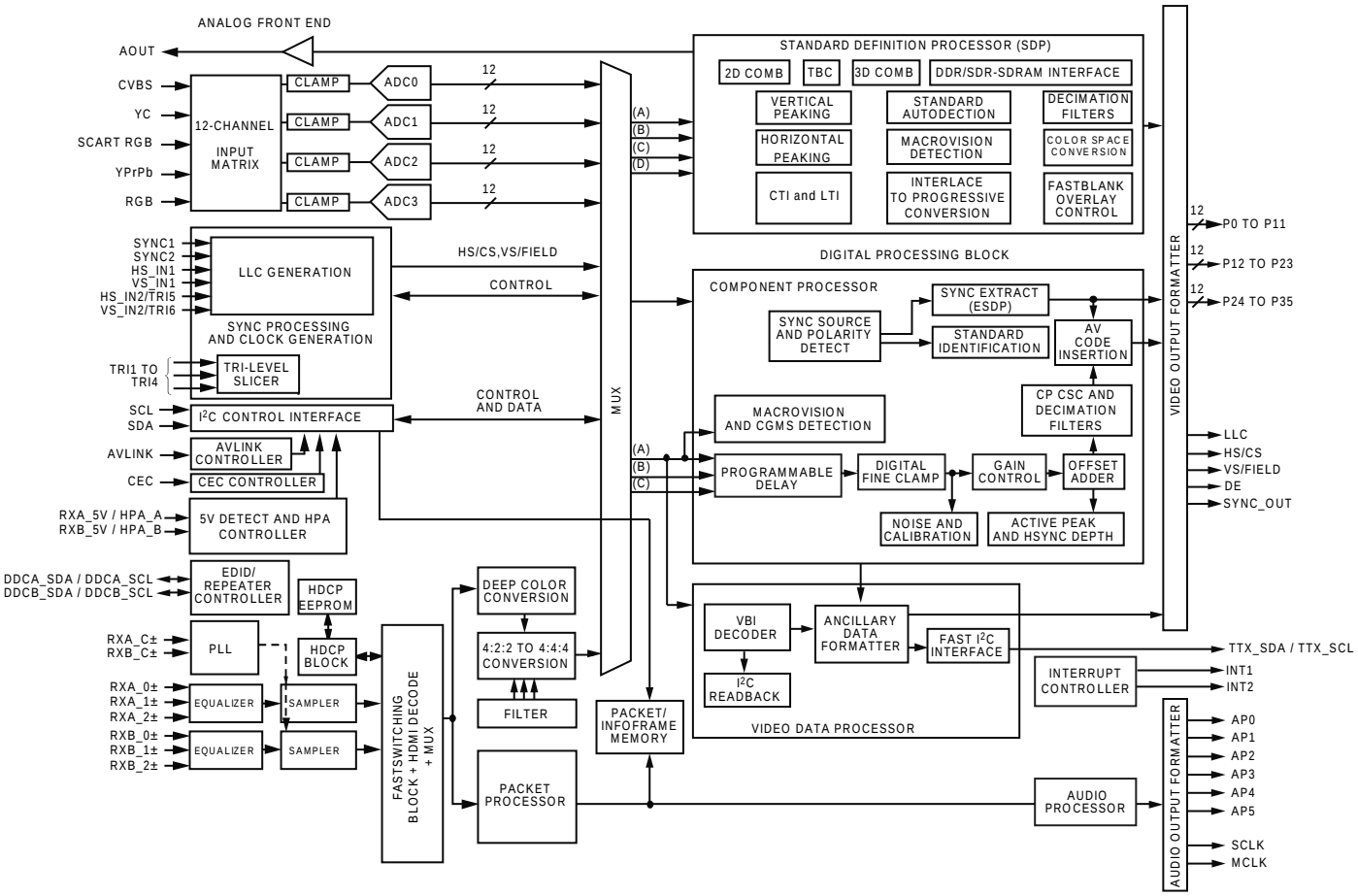


Figure 2: Functional Block Diagram

2.7 **PIN DESCRIPTION**

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
A	GND	P3	P1	P0	TVDD	RXB_2-	RXB_1-	RXB_0-	RXB_C-	GND	RXA_2-	RXA_1-	RXA_0-	RXA_C-	TVDD	GND	A
B	P5	P4	P2	SYNC_OUT	TVDD	RXB_2+	RXB_1+	RXB_0+	RXB_C+	GND	RXA_2+	RXA_1+	RXA_0+	RXA_C+	TVDD	XTALP	B
C	P7	P6	VS/FIELD	HS/CS	GND	HPA_B	HPA_A	RXB_5V	RXA_5V	$\overline{\text{PWRDN1}}$	TEST8	CVDD	CVDD	CVDD	GND	XTALN	C
D	P9	P8	EP_MISO	FIELD/DE	GND	CEC	DDCB_SDA	DDCB_SCL	DDCA_SDA	DDCA_SCL	RTERM	CVDD	VGA_SCL	VGA_SDA	PVDD	PVDD	D
E	GND	P10	$\overline{\text{EP\_CS}}$	EP_MOSI	GND	GND	GND	GND	GND	GND	CVDD	CVDD	HS_IN2/TRI7	VS_IN2/TRI8	REFP	REFN	E
F	P12	P11	EP_SCK	TTX_SCL	DVDDIO	GND	GND	GND	GND	GND	GND	AVDD	AIN12	AIN11	TRI4	TRI3	F
G	P14	P13	TTX_SDA	MCLK	DVDDIO	GND	GND	GND	GND	GND	GND	AVDD	AIN10	SYNC4	AIN9	AIN8	G
H	P16	P15	AP0	AP5	DVDDIO	GND	GND	GND	GND	GND	GND	AVDD	TRI1	TRI2	AIN7	SYNC3	H
J	P18	P17	SCLK	AP4	DVDDIO	GND	VDD	GND	GND	GND	GND	AVDD	AIN6	AIN4	SYNC2	GND	J
K	P20	P19	AP3	AP1	DVDDIO	VDD	VDD	VDD	VDD	GND	GND	AVDD	AIN5	VS_IN1/TRI6	AIN2	AIN3	K
L	P22	P21	SCL	AP2	DVDDIO	VDD	VDD	VDD	VDD	GND	GND	AVDD	AOUT	HS_IN1/TRI5	AIN1	SYNC1	L
M	GND	P23	SDA	INT1	DVDDIO	VDD_SDRAM	VDD_SDRAM	VDD_SDRAM	VDD_SDRAM	VDD_SDRAM	GND	GND	GND	GND	GND	GND	M
N	LLC	P24	INT2	TEST4	$\overline{\text{RESET}}$	TEST6	SDRAM_A8	SDRAM_A4	SDRAM_A0	$\overline{\text{SDRAM\_CS}}$	SDRAM_LDQS	SDRAM_DQ4	SDRAM_DQ15	SDRAM_DQ11	SDRAM_CK	SDRAM_CKE	N
P	P25	P26	TEST5	AVLINK	TEST7	SDRAM_A11	SDRAM_A7	SDRAM_A3	SDRAM_A10	$\overline{\text{SDRAM\_RAS}}$	SDRAM_DQ7	SDRAM_DQ3	SDRAM_VREF	SDRAM_DQ12	SDRAM_UDQS	$\overline{\text{SDRAM\_CK}}$	P
R	P27	P28	P30	P32	P34	SDRAM_A9	SDRAM_A6	SDRAM_A2	SDRAM_BA1	$\overline{\text{SDRAM\_CAS}}$	SDRAM_DQ6	SDRAM_DQ2	SDRAM_DQ0	SDRAM_DQ13	SDRAM_DQ9	SDRAM_DQ8	R
T	GND	P29	P31	P33	P35	GND	SDRAM_A5	SDRAM_A1	SDRAM_BA0	$\overline{\text{SDRAM\_WE}}$	SDRAM_DQ5	GND	SDRAM_DQ1	SDRAM_DQ14	SDRAM_DQ10	GND	T
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	

Figure 3: ADV7842 Pin Configuration (Top View)

Table 1: Function Descriptions

Pin No.	Mnemonic	Type	Description
A1	GND	Ground	Ground.
A2	P3	Digital video output	Video Pixel Output Port.
A3	P1	Digital video output	Video Pixel Output Port.
A4	P0	Digital video output	Video Pixel Output Port.
A5	TVDD	Power	Terminator Supply Voltage (3.3 V).
A6	RXB_2-	HDMI input	Digital Input Channel 2 Complement of Port B in the HDMI Interface.

Pin No.	Mnemonic	Type	Description
A7	RXB_1-	HDMI input	Digital Input Channel 1 Complement of Port B in the HDMI Interface.
A8	RXB_0-	HDMI input	Digital Input Channel 0 Complement of Port B in the HDMI Interface.
A9	RXB_C-	HDMI input	Digital Input Clock Complement of Port B in the HDMI Interface.
A10	GND	Ground	Ground.
A11	RXA_2-	HDMI input	Digital Input Channel 2 Complement of Port A in the HDMI Interface.
A12	RXA_1-	HDMI input	Digital Input Channel 1 Complement of Port A in the HDMI Interface.
A13	RXA_0-	HDMI input	Digital Input Channel 0 Complement of Port A in the HDMI Interface.
A14	RXA_C-	HDMI input	Digital Input Clock Complement of Port A in the HDMI Interface.
A15	TVDD	Power	Terminator Supply Voltage (3.3 V).
A16	GND	Ground	Ground.
B1	P5	Digital video output	Video Pixel Output Port.
B2	P4	Digital video output	Video Pixel Output Port.
B3	P2	Digital video output	Video Pixel Output Port.
B4	SYNC_OUT	Miscellaneous digital	Sliced Synchronization Output
B5	TVDD	Power	Terminator Supply Voltage (3.3 V).
B6	RXB_2+	HDMI input	Digital Input Channel 2 True of Port B in the HDMI Interface.
B7	RXB_1+	HDMI input	Digital Input Channel 1 True of Port B in the HDMI Interface.
B8	RXB_0+	HDMI input	Digital Input Channel 0 True of Port B in the HDMI Interface.
B9	RXB_C+	HDMI input	Digital Input Clock True of Port B in the HDMI Interface.
B10	GND	Ground	Ground.
B11	RXA_2+	HDMI input	Digital Input Channel 2 True of Port A in the HDMI Interface.
B12	RXA_1+	HDMI input	Digital Input Channel 1 True of Port A in the HDMI Interface.
B13	RXA_0+	HDMI input	Digital Input Channel 0 True of Port A in the HDMI Interface.
B14	RXA_C+	HDMI input	Digital Input Clock True of Port A in the HDMI Interface.
B15	TVDD	Power	Terminator Supply Voltage (3.3 V).
B16	XTALP		Input pin for 28.63636 MHz crystal or external 1.8V, 28.63636 MHz Clock Oscillator Source to Clock the ADV7842.
C1	P7	Digital video output	Video Pixel Output Port.
C2	P6	Digital video output	Video Pixel Output Port.
C3	VS/FIELD	Digital video output	Vertical Synchronization/Field Synchronization. VS is a vertical synchronization output signal in the CP and HDMI processor. FIELD is a field synchronization output signal in all interlaced video modes. VS or FIELD can be configured for this pin.
C4	HS/CS	Digital video output	Horizontal Synchronization/Composite Synchronization. HS is a horizontal synchronization output signal in the CP and HDMI processor. CS (composite synchronization) signal is a single signal containing both horizontal and vertical synchronization pulses. HS or CS can be configured for this pin.
C5	GND	Ground	Ground.
C6	HPA_B	Miscellaneous digital	Hot Plug Assert signal output for HDMI port B.
C7	HPA_A	Miscellaneous digital	Hot Plug Assert signal output for HDMI port A.
C8	RXB_5V	HDMI input	5 V Detect Pin for Port B in the HDMI Interface.
C9	RXA_5V	HDMI input	5 V Detect Pin for Port A in the HDMI Interface.
C10	PWRDN1	Miscellaneous digital	Controls the Power-Up of the ADV7842. Should be connected to a digital 3.3 V I/O supply to power up the ADV7842.
C11	Test8	Test pin	Tie to 3.3V via 4.7k resistor
C12	CVDD	Power	Comparator Supply Voltage (1.8 V).

Pin No.	Mnemonic	Type	Description
C13	CVDD	Power	Comparator Supply Voltage (1.8 V).
C14	CVDD	Power	Comparator Supply Voltage (1.8 V).
C15	GND	Ground	Ground.
C16	XTALN		Input Pin for 28.63636 MHz Crystal.
D1	P9	Digital video output	Video Pixel Output Port.
D2	P8	Digital video output	Video Pixel Output Port.
D3	EP_MISO	Digital input	SPI Master In/Slave Out for External EDID Interface.
D4	FIELD/DE	Miscellaneous digital	DE (data enable) is a signal that indicates active pixel data. FIELD is a field synchronization output signal in all interlaced video modes. DE or FIELD can be configured for this pin.
D5	GND	Ground	Ground.
D6	CEC	Digital input/output	Consumer Electronic Control Channel.
D7	DDCB_SDA	Digital input/output	HDCP Slave Serial Data Port B. DDCB_SDA is a 3.3 V input/output that is 5 V tolerant.
D8	DDCB_SCL	Digital input	HDCP Slave Serial Clock Port B. DDCB_SCL is a 3.3 V input that is 5 V tolerant.
D9	DDCA_SDA	Digital input/output	HDCP Slave Serial Data Port A. DDCA_SDA is a 3.3 V input/output that is 5 V tolerant.
D10	DDCA_SCL	Digital input	HDCP Slave Serial Clock Port A. DDCA_SCL is a 3.3 V input that is 5 V tolerant.
D11	RTERM	Miscellaneous analog	Sets Internal Termination Resistance. A 500Ω resistor between this pin and GND should be used.
D12	CVDD	Power	Comparator Supply Voltage (1.8 V).
D13	VGA_SCL	Miscellaneous digital	DDC Port Serial Clock Input for VGA
D14	VGA_SDA	Miscellaneous digital	DDC Port Serial Data Input/Output for VGA
D15	PVDD	Power	PLL Supply Voltage (1.8 V).
D16	PVDD	Power	PLL Supply Voltage (1.8 V).
E1	GND	Ground	Ground.
E2	P10	Digital video output	Video Pixel Output Port.
E3	$\overline{\text{EP\_CS}}$	Digital output	SPI Chip Select for External EDID Interface.
E4	EP_MOSI	Digital output	SPI Master Out/Slave In for External EDID Interface.
E5	GND	Ground	Ground.
E6	GND	Ground	Ground.
E7	GND	Ground	Ground.
E8	GND	Ground	Ground.
E9	GND	Ground	Ground.
E10	GND	Ground	Ground.
E11	CVDD	Power	Comparator Supply Voltage (1.8 V).
E12	CVDD	Power	Comparator Supply Voltage (1.8 V).
E13	HS_IN2/TRI7	Miscellaneous analog	HS on Graphics Port 2. The HS input signal is used for 5-wire timing mode. This pin can also be used as a trilevel/bilevel input on the SCART or D-terminal connector. (Selection available via I^2C.)
E14	VS_IN2/TRI8	Miscellaneous analog	VS on Graphics Port 2. The VS input signal is used for 5-wire timing mode. This pin can also be used as a trilevel/bilevel input on the SCART or D-terminal connector. (Selection available via I^2C.)
E15	REFP	Miscellaneous analog	Internal Voltage Reference Output.
E16	REFN	Miscellaneous analog	Internal Voltage Reference Output.
F1	P12	Digital video output	Video Pixel Output Port.

Pin No.	Mnemonic	Type	Description
F2	P11	Digital video output	Video Pixel Output Port.
F3	EP_SCK	Digital output	SPI Clock for External EDID Interface.
F4	TTX_SCL	Miscellaneous digital	Fast I^2C Interface for Teletext Data Extraction. TTX_SCL is used as the I^2C port serial clock input.
F5	DVDDIO	Power	Digital I/O Supply Voltage (3.3 V).
F6	GND	Ground	Ground.
F7	GND	Ground	Ground.
F8	GND	Ground	Ground.
F9	GND	Ground	Ground.
F10	GND	Ground	Ground.
F11	GND	Ground	Ground.
F12	AVDD	Power	Analog Supply Voltage (1.8 V).
F13	AIN12	Analog video input	Analog Video Input Channel.
F14	AIN11	Analog video input	Analog Video Input Channel.
F15	TRI4	Miscellaneous analog	Trilevel or Bilevel Input on the SCART or D-Type Connector. (Selection available via I^2C.)
F16	TRI3	Miscellaneous analog	Trilevel or Bilevel Input on the SCART or D-Type Connector. (Selection available via I^2C.)
G1	P14	Digital video output	Video Pixel Output Port.
G2	P13	Digital video output	Video Pixel Output Port.
G3	TTX_SDA	Miscellaneous digital	Fast I^2C Interface for Teletext Data Extraction. TTX_SDA is used as the I^2C port serial data input/output pins.
G4	MCLK	Miscellaneous	Audio Master Clock Output.
G5	DVDDIO	Power	Digital I/O Supply Voltage (3.3 V).
G6	GND	Ground	Ground.
G7	GND	Ground	Ground.
G8	GND	Ground	Ground.
G9	GND	Ground	Ground.
G10	GND	Ground	Ground.
G11	GND	Ground	Ground.
G12	AVDD	Power	Analog Supply Voltage (1.8 V).
G13	AIN10	Analog video input	Analog Video Input Channel.
G14	SYNC4	Miscellaneous analog	This is a synchronization on green or luma input (SOG/SOY) used in embedded synchronization mode. User configurable.
G15	AIN9	Analog video input	Analog Video Input Channel.
G16	AIN8	Analog video input	Analog Video Input Channel.
H1	P16	Digital video output	Video Pixel Output Port.
H2	P15	Digital video output	Video Pixel Output Port.
H3	AP0	Miscellaneous	Audio Output Pin.
H4	AP5	Miscellaneous	Audio Output Pin.
H5	DVDDIO	Power	Digital I/O Supply Voltage (3.3 V).
H6	GND	Ground	Ground.
H7	GND	Ground	Ground.
H8	GND	Ground	Ground.
H9	GND	Ground	Ground.
H10	GND	Ground	Ground.
H11	GND	Ground	Ground.
H12	AVDD	Power	Analog Supply Voltage (1.8 V).

Pin No.	Mnemonic	Type	Description
H13	TRI1	Miscellaneous analog	Trilevel or Bilevel Input on the SCART or D-Type Connector. (Selection available via I²C.)
H14	TRI2	Miscellaneous analog	Trilevel or Bilevel Input on the SCART or D-Type Connector. (Selection available via I²C.)
H15	AIN7	Analog video input	Analog Video Input Channel.
H16	SYNC3	Miscellaneous analog	This is a synchronization on green or luma input (SOG/SOY) used in embedded synchronization mode. User configurable.
J1	P18	Digital video output	Video Pixel Output Port.
J2	P17	Digital video output	Video Pixel Output Port.
J3	SCLK	Miscellaneous digital	Audio Serial Clock Output.
J4	AP4	Miscellaneous	Audio Output Pin.
J5	DVDDIO	Power	Digital I/O Supply Voltage (3.3 V).
J6	GND	Ground	Ground.
J7	VDD	Power	Digital Core Supply Voltage (1.8 V).
J8	GND	Ground	Ground.
J9	GND	Ground	Ground.
J10	GND	Ground	Ground.
J11	GND	Ground	Ground.
J12	AVDD	Power	Analog Supply Voltage (1.8 V).
J13	AIN6	Analog video input	Analog video input channel.
J14	AIN4	Analog video input	Analog Video Input Channel.
J15	SYNC2	Miscellaneous analog	This is a synchronization on green or luma input (SOG/SOY) used in embedded synchronization mode. User configurable.
J16	GND	Ground	Ground.
K1	P20	Digital video output	Video Pixel Output Port.
K2	P19	Digital video output	Video Pixel Output Port.
K3	AP3	Miscellaneous	Audio Output Pin.
K4	AP1	Miscellaneous	Audio Output Pin.
K5	DVDDIO	Power	Digital I/O Supply Voltage (3.3 V).
K6	VDD	Power	Digital Core Supply Voltage (1.8 V).
K7	VDD	Power	Digital Core Supply Voltage (1.8 V).
K8	VDD	Power	Digital Core Supply Voltage (1.8 V).
K9	VDD	Power	Digital Core Supply Voltage (1.8 V).
K10	GND	Ground	Ground.
K11	GND	Ground	Ground.
K12	AVDD	Power	Analog Supply Voltage (1.8 V).
K13	AIN5	Analog video input	Analog Video Input Channel.
K14	VS_IN1/TRI6	Miscellaneous analog	VS on Graphics Port 1. The VS input signal is used for 5-wire timing mode. This pin can also be used as a trilevel/bilevel input on the SCART or D-terminal connector. (Selection available via I²C.)
K15	AIN2	Analog video input	Analog Video Input Channel.
K16	AIN3	Analog video input	Analog Video Input Channel.
L1	P22	Digital video output	Video Pixel Output Port.
L2	P21	Digital video output	Video Pixel Output Port.
L3	SCL	Miscellaneous digital	I²C Port Serial Clock Input. SCL is the clock line for the control port.
L4	AP2	Miscellaneous	Audio Output Pin.

Pin No.	Mnemonic	Type	Description
L5	DVDDIO	Power	Digital I/O Supply Voltage (3.3 V).
L6	VDD	Power	Digital Core Supply Voltage (1.8 V).
L7	VDD	Power	Digital Core Supply Voltage (1.8 V).
L8	VDD	Power	Digital Core Supply Voltage (1.8 V).
L9	VDD	Power	Digital Core Supply Voltage (1.8 V).
L10	GND	Ground	Ground.
L11	GND	Ground	Ground.
L12	AVDD	Power	Analog Supply Voltage (1.8 V).
L13	AOUT	Analog monitor output	Analog Monitor Output.
L14	HS_IN1/TRI5	Miscellaneous analog	HS on Graphics Port 1. The HS input signal is used for 5-wire timing mode. This pin can also be used as a trilevel/bilevel input on the SCART or D-terminal connector. (Selection available via I²C.)
L15	AIN1	Analog video input	Analog Video Input Channel.
L16	SYNC1	Miscellaneous analog	This is a synchronization on green or luma input (SOG/SOY) used in embedded synchronization mode. User configurable.
M1	GND	Ground	Ground.
M2	P23	Digital video output	Video Pixel Output Port.
M3	SDA	Miscellaneous digital	I²C Port Serial Data Input/Output Pin. SDA is the data line for the control port.
M4	INT1	Miscellaneous digital	Interrupt. This pin can be active low or active high. When status bits change, this pin is triggered. The events that trigger an interrupt are under user control.
M5	DVDDIO	Power	Digital I/O Supply Voltage (3.3 V).
M6	VDD_SDRAM	Power	External Memory Interface Digital Input/Output Supply (DDR 2.5 V or SDR 3.3 V).
M7	VDD_SDRAM	Power	External Memory Interface Digital Input/Output Supply (DDR 2.5 V or SDR 3.3 V).
M8	VDD_SDRAM	Power	External Memory Interface Digital Input/Output Supply (DDR 2.5 V or SDR 3.3 V).
M9	VDD_SDRAM	Power	External Memory Interface Digital Input/Output Supply (DDR 2.5 V or SDR 3.3 V).
M10	VDD_SDRAM	Power	External Memory Interface Digital Input/Output Supply (DDR 2.5 V or SDR 3.3 V).
M11	GND	Ground	Ground.
M12	GND	Ground	Ground.
M13	GND	Ground	Ground.
M14	GND	Ground	Ground.
M15	GND	Ground	Ground.
M16	GND	Ground	Ground.
N1	LLC	Digital video output	Line-Locked Output Clock for the Pixel Data.
N2	P24	Digital video output	Video Pixel Output Port.
N3	INT2	Miscellaneous digital	Interrupt. This pin can be active low or active high. When status bits change, this pin is triggered. The events that trigger an interrupt are under user control.
N4	TEST4	Test	Connect this pin to ground.
N5	RESET	Miscellaneous digital	System Reset Input. Active low. A minimum low reset pulse width of 5 ms is required to reset the ADV7842 circuitry.
N6	TEST6	Test	Float this pin.
N7	SDRAM_A8	SDRAM interface	Address Output. Interface to external RAM address lines.
N8	SDRAM_A4	SDRAM interface	Address Output. Interface to external RAM address lines.
N9	SDRAM_A0	SDRAM interface	Address Output. Interface to external RAM address lines.

Pin No.	Mnemonic	Type	Description
N10	SDRAM_CS	SDRAM interface	Chip Select. SDRAM_CS enables and disables the command decoder on the RAM. One of four command signals to the external SDRAM.
N11	SDRAM_LDQS	SDRAM interface	Lower Data Strobe Pin. Data strobe pins are used for the RAM interface. This is an output with read data and an input with write data. It is edge aligned with write data and centered in read data. SDRAM_ LDQS corresponds to the data on SDRAM_DQ0 to SDRAM_ DQ7
N12	SDRAM_DQ4	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
N13	SDRAM_DQ15	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
N14	SDRAM_DQ11	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
N15	$\overline{\text{SDRAM\_CK}}$	SDRAM interface	Differential Clock Output. All address and control output signals to the RAM should be sampled on the positive edge of SDRAM_CK and on the negative edge of $\overline{\text{SDRAM\_CK}}$.
N16	SDRAM_CKE	SDRAM interface	Clock Enable. This pin acts as an enable to the clock signals of the external RAM.
P1	P25	Digital video output	Video Pixel Output Port.
P2	P26	Digital video output	Video Pixel Output Port.
P3	TEST5	Test	Connect this pin to ground.
P4	AVLINK	Digital input/output	Digital SCART Control Channel.
P5	TEST7	Test	Float this pin.
P6	SDRAM_A11	SDRAM interface	Address Output. Interface to external RAM address lines.
P7	SDRAM_A7	SDRAM interface	Address Output. Interface to external RAM address lines.
P8	SDRAM_A3	SDRAM interface	Address Output. Interface to external RAM address lines.
P9	SDRAM_A10	SDRAM interface	Address Output. Interface to external RAM address lines.
P10	$\overline{\text{SDRAM\_RAS}}$	SDRAM interface	Row Address Select Command Signal. One of four command signals to the external SDRAM.
P11	SDRAM_DQ7	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
P12	SDRAM_DQ3	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
P13	SDRAM_VREF	SDRAM interface	1.25 V Reference for DDR SDRAM Interface or 1.65 V for SDR SDRAM Interface.
P14	SDRAM_DQ12	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
P15	SDRAM_UDQS	SDRAM interface	Upper Data Strobe Pin. Data strobe pins for the RAM interface. This is an output with read data and an input with write data. It is edge aligned with write data and centered in read data. SDRAM_UDQS corresponds to the data on SDRAM_DQ8 to SDRAM_DQ15.
P16	$\overline{\text{SDRAM\_CK}}$	SDRAM interface	Differential Clock Output. All address and control output signals to the RAM should be sampled on the positive edge of SDRAM_CK and on the negative edge of $\overline{\text{SDRAM\_CK}}$.
R1	P27	Digital video output	Video Pixel Output Port.
R2	P28	Digital video output	Video Pixel Output Port.
R3	P30	Digital video output	Video Pixel Output Port.
R4	P32	Digital video output	Video Pixel Output Port.
R5	P34	Digital video output	Video Pixel Output Port.
R6	SDRAM_A9	SDRAM interface	Address Output. Interface to external RAM address lines.
R7	SDRAM_A6	SDRAM interface	Address Output. Interface to external RAM address lines.
R8	SDRAM_A2	SDRAM interface	Address Output. Interface to external RAM address lines.
R9	SDRAM_BA1	SDRAM interface	Bank Address Output. Interface to external RAM bank address lines.

Pin No.	Mnemonic	Type	Description
R10	$\overline{\text{SDRAM\_CAS}}$	SDRAM interface	Column Address Select Command Signal. One of four command signals to the external SDRAM.
R11	SDRAM_DQ6	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
R12	SDRAM_DQ2	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
R13	SDRAM_DQ0	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
R14	SDRAM_DQ13	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
R15	SDRAM_DQ9	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
R16	SDRAM_DQ8	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
T1	GND	Ground	Ground.
T2	P29	Digital video output	Video Pixel Output Port.
T3	P31	Digital video output	Video Pixel Output Port.
T4	P33	Digital video output	Video Pixel Output Port.
T5	P35	Digital video output	Video Pixel Output Port.
T6	GND	Ground	Ground.
T7	SDRAM_A5	SDRAM interface	Address Output. Interface to external RAM address lines.
T8	SDRAM_A1	SDRAM interface	Address Output. Interface to external RAM address lines.
T9	SDRAM_BA0	SDRAM interface	Bank Address Output. Interface to external RAM bank address lines.
T10	$\overline{\text{SDRAM\_WE}}$	SDRAM interface	Write Enable Output Command Signal. One of four command signals to the external SDRAM.
T11	SDRAM_DQ5	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
T12	GND	Ground	Ground.
T13	SDRAM_DQ1	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
T14	SDRAM_DQ14	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
T15	SDRAM_DQ10	SDRAM interface	Data Bus. Interface to external RAM 16-bit data bus.
T16	GND	Ground	Ground.

3 GLOBAL CONTROL REGISTERS

The register control bits described in this section deal with the general control of the chip and the three main sections of the ADV7842: the SDP, the CP, and the HDMI receiver.

3.1 ADV7842 REVISION IDENTIFICATION

RD_INFO[15:0], Addr 40 (IO), *Address 0xEA[7:0]; Address 0xEB[7:0] (Read Only)*

Silicon Revision ID.

Function

RD_INFO[15:0]	Description
0x2001	ADV7842 ES2 Silicon
All other values	Invalid for ADV7842

3.2 POWER-DOWN CONTROLS

3.2.1 Primary Power-down Controls

POWER_DOWN is the main power-down control. It is the main control for power-down mode 0 and mode 1. Refer to Section 3.2.3 for more details.

POWER_DOWN, Addr 40 (IO), *Address 0x0C[5]*

This control enables power-down mode. It is the main I2C power-down control.

Function

POWER_DOWN	Description
0	Chip is operational
1 «	Enables chip power down

3.2.2 Secondary Power-down Controls

The CP_PWRDN, XTAL_PDN, CORE_PDN, and VDP_PDN controls allow various sections of the ADV7842 to be powered down.

For a power-sensitive application, it is possible to stop the clock to the CP to reduce power. The CP_PWRDN bit enables this power-save mode. The SDP, VDP, and HDMI blocks are not affected by this power-save mode. This allows the use of limited HDMI, SSPD, and STDI monitoring features while reducing the power consumption. For full processing of HDMI input the CP core needs to be powered up.

The XTAL_PDN bit allows the user to power down the XTAL clock in the following sections:

- STDI blocks
- SSPD blocks
- Free run synchronization generation block
- I^2C sequencer block, which is used for the configuration of the gain, clamp, and offset
- CP and HDMI section

- Frequency measurement block, and its derivatives – e.g. frequency settings for equalizers
- DDC pads (active pull-up depends on Xtal)
- Reset

The XTAL clock is also provided to the HDCP engine, EDID, and the repeater controller within the HDMI receiver. The XTAL clock within these sections is not affected by XTAL_PDN.

The CORE_PDN bit allows the user to power down clocks, with the exception of the XTAL clock, in the following sections:

- DPP block
- CP block
- SDP block
- Digital section of the HDMI block

The following sections remain active when CORE_PDN is set:

- STDI block
- SSPD block

CP_PWRDN, Addr 40 (IO), *Address 0x0C[2]*

This is a power-down control for the CP core.

Function

CP_PWRDN	Description
0 «	Powers up the clock to the CP core.
1	Powers down the clock to the CP core. VDP, SDP and HDMI blocks are not affected by this bit.

XTAL_PDN, Addr 40 (IO), *Address 0x0B[0]*

This is a power-down control for the XTAL in the digital blocks.

Function

XTAL_PDN	Description
0 «	Powers up XTAL buffer to digital core
1	Powers down XTAL buffer to digital core

CORE_PDN, Addr 40 (IO), *Address 0x0B[1]*

This is a power-down control for the DPP, CP core and digital sections of the HDMI core.

Function

CORE_PDN	Description
0 «	Powers up DPP, CP, SDP and digital sections of HDMI block
1	Powers down the DPP, CP, SDP and digital section of HDMI block. STDI and SSPD are still active when CORE_PDN is set.

VDP_PDN, Addr 40 (IO), *Address 0x0C[1]*

This is a power-down control for the VDP. It is recommended to power-down the VDP when this feature is not required.

Function

VDP_PDN	Description
0	Powers up the VDP section
1 «	Powers down the VDP section

3.2.3 Power-down Modes

The ADV7842 supports the following power-down modes.

- Power-down mode 0
- Power-down mode 1

Table 2 shows the power-down and normal modes of the ADV7842.

Table 2: Power Down Modes Overview

PWRDN1 Pin	CEC_POWER_UP Bit	AVL_AVLINK_POWER_UP Bit	CEC	AVLINK	Internal E-EDID Access	Power-down Mode
0	0	0	Disabled	Disabled	Yes	Power-down mode 0
0	0	1	Disabled	Enabled	Yes	Power-down mode 1
0	1	0	Enabled	Disabled	Yes	Power-down mode 1
0	1	1	Enabled	Enabled	Yes	Power-down mode 1
1	0	0	Disabled	Disabled	Yes [1]	Normal mode
1	0	1	Disabled	Enabled	Yes [1]	Normal mode
1	1	0	Enabled	Disabled	Yes [1]	Normal mode
1	1	1	Enabled	Enabled	Yes [1]	Normal mode

3.2.3.1 Power-down Mode 0

In power-down mode 0, selected sections and pads are kept active to provide E-EDID and +5 V antiglitch filter functionality.

In power-down mode 0, all the sections of the ADV7842 are disabled except for the following blocks:

- I2C slave section
- E-EDID/Repeater controller
- E-EDID ring oscillator
  The ring oscillator provides a clock to the E-EDID/Repeater controller (refer to Section 8.3) and the +5 V power supply antiglitch filter. The clock output from the ring oscillator runs approximately at 50 MHz.

The following are the only pads enabled in power-down mode 0:

[1] Dependant on the value of EDID_X_ENABLE_CPU and EDID_X_ENABLE for each HDMI port (where X is A or B)

- I2C pads:
  - SDA
  - SCL

- +5 V pads:
  - RXA_5V
  - RXB_5V
  - HPA_A
  - HPA_B

- DDC pads:
  - DDCA_SCL
  - DDCA_SDA
  - DDCB_SCL
  - DDCB_SDA
  - DDCC_SCL
  - DDCC_SDA
  - DDCD_SCL
  - DDCD_SDA

- SPI EEPROM interface pads:
  - EP_MOSI
  - EP_MISO
  - $\overline{\text{EP\_CS}}$
  - EP_SCK

- Reset pad :
  - $\overline{\text{RESET}}$

The power-down mode 0 is initiated through either a software (I2C register) configuration or a hardware configuration.

Entering Power-down Mode 0 via Software

The ADV7842 should be put into power-down mode 0 via software as follows:

- ADI Recommend Write: HDMI Map (68 C0 00)
- Set the POWER_DOWN bit to 1

Leaving Power-down Mode 0 via Software

The ADV7842 should be removed from power-down mode 0 via software as follows:

- ADI Recommend Write: HDMI Map (68 C0 F0)
- Clear the POWER_DOWN bit to 0

Note: If not using the fast switching HDMI feature of the ADV7842, register 0xC0 in the HDMI Map should always be set to 0x00 regardless of the power down status of the part. The recommended write above for entering/leaving power-down mode 0 via software is not required.

Entering Power-down Mode 0 via Hardware

The ADV7842 should be put into power-down mode 0 by feeding a low voltage on the $\overline{\text{PWRDWN1}}$ pin. Refer to Figure 4 for the required hardware configuration.

This hardware functionality allows the internal E-EDID to be available even if the main supply (e.g. AC power) is not available to power up the ADV7842. The +5 V from the HDMI source(s) connected to the system can be used to provide sufficient power for the sections and pads activated in power-down mode 0.

Note: The recommended PSS (refer to Section 0) must be adhered to when using the +5 V line from an HDMI source to power the part.

The power consumption of the ADV7842 is less than 50 mA in power-down mode 0. This allows the part to be powered by the +5 V signal from one or more HDMI ports when the ADV7842 enters power-down mode 0. The internal E-EDID is also accessible through the DDC bus for ports A, B, C, and D in both power-down mode 0 and mode 1.

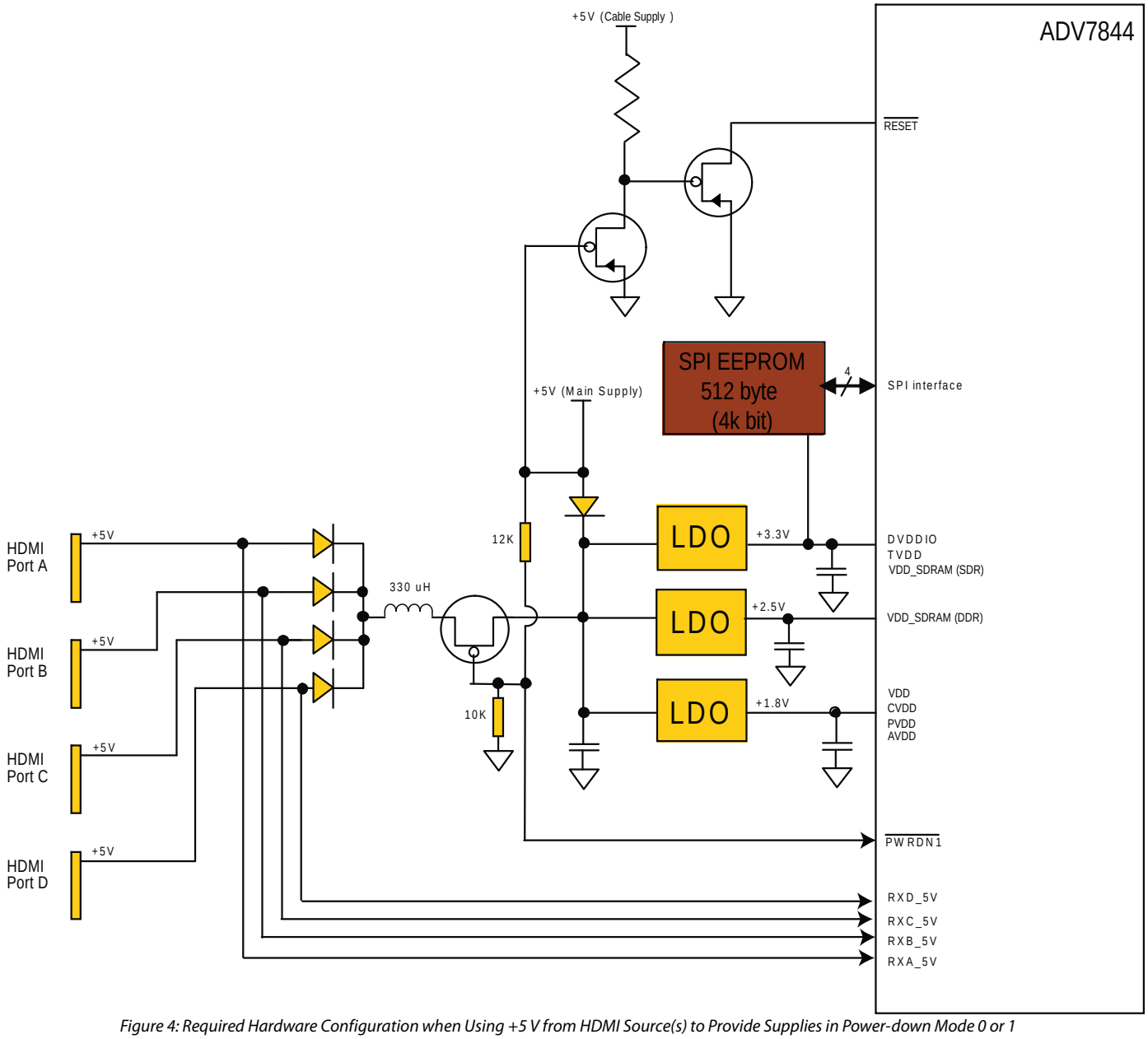


Figure 4: Required Hardware Configuration when Using +5 V from HDMI Source(s) to Provide Supplies in Power-down Mode 0 or 1

Leaving Power-down Mode 0 via Hardware

The ADV7842 should be removed from power-down mode 0 by feeding a high voltage on the $\overline{\text{PWRDWN1}}$ pin.

$\overline{\text{PWRDN1}}$ Configuration

The configuration shown in Figure 4 is required for hardware power-down with internal E-EDID availability.

When accessing the internal E-EDID with the main system power removed from ADV7842, the following circuit design precautions must be strictly adhered to:

- Each of the power supplies must be powered from the cable. All 3.3 V, 2.5 V, and 1.8 V supplies must be powered up.
- Power supply sequencing must be adhered to when powering from the HDMI cable (refer to Section 0).
- The HDMI specification limits the HDMI cable to 50 mA. Therefore, when using this supply to provide power to the ADV7842 in this mode, it is not recommended that any other circuitry is powered from the cable other than the ADV7842 and the SPI EEPROM.

It is important to ensure the correct operation of the internal E-EDID. In system power-down mode, the following requirements must always be met when removing system power from the chip:

- The $\overline{\text{RESET}}$ pin must go low before or at the same time as the main system supply is removed.
- The $\overline{\text{RESET}}$ pin must go low before or at the same time as the $\overline{\text{PWRDN1}}$ goes low. The circuitry in Figure 4 is required to ensure this occurs. It is important to note that this requirement must be met even if the main supply is removed without notice, for example, if a TV is disconnected abruptly from the wall socket.

The configuration shown in Figure 5 can be implemented if the $\overline{\text{PWRDN1}}$ pin is unused in the design. Alternatively, the $\overline{\text{PWRDN1}}$ pin can be left floating and its effect on the powerdown status can be disabled via I2C by setting DIS_PWRDNB to 1.

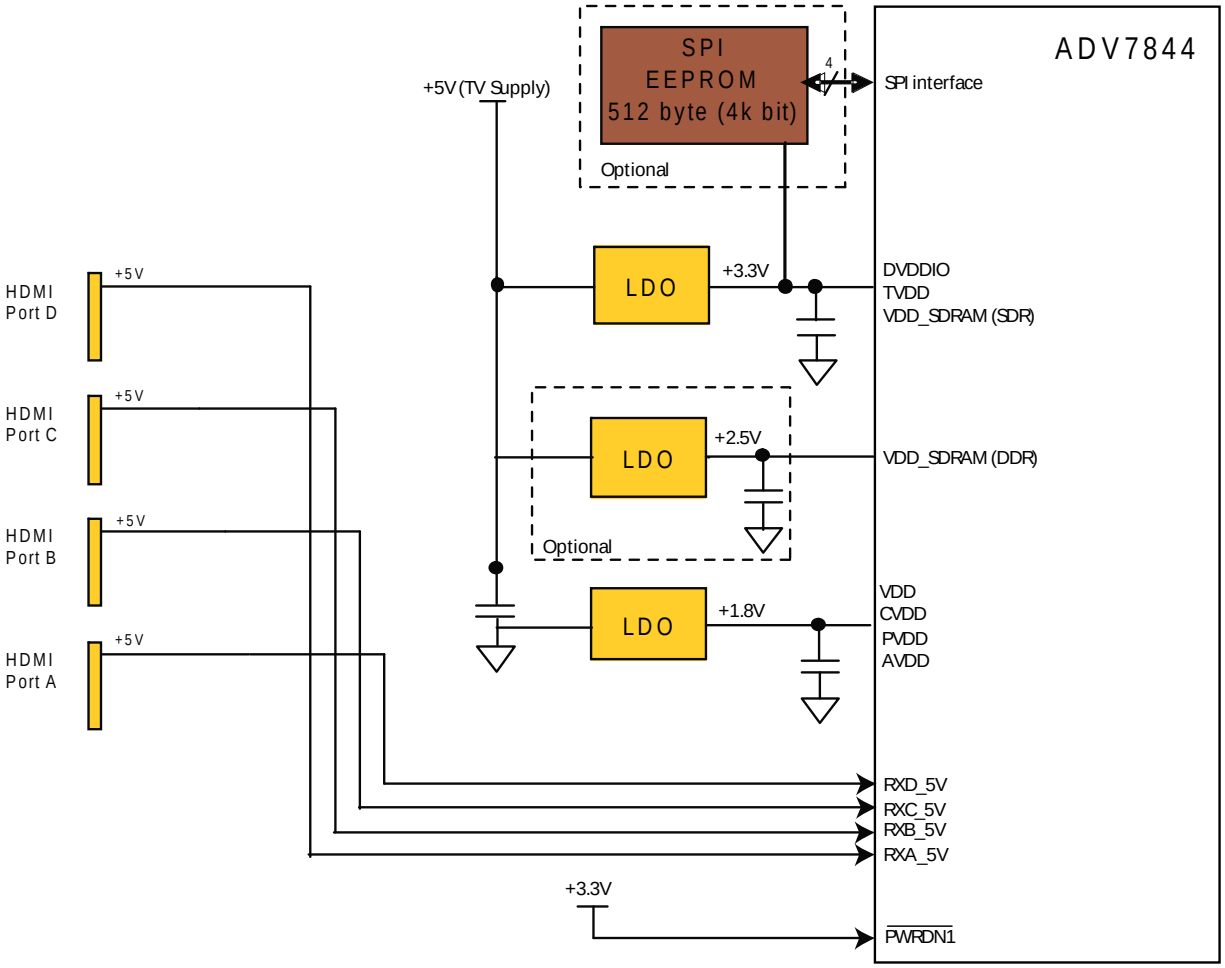


Figure 5: Hardware Configuration Recommended When PWRDN1 is Unused

3.2.3.2 Power-Down Mode 1

Power-down mode 1 is enabled when the following conditions are met:

- The ADV7842 is configured to be in power-down mode 0, via either a hardware or software configuration (refer to Section 3.2.3.1)
- The CEC section is enabled by setting CEC_POWER_UP to 1 or the AV.link section is enabled by setting bit AVL_AVLINK_POWER_UP to 1.

Power-down mode 1 provides the same functionality as the power-down mode 0 with the addition of the following sections:

- XTAL clock
- CEC section
- AV.link section
- Interrupt controller section

The following pads are enabled in power-down mode 1:

- The same pads that are enabled in power-down mode 0
- CEC pad
- AVLink pad
- INT1 and INT2 interrupt pads

The internal E-EDID is also accessible through the DDC bus for ports A, B, C, and D in power-down modes 0 and 1.

The effect of the $\overline{\text{PWRDN1}}$ pin can be disabled by setting DIS_PWRDNB to 1. When this is done, the part is in power-down mode 1 when:

- POWER_DOWN is set to 1
- CEC section is enabled by setting CEC_POWER_UP to 1 or the AV.link section is enabled by setting AVL_AVLINK_POWER_UP to 1

DIS_PWRDNB, Addr 68 (HDMI), *Address 0x48[7]*

This control is used to disable the effect of the PWRDN1 pin. DIS_PWRDNB should be set to 1 if the PWRDN1 pin is unused and unconnected.

Function

DIS_PWRDNB	Description
0 «	PWRDN1 pin is used to set the power mode of the part (e.g. Power Down mode 0, Power Down mode 1 or Normal mode).
1	PWRDN1 has no effect

3.2.4 ADC Power-Down Control

The ADV7842 contains 12-bit ADCs (ADC 0, ADC 1, ADC 2, and ADC3). It is possible to power down each ADC individually, if required.

PDN_ADC0, Addr 4C (AFE), *Address 0x00[0]*

This control is used to power down ADC0.

Function

PDN_ADC0	Description
0	Powered up.
1 «	Powered down.

PDN_ADC1, Addr 4C (AFE), *Address 0x00[1]*

This control is used to power down ADC1.

Function

PDN_ADC1	Description
0	Powered up.
1 «	Powered down.

PDN_ADC2, Addr 4C (AFE), *Address 0x00[2]*

This control is used to power down ADC2.

Function

PDN_ADC2	Description
0	Powered up.
1 «	Powered down.

PDN_ADC3, Addr 4C (AFE), *Address 0x00[3]*

This control is used to power down ADC3.

Function

PDN_ADC3	Description
0	Powered up.
1 «	Powered down.

PDN_ADC_CLK0, Addr 4C (AFE), *Address 0x00[4]*

Powerdown the clock to the ADC 0 circuitry

Function

PDN_ADC_CLK0	Description
0 «	Powered up.
1	Powered down.

PDN_ADC_CLK1, Addr 4C (AFE), *Address 0x00[5]*

Powerdown the clock to the ADC 1 circuitry

Function

PDN_ADC_CLK1	Description
0 «	Powered up.
1	Powered down.

PDN_ADC_CLK2, Addr 4C (AFE), *Address 0x00[6]*

Powerdown the clock to the ADC 2 circuitry

Function

PDN_ADC_CLK2	Description
0 «	Powered up.
1	Powered down.

PDN_ADC_CLK3, Addr 4C (AFE), *Address 0x00[7]*

Powerdown the clock to the ADC3 circuitry

Function

PDN_ADC_CLK3	Description
0 «	Powered up.
1	Powered down.

3.2.5 DDC and VGA Pins Power-Down

VGA_PWRDN, Addr 68 (HDMI), *Address 0x72[4]*

This control is used to power down for the VGA EDID pads.

Function

VGA_PWRDN	Description
0 «	Power up VGA EDID pads
1	Powerdown VGA EDID pads

3.3 RESET CONTROLS AND GLOBAL PIN CONTROLS

3.3.1 Reset Pin

The ADV7842 can be reset by a low reset pulse on the $\overline{\text{RESET}}$ pin with a minimum width of 5 ms. It is recommended to wait 5 ms after the low pulse before an I^2C write is performed to the ADV7842.

3.3.2 Reset Controls

MAIN_RESET, Addr 40 (IO), *Address 0xFF[7] (Self-Clearing)*

This control is used to reset the I2C registers to their default values.

Function

MAIN_RESET	Description
0 «	No function
1	Applies main I2C reset

VDP_RESET, Addr 40 (IO), *Address 0xFF[5] (Self-Clearing)*

This control is used to reset the VDP FIFO and controller.

Function

VDP_RESET	Description
0 «	No function
1	Apply VDP reset

SDP_RESET, Addr 40 (IO), *Address 0xFF[3] (Self-Clearing)*

This control is used to reset the SDP.

Function

SDP_RESET	Description
0 «	No function
1	Applies SDP reset

CEC_SOFT_RESET, Addr 80 (CEC), *Address 0x2C[0] (Self-Clearing)*

CEC module software reset.

Function

CEC_SOFT_RESET	Description
0 «	No function
1	Reset the CEC module

AVL_SOFT_RESET, Addr 84 (AVLINK), *Address 0x29[0] (Self-Clearing)*

AV.link module software reset.

Function

AVL_SOFT_RESET	Description
0 «	No function
1	Reset the AV.link module

3.3.3 Tristate Output Drivers

TRI_PIX allows the user to tristate the output driver of pixel outputs.

TRI_PIX, Addr 40 (IO), *Address 0x15[1]*

This control is used to tristate the pixel data on the pixel pins P[35:0].

Function

TRI_PIX	Description
0	Pixel bus active
1 «	Tristates pixel bus

3.3.4 Tristate LLC Driver

TRI_LLC, Addr 40 (IO), *Address 0x15[2]*

This control is used to tristate the output pixel clock on the LLC pin.

Function

TRI_LLC	Description
0	LLC pin active
1 «	Tristates LLC pin

3.3.5 Tristate Synchronization Output Drivers

When TRI_SYNCS is set, the following output synchronization signals are tristated:

- VS/FIELD
- HS/CS
- FIELD/DE

When TRI_SYNC_OUT is set, the synchronization signal on the SYNC_OUT pin is tristated.

The drive strength controls for these signals are provided via the DR_STR_SYNC bits. The ADV7842 does not support tristating via a dedicated pin.

TRI_SYNCS, Addr 40 (IO), *Address 0x15[3]*

Synchronization output pins tristate control. The synchronization pins under this control are HS/CS, VS/FIELD, and FIELD/DE.

Function

TRI_SYNCS	Description
0	Sync output pins active
1 «	Tristate sync output pins

TRI_SYNC_OUT, Addr 40 (IO), *Address 0x15[5]*

Tristate control for SYNC_OUT pin.

Function

TRI_SYNC_OUT	Description
0	SYNC_OUT pin active
1 «	Tristate SYNC_OUT pin

3.3.6 Tristate Audio Output Drivers

TRI_AUDIO is used to tristate the drivers of the following audio output signals:

- AP0
- AP1
- AP2
- AP3
- AP4
- AP5
- SCLK
- MCLK

The drive strength for the output pins can be controlled by the DR_STR[1:0] bits. The ADV7842 does not support tristating via a dedicated pin.

TRI_AUDIO, Addr 40 (IO), *Address 0x15[4]*

This control is used to tristate the audio output interface pins, AP[5:0], SCLK and MCLK.

Function

TRI_AUDIO	Description
0	Audio output pins active
1 «	Tristates audio output pins

3.3.7 Tristate Memory Interface

SDP_TRI_MEMORY_IF, Addr 94 (SDP_IO), *Address 0x29[4]*

This control is used to tristate memory interface pins. Setting this bit tristates the interface address, data control, data strobe, and mask lines.

Function

SDP_TRI_MEMORY_IF_B	Description
0 «	Tristates memory interface pins
1	Does not tristate memory interface pins

3.3.8 Drive Strength Selection

It may be desirable to strengthen or weaken the drive strength of the output drivers for Electromagnetic Compatibility (EMC) and crosstalk reasons.

The drive strength DR_STR[1:0] bits affect the output drivers for the following output pins:

- P[35:0]
- AP0-AP5
- SCLK
- SDA
- SCL

Refer to the descriptions of DR_STR[1:0] and DR_STR_CLK[1:0] for the drive strength control of the LLC, synchronization, and audio output signals.

DR_STR[1:0], Addr 40 (IO), *Address 0x14[5:4]*

This control is used to set the drive strength of the data output drivers.

Function

DR_STR[1:0]	Description
00	Low (1x)
01	Medium low (2x)
10 «	Medium high (3x)
11	High (4x)

DR_STR_CLK[1:0], Addr 40 (IO), *Address 0x14[3:2]*

This control is used to set the drive strength control for the output pixel clock out signal on the LLC pin.

Function

DR_STR_CLK[1:0]	Description
00	Low (1x)
01	Medium low (2x) for LLC up to 60 MHz
10 «	Medium high (3x) for LLC from 44 MHz to 105 MHz
11	High (4x) for LLC greater than 100 MHz

DR_STR_SYNC[1:0], Addr 40 (IO), *Address 0x14[1:0]*

This control is used to set the drive strength of the synchronization pins, HS/CS, VS/FIELD, FIELD/DE and SYNC_OUT.

Function

DR_STR_SYNC[1:0]	Description
00	Low (1x)
01	Medium low (2x)
10 «	Medium high (3x)
11	High (4x)

The DR_STR_SYNC[1:0] bits allow the user to select the strength of the following synchronization signals:

- FIELD/DE
- HS/CS
- VS/FIELD
- SYNC_OUT

Refer to DR_STR_CLK[1:0] and DR_STR_SYNC[1:0] for the drive strength controls of the pixel bus, pixel clock, and audio output signals.

3.3.9 Output Synchronization Selection

VS_OUT_SEL, Addr 40 (IO), *Address 0x06[7]*

This control is used to select a VSync signal or Field signal to be output on the VS/Field pin.

Function

VS_OUT_SEL	Description
0	Field output on VS/FIELD pin
1 «	VSync output on VS/FIELD pin

F_OUT_SEL, Addr 40 (IO), *Address 0x05[4]*

This control is used to select a DE signal or a Field signal to be output on the FIELD/DE pin.

Function

F_OUT_SEL	Description
0 «	DE output selected
1	Field output selected

HS_OUT_SEL[1:0], Addr 40 (IO), *Address 0x06[6:5]*

This control is used to select the signal to be output on the HS/CS output pin.

Function

HS_OUT_SEL[1:0]	Description
00	Regenerates CSync signal, synchronous to LLC
01 «	Regenerates HSync signal, synchronous to LLC
10	Asynchronous HSync
11	Depending on result of SSPD, signal output on HS/CS pin is: Logic AND of HS and VS input after polarity correction, if HS/VS detected by SSPD. CS input if CS/VS detected by SSPD. Sliced embedded sync if embedded synchronization detected by SSPD

3.3.10 Output Synchronization Signals Polarity

INV_LLC_POL, Addr 40 (IO), *Address 0x06[0]*

This control is used to invert the polarity of the LLC.

Function

INV_LLC_POL	Description
0 «	Does not invert LLC
1	Inverts LLC

The output synchronization signals HS/CS, VS/FIELD, FIELD/DE and SYNC_OUT can be inverted using the following control bits:

- INV_HS_POL
- INV_VS_POL
- INV_F_POL
- INV_SYNC_OUT_POL

The synchronization path is shown in Figure 6. The polarity of the synchronization signals at the different sections of the path is as follows:

- **Section 1**: The SSPD channel 1 receives the following signals from the AFE and HDMI sections:
  - HS/CS
  - VS
- **Section 2**: The SSPD channel 2 receives the following signals from the AFE and HDMI sections:
  - HS/CS
  - VS
- **Section 3:** The SSPD channel 1 outputs the following signals to the CP core:
  - HS with negative polarity
  - VS with negative polarity
- **Section 4**: The SSPD channel 2 outputs the following signals to the CP core:
  - HS with negative polarity
  - VS with negative polarity
- **Section 5**: The CP core outputs the following signals to the output formatter section:
  - HS with negative polarity
  - VS with negative polarity
  - DE with negative polarity
    Note that the CP core generates the DE signal for analog input. In HDMI mode, the CP core regenerates the DE signal from the DE output by the HDMI receiver section.
- **Section 6**: The output formatter outputs the synchronization signal through the pins as follows:
  - HS with positive polarity only if INV_HS_POL is set to 1
  - VS with positive polarity only if INV_VS_POL is set to 1
  - FIELD/DE with positive polarity only if INV_F_POL is set to 1

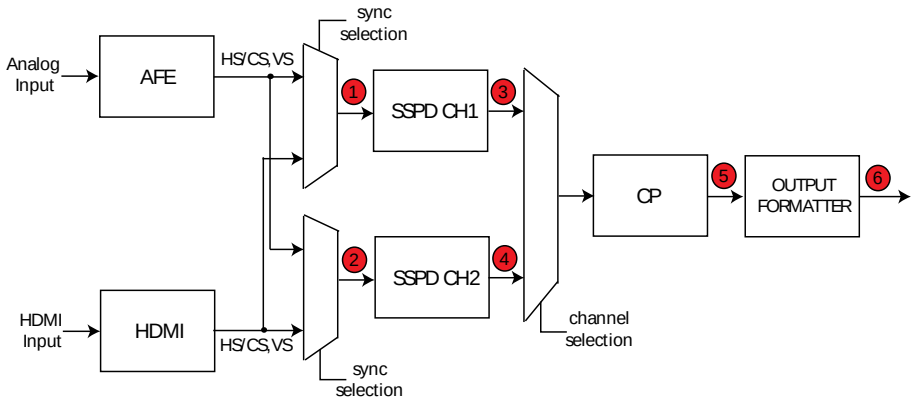


Figure 6: Synchronization Path

INV_HS_POL, Addr 40 (IO), *Address 0x06[1]*

This control is used to select the polarity of the HS/CS signal.

Function

INV_HS_POL	Description
0 «	Negative polarity HS/CS
1	Positive polarity HS/CS

INV_VS_POL, Addr 40 (IO), *Address 0x06[2]*

This control is used to select the polarity of the VS/FIELD signal

Function

INV_VS_POL	Description
0 «	Negative polarity VS/FIELD
1	Positive polarity VS/FIELD

INV_F_POL, Addr 40 (IO), *Address 0x06[3]*

This control is used to select the polarity of the FIELD/DE signal.

Function

INV_F_POL	Description
0 «	Negative FIELD/DE polarity
1	Positive FIELD/DE polarity

INV_SYNC_OUT_POL, Addr 40 (IO), *Address 0x06[4]*

This control is used to set the polarity of the SYNC_OUT signal. This bit is not valid in power-save mode.

Function

INV_SYNC_OUT_POL	Description
0 «	Negative polarity SYNC_OUT signal
1	Positive polarity SYNC_OUT signal

3.3.11 Digital Synthesizer Controls

The ADV7842 features two digital encoder synthesizers that generate the following clocks:

- Video DPLL: This clock synthesizer generates an ADC sampling clock/pixel clock for analog, DVI, and HDMI inputs. For analog input, the video DPLL locks when it has received a horizontal synchronization with 128 consistent periods. For DVI/HDMI input, the video DPLL locks within 128 TMDS clock periods after the TMDS PLL has locked.

- Audio DPLL: This clock synthesizer generates an audio master clock HDMI input. The audio PLL locks within two periods of the audio sampling clock after the TMDS PLL has locked and when the part has received valid ACR packets.

3.3.12 Pixel Clock Generation for Analog CP Control

ADV7842 uses PLL to generate the pixel clock for Analog CP modes. PLL_DIV_RATIO is a register that sets the divide ratio for that PLL. For standard modes, the PLL divide ratio is automatically decoded from the programmed PRIM_MODE[3:0] and VID_STD[5:0] registers. PLL_DIV_RATIO can also be programmed manually via registers PLL_DIV_RATIO[12:0] and PLL_DIV_RATIO_MAN_EN in the IO Map for modes that are not supported by primary mode and video standard settings. This feature is used also in the configuration of the autographics mode (refer to Section 10.15).

PLL_DIV_MAN_EN, Addr 40 (IO), *Address 0x16[7]*

This control is used to manually override the PLL divider ratio value.

Function

PLL_DIV_MAN_EN	Description
0 «	Disables manual PLL divider ratio settings. PLL divider ratio set by PRIM_MODE[3:0] and VID_STD[5:0]
1	Manually sets PLL_DIV ratio as defined by PLL_DIV[12:0]

PLL_DIV_RATIO[12:0], Addr 40 (IO), *Address 0x16[4:0]; Address 0x17[7:0]*

This control is used to manually set the PLL divide ratio. These registers are sequenced and require sequential writes in order to update the value.

Function

PLL_DIV_RATIO[12:0]	Description
xxxxxxxxxxxxx	Synthesizer feedback value. PLL_DIV_MAN_EN must be set for this value to be active.

In analog CP mode, the digital encoder synthesizer locks to the incoming HSync signal and generates the pixel clock from the HSync signal. The relationship between the HSync and pixel clock frequencies is shown in Equation 1.

$$F_{TLLC} = PLL\_DIV\_RATIO \cdot F_{HS}$$

where:

F_{TLLC} is the frequency of the pixel clock

F_{HS} is the frequency of the HSync signal

Equation 1: HSync to TLLC Frequency

Important: If the PLL_DIV_RATIO is set manually, it should only be programmed with an even value when the part processes video data in oversampling mode. (Refer to Section 4 for information on oversampling mode.)

Note: In digital input mode, the pixel and audio master clocks are generated from the digital encoder synthesizer provided with the

incoming TMDS clock. PLL_DIV_RATIO is not used in this mode.

3.3.13 ADC Phase Control

DLL_PHASE[5:0], Addr 4C (DPLL), *Address 0xC8[5:0]*

This control is used to adjust the phase of the ADC sampling clock.

Function

DLL_PHASE[5:0]	Description
000000 «	Default
xxxxxx	Adjusts the phase of the ADC sampling clock

3.4 ADC-HDMI SIMULTANEOUS MODE

The ADV7842 can be configured to be in either simultaneous mode or non simultaneous mode, as follows:

- **Non simultaneous mode**
  In this mode, the ADV7842 will process either analog or HDMI/DVI inputs. The HDMI section is disabled when the ADV7842 is configured to process analog inputs. The ADCs are powered down when the part is configured to process HDMI/DVI inputs in HDMI mode.

- **Simultaneous mode**
  In this mode, specific subsections of the HDMI block remain enabled when the ADV7842 is programmed to process analog inputs through the CP and SD core. Simultaneous mode keeps the HDCP engine and the E-EDID/Repeater controller active, allowing an upstream transmitter to authenticate the ADV7842 even when the latter is in analog mode. Keeping the HDCP engine active allows for fast switching from analog mode to HDMI mode, as the transmitter will have already authenticated the ADV7842 when the latter is switched into HDMI mode. Simultaneous mode is also used for SDP and HDMI audio mode.

Notes:

- Simultaneous mode has no effect when the part is programmed in HDMI mode as the full HDMI section is enabled in this mode.
- The ADCs are powered down automatically in HDMI mode – with the exception of simultaneous CVBS+HDMI audio mode, mentioned above
- The following HDMI subsections are active and functional when the ADV7842 runs in simultaneous mode:
  - TMDS equalizer (refer to Section 8.13)
  - TMDS clock detection circuitry (refer to Section 8.16)
  - TMDS clock measurement circuitry (refer to Section 8.17)
  - HDCP decryption engine (refer to Section 0)
  - HDMI synchronization filters (refer to Section 8.22)
  - InfoFrame and packet extraction processor (refer to Section 8.32)
- The following HDMI subsections are active and functional irrespective of the simultaneous mode:
  - 5 V deglitch filter and detection circuitry (refer to Section 8.1)
  - E-EDID/Repeater controller (refer to Section 8.3)

ADC_HDMI_SIMULT_MODE, Addr 40 (IO), *Address 0x01[7]*

This control is used to enable ADC and HDMI simultaneous mode. In this mode certain HDMI functionality is available when processing analog inputs.

Function

ADC_HDMI_SIMULT_MODE	Description
0 «	Disables simultaneous mode
1	Enables simultaneous mode

HDCP_ONLY_MODE, Addr 68 (HDMI), *Address 0x01[2]*

This control is used to configure a HDCP only mode for simultaneous analog and HDMI modes. Refer to the ADC_HDMI_SIMULTANEOUS_MODE bit. By selecting HDCP only mode, HDMI activity is reduced and it can be used as a power saving feature in simultaneous analog and HDMI operation.

Function

HDCP_ONLY_MODE	Description
0 «	Normal operation
1	HDCP only mode

4 PRIMARY MODE AND VIDEO STANDARD

Setting the primary mode and choosing a video standard are the most fundamental settings when configuring the ADV7842.
here are four main modes of operation on the ADV7842. These modes are enabled by setting PRIM_MODE[3:0] appropriately.

- **SDP mode**
  Analog-composite and analog S-Video mode. This covers all standard definition modes that have a modulated color subcarrier. Typical examples are PAL-BGHID, PAL-M/N, NTSC-M/N and SECAM. SDP also supports SD YPrPb and SD SCART RGB. The SDP core is the main processing block in this mode.

- **COMP mode**
  Analog-component video mode. This includes all video signals that arrive in a YPbPr (or YUV) analog format. Typical examples are progressive and high definition video signals. The CP core is the main processing block in this mode.

- **GR mode**
  Analog-graphic mode. This mode is intended for RGB input signals with high bandwidth. The CP core is the main processing block in this mode.

- **HDMI mode**
  HDMI mode. In HDMI mode the ADV7842 can receive and decode HDMI or DVI data throughout the DVI/HDMI receiver front end. Video data from the HDMI receiver is routed to the DPP and CP blocks while audio data is available on the audio interface. This mode is enabled by selecting either the HDMI (Graphics) primary mode or HDMI (Comp) primary mode. The HDMI and CP core are the main processing blocks in this mode.

4.1 PRIMARY MODE AND VIDEO STANDARD CONTROLS

PRIM_MODE[3:0], Addr 40 (IO), *Address 0x01[3:0]*

A control to selects the primary mode of operation of the decoder. To be used with VID_STD[5:0].

Function

PRIM_MODE[3:0]	Description
0000	SDP mode
0001	Component Mode
0010	Graphics mode
0011	Reserved
0100	CVBS & HDMI AUDIO Mode
0101	HDMI-Comp
0110 «	HDMI-GR
0111 - 1111	Reserved

VID_STD[5:0], Addr 40 (IO), *Address 0x00[5:0]*

Sets the expected video standard and desired oversampling mode. The configuration values vary with PRIM_MODE[3:0] setting. A detailed table with Primary Mode and Video Standard settings is available in the hardware manual.

Function

VID_STD[5:0]	Description
001000 «	Default value

PRIM_MODE[3:0] should be used with **VID_STD[5:0]** to set the desired video mode. Both controls should be set accordingly to below table.

Table 3: Primary Mode and Video Standard Selection

PRIM_MODE[3:0]		VID_STD[5:0]				
Code	**Description**	**Processor**	**Code**	**Input Video**	**Output Resolution**	**Comment**
0000	**SDP** (Standard Definition) e.g. CVBS	SDP	00000	SD 2x1	720x480i/576i	CVBS
			00001	SD 4x1	720x480i/576i	CVBS
			00010	SD 8x1	720x480i/576i	CVBS
			00011	Reserved	Reserved	
			00100	SD 2x1	720x480i/576i	SCART RGB
			00101	SD 4x1	720x480i/576i	SCART RGB
			00110	SD 8x1	720x480i/576i	SCART RGB
			00111	Reserved	Reserved	
			01000	SD 2x1	720x480i/576i	YC
			01001	SD 4x1	720x480i/576i	YC
			01010	SD 8x1	720x480i/576i	YC
			01011	Reserved	Reserved	
			01100	SD 2x1	720x480i/576i	YC auto
			01101	SD 4x1	720x480i/576i	YC auto
			01110	SD 8x1	720x480i/576i	YC auto
			01111	Reserved	Reserved	
			11111	Reserved	Reserved	
			10000	SD 2X1 YPrPb	720X480i/576i	SD Component
			10001	SD 4X1 YPrPb	720X480i/576i	SD Component
			10010	SD 8X1 YPrPb	720X480i/576i	SD Component
			10011	Reserved	Reserved	
			10100	Reserved	Reserved	
			10101	Reserved	Reserved	
			10110	Reserved	Reserved	
			10111	Reserved	Reserved	
			11000	SD 2x1 RGB	720x480i/576i	SD Component
			11001	SD 4x1 RGB	720x480i/576i	SD Component
			11010	SD 8x1 RGB	720x480i/576i	SD Component
			11110	Reserved	Reserved	
			11111	Reserved	Reserved	
0001	**COMP** (Component Video)		000000	Reserved	Reserved	
			000001	Reserved	Reserved	
			000010	SD 2x1 525i	720 x 480i	
			000011	SD 2x1 625i	720 x 576i	

PRIM_MODE[3:0]		VID_STD[5:0]				
Code	**Description**	**Processor**	**Code**	**Input Video**	**Output Resolution**	**Comment**
	e.g. YPrPb		000100	SD 4x1 525i	720 x 480i	
			000101	SD 4x1 625i	720 x 576i	
			000110	SD 2x2 525i	1440 x 480i	
			000111	SD 2x2 625i	1440 x 576i	
		CP	001000	SD 4x2 525i	1440 x 480i	
			001001	SD 4x2 625i	1440 x 576i	
			001010	Reserved	Reserved	
			001011	Reserved	Reserved	
			001100	PR 2x1 525p	720 x 480p	
			001101	PR 2x1 625p	720 x 576p	
			001110	PR 4x1 525p	720 x 480p	
			001111	PR 4x1 625p	720 x 576p	
			010000	PR 2x2 525p	1440 x 480p	
			010001	PR 2x2 625p	1440 x 576p	
			010010	Reserved	Reserved	
			010011	HD 1x1 720p	1280 x 720p	SMPTE 296M
			010100	HD 1x1 1125	1920 x 1080i	SMPTE 274M
			010101	HD 1x1 1125	1920 x 1035i	SMPTE 240M
			010110	HD 1x1 1250	1920 x 1080i	SMPTE 295M
			010111	HD 1x1 1250	1920 x 1152i	
			011000	Reserved	Reserved	
			011001	HD 2x1 720p	1280 x 720p	SMPTE 296M
			011010	HD 2x1 1125	1920 x 1080i	SMPTE 274M
			011011	HD 2x1 1125	1920 x 1035i	SMPTE 240M
			011100	HD 2x1 1250	1920 x 1080i	SMPTE 295M
			011101	HD 2x1 1250	1920 x 1152i	
			011110	HD 1x1 1125p	1920 x 1080p	SMPTE 274M
			011111	HD 1x1 1250p	1920 x 1080p	SMPTE 295M
			~	Reserved	Reserved	
			~	Reserved	Reserved	
			111110	HD 1x1 1125p	1920 x 1080p	Reduced blanking
			111111	Reserved	Reserved	
0010	**GR**		000000	SVGA	800 x 600p @ 56	
			000001	SVGA	800 x 600p @ 60	
	(Graphics)		000010	SVGA	800 x 600p @ 72	
			000011	SVGA	800 x 600p @ 75	
	e.g. RGB		000100	SVGA	800 x 600p @ 85	
			000101	SXGA	1280 x 1024p @ 60	
			000110	SXGA	1280 x 1024p @ 75	
			000111	Auto-graphics Mode	Various	
			001000	VGA	640 x 480p @ 60	

PRIM_MODE[3:0]		VID_STD[5:0]				
Code	**Description**	**Processor**	**Code**	**Input Video**	**Output Resolution**	**Comment**
		CP	001001	VGA	640 x 480p @ 72	
			001010	VGA	640 x 480p @ 75	
			001011	VGA	640 x 480p @ 85	
			001100	XGA	1024 x 768p @ 60	
			001101	XGA	1024 x 768p @ 70	
			001110	XGA	1024 x 768p @ 75	
			001111	XGA	1024 x 768p @ 85	
			010000	WXGA	1280 x768p @ 60	
			010001	WXGA	1280 x768p @ 60	With reduced blanking
			010010	WXGA	1360 x768p @ 60	
			010011	WXGA	1366 x768p @ 60	
			010100	SXGA+	1400x1050p @ 60	
			010101	SXGA+	1400x1050p @ 75	
			010110	UXGA	1600x1200p @ 60	
			010111	UXGAR	1600x1200p @ 60	With reduced blanking
			011000	WSXGA	1680x1050p @ 60	
			011001	WUXGAR	1920x1200p @ 60	With reduced blanking
			~	Reserved	Reserved	
			111111	Reserved	Reserved	
0011	**Reserved**		xxxxx	Reserved	Reserved	
0100	**SDP + HDMI Audio**	SD + CP	000000	SD 2x1	720 x 480i/576i	CVBS
			000001	Reserved	Reserved	
			000010	Reserved	Reserved	
			000011	Reserved	Reserved	
			000100	Reserved	Reserved	
			000101	Reserved	Reserved	
			000110	Reserved	Reserved	
			000111	Reserved	Reserved	
			001000	SD 2x1 YUV	720x480i/576i	YC
			001001	Reserved	Reserved	
			001010	SD 2x1 RGB	720x480i/576i	YPrPb
			~	Reserved	Reserved	
			111111	Reserved	Reserved	
0101	**HDMI-COMP** (Component Video)	HDMI + CP	000000	SD 1x1 525i	720 x 480i	HDMI Receiver Support
			000001	SD 1x1 625i	720 x 576i	
			000010	SD 2x1 525i	720 x 480i	
			000011	SD 2x1 625i	720 x 576i	
			000100	Reserved	Reserved	
			000101	Reserved	Reserved	
			000110	Reserved	Reserved	
			000111	Reserved	Reserved	
			001000	Reserved	Reserved	
			001001	Reserved	Reserved	

PRIM_MODE[3:0]		VID_STD[5:0]				
Code	**Description**	**Processor**	**Code**	**Input Video**	**Output Resolution**	**Comment**
			001010	PR 1x1 525p	720 x 480p	
			001011	PR 1x1 625p	720 x 576p	
			001100	PR 2x1 525p	720 x 480p	
			001101	PR 2x1 625p	720 x 576p	
			001110	Reserved	Reserved	
			001111	Reserved	Reserved	
			010000	Reserved	Reserved	
			010001	Reserved	Reserved	
			010010	Reserved	Reserved	
			010011	HD 1x1	1280 x 720p	
			010100	HD 1x1	1920 x 1080i	
			010101	HD 1x1	1920 x 1035	
			010110	HD 1x1	1920 x 1080	
			010111	HD 1x1	1920 x 1152	
			011000	Reserved	Reserved	
			011001	HD 2x1 720p	1280 x 720p	
			011010	HD 2x1 1125	1920 x 1080i	
			011011	HD 2x1 1125	1920 x 1035i	
			011100	HD 2x1 1250	1920 x 1080i	
			011101	HD 2x1 1250	1920 x 1152i	
			011110	HD 1x1 1125p	1920 x 1080i	
			011111	HD 1x1 1250p	1920 x 1080p	
0110			000000	SVGA	800 x 600p @ 56	HDMI Receiver Support
			000001	SVGA	800 x 600p @ 60	
			000010	SVGA	800 x 600p @ 72	
			000011	SVGA	800 x 600p @ 75	
			000100	SVGA	800 x 600p @ 85	
			000101	SXGA	1280 x 1024p @ 60	
			000110	SXGA	1280 x 1024p @ 75	
	HDMI-GR		000111	Reserved	Reserved	
		HDMI + CP	001000	VGA	640 x 480p @ 60	
	(Graphics)		001001	VGA	640 x 480p @ 72	
			001010	VGA	640 x 480p @ 75	
			001011	VGA	640 x 480p @ 85	
			001100	XGA	1024 x 768p @ 60	
			001101	XGA	1024 x 768p @ 70	
			001110	XGA	1024 x 768p @ 75	
			001111	XGA	1024 x 768p @ 85	
			01xxxx	Reserved		
0111	**Reserved**		xxxxxx	Reserved	Reserved	
1000	**Reserved**		xxxxxx	Reserved	Reserved	
1001	**Reserved**		xxxxxx	Reserved	Reserved	

PRIM_MODE[3:0]		VID_STD[5:0]				
Code	**Description**	**Processor**	**Code**	**Input Video**	**Output Resolution**	**Comment**
1010	**Reserved**		xxxxxx	Reserved	Reserved	
1011	**Reserved**		xxxxxx	Reserved	Reserved	
1100	**Reserved**		xxxxxx	Reserved	Reserved	
1101	**Reserved**		xxxxxx	Reserved	Reserved	
1110	**Reserved**		xxxxxx	Reserved	Reserved	
1111	**Reserved**		xxxxxx	Reserved	Reserved	

Setting the Vertical Frequency

V_FREQ[2:0] is used when the decoder is required to support HD standards SMPTE 274, systems 6, 7, 8, 9, 10, and 11; and SMPTE 296, systems 3, 4, 5, and 6. These are the 50 Hz, 30 Hz, and 24 Hz standards listed within these standards.

This control should be set when the primary mode is configured for component mode. The primary mode is set by the PRIM_MODE[3:0] control. Table 4 details the vertical frequencies supported by the ADV7842 for various resolutions.

Note that this control may also be used when setting the HDMI free run mode (refer to Section 10.13.3).

V_FREQ[2:0], Addr 40 (IO), *Address 0x01[6:4]*

A control to set vertical frequency of HD component standards.

Function

V_FREQ[2:0]	Description
000 «	60 Hz
001	50 Hz
010	30 Hz
011	25 Hz
100	24 Hz
101	Reserved
110	Reserved
111	Reserved

Table 4: Vertical Frequencies Supported in HD Modes

Standard	SM296 720p 1x1	BT709 1035i 1x1	SM274 1080i 1x1	SM295 1250i 1x1	BT709 1250i 1x1	SM274 1080p 1x1	SM295 1250p 1x1
Vertical Frequency ↓ (Hz)							
SM274-1,2	✓	✓	✓	✓	✓	✓	✓
SM274-3	✓		✓			✓	
SM274-7,8	✓					✓	
SM274-9	✓					✓	
SM274-10, 11	✓					✓	

4.2 SELECTING PRIMARY MODE AND VIDEO STANDARD FOR HDMI MODES

The HDMI receiver decodes and processes any applied HDMI stream irrespective of the video resolution. However, there are many

Primary Mode and Video Standards settings in order to define how the decoded video data that is routed to the DPP and CP blocks is processed. This allows for free run features and data decimation modes that some systems may require.

If free run and decimation are not required it is recommended to set the following configuration for HDMI mode:

- PRIM_MODE[3:0]: 0x06
- VID_STD[5:0]: 0x02

4.3 HDMI DECIMATION MODES

Some of the modes defined by VID_STD have an inherent 2x1 decimation. For these modes, the main clock generator and the decimation filters in the DPP block are configured automatically. This ensures the correct data rate at the input to the CP block. Refer to Section **9** for more information on the automatic configuration of the DPP block.

The ADV7842 correctly decodes and processes any incoming HDMI stream with the required decimation irrespective of its video resolution.

- In 1x1 mode (i.e. without decimation) as long the PRIM_MODE and VID_STD registers are programmed for any HDMI mode without decimation.
  For example:
  - Set PRIM_MODE to 0x5 and VID_STD to 0x00
  - Set PRIM_MODE to 0x5 and VID_STD to 0x13
  - Set PRIM_MODE to 0x6 and VID_STD to 0x02

- In 2x1 decimation mode as long the PRIM_MODE and VID_STD registers are programmed for any HDMI mode with 2x1 decimation.
  For example:
  - Set PRIM_MODE to 0x5 and VID_STD to 0x0C
  - Set PRIM_MODE to 0x5 and VID_STD to 0x19

Note that decimating the video data from an HDMI stream is optional and should only be performed if it is required by the downstream devices connected to the ADV7842.

4.4 PRIMARY MODE AND VIDEO STANDARD CONFIGURATION FOR HDMI FREE RUN

If free run is enabled in HDMI mode, PRIM_MODE[3:0] and VID_STD[5:0] specify the input resolution expected by the ADV7842 (for free run mode 1) and/or the output resolution to which the ADV7842 free runs (for free run mode 0 and mode 1). Refer to Section **10.13** for additional details on the free run feature for HDMI inputs.

4.5 RECOMMENDED SETTINGS FOR HDMI INPUTS

This section provides the recommended settings for an HDMI input encapsulating a video resolution corresponding to a selection Video ID Code described in the CEA861 specification.

Table 5 provides recommended settings for the following registers:

- PRIM_MODE
- VID_STD
- V_FREQ. V_FREQ should be set to 0x0 if not specified in Table 5.
- INV_HS_POL = 1. INV_HS_POL should be set to 1 if not specified in Table 5.
- INV_VS_POL = 1. INV_VS_POL should be set to 1 if not specified in Table 5.

Table 5: Recommended Settings for HDMI Inputs

Video ID Codes (861 Specification)	Formats	Pixel Repetition	Recommended Settings if Free Run is not Used OR Free Run is Used and DIS_AUTOPARAM_BUFFER = 0	Recommended Settings if Free Run is Used and DIS_AUTOPRAM_BUFFER = 1
2, 3	720x480p @ 60 Hz	0	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE = 0x5 VID_STD = 0xA
4	1280x720p @ 60 Hz	0	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE = 0x5 VID_STD = 0x13
5	1920x1080i @ 60 Hz	0	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE = 0x5 VID_STD = 0x14
6, 7	720 (1440)x 480i @ 60 Hz	1	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE = 0x5 VID_STD = 0x0
10, 11	2880x480i @ 60 Hz	3	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE = 0x5 VID_STD = 0x0
14, 15	1440x480p @ 60 Hz	1	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE=0x5 VID_STD = 0xA
16	1920x1080p @ 60 Hz	0	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE = 0x5 VID_STD = 0x1E
17, 18	720x576p @ 60 Hz	0	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE = 0x5 VID_STD = 0xB
19	1280x720p @ 50 Hz	0	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE = 0x5 VID_STD = 0xA3 V_FREQ = 0x1
20	1920x1080i @ 50 Hz	0	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE = 0x5 VID_STD = 0x14 V_FREQ = 0x1
21, 22	720 (1440)x576i @ 60 Hz	1	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE = 0x5 VID_STD = 0x1
25, 26	2880x480i @ 60 Hz	3	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE=0x5 VID_STD = 0x1
29, 30	1440x576p @ 60 Hz	1	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE = 0x5 VID_STD = 0xA
31	1920x1080p @ 50 Hz	0	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE = 0x5 VID_STD = 0x1E V_FREQ = 0x1
32	1920x1080p @ 24 Hz	0	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE = 0x5 VID_STD = 0x1E V_FREQ = 0x4
33	1920x1080p @ 25 Hz	0	PRIM_MODE = 0x6 VID_STD = 0x2	PRIM_MODE = 0x5 VID_STD = 0x1E

Video ID Codes (861 Specification)	Formats	Pixel Repetition	Recommended Settings if Free Run is not Used OR Free Run is Used and DIS_AUTOPARAM_BUFFER = 0	Recommended Settings if Free Run is Used and DIS_AUTOPRAM_BUFFER = 1
35, 36	2880x480p @ 60 Hz	3	V_FREQ = 0x3 PRIM_MODE = 0x6 VID_STD = 0x2	V_FREQ = 0x3 PRIM_MODE = 0x5 VID_STD = 0xA
37, 38	2880x576p @ 60 Hz	3	V_FREQ = 0x3 PRIM_MODE = 0x6 VID_STD = 0x2	V_FREQ = 0x3 PRIM_MODE = 0x5 VID_STD = 0xA

4.6 STANDARD CONFIGURATION FOR SDP – HDMI SIMULTANEOUS AUDIO

CVBS video can be processed simultaneously with HDMI audio. In this configuration, CVBS is only supported in 2x1 over sampling mode. This mode is available for S-Video and 525i and 625i SD modes. This mode can be selected by the relevant PRIM_MODE and VID_STD. Frame TBC and 3D comb are not available in this mode. To configure the ADV7842 for this mode, the following I2C writes should be carried out.

- Primary mode should be configured for SDP and HDMI audio mode (PRIM_MODE [3:0] set to 0x04)
- ADC simultaneous mode should be enabled (ADC_HDMI_SIMULT_MODE set to 0x01).
- The desired 2x1 SDP mode should be configured by setting VID_STD[5:0]
    - 2x1 CVBS (VID_STD[5:0] set to 0x00)
    - 2x1 YC (VID_STD[5:0] set to 0x08)
    - 2x1 YPrPb (VID_STD[5:0] set to 0x10)-

5 PIXEL PORT CONFIGURATION

The ADV7842 has a very flexible pixel port, which can be configured in a variety of formats to accommodate downstream ICs. The ADV7842 can provide output modes up to 36 bits.

This section details the controls required to configure the ADV7842 pixel port. The Appendix contains tables detailing some example pixel port configurations. A spreadsheet tool, the ADV7842-output-pixel-port-mapping.xls, can generate the pinout for all combinations of pixel port configuration controls. Refer to this spreadsheet for a full overview of the available output formats.

Figure 7 provides a screen shot of the spreadsheet.

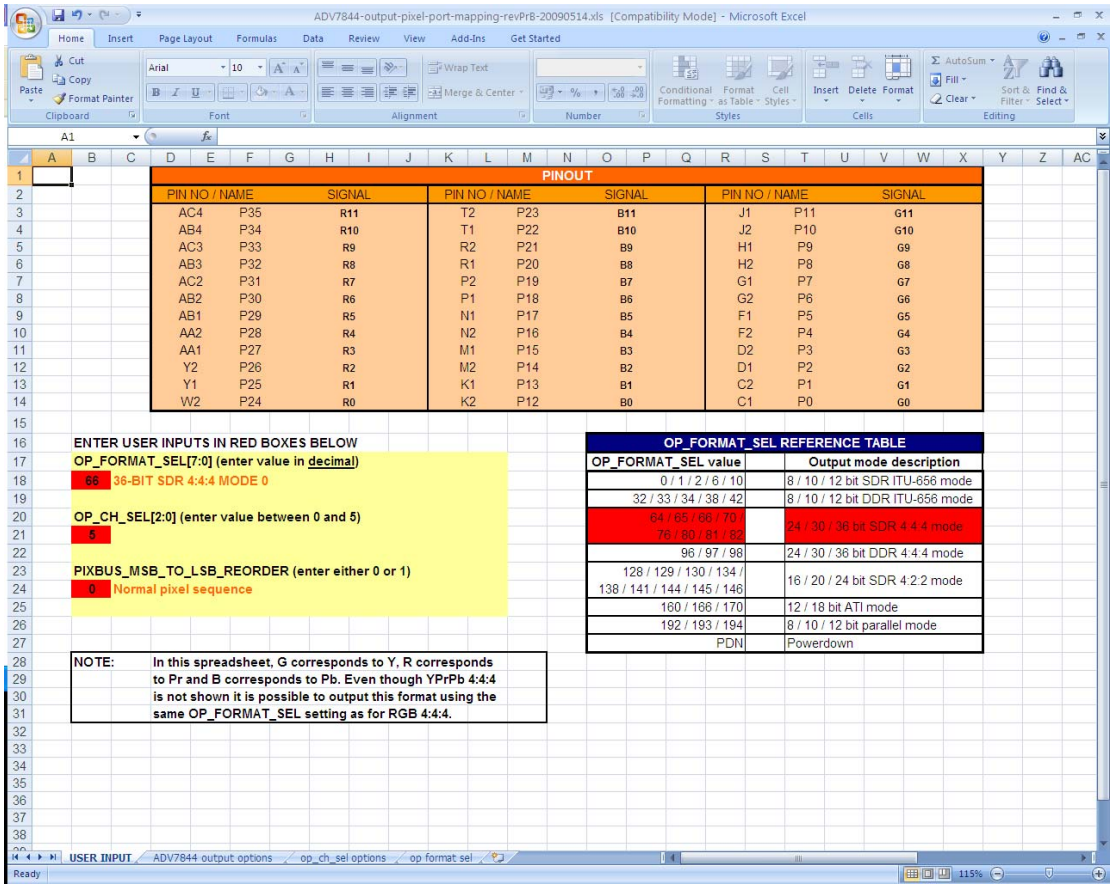


Figure 7: ADV7842-Output-Pixel-Port-Mapping.xls

5.1 PIXEL PORT OUTPUT MODES

OP_FORMAT_SEL[7:0], Addr 40 (IO), *Address 0x03[7:0]*

A control to select the data format and pixel bus configuration. Refer to the pixel port configuration spreadsheet tool for full information on pixel port modes and configuration settings.

Function

OP_FORMAT_SEL[7:0]	Description
0x00 «	8-bit SDR ITU-656 mode
0x01	10-bit SDR ITU-656 mode
0x02	12-bit SDR ITU-656 mode 0
0x06	12-bit SDR ITU-656 mode 1
0x0A	12-bit SDR ITU mode 2
0x20	8-bit 4:2:2 DDR mode
0x21	10-bit 4:2:2 DDR mode
0x22	12-bit 4:2:2 DDR mode 0
0x23	12-bit 4:2:2 DDR mode 1
0x24	12-bit 4:2:2 DDR mode 2
0x40	24-bit 4:4:4 SDR mode
0x41	30-bit 4:4:4 SDR mode
0x42	36-bit 4:4:4 SDR mode 0
0x46	36-bit SDR 4:4:4 mode 1
0x4C	24-bit SDR 4:4:4 mode 3
0x50	24-bit SDR 4:4:4 mode 4
0x51	30-bit SDR 4:4:4 mode 4
0x52	36-bit SDR 4:4:4 mode 4
0x60	24-bit 4:4:4 DDR mode
0x61	30-bit 4:4:4 DDR mode
0x62	36-bit 4:4:4 DDR mode
0x80	16-bit ITU-656 SDR mode
0x81	20-bit ITU-656 SDR mode
0x82	24-bit ITU-656 SDR mode 0
0x86	24-bit ITU-656 SDR mode 1
0x8A	24-bit ITU-656 SDR mode 2
0x8D	20-bit SDR 4:2:2 mode 3
0x90	16-bit SDR 4:2:2 mode 4
0x91	20-bit SDR 4:2:2 mode 4
0x92	24-bit SDR 4:2:2 mode 4
0xC0	8-bit PAR mode 0
0xC1	10-bit PAR mode 0
0xC2	12-bit PAR mode 0

5.1.1 Bus Rotation and Reordering Controls

Bus swapping and bus reordering controls are available for ADV7842. OP_CH_SEL[2:0] allows the three output buses to be rearranged, providing six different output possibilities. PIXBUS_MSB_TO_LSB_REORDER allows the pixel sequence to be modified. These controls operate on the CP output bus only.

OP_CH_SEL[2:0], Addr 40 (IO), *Address 0x04[7:5]*

A control to select the configuration of the pixel data bus on the pixel pins. Refer to the pixel port configuration spreadsheet tool for full information on pixel port modes and configuration settings.

Function

OP_CH_SEL[2:0]	Description
000	P[35:24] Y/G, P[23:12] U/CrCb/B, P[11:0] V/R
001	P[35:24] Y/G, P[23:12] V/R, P[11:0] U/CrCb/B
010	P[35:24] U/CrCb/B, P[23:12] Y/G, P[11:0] V/R
011 «	P[35:24] V/R, P[23:12] Y/G, P[11:0] U/CrCb/B
100	P[35:24] U/CrCb/B, P[23:12] V/R, P[11:0]Y/G
101	P[35:24] V/R, P[23:12] U/CrCb/B, P[11:0] Y/G
110	Reserved
111	Reserved

PIXBUS_MSB_TO_LSB_REORDER, Addr 40 (IO), *Address 0x30[4]*

A control to swap the MSB to LSB orientation on the pixel bus.

Function

PIXBUS_MSB_TO_LSB_ REORDER	Description
0 «	Output bus goes from MSB to LSB
1	Output bus goes from LSB to MSB

5.1.2 Pixel Data and Synchronization Signals Control

The polarity of the LLC and synchronization signals can be inverted, and the LLC, synchronization signals, and the pixel data output can be tristated. Refer to the information on the following controls:

- INV_SYNC_OUT_POL
- INV_F_POL
- INV_VS_POL
- INV_HS_POL
- TRI_PIX
- TRI_LLC
- TRI_SYNCS

OP_SWAP_CB_CR, Addr 40 (IO), *Address 0x05[0]*

A controls the swapping of Cr and Cb data on the pixel buses.

Function

OP_SWAP_CB_CR	Description
0 «	Outputs Cr and Cb as per OP_FORMAT_SEL
1	Inverts the order of Cb and Cr in the interleaved data stream

OP_SWAP_CB_CR swaps the order in which Cb and Cr are interleaved in the output data stream. It caters for cases in which the data on channels B and C are swapped.

It is effective only if OP_FORMAT_SEL[7:0] is set to a 4:2:2 compatible output mode

Note: It has no effect for 24-bit SDR modes and DDR modes.

5.1.3 Rounding and Truncating Data

CP_PREC[1:0], Addr 44 (CP), *Address 0x77[7:6]*

A control to set the precision of the data output by the CP core for channels A, B and C.

Function

CP_PREC[1:0]	Description
00	Rounds and truncates data in channels A, B and C to 10-bit precision
01	Rounds and truncates data in channels A, B and C to 12-bit precision
10	Rounds and truncates data in channels A, B and C to 8 bit precision
11 «	Rounds and truncates data in channels A, B, and C to the precision set in OP_FORMAT_SEL[6:0]

This rounding and truncation provides simple truncation of the unwanted LSBs. If the discarded LSBs are greater or equal to mid-scale, one LSB is added to the remaining bits.

For example, if the internal 12-bit data is 0b100101010101 and it is to be truncated to 10 bits, the two LSBs (0b01) are discarded. The discarded LSBs are less than mid-scale, so the 10-bit output word is 0b1001010101. However, if the internal 12-bit data is 0b100101010110, the discarded LSB is 0b10, and one LSB is added to the remaining 10-bit word, making the output data 0b1001010110.

It is also possible to use noise shaping when truncating the data word from 12-bits to 10-bits or 8-bits.

BR_NOISE_SHAPING_EN, Addr 44 (CP), *Address 0x36[5]*

Enables a noise shaped truncation of the data from 12 bits to 10 bits or 8 bits (as specified by CP_PREC[1:0])

Function

BR_NOISE_SHAPING_EN	Description
0 «	Disables noise shaped bit reduction. Simple rounding is used for bit reduction
1	Enables noise shaped bit width reduction.

BR_NOISE_SHAPING_MODE, Addr 44 (CP), *Address 0x36[4]*

A control to select the bit reduction noise shaping mode. This bit is effective if BR_NOISE_SHAPING_EN is set 1. This feature should only be used in HDMI modes.

Function

BR_NOISE_SHAPING_MODE	Description
0 «	Noise Shaping Mode 0
1	Noise Shaping Mode 1

BR_NOISE_SHAPING_GAIN[1:0], Addr 44 (CP), *Address 0x36[3:2]*

A control set the gain applied to the noise shaping bit in mode 1.

Function

BR_NOISE_SHAPING_GAIN[1:0]	Description
00 «	Gain of 1
01	Gain of 2
10	Gain of 4
11	Gain of 8

TEN_TO_EIGHT_CONV, CP Map, *Address 0x36, [0]*

This control is used to configure when the data to be rounded and truncated is 10-bit and not of 12-bit data. This control is for HDMI use only, where the data can be 8-, 10- or 12-bit precision.

TEN_TO_EIGHT_ CONV should be set to 0 in any of the following cases:

- Pixel data input has 12-bit precision per channel components (i.e. channel Y, U, and V). The part has been configured via OP_FORMAT_SEL to output 12-bit (2-LSBs of OP_FORMAT_SEL set to 10b or 11b), 10-bit (2-LSBs of OP_FORMAT_SEL set to 01b), or 8-bit per channel component (2-LSBs of OP_FORMAT_SEL set to 00b).
- Pixel data input has 10-bit precision per channel component. The part has been configured via OP_FORMAT_SEL to output 10-bit per channel component.
- Pixel data input has 8-bit precision per channel component. The part has been configured via OP_FORMAT_SEL to output 8-bit per channel component.

TEN_TO_EIGHT_ CONV should be set to 1 if the pixel data input has 10-bit precision per channel component and the part has been configured via OP_FORMAT_SEL to output 8-bit per channel component.

TEN_TO_EIGHT_CONV, Addr 44 (CP), *Address 0x36[0]*

A control to indicate if the precision of the data to be rounded and truncated to 8-bit has 10 bit precision. This control is for HDMI use only.

Function

TEN_TO_EIGHT_CONV	Description
0 «	If the input data has got 12 bit precision - then the output data will have 12-, 10- or 8-bits per channel. If the input data has got 10 bit precision - then the output data will have 10-bits per channel. If the input data has got 8 bit precision - then the output data will have 8-bits per channel.
1	If The input data has got 10 bit precision, the output data will be 8 bits per channel.

Note: BR_NOISE_SHAPING_EN and TEN_TO_EIGHT_CONV should be enabled only in HDMI modes.

5.1.4 DDR Output Interface

The ADV7842 allows data to be output in a DDR mode.

Internally, the ADV7842 produces three data streams: either R, G, and B, or Y, Cr, and Cb, depending on whether the input is RGB or YPrPb. If a 4:4:4 output is selected, the three data streams are presented on separate buses. The data values are normally updated on every clock rising edge (SDR 4:4:4) or on both clock edges – rising and falling edges with 50% clock rate (DDR 4:4:4).

For a YPrPb input ADV7842 can decimate and interleave the Pr and Pb data streams, and output the video data on two buses instead of three (SDR 4:2:2).

The ADV7842 can also output this data stream with the half clock rate, having Y samples presented on the clock falling edge (depending on LLC polarity settings), and alternative Pb/Pr samples presented on the clock rising edge (depending on LLC polarity settings) (DDR 4:2:2).

All these modes are available via OP_FORMAT_SEL[7:0]. Refer also to the Appendix .

Important: The maximum frequency of the DDR clock supported by the ADV7842 is 50 MHz. The DDR clock is output though the LLC pin. Note also that SDR 4:4:4 and DDR 4:4:4 modes from SDP core are not available.

5.1.5 *Parallel Output Modes*

The ADV7842 can provide three parallel output modes. These modes are selected via OP_FORMAT_SEL[7:0]. These parallel modes are available when processing a CVBS, an SVideo, or a component input up to 480i/576i input. There are two output pixel buses in these modes. The main bus can either be converted to a progressive format by using the I-to-P converter (refer to Section 7.18) or it can be in an interlaced format by bypassing the I-to-P converter. The auxiliary pixel bus will always be in an interlaced format.

Figure 8 provides a block diagram representation of this functionality.

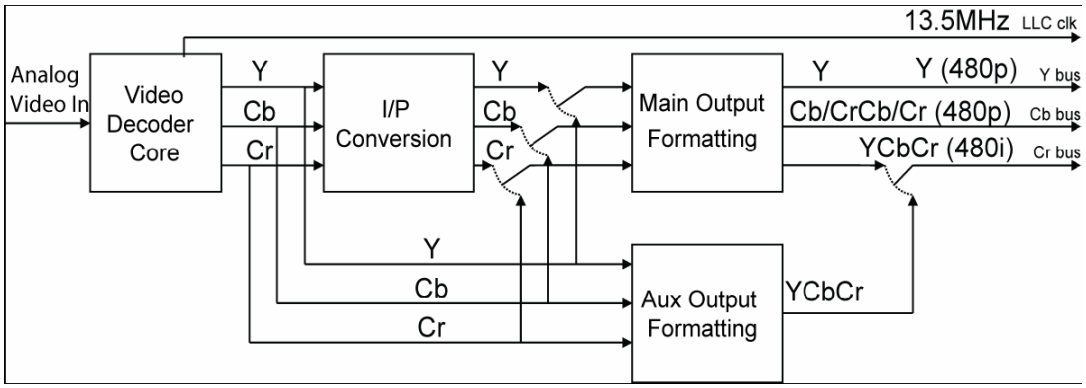


Figure 8: Parallel Output Mode

In parallel mode, both the embedded synchronization and external synchronization are available to the main output block but only SAV/EAV is available to the auxiliary output block.

The auxiliary block has independent SAV/EAV controls to allow the output timing to be adjusted for this block. These controls are outlined in Section 7.22.7. Only one LLC clock output is available in parallel mode.

5.2 LLC CONTROLS

The ADV7842 has a limited number of adjustment features available for the line locked clock (LLC) output. The polarity of the LLC can be inverted and the LLC the output driver tristated. Controls also exist to skew the LLC versus the output data to achieve suitable setup and hold times for any back end device.

The LLC controls are as follows:

- INV_LLC_POL
- TRI_LLC
- LLC_DLL_EN
- LLC_DLL_MUX
- LLC_DLL_PHASE[4:0]

The frequency of LLC can also be multiplied by 2, divided by 2, or phase shifted.

5.3 DLL ON LLC CLOCK PATH

A DLL block is implemented on the LLC clock path. This DLL allows the changing of the phase of the output pixel clock on the LLC pin.

5.3.1 Adjusting DLL phase in all modes

LLC_DLL_EN, Addr 40 (IO), *Address 0x19[7]*

A control to enable the Delay Locked Loop for output pixel clock. LLC_DLL_MUX must be set to 1 for this setting to be effective.

Function

LLC_DLL_EN	Description
1	Enable LLC DLL
0 «	Disable LLC DLL

LLC_DLL_MUX, Addr 40 (IO), *Address 0x33[6]*

A control to apply the pixel clock DLL to the pixel clock output on the LLC pin.

Function

LLC_DLL_MUX	Description
0 «	Bypasses the DLL
1	Muxes the DLL output on LLC output

LLC_DLL_PHASE[4:0], Addr 40 (IO), *Address 0x19[4:0]*

A control to adjust LLC DLL phase in increments of 1/32 of a clock period. LLC_DLL_MUX must be set to 1 for this setting to be effective.

Function

LLC_DLL_PHASE[4:0]	Description
00000 «	Default
xxxxx	Sets on of 32 phases of DLL to vary LLC CLK

6 ANALOG FRONT END

The analog front end (AFE) comprises the following:

- High performance 12-bit analog to digital converters (ADCs) with clamping circuitry
- Twelve analog inputs and multiplexing capability
- Four synchronization (SYNC1, SYNC2, SYNC3 and SYNC4) input multiplexers with synchronization slicers and filtering
- Variable bandwidth anti aliasing filters
- LLC-DLL (Line Locked Clock – Delay Locked Loop)
- Eight trilevel input detection blocks

6.1 ADC SAMPLING CLOCK

The ADV7842 has two main modes of operation for sampling the input analog video: CP mode and SDP mode. This is determined by the primary mode setting.

- When the SDP is enabled, fixed 114.545 MHz sampling is applied to the ADCs. The SDP processes the video signal and, using a line length tracking processor, resamples the incoming video so that 720 active pixels are always generated per line. Note that no user I^2C settings are available for the PLL when in SDP mode as the PLL is controlled directly by the SDP.

- When the CP is enabled, true line locked sampling is applied to the video signal being processed. This means that the horizontal synchronization signal of the incoming video signal is applied to the PLL and multiplied up by the desired number of samples per line, which yields the pixel sampling clock used in CP mode.

The CP ADC sampling clock is a line locked clock that is generated automatically by a digital encoder synthesizer. The following controls enable the user to adjust the ADC sampling clock:

- PLL_DIV_MAN_EN
- PLL_DIV_RATIO[12:0]
- DLL_PHASE[5:0]

6.2 ADCS AND VOLTAGE CLAMPS

6.2.1 Analog Input Hardware Configuration

The ADV7842 supports 12 analog inputs. The analog inputs have an input range of 0 to 1 V.

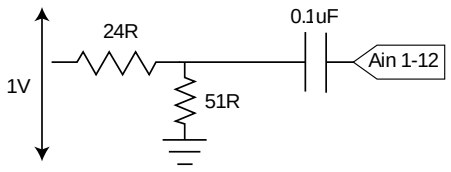


Figure 9: Analog Inputs Hardware Configuration

6.2.2 Clamp Operation

The ADV7842 has a clamp in front of each of its ADCs. The purpose of the clamp is to ensure that the video input signal lies inside the range of the ADC. In component and graphics modes, voltage clamps are used; and in standard definition modes, current clamps are used. Clamping is done automatically and does not require user programming. Optimum performance is selected by default. Figure 9 shows the resistor divider network required on each analog input to scale the input video to the input range of the ADCs.

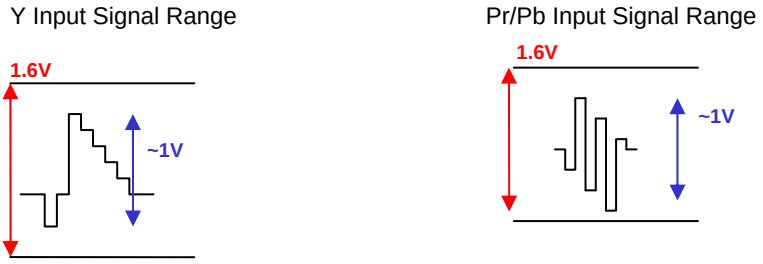


Figure 10: Video Input Signal Level Prior to 24 Ohm to 51 Ohm Resistor Divider

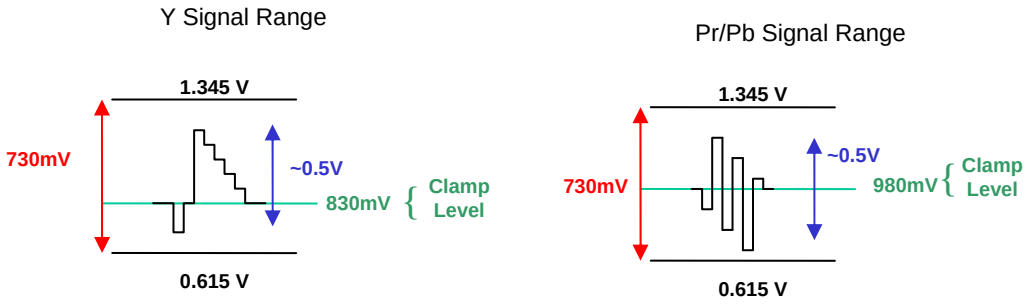


Figure 11: Video Input Signal Level After Voltage Clamps

6.2.3 SDP Clamp Operation

Standard definition video signals can have excessive noise on them; especially CVBS signals transmitted by terrestrial broadcast and demodulated using a tuner. These usually show very large levels of noise. A voltage clamp would be unsuitable for this type of video signal. Instead, the ADV7842 employs a set of current sources that can cause currents to flow into and away from the high impedance node that carries the video signal (refer to Figure 12).

Since the input video is AC coupled into the ADV7842, its DC value needs to be restored. This process is referred to as clamping the video. This section explains the general process of clamping on the ADV7842 for the SDP and shows the different ways in which a user can configure its behavior.

The SDP block uses a combination of current sources and a digital processing block for clamping, as shown in Figure 12.

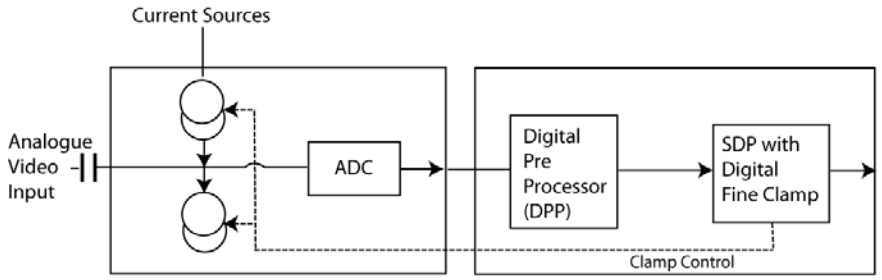


Figure 12: SDP Clamping Overview

The clamping can be divided into two sections:

- Clamping before the ADC (analog domain): digitally controlled current sources.
- Clamping after the ADC (digital domain): digital processing block.

The ADCs can digitize an input signal if it resides within the ADC input voltage range of 1.0 V. An input signal with a DC level that is too large or too small will be clipped at the top or bottom of the ADC range.

The primary task of the analog clamping circuits is to ensure that the video signal stays within the valid ADC input window so that the analog-to-digital conversion can take place. After digitization, the digital fine clamp block corrects for any remaining variations in DC level. Since the DC level of an input video signal refers directly to the brightness of the picture transmitted, it is important to perform a fine clamp with high accuracy, otherwise brightness variations can occur.

This section describes the I^2C signals used to influence the behavior of the SDP clamping.

SDP_DCLP_SPEED[4:0], Addr 90 (SDP), *Address 0x0C[4:0]*

Control to adjust the speed of digital clamp operation

Function

SDP_DCLP_SPEED[4:0]	Description
0x00	Freeze digital gain
0x05 «	Default value
0x06	Max value
>0x06	Reserved

SDP_DCLP_SPEED[4:0] determines the time constant of the digital clamp circuitry. It is important to realize that the digital clamp reacts very fast to correct immediately any residual DC level error for the active line. The time constant of the digital clamp must be a lot quicker than the one from the analog blocks.

This register also allows the user to freeze the digital clamp loop at any point in time.

By default, the time constant of the digital clamp is adjusted dynamically to suit the currently connected input signal.

SDP_ACLP_SPEED[4:0], Addr 90 (SDP), *Address 0x0D[4:0]*

A control to adjust the speed of the analog clamp operation.

Function

SDP_ACLP_SPEED[4:0]	Description
0x00	Freeze analog clamp
0x04 «	Default value
0x06	Max value
>0x06	Reserved

SDP_ACLP_SPEED[4:0] determines the time constant of the analog clamp circuitry. It is important to realize that the analog clamp reacts quickly to correct any residual DC level error for the active line.

This register also allows the user to freeze the analog clamp loop at any point in time.

By default, the time constant of the digital clamp is adjusted dynamically to suit the currently connected input signal.

6.3 ANALOG INPUT MUXING

The ADV7842 has twelve analog input pins, Ain1 to Ain12. The user must select the Ain pin signals are routed to the ADC in order to process the video signals that appear on the analog inputs. The ADV7842 has an integrated analog muxing section to route the video signals to the ADCs. This allows more than one source of video signal to be connected to the decoder and routes the desired video signals to the ADCs. A selection of predefined routing options are available and are controlled by the AIN_SEL[2:0]; this is referred to as automatic input muxing selection. It is also possible to manually select routing of analog inputs to individual ADCs; this is referred to as manual input muxing.

6.3.1 Analog Input Routing Recommendation

ADI has specific Ain pin recommendations for specific video processing modes in order to ensure the best performance. ADI recommends the following:

- RGB Graphics – Ain 1, 2, 3
- Component – Ain 4, 5, 6
- SCART RGB – Ain 7, 8, 9
- SCART CVBS – Ain 10
- CVBS – Ain11
- YC – Ain 10, 12

Refer to Appendix A for the recommended board layout.

Figure 13 proposes three setup configurations that can be used in various geographical regions:

- Japan
- Europe
- North America

The D-terminal is predominantly used in Japan for the transfer of analog component video. North America uses the phono/RCA type

component connecters while the SCART connector is popular in Europe. The VGA graphics connector for PC hook-up is common to all three areas.

The ADV7842 is suited to interface to all the connector types mentioned above. It also has the ability to handle special bilevel and trilevel signaling that the D-terminal and SCART connectors support.

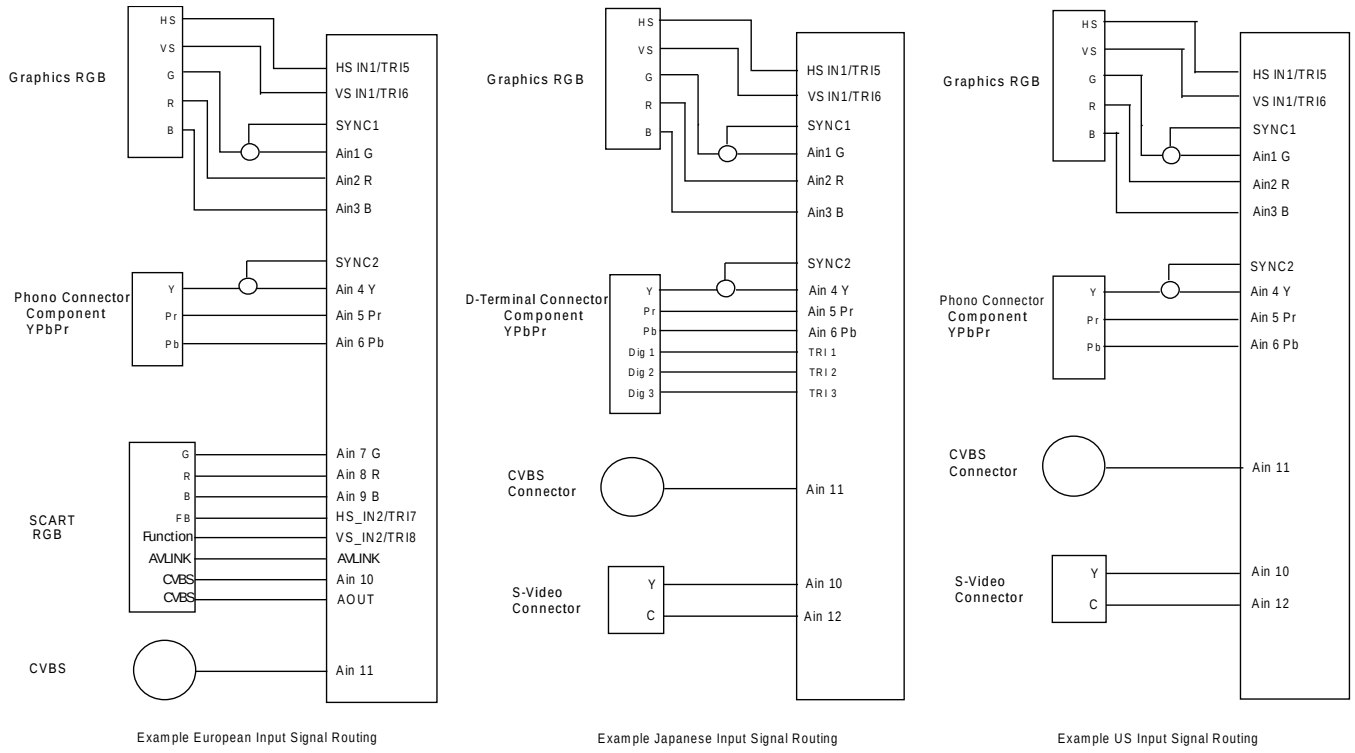


Figure 13: ADV7842 Typical Configurations

6.4 AUTOMATIC INPUT MUXING SELECTION

By selecting the various AIN_SEL[2:0] values, the user can direct the predefined analog input to ADC routing options.

The AIN_SEL also controls the routing of embedded synchronization on the SYNC pins to the embedded synchronization slicers.

Embedded synchronization refers to a synchronization signal that is 'embedded' in the input video signal itself, for example, SOG (Sync on green) and SOY (Sync on Y) type signals.

Note that CVBS, YC and SD components have to be manually routed using ADC0_SW_MAN[3:0], ADC1_SW_MAN[3:0], ADC2_SW_MAN[3:0], ADC3_SW_MAN[3:0] while ADC_SWITCH_MAN = 1. Refer to Section 6.5.1.

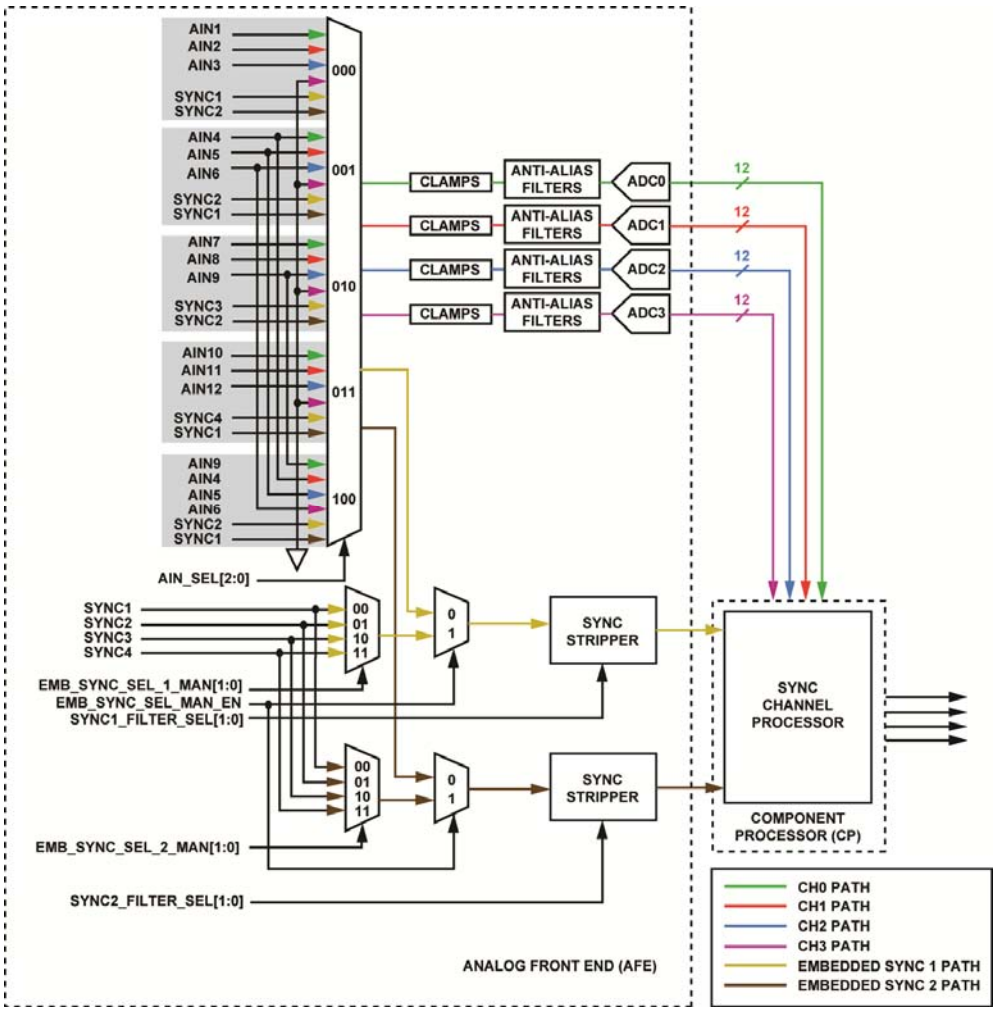


Figure 14: ADV7842 Input Functional Diagram

AIN_SEL[2:0], Addr 4C (AFE), *Address 0x02[2:0]*

Input Muxing Mode

Function

AIN_SEL[2:0]	Description					
Code	ADC0	ADC1	ADC2	ADC3	EMB_SYNC_SEL1	EMB_SYNC_SEL2
000 «	Ain1	Ain2	Ain3	NC	Sync1	Sync2
001	Ain4	Ain5	Ain6	NC	Sync2	Sync1
010	Ain7	Ain8	Ain9	NC	Sync3	Sync1
011	Ain10	Ain11	Ain12	NC	Sync4	Sync1
100	Ain9	Ain4	Ain5	Ain6	Sync2	Sync1

The EMB_SYNC_SEL_1 signal should be considered as the main synchronization input for this muxing arrangement, and the EMB_SYNC_SEL_2 signal will be monitored in parallel through the second synchronization stripper.

6.5 MANUAL INPUT MUXING OVERVIEW

By accessing a set of manual override muxing registers, the analog input muxes of the ADV7842 can be controlled directly. This is referred to as Manual Input Muxing.

Notes:

- Manual input muxing overrides **all** other input muxing control bits, for example, AIN_SEL[2:0]. The manual muxing is activated by setting the ADC_SWITCH_MAN. It only affects the analog switches in front of the ADCs. PRIM_MODE[3:0] and VID_STD[5:0] still have to be set so the follow on blocks process the video data in the correct format.
- Manual input muxing **only** controls the analog input muxes.
- The ADI recommended input combinations are designed to minimize crosstalk between input channels. When using the manual input muxing, special care must be taken by the user/PCB designer to take care of cross coupling.

Not every input pin can be routed to any ADC. There are restrictions in the channel routing imposed by the analog signal routing inside the IC. This is described in Section 6.5.1.

6.5.1 Manual Input Muxing

The manual muxing controls listed in this section configure the first six analog inputs. The following controls should be used to override the auto configuration described in Section 6.4.

Note that when selecting analog input routing to the ADCs the following rules must be adhered to:

- The CVBS, Y or G signal must be routed to ADC0
- The C signal in S-Video(YC) mode must be routed to ADC2

Refer to Table 7 for more information.

ADC_SWITCH_MAN, Addr 4C (AFE), *Address 0x02[7]*

A control to enable manual input muxing to the ADCs.

Function

ADC_SWITCH_MAN	Description
0 «	Automatic Muxing
1	Manual Muxing

Once ADC_SWITCH_MAN is set to 1, the following controls take effect.

ADC0_SW_MAN[3:0], Addr 4C (AFE), *Address 0x03[7:4]*

ADC0 Manual Input Muxing. A control to manually route analog inputs to ADC0.

Function

ADC0_SW_MAN[3:0]	Description
0001	Ain1
0100	Ain4
0111	Ain7
1001	Ain9
1010	Ain10
1011	Ain11
All Other Values	Not Connected

ADC1_SW_MAN[3:0], Addr 4C (AFE), *Address 0x03[3:0]*

ADC1 Manual Input Muxing. A control to manually route analog inputs to ADC 1.

Function

ADC1_SW_MAN[3:0]	Description
0010	Ain2
0100	Ain4
0101	Ain5
1000	Ain8
1011	Ain11
All Other Values	Not Connected

ADC2_SW_MAN[3:0], Addr 4C (AFE), *Address 0x04[7:4]*

ADC2 Manual Input Muxing. A control to manually rout analog inputs to ADC2

Function

ADC2_SW_MAN[3:0]	Description
0011	Ain3
0100	Ain4
0101	Ain5
0110	Ain6
1001	Ain9
1100	Ain12
All Other Values	Not Connected

ADC3_SW_MAN[3:0], Addr 4C (AFE), *Address 0x04[3:0]*

ADC3 Manual Input Muxing. A control to manually rout analog inputs to ADC3

Function

ADC3_SW_MAN[3:0]	Description
0100	Ain4
0110	Ain6
0111	Ain7
All Other Values	Not Connected

Table 6: Manual Input Muxing

ADC0_SW_MAN[3:0], ADC1_SW_MAN[3:0], ADC2_SW_MAN[3:0], ADC3_SW_MAN[3:0]	ADC0 Connected to	ADC1 Connected to	ADC2 Connected to	ADC3 Connected to
0000	-	-	-	-
0001	AIN1	-	-	-

ADC0_SW_MAN[3:0], ADC1_SW_MAN[3:0], ADC2_SW_MAN[3:0], ADC3_SW_MAN[3:0]	ADC0 Connected to	ADC1 Connected to	ADC2 Connected to	ADC3 Connected to
0010	-	AIN2	-	-
0011	-	-	AIN3	-
0100	AIN4	AIN4	AIN4	AIN4
0101	-	AIN5	AIN5	-
0110	-	-	AIN6	AIN6
0111	AIN7	-	-	AIN7
1000	-	AIN8	-	-
1001	AIN9	-	AIN9	-
1010	AIN10	-	-	-
1011	AIN11	AIN11	-	-
1100	-	-	AIN12	-
1101	-	-	-	-
1110	-	-	-	-
1111	-	-	-	-

ADC Mapping

Table 7: Recommended Video Signal to ADC Routing

Mode	Required ADC Mapping	ADI Recommended Ain Channel
CVBS	CVBS = ADC0	CVBS = Ain 11
YC/YC auto	Y = ADC0	Y= Ain 10
	C = ADC2	C = Ain12
SCART RGB[1]	CBVS = ADC0	CVBS = Ain 10
	G = ADC3	G = Ain 7
	B = ADC2	B = Ain 8
	R = ADC1	R = Ain 9
CP YPrPb	Y = ADC0	G = Ain 4
	Pr = ADC1	R = Ain 5
	Pb = ADC2	B = Ain 6
Graphics RGB	G = ADC0	G = Ain 1
	B = ADC2	R = Ain 2
	R = ADC1	B = Ain 3

6.6 SYNC1-4 INPUT CONTROL

As shown in Figure 15, the ADV7842 has four input synchronization control pins: SYNC1, SYNC2, SYNC3 and SYNC4. These pins are used for signals with embedded synchronization, for example, Sync on Green (SOG) or Sync On Y (SOY) type signals . SOG is associated with RGB input video; SOY is associated with component YPrPb input video. These synchronization inputs can be muxed to two synchronization strippers also in the AFE. From here, the signals are routed to synchronization processors in the CP. This muxing can be performed in an automatic configuration mode or with manual control. These are described in Sections 6.6.1 and 6.6.2 respectively.

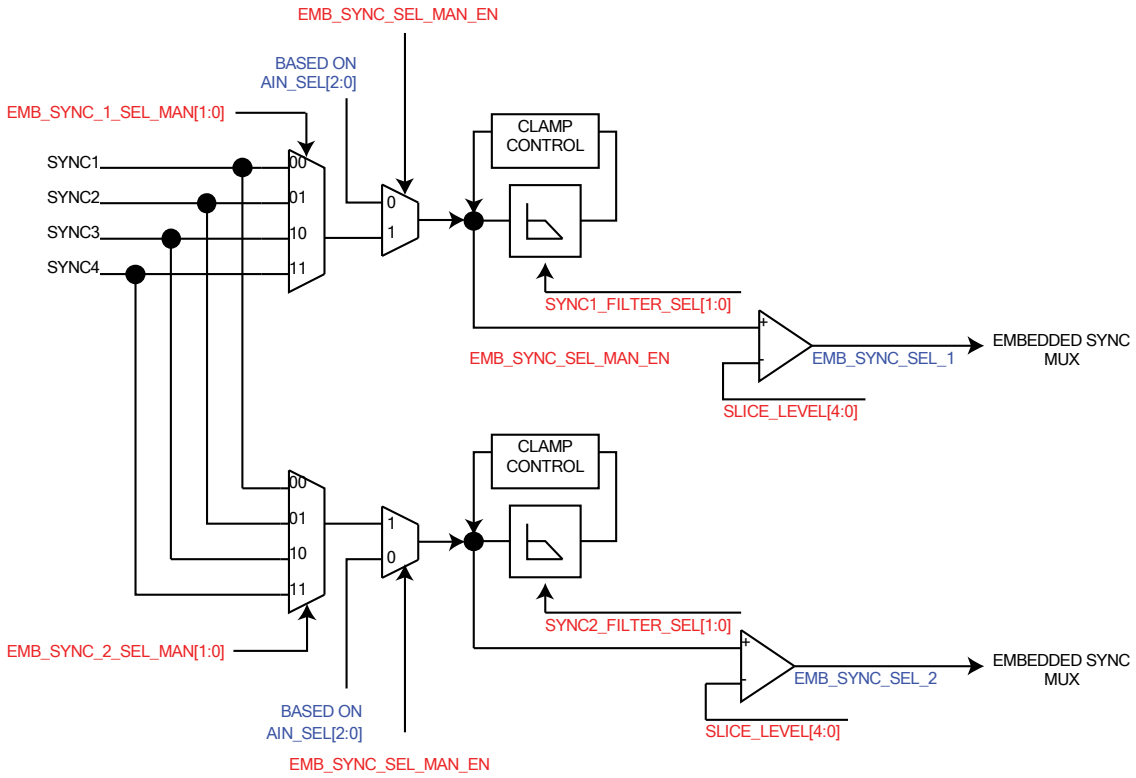


Figure 15: Synchronization Stripper Circuit

6.6.1 Automatic Synchronization Configuration

In addition to configuring the analog input muxes, AIN_SEL[2:0]controls automatically the associated synchronization channel routing. Refer to Figure 13 and the AIN_SEL[2:0] description for the routing details when AIN_SEL[2:0] is programmed.

6.6.2 Manual Synchronization Configuration

Figure 14 shows the input synchronization muxes, which can be controlled manually as well as automatically. Manual control is enabled via EMB_SYNC_SEL_MAN_EN, and EMB_SYNC_1_SEL_MAN[1:0] and EMB_SYNC_2_SEL_MAN[1:0] select the SYNC source for each embedded sync processor. This section describes the control bits for manual operation.

EMB_SYNC_SEL_MAN_EN, Addr 4C (AFE), *Address 0x02[6]*

Enable manual selection of embedded synchronization inputs to synchronization strippers. In automatic mode AIN_SEL[3:0] makes the selection. In manual mode EMB_SYNC_1_SELMAN[1:0] and EMB_SYNC_2_SELMAN[1:0] makes the selection.

Function

EMB_SYNC_SEL_MAN_EN	Description
0 «	Automatic Sync Selection
1	Manual Sync Selection

EMB_SYNC_1_SEL_MAN[1:0], Addr 4C (AFE), *Address 0x15[7:6]*

Manual embedded synchronization selection for EMB_SYNC1

Function

EMB_SYNC_1_SEL_MAN [1:0]	Description
00 «	Sync1 pin
01	Sync2 pin
10	Sync3 pin
11	Sync4 pin

EMB_SYNC_2_SEL_MAN[1:0], Addr 4C (AFE), *Address 0x15[5:4]*

Manual embedded synchronization selection for EMB_SYNC2

Function

EMB_SYNC_2_SEL_MAN [1:0]	Description
00 «	Sync1 pin
01	Sync2 pin
10	Sync3 pin
11	Sync4 pin

6.7 SYNCHRONIZATION SLICERS

The ADV7842 has two synchronization slicer blocks, which are placed before the synchronization processing sections, as shown in Figure 14. The purpose of a synchronization slicer is to provide a reliable synchronization signal to the STDI and SSPD circuits in the component processor so that a robust identification of standard is made. A second synchronization slicer is provided to allow processing of a second or, possibly, a third channel.

A more in depth picture of Sync Slicer 1 is shown in Figure 15. The circuit is duplicated for Sync Slicer 2.

6.7.1 Synchronization Filter Stage

There are two synchronization strippers in the ADV7842 similar to that shown in Figure 15. Any of the four SYNC pins; SYNC 1, SYNC 2, SYNC 3 and SYNC 4 can be routed to the filter stage.

SYNC1_FILTER_SEL[1:0], Addr 4C (AFE), *Address 0x15[3:2]*

Select the clamp filter on the Sync Channel 1

Function

SYNC1_FILTER_SEL[1:0]	Description
00	No filter
01	Sync > 250ns
10 «	Sync > 1us
11	Sync > 2.5us

SYNC2_FILTER_SEL[1:0], Addr 4C (AFE), *Address 0x15[1:0]*

Select the clamp filter on the Sync Channel 2

Function

SYNC2_FILTER_SEL[1:0]	Description
00	No filter
01	Sync > 250ns
10 «	Sync > 1us
11	Sync > 2.5us

6.7.2 Sync Stripper Slice Level

A comparator stage is located after the filter stage. This has programmable thresholds, which offer the user various slice levels that determine the presence of a valid synchronization pulse. The register SLICE_LEVEL[4:0] is a 5-bit register that sets up slice levels for both Sync Stripper 1 **and** Sync Stripper 2.

SLICE_LEVEL[4:0], Addr 4C (AFE), *Address 0x16[4:0]*

Set the slice level in the synchronization strippers. A smaller value corresponds to a higher slice level.
For clamp at 300mV slice level is equal to 600mV - ((SLICE_LEVEL + 1) * 9.375mV).

Function

SLICE_LEVEL[4:0]	Description
00000	Highest slice level
XXXXX	Clamp at 300mV and slice at 600mV - ((XXXXX + 1) * 9.375mV)
11000 «	Default value
11111	Lowest slice Level

6.8 TRILEVEL SLICERS

Several video connectors and cables have signals that are structured as three-level signals. These include the European SCART connector, which has two such signals; and the Japanese D-connector, which has three. These signals convey much information about the parameters of the signal being sent. In order for the ADV7842 to use this information, it must slice the voltage levels appearing on TRI1-TRI8.
The following eight pins are capable of slicing a trilevel signal in the ADV7842:

- TRI1
- TRI2
- TRI3
- TRI4
- HS_IN1/TRI5 (known also as TRI5)
- VS_IN2/TRI6 (known also as TRI6)
- HS_IN2/TRI7 (known also as TRI7)
- VS_IN2/TRI8 (known also as TRI8)

6.8.1 D-Terminal Connector

Table 8 and Table 9 show details of the D-terminal connector used in Japan. The Data Line 1, Data Line 2, and Data Line 3 signals are the three signals that can be applied to any of the eight TRI inputs. The ADV7842 can process two D-terminal connectors. In this case, six of the available eight TRI inputs would be utilized.

	Voltage	Voltage		
A	0 V	0 V	525 lines	4:3
B	1.4 V	2.4 V	750 lines	4:3 letterbox
C	3.5 V	5 V	1125 lines	16:9

Table 9: D-Terminal Connector Characteristics (Bilevel)

Level	Typical Voltage	Pin 9 (Dataline 2)
A	0 V	59.94i/60i
B	5 V	59.94p/60p

6.8.2 TRI 1-8 Input Resistor Selection

As can be seen from Table 8, Table 9, Table 10, and Table 11, the voltage levels to be sliced exceed the power supplies of the ADV7842 and, therefore, are beyond the range of the ADV7842 TRI inputs. The applied signals need to be reduced to fit in the range of the slicers. This is done by utilizing resistor divider networks at the inputs to TRI1-9.

The recommended resistor values for voltages related to the D-terminal connector are shown in Figure 16.

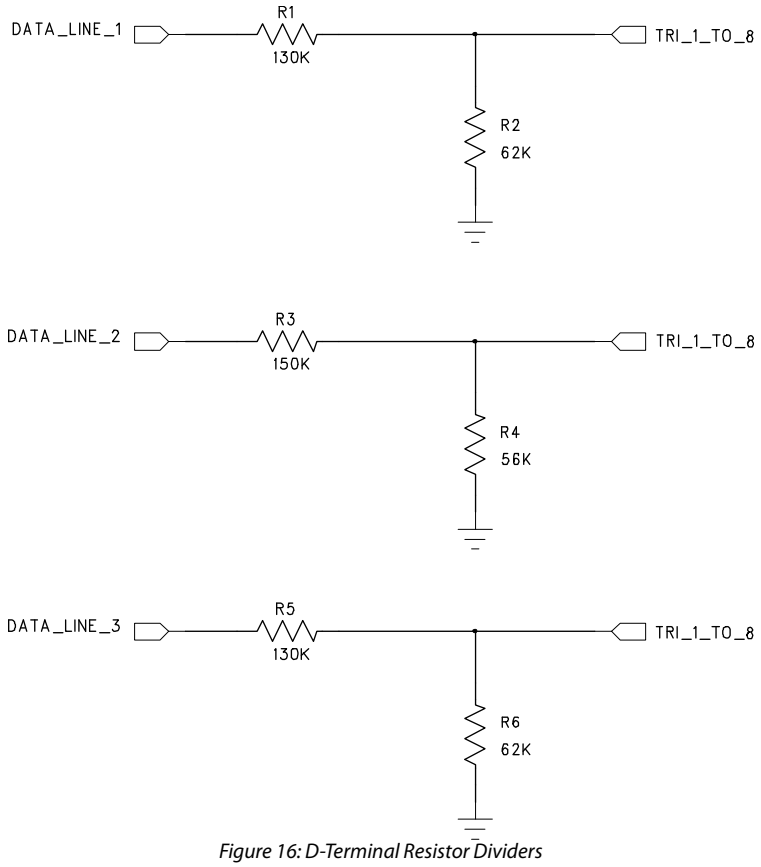


Figure 16: D-Terminal Resistor Dividers

6.8.3 Trilevel Input Controls

The ADV7842 has eight trilevel slicers, as shown in Figure 17. Each trilevel slicer is capable of operating in two modes. The first mode is bilevel mode where the input signal can be sliced at a single voltage level to determine what is the voltage level (refer to Table 9). The second mode is trilevel mode. This mode can be used to determine voltage level of trilevel signals that are present from inputs such as D-

terminal connectors.

All levels are programmable. The outputs from these slicers are available via readback registers. The output from these slicers are also sent to a digital processor in the AFE that generates a system interrupt if any inputs to the TRI 1-8 pins change state. All interrupts can be enabled and serviced via I²C.

Note that a trilevel input pin may also be configured as a fast blank control pin for SCART functionality. Refer to Section 6.9.

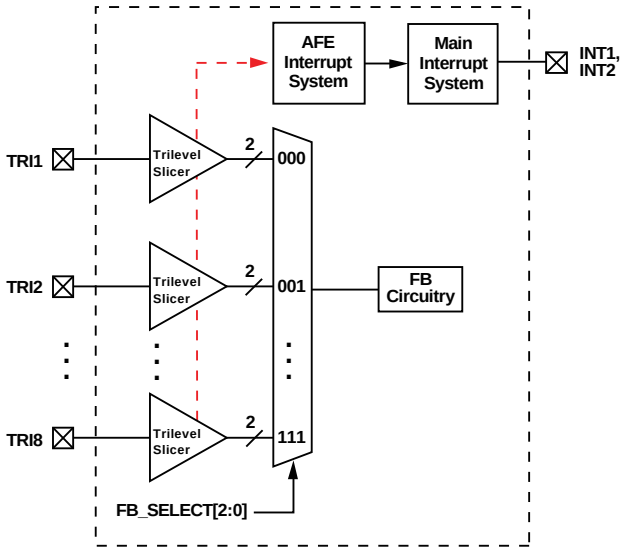


Figure 17: Trilevel Slicer

6.8.4 Trilevel Slicer Operation

Each trilevel slicer can be powered up or down.

TRI1_SLICER_PWRDN, Addr 4C (AFE), *Address 0x1D[6]*

Powerdown the Tri1 slicer

Function

TRI1_SLICER_PWRDN	Description
0	Powered up
1 «	Powerdown down

TRI2_SLICER_PWRDN, Addr 4C (AFE), *Address 0x1E[6]*

Powerdown the Tri2 slicer

Function

TRI2_SLICER_PWRDN	Description
0	Powered up
1 «	Powerdown down

TRI3_SLICER_PWRDN, Addr 4C (AFE), *Address 0x1F[6]*

Powerdown the Tri3 slicer

Function

TRI3_SLICER_PWRDN	Description
0	Powered up
1 «	Powerdown down

TRI4_SLICER_PWRDN, Addr 4C (AFE), *Address 0x20[6]*

Powerdown the Tri4 slicer

Function

TRI4_SLICER_PWRDN	Description
0	Powered up
1 «	Powerdown down

TRI5_SLICER_PWRDN, Addr 4C (AFE), *Address 0x21[6]*

Powerdown the Tri5 slicer

Function

TRI5_SLICER_PWRDN	Description
0	Powered up
1 «	Powerdown down

TRI6_SLICER_PWRDN, Addr 4C (AFE), *Address 0x22[6]*

Powerdown the Tri6 slicer

Function

TRI6_SLICER_PWRDN	Description
0	Powered up
1 «	Powerdown down

TRI7_SLICER_PWRDN, Addr 4C (AFE), *Address 0x23[6]*

Powerdown the Tri7 slicer

Function

TRI7_SLICER_PWRDN	Description
0	Powered up
1 «	Powerdown down

TRI8_SLICER_PWRDN, Addr 4C (AFE), *Address 0x24[6]*

Powerdown the Tri8 slicer

Function

TRI8_SLICER_PWRDN	Description
0	Powered up
1 «	Powerdown down

6.8.5 Bilevel/Trilevel Selection

Each slicer operates in trilevel slice mode by default. The following controls are used to switch between bilevel mode and trilevel mode.

TRI1_BILEVEL_SLICE_EN, Addr 4C (AFE), *Address 0x1D[5]*

Enable bi-level slicing on Tri1 input

Function

TRI1_BILEVEL_SLICE_EN	Description
0	Bilevel slicing
1 «	Trilevel slicing

TRI2_BILEVEL_SLICE_EN, Addr 4C (AFE), *Address 0x1E[5]*

Enable bi-level slicing on Tri2 input

Function

TRI2_BILEVEL_SLICE_EN	Description
0	Bilevel slicing
1 «	Trilevel slicing

TRI3_BILEVEL_SLICE_EN, Addr 4C (AFE), *Address 0x1F[5]*

Enable bi-level slicing on Tri3 input

Function

TRI3_BILEVEL_SLICE_EN	Description
0	Bilevel slicing
1 «	Trilevel slicing

TRI4_BILEVEL_SLICE_EN, Addr 4C (AFE), *Address 0x20[5]*

Enable bi-level slicing on Tri4 input

Function

TRI4_BILEVEL_SLICE_EN	Description
0	Bilevel slicing
1 «	Trilevel slicing

TRI5_BILEVEL_SLICE_EN, Addr 4C (AFE), *Address 0x21[5]*

Enable bi-level slicing on Tri5 input

Function

TRI5_BILEVEL_SLICE_EN	Description
0	Bilevel slicing
1 «	Trilevel slicing

TRI6_BILEVEL_SLICE_EN, Addr 4C (AFE), *Address 0x22[5]*

Enable bi-level slicing on Tri6 input

Function

TRI6_BILEVEL_SLICE_EN	Description
0	Bilevel slicing
1 «	Trilevel slicing

TRI7_BILEVEL_SLICE_EN, Addr 4C (AFE), *Address 0x23[5]*

Enable bi-level slicing on Tri7 input

Function

TRI7_BILEVEL_SLICE_EN	Description
0	Bilevel slicing
1 «	Trilevel slicing

TRI8_BILEVEL_SLICE_EN, Addr 4C (AFE), *Address 0x24[5]*

Enable bi-level slicing on Tri8 input

Function

TRI8_BILEVEL_SLICE_EN	Description
0	Bilevel slicing
1 «	Trilevel slicing

6.8.6 Trilevel Slicer Readbacks

The input has two comparators on it for each trilevel slicer (refer to Figure 17). The raw results of these comparators are available via the TRIx_READBACK[1:0] registers. Upper level in descriptions below refers to TRIx_UPPER_SLICE_LEVEL (where x is 1 to 8). Lower level in descriptions below refers to TRIx_LOWER_SLICE_LEVEL (where x is 1 to 8).

TRI1_READBACK[1:0], Addr 4C (AFE), *Address 0x27[7:6] (Read Only)*

Readback Tri1 DC levels

Function

TRI1_READBACK[1:0]	Description
1x	Signal is higher than upper level
0x	Signal is lower than upper level
x1	Signal is higher than lower level
x0	Signal is lower than lower level

TRI2_READBACK[1:0], Addr 4C (AFE), *Address 0x27[5:4] (Read Only)*

Readback Tri2 DC levels

Function

TRI2_READBACK[1:0]	Description
1x	Signal is higher than upper level
0x	Signal is lower than upper level
x1	Signal is higher than lower level
x0	Signal is lower than lower level

TRI3_READBACK[1:0], Addr 4C (AFE), *Address 0x27[3:2] (Read Only)*

Readback Tri3 DC levels

Function

TRI3_READBACK[1:0]	Description
1x	Signal is higher than upper level
0x	Signal is lower than upper level
x1	Signal is higher than lower level
x0	Signal is lower than lower level

TRI4_READBACK[1:0], Addr 4C (AFE), *Address 0x27[1:0] (Read Only)*

Readback Tri4 DC levels

Function

TRI4_READBACK[1:0]	Description
1x	Signal is higher than upper level
0x	Signal is lower than upper level
x1	Signal is higher than lower level
x0	Signal is lower than lower level

TRI5_READBACK[1:0], Addr 4C (AFE), *Address 0x28[7:6] (Read Only)*

Readback Tri5 DC levels

Function

TRI5_READBACK[1:0]	Description
1x	Signal is higher than upper level
0x	Signal is lower than upper level
x1	Signal is higher than lower level
x0	Signal is lower than lower level

TRI6_READBACK[1:0], Addr 4C (AFE), *Address 0x28[5:4] (Read Only)*

Readback Tri6 DC levels

Function

TRI6_READBACK[1:0]	Description
1x	Signal is higher than upper level
0x	Signal is lower than upper level
x1	Signal is higher than lower level
x0	Signal is lower than lower level

TRI7_READBACK[1:0], Addr 4C (AFE), *Address 0x28[3:2] (Read Only)*

Readback Tri7 DC levels

Function

TRI7_READBACK[1:0]	Description
1x	Signal is higher than upper level
0x	Signal is lower than upper level
x1	Signal is higher than lower level
x0	Signal is lower than lower level

TRI8_READBACK[1:0], Addr 4C (AFE), *Address 0x28[1:0] (Read Only)*

Readback Tri8 DC levels

Function

TRI8_READBACK[1:0]	Description
1x	Signal is higher than upper level
0x	Signal is lower than upper level
x1	Signal is higher than lower level
x0	Signal is lower than lower level

6.8.7 Programming Trilevel Slicers

When the slicers are in a bilevel mode of operation (refer to Section 6.8.5), the upper slicer is utilized. This offers the programmability described in Section 6.8.7.1. When any of the trilevel slicer circuits are in trilevel mode (refer to Section 6.8.5), the upper levels (UL) are sliced by the UPPER_SLICER, as described in Section 6.8.7.1. The lower levels are sliced by the LOWER_SLICER (LL), as described in Section 6.8.7.2).

6.8.7.1 Upper Slice Levels

TRI1_UPPER_SLICE_LEVEL[2:0], Addr 4C (AFE), *Address 0x1D[4:2]*

Set the upper slice level on the Tri1 input

Function

TRI1_UPPER_SLICE_LEVEL[2:0]	Description
000	75mV
001	225mV
010	375mV
011 «	525mV
100	675mV
101	825mV
110	975mV
111	1.125V

TRI2_UPPER_SLICE_LEVEL[2:0], Addr 4C (AFE), *Address 0x1E[4:2]*

Set the upper slice level on the Tri2 input

Function

TRI2_UPPER_SLICE_LEVEL[2:0]	Description
000	75mV
001	225mV
010	375mV
011 «	525mV
100	675mV
101	825mV
110	975mV
111	1.125V

TRI3_UPPER_SLICE_LEVEL[2:0], Addr 4C (AFE), *Address 0x1F[4:2]*

Set the upper slice level on the Tri3 input

Function

TRI3_UPPER_SLICE_LEVEL[2:0]	Description
000	75mV
001	225mV
010	375mV
011 «	525mV
100	675mV
101	825mV
110	975mV
111	1.125V

TRI4_UPPER_SLICE_LEVEL[2:0], Addr 4C (AFE), *Address 0x20[4:2]*

Set the upper slice level on the Tri4 input

Function

TRI4_UPPER_SLICE_LEVEL[2:0]	Description
000	75mV
001	225mV
010	375mV
011 «	525mV
100	675mV
101	825mV
110	975mV
111	1.125V

TRI5_UPPER_SLICE_LEVEL[2:0], Addr 4C (AFE), *Address 0x21[4:2]*

Set the upper slice level on the Tri5 input

Function

TRI5_UPPER_SLICE_LEVEL[2:0]	Description
000	75mV
001	225mV
010	375mV
011 «	525mV
100	675mV
101	825mV
110	975mV
111	1.125V

TRI6_UPPER_SLICE_LEVEL[2:0], Addr 4C (AFE), *Address 0x22[4:2]*

Set the upper slice level on the Tri6 input

Function

TRI6_UPPER_SLICE_LEVEL[2:0]	Description
000	75mV
001	225mV
010	375mV
011 «	525mV
100	675mV
101	825mV
110	975mV
111	1.125V

TRI7_UPPER_SLICE_LEVEL[2:0], Addr 4C (AFE), *Address 0x23[4:2]*

Set the upper slice level on the Tri7 input

Function

TRI7_UPPER_SLICE_LEVEL[2:0]	Description
000	75mV
001	225mV
010	375mV
011 «	525mV
100	675mV
101	825mV
110	975mV
111	1.125V

TRI8_UPPER_SLICE_LEVEL[2:0], Addr 4C (AFE), *Address 0x24[4:2]*

Set the upper slice level on the Tri8 input

Function

TRI8_UPPER_SLICE_LEVEL[2:0]	Description
000	75mV
001	225mV
010	375mV
011 «	525mV
100	675mV
101	825mV
110	975mV
111	1.125V

6.8.7.2 Lower Slice Levels

TRI1_LOWER_SLICE_LEVEL[1:0], Addr 4C (AFE), *Address 0x1D[1:0]*

Set the lower slice level on the Tri1 input

Function

TRI1_LOWER_SLICE_LEVEL[1:0]	Description
00	75mV
01 «	225mV
10	375mV
11	525mV

TRI2_LOWER_SLICE_LEVEL[1:0], Addr 4C (AFE), *Address 0x1E[1:0]*

Set the lower slice level on the Tri2 input

Function

TRI2_LOWER_SLICE_LEVEL[1:0]	Description
00	75mV
01 «	225mV
10	375mV
11	525mV

TRI3_LOWER_SLICE_LEVEL[1:0], Addr 4C (AFE), *Address 0x1F[1:0]*

Set the lower slice level on the Tri3 input

Function

TRI3_LOWER_SLICE_LEVEL[1:0]	Description
00	75mV
01 «	225mV
10	375mV
11	525mV

TRI4_LOWER_SLICE_LEVEL[1:0], Addr 4C (AFE), *Address 0x20[1:0]*

Set the lower slice level on the Tri4 input

Function

TRI4_LOWER_SLICE_LEVEL[1:0]	Description
00	75mV
01 «	225mV
10	375mV
11	525mV

TRI5_LOWER_SLICE_LEVEL[1:0], Addr 4C (AFE), *Address 0x21[1:0]*

Set the lower slice level on the Tri5 input

Function

TRI5_LOWER_SLICE_LEVEL[1:0]	Description
00	75mV
01 «	225mV
10	375mV
11	525mV

TRI6_LOWER_SLICE_LEVEL[1:0], Addr 4C (AFE), *Address 0x22[1:0]*

Set the lower slice level on the Tri6 input

Function

TRI6_LOWER_SLICE_LEVEL[1:0]	Description
00	75mV
01 «	225mV
10	375mV
11	525mV

TRI7_LOWER_SLICE_LEVEL[1:0], Addr 4C (AFE), *Address 0x23[1:0]*

Set the lower slice level on the Tri7 input

Function

TRI7_LOWER_SLICE_LEVEL[1:0]	Description
00	75mV
01 «	225mV
10	375mV
11	525mV

TRI8_LOWER_SLICE_LEVEL[1:0], Addr 4C (AFE), *Address 0x24[1:0]*

Set the lower slice level on the Tri8 input

Function

TRI8_LOWER_SLICE_LEVEL[1:0]	Description
00	75mV
01 «	225mV
10	375mV
11	525mV

6.9 SCART AND FAST BLANKING

The ADV7842 can support simultaneous processing of CVBS and RGB standard definition signals to enable SCART compatibility and overlay functionality. This is a standard definition mode and is enabled by PRIM_MODE[3:0] and VID_STD[5:0] controls.

In this mode, timing extraction is always performed by the SDP on the CVBS signal. However, a combination of the CVBS and RGB video inputs can be mixed and output on the pixel port. The video selection is under the control of I^2C registers or a fast blank signal applied to one of the trilevel inputs. (The trilevel input is selected using FB_SELECT[3:0]).

By default, the ADV7842 operates in a dynamic switching mode in which the source selection is under the control of the fast blank signal. This enables dynamic multiplexing between the CVBS and RGB sources. When fast blank signal is logic HI, the RGB source is selected; when the fast blank signal is logic LO, the CVBS source is selected. This is suitable for the overlay of subtitles, teletext, or other material. Typically, the CVBS source carries the main picture and the RGB source has the overlay data.

The source selection can also be controlled manually via the I2C registers. Manual source selection mode is enabled using the SDP_MAN_FB_EN bit. In manual mode, either the CVBS content or the RGB content can be output under the control of

SDP_MAN_FB. This mode allows the selection of a full screen picture from either source. Overlay is not possible in manual mode.

FB_SELECT[3:0], Addr 4C (AFE), *Address 0x14[3:0]*

Select the Trilevel input to use as Fast Blank

Function

FB_SELECT[3:0]	Description
0000 «	TRI1
0001	TRI2
0010	TRI3
0011	TRI4
0100	TRI5/HS_IN1
0101	TRI6/VS_IN2
0110	TRI7/HS_IN2
0111	TRI8/VS_IN2
1000	SOG1
1001	SOG2
1010	HSIN1
1011	HSIN2
1100	VSIN1
1101	VSIN2

6.9.1 Fast Blanking Configuration

A block diagram of the ADV7842 fast blanking configuration is shown in Figure 18.

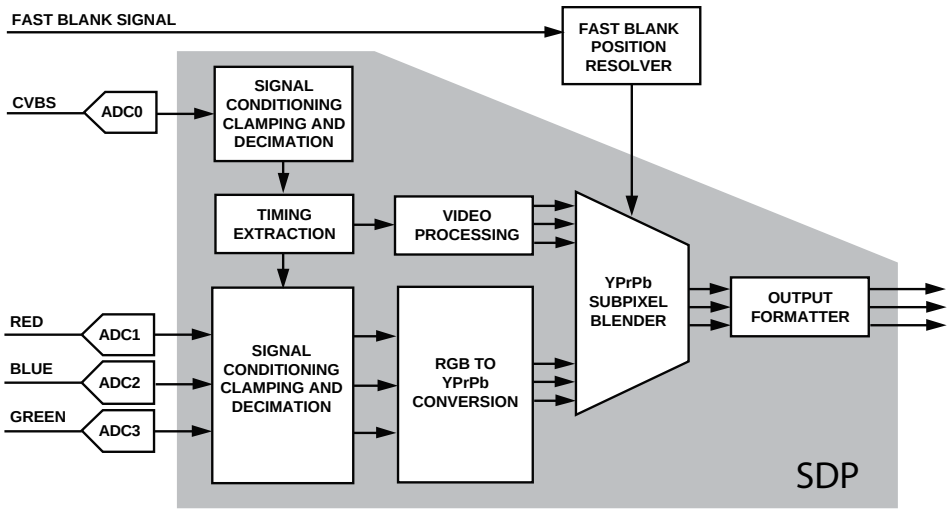


Figure 18: ADV7842 Fast Blanking Configuration

The CVBS signal is processed by the SDP and converted to YPbPr. The RGB signals are also processed by the SDP and are converted to YPbPr. Both sets of YPbPr signals are input to the subpixel blender, which can be configured to operate manually using SDP_MAN_FB control or dynamically using the fast blank input signal.

The fast blank position resolver determines the time position of the fast blank to a very high accuracy and this position information is then used by the subpixel blender in dynamic switching modes. This enables the ADV7842 to implement high performance multiplexing between the CVBS and RGB sources, even when the RGB data source is completely asynchronous to the sampling crystal reference.

The switched or blended data is output from the ADV7842 in the standard formats that exist for the SDP.

6.9.2 SCART Source Selection Control

SDP_MAN_FB_EN, Addr 90 (SDP), *Address 0x2A[3]*

A control to select between manual fast blank control via SDP_MAN_FB and automatic fast blank control via the FB signal (please refer to FB_SELECT in AFE Map).

Function

SDP_MAN_FB_EN	Description
0 «	Auto fast blank controlled by FB signal
1	Allow manual control of FB signal

SDP_MAN_FB, Addr 90 (SDP), *Address 0x2A[7]*

A control to select video source for fast blank operation. This control is only valid SDP_MAN_FB_EN is set to 1.

Function

SDP_MAN_FB	Description
0 «	Select CVBS
1	Select RGB

6.9.3 SCART Fast Blank Timing

The critical information extracted from the SCART fast blank signal is the time at which it switches relative to the input video. Due to small timing inequalities, either on the IC or on the PCB, it may be necessary to adjust the result by fractions of one clock cycle. This is controlled by FB_PHASE_ADJUST[3:0].

Each LSB of FB_PHASE_ADJUST[3:0] corresponds to 1/8 of an ADC clock cycle. Increasing the value is equivalent to adding delay to the SCART fast blank signal. The reset value is chosen to give equalized channels when the ADV7842 internal anti aliasing filters are enabled and there is no unintentional delay on the PCB.

FB_PHASE_ADJUST[3:0], Addr 4C (DPLL), *Address 0xC9[3:0]*

SCART fast blank phase delay adjustment in increments of 1/8th of the ADC clock cycle. The critical information extracted from the SCART fast blank signal is the time at which it switches relative to the input video. Due to small timing inequalities, either on the IC or on the PCB, it may be necessary to adjust the result by fractions of one clock cycle. This is controlled by FB_PHASE_ADJ[3:0].

Function

FB_PHASE_ADJUST[3:0]	Description
1001	Default
xxxx	Adjust the phase in increments of 1/8th of a ADC clock cycle.

The fast blank and RGB signals can also be advanced or delayed by 1, 2, or 3 pixels

SDP_FB_DELAY_ADJ[2:0], Addr 90 (SDP), *Address 0x2A[2:0]*

A signed control to advance or delay for FB signal in increments of one burst-locked pixel

Function

SDP_FB_DELAY_ADJ[2:0]	Description
000 «	No delay.
001	Delay by 1 pixel
010	Delay by 2 pixels.
011	Delay by 3 pixels.
100	No advance.
101	Advance by 1 pixel.
110	Advance by 2 pixels.
111	Advance by 3 pixels.

SDP_RGB_DELAY_ADJ[2:0], Addr 90 (SDP), *Address 0x2A[6:4]*

A signed control to advance or delay for SCART RGB signals in increments of one burst-locked pixel.

Function

SDP_RGB_DELAY_ADJ[2:0]	Description
000 «	No delay.
001	Delay by 1 pixel.
010	Delay by 2 pixels.
011	Delay by 3 pixels.
100	No advance.
101	Advance by 1 pixel.
110	Advance by 2 pixels.
111	Advance by 3 pixels.

6.10 ANTI ALIASING FILTERS

6.10.1 Description

The ADV7842 has optional anti aliasing filters on each of the input channels. The filters are designed for SD, ED, and HD video with various bandwidths selectable via I^2C. These filters are most effective when ADC oversampling is selected.

The filters can be individually enabled and disabled via I^2C under the control of AA_FILTER_EN0, AA_FILTER_EN1, AA_FILTER_EN2 and AA_FILTER_EN3. All filters are disabled when the ADV7842 powers up.

AA_FILTER_EN3, Addr 4C (AFE), *Address 0x05[3]*

ADC3 Anti-Aliasing Filter control.

Function

AA_FILTER_EN3	Description
0 «	Disabled.
1	Enabled.

AA_FILTER_EN2, Addr 4C (AFE), *Address 0x05[2]*

ADC2 Anti-Aliasing Filter control

Function

AA_FILTER_EN2	Description
0 «	Disabled.
1	Enabled.

AA_FILTER_EN1, Addr 4C (AFE), *Address 0x05[1]*

ADC1 Anti-Aliasing Filter control

Function

AA_FILTER_EN1	Description
0 «	Disabled.
1	Enabled.

AA_FILTER_EN0, Addr 4C (AFE), *Address 0x05[0]*

ADC0 Anti-Aliasing Filter control

Function

AA_FILTER_EN0	Description
0 «	Disabled.
1	Enabled.

AA_FILT_PROG_BW and **AA_FILT_HIGH_BW**

AA_FILT_PROG_BW[1:0] combined with the AA_FILT_HIGH_BW[1:0] bits control the anti alias filter response to be selected. The chosen response is applied to all four AA filters simultaneously. Table 10 shows how to choose the various filter characteristics. The 3db cutoff frequency (Fc) is also shown in the table. All possible filter responses are shown in Figure 19.

AA_FILT_HIGH_BW[1:0], Addr 4C (AFE), *Address 0x06[5]; Address 0x07[7]*

Anti Aliasing Filter Bandwidth Control. AA_FILT_PROG_BW combined with AA_FILT_HIGH_BW controls the anti aliasing filter response. Refer to the Anti Alias Filter Frequency Characteristics table.

Function

AA_FILT_HIGH_BW[1:0]	Description
00 «	Default value; pass band < 17 MHz;
01	Pass band < 42 MHz;
10	Pass band < 92 MHz;
11	Pass band < 146 MHz;

AA_FILT_PROG_BW[1:0], Addr 4C (AFE), *Address 0x07[6:5]*

Anti-Alias Filter Bandwidth Control. To be used in conjunction with AA_FILT_HIGH_BW. Please refer to Anti Alias Filter Frequency Characteristics table

Function

AA_FILT_PROG_BW[1:0]	Description
00 «	Default value;

Table 10: Anti Alias Filter Frequency Characteristics

AA_FILT_HIGH_BW[1:0]	AA_FILT_PROG_BW[1:0]	

[1]	[0]	[1]	[0]	Fc (MHz)	Frequency Response No
0	0	0	0	10	0_0
		0	1	12	0_1
		1	0	14	0_2
		1	1	16	0_3
0	1	0	0	27	1_0
		0	1	32	1_1
		1	0	36	1_2
		1	1	41	1_3
1	0	0	0	59	2_0
		0	1	69	2_1
		1	0	80	2_2
		1	1	91	2_3
1	1	0	0	95	3_0
		0	1	109	3_1
		1	0	126	3_2
		1	1	145	3_3

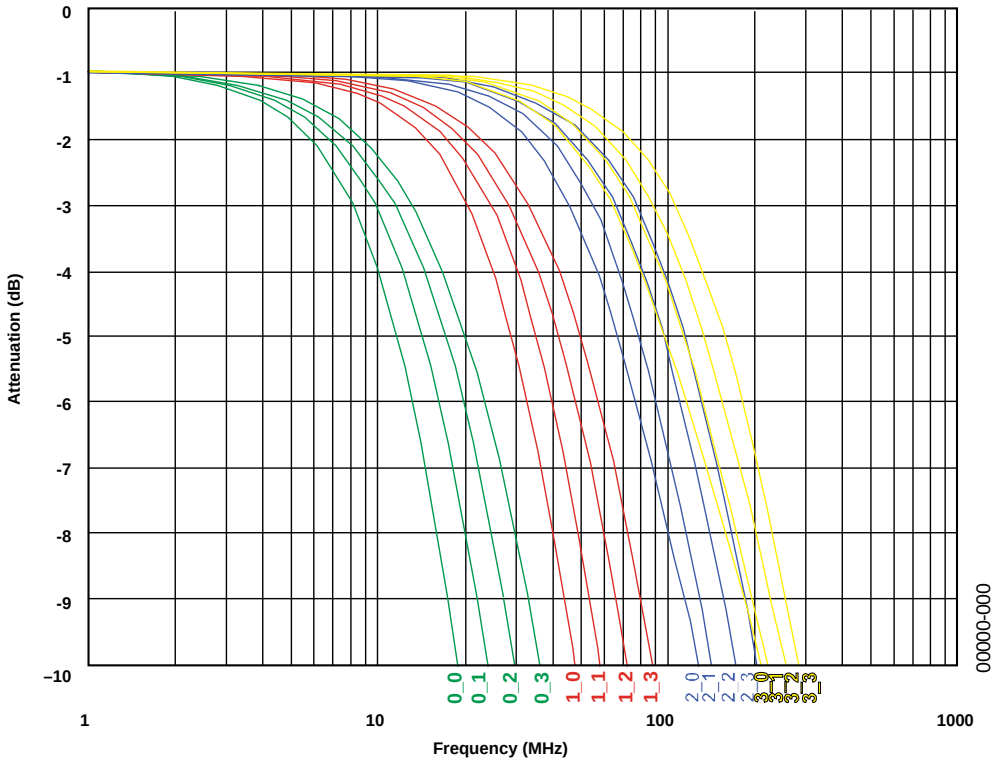


Figure 19: Response of Anti Aliasing Filters

7 STANDARD DEFINITION PROCESSOR

Figure 20 provides a block diagram of the ADV7842 Standard Definition Processor (SDP).

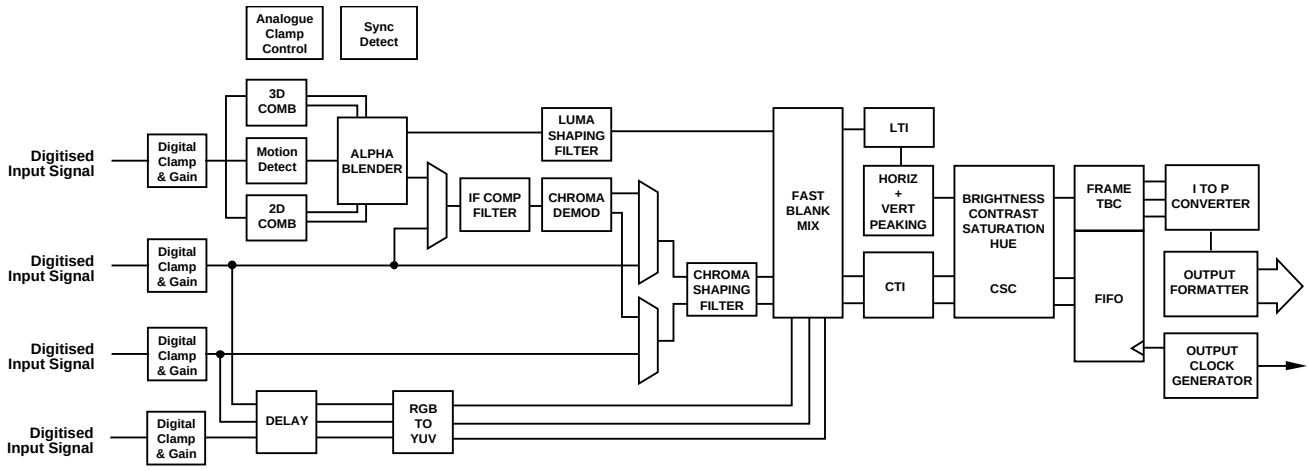


Figure 20: Block Diagram of SDP

The SDP block can handle standard definition video in CVBS, YC, YPbPr, and RGB formats. Figure 20 outlines the signal flow through the SDP.

7.1 SDP BLOCK

The SDP contains the following blocks:

- **Digital clamp and gain**
  This block uses a high precision algorithm to clamp and gain the video signal.
- **3D comb**
  A 3D comb provides YC separation by temporal combining certain lines. The 3D comb combines the current line with the same line of a time delayed version.
- **Motion detection**
  Motion detection is used to detect motion and provides the basis for a decision to use the 3D or 2D comb filters.
- **2D comb**
  The two-dimensional comb filters provide YC separation.
- **IF compensation filter**
  The IF compensation filter allows the user to compensate for saw filter characteristics on a composite input that can be observed on a tuner output.
- **Chroma demodulation**
  This block employs a color subcarrier recovery unit to regenerate the color subcarrier for any modulated chroma scheme.
- **AV code insertion**
  At this point, the decoded luma (Y) signal is merged with the retrieved chroma values; AV codes (as per ITU-R. BT-656) can be inserted. Timing adjustments can also be made.
- **Horizontal peaking/vertical peaking**
  Horizontal and vertical luma peaking enhance the picture produced by the ADV7842. The luma peaking function operates to boost or attenuate the mid to high frequency component of the Y signal.

- **Luma transient improvement (LTI)/chroma transient improvement (CTI)**
  LTI/CTI improves picture quality by restoring sharpness on transitions of chroma and luma, without affecting picture quality.
- **Color space conversion (CSC) matrix**
  The SDP processor has its own independent CSC matrix.

7.2 SDP SYNCHRONIZATION PROCESSING

The SDP extracts HSync and VSync synchronization that are embedded in the video data stream. The SDP does not support external HSync and VSync inputs. The synchronization extraction has been optimized to support imperfect video sources, such as VCRs with head switches. The extracted HSync and VSync synchronization is then used to drive the digital resampling section to ensure 720 active pixels per line are output by the SDP.

The ADV7842 uses a VSync and HSync PLL to provide an accurate and reliable lock for both perfect and imperfect video sources.

- VSync PLL: provides extra filtering of the detected VSyncs to give improved vertical lock.
- HSync PLL: designed to filter incoming HSyncs corrupted by noise; providing much improved performance for video signals with stable time base but poor signal to noise ratio (SNR).

Two registers, SDP_EXTEND_VS_MAX_FREQ and SDP_EXTEND_VS_MIN_FREQ, can be used to additionally extend Vsync lock ranges.

SDP_EXTEND_VS_MAX_FREQ, Addr 90 (SDP), *Address 0x7B[2]*

A control to extend the minimum frequency VSync lock range.

Function

SDP_EXTEND_VS_MAX_FREQ	Description
0 «	Normal minimum frequency VSync lock range
1	Extended minimum frequency VSync lock range

SDP_EXTEND_VS_MIN_FREQ, Addr 90 (SDP), *Address 0x7B[1]*

A control to extend the maximum frequency VSync lock range.

Function

SDP_EXTEND_VS_MIN_FREQ	Description
0 «	Normal maximum frequency VSync lock range
1	Extended maximum frequency VSync lock range

7.3 SDP GENERAL SETUP

7.3.1 Autodetection of SDP Modes

Automatic standard detection or specific standard forcing on the ADV7842 is achieved by specifying the standards to be automatically detected. The autodetection registers allows the user to force the digital core into a specific video standard. Setting the relevant bit to 0 inhibits the standard from being detected automatically. For example, to force the standard to PAL only the PAL autodetection bit should be set. To allow the decoder to detect automatically between PAL, NTSC, and SECAM, the appropriate bits should be set. The results of

the SDP autodetection can be read back via the status registers.

SDP_AD_SECAM_EN, Addr 90 (SDP), *Address 0x00[6]*

A control to enable autodetection of SECAM standard. Setting this bit to 1 enables the corresponding standard to be detected. In order to force the part into a particular standard, the corresponding enable bit for that standard should only be set. To allow full autodetect enable all standards via the respective bits.

Function

SDP_AD_SECAM_EN	Description
0 «	Do not enable SECAM to be detected.
1	Enable SECAM to be detected.

SDP_AD_N443_EN, Addr 90 (SDP), *Address 0x00[5]*

A control to enable autodetection of NTSC-443 standard. Setting this bit to 1 enables the corresponding standard to be detected. In order to force the part into a particular standard, the corresponding enable bit for that standard should only be set. To allow full autodetect enable all standards via the respective bits.

Function

SDP_AD_N443_EN	Description
0 «	Do not enable NTSC-443 to be detected
1	Enable NTSC-443 to be detected

SDP_AD_PAL60_EN, Addr 90 (SDP), *Address 0x00[4]*

A control to enable autodetection of PAL-60 standard. Setting this bit to 1 enables the corresponding standard to be detected. In order to force the part into a particular standard,the corresponding enable bit for that standard should only be set. To allow full autodetect enable all standards via the respective bits.

Function

SDP_AD_PAL60_EN	Description
0 «	Don't enable PAL-60 to be detected
1	Enable PAL-60 to be detected

SDP_AD_PALCN_EN, Addr 90 (SDP), *Address 0x00[3]*

A control to enable autodetection of PAL-Comb N standard. Setting this bit to 1 enables the corresponding standard to be detected. In order to force the part into a particular standard, the corresponding enable bit for that standard should only be set. To allow full autodetect enable all standards via the respective bits.

Function

SDP_AD_PALCN_EN	Description
0 «	Do not enable PAL-CombN to be detected
1	Enable PAL-CombN to be detected

SDP_AD_PALM_EN, Addr 90 (SDP), *Address 0x00[2]*

A control to enable autodetection of PAL-M standard. Setting this bit to 1 enables the corresponding standard to be detected. In order to force the part into a particular standard, the corresponding enable bit for that standard should only be set. To allow full autodetect enable all standards via the respective bits.

Function

SDP_AD_PALM_EN	Description
0 «	Do not enable PAL-M to be detected
1	Enable PAL-M to be detected

SDP_AD_NTSC_EN, Addr 90 (SDP), *Address 0x00[1]*

A control to enable autodetection of NTSC-M standard. Setting this bit to 1 enables the corresponding standard to be detected. In order to force the part into a particular standard, the corresponding enable bit for that standard should only be set. To allow full autodetect enable all standards via the respective bits.

Function

SDP_AD_NTSC_EN	Description
0	Do not enable NTSC-M to be detected
1 «	Enable NTSC-M to be detected

SDP_AD_PAL_EN, Addr 90 (SDP), *Address 0x00[0]*

A control to enable autodetection of PAL-BGHID standard. Setting this bit to 1 enables the corresponding standard to be detected. In order to force the part into a particular standard, the corresponding enable bit for that standard should only be set. To allow full autodetect enable all standards via the respective bits.

Function

SDP_AD_PAL_EN	Description
0 «	Do not enable PAL-BGHID to be detected
1	Enable PAL-BGHID to be detected

7.3.2 Pedestal Configuration in SDP Modes

The following controls dictate which standards the SDP core expects to have a pedestal. Standards that are expected to have a pedestal are clamped to the back porch level.

SDP_PAL_PED_EN, Addr 90 (SDP), *Address 0x01[0]*

A control to force the part to assume that the corresponding standard has a pedestal. Standards with pedestal are clamped to the pedestal level, standards without pedestal are clamped to the back porch level.

Function

SDP_PAL_PED_EN	Description
0 «	Assume PAL-BGHID inputs don't have a pedestal
1	Assume PAL-BGHID inputs have a pedestal

SDP_SECAM_PED_EN, Addr 90 (SDP), *Address 0x01[6]*

A control to force the part to assume that the corresponding standard has a pedestal. Standards with pedestal are clamped to the pedestal level, standards without pedestal are clamped to the back porch level.

Function

SDP_SECAM_PED_EN	Description
0 «	Assume SECAM inputs do not have a pedestal
1	Assume SECAM inputs have a pedestal

SDP_N443_PED_EN, Addr 90 (SDP), *Address 0x01[5]*

A control to force the part to assume that the corresponding standard has a pedestal. Standards with pedestal are clamped to the pedestal level, standards without pedestal are clamped to the back porch level.

Function

SDP_N443_PED_EN	Description
0	Assume NTSC-443 inputs do not have a pedestal
1 «	Assume NTSC-443 inputs have a pedestal

SDP_PAL60_PED_EN, Addr 90 (SDP), *Address 0x01[4]*

A control to force the part to assume that the corresponding standard has a pedestal. Standards with pedestal are clamped to the pedestal level, standards without pedestal are clamped to the back porch level.

Function

SDP_PAL60_PED_EN	Description
0	Assume PAL-60 inputs don't have a pedestal
1 «	Assume PAL-60 inputs have a pedestal

SDP_PALCN_PED_EN, Addr 90 (SDP), *Address 0x01[3]*

A control to force the part to assume that the corresponding standard has a pedestal. Standards with pedestal are clamped to the pedestal level, standards without pedestal are clamped to the back porch level.

Function

SDP_PALCN_PED_EN	Description
0 «	Assume PAL-CombN inputs do not have a pedestal
1	Assume PAL-CombN inputs have a pedestal

SDP_PALM_PED_EN, Addr 90 (SDP), *Address 0x01[2]*

A control to force the part to assume that the corresponding standard has a pedestal. Standards with pedestal are clamped to the pedestal level, standards without pedestal are clamped to the back porch level.

Function

SDP_PALM_PED_EN	Description
0	Assume PAL-M inputs don't have a pedestal
1 «	Assume PAL-M inputs have a pedestal

SDP_NTSC_PED_EN, Addr 90 (SDP), *Address 0x01[1]*

A control to force the part to assume that the corresponding standard has a pedestal. Standards with pedestal are clamped to the pedestal level, standards without pedestal are clamped to the back porch level.

Function

SDP_NTSC_PED_EN	Description
0	Assume NTSC-M inputs do not have a pedestal
1 «	Assume NTSC-M inputs have a pedestal

7.3.3 Subcarrier Frequency Lock Controls

The ADV7842 allows the user to configure manually the Subcarrier Frequency Lock (SFL) output.

SDP_SFL_INV_PSW, Addr 94 (SDP_IO), *Address 0x67[0]*

A control to invert PAL switch in SFL stream.

Function

SDP_SFL_INV_PSW	Description
0 «	Do not invert PAL switch in SFL stream
1	Invert PAL switch in SFL stream, (compatibility with some older video encoders)

SDP_SFL_INV_PSW controls the behavior of the PAL switch bit in the SFL (GenLock Telegram) data stream. It was implemented to solve some compatibility issues with video encoders.

It solves the following two problems:

- The PAL switch bit is only meaningful in PAL. Some encoders (including Analog Devices), however, do look at the state of this bit in NTSC too.
- There was a design change in Analog Devices encoders from ADV717x to ADV719x. The older versions used the SFL (GenLock Telegram) bit directly; the latter versions invert the bit prior to using it. The reason for this was that the inversion compensated for the one line delay of an SFL (GenLock Telegram) transmission.

As a result:

- ADV717x encoders need the PAL switch bit in the SFL (GenLock Telegram) to be 1 for NTSC to work
- ADV73xx, ADV719x encoders need the PAL switch bit in the SFL to be 0 to work in NTSC

If the state of the PAL switch bit is wrong, a phase shift of 180 degrees occurs. In a decoder/encoder back-to-back system where SFL is used, this bit needs to be set up properly for the specific encoder used.

SDP_AUTO_SFL_STD_EN, Addr 94 (SDP_IO), *Address 0x66[0]*

A control to enable automatic standard selection for SFL

Function

SDP_AUTO_SFL_STD_EN	Description
0	Uses SDP_MAN_SFL_STD, dependant on SDP_MAN_SFL_STD_EN
1 «	Automatic configuration of SFL output to convert all 50Hz inputs to PAL-BGHID and all 60Hz inputs to NTSC-MJ

SDP_MAN_SFL_STD_EN, Addr 94 (SDP_IO), *Address 0x66[1]*

A control to enable manual standard selection for SFL

Function

SDP_MAN_SFL_STD_EN	Description
0 «	Automatically configuration of SFL output to convert all 50 Hz inputs (including SECAM) to PAL-BGHID and all 60 Hz to inputs to NTSC-MJ.
1	Configure SFL output to convert input to standard given by SDP_MAN_SFL_STDconverted to PAL-BGHID and 60Hz SECAM is converted to NTSC-MJ.

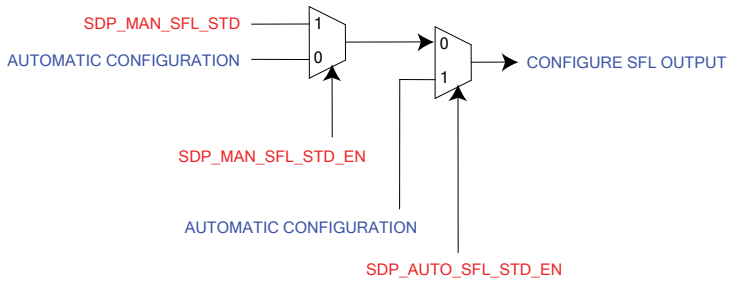


Figure gh-Z	Main Cb0,Cr0

SDP_MAN_SFL_STANDARD

Table : SFL Output Configuration

Manual SFL output format used if SDP_MAN_SFL_STD_EN set to 1.

Function

SDP_MAN_SFL_STANDARD[3:0]	Description
0000 «	NTSC-MJ
0001	Reserved
0010	NTSC-MJ
0011	NTSC-MJ
0100	PAL-M
0101	NTSC-MJ
0110	Reserved
0111	NTSC-MJ
1000	PAL-CombN
1001	PAL-BGHID/NTSC443
1010	PAL-BGHID/NTSC443
1011	PAL-BGHID/NTSC443
1100	PAL-CombN
1101	PAL-BGHID/NTSC443
1110	PAL-BGHID/NTSC443
1111	PAL-BGHID/NTSC443

7.4 SDP STATUS REGISTERS

The SDP contains registers that provide summary information about the video decoder. This information allows the user to read back information such as decoder lock status, input format and type, and autodetection result.

7.4.1 SDP Autodetection Result

SDP_STD[3:0], Addr 90 (SDP), *Address 0x52[3:0] (Read Only)*

An input status readback indicating if the currently active standard, detected mode in autodetect mode, manually programmed mode if in manual mode. Valid results are

Function

SDP_STD[3:0]	Description
0x00 «	NTSC-M/J
0x02	NTSC-443
0x03	60HzSECAM
0x04	PAL-M
0x06	PAL-60
0x0C	PAL-CombN
0x0E	PAL-BGHID
0x0F	SECAM

Refer to Section 7.3.1 for more information on the use of the autodetection block.

7.4.2 SDP Video Detection

SDP_VIDEO_DETECTED, Addr 90 (SDP), *Address 0x5A[0] (Read Only)*

A input status readback indicating the detection of a valid video input.

Function

SDP_VIDEO_DETECTED	Description
0 «	Input is invalid or no input is connected
1	Indicates valid SD/PR video input detected

SDP_C_CHAN_ACTIVE, Addr 90 (SDP), *Address 0x54[5] (Read Only)*

A input status readback indicating the result of the CVBS\YC detection feature.

Function

SDP_C_CHAN_ACTIVE	Description
0 «	CVBS input is detected
1	Y/C input is detected

SDP_PR_DETECTED_IN_SD, Addr 90 (SDP), *Address 0x58[6] (Read Only)*

An input status readback indicating mismatch between selected SD mode, and type of input signal.

Function

SDP_PR_DETECTED_IN_SD	Description
0 «	Normal operation
1	Indicates SDP is in SD mode but PR input is detected

7.4.3 Input Status

Register SDP_STATUS_INPUT_TYPE_1 consists of the following bit fields.

SDP_HSWITCH_PRESENT, Addr 90 (SDP), *Address 0x56[7] (Read Only)*

An input status readback indicating the result of the Head switch detection using algorithm 1.

Function

SDP_HSWITCH_PRESENT	Description
0 «	Head switch not detected by algorithm 1
1	Head switch detected by algorithm 1

SDP_BLK_NSTD, Addr 90 (SDP), *Address 0x56[6] (Read Only)*

An input status readback indicating the length of 192 line block of pixels in clock cycles it within the threshold set by SDP_FRM_NSTD_THR of a nominal standard.

Function

SDP_BLK_NSTD	Description
0 «	Length of 192 line block of pixels in clock cycles is within +- SDP_FRM_NSTD_THR of nominal value
1	Length of 192 line block of pixels in clock cycles is not within +- SDP_FRM_NSTD_THR of nominal value

SDP_FLD_NSTD, Addr 90 (SDP), *Address 0x56[5] (Read Only)*

An input status readback indicating if the Field length in clock cycles is within the threshold set by SDP_FRM_NSTD_THR of a nominal standard.

Function

SDP_FLD_NSTD	Description
0 «	Field length in clock cycles is within +- SDP_FRM_NSTD_THR of nominal value
1	Field length in clock cycles is not within +- SDP_FRM_NSTD_THR of nominal value

SDP_FRM_NSTD, Addr 90 (SDP), *Address 0x56[4] (Read Only)*

An input status readback indicating if the Frame length in clock cycles is within the threshold set by SDP_FRM_NSTD_THR.

Function

SDP_FRM_NSTD	Description
0 «	Frame length in clock cycles is within +- SDP_FRM_NSTD_THR of nominal value
1	Frame length in clock cycles is not within +- SDP_FRM_NSTD_THR of nominal value

SDP_LC_NSTD, Addr 90 (SDP), *Address 0x56[3] (Read Only)*

An input status readback indicating if the field length varies by more than one line from field to field.

Function

SDP_LC_NSTD	Description
0 «	Field length in terms of number of lines does not vary by more than 1 line from field to field
1	Field length in terms of number of lines varies by more than 1 line from field to field

SDP_ALLOW_MED_PLL, Addr 90 (SDP), *Address 0x56[2] (Read Only)*

An input status readback indicating if the input could be from a VCR source. Only valid if SDP_ALLOW_SLOW_PLL is 0. If SDP_ALLOW_SLOW_PLL is 1 this bit is ignored.

Function

SDP_ALLOW_MED_PLL	Description
0 «	Input is a VCR, a fast HSync PLL speed used
1	Input may be a VCR, a medium HSync PLL speed used

SDP_ALLOW_SLOW_PLL, Addr 90 (SDP), *Address 0x56[1] (Read Only)*

An input status readback indicating if the input could be from a VCR source. To be used in conjunction SDP_ALLOW_MED_PLL.

Function

SDP_ALLOW_SLOW_PLL	Description
0 «	Input may be a VCR, refer to SDP_ALLOW_MED_PLL
1	Input is not a VCR slow HSync PLL speed used

SDP_FREE_RUN, Addr 90 (SDP), *Address 0x56[0] (Read Only)*

A status readback indicating free-run status. If set the part is free-running due to no video detected on input or forced free run mode.

Function

SDP_FREE_RUN	Description
0 «	Part is not free running
1	Part is free running

SDP_CKILL_ACT, Addr 90 (SDP), *Address 0x57[7] (Read Only)*

A status readback indicating the Colour kill status.

Function

SDP_CKILL_ACT	Description
0 «	Colour kill is not active
1	Colour kill is active (and enabled)

SDP_VS_STD_MODE, Addr 90 (SDP), *Address 0x57[6] (Read Only)*

An input status readback indicating the detection for regular frame lengths on the input.

Function

SDP_VS_STD_MODE	Description
0 «	Regular frame lengths not detected on input
1	Regular frame lengths detected on input

SDP_ALLOW_3D_COMB, Addr 90 (SDP), *Address 0x57[4] (Read Only)*

An input status readback indicating the suitability of the input for 3D combing.

Function

SDP_ALLOW_3D_COMB	Description
0 «	Non-standard input detected, 3D comb not allowed, 2d comb used
1	Standard input detected, 3D comb allowed

SDP_INTERLACED, Addr 90 (SDP), *Address 0x57[3] (Read Only)*

A input status readback indicating the detection of an interlaced format on the input.

Function

SDP_INTERLACED	Description
0 «	Alternating field sequence not detected on input
1	Alternating field sequence detected on input

SDP_TRICK_MODE, Addr 90 (SDP), *Address 0x57[2] (Read Only)*

An input status readback indicating the detection of a VCR trick mode operation on the input.

Function

SDP_TRICK_MODE	Description
0 «	VCR trick mode not detected, line TBC not allowed
1	VCR trick mode detected, line TBC allowed if enabled

SDP_BURST_LOCKED_RB, Addr 90 (SDP), *Address 0x59[7] (Read Only)*

A readback indicating the status of the burst locking loop.

Function

SDP_BURST_LOCKED_RB	Description
0 «	Burst locking loop is not locked
1	Burst locking loop is locked

SDP_AD_50_60_HZ, Addr 90 (SDP), *Address 0x59[3] (Read Only)*

A readback indicating the result of the field rate detection on the input.

Function

SDP_AD_50_60_HZ	Description
0 «	Vertical refresh rate of 60Hz detected on the input
1	Vertical refresh rate of 50Hz detected on the input

SDP_PAL_SW_LOCKED, Addr 90 (SDP), *Address 0x59[2] (Read Only)*

A input status readback indicating the detection of a PAL swinging burst sequence on the input.

Function

SDP_PAL_SW_LOCKED	Description
0 «	PAL swinging burst sequence is not detected.
1	PAL swinging burst sequence is detected.

SDP_FSC_FREQ_OK, Addr 90 (SDP), *Address 0x59[1] (Read Only)*

An input status readback indicating if detected frequency subcarrier is close to that of the selected standard.

Function

SDP_FSC_FREQ_OK	Description
0 «	Detected Fsc frequency is not close to that of selected standard
1	Detected Fsc frequency is close to that of selected standard.

SDP_SCM_LOCKED, Addr 90 (SDP), *Address 0x59[0] (Read Only)*

A input status readback indicating the detection of a SECAM input.

Function

SDP_SCM_LOCKED	Description
0 «	SECAM is not detected on the input
1	SECAM is detected on the input

7.4.4 Macrovision Status

Macrovision status bits are split into two registers, **STATUS_MACROVISION_DETECTION_1** and **STATUS_MACROVISION_DETECTION_2** located in the SDP Map at address 0x50 and 0x51 respectively.

STATUS_MACROVISION_DETECTION_1 Register

SDP_MV_AGC_DETECTED, Addr 90 (SDP), *Address 0x50[3] (Read Only)*

A Macrovision status readback indicating the detection of Macrovision AGC pulses.

Function

SDP_MV_AGC_DETECTED	Description
0 «	Macrovision AGC pulses part of AGC process not detected
1	Macrovision AGC pulses part of AGC process detected

SDP_MV_PS_DETECTED, Addr 90 (SDP), *Address 0x50[2] (Read Only)*

A Macrovision status readback indicating the detection of Macrovision AGC pseudo syncs.

Function

SDP_MV_PS_DETECTED	Description
0 «	Macrovision pseudo-sync part of AGC process not detected
1	Macrovision pseudo-sync part of AGC process detected

SDP_MVCS_TYPE3, Addr 90 (SDP), *Address 0x50[1] (Read Only)*

A Macrovision status readback indicating the detection of Macrovision type 3 colourstripe process. This bit is only valid if SDP_MVCS_DETECT is set to 1

Function

SDP_MVCS_TYPE3	Description
0 «	Macrovision type 3 colourstripe process not detected
1	Macrovision type 3 colourstripe process detected, only valid if SDP_MVCS_DETECT = 1

SDP_MVCS_DETECT, Addr 90 (SDP), *Address 0x50[0] (Read Only)*

A Macrovision status readback indicating the detection of colourstripe process.

Function

SDP_MVCS_DETECT	Description
0 «	Macrovision colourstripe process not detected
1	Macrovision colourstripe process detected

STATUS_MACROVISION_DETECTION_2 Register

SDP_BP_TOTAL_PULSE_BEG[3:0], Addr 90 (SDP), *Address 0x51[7:4] (Read Only)*

A Macrovision readback indicating the total Macrovision back porch pulses detected at the beginning of the field.

Function

SDP_BP_TOTAL_PULSE_BEG[3:0]	Description
xxxx	Number of Macrovision back porch pulses detected at the beginning of the field

SDP_BP_TOTAL_PULSES_END[3:0], Addr 90 (SDP), *Address 0x51[3:0] (Read Only)*

A Macrovision readback Total Macrovision back porch pulses detected at the end of the field

Function

SDP_BP_TOTAL_PULSES_END[3:0]	Description
xxxx	Number of Macrovision back porch pulses detected at the end of the field

7.4.5 Synctip Noise Measurement, Noisy and Very Noisy Signal Detection

SDP_SYNCTIP_NOISE[11:0] is a readback value of the average of the noise in the HSync tip. The SDP_SYNCTIP_NOISE[11:0] register is spread over two complete I^2C registers, allowing the user to read eight MSBs or eight LSBs of synctip noise in one I^2C read transfer.

1 bit of SDP_SYNCTIP _NOISE[11:0] = 1 ADC code

1 bit of SDP_SYNCTIP _NOISE[11:0] = 398.883uV

SDP_SYNCTIP_NOISE[11:0], Addr 90 (SDP), *Address 0x53[7:4]; Address 0x4F[7:0] (Read Only)*

HSync noise power readback 12-bit ADC codes, larger values indicate more measured noise, not saturated.

Function

SDP_SYNCTIP_NOISE[11:0]	Description
xxxxxxxxxxxx	Readback of Hsync noise power

Note that register 0x53[3:0] contains SDP_SYNCTIP_NOISE[7:4]. The polling of a single register in 0x53[7:0] reads back the MSBs of the SDP_SYNCTIP_NOISE[11:4]. The polling of a single register in 0x4F[7:0] reads back the LSBs of SDP_SYNCTIP_NOISE[7:0]. The user has to read back both registers (0x53 and 0x4F) to obtain the full noise measurement from SDP_SYNCTIP_NOISE[11:0].

The SDP_NOISY_IP and SDP_VERY_NOISY_IP bits can be used to check the quality of the video signal. These bits are located in the SDP map in register 0x54. By doing single a readback of register 0x54, the quality of video signal can be determined quickly. Registers SDP_NOISY_THR[7:0] and SDP_VERY_NOISY_THR[7:0] are used to set the threshold for input to detect signal as noisy or very noisy.

The following controls are used in conjunction with the bits used for 3D comb activation (refer to Section 7.8.1):

- SDP_NOISY_HSW2_DIS_3D
- SDP_NOISY_HSW1_DIS_3D
- SDP_NOISY_LC_DIS_3D
- SDP_NOISY_BLK_DIS_3D
- SDP_NOISY_FLD_DIS_3D
- SDP_NOISY_FRM_DIS_3D
- SDP_NOISY_DIS_3D
- SDP_VNOISY_HSW2_DIS_3D
- SDP_VNOISY_HSW1_DIS_3D
- SDP_VNOISY_LC_DIS_3D
- SDP_VNOISY_BLK_DIS_3D
- **SDP_VNOISY_FLD_DIS_3D**
- SDP_VNOISY_FRM_DIS_3D
- SDP_VNOISY_DIS_3D

SDP_NOISY_IP, Addr 90 (SDP), *Address 0x54[7] (Read Only)*

A input status readback indicating the detection of a noisy input signal. Refer also to (SDP_VERY_NOISY_IP and SDP_SYNCTIP_NOISE)

Function

SDP_NOISY_IP	Description
0 «	Noisy input not detected
1	Noisy input detected

SDP_VERY_NOISY_IP, Addr 90 (SDP), *Address 0x54[6] (Read Only)*

A input status readback indicating the detection of a very noisy input signal. Refer also to (SDP_NOISY_IP and SDP_SYNCTIP_NOISE)

Function

SDP_VERY_NOISY_IP	Description
0 «	Very noisy input not detected
1	Very noisy input detected

SDP_NOISY_THR[7:0], Addr 90 (SDP), *Address 0xA1[7:0]*

A control to set the threshold for input to be detected as noisy. A higher value reduces the possibility of detecting the input as noisy.

Function

SDP_NOISY_THR[7:0]	Description
0x50 «	Default value

SDP_VERY_NOISY_THR[7:0], Addr 90 (SDP), *Address 0xA2[7:0]*

A control to set the threshold for input to be detected as very noisy. A higher value reduces the possibility of detecting the input to be very noisy.

Function

SDP_VERY_NOISY_THR[7:0]	Description
0xA0 «	Default value

7.5 SDP COLOR CONTROLS

The following registers provide user control over the picture appearance. They are independent of any other controls. For instance, the brightness control is independent from the picture clamping, although both controls affect the DC level of the signal. The ADV7842 provides 10-bit control of contrast, brightness, saturation, and hue. Refer to Section 7.22.7 .

SDP_CONTRAST[9:0], Addr 90 (SDP), *Address 0x13[7:0]; Address 0x17[1:0]*

A control to set the Contrast level (luma gain). This control has a range of gain from 0 to 2. It is an unsigned number and has a range from 0x000 (lowest contrast, all black) to 0x3FF (highest contrast).

Function

SDP_CONTRAST[9:0]	Description
0x000	Lowest contrast
0x200 «	Default contrast
0x3FF	Highest contrast

SDP_BRIGHTNESS[9:0], Addr 90 (SDP), *Address 0x14[7:0]; Address 0x17[3:2]*

A control to set the brightness level (luma offset). It is a 2s complement number and has a range of 0x200 (darkest) to 0x1FF (brightest).

Function

SDP_BRIGHTNESS[9:0]	Description
0x200	Darkest
0x000 «	Default brightness
0x1FF	Brightest

SDP_SATURATION[9:0], Addr 90 (SDP), *Address 0x15[7:0]; Address 0x17[5:4]*

This is a control to set the saturation level (chroma gain). It has a valid range 0 -> 1.75. It is an unsigned number and has a range of 0x000 (lowest saturation, no color) to 0x3FF (highest saturation).

Function

SDP_SATURATION[9:0]	Description
0x000	Lowest saturation (No Color)
0x200 «	Default value
0x3FF	Highest saturation

SDP_HUE[9:0], Addr 90 (SDP), *Address 0x16[7:0]; Address 0x17[7:6]*

A control to set the Hue (chroma phase rotation). It is a twos compliment number and has a range of 0x200 (-180 degrees) to 0x1FF (+180 degrees).

Function

SDP_HUE[9:0]	Description
0x1FF	+180°
0x000 «	0°
0x200	-180°

7.6 DECIMATION FILTERS FOR SDP

Along with the CP decimation filters, the ADV7842 also includes SDP decimation filters. These are positioned directly before and after the SDP CSC section. These decimation filters are automatically controlled using VID_STD. Refer to Figure 21 for a block diagram of these filters.

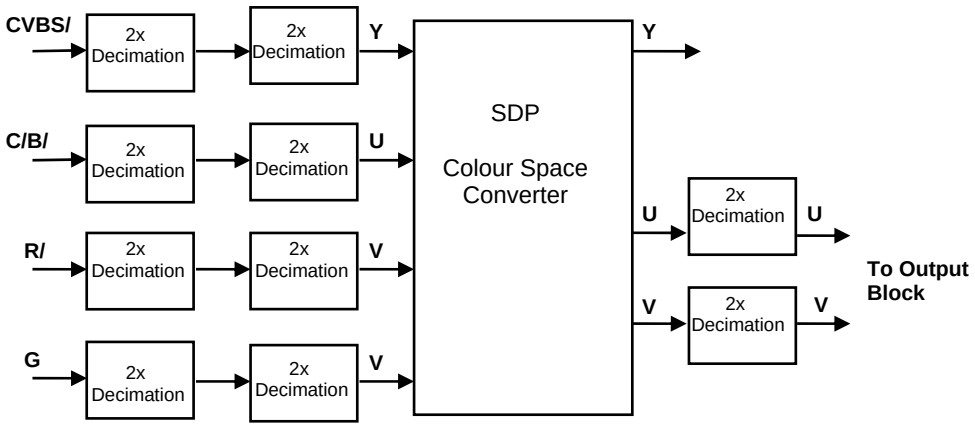


Figure 21: SDP Decimation Filters

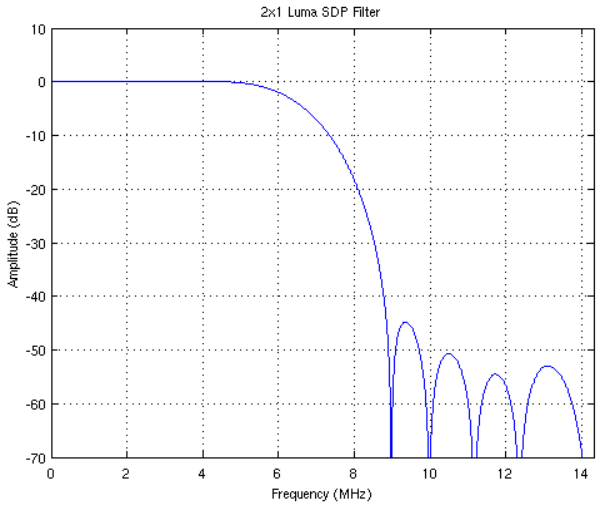


Figure 22: 2x1 Luma Filter Response (SDP Only)

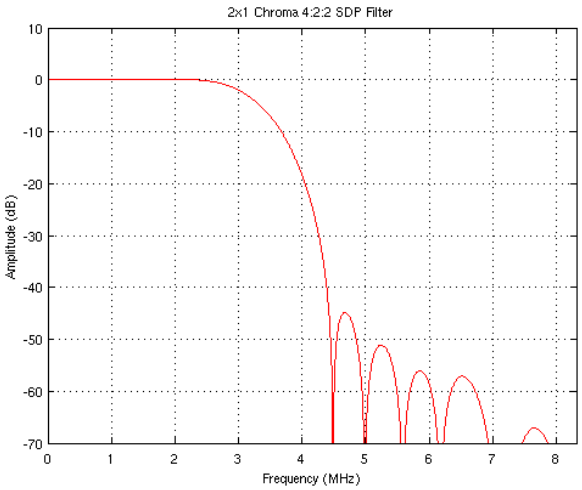


Figure 23: 2x1 Chroma 4:2:2 Filter Response (SDP Only)

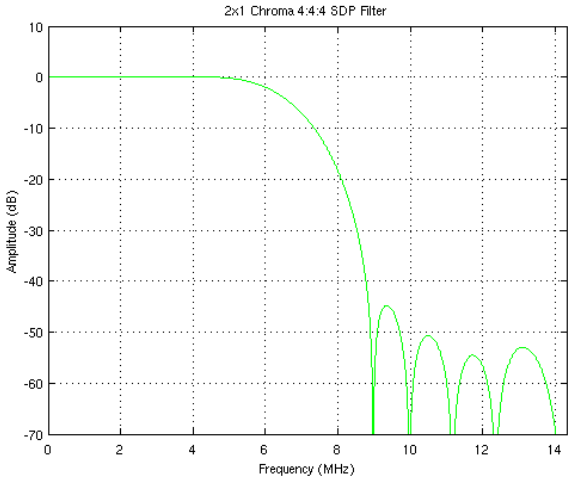


Figure 24: 2x1 Chroma 4:4:4 Filter Response (SDP Only)

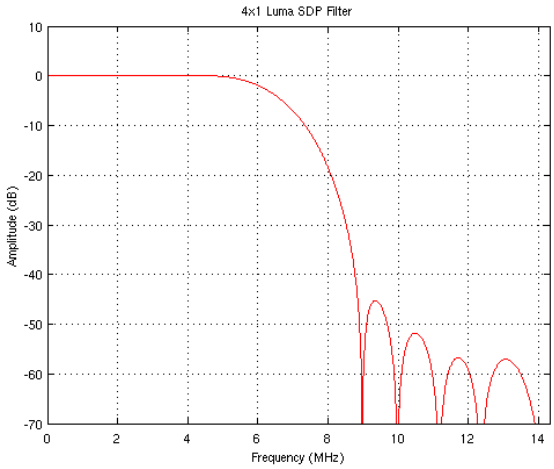


Figure 25: 4x1 Luma Filter Response (SDP Only)

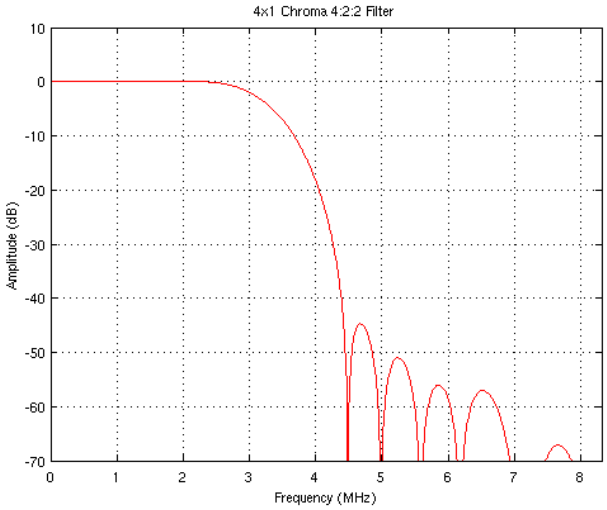


Figure 26: 4x1 Chroma 4:2:2 Filter Response (SDP Only)

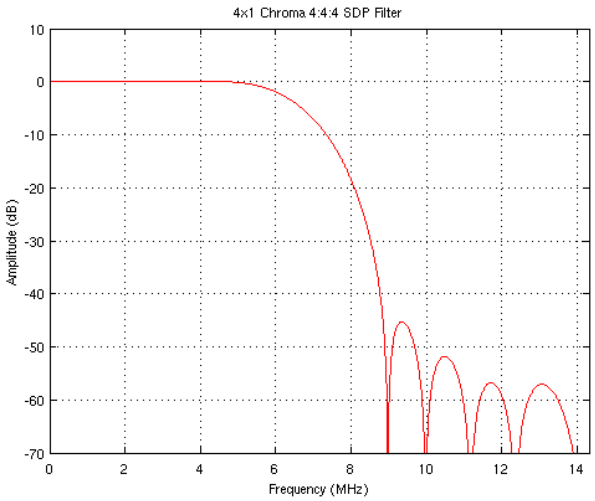


Figure 27: 4x1 Chroma 4:4:4 Filter Response (SDP Only)

7.7 SDP GAIN OPERATION

The SDP gain control within the ADV7842 is done on a purely digital basis. SDP gain correction takes place after digitization in the form of a digital multiplier.

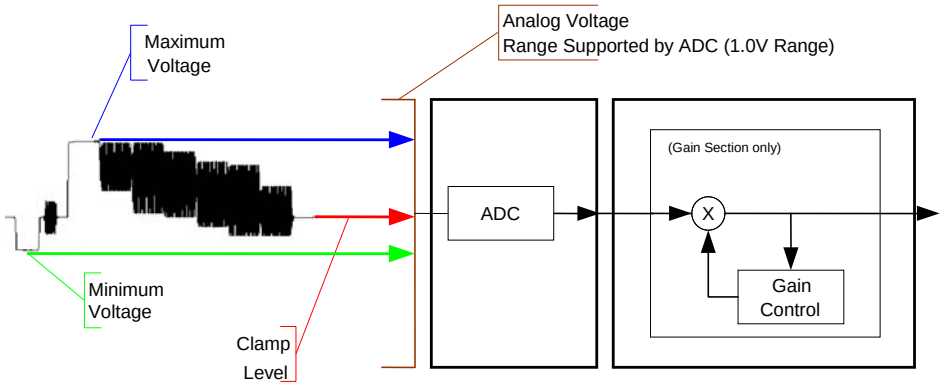


Figure 28: SDP Gain Control Overview

As shown in Figure 28, the ADV7842 can decode a video signal as long as it fits into the ADC window. There are two components; the amplitude of the input signal, and the DC level on which it resides. The DC level is set by the clamping circuitry.

If the amplitude of the analog video signal is too high, clipping can occur and visual artifacts appear. The analog input range of the ADC and the clamp level determine the maximum supported amplitude of the video signal.

The minimum supported amplitude of the input video is determined by the core ability of the SDP to retrieve horizontal and vertical timing and to lock to the color burst, if present.

There are two gain control units, one for luma data and the other for chroma data. Both operate independently of each other.

The luma gain controls operate in three basic modes:

- Manual gain.
  The gain can be controlled manually irrespective of peak white.
- Automatic gain control (AGC) with peak white adjustment.
  The peak white feature overrides and reduces the gain of the luma AGC if the input signal exceeds a specific threshold.
- Automatic gain control without peak white adjustment.

The applied luma gain values can be read back using the SDP_Y_GAIN_MAN_RB[12:0] register. The chroma gain controls operate in a similar way.

Peak black overrides and increases the gain of the luma AGC if the input signal becomes lower than a set threshold. Peak black is not user controllable.

7.7.1 SDP Luma Gain

SDP_Y_AGC_EN, Addr 90 (SDP), *Address 0x03[7]*

A control to select between automatic and manual luma gain control.

Function

SDP_Y_AGC_EN	Description
0	Enable manual luma gain, set by SDP_Y_GAIN_MAN
1 «	Enable automatic luma gain based on the sync.

SDP_MAN_GAIN_VCR, Addr 90 (SDP), *Address 0x03[5]*

A control to select gain method used when a VCR input is detected.

Function

SDP_MAN_GAIN_VCR	Description
0 «	Automatic gain used for VCR inputs (Valid only if SDP_Y_AGC_EN set to 1)
1	Manual gain used for VCR inputs value is SDP_Y_GAIN_MAN

SDP_Y_GAIN_MAN[12:0], Addr 90 (SDP), *Address 0x03[4:0]; Address 0x04[7:0]*

Manual luma gain value, used if SDP_Y_AGC_EN = 0. Also applies to G channel in component modes. Range of 0.5 to 4.

Function

SDP_Y_GAIN_MAN[12:0]	Description
0x040B «	Default value

$$\text{Luma\_Gain} = \frac{512 < SDP\_Y\_GAIN\_MAN[12:0] \leq 4095}{1024} = 0.5 \ldots 4.0$$

Equation 2: SDP Luma Gain Formula

The luma gain range is from 0.5 to 4.0.

Example:
To program the ADV7842 into manual fixed gain mode with a desired gain of 0.95:

- Use Equation 2 to convert the gain:
  0.95 * 1024 = 972.8
- Truncate to integer value:
  972.8➔ 972
- Convert to hexadecimal:
  972 ➔ 0x3CC
- Split into two registers and program:
  SDP_Y_GAIN_MAN[12:8] = 0x3
  SDP_Y_GAIN_MAN[7:0] = 0xCC
- Enable manual fixed gain mode:
  Set SDP_Y_AGC_EN to 0

SDP_Y_GAIN_MAN_RB[12:0], Addr 90 (SDP), *Address 0x54[4:0]; Address 0x55[7:0] (Read Only)*

A readback providing the current luma gain.

Function

SDP_Y_GAIN_MAN_RB[12:0]	Description
xxxxxxxxxxxxx	Readback

SDP_LIMIT_Y_GAIN, Addr 90 (SDP), *Address 0x89[7]*

A control to limit the luma gain.

Function

SDP_LIMIT_Y_GAIN	Description
0 «	Normal operation.
1	Limits the luma gain to a range of 50%-200%

SDP_DGAIN_SPEED[4:0], Addr 90 (SDP), *Address 0x0A[4:0]*

Control to adjust the speed of luma digital gain operation. Only values of 1 to 6 are within a valid range.

Function

SDP_DGAIN_SPEED[4:0]	Description
0x00	Freeze digital gain
0x05 «	Default value
0x06	Max value

7.7.2 Chroma Gain

SDP_C_AGC_EN, Addr 90 (SDP), *Address 0x05[7]*

A control to select between automatic and manual chroma gain (C/Pr/Pb/R/B channels also used for G in case of SCART)

Function

SDP_C_AGC_EN	Description
0	Enable manual chroma gain, gain value set by SDP_C_GAIN_ACT_MAN
1 «	Enable automatic chroma gain based on burst power

SDP_C_GAIN_ACT_MAN[12:0], Addr 90 (SDP), *Address 0x05[4:0]; Address 0x06[7:0]*

Manual chroma gain value, used if SDP_C_AGC_EN = 0. Also applies to Pr, Pb, R, B channels in component modes Range of 0.5 to 8.

Function

SDP_C_GAIN_ACT_MAN [12:0]	Description
0x03C0 «	Default value

$$\text{Chroma\_Gain} = \frac{512 < SDP\_C\_GAIN\_ACT\_MAN < 8191}{1024} = 0.5\ldots8.0$$

Equation 3: SDP Chroma Gain Formula

The chroma gain range is from 0.5 to 8.0.

Example:

To program the ADV7842 into manual fixed chroma gain with a desired gain of 0.70 (signal):

- Use Equation 3 to convert the gain:
  0.70 * 1024 = 716.8
- Truncate to integer value:
  716.8➔ 716
- Convert to hexadecimal:
  716 ➔ 0x2CC
- Split into two registers and program:
  SDP_C_GAIN_ACT _MAN[12:8] = 0x2
  SDP_C_GAIN_ACT _MAN[7:0] = 0xCC
- Enable manual fixed gain mode:
  Set SDP_C_AGC_EN to 0

SDP_LIMIT_C_GAIN, Addr 90 (SDP), *Address 0x89[6]*

A control to limit the chroma gain.

Function

SDP_LIMIT_C_GAIN	Description
0 «	Normal operation.
1	Limits the chroma gain to a range of 50%-200%.

SDP_LIMIT_UV_GAIN, Addr 90 (SDP), *Address 0x89[5]*

A control to limit U/V gain.

Function

SDP_LIMIT_UV_GAIN	Description
0 «	Normal operation.
1	limits the U&V gain to a range of 50%-200%

SDP_LIMIT_G_GAIN, Addr 90 (SDP), *Address 0x89[4]*

A control to limit the SD SCART FB RGB gain.

Function

SDP_LIMIT_G_GAIN	Description
0 «	Normal operation.
1	Limits the SD (FB) RGB gain to a range of 50%-200%

SDP_C_DGAIN_SPEED[4:0], Addr 90 (SDP), *Address 0x0B[4:0]*

Control to adjust the speed of chroma digital gain operation. Only values of 1 to 6 are within a valid range. This register has an effect only if the SDP_C_AGC_EN bit is set to 1

Function

SDP_C_DGAIN_SPEED[4:0]	Description
0x00	Freeze digital gain
0x05 «	Default value
0x06	Max value
>0x06	Reserved

SDP_C_DGAIN_SPEED[4:0] allows the user to influence the tracking speed of the chroma automatic gain control. Note that this register has an effect only if the SDP_C_AGC_EN bit is set to 1 (automatic gain). A setting of 0 freezes the chroma digital gain. The larger the register value, the faster the loop operates. Values above 0x06 are reserved.

7.7.3 Peak White Feature

SDP_PW_EN, Addr 90 (SDP), *Address 0x03[6]*

A control to enable Peak-white luma gain feature.

Function

SDP_PW_EN	Description
0	Disable peak-white luma gain control
1 «	Enable peak-white luma gain control

SDP_PW_REC_RATE[11:0], Addr 90 (SDP), *Address 0x0F[3:0]; Address 0x10[7:0]*

A control to adjust the peak-white gain recovery speed. The speed at which the luma gain is increased following a gain reduction due to a peak white violation. A larger value corresponds to a faster speed.

Function

SDP_PW_REC_RATE[11:0]	Description
000000000001 «	Default value

7.7.4 Peak Chroma

SDP_PC_EN, Addr 90 (SDP), *Address 0x05[6]*

A control to enable Peak chroma gain feature. Peak-colour chroma overrides and reduces the gain of the chroma AGC if the chroma signal path becomes larger than a set threshold. Peak chroma can only act to reduce the AGC gain. When there are no more violations of the peak white threshold, the peak chrome algorithm allows the chroma AGC to restore the gain (based on synchronization depth). The recovery rate of the AGC gain is set by the peak chroma recovery register.

Function

SDP_PC_EN	Description
0	Disable peak-chroma gain.
1 «	Enable peak-chroma gain.

SDP_PC_REC_RATE[11:0], Addr 90 (SDP), *Address 0x0F[7:4]; Address 0x11[7:0]*

A control to adjust the Peak-chroma gain recovery speed. The speed at which the chroma gain is increased following a gain reduction due to peak colour violation. A larger value results in the faster speed.

Function

SDP_PC_REC_RATE[11:0]	Description
000000010000 «	Default value

7.7.5 Color Kill

Color kill mode is the feature of the ADI decoder that allows removing chroma when color burst level of input is poor. This feature is

especially useful when inputting weak-signal tuner signals.

SDP_CKILL_EN, Addr 90 (SDP), *Address 0x07[7]*

A control to enable the color kill feature.

Function

SDP_CKILL_EN	Description
0	Disable colour-kill feature.
1 «	Enable colour-kill feature.

SDP_CK_LOW_THR[6:0], Addr 90 (SDP), *Address 0x07[6:0]*

A control to set colour kill low threshold. If the burst power is below this threshold, enter colour kill mode.

Function

SDP_CK_LOW_THR[6:0]	Description
0001011 «	Color Kill low threshold

SDP_CK_HIGH_THR[7:0], Addr 90 (SDP), *Address 0x08[7:0]*

A control to set colour kill high threshold. If the burst power is above this threshold, enter colour kill mode.

Function

SDP_CK_HIGH_THR[7:0]	Description
00011010 «	Color Kill high threshold

7.8 3D COMB

The ADV7842 has the ability to separate the luminance (Y) and chrominance (C) components using 1D (horizontal), 2D (horizontal and vertical), or 3D (temporal) processing.

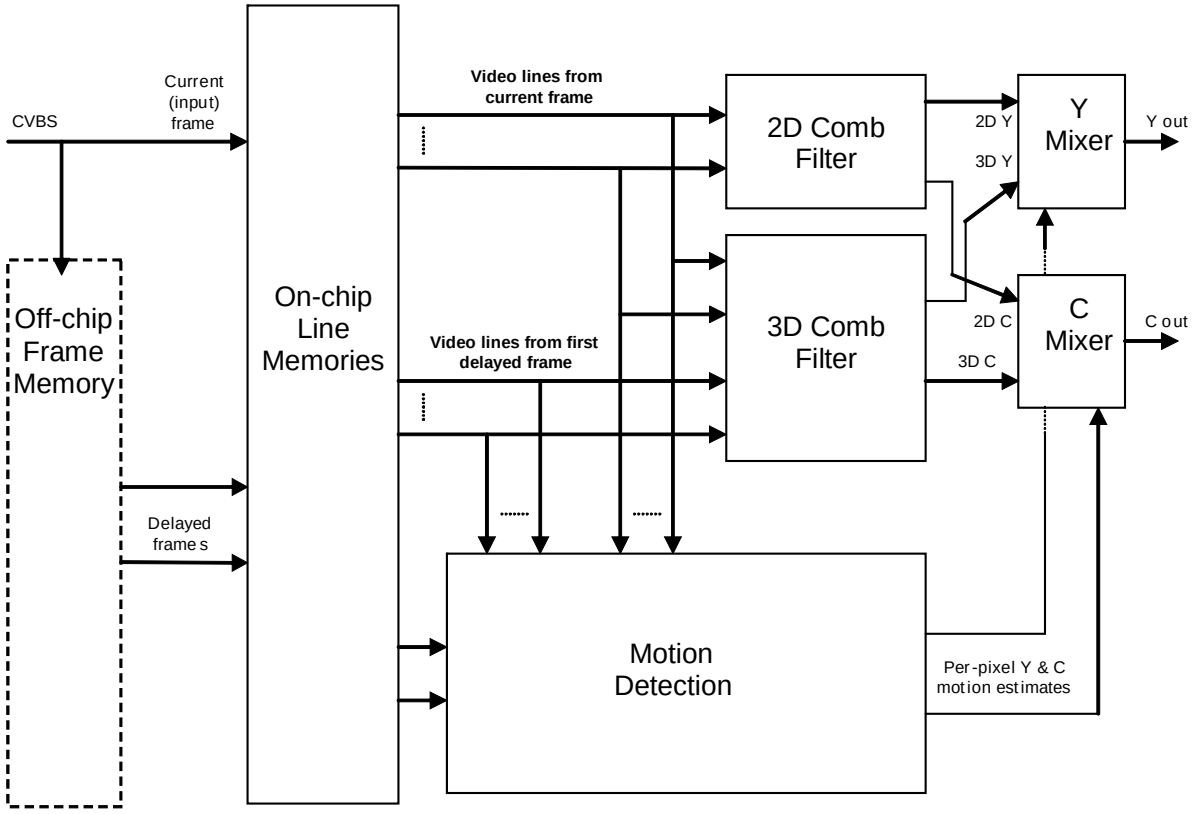


Figure 29: 3D Comb and Motion Detection Operation

When the picture content is static, the 3D comb filter in the ADV7842 combines video frames to give perfect Y/C separation results. Artifacts of 1D and 2D methods such as dot crawl, cross color, and hanging dots are eliminated. The sharpness and stillness of the decoded picture are improved using the 3D comb.

Once the picture content begins to move, 3D Y/C separation is no longer possible and 2D or 1D combing must be used instead. The ADV7842 features advanced motion detection, enabling motion adaptive combing between 3D and 2D or 1D Y/C separation methods. This maximizes the benefit from 3D processing while minimizing temporal artifacts on moving picture regions.

Note: 3D combing requires video frame storage in external memory. Where external memory is not connected, SDP_3D_COMB_EN must be disabled.

7.8.1 3D Comb Activation

SDP_3D_COMB_EN, Addr 90 (SDP), *Address 0x12[0]*

A control to enable the 3D comb filter. When the 3D comb is enabled automatic 2D/3D comb switching is applied based on detected video type. When the 3D comb is disabled 2D combing only is applied.

Function

SDP_3D_COMB_EN	Description
0	Disable 3D comb filtering, enable 2D comb mode only
1 «	Allow 3D comb filtering when it is possible

SDP_ALLOW_3D_COMB indicates if the ADV7842 has detected a non standard input for which 3D processing is not possible. For high

quality PAL and NTSC input signals with nominal timebase, 3D processing is possible. For unstable, noisy, or otherwise non standard video signals, 3D processing is impossible. The user can disable 3D processing manually via SDP_3D_COMB_EN.

The ADV7842 applies a deglitching filter to the result of this decision, with the programmable time constant controlled by SDP_ALLOW_3D_FILT_SEL[2:0].

SDP_ALLOW_3D_FILT_SEL[2:0], Addr 90 (SDP), *Address 0x9A[2:0]*

SDP_ALLOW_3D_FILT_SEL[2:0] controls the time constant applied to the deglitching filter for the 3D comb decision.

Function

SDP_ALLOW_3D_FILT_SEL[2:0]	Description
000	No filtering
001 «	0.25s
010	0.55s
011	0.81s
100	1.10s
101	1.36s
110	1.63s
111	2.00s

The ADV7842 monitors the input signal for noise and characteristics that are undesirable for 3D processing (information about noisy or very noisy input signal is accessible via SDP_VERY_NOISEY_IPand SDP_NOISY_IP).

ADV7842 characterizes the signal as a clean, noisy, or very noisy input signal, and assesses whether or not the input signal has the following characteristics:

- Incorrect number of lines
- Non standard block length
- Non standard frame length
- Non standard field length
- VCR head switches (unstable time base)
- No color burst (color kill is active)

For each input signal characterization (clean, noisy, or very noisy), it is possible to select whether or not the 3D comb is disabled on detection of the characteristics listed above. Note that 3D processing, if enabled with non standard inputs, can cause undesirable artifacts.

The following controls allow the user to select the non standard characteristics that contribute to the SDP_ALLOW_3D_COMB decision.

SDP_CKILL_DIS_3D, Addr 90 (SDP), *Address 0xA3[7]*

A control to enable 3D combing if colour kill mode is active.

Function

SDP_CKILL_DIS_3D	Description
0	Allow 3D comb even if colour kill is active
1 «	Disable 3D comb if colour kill is active

SDP_CKILL_DIS_2D, Addr 90 (SDP), *Address 0xA4[7]*

A control to enable 2D combing even if colour kill mode is active. This would effectively be pass through mode.

Function

SDP_CKILL_DIS_2D	Description
0	Use 2D comb even if colour kill is active.
1 «	Disable 2D comb if colour kill is active.

SDP_NOISY_HSW2_DIS_3D, Addr 90 (SDP), *Address 0xA4[6]*

A control to enable 3D combing if a noisy input has been detected and a headswitch has been detected on the input by algorithm 2.

Function

SDP_NOISY_HSW2_DIS_3D	Description
0 «	Allow 3D comb if noisy input detected even if head switch detection algorithm 2 detects head switches.
1	Disable 3D comb if noisy input detected and head switch detection algorithm 2 detects head switches.

SDP_NOISY_HSW1_DIS_3D, Addr 90 (SDP), *Address 0xA4[5]*

A control to enable 3D combing if a noisy input has been detected and a headswitch has been detected on the input by algorithm 1.

Function

SDP_NOISY_HSW1_DIS_3D	Description
0	Allow 3D comb if noisy input detected even if head switch detection algorithm 1 detects head switches.
1 «	Disable 3D comb if noisy input detected and head switch detection algorithm 1 detects head switches.

SDP_NOISY_LC_DIS_3D, Addr 90 (SDP), *Address 0xA4[4]*

A control to enable 3D combing if a noisy input has been detected and a non-standard number of lines per frame have been detected on the input

Function

SDP_NOISY_LC_DIS_3D	Description
0	Allow 3D comb if noisy input detected even if incorrect number of lines per frame detected.
1 «	Disable 3D comb if noisy input detected and incorrect lines per frame detected.

SDP_NOISY_BLK_DIS_3D, Addr 90 (SDP), *Address 0xA4[3]*

A control to enable 3D combing if a noisy input is detected and a non standard block length is detected on the input.

Function

SDP_NOISY_BLK_DIS_3D	Description
0	Allow 3D comb if noisy input detected even if SDP_BLK_NSTD detected
1 «	Disable 3D comb if noisy input detected and SDP_BLK_NSTD detected

SDP_NOISY_FLD_DIS_3D, Addr 90 (SDP), *Address 0xA4[2]*

A control to enable 3D combing if a noisy input is detected and a non standard field length is detected.

Function

SDP_NOISY_FLD_DIS_3D	Description
0	Allow 3D comb if noisy input detected even if SDP_FLD_NSTD detected
1 «	Disable 3D comb if noisy input detected and SDP_FLD_NSTD detected

SDP_NOISY_FRM_DIS_3D, Addr 90 (SDP), *Address 0xA4[1]*

A control to enable 3D combing if a noisy input is detected and a non standard frame length is detected.

Function

SDP_NOISY_FRM_DIS_3D	Description
0	Allow 3D comb if noisy signal even if SDP_FRM_NSTD detected.
1 «	Disable 3D comb if noisy input detected and SDP_FRM_NSTD detected.

SDP_NOISY_DIS_3D, Addr 90 (SDP), *Address 0xA4[0]*

A control to enable 3D combing if a noisy input is detected .

Function

SDP_NOISY_DIS_3D	Description
0	Allow 3D comb if noisy input detected.
1 «	Disable 3D comb if noisy input detected.

SDP_P60_N443_DIS_3D, Addr 90 (SDP), *Address 0xA5[7]*

A control to enable 3D combing for PAL-60 and NTSC-443 even though it does not work perfectly due to sub-optimal mathematical relationship of subcarrier frequency versus horizontal frequency for those standards.

Function

SDP_P60_N443_DIS_3D	Description
0	Use 3D comb on PAL-60 and NTSC-443 inputs.
1 «	Disable 3D comb for PAL-60 and NTSC-443 inputs.

SDP_VNOISY_HSW2_DIS_3D, Addr 90 (SDP), *Address 0xA5[6]*

A control to enable 3D combing if a very noisy input is detected and head switch is detected on the input by algorithm 2.

Function

SDP_VNOISY_HSW2_DIS_3D	Description
0 «	Allow 3D comb if very noisy input detected even if head switch detection algorithm 2 detects head switches.
1	Disable 3D comb if very noisy input detected and head switch detection algorithm 2 detects head switches.

SDP_VNOISY_HSW1_DIS_3D, Addr 90 (SDP), *Address 0xA5[5]*

A control to enable 3D combing if a very noisy input is detected and head switch is detected on the input by algorithm 1

Function

SDP_VNOISY_HSW1_DIS_3D	Description
0	Allow 3D comb if very noisy input detected even if head switch detection algorithm 1 detects head switches.
1 «	Disable 3D comb if very noisy input detected and head switch detection algorithm 1 detects head switches.

SDP_VNOISY_LC_DIS_3D, Addr 90 (SDP), *Address 0xA5[4]*

A control to enable 3D combing if a very noisy input is detected and an incorrect frame length is detected on the input.

Function

SDP_VNOISY_LC_DIS_3D	Description
0	Allow 3D comb if very noisy input detected even if incorrect number of lines per frame detected.
1 «	Disable 3D comb if very noisy input detected and incorrect lines per frame detected.

SDP_VNOISY_BLK_DIS_3D, Addr 90 (SDP), *Address 0xA5[3]*

A control to enable 3D combing if a very noisy input is detected and a non-standard block length is detected on the input.

Function

SDP_VNOISY_BLK_DIS_3D	Description
0	Allow 3D comb if very noisy input detected even if SDP_BLK_NSTD detected
1 «	Disable 3D comb if very noisy input detected and SDP_BLK_NSTD detected

SDP_VNOISY_FLD_DIS_3D, Addr 90 (SDP), *Address 0xA5[2]*

A control to enable 3D combing if a very noisy input is detected and a non-standard field length has been detected on the input.

Function

SDP_VNOISY_FLD_DIS_3D	Description
0	Allow 3D comb if very noisy input detected even if SDP_FLD_NSTD detected
1 «	Disable 3D comb if very noisy input detected and SDP_FLD_NSTD detected

SDP_VNOISY_FRM_DIS_3D, Addr 90 (SDP), *Address 0xA5[1]*

A control to enable 3D combing if a very noisy input is detected and a non-standard frame length has been detected on the input.

Function

SDP_VNOISY_FRM_DIS_3D	Description
0	Allow 3D comb if very noisy signal even if SDP_FRM_NSTD detected
1 «	Disable 3D comb if very noisy input detected and SDP_FRM_NSTD detected

SDP_VNOISY_DIS_3D, Addr 90 (SDP), *Address 0xA5[0]*

A control to enable 3D combing if a very noisy input has been detected.

Function

SDP_VNOISY_DIS_3D	Description
0	Allow 3D comb if very noisy input detected
1 «	Disable 3D comb if very noisy input detected

7.8.2 3D Comb Sensitivity

SDP_3D_COMB_LUMA_SNS[3:0], Addr 90 (SDP), *Address 0xAA[3:0]*

A control to set the 3D Comb Luma Sensitivity. Larger values increase 3D comb motion detection sensitivity to luma motion and noise. This is an unsigned control.

Function

SDP_3D_COMB_LUMA_SNS[3:0]	Description
0000	Minimum value
1000 «	Default value
1111	Maximum value

SDP_3D_COMB_LUMA_CORE[3:0], Addr 90 (SDP), *Address 0xAA[7:4]*

A control to set the 3D Comb Luma Coring. Larger values decrease 3D comb motion detection sensitivity to luma motion and noise. This is an unsigned control.

Function

SDP_3D_COMB_LUMA_CORE[3:0]	Description
1000 «	Larger values increase 3D processing

SDP_3D_COMB_CHROMA_SNS[3:0], Addr 90 (SDP), *Address 0xA9[3:0]*

A control to set 3D Comb Chroma Sensitivity. Larger values increase 3D comb motion detection sensitivity to chroma motion and noise. This is an unsigned control.

Function

SDP_3D_COMB_CHROMA_SNS[3:0]	Description
0000	Minimum value
1000 «	Default value
1111	Maximum value

SDP_3D_COMB_CHROMA_CORE[3:0], Addr 90 (SDP), *Address 0xA9[7:4]*

A control to set 3D Comb Chroma Coring. Larger values decrease 3D comb motion detection sensitivity to chroma motion and noise. This is an unsigned control.

Function

SDP_3D_COMB_CHROMA_CORE[3:0]	Description
0000	Minimum value
1000 «	Default value
1111	Maximum value

SDP_3D_COMB_NOISE_SNS[6:0], Addr 90 (SDP), *Address 0xA8[6:0]*

A control to set the 3D Comb Noise Sensitivity. Larger values allow more temporal comb for noisy RF signals but may also introduce motion error. This is an unsigned control.

Function

SDP_3D_COMB_NOISE_SNS[6:0]	Description
0000000	Minimum value
1000000 «	Default value
1111111	Maximum value

7.9 Y SHAPING FILTER

The Y shaping filter block is a programmable low pass filter with a wide variety of responses. It can be used to selectively reduce the bandwidth of the luma video signal. For some video sources that contain high frequency noise, reducing the bandwidth of the luma signal improves visual picture quality. This feature can also be used to reduce the bandwidth of the luma video as required prior to scaling.

For input signals in CVBS format, the luma shaping filters play an important role in removing the chroma component from a composite signal. YC separation must aim for the best possible crosstalk reduction while still retaining as much bandwidth as possible, especially on the luma component.

The ADV7842 contains 33 different Y shaping filters available for selection using a 6-bit selection value. Four selection registers are provided so that different Y shaping filter can be selected for the VBI region, high quality inputs, low quality inputs, and SECAM inputs.

The quality of the input signal is based on the HSync PLL speed, and some or all of the following metrics:

- Input is noisy (based on synchronization tip noise measurement and noisy threshold)
- Head switch detected (indicating the input is from a VCR)
- Input has incorrect number of lines per frame
- Block length is non standard
- Field length is non standard
- Frame length is non standard
- Input is very noisy (based on synchronization tip nose measurement and very noisy threshold)

Each of the above metrics has an enable/disable bit to specify if it is included in the quality determination of the input signal. The amount of filtering on the standard/non standard decision for these metrics is user programmable from no filtering up to two seconds. There is also a single bit control to disable all of these metrics, and base the quality decision solely on the HSync PLL speed.

A filter selection guide is used to choose the appropriate filter depending on the input signal format, type, quality, and user settings. In automatic mode, the system preserves the maximum possible bandwidth for good CVBS sources since they can be combed successfully, as well as for luma components of YUV and YC sources since they need not be combed. For poor quality signals, the system selects from a set of proprietary shaping filter responses that complement the comb filter operation in order to reduce visual artifacts. Table 11 lists the selectable filters types.

A flow diagram showing the selection process is shown in Figure 30. Filter response plots are included from Figure 31 to Figure 36.

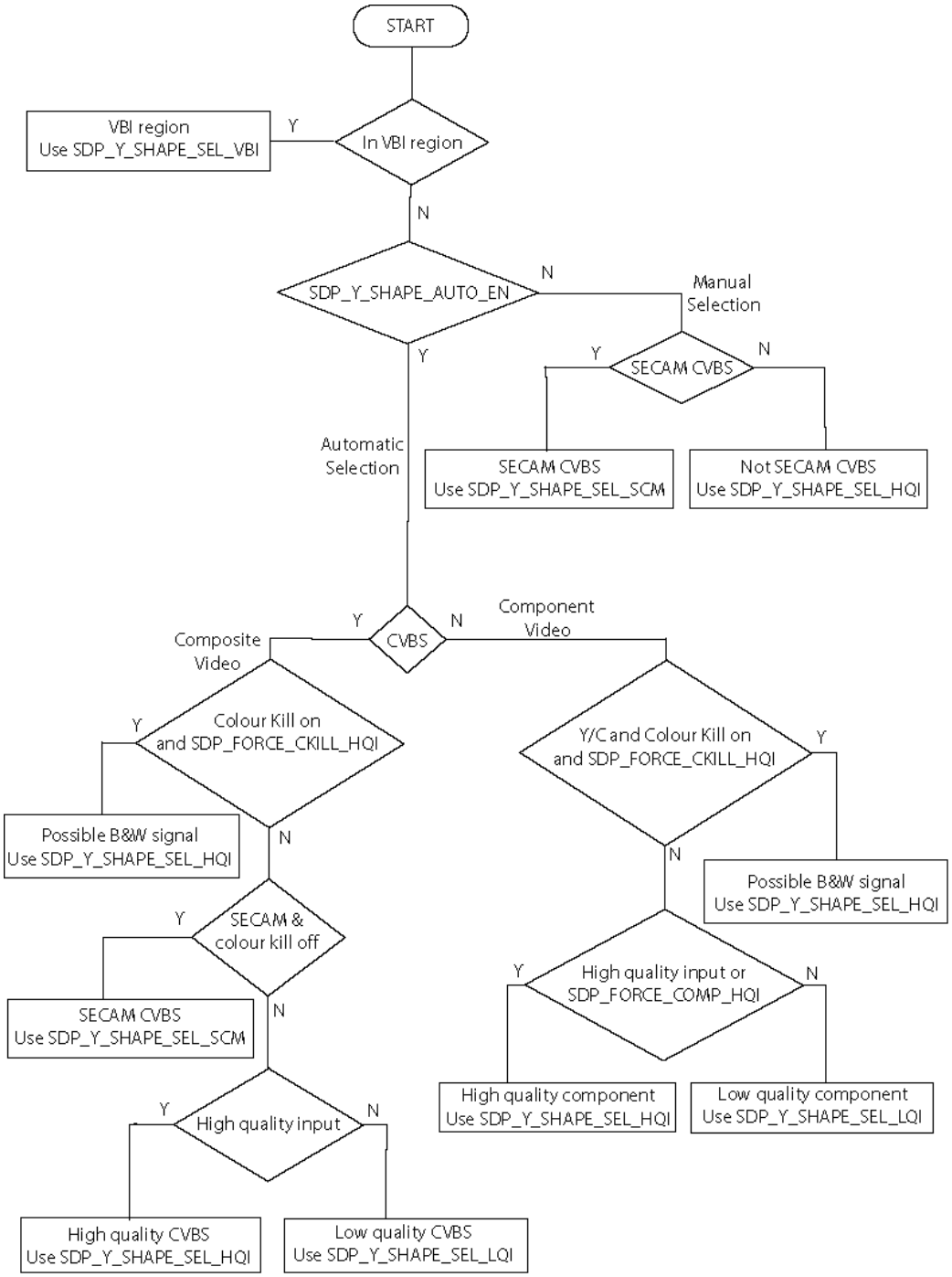


Figure 30: Y Shaping Filter Flowchart

Table 11: Y Shaping Filter Selection

No.	Y Shape Response	Filter Type	Comment
0	General purpose low-pass luma shaping filter 1	FIR, linear	Narrowest GPLPF
1	General purpose low-pass luma shaping filter 2	FIR, linear	-
2	General purpose low-pass luma shaping filter 3	FIR, linear	-
3	General purpose low-pass luma shaping filter 4	FIR, linear	-
4	General purpose low-pass luma shaping filter 5	FIR, linear	-
5	General purpose low-pass luma shaping filter 6	FIR, linear	-
6	General purpose low-pass luma shaping filter 7	FIR, linear	-
7	General purpose low-pass luma shaping filter 8	FIR, linear	-
8	General purpose low-pass luma shaping filter 9	FIR, linear	-
9	General purpose low-pass luma shaping filter 10	FIR, linear	-
10	General purpose low-pass luma shaping filter 11	FIR, linear	-
11	General purpose low-pass luma shaping filter 12	FIR, linear	
12	General purpose low-pass luma shaping filter 13	FIR, linear	-
13	General purpose low-pass luma shaping filter 14	FIR, linear	Widest Fsc GPLPF
14	PAL/NTSC CVBS combined low-pass and Fsc notch shaping filter 1	FIR, linear	Narrowest Fsc notch
15	PAL/NTSC CVBS combined low-pass and Fsc notch shaping filter 2	FIR, linear	-
16	PAL/NTSC CVBS combined low-pass and Fsc notch shaping filter 3	FIR, linear	-
17	PAL/NTSC CVBS combined low-pass and Fsc notch shaping filter 4	FIR, linear	-
18	PAL/NTSC CVBS combined low-pass and Fsc notch shaping filter 5	FIR, linear	-
19	PAL/NTSC CVBS combined low-pass and Fsc notch shaping filter 6	FIR, linear	-
20	PAL/NTSC CVBS combined low-pass and Fsc notch shaping filter 7	FIR, linear	Widest Fsc notch
21	PAL/NTSC CVBS Fsc notch shaping filter 1	FIR, linear	Narrowest Db notch
22	PAL/NTSC CVBS Fsc notch shaping filter 2	FIR, linear	-
23	PAL/NTSC CVBS Fsc notch shaping filter 3	FIR, linear	-
24	PAL/NTSC CVBS Fsc notch shaping filter 4	FIR, linear	-
25	SECAM CVBS Db notch shaping filter 1	FIR, linear	-
26	SECAM CVBS Db notch shaping filter 2	FIR, linear	-
27	SECAM CVBS Db notch shaping filter 3	FIR, linear	-
28	SECAM CVBS Dr notch shaping filter 4	FIR, linear	-
29	SECAM CVBS Dr notch shaping filter 5	FIR, linear	-
30	SECAM CVBS Dr notch shaping filter 6	FIR, linear	Widest Db notch
31	SECAM CVBS DbDr notch shaping filter 7	FIR, linear	Wide combined DbDr notch
32 to 62	Reserved	-	Do not select
63	Luma shape filter (bypass) mode	-	No filter applied

SDP_Y_SHAPE_AUTO_EN, Addr 90 (SDP), *Address 0x19[7]*

A control to allow manual or automatic selection of Y shaping filter. In manual mode the Y shaping filter is determined by the value of using SDP_Y_SHAPE_SEL_HQI[5:0]

Function

SDP_Y_SHAPE_AUTO_EN	Description
0	Manual Y shaping filter selection.
1 «	Enable automatic selection of Y shaping filter.

SDP_HQI_REQ_STD, Addr 90 (SDP), *Address 0x1A[7]*

Control that allows user to set conditions required to qualify signal as a high-quality

Function

SDP_HQI_REQ_STD	Description
0	High-quality input (HQI) requires only stable timebase
1 «	High-quality input (HQI) requires both stable and standard (nominal) timebase

SDP_FORCE_COMP_HQI, Addr 90 (SDP), *Address 0x19[6]*

A control to force the use of high-quality input (HQI) Y shaping filter when a component input is applied. When this bit is disabled the autoselection of the Y shaping filter is employed.

Function

SDP_FORCE_COMP_HQI	Description
0	Automatic selection of Y shaping filter used in component modes
1 «	Force Y shaping filter selection to use HQI filter selection in component modes

SDP_FORCE_CKILL_HQI, Addr 90 (SDP), *Address 0x18[6]*

A control to force the use of high-quality input (HQI) Y shaping filter when color kill is active. When this bit is disabled the autoselection of the Y shaping filter does not consider colour-kill mode.

Function

SDP_FORCE_CKILL_HQI	Description
0	HQI Y shaping filter is not used when colour-kill is active.
1 «	Force use of HQI Y shaping filter when colour-kill is active.

SDP_Y_SHAPE_SEL_VBI[5:0], Addr 90 (SDP), *Address 0x18[5:0]*

Y shaping filter user selection for VBI region.

Function

SDP_Y_SHAPE_SEL_VBI [5:0]	Description
111111 «	Default

SDP_Y_SHAPE_SEL_HQI[5:0], Addr 90 (SDP), *Address 0x19[5:0]*

A control to allow manual Y shaping user filter selection for high quality input signals, selects filters as shown in Y shaping filter selection table

Function

SDP_Y_SHAPE_SEL_HQI [5:0]	Description
001101 «	Default

SDP_Y_SHAPE_SEL_LQI[5:0], Addr 90 (SDP), *Address 0x1A[5:0]*

Y shaping filter manual selection for low-quality inputs (LQI), selects filters as shown in Y shaping filter selection table

Function

SDP_Y_SHAPE_SEL_LQI [5:0]	Description
010101 «	Default

SDP_Y_SHAPE_SEL_SCM[5:0], Addr 90 (SDP), *Address 0x1B[5:0]*

This control is used to select the Y shaping filter for SECAM input signals.

Function

SDP_Y_SHAPE_SEL_SCM[5:0]	Description
011110 «	Default

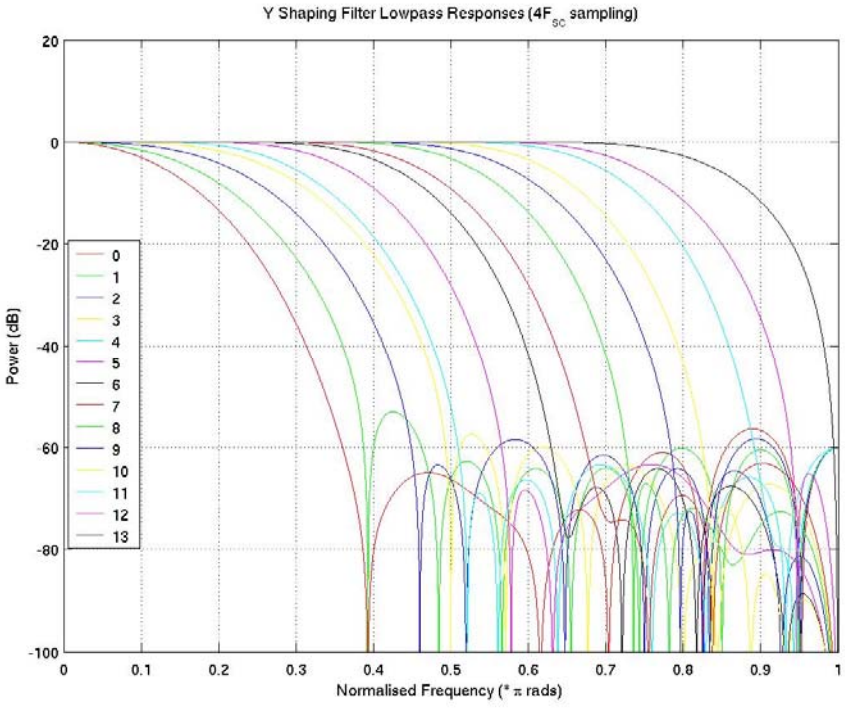


Figure 31: Y Shaping Filter Selection from No. 0 to No. 13 in Table 11

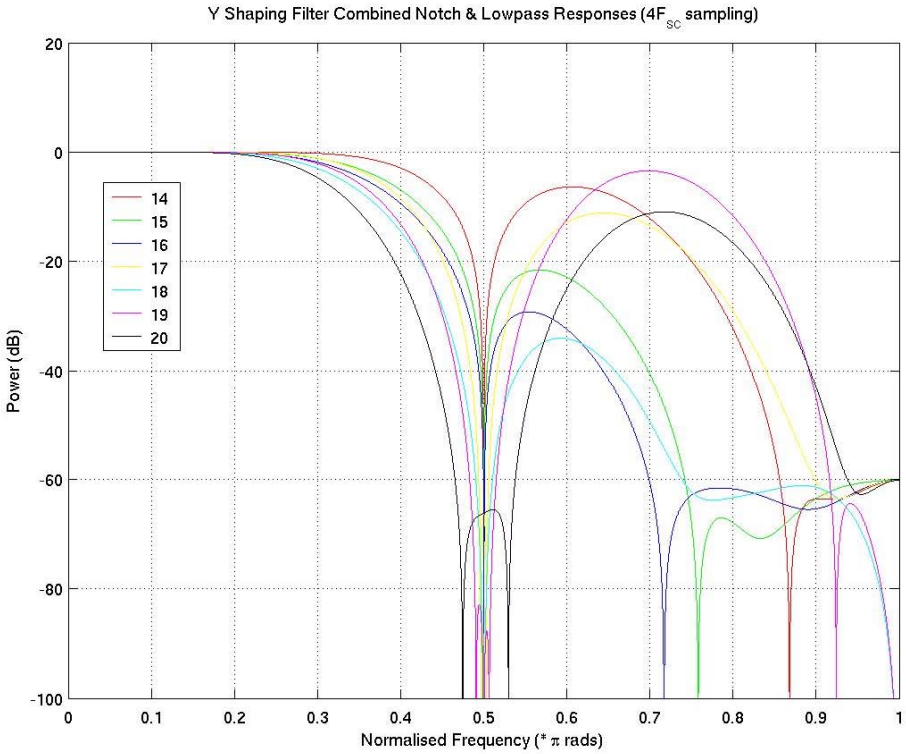


Figure 32: Y Shaping Filter Selection from No. 14 to 20 in Table 11

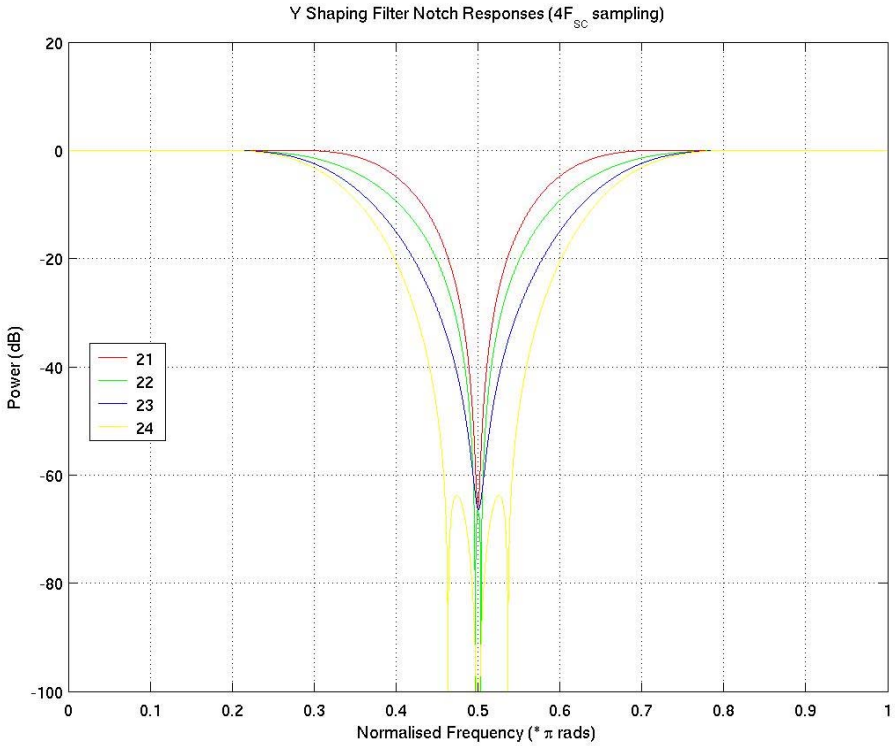


Figure 33: Y Shaping Filter Selection from No. 21 to 24 in Table 11

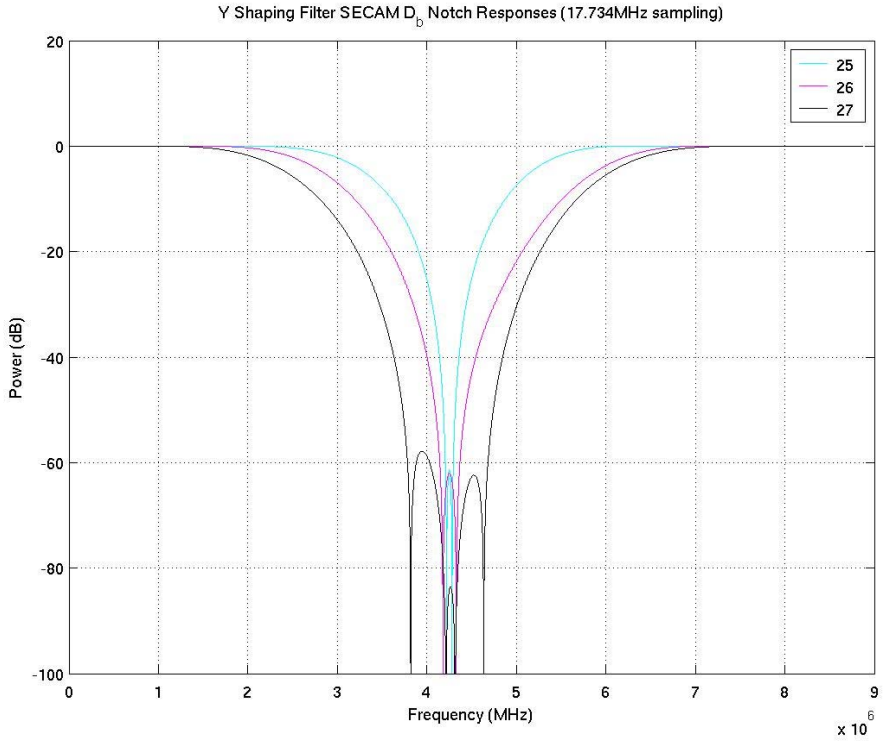


Figure 34: Y Shaping Filter Selection from No. 25 to 27 in Table 11

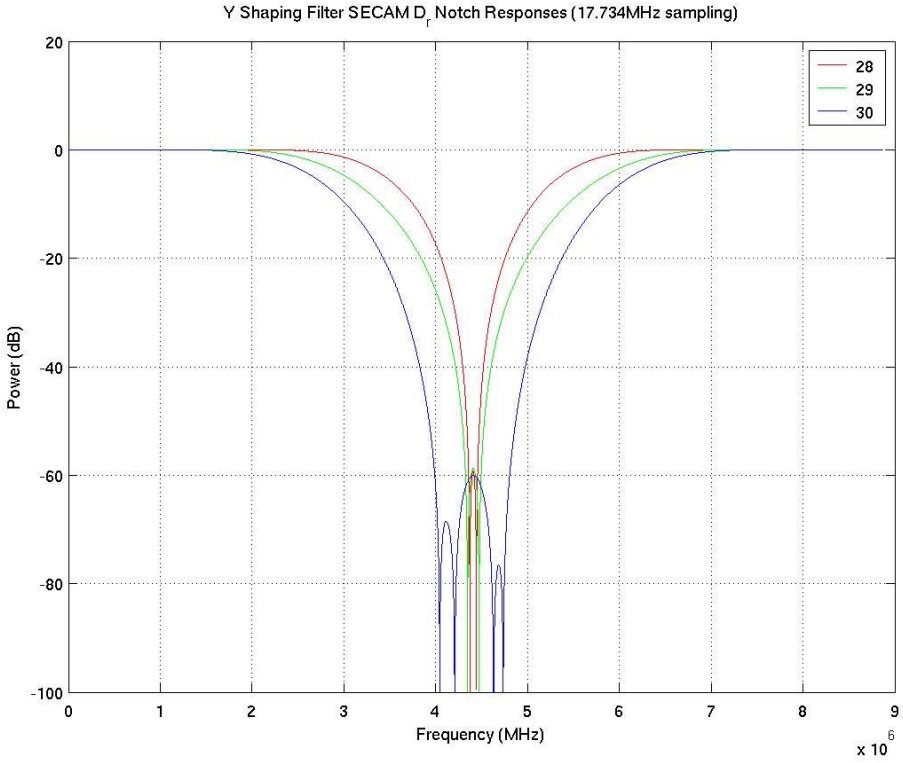


Figure 35: Y Shaping Filter Selection from No. 28 to 30 in Table 11

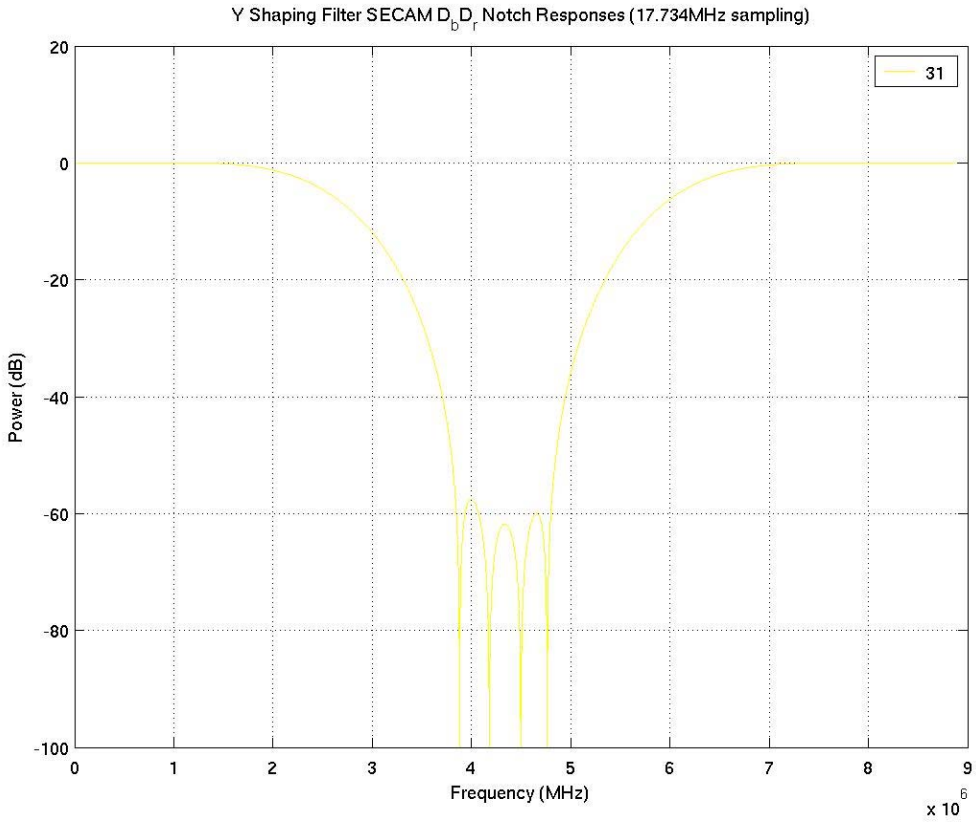


Figure 36: Y Shaping Filter Selection from No. 31 in Table 11

7.9.1 Input Shaping Filter Enables

Refer to Section 7.4.5 for synctip noise measurement, noisy, and very noisy signal detection information.

HQI_Shaping_Filter_Disable, SDP Map, *Address 0x98, [7:0]*

Function

HQI_Shaping_Filter_Disable[7:0]	Bit Name	Description
0	SDP_VNSY_DIS_SFS_STD	Allows high quality input shaping filter even if a very noisy input is detected
1	SDP_FRM_ DIS_SFS_STD	Allows high quality input shaping filter even if the input frame length is non standard
2	SDP_FLD_SFS_STD	Allows high quality input shaping filter even if the input field length is non standard
3	SDP_BLK_DIS_SFS_STD	Allows high quality input shaping filter even if the input block length is non standard
4	SDP_LC_ DIS_SFS_STD	Allows high quality input shaping filter even if the incorrect number of lines are detected on a field
5	SDP_HSW1_DIS_SFS_STD	Allows high quality input shaping filter even if head switch detect type 1 detects head switches on the input
6	SDP_HSW2_DIS_SFS_STD	Allows high quality input shaping filter even if head switch detect type 2 detects head switches on the input
7	SDP_NSY_DIS_SFS_STD	Allows high quality input shaping filter even if a noisy input is detected

SDP_VNSY_DIS_SFS_STD, Addr 90 (SDP), *Address 0x98[0]*

A control to enable HQI shaping filter when a very noisy input is detected.

Function

SDP_VNSY_DIS_SFS_STD	Description
0	Allow HQI shape filter even if noisy input detected
1 «	Disable HQI shape filter if very noisy input detected

Very noisy input is defined as a SDP_VERY_NOISY_IP = 1. Refer to the Synctip Noise Measurement, Noisy and Very Noisy Signal Detection section.

SDP_FRM_DIS_SFS_STD, Addr 90 (SDP), *Address 0x98[1]*

A control to enable HQI shaping filter when a non-standard frame is detected.

Function

SDP_FRM_DIS_SFS_STD	Description
0	Allow hqi shape filter even if SDP_FRM_NSTD detected
1 «	Disable HQI shape filter if clean input and SDP_FRM_NSTD detected

SDP_FLD_SFS_STD, Addr 90 (SDP), *Address 0x98[2]*

A control to enable HQI shaping filter when a non-standard field is detected.

Function

SDP_FLD_SFS_STD	Description
0	Allow HQI shape filter even if SDP_FLD_NSTD detected
1 «	Disable HQI shape filter if clean and SDP_FLD_NSTD

SDP_BLK_DIS_SFS_STD, Addr 90 (SDP), *Address 0x98[3]*

A control to enable HQI shaping filter when a non standard block length is detected.

Function

SDP_BLK_DIS_SFS_STD	Description
0	Allow HQI shape filter even if SDP_BLK_NSTD detected
1 «	Disable HQI shape filter if clean and SDP_BLK_NSTD detected

SDP_LC_DIS_SFS_STD, Addr 90 (SDP), *Address 0x98[4]*

A control to enable HQI shaping filter when incorrect number of lines per frame have been detected.

Function

SDP_LC_DIS_SFS_STD	Description
0	Allow HQI shape filter even if incorrect number of lines per frame detected.
1 «	Disable HQI shape filter if clean and incorrect lines per frame detected.

SDP_HSW1_DIS_SFS_STD, Addr 90 (SDP), *Address 0x98[5]*

A control to enable HQI shaping filter when a head switch is detected on the input by head switch algorithm 2.

Function

SDP_HSW1_DIS_SFS_STD	Description
0	Allow HQI shape filter even if head switch detection algorithm 1 detects head switches.
1 «	Disable HQI shape filter if head switch detection algorithm 1 detects head switches.

SDP_HSW2_DIS_SFS_STD, Addr 90 (SDP), *Address 0x98[6]*

A control to enable HQI shaping filter when a head switch is detected on the input by head switch algorithm 1.

Function

SDP_HSW2_DIS_SFS_STD	Description
0 «	Allow HQI shape filter even if head switch detection algorithm 2 detects head switches.
1	Disable HQI shape filter if head switch detection algorithm 2 detects head switches.

SDP_NSY_DIS_SFS_STD, Addr 90 (SDP), *Address 0x98[7]*

A control to enable HQI shaping filter when a noisy input is detected.

Function

SDP_NSY_DIS_SFS_STD	Description
0	Allow HQI shape filter even if noisy input detected
1 «	Disable HQI shape filter if noisy input detected

A noisy signal is defined as SDP_NOISY_IP = 1. Refer to the Synctip Noise Measurement, Noisy and Very Noisy Signal Detection section.

SDP_SHAPE_STD_FILT_SEL[2:0], Addr 90 (SDP), *Address 0x99[6:4]*

A control to select the length of filter that is used when selecting the shaping filters.

Function

SDP_SHAPE_STD_FILT_SEL[2:0]	Description
000	No filtering
001 «	0.25s
010	0.55s
011	0.81s
100	1.10s
101	1.36s
110	1.63s
111	2.00s

7.10 CHROMA SHAPING FILTER

The chroma shaping filter block can be programmed to perform a variety of low pass responses. It is used to selectively reduce the bandwidth of the chroma signal for scaling or compression.

The ADV7842 contains 20 different C shaping filters that are available for selection using a 5-bit selection value. Three selection registers are provided so that different C shaping filters can be selected for high quality inputs, low quality inputs, and SECAM inputs.

The metrics for the quality determination of the input signal are selected by SDP_HQI_REQ_STD. If this bit is set then HSync PLL speed, time base stability, and frame, field, block, or line count non standard detection are all used to determine if the input signal is high quality. If this bit is cleared then only the HSync PLL speed is used to determine the quality.

A filter selection algorithm is used to choose the appropriate filter, depending on the input signal format, type, quality, and user settings. A flow diagram showing the selection process is given in Figure 37.

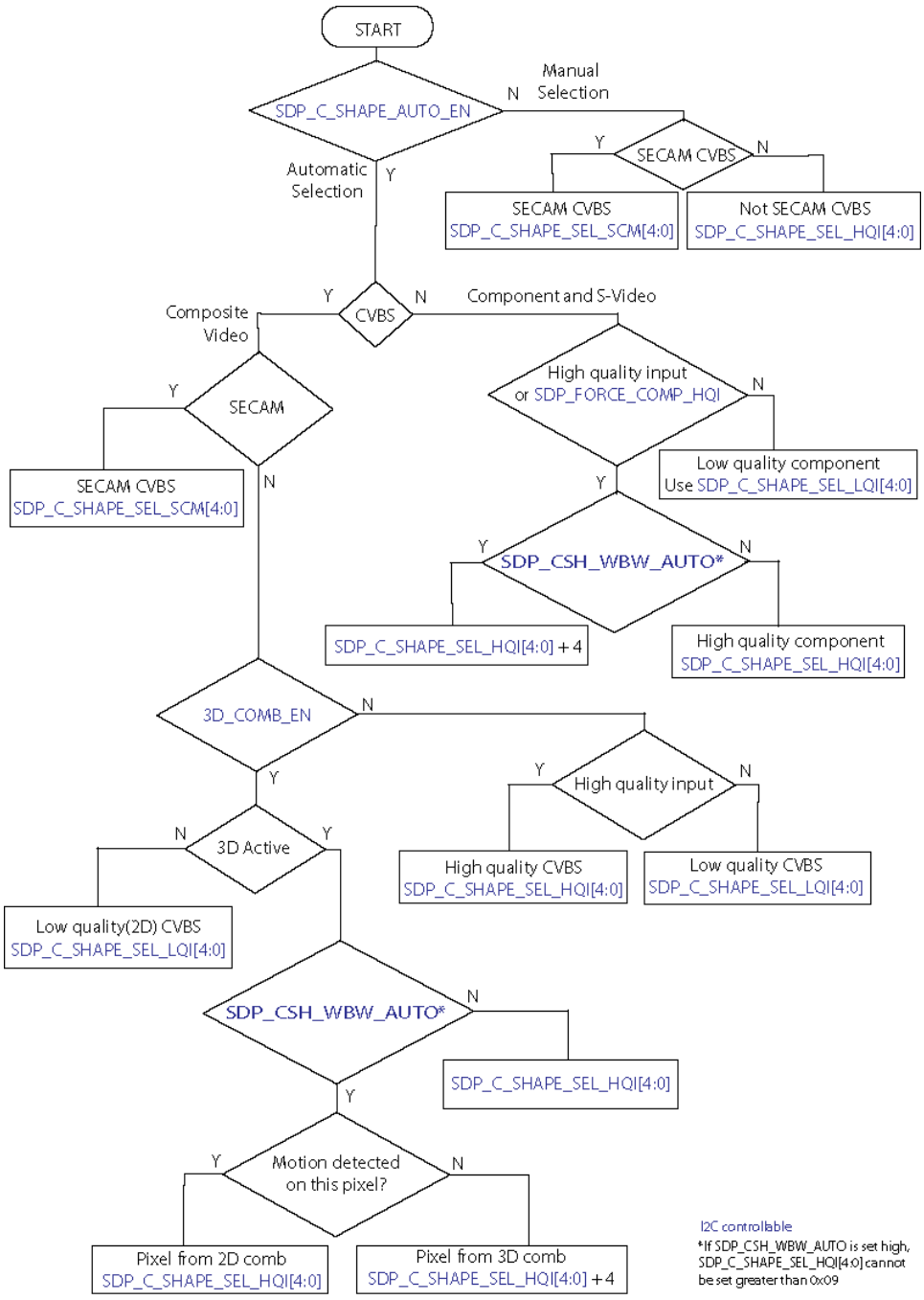


Figure 37: C Shaping Filter Flowchart

Table 12: C Shaping Filter Selection

No.	C Shape Response	Filter Type	Comment
0	CVBS-only narrow-band low-pass chroma shaping filter 1	FIR, linear	Narrowest CVBS-only LPF
1	CVBS-only narrow-band low-pass chroma shaping filter 2	FIR, linear	-
2	CVBS-only narrow-band low-pass chroma shaping filter 3	FIR, linear	-
3	CVBS-only narrow-band low-pass chroma shaping filter 4	FIR, linear	Widest CVBS-only LPF
4	General purpose low-pass chroma shaping filter 1	FIR, linear	Narrowest GPLPF
5	General purpose low-pass chroma shaping filter 2	FIR, linear	-
6	General purpose low-pass chroma shaping filter 3	FIR, linear	-
7	General purpose low-pass chroma shaping filter 4	FIR, linear	-
8	General purpose low-pass chroma shaping filter 5	FIR, linear	-
9	General purpose low-pass chroma shaping filter 6	FIR, linear	-
10	General purpose low-pass chroma shaping filter 7	FIR, linear	-
11	General purpose low-pass chroma shaping filter 8	FIR, linear	-
12	General purpose low-pass chroma shaping filter 9	FIR, linear	-
13	General purpose low-pass chroma shaping filter 10	FIR, linear	Widest GPLPF
14	SECAM narrow-band low-pass chroma shaping filter 1	IIR, non linear	Narrowest IIR LPF
15	SECAM narrow-band low-pass chroma shaping filter 2	IIR, non linear	-
16	SECAM narrow-band low-pass chroma shaping filter 3	IIR, non linear	-
17	SECAM narrow-band low-pass chroma shaping filter 4	IIR, non linear	-
18	SECAM narrow-band low-pass chroma shaping filter 5	IIR, non linear	Widest IIR LPF
19 to 30	Reserved	-	Do not select
31	Chroma shape filter (bypass) mode	-	No filter applied

SDP_C_SHAPE_AUTO_EN, Addr 90 (SDP), *Address 0x1C[7]*

A control to allow manual or automatic selection of C shaping filter. Manual selection is determined by SDP_C_SHAPE_SEL_HQI[4:0]

Function

SDP_C_SHAPE_AUTO_EN	Description
0	Manual C shaping filter selection using SDP_C_SHAPE_SEL_HQI[4:0]
1 «	Enable automatic selection of C shaping filter.

SDP_CSH_WBW_AUTO, Addr 90 (SDP), *Address 0x1C[6]*

A control to allow automatic selection of the C shaping filter to be influenced by motion detection. In areas where motion is detected and 2D combing is in operation a narrow C shaping filter is used. For still areas where no motion is detected and 3D combing is in operation a wide C shaping filter is applied. Please refer to C shaping filter flowchart.

Function

SDP_CSH_WBW_AUTO	Description
0	Disable auto C shaping filter selection based on motion, default c-shape filter selection.
1 «	Enable auto C shaping filter selection based on motion

SDP_C_SHAPE_SEL_HQI[4:0], Addr 90 (SDP), *Address 0x1C[4:0]*

This control is used to manually select C shaping filter for high-quality inputs (HQI).

Function

SDP_C_SHAPE_SEL_HQI [4:0]	Description
00100 «	Default

SDP_C_SHAPE_SEL_LQI[4:0], Addr 90 (SDP), *Address 0x1D[4:0]*

This control is used to select C shaping filter for low-quality inputs (LQI).

Function

SDP_C_SHAPE_SEL_LQI [4:0]	Description
00010 «	Default

SDP_C_SHAPE_SEL_SCM[4:0], Addr 90 (SDP), *Address 0x1E[4:0]*

This control is used to set the C shaping filter for SECAM input signals

Function

SDP_C_SHAPE_SEL_SCM[4:0]	Description
00100 «	Default

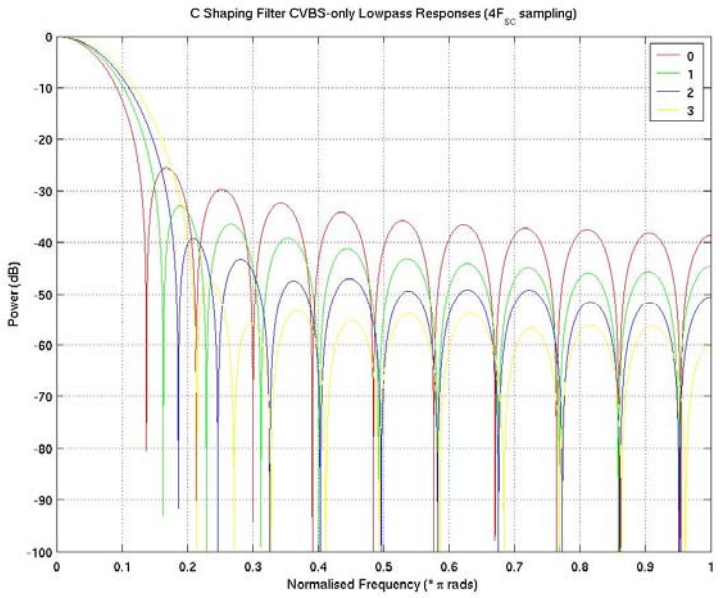


Figure 38: C Shaping Filter Selection No. 0 to 3 in Table 12

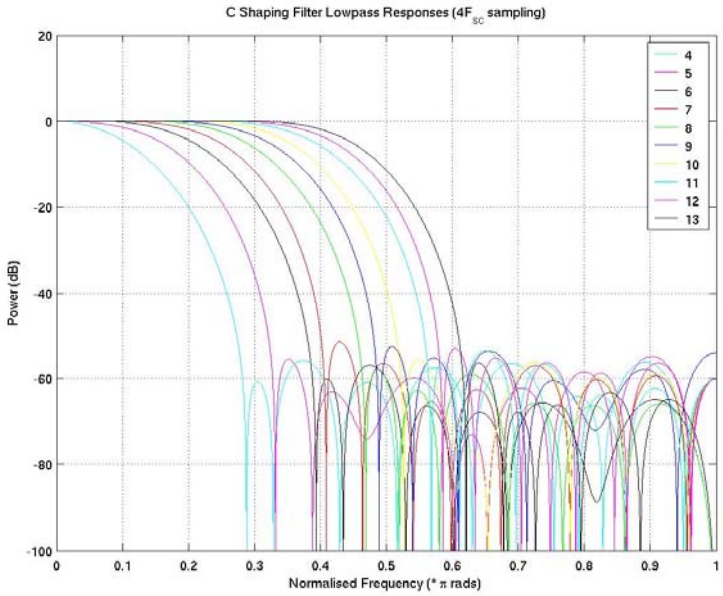


Figure 39: C Shaping Filter Selection No. 4 to 13 in Table 12

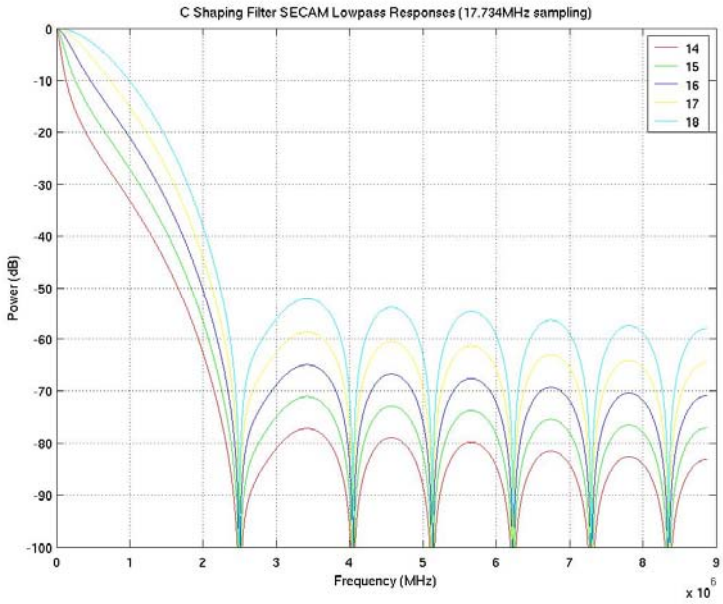


Figure 40: C Shaping Filter No. 14 to 18 in Table 12

7.11 SPLIT FILTER SELECTION

The ADV7842 offers a dynamic, pixel-by-pixel split filter alpha blending between a fixed, wide split filter and the split filter selected in the split filter selection register. The alpha blending determines if the region being combed would benefit most from a wide or a narrow split filter. The decision is based on splitting the picture into areas where a wide split filter is useable (content with very low vertical change and areas with low/moderate frequency color) and everywhere else (areas with high detail horizontally and vertically). A narrow split filter selection gives better performance on diagonal lines, but leaves more dot crawl in the final output image. The opposite is true for selecting a wide bandwidth split filter.

The split filter selection register allows one of six split filters to be chosen, thereby allowing the user control over the narrow split filter that is used in the alpha blending. The filter selection is independent of the input format since the decoder is burst locked and effectively changes its internal operating clock based on the color burst frequency. Therefore, the characteristics of the selected filter will scale with the Fsc frequency of the input signal. Figure 41 shows the frequency responses of the selectable split filters.

SDP_SPLIT_FILTER_SEL[4:0], Addr 90 (SDP), *Address 0x1F[4:0]*

A control to select the split filter frequency response for the pixel-by-pixel split filter alpha blending.

Function

SDP_SPLIT_FILTER_SEL[4:0]	Description
0xxxx	Reserved
10000	Filter no. 0
10001	Filter no. 1
10010 «	Filter no. 2
10011	Filter no. 3
10100	Filter no. 4
10101	Filter no. 5
10110	Filter no. 6
10111	Reserved
11xxx	Reserved

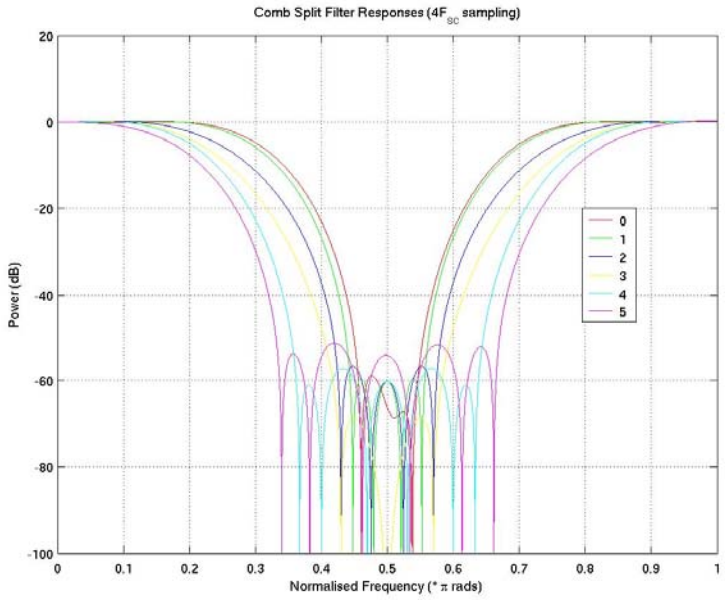


Figure 41: Split Filter Frequency Response

7.12 IF FILTER COMPENSATION

The ADV7842 offers ten different chroma path filters; four band-pass and six IF compensation, as detailed on Figure 42 and Figure 43.

SDP_IF_FILT_SEL[4:0], Addr 90 (SDP), *Address 0x20[4:0]*

The SDP_IF_FILT_SEL bits allow the user to compensate for SAW filter characteristics on a composite input as would be observed on a tuner output.

Function

SDP_IF_FILT_SEL[4:0]	Description
00000 «	Default

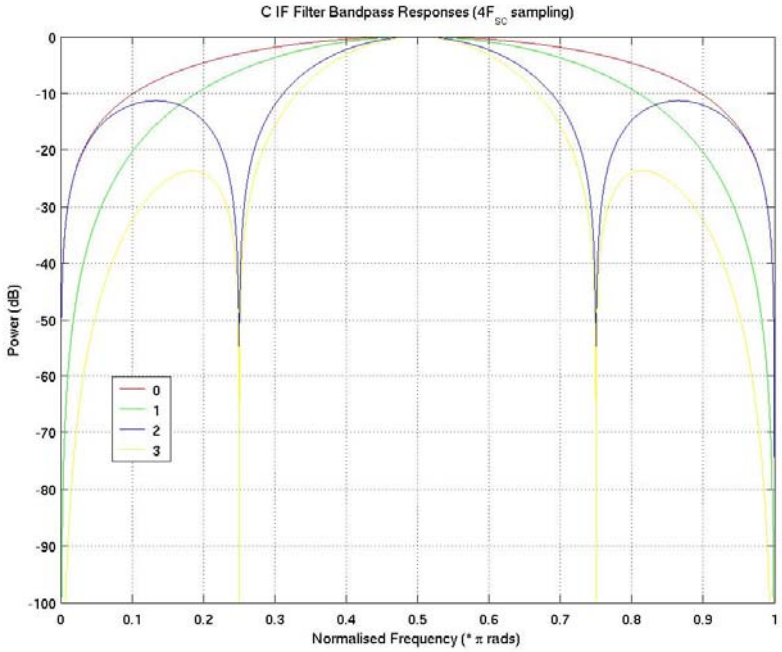


Figure 42: IF Compensation Filter Responses 0 to 3

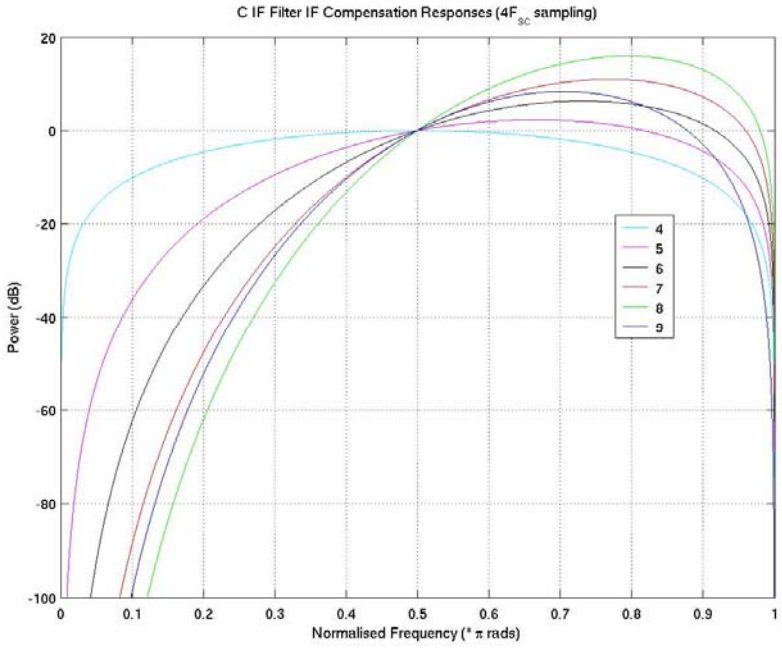


Figure 43: IF Compensation Filter Responses 4 to 9

7.13 LUMA TRANSIENT IMPROVEMENT AND CHROMA TRANSIENT IMPROVEMENT

The Luma Transient Improvement (LTI)/Chroma Transient Improvement (CTI) block enhances the picture produced in the ADV7842. The LTI/CTI block improves the steepness of the transitions. It uses adaptive peaking and non linear methods to provide enhancements without increasing noise or artifacts. The LTI block sharpens the transitions on the Y luma channel without causing adverse effects to picture quality. The CTI block operates in a similar manner on the U and V channels. Figure 44 shows a basic operational diagram.

Figure 44: LTI/CTI Operation Diagram

SDP_LTI_EN, Addr 90 (SDP), *Address 0x0E[1]*

A control to enable Luma Transient Improvement (LTI)

Function

SDP_LTI_EN	Description
0 «	Disable LTI
1	Enable LTI

SDP_CTI_EN, Addr 90 (SDP), *Address 0x0E[0]*

A control to enable Chroma Transient Improvement (CTI).

Function

SDP_CTI_EN	Description
0	Disable CTI
1 «	Enable CTI

SDP_SCM_CTI_EN, Addr 90 (SDP), *Address 0x0E[5]*

A control to enable CTI in SECAM modes.

Function

SDP_SCM_CTI_EN	Description
0	Disable CTI in SECAM modes
1 «	Enable CTI in SECAM modes

The SDP_SCM_CTI_EN bit enables an independent CTI block for SECAM modes. The SECAM signal is often clipped because of the LF-emphasis filter that is used on the transmitter side. Once a signal is restored on the receiver side, its chroma transitions are slower.

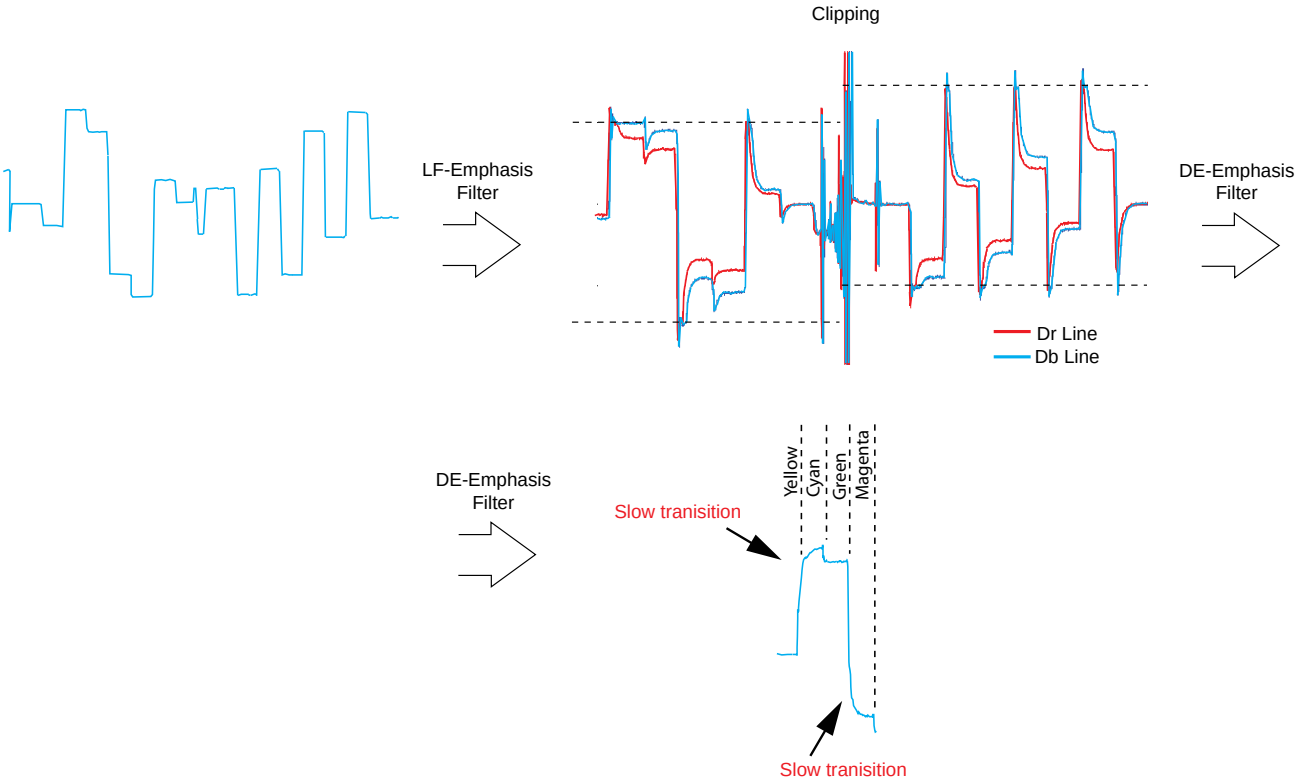


Figure 45: SECAM Signal Distortion

SECAM CTI block is designed to improve this distortion, by making transitions steeper. Amount of improvement can be selected by SDP_SCM_CTI_GAIN[1:0].

The LTI/CTI levels can be adjusted to improve picture sharpness for the luma and chroma elements of the picture. Figure 46 and Figure 47 show the LTI filter responses.

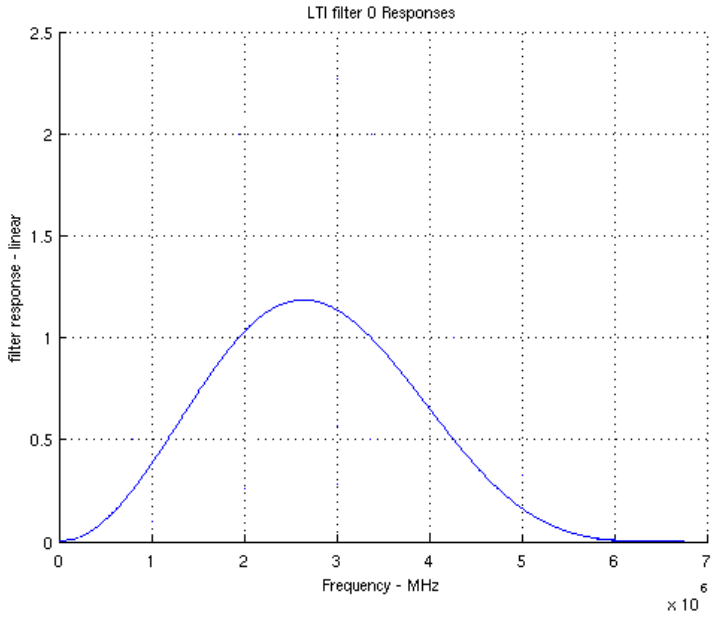


Figure 46: LTI Filter Response 0

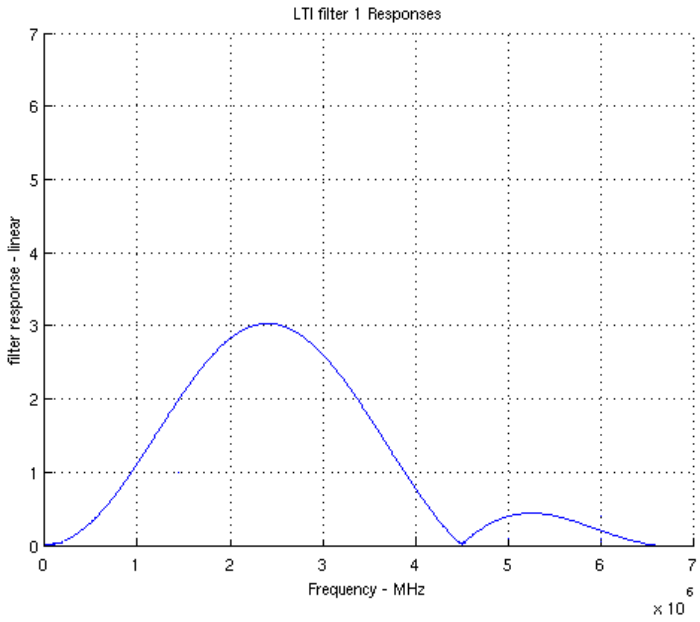


Figure 47: LTI Filter Response 1

SDP_LTI_LEVEL[6:0], Addr 90 (SDP), *Address 0x25[6:0]*

A control to set the amount of LTI applied. A larger value corresponds to the sharpening of luma transients.

Function

SDP_LTI_LEVEL[6:0]	Description
0000000 «	No transient improvement
xxxxxxx	More sharpening of luma transients

SDP_CTI_LEVEL[5:0], Addr 90 (SDP), *Address 0x26[5:0]*

A control to set the amount of CTI applied. A larger value corresponds to the sharpening of chroma transients.

Function

SDP_CTI_LEVEL[5:0]	Description
000000	No transient improvement
001111 «	Default value
xxxxxx	More sharpening of chroma transients

SDP_SCM_CTI_GAIN[1:0], Addr 90 (SDP), *Address 0x28[2:1]*

Changes Gain used for CTI (SECAM modes)

Function

SDP_SCM_CTI_GAIN[1:0]	Description
00	*0.125
01 «	*0.25
10	*0.375
11	*0.5

The LTI/CTI block allows the user to select between two filter responses. Figure 48 and Figure 49 illustrate filter response 0 and filter response 1 for CTI.

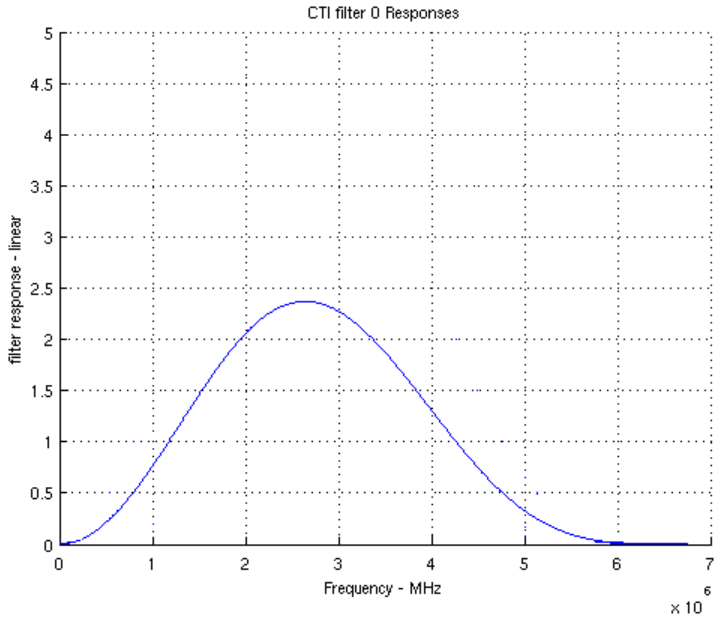


Figure 48: CTI Filter Response 0

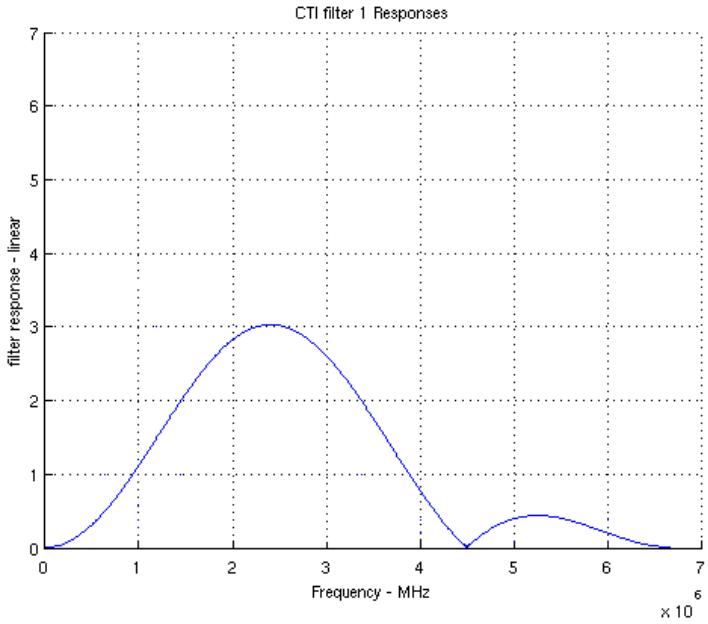


Figure 49: CTI Filter Response 1

SDP_LTI_FILT_SEL, Addr 90 (SDP), *Address 0x25[7]*

A control to select one of two filter response available in LTI operation.

Function

SDP_LTI_FILT_SEL	Description
0 «	Select filter response 0 as part of LTI
1	Select filter response 1 as part of LTI

SDP_CTI_FILT_SEL, Addr 90 (SDP), *Address 0x26[7]*

A control to select one of two filter response available for CTI operation.

Function

SDP_CTI_FILT_SEL	Description
0	Select filter response 0 as part of CTI
1 «	Select filter response 1 as part of CTI

The LTI/CTI flip registers allow the user to set a threshold on the filtered signal amplitude. Above this threshold, no LTI/CTI edge enhancements are applied. Adjusting this range allows LTI/CTI to be applied to mid to high range frequencies.

SDP_LTI_FLIP[1:0], Addr 90 (SDP), *Address 0x27[3:2]*

Filtered input signal amplitudes above this threshold receive no LTI edge enhancement

Function

SDP_LTI_FLIP[1:0]	Description
00	128
01	512
10 «	1024
11	4096

SDP_CTI_FLIP[1:0], Addr 90 (SDP), *Address 0x27[7:6]*

Filtered input signal amplitudes above this threshold receive no CTI edge enhancement

Function

SDP_CTI_FLIP[1:0]	Description
00	128
01	512
10 «	1024
11	4096

7.14 RINGING REDUCTION

The ADV7842 incorporates a ringing reduction block. This block is used to reduce ringing artifacts that can appear around sharp edges. There are two controls to use a ringing reduction block. SDP_RING_RED_EN enables a block, and SDP_RING_RED_LEVEL[6:0] sets the level of reduction applied to the signal.

SDP_RING_RED_EN, Addr 94 (SDP_IO), *Address 0x51[7]*

Enable ringing reduction block to remove ringing artifact from around sharp edges

Function

SDP_RING_RED_EN	Description
0 «	Disables ringing reduction block
1	Enables ringing reduction block

SDP_RING_RED_LEVEL[6:0], Addr 94 (SDP_IO), *Address 0x51[6:0]*

Level control for ringing reduction algorithm. Higher values give more dramatic ringing reduction

Function

SDP_RING_RED_LEVEL[6:0]	Description
0000000 «	No ringing reduction

7.15 HORIZONTAL AND VERTICAL PEAKING

The ADV7842 offers horizontal and vertical luma peaking to enhance the picture produced by the decoder. The luma peaking function operates to boost or attenuate the mid to high frequency component of the Y signal.

In horizontal and vertical peaking, the input is passed through a high-pass or band-pass filter. The input is cored and gained before it is added to/subtracted from the original signal. This ensures that all the information in the input signal is preserved. Whether the result is added to or subtracted from the input signal is determined by the SDP_H_PK_INV and SDP_V_PK_INV controls.

The horizontal and vertical peaking controls are described in Section 7.15.1 and Section 7.15.2.

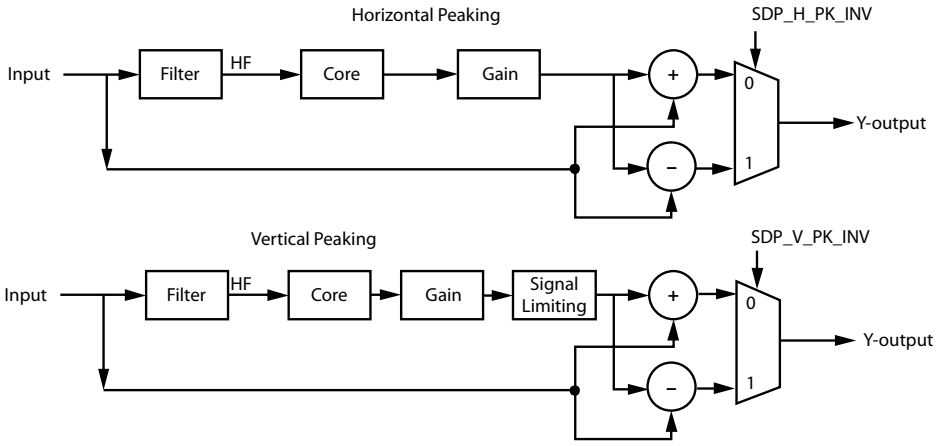


Figure 50: Peaking Block Diagram

7.15.1 Horizontal Peaking

Horizontal peaking contains several controls that are described in this section. SDP_H_PK_EN is the main control that enables or disables horizontal peaking. SDP_Y_2D_PK_EN is an additional control to enable or disable horizontal peaking filter on the 2D combed output. SDP_H_PK_GAIN[3:0], in conjunction with SDP_H_PK_INV**,** sets the gain of the output. This is in a range of -1 to 0 (for SDP_H_PK_INV = 1) or 0 to 4 (for SDP_H_PK_INV = 0). SDP_H_PK_BAND[1:0] sets the band of horizontal peaking. SDP_H_PK_CORE[2:0] is the last control that is used to set the threshold of the output, below which no high frequency is added back to the input, as shown on the Figure 51.

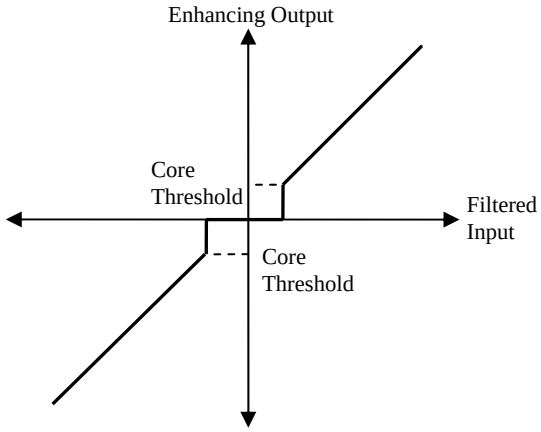


Figure 51: Core Threshold in Horizontal Peaking

SDP_H_PK_EN, Addr 90 (SDP), *Address 0x0E[2]*

A control to enable horizontal peaking filter. This is a universal peaking control it is applied after the 2D/3D mixing. It is applied regardless of whether or not 3D has been enabled.

Function

SDP_H_PK_EN	Description
0 «	Disable horizontal peaking
1	Enable horizontal peaking

SDP_Y_2D_PK_EN, Addr 90 (SDP), *Address 0x0E[4]*

A control to enable horizontal peaking filter on the 2D combed output. This peaking filter is applied to the 2D portion of the image before it is mixed with the 3D. It is important to note that it will always be applied to the 2D portion regardless or not as to weather 3D comb is enabled. The purpose of this control is that where 3D comb is enabled, 2D peaking reduces the sharpness/resolution difference perceived in areas where motion occurs. 3D areas are always very sharp due to temporal comb; 2D areas need to be peaked to compensate for softness of 2D/1D separation.

Function

SDP_Y_2D_PK_EN	Description
0	Disable horizontal peaking
1 «	Enable horizontal peaking

SDP_H_PK_INV, Addr 90 (SDP), *Address 0x22[7]*

A control to inverse Horizontal peaking filter operation.

Function

SDP_H_PK_INV	Description
0 «	Normal (gain HF)
1	Inverse peaking (attenuate HF)

SDP_H_PK_BAND[1:0], Addr 90 (SDP), *Address 0x24[1:0]*

Control to set Horizontal peaking filter band.

Function

SDP_H_PK_BAND[1:0]	Description
00 «	No filtering
01	High-Pass Filter
10	Band-Pass Filter 1
11	Band-Pass Filter 2

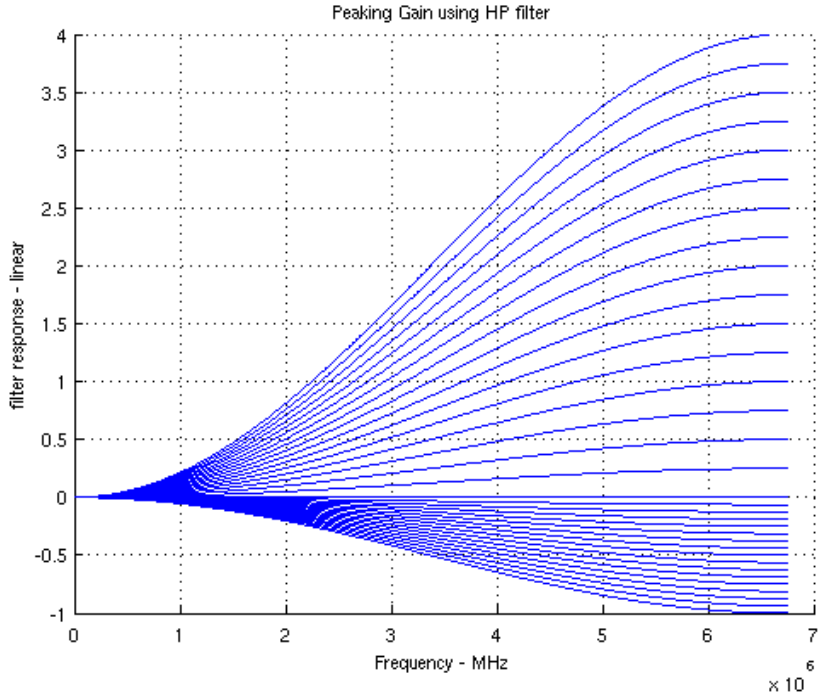


Figure 52: Horizontal Peaking High-Pass Filter

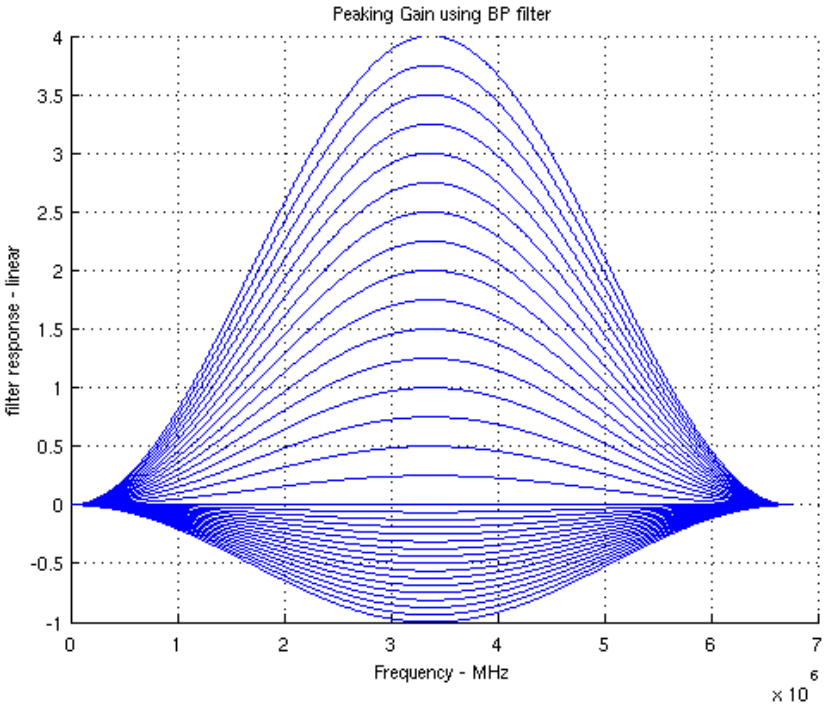


Figure 53: Horizontal Peaking Band-Pass Filter 1

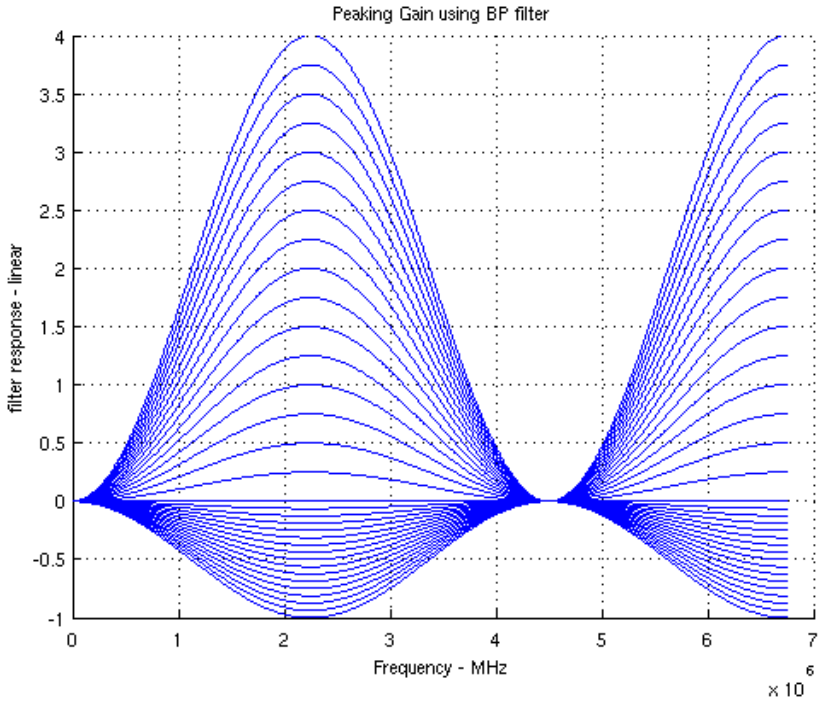


Figure 54: Horizontal Peaking Band-Pass Filter 2

SDP_H_PK_CORE[2:0], Addr 90 (SDP), *Address 0x22[2:0]*

The SDP_H_PK_CORE[2:0] bits select the horizontal threshold from the eight possible values listed in the following table. If the filtered output is less than the coring threshold, no high frequency is added back to the input. If the filter output is greater than the core threshold, it is passed through unchanged to the next stage.

Function

SDP_H_PK_CORE[2:0]	Description
000 «	0
001	8
010	16
011	24
100	32
101	40
110	48
111	56

SDP_H_PK_GAIN[3:0], Addr 90 (SDP), *Address 0x22[6:3]*

A control to adjust the gain of the horizontal peaking filter. The peaking filter can visually improve the picture by showing more definition on the picture details that contain frequency components around 3 MHz. The filter response is also user selectable using the SDP_H_PK_BAND and SDP_H_PK_INV controls. Range of 0 to 4 or 0 to -1 depending on SDP_H_PK_INV.

Function

SDP_H_PK_GAIN[3:0]	Description
0000	Minimum value; Gain = 0
0100 «	Default value; Gain = 1.06, if SDP_V_PK_INV = 0; Gain = -0.27 if SDP_V_PK_INV = 1
1111	Maximum value; Gain = 4, if SDP_V_PK_INV = 0; Gain = -1 if SDP_V_PK_INV = 1

Plots of the filter responses are shown in Figure 52 to Figure 54.

7.15.2 Vertical Peaking

Vertical peaking contains several controls that are described in this section. SDP_V_PK_EN is the main control that enables or disables Vertical peaking.

SDP_V_PK_FLIP[2:0] with SDP_V_PK_CLIP[1:0] are used to set the maximum amount of enhancement that can be added before the gain is applied. SDP_V_PK_FLIP[2:0] sets the maximum amount of enhancement. Above this value amount of enhancements will be reduced. SDP_V_PK_CLIP[1:0] is a control that is applied after SDP_V_PK_FLIP[2:0]. It sets the positive and negative values of saturation. This mechanism is explained in Figure 55.

SDP_V_PK_CORE[2:0] is the another control that is used to set the threshold of the output before applying it to gain block, below which values are cored to 0. Refer to Figure 55.

SDP_V_PK_GAIN[3:0] in conjunction with SDP_V_PK_INV sets the gain of the output that can be in the ranges of -1 to 0 (for SDP_V_PK_INV = 1) or 0 to 4 (for SDP_V_PK_INV = 0).

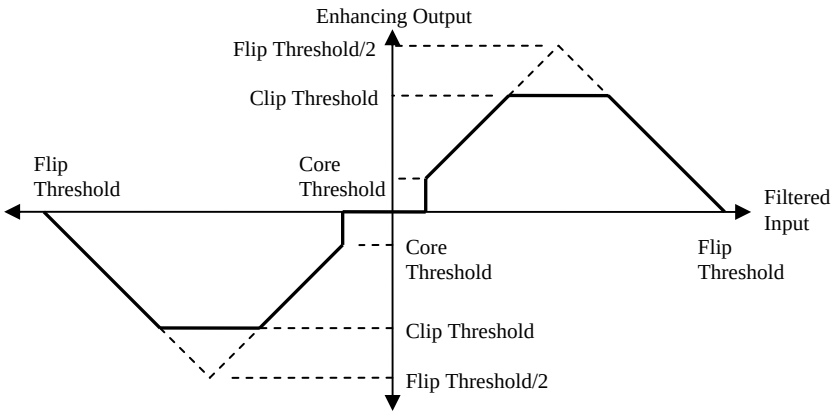


Figure 55: Peaking Controls

SDP_V_PK_EN, Addr 90 (SDP), *Address 0x0E[3]*

A control to enable vertical peaking filter

Function

SDP_V_PK_EN	Description
0 «	Disable vertical peaking
1	Enable vertical peaking

SDP_V_PK_INV, Addr 90 (SDP), *Address 0x23[7]*

A control to inverse Vertical peaking filter operation.

Function

SDP_V_PK_INV	Description
0 «	Normal (gain HF)
1	Inverse peaking (attenuate HF)

SDP_V_PK_GAIN[3:0], Addr 90 (SDP), *Address 0x23[6:3]*

A control to adjust the gain for the vertical peaking filter. The user can select to boost or attenuate the mid region of the Y spectrum around 3 MHz. The peaking filter can visually improve the picture by showing more definition on the picture details that contain frequency components around 3 MHz. To be used in conjunction with SDP_V_PK_INV range of 0 to 4 or 0 to -1 depending on SDP_V_PK_INV.

Function

SDP_V_PK_GAIN[3:0]	Description
0000	Minimum value; Gain = 0
0010 «	Default value, Gain = 0.53, if SDP_V_PK_INV = 0; Gain = -0.13 if SDP_V_PK_INV = 1
1111	Maximum value; Gain = 4, if SDP_V_PK_INV = 0; Gain = -1 if SDP_V_PK_INV = 1

The vertical peaking filter response is shown in Figure 56.

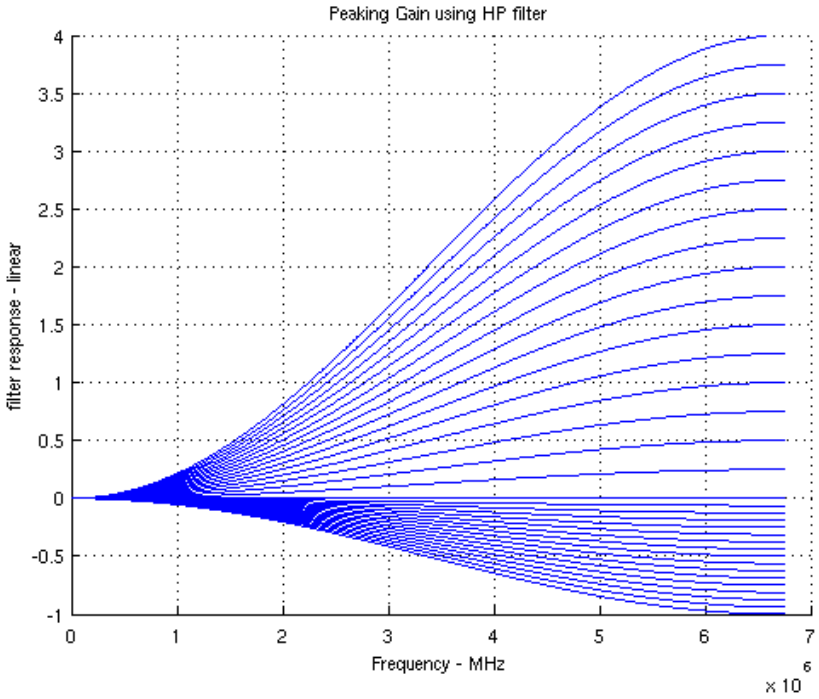


Figure 56: Vertical Peaking Filter

SDP_V_PK_CORE[2:0], Addr 90 (SDP), *Address 0x23[2:0]*

This control sets the coring threshold for Vertical filter. Signals in output of the filter below this level are cored to 0.

Function

SDP_V_PK_CORE[2:0]	Description
000 «	0
001	8
010	16
011	24
100	32
101	40
110	48
111	56

SDP_V_PK_CLIP[1:0], Addr 90 (SDP), *Address 0x24[3:2]*

A control to set the maximum amount of enhancement that can be added before the gain is applied. Set the saturation threshold on output of peaking filter. To be used in conjunction with SDP_V_PK_FLIP.

Function

SDP_V_PK_CLIP[1:0]	Description
00	Flip threshold divided by 2
01	Flip threshold divided by 7/16
10	Flip threshold divided by 3/8
11 «	Flip threshold divided by 4

SDP_V_PK_FLIP[2:0], Addr 90 (SDP), *Address 0x24[6:4]*

This bit control the upper convergence limit. Filtered input signal amplitude above this threshold receive no peaking enhancement

Function

SDP_V_PK_FLIP[2:0]	Description
000	64
001	128
010	256
011	512
100 «	1024
101	2048
110	3072
111	4095

7.16 FRAME SYNCHRONIZATION (FRAME TIME BASE CORRECTION)

The frame synchronizer block in the ADV7842 provides stable timing information and stable clock frequency in the event of irregularities on the input signal. These irregularities include items such as VCR head switches, extra lines in a frame, too few lines in a frame, non standard input frequency, or interrupted field sequence that occur during VCR trick mode inputs and during input channel changes. Nominal 480i/576i H/V/F and/or CCIR-656 type timing and data is output from the system under all conditions. Figure 57 provides a block diagram for the system.

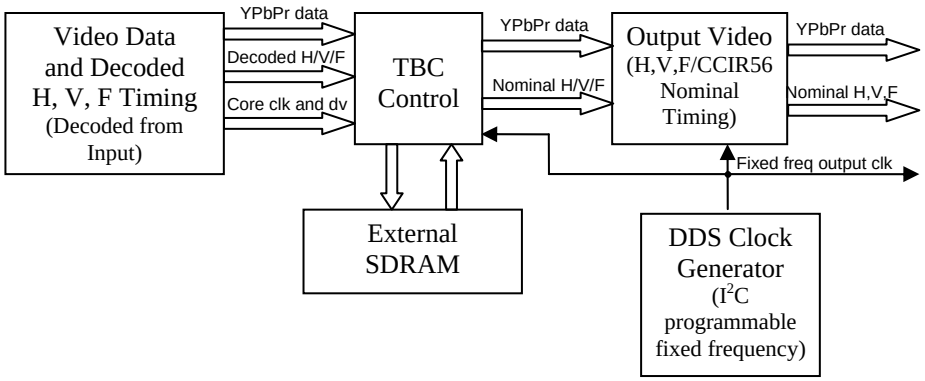


Figure 57: Frame Synchronization Block Diagram

Input video is decoded in the decoder core to component YPrPb (or equivalent) with associated HSync, VSync, and field signals. The time base correction (TBC) control block writes YPrPb data to the frame memories in the external SDRAM. The TBC control block synthesizes H, V, and F timing signals with nominal 480i/576i timing and reads YUV data from the frame memories based on nominal timing signals generated using a fixed frequency clock. This fixed frequency clock is also used to clock data out of the ADV7842. The fixed frequency clock is user programmable but is intended to operate at 13.5 MHz, 27 MHz, or 54 MHz depending on the chosen output format.

The external SDRAM memory is used to provide storage for fields of data.

The frame synchronization controls are described below.

SDP_FR_TBC_EN, Addr 90 (SDP), *Address 0x12[2]*

A control to enable frame time-based correction (TBC)

Function

SDP_FR_TBC_EN	Description
0 «	Disable frame TBC
1	Enable frame TBC

SDP_TBC_EN, Addr 90 (SDP), *Address 0x34[7]*

A control to enable the line Time Base Correction (TBC) feature.

Function

SDP_TBC_EN	Description
0	Disable line TBC
1 «	Enable line TBC

SDP_FREEZE_FRAME, Addr 94 (SDP_IO), *Address 0x6F[1]*

A control to continuously loop out a frame of video data from the TBC block. This feature will effectively allow the image to be paused on screen. When this bit is set, new data will not be updated into the Frame memory. This bit is only valid when frame TBC is enabled.

Function

SDP_FREEZE_FRAME	Description
0 «	Don't freeze frame TBC input
1	Freeze frame TBC input

7.17 FREE RUN MODE

Free Run mode is a mode that was designed to provide stable clock and predictable video, when no signal is applied to input or signal cannot be decoded. The ADV7842 can automatically activate Free Run mode (when SDP_FREE_RUN_AUTO = 1). Free run mode can be also forced regardless of input state – by SDP_FORCE_FREE_RUN.

Free Run controls default color insertion and causes ADV7842 to generate a default clock. The state in which this happens can be monitored via readback SDP_FREE_RUN.

Default color of Free Run mode can be set by registers:

- SDP_FREE_RUN_MAN_COL_EN
- SDP_FREE_RUN_Y[7:0]
- SDP_FREE_RUN_U[3:0]
- SDP_FREE_RUN_V[3:0]

ADV7842 may also output color bars instead of flat picture. To do this SDP_FREE_RUN_CBAR_EN should be set to 1.

SDP_FREE_RUN_AUTO, Addr 90 (SDP), *Address 0xDD[3]*

A control to enable automatic free-run operation. The part will enter free run if no valid input video detected.

Function

SDP_FREE_RUN_AUTO	Description
0	Do not free run even if no valid input video detected
1 «	Free run if no valid input video detected.

SDP_FREE_RUN_MAN_COL_EN, Addr 90 (SDP), *Address 0xDD[2]*

A control to enable manual setting of video data output in video mode. If set free run luma and chroma values are set by SDP_FREE_RUN_Y, SDP_FREE_RUN_V, SDP_FREE_RUN_U.

Function

SDP_FREE_RUN_MAN_COL_EN	Description
0	If in free run output decoded video data.
1 «	If in free run output manual luma and chroma values as set by SDP_FREE_RUN_Y, SDP_FREE_RUN_V, SDP_FREE_RUN_U

SDP_FREE_RUN_CBAR_EN, Addr 90 (SDP), *Address 0xDD[1]*

A control to select colour bar data to be output in manual mode.

Function

SDP_FREE_RUN_CBAR_EN	Description
0 «	If in free run mode output free- run mode data.
1	If in free run mode output colour bar data

SDP_FORCE_FREE_RUN, Addr 90 (SDP), *Address 0xDD[0]*

A control to force free-run mode irrespective of input lock status .

Function

SDP_FORCE_FREE_RUN	Description
0 «	Normal operation
1	Force free run

SDP_FREE_RUN_Y[7:0], Addr 90 (SDP), *Address 0xDE[7:0]*

Control to set Luma level to output in free run mode if SDP_FREE_RUN_MAN_COL_EN = 1

Function

SDP_FREE_RUN_Y[7:0]	Description
0x23 «	Default value

SDP_FREE_RUN_V[3:0], Addr 90 (SDP), *Address 0xDF[7:4]*

Control to set V level to output in free run mode if SDP_FREE_RUN_MAN_COL_EN = 1

Function

SDP_FREE_RUN_V[3:0]	Description
0111 «	Default value

SDP_FREE_RUN_U[3:0], Addr 90 (SDP), *Address 0xDF[3:0]*

Control to set U level to output in free run mode if SDP_FREE_RUN_MAN_COL_EN = 1

Function

SDP_FREE_RUN_U[3:0]	Description
1101 «	Default value

7.18 INTERLACED TO PROGRESSIVE CONVERSION

The ADV7842 incorporates an interlaced to progressive block that allows 480i to 480p and 576i to 576p conversions. This block operates in two basic modes: scan line duplication and scan line interpolation, as illustrated in Figure 58 and Figure 59 respectively.

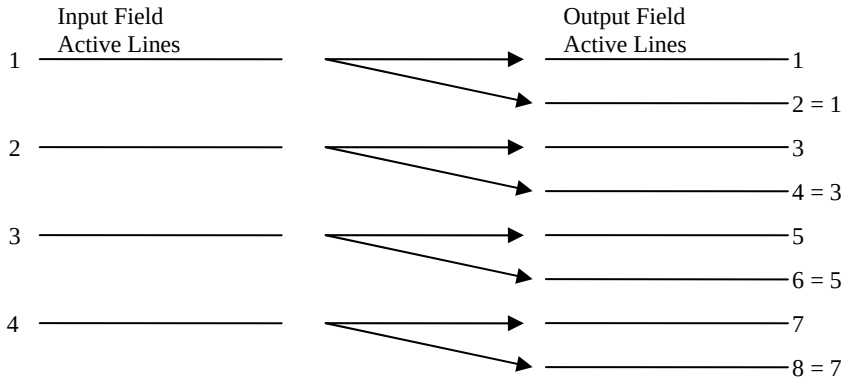


Figure 58: Scan Line Duplication

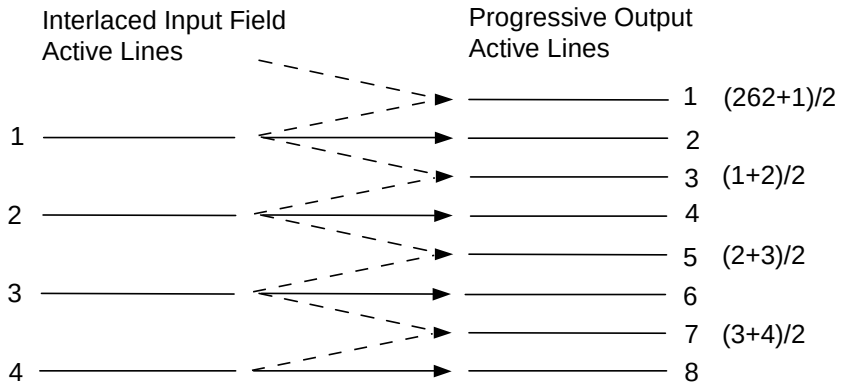


Figure 59: Scan Line Interpolation

SDP_SHIP_EN, Addr 90 (SDP), *Address 0x12[3]*

Enable interlaced to progressive conversion for 480i and 576i

Function

SDP_SHIP_EN	Description
0 «	Disable 480i - 480p and 576i - 576p conversion
1	Enable 480i - 480p and 576i - 576p conversion

SDP_SHIP_INT_EN, Addr 90 (SDP), *Address 0xD9[6]*

Selects method of interlaced to progressive conversion. This control is valid only if SDP_SHIP_EN is set to 1.

Function

SDP_SHIP_INT_EN	Description
0	Use line doubling mode for 480i-> 480p and 576i-> 576p conversion
1 «	Use line interpolation mode for 480i-> 480p and 576i-> 576p conversion

7.19 LETTERBOX DETECTION

Incoming video signals can conform to different aspect ratios (16:9 wide screen of 4:3 standard). For transmissions in the wide screen format, a digital sequence (WSS) is transmitted with the video signal. If a WSS sequence is provided, the aspect ratio of the video can be derived from digitally decoded bits contained within it.

In the absence of a WSS sequence, the letterbox detection can be used to find wide screen signals. The detection algorithm examines the active video content of lines at the start and at the end of a field. If the presence of black lines is detected, this can serve as an indication that the currently shown picture is in wide screen format.

The active video content (luminance magnitude) over a line of video is summed together. At the end of a line, this accumulated value is compared to a threshold and a decision is made whether or not a particular line is considered to be black. The threshold value needed can depend on the type of input signal, and some control is provided via SDP_LBOX_THR[4:0].

SDP_LBOX_THR[4:0], Addr 90 (SDP), *Address 0xDC[4:0]*

A control to set threshold for black line detection in letterbox detection. A larger value increases the possibility of detecting a line as black.

Function

SDP_LBOX_THR[4:0]	Description
00010 «	Default threshold for detection of black lines. The larger the value the more likely to detect the line as black.

SDP_LBOX_BLK_LVL[2:0], Addr 90 (SDP), *Address 0xDC[7:5]*

A control to set expected blank level at lbox detection block. A larger value corresponds to a higher blank level.

Function

SDP_LBOX_BLK_LVL[2:0]	Description
000 «	Default value

7.19.1 Detection at Start of Field

At the top of a field, the ADV7842 expects a section of at least six consecutive black lines of video. Once those lines are detected, the register SDP_LBOX_BLK_TOP[7:0] reports back the number of black lines actually found. By default, the ADV7842 starts looking for those black lines in synchronization with the beginning of active video, for example, straight after the last VBI video line. SDP_LBOX_BEG_DEL[3:0] allows the user to set the start of letterbox detection from the beginning of a frame on a line by line basis. The detection window closes in the middle of the field.

7.19.2 Detection at End of Field

The ADV7842 expects at least six continuous lines of black video at the bottom of a field before reporting back the number of lines actually found via the SDP_LBOX_BLK_BOT[7:0] value. The activity window for the letterbox detection (end of field) starts in the middle of the active field. Its end is programmable via SDP_LBOX_END_DEL[3:0].

7.19.3 Detection at Mid Range

Some transmissions of wide screen video include subtitles within the lower black box. If the ADV7842 finds at least two black lines, followed by some more non black video (for example, the subtitle), and finally followed by the remainder of the bottom black block, it

reports back a mid count via SDP_LBOX_BLK_SUB_BOT[7:0]. If no subtitles are found, SDP_LBOX_BLK_SUB_BOT[7:0] reports the same number as SDP_LBOX_BLK_BOT[7:0].

Notes:

There is a two field delay in the reporting of any line count parameters. There is no letterbox detected bit. The user is requested to read the SDP_LBOX_BLK_TOP[7:0] and SDP_LBOX_BLK_BOT[7:0] register values, and come to a conclusion about the presence of letterbox type video in the software.

SDP_LBOX_BLK_TOP[7:0], SDP Map, *Address 0x4C, [7:0]*
SDP_LBOX_BLK_BOT[7:0], SDP Map, *Address 0x4D, [7:0]*
SDP_LBOX_BLK_SUB_BOT[7:0], SDP Map, *Address 0x4E, [7:0]*

Access Information			
Signal Name	**Address**	**Default**	**Description**
SDP_LBOX_BLK_TOP[7:0]	0x4C	Read back only	Number of black lines detected at top of the field
SDP_LBOX_BLK_BOT[7:0]	0x4D	Read back only	Number of black lines detected at bottom of the field
SDP_LBOX_BLK_SUB_ BOT[7:0]	0x4E	Read back only	Number of black lines detected at bottom of the field (including subtitle lines)

SDP_LBOX_BLK_TOP[7:0], Addr 90 (SDP), *Address 0x4C[7:0] (Read Only)*

A letterbox readback control to indicate the number of black lines detected at the top of the field.

Function

SDP_LBOX_BLK_TOP[7:0]	**Description**
xxxxxxxx	Readback

SDP_LBOX_BLK_BOT[7:0], Addr 90 (SDP), *Address 0x4D[7:0] (Read Only)*

A letterbox readback control to indicate the number of black lines detected at the bottom of the field.

Function

SDP_LBOX_BLK_BOT[7:0]	**Description**
xxxxxxxx	Readback

SDP_LBOX_BLK_SUB_BOT[7:0], Addr 90 (SDP), *Address 0x4E[7:0] (Read Only)*

A letterbox readback control to indicate the number of black lines detected at the bottom of the field. This includes subtitle lines.

Function

SDP_LBOX_BLK_SUB_BOT[7:0]	**Description**
xxxxxxxx	Readback

SDP_LBOX_BEG_DEL[3:0], Addr 90 (SDP), *Address 0xDB[3:0]*

A control to delay letterbox detection begin line versus default position.

Function

SDP_LBOX_BEG_DEL[3:0]	Description
1000 «	Letterbox detection aligned with the start line of active video. Window starts after VBI data line.

SDP_LBOX_END_DEL[3:0], Addr 90 (SDP), *Address 0xDB[7:4]*

A control to set delay letterbox detection end line versus default position.

Function

SDP_LBOX_END_DEL[3:0]	Description
1000 «	Letterbox detection ends with the last active line of video on a field

7.20 SDP AV CODE INSERTION AND CONTROLS

This section describes the I^2C based controls that affect:

- Insertion of AV codes into the data stream
- Data blanking during the vertical blank interval (VBI)

Note that some of the decoded VBI data is being inserted during the horizontal blanking interval.

SDP_SAV_EN, Addr 94 (SDP_IO), *Address 0xB2[2]*

A control to enable the insertion of SAV codes into the digital data-stream for SD core modes.

Function

SDP_SAV_EN	Description
0	Do not insert SAV codes
1 «	Insert SAV codes

SDP_EAV_EN, Addr 94 (SDP_IO), *Address 0xB2[3]*

A control to enable the insertion of EAV codes into the digital data-stream for SD core modes.

Function

SDP_EAV_EN	Description
0	Do not insert EAV codes
1 «	Insert EAV codes

SDP_SPLIT_AV_CODE, Addr 94 (SDP_IO), *Address 0xB3[5]*

A control to enable splitting SDP SAV/EAV codes across all channels

Function

SDP_SPLIT_AV_CODE	Description
0	Don't split SDP SAV/EAV codes across all channels.
1 «	Split SDP SAV/EAV codes across channels

REPL_AV_CODE

See the description of the REPL_AV_CODE register to replicate the AV codes from the luma path into the chroma path.

SDP_SAV_POS_ADJ[11:0], Addr 94 (SDP_IO), *Address 0x92[3:0]; Address 0x93[7:0]*

A control to adjust the SAV position from its default position. This a 2s complement control.

Function

SDP_SAV_POS_ADJ[11:0]	Description
000000000000 «	SAV code default position
xxxxxxxxxxxx	SAV code adjustment

SDP_EAV_POS_ADJ[11:0], Addr 94 (SDP_IO), *Address 0x90[3:0]; Address 0x91[7:0]*

A control to adjust the EAV position from its default position. This a 2s complement control.

Function

SDP_EAV_POS_ADJ[11:0]	Description
000000000000 «	EAV code default position
xxxxxxxxxxxx	EAV code position adjustment

SDP_BLANK_C_VBI, Addr 90 (SDP), *Address 0x18[7]*

Setting SDP_BLANK_C_VBI high, the Cr and Cb values of all VBI lines are blanked. This is done so that any data that comes during VBI is not decoded as color and output through Cr and Cb. As a result, it should be possible to send VBI lines into the decoder, then output them through an encoder again and they should appear undistorted. Without this blanking, any wrongly decoded color would are encoded by the video encoder and, therefore, the VBI lines would be distorted.

Function

SDP_BLANK_C_VBI	Description
0	Pass through colour as decoded during VBI lines
1 «	Blank colour during VBI lines

7.21 SDP BLANKING CODES INSERTION

The ADV7842 inserts blanking codes by default. However it is possible to pass through decoded video data during horizontal and/or vertical blanking interval. These settings for main bus can be changed with SDP_HBLANK_EN and SDP_VBLANK_EN bits. For auxiliary bus SDP_AUX_HBLANK_EN and SDP_AUX_VBLANK_EN are appropriate.

SDP_VBLANK_EN, Addr 94 (SDP_IO), *Address 0xB0[7]*

A control to insert blanking codes or pass-through decoded video data during vertical blanking interval.

Function

SDP_VBLANK_EN	Description
0	Pass through decoded video data during vertical blanking interval
1 «	Insert blanking codes during vertical blanking interval (location equal to V bit)

SDP_HBLANK_EN, Addr 94 (SDP_IO), *Address 0xB0[6]*

A control to insert blanking codes or pass-through decoded video data during horizontal blanking interval.

Function

SDP_HBLANK_EN	Description
0	Pass through decoded video data during horizontal blanking interval
1 «	Insert blanking codes during horizontal blanking interval (location equal to H bit)

SDP_AUX_HBLANK_EN, Addr 94 (SDP_IO), *Address 0xC8[0]*

A control to select the insertion of blanking codes or decoded video data during the horizontal interval (location equal to H bit) for the auxiliary pixel bus datastream.

Function

SDP_AUX_HBLANK_EN	Description
0	Pass through decoded video data during horizontal blanking interval
1 «	Insert blanking codes during horizontal blanking interval (location equal to H bit)

SDP_AUX_VBLANK_EN, Addr 94 (SDP_IO), *Address 0xC8[1]*

A control to select the insertion of blanking codes or decoded video data during the vertical interval (location equal to V bit) for the auxiliary pixel bus datastream.

Function

SDP_AUX_VBLANK_EN	Description
0	Pass through decoded video data during vertical blanking interval
1 «	Insert blanking codes during vertical blanking interval (location equal to V bit)

7.22 SDP SYNCHRONIZATION OUTPUT SIGNALS

7.22.1 HSync Timing Configuration

The following controls allow the user to configure the behavior of the HSync on the output pin only:

HSync timing adjustments:

- SDP_HS_BEG_ADJ[11:0]
- SDP_HS_WIDTH[11:0]
- SDP_FHE_TOG_INV
- SDP_FHO_TOG_INV

HSync polarity adjustment:

- SDP_HS_POL

SDP_HS_POL, Addr 94 (SDP_IO), *Address 0xB1[0]*

A control to change polarity of HS/CS

Function

SDP_HS_POL	Description
0	Inverted HS/CS pin polarity
1 «	Default HS/CS pin polarity

SDP_HS_BEG_ADJ[11:0], Addr 94 (SDP_IO), *Address 0x94[3:0]; Address 0x95[7:0]*

The SDP_HS_BEG_ADJ[11:0] and SDP_HS_WIDTH[11:0] bits allow the user to freely position the HSync signal applied to the output pin within the video line. The values in the SDP_HS_BEG_ADJ[11:0] and SDP_HS_WIDTH[11:0] bits are measured in pixel units from the default falling edge position of the HSync. Using both values, the user can program both the position and the width of the HSync output signal. The SDP_HS_BEG_ADJ[11:0] adjusts the leading and trailing edge positions, hence adjusting the HSync pulse. The number applied to the register offsets the HSync pulse position with respect to the default value. The number is a twos complement value, which allows both positive and negative edge movement.

Function

SDP_HS_BEG_ADJ[11:0]	Description
0x000 «	Default value

SDP_HS_WIDTH[11:0], Addr 94 (SDP_IO), *Address 0x96[3:0]; Address 0x97[7:0]*

The SDP_HS_WIDTH[11:0] bits allow the user to freely adjust the width of the HSync pulse within the video line. The values in the SDP_HS_WIDTH[11:0] bits are measured in pixel units from the falling edge of HSync. The position of this edge is controlled by placing an unsigned binary number into the SDP_HS_BEG_ADJ[11:0] bits.

Function

SDP_HS_WIDTH[11:0]	Description
0x020 «	Default value (unsigned control)

SDP_FHE_TOG_INV, Addr 94 (SDP_IO), *Address 0xB0[5]*

A control to change the default field transition position for the field signal for even fields. The field transition can be at the beginning or in the middle of the line.

Function

SDP_FHE_TOG_INV	Description
0 «	Use default horizontal transition position for field signal on even fields
1	Swap horizontal transition position for field signal on even fields. Swap between beginning of the line and middle of the line

SDP_FHO_TOG_INV, Addr 94 (SDP_IO), *Address 0xB0[4]*

A control to change the default field transition position for the field signal for odd fields. The field transition can be at the beginning or in the middle of the line.

Function

SDP_FHO_TOG_INV	Description
0 «	Use default horizontal transition position for field signal on odd fields
1	Swap horizontal transition position for field signal on odd fields. Swap between beginning of the line and middle of the line

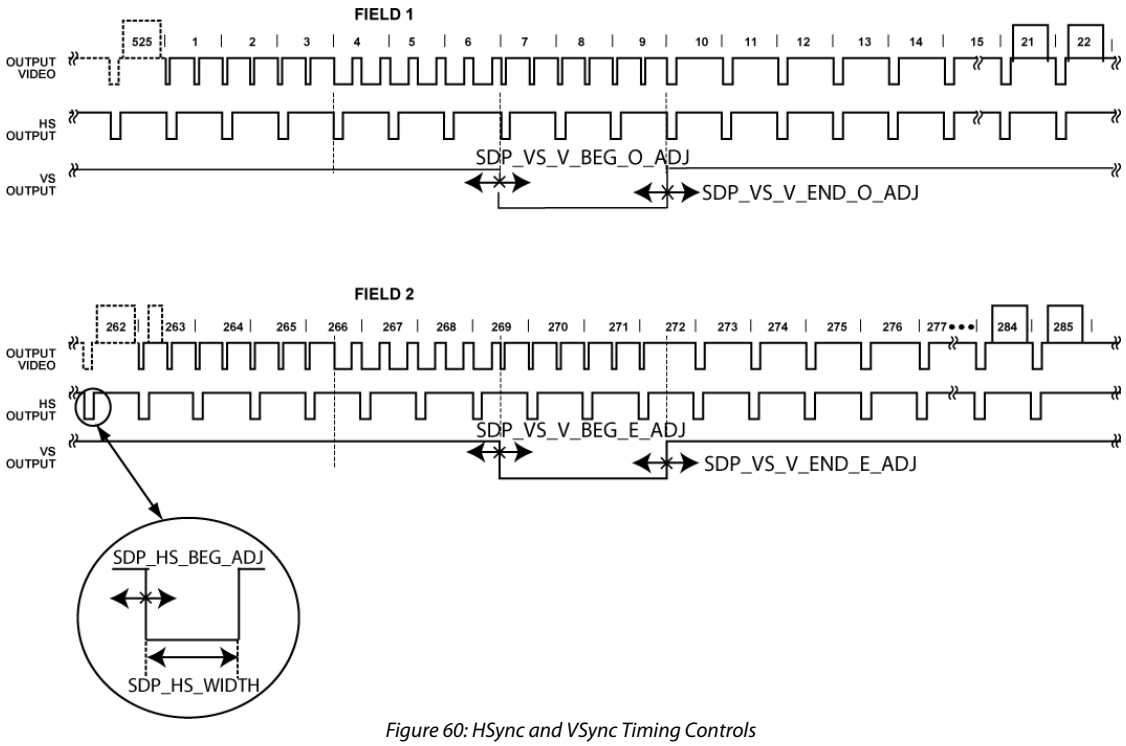


Figure 60: HSync and VSync Timing Controls

7.22.2 VSync and FIELD Configuration

The following controls allow the user to configure the behavior of the VSync and FIELD signal on the output pins VS/FIELD and FIELD/DE:

- Polarity pin adjustments:
  - SDP_VS_POL
  - SDP_FLD_POL
- VSync timing adjustments relative to HSync:
  - SDP_VSF_H_BEG_ADJ[11:0]
  - SDP_VSF_H_MID_ADJ[11:0]
- VSync timing adjustments:
  - SDP_VS_V_BEG_O_ADJ[5:0]
  - SDP_VS_V_BEG_E_ADJ[5:0]
  - SDP_VS_V_END_O_ADJ[5:0]
  - SDP_VS_V_END_E_ADJ[5:0]
  - SDP_VHO_BEG_INV
  - SDP_VHO_END_INV
  - SDP_VHE_BEG_INV
  - SDP_VHE_END_INV

SDP_VS_POL, Addr 94 (SDP_IO), *Address 0xB1[1]*

A control to change polarity of VS/FIELD

Function

SDP_VS_POL	Description
0 «	Default VS/FIELD pin polarity
1	Inverted VS/FIELD pin polarity

SDP_FLD_POL, Addr 94 (SDP_IO), *Address 0xB1[2]*

A control to change polarity of FIELD/DE

Function

SDP_FLD_POL	Description
0	Inverted FIELD/DE pin polarity
1 «	Default FIELD/DE pin polarity

SDP_VSF_H_BEG_ADJ[11:0], Addr 94 (SDP_IO), *Address 0x9C[3:0]; Address 0x9D[7:0]*

The SDP_VSF_H_BEG_ADJ[11:0] bits adjust the VS/FIELD output relative to the HSync position. The values are measured in pixel units from the falling edge of HSync. This control is used when the VSync and Field outputs are coincident with HSync. The position of the VSync and Field relative to the HSync is controlled by placing a two's complement number into the SDP_VSF_H_BEG_ADJ[11:0] bits.

Function

SDP_VSF_H_BEG_ADJ[11:0]	Description
0x000 «	Default value

SDP_VSF_H_MID_ADJ[11:0], Addr 94 (SDP_IO), *Address 0x9E[3:0]; Address 0x9F[7:0]*

The SDP_VSF_H_MID_ADJ[11:0] bits adjust the SDP VS/FIELD output relative to the HSync position within the video line. The values are measured in pixel units from the falling edge of HSync. This control is used when the VSync or Field changes approximately midway between HSyncs. The position of the VSync and Field relative to the HSync is controlled by placing a twos complement number into the SDP_VSF_H_MID_ADJ[11:0] bits.

Function

SDP_VSF_H_MID_ADJ[11:0]	Description
0x000 «	Default value

SDP_VS_V_BEG_O_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xA8[5:0]*

Adjust SDP VSync pin begin line relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_VS_V_BEG_O_ADJ[5:0]	Description
000100 «	Default value

SDP_VS_V_BEG_E_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xA9[5:0]*

Adjust SDP VSync pin begin line relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_VS_V_BEG_E_ADJ[5:0]	Description
000100 «	Default value

SDP_VS_V_END_O_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xAA[5:0]*

Adjust SDP VSync pin end line relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_VS_V_END_O_ADJ [5:0]	Description
000100 «	Default value

SDP_VS_V_END_E_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xAB[5:0]*

Adjust SDP VSync pin end line relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_VS_V_END_E_ADJ[5:0]	Description
000100 «	Default value

SDP_FLD_TOG_O_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xA6[5:0]*

Adjust SDP field pin transition relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_FLD_TOG_O_ADJ[5:0]	Description
000100 «	Default value

SDP_FLD_TOG_E_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xA7[5:0]*

Adjust SDP field pin transition relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_FLD_TOG_E_ADJ[5:0]	Description
000100 «	Default value

SDP_VHO_BEG_INV, Addr 94 (SDP_IO), *Address 0xB0[0]*

A control to change the default field transition and beginning of Vsync positions for the VSync signal for odd fields. The field transition can be at the beginning or in the middle of the line.

Function

SDP_VHO_BEG_INV	Description
0 «	Use default horizontal field transition and beginning of Vsync positions for VSync signals on odd fields.
1	Swap horizontal field transition position and beginning of VSync positions for Vsync signals on odd fields. Swap between the beginning and middle of the line

SDP_VHE_BEG_INV, Addr 94 (SDP_IO), *Address 0xB0[1]*

A control to change the default field transition and beginning of Vsync positions for the VSync signal for even fields. The field transition can be at the beginning or in the middle of the line.

Function

SDP_VHE_BEG_INV	Description
0 «	Use default horizontal field transition and beginning of Vsync positions for VSync signals on even fields.
1	Swap horizontal field transition position and beginning of VSync positions for Vsync signals on even fields. Swap between the beginning and middle of the line

SDP_VHO_END_INV, Addr 94 (SDP_IO), *Address 0xB0[2]*

A control to change the default field transition and end of Vsync positions for the VSync signal for odd fields. The field transition can be at the beginning or in the middle of the line.

Function

SDP_VHO_END_INV	Description
0 «	Use default horizontal field transition and end of Vysnc positions for VSync signals on odd fields.
1	Swap horizontal field transition and end of Vsync positions for Vsync on odd fields. Swap between the beginning and middle of the line.

SDP_VHE_END_INV, Addr 94 (SDP_IO), *Address 0xB0[3]*

A control to change the default field transition and end of Vsync positions for the VSync signal for even fields. The field transition can be at the beginning or in the middle of the line.

Function

SDP_VHE_END_INV	Description
0 «	Use default horizontal field transition and end of Vsync positions for VSync signals on even fields.
1	Swap horizontal field transition and end of Vsync positions for Vsync signals on even fields. Swap between beginning and middle of the line.

7.22.3 DE Configuration

The following controls allow the user to configure the behavior of the DE signal on the output pin:

- DE polarity adjustment:
  - SDP_DE_POL
- DE timing adjustments relative to VSync:
  - SDP_DE_V_BEG_O_ADJ[5:0]
  - SDP_DE_V_BEG_E_ADJ[5:0]

- SDP_DE_V_END_O_ADJ[5:0]
- SDP_DE_V_END_E_ADJ[5:0]

- DE timing adjustments relative to HSync:
  - SDP_DE_H_BEG_ADJ[11:0]
  - SDP_DE_H_END_ADJ[11:0]

SDP_DE_POL, Addr 94 (SDP_IO), *Address 0xB1[4]*

A control to change polarity of DE

Function

SDP_DE_POL	Description
0	Inverted DE polarity
1 «	Default DE polarity

SDP_DE_V_BEG_O_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xAC[5:0]*

Adjust SDP DE pin begin line relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_DE_V_BEG_O_ADJ [5:0]	Description
000100 «	Default value

SDP_DE_V_BEG_E_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xAD[5:0]*

Adjust SDP DE pin begin line relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_DE_V_BEG_E_ADJ[5:0]	Description
000100 «	Default value

SDP_DE_V_END_O_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xAE[5:0]*

Adjust SDP DE pin end line relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_DE_V_END_O_ADJ [5:0]	Description
000100 «	Default value

SDP_DE_V_END_E_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xAF[5:0]*

Adjust SDP DE pin end line relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_DE_V_END_E_ADJ[5:0]	Description
000100 «	Default value

SDP_DE_H_BEG_ADJ[11:0], Addr 94 (SDP_IO), *Address 0x98[3:0]; Address 0x99[7:0]*

Adjust SDP DE horizontal begin position versus default, 2s complement

Function

SDP_DE_H_BEG_ADJ[11:0]	Description
0x000 «	Default value

The SDP_DE_H_END_ADJ[11:0] bits allow the user to adjust the horizontal DE trailing edge position relative to the HSync. The values are measured in pixel units from the falling edge of HSync.

The position of the horizontal DE trailing edge is controlled by placing a twos complement number into the SDP_DE_H_END_ADJ[11:0] bits. The number is a twos complement value that allows both positive and negative edge movement.

SDP_DE_H_END_ADJ[11:0], Addr 94 (SDP_IO), *Address 0x9A[3:0]; Address 0x9B[7:0]*

Adjust SDP DE horizontal end position versus default, 2s complement

Function

SDP_DE_H_END_ADJ[11:0]	Description
0x000 «	Default value

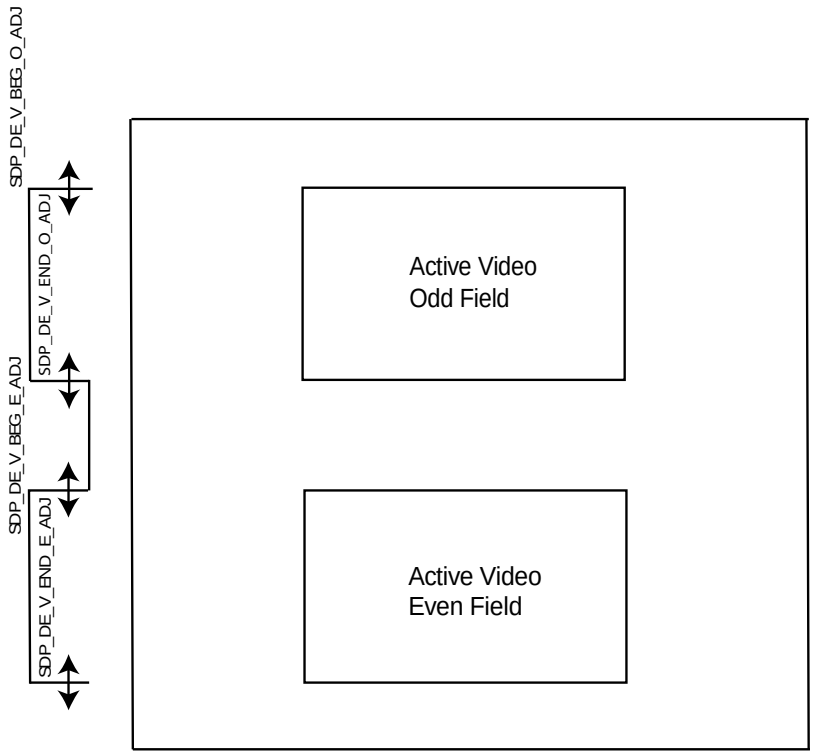


Figure 61: DE Timing Controls

7.22.4 CSync Signal Configuration

The SDP_CS_POL control allows the user to configure the polarity of the CSync signal on the output pin.

SDP_CS_POL, Addr 94 (SDP_IO), *Address 0xB1[3]*

A control to change polarity of HS/CS

Function

SDP_CS_POL	Description
0	Inverted HS/CS polarity
1 «	Default HS/CS polarity

7.22.5 656 Codes Configuration

The following controls allow the user to configure the embedded AV codes:

- SDP_F_BIT_POL
- SDP_V_BIT_POL
- SDP_V_BEG_O_ADJ[5:0]
- SDP_V_BEG_E_ADJ[5:0]
- SDP_V_END_O_ADJ[5:0]
- SDP_V_END_E_ADJ[5:0]
- SDP_F_TOG_O_ADJ[5:0]
- SDP_F_TOG_E_ADJ[5:0]

SDP_F_BIT_POL, Addr 94 (SDP_IO), *Address 0xB1[5]*

A control to change polarity of FIELD bit

Function

SDP_F_BIT_POL	Description
0	Inverted FIELD bit polarity
1 «	Default FIELD bit polarity

SDP_V_BIT_POL, Addr 94 (SDP_IO), *Address 0xB1[6]*

A control to change polarity of V bit

Function

SDP_V_BIT_POL	Description
0	Inverted V bit polarity
1 «	Default V bit polarity

SDP_V_BEG_O_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xA0[5:0]*

Adjust SDP 656 code V bit low to high transition relative to default, only +ve recommended, 2s complement

Function

SDP_V_BEG_O_ADJ[5:0]	Description
000100 «	Default value

SDP_V_BEG_E_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xA1[5:0]*

Adjust SDP 656 code V bit low to high transition relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_V_BEG_E_ADJ[5:0]	Description
000100 «	Default value

SDP_V_END_O_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xA2[5:0]*

Adjust SDP 656 code V bit high to low transition relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_V_END_O_ADJ[5:0]	Description
000100 «	Default value

SDP_V_END_E_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xA3[5:0]*

Adjust SDP 656 code V bit high to low transition relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_V_END_E_ADJ[5:0]	Description
000100 «	Default value

SDP_F_TOG_O_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xA4[5:0]*

Adjust SDP 656 code F bit transition relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_F_TOG_O_ADJ[5:0]	Description
000100 «	Default value

SDP_F_TOG_E_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xA5[5:0]*

Adjust SDP 656 code F bit transition relative to default. This is a 2's complement adjustment control, only positive adjustment is recommended.

Function

SDP_F_TOG_E_ADJ[5:0]	Description
000100 «	Default value

SDP_V_BEG_TRICK_O_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xB4[5:0]*

Adjust SDP 656 code V bit low to high transition in VCR trick modes relative to default. This is a 2's complement control, only positive adjustments are recommended. Adjustment applied when a VCR trick modes is detected.

Function

SDP_V_BEG_TRICK_O_ADJ[5:0]	Description
000100 «	Default value

SDP_V_BEG_TRICK_E_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xB5[5:0]*

Adjust SDP 656 code V bit low to high transition in VCR trick modes relative to default. This is a 2's complement control, only positive adjustments are recommended. Adjustment applied when a VCR trick modes is detected.

Function

SDP_V_BEG_TRICK_E_ADJ[5:0]	Description
000100 «	Default value

SDP_V_END_TRICK_O_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xB6[5:0]*

Adjust SDP 656 code V bit high to low transition in VCR trick modes relative to default. This is a 2's complement control, only positive adjustments are recommended. Adjustment applied when a VCR trick modes is detected.

Function

SDP_V_END_TRICK_O_ADJ[5:0]	Description
000100 «	Default value

SDP_V_END_TRICK_E_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xB7[5:0]*

Adjust SDP 656 code V bit high to low transition in VCR trick modes relative to default. This is a 2's complement control, only positive adjustments are recommended. Adjustment applied when a VCR trick modes is detected.

Function

SDP_V_END_TRICK_E_ADJ[5:0]	Description
000100 «	Default value

SDP_F_TOG_TRICK_O_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xB8[5:0]*

Adjust SDP 656 code F bit transition in VCR trick modes relative to default. This is a 2's complement control, only positive adjustments are recommended. Adjustment applied when a VCR trick modes is detected.

Function

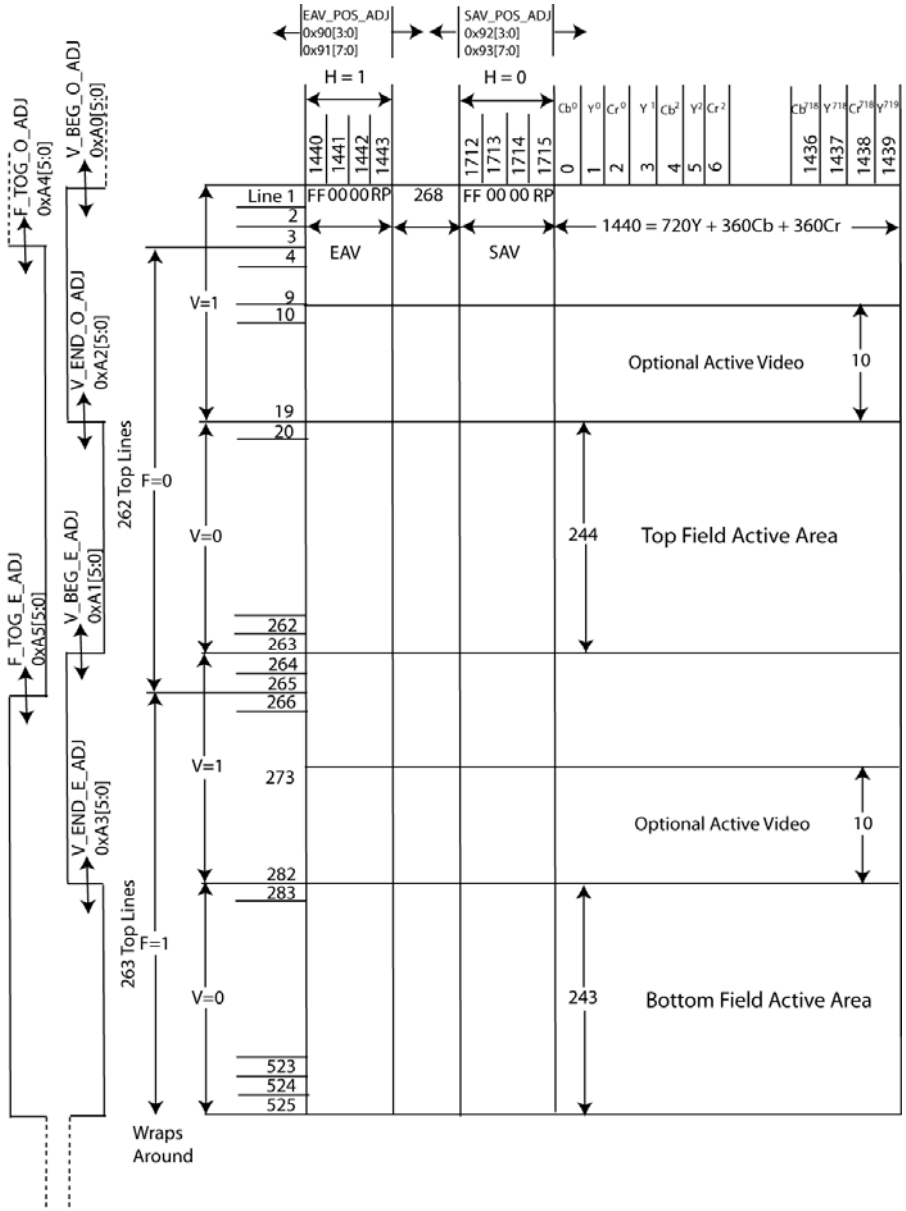
SDP_F_TOG_TRICK_O_ADJ[5:0]	Description
000100 «	Default value

SDP_F_TOG_TRICK_E_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xB9[5:0]*

Adjust SDP 656 code F bit transition in VCR trick modes relative to default. This is a 2's complement control, only positive adjustments are recommended. Adjustment applied when a VCR trick modes is detected.

Function

SDP_F_TOG_TRICK_E_ADJ[5:0]	Description
000100 «	Default value



Note: Timing example shown for 525i through SD core.

Figure 62: Embedded Timing Controls (SDP)

7.22.6 Retiming Video Adjustments

Register SDP_RET_VID_ADJ allows retiming video adjustments in VBI.

SDP_RET_VID_ADJ, Addr 94 (SDP_IO), *Address 0xE0[5]*

A control to enable retiming video adjustments to VBI.

Function

SDP_RET_VID_ADJ	Description
0 «	Apply video adjustments when programmed
1	Retime video adjustments to VBI

7.22.7 Manual Color Space Conversion Matrix

The ADV7842 features Color Space Converter (CSC) in SDP core. CSC provides any-to-any color space conversion support, that is, it support formats such as RGB, YUV, YCrCb, and many other color spaces.

The CSC matrix in the ADV7842 is a 3 x 3 matrix with full programmability of all coefficients in the matrix in manual mode. Each coefficient is 12-bit wide to ensure signal integrity is maintained in the CSC section. The CSC contains three identical processing channels, one of which is shown in Figure 63. The main inputs labeled In_A, In_B, and In_C comes from SDP core output. Each input to the individual channels to the CSC is multiplied by a separate coefficient for each channel. In Figure 63, these coefficients are marked A1, A2, and A3. The variable labeled A4 in Figure 63 is used as an offset control for channel A in the CSC. There is also a further CSC control bit labeled SDP_CSC_SCALE; this bit can be used to accommodate coefficients that extend the supported range. The functional diagram for a single channel in the CSC as per Figure 63 is repeated for the other two remaining channels B and C. The coefficients for these channels are called B1, B2, B3, B4, C1, C2, C3, and C4. To enable Manual Color Space Conversion Matrix SDP_CSC_AUTO bit should be set high.

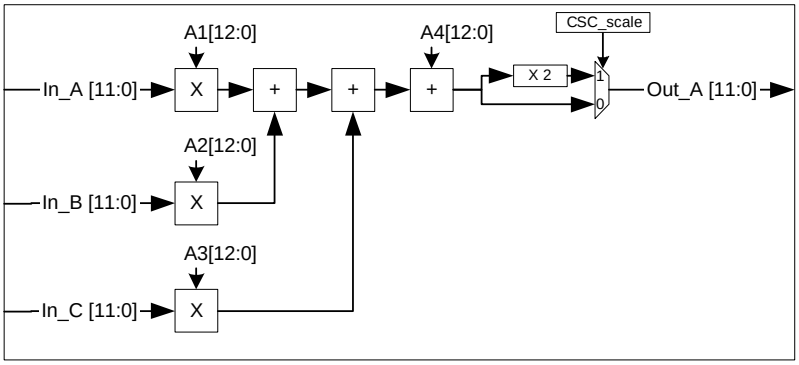


Figure 63: Single CSC Channel

The coefficients mentioned previously are detailed in Table 13 along with the default values for these coefficients.

Table 13: CSC Coefficients

Function Bit	SDP_IO Map Address	Reset Value (Hex)	Description
SDP_A1[12:0]	0xE0[4:0], 0xE1[7:0]	0x07D2	Coefficients for channel A
SDP_A2[12:0]	0xE2[4:0], 0xE3[7:0]	0x0000	
SDP_A3[12:0]	0xE4[4:0], 0xE5[7:0]	0x0040	
SDP_B1[12:0]	0xE8[4:0], 0xE9[7:0]	0x0000	Coefficients for channel B
SDP_B2[12:0]	0xEA[4:0], 0xEB[7:0]	0x0926	
SDP_B3[12:0]	0xEC[4:0], 0xED[7:0]	0x0000	
SDP_C1[12:0]	0xF0[4:0], 0xF1[7:0]	0x0000	Coefficients for channel C
SDP_C2[12:0]	0xF2[4:0], 0xF3[7:0]	0x0000	
SDP_C3[12:0]	0xF4[4:0], 0xF5[7:0]	0x0681	
SDP_CSC_SCALE	0xE0[7]	0x00	Scaling for CSC formula
SDP_A4[14:0]	0xE6[6:0], 0xE7[7:0]	0x7F00	Offsets for the three channels

Function Bit	SDP_IO Map Address	Reset Value (Hex)	Description
SDP_B4[14:0]	0xEE[6:0], 0xEF[7:0]	0x0000	
SDP_C4[14:0]	0xF6[6:0], 0xF7[7:0]	0x0000	

SDP_CSC_AUTO, Addr 94 (SDP_IO), *Address 0xE0[6]*

A control to select CSC operation

Function

SDP_CSC_AUTO	Description
0	Use manual CSC coefficients
1 «	Use automatic CSC coefficients

SDP_CSC_SCALE, Addr 94 (SDP_IO), *Address 0xE0[7]*

A control to set CSC gain

Function

SDP_CSC_SCALE	Description
0 «	CSC scaler set to 1
1	CSC scaler set to 2

This bit allows the control to accommodate coefficients that extend the supported range of the DPP. For further details, refer to the examples in Section 7.22.7.2.

SDP_A1[12:0], Addr 94 (SDP_IO), *Address 0xE0[4:0]; Address 0xE1[7:0]*

CSC A1 coefficient for SDP output colour space converter

Function

SDP_A1[12:0]	Description
0x07D2 «	Default value

SDP_A2[12:0], Addr 94 (SDP_IO), *Address 0xE2[4:0]; Address 0xE3[7:0]*

CSC A2 coefficient for SDP output colour space converter

Function

SDP_A2[12:0]	Description
0x0000 «	Default value

SDP_A3[12:0], Addr 94 (SDP_IO), *Address 0xE4[4:0]; Address 0xE5[7:0]*

CSC A3 coefficient for SDP output colour space converter

Function

SDP_A3[12:0]	Description
0x0040 «	Default value

SDP_A4[14:0], Addr 94 (SDP_IO), *Address 0xE6[6:0]; Address 0xE7[7:0]*

CSC A4 coefficient for SDP output colour space converter

Function

SDP_A4[14:0]	Description
0x7F00 «	Default value

SDP_B1[12:0], Addr 94 (SDP_IO), *Address 0xE8[4:0]; Address 0xE9[7:0]*

CSC B1 coefficient for SDP output colour space converter

Function

SDP_B1[12:0]	Description
0x0000 «	Default value

SDP_B2[12:0], Addr 94 (SDP_IO), *Address 0xEA[4:0]; Address 0xEB[7:0]*

CSC B2 coefficient for SDP output colour space converter

Function

SDP_B2[12:0]	Description
0x0926 «	Default value

SDP_B3[12:0], Addr 94 (SDP_IO), *Address 0xEC[4:0]; Address 0xED[7:0]*

CSC B3 coefficient for SDP output colour space converter

Function

SDP_B3[12:0]	Description
0x0000 «	Default value

SDP_B4[14:0], Addr 94 (SDP_IO), *Address 0xEE[6:0]; Address 0xEF[7:0]*

CSC B4 coefficient for SDP output colour space converter

Function

SDP_B4[14:0]	Description
0x0000 «	Default value

SDP_C1[12:0], Addr 94 (SDP_IO), *Address 0xF0[4:0]; Address 0xF1[7:0]*

CSC C1 coefficient for SDP output colour space converter

Function

SDP_C1[12:0]	Description
0x0000 «	Default value

SDP_C2[12:0], Addr 94 (SDP_IO), *Address 0xF2[4:0]; Address 0xF3[7:0]*

CSC C2 coefficient for SDP output colour space converter

Function

SDP_C2[12:0]	Description
0x0000 «	Default value

SDP_C3[12:0], Addr 94 (SDP_IO), *Address 0xF4[4:0]; Address 0xF5[7:0]*

CSC C3 coefficient for SDP output colour space converter

Function

SDP_C3[12:0]	Description
0x0681 «	Default value

SDP_C4[14:0], Addr 94 (SDP_IO), *Address 0xF6[6:0]; Address 0xF7[7:0]*

CSC C4 coefficient for SDP output colour space converter

Function

SDP_C4[14:0]	Description
0x0000 «	Default value

7.22.7.1 CSC Manual Programming

The equation performed by SDP CSC is as follows:

$$\begin{bmatrix} Out\_A \\ Out\_B \\ Out\_C \end{bmatrix} = \left[\frac{1}{4096} \cdot \begin{bmatrix} SDP\_A1[12:0] & SDP\_A2[12:0] & SDP\_A3[12:0] \\ SDP\_B1[12:0] & SDP\_B2[12:0] & SDP\_B3[12:0] \\ SDP\_C1[12:0] & SDP\_C2[12:0] & SDP\_C3[12:0] \end{bmatrix} \begin{bmatrix} In\_A \\ In\_B \\ In\_C \end{bmatrix} + \begin{bmatrix} A4[14:0] \\ B4[14:0] \\ C4[14:0] \end{bmatrix} \right] \cdot 2^{SDP\_CSC\_SCALE}$$

Where:

Out_A – Channel A output

Out_B – Channel B output

Out_C – Channel C Output

Equation 4: CSC Channels A, B, C

As can be seen from Equation 4 the SDP_A1[12:0], SDP_A2[12:0], SDP_A3[12:0]; SDP_B1[12:0], SDP_B2[12:0], SDP_B3[12:0]; and SDP_C1[12:0], SDP_C2[12:0], SDP_C3[12:0] coefficients are used to scale the primary inputs. The values of SDP_A4[14:0], SDP_B4[14:0], and SDP_C4[14:0]are added as offsets. The SDP_CSC_SCALE bit allows the user to implement conversion formulae in which the coefficients exceed the standard range of [-4095/4096 .. 4095/4096]. The overall range of the CSC is [0..1) for unipolar signals (for example, Y, R, G, and B) and [-0.5..+0.5) for bipolar signals (for example, Pr and Pb).

Note: The bipolar signals must be offset to mid range, for example, 2048.

To arrive at programming values from typical formulas, the following steps are performed:

1. Determine the dynamic range of the equation.
   The dynamic range of the CSC is [0 … 1) or [-0.5 … +0.5). Equations with a gain larger than 1 need to be scaled back. Errors in the gain can be compensated for in the gain stages of the follow on blocks.
   ➔ Scale the equations, if necessary.

2. Check the value of each coefficient.
   The coefficients can only be programmed in the range [-1 … +1). To support larger coefficients, the SDP_CSC_SCALE function should be used.

➔ Determine the setting for SDP_CSC_SCALE and adjust coefficients, if necessary.

3. Program the coefficient values.
   Convert the float point coefficients into 12-bit fixed decimal format. Convert into binary format, using twos complement for negative values.
   ➔ Program A1 .. A3, B1 .. B3, C1 .. C3.

4. Program the offset values.
   Depending on the type of CSC, offsets may have to be used.
   ➔ Program A4, B4, C4.

7.22.7.2 CSC Example

The following set of equations gives an example of a conversion from a gamma corrected RGB signal into a YCrCb color space signal.

$$\begin{bmatrix} Out\_A \\ Out\_B \\ Out\_C \end{bmatrix} = \left[\frac{1}{4096} \cdot \begin{bmatrix} SDP\_A1[12:0] & SDP\_A2[12:0] & SDP\_A3[12:0] \\ SDP\_B1[12:0] & SDP\_B2[12:0] & SDP\_B3[12:0] \\ SDP\_C1[12:0] & SDP\_C2[12:0] & SDP\_C3[12:0] \end{bmatrix} \begin{bmatrix} In\_A \\ In\_B \\ In\_C \end{bmatrix} + \begin{bmatrix} A4[14:0] \\ B4[14:0] \\ C4[14:0] \end{bmatrix} \right] \cdot 2^{SDP\_CSC\_SCALE}$$

Note: The original equations give offset values of 128 for the Pr and Pb components. The value of 128 equates to half the range on an 8-bit system. It must be noted that the CSC operates on a 12-bit range. The offsets, therefore, must be changed from 128 to half the range of a 12-bit system, which equates to 2048.

The maximum range for each channel, can only be [0 ... 1) or [-0.5 ... +0.5). Channels with a larger gain must be scaled back into range. The gain error can be compensated for in the gain stage of the follow on blocks.

The ranges of the three equations are:

Equation	Minimum Value	Maximum Value	Range
Y	0 + 0 + 0 = 0	0.59 + 0.3 + 0.11 = 1	[0 … 1] = 1
Pb	(-0.34) + (-0.17) = -0.51	0.51	[-0.51 … 0.51] = 1.02
Pr	(-0.43) + (-0.08) = -0.51	0.51	[-0.51 … 0.51] = 1.02

As can be seen from this table, the range for the Y component fits into the CSC operating range. However, the Pb and Pr ranges slightly exceed the range. To bring all equations back into the supported range, they should be scaled back by 1/1.02.

If equations fall outside the supported range, overflow or underflow can occur and undesirable wrap around effects (large number overflowing to small ones) can happen.

$$\begin{bmatrix} Y \\ Pb \\ \Pr \end{bmatrix} = \frac{1}{1.02} \begin{bmatrix} 0.59 & 0.3 & 0.11 \\ -0.34 & -0.17 & 0.51 \\ -0.43 & 0.51 & -0.08 \end{bmatrix} \begin{bmatrix} G \\ R \\ B \end{bmatrix} + \begin{bmatrix} 0 \\ 2048 \\ 2048 \end{bmatrix} = \begin{bmatrix} 0.58 & 0.29 & 0.11 \\ -0.33 & -0.17 & 0.5 \\ -0.42 & 0.5 & -0.08 \end{bmatrix} + \begin{bmatrix} 0 \\ 2048 \\ 2048 \end{bmatrix}$$

So:

SDP_A1 = 0.58 * 4096 = 2376

SDP_A1 = 0x0948

SDP_A2 = 0.29 * 4096 = 1188

SDP_A2 = 0x04A4

SDP_A3 = -0.42 * 4096 = -1720 = 2376+ (-4096)
SDP_A3 = 0x1948

SDP_B1 = -0.33 * 4096 = -1352 = 2744 + (-4096)
SDP_B1 = 0x1AB8
SDP_B2 = -0.17 * 4096 = -696 = 3400 + (-4096)
SDP_B2 = 0x1D48
SDP_B3 = 0.5 * 4096 = 2048
SDP_B3 = 0x800

SDP_C1 = -0.42 * 4096 = -1720 = 2376+ (-4096)
SDP_C1 = 0x1948
SDP_C2 = 0.5 * 4096 = 2048
SDP_C2 = 0x0800
SDP_C3 = -0.08 * 4096 = -327 = 3769 + (-4096)
SDP_C3 = 1EB9

Note: The scaling of the dynamic range does not affect the static offset.

Check the value of each coefficient:

The maximum value for each coefficient on its own can only be within the range of -4096/4096 to 4095/4096, which equals [-1 .. 0.999755859375). Values outside this range do not fit into the 12-bit fixed point format used to program the coefficients.

If the value of one or more coefficients after scaling of the overall equation exceeds the supported coefficient range, the SDP_CSC_SCALE bit should be set.

With the SDP_CSC_SCALE bit set high, all coefficients must be scaled by half, which makes them fit into the given coefficient range. The overall outputs of the CSC are gained up by a fixed value of two, thus compensating for the scaled down coefficients.

In the above example:

Each coefficient on its own is within the range of $\frac{-4096}{4096} \le Coeff \le \frac{4095}{4096}$

Therefore, all coefficients can be programmed directly and the SDP_CSC_SCALE bit should be set to 0.

Notes:

- To achieve a coefficient value of 1.0 for any given coefficient, the SDP_CSC_SCALE bit should be set high and the coefficient should actually be programmed to a value of 0.5. Otherwise, the largest value would be 4095/4096 = 0.9997, which is not exactly 1. While this value could be interpreted as a 1, it is recommended to use the value of 0.5 and the SDP_CSC_SCALE bit for maximum accuracy.

- For very large coefficient values, for example, 2.58, a combination of SDP_CSC_SCALE and equation scaling should be used. Set SDP_CSC_SCALE high (2.58/2 = 1.29) and scale the overall equation by slightly more than 1.28 (coefficient falls within the supported range of [-1 … +1)).

7.23 AUXILIARY EMBEDDED TIMING CONTROLS

The ADV7842 supports parallel output modes. The channel synchronization controls for the auxiliary pixel bus are described in the following sections. This auxiliary channel supports inputs up to 525i/625i. Only embedded timing is used on the auxiliary channel. These main timing control adjustments include:

- SDP_AUX_SAV_EN, SDP_AUX_EAV_EN
- SDP_AUX_SAV_POS_ADJ[11:0], SDP_AUX_EAV_POS_ADJ[11:0]
- SDP_AUX_V_BIT_POL, SDP_AUX_F_BIT_POL
- SDP_AUX_V_BEG_O_ADJ[5:0], SDP_AUX_V_END_O_ADJ[5:0]
- SDP_AUX_V_BEG_E_ADJ[5:0], SDP_AUX_V_END_E_ADJ[5:0]
- SDP_AUX_F_TOG_O_ADJ[5:0], SDP_AUX_F_TOG_E_ADJ[5:0]

Other auxiliary controls include:

- SDP_AUX_HBLANK_EN, SDP_ANC_AUX_EN
- SDP_ANC_MAIN_EN
- SDP_REPL_ANC_DATA
- SDP_AUX_REPL_AV_CODE
- SDP_SPLIT_ANC_DATA

SDP_AUX_SAV_EN, Addr 94 (SDP_IO), *Address 0xC9[2]*

A control to enable the insertion of SAV codes into the datastream on the auxiliary pixel bus in parallel modes. Note only embedded timing is supported on the auxiliary pixel bus.

Function

SDP_AUX_SAV_EN	Description
0	Don't insert SAV codes, only valid for SDP modes
1 «	Insert SAV codes

SDP_AUX_EAV_EN, Addr 94 (SDP_IO), *Address 0xC9[3]*

A control to enable the insertion of EAV codes into the datastream on the auxiliary pixel bus in parallel modes. Note only embedded timing is supported on the auxiliary pixel bus.

Function

SDP_AUX_EAV_EN	Description
0	Don't insert EAV codes, only valid for SDP modes
1 «	Insert EAV codes

SDP_AUX_SAV_POS_ADJ[11:0], Addr 94 (SDP_IO), *Address 0x8E[3:0]; Address 0x8F[7:0]*

A control to adjust the SAV position in the auxiliary pixel bus datastream. This a 2s complement control. To be used in parallel modes only.

Function

SDP_AUX_SAV_POS_ADJ[11:0]	Description
0x000 «	Default position

SDP_AUX_EAV_POS_ADJ[11:0], Addr 94 (SDP_IO), *Address 0x8C[3:0]; Address 0x8D[7:0]*

A control to adjust the EAV position in the auxiliary pixel bus datastream. This a 2s complement control. To be used in parallel modes only.

Function

SDP_AUX_EAV_POS_ADJ[11:0]	Description
0x000 «	Default position

SDP_AUX_V_BIT_POL, Addr 94 (SDP_IO), *Address 0xC8[6]*

A control to invert the V bit polarity inserted in the auxiliary pixel bus datastream. This should be used in parallel modes only.

Function

SDP_AUX_V_BIT_POL	Description
0	Inverted V bit polarity
1 «	Default V bit polarity

SDP_AUX_F_BIT_POL, Addr 94 (SDP_IO), *Address 0xC8[5]*

A control to invert the F bit polarity inserted in the auxiliary pixel bus datastream. This should be used in parallel modes only.

Function

SDP_AUX_F_BIT_POL	Description
0	Inverted F bit polarity
1 «	Default F bit polarity

SDP_AUX_V_BEG_O_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xC2[5:0]*

A control to adjust the 656 code V bit low to high transition on the odd field relative to default position in the auxiliary pixel bus datastream. This is a 2s complement control. Only positive adjustments recommended.

Function

SDP_AUX_V_BEG_O_ADJ[5:0]	Description
000100 «	Default value

SDP_AUX_V_END_O_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xC4[5:0]*

A control to adjust the 656 code V bit high to low transition on the odd field relative to default position in the auxiliary pixel bus datastream. This is a 2s complement control. Only positive adjustments are recommended.

Function

SDP_AUX_V_END_O_ADJ[5:0]	Description
000100 «	Default value

SDP_AUX_V_BEG_E_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xC3[5:0]*

A control to adjust the 656 code V bit low to high transition on the even field relative to default position in the auxiliary pixel bus datastream. This is a 2s complement control. Only positive adjustments recommended.

Function

SDP_AUX_V_BEG_E_ADJ[5:0]	Description
000100 «	Default value

SDP_AUX_V_END_E_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xC5[5:0]*

A control to adjust the 656 code V bit high to low transition on the even field relative to default position in the auxiliary pixel bus datastream. This is a 2s complement control. Only positive adjustments are recommended.

Function

SDP_AUX_V_END_E_ADJ[5:0]	Description
000100 «	Default value

SDP_AUX_F_TOG_O_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xC6[5:0]*

A control to adjust the 656 code F bit transition position on the odd field relative to default in the auxiliary pixel bus datastream. This is a 2s complement control. Only positive adjustments are recommended.

Function

SDP_AUX_F_TOG_O_ADJ[5:0]	Description
000100 «	Default value

SDP_AUX_F_TOG_E_ADJ[5:0], Addr 94 (SDP_IO), *Address 0xC7[5:0]*

A control to adjust the 656 code F bit transition position on the even field relative to default in the auxiliary pixel bus datastream. This is a 2s complement control. Only positive adjustments are recommended.

Function

SDP_AUX_F_TOG_E_ADJ[5:0]	Description
000100 «	Default value

SDP_ANC_AUX_EN, SDP_ANC_MAIN_EN

These bits controls whether or not the ancillary data, will be output on the main or on the auxiliary pixel bus data stream. Only the main or auxiliary pixel bus can have ancillary data enabled at any time. **SDP_ANC_AUX_EN**, Addr 94 (SDP_IO), *Address 0xC8[2]*

A control to enable the ancillary data on the auxiliary pixel bus datastream. Note only one pixelbus, main or auxiliary, should be enabled for ancillary data at a time.

Function

SDP_ANC_AUX_EN	Description
0 «	Ancillary data if enabled does not come on aux channel
1	Ancillary data if enabled comes on aux channel

SDP_ANC_MAIN_EN, Addr 94 (SDP_IO), *Address 0xC8[3]*

A control to enable the ancillary data on the main pixel bus datastream. Note only one pixelbus, main or auxiliary, should be enabled for ancillary data at a time.

Function

SDP_ANC_MAIN_EN	Description
0	Ancillary data if enabled does not come on main channel
1 «	Ancillary data if enabled comes on main channel

SDP_REPL_ANC_DATA, Addr 94 (SDP_IO), *Address 0xB3[7]*

A control to enable replication ancillary data on all channels

Function

SDP_REPL_ANC_DATA	Description
0 «	Ancillary data on Y/G channel only (if CLK is fast enough)
1	Replicate SDP ancillary data on all channels

SDP_AUX_REPL_AV_CODE, Addr 94 (SDP_IO), *Address 0xC9[4]*

A control to select that embedded timing is replicated on the chroma channel on the auxiliary pixel bus.

Function

SDP_AUX_REPL_AV_CODE	Description
0 «	Output single SAV/EAV codes on auxiliary luma channel only.
1	Replicate SAV/EAV codes on auxiliary Cr/Cb channel.

SDP_SPLIT_ANC_DATA, Addr 94 (SDP_IO), *Address 0xB3[6]*

A control to enable splitting ancillary data across channels.

Function

SDP_SPLIT_ANC_DATA	Description
0 «	Don't split SDP ancillary data across channels. Overwritten by SDP_REPL_ANC_DATA
1	Split SDP ancillary data across channels

8 HDMI RECEIVER

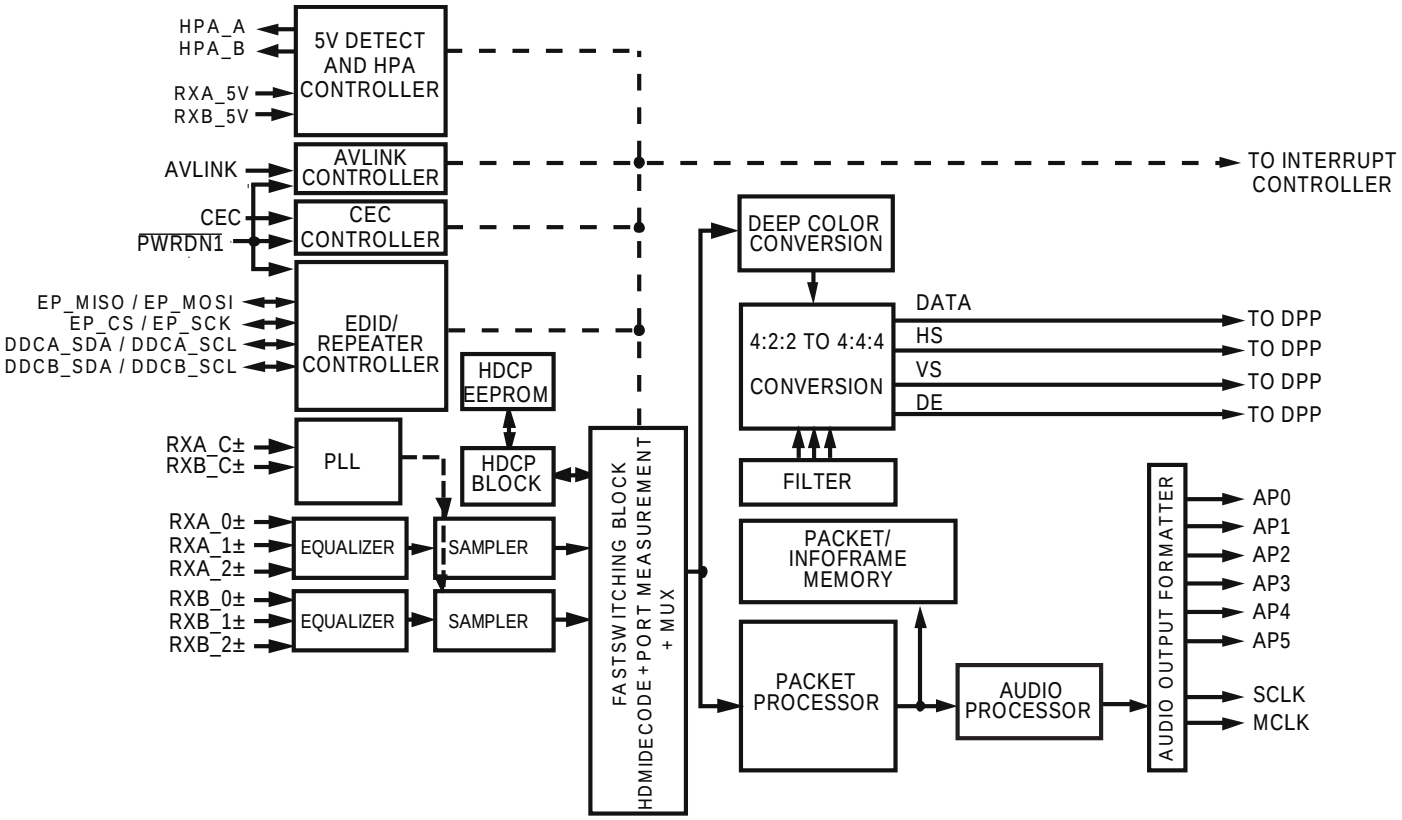


Figure 64: Functional Block Diagram of HDMI Core

8.1 +5 V CABLE DETECT

The HDMI receiver in the ADV7842 can monitor the level on the +5 V power signal pin of each connected HDMI port. The results of this detection can be read back from the following I^2C registers. These readbacks are valid even when the part is not configured for HDMI mode.

Note: If not using the RXx_5V pins (where x =A or B) these pins should be pulled to 5 V.

CABLE_DET_A_RAW, Addr 40 (IO), *Address 0x6F[1] (Read Only)*

Raw status of Port A +5 V cable detection signal.

Function

CABLE_DET_A_RAW	Description
0 «	No cable detected on Port A
1	Cable detected on Port A

CABLE_DET_B_RAW, Addr 40 (IO), *Address 0x6F[0] (Read Only)*

Raw status of Port B +5 V cable detection signal.

Function

CABLE_DET_B_RAW	Description
0 «	No cable detected on Port B
1	Cable detected on Port B

The ADV7842 provides a digital glitch filter on the +5 V power signals from the HDMI ports. The output of this filter is used to reset the HDMI block (refer to Section 8.40).

The +5 V power signal must be constantly high for the duration of the timer (controlled by FILT_5V_DET_TIMER[6:0]) otherwise the output of the filter is low. The output of the filter returns low as soon as any change in the +5 V power signal is detected.

FILT_5V_DET_DIS, Addr 68 (HDMI), *Address 0x56[7]*

This bit is a control to disable the digital glitch filter on the HDMI 5V detect signals. The filtered signals are used as interrupt flags, and also used to reset the HDMI section. The filter works from an internal ring oscillator clock and is therefore available in power-down mode. The clock frequency of the ring oscillator is 42MHz +/-10%.

Function

FILT_5V_DET_DIS	Description
0 «	Enabled
1	Disabled

FILT_5V_DET_TIMER[6:0], Addr 68 (HDMI), *Address 0x56[6:0]*

These bits control the timer for the digital glitch filter on the HDMI +5 V detect inputs. The unit of this parameter is 2 clock cycles of the ring oscillator (~ 47ns). The input must be constantly high for the duration of the timer, otherwise the filter output remains low. The output of the filter returns low as soon as any change in the +5 V power signal is detected.

Function

FILT_5V_DET_TIMER[6:0]	Description
1011000 «	Approximately 4.2us
xxxxxxx	Time duration of +5 V deglitch filter. The unit of this parameter is 2 clock cycles of the ring oscillator (~ 47ns)

DIS_CABLE_DET_RST, Addr 68 (HDMI), *Address 0x48[6]*

This control disables the reset effect of cable detection.

Function

DIS_CABLE_DET_RST	Description
0 «	Resets the HDMI section if the 5 V input pin corresponding to the selected HDMI port (e.g. RXA_5V for port A) is inactive
1	Do not use the 5 V input pins as reset signal for the HDMI section

8.2 HOT PLUG ASSERT

The ADV7842 features hot plug assert (HPA) controls for its two HDMI ports. The purpose of these controls and their corresponding output pins is to communicate to an HDMI Transmitter that the HDMI Receivers' E-EDID connected to the DDC bus can be accessed by

the source.

Note: In order to comply with the required output characteristics of section 4.2.9 "Hot Plug Detect Signal" of the HDMI 1.4 specification the output resistance on the Hot Plug Assert pins must be 1000 Ohm +/-20%. This may easily be implemented by connecting the Hot Plug Assert line to the corresponding +5 V power signal through a 1000 Ohm resistor.

HPA_MANUAL, Addr 68 (HDMI), *Address 0x69[0]*

Manual control enable for the Hot Plug Assert output pins. By setting this bit any automatic control of these pins is disabled. Manual control is determined by the HPA_MAN_VALUE_X (where X = A, B)

Function

HPA_MANUAL	Description
0 «	HPA takes its value based on HPA_AUTO_INT_EDID
1	HPA takes its value from HPA_MAN_VALUE_X

HPA_MAN_VALUE_A, Addr 40 (IO), *Address 0x20[5]*

A manual control for the value of HPA on Port A. Only valid if HPA_MANUAL is set to 1.

Function

HPA_MAN_VALUE_A	Description
0	0 V applied to HPA_A pin.
1 «	High level applied to HPA_A pin.

HPA_MAN_VALUE_B, Addr 40 (IO), *Address 0x20[4]*

A manual control for the value of HPA on Port B. Only valid if HPA_MANUAL is set to 1.

Function

HPA_MAN_VALUE_B	Description
0	0 V applied to HPA_B pin.
1 «	High level applied to HPA_B pin.

Note: The HPA_X pins (where X = A or B) are open drain. An external pull-up resistor is required to pull them high.

HPA_AUTO_INT_EDID[1:0], Addr 68 (HDMI), *Address 0x69[2:1]*

Selects the type of automatic control on the HPA output pins. This bit has no effect when HPA_MANUAL is set to 1.

Function

HPA_AUTO_INT_EDID[1:0]	Description
00	The HPA of an HDMI port is asserted high immediately after the internal EDID has been activated for that port. The HPA of a specific HDMI port is de-asserted low immediately after the internal E-EDID is de-activated for that port.
01 «	The HPA of an HDMI port is asserted high following a programmable delay after the part detects an HDMI cable plug on that port. The HPA of an HDMI port is immediately de-asserted after the part detects a cable disconnect on that HDMI port.
10	The HPA of an HDMI port is asserted high after two conditions have been met. The conditions are detailed as follows. 1. The internal EDID is active for that port. 2. The delayed version of the cable detect signal CABLE_DET_X_RAW for that port is high. The HPA of an HDMI port is immediately de-asserted after any of the following two conditions have been met 1. The internal EDID is de-activated for that port 2. The cable detect signal CABLE_DET_X_RAW for that port is low.
11	The HPA of an HDMI port is asserted high after three conditions have been met. The conditions are detailed as follows. 1. The internal EDID is active for that port. 2. The delayed version of the cable detect signal CABLE_DET_X_RAW for that port is high. 3. The user has set the manual HPA control for that port to 1 via the HPA_MAN_VALUE_X controls. The HPA of an HDMI port is immediately de-asserted after any of the following three conditions have been met 1.The internal EDID is de-activated for that port 2.The cable detect signal CABLE_DET_X_RAW for that port, is low. 3.The user sets the manual HPD control for that port to 0 via the HPA_MAN_VALUE_X controls

Note: The delay is programmable. Refer to EDID_ENABLE for details on enabling the internal E-EDID for an HDMI port. In HPA_MAN_VALUE_X and CABLE_DET_X_RAW. 'X' refers to A and B.

HPA_TRISTATE_A, Addr 40 (IO), *Address 0x20[1]*

Tristate HPA output pin for Port A.

Function

HPA_TRISTATE_A	Description
0 «	HPA_A pin active.
1	Tristate HPA_A pin

HPA_TRISTATE_B, Addr 40 (IO), *Address 0x20[0]*

Tristate HPA output pin for Port B.

Function

HPA_TRISTATE_B	Description
0 «	HPA_B pin active
1	Tristate HPA_B pin.

HPA_STATUS_PORT_A, Addr 40 (IO), *Address 0x21[1] (Read Only)*

Readback of HPA status for Port A

Function

HPA_STATUS_PORT_A	Description
0 «	+5V not applied to HPA_A pin by chip.
1	+5V applied to HPA_A pin by chip.

HPA_STATUS_PORT_B, Addr 40 (IO), *Address 0x21[0] (Read Only)*

Readback of HPA status for Port B

Function

HPA_STATUS_PORT_B	Description
0 «	+5V not applied to HPA_B pin by chip
1	+5V applied to HPA_B pin by chip

HPA_OVR_TERM, Addr 68 (HDMI), *Address 0x69[3]*

A control to set termination control to be overridden by the HPA setting. When this bit is set, termination on a specific port will be set according to the HPA status of that port.

Function

HPA_OVR_TERM	Description
0 «	Automatic or manual I2C control of port termination.
1	Termination controls disabled and overridden by HPA controls.

8.3 E-EDID/REPEATER CONTROLLER

The HDMI section incorporates an E-EDID/Repeater controller, which performs the following tasks:

Computes the E-EDID checksums for the two ports
Updates the SPA value after the E-EDID image has been loaded from the SPI EEPROM into the internal E-EDID RAM
Performs the repeater routines described in Section 8.35

The E-EDID/Repeater controller is powered from the VDD supply and clocked by an internal ring oscillator. The controller and the internal DDC bus arbiter are kept active in power-down mode 0 and power-down mode 1, which allows the internal E-EDID to be functional and accessible through the DDC port even when the part is powered down (refer to Section 3.2.3). In these power-down modes, all the power needed by the ADV7842 can be provided by one or more HDMI transmitters connected to the HDMI ports. These HDMI transmitters can then read the capabilities of the powered down application integrating the ADV7842 by accessing its internal E-EDID through the DDC ports.

The E-EDID/Repeater controller is reset when the VDD supplies go low or when HDCP_REPT_EDID_RESET is set high. When the E-EDID/Repeater controller reboots, it performs the following tasks:

- Clears the internal E-EDID and KSV RAM (refer to Section 8.4 and Section 8.21.2)
- Computes a total of seven checksums for all two ports (refer to Section 8.10)
- Updates the SPA registers (refer to Section 8.11)

HDCP_REPT_EDID_RESET, Addr 68 (HDMI), *Address 0x5A[3] (Self-Clearing)*

A reset control for the E-EDID/Repeater controller. When asserted it resets the E-EDID/Repeater controller.

Function

HDCP_REPT_EDID_RESET	Description
0 «	Normal operation
1	Resets the E-EDID/Repeater controller.

8.4 E-EDID DATA CONFIGURATION

The ADV7842 features an SRAM memory that can store an Enhanced-Extended Display Identification (E-EDID). This internal E-EDID feature can be used for the two HDMI ports A, B, C, and D. It is also possible to use an external device storage for the E-EDID data on each port, or a combination of internal E-EDID for some port(s) and external storage for the other port(s).

The following controls are provided to enable the internal E-EDID for each of the two HDMI ports.

EDID_A_ENABLE, Addr 64 (Repeater), *Address 0x77[2]*

Enables I2C access to internal EDID RAM from DDC Port A

Function

EDID_A_ENABLE	Description
0 «	E-EDID for Port A disabled
1	E-EDID for Port A enabled

EDID_B_ENABLE, Addr 64 (Repeater), *Address 0x77[3]*

Enables I2C access to internal EDID RAM from DDC Port B

Function

EDID_B_ENABLE	Description
0 «	E-EDID for Port B disabled
1	E-EDID for Port B enabled

When the internal E-EDID is enabled on any of the two ports (e.g. port A by setting EDID_A_ENABLE to 1), the ADV7842 must first calculate the E-EDID checksums for that port before the E-EDID is actually enabled.

The following read only flags can be utilized to determine if the E-EDID is actually enabled on any of the two HDMI ports.

EDID_A_ENABLE_CPU, Addr 64 (Repeater), *Address 0x7D[2] (Read Only)*

Flags internal EDID enabling on Port A

Function

EDID_A_ENABLE_CPU	Description
0 «	Disabled
1	Enabled

EDID_B_ENABLE_CPU, Addr 64 (Repeater), *Address 0x7D[3] (Read Only)*

Flags internal EDID enabling on Port B

Function

EDID_B_ENABLE_CPU	Description
0 «	Disabled
1	Enabled

Notes:

- When the internal E-EDID is enabled on more than one port (e.g. ports A and B), the corresponding enable controls (e.g. EDID_A_ENABLE and EDID_B_ENABLE) should be set high in one single I²C write. This ensures the fastest calculation of the checksums.

- If the internal E-EDID RAM is enabled for one specific port (e.g. port A), an external E-EDID storage device should not be connected on the DDC bus of that port.
- The internal E-EDID can be read by Current Address Read sequences on the DDC ports.
- The ADV7842 supports the segment pointer, which is set at device address 0x60 through the DDC bus, and used in combination with the internal E-EDID address (0xA0) to access the internal E-EDID.

8.4.1 E-EDID Support for Power-down Modes

The ADV7842 supports internal E-EDID access in power-down mode 0 and power-down mode 1. Using this feature, an application that integrates the ADV7842 in standby can make its E-EDID available to the HDMI transmitter. This allows support of CEC and provides compatibility with HDMI transmitters that require the E-EDID to be available when the HDMI receiver is powered down.

In power-down mode 0, the part operates in a very low power state with only the minimum of internal circuitry enabled for the internal E-EDID. This allows the E-EDID/Repeater controller to load the E-EDID image from an external SPI EEPROM into the internal E-EDID RAM. The E-EDID/Repeater controller also updates the SPA of each port (refer to Section 8.11), computes the required E-EDID checksums, and enables the internal E-EDID.

For more details on E-EDID accessibility in power down modes, refer to Section 3.2.3.

8.5 +5 V SUPPLY

The ADV7842 can receive power from the +5 V power signal line of one of the connected HDMI cable(s) and enter into power-down mode 0 (refer to Section 3.2.3). This feature requires specific ADV7842 supplies to be capable of taking their input from either the generic power supply or the +5 V power signal line of the two input HDMI ports.

Special care should be taken to ensure the generic power supply does not supply power back onto the +5 V power signal lines in normal operation and that power is only used from the +5 V power signal lines when the generic power supply is not available. Refer to Figure 4.

8.6 POWER-DOWN PIN

The ADV7842 features a hardware pin control that can be used to configure the ADV7842 into power-down mode 0. This hardware functionality allows the internal E-EDID to be available even if the main supply (e.g. AC power) is not available to power up the ADV7842 (refer to Section 3.2.3).

When power-down mode 0 is initiated, the internal E-EDID is automatically configured and the part loads its internal E-EDID with the information in the SPI EEPROM and internal E-EDID is enabled on all ports.

Note: The $\overline{\text{PWRDN1}}$ pin must not be pulled high when the part is powered from the cable supply.

8.7 SPI INTERFACE

The ADV7842 has a 4-pin SPI interface to load the E-EDID information from the SPI EEPROM into the internal E-EDID RAM:

- EP_MOSI
- EP_CS

- EP_MISO
- EP_SCK

The SPI interface offers the user controls to tristate the SPI pins, load the E-EDID data image from the SPI EEPROM into the internal E-EDID RAM or store the E-EDID data image from the internal E-EDID RAM into the SPI EEPROM.

EXT_EEPROM_TRI, Addr 64 (Repeater), *Address 0x78[6]*

Tri-states the output pins to the external SPI EEPROM

Function

EXT_EEPROM_TRI	Description
0 «	SPI interface outputs enabled
1	SPI interface outputs tri-stated

Note: VGA_EDID_ENABLE must be left at its default value of 0 in order to use the LOAD_EDID and STORE_EDID functions.

LOAD_EDID, Addr 64 (Repeater), *Address 0x7E[1] (Self-Clearing)*

Force loading internal E-EDID RAM with SPI EEPROM contents. This self clearing bit returns to 0 after successfully loading the internal E-EDID RAM with the SPI EEPROM contents.

Function

LOAD_EDID	Description
0 «	No effect
1	Load internal E-EDID RAM with SPI EEPROM contents

STORE_EDID, Addr 64 (Repeater), *Address 0x7E[0] (Self-Clearing)*

Write internal E-EDID RAM contents to SPI EEPROM. This self clearing bit returns to 0 after successfully writing the contents of the internal E-EDID RAM to SPI EEPROM.

Function

STORE_EDID	Description
0 «	No effect
1	Write contents of internal E-EDID RAM to SPI EEPROM

8.7.1 SPI EEPROM Data Structure

The ADV7842 requires data in the SPI EEPROM to be stored as shown in Figure 65.

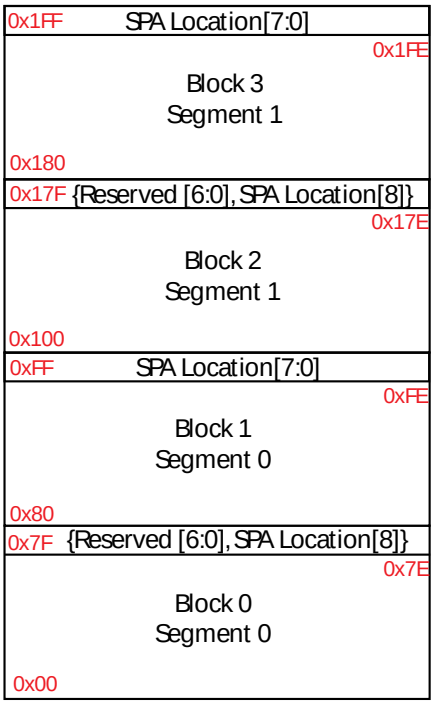


Figure 65: SPI EEPROM Data Image Structure

The SPA location is stored in lieu of the checksums – the part recalculates the checksums once the E-EDID data has been read.

Notes:

- 4 Kb SPI EEPROM must be used to store a 3 to 4 block E-EDID image
- 2 Kb or 4 Kb SPI EEPROM can be used to store a 2 block E-EDID image
- Although the SPA location is duplicated in both segments of the SPI EEPROM structure, the E-EDID controller uses only the SPA location that is in the first segment of the SPI EEPROM.

8.8 TRANSITIONING FROM POWER MODE

If the part starts in power-down mode 0 and then transitions into a different power mode (i.e. power-down mode 1 or normal operation mode), the information in the internal E-EDID is not overwritten. The internal E-EDID remains active on the HDMI port(s) for which the E-EDID has been accessed. This prevents disturbing E-EDID read requests from HDMI sources connected to the ADV7842.

It is possible to disable the automatic enable of internal E-EDID on the HDMI ports when the part comes out of powerdown mode, by setting the DISABLE_AUTO_EDID bit.

DISABLE_AUTO_EDID, Addr 64 (Repeater), *Address 0x77[5]*

Disables all automatic enables for internal E-EDID

Function

DISABLE_AUTO_EDID	Description
0 «	Automatic enable of internal E-EDID on HDMI ports when the part comes out of powerdown mode
1	Disable automatic enable of internal E-EDID on HDMI ports when the part comes out of powerdown mode

8.9 STRUCTURE OF INTERNAL E-EDID FOR PORT A

The internal E-EDID is enabled on port A by setting EDID_A_ENABLE to 1. The structure of the internal E-EDID that is accessible on the DDC line of port A is shown in Figure 66.

The image of the internal E-EDID that is accessed on the DDC bus of port A corresponds to the data image contained in the internal E-EDID RAM.

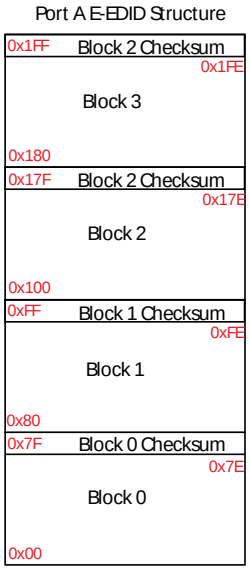


Figure 66: Port A E-EDID Structure and Mapping

Notes:

- After EDID_A_ENABLE is set to 1, the ADV7842 E-EDID/Repeater controller calculates the two checksums of the E-EDID image for port A and updates the internal RAM address locations 0x7F, 0xFF, 0x17F, and 0x1FF in the internal E-EDID RAM with the computed checksums.
- After power up, the ADV7842 E-EDID/Repeater controller sets all bytes in the internal E-EDID RAM to 0, this operation takes less than 1 ms. It is recommended to wait for at least 1 ms before initializing the EDID Map with an E-EDID image.
- When internal E-EDID is enabled on port A, the Hot Plug should not be asserted until the EDID Map has been completely initialized with E-EDID.
- The internal E-EDID can be accessed in read-only mode through the DDC interface at the I^2C address 0xA0.
- The internal E-EDID can be accessed in read/write mode through the general I^2C interface at the EDID Map I^2C address.

8.10 STRUCTURE OF INTERNAL E-EDID OF PORT B

This section describes the structure of the internal E-EDID accessible through the DDC bus of port B.

The internal E-EDID is enabled for port B by setting the EDID_B_ENABLE bit to 1. The image of the internal E-EDID that is accessed on the DDC bus of port B corresponds to the data image contained in the internal E-EDID RAM except for the SPA, SPA location, and the checksum of the E-EDID block where the SPA is located.

The structure of the internal E-EDID image for port B is shown in the following figures:

- Figure 67 – SPA located in E-EDID block 1
- Figure 68 – SPA located in E-EDID block 2
- Figure 69 – SPA located in E-EDID block 3

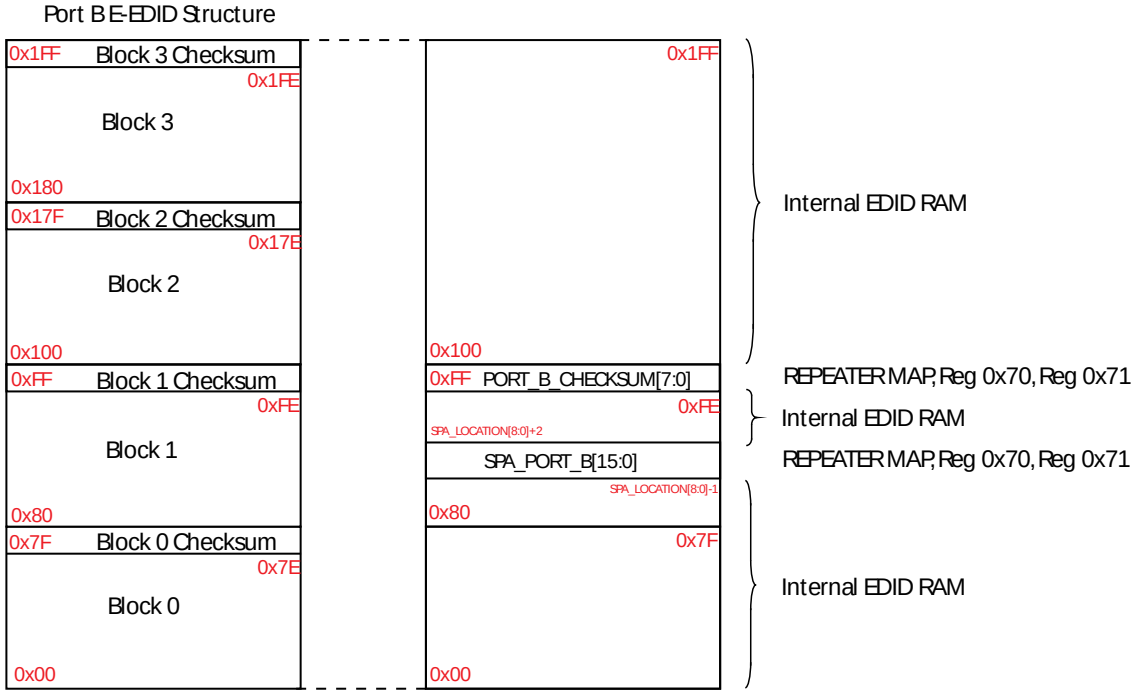


Figure 67: Port B E-EDID Structure and Mapping for SPA Located in E-EDID Block 1

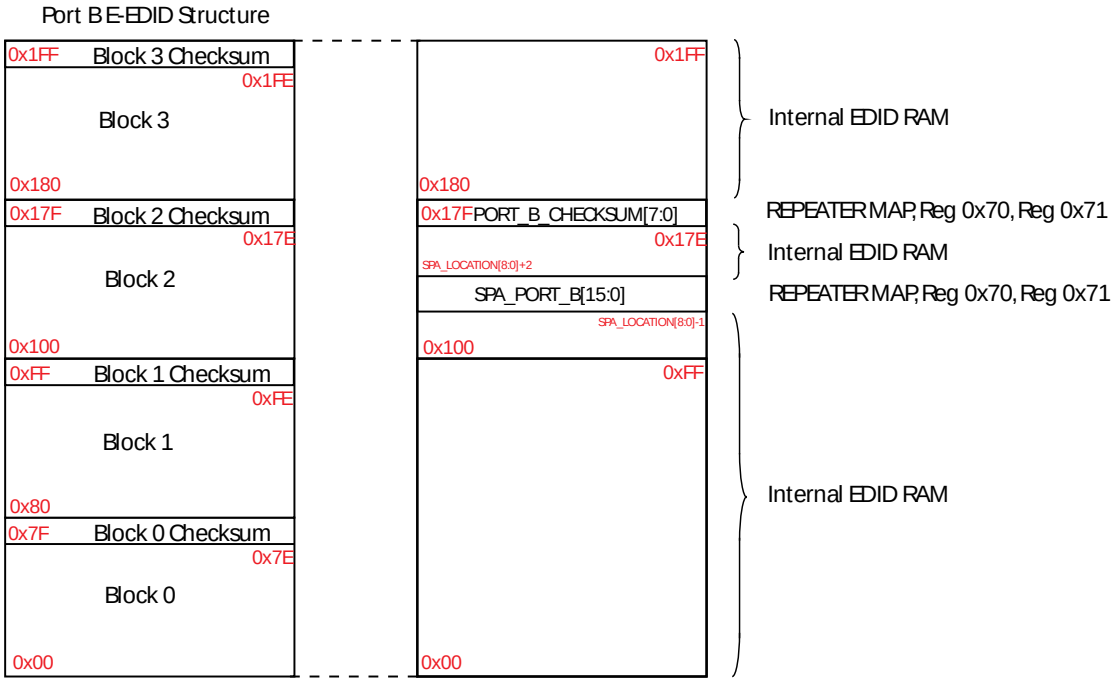


Figure 68: Port B E-EDID Structure and Mapping for SPA Located in E-EDID Block 2

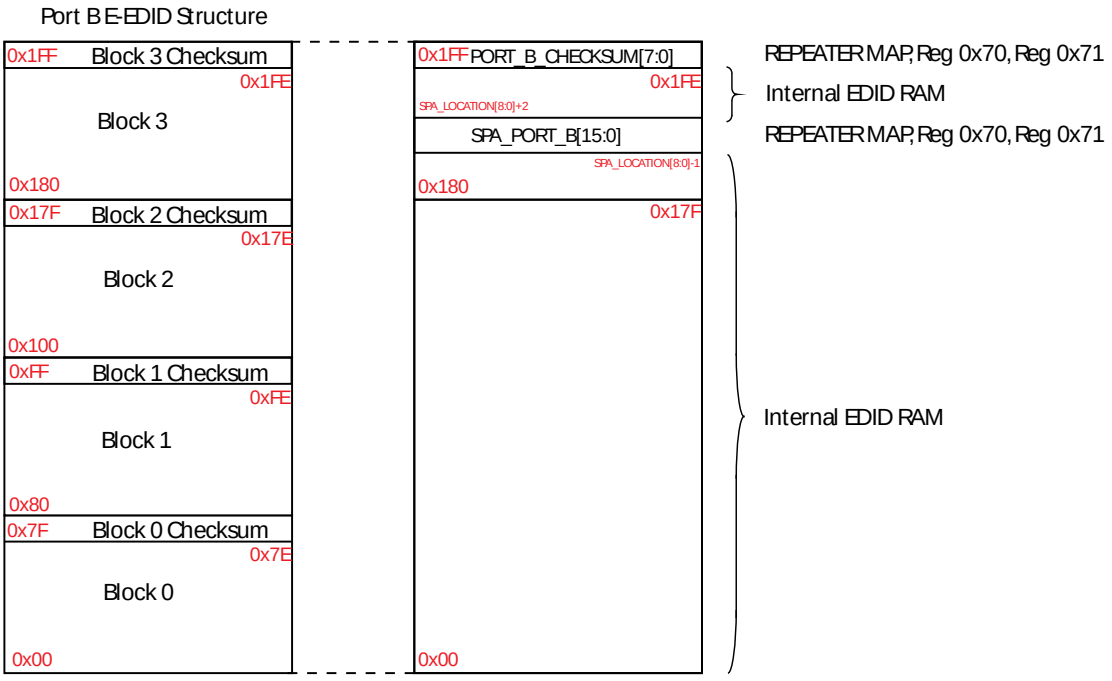


Figure 69: Port B E-EDID Structure and Mapping for SPA Located in E-EDID Block 3

The SPA of port B is programmed in the SPA_PORT_B[15:0] register. The SPA location is programmed in the SPA_LOCATION[7:0] register. This register should contain a value greater than 0x7F since the SPA is located in an upper block of the E-EDID.

Notes:

- When internal E-EDID is required for port B, the SPA along with its location address in the E-EDID must be programmed in the Repeater Map, registers SPA_PORT_B[15:0]and SPA_LOCATION[7:0] respectively.
- After EDID_B_ENABLE is set to 1, the ADV7842 E-EDID/Repeater controller computes the two checksums of the E-EDID

image for port B. The E-EDID controller then updates the checksum registers in the E-EDID RAM memory location 0x7F and the following three locations:

- 0x17F, 0x1FF, and the register SPA_LOCATION[7:0], which is the SPA located in the E-EDID block 1
- 0xFF, 0x1FF, and the register SPA_LOCATION[7:0], which is the SPA located in the E-EDID block 2
- 0xFE, 0x17F, and the register SPA_LOCATION[7:0], which is the SPA located in the E-EDID block 3

- After power up, the ADV7842 E-EDID controller sets all bytes in the internal E-EDID RAM to 0, this operation takes less than 1 ms. It is recommended to wait for at least 1 ms before initializing the EDID Map with E-EDID.
- SPA_LOCATION[7:0] must be programmed with a value greater than 0x7F, as SPA is always located in the E-EDID blocks 1, 2 or 3.
- When internal E-EDID is enabled on port B, the Hot Plug should not be asserted until the EDID Map has been completely initialized with E-EDID.
- The internal E-EDID can be accessed in read-only mode through the DDC interface at the I²C address 0xA0.
- The internal E-EDID can be accessed in read/write mode through the general I²C interface at the EDID Map I²C address.
- The SPA_PORT_B[15:0] register does not have to be programmed with an actual SPA value. It can be programmed with any value that must be read from the location SPA_LOCATION[7:0] when the internal E-EDID is accessed from the DDC lines of port B. This allows support for non CEA-861 compliant E-EDIDs (e.g. VESA-only compliant E-EDID for analog inputs).

The SPA of port B is the address of the port B in the CEC interface. The SPA is comprised of two components, A, B, C, and D as defined in the HDMI specification, which are programmed as follows:

- SPA_PORT_B[15:12] = A
- SPA_PORT_B[11:8] = B
- SPA_PORT_B[7:4] = C
- SPA_PORT_B[3:0] = D

SPA_PORT_B[15:0], Addr 64 (Repeater), *Address 0x74[7:0]; Address 0x75[7:0]*

Source Physical Address for Port B. This is used for CEC and is located in the HDMI Vendor Specific data block in the E-EDID.

Function

SPA_PORT_B[15:0]	Description
0000000000000000 «	Default value
xxxxxxxxxxxxxxxx	Source physical address of Port B

SPA_LOCATION[7:0], Addr 64 (Repeater), *Address 0x76[7:0]*

This is the location in the E-EDID data where the SPA is located.

Function

SPA_LOCATION[7:0]	Description
11000000 «	Default value
xxxxxxxx	Location of source physical address in internal E-EDID for ports A and B

PORT_B_CHECKSUM[7:0], Addr 64 (Repeater), *Address 0x7C[7:0]*

This is the checksum for the second half of the Port B EDID. This is calculated automatically.

Function

PORT_B_CHECKSUM[7:0]	Description
00000000 «	Default value
xxxxxxxx	Checksum for E-EDID block containing SPA for Port B

8.11 SPA CONFIGURATION

When the E-EDID/Repeater controller configures the internal E-EDID in power-down mode 0 or power-down mode 1, it also updates the SPA registers for each port according to the SPA read from the external SPI EEPROM. The 2-byte SPA is located at the address specified by SPA_LOCATION in addresses 0x7F and 0xFF of the SPI EEPROM. The SPA of each port is set as follows:

- SPA for port A located in E-EDID RAM is set to A.B.C.D
- SPA for port B, SPA_PORT_B[15:0] is set to A+1.B.C.D

where A.B.C.D is the 2-byte SPA read from the SPI EEPROM. The format A.B.C.D is described in the HDMI specification.

8.12 EXTERNAL E-EDID

It is possible to use an external device such as an EEPROM to store E-EDID data. When an external storage device is used for the E-EDID data of a specific HDMI port, the storage device must be connected to the DDC lines of that HDMI port. The internal E-EDID should not be enabled for that specific port.

8.13 TMDS EQUALIZATION

The ADV7842 incorporates active equalization of the HDMI data signals. This equalization compensates for the high frequency losses inherent in HDMI and DVI cabling, especially at long lengths and higher frequencies. The ADV7842 is capable of equalizing for cable lengths up to 30 meters and for pixel clock frequencies up to 225 MHz. The equalization is adaptive but also has a semi automatic mode of operation.

Note: The Transition Minimized Differential Signaling (TMDS) Equalization frequency of the active HDMI port can be read back in the TMDSFREQ[8:0] and TMDSFREQ_FRAC[6:0] registers.

EQ_DYN_EN, Addr 68 (HDMI), *Address 0x96[0]*

Enable for HDMI Equalizer Dynamic Control

Function

EQ_DYN_EN	Description
0 «	Disables equalizer dynamic mode. The equalizer is configured in static mode.
1	Enables equalizer dynamic mode. Equaliser is configured via EQ_DYNx_HF and EQ_DYNx_LF settings.

The ADV7842 features a dynamic equalizer mode. In this mode, the ADV7842 allows the programming of equalizer settings for three frequency bandwidth ranges. EQ_DYN_FREQ1[3:0] and EQ_DYN_FREQ2[3:0] set the frequency limits for these ranges. The frequency limits set by these registers are expressed in MHz divided by 16.

The internally measured TMDS frequency is compared against these limits. The ADV7842 then applies the corresponding equalizer settings according to the detected range. The specific settings are placed in EQ_DYNX_LF and EQ_DYNX_HF (where X = 1, 2, or 3).

EQ_DYN_FREQ1[3:0], Addr 68 (HDMI), *Address 0x8C[3:0]*

A control to set the lower limit, limit 1, for the HDMI equalizer dynamic control frequency range. The frequency must be specified in MHz divided by 16.

Function

EQ_DYN_FREQ1[3:0]	Description
0000	Reserved. Do not use.
0011 «	Default dynamic equalizer frequency limit 1. The default value corresponds to 48 MHz.
xxxx	Frequency for limit 1

EQ_DYN_FREQ2[3:0], Addr 68 (HDMI), *Address 0x8C[7:4]*

A control to set the upper limit, limit 2, for the HDMI Equalizer Dynamic Control Frequency range. The frequency must be specified in MHz divided by 16.

Function

EQ_DYN_FREQ2[3:0]	Description
0000	Reserved. Do not use.
1010 «	Default dynamic equalizer frequency limit 2. The default value corresponds to 160 MHz.
xxxx	Frequency for limit 2.

The internally measured TMDS frequency value is compared to the limits controlled by EQ_DYN_FREQ1[3:0] and EQ_DYN_FREQ2[3:0]. If the measured frequency is outside a range by a given tolerance, the equalizer setting is updated as appropriate for the frequency range. This tolerance is set via the control register FREQTOLERANCE[3:0]. This form of hysteresis allows for oscillations in the TMDSFREQ.

The ADV7842 equalizer block consist of two different gain boost response blocks, as shown in Figure 70 and Figure 71.

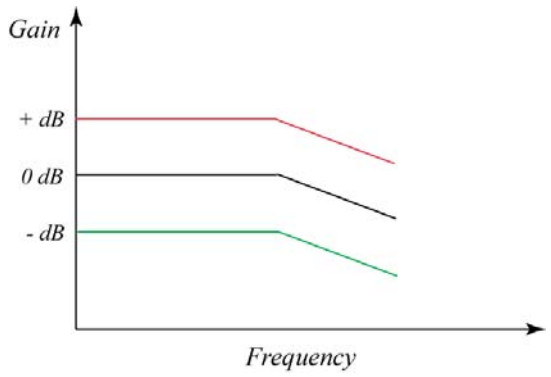


Figure 70: Low Frequency Gain Response

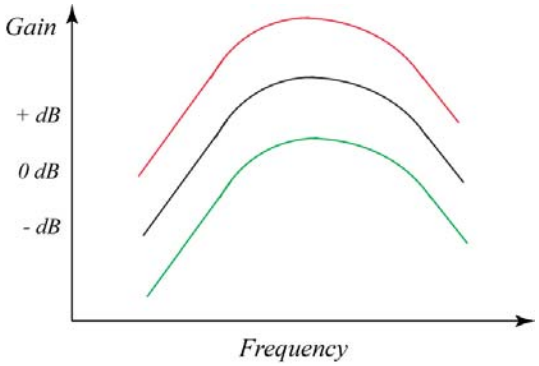


Figure 71: High Frequency Gain Response

These two gain boost block outputs are combined together and perform an overall low frequency/high frequency compensated equalizing processing, as shown in Figure 72.

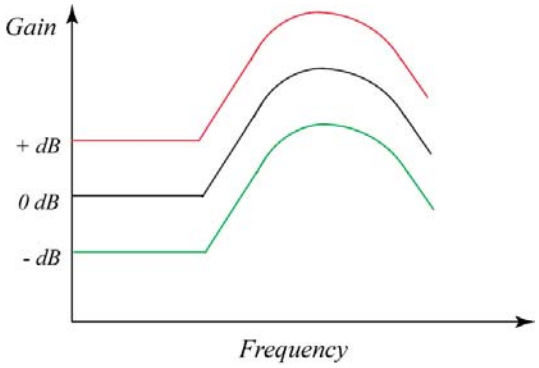


Figure 72: Overall Gain of Equalizer Process

EQ_DYN1_LF[7:0], Addr 68 (HDMI), *Address 0x8D[7:0]*

HDMI Equalizer Dynamic Control LF for frequencies below limit1, i.e. range1

Function

EQ_DYN1_LF[7:0]	Description
00001011 «	Default LF gain equalizer settings for dynamic mode range 1
xxxxxxxx	LF gain equalizer settings for dynamic mode range 1

EQ_DYN1_HF[7:0], Addr 68 (HDMI), *Address 0x8E[7:0]*

HDMI Equalizer Dynamic Control HF for frequencies below limit1, i.e. range1

Function

EQ_DYN1_HF[7:0]	Description
00100000 «	Default HF gain equalizer settings for dynamic mode range 1
xxxxxxxx	HF gain equalizer settings for dynamic mode range 1

EQ_DYN2_LF[7:0], Addr 68 (HDMI), *Address 0x90[7:0]*

HDMI Equalizer Dynamic Control LF for frequencies below limit2 and above limit1, i.e. range2

Function

EQ_DYN2_LF[7:0]	Description
00001011 «	Default LF gain equalizer settings for dynamic mode range 2
xxxxxxxx	LF gain equalizer settings for dynamic mode range 2

EQ_DYN2_HF[7:0], Addr 68 (HDMI), *Address 0x91[7:0]*

HDMI Equalizer Dynamic Control HF for frequencies below limit2 and above limit1, i.e. range2

Function

EQ_DYN2_HF[7:0]	Description
00100000 «	Default HF gain equalizer settings for dynamic mode range 2
xxxxxxxx	HF gain equalizer settings for dynamic mode range 2

EQ_DYN3_LF[7:0], Addr 68 (HDMI), *Address 0x93[7:0]*

HDMI Equalizer Dynamic Control LF for frequencies above limit2, i.e. range3

Function

EQ_DYN3_LF[7:0]	Description
00001011 «	Default LF gain equalizer settings for dynamic mode range 3
xxxxxxxx	LF gain equalizer settings for dynamic mode range 3

EQ_DYN3_HF[7:0], Addr 68 (HDMI), *Address 0x94[7:0]*

HDMI Equalizer Dynamic Control HF for frequencies above limit2, i.e. range3

Function

EQ_DYN3_HF[7:0]	Description
00100000 «	Default HF gain equalizer settings for dynamic mode range 3
xxxxxxxx	HF gain equalizer settings for dynamic mode range 3

8.14 PORT SELECTION

HDMI_PORT_SELECT allows the selection of the active HDMI port. On the ADV7842 no HDMI port is selected by default, therefore this register must be set to activate either HDMI port A or HDMI port B.

HDMI_PORT_SELECT[1:0], Addr 68 (HDMI), *Address 0x00[1:0]*

HDMI primary port selection control.

Function

HDMI_PORT_SELECT[1:0]	Description
10	Port A
11	Port B

8.15 FAST SWITCHING AND BACKGROUND PORT SELECTION

The ADV7842 incorporates a fast switching feature. This feature allows the user of a system containing the ADV7842 to seamlessly switch between HDCP encrypted sources. There is no delay in achieving video output which was previously caused by HDCP authentication. The time required to switch between HDMI sources with HDCP encryption is reduced to a fraction of a second.

If an HDMI port is not selected by HDMI_PORT_SELECT then by default this port is disabled. Asserting EN_BG_PORT_X allows this unselected port to be enabled in background mode (where X = A or B). Once a port is in background mode the ADV7842 establishes a HDCP link with its source even though it is not selected by HDMI_PORT_SELECT. This background authentication allows for fast switching of the HDMI ports.

Note: EN_BG_PORT_X has no effect if the port is selected by HDMI_PORT_SELECT.

EN_BG_PORT_A, Addr 68 (HDMI), *Address 0x00[5]*

Background mode enable for Port A. Sets Port A in background mode to establish a HDCP link with its source, even if the port is not selected by HDMI_PORT_SELECT. This control has no effect if the port is selected by HDMI_PORT_SELECT.

Function

EN_BG_PORT_A	Description
0 «	Port disabled, unless selected with HDMI_PORT_SELECT
1	Port enabled in background mode.

EN_BG_PORT_B, Addr 68 (HDMI), *Address 0x00[4]*

Background mode enable for Port B. Sets Port B in background mode to establish a HDCP link with its source, even if the port is not selected by HDMI_PORT_SELECT. This control has no effect if the port is selected by HDMI_PORT_SELECT.

Function

EN_BG_PORT_B	Description
0 «	Port disabled, unless selected with HDMI_PORT_SELECT
1	Port enabled in background mode.

The ADV7842 can also perform HDMI parameter measurements on one background port.
The following information can then be read from the background measurement and parameter registers.

- BG_TMDSFREQ[8:0]
- BG_TMDSFREQ_FRAC[6:0]
- BG_DEEP_COLOR_MODE[1:0]
- BG_PIX_REP[3:0]
- BG_PARAM_LOCK
- BG_TOTAL_LINE_WIDTH[13:0]
- BG_LINE_WIDTH[12:0]
- BG_TOTAL_FIELD_HEIGHT[12:0]
- BG_FIELD_HEIGHT[12:0]
- BG_HDMI_INTERLACED

BG_MEAS_PORT_SEL[1:0], Addr 68 (HDMI), *Address 0x00[3:2]*

BG_MEAS_PORT_SEL[1:0] selects a background port on which HDMI measurements are to be made and provided in the background measurement registers. The port in question must be set as a background port in order for this setting to be effective. There is no conflict if this matches the port selected by HDMI_PORT_SELECT.

Function

BG_MEAS_PORT_SEL[1:0]	Description
10	Port A
11	Port B

BG_MEAS_REQ, Addr 68 (HDMI), *Address 0x5A[5] (Self-Clearing)*

This bit must be set to get correct measurements of the selected background port. Setting this control sends a request to update the synchronization parameter measurements of the currently selected background port. The port on which the measurement will be made is selected by BG_MEAS_PORT_SEL[1:0].

Function

BG_MEAS_REQ	Description
0 «	No request to update selected background port synchronization parameter measurements
1	Requests an update of the selected background port synchronization parameter measurements

Note: After setting the self clearing BG_MEAS_REQ bit, the measurements of the TMDS frequency and video parameters of the background ports are valid when BG_MEAS_DONE_RAW goes high.

BG_MEAS_DONE_RAW, Addr 40 (IO), *Address 0x8D[1] (Read Only)*

Status of Background port Measurement completed interrupt signal. When set to 1 it indicates measurements of TMDS frequency and video parameters on the selected background port have been completed. Once set, this bit will remain high until it is cleared via BG_MEAS_DONE_CLR.

Function

BG_MEAS_DONE_RAW	Description
0 «	Measurements of TMDS frequency and video parameters of background port not finished or not requested.
1	Measurements of TMDS frequency and video parameters of background port are ready

Note: This bit only informs the user that the measurement is complete and can be read back. One should ensure that BG_PARAM_LOCK is asserted so that the background parameter filters are locked and the measurement values are valid.

8.16 TMDS CLOCK ACTIVITY DETECTION

The ADV7842 provides circuitry to monitor TMDS clock activity on each of its two HDMI ports. The firmware can poll the appropriate registers for TMDS clock activity detection and configure the ADV7842 as desired.

HDMI_MODE, Addr 68 (HDMI), *Address 0x05[7] (Read Only)*

A readback to indicate whether the stream processed by the HDMI core is a DVI or an HDMI stream.

Function

HDMI_MODE	Description
0 «	DVI Mode Detected
1	HDMI Mode Detected

BG_HDMI_MODE, Addr 68 (HDMI), *Address 0xEB[0] (Read Only)*

This readback provides the HDMI/DVI mode status of the background port determined by BG_MEAS_PORT_SEL[1:0] and is updated continuously.

Function

BG_HDMI_MODE	Description
0 «	DVI Mode Detected
1	HDMI Mode Detected

TMDS_CLK_A_RAW, Addr 40 (IO), *Address 0x6A[1] (Read Only)*

Raw status of Port A TMDS Clock detection signal.

Function

TMDS_CLK_A_RAW	Description
0 «	No TMDS clock detected on Port A
1	TMDS clock detected on Port A.

TMDS_CLK_B_RAW, Addr 40 (IO), *Address 0x6A[0] (Read Only)*

Raw status of Port B TMDS Clock detection signal.

Function

TMDS_CLK_B_RAW	Description
0 «	No TMDS clock detected on Port B
1	TMDS clock detected on Port B.

Important:

- The clock detection flags are valid for a specific port as long as the TMDS clock and data termination have been enabled for that port.
- The clock detection flags are valid if the CP and HDMI core have been powered down by setting POWER_DOWN to 0.
- The clock detection flags are valid, irrespective of the mode the part is set into via the PRIM_MODE[3:0] register.

8.16.1 Clock and Data Termination Control

The ADV7842 provides controls for the TMDS clock and data termination on all HDMI ports. The ADV7842 also offers automatic or manual termination closure of the selected port, and individual manual control over the two ports.

Note: The clock termination of the port selected by HDMI_PORT_SELECT[1:0] must always be enabled.

TERM_AUTO, Addr 68 (HDMI), *Address 0x01[7]*

This bit allows the user to select automatic or manual control of clock termination. If automatic mode termination is enabled, then the termination on the port selected via HDMI_PORT_SELECT[1:0] is enabled. The termination is disabled on all other ports. When automatic mode is disabled the termination for each port is set individually by the CLOCK_TERMx_DISABLE.

Function

TERM_AUTO	Description
0 «	Disable Termination automatic control
1	Enable Termination automatic control

Note: To enable the fast switching feature the termination should be set manually for each port. When manual mode is enabled the termination for each port is set individually by the CLOCK_TERMX_DISABLE control bits (were X = A or B)

CLOCK_TERMA_DISABLE, Addr 68 (HDMI), *Address 0x01[5]*

Disable clock termination on Port A. Can be used when TERM_AUTO set to 0

Function

CLOCK_TERMA_DISABLE	Description
0	Enable Termination Port A
1 «	Disable Termination Port A

CLOCK_TERMB_DISABLE, Addr 68 (HDMI), *Address 0x01[6]*

Disable clock termination on Port B. Can be used when TERM_AUTO set to 0

Function

CLOCK_TERMB_DISABLE	Description
0	Enable Termination Port B
1 «	Disable Termination Port B

8.17 TMDS MEASUREMENT

The ADV7842 contains logic that measures the frequency of the TMDS clock transmitted on the TMDS clock channel. The TMDS frequency can be read back via the TMDSFREQ[8:0] and TMDSFREQ_FRAC[6:0] registers.

8.17.1 TMDS Measurement After TMDS PLL

The TMDSFREQ measurement is provided by a clock measurement circuit located after the TMDS PLL. The TMDS PLL must, therefore, be locked to the incoming TMDS clock in order for the TMDSFREQ and TMDSFREQ_FRAC registers to return a valid measurement. The TMDS frequency can be obtained using Equation 5.

$$F_{TMDS} = TMDSFREQ + \frac{TMDSFREQ\_FRAC}{128}$$

Equation 5: TMDS Frequency in MHz (Measured After TMDS PLL)

Notes:

- The TMDS PLL lock status can be monitored via TMDS_PLL_LOCKED. Figure 73 shows the algorithm that can be implemented on an external controller to monitor the TMDS clock frequency.
- The TMDS_PLL_LOCKED flag should be considered valid if a TMDS clock is input on the HDMI port selected via HDMI_PORT_SELECT[1:0].
- The NEW_TMDS_FRQ_RAW flag can be used to monitor if the TMDS frequency on the selected HDMI port changes by a programmable threshold
- The ADV7842 can be configured to trigger an interrupt when the bit NEW_TMDS_FRQ_RAW changes from 0 to 1. In that configuration, the interrupt status NEW_TMDS_FRQ_ST indicates that NEW_TMDS_FRQ_RAW has changed from 0 to 1.

Refer to Section 16 for additional information on the configuration of interrupts.

TMDSFREQ[8:0], Addr 68 (HDMI), *Address 0x51[7:0]; Address 0x52[7] (Read Only)*

This register provides a full precision integer TMDS frequency measurement

Function

TMDSFREQ[8:0]	Description
000000000 «	Default value
xxxxxxxxx	Outputs 9-bit TMDS frequency measurement in MHz

TMDSFREQ_FRAC[6:0], Addr 68 (HDMI), *Address 0x52[6:0] (Read Only)*

A readback to indicate the fractional bits of measured frequency of PLL recovered TMDS clock. The unit is 1/128 MHz.

Function

TMDSFREQ_FRAC[6:0]	Description
0000000 «	Outputs 7-bit TMDS fractional frequency measurement in 1/128MHz
xxxxxxx	Outputs 7-bit TMDS fractional frequency measurement in 1/128MHz

BG_TMDSFREQ[8:0], Addr 68 (HDMI), *Address 0xE0[7:0]; Address 0xE1[7] (Read Only)*

This register provides a precision integer TMDS frequency measurement on the background port selected by BG_MEAS_PORT_SEL. The value provided is the result of a single measurement of the TMDS PLL frequency in MHz. This value is updated when an update request is made via the BG_MEAS_REQ control bit. This measurement is only valid when BG_PARAM_LOCK is set to 1.

Function

BG_TMDSFREQ[8:0]	Description
000000000 «	Outputs 9-bit TMDS frequency measurement in MHz
xxxxxxxxx	Outputs 9-bit TMDS frequency measurement in MHz

BG_TMDSFREQ_FRAC[6:0], Addr 68 (HDMI), *Address 0xE1[6:0] (Read Only)*

This register provides a precision fractional measurement of the TMDS frequency on the background port selected by BG_MEAS_PORT_SEL. The unit is 1/128 MHz and the value is updated when a update request is made via the BG_MEAS_REQ control bit. This measurement is only valid when BG_PARAM_LOCK is set to 1.

Function

BG_TMDSFREQ_FRAC[6:0]	Description
0000000 «	Outputs 7-bit TMDS fractional frequency measurement in 1/128MHz
xxxxxxx	Outputs 7-bit TMDS fractional frequency measurement in 1/128MHz

TMDS_PLL_LOCKED, Addr 68 (HDMI), *Address 0x04[1] (Read Only)*

A readback to indicate if the TMDS PLL is locked to the TMDS clock input to the selected HDMI port.

Function

TMDS_PLL_LOCKED	Description
0 «	The TMDS PLL is not locked
1	The TMDS PLL is locked to the TMDS clock input to the selected HDMI port.

TMDSPLL_LCK_A_RAW, Addr 40 (IO), *Address 0x6A[5] (Read Only)*

A readback to indicate the raw status of the Port A TMDS PLL lock signal.

Function

TMDSPLL_LCK_A_RAW	Description
0 «	TMDS PLL on Port A is not locked.
1	TMDS PLL on Port A is locked to the incoming clock.

TMDSPLL_LCK_B_RAW, Addr 40 (IO), *Address 0x6A[4] (Read Only)*

A readback to indicate the raw status of the port B TMDS PLL lock signal.

Function

TMDSPLL_LCK_B_RAW	Description
0 «	TMDS PLL on Port B is not locked.
1	TMDS PLL on Port B is locked to the incoming clock.

NEW_TMDS_FRQ_RAW, Addr 40 (IO), *Address 0x83[1] (Read Only)*

Status of New TMDS Frequency interrupt signal. When set to 1 it indicates the TMDS Frequency has changed by more than the tolerance set in FREQTOLERANCE[3:0]. Once set, this bit will remain high until it is cleared via NEW_TMDS_FREQ_CLR.

Function

NEW_TMDS_FRQ_RAW	Description
0 «	TMDS frequency has not changed by more than tolerance set in FREQTOLERANCE[3:0] in the HDMI Map
1	TMDS frequency has changed by more than tolerance set in FREQTOLERANCE[3:0] in the HDMI Map

FREQTOLERANCE[3:0], Addr 68 (HDMI), *Address 0x0D[3:0]*

Sets the tolerance in MHz for new TMDS frequency detection. This tolerance is used for the audio mute mask MT_MSK_VCLK_CHNG and the HDMI status bit NEW_TMDS_FRQ_RAW.

Function

FREQTOLERANCE[3:0]	Description
0100 «	Default tolerance in MHz for new TMDS frequency detection
xxxx	Tolerance in MHz for new TMDS frequency detection

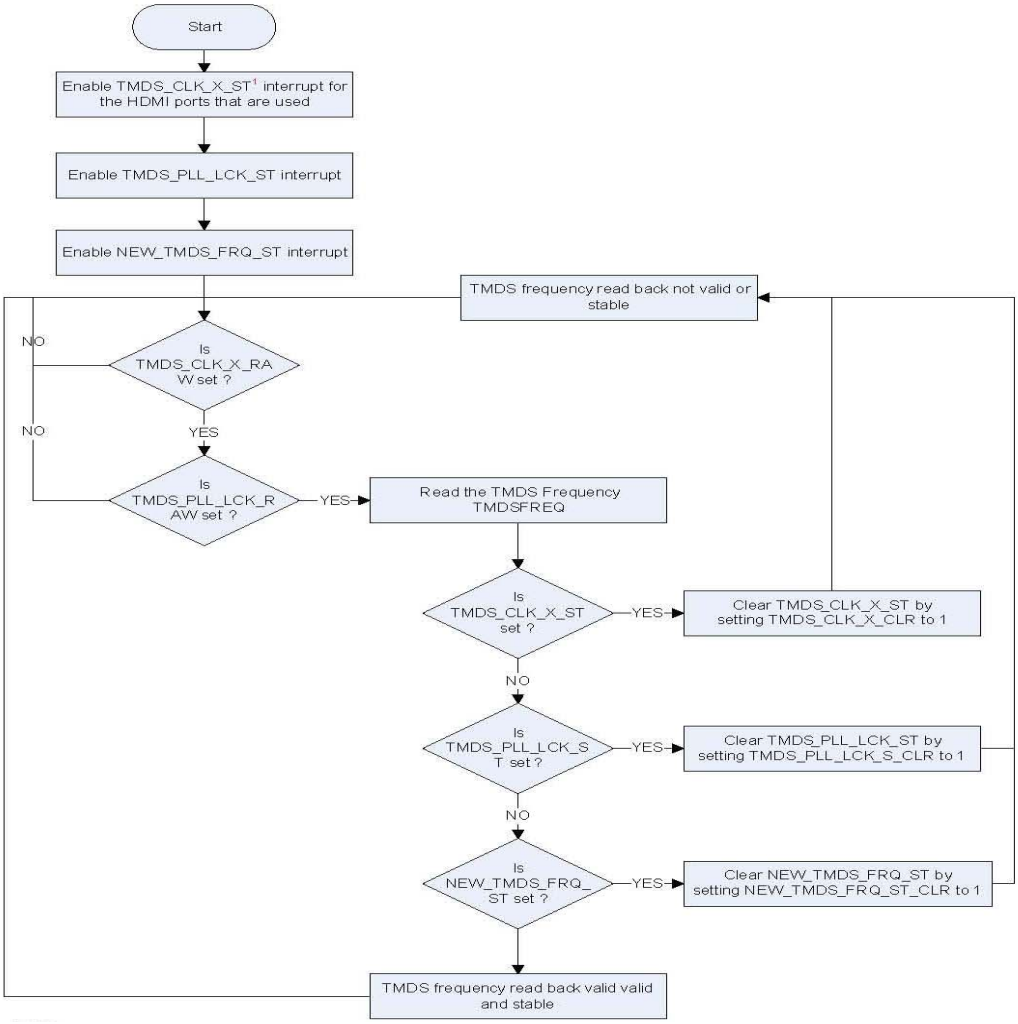


Figure 73: Monitoring TMDS Clock Frequency

8.18 DEEP COLOR MODE SUPPORT

The ADV7842 supports HDMI streams with 24 bit per sample and deep color modes of 30 or 36 bits per sample. The addition of a video FIFO (refer to Section 8.19) allows for the robust support of these modes.

The deep color mode information that the ADV7842 extracts from the general control packet can be read back from DEEP_COLOR_MODE[1:0]. It is possible to over ride the deep color mode that the ADV7842 unpacks from the video data encapsulated in the processed HDMI stream. This is achieved by configuring the OVERRIDE_DEEP_COLOR_MODE and DEEP_COLOR_MODE_USER[1:0] controls.

DEEP_COLOR_MODE[1:0], Addr 68 (HDMI), *Address 0x0B[7:6] (Read Only)*

A readback of the deep color mode information extracted from the general control packets.

Function

DEEP_COLOR_MODE[1:0]	Description
00 «	8-bits per channel
01	10-bits per channel
10	12-bits per channel
11	16-bits per channel (not supported)

OVERRIDE_DEEP_COLOR_MODE, Addr 68 (HDMI), *Address 0x40[6]*

A control to override the Deep Color mode.

Function

OVERRIDE_DEEP_COLOR_MODE	Description
0 «	The HDMI section unpacks the video data according to the deep-color information extracted from the General Control packets. (Normal operation)
1	Override the deep color mode extracted from the General Control Packet. The HDMI section unpacks the video data according to the Deep Color mode set in DEEP_COLOR_MODE_USER[1:0].

DEEP_COLOR_MODE_USER[1:0], Addr 68 (HDMI), *Address 0x40[5:4]*

A control to manually set the Deep Color mode. The value set in this register is only effective when OVERRIDE_DEEP_COLOR_MODE is set to 1.

Function

DEEP_COLOR_MODE_USER[1:0]	Description
00 «	8 bits per channel
01	10 bits per channel
10	12 bits per channel
11	16 bits per channel (not supported)

__ENDFIELD__**Notes:**

- Deep color mode can be monitored via DEEP_COLOR_CHNG_RAW, which indicates if the color depth of the processed HDMI stream has changed.
- The ADV7842 can be configured to trigger an interrupt when the DEEP_COLOR_CHNG_RAW bit changes from 0 to 1. In that configuration, the interrupt status DEEP_COLOR_CHNG_ST indicates that DEEP_COLOR_CHNG_RAW has changed from 0 to 1. Refer to Section 16 for additional information on the configuration of interrupts.

DEEP_COLOR_CHNG_RAW, Addr 40 (IO), *Address 0x83[7] (Read Only)*

Status of Deep Color Mode Changed Interrupt signal. When set to 1 it indicates a change in the deep color mode has been detected. Once set, this bit will remain high until it is cleared via DEEP_COLOR_CHNG_CLR.

Function

DEEP_COLOR_CHNG_RAW	Description
0 «	Deep color mode has not changed
1	Change in deep color triggered this interrupt

BG_DEEP_COLOR_MODE[1:0], Addr 68 (HDMI), *Address 0xEA[3:2] (Read Only)*

This readback provides the deep-color status for the background HDMI port determined by BG_MEAS_PORT_SEL[1:0]. The readback provides the HDMI color depth and is updated when an update request is made via the BG_MEAS_REQ control bit. This measurement is only valid when BG_PARAM_LOCK is set to 1.

Function

BG_DEEP_COLOR_MODE[1:0]	Description
00 «	8-bit color per channel
01	10-bit color per channel
10	12-bit color per channel
11	16-bit color per channel

8.19 VIDEO FIFO

The ADV7842 contains a FIFO located between the incoming TMDS data and the CP core (refer to Figure 74). Data arriving over the HDMI link will be at 1X for non deep color mode (8 bits per channel), and 1.25X, 1.5X, or 2X for deep color modes (30, 36 and 48 bits respectively). Data unpacking and data rate reduction must be performed on the incoming HDMI data to provide the CP core with the correct data rate and data bit width. The video FIFO is used to pass data safely across the clock domains.

The video FIFO also provides extreme robustness to jitter on the TMDS clock. The CP clock is generated by a DPLL running on the incoming TMDS clock, and the CP clock may contain less jitter than the incoming TMDS clock. The video FIFO provides immunity to the incoming jitter and the resultant clock phase mismatch between the CP clock and the TMDS clock.

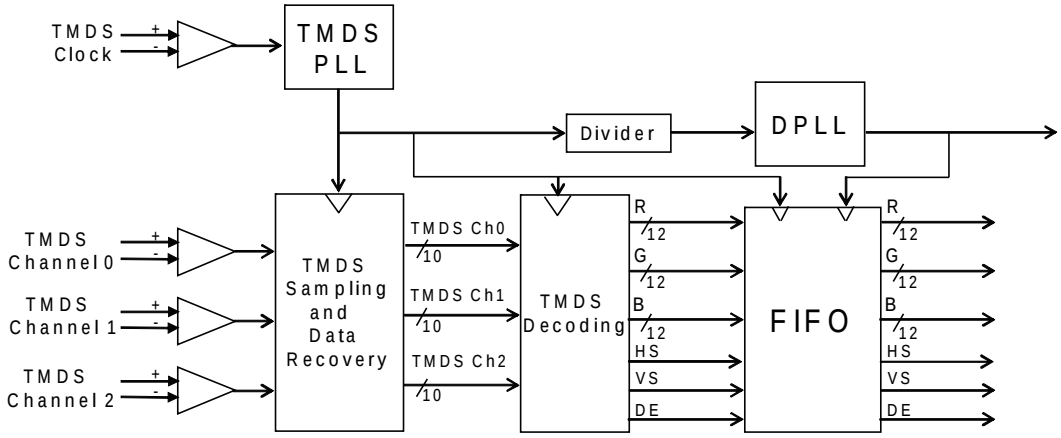


Figure 74: HDMI Video FIFO

The video FIFO is designed to operate completely autonomously. It automatically resynchronizes the read and write pointers if they are about to point to the same location. However, it is also possible for the user to observe and control the FIFO operation with a number of FIFO status and control registers.

DCFIFO_LEVEL[2:0], Addr 68 (HDMI), *Address 0x1C[2:0] (Read Only)*

A readback that indicates the distance between the read and write pointers. Overflow/underflow would read as level 0. Ideal centered functionality would read as 0b100.

Function

DCFIFO_LEVEL[2:0]	Description
000 «	FIFO has underflowed or overflowed
001	FIFO is about to overflow
010	FIFO has some margin.
011	FIFO has some margin.
100	FIFO perfectly balanced
101	FIFO has some margin.
110	FIFO has some margin.
111	FIFO is about to underflow

DCFIFO_LOCKED, Addr 68 (HDMI), *Address 0x1C[3] (Read Only)*

A readback to indicate if Video FIFO is locked.

Function

DCFIFO_LOCKED	Description
0 «	Video FIFO is not locked. Video FIFO had to resynchronize between previous two Vsyncs
1	Video FIFO is locked. Video FIFO did not have to resynchronize between previous two Vsyncs

DCFIFO_RECENTER, Addr 68 (HDMI), *Address 0x5A[2] (Self-Clearing)*

A reset to recenter the Video FIFO. This is a self clearing bit.

Function

DCFIFO_RECENTER	Description
0 «	Video FIFO normal operation.
1	Video FIFO to re-centre.

DCFIFO_KILL_DIS, Addr 68 (HDMI), *Address 0x1B[2]*

The Video FIFO output is zeroed if there is more than one resynchronization of the pointers within 2 FIFO cycles. This behavior can be disabled with this bit.

Function

DCFIFO_KILL_DIS	Description
0 «	FIFO output set to zero if more than one resynchronization is necessary during two FIFO cycles
1	FIFO output never set to zero regardless of how many resynchronizations occur

DCFIFO_KILL_NOT_LOCKED, Addr 68 (HDMI), *Address 0x1B[3]*

DCFIFO_KILL_NOT_LOCKED controls whether or not the output of the Video FIFO is set to zero when the video PLL is unlocked.

Function

DCFIFO_KILL_NOT_LOCKED	Description
0	FIFO data is output regardless of video PLL lock status
1 «	FIFO output is zeroed if video PLL is unlocked

The DCFIFO is programmed to reset itself automatically when the video PLL transitions from unlocked to locked. Note that the video PLL transition does not necessarily indicate that the overall system is stable.

DCFIFO_RESET_ON_LOCK, Addr 68 (HDMI), *Address 0x1B[4]*

Enables the reset/re-centering of video FIFO on video PLL unlock

Function

DCFIFO_RESET_ON_LOCK	Description
0	Do not reset on video PLL lock
1 «	Reset FIFO on video PLL lock

8.20 PIXEL REPETITION

In HDMI mode, video formats with TMDS rates below 25 Mpixels/s require pixel repetition in order to be transmitted over the TMDS link. When the ADV7842 receives this type of video format, it discards repeated pixel data automatically, based on the Pixel Repetition field available in the AVI InfoFrame.

When HDMI_PIXEL_REPETITION is non zero, video pixel data is discarded and the pixel clock frequency is divided by (HDMI_PIXEL_REPETITION) + 1.

HDMI_PIXEL_REPETITION[3:0], Addr 68 (HDMI), *Address 0x05[3:0] (Read Only)*

A readback to provide the current HDMI pixel repetition value decoded from the AVI Infoframe received. The HDMI receiver automatically discards repeated pixel data and divides the pixel clock frequency appropriately as per the pixel repetition value.

Function

HDMI_PIXEL_REPETITION[3:0]	Description
0000 «	1x
0001	2x
0010	3x
0011	4x
0100	5x
0101	6x
0110	7x
0111	8x
1000	9x
1001	10x
1010 - 1111	Reserved

DEREP_N_OVERRIDE, Addr 68 (HDMI), *Address 0x41[4]*

This control allows the user to override the pixel repetition factor. The ADV7842 then uses DEREP_N instead of HDMI_PIXEL_REPETITION[3:0] to discard video pixel data from the incoming HDMI stream.

Function

DEREP_N_OVERRIDE	Description
0 «	Automatic detection and processing of procession of pixel repeated modes using the AVI infoframe information.
1	Enables manual setting of the pixel repetition factor as per DEREP_N[3:0].

DEREP_N[3:0], Addr 68 (HDMI), *Address 0x41[3:0]*

Sets the derepetition value if derepetition is overridden by setting DEREP_N_OVERRIDE.

Function

DEREP_N[3:0]	Description
0000 «	DEREP_N+1 indicates the pixel and clock discard factor
xxxx	DEREP_N+1 indicates the pixel and clock discard factor

BG_PIX_REP[3:0], Addr 68 (HDMI), *Address 0xEA[7:4] (Read Only)*

Background port pixel repetition status for the background HDMI port determined by BG_MEAS_PORT_SEL[1:0]. The readback provides the pixel repetition value in AVI Infoframe and is updated when an update request is made via the BG_MEAS_REQ control bit. This measurement is only valid when BG_PARAM_LOCK is set to 1.

Function

BG_PIX_REP[3:0]	Description
0000 «	1x
0001	2x
0010	3x
0011	4x
0100	5x
0101	6x
0110	7x
0111	8x
1000	9x
1001	10x
1010 - 1111	Reserved

8.21 HDCP SUPPORT

8.21.1 HDCP Decryption Engine

The HDCP decryption engine allows for the reception and decryption of HDCP content-protected video and audio data. In the HDCP authentication protocol, the transmitter authenticates the receiver by accessing the HDCP registers of the ADV7842 over the DDC bus. Once the authentication is initiated, the HDCP decryption integrated in the ADV7842 computes and updates a decryption mask for every video frame. This mask is applied to the incoming data at every clock cycle to yield decrypted video and audio data.

HDMI_CONTENT_ENCRYPTED, Addr 68 (HDMI), *Address 0x05[6] (Read Only)*

A readback to indicate the use of HDCP encryption.

Function

HDMI_CONTENT_ENCRYPTED	Description
0 «	The input stream processed by the HDMI core is not HDCP encrypted
1	The input stream processed by the HDMI core is HDCP encrypted

HDMI_ENCRPT_X_RAW reports the encryption status of the data present on each individual HDMI port (where X = A or B).

Note: These bits are reset to 0 if an HDMI packet detection reset occurs. (Refer to Section 8.41)

HDMI_ENCRPT_A_RAW, Addr 40 (IO), *Address 0x6F[5] (Read Only)*

Raw status of Port A Encryption detection signal.

Function

HDMI_ENCRPT_A_RAW	Description
0 «	Current frame in Port A is not encrypted.
1	Current frame in Port A is encrypted.

HDMI_ENCRPT_B_RAW, Addr 40 (IO), *Address 0x6F[4] (Read Only)*

Raw status of Port B Encryption detection signal.

Function

HDMI_ENCRPT_B_RAW	Description
0 «	Current frame in Port B is not encrypted.
1	Current frame in Port B is encrypted.

Notes:

- The ADV7842 supports the 1.1_FEATURES, FAST_REAUTHENTICATION, and FAST_I2C speed HDCP features. The BCAPS register must be initialized appropriately if these features are to be supported by the application integrating the ADV7842, e.g. set BCAPS[0] to 1 to support FAST_REAUTHENTICATION.

- It is recommended to set BCAPS[7:0] bit [7] to 1 if the ADV7842 is used as the front end of an HDMI receiver. This bit should be set to 0 for DVI applications.

8.21.2 Internal HDCP Key OTP ROM

The ADV7842 features an on-chip nonvolatile memory that is preprogrammed with a set of HDCP keys.

8.21.3 HDCP Keys Access Flags

The ADV7842 accesses the internal HDCP key OTP ROM (also referred to as HDCP ROM) on two different occasions:

- After a power up, the ADV7842 reads the KSV from the internal HDCP ROM (refer to Figure 75).
- After a KSV update from an HDCP transmitter, the ADV7842 reads the KSV and all keys in order to carry out the link verification response (refer to Figure 76).

The host processor can read the HDCP_KEYS_READ and HDCP_KEY_ERROR flags to check that the ADV7842 successfully accessed the HDCP ROM.

HDCP_KEYS_READ, Addr 68 (HDMI), *Address 0x04[5] (Read Only)*

A readback to indicate a successful read of the HDCP keys and/or KSV from the internal HDCP Key OTP ROM. A logic high is returned when the read is successful.

Function

HDCP_KEYS_READ	Description
0 «	HDCP keys and/or KSV not yet read
1	HDCP keys and/or KSV read

HDCP_KEY_ERROR, Addr 68 (HDMI), *Address 0x04[4] (Read Only)*

A readback to indicate if a checksum error occurred while reading the HDCP and/or KSV from the HDCP Key ROM. Returns 1 when the HDCP Key master encounters an error while reading the HDCP Key OTP ROM.

Function

HDCP_KEY_ERROR	Description
0 «	No error occurred while reading HDCP keys
1	HDCP keys read error

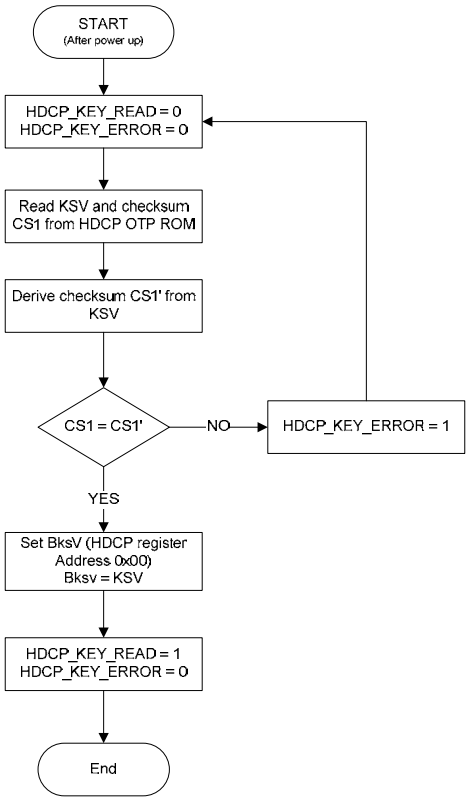


Figure 75: HDCP ROM Access After Power Up

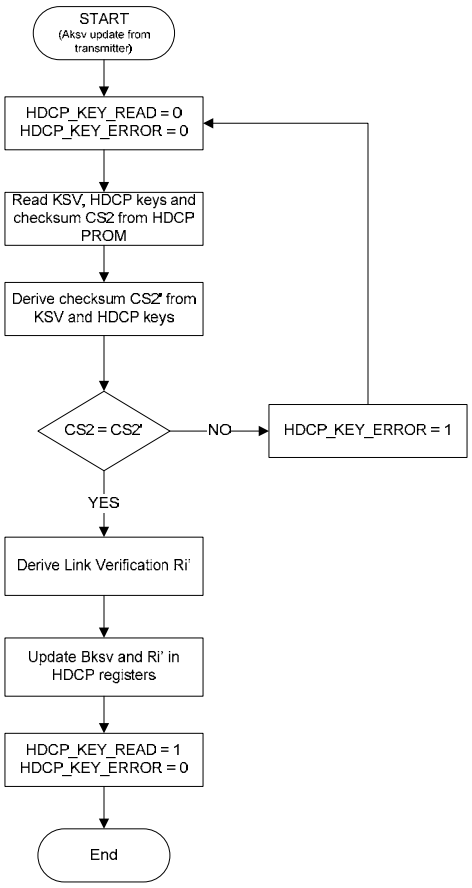


Figure 76: HDCP ROM Access After KSV Update from the Transmitter

Notes:

- After the part has powered up, it is recommended to wait for 1 ms before checking the HDCP_KEYS_READ and HDCP_KEY_ERROR flag bits. This ensures that the ADV7842 had sufficient time to access the internal HDCP ROM and set the HDCP_KEYS_READ and HDCP_KEY_ERROR flag bits.
- After an AKSV update from the transmitter, it is recommended to wait for 2 ms before checking the HDCP_KEYS_READ and HDCP_KEY_ERROR flag bits. This ensures that the ADV7842 had sufficient time to access the internal HDCP ROM, and set the HDCP_KEYS_READ and HDCP_KEY_ERROR flag bits.
- When the ADV7842 successfully retrieves the HDCP keys and/or KSV from the internal HDCP ROM, the HDCP_KEYS_READ flag bit is set to 1 and the HDCP_KEY_ERROR flag bit is set to 0.
- The I^2C controllers for the main I^2C lines and the HDCP lines are independent of each other. It is, therefore, possible to access the internal registers of the ADV7842 while it reads the HDCP keys and/or the KSV from the internal HDCP ROM.
- A hardware reset (i.e. reset via the $\overline{\text{RESET}}$ pin) does not lead the ADV7842 to read the KSV or the keys from the HDCP ROM.
- The ADV7842 takes 1.8 ms to read the keys from the HDCP ROM

8.22 HDMI SYNCHRONIZATION PARAMETERS

The ADV7842 contains the logic required to measure the details of the incoming video resolution. The HDMI synchronization parameters readback registers from the HDMI Map can be used, in addition to the STDI registers from the CP (refer to Section 10.8.2), to estimate the video resolution of the incoming HDMI stream.

Notes:

- The synchronization parameters are valid if the part is configured in HDMI mode via PRIM_MODE[3:0].
- The HDMI synchronization filter readback parameters are valid even while the part free runs (refer to Section 10.13) and/or when it is configured to process analog inputs in simultaneous mode (refer to Section 3.4) on the condition that the measurement filters have locked.

8.22.1 Horizontal Filter and Measurements

The HDMI horizontal filter performs measurements on the DE and HSync of the HDMI stream on the selected port. The ADV7842 also performs horizontal measurements on the background port as selected by BG_MEAS_PORT_SEL[1:0]. These measurements are available in the HDMI Map and can be used to determine the resolution of the incoming video datastreams.

8.22.2 Primary Port Horizontal Filter Measurements

The HDMI horizontal filter performs the measurements described in this section on the HDMI port selected by HDMI_PORT_SELECT[1:0].

Notes:

- The horizontal measurements are valid only if DE_REGEN_LCK_RAW is set to 1.
- The HDMI horizontal filter is used solely to measure the horizontal synchronization signals decoded from the HDMI stream. The HDMI horizontal filter is not in the main path of the synchronization processed by the part and does not delay the overall HDMI data in, to video data out latency.
- The unit for horizontal filter measurement is a pixel, which is the actual element of the picture content encapsulated in the HDMI/DVI stream which the ADV7842 processes. A pixel has a duration T_{Pixel} which is provided in Equation 6.

$$T_{Pixel} = T_{FTMDS} \cdot DEEP\_COLOR\_RATIO \cdot (PIXEL\_REPETITION + 1)$$

where:

T_{FTMDS} is the TMDS frequency
$DEEP\_COLOR\_RATIO = 1$ for 24-bit deep color
$DEEP\_COLOR\_RATIO = 5/4$ for 30-bit deep color
$DEEP\_COLOR\_RATIO = 3/2$ for 36-bit deep color
$DEEP\_COLOR\_RATIO = 2$ for 48-bit deep color
$PIXEL\_REPETITION$ is the number of repeated pixels in the input HDMI stream

Equation 6: Unit Time of Horizontal Filter Measurements

DE_REGEN_FILTER_LOCKED, Addr 68 (HDMI), *Address 0x07[5] (Read Only)*

DE regeneration filter lock status. Indicates that the DE regeneration section has locked to the received DE and horizontal synchronization parameter measurements are valid for readback.

Function

DE_REGEN_FILTER_LOCKED	Description
0 «	DE regeneration filter is not locked.
1	DE regeneration filter is locked.

DE_REGEN_LCK_RAW, Addr 40 (IO), *Address 0x74[0] (Read Only)*

Raw status of the DE regeneration lock signal.

Function

DE_REGEN_LCK_RAW	Description
0 «	DE regeneration block has not been locked
1	DE regeneration block has been locked to the incoming DE signal

TOTAL_LINE_WIDTH[13:0], Addr 68 (HDMI), *Address 0x1E[5:0]; Address 0x1F[7:0] (Read Only)*

Total line width is a horizontal synchronization measurement. This gives the total number of pixels per line. This measurement is valid only when the DE regeneration filter has locked.

Function

TOTAL_LINE_WIDTH[13:0]	Description
00000000000000 «	Default value
xxxxxxxxxxxxxx	Total number of pixels per line.

LINE_WIDTH[12:0], Addr 68 (HDMI), *Address 0x07[4:0]; Address 0x08[7:0] (Read Only)*

Line width is a horizontal synchronization measurement. The gives the number of active pixels in a line. This measurement is only valid when the DE regeneration filter is locked.

Function

LINE_WIDTH[12:0]	Description
00000000000 «	Default value
xxxxxxxxxxx	Total number of active pixels per line.

HSYNC_FRONT_PORCH[12:0], Addr 68 (HDMI), *Address 0x20[4:0]; Address 0x21[7:0] (Read Only)*

HSync front porch width is a horizontal synchronization measurement. The unit of this measurement is unique pixels. This measurement is valid only when the DE regeneration filter has locked.

Function

HSYNC_FRONT_PORCH[12:0]	Description
0000000000000 «	Default value
xxxxxxxxxxxxx	Total number of pixels in the front porch.

HSYNC_PULSE_WIDTH[12:0], Addr 68 (HDMI), *Address 0x22[4:0]; Address 0x23[7:0] (Read Only)*

HSync pulse width is a horizontal synchronization measurement. The unit of this measurement is unique pixels. This measurement is valid only when the DE regeneration filter has locked.

Function

HSYNC_PULSE_WIDTH[12:0]	Description
0000000000000 «	Default value
xxxxxxxxxxxxx	Total number of pixels in the hsync pulse.

HSYNC_BACK_PORCH[12:0], Addr 68 (HDMI), *Address 0x24[4:0]; Address 0x25[7:0] (Read Only)*

HSync Back Porch width is a horizontal synchronization measurement. The unit of this measurement is unique pixels. This measurement is valid only when the DE regeneration filter has locked.

Function

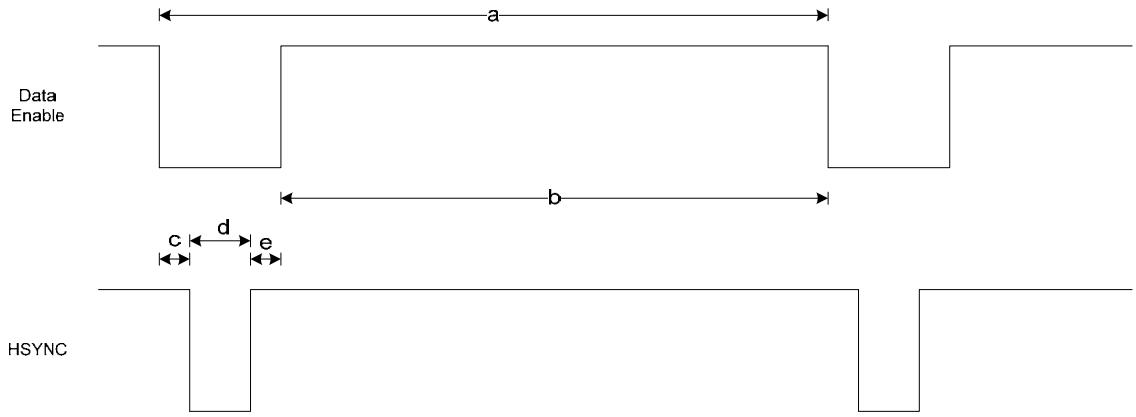
HSYNC_BACK_PORCH[12:0]	Description
0000000000000 «	Default value
xxxxxxxxxxxxx	Total number of pixels in the back porch.

DVI_HSYNC_POLARITY, Addr 68 (HDMI), *Address 0x05[5] (Read Only)*

A readback to indicate the polarity of the HSync encoded in the input stream

Function

DVI_HSYNC_POLARITY	Description
0 «	The HSync is active low
1	The HSync is active high



a Total number of pixels per line
b Active number of pixels per line
c HSync front porch width in pixel unit
d HSync width in pixel unit
e HSync back porch width in pixel unit

Figure 77: Horizontal Timing Parameters

8.22.3 Background Port Horizontal Filter Measurements

The HDMI horizontal filter performs the measurements described in this section on the HDMI port selected by BG_MEAS_PORT_SEL[1:0].

Note: BG_PARAM_LOCK must be high for background horizontal and vertical measurements to be valid.

BG_PARAM_LOCK, Addr 68 (HDMI), *Address 0xEA[1] (Read Only)*

A flag to indicate that vertical and horizontal parameters have been locked during a background measurement.

Function

BG_PARAM_LOCK	Description
0 «	Horizontal and Vertical were not locked when measurements for selected background HDMI port were taken.
1	Horizontal and Vertical were locked when measurements for selected background HDMI port were taken.

BG_TOTAL_LINE_WIDTH[13:0], Addr 68 (HDMI), *Address 0xE4[5:0]; Address 0xE5[7:0] (Read Only)*

Background port total line width, a horizontal synchronization measurement for the background HDMI Port determined by BG_MEAS_PORT_SEL[1:0]. The value represents the total number of pixels in a line and is updated when a update request is made via the BG_MEAS_REQ control bit. This measurement is only valid when BG_PARAM_LOCK is set to 1.

Function

BG_TOTAL_LINE_WIDTH[13:0]	Description
xxxxxxxxxxxxx	The total number of pixels per line on the background measurement port

BG_LINE_WIDTH[12:0], Addr 68 (HDMI), *Address 0xE2[4:0]; Address 0xE3[7:0] (Read Only)*

Background port line width, a horizontal synchronization measurement for the background HDMI Port determined by BG_MEAS_PORT_SEL[1:0]. The value represents the number of active pixels in a line and is updated when an update request is made via the BG_MEAS_REQ control bit.

Function

BG_LINE_WIDTH[12:0]	Description
0000000000000 «	The number of active pixels per line on the background measurement port.
xxxxxxxxxxxxx	The number of active pixels per line on the background measurement port.

8.22.4 Horizontal Filter Locking Mechanism

The locking/unlocking mechanism of the HDMI horizontal filter is as follows:

- The HDMI horizontal filter locks if the following two conditions are met:
  - The DE transitions occur at the exact same pixel count for 8 consecutive video lines
  - The HSync transitions occur at the exact pixel count for 8 consecutive video lines

- The HDMI horizontal filter unlocks if *either* of the two following conditions are met:
  - The DE transitions occur on different pixels count for 15 consecutive video lines
  - The HSync transitions occur on different pixels count for 15 consecutive video lines

8.22.5 Vertical Filters and Measurements

The ADV7842 integrates a HDMI vertical filter which performs measurements on the VSync of the HDMI stream on the selected port. The ADV7842 also performs vertical measurements on the background port as selected by BG_MEAS_PORT_SEL[1:0].These measurements are available in the HDMI Map and can be used to determine the resolution of the incoming video datastreams.

8.22.6 Primary Port Vertical Filter Measurements

The HDMI vertical filter performs the measurements described in this section on the HDMI port selected by HDMI_PORT_SELECT[1:0].

The field 0 measurements are adequate to determine the standard of incoming progressive modes. A combination of field 0 and field 1 measurements should be used to determine the standard of interlaced modes.

Notes:

- The vertical measurements are valid only if V_LOCKED_RAW is set to 1.
- The HDMI vertical filter is used solely to measure the vertical synchronization signals decoded from the HDMI stream. This filter is not in the main path of the synchronization processed by the part and does not delay the overall HDMI data in to video data out latency.

VERT_FILTER_LOCKED, Addr 68 (HDMI), *Address 0x07[7] (Read Only)*

Vertical filter lock status. Indicates whether or not the vertical filter is locked and vertical synchronization parameter measurements are valid for readback.

Function

VERT_FILTER_LOCKED	Description
0 «	Vertical filter has not locked
1	Vertical filter has locked

V_LOCKED_RAW, Addr 40 (IO), *Address 0x74[1] (Read Only)*

Raw status of the Vertical Sync Filter Locked signal.

Function

V_LOCKED_RAW	Description
0 «	Vertical sync filter has not locked and vertical sync parameters are not valid
1	Vertical sync filter has locked and vertical sync parameters are valid

Note: Field 0 measurements are used to determine the video modes that are progressive.

FIELD0_TOTAL_HEIGHT[13:0], Addr 68 (HDMI), *Address 0x26[5:0]; Address 0x27[7:0] (Read Only)*

Field 0 total height is a vertical synchronization measurement. This readback gives the total number of half lines in Field 0. (Divide readback value by 2 to get number of lines.) This measurement is valid only when the vertical filter has locked.

Function

FIELD0_TOTAL_HEIGHT [13:0]	Description
0000000000000 «	Default value
xxxxxxxxxxxxx	The total number of half lines in Field 0.

FIELD0_HEIGHT[12:0], Addr 68 (HDMI), *Address 0x09[4:0]; Address 0x0A[7:0] (Read Only)*

Field 0 Height is a vertical filter measurement. This readback gives the number of active lines in field 0. This measurement is valid only when the vertical filter has locked.

Function

FIELD0_HEIGHT[12:0]	Description
0000000000000 «	Default value
xxxxxxxxxxxxx	The number of active lines in Field 0

FIELD0_VS_FRONT_PORCH[13:0], Addr 68 (HDMI), *Address 0x2A[5:0]; Address 0x2B[7:0] (Read Only)*

Field 0 VSync front porch width is a vertical synchronization measurement. The unit of this measurement is half lines. (Divide readback by 2 to get number of lines) This measurement is valid only when the vertical filter has locked.

Function

FIELD0_VS_FRONT_PORCH[13:0]	Description
00000000000000 «	Default value
xxxxxxxxxxxxxx	The total number of half lines in the VSync Front Porch of Field 0.

FIELD0_VS_PULSE_WIDTH[13:0], Addr 68 (HDMI), *Address 0x2E[5:0]; Address 0x2F[7:0] (Read Only)*

Field 0 VSync width is a vertical synchronization measurement. The unit for this measurement is half lines. (Divide readback value by 2 to get number of lines) This measurement is valid only when the vertical filter has locked.

Function

FIELD0_VS_PULSE_WIDTH[13:0]	Description
00000000000000 «	Default value
xxxxxxxxxxxxxx	The total number of half lines in the VSync Pulse of Field 0.

FIELD0_VS_BACK_PORCH[13:0], Addr 68 (HDMI), *Address 0x32[5:0]; Address 0x33[7:0] (Read Only)*

Field 0 VSync back porch width is a vertical synchronization measurement. The unit for this measurement is half lines. (Divide readback value by 2 to get number of lines)

Function

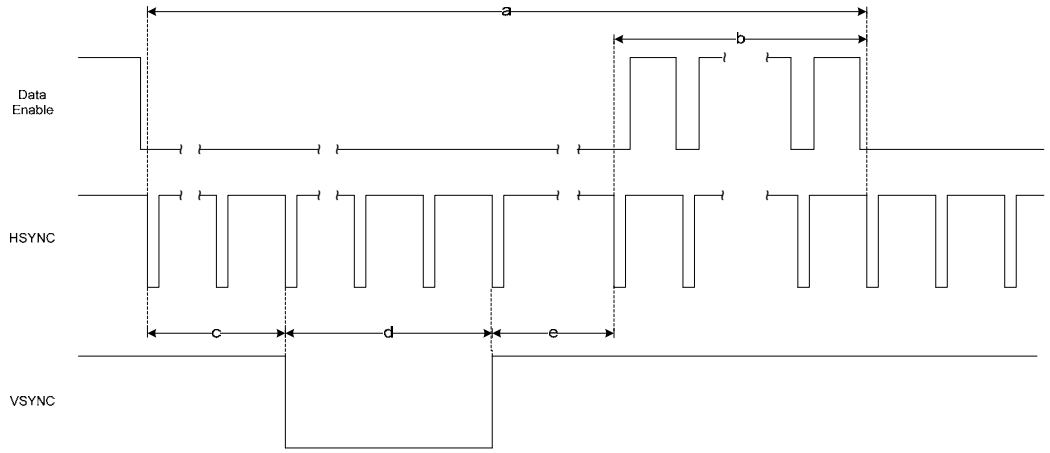
FIELD0_VS_BACK_PORCH[13:0]	Description
00000000000000 «	Default value
xxxxxxxxxxxxxx	The total number of half lines in the VSync Back Porch of Field 0.

DVI_VSYNC_POLARITY, Addr 68 (HDMI), *Address 0x05[4] (Read Only)*

A readback to indicate the polarity of the VSync encoded in the input stream

Function

DVI_VSYNC_POLARITY	Description
0 «	The VSync is active low
1	The VSync is active high



a Total number of lines in field 0. Unit is in half lines.
b Actives number of lines in field 0. Unit is in lines.
c VSync front porch width in field 0. Unit is in half lines.
d VSync pulse width in field 0. Unit is in half lines.
e VSync back porch width in field 0. Unit is in half lines.

Figure 78: Vertical Parameters for Field 0

Note: Field 1 measurements should not be used for progressive video modes.

FIELD1_TOTAL_HEIGHT[13:0], Addr 68 (HDMI), *Address 0x28[5:0]; Address 0x29[7:0] (Read Only)*

Field 1 total height is a vertical synchronization measurement. This readback gives the total number of half lines in Field 1. (Divide readback by 2 to get number of lines) This measurement is valid only when the vertical filter has locked. Field 1 measurements are valid when HDMI_INTERLACED is set to 1.

Function

FIELD1_TOTAL_HEIGHT [13:0]	Description
00000000000000 «	Default value
xxxxxxxxxxxxxx	The total number of half lines in Field 1.

FIELD1_HEIGHT[12:0], Addr 68 (HDMI), *Address 0x0B[4:0]; Address 0x0C[7:0] (Read Only)*

Field 1 height is a vertical filter measurement. This readback gives the number of active lines in field. This measurement is valid only when the vertical filter has locked. Field 1 measurements are only valid when HDMI_INTERLACED is set to 1.

Function

FIELD1_HEIGHT[12:0]	Description
0000000000000 «	Default value
xxxxxxxxxxxxx	The number of active lines in Field 1

FIELD1_VS_FRONT_PORCH[13:0], Addr 68 (HDMI), *Address 0x2C[5:0]; Address 0x2D[7:0] (Read Only)*

Field 1 VSync front porch width is a vertical synchronization measurement. The unit of this measurement is half lines. (Divide readback value by 2 to get number of lines) This measurement is valid only when the vertical filter has locked. Field 1 measurements are valid when HDMI_INTERLACED is set to 1

Function

FIELD1_VS_FRONT_PORCH[13:0]	Description
00000000000000 «	Default value
xxxxxxxxxxxxxx	The total number of half lines in the VSync Front Porch of Field 1.

FIELD1_VS_PULSE_WIDTH[13:0], Addr 68 (HDMI), *Address 0x30[5:0]; Address 0x31[7:0] (Read Only)*

Field 1 VSync width is a vertical synchronization measurement. The unit for this measurement is half lines. (Divide readback value by 2 to get number of lines) This measurement is valid only when the vertical filter has locked. Field 1 measurements are valid when HDMI_INTERLACED is set to 1

Function

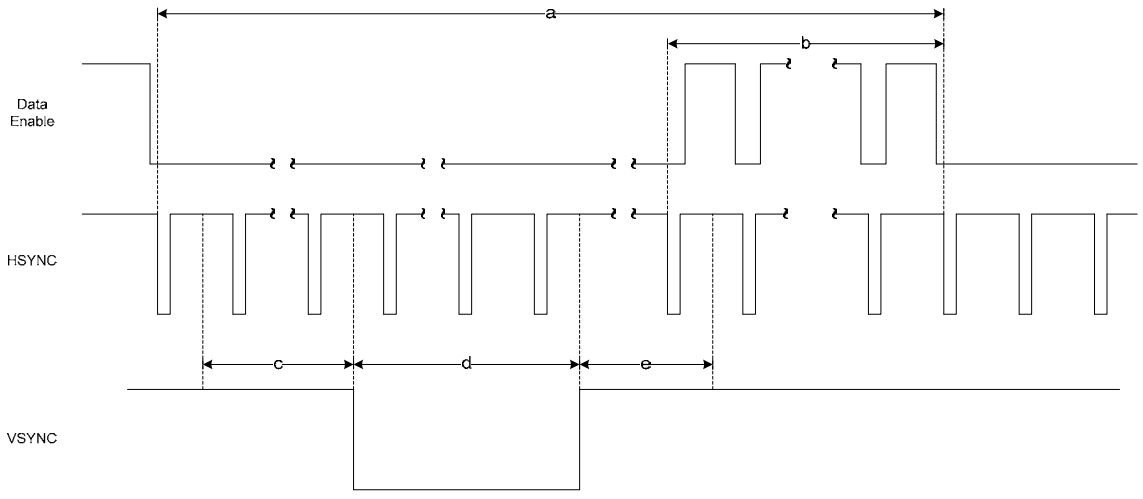
FIELD1_VS_PULSE_WIDTH[13:0]	Description
00000000000000 «	Default value
xxxxxxxxxxxxxx	The total number of half lines in the VSync Pulse of Field 1.

FIELD1_VS_BACK_PORCH[13:0], Addr 68 (HDMI), *Address 0x34[5:0]; Address 0x35[7:0] (Read Only)*

Field 1 VSync back porch width is a vertical synchronization measurement. The unit for this measurement is half lines. (Divide readback by 2 to get number of lines)This measurement is valid only when the vertical filter has locked. Field 1 measurements are valid when HDMI_INTERLACED is set to 1.

Function

FIELD1_VS_BACK_PORCH[13:0]	Description
00000000000000 «	Default value
xxxxxxxxxxxxx	The number of half lines in the VSync Back Porch of Field 1.



a Total number of lines in field 1. Unit is in half lines.

b Actives number of lines in field . Unit is in lines.

c	VSync front porch width in field 1. Unit is in half lines.
d	VSync pulse width in field 1. Unit is in half lines.
e	VSync back porch width in field 1. Unit is in half lines.

Figure 79: Vertical Parameters for Field 1

The vertical filter provides the interlaced status of the video stream. The interlaced status INTERLACED_HDMI is valid only if the vertical filter is locked and V_LOCKED_RAW is set to 1.

HDMI_INTERLACED, Addr 68 (HDMI), *Address 0x0B[5] (Read Only)*

HDMI input Interlace status, a vertical filter measurement.

Function

HDMI_INTERLACED	Description
0 «	Progressive Input
1	Interlaced Input

8.22.7 Background Port Vertical Filter Measurements

The HDMI vertical filter performs the measurements described in this section on the HDMI port selected by BG_MEAS_PORT_SEL[1:0].

Note: BG_PARAM_LOCK must be high for background horizontal and vertical measurements to be valid.

BG_TOTAL_FIELD_HEIGHT[12:0], Addr 68 (HDMI), *Address 0xE8[4:0]; Address 0xE9[7:0] (Read Only)*

Background port total field height is a vertical synchronization measurement for the background HDMI Port determined by BG_MEAS_PORT_SEL[1:0]. The value represents the total number of lines in a field and is updated when an update request is made via the BG_MEAS_REQ control bit.

Function

BG_TOTAL_FIELD_HEIGHT[12:0]	Description
0000000000000 «	The total number of lines in a Field on the background measurement port
xxxxxxxxxxxxx	The total number of lines in a Field on the background measurement port

BG_FIELD_HEIGHT[12:0], Addr 68 (HDMI), *Address 0xE6[4:0]; Address 0xE7[7:0] (Read Only)*

Background port field height is a vertical synchronization measurement for a background HDMI Port determined by BG_MEAS_PORT_SEL[1:0]. The value represents the number of active lines in a field and is updated when a update request is made via the BG_MEAS_REQ control bit.

Function

BG_FIELD_HEIGHT[12:0]	Description
0000000000000 «	The number of active lines in a Field on the background measurement port
xxxxxxxxxxxxx	The number of active lines in a Field on the background measurement port

BG_HDMI_INTERLACED, Addr 68 (HDMI), *Address 0xEA[0] (Read Only)*

Background port HDMI input interlace status is a vertical filter measurement for a background HDMI Port determined by BG_MEAS_PORT_SEL[1:0]. The status readback is updated when a update request is made via the BG_MEAS_REQ control bit. This measurement is only valid when BG_PARAM_LOCK is set to 1.

Function

BG_HDMI_INTERLACED	Description
0 «	Progressive Input
1	Interlaced Input

8.22.8 Vertical Filter Locking Mechanism

The HDMI vertical filter locks if the input VSync comes at exactly the same line count for two consecutive frames. The HDMI vertical filter unlocks if the VSync comes at a different pixels count for two consecutives frames.

8.23 AUDIO CONTROL AND CONFIGURATION

The ADV7842 extracts an L-PCM, IEC 61937 compressed, DSD, or high-bit rate (HBR) audio datastream from their corresponding audio packets (i.e. audio sample, DSD, or HBR packets) encapsulated inside the HDMI datastream.

The ADV7842 also regenerates an audio master clock along with the extraction of the audio data. The clock regeneration is performed by an integrated DPLL. The regenerated clock is used to output audio data from the 64 stereo sample depth FIFO to the audio interface configuration pins.

Important:

- The ADV7842 supports the extraction of stereo audio data (non compressed or compressed) at audio sampling frequency up to 192 kHz
- The ADV7842 supports the extraction of multichannel audio data

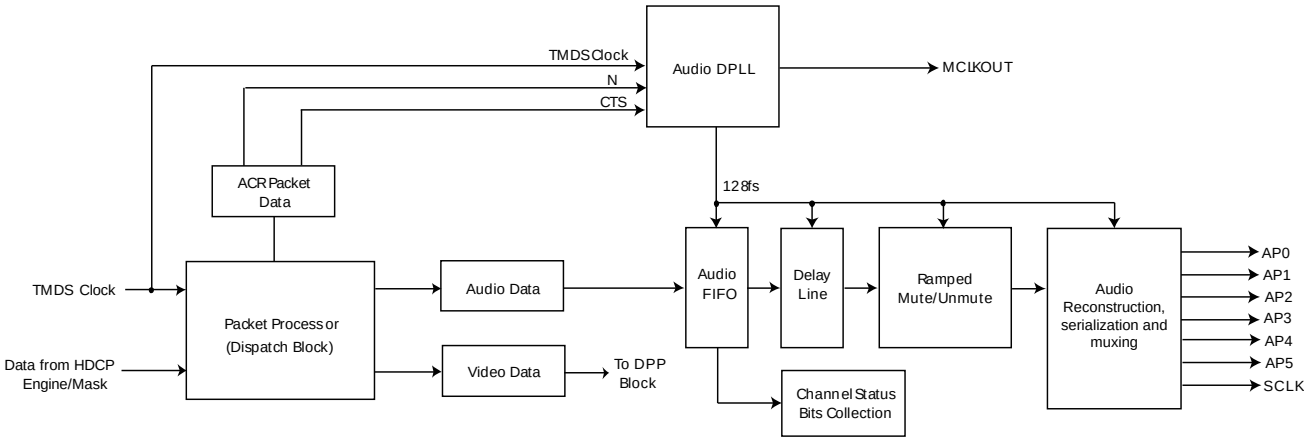


Figure 80: Audio Processor Block Diagram

8.23.1 Audio DPLL

The audio DPLL generates an internal audio master clock with a frequency of 128 times the audio sampling frequency, usually called fs. The audio master clock is used to clock the audio processing section.

8.23.2 Locking Mechanism

When the upstream HDMI transmitter outputs a stable TMDS frequency and consistent audio clock regeneration values, the audio DPLL locks within two cycles of the audio master clock after the following two conditions are met:

- TMDS PLL is locked (refer to TMDS_PLL_LOCKED)
- ADV7842 has received an ACR packet with N and CTS parameters within a valid range

The audio DPLL lock status can be monitored via AUDIO_PLL_LOCKED.

AUDIO_PLL_LOCKED, Addr 68 (HDMI), *Address 0x04[0] (Read Only)*

A readback to indicate the Audio DPLL lock status.

Function

AUDIO_PLL_LOCKED	Description
0 «	The audio DPLL is not locked
1	The audio DPLL is locked

8.23.3 ACR Parameters Loading Method

The N and CTS parameters from the ACR packets are used to regenerate the audio clock and are reloaded into the DPLL anytime they change. The self-clearing bit FORCE_N_UPDATE provides a means to reset the audio DPLL by forcing a reload of the N and CTS parameters from the ACR packet into the audio DPLL.

FORCE_N_UPDATE, Addr 68 (HDMI), *Address 0x5A[0] (Self-Clearing)*

A control to force an N and CTS value update to the audio DPLL. The audio DPLL regenerates the audio clock.

Function

FORCE_N_UPDATE	Description
0 «	No effect
1	Forces an update on the N and CTS values for audio clock regeneration

8.23.4 Audio DPLL Coast Feature

The audio DPLL incorporates a coast feature that allows it to indefinitely output a stable audio master clock when selectable events occur. The coast feature allows the audio DPLL to provide an audio master clock when the audio processor mutes the audio following a mute condition (refer to Section 8.29). The events that cause the audio DPLL to coast are selected via the coasts masks listed in Table 14.

Table 14: Selectable Coast Conditions

Bit Name	HDMI Map Address	Description	Corresponding Status Registers(s)
AC_MSK_VCLK_CHNG	0x13[6]	When set to 1, audio DPLL coasts if TMDS clock has any irregular/missing pulses	VCLK_CHNG_RAW
AC_MSK_VPLL_UNLOCK	0x13[5]	When set to 1, audio DPLL coasts if TMDS PLL unlocks	TMDS_PLL_LOCKED
AC_MSK_NEW_CTS	0x13[3]	When set to 1, audio DPLL coasts if CTS	CTS_PASS_THRSH_RAW

Bit Name	HDMI Map Address	Description	Corresponding Status Registers(s)
		changes by more than threshold set in CTS_CHANGE_THRESHOLD[5:0]	
AC_MSK_NEW_N	0x13[2]	When set to 1, audio DPLL coasts if N changes	CHANGE_N_RAW
AC_MSK_CHNG_PORT	0x13[1]	When set to 1, audio DPLL coasts if active port is changed	HDMI_PORT_SELECT[1:0]
AC_MSK_VCLK_DET	0x13[0]	When set to 1, audio DPLL coasts if no TMDS clock is detected on the active port	TMDS_CLK_A_RAW TMDS_CLK_B_RAW

8.24 AUDIO FIFO

The audio FIFO can store up to 128 audio stereo data from the audio sample, DSD, or HBR packets. Stereo audio data are added into the FIFO from the audio packet received. Stereo audio data are retrieved from the FIFO at a rate corresponding to 128 times the audio sampling frequency, f_s.

The status of the audio FIFO can be monitored through the status flags FIFO_UNDERFLO_RAW, FIFO_OVERFLO_RAW, FIFO_NEAR_OVFL_RAW, and FIFO_NEAR_UFLO_RAW.

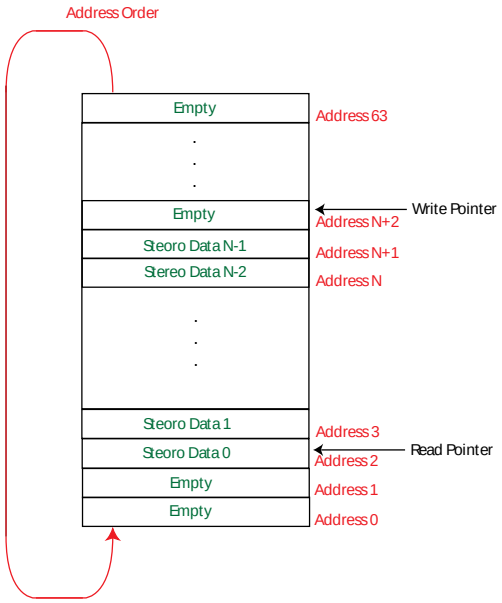


Figure 81: Audio FIFO

FIFO_UNDERFLO_RAW, Addr 40 (IO), *Address 0x7E[6] (Read Only)*

Status of Audio FIFO Underflow interrupt signal. When set to 1 it indicates the Audio FIFO read pointer has reached the write pointer causing the audio FIFO to underflow. Once set, this bit will remain high until it is cleared via AUDIO_FIFO_UNDERFLO_CLR.

Function

FIFO_UNDERFLO_RAW	Description
0 «	Audio FIFO has not underflowed
1	Audio FIFO has underflowed

FIFO_OVERFLO_RAW, Addr 40 (IO), *Address 0x7E[5] (Read Only)*

Status of Audio FIFO Overflow interrupt signal. When set to 1 it indicates Audio FIFO write pointer has reached the read pointer causing the audio FIFO to overflow. Once set, this bit will remain high until it is cleared via AUDIO_FIFO_OVERFLO_CLR.

Function

FIFO_OVERFLO_RAW	Description
0 «	Audio FIFO has not overflowed
1	Audio FIFO has overflowed

FIFO_NEAR_UFLO_RAW, Addr 40 (IO), *Address 0x83[0] (Read Only)*

Status of Audio FIFO Near Underflow interrupt signal. When set to 1 it indicates the Audio FIFO is near underflow as the number of FIFO registers containing stereo data is less or equal to value set in AUDIO_FIFO_ALMOST_EMPTY_THRESHOLD. Once set, this bit will remain high until it is cleared via FIFO_NEAR_UFLO_CLR.

Function

FIFO_NEAR_UFLO_RAW	Description
0 «	Audio FIFO has not reached low threshold defined in AUDIO_FIFO_ALMOST_EMPTY_THRESHOLD [5:0]
1	Audio FIFO has reached low threshold defined in AUDIO_FIFO_ALMOST_EMPTY_THRESHOLD [5:0]

FIFO_NEAR_OVFL_RAW, Addr 40 (IO), *Address 0x7E[7] (Read Only)*

Status of Audio FIFO Near Overflow interrupt signal. When set to 1 it indicates the Audio FIFO is near overflow as the number FIFO registers containing stereo data is greater or equal to value set in AUDIO_FIFO_ALMOST_FULL_THRESHOLD. Once set, this bit will remain high until it is cleared via FIFO_NEAR_OVFL_CLR.

Function

FIFO_NEAR_OVFL_RAW	Description
0 «	Audio FIFO has not reached high threshold defined in AUDIO_FIFO_ALMOST_FULL_THRESHOLD [5:0]
1	Audio FIFO has reached high threshold defined in AUDIO_FIFO_ALMOST_FULL_THRESHOLD [5:0]

AUDIO_FIFO_ALMOST_EMPTY_THRESHOLD[6:0], Addr 68 (HDMI), *Address 0x12[6:0]*

Sets the threshold used for FIFO_NEAR_UFLO_RAW. FIFO_NEAR_UFLO_ST interrupt is triggered if audio FIFO goes below this level

Function

AUDIO_FIFO_ALMOST_EMPTY_THRESHOLD[6:0]	Description
0000010 «	Default value
xxxxxxx	Sets threshold used for FIFO_NEAR_UFLO_RAW

AUDIO_FIFO_ALMOST_FULL_THRESHOLD[6:0], Addr 68 (HDMI), *Address 0x11[6:0]*

Sets the threshold used for FIFO_NEAR_OVRFL_RAW. FIFO_NEAR_OVRFL_ST interrupt is triggered if audio FIFO reaches this level

Function

AUDIO_FIFO_ALMOST_FULL_THRESHOLD[6:0]	Description
1111101 «	Default value
xxxxxx	Sets threshold used for FIFO_NEAR_OVFLOW_RAW

8.25 AUDIO PACKET TYPE FLAGS

The ADV7842 can receive and process three types of audio packet:

- Audio sample packets
- DSD packets
- HBR packets

The following flags are provided to monitor the type of audio packets received by the ADV7842. Figure 82 shows the algorithm that can be implemented to monitor the type of audio packet processed by the ADV7842.

AUDIO_MODE_CHNG_RAW, Addr 40 (IO), *Address 0x83[5] (Read Only)*

Status of Audio Mode Change Interrupt signal. When set to 1 it indicates that the type of audio packet received has changed. The following are considered Audio modes, No Audio Packets, Audio Sample Packet, DSD packet or HBR Packet. Once set, this bit will remain high until it is cleared via AUDIO_MODE_CHNG_CLR.

Function

AUDIO_MODE_CHNG_RAW	Description
0 «	Audio mode has not changed.
1	Audio mode has changed.

AUDIO_SAMPLE_PCKT_DET, Addr 68 (HDMI), *Address 0x18[0] (Read Only)*

Audio Sample Packet Detection bit. This bit resets to zero on the 11th HSync leading edge following an Audio packet if a subsequent audio sample packet has not been received or if a DSD or HBR Audio packet sample packet has been received.

Function

AUDIO_SAMPLE_PCKT_DET	Description
0 «	No L_PCM or IEC 61937 compressed audio sample packet received within the last 10 HSync.
1	L_PCM or IEC 61937 compressed audio sample packet received within the last 10 HSyncs.

DSD_PACKET_DET, Addr 68 (HDMI), *Address 0x18[1] (Read Only)*

DSD Audio Packet Detection bit. This bit resets to zero on the 11th HSync leading edge following a DSD packet or if an Audio sample packet or HBR packet has been received or after an HDMI reset condition.

Function

DSD_PACKET_DET	Description
0 «	No DSD packet received within the last 10 HSync.
1	DSD packet received within the last 10 HSync.

HBR_AUDIO_PCKT_DET, Addr 68 (HDMI), *Address 0x18[3] (Read Only)*

HBR Packet detection bit. This bit resets to zero on the 11th HSync leading edge following an HBR packet if a subsequent HBR packet has not been detected. It also resets if an Audio Sample Packet or DSD packet has been received and after an HDMI reset condition

Function

HBR_AUDIO_PCKT_DET	Description
0 «	No HBR audio packet received within the last 10 HSync.
1	HBR audio packet received within the last 10 HSync.

Notes:

- The ADV7842 processes only one type of audio packet at a time.
- The ADV7842 processes the latest type of audio packet that it received.
- A corresponding interrupt can be enabled for AUDIO_MODE_CHNG_RAW by setting the mask AUDIO_MODE_CHNG_MB1 or AUDIO_MODE_CHNG_MB2. Refer to Section 16 for additional information on the interrupt feature.

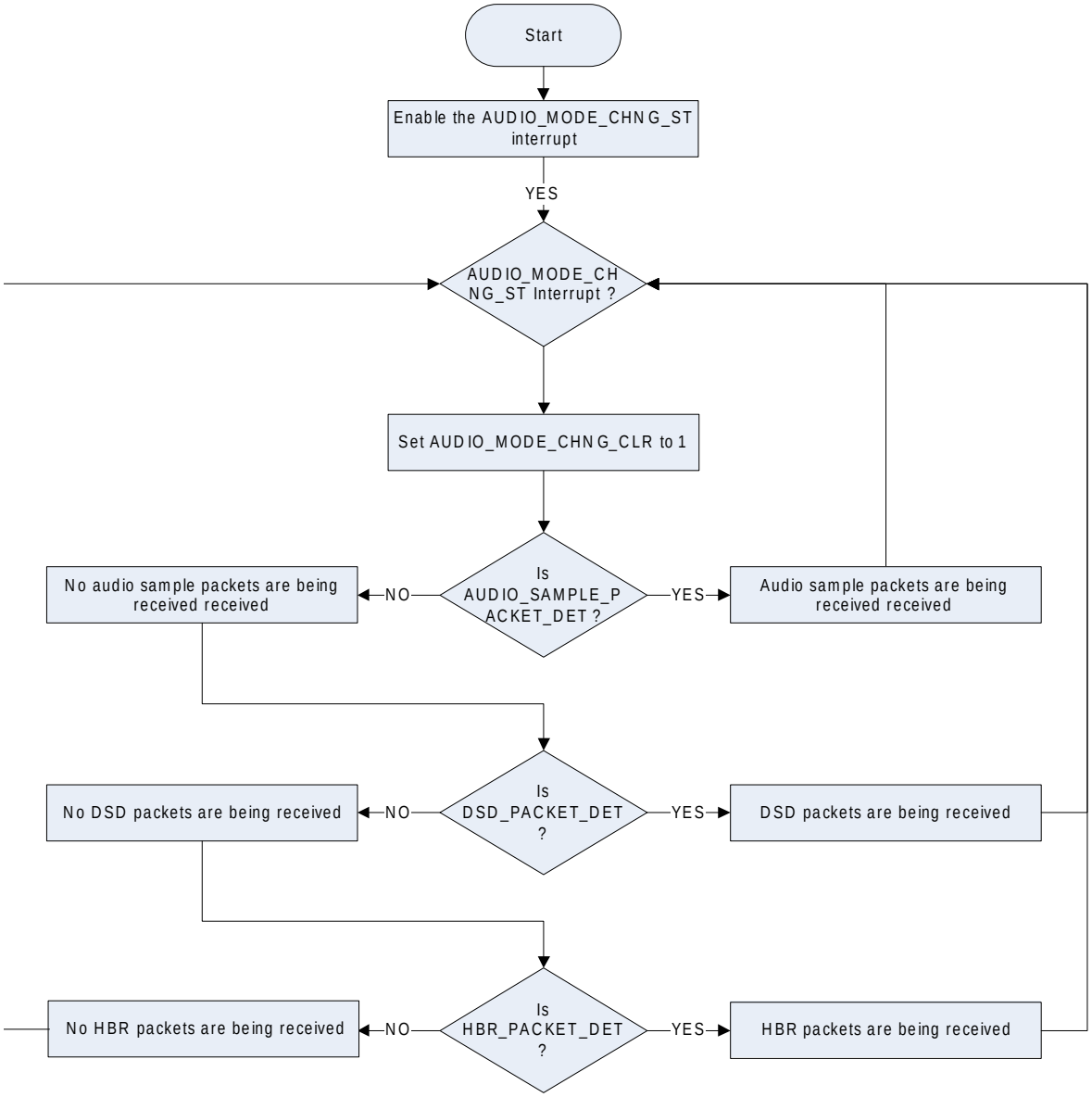


Figure 82: Monitoring Audio Packet Type Processed by ADV7842

8.26 AUDIO OUTPUT INTERFACE

The ADV7842 has a dedicated eight pin audio output interface. The output pin names and descriptions are shown in Table 15.

Table 15: Audio Outputs and Clocks

Output Pixel Port	Description
AP0	Audio Output Port 0
AP1	Audio Output Port 1
AP2	Audio Output Port 2
AP3	Audio Output Port 3
AP4	Audio Output Port 4
AP5	Audio Output Port 5
SCLK	Bit Clock
MCLKOUT	Audio Master Clock

The audio output interface can be adjusted into numerous configurations for the different audio formats. This flexibility helps to increase interconnectivity with downstream audio devices and HDMI transmitters. Table 16 shows the default configurations for the various possible output interfaces.

Table 16: Default Audio Output Pixel Port Mapping

Output Pixel Port	I2S/SPDIF Interface	DSD Interface
AP0	SPDIF0	DSD0A
AP1	I2S0/SDPIF0	DSD0B
AP2	I2S1/SDPIF1	DSD1A
AP3	I2S2/SPDIF2	DSD1B
AP4	I2S3/SPDIF3	DSD2A
AP5	LRCLK	DSD2B

Note: It is possible to tristate the audio pins using the global controls, as described in Section 3.3.6..

8.26.1 I2S/SPDIF Audio Interface and Output Controls

Two controls are provided to change the mapping between the audio output ports and the I2S and SPDIF signals.

I2S_SPDIF_MAP_ROT[1:0], Addr 68 (HDMI), *Address 0x6A[5:4]*

A control to select the arrangement of the I2S/SPDIF interface on the audio output port pins.

Function

I2S_SPDIF_MAP_ROT[1:0]	Description
00 «	[I2S0/SPDIF0 on AP1] [I2S1/SPDIF1 on AP2] [I2S2/SPDIF2 on AP3] [I2S3/SPDIF3 on AP4]
01	[I2S3/SPDIF3 on AP1] [I2S0/SPDIF0 on AP2] [I2S1/SPDIF1 on AP3] [I2S2/SPDIF2 on AP4]
10	[I2S2/SPDIF2 on AP1] [I2S3/SPDIF3 on AP2] [I2S0/SPDIF0 on AP3] [I2S1/SPDIF1 on AP4]
11	[I2S1/SPDIF1 on AP1] [I2S2/SPDIF2 on AP2] [I2S3/SPDIF3 on AP3] [I2S0/SPDIF0 on AP4]

I2S_SPDIF_MAP_INV, Addr 68 (HDMI), *Address 0x6A[6]*

A control to invert the arrangement of the I2S/SPDIF interface on the audio output port pins. Note the arrangement of the I2S/SPDIF interface on the audio output port pins is determined by I2S_SPDIF_MAP_ROT.

Function

I2S_SPDIF_MAP_INV	Description
0 «	Do not invert arrangement of I2S/SPDIF channels in audio output port pins
1	Invert arrangement of I2S/SPDIF channels in audio output port pins

I2S_SPDIF_MAP_ROT and I2S_SPDIF_MAP_INV are independent controls. Any combination of values is therefore allowed for I2S_SPDIF_MAP_ROT and I2S_SPDIF_MAP_INV. Table 17 and Table 18 show examples of mappings for the I2S/SPDIF signals.

Table 17: Audio Mappings for I2S_SPDIF_MAP_ROT = 00, I2S_SPDIF_MAP_INV = 0 (Default)

Output Pixel Port	I2S/SPDIF Interface
AP1	I2S0/SDPIF0
AP2	I2S1/SDPIF1
AP3	I2S2/SDPIF2
AP4	I2S3/SDPIF3

Table 18: Audio Mappings for I2S_SPDIF_MAP_ROT = 00, I2S_SPDIF_MAP_INV = 1

Output Pixel Port	I2S/SPDIF Interface
AP1	I2S3/SDPIF3
AP2	I2S2/SDPIF2
AP3	I2S1/SDPIF1
AP4	I2S0/SDPIF0

I2SBITWIDTH[4:0], Addr 68 (HDMI), *Address 0x03[4:0]*

A control to adjust the bit width for right justified mode on the I2S interface.

Function

I2SBITWIDTH[4:0]	Description
00000	0 bit
00001	1 bit
00010	2 bits
	...
11000 «	24 bits
11110	30 bits
11111	31 bits

I2SOUTMODE[1:0], Addr 68 (HDMI), *Address 0x03[6:5]*

A control to configure the I2S output interface.

Function

I2SOUTMODE[1:0]	Description
00 «	I2S Mode
01	Right Justified
10	Left Justified
11	Raw SPDIF (IEC60958) Mode

Notes:

I2SOUTMODE is effective when the ADV7842 is configured to output I2S streams or AES3 streams. This is the case in the following situation:

- The ADV7842 receives audio sample packets
- The ADV7842 receives HBR packets, OVR_ MUX_HBR is set to 1, and MUX_HBR_OUT is set to 2'b00, 2'b01, 2'b10 or 2'b11
- In HBR mode, it is required that the part outputs four SPDIF, I²S, or raw IEC60958 streams encapsulating a 24-bit audio sample word. Therefore, I2SBITWIDTH should always be set to 0b11000.

The following audio formats can be output when the ADV7842 receives audio sample packets:

- L-PCM audio data is output on the audio output pins if the part received audio sample packets with L-PCM encoded audio data. Each audio output pin carries stereo data that can be output in I²S, right justified, or left justified mode (refer to Figure 83, Figure 84, and Figure 85). The I2SOUTMODE[1:0] control must be set to 0x0, 0x01 or 0x2 to output I²S, right justified, and left justified respectively on the audio output pins.

- A stream conforming to the IEC60958 specification when the part receives audio sample packets with L-PCM encoded data (refer to Figure 86).

- An AES3 stream if the I2SOUTMODE[1:0] control is set to 0x3 (refer to Figure 87 and Figure 88). Note that AES3 is also referred to as raw SPDIF. Each AES3 stream may encapsulate stereo L-PCM audio data or multichannel non L-PCM audio data (e.g. 5.1 Dolby Digital).

- Binary stream on the audio output pins when the part receives audio sample packets with non L-PCM encoded audio data (i.e. AC-3 compressed audio) and if the following configuration is used:
  - I2SOUTMODE must be set to 0x0, 0x01, or 0x2 for I²S, right justified, and left justified format, respectively (refer to Figure 83, Figure 84, and Figure 85)
  - MT_MSK_COMPRS_AUD is set to 0

    Note that no audio flags are output by the part in that configuration. Each binary stream output by the part may encapsulate stereo L-PCM audio data or multichannel non L-PCM audio data (e.g. 5.1 Dolby Digital).

- A stream conforming to the IEC61937 specification when the part receives audio sample packets with non L-PCM encoded audio data (e.g. AC-3 compressed audio). The audio outputs can carry an audio stream that may be stereo or multichannel audio (e.g. 5.1 Dolby Digital).

Table 19: I²S/SPDIF Interface Description

I²S/SPDIF Interface IO	Function
SPDIF0	SPDIF audio output
I2S0/SDPIF0	I²S audio (channel 1, 2) / SPDIF0
I2S1/SDPIF1	I²S audio (channel 3, 4) / SPDIF1
I2S2/SDPIF2	I²S audio (channel 5, 6) / SPDIF2
I2S3/SDPIF3	I²S audio (channel 7, 8) / SPDIF3
SCLK	Bit Clock
LRCLK	Data output clock for left and right channel
MCLKOUT	Audio master clock output

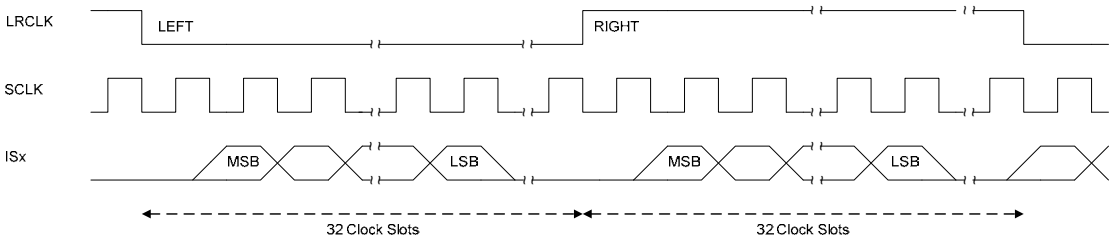


Figure 83: Timing Audio Data Output in I²S Mode

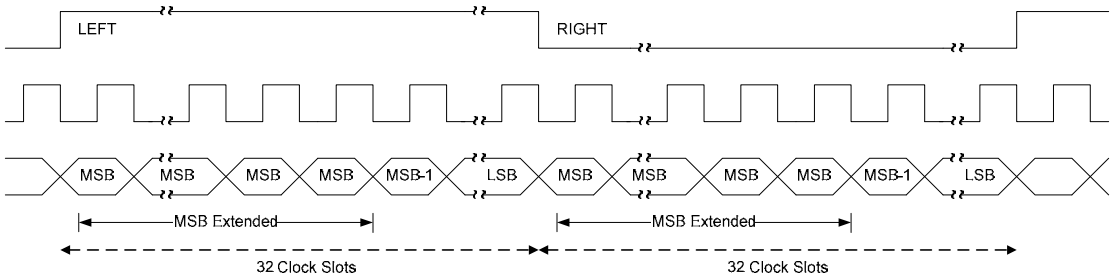


Figure 84: Timing Audio Data Output in Right Justified Mode

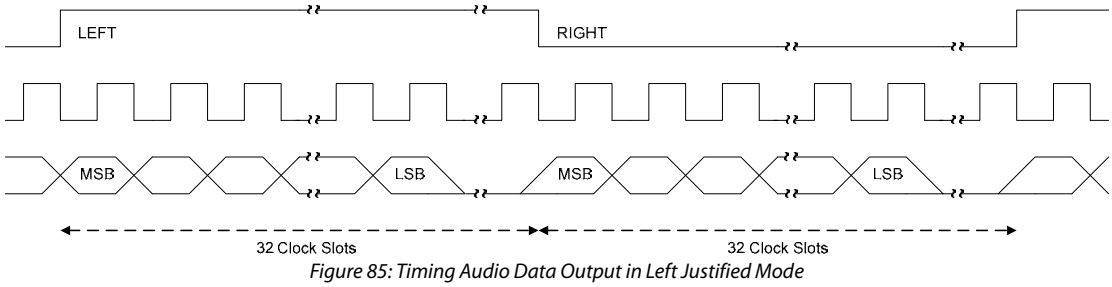


Figure 85: Timing Audio Data Output in Left Justified Mode

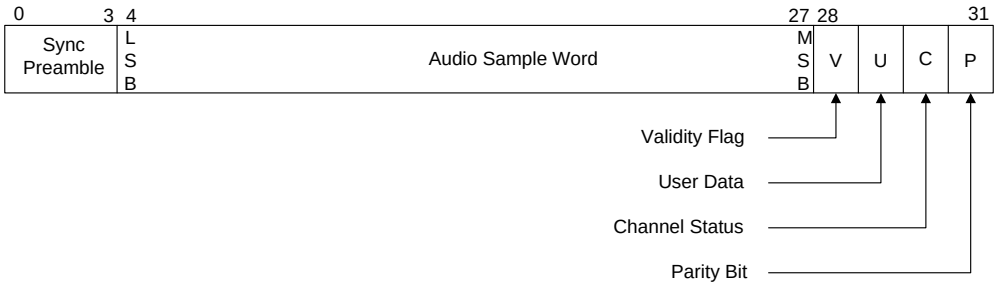


Figure 86: IEC 60958 Sub-frame Timing Diagram

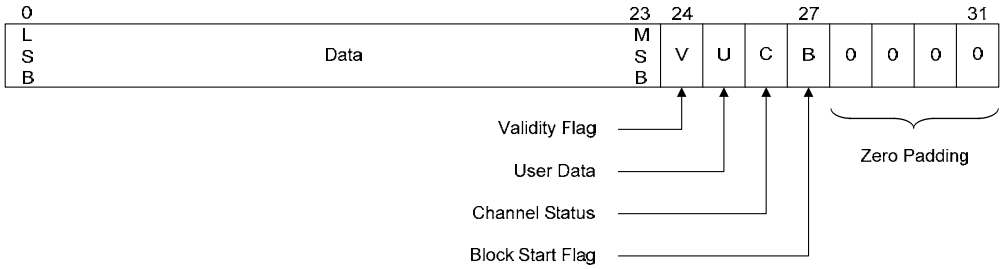


Figure 87: AES3 Sub-frame Timing Diagram

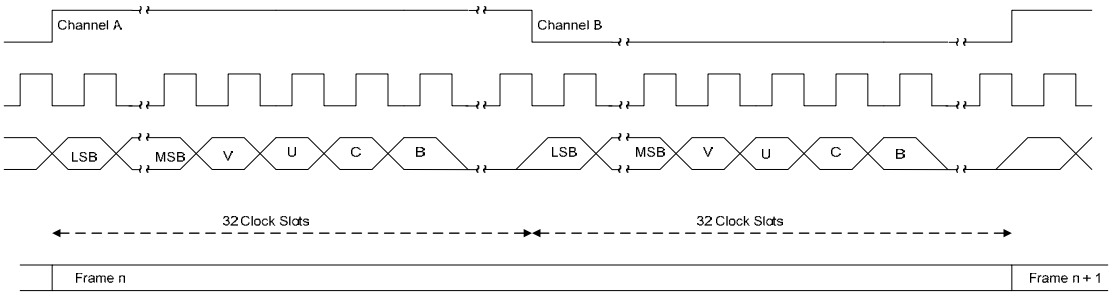


Figure 88: AES3 Stream Timing Diagram

8.26.2 DSD Audio Interface and Output Controls

The ADV7842 incorporates a 6-DSD channel interface used to output the audio stream extracted from DSD packets. Each of the DSD channels carries an over-sampled 1-bit representation of the audio signal as delivered on Super Audio CDs (SACDs).

Table 20: DSD Interface Description

DSD Interface IO	Function
DSD0A	1st DSD data channel
DSD0B	2nd DSD data channel
DSD1A	3rd DSD data channel
DSD1B	4th DSD data channel
DSD2A	5th DSD data channel
DSD2B	6th DSD data channel
SCLK	Bit clock
MCLKOUT	Audio master clock output

Two controls are provided to change the mapping between the audio output ports and DSD signals.

DSD_MAP_ROT[2:0], Addr 68 (HDMI), *Address 0x6A[2:0]*

A control to select the arrangement of the DSD interface on the audio output port pins.

Function

DSD_MAP_ROT[2:0]	Description
000 «	[DSD0A on AP0] [DSD0B on AP1] [DSD1A on AP2] [DSD1B on AP3] [DSD2A on AP4] [DSD2B on AP5]
001	[DSD2B on AP0] [DSD0A on AP1] [DSD0B on AP2] [DSD1A on AP3] [DSD1B on AP4] [DSD2A on AP5]
010	[DSD2A on AP0] [DSD2B on AP1] [DSD0A on AP2] [DSD0B on AP3] [DSD1A on AP4] [DSD1B on AP5]
011	[DSD1B on AP0] [DSD2A on AP1] [DSD2B on AP2] [DSD0A on AP3] [DSD0B on AP4] [DSD1A on AP5]
100	[DSD1A on AP0] [DSD1B on AP1] [DSD2A on AP2] [DSD2B on AP3] [DSD0A on AP4] [DSD0B on AP5]
101	[DSD0B on AP0] [DSD1A on AP1] [DSD1B on AP2] [DSD2A on AP3] [DSD2B on AP4] [DSD0A on AP5]
110	Reserved
111	Reserved

DSD_MAP_INV, Addr 68 (HDMI), *Address 0x6A[3]*

A control to invert the arrangement of the DSD interface on the audio output port pins. Note the arrangement of the DSD interface on the audio output port pins is determined by DSD_MAP_ROT.

Function

DSD_MAP_INV	Description
0 «	Do not invert arrangement of the DSD channels on the audio output port pins
1	Invert arrangement of the DSD channels on the audio output port pins

DSD_MAP_ROT and DSD_MAP_INV are independent controls. Any combination of values is therefore allowed for DSD_MAP_ROT and DSD_MAP_INV. Table 21 and Table 22 show examples of mappings for the DSD signals.

Table 21: Audio Mapping for DSD_MAP_ROT = 00, DSD_MAP_INV = 0 (Default)

Output Pixel Port Name	DSD Interface
AP0	DSD0A
AP1	DSD0B
AP2	DSD1A
AP3	DSD1B
AP4	DSD2A
AP5	DSD2B

Table 22: Audio Mapping for DSD_MAP_ROT = 00, DSD_MAP_INV = 1

Output Pixel Port Name	DSD Interface
AP0	DSD2B
AP1	DSD2A
AP2	DSD1B
AP3	DSD1A
AP4	DSD0B
AP5	DSD0A

Notes:

DSD0A and DSD0B output must be used when in stereo mode only. DSD0A and DSD0B always carry the main 2-channel audio data. DSD1A, DSD1B, DSD2A, and DSD2B are the surround channels.

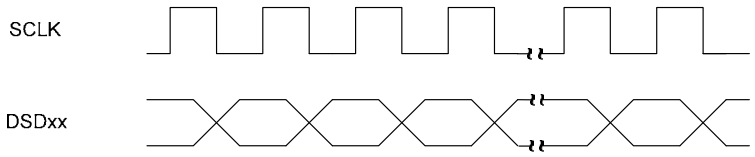


Figure 89: DSD Timing Diagram[1]

[1]Where xx is the channel, for example, 0A, 0B

By default, the ADV7842 automatically enables the DSD interface if it receives DSD packets. The ADV7842 also automatically enables the I2S interface if it receives audio sample packets or if it does not receive any audio packets. However, it is possible to override the audio interface that is used via the OVR_AUTO_MUX_DSD_OUT and MUX_DSD_OUT controls.

OVR_MUX_DSD_OUT, Addr 68 (HDMI), *Address 0x02[1]*

DSD override control. In automatic control, DSD or I2S interface is selected according to the type of packet received DSD or audio sample packet received. I2S interface is enabled when part receives audio sample packets or when no packet is received. DSD interface is enabled when DSD packets are received. In manual mode MUX_DSD_OUT selects the output interface for DSD operation.

Function

OVR_MUX_DSD_OUT	Description
0 «	Automatic DSD output control
1	Manual DSD output control

MUX_DSD_OUT, Addr 68 (HDMI), *Address 0x02[0]*

An override control for the DSD output

Function

MUX_DSD_OUT	Description
0 «	Output I2S data in DSD mode
1	Output DSD data in DSD Mode

8.26.3 HBR Interface and Output Controls

The ADV7842 can receive HBR audio stream packets. The ADV7842 outputs HBR data over five of the audio output pins in any of the following formats:

- An SDPIF stream conforming to the IEC60958 specification (refer to Figure 86). The following configuration is required to output an SPDIF stream on the HBR output pins:
  - OVR_MUX_HBR is set to 0
    or
  - OVR_ MUX_HBR is set to 1 and MUX_HBR_OUT is set to 1

- A binary stream if one of the following configurations is used:
  - OVR_ MUX_HBR is set to 1, MUX_HBR_OUT is set to 0, and I2SOUTMODE[1:0] is set to 0x0 for an I^2S mode binary stream (refer to Figure 83).
  - OVR_ MUX_HBR is set to 1, MUX_HBR_OUT is set to 0, and I2SOUTMODE is set to 0x1 for a right justified stream (refer to Figure 84)
  - OVR_ MUX_HBR is set to 1, MUX_HBR_OUT is set to 0, and I2SOUTMODE is set to 0x2 for a left justified stream (refer to Figure 85)

  Note: No audio flags are output by the part in these configuration.

- An AES3 stream on each HBR interface output pin (refer to Figure 87 and Figure 88). The following configuration is required to output AES3 streams:
  - OVR_ MUX_HBR is set to 1
  - I2SOUTMODE is set to 0b11

Important:

- Each of the four HBR outputs carry one of four consecutive blocks of the HBR stream
- The four streams on the four HBR pin are output at one quarter of the audio sample rate, fs

Table 23: HBR Interface Description

HBR Interface IO	Function
AP0	1st block of HBR stream (SPDIF Format Only)
AP1	1st block of HBR stream
AP2	2nd block of HBR stream
AP3	3rd block of HBR stream
AP4	4th block of HBR stream
SCLK	Bit clock
LRCLK	Data output clock for left and right channel
MCLKOUT	Audio master clock output

Note: The audio output mapping controls: I2S_SPDIF_MAP_ROT[1:0] and I2S_SPDIF_MAP_INV also apply to the HBR output signals.

OVR_MUX_HBR, Addr 68 (HDMI), *Address 0x01[1]*

A control to select automatic or manual configuration for HBR outputs. Automatically, HBR outputs are encoded as SPDIF streams. In manual mode MUX_HBR_OUT selects the audio output interface.

Function

OVR_MUX_HBR	Description
0 «	Automatic HBR output control
1	Manual HBR output control

MUX_HBR_OUT, Addr 68 (HDMI), *Address 0x01[0]*

A control to manually select the audio output interface for HBR data. Valid when OVR_MUX_HBR is set to 1.

Function

MUX_HBR_OUT	Description
0 «	Override by outputting I2S data
1	Override by outputting SPDIF data

8.27 MCLKOUT SETTING

The audio master clock MCLKOUT is set using the MCLK_FS_N[2:0] register, as shown in Equation 7.

$$MCLKOUT = (MCLKFS\_N + 1) \times 128 \times f_s$$

Equation 7: Relationship Between MCLKOUT, MCLKFS_N, and f_s

MCLK_FS_N[2:0], Addr 4C (DPLL), *Address 0xB5[2:0]*

Selects the multiple of 128fs used for MCLK out.

Function

MCLK_FS_N[2:0]	Description
000	128fs
001 «	256fs
010	384fs
011	512fs
100	640fs
101	768fs
110	Not Valid
111	Not Valid

8.28 AUDIO CHANNEL MODE

AUDIO_CH_MD_RAW indicates if 2-channel audio data or multi-channel audio data is received.

AUDIO_CH_MD_RAW, Addr 40 (IO), *Address 0x65[4] (Read Only)*

Raw status signal indicating the layout value of the audio packets that were last received

Function

AUDIO_CH_MD_RAW	Description
0 «	The last audio packets received have a layout value of 1. (e.g. Layout-1 corresponds to 2-channel audio when Audio Sample packets are received).
1	The last audio packets received have a layout value of 0 (e.g. Layout-0 corresponds to 8-channel audio when Audio Sample packets are received).

Note: The Audio CH_MD_RAW flag is valid for audio sample packets and DSD packets.

AUDIO_CHANNEL_MODE, Addr 68 (HDMI), *Address 0x07[6] (Read Only)*

Flags stereo or multichannel audio packets. Note stereo packets may carry compressed multi-channel audio.

Function

AUDIO_CHANNEL_MODE	Description
0 «	Stereo Audio (may be compressed multichannel)
1	Multichannel uncompressed audio detected (3-8 channels).

8.29 AUDIO MUTING

The ADV7842 integrates an advanced audio mute function that is designed to remove all extraneous noise and pops from a 2-channel L-PCM audio stream at sample frequencies up to 192 kHz.

The hardware for audio mute function is composed of the following three blocks:

- Audio delay line that delays channels 1 and 2 by 512 stereo samples.
- Audio mute controller takes in event detection signals that can be used to determine when an audio mute is needed. The controller generates a mute signal to the ramped audio block and a coast signal to the digital PLL generating the audio clock.
- Ramped audio mute block that can mute the audio over the course of 512 stereo samples.

Notes:

- The ADV7842 mutes only the non compressed data from the audio sample packets output through the I^2S and the SPDIF interface.

- The audio delay line is automatically bypassed when the ADV7842 receives multichannel audio or when it receives the following audio packets:
  - DSD packets
  - HBR packets

- The ramped audio mute block is always bypassed when the part received compressed audio or when it received the following audio packets:
  - DSD packets
  - HBR packets

8.29.1 Delay Line Control

The audio delay line should be enabled when the ADV7842 is configured for automatic mute. The audio delay line is controlled by the

MAN_AUDIO_DL_BYPASS and AUDIO_DELAY_LINE_BYPASS bits.

MAN_AUDIO_DL_BYPASS, Addr 68 (HDMI), *Address 0x0F[7]*

Audio Delay Bypass Manual Enable. The audio delay line is automatically active for stereo samples and bypassed for multichannel samples. By setting MAN_AUDIO_DL_BYPASS to 1 the Audio delay bypass configuration can be set by the user with the AUDIO_DELAY_LINE_BYPASS control.

Function

MAN_AUDIO_DL_BYPASS	Description
0 «	Audio delay line is automatically bypassed if multichannel audio is received. The audio delay line is automatically enabled if stereo audio is received.
1	Overrides automatic bypass of audio delay line. Audio delay line is applied depending on the AUDIO_DELAY_LINE_BYPASS control.

AUDIO_DELAY_LINE_BYPASS, Addr 68 (HDMI), *Address 0x0F[6]*

Manual bypass control for the audio delay line. Only valid if MAN_AUDIO_DL_BYPASS is set to 1.

Function

AUDIO_DELAY_LINE_BYPASS	Description
0 «	Enables the audio delay line.
1	Bypasses the audio delay line.

8.29.2 Audio Mute Configuration

The ADV7842 can be configured to automatically mute an L-PCM audio stream when selectable mute conditions occur. The audio muting is configured as follows:

- Set the audio muting speed via AUDIO_MUTE_SPEED[4:0].
- Set NOT_AUTO_UNMUTE, as follows:
  - Set AUDIO_UNMUTE[2:0] to 0 if the audio must be unmuted automatically after a delay set in WAIT_UNMUTE[2:0] after all selected mute conditions have become inactive.
  - Set NOT_AUTO_UNMUTE to 1 if the audio must be unmuted manually (e.g. by an external controller) when all selected mute conditions have become inactive.
  - Select the mute conditions that trigger an audio mute (refer to Table 24).
- Select the Audio PLL coast conditions (refer to Section 8.23.4).
- Set WAIT_UNMUTE[2:0] to configure the audio counter that triggers the audio unmute when it has timed out after all selected mute conditions have become inactive.

The ADV7842 internally unmutes the audio if the following three conditions (listed in order of priority) are met:

1. Mute conditions are inactive
2. NOT_AUTO_UNMUTE is set to 0
3. Audio unmute counter has finished counting down or is disabled

Notes:

- Both Table 14 and Table 24 provide a column with the heading 'Corresponding Status Registers(s)'. This column lists the status registers that convey information related to their corresponding audio mute masks or coast masks.

- The ADV7842 also mutes the DSD stream when one of the selected mute conditions occurs (refer to Table 24) by outputting the DSD mute pattern 0101010101… A DSD decoder receiving this stream outputs a 0 V mean analog stream.
- The ADV7842 never mutes the audio data when it receives an audio sample packet with compressed audio data or HBR packets.
- For the best audio muting performance, the following setting is recommended when the ADV7842 receives multichannel sample packets:
  - Set AUDIO_MUTE_SPEED to 1
- For best audio muting performance, the following settings are recommended when the audio sampling frequency of the audio stream is greater than 48 kHz:
  - Set AUDIO_MUTE_SPEED to 1
  - Set MAN_AUDIO_DL_BYPASS to 1
  - Set AUDIO_DELAY_LINE_BYPASS to 1
- For best audio muting performance, the following settings are recommended when the audio sampling frequency of the audio stream is equal to or lower than 48 kHz:
  - Set AUDIO_MUTE_SPEED to 0x1F
  - Set MAN_AUDIO_DL_BYPASS to 0

MUTE_AUDIO, Addr 68 (HDMI), *Address 0x1A[4]*

A control to force an internal mute independently of the mute mask conditions

Function

MUTE_AUDIO	Description
0 «	Audio in normal operation
1	Force audio mute

AUDIO_MUTE_SPEED[4:0], Addr 68 (HDMI), *Address 0x0F[4:0]*

Number of samples between each volume change of 1.5dB when muting and unmuting

Function

AUDIO_MUTE_SPEED[4:0]	Description
11111 «	31 samples between each volume change of 1.5dB when muting and unmuting
xxxxx	Number of samples between each volume change of 1.5dB when muting and unmuting

__ENDFIELD__

NOT_AUTO_UNMUTE, Addr 68 (HDMI), *Address 0x1A[0]*

A control to disable the auto unmute feature. When set to 1 audio can be unmuted manually if all mute conditions are inactive by setting NOT_AUTO_UNMUTE to 0 and then back to 1.

Function

NOT_AUTO_UNMUTE	Description
0 «	Audio unmutes following a delay set by WAIT_UNMUTE after all mute conditions have become inactive.
1	Prevents audio from unmuting automatically

WAIT_UNMUTE[2:0], Addr 68 (HDMI), *Address 0x1A[3:1]*

A control to delay audio unmute. Once all mute conditions are inactive WAIT_UNMUTE[2:0] can specify a further delay time before unmuting. NOT_AUTO_UNMUTE must be set to 0 for this control to be effective.

Function

WAIT_UNMUTE[2:0]	Description
000 «	0ms Disables/cancels delayed unmute. Audio unmutes directly after all mute conditions become inactive
001	Unmutes 10 ms after all mute conditions become inactive
010	Unmutes 25 ms after all mute conditions become inactive
011	Unmutes 50 ms after all mute conditions become inactive
100	Unmutes 75 ms after all mute conditions become inactive
101	Unmutes 100 ms after all mute conditions become inactive
110	Unmutes 250 ms after all mute conditions become inactive
111	Unmutes 1000 ms (1s) after all mute conditions become inactive

Table 24: Selectable Mute Conditions

Bit Name	HDMI Map Address	Description	Corresponding Status Registers(s)
MT_MSK_COMPRS_AUD	0x14[5]	Causes audio mute if audio is compressed	CS_DATA[1]
MT_MSK_ AUD_MODE_CHNG	0x14[4]	Causes audio mute if audio mode changes between PCM, DSD, or HBR formats	AUDIO_SAMPLE_PCKT_DET
MT_MSK_PARITY_ERR	0x14[1]	Causes audio mute if parity bits in audio samples are not correct	PARITY_ERROR_RAW
MT_MSK_VCLK_CHNG	0x14[0]	Causes audio mute if TMDS clock has irregular/missing pulses	VCLK_CHNG_RAW
MT_MSK_APLL_UNLOCK	0x15[7]	Causes audio mute if audio PLL unlocks	AUDIO_PLL_LOCKED
MT_MSK_VPLL_UNLOCK	0x15[6]	Causes audio mute if TMDS PLL unlocks	TMDS_PLL_LOCKED
MT_MSK_ACR_NOT_DET	0x15[5]	Causes audio mute if ACR packets are not received within one VSync	AUDIO_C_PCKT_RAW
MT_MSK_FLATLINE_DET	0x15[3]	Causes audio mute if flatline bit in audio packets is set	AUDIO_FLT_LINE_RAW
MT_MSK_FIFO_UNDERFLOW	0x15[1]	Causes audio mute if audio FIFO underflows	FIFO_UNDERFLO_RAW
MT_MSK_FIFO_OVERFLOW	0x15[0]	Causes audio mute if audio FIFO overflows	FIFO_OVERFLO_RAW
MT_MSK_AVMUTE	0x16[7]	Causes audio mute if AVMute is set in the general control packet	AV_MUTE_RAW
MT_MSK_NOT_HDMIMODE	0x16[6]	Causes audio mute if HDMI_MODE bit goes low	HDMI_MODE
MT_MSK_NEW_CTS	0x16[5]	Causes audio mute if CTS changes by more than the threshold set in CTS_CHANGE_THRESHOLD[5:0]	CTS_PASS_THRSH_RAW
MT_MSK_NEW_N	0x16[4]	Causes audio mute if N changes	CHANGE_N_RAW
MT_MSK_CHMODE_CHNG	0x16[3]	Causes audio mute if the channel mode changes from stereo to multichannel, or visa versa	AUDIO_MODE_C HNG_RAW
MT_MSK_APCKT_ECC_ERR	0x16[2]	Causes audio mute if uncorrectable error is detected in the audio packets by the ECC block	AUDIO_PCKT_ERR_RAW
MT_MSK_CHNG_PORT	0x16[1]	Causes audio mute if HDMI port is changed	HDMI_PORT_SELECT[1:0]
MT_MSK_VCLK_DET	0x16[0]	Causes audio mute if TMDS clock is not detected	TMDS_CLK_A_RAW TMDS_CLK_B_RAW

8.29.3 Internal Mute Status

The internal mute status is provided through the INTERNAL_MUTE_RAW bit.

INTERNAL_MUTE_RAW, Addr 40 (IO), *Address 0x65[6] (Read Only)*

Raw status signal of Internal Mute signal.

Function

INTERNAL_MUTE_RAW	Description
0 «	Audio is not muted
1	Audio is muted

8.29.4 AV Mute Status

AV_MUTE, Addr 68 (HDMI), *Address 0x04[6] (Read Only)*

Readback of AVMUTE status received in the last General Control packet received.

Function

AV_MUTE	Description
0 «	AVMUTE not set
1	AVMUTE set

8.29.5 Audio Mute Signal

The ADV7842 can output an audio mute signal that can be used to control the muting in a back end audio device processing the audio data output by the ADV7842 (e.g. DSP).

The audio mute signal is output on the INT1 pin by setting EN_MUTE_OUT_INTRQ to 1. The active level of the mute signal on INT1 is set via the INTRQ_OP_SEL control.

The audio mute signal can also be output on the INT2 pin by setting INT2_EN to 1 and EN_MUTE_OUT_INTRQ2 to 1. The active level of the mute signal output on the INT2 pin is set via INTRQ2_OP_SEL.

Important:

The ADV7842 may interface with an audio processor (e.g. DSP) in which the muting of the audio is implemented. In this case, the audio processor typically features a delay line followed by a mute block for audio mute and unmuting purposes. The following hardware and software configuration is recommended for optimum muting performance of the ADV7842 and audio processor system:

- Connect the mute signal of the ADV7842 to the audio processor mute input. The ADV7842 mute signal can now drive the muting/unmuting of the audio data inside the audio processor.
- Bypass the audio delay line of the ADV7842 with the following settings:
  - Set MAN_AUDIO_DL_BYPASS to 1
  - Set AUDIO_DELAY_LINE_BYPASS to 1
  - Configure the ADV7842 to mute the audio over one audio sample clock as follows:
    - Set AUDIO_MUTE_SPEED[4:0] to 1. This ensures that the ADV7842 never outputs invalid audio data out

to the audio processor.

EN_MUTE_OUT_INTRQ, Addr 40 (IO), *Address 0x40[3]*

A control to apply the audio mute signal on INT1 interrupt pin.

Function

EN_MUTE_OUT_INTRQ	Description
0 «	Does not output audio mute signal on INT1
1	Outputs audio mute signal on INT1

EN_MUTE_OUT_INTRQ2, Addr 40 (IO), *Address 0x41[3]*

A control to apply the internal audio mute signal on INT2 interrupt pin.

Function

EN_MUTE_OUT_INTRQ2	Description
0 «	Does not output audio mute signal on INT2
1	Outputs audio mute signal on INT2

8.29.6 Audio Stream with Incorrect Parity Error

The ADV7842 discards audio sample packets that have an incorrect parity bit. When these samples are received, the ADV7842 repeats the previous audio sample with a valid parity bit. The audio stream out of the ADV7842 can be muted in this situation if the audio mute mask MT_MSK_PARITY_ERR is set.

It is possible to configure the ADV7842 so that it processes audio sample packets that have an incorrect parity bit and corrects the parity bit. The ADV7842 can then output an audio stream even when the parity bits from the audio sample packet are invalid. This configuration is activated by setting MT_MSK_PARITY_ERR 0 and IGNORE_PARITY_ERR to 1.

IGNORE_PARITY_ERR, Addr 68 (HDMI), *Address 0x1A[6]*

A control to select the processing of audio samples even when they have a parity error.

Function

IGNORE_PARITY_ERR	Description
0 «	Discard audio sample packet that have an invalid parity bit.
1	Process audio sample packets that have an invalid parity bit.

MT_MSK_PARITY_ERR, Addr 68 (HDMI), *Address 0x14[1]*

Audio Mute Mask for a parity error. It sets the audio mutes if an audio sample packet is received with an incorrect parity bit.

Function

MT_MSK_PARITY_ERR	Description
1 «	Audio mute occurs if an audio sample packet is received with an incorrect parity bit.

8.30 AUDIO CLOCK REGENERATION PARAMETERS

The ADV7842 recreates an internal audio master clock using Audio Clock Regeneration (ACR) values transmitted by the HDMI source.

8.30.1 ACR Parameters Readbacks

The registers N and CTS can be read back from the HDMI Map.

CTS[19:0], Addr 68 (HDMI), *Address 0x5B[7:0]; Address 0x5C[7:0]; Address 0x5D[7:4] (Read Only)*

A readback for the CTS value received in the HDMI datastream.

Function

CTS[19:0]	Description
00000000000000000000 «	Default CTS value readback from HDMI stream
xxxxxxxxxxxxxxxxxxxx	CTS value readback from HDMI stream

N[19:0], Addr 68 (HDMI), *Address 0x5D[3:0]; Address 0x5E[7:0]; Address 0x5F[7:0] (Read Only)*

A readback for the N value received in the HDMI datastream

Function

N[19:0]	Description
00000000000000000000 «	Default N value readback from HDMI stream
xxxxxxxxxxxxxxxxxxxx	N value readback from HDMI stream

Note: A buffer has been implemented for the N and CTS readback registers. A read of the HDMI Map, Address 0x5B register updates the buffer that stores the N and CTS readback registers. The buffer implemented for N and CTS readback allows the reading of both N and CTS registers within an I^2C block read.

8.30.2 Monitoring ACR Parameters

The reception of ACR packets can be notified via the AUDIO_C_PCKT_RAW flag. Changes in N and CTS can be monitored via the CHANGE_N_RAW and CTS_PASS_THRSH_RAW flags, as described in this section.

AUDIO_C_PCKT_RAW, Addr 40 (IO), *Address 0x65[1] (Read Only)*

Raw status signal of Audio Clock Regeneration Packet detection signal.

Function

AUDIO_C_PCKT_RAW	Description
0 «	No audio clock regeneration packets received since the last HDMI reset condition
1	Audio clock regeneration packets received

CHANGE_N_RAW, Addr 40 (IO), *Address 0x7E[3] (Read Only)*

Status of the ACR N Value changed interrupt signal. When set to 1 it indicates the N Value of the ACR packets has changed. Once set, this bit will remain high until it is cleared via CHANGE_N_CLR.

Function

CHANGE_N_RAW	Description
0 «	Audio clock regeneration N value has not changed
1	Audio clock regeneration N value has changed

CTS_PASS_THRSH_RAW, Addr 40 (IO), *Address 0x7E[4] (Read Only)*

Status of the ACR CTS value exceed threshold interrupt signal. When set to 1 it indicates the CTS Value of the ACR packets has exceeded the threshold set by CTS_CHANGE_THRESHOLD. Once set, this bit will remain high until it is cleared via CTS_PASS_THRSH_CLR.

Function

CTS_PASS_THRSH_RAW	Description
0 «	Audio clock regeneration CTS value has not passed the threshold
1	Audio clock regeneration CTS value has changed more than threshold

CTS_CHANGE_THRESHOLD[5:0], Addr 68 (HDMI), *Address 0x10[5:0]*

Sets the tolerance for change in the CTS value. This tolerance is used for the audio mute mask MT_MSK_NEW_CTS and the HDMI status bit CTS_PASS_THRSH_RAW and the HDMI interrupt status bit CTS_PASS_THRSH_ST. This register controls the amounts of LSBs that the CTS can change before an audio mute, status change or interrupt is triggered.

Function

CTS_CHANGE_THRESHOLD[5:0]	Description
100101 «	Tolerance of CTS value for CTS_PASS_THRSH_RAW and MT_MSK_NEW_CTS
xxxxxx	Tolerance of CTS value for CTS_PASS_THRSH_RAW and MT_MSK_NEW_CTS

8.31 CHANNEL STATUS

Channel status bits are extracted from the HDMI audio packets of the 1st audio channel (i.e. channel 0) and stored in registers CHANNEL_STATUS_DATA_X of the HDMI Map (where X = 1, 2, 3, 4 and 5).

8.31.1 Validity Status Flag

The channel status readback described in Section 8.31 should be considered valid if CS_DATA_VALID_RAW is set to 1. Figure 90 shows the algorithm that can be implemented to monitor the read valid channel status bit using the CS_DATA_VALID_RAW flag.

CS_DATA_VALID_RAW, Addr 40 (IO), *Address 0x65[7] (Read Only)*

Raw status signal of Channel Status Data Valid signal.

Function

CS_DATA_VALID_RAW	Description
0 «	Channel status data is not valid
1	Channel status data is valid

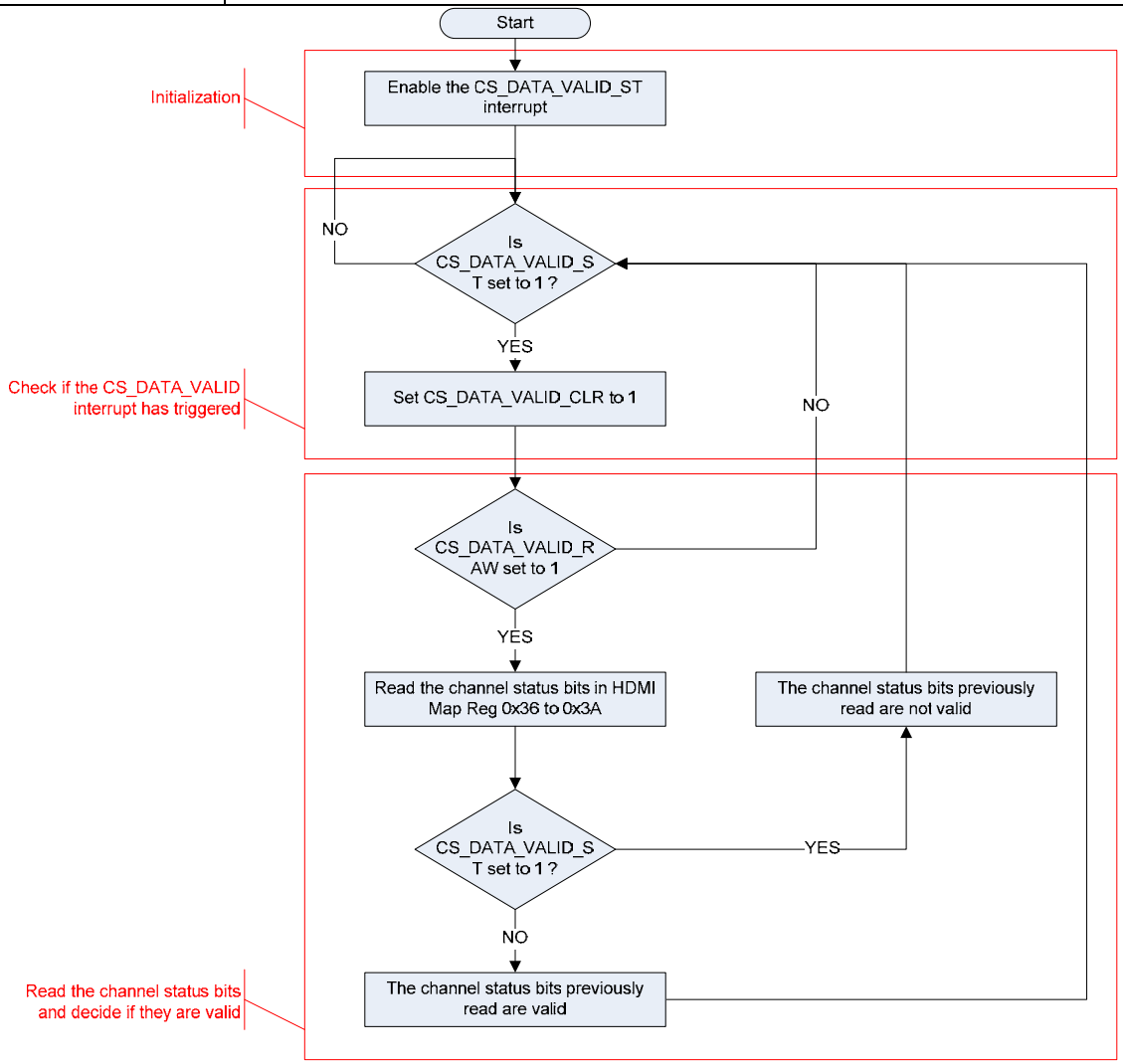


Figure 90: Reading Valid Channel Status Flags

Notes:

- CS_DATA_VALID_RAW indicates that the first 40 of the channel status bits sent by the upstream transmitter have been correctly collected. This bit does not indicate if the content of the channel status bit is corrupted as this is indeterminable.
- A corresponding interrupt can be enabled for CS_DATA_VALID_RAW by setting the mask CS_DATA_VALID_MB1 or CS_DATA_VALID_MB2. Refer to section 16 for additional information on the interrupt feature.

8.31.2 General Control and Mode Information

The general control and mode information are specified in Byte 0 of the channel status. For more information, refer to the IEC60958 standards.

CS_DATA[0], Consumer/Professional Application, HDMI Map, *Address 0x36, [0]*

Function

CS_DATA[0]	Description
0 «	Consumer application
1	Professional application

CS_DATA[1], PCM/non-PCM Audio Sample, HDMI Map, *Address 0x36, [1]*

Function

CS_DATA[1]	Description
0 «	Audio sample word represents linear PCM samples
1	Audio sample word used for other purposes

CS_DATA[2], Copyright, HDMI Map, *Address 0x36, [2]*

Function

CS_DATA[2]	Description
0 «	Software for which copyright is asserted
1	Software for which no copyright is asserted

CS_DATA[5:3], Emphasis, HDMI Map, *Address 0x36, [5:3]*

Function

CS_DATA[5:3][1]	Description
000 «	Two audio channels without pre-emphasis
001	Two audio channels with 50/15 pre-emphasis

[1] Unspecified values are reserved

CS_DATA[7:6], Channel Status Mode, HDMI Map, *Address 0x36, [7:6]*

Function

CS_DATA[7:6][1]	Description
00 «	Mode 0

[1]Unspecified values are reserved

8.31.3 Category Code

The category code is specified in Byte 1 of the channel status. The category code indicates the type of equipment that generates the digital audio interface signal. For more information, refer to the IEC60958 standards.

CS_DATA[15:8], Category Code, HDMI Map, *Address 0x37, [7:0]*

Function

CS_DATA[15:8]	Description
xxxx xxxx	Category code[1]
0000 0000 «	Reset value

[1]Refer to IEC60958-3 standards

8.31.4 Source Number and Channel Number

CS_DATA[19:16], Source Number, HDMI Map, *Address 0x38, [3:0]*

Function

CS_DATA[19:16]	Description
xxxx	Source number[1]
0000 «	Reset value

[1]Refer to IEC60958-3 standards

CS_DATA[23:20], Channel Number, HDMI Map, *Address 0x38, [7:4]*

Function

CS_DATA[23:20]	Description
xxxxx	Channel number[1]
00000 «	Reset value

[1]Refer to IEC60958-3 standards

8.31.5 Sampling and Frequency Accuracy

The sampling frequency and clock accuracy are specified by Byte 3 of the channel status. For additional information, refer to the IEC60958 standards.

CS_DATA[27:24], Sampling Frequency, HDMI Map, *Address 0x39, [3:0]*

Function

CS_DATA[27:24][1]	Description
0000 «	44.1 kHz
0010	48 kHz
0011	32 kHz
1000	88.2 kHz
1010	96 kHz
1100	176 kHz
1110	192 kHz

[1] Unspecified values are reserved

CS_DATA[29:28], Clock Accuracy, HDMI Map, *Address 0x39, [5:4]*

Function

CS_DATA[29:28]	Description
00 «	Level II, ±1000 ppm
01	Level I, ±50 ppm
10	Level III, variable pitch shifted
11	Reserved

CS_DATA[31:30], Reserved Register, HDMI Map, *Address 0x39, [7:6]*

Function

CS_DATA[31:30]	Description
XX	Reserved
00 «	Reset value

8.31.6 Word Length

Word length information is specified in Byte 4 of the channel status bit. For more information, refer to the IEC60958 standards.

CS_DATA[32], Maximum Word Length Size, HDMI Map, *Address 0x3A, [0]*

Function

CS_DATA[32]	Description
0 «	Maximum audio sample word length is 20 bits
1	Maximum audio sample word length is 24 bits

CS_DATA[35:33], Word Length, HDMI Map, *Address 0x3A, [3:1]*

Function

CS_DATA[35:33][1]	Description	
	Audio sample word length if maximum length is 24 as indicated by CS_DATA_[32]	Audio sample word length if maximum length is 20 as indicated by CS_DATA_[32]
000 «	Word length not indicated	Word length not indicated
001	20 bits	16 bits
010	22 bits	18 bits
100	23 bits	19 bits
101	24 bits	20 bits
110	21 bits	21bits

[1]Unspecified values are reserved

8.31.7 Channel Status Copyright Value Assertion

It is possible to overwrite the copyright value of the channel status bit that is passed to the SPDIF output. This is done via the CS_COPYRIGHT_MANUAL and CS_COPYRIGHT_VALUE controls.

CS_COPYRIGHT_MANUAL, Addr 68 (HDMI), *Address 0x50[1]*

A control to select automatic or manual setting of the copyright value of the channel status bit that is passed to the SPDIF output. Manual control is set with the CS_COPYRIGHT_VALUE bit.

Function

CS_COPYRIGHT_MANUAL	Description
0 «	Automatic CS copyright control
1	Manual CS copyright control. Manual value is set by CS_COPYRIGHT_VALUE

CS_COPYRIGHT_VALUE, Addr 68 (HDMI), *Address 0x50[0]*

A control to set the CS Copyright value when in manual configuration of the CS Copyright bit that is passed to the SPDIF output.

Function

CS_COPYRIGHT_VALUE	Description
0 «	Copyright value of channel status bit is 0. Valid only if CS_COPYRIGHT_MANUAL is set to 1
1	Copyright value of channel status bit is 1. Valid only if CS_COPYRIGHT_MANUAL is set to 1

8.31.8 Monitoring Change of Audio Sampling Frequency

The ADV7842 features the NEW_SAMP_RT_RAW flag to monitor changes in the audio sampling frequency field of the channel status bits.

NEW_SAMP_RT_RAW, Addr 40 (IO), *Address 0x83[3] (Read Only)*

Status of new sampling rate interrupt signal. When set to 1 it indicates that audio sampling frequency field in channel status data has changed. Once set, this bit will remain high until it is cleared via NEW_SAMP_RT _CLR.

Function

NEW_SAMP_RT_RAW	Description
0 «	Sampling rate bits of the channel status data on audio channel 0 have not changed
1	Sampling rate bits of the channel status data on audio channel 0 have changed

Important: The NEW_SAMP_RT_RAW flag does not trigger if CS_DATA_VALID_RAW is set to 0. This prevents the notification of a change from a valid to an invalid audio sampling frequency readback in the channel status bits, and vice versa.

8.32 PACKETS AND INFOFRAMES REGISTERS

In HDMI, auxiliary data is carried across the digital link using a series of packets. The ADV7842 automatically detects and stores the following HDMI packets:

- InfoFrames
- Audio Content Protection (ACP)
- International Standard Recording Code (ISRC)
- Gamut Metadata

When the ADV7842 receives one of these packets, it computes the packet checksum and compares it with the checksum available in the packet. If these checksums are the same, the packets are stored in the corresponding registers. If the checksums are not the same, the packets are discarded. Refer to the EIA/CEA-861D specifications for more information on the packets fields.

8.32.1 InfoFrames Registers

The ADV7842 can store the following InfoFrames:

- Auxiliary Video Information (AVI) InfoFrame
- Source Production Descriptor (SPD) InfoFrame
- Audio InfoFrame

- Moving Picture Expert Group (MPEG) Source InfoFrame

8.32.2 InfoFrame Collection Mode

The ADV7842 has two modes for storing the InfoFrame packet sent from the source into the internal memory. By default, the ADV7842 only stores the InfoFrame packets received if the checksum is correct for each InfoFrame.

The ADV7842 also provides a mode to store every InfoFrame sent from the source, regardless of a InfoFrame packet checksum error.

ALWAYS_STORE_INF, Addr 68 (HDMI), *Address 0x47[0]*

A control to force InfoFrames with checksum errors to be stored.

Function

ALWAYS_STORE_INF	Description
0 «	Stores data from received InfoFrames only if their checksum is correct
1	Always store the data from received InfoFrame regardless of their checksum

8.32.3 InfoFrame Checksum Error Flags

The following checksum error status registers flag when the last InfoFrame received has a checksum error. Once set these bits will remain high until the interrupt has been cleared via their corresponding clear bits.

AVI_INF_CKS_ERR_RAW, Addr 40 (IO), *Address 0x88[4] (Read Only)*

Status of AVI Infoframe Checksum Error interrupt signal. When set to 1 it indicates that a checksum error has been detected for an AVI InfoFrame. Once set, this bit will remain high until it is cleared via AVI_INF_CKS_ERR_CLR.

Function

AVI_INF_CKS_ERR_RAW	Description
0 «	No AVI infoframe checksum error has occurred
1	An AVI infoframe checksum error has occurred

AUD_INF_CKS_ERR_RAW, Addr 40 (IO), *Address 0x88[5] (Read Only)*

Status of Audio Infoframe Checksum Error interrupt signal. When set to 1 it indicates that a checksum error has been detected for an Audio Infoframe. Once set, this bit will remain high until it is cleared via AUDIO_INF_CKS_ERR_CLR.

Function

AUD_INF_CKS_ERR_RAW	Description
0 «	No Audio infoframe checksum error has occurred
1	An Audio infoframe checksum error has occurred

SPD_INF_CKS_ERR_RAW, Addr 40 (IO), *Address 0x88[6] (Read Only)*

Status of SPD Infoframe Checksum Error interrupt signal. When set to 1 it indicates that a checksum error has been detected for an SPD Infoframe. Once set, this bit will remain high until it is cleared via ASPD_INF_CKS_ERR_CLR.

Function

SPD_INF_CKS_ERR_RAW	Description
0 «	No SPD infoframe checksum error has occurred
1	An SPD infoframe checksum error has occurred

MS_INF_CKS_ERR_RAW, Addr 40 (IO), *Address 0x88[7] (Read Only)*

Status of MPEG Source Infoframe Checksum Error interrupt signal. When set to 1 it indicates that a checksum error has been detected for an MPEG Source Infoframe. Once set, this bit will remain high until it is cleared via MS_INF_CKS_ERR_CLR.

Function

MS_INF_CKS_ERR_RAW	Description
0 «	No MPEG source infoframe checksum error has occurred
1	An MPEG source infoframe checksum error has occurred

VS_INF_CKS_ERR_RAW, Addr 40 (IO), *Address 0x8D[0] (Read Only)*

Status of Vendor Specific Infoframe Checksum Error interrupt signal. When set to 1 it indicates that a checksum error has been detected for a Vendor Specific Infoframe. Once set, this bit will remain high until it is cleared via VS_INF_CKS_ERR_CLR.

Function

VS_INF_CKS_ERR_RAW	Description
0 «	No VS infoframe checksum error has occurred
1	A VS infoframe checksum error has occurred

8.32.4 AVI InfoFrame Registers

Table 25 provides a list of readback registers for the AVI InfoFrame data. Refer to the EIA/CEA-861D specifications for a detailed explanation of the AVI InfoFrame fields.

Table 25: AVI InfoFrame Registers

InfoFrame Map Address	Access Type	Register Name	Byte Name[2]
0xE0	R/W	AVI_PACKET_ID[7:0]	Packet Type Value
0xE1	R	AVI_INF_VER	InfoFrame version number
0xE2	R	AVI_INF_LEN	InfoFrame length
0x00	R	AVI_INF_PB_0_1	Checksum
0x01	R	AVI_INF_PB_0_2	Data Byte 1
0x02	R	AVI_INF_PB_0_3	Data Byte 2
0x03	R	AVI_INF_PB_0_4	Data Byte 3
0x04	R	AVI_INF_PB_0_5	Data Byte 4
0x05	R	AVI_INF_PB_0_6	Data Byte 5
0x06	R	AVI_INF_PB_0_7	Data Byte 6
0x07	R	AVI_INF_PB_0_8	Data Byte 7
0x08	R	AVI_INF_PB_0_9	Data Byte 8
0x09	R	AVI_INF_PB_0_10	Data Byte 9
0x0A	R	AVI_INF_PB_0_11	Data Byte 10
0x0B	R	AVI_INF_PB_0_12	Data Byte 11

InfoFrame Map Address	Access Type	Register Name	Byte Name[2]
0x0C	R	AVI_INF_PB_0_13	Data Byte 12
0x0D	R	AVI_INF_PB_0_14	Data Byte 13
0x0E	R	AVI_INF_PB_0_15	Data Byte 14
0x0F	R	AVI_INF_PB_0_16	Data Byte 15
0x10	R	AVI_INF_PB_0_17	Data Byte 16
0x11	R	AVI_INF_PB_0_18	Data Byte 17
0x12	R	AVI_INF_PB_0_19	Data Byte 18
0x13	R	AVI_INF_PB_0_20	Data Byte 19
0x14	R	AVI_INF_PB_0_21	Data Byte 20
0x15	R	AVI_INF_PB_0_22	Data Byte 21
0x16	R	AVI_INF_PB_0_23	Data Byte 22
0x17	R	AVI_INF_PB_0_24	Data Byte 23
0x18	R	AVI_INF_PB_0_25	Data Byte 24
0x19	R	AVI_INF_PB_0_26	Data Byte 25
0x1A	R	AVI_INF_PB_0_27	Data Byte 26
0x1B	R	AVI_INF_PB_0_28	Data Byte 27

[1]As defined by the EIA/CEA-861D specifications

The AVI InfoFrame registers are considered valid if the following two conditions are met:

- AVI_INFO_RAW is 1.
- AVI_INF_CKS_ERR_RAW is 0. This condition applies only if ALWAYS_STORE_INF is set to 1.

AVI_INFO_RAW is described in Section 16.1.

8.32.5 Audio InfoFrame Registers

Table 26 provides the list of readback registers available for the Audio InfoFrame. Refer to the EIA/CEA-861D specifications for a detailed explanation of the Audio InfoFrame fields.

Table 26: Audio InfoFrame Registers

InfoFrame Map Address	Access Type	Register Name	Byte Name[1]
0xE3	R/W	AUD_PACKET_ID[7:0]	Packet Type Value
0xE4	R	AUD_INF_VERS	InfoFrame version number
0xE5	R	AUD_INF_LEN	InfoFrame length
0x1C	R	AUD_INF_PB_0_1	Checksum
0x1D	R	AUD_INF_PB_0_2	Data Byte 1
0x1E	R	AUD_INF_PB_0_3	Data Byte 2
0x1F	R	AUD_INF_PB_0_4	Data Byte 3
0x20	R	AUD_INF_PB_0_5	Data Byte 4
0x21	R	AUD_INF_PB_0_6	Data Byte 5
0x22	R	AUD_INF_PB_0_7	Data Byte 6
0x23	R	AUD_INF_PB_0_8	Data Byte 7
0x24	R	AUD_INF_PB_0_9	Data Byte 8
0x25	R	AUD_INF_PB_0_10	Data Byte 9

InfoFrame Map Address	Access Type	Register Name	Byte Name[1]
0x26	R	AUD_INF_PB_0_11	Data Byte 10
0x27	R	AUD_INF_PB_0_12	Data Byte 11
0x28	R	AUD_INF_PB_0_13	Data Byte 12
0x29	R	AUD_INF_PB_0_14	Data Byte 13

[1]As defined by the EIA/CEA-861D specifications

The audio InfoFrame registers are considered valid if the following two conditions are met:

- AUDIO_INFO_RAW is 1.
- AUD_INF_CKS_ERR_RAW is 0. This condition applies only if ALWAYS_STORE_INF is set to 1.

AUDIO_INFO_RAW, Addr 40 (IO), *Address 0x60[1] (Read Only)*

Raw status of Audio InfoFrame detected signal.

Function

AUDIO_INFO_RAW	Description
0 «	No AVI InfoFrame has been received within the last three VSyncs or since the last HDMI packet detection reset.
1	An Audio InfoFrame has been received within the last three VSyncs. This bit will reset to zero on the fourth VSync leading edge following an Audio InfoFrame, after an HDMI packet detection reset or upon writing to AUD_PACKET_ID.

8.32.6 SPD InfoFrame Registers

Table 27 provides a list of readback registers available for the SPD InfoFrame. Refer to the EIA/CEA-861D specifications for a detailed explanation of the SPD InfoFrame fields.

Table 27: SPD InfoFrame Registers

InfoFrame Map Address	Access Type	Register Name	Byte Name[1]
0xE6	R/W	SPD_PACKET_ID[7:0]	Packet Type Value
0xE7	R	SPD_INF_VER	InfoFrame version number
0xE8	R	SPD_INF_LEN	InfoFrame length
0x2A	R	SPD_INF_PB_0_1	Checksum
0x2B	R	SPD_INF_PB_0_2	Data Byte 1
0x2C	R	SPD_INF_PB_0_3	Data Byte 2
0x2D	R	SPD_INF_PB_0_4	Data Byte 3
0x2E	R	SPD_INF_PB_0_5	Data Byte 4
0x2F	R	SPD_INF_PB_0_6	Data Byte 5
0x30	R	SPD_INF_PB_0_7	Data Byte 6
0x31	R	SPD_INF_PB_0_8	Data Byte 7
0x32	R	SPD_INF_PB_0_9	Data Byte 8
0x33	R	SPD_INF_PB_0_10	Data Byte 9
0x34	R	SPD_INF_PB_0_11	Data Byte 10
0x35	R	SPD_INF_PB_0_12	Data Byte 11
0x36	R	SPD_INF_PB_0_13	Data Byte 12

InfoFrame Map Address	Access Type	Register Name	Byte Name[1]
0x37	R	SPD_INF_PB_0_14	Data Byte 13
0x38	R	SPD_INF_PB_0_15	Data Byte 14
0x39	R	SPD_INF_PB_0_16	Data Byte 15
0x3A	R	SPD_INF_PB_0_17	Data Byte 16
0x3B	R	SPD_INF_PB_0_18	Data Byte 17
0x3C	R	SPD_INF_PB_0_19	Data Byte 18
0x3D	R	SPD_INF_PB_0_20	Data Byte 19
0x3E	R	SPD_INF_PB_0_21	Data Byte 20
0x3F	R	SPD_INF_PB_0_22	Data Byte 21
0x40	R	SPD_INF_PB_0_23	Data Byte 22
0x41	R	SPD_INF_PB_0_24	Data Byte 23
0x42	R	SPD_INF_PB_0_25	Data Byte 24
0x43	R	SPD_INF_PB_0_26	Data Byte 25
0x44	R	SPD_INF_PB_0_27	Data Byte 26
0x45	R	SPD_INF_PB_0_28	Data Byte 27

[1]As defined by the EIA/CEA-861D specifications

The Source Product Descriptor InfoFrame registers are considered valid if the following two conditions are met:

- SPD_INFO_RAW is 1.
- SPD_INF_CKS_ERR_RAW is 0. This condition only applies if ALWAYS_STORE_INF is set to 1.

SPD_INFO_RAW, Addr 40 (IO), *Address 0x60[2] (Read Only)*

Raw status of SPD Infoframe detected signal.

Function

SPD_INFO_RAW	Description
0 «	No source product description InfoFrame received since the last HDMI packet detection reset.
1	Source product description InfoFrame received. This bit will reset to zero after an HDMI packet detection reset or upon writing to SPD_PACKET_ID.

8.32.7 MPEG Source InfoFrame Registers

Table 28 provides a list of readback registers available for the MPEG InfoFrame. Refer to the EIA/CEA-861D specifications for a detailed explanation of the MPEG InfoFrame fields.

Table 28: MPEG InfoFrame Registers

InfoFrame Map Address	Access Type	Register Name	Byte Name[1]
0xE9	R/W	MS_PACKET_ID[7:0]	Packet Type Value
0xEA	R	MS_INF_VERS	InfoFrame version number
0xEB	R	MS_INF_LEN	InfoFrame length
0x46	R	MS_INF_PB_0_1	Checksum
0x47	R	MS_INF_PB_0_2	Data Byte 1
0x48	R	MS_INF_PB_0_3	Data Byte 2
0x49	R	MS_INF_PB_0_4	Data Byte 3
0x4A	R	MS_INF_PB_0_5	Data Byte 4
0x4B	R	MS_INF_PB_0_6	Data Byte 5
0x4C	R	MS_INF_PB_0_7	Data Byte 6
0x4D	R	MS_INF_PB_0_8	Data Byte 7
0x4E	R	MS_INF_PB_0_9	Data Byte 8
0x4F	R	MS_INF_PB_0_10	Data Byte 9
0x50	R	MS_INF_PB_0_11	Data Byte 10
0x51	R	MS_INF_PB_0_12	Data Byte 11
0x52	R	MS_INF_PB_0_13	Data Byte 12
0x53	R	MS_INF_PB_0_14	Data Byte 13

[1]As defined by the EIA/CEA-861D specifications

The MPEG InfoFrame registers are considered valid if the following two conditions are met:

- MS_INFO_RAW is 1.
- MS_INF_CKS_ERR_RAW is 0. This condition applies only if ALWAYS_STORE_INF is set to 1.

MS_INFO_RAW, Addr 40 (IO), *Address 0x60[3] (Read Only)*

Raw status signal of MPEG Source Infoframe detection signal.

Function

MS_INFO_RAW	Description
0 «	No source product description Infoframe received within the last three VSyncs or since the last HDMI packet detection reset.
1	MPEG Source InfoFrame received. This bit will reset to zero after an HDMI packet detection reset or upon writing to MS_PACKET_ID.

8.32.8 Vendor Specific InfoFrame Registers

Table 29 provides a list of readback registers available for the Vendor Specific InfoFrame.

Table 29: VS InfoFrame Registers

InfoFrame Map Address	R/W	Register Name	Byte Name
0xEC	R	VS_PACKET_ID[7:0]	Packet Type Value
0xED	R	VS_INF_VERS	InfoFrame version number
0xEE	R	VS_INF_LEN	InfoFrame length
0x54	R	VS_INF_PB_0_1	Checksum
0x55	R	VS_INF_PB_0_2	Data Byte 1
0x56	R	VS_INF_PB_0_3	Data Byte 2
0x57	R	VS_INF_PB_0_4	Data Byte 3
0x58	R	VS_INF_PB_0_5	Data Byte 4
0x59	R	VS_INF_PB_0_6	Data Byte 5
0x5A	R	VS_INF_PB_0_7	Data Byte 6
0x5B	R	VS_INF_PB_0_8	Data Byte 7
0x5C	R	VS_INF_PB_0_9	Data Byte 8
0x5D	R	VS_INF_PB_0_10	Data Byte 9
0x5E	R	VS_INF_PB_0_11	Data Byte 10
0x5F	R	VS_INF_PB_0_12	Data Byte 11
0x60	R	VS_INF_PB_0_13	Data Byte 12
0x61	R	VS_INF_PB_0_14	Data Byte 13
0x62	R	VS_INF_PB_0_15	Data Byte 14
0x63	R	VS_INF_PB_0_16	Data Byte 15
0x64	R	VS_INF_PB_0_17	Data Byte 16
0x65	R	VS_INF_PB_0_18	Data Byte 17
0x66	R	VS_INF_PB_0_19	Data Byte 18
0x67	R	VS_INF_PB_0_20	Data Byte 19
0x68	R	VS_INF_PB_0_21	Data Byte 20
0x69	R	VS_INF_PB_0_22	Data Byte 21
0x6A	R	VS_INF_PB_0_23	Data Byte 22
0x6B	R	VS_INF_PB_0_24	Data Byte 23
0x6C	R	VS_INF_PB_0_25	Data Byte 24
0x6D	R	VS_INF_PB_0_26	Data Byte 25
0x6E	R	VS_INF_PB_0_27	Data Byte 26
0x6F	R	VS_INF_PB_0_28	Data Byte 27

The Vendor Specific InfoFrame registers are considered valid if the following two conditions are met:

- VS_INFO_RAW is 1.
- VS_INF_CKS_ERR_RAW is 0. This condition applies only if ALWAYS_STORE_INF is set to 1.

VS_INFO_RAW, Addr 40 (IO), *Address 0x60[4] (Read Only)*

Raw status signal of Vendor specific Infoframe detection signal.

Function

VS_INFO_RAW	Description
0 «	No new VS infoframe has been received since the last HDMI packet detection reset.
1	A new VS infoframe has been received. This bit will reset to zero after an HDMI packet detection reset or upon writing to VS_PACKET_ID.

8.33 PACKET REGISTERS

8.33.1 ACP Packet Registers

Table 30 provides the list of readback registers available for the ACP packets. Refer to the HDMI 1.4 specifications for a detailed explanation of the ACP packet fields.

Table 30: ACP Packet Registers

InfoFrame Map Address	R/W	Register Name	Packet Byte No.[1]
0xEF	R/W	ACP_PACKET_ID[7:0]	Packet Type Value
0xF0	R	ACP_TYPE	HB1
0xF1	R	ACP_HEADER2	HB2
0x70	R	ACP_PB_0_1	PB0
0x71	R	ACP_PB_0_2	PB1
0x72	R	ACP_PB_0_3	PB2
0x73	R	ACP_PB_0_4	PB3
0x74	R	ACP_PB_0_5	PB4
0x75	R	ACP_PB_0_6	PB5
0x76	R	ACP_PB_0_7	PB6
0x77	R	ACP_PB_0_8	PB7
0x78	R	ACP_PB_0_9	PB8
0x79	R	ACP_PB_0_10	PB9
0x7A	R	ACP_PB_0_11	PB10
0x7B	R	ACP_PB_0_12	PB11
0x7C	R	ACP_PB_0_13	PB12
0x7D	R	ACP_PB_0_14	PB13
0x7E	R	ACP_PB_0_15	PB14
0x7F	R	ACP_PB_0_16	PB15
0x80	R	ACP_PB_0_17	PB16
0x81	R	ACP_PB_0_18	PB17
0x82	R	ACP_PB_0_19	PB18
0x83	R	ACP_PB_0_20	PB19
0x84	R	ACP_PB_0_21	PB20
0x85	R	ACP_PB_0_22	PB21
0x86	R	ACP_PB_0_23	PB22
0x87	R	ACP_PB_0_24	PB23
0x88	R	ACP_PB_0_25	PB24
0x89	R	ACP_PB_0_26	PB25
0x8A	R	ACP_PB_0_27	PB26
0x8B	R	ACP_PB_0_28	PB27

[1]As defined by the HDMI 1.4 specifications

The ACP InfoFrame registers are considered valid if ACP_PCKT_RAW is set to 1.

ACP_PCKT_RAW, Addr 40 (IO), *Address 0x60[5] (Read Only)*

Raw status signal of Audio Content Protection Packet detection signal.

Function

ACP_PCKT_RAW	Description
0 «	No ACP packet received within the last 600 ms or since the last HDMI packet detection reset.
1	ACP packets have been received within the last 600 ms. This bit will reset to zero after an HDMI packet detection reset or upon writing to ACP_PACKET_ID.

8.33.2 ISRC Packet Registers

Table 31: ISRC1 Packet Registers

InfoFrame Map Address	R/W	Register Name	Packet Byte No.[1]
0xF2	R/W	ISRC1_PACKET_ID[7:0]	Packet Type Value
0xF3	R	ISRC1_HEADER1	HB1
0xF4	R	ISRC1_HEADER2	HB2
0x8C	R	ISRC1_PB_0_1	PB0
0x8D	R	ISRC1_PB_0_2	PB1
0x8E	R	ISRC1_PB_0_3	PB2
0x8F	R	ISRC1_PB_0_4	PB3
0x90	R	ISRC1_PB_0_5	PB4
0x91	R	ISRC1_PB_0_6	PB5
0x92	R	ISRC1_PB_0_7	PB6
0x93	R	ISRC1_PB_0_8	PB7
0x94	R	ISRC1_PB_0_9	PB8
0x95	R	ISRC1_PB_0_10	PB9
0x96	R	ISRC1_PB_0_11	PB10
0x97	R	ISRC1_PB_0_12	PB11
0x98	R	ISRC1_PB_0_13	PB12
0x99	R	ISRC1_PB_0_14	PB13
0x9A	R	ISRC1_PB_0_15	PB14
0x9B	R	ISRC1_PB_0_16	PB15
0x9C	R	ISRC1_PB_0_17	PB16
0x9D	R	ISRC1_PB_0_18	PB17
0x9E	R	ISRC1_PB_0_19	PB18
0x9F	R	ISRC1_PB_0_20	PB19
0xA0	R	ISRC1_PB_0_21	PB20
0xA1	R	ISRC1_PB_0_22	PB21
0xA2	R	ISRC1_PB_0_23	PB22
0xA3	R	ISRC1_PB_0_24	PB23
0xA4	R	ISRC1_PB_0_25	PB24
0xA5	R	ISRC1_PB_0_26	PB25
0xA6	R	ISRC1_PB_0_27	PB26
0xA7	R	ISRC1_PB_0_28	PB27

[1]As defined by the HDMI 1.4 specifications

The ISRC1 packet registers are considered valid if ISRC1_PCKT_RAW is set to 1.

ISRC1_PCKT_RAW, Addr 40 (IO), *Address 0x60[6] (Read Only)*

Raw status signal of International Standard Recording Code 1 (ISRC1) Packet detection signal.

Function

ISRC1_PCKT_RAW	Description
0 «	No ISRC1 packets received since the last HDMI packet detection reset.
1	ISRC1 packets have been received. This bit will reset to zero after an HDMI packet detection reset or upon writing to ISRC1_PACKET_ID.

Table 32: ISRC2 Packet Registers

InfoFrame Map Address	R/W	Register Name	Packet Byte No.[1]
0xF5	R/W	ISRC2_PACKET_ID[7:0]	Packet Type Value
0xF6	R	ISRC2_HEADER1	HB1
0xF7	R	ISRC2_HEADER2	HB2
0xA8	R	ISRC2_PB_0_1	PB0
0xA9	R	ISRC2_PB_0_2	PB1
0xAA	R	ISRC2_PB_0_3	PB2
0xAB	R	ISRC2_PB_0_4	PB3
0xAC	R	ISRC2_PB_0_5	PB4
0xAD	R	ISRC2_PB_0_6	PB5
0xAE	R	ISRC2_PB_0_7	PB6
0xAF	R	ISRC2_PB_0_8	PB7
0xB0	R	ISRC2_PB_0_9	PB8
0xB1	R	ISRC2_PB_0_10	PB9
0xB2	R	ISRC2_PB_0_11	PB10
0xB3	R	ISRC2_PB_0_12	PB11
0xB4	R	ISRC2_PB_0_13	PB12
0xB5	R	ISRC2_PB_0_14	PB13
0xB6	R	ISRC2_PB_0_15	PB14
0xB7	R	ISRC2_PB_0_16	PB15
0xB8	R	ISRC2_PB_0_17	PB16
0xB9	R	ISRC2_PB_0_18	PB17
0xBA	R	ISRC2_PB_0_19	PB18
0xBB	R	ISRC2_PB_0_20	PB19
0xBC	R	ISRC2_PB_0_21	PB20
0xBD	R	ISRC2_PB_0_22	PB21
0xBE	R	ISRC2_PB_0_23	PB22
0xBF	R	ISRC2_PB_0_24	PB23
0xC0	R	ISRC2_PB_0_25	PB24
0xC1	R	ISRC2_PB_0_26	PB25
0xC2	R	ISRC2_PB_0_27	PB26
0xC3	R	ISRC2_PB_0_28	PB27

[1]As defined by the HDMI 1.4 specifications

The ISRC2 packet registers are considered valid if, and only, if ISRC1_PCKT_RAW is set to 1.

ISRC2_PCKT_RAW, Addr 40 (IO), *Address 0x60[7] (Read Only)*

Raw status signal of International Standard Recording Code 2 (ISRC2) Packet detection signal.

Function

ISRC2_PCKT_RAW	Description
0 «	No ISRC2 packets received since the last HDMI packet detection reset.
1	ISRC2 packets have been received. This bit will reset to zero after an HDMI packet detection reset or upon writing to ISRC2_PACKET_ID.

8.33.3 Gamut Metadata Packets

Refer to the HDMI 1.4 specifications for a detailed explanation of the Gamut Metadata packet fields.

Table 33: Gamut Metadata Packet Registers

HDMI Map Address	R/W	Register Name	Packet Byte No.[1]
0xF8	R/W	GAMUT_PACKET_ID[7:0]	Packet Type Value
0xF9	R	GAMUT_HEADER1	HB1
0xFA	R	GAMUT_HEADER2	HB2
0xC4	R	GAMUT_MDATA_PB_0_1	PB0
0xC5	R	GAMUT_MDATA_PB_0_2	PB1
0xC6	R	GAMUT_MDATA_PB_0_3	PB2
0xC7	R	GAMUT_MDATA_PB_0_4	PB3
0xC8	R	GAMUT_MDATA_PB_0_5	PB4
0xC9	R	GAMUT_MDATA_PB_0_6	PB5
0xCA	R	GAMUT_MDATA_PB_0_7	PB6
0xCB	R	GAMUT_MDATA_PB_0_8	PB7
0xCC	R	GAMUT_MDATA_PB_0_9	PB8
0xCD	R	GAMUT_MDATA_PB_0_10	PB9
0xCE	R	GAMUT_MDATA_PB_0_11	PB10
0xCF	R	GAMUT_MDATA_PB_0_12	PB11
0xD0	R	GAMUT_MDATA_PB_0_13	PB12
0xD1	R	GAMUT_MDATA_PB_0_14	PB13
0xD2	R	GAMUT_MDATA_PB_0_15	PB14
0xD3	R	GAMUT_MDATA_PB_0_16	PB15
0xD4	R	GAMUT_MDATA_PB_0_17	PB16
0xD5	R	GAMUT_MDATA_PB_0_18	PB17
0xD6	R	GAMUT_MDATA_PB_0_19	PB18
0xD7	R	GAMUT_MDATA_PB_0_20	PB19
0xD8	R	GAMUT_MDATA_PB_0_21	PB20
0xD9	R	GAMUT_MDATA_PB_0_22	PB21
0xDA	R	GAMUT_MDATA_PB_0_23	PB22
0xDB	R	GAMUT_MDATA_PB_0_24	PB23
0xDC	R	GAMUT_MDATA_PB_0_25	PB24
0xDD	R	GAMUT_MDATA_PB_0_26	PB25
0xDE	R	GAMUT_MDATA_PB_0_27	PB26
0xDF	R	GAMUT_MDATA_PB_0_28	PB27

[1]As defined by the HDMI 1.4 specifications

The Gamut Metadata packet registers are considered valid if GAMUT_MDATA_RAW is set to 1.

GAMUT_MDATA_RAW, Addr 40 (IO), *Address 0x65[0] (Read Only)*

Raw status signal of Gamut Metadata Packet detection signal.

Function

GAMUT_MDATA_RAW	Description
0 «	No Gamut Metadata packet has been received in the last video frame or since the last HDMI packet detection reset.
1	A Gamut Metadata packet has been received in the last video frame. This bit will reset to zero after an HDMI packet detection reset or upon writing to GAMUT_PACKET_ID.

GAMUT_IRQ_NEXT_FIELD, Addr 68 (HDMI), *Address 0x50[4]*

A control to set the NEW_GAMUT_MDATA_RAW interrupt to detect when the new contents are applicable to next field or to indicate that the Gamut packet is new. This is done using header information of the gamut packet.

Function

GAMUT_IRQ_NEXT_FIELD	Description
0 «	Interrupt flag indicates that Gamut packet is new
1	Interrupt flag indicates that Gamut packet is to be applied next field

8.34 CUSTOMIZING PACKET/INFOFRAME STORAGE REGISTERS

The packet type value of each set of packet and InfoFrame registers in the InfoFrame Map is programmable. This allows the user to configure the ADV7842 to store the payload data of any packet and InfoFrames sent by the transmitter connected on the selected HDMI port.

Note: Writing to any of the nine following packet ID registers also clears the corresponding raw InfoFrame / Packet detection bit. E.g. Writing 0x82 -or any other value- to AVI_PACKET_ID clears AVI_INFO_RAW.

AVI_PACKET_ID[7:0], Addr 7C (Infoframe), *Address 0xE0[7:0]*

AVI infoframe ID

Function

AVI_PACKET_ID[7:0]	Description
0xxxxxxx	Packet type value of packet stored in InfoFrame Map, Address 0x00 to 0x1B
1xxxxxxx	Packet type value of InfoFrame stored in InfoFrame Map, Address 0x00 to 0x1B

AUD_PACKET_ID[7:0], Addr 7C (Infoframe), *Address 0xE3[7:0]*

Audio infoframe ID

Function

AUD_PACKET_ID[7:0]	Description
0xxxxxxx	Packet type value of packet stored in InfoFrame Map, Address 0x1C to 0x29
1xxxxxxx	Packet type value of InfoFrame stored in InfoFrame Map, Address 0x1C to 0x29

SPD_PACKET_ID[7:0], Addr 7C (Infoframe), *Address 0xE6[7:0]*

Source Prod infoframe ID

Function

SPD_PACKET_ID[7:0]	Description
0xxxxxxx	Packet type value of packet stored in InfoFrame Map, Address 0x2A to 0x45
1xxxxxxx	Packet type value of InfoFrame stored in InfoFrame Map, Address 0x2A to 0x45

MS_PACKET_ID[7:0], Addr 7C (Infoframe), *Address 0xE9[7:0]*

MPEG Source infoframe ID

Function

MS_PACKET_ID[7:0]	Description
0xxxxxxx	Packet type value of packet stored in InfoFrame Map, Address 0x46 to 0x53
1xxxxxxx	Packet type value of InfoFrame stored in InfoFrame Map, Address 0x46 to 0x53

VS_PACKET_ID[7:0], Addr 7C (Infoframe), *Address 0xEC[7:0]*

Vendor Specific infoframe ID

Function

VS_PACKET_ID[7:0]	Description
0xxxxxxx	Packet type value of packet stored in InfoFrame Map, Address 0x54 to 0x6F
1xxxxxxx	Packet type value of packet stored in InfoFrame Map, Address 0x54 to 0x6F

ACP_PACKET_ID[7:0], Addr 7C (Infoframe), *Address 0xEF[7:0]*

ACP infoframe ID

Function

ACP_PACKET_ID[7:0]	Description
0xxxxxxx	Packet type value of packet stored in InfoFrame Map, Address 0x70 to 0x8B
1xxxxxxx	Packet type value of InfoFrame stored in InfoFrame Map, Address 0x70 to 0x8B

ISRC1_PACKET_ID[7:0], Addr 7C (Infoframe), *Address 0xF2[7:0]*

ISRC1 infoframe ID

Function

ISRC1_PACKET_ID[7:0]	Description
0xxxxxxx	Packet type value of packet stored in InfoFrame Map, Address 0x8C to 0xA7
1xxxxxxx	Packet type value of InfoFrame stored in InfoFrame Map, Address 0x8C to 0xA7

ISRC2_PACKET_ID[7:0], Addr 7C (Infoframe), *Address 0xF5[7:0]*

ISRC2 infoframe ID

Function

ISRC2_PACKET_ID[7:0]	Description
0xxxxxxx	Packet type value of packet stored in InfoFrame Map, Address 0xA8 to 0xC3
1xxxxxxx	Packet type value of InfoFrame stored in InfoFrame Map, Address 0xA8 to 0xC3

GAMUT_PACKET_ID[7:0], Addr 7C (Infoframe), *Address 0xF8[7:0]*

Gamut infoframe ID

Function

GAMUT_PACKET_ID[7:0]	Description
0xxxxxxx	Packet type value of packet stored in InfoFrame Map, Address 0xC4 to 0xDF
1xxxxxxx	Packet type value of InfoFrame stored in InfoFrame Map, Address 0xC4 to 0xDF

Note: The packet type values and corresponding packets should not be programmed in the packet type values registers. These packets are always processed internally and cannot be stored in the packet/InfoFrame registers in the InfoFrame Map:

- 0x01: Audio Clock Regeneration Packet
- 0x02: Audio Sample Packet
- 0x03: General Control Packet
- 0x07: DSD Audio Sample Packet
- 0x09: HBR Audio Stream Packet

8.35 REPEATER SUPPORT

The ADV7842 incorporates an E-EDID/Repeater controller that provides all the features required for a receiver front end of a fully HDCP 1.3 compliant repeater system. The ADV7842 has a RAM that can store up to 24 KSVs, which allows it to handle up to 24 downstream devices in repeater mode (refer to Table 34).

The ADV7842 features a set of HDCP registers, defined in the HDCP specifications, which are accessible through the DDC bus (refer to Section 15.2) of the selected port. A subset of the HDCP registers (defined in the following subsections) are also available in the Repeater Map and are accessible through the main I^2C port (refer to Section 15.1).

8.35.1 Repeater Routines Performed by the E-EDID/Repeater Controller

1. Power Up

A power-on reset circuitry on the VDD supply is used to reset the E-EDID/Repeater controller when the ADV7842 is powered up. When the E-EDID/Repeater controller reboots after reset, it resets all the KSV registers listed in Table 34 to 0x00.

2. AKSV Update

The E-EDID/Repeater controller resets automatically the BCAPS [5] bit to 0 when an HDCP transmitter writes its AKSV into the ADV7842 HDCP registers through the DDC bus of the selected HDMI port.

Note: Writing a value in the AKSV[39:32] triggers an AKSV update and AKSV_UPDATE_ST interrupt if AKSV_UPDATE_MB1 or AKSV_UPDATE_MB2 has been set to 1. This triggers the E-EDID/Repeater controller to reset the BCAPS [5] bit back to 0.

3. KSV List Ready

The KSV_LIST_READY bit is set by an external controller driving the ADV7842. This notifies the ADV7842 on-chip E-EDID/Repeater controller that the KSV list registers have been updated with the KSVs of the attached and active downstream HDCP devices.

When KSV_LIST_READY is set to 1, the E-EDID/Repeater controller computes the SHA-1 hash value V', updates the corresponding V' registers (refer to Table 35), and sets the READY bit (i.e. BCAPS[7:0][5]) to 1. This indicates to the transmitter attached to the ADV7842

that the KSV FIFO and SHA-1 hash value V' are ready to be read.

KSV_LIST_READY, Addr 64 (Repeater), *Address 0x77[7]*

The system sets this bit in order to indicate that the KSV list has been read from the Tx IC(s) and written into the Repeater Map. The system must also set bits [11:0] of Bstatus before setting this bit.

Function

KSV_LIST_READY	Description
0 «	Not Ready
1	Ready

Notes:

- The SHA-1 hash value will be computed if the bit KSV_LIST_READY is set after the part has received an AKSV update from the upstream source. The external controller should therefore set KSV_LST_READY to 1 only after the part has received an AKSV update from the upstream source.
- The ADV7842 does not automatically clear KSV_LIST_READY to 0, after it has finished computed the SHA-1 has value. Therefore the external controller needs to clear KSV_LIST_READY.

1 HDMI Mode

The BSTATUS[15:0][12] bit is updated automatically by the ADV7842 and follows the HDMI mode status of the HDMI/DVI stream input on the active HDMI port. BSTATUS[15:0][12] is set to 1 if the ADV7842 receives an HDMI stream, and set to 0 if the ADV7842 receives a DVI stream.

8.35.2 Repeater Actions Required by External Controller

The external controller must set the BCAPS register and notify the ADV7842 when the KSV list is updated, as described in the following actions 1 to 4.

Note that many more routines must be implemented into the external controller driving the ADV7842 to implement a full repeater. Such routines are described in the HDCP and HDMI specifications (e.g. copying InfoFrame and packet data image from the HDMI receiver into the HDMI transmitter, momentary de-asserting the hot plug detect and disabling the clock termination on a change of downstream topology, and so on).

1. Repeater Bit

The Repeater bit (i.e. BCAPS[7:0][6]) must be set to 1 by the external controller in the routine that initializes the ADV7842. The repeater bit must be left as such as long as the ADV7842 is configured as the front end of a repeater system.

Note: The registers in the KSV list (refer to Table 34) should always be set to 0x0 if the REPEATER bit is set to 0. The firmware running on the external controller, therefore, always sets the registers in the KSV list to 0x0 if the repeater bit is changed from 1 to 0.

2. KSV FIFO Read from HDCP Registers

The KSV FIFO read at address 0x43 through the HDCP port of the selected HDMI port is dependant on the value of the REPEATER bit (i.e. BCAPS[7:0][6]):

- When the Repeater bit is set to 0, the KSV FIFO read from the HDCP port always returns 0x0
- When the Repeater bit is set to 1, the KSV FIFO read from the HDCP port matches the KSV list which is set in the Repeater Map at addresses 0x80 to 0xF7 (refer to Table 34)

3. First AKSV Update

When the upstream transmitter writes its AKSV for the first time into the ADV7842 HDCP registers, the external controller driving the ADV7842 should perform the following tasks:

- Update BSTATUS[15:0][11:0] according to the topology of the downstream device attached to the repeater.
- Update the KSV list (refer to Table 34) with the KSV from the transmitter on the back end of the repeater as well as the KSV from all the downstream devices connected to the repeater.
- Set KSV_LIST_READY to 1.
- The external controller can monitor the AKSV_UPDATE_X_RAW bits to be notified when the transmitter writes its AKSV into the HDCP registers of the ADV7842 (Where X = A or B).

AKSV_UPDATE_A_RAW, Addr 40 (IO), *Address 0x88[1] (Read Only)*

Status of Port A AKSV Update Interrupt signal. When set to 1 it indicates that transmitter has written its AKSV into HDCP registers for Port A. Once set, this bit will remain high until it is cleared via AKSV_UPDATE_A_CLR.

Function

AKSV_UPDATE_A_RAW	Description
0 «	No AKSV updates on Port A.
1	Detected a write access to the AKSV register on Port A.

AKSV_UPDATE_B_RAW, Addr 40 (IO), *Address 0x88[0] (Read Only)*

Status of Port B AKSV Update Interrupt signal. When set to 1 it indicates that transmitter has written its AKSV into HDCP registers for Port B. Once set, this bit will remain high until it is cleared via AKSV_UPDATE_B_CLR.

Function

AKSV_UPDATE_B_RAW	Description
0 «	No AKSV updates on Port B.
1	Detected a write access to the AKSV register on Port A.

4. Second and Subsequent AKSV Updates

When the upstream transmitter writes its AKSV for the second time or more into the ADV7842 HDCP registers, the external controller driving the ADV7842 should set KSV_LIST_READY to 1.

8.35.3 HDCP Registers Available in Repeater Map

In order to enable fast switching of the HDCP encrypted HDMI ports, the registers 0x00 to 0x42 in the repeater map are replicated for each port. AUTO_HDCP_MAP_ENABLE and HDCP_MAP_SELECT[1:0] determine which port is currently visible to the user.

AUTO_HDCP_MAP_ENABLE, Addr 64 (Repeater), *Address 0x7F[2]*

Selects which port will be accessed for HDCP addresses: the HDMI active port (selected by HDMI_PORT_SELECT, HDMI map) or the one selected in HDCP_MAP_SELECT

Function

AUTO_HDCP_MAP_ENABLE	Description
0	HDCP data read from port given by HDCP_MAP_SELECT
1 «	HDCP data read from the active HDMI port

HDCP_MAP_SELECT[1:0], Addr 64 (Repeater), *Address 0x7F[1:0]*

Selects which port will be accessed for HDCP addresses (0x00 to 0x42 in Repeater map). This only takes effect when AUTO HDCP MAN ENABLE is 0

Function

HDCP_MAP_SELECT[1:0]	Description
00 «	Reserved
01	Reserved
10	Select port A
11	Select port B

BKSV[39:0], Addr 64 (Repeater), *Address 0x04[7:0]; Address 0x03[7:0]; Address 0x02[7:0]; Address 0x01[7:0]; Address 0x00[7:0] (Read Only)*

The receiver Key Selection Vector (BKSV) can be read back once the part has successfully accessed the HDCP ROM. The following registers contain the BKSV read from the EEPROM.

Function

BKSV[39:0]	Description
0x00[7:0]	BKSV[7:0]
0x01[7:0]	BKSV[15:8]
0x02[7:0]	BKSV[23:16]
0x03[7:0]	BKSV[31:24]
0x04[7:0]	BKSV[39:32]

AKSV[39:0], Addr 64 (Repeater), *Address 0x14[7:0]; Address 0x13[7:0]; Address 0x12[7:0]; Address 0x11[7:0]; Address 0x10[7:0]*

The AKSV of the transmitter attached to the active HDMI port can be read back after an AKSV update. The following registers contain the AKSV written by the Tx.

Function

AKSV[39:0]	Description
0x10[7:0]	AKSV[7:0]
0x11[7:0]	AKSV[15:8]
0x12[7:0]	AKSV[23:16]
0x13[7:0]	AKSV[31:24]
0x14[7:0]	AKSV[39:32]

BCAPS[7:0], Addr 64 (Repeater), *Address 0x40[7:0]*

This is the BCAPS register presented to the Tx attached to the active HDMI port.

Function

BCAPS[7:0]	Description
10000011 «	Default BCAPS register value presented to the Tx
xxxxxxxx	BCAPS register value presented to the Tx

BSTATUS[15:0], Addr 64 (Repeater), *Address 0x42[7:0]; Address 0x41[7:0]*

These registers contain the BSTATUS information presented to the Tx attached to the active HDMI port. Bits [11:0] must be set by the system software acting as a repeater.

Function

BSTATUS[15:0]	Description
xxxxxxxxxxxxxxxx	BSTATUS register presented to Tx
0000000000000000 «	Reset value. BSTATUS register is reset only after power up.
0x41[7:0]	BSTATUS[7:0]
0x42[7:0]	BSTATUS[15:8]

Table 34: KSV List Registers Location

KSV Number	Register Name	Register Addresses[1]
0	KSV0[39:0]	0x80[7:0]: KSV0[7:0]
		0x81[7:0]: KSV0[15:8]
		0x82[7:0]: KSV0[23:16]
		0x83[7:0]: KSV0[31:24]
		0x84[7:0]: KSV0[39:32]
1	KSV1[39:0]	0x85[7:0]: KSV1[7:0]
		0x86[7:0]: KSV1[15:8]
		0x87[7:0]: KSV1[23:16]
		0x88[7:0]: KSV1[31:24]
		0x89[7:0]: KSV1[39:32]
2	KSV2[39:0]	0x8A[7:0]: KSV2[7:0]
		0x8B[7:0]: KSV2[15:8]
		0x8C[7:0]: KSV2[23:16]
		0x8D[7:0]: KSV2[31:24]
		8x8E[7:0]: KSV2[39:32]
3	KSV3[39:0]	0x8F[7:0]: KSV3[7:0]
		0x90[7:0]: KSV3[15:8]
		0x91[7:0]: KSV3[23:16]
		0x92[7:0]: KSV3[31:24]
		0x93[7:0]: KSV3[39:32]
4	KSV4[39:0]	0x94[7:0]: KSV4[7:0]
		0x95[7:0]: KSV4[15:8]
		0x96[7:0]: KSV4[23:16]
		0x97[7:0]: KSV4[31:24]
		0x98[7:0]: KSV4[39:32]
5	KSV5[39:0]	0x99[7:0]: KSV5[7:0]
		0x9A[7:0]: KSV5[15:8]
		0x9B[7:0]: KSV5[23:16]
		0x9C[7:0]: KSV5[31:24]
		0x9D[7:0]: KSV5[39:32]
6	KSV6[39:0]	0x9E[7:0]: KSV6[7:0]
		0x9F[7:0]: KSV6[15:8]
		0xA0[7:0]: KSV6[23:16]
		0xA1[7:0]: KSV6[31:24]

KSV Number	Register Name	Register Addresses[1]
		0xA2[7:0]: KSV6[39:32]
7	KSV7[39:0]	0xA3[7:0]: KSV7[7:0]
		0xA4[7:0]: KSV7[15:8]
		0xA5[7:0]: KSV7[23:16]
		0xA6[7:0]: KSV7[31:24]
		0xA7[7:0]: KSV7[39:32]
8	KSV8[39:0]	0xA8[7:0]: KSV8[7:0]
		0xA9[7:0]: KSV8[15:8]
		0xAA[7:0]: KSV8[23:16]
		0xAB[7:0]: KSV8[31:24]
		0xAC[7:0]: KSV8[39:32]
9	KSV9[39:0]	0xAD[7:0]: KSV9[7:0]
		0xAE[7:0]: KSV9[15:8]
		0xAF[7:0]: KSV9[23:16]
		0xB0[7:0]: KSV9[31:24]
		0xB1[7:0]: KSV9[39:32]
10	KSV10[39:0]	0xB2[7:0]: KSV10[7:0]
		0xB3[7:0]: KSV10[15:8]
		0xB4[7:0]: KSV10[23:16]
		0xB5[7:0]: KSV10[31:24]
		0xB6[7:0]: KSV10[39:32]
11	KSV11[39:0]	0xB7[7:0]: KSV11[7:0]
		0xB8[7:0]: KSV11[15:8]
		0xB9[7:0]: KSV11[23:16]
		0xBA[7:0]: KSV11[31:24]
		0xBB[7:0]: KSV11[39:32]
12	KSV12[39:0]	0xBC[7:0]: KSV12[7:0]
		0xBD[7:0]: KSV12[15:8]
		0xBE[7:0]: KSV12[23:16]
		0xBF[7:0]: KSV12[31:24]
		0xC0[7:0]: KSV12[39:32]
13	KSV13[39:0]	0xC1[7:0]: KSV13[7:0]
		0xC2[7:0]: KSV13[15:8]
		0xC3[7:0]: KSV13[23:16]
		0xC4[7:0]: KSV13[31:24]
		0xC5[7:0]: KSV13[39:32]
14	KSV14[39:0]	0xC6[7:0]: KSV14[7:0]
		0xC7[7:0]: KSV14[15:8]
		0xC8[7:0]: KSV14[23:16]
		0xC9[7:0]: KSV14[31:24]
		0xCA[7:0]: KSV14[39:32]
15	KSV15[39:0]	0xCB[7:0]: KSV15[7:0]
		0xCC[7:0]: KSV15[15:8]
		0xCD[7:0]: KSV15[23:16]
		0xCE[7:0]: KSV15[31:24]

KSV Number	Register Name	Register Addresses[1]
		0xCF[7:0]: KSV15[39:32]
16	KSV16[39:0]	0xD0[7:0]: KSV16[7:0]
		0xD1[7:0]: KSV16[15:8]
		0xD2[7:0]: KSV16[23:16]
		0xD3[7:0]: KSV16[31:24]
		0xD4[7:0]: KSV16[39:32]
17	KSV17[39:0]	0xD5[7:0]: KSV17[7:0]
		0xD6[7:0]: KSV17[15:8]
		0xD7[7:0]: KSV17[23:16]
		0xD8[7:0]: KSV17[31:24]
		0xD9[7:0]: KSV17[39:32]
18	KSV18[39:0]	0xDA[7:0]: KSV18[7:0]
		0xDB[7:0]: KSV18[15:8]
		0xDC[7:0]: KSV18[23:16]
		0xDD[7:0]: KSV18[31:24]
		0xDE[7:0]: KSV18[39:32]
19	KSV19[39:0]	0xDF[7:0]: KSV19[7:0]
		0xE0[7:0]: KSV19[15:8]
		0xE1[7:0]: KSV19[23:16]
		0xE2[7:0]: KSV19[31:24]
		0xE3[7:0]: KSV19[39:32]
20	KSV20[39:0]	0xE4[7:0]: KSV20[7:0]
		0xE5[7:0]: KSV20[15:8]
		0xE6[7:0]: KSV20[23:16]
		0xE7[7:0]: KSV20[31:24]
		0xE8[7:0]: KSV20[39:32]
21	KSV21[39:0]	0xE9[7:0]: KSV21[7:0]
		0xEA[7:0]: KSV21[15:8]
		0xEB[7:0]: KSV21[23:16]
		0xEC[7:0]: KSV21[31:24]
		0xED[7:0]: KSV21[39:32]
22	KSV22[39:0]	0xEE[7:0]: KSV22[7:0]
		0xEF[7:0]: KSV22[15:8]
		0xF0[7:0]: KSV22[23:16]
		0xF1[7:0]: KSV22[31:24]
		0xF2[7:0]: KSV22[39:32]
23	KSV23[39:0]	0xF3[7:0]: KSV23[7:0]
		0xF4[7:0]: KSV23[15:8]
		0xF5[7:0]: KSV23[23:16]
		0xF6[7:0]: KSV23[31:24]
		0xF7[7:0]: KSV23[39:32]

[1] All KSVs are located in the Repeater Map

Table 35: Registers Location for SHA-1 Hash Value V'

Register Name	Address Location[1]	Function
SHA_A[31:0]	0x20[7:0]: SHA_A[7:0] 0x21[7:0]: SHA_A[15:8] 0x22[7:0]: SHA_A[23:16] 0x23[7:0]: SHA_A[31:24]	H0 part of SHA-1 hash value V'. Register also called (V'.H1)[2]
SHA_B[31:0]	0x24[7:0]: SHA_B[7:0] 0x25[7:0]: SHA_B[15:8] 0x26[7:0]: SHA_B[23:16] 0x27[7:0]: SHA_B[31:24]	H1 part of SHA-1 hash value V'. Register also called (V'.H1)[2]
SHA_C[31:0]	0x28[7:0]: SHA_C[7:0] 0x29[7:0]: SHA_C[15:8] 0x2A[7:0]: SHA_C[23:16] 0x2B[7:0]: SHA_C[31:24]	H2 part of SHA-1 hash value V'. Register also called (V'.H2)[2]
SHA_D[31:0]	0x2C[7:0]: SHA_D[7:0] 0x2D[7:0]: SHA_D[15:8] 0x2E[7:0]: SHA_D[23:16] 0x2F[7:0]: SHA_D[31:24]	H3 part of SHA-1 hash value V'. Register also called (V'.H3)[2]
SHA_E[31:0]	0x30[7:0]: SHA_E[7:0] 0x31[7:0]: SHA_E[15:8] 0x32[7:0]: SHA_E[23:16] 0x33[7:0]: SHA_E[31:24]	H4 part of SHA-1 hash value V'. Register also called (V'.H4)[2]

[1]All registers specified in table are located in the Repeater Map

[2] Refer to HDCP Protection System Standards

8.36 INTERFACE TO DPP SECTION

The video data from the HDMI section is sent to the CP section via the DPP block. The video data output by the HDMI section is always in a 4:4:4 format with 36 bits per pixel. This is irrespective of the encoding format of the video data encapsulated in the HDMI/DVI stream input to the HDMI receiver section (i.e. 4:2:2 or 4:4:4).

- If the HDMI section receives a stream with video encoded in a 4:4:4 format, it passes the video data to the DPP section.

- If the HDMI section receives a stream with video encoded in a 4:2:2 format (refer to
- Figure 91), the HDMI section upconverts the video data into a 4:4:4 format, according to the UP_CONVERSION_MODE bit, and passes the upconverted video data to the DPP section (refer to
- Figure 92).

- If the HDMI receiver receives video data with fewer than 12 bits used per channel, the valid bits are left-shifted on each component channel with zeroes padding the bit below the LSB, before being sent to the DPP section.

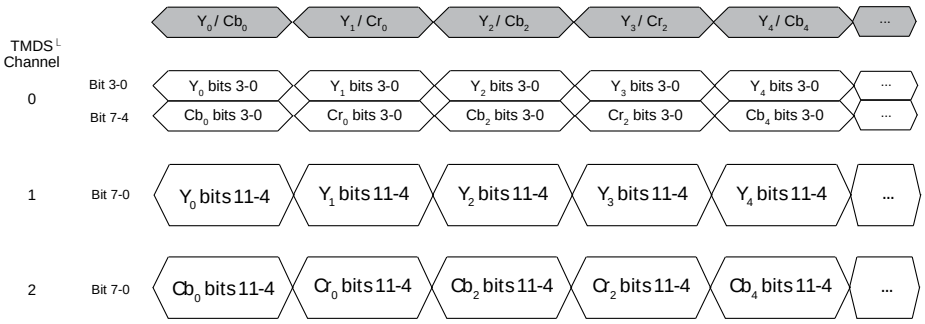


Figure 91: YCbCr 4:2:2 Video Data Encapsulated in HDMI Stream

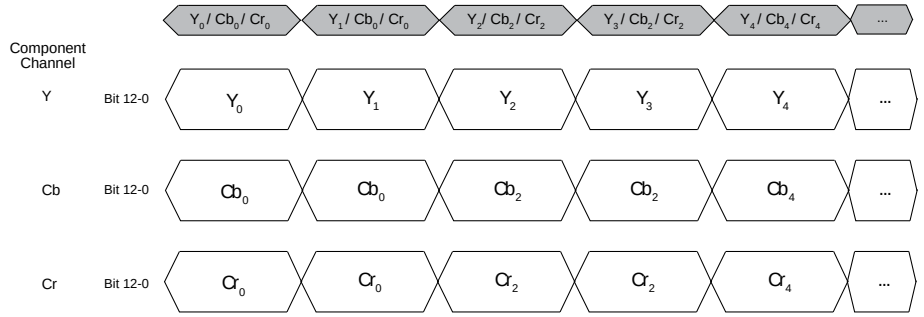


Figure 92: Video Stream Output by HDMI Core for YCbCr 4:2:2 Input and UP_CONVERSION = 0

UP_CONVERSION_MODE, Addr 68 (HDMI), *Address 0x1D[5]*

A control to select linear or interpolated 4:2:2 to 4:4:4 conversion. A 4:2:2 incoming stream is upconverted to a 4:4:4 stream before being sent to the CP.

Function

UP_CONVERSION_MODE	Description
0 «	Cr and Cb samples are repeated in their respective channel.
1	Interpolate Cr and Cb values.

Notes:

- When the ADV7842 pixel output format is set to 4:2:2 (refer to Section 5.1), the DPP section down converts the 4:4:4 stream from the HDMI section according to DS_WITHOUT_FILTER:
  - For a 4:4:4 HDMI input stream to the ADV7842
- The DPP section filters and downsamples the video data from 4:4:4 to 4:2:2 format if DS_WITHOUT_FILTER is set to 0. The DPP section only downsamples, without filtering, the video data from the HDMI section if DS_WITHOUT_FILTER is set to 1.
  - For a 4:2:2 HDMI input stream, the functionality of DS_WITHOUT_FILTER is reversed
- This inversion ensures that for a 4:2:2 HDMI input stream no filtering will be applied if DS_WITHOUT_FILTER is left to its default value 0. When a 4:2:2 HDMI input stream is input to the ADV7842, the DPP section downsamples, without filtering, the video data from 4:4:4 to 4:2:2 format if DS_WITHOUT_FILTER is set to 0. If DS_WITHOUT_FILTER is set to 1, the DPP filters and downsamples the video data from 4:4:4 to 4:2:2 format.

8.37 PASS THROUGH MODE

The ADV7842 can pass through the video data of an HDMI stream with no formatting. The video is passed from the HDMI section through the DPP and CP cores, out through the pixel output formatter without filtering or alteration. This can be achieved with the following settings:

4:2:2 Pass Through

- Set DPP_BYPASS_EN to 1
- Set UP_CONVERSION_MODE to 0
- Set DS_WITHOUT_FILTER to 0
- Configure the pixel output formatter to output a 4:2:2 stream (refer to Section 5.1)

4:4:4 Pass Through

- Set UP_CONVERSION_MODE to 0 or to 1
- Set DS_WITHOUT_FILTER to 0 or to 1
- Configure the pixel output formatter to output a 4:4:4 stream (refer to Section 5.1)

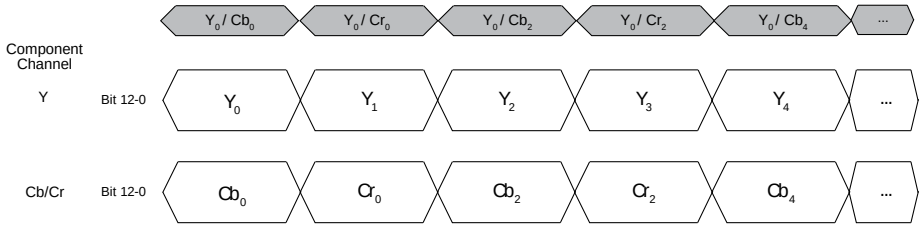


Figure 93: Video Data Output by DPP in 4:2:2 Pass Through Mode

8.38 COLOR SPACE INFORMATION SENT TO THE DPP AND CP SECTIONS

The HDMI section sends information regarding the color space of the video it outputs to the DPP and the CP sections. This color space information is derived from the DVI/HDMI status of the input stream the HDMI section processes and from the AVI InfoFrame that the HDMI section decodes from the input stream.

The color space information sent by the HDMI section to the DDP and CP sections can be read via the HDMI_COLORSPACE[3:0].
HDMI_COLORSPACE[3:0], Addr 68 (HDMI), *Address 0x53[3:0] (Read Only)*

A readback of the HDMI input colorspace decoded from several fields in the AVI infoframe.

Function

HDMI_COLORSPACE[3:0]	Description
0000 «	RGB_LIMITED
0001	RGB_FULL
0010	YUV_601
0011	YUV_709
0100	XVYCC_601
0101	XVYCC_709
0110	YUV_601_FULL
0111	YUV_709_FULL
1000	sYCC 601
1001	Adobe YCC 601
1010	Adobe RGB

8.39 STATUS REGISTERS

Many status bit are available throughout the IO and HDMI Maps. These status bits are listed in the following tables.

Table 36: HDMI Flags in IO Map Register 0x60

Bit Name	Bit Position	Description
AVI_INFO_RAW	0(LSB)	Returns 1 if an AVI InfoFrame was received within last seven VSync. Additional description available here.
AUDIO_INFO_RAW	1	Returns 1 if an AVI InfoFrame was received within last three VSync. Additional description available here.
SPD_INFO_RAW	2	Returns 1 if a Source Product Descriptor InfoFrame has been received. Additional description available here.
MS_INFO_RAW	3	Returns 1 if a MPEG InfoFrame was received within the last three VSyncs Additional description available here.
VS_INFO_RAW	4	Returns 1 if a Vendor Specific InfoFrame has been received. Additional description available here.
ACP_PCKT_RAW	5	Returns 1 if an ACP packet was received within last 600 ms. Additional description available here.
ISRC1_PCKT_RAW	6	Returns 1 if an ISRC1 packet was received. Additional description available here.
ISRC2_PCKT_RAW	7 (MSB)	Returns 1 if an ISRC2 packet was received. Additional description available here.

Table 37: HDMI Flags in IO Map Register 0x65

Bit Name	Bit Position	Description
GAMUT_MDATA_RAW	0 (LSB)	Returns 1 if a Gamut Metadata packet was received. Additional description available here.
AUDIO_C_PCKT_RAW	1	Returns 1 if an audio clock regeneration packet has been received. Reset to 0 following

Bit Name	Bit Position	Description
		a packet detection flag reset condition.
GEN_CTL_PCKT_RAW	2	Returns 1 if general control packet has been received. Reset to 0 following a packet detection flag reset condition.
HDMI_MODE_RAW	3	Returns 1 if a HDMI stream is being received. Additional description available here.
AUDIO_CH_MD_RAW	4	Returns 1 if the audio channel mode is multi-channel (2, 4, 6 or 8 channel) audio. Reset to 0 following a packet detection flag reset condition. Additional description available here.
AV_MUTE_RAW	5	Returns 1 if the latest general control packet received has AV_MUTE asserted. Reset to 0 following packet detection flag reset condition.
INTERNAL_MUTE_RAW	6	Returns 1 if ADV7842 has internally muted the audio data. Additional information available here.
CS_DATA_VALID_RAW	7 (MSB)	Returns 1 if channel status bit readback registers in HDMI Map, Address 0x36 to 0x3A are valid. Additional information available here.

Table 38: HDMI Flags in IO Map Register 0x6A

Bit Name	Bit Position	Description
TMDS_CLK_A_RAW	1	Description available here.
TMDS_CLK_B_RAW	0 (LSB)	Description available here.
TMDSPLL_LCK_A_RAW	5	Description available here.
TMDSPLL_LCK_B_RAW	4	Description available here.

Table 39: HDMI Flags in IO Map Register 0x6F

Bit Name	Bit Position	Description
CABLE_DET_A_RAW	1	Description available here.
CABLE_DET_B_RAW	0 (LSB)	Description available here.
HDMI_ENCRPT_A_RAW	5	Description available here.
HDMI_ENCRPT_B_RAW	4	Description available here.

Table 40: HDMI Flags in IO Map Register 0x74

Bit Name	Bit Position	Description
DE_REGEN_LCK_RAW	0 (LSB)	Description available here.
V_LOCKED_RAW	1	Description available here.

Table 41: HDMI Flags in IO Map Register 0x79

Bit Name	Bit Position
NEW_AVI_INFO_RAW	0 (LSB)
NEW_AUDIO_INFO_RAW	1
NEW_SPD_INFO_RAW	2
NEW_MS_INFO_RAW	3
NEW_VS_INFO_RAW	4
NEW_ACP_PCKT_RAW	5
NEW_ISRC1_PCKT_RAW	6
NEW_ISRC2_PCKT_RAW	7 (MSB)

Table 42: HDMI Flags in IO Map Register 0x7E

Bit Name	Bit Position	Description
NEW_GAMUT_MDATA_RAW	0 (LSB)	When set to 1 indicates that a Gamut Metadata packet with new content has been received. Once set this bit will remain high until the interrupt has been cleared via NEW_GAMUT_MDATA_PCKT_CLR. (IO Map 0x80 [0])
AUDIO_PCKT_ERR_RAW	1	When set to 1 indicates that an uncorrectable error was detected in the body of an audio packet. Once set this bit will remain high until the interrupt has been cleared via AUDIO_PCKT_ERR_CLR (IO Map 0x80 [1])
PACKET_ERROR_RAW	2	When set to 1 it indicates an uncorrectable EEC error was detected in the body or header of any packet. Once set this bit will remain high until the interrupt has been cleared via PACKET_ERROR_CLR (IO Map 0x80 [2])
CHANGE_N_RAW	3	When set to 1 it indicates the N Value of the ACR packets has changed. Once set this bit will remain high until the interrupt has been cleared via CHANGE_N_CLR (IO Map 0x80 [3])
CTS_PASS_THRSH_RAW	4	When set to 1 it indicates the CTS Value of the ACR packets has exceeded the threshold set by CTS_CHANGE_THRESHOLD. Once set this bit will remain high until the interrupt has been cleared via CTS_PASS_THRSH_CLR (IO Map 0x80 [4])
FIFO_OVERFLO_RAW	5	When set to 1 it indicates the Audio FIFO write pointer has reached the read pointer causing the audio FIFO to overflow. Once set this bit will remain high until the interrupt has been cleared via FIFO_OVERFLO_CLR (IO Map 0x80 [5])
FIFO_UNDERFLO_RAW	6	When set to 1 it indicates the Audio FIFO read pointer has reached the write pointer causing the audio FIFO to underflow. Once set this bit will remain high until the interrupt has been cleared via FIFO_UNDERFLO_CLR (IO Map 0x80 [6])
FIFO_NEAR_OVFL_RAW	7 (MSB)	When set to 1 it indicates the Audio FIFO is near overflow as the number FIFO registers containing stereo data is greater or equal to value set in AUDIO_FIFO_ALMOST_FULL_THRESHOLD. Once set this bit will remain high until the interrupt has been cleared via FIFO_NEAR_OVFL_CLR (IO Map 0x80 [7])

Table 43: HDMI Flags in IO Map Register 0x83

Bit Name	Bit Position	Description
FIFO_NEAR_UFLO_RAW	0 (LSB)	When set to 1 it indicates the Audio FIFO is near underflow as the number of FIFO registers containing stereo data is less or equal to value set in AUDIO_FIFO_ALMOST_EMPTY_THRESHOLD. Once set this bit will remain high until the interrupt has been cleared via FIFO_NEAR_UFLO_CLR (IO Map 0x85 [0])
NEW_TMDS_FRQ_RAW	1	When set to 1 it indicates the TMDS Frequency has changed by more than the tolerance set in FREQTOLERANCE[3:0] Once set this bit will remain high until the interrupt has been cleared via NEW_TMDS_FREQ_CLR (IO Map 0x85 [1])
AUDIO_FLT_LINE_RAW	2	When set to 1 it indicates audio sample packet has been received with the Flat line bit set to 1. Once set this bit will remain high until the interrupt has been cleared via AUDIO_FLT_LINE_CLR (IO Map 0x85 [2])
NEW_SAMP_RT_RAW	3	When set to 1 it indicates that audio sampling frequency field in channel status data has changed. Once set this bit will remain high until the interrupt has been cleared via NEW_SAMP_RT_CLR (IO Map 0x85 [3])
PARITY_ERROR_RAW	4	When set to 1 it indicates an audio sample packet has been received with parity error. Once set this bit will remain high until the interrupt has been cleared via PARITY_ERROR_CLR (IO Map 0x85 [4])
AUDIO_MODE_CHNG_RAW	5	When set to 1 it indicates that the type of audio packet received has changed. The following are considered Audio modes, No Audio, PCM, DSD, or HBR. AUDIO_SAMPL_PCKT_DET, DSD_PACKET_DET, and HBR_AUDIO_PCKT_DET used identify type of audio packet currently received. Once set this bit will remain high until the interrupt has been cleared via AUDIO_MODE_CHNG_CLR (IO Map 0x85 [5])
VCLK_CHNG_RAW	6	When set to 1 it indicates that irregular or missing pulses are detected in the TMDS clock. Once set this bit will remain high until the interrupt has been cleared via VCLK_CHNG_CLR (IO Map 0x85 [6])
DEEP_COLOR_CHNG_RAW	7 (MSB)	When set to 1 it indicates a change in the deep color mode has been detected. Once set this bit will remain high until the interrupt has been cleared via DEEP_COLOR_CHNG_CLR (IO Map 0x85 [7])

Table 44: HDMI InfoFrame Checksum Error Flags in IO Map

Bit Name	IO Map Location	Description
AVI_INF_CKS_ERR_RAW	0x88[4]	Description available here.
AUD_INF_CKS_ERR_RAW	0x88[5]	Description available here.
SPD_INF_CKS_ERR_RAW	0x88[6]	Description available here.
MS_INF_CKS_ERR_RAW	0x88[7]	Description available here.
VS_INF_CKS_ERR_RAW	0x8D[0]	Description available here.

Table 45: AKSV Update Flags in IO Map Register 0x88

Bit Name	Bit Position	Description
AKSV_UPDATE_A_RAW	1	When set to 1 it indicates that transmitter has written its AKSV into HDCP registers for Port A. Once set this bit will remain high until the interrupt has been cleared via AKSV_UPDATE_A_CLR (IO Map 0x8A [1])
AKSV_UPDATE_B_RAW	0 (LSB)	When set to 1 it indicates that transmitter has written its AKSV into HDCP registers for Port B. Once set this bit will remain high until the interrupt has been cleared via AKSV_UPDATE_B_CLR (IO Map 0x8A [0])

Table 46: HDMI Flags in HDMI Map

Bit Name	HDMI Map Location	Description
AUDIO_PLL_LOCKED	0x04[0]	Description available here.
AUDIO_SAMPLE_PCKT_DET	0x18[0]	Description available here.
DSD_PACKET_DET	0x18[1]	Description available here.
HBR_PACKET_DET	0x18[3]	Description available here.
DCFIFO_LOCKED	0x1C[3]	Description available here.

8.40 HDMI SECTION RESET STRATEGY

The reset strategy implemented for the HDMI section is described here.

Global Chip Reset

A global chip reset is triggered by asserting the $\overline{\text{RESET}}$ pin to a low level. The HDMI section, excluding the E-EDID/Repeater controller, is reset when a global reset is triggered.

Loss of TMDS Clock or 5 V Signal Reset

A loss of TMDS clock or 5 V signal on the HDMI port selected via HDMI_PORT_SELECT[1:0] resets the entire HDMI section except for the E-EDID/Repeater controller and the audio section. The loss of a 5 V signal condition is discarded if DIS_CABLE_DET_RST is set high.

DVI Mode Reset

The packet processing block, including InfoFrame memory is held in reset when the HDMI section processes a DVI stream.

E-EDID/Repeater Controller Reset

The E-EDID/Repeater controller is reset when the VDD supplies go low or when HDCP_REPT_EDID_RESET is set high.

8.41 HDMI PACKET DETECTION FLAG RESET

A packet detection flag reset is triggered when any of the following events occur:

- The ADV7842 is powered up
- The ADV7842 is reset
- A TMDS clock is detected, after a period of no clock activity, on the selected HDMI port
- The selected HDMI port is changed
- The signal from the 5 V input pin of the HDMI port selected through HDMI_PORT_SELECT transitions to a high. This condition is discarded if DIS_CABLE_DET_RST is set high.

9 DATA PREPROCESSOR, COLOR SPACE CONVERSION AND COLOR CONTROLS

9.1 DATA PREPROCESSOR

The Data Preprocessor (DPP) is positioned after the AFE and HDMI receiver. It receives data directly from either the ADCs in CP mode or from the HDMI receiver section. The DPP block is configured automatically by the PRIM_MODE and VID_STD controls. Selection of different decimation filter responses is possible.

The DPP comprises three main sections, as illustrated in Figure 94

- Stage 1 decimation and filtering
- Color space converter (CSC) matrix
- Stage 2 decimation and filtering

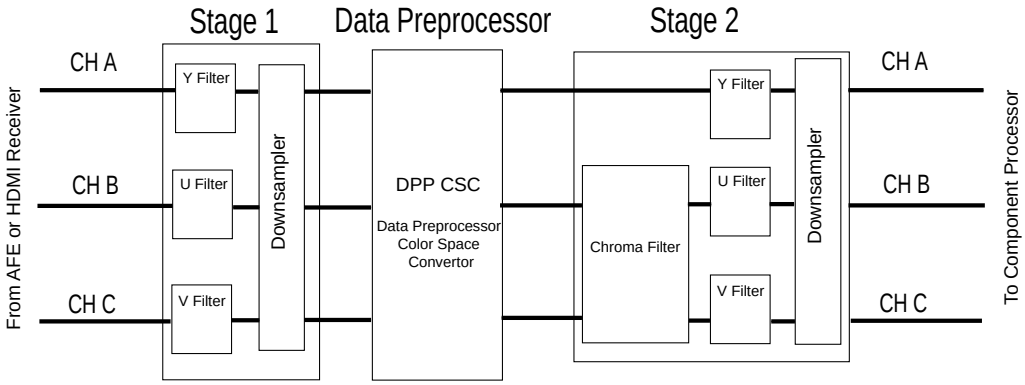


Figure 94 DPP Block Diagram

9.2 DPP ENABLE CONTROL

The DPP_BYPASS_EN bit allows the user to fully bypass the DPP block with no decimation or color space conversion.

The DPP is bypassed automatically in non decimated modes such as 1x1, 2x2, and 4x4 (selected by the PRIM_MODE[3:0] and VID_STD[5:0] controls).

Note:

- The DPP_BYPASS_EN bit is only effective in non decimated modes.
- The DPP is never bypassed in decimation modes such 2x1, 4x2, and 4x1.

DPP_BYPASS_EN, Addr 44 (CP), *Address 0xBD[4]*

Manual control to enable DPP block.

Function

DPP_BYPASS_EN	Description
1 «	DPP Bypassed
0	DPP Enabled

9.3 DECIMATION FILTERS

In the ADV7842, the DPP contains two sets of decimation filters.

The decimation filters are designed as linear phase FIR filters with a low pass response. They should be enabled to reduce the bandwidth of the video stream prior to decimation. This can be necessary under the following conditions:

- The input is 2x oversampled at the ADCs and data must be decimated, for example, PR-2X1 mode of operation:
  - If the intended output interface is 4:4:4, all three datastreams must be decimated by 2.
  - If the user wants a 4:2:2 output interface, channel A must be decimated by 2, channels B and C must be decimated by 4.
- In case the mode of operation is not oversampled (for example, 1x1) or the output is not to be decimated (for example, 2X2):
  - For a 4:4:4 output interface, no decimation is to be performed.
  - To achieve a 4:2:2 output datastream, channel A is not decimated, channels B and C must be decimated by 2.
- The input is 4x oversampled at the ADCs and data must be decimated, for example, SD-4X1 mode of operation:
  - If the intended output interface is 4:4:4, all three datastreams must be decimated by 4.
  - If the user wants a 4:2:2 output interface, channel A must be decimated by 4, channels B and C must be decimated by 8.

Notes:

- The decimation filters are automatically bypassed for HDMI 4:2:2 inputs.
- Decimation filters for the SDP core are described in Section 7.6

9.3.1 DPP Decimation Filter Automatic Selection

The ADV7842 has an automatic selection algorithm for the decimation filters in the DPP block. Based on the selected PRIM_MODE[3:0], VID_STD[5:0] and OP_FORMAT_SEL[7:0], the ADV7842 decides on the best filter mode to be used, as indicated in Table 47: DPP Filter Auto Selection

Table 47: DPP Filter Auto Selection

Mode of Operation	Example		Output Interface	Decimation Factor for Channel A	Decimation Factor for Channel B/C
Quad rate oversampled modes	COMP	SD 4X2	4:4:4 (e.g. 30-bit)	2	2
		SD 4X2	4:2:2 (e.g. 20-bit)	2	4
		SD/PR 4x1	4:4:4 (e.g. 30-bit)	4	4
		SD/PR 4x1	4:2:2 (e.g. 20-bit)	4	8
Double rate oversampled modes	COMP	SD/HD/PR 2X1	4:2:2 (e.g. 20-bit)	2	4
No oversampling mode	COMP or HDMI(COMP)	HD 1x1, HDMI 1x1	4:4:4 (e.g. 30-bit)	n/a[1]	n/a[1]
		PR 1x1	4:4:4 (e.g. 30-bit)	n/a[1]	n/a[1]
		HD 1x1, HDMI 2x1	4:2:2 (e.g. 20-bit)	n/a[1]	2
		PR 1x1	4:2:2 (e.g. 20-bit)	n/a[1]	2
	GR or HDMI(GR)	All	4:2:2 (e.g. 20-bit)	n/a[1]	2
		All	4:4:4 (e.g. 30-bit)	n/a[1]	n/a[1]

1 No decimation

9.3.2 Decimation Filter Selection for Stage 1

The first set of three decimation filters are positioned before the DPP CSC in stage 1. The decimation filters in stage 1 of the DPP have a fixed filter response.

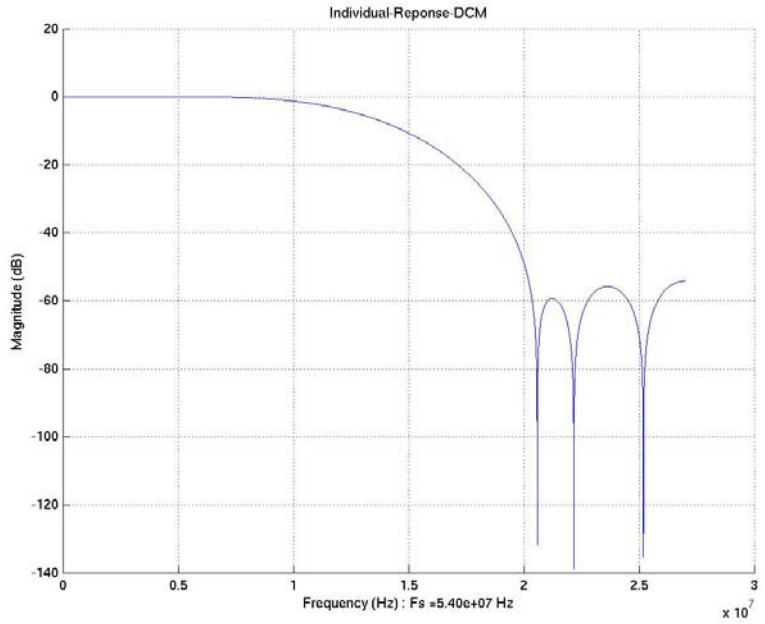


Figure 95: DPP Stage 1 Fixed Decimation Filter Frequency Response

9.3.3 Decimation Filter Selection for Stage 2

The second set of three decimation filters is positioned after the DPP CSC in stage 2. The decimation filters in stage 2 of the DPP have two possible filter responses.

The filter response on channel A is selected by DPP_LUMA_HBW_SEL. The alternative filter response is a higher bandwidth response. This also has lower stop-band attenuation.

DPP_LUMA_HBW_SEL, Addr 40 (IO), *Address 0xE7[5]*

A control to select the DPP Luma filter bandwidth for stage 2 filters.

Function

DPP_LUMA_HBW_SEL	Description
0 «	Select Low bandwidth (0.44Fs) Higher stopband attenuation
1	Select High bandwidth (0.47 Fs) Lower stop-band attenuation

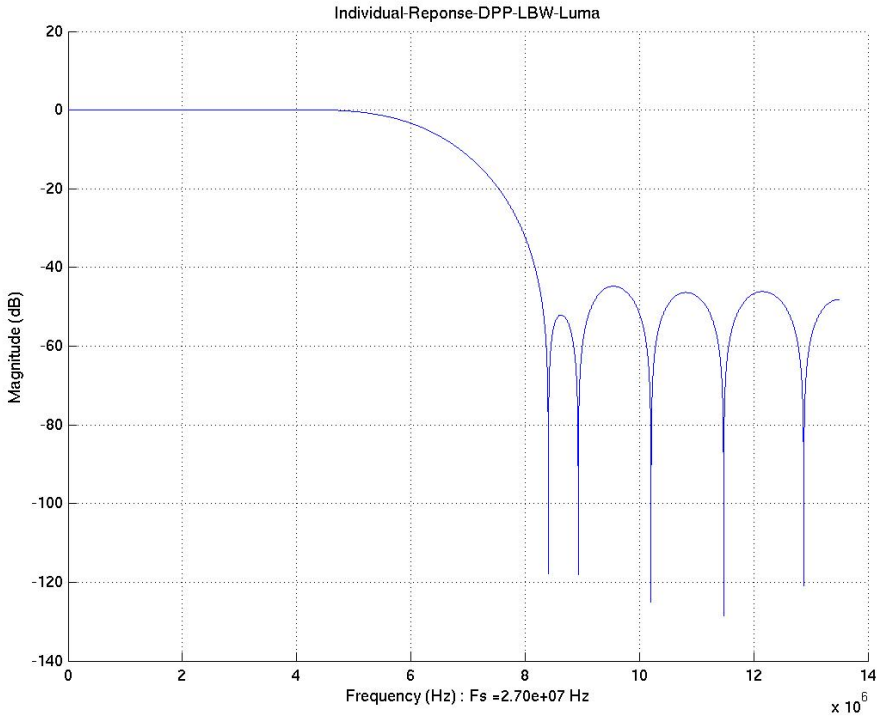


Figure 96: DPP Stage 2 Default Luma Filter Frequency Response

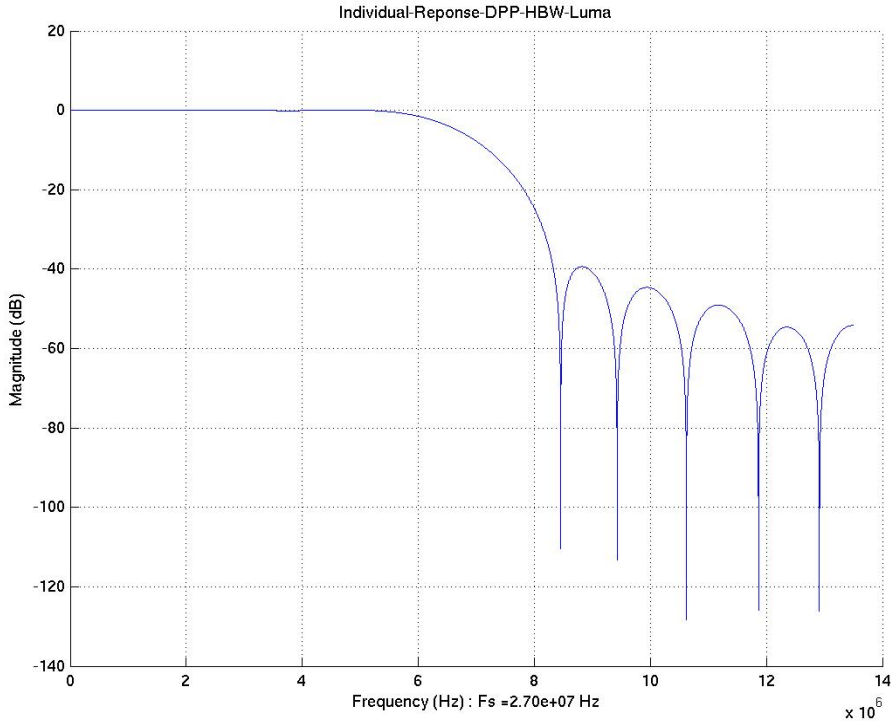


Figure 97: DPP Stage 2 High Bandwidth Luma Filter Frequency Response

The filter response on channels B and C is selected by DPP_CHROMA_LOW_EN. The default filter has high bandwidth and a sharp roll off. The alternative filter response has a softer roll off which results in less ringing.

DPP_CHROMA_LOW_EN, Addr 40 (IO), *Address 0xE7[4]*

A control to select DPP Chroma filter bandwidth for stage 2 filters.

Function

DPP_CHROMA_LOW_EN	Description
0 «	High bandwidth, sharp transition filter for channels B/C
1	Soft filter with minimized ringing for channels B/C

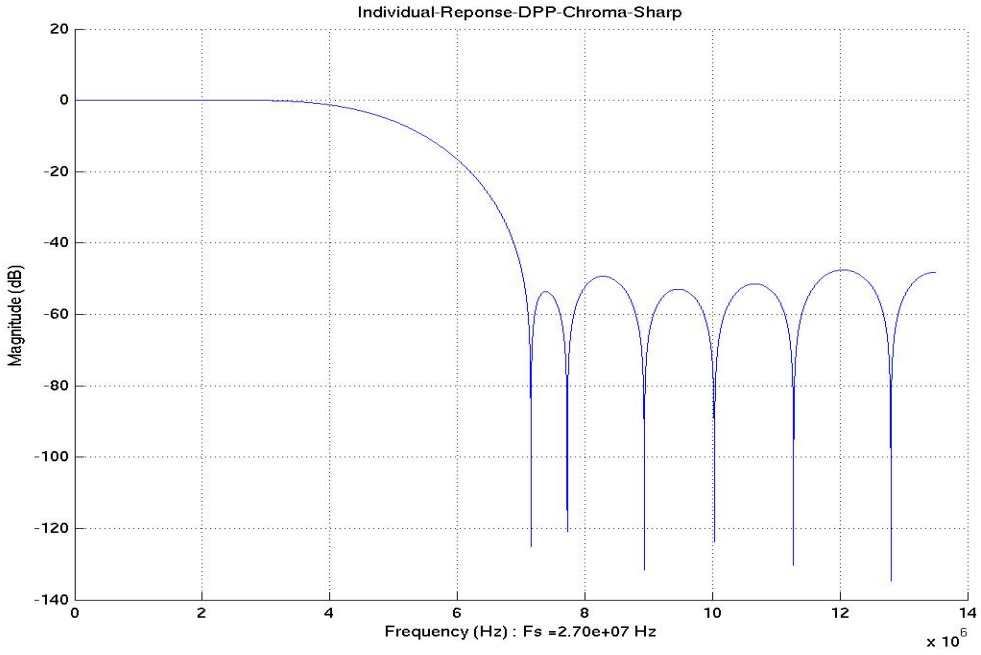
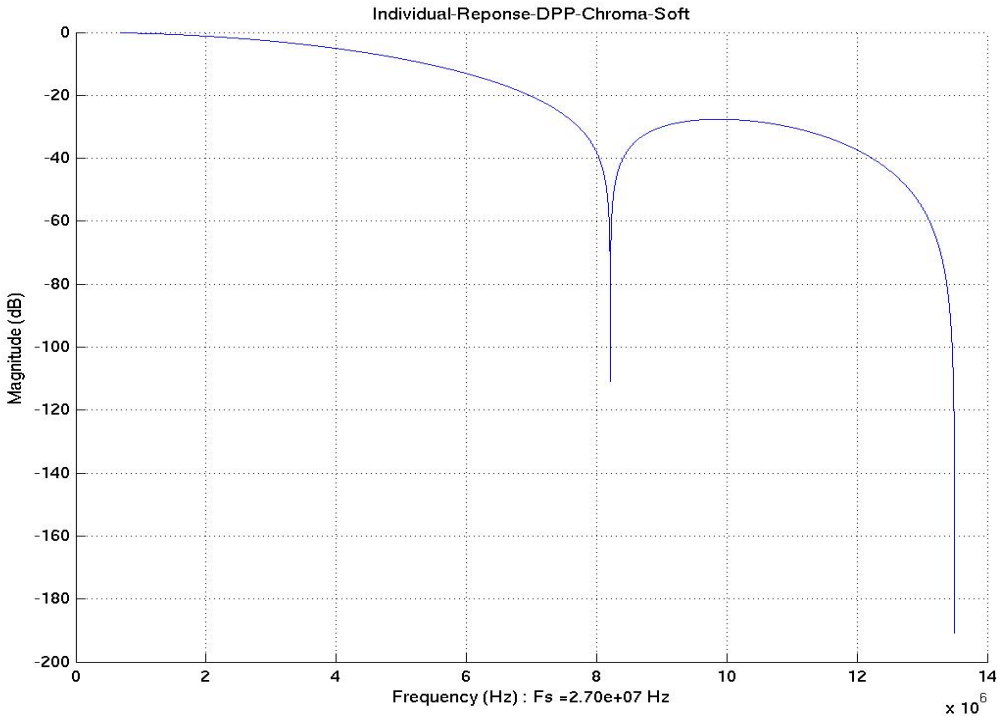


Figure 98: DPP Stage 2 Sharp Chroma Filter Frequency Response



9.3.4 DPP Decimation Only Selection

In some systems, it may be desirable to downsample the channels without any filter operation. To achieve this, the DS_WITHOUT_FILTER bit can be selected. The DS_WITHOUT_FILTER bit disables all filters on channels B, and C while keeping the downsampler (data dropping) functional.

Important: In HDMI mode, when a 4:2:2 input is received the functionality of this bit is reversed.

DS_WITHOUT_FILTER, Addr 40 (IO), *Address 0xE0[7]*

Disables the chroma filters on channel B and C while keeping the downsampler functional

Function

DS_WITHOUT_FILTER	Description
0 «	Filters and downsamples
1	Downsamples only (no filtering)

9.3.5 Decimation Filters for CP

The CP section features a control that selects the filter response on channels B and C in the CP section. This is used for 444 to 422 decimation. The control bit is CP_CHROMA_LOW_EN.

CP_CHROMA_LOW_EN, Addr 44 (CP), *Address 0x68[3]*

Filter Response Control for the 444 to 422 Chroma decimation filter

Function

CP_CHROMA_LOW_EN	Description
0 «	High bandwidth, sharp transition filter for channels B/C
1	Soft filter with minimized ringing for channels B/C

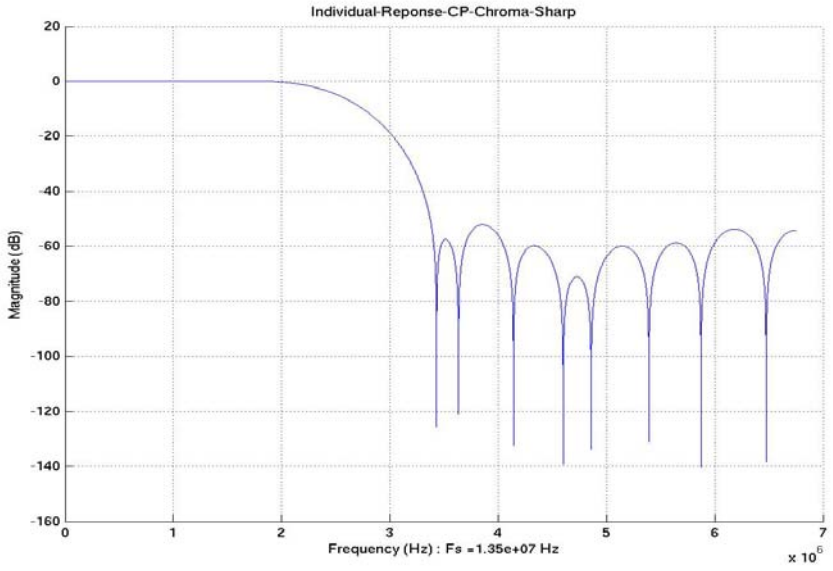


Figure 99: CP Sharp Chroma Filter Frequency Response

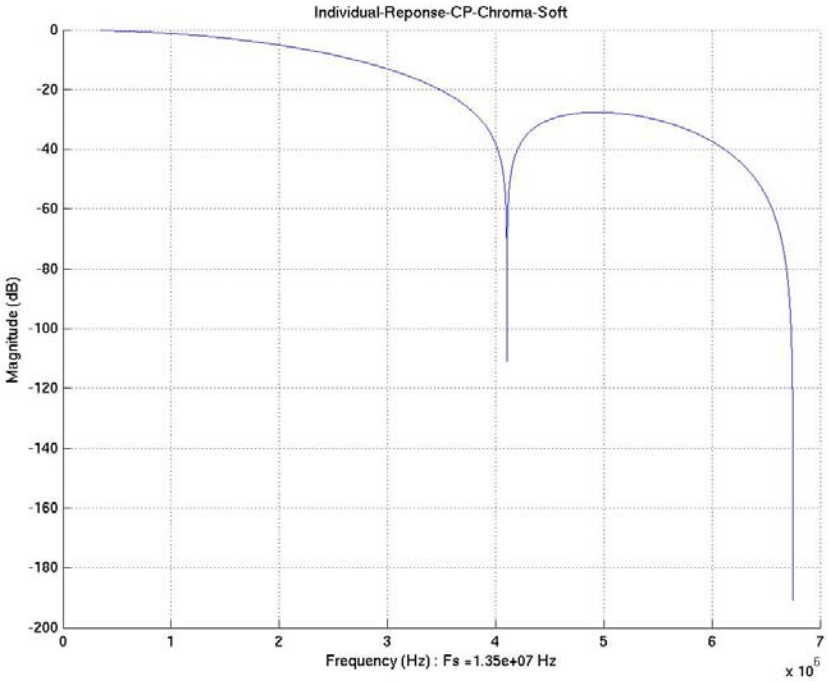


Figure 100: CP Soft Chroma Filter Frequency Response

9.4 COLOR SPACE CONVERSION MATRIX

The ADV7842 provides any-to-any color space conversion support. It supports formats such as RGB, YUV, and YCbCr. The DPP Color Space Converter (CSC) and CP CSC are designed to run at speeds of up to 170 MHz.

The CSC is in the CP block (CP CSC). The CP also provides color controls for brightness, contrast, saturation and hue adjustments. The CP CSC is the main color space converter. The DPP block also has an automatic non programmable CSC. The ADV7842 will automatically configure the DPP CSC for certain modes, depending on the input and output formats and the use of the color control feature.

The configuration of the color space conversion using the CP CSC block and a description of the adjustable register bits are provided in Figure 101.

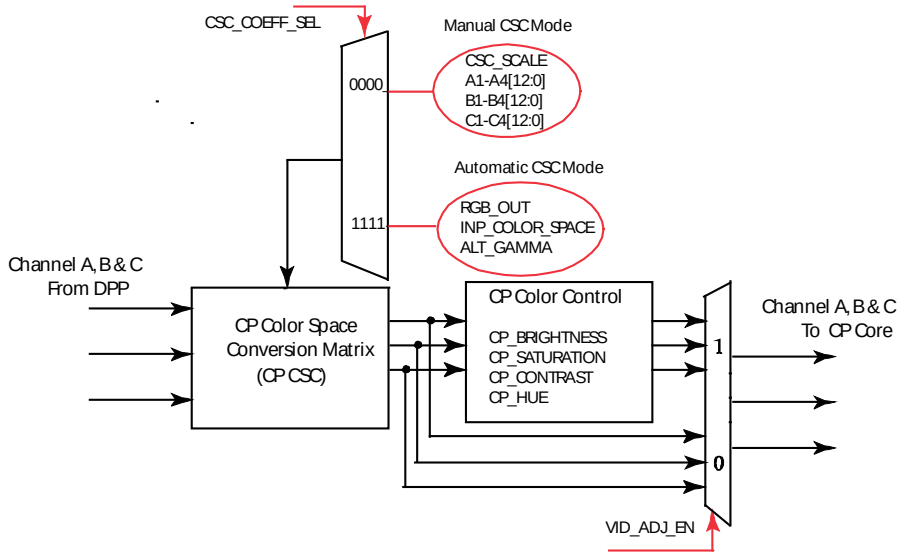


Figure 101: Configuring CP CSC Block

9.4.1 CP CSC Selection

MAN_CP_CSC_EN, Addr 44 (CP), *Address 0x69[4]*

A control to manually enable the CP CSC. By default the CP CSC will be automatically enabled in the case that either a color-space conversion or video-adjustments (Hue, Saturation, Contrast, Brightness) is determined to be required due to other I2C settings. If MAN_CP_CSC_EN is set to one the CP CSC is forced into the enabled state.

Function

MAN_CP_CSC_EN	Description
0 «	CP CSC will be automatically enabled if required. For example if either a color-space conversion or video-adjustments (Hue, Saturation, Contrast, Brightness) is determined to be required due to other I2C settings.
1	Manual override to force CP-CSC to be enabled

9.4.2 Selecting Automatic or Manual CP CSC Conversion Mode

The ADV7842 CP CSC provides two modes for the CSC configuration: automatic CSC mode and manual CSC mode.

In automatic CSC mode, the user is required to program the input color space and the output color space for the correct operation of the CSC matrix. Manual CSC mode allows the user to program all the color space conversion by manually programming CSC coefficients.

CSC_COEFF_SEL[3:0], Addr 44 (CP), *Address 0x68[7:4]*

A control to select the mode the CP CSC operates in.

Function

CSC_COEFF_SEL[3:0]	Description
0000	CP CSC configuration in manual mode
1111 «	CP CSC configured in automatic mode
xxxx	Reserved

The CSC configuration mode is automated in the ADV7842. Automatic or manual CSC mode can be selected by setting the CSC_COEFF_SEL[3:0] bits. When CSC_COEFF_SEL[3:0] is set to 0b1111, the CSC mode is automatically selected, based on the input color space and output color space required and set through the following registers:

- INP_COLOR_SPACE[3:0]
- RGB_OUT
- ALT_GAMMA

9.4.3 Automatic Color Space Conversion Matrix

In automatic mode, the CSC matrix, AGC target gain values, and offset values can be configured automatically via the following set of registers:

- INP_COLOR_SPACE[3:0]
- RGB_OUT
- ALT_GAMMA
- OP_656_RANGE_SEL

INP_COLOR_SPACE[3:0], Addr 40 (IO), *Address 0x02[7:4]*

A control to set the colorspace of the input video. To be used in conjunction with ALT_GAMMA and RGB_OUT to configure the color space converter. A value of 4'b1111 selects automatic setting of the input color space base on the primary mode and video standard settings. Settings 1000 to 1110 are undefined.

Function

INP_COLOR_SPACE[3:0]	Description
0000	Forces RGB (range 16 to 235) input
0001	Forces RGB (range 0 to 255) input
0010	Forces YCrCb input (601 color space) (range 16 to 235)
0011	Forces YCrCb input (709 color space) (range 16 to 235)
0100	Forces XVYCC 601
0101	Forces XVYCC 709
0110	Forces YCrCb input (601 color space) (range 0 to 255)
0111	Forces YCrCb input (709 color space) (range 0 to 255)
1111 «	In analog mode, input color space depends on Primary Mode and Video Standard. In HDMI mode, input color space depends on color space reported by HDMI block.

Table 48: Automatic Input Color Space Selection

PRIM_MODE[3:0]	VID_STD[5:0]	Input Color Space	Input Range	Comments
0001	≤ 1001	YcrCb601	0:255	Analog SD/ED modes
0001	> 1001	YcrCb709	0:255	Analog HD modes
0010	xxxx	RGB	0:255	Analog GR modes
0101	xxxx	Dependant on AVI InfoFrame	0:255 for YUV Dependant on AVI InfoFrame for RGB	HDMI component modes
0110	xxxx	Dependant on AVI InfoFrame	0:255 for YUV Dependant on AVI InfoFrame for RGB	HDMI graphic modes

RGB_OUT, Addr 40 (IO), *Address 0x02[1]*

A control to select output color space and the correct digital blank level and offsets on the RGB or YPrPb outputs. It is used in conjunction with the INP_COLOR_SPACE[3:0] and ALT_GAMMA bits to select the applied CSC.

Function

RGB_OUT	Description
0 «	YPbPr color space output
1	RGB color space output

ALT_GAMMA, Addr 40 (IO), *Address 0x02[3]*

A control to select the type of YPbPr colorspace conversion. This bit is to be used in conjunction with INP_COLOR_SPACE[3:0] and RGB_OUT. If ALT_GAMMA is set to 1 and RGB_OUT= 0 a colorspace conversion is applied to convert from 601 to 709 or 709 to 601. Valid only if RGB_OUT set to 0.

Function

ALT_GAMMA	Description
0 «	No conversion
1	YUV601 to YUV709 conversion applied if input is YUV601. YUV709 to YUV601 conversion applied if input is YUV709

Table 49: Automatic CSC Selection

INP_COLOR_SPACE[3:0] (Input Color Space)	RGB_OUT	CSC Mode Used (Output)	
		ALT_GAMMA = 0	ALT_GAMMA = 1
00 - RGB	0	YCbCr 601	YCbCr 709
	1	RGB	RGB
01 - (YCbCr /YUV 601)	0	YCbCr 601	YCbCr 709
	1	RGB	RGB
10 - (YCbCr /YUV 709)	0	YCbCr 709	YCbCr 601
	1	RGB	RGB

CSC_COEFF_SEL_RB[3:0], Addr 44 (CP), *Address 0xF4[7:4] (Read Only)*

Readback of the CP CSC conversion when configured in automatic mode

Function

CSC_COEFF_SEL_RB[3:0]	Description
0000 «	CSC is bypassed
0001	YPbPr 601 to RGB
0011	YPbPr 709 to RGB
0101	RGB to YPbPr 601
0111	RGB to YPbPr 709
1001	YPbPr 709 to YPbPr 601
1010	YPbPr 601 to YPbPr 709
1111	CSC in manual mode
xxxx	Reserved

Table 50: CSC Configuration for All CSC Modes Reported by CSC_COEFF_SEL_RB

CSC Mode	CSC_SCALE[1:0]	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4
0b0000	CSC in bypass mode. In this mode the CSC effectively performs a color conversion based on the CSC coefficients set in registers CSC_SCALE, A1, A2, A3, A4, B1, B2, B3, B4, C1, C2, C3, and C4.												
0b0001	0b01	0x0800	0x1A6A	0x1D50	0x0423	0x0800	0x0AF8	0x0000	0x1A84	0x0800	0x0000	0x0DDB	0x1912
0b0011	0b01	0x0800	0x1C54	0x1E89	0x0291	0x0800	0x0C52	0x0000	0x19D7	0x0800	0x0000	0x0E87	0x18BC
0b0101	0b00	0x0964	0x04C9	0x01D3	0x0000	0x1927	0x082D	0x1EAC	0x0800	0x1A93	0x1D3F	0x082D	0x0800
0b0111	0b00	0x0B71	0x0368	0x0127	0x0000	0x1893	0x082D	0x1F3F	0x0800	0x19B2	0x1E21	0x082D	0x0800
0b1001	0b01	0x0800	0x0188	0x00CB	0x1ED7	0x0000	0x07DE	0x1F6C	0x005B	0x0000	0x1F1D	0x07EB	0x007B
0b1010	0b01	0x0800	0x1E56	0x1F14	0x014A	0x0000	0x0834	0x009A	0x1F9A	0x0000	0x00EB	0x0826	0x1F78

9.4.4 HDMI Automatic CSC Operation

In HDMI mode, the ADV7842 provides an automatic CSC function based on the AVI InfoFrame sent from the source. The flowchart in Figure 102 shows the mechanism of the ADV7842 automatic CSC functionality in HDMI mode.

Note: In the following flowcharts a red dashed line represents a state that is undefined according to the CEA-861D spec, and therefore should never happen. In the event that it did somehow occur the ADV7842 retains the previous colorimetry.

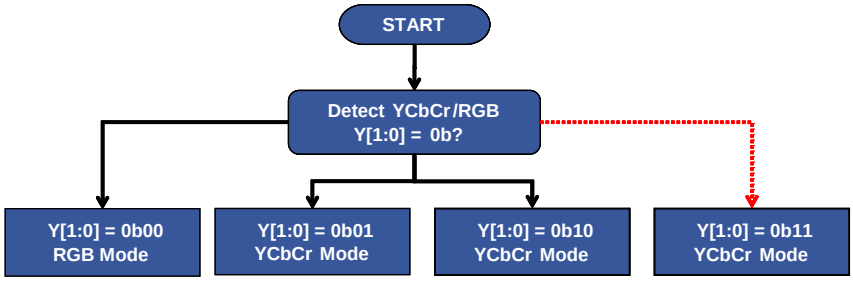


Figure 102: HDMI Automatic CSC Flowchart

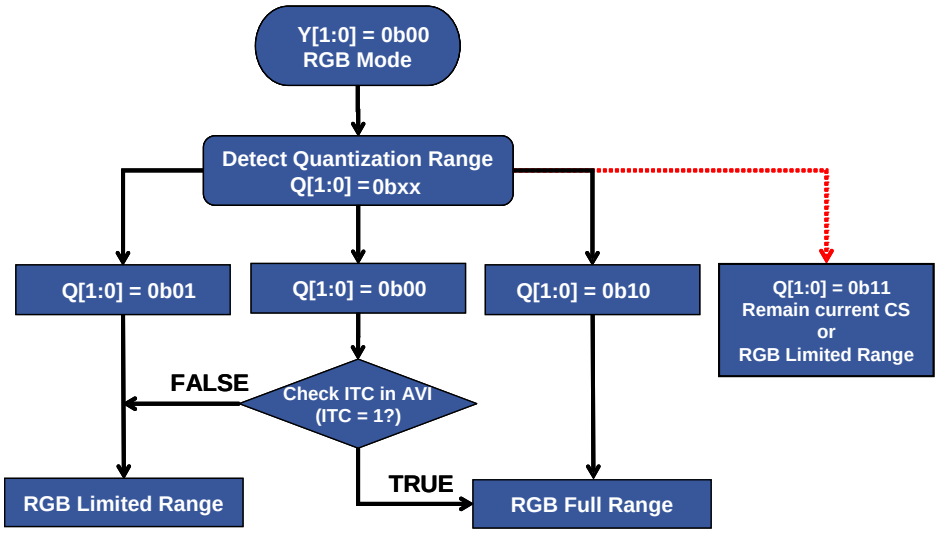


Figure 103: HDMI Automatic CSC Flowchart (Case RGB)

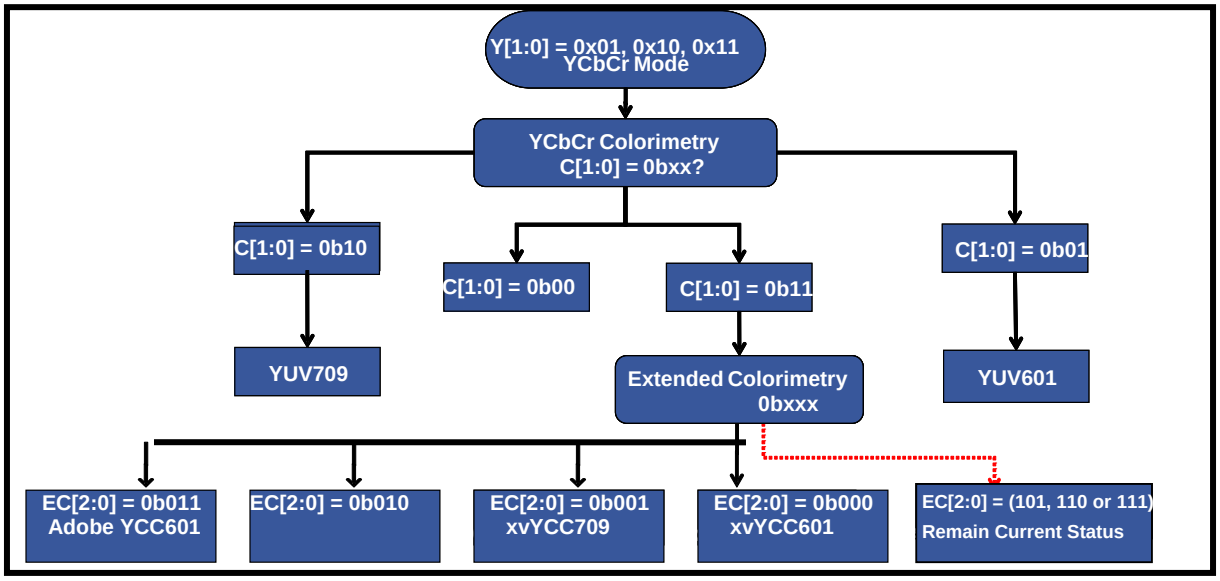


Figure 104: HDMI Automatic CSC Flowchart (Case YCbCr-1)

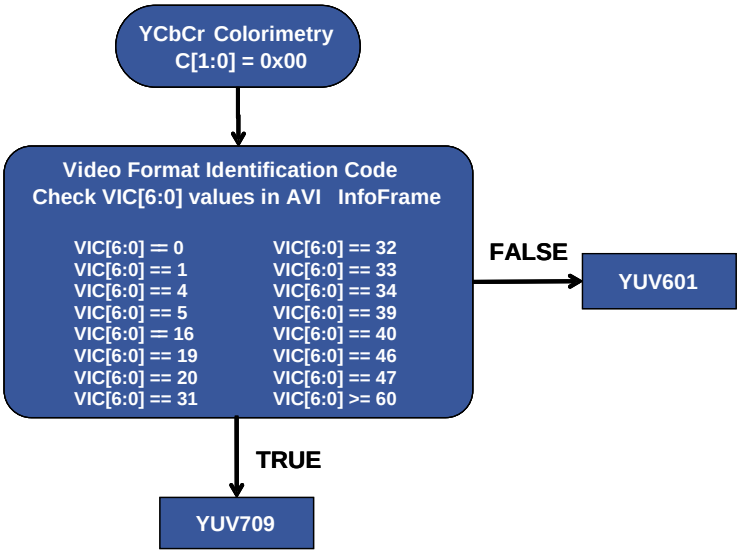


Figure 105: HDMI Automatic CSC Flowchart (Case YCbCr-2)

In the RGB case (refer to Figure 106), the ADV7842 has the programmability to control manually the RGB limited/full range regardless of the ITC bit.

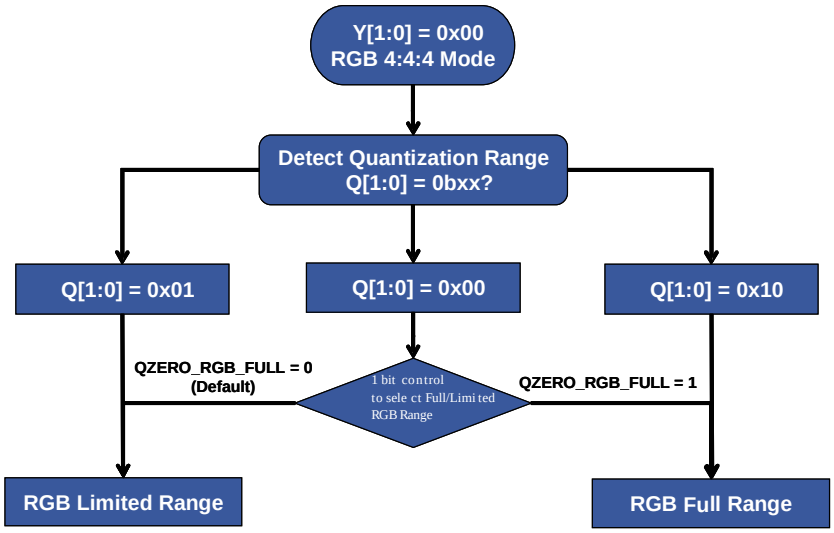


Figure 106: Manual RGB Range Control Flowchart for Automatic CSC (Case RGB)

QZERO_ITC_DIS, Addr 68 (HDMI), *Address 0x47[2]*

A control to select manual control of the RGB colorimetry when the AVI infoframe field Q[1:0]=00. To be used in conjunction with QZERO_RGB_FULL

Function

QZERO_ITC_DIS	Description
0 «	AVI InfoFrame ITC bit decides RGB-full or limited range in case Q[1:0]=00
1	Manual RGB range as per QZERO_RGB_FULL.

QZERO_RGB_FULL, Addr 68 (HDMI), *Address 0x47[1]*

A control to manually select the HDMI colorimetry when AVI infoframe field Q[1:0]=00. Valid only when QZERO_ITC_DIS is set to 1.

Function

QZERO_RGB_FULL	Description
0 «	RGB-limited range when Q[1:0]=00
1	RGB-full when Q[1:0]=00

9.4.5 Manual Color Space Conversion Matrix

The CP CSC matrix in the ADV7842 is a 3 x 3 matrix with full programmability of all coefficients in the matrix in manual mode. Each coefficient is 12-bits wide to ensure signal integrity is maintained in the CP CSC section. The CP CSC contains three identical processing channels, one of which is shown in Figure 107.

The main inputs labeled In_A, In_B, and In_C can come from each ADC or from the 36-bit digital input from the HDMI section. Each input to the individual channels to the CP CSC is multiplied by a separate coefficient for each channel.

In Figure 107, these coefficients are marked A1, A2 and A3. The variable labeled A4 is used as an offset control for channel A in the CSC. There is also a further CP CSC control bit labeled CSC_SCALE[1:0]; this control can be used to accommodate coefficients that extend the supported range. The functional diagram for a single channel in the CP CSC as per Figure 107 is repeated for the other two remaining channels B and C. The coefficients for these channels are called B1, B2, B3, B4, C1, C2, C3 and C4.

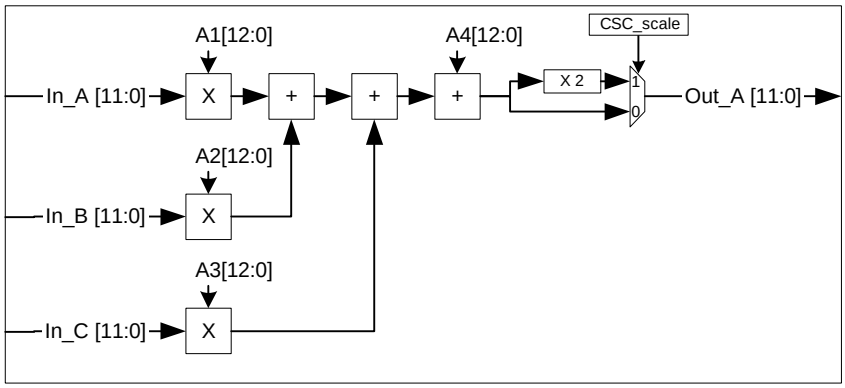


Figure 107: Single CSC Channel

The coefficients mentioned previously are detailed in Table 51 along with the default values for these coefficients.

Table 51. CSC Coefficients

Function Bit	CP Map Address	Reset Value (Hex)	Description
A1[12:0]	0x57 [4:0], 0x58 [7:0]	0x800	Coefficient for channel A
A2[12:0]	0x55 [1:0], 0x56 [7:0] 0x57 [7:5]	0x000	Coefficient for channel A
A3[12:0]	0x54 [6:0], 0x55 [7:2]	0x000	Coefficient for channel A
B1[12:0]	0x5E [4:0], 0x5F [7:0]	0x000	Coefficient for channel B
B2[12:0]	0x5C [1:0], 0x5D [7:0] 0x5E [7:5]	0x800	Coefficient for channel B
B3[12:0]	0x5B [6:0], 0x5C [7:2]	0x000	Coefficient for channel B
C1[12:0]	0x65 [4:0], 0x66 [7:0]	0x000	Coefficient for channel C
C2[12:0]	0x63 [1:0], 0x64 [7:0] 0x65 [7:5]	0x000	Coefficient for channel C
C3[12:0]	0x62 [6:0], 0x63 [7:2]	0x800	Coefficient for channel C
CSC_SCALE[1:0]	0x52 [7:6]	0x01	Scaling for CSC formula
A4[12:0]	0x52 [4:0], 0x53 [7:0]	0x000	Offset for channel A
B4[12:0]	0x59 [4:0], 0x5A [7:0]	0x000	Offset for channel B
C4[12:0]	0x60[4:0], 0x61 [7:0]	0x000	Offset for channel C

CSC_SCALE[1:0], Addr 44 (CP), *Address 0x52[7:6]*

A control to set the CSC coefficient scalar.

Function

CSC_SCALE[1:0]	Description
00	CSC scalar set to 1
01 «	CSC scalar set to 2
10	Reserved. Do not use
11	Reserved. Do not use

RB_CSC_SCALE[1:0], Addr 44 (CP), *Address 0x0B[7:6] (Read Only)*

Readback of CSC scale applied to CSC coefficients

Function

RB_CSC_SCALE[1:0]	Description
xx	Readback value

RB_A1[12:0], Addr 44 (CP), *Address 0x10[4:0]; Address 0x11[7:0] (Read Only)*

Readback of CSC coefficient A1 modified by video adjustment block.

Function

RB_A1[12:0]	Description
xxxxxxxxxxxxx	Readback value

RB_A2[12:0], Addr 44 (CP), *Address 0x0E[1:0]; Address 0x0F[7:0]; Address 0x10[7:5] (Read Only)*

Readback of CSC coefficient A2 modified by video adjustment block.

Function

RB_A2[12:0]	Description
xxxxxxxxxxxxx	Readback value

RB_A3[12:0], Addr 44 (CP), *Address 0x0D[6:0]; Address 0x0E[7:2] (Read Only)*

Readback of CSC coefficient A3 modified by video adjustment block.

Function

RB_A3[12:0]	Description
xxxxxxxxxxxxx	Readback value

RB_A4[12:0], Addr 44 (CP), *Address 0x0B[4:0]; Address 0x0C[7:0] (Read Only)*

Readback of CSC coefficient A4 modified by video adjustment block.

Function

RB_A4[12:0]	Description
xxxxxxxxxxxxx	Readback value

RB_B1[12:0], Addr 44 (CP), *Address 0x17[4:0]; Address 0x18[7:0] (Read Only)*

Readback of CSC coeff B1 modified by video adjustment block

Function

RB_B1[12:0]	Description
xxxxxxxxxxxxx	Readback value

RB_B2[12:0], Addr 44 (CP), *Address 0x15[1:0]; Address 0x16[7:0]; Address 0x17[7:5] (Read Only)*

Readback of CSC coeff B2 modified by video adjustment block.

Function

RB_B2[12:0]	Description
xxxxxxxxxxxxx	Readback value

RB_B3[12:0], Addr 44 (CP), *Address 0x14[6:0]; Address 0x15[7:2] (Read Only)*

Readback of CSC coeff B3 modified by video adjustment block.

Function

RB_B3[12:0]	Description
xxxxxxxxxxxxx	Readback value

RB_B4[12:0], Addr 44 (CP), *Address 0x12[4:0]; Address 0x13[7:0] (Read Only)*

Readback of CSC coefficient B4 modified by video adjustment block.

Function

RB_B4[12:0]	Description
xxxxxxxxxxxxx	Readback value

RB_C1[12:0], Addr 44 (CP), *Address 0x1E[4:0]; Address 0x1F[7:0] (Read Only)*

Readback of CSC coeff C1 modified by video adjustment block.

Function

RB_C1[12:0]	Description
xxxxxxxxxxxxx	Readback value

RB_C2[12:0], Addr 44 (CP), *Address 0x1C[1:0]; Address 0x1D[7:0]; Address 0x1E[7:5] (Read Only)*

Readback of CSC coefficient C2 modified by video adjustment block.

Function

RB_C2[12:0]	Description
xxxxxxxxxxxxx	Readback value

RB_C3[12:0], Addr 44 (CP), *Address 0x1B[6:0]; Address 0x1C[7:2] (Read Only)*

Readback of CSC coefficient C3 modified by video adjustment block.

Function

RB_C3[12:0]	Description
xxxxxxxxxxxxx	Readback value

RB_C4[12:0], Addr 44 (CP), *Address 0x19[4:0]; Address 0x1A[7:0] (Read Only)*

Readback of CSC coefficient C4 modified by video adjustment block

Function

RB_C4[12:0]	Description
xxxxxxxxxxxxx	Readback value

9.4.5.1 CSC Manual Programming

The equations performed by the CP CSC are as follows:

$$Out\_A = \left[In\_A * \frac{A1[12:0]}{4096} + In\_B * \frac{A2[12:0]}{4096} + In\_C * \frac{A3[12:0]}{4096} + A4[12:0] \right] * 2^{CSC\_scale}$$

Equation 8 CSC Channel A

$$Out\_B = \left[In\_A * \frac{B1[12:0]}{4096} + In\_B * \frac{B2[12:0]}{4096} + In\_C * \frac{B3[12:0]}{4096} + B4[12:0] \right] * 2^{CSC\_scale}$$

Equation 9 CSC Channel B

$$Out\_C = \left[In\_A * \frac{C1[12:0]}{4096} + In\_B * \frac{C2[12:0]}{4096} + In\_C * \frac{C3[12:0]}{4096} + C4[12:0] \right] * 2^{CSC\_scale}$$

Equation 10: CSC Channel C

As can be seen from Equation 8, Equation 9 and Equation 10, the A1, A2, A3; B1, B2, B3; and C1, C2, C3 coefficients are used to scale the primary inputs. The values of A4, B4, and C4 are added as offsets. The CSC_SCALE[1:0] bits allows the user to implement conversion formulae in which the coefficients exceed the standard range of [-4095/4096 .. 4095/4096]. The overall range of the CSC is [0..1] for unipolar signals (for example, Y, R, G, and B) and [-0.5..+0.5] for bipolar signals (for example, Pr and Pb).

Note: The bipolar signals must be offset to mid range, for example, 2048.

To arrive at programming values from typical formulas, the following steps are performed:

1. Determine the dynamic range of the equation.
   The dynamic range of the CSC is [0 … 1] or [-0.5 … +0.5]. Equations with a gain larger than 1 need to be scaled back. Errors in the gain can be compensated for in the gain stages of the follow on blocks. ➔ Scale the equations, if necessary.

2. Check the value of each coefficient. The coefficients can only be programmed in the range [-0.99 … +0.99].

3. To support larger coefficients, the CSC_SCALE[1:0] function should be used. ➔

4. Determine the setting for CSC_SCALE[1:0] and adjust coefficients, if necessary.

5. Program the coefficient values. Convert the float point coefficients into 12-bit fixed decimal format. Convert into binary format, using twos complement for negative values.
   ➔ Program A1 .. A3, B1 .. B3, C1 .. C3.

6. Program the offset values. Depending on the type of CSC, offsets may have to be used.
   ➔ Program A4, B4, C4.

9.4.5.2 CSC Example

The following set of equations gives an example of a conversion from a gamma corrected RGB signal into a YCbCr color space signal.

$$Out\_A = \left[In\_A * \frac{A1[12:0]}{4096} + In\_B * \frac{A2[12:0]}{4096} + In\_C * \frac{A3[12:0]}{4096} + A4[12:0] \right] * 2^{CSC\_scale}$$

$$Out\_B = \left[In\_A * \frac{B1[12:0]}{4096} + In\_B * \frac{B2[12:0]}{4096} + In\_C * \frac{B3[12:0]}{4096} + B4[12:0] \right] * 2^{CSC\_scale}$$

$$Out\_C = \left[In\_A * \frac{C1[12:0]}{4096} + In\_B * \frac{C2[12:0]}{4096} + In\_C * \frac{C3[12:0]}{4096} + C4[12:0] \right] * 2^{CSC\_scale}$$

Note: The original equations give offset values of 128 for the Pr and Pb components. The value of 128 equates to half the range on an 8-bit system. It must be noted that the CSC operates on a 12-bit range. The offsets, therefore, must be changed from 128 to half the range of a 12-bit system, which equates to 2048.

The maximum range for each equation, that is, each output data path, can only be [0 ... 1] or [-0.5 ... +0.5]. Equations with a larger gain must be scaled back into range. The gain error can be compensated for in the gain stage of the follow on blocks.

The ranges of the three equations are:

Equation	Minimum Value	Maximum Value	Range
Y	0 + 0 + 0 = 0	0.59 + 0.3 + 0.11 = 1	[0 … 1] = 1
Pb	(-0.34) + (-0.17) = -0.51	0.51	[-0.51 … 0.51] = 1.02
Pr	(-0.43) + (-0.08) = -0.51	0.51	[-0.51 … 0.51] = 1.02

As can be seen from this table, the range for the Y component fits into the CSC operating range. However, the Pb and Pr ranges slightly exceed the range. To bring all equations back into the supported range, they should be scaled back by 1/1.02.

If equations fall outside the supported range, overflow or underflow can occur and undesirable wrap around effects (large number overflowing to small ones) can happen.

$$Y = \frac{0.59}{1.02} * G + \frac{0.3}{1.02} * R + \frac{0.11}{1.02} * B = 0.58 * G + 0.29 * R + 0.11 * B$$

$$Pb = \frac{-0.34}{1.02} * G + \frac{-0.17}{1.02} * R + \frac{0.51}{1.02} * B + 2048 = -0.33 * G - 0.17 * R + 0.5 * B + 2048$$

$$\Pr = \frac{-0.43}{1.02} * G + \frac{0.51}{1.02} * R + \frac{-0.08}{1.02} * B + 2048 = -0.42 * G + 0.5 * R - 0.08 * B + 2048$$

Note: The scaling of the dynamic range does not affect the static offset.

Check the value of each coefficient:

The maximum value for each coefficient on its own can only be within the range of -4095/4096 to 4095/4096, which equals [-0.999755859375 .. 0.999755859375]. Values outside this range do not fit into the 12-bit fixed point format used to program the coefficients.

If the value of one or more coefficients after scaling of the overall equation exceeds the supported coefficient range, the CSC_SCALE[1:0] should be set.

With the CSC_SCALE[1:0] set high, all coefficients must be scaled by half, which makes them fit into the given coefficient range. The overall outputs of the CSC are gained up by a fixed value of two, thus compensating for the scaled down coefficients.

In the above example:

Each coefficient on its own is within the range of $\frac{-4095}{4096} \le Coeff \le \frac{4095}{4096}$.

Therefore, all coefficients can be programmed directly and the CSC_SCALE[1:0] bit should be set to 0.

Notes:

- To achieve a coefficient value of 1.0 for any given coefficient, CSC_SCALE should be set high and the coefficient should actually be programmed to a value of 0.5. Otherwise, the largest value would be 4095/4096 = 0.9997, which is not exactly 1. While this value could be interpreted as a 1, it is recommended to use the value of 0.5 and the CSC_SCALE bit for maximum accuracy.
- For very large coefficient values, for example, 2.58, a combination of CSC_SCALE[1:0] and equation scaling should be used.
- Set CSC_SCALE high (2.58/2 = 1.29) and scale the overall equation by slightly more than 1.28 (coefficient falls within the supported range of [-0.999 … +0.999]).

9.4.6 CSC in Pass-through Mode

It is possible to configure the CP CSC in a pass-through mode. In this mode, the CP CSC is used but does not alter the data it processes.

The CP CSC pass-through mode is obtained using the following settings:

1. Set MAN_CP_CSC_EN to 1'b1
2. CSC_COEFF_SEL[3:0] to 4'b0000
3. Leave the following registers from the CP Map at the default:
   - CSC_SCALE = 1 (default value)
   - A4 = A3 = A2 = 0x000 (default value)
   - B4 = B3 = B1 = 0x000 (default value)
   - C4 = C2 = C1 = 0x000 (default value)
   - A1 = B2 = C3 = 0x800 (default value)

Note: The DPP CSC is always in pass-through mode unless the ADV7842 is processing an RGB input, outputting this input in the RGB color space and VID_ADJ_EN is enabled.

9.5 COLOR CONTROLS

The ADV7842 has a color control feature that can adjust the brightness, contrast, saturation, and hue properties.

VID_ADJ_EN, Addr 44 (CP), *Address 0x3E[7]*

Video Adjustment Enable. This control selects whether or not the color controls feature is enabled. The color controls feature is configured via the parameters CP_CONTRAST[7:0], CP_SATURATION[7:0], CP_BRIGHTNESS[7:0] and CP_HUE[7:0]. The CP CSC must also be enabled for the color controls to be effective.

Function

VID_ADJ_EN	Description
0 «	Disable color controls.
1	Enable color controls.

CP_CONTRAST[7:0], Addr 44 (CP), *Address 0x3A[7:0]*

A control to set the contrast. This field is an unsigned value represented in a 1.7 binary format. The MSB represents the integer part of the contrast value which is either 0 or 1. The seven LSBs represents the fractional part of the contrast value. The fractional part has the range [0 to 0.99]. This control is functional if VID_ADJ_EN is set to 1.

CP_SATURATION[7:0], Addr 44 (CP), *Address 0x3B[7:0]*

A control to set the saturation. This field is an unsigned value represented in a 1.7 binary format. The MSB represents the integer part of the contrast value which is either 0 or 1. The seven LSBs represent the fractional part of the saturation value. The fractional part has a [0 to 0.99] range. This control is functional if VID_ADJ_EN is set to 1.

CP_BRIGHTNESS[7:0], Addr 44 (CP), *Address 0x3C[7:0]*

A control to set the brightness. This field is a signed value. The effective brightness value applied to the Luma is obtained by multiplying the programmed value CP_BRIGHTNESS with a gain of 4. The brightness applied to the Luma has a range of [-512 to 508]. This control is functional if VID_ADJ_EN is set to 1.

Function

CP_BRIGHTNESS[7:0]	Description
00000000 «	The offset applied to the Luma is 0 .
01111111	The offset applied to the Luma is 508d. This value corresponds to the brightness setting.
11111111	The offset applied to the Luma is -512d. This value corresponds to the darkest setting.

CP_HUE[7:0], Addr 44 (CP), *Address 0x3D[7:0]*

A control to set the hue. This register a represent an unsigned value which provides hue adjustment. The effective hue applied to the Chroma is [(CP_HUE[7:0] * 180)/256 - 90]. The range of the effective hue applied to the Chroma is [-90° to 90°]. This control is functional if VID_ADJ_EN is set to 1.

Function

CP_HUE[7:0]	Description
00000000 «	A hue of -90° is applied to the Chroma
00001111	A hue of 0° is applied to the Chroma
11111111	A hue of 90° is applied to the Chroma

The effective hue applied to the Chroma is given by Equation 11. The range of the effective hue applied to the Chroma is [-90° to 90°].

$$HUE = \frac{180 \cdot CP\_HUE[7:0]}{256 - 90}$$

Equation 11: Hue in Degree Unit Applied to Chroma via CP_HUE[7:0] Control

10 COMPONENT PROCESSOR

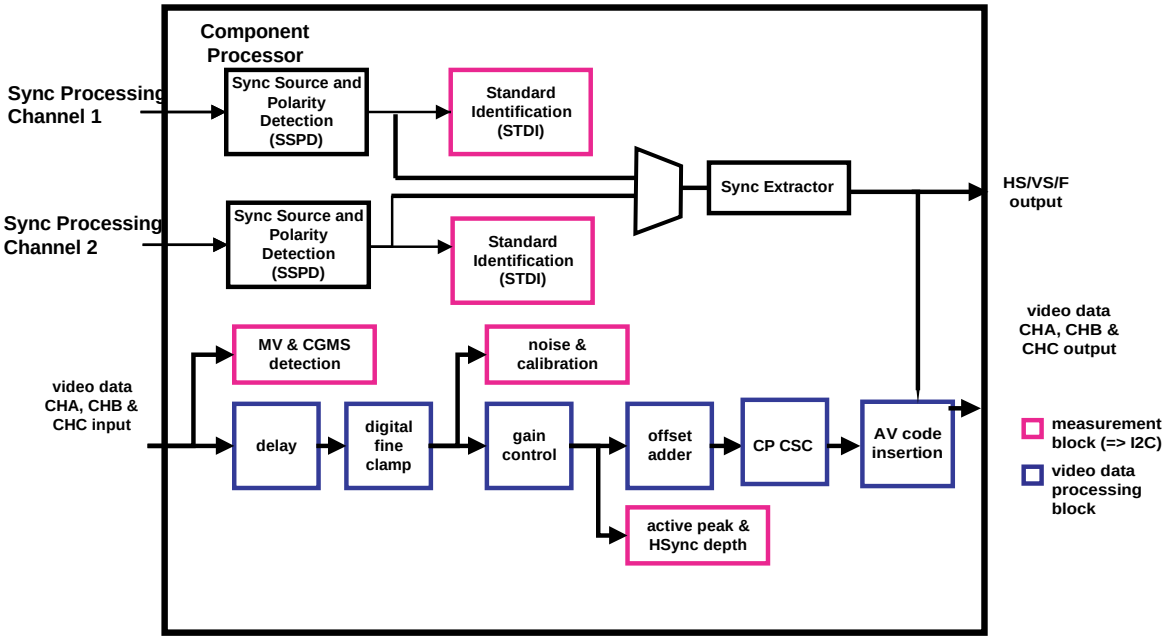


Figure 108: Component Processor Block Diagram

10.1 INTRODUCTION TO COMPONENT PROCESSOR

A simplified block diagram of the CP on the ADV7842 is shown in Figure 108. Data is supplied to the CP from the Data Preprocessor (DPP). The CP circuitry is activated under the control of PRIM_MODE[3:0] and VID_STD[5:0].

The CP is activated for the following modes of operation:

- GR modes: PC graphic-based signals in RGB format
- HD modes: high definition video signals in YPbPr/RGB format
- PR mode: progressive scan video signals in YPbPr/RGB format, for example, 525p and 625p
- SD modes: component standard definition in YPbPr/RGB format, for example, 525i and 625i
- HDMI modes: digital video data from the HDMI receiver block

The CP performs the following functions:

- Digital fine clamping of the video signal
- Manual and automatic gain control
- Manual offset correction
- Saturation
- Insertion of timing codes and blanking data

The CP has the following capabilities:

- Generates HSync, VSync, FIELD, and Data Enable (DE) timing reference outputs
- Detects the source from which the video is to be synchronized
- Measures noise and calibration levels
- Measures the depth of the horizontal synchronization pulse used for AGC
- Detects the presence of Macrovision encoded signals
- Color space conversion
- Color control adjustment

10.2 CLAMP OPERATION

For analog signals that enter the CP block, there are two clamp methods applied to the video signal:

- An analog voltage clamp block prior to the ADCs
- A digital fine clamp that operates after the DPP block

The analog voltage clamp signal operates on the input video prior to digitization. Figure 109 shows the position within the active video lines where the voltage clamp switches on. The position of the window is changed automatically (depending on PRIM_MODE[3:0] and VID_STD[5:0]) to suit the video standard in question.

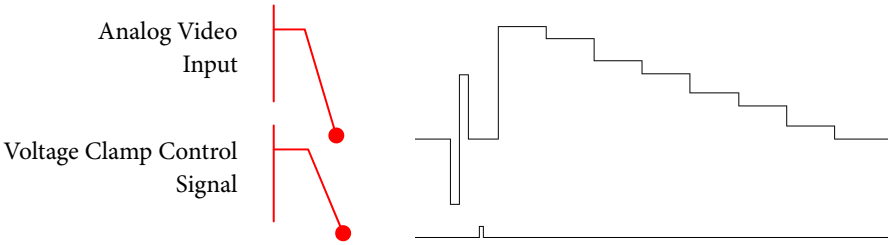


Figure 109: Position of Voltage Clamp Window

The CP contains a digital fine clamp block. Its main purposes are:

1. To compensate for variations of the voltage clamps in the analog domain
2. To allow a clamp to operate even if the input signal is coming from a digital source, for example, external ADC, HDMI/DVI receiver, and so on

The digital fine clamp operates in three separate feedback loops, one for each channel. The incoming video signal level is measured at the back porch. The level error, that is, clamp error, is compensated for by subtracting or adding a digital number to the datastream.

The digital clamp loop can be operated in an automatic or a manual mode with the following options:

- The clamp values for channels B and C can be set manually. This is the recommended mode.
- The clamp value is determined automatically on a line-by-line basis.
- The clamp loops can be frozen. This means that the currently active offsets will no longer be updated but will be applied permanently.

- The clamp value for channel A can be set manually (static value).

Note: The target clamp level for black input is a digital code of 0. This is to facilitate the highest possible signal to noise ratio (SNR). Some interfaces, for example, ITU-R. BT656, require black to correspond to a value other than 0. To facilitate this, there is an additional independent offset adder block after the gain multipliers for which separate fixed offset values can be supplied. Refer to Section 10.4 for additional information.

CLMP_FREEZE Freeze Digital Clamp, CP Map, *Address 0x6C, [5]*

The CLMP_FREEZE bit stops the three digital fine clamp loops for channels A, B, and C from updating. The currently active clamp values are applied continuously. All three loops are affected together; it is not possible to freeze the clamps for the channels individually.

CLMP_FREEZE, Addr 44 (CP), *Address 0x6C[5]*

Stops the digital fine clamp loops for channels A, B and C from updating.

Function

CLMP_FREEZE	Description
0 «	Clamp value updated on every active video line.
1	Clamp loops are stopped and not updated.

To facilitate an external clamp loop for channel A, the internal clamp value determined by the digital fine clamp block can be overridden by a manual value programmed in the I^2C. The two corresponding control values are CLMP_A_MAN and CLMP_A[11:0].

CLMP_A_MAN, Addr 44 (CP), *Address 0x6C[7]*

Manual clamping enable for channel A.

Function

CLMP_A_MAN	Description
0 «	Use the digital fine clamp value determined by the on-chip clamp loop
1	Ignore internal digital fine clamp loop result. Use CLMP_A[11:0]

CLMP_A[11:0], Addr 44 (CP), *Address 0x6C[3:0]; Address 0x6D[7:0]*

Manual clamp value for channel A. This field is an unsigned 12-bit value to be subtracted from the incoming video signal. This value programmed in this register is effective if the CLMP_A_MAN is set to 1. To change the CLMP_A[11:0], the register addresses 0x6C and 0x6D must be updated with the desired clamp value written to in this order and with no other I2C access in between.

Function

CLMP_A[11:0]	Description
0x000 «	minimum range,
	...
0xFFF	maximum range

To facilitate an external clamp loop for channels B and C, the internal clamp value determined by the digital fine clamp block can be overridden by manual values programmed in the CP Map. Both channels B and C are either in manual or automatic mode. There is no individual control for them.

The corresponding control values are CLMP_BC_MAN, CLMP_B[11:0], and CLMP_C[11:0].

CLMP_BC_MAN, Addr 44 (CP), *Address 0x6C[6]*

Manual clamping enable for channel B and C.

Function

CLMP_BC_MAN	Description
0 «	Use the digital fine clamp value determined by the on-chip clamp loop.
1	Ignore internal digital fine clamp loop result. use CLMP_B[11:0] for channel B and CLMP_C[11:0] for channel C.

CLMP_B[11:0], Addr 44 (CP), *Address 0x6E[7:0]; Address 0x6F[7:4]*

Manual clamp value for channel B. This field is an unsigned 12-bit value to be subtracted from the incoming video signal. This value programmed in this register is effective if the CLMP_BC_MAN is set to 1. To change the CLMP_B[11:0], the register addresses 0x6E and 0x6F must be updated with the desired clamp value written to in this order and with no other I2C access in between.

Function

CLMP_B[11:0]	Description
0x000 «	minimum range,
	...
0xFFF	maximum range

CLMP_C[11:0], Addr 44 (CP), *Address 0x6F[3:0]; Address 0x70[7:0]*

Manual clamp value for channel C. This field is an unsigned 12-bit value to be subtracted from the incoming video signal. This value programmed in this register is effective if the CLMP_BC_MAN is set to 1. To change the CLMP_C[11:0], the register addresses 0x6F and 0x70 must be updated with the desired clamp value written to in this order and with no other I2C access in between.

Function

CLMP_C[11:0]	Description
0x000 «	minimum range,
	...
0xFFF	maximum range

CLAMP_AVG_FCTR

The ADV7842 provides a special filter option for the auto clamp mode. The purpose of this filter is to provide a smoothening mechanism when the clamping value for each channel is being changed continuously in significant amounts by the autoclamping mechanism.

The filter is an IIR filter with an effective function of:

$$Y_N = (1\text{-}A) * Y_{N\text{-}1} + A * X_N$$

Where A is the filter coefficient.

The value of A can vary from 1 to 1/32 lines. A value of 1 indicates no filtering of the clamp and is a pass through option for the autoclamp value.

CLAMP_AVG_FCTR[1:0], Addr 44 (CP), *Address 0xC5[7:6]*

A control to set the coefficient A of the IIR filter used for auto clamp mode.
The function transfer is Y[N]=(1-A)*Y[N-1]+A*X[N]

Function

CLAMP_AVG_FCTR[1:0]	Description
00	No filtering, A=1
01	The clamp is averaged over 8 lines. A=1/8
10 «	The clamp is averaged over 16 lines. A=1/16
11	The clamp is averaged over 32 lines. A=1/32

10.3 CP GAIN OPERATION

The digital gain block of the CP consists of three multipliers in the data paths of channel A, B, and C, as well as one single automatic gain control loop. The gain control can be operated in manual or automatic mode.

10.3.1 Features of Manual Gain Control

The gain values for the three channels can be programmed separately via I²C registers. This is the recommended mode.

10.3.2 Features of Automatic Gain Control

The gain value is determined automatically, based on a signal with an embedded horizontal synchronization pulse on channel A. The automatic gain control loop can be frozen, for example, after settling.

The gain value for analog inputs with separate HSync and VSync timing signals, and HDMI receiver inputs are controlled via the OP_656_RANGE bit.

10.3.3 Manual Gain and Automatic Gain Control Selection

Figure 110 shows how the gain is applied to the to the video data processes by the CP section. The following gain configurations are available:

- **Automatic Gain configuration in HDMI Mode**
  This configuration is enabled by setting AGC_MODE_MAN to 0 and by setting the part in HDMI mode via PRIM_MODE[3:0] and VID_STD[5:0]. In that case the gain applied to the video data depends on the input and output range configuration. The input range is set by control register INP_COLOR_SPACE and the read back register HDMI_COLORSPACE[3:0] as per Table 52. The output color space is determined the control bit OP_656_RANGE.

- **Gain Configuration Dependant on the Output Range**
  This configuration is enabled by setting AGC_MODE_MAN to 0 and by setting the part is SDP, COMP or GR mode via PRIM_MODE and VID_STD. In this case the gain applied to the video data processed by the CP core is determined by the control bit OP_656_RANGE.

- **Manual Gain Configuration**
  This configuration is enabled by setting AGC_MODE_MAN to 1 and GAIN_MAN to 1. In this case the gain applied to the video data processed by the CP core is configured via the control registers A_GAIN[9:0], B_GAIN[9:0] and C_GAIN[9:0].

- **Automatic Gain Control Configuration**
  This configuration is enabled by setting AGC_MODE_MAN to 1 and GAIN_MAN to 0.
  The detection block, Synchronization Source and Polarity Detector (SSPD), is used to determine automatically the presence of external digital synchronizations, for example, HSync/VSync, or embedded synchronization. The detection result of the SSPD block is used to enable/disable the automatic gain control mode.
  If SSPD detects the presence of external (that is, digital) synchronization signals, the gain block in the CP core is controlled by the OP_656_RANGE bit because it is assumed that there is no embedded synchronization present and it is, therefore, not

possible to adjust the gain automatically

- If, however, SSPD does not find any external synchronization signal, it concludes that the synchronization must be embedded. This switches the gain block in the CP core into automatic mode (refer to Section 10.8). This function can be disabled using the AGC mode manual enable control, AGC_MODE_MAN, as illustrated in Figure 110.

Table 52. Input Ranges for HDMI Modes

INP_COLOR_SPACE	Input Range
0b0000	16-235
0b0001	0-255
0b0010	16-235
0b0011	16-235
0b0100	0-255
0b0101	0-255
0b0110	0-255
0b0111	0-255
0b1111	16-235 if HDMI_COLORSPACE = 0b000
	0-255 if HDMI_COLORSPACE = 0b001
	16-235 if HDMI_COLORSPACE = 0b010
	16-235 if HDMI_COLORSPACE = 0b011
	0-255 if HDMI_COLORSPACE = 0b100
	0-255 if HDMI_COLORSPACE = 0b101
	0-255 if HDMI_COLORSPACE = 0b110
	0-255 if HDMI_COLORSPACE = 0b111
0b1000 to 0b1110	Reserved

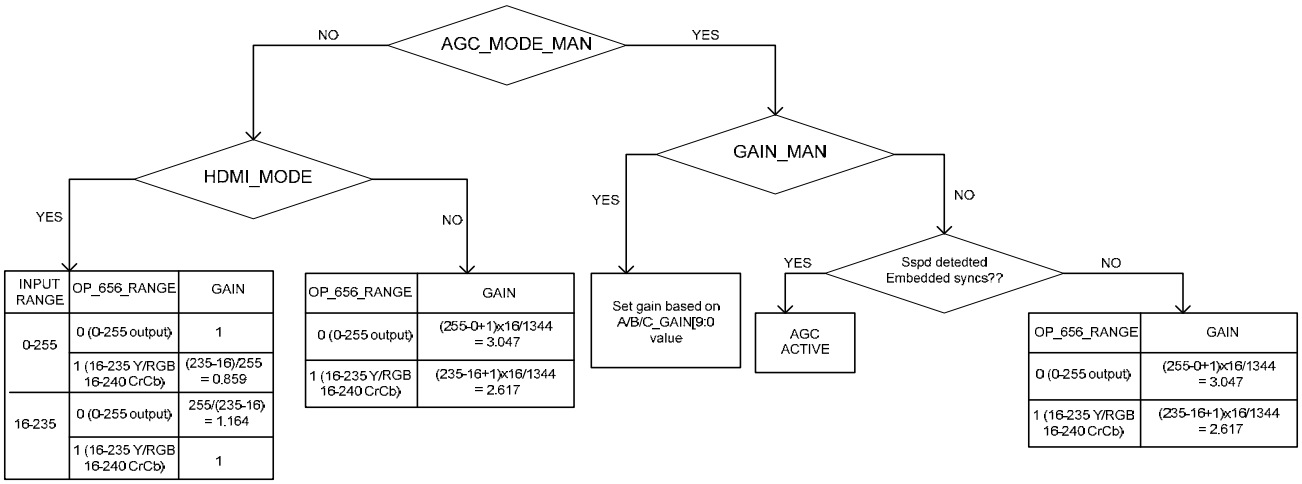


Figure 110: CP Automatic Gain Controls

AGC_MODE_MAN, Addr 44 (CP), *Address 0x73[6]*

A control to set how the gains for all 3 channels is configured

Function

AGC_MODE_MAN	Description
0 «	The gain is dependant on the type of input and OP_656_RANGE
1	Gain operation controlled by GAIN_MAN

10.3.4 Manual Gain Control

The automatic gain control (AGC) can be completely disabled by setting the gain control block into a manual mode. By setting the GAIN_MAN bit, the gain factors for channels A, B, and C are no longer taken from the AGC, but are replaced by three dedicated I^2C registers.

Using these factors with the HSD_FB[9:0] register, it is possible to implement an off-chip AGC if desired. The range for the gain is [0…3.999]. The registers A_GAIN[9:0], B_GAIN[9:0], and C_GAIN are in 2.8 binary format and can be set as shown in Equation 12.

$$X\_GAIN[9:0] = floor(GAIN \times 256)$$

Where:

$0 \le GAIN < 4$

$floor(\ )$ is the floor function that returns the largest integer not greater than its input parameter

X refers to A, B, and C

Equation 12: CP Manual Gain

Example:

Example $Gain_{dec}$	A_GAIN[9:0]
0.5	0x80
0.98887	0xFD
2.5	0x280

GAIN_MAN, Addr 44 (CP), *Address 0x73[7]*

Enables the gain factor to be set by the AGC or manually.

Function

GAIN_MAN	Description
0 «	AGC controls the gain for all three channels,
1	Manual gains are used for all three channels

A_GAIN[9:0], Addr 44 (CP), *Address 0x73[5:0]; Address 0x74[7:4]*

A control to set the manual gain value for channel A.
This register is an unsigned value in a 2.8 binary format. To change A_GAIN[9:0], the register at addresses 0x73 and 0x74 must be written to in this order with no I2C access in between.

Function

A_GAIN[9:0]	Description
0x000	Gain of 0
0x100 «	Unity Gain
0x3FF	Gain of 3.99

B_GAIN[9:0], Addr 44 (CP), *Address 0x74[3:0]; Address 0x75[7:2]*

A control to set the manual gain value for channel B.
This register stores an unsigned value in a 2.8 binary format. To change A_GAIN[9:0], the register at addresses 0x74 and 0x75 must be written to in this order with no I2C access in between.

Function

B_GAIN[9:0]	Description
0x000	Gain of 0
0x100 «	Unity Gain
0x3FF	Gain of 3.99

C_GAIN[9:0], Addr 44 (CP), *Address 0x75[1:0]; Address 0x76[7:0]*

A control to set the manual gain value for channel C. This register stores an unsigned value in a 2.8 binary format. To change C_GAIN[9:0], the registers at addresses 0x75 and 0x76 must be written to in sequence with no I2C access in between.

Function

C_GAIN[9:0]	Description
0x000	Gain of 0
0x100 «	Unity Gain
0x3FF	Gain of 3.99

10.3.5 Manual Gain Filter Mode

The ADV7842 provides a special filter option for the manual gain mode. This is functional only when manual gain is enabled. The purpose of this filter is a smoothing mechanism when the manual gain value is updated continuously by an external system based on either external or readback conditions in the ADV7842. The filter designed is an IIR filter with a transfer function of the form:

$$Y_N = (1-A)*Y_{N-1} + A*X_N$$

where A is the filter coefficient.

The values possible for A can vary from 1 (no filtering) to 1/128K (K = 1024). The value of coefficient A is chosen by programming CP_GAIN_FILT[3:0].

CP_GAIN_FILT[3:0], Addr 44 (CP), *Address 0x84[7:4]*

A control to set the coefficient A of the IIF filter to filter the gain applied to the video signal when the gain is manually set. The value set in this register is effective only when manual gain is enabled. The filter is designed as and IIR filter with a transfer function of the form Y[N]=(1-A)*y[N-1]+A*X[N]

Function

CP_GAIN_FILT[3:0]	Description
0000 «	No filtering, i.e. coefficient A = 1
0001	Coefficient A = 1/128 lines
0010	Coefficient A = 1/256 lines
0011	Coefficient A = 1/512 lines
0100	Coefficient A = 1/1024 lines
0101	Coefficient A = 1/2048 lines
0110	Coefficient A = 1/4096 lines
0111	Coefficient A = 1/8192 lines
1000	Coefficient A = 1/16 384 lines
1001	Coefficient A = 1/32 768 lines
1010	Coefficient A = 1/65 536 lines
1011	Coefficient A = 1/131 072 lines
All other values	Reserved. Do not use.

10.3.6 Automatic Gain Control

The AGC of the CP takes measurements of the signal on channel A and determines an appropriate gain value for all three channels. For the block to operate, it is necessary that a signal with an embedded synchronization pulse is fed through to channel A, for example, Y or G. The AGC measures the depth of this synchronization pulse and compares it against a target value. The HSD_CHA[9:0] readback register is used to determine if there is a synchronization pulse on the data. If no synchronization pulse is found, AGC cannot work and the manual gain control should be enabled.

The target value for the AGC can come from three sources. There are two predefined values of 300 mV and 286 mV (the HS_NORM bit is used to decide between the two values) and there is the option of setting an arbitrary target value by setting the AGC_TAR_MAN bit, which enables the usage, and AGC_TAR[9:0], which sets the arbitrary target level.

In some applications, it is desirable to use the AGC to gain the signal to a smaller range, use the Offset block to preserve the synchronizations (by lifting the entire video signal up), and thus output the full digitized waveform (including synchronizations) within the 12-bit output range. For this application, the AGC_TAR[9:0] value is very important. For more information, refer to Section 10.4.

$$AGC\_TAR[9:0] = (Code_{White} - Code_{Black}) \bullet \frac{SyncHeight_{mV}}{VideoHeight_{mV}}$$

Equation 13: CP AGC Target Value

Note: The 12-bit target code for white is nominally 940, the target code for black is 64.

Examples:

$$AGC\_TAR_{HSync=286mV} = (940-64) \bullet \frac{286mV}{714mV} = 351_{dec}$$

$$AGC\_TAR_{HSync=300mV} = (940-64) \bullet \frac{300mV}{700mV} = 375_{dec}$$

An error signal is derived from the comparison of the measured synchronization depth and the target value. The error signal is weighted by a factor that allows different response times to be selected (AGC_TIM[2:0] is used to select different time constants). The resulting gain value is applied to all three channels A, B, and C.

The AGC_FREEZE bit allows the AGC loop to be stopped, that is, frozen. If frozen, the currently active gain is no longer updated but is applied continuously to all three datastreams.

HS_NORM, Addr 44 (CP), *Address 0x71[3]*

Nominal Hsync Depth Selection.

Function

HS_NORM	Description
0 «	The AGC target scales the video as per 300 mV horizontal synchronization depth
1	The AGC target scales the video as per 286 mV horizontal synchronization depth

AGC_TAR_MAN, Addr 44 (CP), *Address 0x71[5]*

Manual Target Level Enable.

Function

AGC_TAR_MAN	Description
0 «	The AGC operates based on a 300 mV or 286 mV horizontal synchronization depth. Use HS_NORM to select between the two.
1	AGC operates based on AGC_TAR[9:0].

AGC_TAR[9:0], Addr 44 (CP), *Address 0x71[7:6]; Address 0x72[7:0]*

Manual AGC Target Value Enable. The register is used to set the target value for horizontal synchronization depth after gain has been applied. The field represents an unsigned value.
Manual AGC Target Value. See also description of AGC_TAR_MAN, AGC_FREEZE, AGC_TIM.

Function

AGC_TAR[9:0]	Description
0x000 «	minimum range,
	…
0x3FF	maximum range

AGC_FREEZE, Addr 44 (CP), *Address 0x71[4]*

AGC Freeze Enable.

Function

AGC_FREEZE	Description
0 «	The AGC loop is operational
1	The AGC loop is frozen and not updated further. the last gain value becomes static.

AGC_TIM[2:0], Addr 44 (CP), *Address 0x71[2:0]*

AGC Time Constant Selection.

Function

AGC_TIM[2:0]	Description
000 «	100 lines
001	1 frame
010	0.5 sec
011	1 sec
100	2 sec
101	3 sec
110	5 sec
111	7 sec

10.3.6.1 Readback Signals from AGC Block

The following readback signals are provided:

- Presently used gain value can be read back through CP_AGC_GAIN[9:0]
- Depth of the synchronization pulse on channel A (before gaining) through HSD_CHA[9:0]
- Depth of the synchronization pulse on channel A (after gaining) through HSD_FB[11:0]
- Depth of the synchronization pulse on channel B (before gaining) through HSD_CHB[9:0]
- Depth of the synchronization pulse on channel C (before gaining) through HSD_CHC[9:0]

Notes:

- HSD_FB[11:0] is provided to allow an off-chip AGC loop to be implemented in a feedback architecture.
- HSD_CHA, HSD_CHB, and HSD_CHC[9:0] are provided to allow the user in GR modes to find out if all three channels have synchronization pulses on them. If the input RGB has a synchronization pulse only on the Green channel and the CSC is used to convert RGB to YPbPr levels, the synchronization depth on Y will be too shallow (compare with conversion formula RGB to YPbPr). AGC_TAR[9:0] must be used to enable proper output levels after the AGC.
- The HSD_CHA[9:0] register information is also used to figure out if an AGC function is possible. Without a proper synchronization pulse on the data in channel A, no AGC loop can work and manual gain control should be used.

CP_AGC_GAIN[9:0], Addr 44 (CP), *Address 0xE0[1:0]; Address 0xE1[7:0] (Read Only)*

A readback value of the gain used gain on the data of channel A. The value stored in this register has is in a 1.9 binary format and composed of one integer and nine fractional bits.

Function

CP_AGC_GAIN[9:0]	Description
xxxxxxxxxx	Readback value of the gain

HSD_CHA[9:0], Addr 44 (CP), *Address 0xE7[1:0]; Address 0xE8[7:0] (Read Only)*

A readback for the measured value of the HSync depth on channel A before the gain multiplier.
The value is presented in 1.9 binary format.

Function

HSD_CHA[9:0]	Description
xxxxxxxxxx	Readback for measured value of the HSync depth on channel A

HSD_CHB[9:0], Addr 44 (CP), *Address 0xE7[3:2]; Address 0xE9[7:0] (Read Only)*

A readback for the measured value of the HSync depth on channel B before the gain multiplier.
The value is presented in 1.9 binary format.

Function

HSD_CHB[9:0]	Description
xxxxxxxxxx	Readback for measured value of the HSync depth on channel B

HSD_CHC[9:0], Addr 44 (CP), *Address 0xE7[5:4]; Address 0xEA[7:0] (Read Only)*

A readback for the measured value of the HSync depth on channel C before the gain multiplier.
The value is presented in 1.9 binary format.

Function

HSD_CHC[9:0]	Description
xxxxxxxxxx	Readback for measured value of the HSync depth on channel C

HSD_FB[11:0], Addr 44 (CP), *Address 0xEB[3:0]; Address 0xEC[7:0] (Read Only)*

A readback for the measured value of HSync depth on channel A, after gain multiplier, for external feedback loop.
The value is presented in twos complement form. This means that only a standard adder is needed to subtract the actual HSync depth (as per HSD_FB) from a nominal value, as the HSD_FB value is already in negative format.

Function

HSD_FB[11:0]	Description
xxxxxxxxxxxx	Readback value

OP_656_RANGE, Addr 40 (IO), *Address 0x02[2]*

A control to set the output range of the digital data. It also automatically sets the gain setting, the offset setting, and the data saturator setting.

Function

OP_656_RANGE	Description
0 «	Enables full output range (0 to 255)
1	Enables limited output range (16 to 235)

The settings of OP_656_RANGE applied depend on the type of video signal and whether the signal is routed from the analog front end or from the HDMI receiver. Refer to Table 53 and Table 54.

Table 53: OP_656_RANGE Description for HDMI Receiver Input Mode

Input Range	OP_656_RANGE	Gain
0 to 255	0 (0 to 255 output)	1
	1 (16 to 235 RGB output, 16 to 240 CrCb output)	(235-16)/255 = 0.859
16 to 235	0 (0 to 255 output)	255/(235-16) = 1.164
	1 (16 to 235 RGB output, 16 to 240 CrCb output)	1

Table 54: OP_656_RANGE Description for Analog Front End Input Mode

OP_656_RANGE	Gain
0 (0 to 255 output)	(255-0+1) x 16/1792 = 2.29
1 (16 to 235 RGB output,	(235-16+1) x 16/1792 = 1.96

OP_656_RANGE	Gain
16 to 240 CrCb output)	

ALT_DATA_SAT, Addr 40 (IO), *Address 0x02[0]*

A control to disable the data saturator that limits the output range independently of OP_656_RANGE. This bit is used to support extended data range modes.

Function

ALT_DATA_SAT	Description
0 «	Data saturator enabled or disabled according to OP_656_RANGE setting.
1	Reverses OP_656_RANGE decision to enable or disable the data saturator

ALT_SAT_UV_MAN, Addr 44 (CP), *Address 0x3E[1]*

U and V Saturation Range Control

Function

ALT_SAT_UV_MAN	Description
0 «	The range of the saturator on the Cr and the Cb channels are determined by OP_656_RANGE and ALT_DATA_SAT.
1	The range of the saturator on the Cr and the Cb channels are determined by ALT_SAT_UV if either OP_656_RANGE or ALT_DATA_SAT is set to 0.

ALT_SAT_UV, Addr 44 (CP), *Address 0x3E[0]*

Cr and Cb Saturation Range. Refer to ALT_SAT_UV_MAN for additional detail.

Function

ALT_SAT_UV	Description
0 «	The range of the saturators on channels Cr and Cb is 15-to-235.
1	The range of the saturators on channels Cr and Cb is 16-to-240.

CP Peak Active Video Readback

The ADV7842 provides circuitry that monitors the active CP video on a field basis and records the largest value encountered during this time. It is intended to be used in a peak-white type AGC for signals that do not have an embedded horizontal synchronization pulse, and to provide feedback on the accurate function of the built-in AGC loop.

The ADV7842 itself does **not** provide a peak-white AGC. It merely monitors the input signal for the largest data value encountered in each of the three channels, and presents those three values for readback via the I^2C. The values are given in an unsigned format. There is no averaging or filtering before the peak detection.

Notes:

- The measurement is taken on a field basis (from one vertical synchronization to the next). The read out at any time refers to the previous field, not necessarily the current one.
- The tap-off point for the measurement is right after the gain multipliers. This means that clamping and AGC/manual gain have an effect on the results.
- The peak video readback is calculated according to Equation 14.

Peak active video readback value =
(Peak video ampl – Clamp level) * VtoCode * GAIN * (1/8)

Equation 14: Peak Active Video Readback Value

Where:

VtoCode – Voltage to ADC code conversion; for ADV7842 it is 4096/1.111V

GAIN – is an: AGC-gain or manual gain set by user or automatic gain in non embedded-sync-mode (depending on the mode used)

PKV_CHA[9:0], Addr 44 (CP), *Address 0xED[5:4]; Address 0xEE[7:0] (Read Only)*

Maximum signal level measured during the active video on channel A.

Function

PKV_CHA[9:0]	Description
xxxxxxxxxx	Readback value

PKV_CHB[9:0], Addr 44 (CP), *Address 0xED[3:2]; Address 0xEF[7:0] (Read Only)*

Maximum signal level measured during the active video on channel B.

Function

PKV_CHB[9:0]	Description
xxxxxxxxxx	Readback value

PKV_CHC[9:0], Addr 44 (CP), *Address 0xED[1:0]; Address 0xF0[7:0] (Read Only)*

Maximum signal level measured during the active video on channel C.

Function

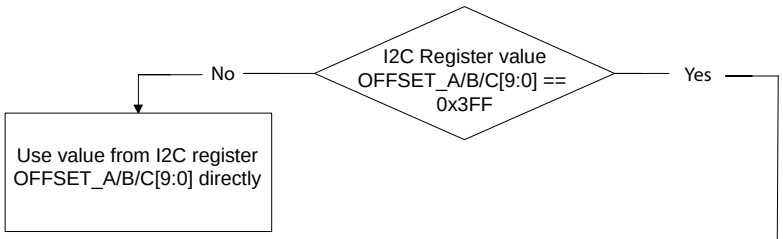
PKV_CHC[9:0]	Description
xxxxxxxxxx	Readback value

10.4 CP OFFSET BLOCK

The offset block consists of three independent adders, one for each channel. Using the registers, A_OFFSET[9:0], B_OFFSET[9:0], and C_OFFSET[9:0] a fixed offset value can be added to the data. The actual offset used can come from two different sources:

1. The ADV7842 includes an automatic selection of the offset value, dependent on the CSC mode that is programmed by the user. The AGC_TAR_MAN and OP_656_RANGE bits are used to derive offset values.
2. A manual, user defined value can be programmed.

When the offset registers (A_OFFSET[9:0], B_OFFSET[9:0], and C_OFFSET[9:0]) contain the value 0x3FF (reset default), the offset used is determined using the automatic selection process. For any other value in the offset registers, the automatic selection is disabled and the user-programmed offset value is applied directly to the video. Refer to the flowchart in Figure 111.



Channel	OFFSET			
	RGB_OUT = 1		RGB_OUT = 0	
	OP_656_RANGE = 1	OP_656_RANGE = 0	OP_656_RANGE = 1	OP_656_RANGE = 0
A	64	0	64	0
B	64	0	512	512
C	64	0	512	512

Figure 111: Channel A, B, and C Automatic Value Selection

In some applications, it is desirable to use the AGC to gain the signal to a smaller range, then use the Offset block to preserve the synchronizations (by lifting the whole video signal up), and thus output the full digitized waveform (including synchronizations) within the 10-bit output range. For this application, the offset values are very important. Refer to the description of AGC_TAR_MAN for additional information.

For RGB type output data, the three offset values should be programmed to 0 or 64 (desired code output for black video). For YPbPr type output data, A_OFFSET[9:0] should be set to 64 (desired code for black); B_OFFSET[9:0] and C_OFFSET[9:0] (for Pr and Pb) are typically set to 512 (mid range).

Notes:

- Adding an excessive offset onto the data will result in clipping of the signal.
- The offset value can only be positive; it is an unsigned number.
- ADV7842 employs sequencers for the offset values that prohibit intermediate wrong values to be applied.
- The I²C sequencer treats the three offset values as separate entities. To update all three offset values, a single sweep of I²C writes to the CP Map, registers 0x77, 0x78, 0x79, and 0x7A is sufficient.

A_OFFSET[9:0], Addr 44 (CP), *Address 0x77[5:0]; Address 0x78[7:4]*

A control to set the manual offset for channel A.
This field stores an unsigned value. To change A_OFFSET[9:0], the register addresses 0x77 and 0x78 must be written to in this order with no I2C access in between.

Function

A_OFFSET[9:0]	Description
0x3FF «	Auto Offset to Ch A ,
Any other value	Ch A offset

B_OFFSET[9:0], Addr 44 (CP), *Address 0x78[3:0]; Address 0x79[7:2]*

A control to set the manual offset for channel B.
This field stores an unsigned value. To change B_OFFSET[9:0], the register addresses 0x78 and 0x79 must be written to in this order with no I2C access in between.

Function

B_OFFSET[9:0]	Description
0x3FF «	Auto Offset to Ch B ,
Any other value	Ch B offset.

C_OFFSET[9:0], Addr 44 (CP), *Address 0x79[1:0]; Address 0x7A[7:0]*

A control to set the manual offset for channel C.
This field stores an unsigned value. To change C_OFFSET[9:0], the register addresses 0x79 and 0x7A must be written to in this order with no I2C access in between.

Function

C_OFFSET[9:0]	Description
0x3FF «	Auto Offset to Ch C
Any other value	Ch C offset.

10.5 AV CODE BLOCK

The AV code block is used to insert AV codes into the video data stream. The codes follow the standards outlined in ITU-R BT.656-4.

The following functions are supported by this block:

- AV code insertion can be enabled or disabled.
- Data between the end of active video (EAV) and the start of active video (SAV) can be blanked, for example, overwritten with default values. This function can be enabled or disabled. In addition, the default blanking value can be set for RGB or YPbPr.
- AV codes can be output on all channels or spread across the Y and PrPb buses for 20-bit output modes.
- F and V bits within the codes can be inserted directly or can be inverted before insertion.
- The position of the codes within the data stream (timing of the insertion) can be set to a default or can be slaved off the signal from the selected HS input pin.

The insertion point for the AV codes is predetermined by default and is adjusted automatically to suit the current video standard as per the PRIM_MODE[3:0] and VID_STD[5:0] settings. To cater for nonstandard signals, however, the AV code insertion point can also be taken off the HSync signal before it goes to the selected HS input pin. This gives the user great flexibility since the HSync signal position can be programmed to quite a wide range with LLC accuracy.

AVCODE_INSERT_EN, Addr 40 (IO), *Address 0x05[2]*

A control to select AV code insertion into the data stream

Function

AVCODE_INSERT_EN	Description
0	Does not insert AV codes into data stream
1 «	Inserts AV codes into data stream

AV_POS_SEL, Addr 44 (CP), *Address 0x7B[2]*

A control to select AV codes position

Function

AV_POS_SEL	Description
0	SAV code at HS falling edge and EAV code at HS rising edge.
1 «	Uses predetermined (default) positions for AV codes.

AV_INV_V, Addr 44 (CP), *Address 0x7B[6]*

A control to invert V bit in AV codes.

Function

AV_INV_V	Description
0 «	Do not invert V bit polarity before inserting it into the AV code,
1	Invert V bit polarity before inserting it into the AV code

AV_INV_F, Addr 44 (CP), *Address 0x7B[7]*

A control to invert the F bit in the AV codes.

Function

AV_INV_F	Description
0 «	Inserts the F bit with default polarity,
1	Inverts the F bit before inserting it into the AV code

DATA_BLANK_EN, Addr 40 (IO), *Address 0x05[3]*

A control to blank data during video blanking sections.

Function

DATA_BLANK_EN	Description
0	Do not blank data during horizontal and vertical blanking periods.
1 «	Blank data during horizontal and vertical blanking periods.

GR_AV_BL_EN, Addr 44 (CP), *Address 0x81[4]*

A control to enable the insertion of data blanking and AV codes for auto-graphics mode.

Function

GR_AV_BL_EN	Description
0 «	Data blanking and AV code insertion for auto graphics mode disabled.
1	Data blanking and AV code insertion for auto graphics mode enabled.

DE_WITH_AVCODE, Addr 44 (CP), *Address 0x7B[0]*

A control to insert AV codes in relation to the DE output signal

Function

DE_WITH_AVCODE	Description
0	AV codes locked to default values. DE position can be moved independently of AV codes.
1 «	Inserted AV codes moves in relation to DE position change.

REPL_AV_CODE, Addr 40 (IO), *Address 0x05[1]*

A control to select the duplication of the AV codes and insertion on all data channels of the output data stream

Function

REPL_AV_CODE	Description
0 «	Outputs complete SAV/EAV codes on all Channels, Channel A, Channel B and Channel C.
1	Spreads AV code across the three channels. Channel B and Channel C contain the first two ten bit words, 0x3FF and 0x000. Channel A contains the final two ten bit words 0x00 and 0xXYZ.

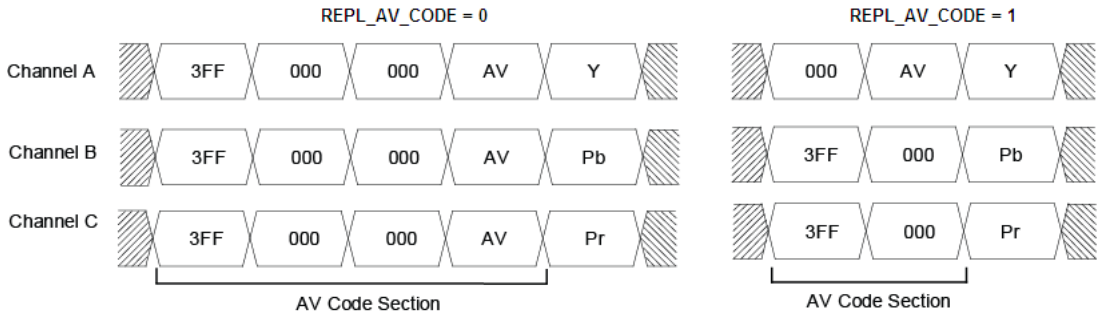


Figure 112: AV Code Output Options (CP)

SWAP_SPLIT_AV, Addr 44 (CP), *Address 0xC9[2]*

A control to swap the Luma and Chroma AV codes in DDR modes

Function

SWAP_SPLIT_AV	Description
0	Swap the Luma and Chroma AV codes in DDR mode
1 «	Do not swap the Luma and Chroma AV codes in DDR mode

10.6 CP DATA PATH FOR ANALOG AND HDMI MODES

Figure 113 to
Figure 117 depict the data path of the video for both analog and HDMI modes. These figures depict the gains and offsets applied when using the automatic control, OP_656_RANGE, and the manual options for setting the clamp level, gain, and offset.

The I²C settings are detailed in Table 55 for use when processing extended range video signals with 'blacker than black' and/or 'whiter than white' video levels.

Table 55: Settings Required to Support Extended Range Video Input

I²C Setting/Mode	Analog Modes	HDMI Mode YUV	HDMI Mode RGB [0-255]	HDMI Mode RGB [16-235]
OP_656_RANGE	1	1	0	1
ALT_DATA_SAT	1	1	0	1

10.6.1 Pregain Block

To compensate for signal attenuation in the analog front end of the ADV7842 and input buffer gain, a pregain block is provided in the CP path. The pregain block is controlled by CP_MODE_GAIN_ADJ[7:0], which represents an unsigned value in a 1.7 binary format. The range of CP_MODE_GAIN_ADJ[7:0] is 0 to 1.99.

The MSB of CP_MODE_GAIN_ADJ[7:0] represents the integer part of the pregain value while the 7 LSBs represents the fractional part of the pregain value.

CP_MODE_GAIN_ADJ[7:0], Addr 44 (CP), *Address 0x40[7:0]*

Pregain adjustment to compensate for the gain of the Analog Front End. This register stores a value in a 1.7 binary format.

Function

CP_MODE_GAIN_ADJ[7:0]	Description
0xxxxxxx	Gain of (0 + (xxxxxxx / 128))
01011100 «	Default pregain (pregain of 0.718)
1xxxxxxx	Gain of (1 + (xxxxxxx / 128))

CP_MODE_GAIN_ADJ_EN, Addr 44 (CP), *Address 0x3E[2]*

A control to enable pregain

Function

CP_MODE_GAIN_ADJ_EN	Description
0	The pregain block is bypassed
1 «	The pregain block is enabled

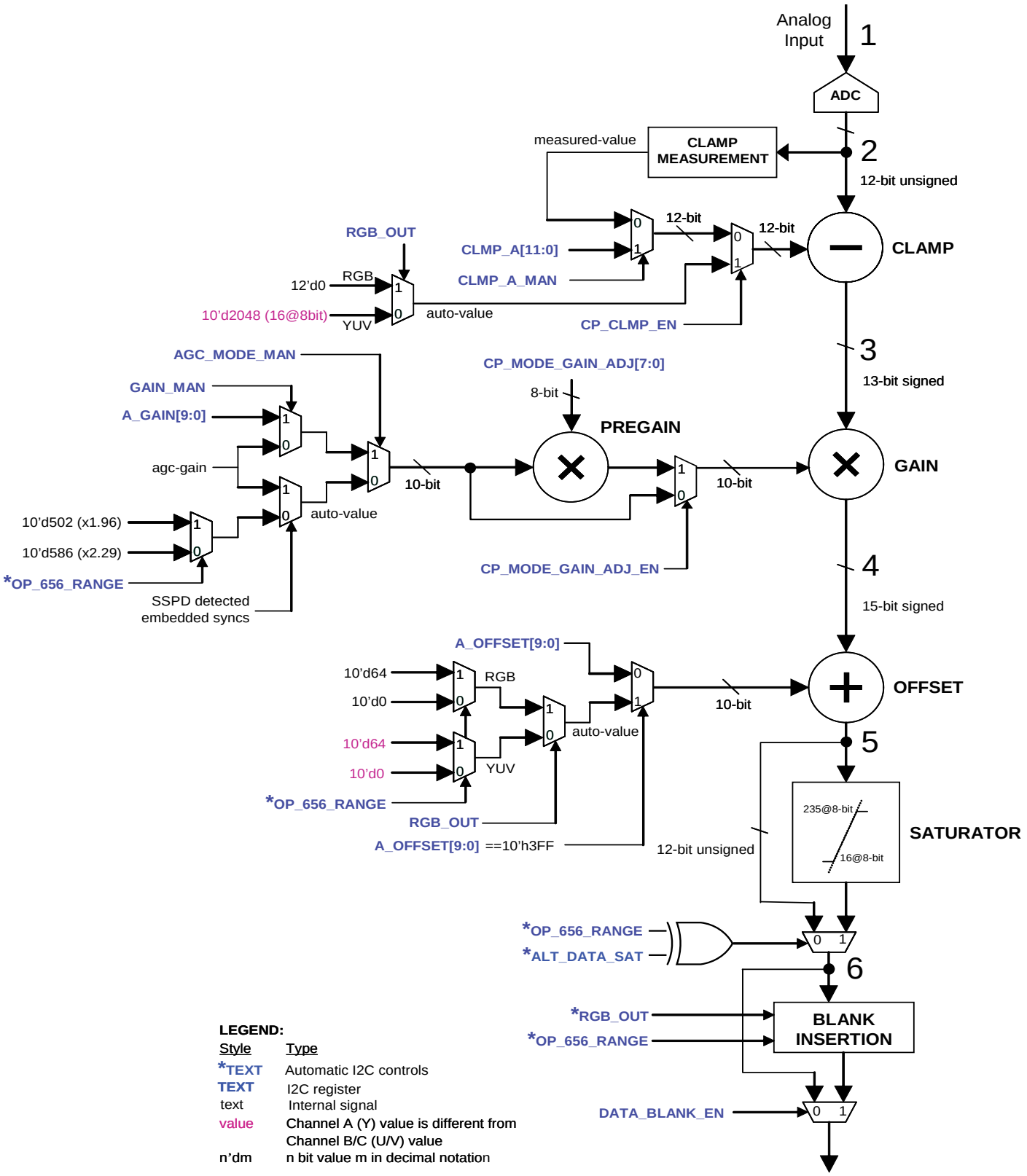


Figure 113: CP DATA Path Channel A (Y) for Analog Mode

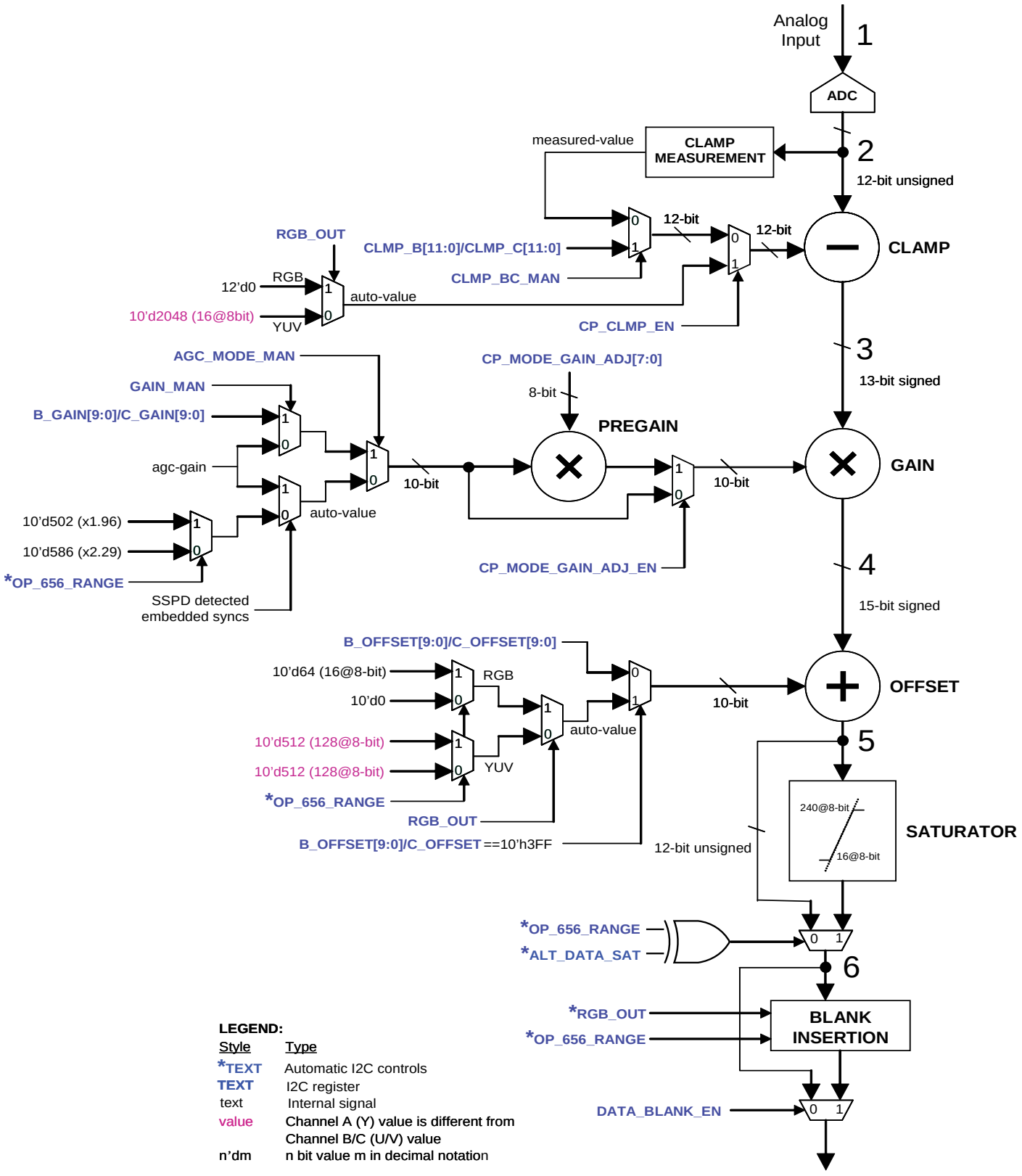


Figure 114: CP Data Path Channel B/C (UV) for Analog Mode

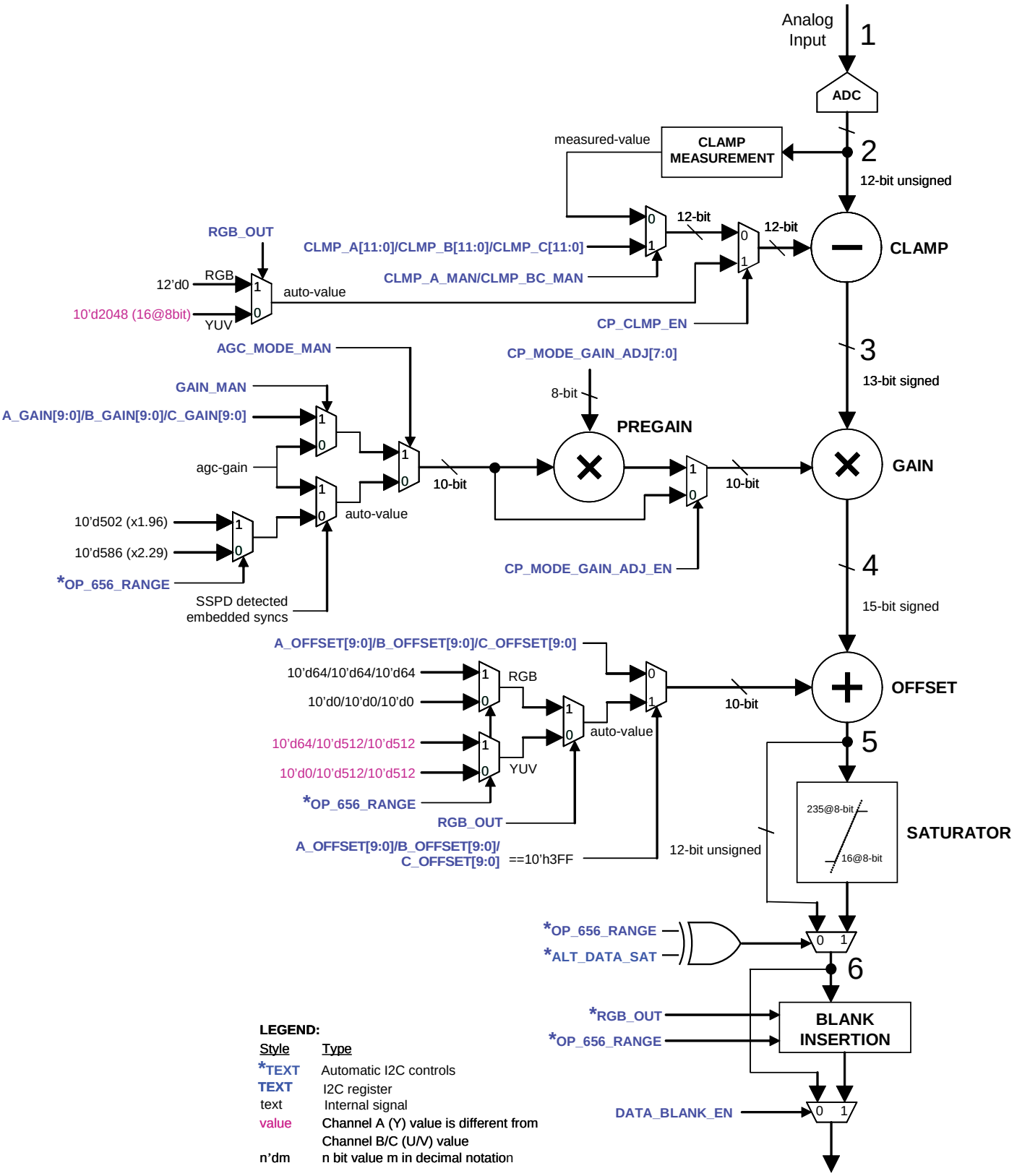
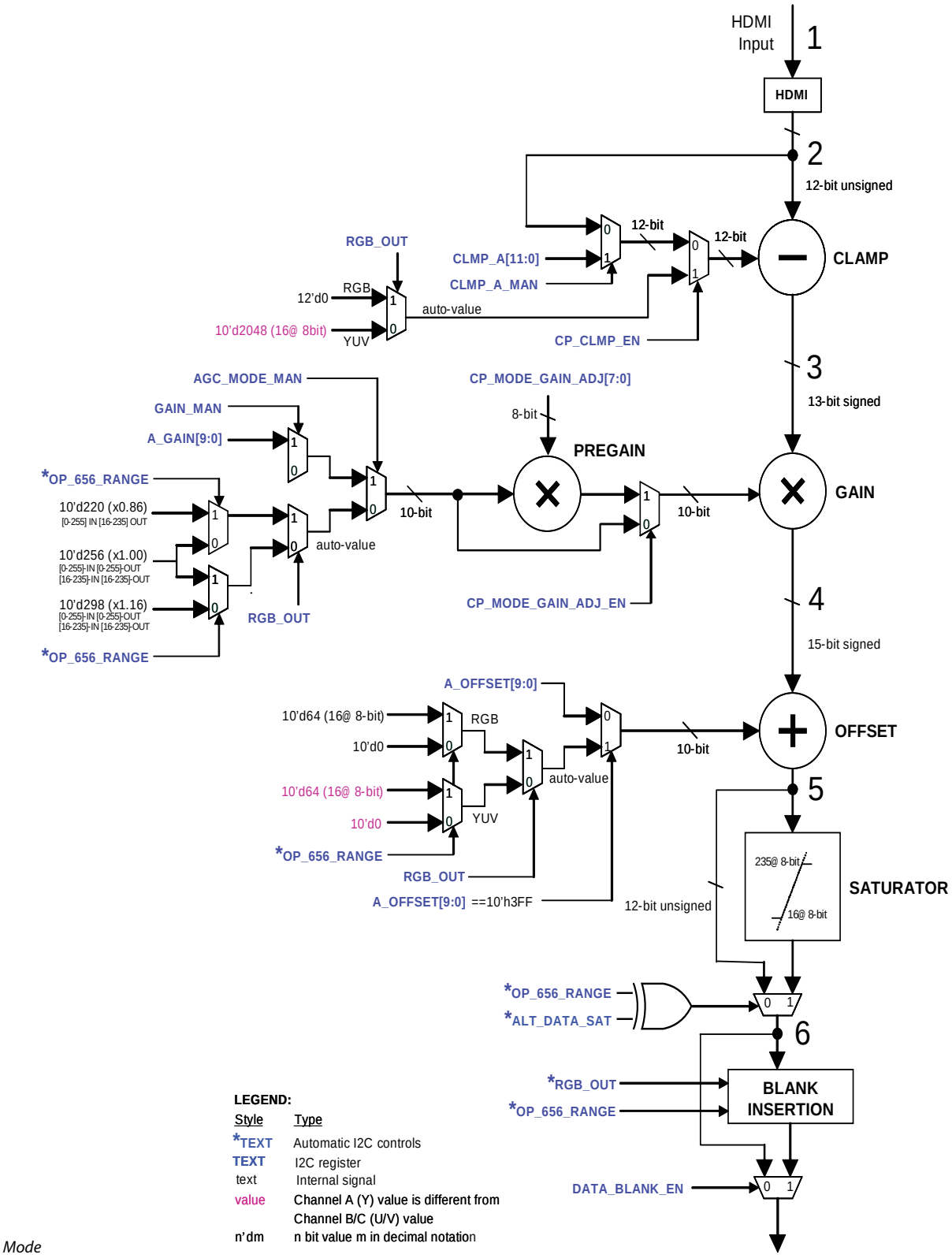


Figure 115: CP Data Path Channel A/B/C (RGB) for Analog



Mode

Figure 116: CP Data Path Channel A (Y) for HDMI Mode

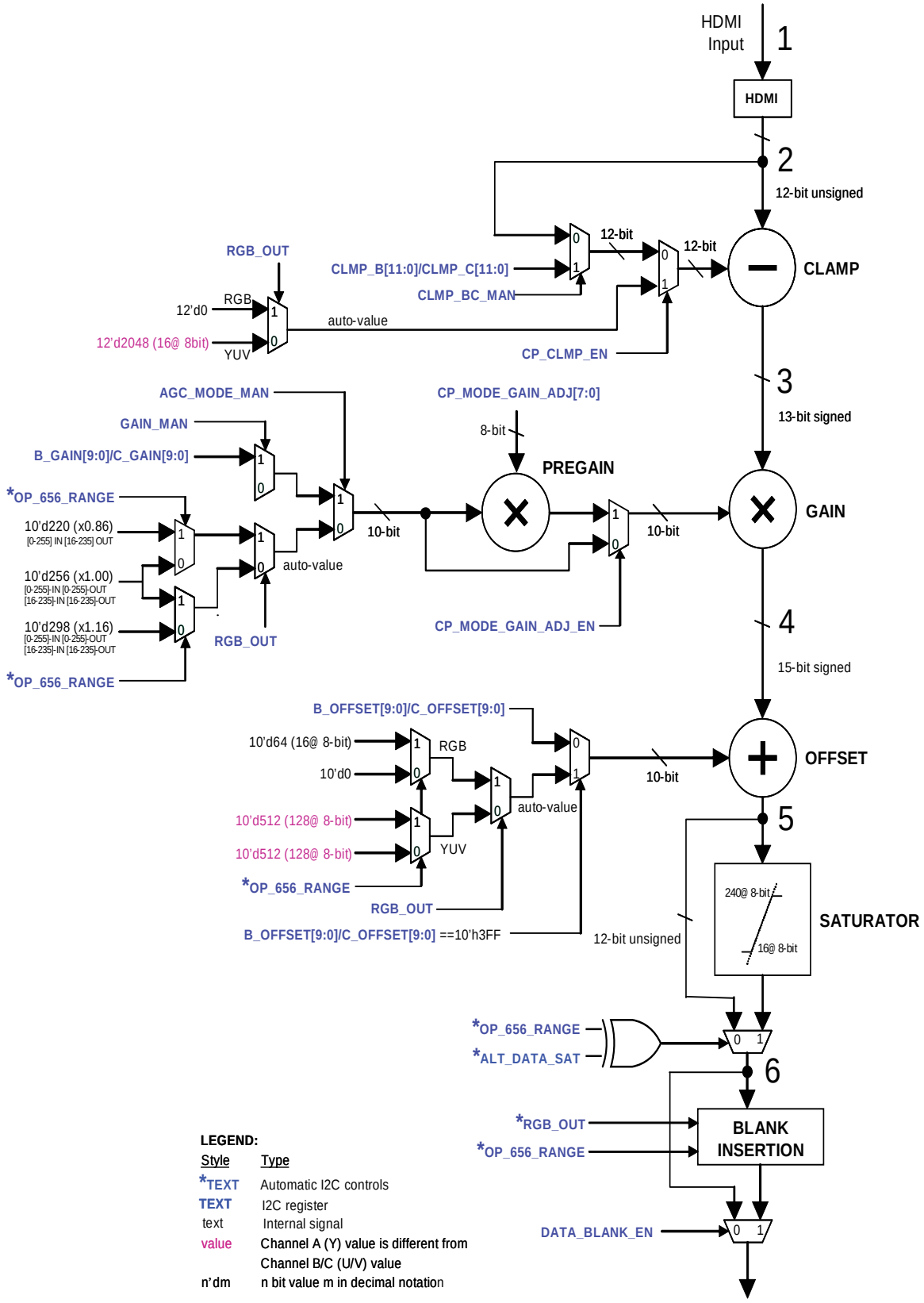


Figure 117: CP Data Path Channel B/C for HDMI Mode

10.7 SYNC PROCESSED BY CP SECTION

The following three sources of HSync and VSync are used in the CP core:

- Embedded sync slicer section
  The sync output by the sync slicers are optionally routed to the CP section via the SSPD block.
- External CSync or HSync and VSync
  The external CSync or HSync and VSync are optionally routed to the CP section via the SSPD block
- HDMI section, which is also a DE signal to the CP
  The sync output from the HDMI section are optionally routed to the CP section via the SSPD block

10.7.1 Sync Extracted by Sync Slicer Section

The ADV7842 has two sync slicers. Each sync slicer can slice one of the four possible embedded sync signals; SYNC1, SYNC 2, SYNC 3 and SYNC 4. The sliced signals are output on the internal sliced signals EMB_SYNC_SEL1 and EMB_SYNC_SEL2 (see AIN_SEL[2:0]). A pair of muxes allows the user to select which sliced signals are output on the internal signals EMB_SYNC_1 and EMB_SYNC_2. These are internal signals that are passed to the STDI/SSPD stage of sync processing (refer to Section 10.8).

The muxes are controlled by SYNC_CH1_EMB_SYNC_SEL[1:0] and SYNC_CH2_EMB_SYNC_SEL[1:0]. By default, both of these controls are 0b00, so EMB_SYNC_1 receives either the output of sync slicer 1 or a LO signal, depending on the selected PRIM_MODE. EMB_SYNC_2 receives the output of sync slicer 2.

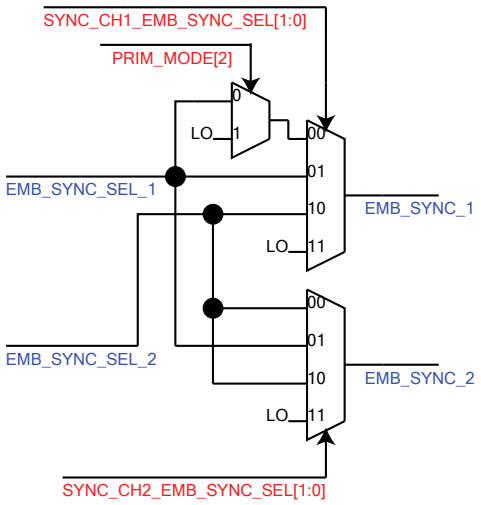


Figure 118: Sliced Signal Path

Both these signals are passed to the STDI/SSPD stage of the sync processing, discussed in Section 10.8. Refer to Section 6.6 for additional information on the sync slicers.

SYNC_CH1_EMB_SYNC_SEL[1:0], Addr 40 (IO), *Address 0x07[1:0]*

A control to select from the outputs of the two synchronization sources as input to sync channel 1.

Function

SYNC_CH1_EMB_SYNC_SEL[1:0]	Description
00 «	Auto-select mode; EMB_SYNC_SEL1 in component or graphics mode or tied LO in HDMI mode. The selection is based on primary mode.
01	EMB_SYNC_SEL1
10	EMB_SYNC_SEL2
11	Tie to GND

SYNC_CH2_EMB_SYNC_SEL[1:0], Addr 40 (IO), *Address 0x08[1:0]*

A control to select from the outputs of the two Sync Slicers as input to Sync Channel 2

Function

SYNC_CH2_EMB_SYNC_SEL[1:0]	Description
00 «	EMB_SYNC_SEL2
01	EMB_SYNC_SEL1
10	EMB_SYNC_SEL2
11	Tie To GND

10.7.2 External Sync and Sync from HDMI Section

The CP section can receive syncs from the external HSync, VSync and CSync inputs or from the HDMI section, as shown in Figure 119. Note also that Figure 119 shows the routing of the internal embedded sync signals, EMB_SYNC_1 and EMB_SYNC_2, which are output by the sync slicers.

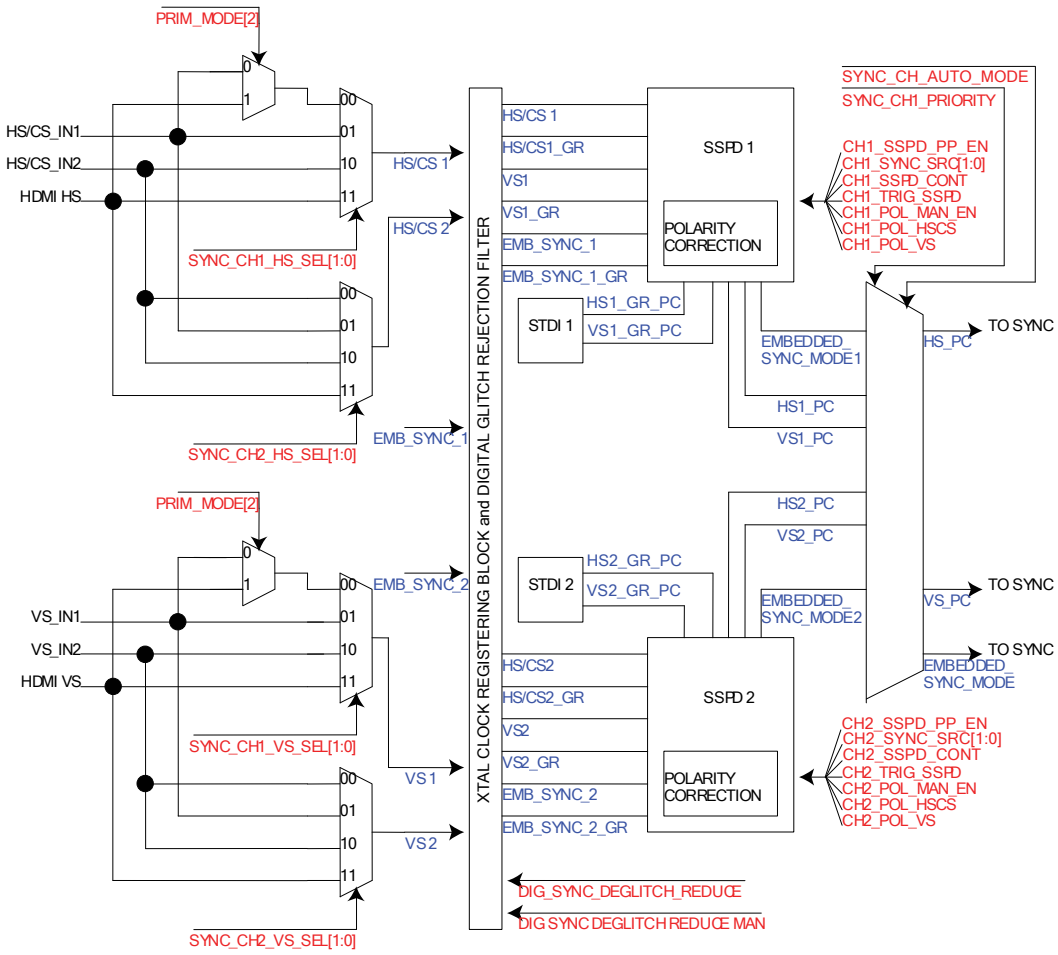


Figure 119: External/HDMI Syncs Routing to CP Section

10.7.2.1 Signals Routing to Synchronization Channels

The ADV7842 has two synchronization channels. Each channel consists of one SSPD and one STDI section. When an HDMI input is applied, the HDMI core will generate HSync, VSync, and DE signals and supply them as input to the each synchronization channel shown in Figure 119. HSync from the HDMI block is denoted as HDMI_HS, and VSync from the HDMI block is denoted as HDMI_VS. DE from the HDMI block is not shown in Figure 119 as it is passed directly to the CP core without any processing when an HDMI input is selected. As well as the HDMI sync signals, external CSync or HSync and VSync signals applied to the pins are provided as inputs to the HSync and VSync muxes.

Two muxes are used to select HS/CS 1 and HS/CS 2 from the three possible HSync signals. Similarly, two muxes are used to select VS 1 and VS 2 from the three possible VS signals. These muxes are controlled by SYNC_CH1_HS_SEL[1:0], SYNC_CH1_VS_SEL[1:0], SYNC_CH2_HS_SEL[1:0], and SYNC_CH2_VS_SEL[1:0]. The outputs of these four muxes, as well as the internal EMB_SYNC_1 and EMB_SYNC_2 signals from the sync slicers (as described in Section 10.7.1) are passed to a clock registering and deglitch block.

SYNC_CH1_HS_SEL[1:0], Addr 40 (IO), *Address 0x07[5:4]*

A control to select the HSync input to sync channel 1.

Function

SYNC_CH1_HS_SEL[1:0]	Description
00 «	Auto-select mode; HS_IN1 or HSync from HDMI (HDMI-HS) set to channel 1 based on primary mode set in PRIM_MODE[3:0]. HDMI-HS in HDMI mode. HS1 input in Component or Graphics mode.
01	Select HS_IN1
10	Select HS_IN2
11	HDMI-HS

SYNC_CH1_VS_SEL[1:0], Addr 40 (IO), *Address 0x07[3:2]*

A control to select the VSync input to sync channel 1.

Function

SYNC_CH1_VS_SEL[1:0]	Description
00 «	Auto-select mode; VS_IN1 or VSync from HDMI (HDMI-VS) set to channel 1 based on primary mode set in PRIM_MODE[3:0]. HDMI-VS in HDMI mode. VS1 input in Component or Graphics mode.
01	Select VS_IN1 input
10	Select VS_IN2 input
11	HDMI-VS

SYNC_CH2_HS_SEL[1:0], Addr 40 (IO), *Address 0x08[5:4]*

A control to select the HSync input to sync channel 2.

Function

SYNC_CH2_HS_SEL[1:0]	Description
00	Select HS2 input
01 «	Select HS1 input
10	Select HS2 input
11	Select HDMI HS

SYNC_CH2_VS_SEL[1:0], Addr 40 (IO), *Address 0x08[3:2]*

A control to select the VSync input to Sync Channel 2

Function

SYNC_CH2_VS_SEL[1:0]	Description
00	Select VS2 input
01 «	Select VS1 input
10	Select VS2 input
11	Select HDMI VS

10.7.2.2 XTAL Clock Registering and Glitch Rejection Filter

The XTAL CLOCK REGISTERING BLOCK and DIGITAL GLITCH REJECTION FILTER have two outputs for each input. As shown in Figure 119, each output has two instances, one with the same name as the input, and one with _GR at the end of the name. The _GR signals are latched by the XTAL clock and have a digital filter applied to them, whereas the output signals with the same name as the input signals are true bypass versions of the input signals – they are not latched by XTAL and they do not have any filtering applied to them.

The _GR signals will have a glitch rejection filter applied to them. The filter can be controlled by the DIG_SYNC_DEGLITCH_REDUCE and DIG_SYNC_DEGLITCH_REDUCE_MAN signals.

When DIG_SYNC_DEGLITCH_REDUCE_MAN is set to 0, the block automatically removes any sync signals that are less than five

XTAL clocks wide for component inputs up to and including 1080i, and sync signals that are less than two XTAL clocks wide for component input 1080p and graphics standards. When this bit is set to 1, the user can select the filter to reduce syncs that are less than five XTAL clocks wide, or less than two XTAL clocks wide by setting the DIG_SYNC_DEGLITCH_REDUCE bit to 0 or 1 respectively.

DIG_SYNC_DEGLITCH_REDUCE, Addr 44 (CP), *Address 0xF5[3]*

A control to configure the deglitch filters that processes synchronization signal before there are input to the SSPD section. The value set in this register is effective if DIG_SYNC_DEGLITCH_REDUCE_MAN is set to 1.

Function

DIG_SYNC_DEGLITCH_REDUCE	Description
1	Remove 2 XTAL clock wide glitches synchronization signals input to the SSPD sections
0 «	Remove 5 XTAL clock wide glitches from synchronization signals input to the SSPD sections

DIG_SYNC_DEGLITCH_REDUCE_MAN, Addr 44 (CP), *Address 0xF5[2]*

A control to manually configure the deglitch filters that process synchronization signals input to the SSPD sections.

Function

DIG_SYNC_DEGLITCH_REDUCE_MAN	Description
1	Manual Configuration. The deglitch filters are configured via DIG_SYNC_DEGLITCH_REDUCE.
0 «	Automatic configuration. The deglitch filters remove 5 XTAL clock wide glitches from the synchronization signals input to the SSPD section.

10.7.2.3 Signal Routed to SSPD Blocks

The SSPD block from each synchronization channel receives six input signals, namely HS, HS_GR, VS, VS_GR, EMB_SYNC, and EMB_SYNC_GR. SSPD analyses the _GR signals to determine which signals have valid sync information, and will 'correct' the polarity of the syncs. In this instance, the definition of a 'correct' sync polarity is 'negative going', so for input formats with positive going syncs, these syncs will be inverted. The outputs from the SSPD section are the signals whose names end with _PC. These signals are the polarity corrected version of the syncs signals input to the SSPD sections.

The SSPD block will pass a corrected, registered, glitch rejected HSync and VSync signal to the STDI block. Note that for embedded sync inputs or external CSync inputs where both HSync and VSync information are contained in one signal, the same signal will be applied to both inputs of the STDI block.

The SSPD block will also pass a corrected HS and VS signal to the final mux shown on the right side of Figure 119, as well as a signal called EMBEDDED_SYNC_MODE which tells the CP core if the source of the HS_PC and VS_PC signals are from an embedded signal or separate signals. These syncs are polarity corrected but are not XTAL registered, nor are they glitch rejected.

The final mux shown in Figure 119 selects between the HSync and VSync (and embedded sync information signal) from sync channel 1 and sync channel 2. Various controls exist to select how the SSPD block works, and which sync channel is passed to the CP core. These controls are described in detail in Section 10.8.

Note that as stated previously, the syncs signals VS_PC and HS_PC passed to the CP core are polarity corrected but not XATL registered and glitch rejected. However, internally in the CP core, a glitch rejection filter is applied which rejects any sync signals less than seven CP clocks in width. This glitch filter is not controllable.

10.7.3 Final Sync Muxing Stage

As described in Section 10.7, the main source for sync information into the CP core is the SSPD block.
Figure 120 shows the final stage of muxing, which selects the syncs to be used in the CP core.

HSync and VSync signals from the selected SSPD block are passed through the glitch rejection filters. Then, depending on whether or not the input was an embedded sync signal type (external CS or SOG/SOY) VS is generated by the VSYNC SLICER block.

For CS/SOG/SOY input types, the embedded sync signal must be sliced to find the VS signal, but this should not be done for separate HS and VS inputs. Therefore, the EMBEDDED_SYNC_MODE control signal that is sent from the SSPD block is used to control a mux to bypass the VSYNC SLICER block when the input is external HSync and VSync.

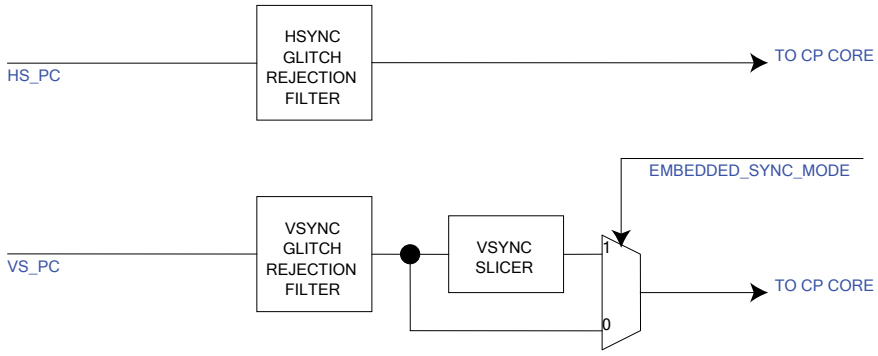


Figure 120: Final Sync Muxing Stage

10.8 SYNCHRONIZATION PROCESSING CHANNEL MUX

The ADV7842 has two synchronization processing channels, as shown in Figure 119. These contain Synchronization Source Polarity Detectors (SSPD) and the Standard Detection and Identification (STDI) functionality that receives synchronization, as described in Section 10.7.2.1.

As explained in Section 10.7.2.1, channel 1 and channel 2 are identical in that they consist of the same basic SSPD and STDI blocks. However, they differ in the way in which the signals are multiplexed so that any synchronization source can be routed to any of the synchronization channels. By default, the synchronization channels are configured into two sets as follows:

- Set 1: HS_IN1, VS_IN1, HDMI_HS and EMB_SYNC_1 feed synchronization channel 1
- Set 2: HS_IN2, VS_IN2, HDMI_VS and EMB_SYNC_2 feed synchronization channel 2

The registers that control the synchronization signals routed to each synchronization channel are described in Section 10.7.1 and Section 10.7.2.1. The output from one synchronization channel is used to provide timing to the CP core. The method that is used to determine the sync channel that is used to output sync signals to the CP core is determined by SYNC_CH_AUTO_MODE. This bit is used to enable and disable the channel auto mode and works in conjunction with the SYNC_CH1_PRIORITY bit as follows:

- **SYNC_CH_AUTO_MODE = 0**
  The priority of the channels is determined by SYNC_CH1_PRIORITY:
  - SYNC_CH1_PRIORITY = 1: Select channel 1
  - SYNC_CH1_PRIORITY = 0: Select channel 2

- **SYNC_CH_AUTO_MODE = 1**

The synchronization channel that output sync signals to the CP core is automatically selected and based on the free run status of each channel. The priority of selection is determined by SYNC_CH1_PRIORITY:

- SYNC_CH1_PRIORITY = 1: Priority for channel 1 – free run if both channels are in free run and monitor channel 1.
- SYNC_CH1_PRIORITY = 0: Priority for channel 2 – free run if both channels are in free run and monitor channel 2.

SYNC_CH_AUTO_MODE, Addr 40 (IO), *Address 0x07[7]*

A control to set automatic synchronization channel selection to CP core. Auto mode selects which synchronization channel drives the CP based on the free run status of each channel.
The priority of selection is determined by SYNC_CH1_PRIORITY when both channels are in free run mode.

Function

SYNC_CH_AUTO_MODE	Description
0 «	Disables auto mode. Priority of channels determined by SYNC_CH1_PRIORITY.
1	Enables auto mode. Automatically selects which synchronization channel drives the CP core based on the free-run status.

SYNC_CH1_PRIORITY, Addr 40 (IO), *Address 0x07[6]*

A control to select which sync channel has priority to CP core.

Function

SYNC_CH1_PRIORITY	Description
0	sync channel 2 sync processing result takes priority
1 «	sync channel 1 sync processing result takes priority

Refer to Section 10.13 for a detailed explanation of free run mode.
The output from the synchronization channel multiplexing can be read from the SEL_SYNC_CHANNEL bit in the IO Map.

SEL_SYNC_CHANNEL, Addr 40 (IO), *Address 0x12[7] (Read Only)*

A readback to indicate the currently selected sync processing channel applied to CP core

Function

SEL_SYNC_CHANNEL	Description
0 «	Sync Channel 2 is being processed by CP core
1	Sync Channel 1 is being processed by CP core

10.8.1 Synchronization Source Polarity Detector

Section 10.8 describes how the various synchronization signals are routed to the two synchronization processing channels. Each of these channels employs an SSPD block to enable it to determine where the synchronization source comes from, and its polarity.

The functions of an SSPD block are:

- Automatic detection of the active synchronization source
- Automatic detection of the synchronization polarity, if applicable
- Readback on SSPD manual override for the synchronization source through synchronization channel multiplexing, as described in Section 10.8
- Manual override for polarity detection via CHx_POL_MAN_EN (where x refers to either 1 or 2)

The SSPD block can operate either in continuous or in single-shot mode. Continuous mode means that the block permanently monitors the inputs and updates its outputs. In single-shot mode, the SSPD block waits for a 0 to 1 transition on the CHx_TRIG_SSPD bit (where x refers to either 1 or 2) before it scans the synchronization inputs once. Single-shot operation is useful to avoid system scheduling conflicts.

The SSPD state machine searches for active synchronization signals in the following order of priority:

1. External HS/VS
2. External CS
3. Embedded synchronization

The ADV7842 by default tries to use separate HS/VS signals - even if there is parallel existence of embedded-sync signals. To change this behavior user may wish to use CH1_SSPD_PP_EN and CH2_SSPD_PP_EN registers. By enabling these registers – first preference for SSPD will be embedded signal (refer also to CH1_RS_ACTIVE and CH2_RS_ACTIVE).

If no synchronization is found, it assumes embedded synchronization.

If external HS or VS is found, the block decides on the synchronization polarity based on a measurement of the mark-space ratio of the HS/VS signals detected. The results from the SSPD detection are read back, but only after they are flagged as valid by the CHx_SSPD_DVALID (where x refers to either 1 or 2) flag. Refer to Figure 121 for information on the data exchange.

The following readback information is available from the SSPD section:

- Active synchronization source (either the result back from manual setting or the result from auto detection)
- Activity report on the CHx (where x refers to either 1 or 2) HS and VS inputs
- Detected polarity on HS and VS inputs

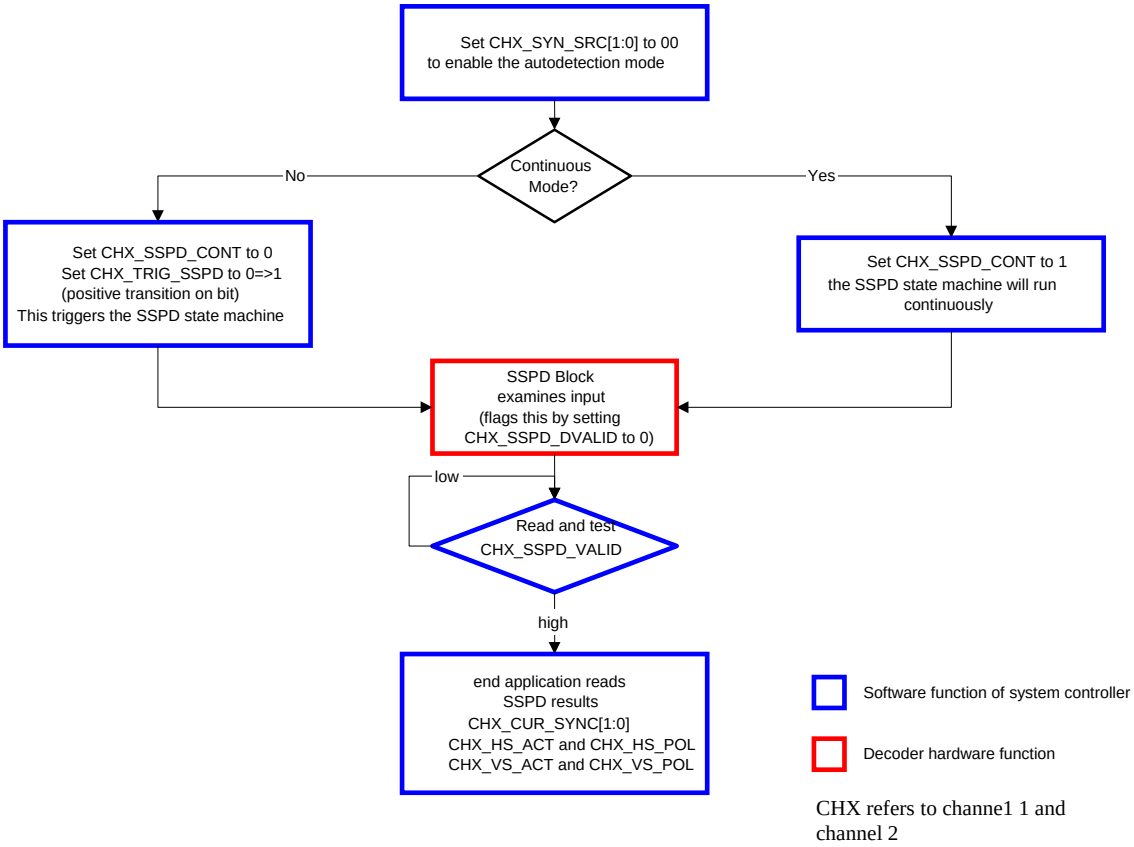


Figure 121: SSPD Auto Detection Flowchart

CH1_SSPD_CONT, Addr 44 (CP), *Address 0x85[1]*

A control to set the synchronization source polarity detection mode for sync channel 1 SSPD.

Function

CH1_SSPD_CONT	Description
0	sync channel 1 SSPD works in one-shot mode (triggered by a 0 to 1 transition on the CH1_TRIG_SSPD bit)
1 «	sync channel 1 SSPD works in continuous mode

CH2_SSPD_CONT, Addr 44 (CP), *Address 0x41[1]*

A control to set the synchronization source polarity detection mode for sync channel 2 SSPD.

Function

CH2_SSPD_CONT	Description
0	sync channel 2 SSPD works in one-shot mode (triggered by a 0 to 1 transition on the CH2_TRIG_SSPD bit)
1 «	sync channel 2 SSPD works in continuous mode

CH1_TRIG_SSPD, Addr 44 (CP), *Address 0x85[2]*

Trigger synchronization source and polarity detector for sync channel 1 SSPD. A 0 to 1 transition in this bit restarts the auto-sync detection algorithm. This is not a self-clearing bit and must be set to 0 to prepare for next trigger.

Function

CH1_TRIG_SSPD	Description
0 «	Default value - transition 0 to 1 restarts auto-sync detection algorithm
1	Transition 0 to 1 restarts auto-sync detection algorithm

CH2_TRIG_SSPD, Addr 44 (CP), *Address 0x41[2]*

Trigger synchronization source and polarity detector for sync channel 2 SSPD. A 0 to 1 transition in this bit restarts the auto-sync detection algorithm. This is not a self-clearing bit and must be set to 0 to prepare for next trigger.

Function

CH2_TRIG_SSPD	Description
0 «	Default value - transition 0 to 1 restarts auto-sync detection algorithm
1	Transition 0 to 1 restarts auto-sync detection algorithm

CH1_SYNC_SRC[1:0], Addr 44 (CP), *Address 0x85[4:3]*

A control to select to synchronization signals processed by sync channel 1 SSPD

Function

CH1_SYNC_SRC[1:0]	Description
00 «	Auto detect mode for synchronization source. Use results of auto detection for synchronization signal routing. Result can be read back via CH1_CUR_SYNC[1:0] bits.
01	Manual setting: separate HSync and VSync to sync channel 1 SSPD
10	Manual setting: CSync on HSync input to sync channel 1
11	Manual setting: embedded synchronization signal input to sync channel 1

CH2_SYNC_SRC[1:0], Addr 44 (CP), *Address 0x41[4:3]*

A control to select to synchronization signals processed by sync channel 2 SSPD

Function

CH2_SYNC_SRC[1:0]	Description
00 «	Auto detect mode for synchronization source. Use results of auto detection for synchronization signal routing. Result can be read back via CH2_CUR_SYNC[1:0] bits.
01	Manual setting: separate HSync and VSync to the sync channel 2 SSPD
10	Manual setting: CSync on HSync input to the sync channel 2
11	Manual setting: embedded synchronization signal input to the sync channel 2

CH1_POL_MAN_EN, Addr 44 (CP), *Address 0x85[7]*

A control to override for polarity detection by sync channel 1 SSPD. CH1_POL_MAN_EN must be set high for this bit to become active.

Function

CH1_POL_MAN_EN	Description
0 «	Use result from sync channel 1 SSPD polarity auto detection
1	Manual override: use CH1_POL_VS and CH1_POL_HS

CH2_POL_MAN_EN, Addr 44 (CP), *Address 0x41[7]*

A control to override the polarity detection by sync channel 2 SSPD

Function

CH2_POL_MAN_EN	Description
0 «	Use result from sync channel 2 SSPD autodetection
1	Use CH2_POL_VS and CH2_POL_HS

CH1_POL_VS, Addr 44 (CP), *Address 0x85[6]*

A control to override for polarity of VSync by sync channel 1 SSPD.

Function

CH1_POL_VS	Description
0 «	VSync input to sync channel 1 carries negative polarity signal.
1	VSync input to sync channel 1 carries positive polarity signal.

CH2_POL_VS, Addr 44 (CP), *Address 0x41[6]*

A control to override for polarity of VSync by sync channel 2 SSPD. CH2_POL_MAN_EN must be set high for this control to be active.

Function

CH2_POL_VS	Description
0 «	VSync input to sync channel 2 carries negative polarity signal.
1	VSync input to sync channel 2 carries positive polarity signal.

CH1_POL_HSCS, Addr 44 (CP), *Address 0x85[5]*

A control to override the polarity of HSync by to sync channel 1 SSPD. CH1_POL_MAN_EN must be set high for this bit to become active.

Function

CH1_POL_HSCS	Description
0 «	HSync input to sync channel 1 carries negative polarity signal (HSync or CSync).
1	HSync input to sync channel 1 carries positive polarity signal (HSync or CSync).

CH2_POL_HSCS, Addr 44 (CP), *Address 0x41[5]*

A control to override the polarity of HSync by to sync channel 2 SSPD. CH2_POL_MAN_EN must be set high for this control to be effective.

Function

CH2_POL_HSCS	Description
0 «	HSync input to sync channel 2 carries negative polarity signal (HSync or CSync).
1	HSync input to sync channel 2 carries positive polarity signal (HSync or CSync).

EMB_SYNC_ON_ALL, Addr 44 (CP), *Address 0x67[5]*

A control to alter the gain computed by the AGC based on the presence of an embedded synchronization on channels A, B and C. Used only in case of RGB input and RGB output with Color-Controls enabled

Function

EMB_SYNC_ON_ALL	Description
0 «	Embedded synchronization is present only on the Luma channel (i.e. channel A)
1	All three input channels have and embedded synchronization

CH1_SSPD_PP_EN, Addr 44 (CP), *Address 0x84[1]*

A control to enable sync channel 1 SSPD post processing

Function

CH1_SSPD_PP_EN	Description
0 «	Disable post processing of the synchronization signals input to sync channel 1 SSPD
1	Check for activity on embedded synchronization signal input to sync channel 1 SSPD when it detects activity on HSync CSync and VSync. Activity on the embedded signal input to sync channel 1 SSPD is reported by CH1_RS_ACTIVE. The post processing of the synchronization signal input to sync channel 1 SSPD works only if the timing on the embedded synchronization signal and the timing on the HSync/CSync and VSync signals are the same.

CH2_SSPD_PP_EN, Addr 44 (CP), *Address 0x41[0]*

A control to enable sync channel 2 SSPD post processing

Function

CH2_SSPD_PP_EN	Description
0 «	Disable post processing of the synchronization signals input to sync channel 2 SSPD
1	Check for activity on embedded synchronization signal input to sync channel 2 SSPD when it detects activity on HSync/CSync and VSync. Activity on the embedded signal input to sync channel 2 SSPD is reported by CH2_RS_ACTIVE. The post processing of the synchronization signal input to sync channel 2 SSPD works only if the timing on the embedded synchronization signal and the timing on the HSync/CSync and VSync signals are the same.

10.8.1.1 SSPD Readback Signals

CH1_SSPD_DVALID, Addr 44 (CP), *Address 0xB5[7] (Read Only)*

CH1_SSPD_DVALID is set to 1 when the read backs from the SSPD section of the synchronization sync channel 1 are valid. This bit is set to 1 after 2^22 crystal clock periods following a reset of the CP section. This bit is set to 0 when the DUT is reset.

Function

CH1_SSPD_DVALID	Description
0 «	Sync channel 1 SSPD results not valid for readback
1	Sync channel 1 SSPD results valid

CH2_SSPD_DVALID, Addr 44 (CP), *Address 0x4F[7] (Read Only)*

CH2_SSPD_DVALID is set to 1 when the read backs from the SSPD section of the synchronization Sync channel 2 are valid. This bit is set to 1 after 2^22 crystal clock periods following a reset of the CP section. This bit is set to 0 when the DUT is reset.

Function

CH2_SSPD_DVALID	Description
0 «	Sync channel 2 SSPD results not valid for readback
1	Sync channel 2 SSPD results valid (detection finished)

CH1_CUR_SYNC_SRC[1:0], Addr 44 (CP), *Address 0xB5[1:0] (Read Only)*

Readback of current synchronization source detected by sync channel 1 SSPD.

Function

CH1_CUR_SYNC_SRC[1:0]	Description
00 «	Not used
01	Activity detected on HSync and VSync input to sync channel 1 SSPD
10	CSync detected in the HSync input to sync channel 1 SSPD
11	Activity detected on embedded synchronization input to sync channel 1 SSPD

CH2_CUR_SYNC_SRC[1:0], Addr 44 (CP), *Address 0x4F[1:0] (Read Only)*

Readback of current synchronization source detected by sync channel 2 SSPD.

Function

CH2_CUR_SYNC_SRC[1:0]	Description
00 «	Not used
01	Activity detected on HSync and VSync input to sync channel 2 SSPD
10	CSync detected in the HSync input to sync channel 2 SSPD
11	Activity detected on embedded synchronization input to sync channel 2 SSPD

CH1_CUR_POL_HS, Addr 44 (CP), *Address 0xB5[3] (Read Only)*

Readback indicating the polarity of the HSync/CSync input to sync channel 1 SSPD

Function

CH1_CUR_POL_HS	Description
0 «	The HSync CSync input to sync channel 1 SSPD has negative polarity
1	The HSync CSync input to sync channel 1 SSPD has positive polarity

CH2_CUR_POL_HS, Addr 44 (CP), *Address 0x4F[3] (Read Only)*

A Readback indicating the polarity of the HSync/CSync input to sync channel 2 SSPD

Function

CH2_CUR_POL_HS	Description
0 «	The HSync CSync input to sync channel 2 SSPD has negative polarity
1	The HSync CSync input to sync channel 2 SSPD has positive polarity

CH1_HS_ACT, Addr 44 (CP), *Address 0xB5[4] (Read Only)*

Readback indicating activity on the HSync/CSync input to sync channel 1 SSPD

Function

CH1_HS_ACT	Description
0 «	No activity detected on the HSync CSync input to sync channel 1 SSPD
1	HSync CSync input to sync channel 1 SSPD carries an active signal

The SSPD section continuously monitors the HSync input signal over timing windows of 2^{22} crystal clock periods (refer to Figure 122). CH1_HS_ACT is updated at the end of each window, as follows:

- CH1_HS_ACT is set to 1 if the SSPD has detected eight edges or four periods on the HSync signal
- CH1_HS_ACT is set to 0 if the SSPD has detected less than eight edges or four periods on the HSync signal

CH2_HS_ACT, Addr 44 (CP), *Address 0x4F[4] (Read Only)*

A Readback indicating activity on the HSync CSync input to sync channel 2 SSPD

Function

CH2_HS_ACT	Description
0 «	No activity detected on the HSync/CSync input to sync channel 2 SSPD
1	HSync/CSync input to sync channel 2 SSPD carries an active signal

CH1_CUR_POL_VS, Addr 44 (CP), *Address 0xB5[5] (Read Only)*

Readback indicating polarity on the HSync/CSync input to sync channel 1 SSPD

Function

CH1_CUR_POL_VS	Description
0 «	The VSync input to sync channel 1 SSPD has negative polarity signal
1	The VSync input to sync channel 1 SSPD has positive polarity signal

CH2_CUR_POL_VS, Addr 44 (CP), *Address 0x4F[5] (Read Only)*

A Readback indicating the polarity of the VSync input to sync channel 2 SSPD

Function

CH2_CUR_POL_VS	Description
0 «	The VSync input to sync channel 2 SSPD has negative polarity signal
1	The VSync input to sync channel 2 SSPD has positive polarity signal

CH1_VS_ACT, Addr 44 (CP), *Address 0xB5[6] (Read Only)*

Readback indicating the activity the VSync input to sync channel 1 SSPD

Function

CH1_VS_ACT	Description
0 «	No activity detected on the VSync input to sync channel 1 SSPD
1	The VSync input to sync channel 1 SSPD carries an active signal

The SSPD section continuously monitors the VSync input signal over timing windows of 2^{22} crystal clock periods (refer to Figure 122). CH1_VS_ACT is updated at the end of each window, as follows:

- CH1_VS_ACT is set to 1 if the SSPD has detected four edges or two periods on the VSync signal
- CH1_VS_ACT is set to 0 if the SSPD has detected less than four edges or two periods on the VSync signal

CH2_VS_ACT, Addr 44 (CP), *Address 0x4F[6] (Read Only)*

A Readback indicating the activity the VSync input to sync channel 2 SSPD.

Function

CH2_VS_ACT	Description
0 «	No activity detected on the VSync input to sync channel 2 SSPD
1	The VSync input to sync channel 2 SSPD carries an active signal

The SSPD section continuously monitors the VSync input signal over timing windows of 2^{22} crystal clock periods (refer to Figure 122). CH2_VS_ACT is updated at the end of each window as follows:

- CH2_VS_ACT is set to 1 if SSPD has detected four edges or two periods on the VSync signal
- CH2_VS_ACT is set to 0 if SSPD has detected less than four edges or two periods on the VSync signal

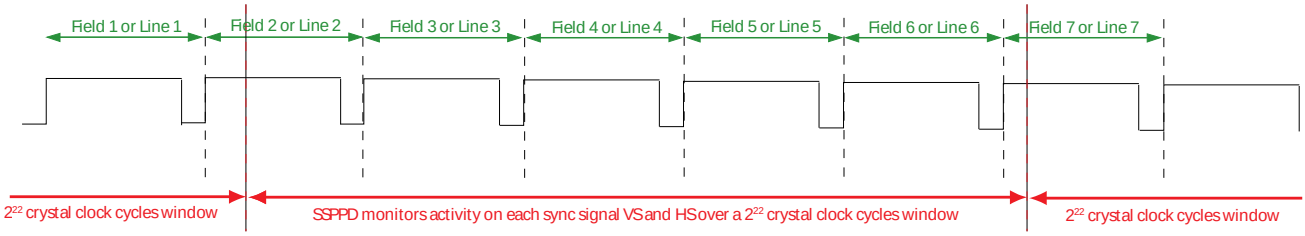


Figure 122: SSPD VSync and HSync Monitoring Operation

CH1_RS_ACTIVE, Addr 44 (CP), *Address 0xB5[2] (Read Only)*

A readback indicating activity in embedded synchronization signal input to sync channel 1 SSPD. CH1_SSPD_PP_EN must be set to 1 and CH1_SSPD_DVALID must return 1 for this readback to be valid. This is readback is only valid when there is a HSync and VSync signal present.
It is not valid to use this bit when only embedded signal is present. The purpose of this bit is to indicate that the user can switch to embedded sync if using HSync and VSync inputs.

Function

CH1_RS_ACTIVE	Description
0 «	Activity detected on the embedded signal input to sync channel 1 SSPD
1	No activity detected on the embedded signal input to sync channel 1 SSPD

CH2_RS_ACTIVE, Addr 44 (CP), *Address 0x4F[2] (Read Only)*

A readback indicating activity in embedded synchronization signal input to sync channel 2 SSPD. CH2_SSPD_PP_EN must be set to 1 and CH1_SSPD_DVALID must return 1 for this readback to be valid. This is readback is only valid when there is a HSync and VSync signal present.
It is not valid to use this bit when only embedded signal is present. The purpose of this bit is to indicate that the user can switch to embedded sync if using HSync and VSync inputs.

Function

CH2_RS_ACTIVE	Description
0 «	Activity detected on the embedded signal input to sync channel 2 SSPD
1	No activity detected on the embedded signal input to sync channel 2 SSPD

Notes:

- It is possible to monitor changes in the CH1_VS_ACT, CH1_HS_ACT and CH1_RS_ACTIVE flags via the SSPD_RSLT_CHNGD_CH1_ST interrupt status.
- Likewise, it is possible to monitor changes in the CH2_VS_ACT and CH2_HS_ACT, CH2_RS_ACTIVE flags via the SSPD_RSLT_CHNGD_CH2_ST interrupt status.

SSPD_RSLT_CHNGD_CH1_ST, Addr 40 (IO), *Address 0x5C[0] (Read Only)*

Latched signal status of SSPD Result Changed for sync channel 1 interrupt signal. Once set this bit will remain high until the interrupt has been cleared via SSPD_RSLT_CHNGD_CH1_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

SSPD_RSLT_CHNGD_CH1_ST	Description
0 «	No SSPD result changed for sync channel 1 interrupt event occurred.
1	A SSPD result changed for sync channel 1 interrupt event has occurred..

SSPD_RSLT_CHNGD_CH1_RAW, Addr 40 (IO), *Address 0x5B[0] (Read Only)*

Status of the SSPD Result Changed on sync channel 1 interrupt signal. When set to 1 it indicates a change in SSPD result of the currently selected sync channel. A change in SSPD result can be either due to a polarity or source change. Once set, this bit will remain high until it is cleared via SSPD_RSLT_CHNGD_CH1_CLR.

Function

SSPD_RSLT_CHNGD_CH1_RAW	Description
0 «	No change in the SSPD result for sync channel 1
1	A change has occurred in SSPD result for sync channel 1

SSPD_RSLT_CHNGD_CH2_ST, Addr 40 (IO), *Address 0x5C[4] (Read Only)*

Latched signal status of SSPD Result Changed for sync channel 2 interrupt signal. Once set this bit will remain high until the interrupt has been cleared via SSPD_RSLT_CHNGD_CH2_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

SSPD_RSLT_CHNGD_CH2_ST	Description
0 «	No SSPD result changed for sync channel 2 interrupt event occurred.
1	A SSPD result changed for sync channel 2 interrupt event has occurred.

SSPD_RSLT_CHNGD_CH2_RAW, Addr 40 (IO), *Address 0x5B[4] (Read Only)*

Status of the SSPD Result Changed on sync channel 2 interrupt signal. When set to 1 it indicates a change in SSPD result of the currently selected sync channel. A change in SSPD result can be either due to a polarity or source change. Once set, this bit will remain high until it is cleared via SSPD_RSLT_CHNGD_CH2_CLR.

Function

SSPD_RSLT_CHNGD_CH2_RAW	Description
0 «	No change in the SSPD result for sync channel 2
1	A change has occurred in SSPD result for sync channel 2

10.8.2 Standard Detection and Identification

As shown in Figure 119, the two synchronization processing channels also contain Standard Detection and Identification (STDI) blocks. These monitor the synchronization signals to determine the video input standard.

The STDI blocks perform four key measurements:

- **Block Length CHx_BL[13:0]**
  This is the number of 28.6363 MHz clock cycles (XTAL frequency) in a block of eight lines. From this, the time duration of one line can be concluded.
- **Line Count in Field CHx_LCF[10:0]**
  The CHx_LCF[10:0] readback value is the number of lines between two VSyncs, that is, over one field measured by channel x.
- **Line Count in VSYNC CHx_LCVS[4:0]**
  The LCVS[4:0] readback value is the number of lines within one VSync period.
- **Field Length CHx_FCL[12:0]**
  This is the number of 28.6363 MHz clock cycles in a 1/256th of a field. Alternately, this value of FCL multiplied by 256 gives one field length count in 28.6363 MHz (XTAL) clocks.

Note: CHx = CH1 or CH2 in the above register descriptions representing channel 1 and channel 2 related registers.

By interpreting these four parameters, it is possible to derive the applied video signal horizontal and vertical resolution information and determine its standard.

In ADV7842, there are three operational modes for the STDI block:

- **Continuous mode**:
  The STDI block performs continuous measurements on lock/unlock bases and updates the corresponding I^2C registers based on the lock status bit (STDI_DVALID).

- **Real-time continuous mode**:
  The STDI block performs continuous measurement regardless of the lock/unlock bases and always updates real-time measurement data to the corresponding I^2C registers.

- **Single shot mode:**
  The STDI block waits for a trigger (0 to 1 transition on CHx_TRIG_STDI) to start the measurements. Single-shot mode can be useful in complex systems where the scheduling of functions is important.

A data valid flag, CHx_STDI_DVALID, is provided, which is based on the status of the horizontal/vertical lock of the block and is held low during the measurements. The four parameters should only be read after the CHx_STDI_DVALID flag has gone high for the continuous/single shot mode. In real-time continuous mode, the ADV7842 allows the user to monitor the real-time timing measurement regardless of the CHx_STDI_DVALID flag. Refer to Section 10.8.3.4 for information on the readback values.

Notes:

- Synchronization type pulses include horizontal synchronization, equalization and serration pulses, and Macrovision pulses.
- Macrovision pseudo synchronization and AGC pulses are counted by the STDI block in normal readback mode. This does not prohibit the identification of the video signal.
- The CHx_TRIG_STDI flag is not self clearing. The measurements are only started upon setting the CHx_TRIG_STDI flag. This means that after setting it, it must be cleared again by writing a 0 to it. This second write (to clear the flag) can be done at any time and does not have any effect on running measurements. It also does **not** invalidate previous measurement results.
- The ADV7842 only **measures** those parameters, but does not take any action based upon them. The part does not reconfigure itself. To avoid unforeseen problems in the scheduling of a system controller, the part merely helps to identify the input.
- Since real-time continuous mode provides the capability to monitor the real-time measurement data regardless of the block lock

status, the user should be aware that the timing readback values may not be a valid readback measurement in this mode.

CH1_STDI_CONT, Addr 44 (CP), *Address 0x86[1]*

A control to set the synchronization source polarity detection mode for sync channel 1 SSPD.

Function

CH1_STDI_CONT	Description
0	sync channel 1 SSPD works in one-shot mode (triggered by a 0 to 1 transition on the CH1_TRIG_SSPD bit)
1 «	sync channel 1 SSPD works in continuous mode

CH2_STDI_CONT, Addr 44 (CP), *Address 0x42[1]*

A control to select the sync channel 2 STDI mode of operation

Function

CH2_STDI_CONT	Description
0	sync channel 2 STDI block operates in single-shot mode. 0 to 1 transition on CH2_TRIG_STDI triggers a measurement of the sync channel 2 STDI block.
1 «	sync channel 2 STDI runs in continuous mode

BYPASS_STDI1_LOCKING, Addr 44 (CP), *Address 0xF5[1]*

Bypass STDI locking for sync channel 1

Function

BYPASS_STDI1_LOCKING	Description
0 «	Update CH1_BL, CH1_LCF and CH1_LCVS only the sync channel 1 STDI locks and CH1_STDI_DVALID is set to 1
1	Update CH1_BL, CH1_LCF,CH1_LCVS from the sync channel 1 STDI as they are measured

BYPASS_STDI2_LOCKING, Addr 44 (CP), *Address 0xF5[0]*

Bypass STDI locking for sync channel 2

Function

BYPASS_STDI2_LOCKING	Description
0 «	Update CH2_BL, CH2_LCF and CH2_LCVS only the sync channel 2 STDI locks and CH2_STDI_DVALID is set to 1
1	Update CH2_BL, CH2_LCF,CH2_LCVS from the sync channel 2 STDI as they are measured

CH1_TRIG_STDI, Addr 44 (CP), *Address 0x86[2]*

Trigger synchronization source and polarity detector for sync channel 1 STDI. A 0 to 1 transition in this bit restarts the auto-sync detection algorithm. This is not a self-clearing bit and must be set to 0 to prepare for next trigger.

Function

CH1_TRIG_STDI	Description
0 «	Default value - transition 0 to 1 restarts auto-sync detection algorithm
1	Transition 0 to 1 restarts auto-sync detection algorithm

CH2_TRIG_STDI, Addr 44 (CP), *Address 0x42[2]*

Triggers standard identification of sync channel 2 STDI. A 0 to 1 transition on this bit triggers the STDI measurements. This is not self-clearing and must be set to 0 to prepare for the next STDI measurements.

Function

CH2_TRIG_STDI	Description
0 «	Default value - transition 0 to 1 restarts auto-sync detection algorithm
1	Transition 0 to 1 restarts auto-sync detection algorithm

CH1_STDI_DVALID, Addr 44 (CP), *Address 0xB1[7] (Read Only)*

This bit is set when the measurements performed by sync channel 1 STDI are completed. High level signals validity for CH1_BL, CH1_LCF, CH1_LCVS, CH1_FCL, and CH1_STDI_INTLCD parameters. To prevent false readouts, especially during signal acquisition, CH1_SDTI_DVALID only goes high after four fields with same length are recorded. As a result, STDI measurements can take up to five fields to finish.

Function

CH1_STDI_DVALID	Description
0 «	Sync channel 1 STDI measurement are not valid
1	Sync channel 1 STDI measurement are valid

CH2_STDI_DVALID, Addr 44 (CP), *Address 0x49[7] (Read Only)*

This bit is set when the measurements performed by sync channel 2 STDI are completed. High level signals validity for CH2_BL, CH2_LCF, CH2_LCVS, CH2_FCL, and CH2_STDI_INTLCD parameters. To prevent false readouts, especially during signal acquisition, CH2_SDTI_DVALID only goes high after four fields with same length are recorded. As a result, STDI measurements can take up to five fields to finish.

Function

CH2_STDI_DVALID	Description
0 «	Sync channel 2 STDI measurement are not valid
1	Sync channel 2 STDI measurement are valid

10.8.3 Detailed Mechanism of STDI Block Horizontal/Vertical Lock Mechanism

10.8.3.1 STDI Horizontal Locking Operation

For the STDI horizontal locking operation, the STDI block compares adjacent line length differences (in XTAL clock cycles) with the programmed threshold. If 128 consecutive adjacent lines lengths are within the threshold, the STDI horizontally locks to the incoming video.

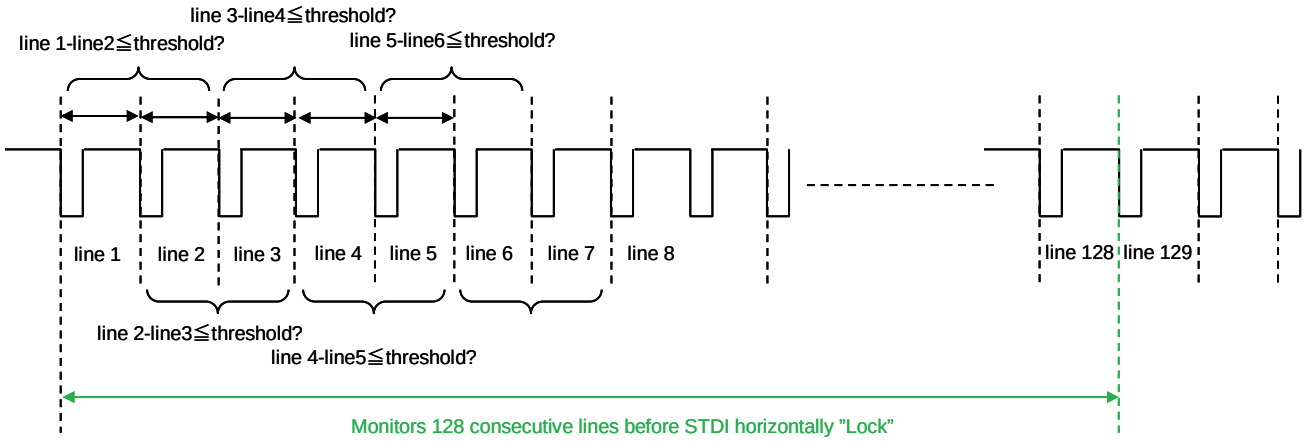


Figure 123: STDI Horizontal Locking Operation

Once the STDI locks to the incoming video, it registers the first BL measurement (first eight lines) as latched data (absolute line length: L) and keeps monitoring and comparing each successive line length with the absolute line length (L/8).

The STDI horizontally unlocks if 128 consecutive lines have a line length greater than the threshold.

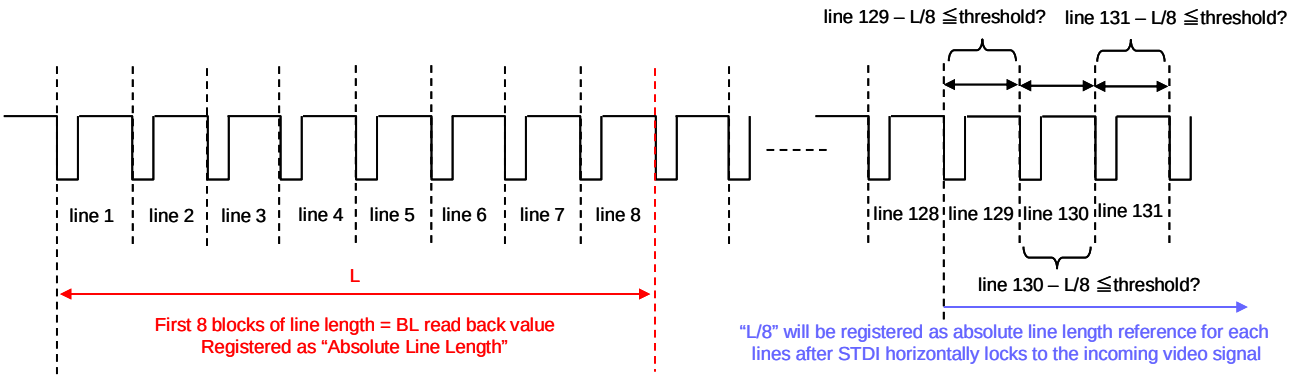


Figure 124: STDI HSync Monitoring Operation

10.8.3.2 STDI Vertical Locking

The STDI block compares adjacent field length differences and VSync lengths in line counts and compares them with a threshold. If four consecutive adjacent field lengths (LCF) and line counts in VSync (LCVS) are within the threshold, the STDI vertically locks to the incoming video.

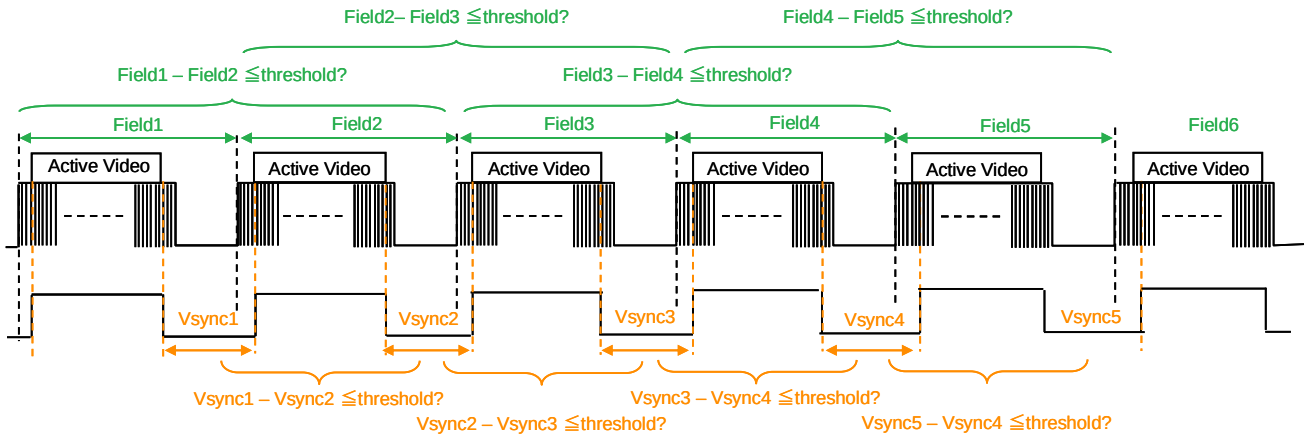


Figure 125: STDI Vertical Locking Operation

Once the STDI locks to the incoming video, the STDI registers the latest field length/VSync length as latched data (absolute field length: F, absolute VSync length: V). The STDI keeps monitoring and comparing Field/VSync lengths with the respective absolute length (F, V) once vertically locked. The STDI vertically unlocks if four consecutive Field or VSync lengths are greater than the respective threshold.

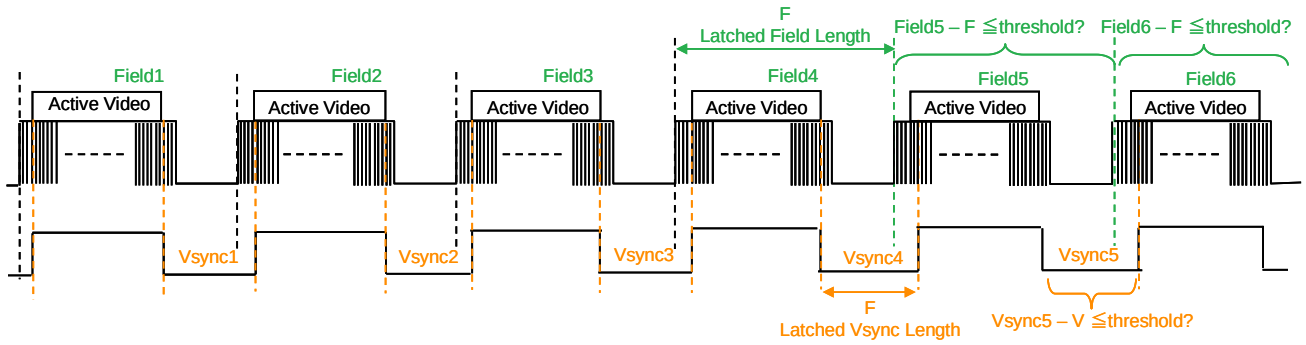


Figure 126: STDI VSync Monitoring Operation

CH1_BL[13:0], Addr 44 (CP), *Address 0xB1[5:0]; Address 0xB2[7:0] (Read Only)*

A readback for the Block Length for sync channel 1. Number of crystal clock cycles in a block of eight lines of incoming video. This readback is valid if CH1_STDI_DVALID is high.

Function

CH1_BL[13:0]	Description
xxxxxxxxxxxxx	Readback value

CH2_BL[13:0], Addr 44 (CP), *Address 0x49[5:0]; Address 0x4A[7:0] (Read Only)*

A readback for the sync channel 2 Block Length. Number of crystal clock cycles in a block of eight lines of incoming video. This readback is valid if CH2_STDI_DVALID is high.

Function

CH2_BL[13:0]	Description
xxxxxxxxxxxxx	Readback value

CH1_LCVS[4:0], Addr 44 (CP), *Address 0xB3[7:3] (Read Only)*

A readback for the sync channel 1 Line Count in a VSync. Number of lines in a VSync period measured on sync channel 1. The readback from this field is valid if CH1_STDI_DVALID is high.

Function

CH1_LCVS[4:0]	Description
xxxxx	Readback value

CH2_LCVS[4:0], Addr 44 (CP), *Address 0x4B[7:3] (Read Only)*

A readback for the sync channel 2 Line Count in a VSync.
Number of lines in a VSync period measured on sync channel 2. The readback from this field is valid if CH2_STDI_DVALID is high.

Function

CH2_LCVS[4:0]	Description
xxxxx	Readback value

CH1_LCF[10:0], Addr 44 (CP), *Address 0xB3[2:0]; Address 0xB4[7:0] (Read Only)*

A readback for the sync channel 1 Line Count in a Field
Number of lines between two VSyncs measured on sync channel 1. The readback from this field is valid if CH1_STDI_DVALID is high.

Function

CH1_LCF[10:0]	Description
xxxxxxxxxxx	Readback value

CH2_LCF[10:0], Addr 44 (CP), *Address 0x4B[2:0]; Address 0x4C[7:0] (Read Only)*

A readback for the sync channel 2 Line Count in a Field
Number of lines between two VSyncs measured on sync channel 2. The readback from this field is valid if CH2_STDI_DVALID is high.

Function

CH2_LCF[10:0]	Description
xxxxxxxxxxx	Readback value

CH1_FCL[12:0], Addr 44 (CP), *Address 0xB8[4:0]; Address 0xB9[7:0] (Read Only)*

A readback for the sync channel 1 Field Count Length
Number of crystal clock cycles between successive VSyncs measured by sync channel 1 STDI or in 1/256th of a field. The readback from this field is valid if CH1_STDI_DVALID is high.

Function

CH1_FCL[12:0]	Description
xxxxxxxxxxxxx	Readback value

CH2_FCL[12:0], Addr 44 (CP), *Address 0x4D[4:0]; Address 0x4E[7:0] (Read Only)*

A readback for the sync channel 2 Field Count Length
Number of crystal clock cycles between successive VSyncs measured by sync channel 2 STDI or in 1/256th of a field. The readback from this field is valid if CH2_STDI_DVALID is high.

Function

CH2_FCL[12:0]	Description
xxxxxxxxxxxx	Readback value

CH1_STDI_INTLCD, Addr 44 (CP), *Address 0xB1[6] (Read Only)*

Interlaced vs. progressive mode detected by sync channel 1 STDI. The readback from this register is valid if CH1_STDI_DVALID is high.

Function

CH1_STDI_INTLCD	Description
0 «	Indicates a video signal on sync channel 1 with non interlaced timing.
1	Indicates a signal on sync channel 1 with interlaced timing.

CH2_STDI_INTLCD, Addr 44 (CP), *Address 0x49[6] (Read Only)*

Interlaced vs. progressive mode detected by sync channel 2 STDI. The readback from this register is valid if CH2_STDI_DVALID is high.

Function

CH2_STDI_INTLCD	Description
0 «	Indicates a video signal on sync channel 2 with non interlaced timing.
1	Indicates a signal on sync channel 2 with interlaced timing.

10.8.3.3 STDI Usage

Figure 127 shows a flowchart of the intended usage of the STDI block.

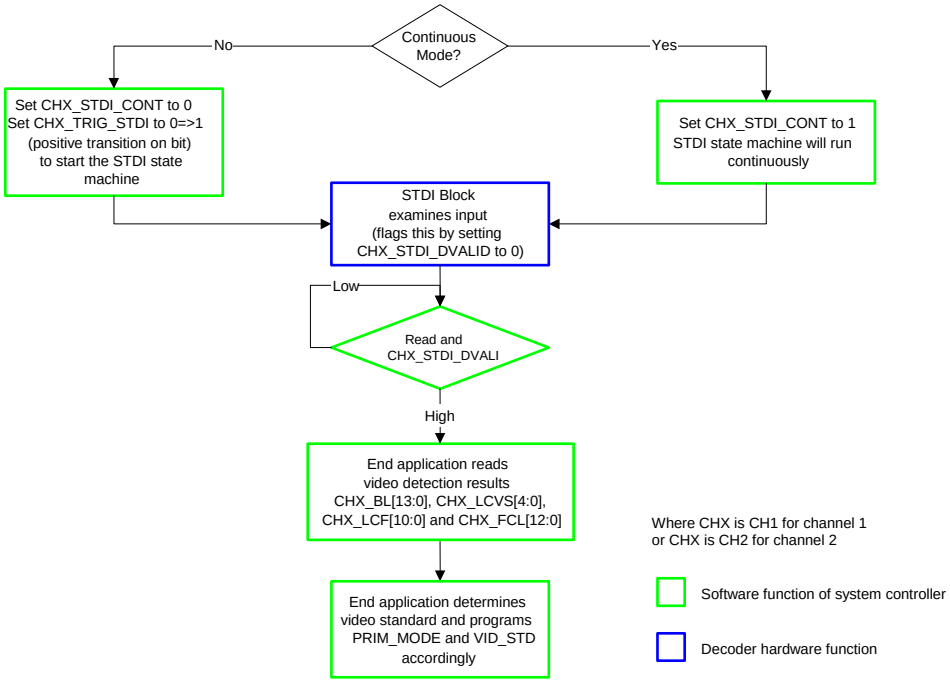


Figure 127: STDI Usage Flowchart

10.8.3.4 STDI Readback Values for SD, PR, and HD

Table 56: STDI Readback Values for SD, PR, and HD

Standard	CHx_BL[13:0] 28.63636 MHz XTAL	CHx_LCF[10:0]	CHx_LCVS[4:0]	FCL[12:0] 28.63636 MHz XTAL
720p SMPTE 296M	5091	750	4 to 5	1868
1125i SMPTE 274M	6788	562 to 563	4 to 5	1868
525p BT 1358	7270	525	5 to 6	1868
625p BT 1358	7331	625	4 to 5	2237
1250i BT 709/SMPTE 295	7331	625	1	4474
1125i SMPTE 274M 6	8145	562 to 563	4 to 5	1868
1125p SMPTE 274M 10	848	1125	4 to 5	1868
525i SD	14560	262 to 263	3	1868
625i SD	14662	312 to 313	2 to 3	2237

10.8.3.5 STDI Readback Values for GR

Table 57: STDI Results for Graphics Standards

Standard	CHx_BL[13:0] 28.63636 MHz XTAL	CHx_LCF[10:0]	CHx_LCVS[4:0]	FCL[12:0] 28.63636 MHz XTAL
XGA 85	3327	805 to 808	0 to 3	1316
SXGA 60	3571	1063 to 1066	0 to 3	1868
XGA 75	3808	797 to 800	0 to 3	1493
XGA 70	4048	800 to 806	0 to 6	1598
SVGA 85	4259	628 to 631	0 to 3	1316
XGA 60	4726	800 to 806	0 to 6	1868
SVGA 72	4756	660 to 666	0 to 6	1554
SVGA 75	4878	622 to 625	0 to 3	1493
VGA 85	5286	506 to 509	0 to 3	1316
VGA 72	6042	517 to 520	0 to 3	1554
SVGA 60	6039	624 to 628	0 to 4	1868
VGA 75	6098	497 to 500	0 to 3	1493
SVGA 56	6508	623 to 625	0 to 2	1997
VGA 60	7272	523 to 525	0 to 2	1868

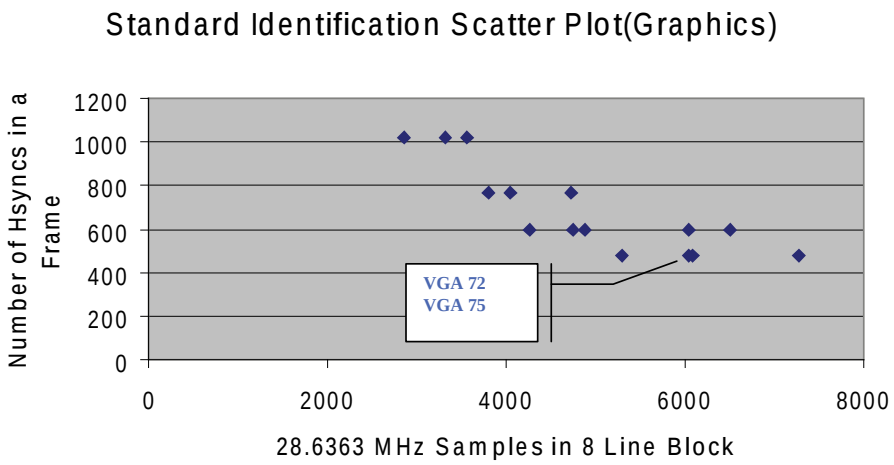


Figure 128: STDI Values for GR Mode (Plot)

Note: Although the two points for VGA72 and VGA75 look very close, it is anticipated that the difference in the parameters is sufficient to distinguish between them.

10.9 CP OUTPUT SYNCHRONIZATION SIGNAL POSITIONING

The ADV7842 overall synchronization processing flow is shown in the block diagram in Figure 129. The user can reposition the synchronization signal output from the regenerated input synchronization signal within the CP block with the control bits marked in red in Figure 129.

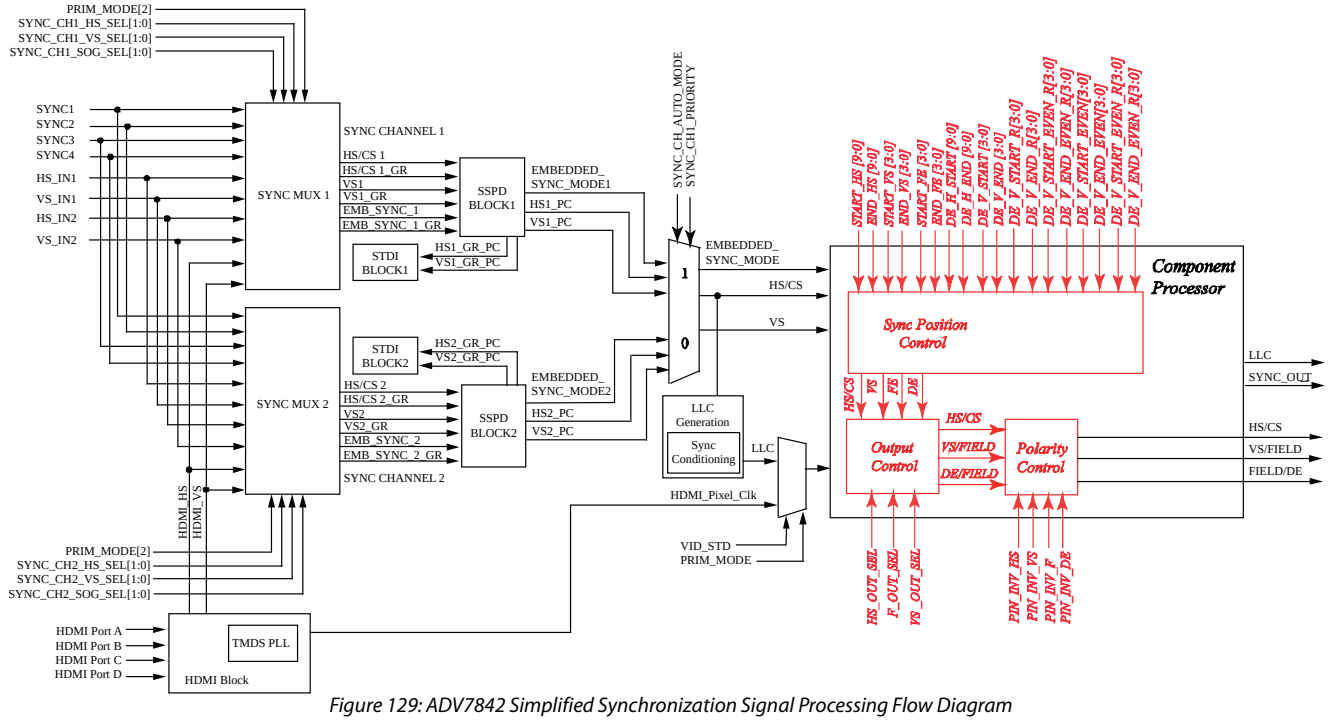


Figure 129: ADV7842 Simplified Synchronization Signal Processing Flow Diagram

As shown in Figure 129, the ADV7842 CP can output following three primary and two secondary synchronization signals, which are

controlled by the output control block in the CP block.

Primary:

- Horizontal synchronization timing reference output on the HS/CS pin
- Vertical synchronization timing reference output on the VS/FIELD pin
- Field timing reference output on the FIELD/DE pin or as a secondary signal on the VS/FIELD pin

Secondary:

- CS timing reference output shared with the HS pin
- DE (indicates active region) shared with the FIELD pin

Timing reference signals with shared pins are controlled via I^2C.

Table 58: CP Synchronization Signal Output Pins

Pin Name	Primary Signal (Default)	Secondary Signal	Controlled by I2C Bit
HS/CS	HS out	CS out	HS_OUT_SEL[1:0]
FIELD/DE	FIELD out	DE out	F_OUT_SEL
VS/FIELD	VS out	FIELD out	VS_OUT_SEL

The user can program the primary and secondary synchronization signals, repositioning them in order to control the display area, as shown in Figure 130. Note that VBI (Vertical Blanking Interval) positioning control is also available for the auto graphics mode. Refer to Section 10.15 for further details.

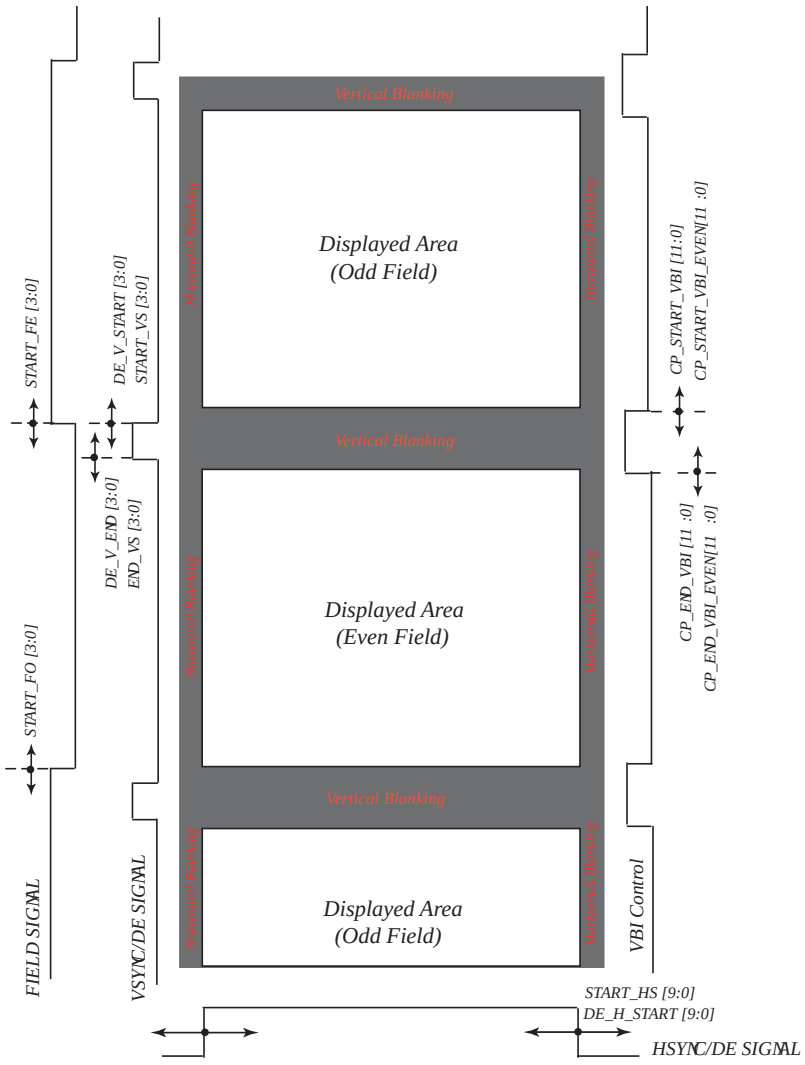


Figure 130: Synchronization Repositioning and Displayed Area

10.9.1 CP Primary Synchronization Signals

The three primary synchronization signals have certain default positions, depending on the video standard in use.

To allow for a glueless interface to downstream ICs, there is the facility to adjust the position of edges on the three primary synchronization signals Figure 131,

Figure 132,

Figure 133,

Figure 134,

Figure 135,
Figure 136, Figure 137, Figure 138, show the nominal position of HS, VS, and FIELD. The positions of those signals can be adjusted in both directions by using the following controls:

- START_HS[9:0]
- END_HS[9:0]
- START_VS **[3:0]**
- END_VS **[3:0]**
- START_FE[3:0]
- START_FO[3:0]

All six above parameters are given as signed values. This means that rather than adjusting the absolute position of a signal, these adjustments allow the user to advance (negative value) or delay (positive value) the respective timing reference signals.

In addition, the polarity of the synchronization output signals can be inverted by using:

- INV_HS_POL
- INV_F_POL
- INV_VS_POL

10.9.2 HSync Timing Controls

Programming the registers listed in this section, the HS signal as shown in Figure 131 can be adjusted in the described manner.

Table 59: HS Default Timing

Symbol	**Characteristic**	**Note**	**525i**	**625i**	**525p**	**625p**	**720p**	**1080i**	**1080p**
a	HS to start of active video	Default	118	128	116	126	256	188	118
			All values are for 1x outputs						
d	HS width	Default	64	64	64	64	40	44	44
b	Active video samples		720	720	720	720	1280	1920	1920
c	Total samples/line		858	864	858	864	1650	2200/ 2376	2200

Table 60: HS Default Timing (Continued 1)

Symbol	Characteristic	Note	680x480 at 60 Hz	640x480 at 72 Hz	640x480 at 75 Hz	640x480 at 85 Hz
a	HS to start of active video	Default	140	164	180	132
			All values are for 1x outputs			
d	HS width	Default	96	40	64	56
b	Active video samples		640	640	640	640
c	Total samples/line		800	832	840	832

Table 61: HS Default Timing (Continued 2)

Symbol	Characteristic	Note	800x600 at 56 Hz	800x600 at 60 Hz	800x600 at 72 Hz	800x600 at 75 Hz	800x600 at 85 Hz
a	HS to start of active video	Default	196	212	180	236	212
			All values are for 1x outputs				
d	HS width	Default	72	128	120	80	64
b	Active video samples		800	800	800	800	800
c	Total samples/line		1024	1056	1040	1056	1048

Table 62: HS Default Timing (Continued 3)

Symbol	Characteristic	Note	1024x768 at 60 Hz	1024x768 at 70 Hz	1024x768 at 75 Hz	1024x768 at 85 Hz
a	HS to start of active video	Default	292	276	268	300
			All values are for 1x outputs			
d	HS width	Default	136	136	96	96
b	Active video samples		1024	1024	1024	1024
c	Total samples/line		1344	1328	1312	1376

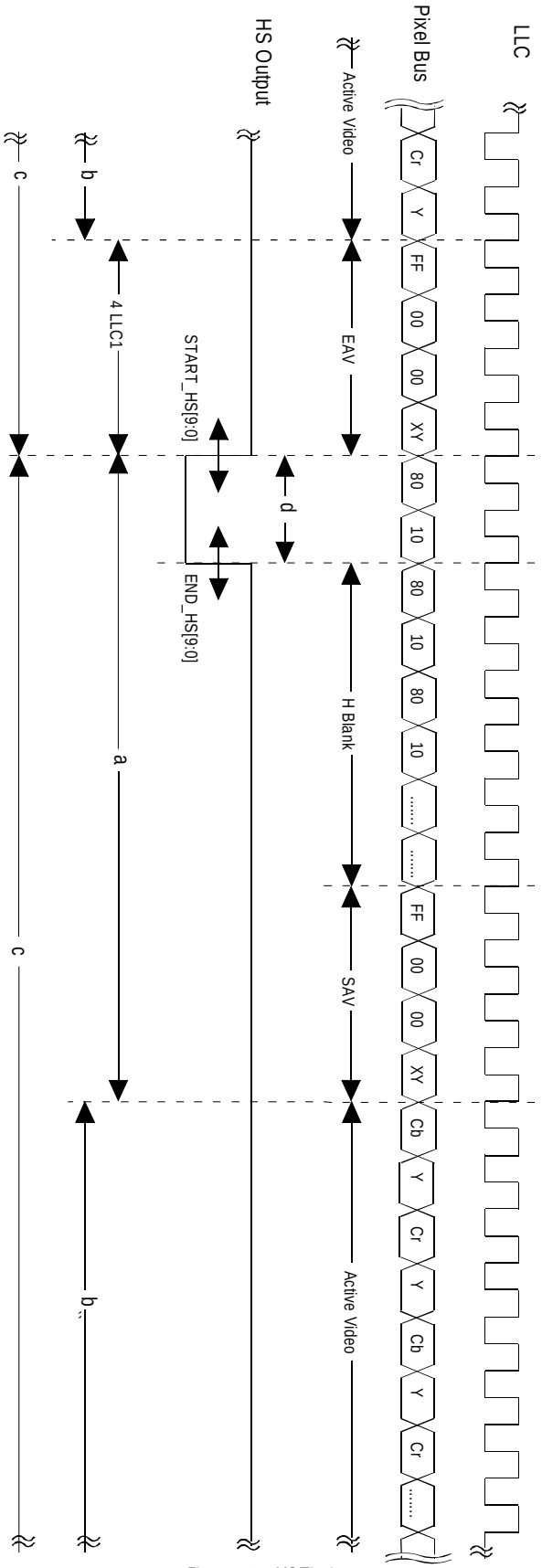


Figure 131: HS Timing

START_HS[9:0], Addr 44 (CP), *Address 0x7C[3:2]; Address 0x7E[7:0]*

A control to shift the position of the leading edge of the HSync output by the CP core.
This register stores a signed value in a 2's complement format. START_HS[9:0] is the number of pixel clocks by which the leading edge of the HSync is shifted (e.g. 0x3FF corresponds to a shift of 1 pixel clock away form the active video, 0x005 corresponds to a shift of 5 pixel clocks toward the active video).

Function

START_HS[9:0]	Description
0x000 «	Default value.
0x000 to 0x1FF	The leading edge of the HSync is shifted toward the active video.
0x200 to 0x3FF	The leading edge of the HSync is shifted away from the active video.

Examples of how to control the BEGIN of the HS timing signal:

START_HS[9:0]	Hex	Result	Note
0000000000 ↻	0x000	No move	Default
0000000001	0x001	1 x $\frac{1}{LLC}$ sec shift later than default[1]	Minimum →
0100000000	0x100	256 x $\frac{1}{LLC}$ sec shift later than default	
0111111111	0x1FF	511 x $\frac{1}{LLC}$ sec shift later than default	Maximum →
1111111111	0x3FF	1 x $\frac{1}{LLC}$ sec shift earlier than default[2]	Minimum ←
1011111111	0x3FE	256 x $\frac{1}{LLC}$ sec shift earlier than default	
1000000000	0x200	512 x $\frac{1}{LLC}$ sec shift earlier than default	Maximum ←

[1]HS START closer to active video

[2]HS START away from active video

END_HS[9:0], Addr 44 (CP), *Address 0x7C[1:0]; Address 0x7D[7:0]*

A control to shift the position of the trailing edge of the HSync output by the CP core.
This register stores a signed value in a 2's complement format. HS_END[9:0] is the number of pixel clock by which the leading edge of the HSync is shifted (e.g. 0x3FF corresponds to a shift of 1 pixel clock away form the active video, 0x005 corresponds to a shift of 5 pixel clocks toward the active video).

Function

END_HS[9:0]	Description
0x000 «	Default value.
0x000 to 0x1FF	The trailing edge of the HSync is shifted toward the active video.
0x200 to 0x3FF	The trailing edge of the HSync is shifted away from the active video.

Examples of how to control the end of the HS timing signal:

END_HS[9:0]	Hex	Result	Note
0000000000 ↻	0x000	No move (default)	

END_HS[9:0]	Hex	Result	Note
0000000001	0x001	1 x $\frac{1}{LLC}$ sec shift later than default[1]	Minimum →
0100000000	0x100	256 x $\frac{1}{LLC}$ sec shift later than default	
0111111111	0x1FF	511 x $\frac{1}{LLC}$ sec shift later than default	Maximum →
1111111111	0x3FF	1 x $\frac{1}{LLC}$ sec shift earlier than default[2]	Minimum ←
1011111111	0x3FE	256 x $\frac{1}{LLC}$ sec shift earlier than default	
1000000000	0x200	512 x $\frac{1}{LLC}$ sec shift earlier than default	Maximum ←

[1]Closer to active video
[2]Away from active video

EIA_861_COMPLIANCE, Addr 44 (CP), *Address 0x69[2]*

A control to implement compliance to the CEA 861 standard for 525p inputs. This bit affects the start of the VBI for the 525p standard only.

Function

EIA_861_COMPLIANCE	Description
0	The VBI region starts on line 1.
1 «	The VBI region starts on line 523. The start of the VBI region is compliant with the 861 specification.

10.9.3 VSync Timing Controls

- Programming of the VS timing signals is listed in this section. The VS signal is shown in
-
- Figure 132,
-
- Figure 133,
-
- Figure 134,
-
- Figure 135,
- Figure 136, Figure 137, Figure 138 and can be adjusted in the described manner.

Table 63: VS Default Timing

Characteristic	Units	Direction	525i	625i	525p	625p	720p	1080i
START_VS range maximum	Lines	→	7	7	7	7	7	7
START_VS range minimum	Lines	←	8	8	8	8	8	8
END_VS range maximum	Lines	→	7	7	7	7	7	7
END_VS range minimum	Lines	←	8	8	8	8	8	8

START_VS [3:0], Addr 44 (CP), *Address 0x7F [7:4]*

A control to shift the position of the leading edge of the VSync output by the CP core.
This register stores a signed value in a 2's complement format. START_VS[3:0] is the number of lines by which the leading edge of the VSync is shifted (e.g. 0x0F corresponds to a shift by 1 line toward the active video, 0x01 corresponds to a shift of 1 line away from the active video).

Function

START_VS[3:0]	Description
0x0 «	Default value.
0x0 to 0x7	The leading edge of the VSync is shifted toward the active video.
0x8 to 0xF	The leading edge of the VSync is shifted away from the active video.

Examples of how to control the start of the VS timing signal:

START_VS[3:0]	Hex	Result	Note
0000 ↻	0x0	No move (default)	
0001	0x1	1 HS shift later than default[1]	Minimum →
0011	0x3	3 HS shift later than default	
0111	0x0	7 HS shift later than default	Maximum →
1111	0xF	1 HS shift earlier than default[2]	Minimum ←
1101	0xD	3 HS shift earlier than default	
1000	0x8	8 HS shift earlier than default	Maximum ←

[1]VS closer to start of active video

[2]VS away from start of active video

END_VS [3:0], Addr 44 (CP), *Address 0x7F [3:0]*

A control to shift the position of the trailing edge of the VSync output by the CP core.
This register stores a signed value in a 2's complement format. SEND_VS[3:0] is the number of lines by which the trailing edge of the VSync is shifted (e.g. 0x0F corresponds to a shift of 1 line toward the active video, 0x01 corresponds to a shift of 1 line away from the active video).

Function

END_VS[3:0]	Description
0x0 «	Default value.
0x0 to 0x7	The trailing edge of the VSync is shifted toward the active video.
0x8 to 0xF	The trailing edge of the VSync is shifted away from the active video.

START_VS_EVEN [3:0], Addr 44 (CP), *Address 0x89[7:4]*

A control to shift the position of the leading edge of the VSync output by the CP core. This register stores a signed value in a 2's complement format. START_VS_EVEN[3:0] is the number of lines by which the leading edge of the Vsync is shifted (e.g. 0x0F corresponds to a shift by 1 line toward the active video, 0x01 corresponds to a shift of 1 line away from the active video).

Function

START_VS_EVEN[3:0]	Description
0x0 to 0x7	The leading edge of the even VSync is shifted toward the active video.
0x8 to 0xF	The leading edge of the even VSync is shifted away from the active video.

END_VS_EVEN [3:0], Addr 44 (CP), *Address 0x89[3:0]*

A control to shift the position of the trailing edge of the VSync output by the CP core. This register stores a signed value in a 2's complement format. END_VS_EVEN[3:0] is the number of lines by which the trailing edge of the Vsync is shifted (e.g. 0x0F corresponds to a shift of 1 line toward the active video, 0x01 corresponds to a shift of 1 line away from the active video).

Function

END_VS_EVEN[3:0]	Description
0x0 to 0x7	The trailing edge of the even VSync is shifted toward the active video.
0x8 to 0xF	The trailing edge of the even VSync is shifted away from the active video.

Examples of how to control the end of the VS timing signal:

END_VS[3:0]	Hex	Result	Note
0000 ↻	0x0	No move (default)	
0001	0x1	1 HS shift later than default[1]	Minimum →
0011	0x3	3 HS shift later than default	
0111	0x0	7 HS shift later than default	Maximum →
1111	0xF	1 HS shift earlier than default[2]	Minimum ←
1101	0xD	3 HS shift earlier than default	
1000	0x8	8 HS shift earlier than default	Maximum ←

[1]VS closer to start of active video

[2]VS away from start of active video

10.9.4 DE Timing Controls

DE_H_END [9:0], Addr 44 (CP), *Address 0x8B [1:0]; Address 0x8C [7:0]*

A control to vary the trailing edge position of the DE signal output by the CP core.
This register stores a signed value in a 2's complement format. The unit of DE_H_END [9:0] is one pixel clock.

Function

DE_H_END[9:0]	Description
0x200	-512 pixels of shift
0x3FF	-1 pixel of shift
0x000 «	Default value (no shift)
0x001	+1 pixel of shift
0x1FF	+511 pixels

DE_H_START [9:0], Addr 44 (CP), *Address 0x8B [3:2]; Address 0x8D [7:0]*

A control to vary the leading edge position of the DE signal output by the CP core.
This register stores a signed value in a 2's complement format. The unit of DE_H_START [9:0] is one pixel clock.

Function

DE_H_START[9:0]	Description
0x200	-512 pixels of shift
0x3FF	-1 pixel of shift
0x000 «	Default value (no shift)
0x001	+1 pixel of shift
0x1FF	+511 pixels

DE_V_START [3:0], Addr 44 (CP), *Address 0x8E [7:4]*

A control to vary the start position of the VBI region.
This register stores a signed value represented in a 2's complement format. The unit of DE_V_START [9:0] is one line.

Function

DE_V_START[3:0]	Description
1000	-8 lines of shift
1111	-1 line of shift
0000 «	Default
0001	+1 line of shift
0111	+7 lines of shift

DE_V_END [3:0], Addr 44 (CP), *Address 0x8E [3:0]*

A control to vary the position of the end of the VBI region.
This register stores a signed value represented in a 2's complement format. The unit of DE_V_START [9:0] is one line.

Function

DE_V_END[3:0]	Description
1000	-8 lines of shift
1111	-1 line of shift
0000 «	Default
0001	+1 line of shift
0111	+7 lines of shift

DE_V_START_EVEN [3:0], Addr 44 (CP), *Address 0x88[7:4]*

A control to adjust the start position of the VBI region in even field. This register stores a signed value represented in a 2's complement format. The unit of adjustment is one pixel clock.

Function

DE_V_START_EVEN[3:0]	Description
1000 … 1111	(-8 lines … -1 line)
0000 «	Default value (0 lines)
0001 … 0111	(1 line … 7 lines)

DE_V_END_EVEN [3:0], Addr 44 (CP), *Address 0x88[3:0]*

A control to adjust the end position of the VBI region in even field. This register stores a signed value represented in a 2's complement format. The unit of adjustment is one pixel clock.

Function

DE_V_END_EVEN[3:0]	Description
1000 … 1111	(-8 lines … -1 line)
0000 «	Default value (0 lines)
0001 … 0111	(1 line … 7 lines)

The following controls are used to adjust 3D HDMI standards only.

DE_V_START_R [3:0], Addr 44 (CP), *Address 0x30[7:4]*

A control to adjust the start position of the extra VBI region between L and R fields during an odd field in 3D TV video field alternative packing format supported by HDMI.
This register stores a signed value represented in a 2's complement format. The unit of adjustment is one line.

Function

DE_V_START_R[3:0]	Description
1000 … 1111	(-8 lines … -1 line)
0000 «	Default value (0 lines)
0001 … 0111	(1 line … 7 lines)

DE_V_END_R [3:0], Addr 44 (CP), *Address 0x30[3:0]*

A control to adjust the end position of the extra VBI region between L and R fields during the odd field in the 3D TV field alternative packing format supported by HDMI.
This register stores a signed value represented in a 2's complement format. The unit of adjustment is one line.

Function

DE_V_END_R[3:0]	Description
1000 … 1111	(-8 lines … -1 line)
0000 «	Default value (0 lines)
0001 … 0111	(1 line … 7 lines)

DE_V_START_EVEN_R [3:0], Addr 44 (CP), *Address 0x31[7:4]*

A control to adjust the start position extra VBI region between L and R fields during even field in the 3D TV field alternative packing format supported by HDMI.
This register stores a signed value represented in a 2's complement format. The unit of adjustment is one line.

Function

DE_V_START_EVEN_R[3:0]	Description
1000 … 1111	(-8 lines … -1 line)
0000 «	Default value (0 lines)
0001 … 0111	(1 line … 7 lines)

DE_V_END_EVEN_R [3:0], Addr 44 (CP), *Address 0x31[3:0]*

A control to adjust the end position of the extra VBI region between L and R fields during even field in the 3D TV field alternative packing format supported by HDMI.
This register stores a signed value represented in a 2's complement format. The unit of adjustment is one line.

Function

DE_V_END_EVEN_R[3:0]	Description
1000 … 1111	(-8 lines … -1 line)
0000 «	Default value (0 lines)
0001 … 0111	(1 line … 7 lines)

10.9.5 FIELD Timing Controls

Programming of the FIELD timing signals is listed in this section. The FIELD signal is shown in

Figure 132,

Figure 133,

Figure 134, Figure 137 can be adjusted in the described manner. (Progressive systems do not have a FIELD signal.)

Table 64: FIELD Default Timing

Characteristic	Units	525i	625i	525p	625p	720p	1080i
START_FO END_FO range maximum	Line	7	7	n/a	n/a	n/a	7
START_FO END_FO range maximum	Line	8	8	n/a	n/a	n/a	8

START_FE[3:0], Addr 44 (CP), *Address 0x80[7:4]*

A control to shift the position of the start of even field edge of the FIELD signal output by the CP core
This register stores a signed value in a 2's complement format. START_FE[3:0] the number of lines by which the start of the even fields edge of the FIELD signal is shifted (e.g. 0x0D corresponds to a shift of 3 lines toward the active video, 0x05 corresponds to a shift of 5 line away from the active video).

Function

START_FE[3:0]	Description
0x0 «	Default value.
0x0 to 0x7	The edge of the FIELD signal corresponding to the start of the even field is shifted toward the active video.
0x8 to 0xF	The trailing of the FIELD signal corresponding to the start of the even field is shifted away from the active video.

Examples of how to control the Even field section of the FIELD timing signal:

START_FE[3:0]	Hex	Result	Note
0000	0x0	No move (default)	
0001	0x1	1 HS shift later than default1	Minimum →
0011	0x3	3 HS shift later than default	
0111	0x7	7 HS shift later than default	Maximum →
1111	0xF	1 HS shift earlier than default2	Minimum ←
1101	0xD	3 HS shift earlier than default	
1000	0x8	8 HS shift earlier than default	Maximum ←

[1]Closer to active video

[2]Away from active video

START_FO[3:0], Addr 44 (CP), *Address 0x80[3:0]*

A control to shift the position of the start of odd field edge of the FIELD signal output by the CP core
This register stores a signed value in a 2's complement format. START_FO[3:0] the number of lines by which the start of the odd fields edge of the FIELD signal is shifted (e.g. 0x0D corresponds to a shift of 3 lines toward the active video, 0x05 corresponds to a shift of 5 line away from the active video).

Function

START_FO[3:0]	Description
0x0 «	Default value.
0x0 to 0x7	The edge of the FIELD signal corresponding to the start of the odd field is shifted toward the active video.
0x8 to 0xF	The trailing of the FIELD signal corresponding to the start of the odd field is shifted away from the active video.

Examples of how to control the Odd field section of FIELD timing signal:

START_FO[3:0]	Hex	Result	Note
0000	0x0	No move (default)	
0001	0x1	1 HS shift later than default1	Minimum →
0011	0x3	3 HS shift later than default	
0111	0x0	7 HS shift later than default	Maximum →
1111	0xF	1 HS shift earlier than default2	Minimum ←
1101	0xD	3 HS shift earlier than default	
1000	0x8	8 HS shift earlier than default	Maximum ←

[1]Closer to active video

[2]Away from active video

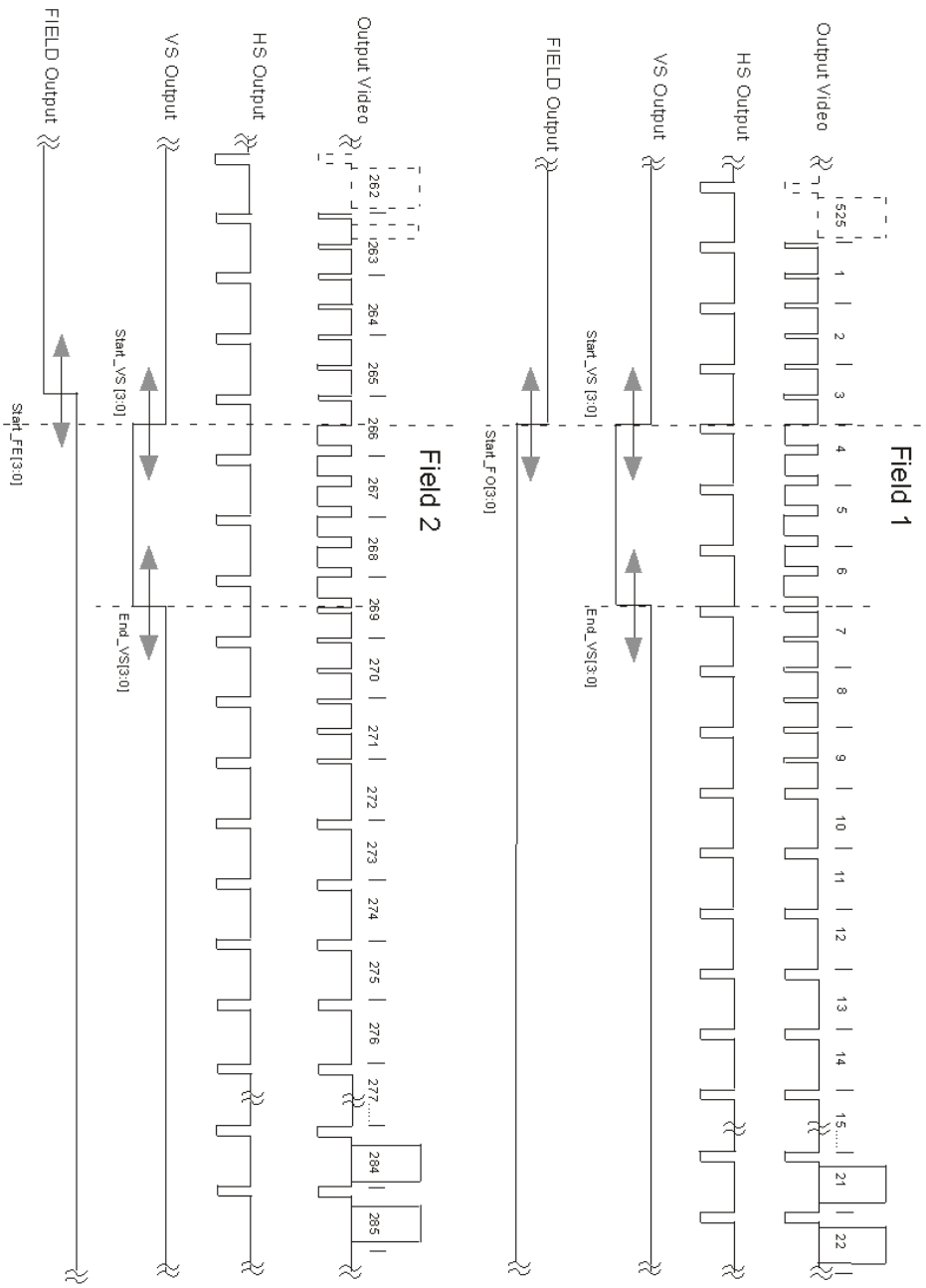


Figure 132: 525i VS Timing

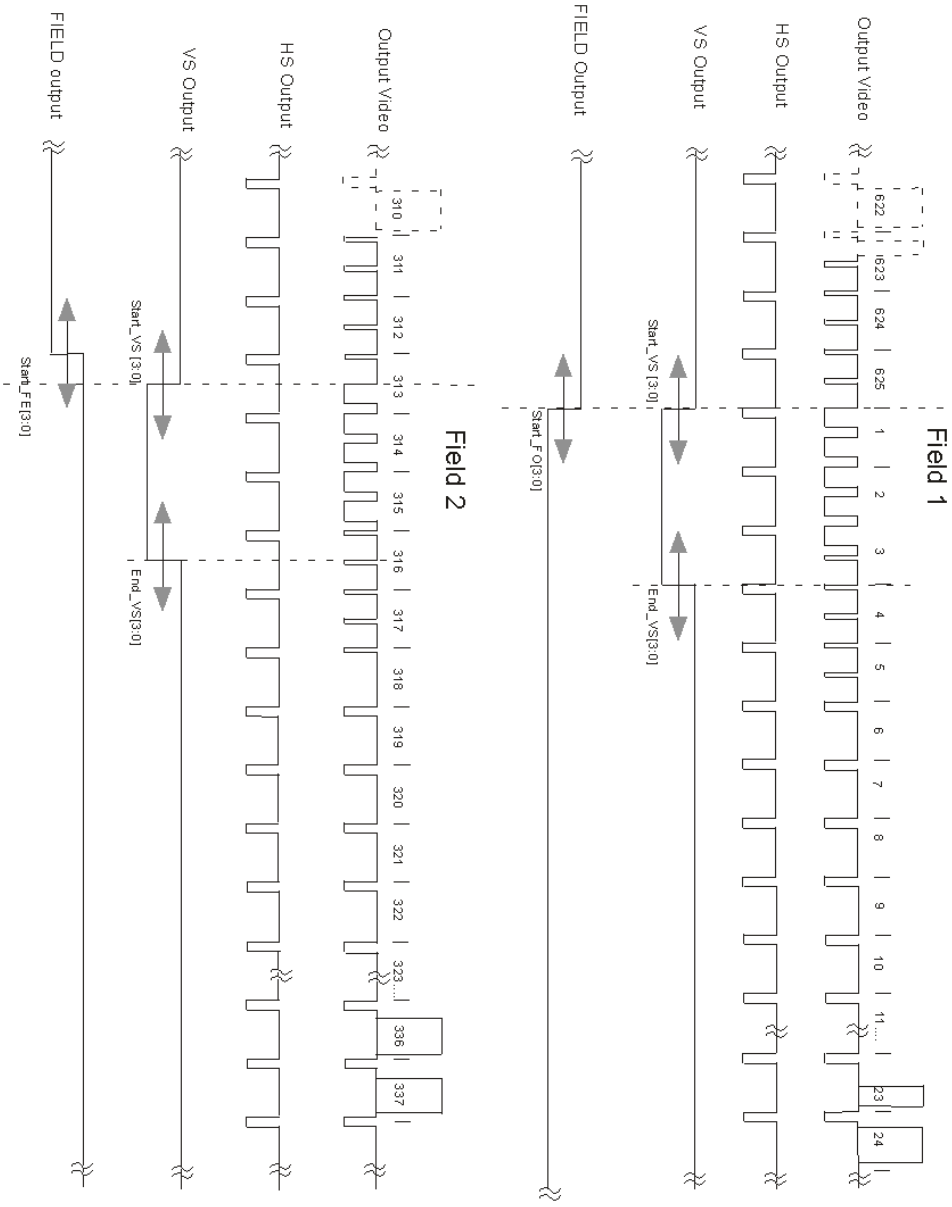


Figure 133: 625i VS Timing

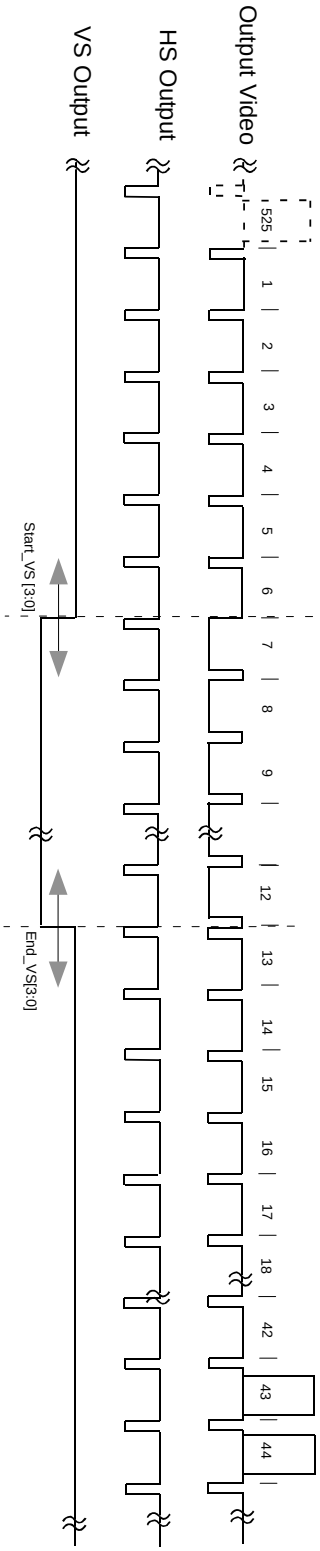


Figure 134: 525p VS Timing

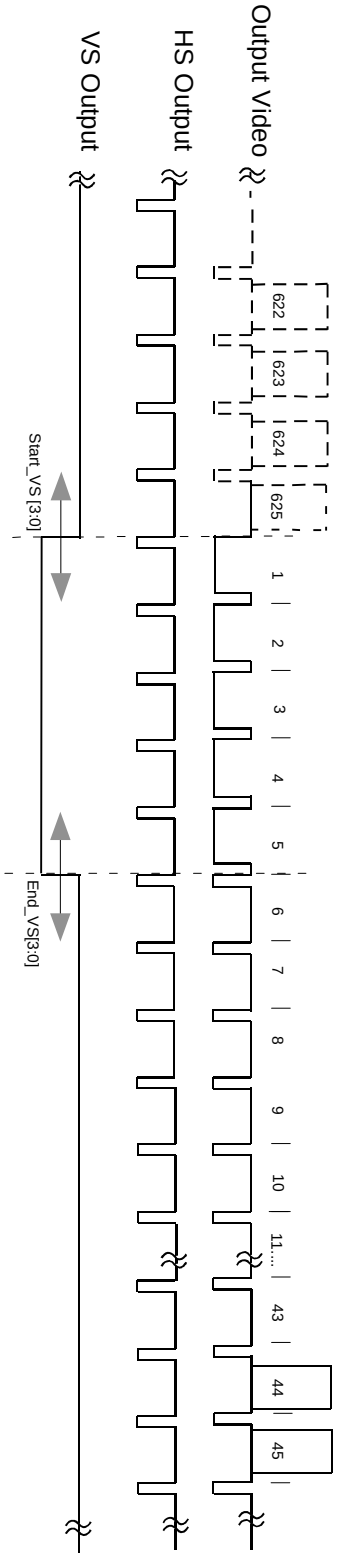


Figure 135: 625p VS Timing

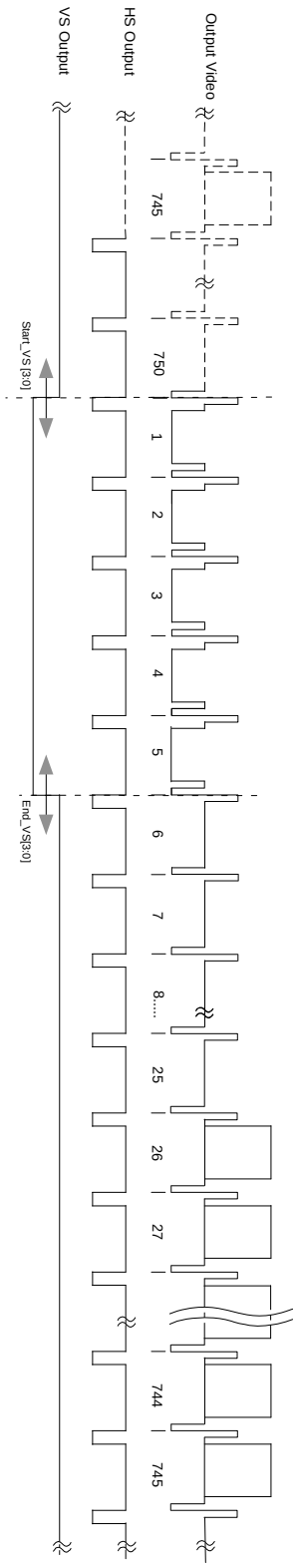


Figure 136: 720p VS Timing

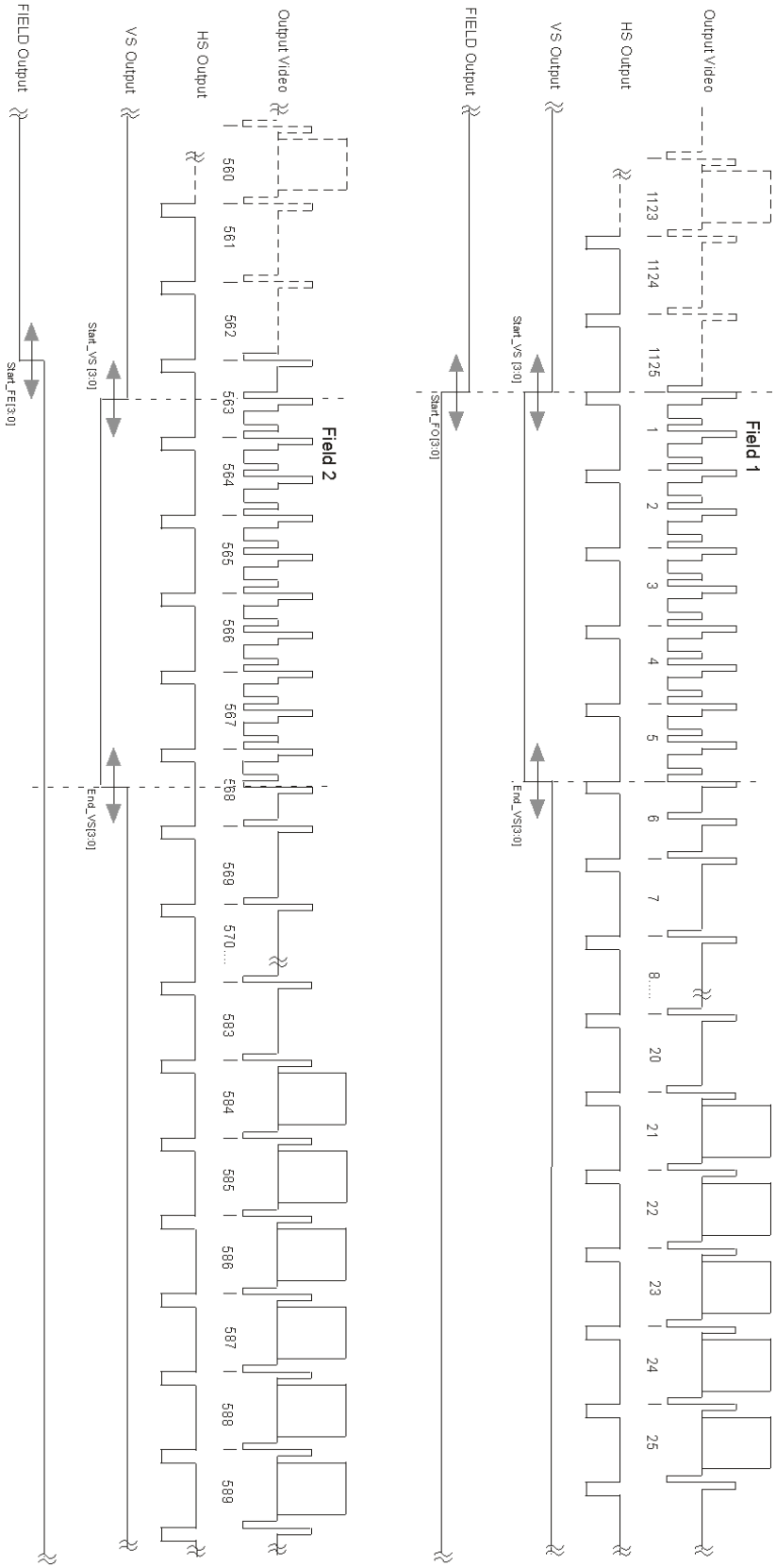


Figure 137: 1080i VS Timing

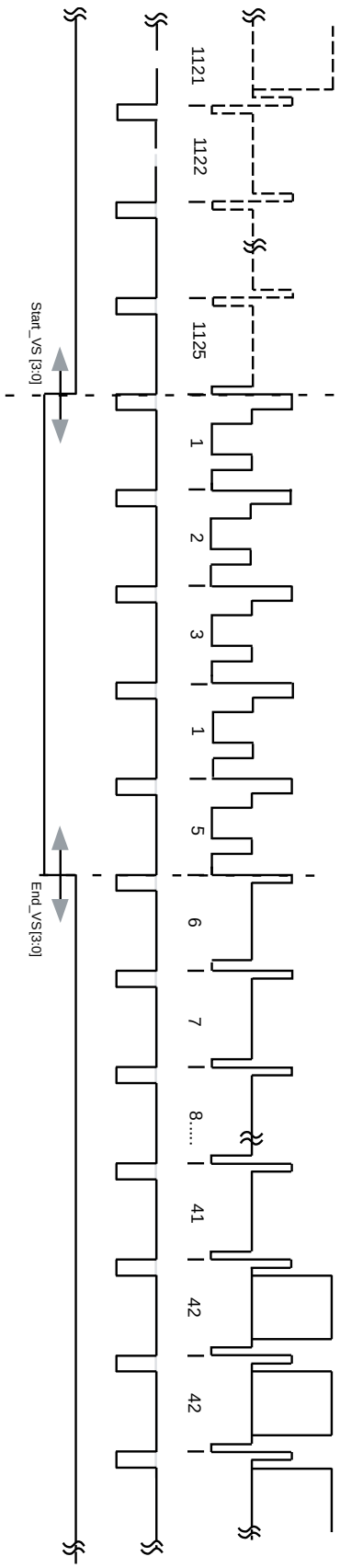


Figure 138: 1080p VS Timing

10.9.6 HCOUNT Timing Control

HCOUNT_ALIGN_ADJ[4:0], Addr 44 (CP), *Address 0xBE[1:0]; Address 0xBF[7:5]*

Manual adjustment for internally generated hcount offset . This register allows an adjustment of 15 pixels to the left or to the right. The MSB sets the direction (left or right) and the 4 LSBs set the number of pixels to move. This is an unsigned control.

Function

HCOUNT_ALIGN_ADJ[4:0]	Description
00000 «	Default value

10.9.7 Secondary Synchronization Signals

The secondary synchronization signals share their output pins with the primary ones, as shown in Table 58. The CS signal is a logic combination of HS and VS. Its polarity can be inverted using the INV_HS_POL bit.

The DE signal allows the ADV7842 to interface to ICs such as DVI or HDMI transmitters. The DE signal marks active video on all active lines and could, therefore, also be described as an inverted blanking signal. The polarity of the DE signal can be changed by the INV_F_POL bit.

Notes:

- The delay units are:
- LLC clock cycles for HS. With nominal sampling, this is equivalent to pixels.
- Video lines for VS and FIELD.

Synchronization information can also be passed on to downstream equipment by means of AV codes. There is an option in the AV code generation block that uses the position of the HS pin to trigger the insertion of SAV/EAV codes into the data stream.

10.9.8 Ancillary Synchronization Signal Outputs

The ADV7842 can provide ancillary synchronization information on the VS/FIELD and the SYNC_OUT output pins. The following sections describe the available signals.

10.9.8.1 Ancillary Synchronization Signals Output on VS/FIELD Pin

Figure 139 outlines the structure implemented in the ADV7842.

The DS_OUT bit enables selection between the regenerated line-locked VSync (synchronous to LLC) and a raw asynchronous version of the vertical synchronization. Depending on the application and the ultimate purpose of the timing signal, both of them can have distinct advantages:

- The **synchronous signals** aligned with the pixel data can be captured with the LLC clock. They accompany the data and determine the position of the vertical synchronization with pixel accuracy. As a prerequisite, the LLC clock must be locked and this requires PRIM_MODE[3:0], VID_STD[5:0], and other registers to be configured correctly.

- The **asynchronous signals** are not aligned with the video pixel data. However, they are valid even if the LLC is not locked to

input video. For a digital VS input signal, the data path to the VS output pin is combinatorial logic. For embedded synchronization, the vertical synchronization is extracted based on the external crystal clock. This makes both paths independent of the status of the LLC clock. These synchronization signals can be used in a system that chooses to implement auto detection of the input video standard downstream with the use of a microprocessor.

The SSPD decides between embedded synchronization and digital input. Refer to Section 10.8 for details on the SSPD.

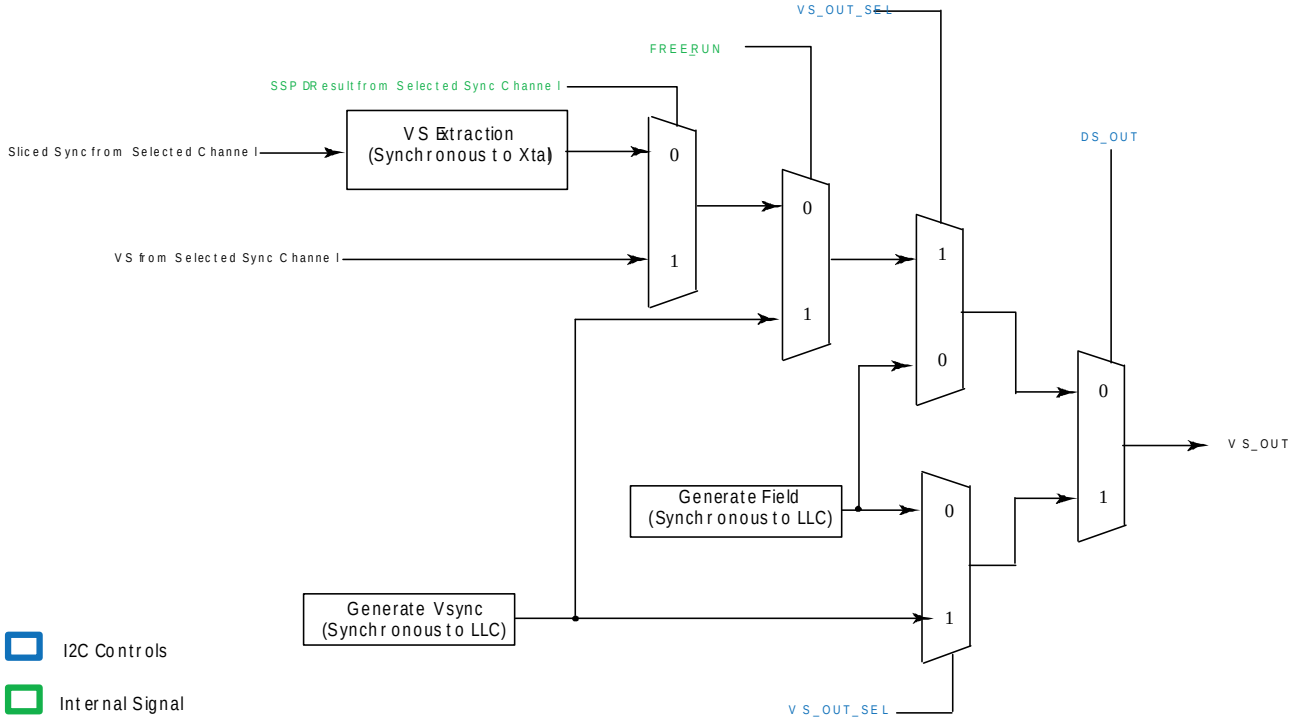


Figure 139: Ancillary Synchronization Information Output on VS/FIELD Pin

DS_OUT, Addr 44 (CP), *Address 0x85[0]*

Digital synchronization output enable.

Function

DS_OUT	Description
1 «	Output synchronous VSync
0	Asynchronous VSync

10.9.8.2 Ancillary Synchronization Signals Output on SYNC_OUT Pin

The ancillary synchronization information on the SYNC_OUT pin is shown in Figure 140.

The SEL_RAW_CS signal selects between the following options:

- When SEL_RAW_CS is set to 0, an HSync type input is passed through to the next stage.
  1. A raw HSync signal with no polarity detection. This raw HSync signal can be passed straight through to the SYNC_OUT pin by selecting SEL_RAW_CS = 0.
  2. A circuit that extracts HSync from an applied CSync signal or embedded synchronization signal. The applied CSync signal must have impairments such as Macrovision pseudo syncs removed.

- When SEL_RAW_CS is set to 1, a CSync signal can be passed through to the SYNC_OUT pin. This synchronization signal can be derived from two possible sources.

    1. The logic AND result of an applied HSync and VSync to give a CSync. Both HSync and VSync are polarity corrected in order to generate a proper CSync signal.
    2. A sliced embedded synchronization or CSync without polarity detection.

All input signals to this block are asynchronous in nature (not line locked) and do not follow fixed set up and hold time specifications with respect to the LLC signal. They are based on either combinatorial signal paths through the ADV7842 or use digital logic that is driven off the external crystal clock. This makes them independent of the lock state of the LLC.

SEL_RAW_CS, Addr 40 (IO), *Address 0x0B[6]*

A control to select the type of signal applied to SYNC_OUT.

Function

SEL_RAW_CS	Description
0	Raw HSync type signal through SYNC_OUT pad
1 «	Raw CSync type signal through SYNC_OUT pad

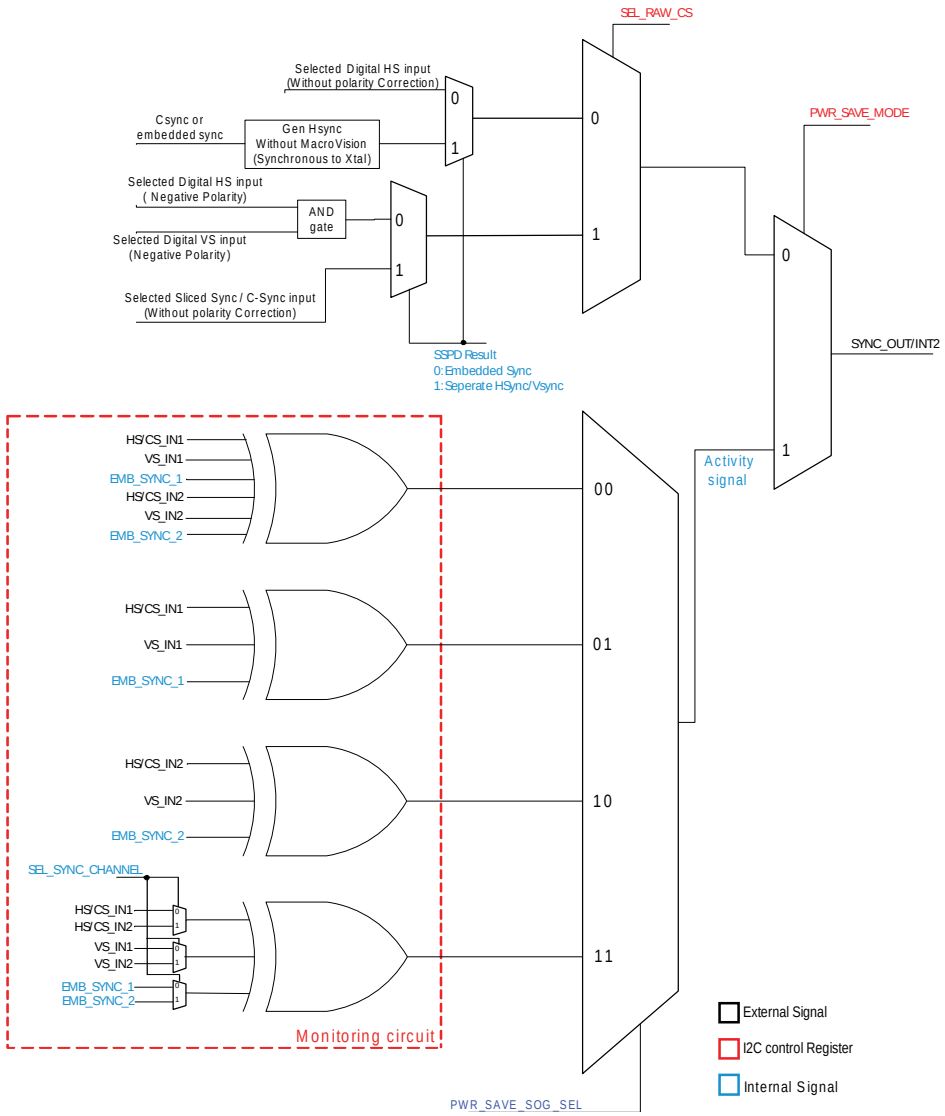


Figure 140: Ancillary Synchronization Information Output on SYNC_OUT Pin

10.10 CP DATA PROCESSING DELAY CONTROLS

The ADV7842 provides controls to delay data by 1 pixel after CP CSC.

DLY_A, Addr 44 (CP), *Address 0xBE[7]*

A control to delay the data on channel A by one pixel clock cycle.

Function

DLY_A	Description
1	Delay the data of channel A by 1 pixel clock cycle
0 «	Do not delay the data of channel A

DLY_B, Addr 44 (CP), *Address 0xBE[6]*

A control to delay the data on channel B by one pixel clock cycle.

Function

DLY_B	Description
1	Delay the data of channel B by 1 pixel clock cycle
0 «	Do not delay the data of channel B

DLY_C, Addr 44 (CP), *Address 0xBE[5]*

A control to delay the data on channel C by one pixel clock cycle.

Function

DLY_C	Description
1	Delay the data of channel C by 1 pixel clock cycle
0 «	Do not delay the data of channel C

10.11 CP HORIZONTAL LOCK STATUS

The ADV7842 provides an I²C readback value for the lock robustness. The measurement is based on an integration of the area of the horizontal synchronization that falls below the slicing threshold, as illustrated by Figure 141. The threshold level can be determined automatically or it can also be set by the customer via I²C.

The quality of horizontal locking depends on the strength, that is, depth, of the horizontal synchronization pulse. For shallow horizontal synchronization pulses, the area measured is low and the locking is not as reliable as for a strong, deep, horizontal synchronization.

The number presented as ISD[8:0] is not intended to be an absolute measurement, but a relative one. A large value indicates robust locking and a small value shows an unreliable lock state. A system controller reading the ISD value via the I²C interface must set appropriate thresholds for fully locked and partially locked.

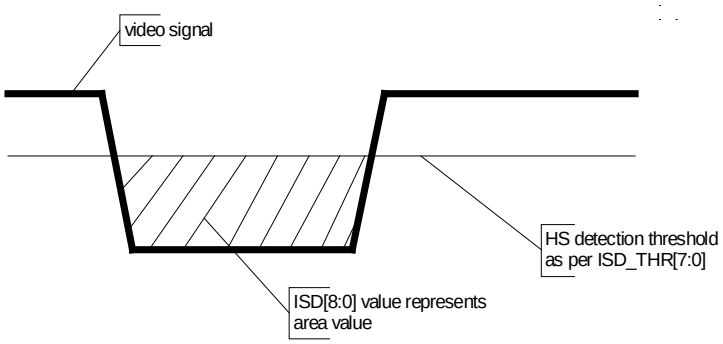


Figure 141: Synchronization Lock Robustness Measurement

The measurements are performed on a line-by-line basis on all video lines but not during the VBI. For video lines during the VBI, the result of the last active video line is kept.

The ISD[8:0] value changes dynamically on a line by line basis; the IFSD[8:0] is an averaged version of the ISD[8:0]. The averaging length can be set to 128 or 256 lines of video.

ISD_THR[7:0], Addr 44 (CP), *Address 0x83[7:0]*

A control used to set the threshold used for the ISD measurement. ISD_THR[7:0] stores a 12-bit unsigned value.

Function

ISD_THR[7:0]	Description
0x00 «	The threshold is calculated automatically and set to (level of HSync tip) + 0.5 * (HSync depth).
>0x01	The threshold is set to (ISD_THR[7:0] * 8)

IFSD_AVG, Addr 44 (CP), *Address 0x84[0]*

A control to set the averaging mode used to compute IFSD[8:0]

Function

IFSD_AVG	Description
0 «	ISD[8:0] is averaged over 128 lines of video to generate IFSD[8:0]
1	ISD[8:0] is averaged over 256 lines of video to generate IFSD[8:0]

ISD[8:0], Addr 44 (CP), *Address 0xE3[0]; Address 0xE4[7:0] (Read Only)*

A readback for representing the area of the of HSync that falls below the slicing threshold set by ISD_THR[7:0]. A high values indicates robust locking.

Function

ISD[8:0]	Description
xxxxxxxxx	Readback value

IFSD[8:0], Addr 44 (CP), *Address 0xE3[1]; Address 0xE5[7:0] (Read Only)*

A readback for the average value of the ISD measurement over 128 or 256 lines. The number of lines used to computes IFSD[8:0] is set in IFSD_AVG.

Function

IFSD[8:0]	Description
xxxxxxxxx	Readback value

10.12 NOISE AND CALIBRATION

The ADV7842 provides hardware for a noise and a calibration measurement. The two measurements share some hardware control (window). However, they are different in the way they examine the input data. The measurements are executed during a time window. The window can be positioned anywhere within a line of video and the length can be selected to be 16, 32, 64, or 128 LLC clock cycles.

Notes:

- Both measurements are performed on the channel A.
- Both measurements work on a video line basis and are performed during the active video.
- The tap-off point for both measurements is right after the gain multiplier. Clamping and AGC/manual gain will affect the numbers reported back.

10.12.1 Measurement Window

The window for the noise and calibration measurement is set via MEAS_WS[11:0] and MEAS_WL[1:0].

MEAS_WS[11:0], Addr 44 (CP), *Address 0x81[3:0]; Address 0x82[7:0]*

A control to set the start value of the measurement window use for noise and calibration. The unit for this parameter is a pixel clock cycle. Refer to NOISE[7:0] and CALIB[10:0]. A value of 0 positions the start of the window at the trailing edge of the incoming HSync.

Function

MEAS_WS[11:0]	Description
0x000	Start value (in LLC clock cycles) of measurement window. Value of 0 positions window at trailing edge of incoming HSync
0x004 «	Default value

MEAS_WL[1:0], Addr 44 (CP), *Address 0x81[7:6]*

A control to set the width of the window length used for noise calibration measurements. The unit for this parameter is a pixel clock cycle. Refer to Noise[7:0] and CALIB[10:0].

Function

MEAS_WL[1:0]	Description
00	Window length is 128 LLC clock cycles,
01	Window length is 64 LLC clock cycles,
10	Window length is 32 LLC clock cycles,
11 «	Window length is 16 LLC clock cycles,

10.12.2 Noise Measurement

For the noise or peak data measurement, the data during the window is monitored for the maximum and the minimum value. After the window is closed, the difference between the two is presented. If programmed during a quiet time of the input video, the value presented can be related back to the level of noise within the video signal. The noise level is presented as an unsigned number. Levels greater than 255 are saturated to 255.

NOISE[7:0], Addr 44 (CP), *Address 0xE2[7:0] (Read Only)*

A readback for the noise value measured on the Luma channel (i.e. channel A).
This register provides an unsigned value representing the difference between the maximum and minimum value measured during the window configured by MEAS_WS[11:0] and MEAS_WL[1:0].

Function

NOISE[7:0]	Description
xxxxxxxx	Readback value

10.12.3 Calibration Measurement

The input signal is accumulated during the measurement window. After the end of the window, the accumulated value is divided by the window length and the result (average signal level over the extent of the window) is presented via the I^2C register CALIB[10:0]. The number format is signed with a possible range of -1024 to +1024. It is envisaged to provide the ADV7842 with a flat gray field and to position the window in the middle of active video for a meaningful measurement.

CALIB[10:0], Addr 44 (CP), *Address 0xE3[4:2]; Address 0xE6[7:0] (Read Only)*

A readback for the calibration value measured on the Luma channel (i.e. channel A).
This register provides a signed value representing the average level over the extent of the window configured by MEAS_WS and MEAS_WL.

Function

CALIB[10:0]	Description
xxxxxxxxxx	Readback value

Note: The calibration measurement can be negative as the samples used for this measurement are taken before the offset block of the A channel. The data is signed before the offset block and unsigned after the offset block.

10.13 FREE RUN MODE

Free run mode provides the user with a stable clock and predictable data if the input signal cannot be decoded, for example, if input video is not present. It controls default color insertion and causes the ADV7842 to generate a default clock. The state in which this happens can be monitored via the CP_FREE_RUN status bit. (Refer to Section 10.14 for more information.) The free run feature is configured automatically for analog modes. The free run feature must be configured for HDMI modes.

10.13.1 Free Run Mode Thresholds

The free run threshold parameters define the horizontal and vertical conditions under which free run mode is entered. The horizontal and vertical parameters of the incoming video signal are measured and compared with internally stored parameters, and the magnitude of the difference decides whether to enter free run. The internally stored parameters are decoded by default from PRIM_MODE[3:0] and VID_STD[5:0]. For video standards other than the preprogrammed settings of PRIM_MODE[3:0] and VID_STD[5:0], the parameters can be set manually.

10.13.1.1 Horizontal Free Run Conditions

The horizontal conditions are based on the length of the incoming video line, which is measured based on the 28.6363 MHz crystal clock. This value is compared with the internally stored horizontal parameter, the ideal line length. The CH1_F_RUN_TH[2:0] and CH2_F_RUN_TH[2:0] control bits allow the user to select the threshold for channel 1 and channel 2 respectively. The ideal line length can be manually set via the Free-run Line Length controls, CH1_FR_LL[10:0] and CH2_FR_LL[10:0].

CH1_F_RUN_THR[2:0], Addr 44 (CP), *Address 0xF3[2:0]*

Free run threshold select for sync channel 1. Determines the horizontal conditions under which free run mode is entered or left. The length of the incoming video line is measured based on the crystal clock and compared to an internally stored parameter. The magnitude of the difference decides whether or not sync channel 1 will enter free run mode.

Function

CH1_F_RUN_THR[2:0]	Description
000	Minimum difference to switch into free run is 2. Maximum difference to switch out of free run is 1.
001	Minimum difference to switch into free run is 256. Maximum difference to switch out of free run is 200.
010	Minimum difference to switch into free run is 128. Maximum difference to switch out of free run is 112.
011	Minimum difference to switch into free run is 64. Maximum difference to switch out of free run is 48.
100 «	Minimum difference to switch into free run is 32. Maximum difference to switch out of free run is 24.
101	Minimum difference to switch into free run is 16. Maximum difference to switch out of free run is 12.
110	Minimum difference to switch into free run is 8. Maximum difference to switch out of free run is 6.
111	Minimum difference to switch into free run is 4. Maximum difference to switch out of free run is 3.

CH2_F_RUN_THR[2:0], Addr 44 (CP), *Address 0x43[2:0]*

Free run threshold select for sync channel 2. Determines the horizontal conditions under which free run mode is entered or left. The length of the incoming video line is measured based on the crystal clock and compared to an internally stored parameter. The magnitude of the difference decides whether or not sync channel 2 will enter free run mode.

Function

CH2_F_RUN_THR[2:0]	Description
000	Minimum difference to switch into free run is 2. Maximum difference to switch out of free run is 1.
001	Minimum difference to switch into free run is 256. Maximum difference to switch out of free run is 200.
010	Minimum difference to switch into free run is 128. Maximum difference to switch out of free run is 112.
011	Minimum difference to switch into free run is 64. Maximum difference to switch out of free run is 48.
100 «	Minimum difference to switch into free run is 32. Maximum difference to switch out of free run is 24.
101	Minimum difference to switch into free run is 16. Maximum difference to switch out of free run is 12.
110	Minimum difference to switch into free run is 8. Maximum difference to switch out of free run is 6.
111	Minimum difference to switch into free run is 4. Maximum difference to switch out of free run is 3.

CH1_FR_LL[10:0], Addr 44 (CP), *Address 0x8F[2:0]; Address 0x90[7:0]*

Free run line length in number of crystal clock cycles in one line of video for sync channel 1 STDI. This register should only be programmed video standards that are not supported by PRIM_MODE[3:0] and VID_STD[5:0].

Function

CH1_FR_LL[10:0]	Description
0x000 «	Internal free run line length is decoded from PRIM_MODE[3:0] and VID_STD[5:0].
All other values	Number of crystal clocks in the ideal line length. Used to enter or exit free run mode.

CH2_FR_LL[10:0], Addr 44 (CP), *Address 0x47[2:0]; Address 0x48[7:0]*

Free run line length in number of crystal clock cycles in one line of video for sync channel 2 STDI. This register should only be programmed for video standards that are not supported by PRIM_MODE[3:0] and VID_STD[5:0].

Function

CH2_FR_LL[10:0]	Description
0x000 «	Actually used internal free run line length is decoded from PRIM_MODE[3:0] and VID_STD[5:0].
All other values	Number of crystal clocks in the ideal line length. Used to enter or exit free run mode.

Notes:

- This parameter has **no** effect on the video decoding.
- If neither CH1_FR_LL[10:0] nor CH2_FR_LL[10:0] are programmed, then the Free-run Line Length parameter is decoded from PRIM_MODE[3:0] and VID_STD[5:0].
- If CH1_FR_LL[10:0] is programmed and CH2_FR_LL[10:0] is not, the Free-run Line Length parameter defined by CH1_FR_LL[10:0], is used for both channels.

10.13.2 Vertical Run Conditions

In the case of the vertical conditions, the number of lines per field of incoming video signal is measured. This value is compared with an internally stored vertical parameter, the ideal Field Length. The CH1_FL_FR_THRESHOLD[1:0] and CH2_FL_FR_THRESHOLD[1:0] control bits allow the user to select the threshold for channel 1 and channel 2 respectively. The ideal number of lines per field can be set manually via the CP_LCOUNT_MAX[11:0] register.

CH1_FL_FR_THRESHOLD[2:0], Addr 44 (CP), *Address 0xF3[5:3]*

Threshold for difference between input video field length and internally stored standard to enter and exit freerun.

Function

CH1_FL_FR_THRESHOLD[2:0]	Description
000	Minimum difference to switch into free run is 36 lines. Maximum difference to switch out of free run is 31 lines.
001	Minimum difference to switch into free run is 18 lines. Maximum difference to switch out of free run is 15 lines.
010 «	Minimum difference to switch into free run is 10 lines. Maximum difference to switch out of free run is 7 lines.
011	Minimum difference to switch into free run is 4 lines. Maximum difference to switch out of free run is 3 lines.
100	Minimum difference to switch into free run is 51 lines. Maximum difference to switch out of free run is 46 lines.
101	Minimum difference to switch into free run is 69 lines. Maximum difference to switch out of free run is 63 lines.
110	Minimum difference to switch into free run is 134 lines. Maximum difference to switch out of free run is 127 lines.
111	Minimum difference to switch into free run is 263 lines. Maximum difference to switch out of free run is 255 lines.

CH2_FL_FR_THRESHOLD[2:0], Addr 44 (CP), *Address 0x43[5:3]*

Threshold for difference between input video field length and internally stored standard to enter and exit freerun. This control is for the sync channel 2 STDI.

Function

CH2_FL_FR_THRESHOLD[2:0]	Description
000	Minimum difference to switch into free run is 36 lines. Maximum difference to switch out of free run is 31 lines.
001	Minimum difference to switch into free run is 18 lines. Maximum difference to switch out of free run is 15 lines.
010 «	Minimum difference to switch into free run is 10 lines. Maximum difference to switch out of free run is 7 lines.
011	Minimum difference to switch into free run is 4 lines. Maximum difference to switch out of free run is 3 lines.
100	Minimum difference to switch into free run is 51 lines. Maximum difference to switch out of free run is 46 lines.
101	Minimum difference to switch into free run is 69 lines. Maximum difference to switch out of free run is 63 lines.
110	Minimum difference to switch into free run is 134 lines. Maximum difference to switch out of free run is 127 lines.
111	Minimum difference to switch into free run is 263 lines. Maximum difference to switch out of free run is 255 lines.

CP_LCOUNT_MAX[11:0], Addr 44 (CP), *Address 0xAB[7:0]; Address 0xAC[7:4]*

Manual value for total number of lines in a frame expected by the CP core. CP_LCOUNT_MAX[11:0] is an unsigned value. This register is used for manual configuration of the free run feature. The value programmed in this register is used for sync channel 1. The value programmed in this register is used also for sync channel 2 if CH2_FR_FIELD_LENGTH[10:0] set to 0x000.

Function

CP_LCOUNT_MAX[11:0]	Description
0x000 «	Ideal number of lines per frame is decoded from PRIM_MODE[3:0] and VID_STD[5:0] for sync channel 1.
All other values	Use the programmed value as ideal number of lines per frame in free run decision for sync channel 1.

INTERLACED, Addr 44 (CP), *Address 0x91[6]*

Sets the interlaced/progressive mode of the incoming video processed in CP mode.

Function

INTERLACED	Description
0	The CP core expects video mode is progressive
1 «	the CP core expects video mode is interlaced

Field Line Count is the vertical parameter that holds the ideal number of lines per field for a given video standard. It affects the way CP handles the unlocked state. If CP_LCOUNT_MAX[11:0] and CH2_FR_FIELD_LENGTH[10:0] are set to 0, the internally used free run line length value is decoded from the current setting of PRIM_MODE[3:0] and VID_STD[5:0].

For standards not covered by the preprogrammed values, the CP_LCOUNT_MAX[11:0], CH2_FR_FIELD_LENGTH[10:0], and INTERLACED parameters must be set to the ideally expected number of lines per field.

Notes:

- The CP_LCOUNT_MAX[11:0] parameter has no effect on the video decoding.
- If neither CP_LCOUNT_MAX[11:0] nor CH2_FR_FIELD_LENGTH[10:0] are programmed, then the Free-run Line Length parameter is decoded from PRIM_MODE[3:0] and VID_STD[5:0].
- If CP_LCOUNT_MAX[11:0] is programmed and CH2_FR_FIELD_LENGTH[10:0] is not, the Free-run Line Length parameter

defined by CP_LCOUNT_MAX[11:0] and INTERLACED, is used for both channel 1 and channel 2.

- If CP_LCOUNT_MAX[11:0] is programmed and CH2_FR_FIELD_LENGTH[10:0] is also programmed, the Free-run Line Length parameter defined by CP_LCOUNT_MAX[11:0] and INTERLACED is used for channel 1. The Free-run Line Length parameter defined by CH2_FR_FIELD_LENGTH[10:0] is used for channel 2.

CH2_FR_FIELD_LENGTH[10:0], Addr 44 (CP), *Address 0x46[7:0]; Address 0x47[7:5]*

Ideal number of lines per field used by the CP core for the free run decision for sync channel 2. If set to 0 the ideal number of lines per field is dictated by CP_LCOUNT_MAX[11:0].

Function

CH2_FR_FIELD_LENGTH [10:0]	Description
0x000 «	Default value

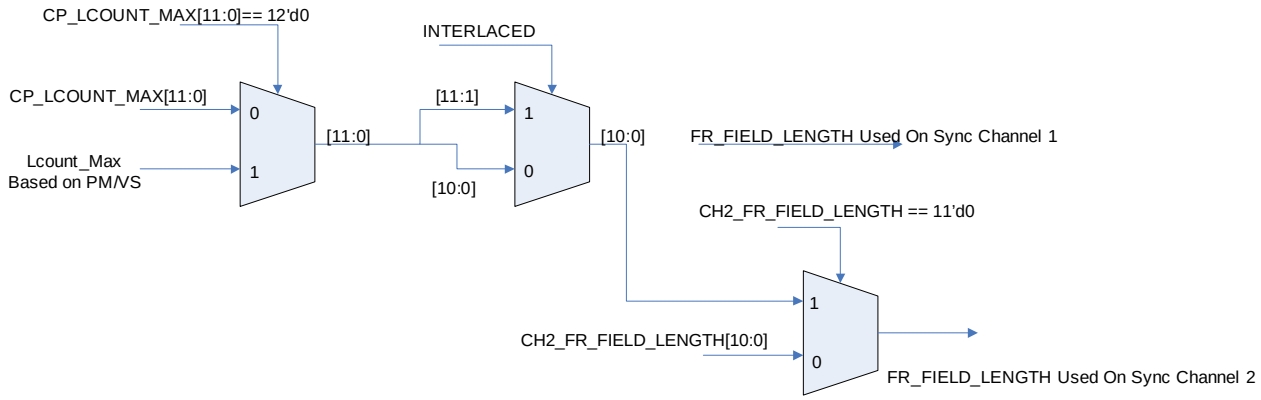


Figure 142: Free Run Field Length Selection for Channel 1 and Channel 2

10.13.3 Free Run Feature in HDMI Mode

This section describes how to configure the free run feature when the ADV7842 is in HDMI mode. The ADV7842 HDMI mode is defined in Section 4.

There are two free run modes in HDMI: free run mode 0 and free run mode 1. The HDMI_FRUN_MODE control selects which free run mode is enabled.

- HDMI free run mode 0:
  The decoder enters free run when the TMDS clock is not detected, for example, in a cable disconnect situation.

- HDMI free run mode 1:
  The decoder enters free run when the TMDS clock is not detected or when the detected input format does not match the format dictated by the PRIM_MODE[3:0] and VID_STD[5:0] settings.

For either free run mode to be implemented, HDMI free run operation must be enabled. This is done via the HDMI_FRUN_EN control.

HDMI_FRUN_EN, Addr 44 (CP), *Address 0xBA[0]*

A control to enable free run in HDMI mode.

Function

HDMI_FRUN_EN	Description
0	Disable the free run feature in HDMI mode
1 «	Enable the free run feature in HDMI mode

HDMI_FRUN_MODE, Addr 44 (CP), *Address 0xBA[1]*

A control to configure the free run feature in HDMI mode.

Function

HDMI_FRUN_MODE	Description
0 «	HDMI free run mode 0. The part free runs when the TMDS clock is not detected on the selected HDMI port
1	HDMI free run mode 1. The CP core free runs when the TMDS clock is not detected on the selected HDMI port or it the video resolution of HDMI stream processed by the part does not match the video resolution programmed in PRIM_MODE[3:0] and VID_STD[5:0].

HDMI_CP_AUTOPARM_LOCKED, Addr 44 (CP), *Address 0xE0[6] (Read Only)*

A readback to report the lock status of the parameter buffering in HDMI mode

Function

HDMI_CP_AUTOPARM_ LOCKED	Description
0 «	The parameter buffering block has not lock to the synchronization signal from the HDMI core.
1	The parameter buffering block has lock to the synchronization signal from the HDMI core.

HDMI_CP_LOCK_THRESHOLD[1:0], Addr 44 (CP), *Address 0xCB[1:0]*

Locking time of filter used for buffering of timing parameters in HDMI mode.

Function

HDMI_CP_LOCK_THRES HOLD[1:0]	Description
00 «	Slowest locking time
01	Medium locking time
10	Fastest locking time
11	Fixed step size of 0.5 pixel

HDMI_AUTOPARM_STS[1:0], Addr 44 (CP), *Address 0xE0[5:4] (Read Only)*

CP status for HDMI mode

Function

HDMI_AUTOPARM_STS[1:0]	Description
00 «	The CP is free running with according to timing parameters programmed in PRIM_MODE and VID_STD
01	The timing buffer filter has locked to the HDMI input
10	The CP is free running according to the HDMI buffered parameters
11	Reserved

DIS_AUTO_PARAM_BUFF, Addr 44 (CP), *Address 0xC9[0]*

A control to disable the buffering of the timing parameters used for free run in HDMI mode.

Function

DIS_AUTO_PARAM_BUFF	Description
0 «	Buffer the last measured parameters in HDMI mode used to determine video resolution the part free runs into.
1	Disable the buffering of measured parameters in HDMI mode. Free run standard determined by PRIM_MODE[3:0], VID_STD[5:0] and V_FREQ[2:0]

The video standard output in HDMI free run mode can be dictated either by the VID_STD[5:0] register setting or by the last valid measured parameters in HDMI mode. The DIS_AUTOPARAM_BUFF bit controls this free run option. If this bit is set to 1, PRIM_MODE[3:0], VID_STD[5:0], and V_FREQ[2:0] parameters must be set according to the required resolution for free run.

It is also possible to custom program the resolution that the ADV7842 should expect for free run mode 1 by programming the Free-run Line Length, Line Count Max, and Interlaced registers. Refer to Section 10.13 for the configuration of these registers.

Note: HDMI parameter based free run mode (i.e. DIS_AUTOPARAM_BUFFER = 0) does not support HDMI input with deep color.

10.13.4 Free Run Default Color Output

In the event of loss of input signal, the ADV7842 may enter free run and can be configured to output a color rather than noise. By default, the ADV7842 is configured to output a blue screen. The default color values are given in Table 65.

The times at which the default colors are inserted can be set as follows:

- Free run is forced: default colors are always output
- Automatic free run mode: default colors are output when the system detects a loss of video signal

Table 65: Default Color Output Values (CP)

Mode	CP_DEF_COL_MAN_VAL	Signal	Value
Default – GR	0	CH_A (G)	0
		CH_B (R)	0
		CH_C (B)	135_d
Default – COMP	0	CH_A (Y)	35_d
		CH_A (Pr)	114_d
		CH_A (Pb)	212_d
Man. Override	1	CH_A	4·DEF_COL_CHA[7:0]
		CH_B	4·DEF_COL_CHB[7:0]
		CH_C	4·DEF_COL_CHC[7:0]

CP_FORCE_FREERUN, Addr 44 (CP), *Address 0xBF[0]*

A control to force the CP to free run.

Function

CP_FORCE_FREERUN	Description
0 «	Do not force the CP core free run.
1	Force the CP core to free run.

CP_DEF_COL_AUTO, Addr 44 (CP), *Address 0xBF[1]*

A control to enable the insertion of default color when the CP free runs.

Function

CP_DEF_COL_AUTO	Description
0	Disable automatic insertion of default color
1 «	Output default colors when the CP free runs

CP_DEF_COL_MAN_VAL, Addr 44 (CP), *Address 0xBF[2]*

A control to enable manual selection of the color used when the CP core free runs.

Function

CP_DEF_COL_MAN_VAL	Description
0 «	Uses default color blue
1	Outputs default colors as given in CP_DEF_COL_CHA, CP_DEF_COL_B and CP_DEF_COL_C

Table 65 shows the default colors for component and graphics based video. The values describe the color blue. Setting the CP_DEF_COL_MAN_VAL bit high enables the user to overwrite the default colors with the values given in DEF_COL_CHA[7:0], DEF_COL_CHB[7:0], and DEF_COL_CHC[7:0].

The three parameters DEF_COL_CHA[7:0], DEF_COL_CHB[7:0], and DEF_COL_CHC[7:0] allow the user to specify their own default values.

Note: CP_DEF_COL_MAN_VAL must be set high for the three parameters to be used. See Table 65 for more information on the automatic values.

DEF_COL_CHA[7:0], Addr 44 (CP), *Address 0xC0[7:0]*

A control the set the default color for channel A. To be used if CP_DEF_COL_MAN_VAL is 1.

Function

DEF_COL_CHA[7:0]	Description
0x00 «	Default value

DEF_COL_CHB[7:0], Addr 44 (CP), *Address 0xC1[7:0]*

A control to set the default color for channel B. To be used if CP_DEF_COL_MAN_VAL is 1

Function

DEF_COL_CHB[7:0]	Description
0x00 «	Default value

DEF_COL_CHC[7:0], Addr 44 (CP), *Address 0xC2[7:0]*

A control to set the default color for channel C. To be used if CP_DEF_COL_MAN_VAL is 1

Function

DEF_COL_CHC[7:0]	Description
0x00 «	Default value

10.14 CP STATUS

CP_REG_FF is a status register that contains status bits for the CP core. Register CP_REG_FF consists of the following fields: MV_PS_DET, MV_AGC_DET, and CP_FREE_RUN.

CP_REG_FF Bit Number	Bit Name	Description
0	Reserved	
1	Reserved	
2	Reserved	
3	Reserved	
4	CP_FREE_RUN	CP is free running (no valid video signal found)
5	Reserved	
6	MV_AGC_DET	Detected Macrovision AGC pulses
7	MV_PS_DET	Detected Macrovision pseudo synchronization pulses

Note: For Bit 7 and Bit 6 to be meaningful, the Macrovision PS and AGC detection circuitry must be enabled (on by default).

CP_FREE_RUN, Addr 44 (CP), *Address 0xFF[4] (Read Only)*

Component processor freerun status

Function

CP_FREE_RUN	Description
0 «	The CP is not free running
1	The CP is free running

MV_AGC_DET, Addr 44 (CP), *Address 0xFF[6] (Read Only)*

Macrovision AGC pulses detection status

Function

MV_AGC_DET	Description
0 «	the CP has not detected Macrovision AGC pulses
1	the CP has detected Macrovision AGC Pulses

MV_PS_DET, Addr 44 (CP), *Address 0xFF[7] (Read Only)*

Macrovision pseudo pulses detection status.

Function

MV_PS_DET	Description
0 «	No Macrovision pseudo synchronization pulses detected.
1	Detected Macrovision Pseudo Synchronization Pulses

10.15 AUTO GRAPHICS MODE

Auto graphics mode is designed to allow the user to configure the ADV7842 to accept an input format not shown in Table 3 with the minimum amount of effort. Auto graphics mode is not limited only to graphics input, it can also be used to support component video input.

10.15.1 Primary Auto Graphics Controls

The user must provide the following key parameters to enable the ADV7842 to sample correctly the incoming video signal:

- PLL_DIV_MAN_EN
  This bit must be set to allow a user programmable PLL divide ratio to be used.

- PLL_DIV_RATIO[12:0]
  The PLL divide ratio is equal to the number of samples per line. The ADV7842 multiplies the incoming HSync frequency by the PLL divide ratio to generate the sampling clock.

- CH1_FR_LL[10:0]/CH2_FR_LL[10:0]
  CH1_FR_LL[10:0]/CH2_FR_LL[10:0] specifies the expected line length of the incoming video. If the actual line length is different from the expected line length by more than a programmable threshold, the decoder will free run.

- CP_LCOUNT_MAX[11:0]/CH2_FR_FIELD_LENGTH[10:0] / INTERLACED
  CP_LCOUNT_MAX[11:0] and CH2_FR_FIELD_LENGTH[10:0] specify the expected number of lines per *frame*. If the actual number of lines per frame is different from the expected number by more than a programmable threshold, the decoder will free run. INTERLACED should be set to 1 if the processed video is interlaced and set to 0 otherwise.

In auto graphics mode, it is assumed that embedded time codes are not required, and are disabled by default – output timing is provided on the HS and VS pins. Data blanking during the horizontal synchronization period and vertical synchronization period is also disabled. To enable embedded time codes and/or data blanking, GR_AV_BL_EN should be set to 1. With this bit set, individual control over time code insertion and data blanking is controlled by AVCODE_INSERT_EN and DATA_BLANK_EN

In the event that it is required to insert time codes and/or blank the data, the ADV7842 cannot determine the start and end of active video on each horizontal line, nor the start and end of the VBI region.

The following three options are available to handle time code insertion.

1. **AV_POS_SEL**, CP Map, *Address 0x7B, [2]* = 0
   EAV/SAV and data blanking are based on the HSync and VSync edges. Only data in the HSync and VSync areas will be blanked.

   AV_POS_SEL, CP Map, *Address 0x7B, [2]* = 1 AND the following user inputs in Section 10.15.2 are set to zero.

   An assumption is made that active video sits between 18.75% and 96.875% of the horizontal line (based on programmed PLL divide ratio), and that the VBI regions extends for IGNR_CLMP_VS_MAR_START[4:0] lines before the VSync and for IGNR_CLMP_VS_MAR_END[4:0] lines after the VSync.

2. **AV_POS_SEL,** CP Map, *Address 0x7B, [2]* = 1 AND the following user inputs in Section 10.15.2 are set to non zero values. The values set in the user inputs will be used to insert the EAV/SAV time codes and blank the data.

10.15.2 Secondary Auto Graphics Control

The following user inputs control time code insertion and/or data blanking.

CP_START_SAV[12:0], Addr 44 (CP), *Address 0x26[4:0]; Address 0x27[7:0]*

Manual value for Start of Active Video (SAV) position. Sets the total number of pixels between the start of non active video and the start of active video. Programming of this parameter is optional and should only be performed when the part is set in auto graphics mode. The value is unsigned.

Function

CP_START_SAV[12:0]	Description
0x0000 «	Default value

CP_START_EAV[12:0], Addr 44 (CP), *Address 0x28[4:0]; Address 0x29[7:0]*

Manual value for End of Active Video (EAV) position. Sets the total number of pixels between the end of non active video and the end of active video. Programming of this parameter is optional and should only be performed when the part is set in auto graphics mode. The value is unsigned.

Function

CP_START_EAV[12:0]	Description
0x0000 «	Default value

CP_START_VBI[11:0], Addr 44 (CP), *Address 0xA5[7:0]; Address 0xA6[7:4]*

Manual value for start of VBI region position (of odd fields in case of interlaced output). This is an unsigned value. It sets the total number of lines at the start of a frame of non-interlaced standard video. It sets the total number of lines at the start of the odd frame of interlaced standard video. Programming of this parameter is optional and should only be performed when the part is set in auto-graphics mode.

Function

CP_START_VBI[11:0]	Description
0x000 «	Default value

CP_END_VBI[11:0], Addr 44 (CP), *Address 0xA6[3:0]; Address 0xA7[7:0]*

Manual value for end of VBI region position (of odd fields in case of interlaced output). This is an unsigned value. It sets the total number of lines at the end of a frame of non-interlaced standard video. It sets the total number of lines at the end of the odd frame of interlaced standard video. Programming of this parameter is optional and should only be performed when the part is set in auto-graphics mode.

Function

CP_END_VBI[11:0]	Description
0x000 «	Default value

CP_START_VBI_EVEN[11:0], Addr 44 (CP), *Address 0xA8[7:0]; Address 0xA9[7:4]*

Manual value for start of VBI in even fields. This is an unsigned value. Total number of lines at the start of the even frame of interlaced standard. Programming of this parameter is optional and should only be performed when the part is set in auto-graphics mode.

Function

CP_START_VBI_EVEN[11:0]	Description
0x000 «	Default value

CP_END_VBI_EVEN[11:0], Addr 44 (CP), *Address 0xA9[3:0]; Address 0xAA[7:0]*

Manual value for end of VBI region position for even fields. This is an unsigned value. Total number of lines at the end of the even frame of interlaced standard. Programming of this parameter is optional and should only be performed when the part is set in auto graphics mode.

Function

CP_END_VBI_EVEN[11:0]	Description
0x000 «	Default value

Below controls are intended only for use with HDMI 3D standards.

CP_START_VBI_R[11:0], Addr 44 (CP), *Address 0x2A[7:0]; Address 0x2B[7:4]*

A control to manually set value for start position of VBI region. That is the extra blank region preceding the odd right (R) field in the 3D TV field alternative packing format supported by HDMI. It is not required to set this value. In normal operation this parameter is automatically calculated from the input.

Function

CP_START_VBI_R[11:0]	Description
0x000 «	Default value

CP_END_VBI_R[11:0], Addr 44 (CP), *Address 0x2B[3:0]; Address 0x2C[7:0]*

A control to manually set the value for end of VBI position. That is the extra blank region preceding the odd R field in 3D TV field alternative packing format supported by HDMI. It is not required to set this value. In normal operation this parameter is automatically calculated from the input.

Function

CP_END_VBI_R[11:0]	Description
0x000 «	Default value

CP_START_VBI_EVEN_R[11:0], Addr 44 (CP), *Address 0x2D[7:0]; Address 0x2E[7:4]*

A control to manually set the value for start position of the VBI region. That is the extra blank region preceding the even R field in 3D TV field alternative packing format supported by HDMI. It is not required to set this value. In normal operation this parameter is automatically calculated from the input.

Function

CP_START_VBI_EVEN_R[11:0]	Description
0x000 «	Default value

CP_END_VBI_EVEN[11:0], Addr 44 (CP), *Address 0xA9[3:0]; Address 0xAA[7:0]*

Manual value for end of VBI region position for even fields. This is an unsigned value. Total number of lines at the end of the even frame of interlaced standard. Programming of this parameter is optional and should only be performed when the part is set in auto graphics mode.

Function

CP_END_VBI_EVEN[11:0]	Description
0x000 «	Default value

It is also possible to adjust the position of the HSync and VSync signals. The following controls to adjust these are signed numbers to allow adjustment in either direction from the current position:

- CP_START_HS[12:0]
- CP_END_HS[12:0]
- CP_START_VS[5:0]
- CP_END_VS[5:0]
- CP_START_VS_EVEN[10:0]
- CP_END_VS_EVEN[10:0]
- CP_START_F_EVEN[10:0]
- CP_START_F_ODD[10:0]

CP_START_HS[12:0], Addr 44 (CP), *Address 0x22[4:0]; Address 0x23[7:0]*

A control to set the position of the start of the HSync output signal in the CP core in Autographic mode only. Programming of this parameter is optional and should only be performed when the part is set in auto graphics mode. The value is unsigned.

Function

CP_START_HS[12:0]	Description
0x0000 «	Default value

CP_END_HS[12:0], Addr 44 (CP), *Address 0x24[4:0]; Address 0x25[7:0]*

A control to set the position of the end of the HSync output signal in the CP core in Autographic mode only. Programming of this parameter is optional and should only be performed when the part is set in auto graphics mode. The value is unsigned.

Function

CP_END_HS[12:0]	Description
0x0000 «	Default value

CP_START_VS[5:0], Addr 44 (CP), *Address 0x9A[4:0]; Address 0x9B[7]*

A control to set the position of the start of the VSync output signal in the CP core in Autographic mode only. In the case of an interlaced signal this register adjusts the odd VS signal. Programming of this parameter is optional and should only be performed when the part is set in auto graphics mode. The value is unsigned.

Function

CP_START_VS[5:0]	Description
000000 «	Default value

CP_END_VS[5:0], Addr 44 (CP), *Address 0x9B[6:1]*

A control to set the position of the end of the VSync output signal in the CP core in Autographic mode only. In the case of an interlaced signal this register adjusts the odd VS signal. Programming of this parameter is optional and should only be performed when the part is set in auto graphics mode. The value is unsigned.

Function

CP_END_VS[5:0]	Description
000000 «	Default value

CP_START_VS_EVEN[10:0], Addr 44 (CP), *Address 0x9C[7:0]; Address 0x9D[7:5]*

A control to set the position of the start of the even VSync output signal in the CP core in Autographic mode only. Programming of this parameter is optional and should only be performed when the part is set in auto graphics mode. The value is unsigned.

Function

CP_START_VS_EVEN[10:0]	Description
0x000 «	Default value

CP_END_VS_EVEN[10:0], Addr 44 (CP), *Address 0x9D[4:0]; Address 0x9E[7:2]*

A control to set the position of the end of the even VSync output signal in the CP core in Autographic mode only. Programming of this parameter is optional and should only be performed when the part is set in auto graphics mode. The value is unsigned.

Function

CP_END_VS_EVEN[10:0]	Description
0x000 «	Default value

CP_START_F_ODD[10:0], Addr 44 (CP), *Address 0x9F[7:0]; Address 0xA0[7:5]*

A control to set the position of the end of the odd field output signal in the CP core in Autographic mode only. Programming of this parameter is optional and should only be performed when the part is set in auto graphics mode. The value is unsigned.

Function

CP_START_F_ODD[10:0]	Description
0x000 «	Default value

CP_START_F_EVEN[10:0], Addr 44 (CP), *Address 0xA0[4:0]; Address 0xA1[7:2]*

A control to set the position of the end of the even field output signal in the CP core in Autographic mode only. Programming of this parameter is optional and should only be performed when the part is set in auto graphics mode. The value is unsigned.

Function

CP_START_F_EVEN[10:0]	Description
0x000 «	Default value

10.15.3 Auxiliary Auto Graphics Controls

This section describes the auxiliary control available for auto graphic controls. It is recommended to leave these controls to default.

IGNR_CLMP_VS_MAR_START[4:0], Addr 44 (CP), *Address 0x8A[0]; Address 0x8B[7:4]*

A control to set the start of the window during which the clamp is ignored.
This register stores the unsigned number of pixel clocks between the start position of the window relative to the leading edge of the VSync. This control should only be used VID_STD[5:0] is set for auto-graphics mode.

Function

IGNR_CLMP_VS_MAR_START[4:0]	Description
0x04 «	Default value

IGNR_CLMP_VS_MAR_END[4:0], Addr 44 (CP), *Address 0x8A[7:3]*

A control to set the end of the window during which the clamp is ignored.
This register stores the unsigned number of pixel clocks between the end position of the window relative to the trailing edge of the VSync. This control should only be used VID_STD[5:0] is set for auto-graphics mode.

Function

IGNR_CLMP_VS_MAR_END[4:0]	Description
0x04 «	Default value

AUTO_SL_FILTER_FREEZE_EN, Addr 44 (CP), *Address 0xCB[5]*

This bit determines if the internally generated parameter for the position of the HSync trailing edge is updated during the VBI region. This control is only intended for auto-graphics mode. It is recommended to leave AUTO_SL_FILTER_FREEZE_EN to default. Unless AUTO_SL_FILTER_FREEZE_EN is left to default, the part may generate an incorrect HSync trailing edge position parameter if the input synchronization is embedded and has serration pulses

Function

AUTO_SL_FILTER_FREEZE_EN	Description
0	Do not freeze the trailing edge position of the HSync during the VBI region.
1 «	Freeze the trailing edge position of the HSync during the VBI region.

10.15.4 Setting Examples for 1280x720p

The section provides settings examples for a video input with resolution 1280x720p.

The PLL divide ratio should be set to 1650 decimal.

If CP_START_SAV = 0x000, CP_START_EAV = 0x5DC, CP_START_VBI = 0x2E9, and CP_END_VBI = 0x00, then:

- SAV will be at pixel 309 (18.75% of 1650 – the PLL divide ratio value)
- EAV will be at pixel 1500 because CP_START_EAV is set to 0x5DC (= 1500 decimal)
- VBI will start on line 745 because CP_START_VBI is set to 0x2E9 (= 745 decimal)
- VBI will end on line 4 because CP_END_VBI is set to 0x00 so the automatic value is used

11 VBI DATA PROCESSOR

The VBI Data Processor (VDP) is capable of processing multiple VBI data standards on analog video.

For low data rate VBI standards like Closed Captioning (CC), Wide Screen Signaling (WSS), or Copy Generation Management System (CGMS), the user can read the decoded data bytes from dedicated I^2C registers for different standards. I^2C readback is also supported for some high data rate standards like PDC, UTC, VPS or Gemstar.

The decoded results can also be made available as ancillary data in the output 656 data stream or through a dedicated fast I^2C port.

The following VBI data standards can be decoded by the VDP block:

PAL

- Teletext system A or C or D — ITU-BT-653
- Teletext system B or WST — ITU-BT-653
- VPS (Video Programming System) — ETSI EN 300 231 V 1.3.1
- VITC (Vertical Interval Time Codes)
- WSS (Wide Screen Signaling) — BT.1119-1 / ETSI.EN.300294
- CCAP (Closed Captioning)

NTSC

- Teletext system B and D — ITU-BT-653
- Teletext system C or NABTS — ITU-BT-653 / EIA-516
- VITC (Vertical Interval Time Codes)
- CGMS (Copy Generation Management System) — EIA-J CPR-1204 / IEC 61880
- Gemstar
- CCAP (Closed Captioning) — EIA-608

525p and 625p

- CGMS
- CGMS TYPE B — (CEA-805-A)

720p

- CGMS
- CGMS TYPE B — (CEA-805-A)

1080i

- CGMS
- CGMS TYPE B — (CEA-805-A)
- VITC

11.1 VDP CONFIGURATION

The VBI data standard that the VDP decodes on a particular line of incoming video has been set by default, as described in Section 11.1.1. This can be over-ridden manually and any VBI data can be decoded on any line. The details of manual programming are described in

Section 11.1.2.

11.1.1 VDP Default Configuration

The VDP can decode different VBI data standards on a line to line basis. List of the standards that VDP can support is shown in Table 66. VBI_DATA_STD[3:0] is a list of VBI standards represented by 4-bit code used in this document. The standard represented by VBI_DATA_STD[3:0] varies, depending on the video standard being processed.

By default, the VDP block is configured to decode certain standards from certain lines, depending on the video standard being processed by the core. The various standards supported by default on different lines of VBI are described in Table 67.

Table 66: VBI_DATA_STD[3:0] Values Corresponding to a Particular VBI Standard

VBI_DATA_STD[3:0]		625/50 – PAL	525/60 – NTSC	525p	625p	720p	1080i
Binary	Dec	(Interlaced)	(Interlaced)				
0001	1	Teletext system[3]	Teletext system[1]	Reserved	Reserved	Reserved	Reserved
0010	2	VPS[4]	Reserved	Reserved	Reserved	Reserved	Reserved
0011	3	VITC	VITC	Reserved	Reserved	Reserved	VITC
0100	4	WSS[5]	CGMS[6]	CGMS[4]	CGMS[4]	CGMS[4]	CGMS[4]
0101	5	Reserved	GEMSTAR_1X	Reserved	Reserved	Reserved	Reserved
0110	6	Reserved	GEMSTAR_2X	Reserved	Reserved	Reserved	Reserved
0111	7	CCAP[5]	CCAP [7]	Reserved	Reserved	Reserved	Reserved
1000	8	Reserved	Reserved	CGMS B[8]	Reserved	CGMS B[6]	CGMS B[6]
1001	9	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
1010	10	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
1011	11	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
1100	12	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
1101	13	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
1110	14	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
1111	15	Disable VDP	-	-	-	-	-

Refer to Table 66 to interpret a standard represented by VBI_DATA_STD (as decoded VBI standard depends on video standard 525i/625i/525p etc.)

Table 67: Default Standards on Lines for Supported Interlaced and Progressive Standards

525i	625i	525p	625p	720p	1080i

[3] Teletext system identified by VDP_TTXT_TYPE

[4] VPS – ETSI EN 300 231 V 1.3.1

[5] WSS BT.1119-1/ ETSI.EN.300294

[6] CGMS EIA-J CPR-1204 / IEC 61880

[7] CCAP EIA-608

[8] CGMS TYPE B (CEA-805-A)

Line No.	Default VBI_DATA_STD [3:0]	Line No.	Default VBI_DATA_STD [3:0]	Line No.	Default VBI_DATA_STD [3:0]	Line No.	Default VBI_DATA_STD [3:0]	Line No.	Default VBI_DATA_STD [3:0]	Line No.	Default VBI_DATA_STD [3:0]
		6	1	6	-	6	-	6	-	6	-
		7	1	7	-	7	-	7	-	7	-
		8	1	8	-	8	-	8	-	8	-
		9	1	9	-	9	-	9	-	9	3
10	1	10	1	10	-	10	-	10	-	10	-
11	1	11	1	11	-	11	-	11	-	11	-
12	1	12	1	12	-	12	-	12	-	12	-
13	1	13	1	13	-	13	-	13	-	13	-
14	3	14	1	14	-	14	-	14	-	14	-
15	1	15	1	15	-	15	-	15	-	15	-
16	1	16	2	16	-	16	-	16	-	16	-
17	1	17	1	17	-	17	-	17	-	17	-
18	1	18	1	18	-	18	-	18	-	18	-
19	1	19	3	19	-	19	-	19	-	19	4
20	4	20	1	20	-	20	-	20	-	20	-
21	7	21	1	21	-	21	-	21	-	21	-
22	1	22	7	22	-	22	-	22	-		
23	6	23	4	23	-	23	-	23	-		
24	6	24	0	24	-	24	-	24	4		
25	6	-	-	25	-	25	-	25	-		
272	1	318	2	26	-	26	-	26	-	569	-
273	1	319	1	27	-	27	-	27	-	570	-
274	1	320	1	28	-	28	-	28	-	571	3
275	1	321	1	29	-	29	-	29	-	572	-
276	1	322	1	30	-	30	-	30	-	573	-
277	3	323	1	31	-	31	-	31	-	574	-
278	1	324	1	32	-	32	-	32	-	575	-
279	1	325	1	33	-	33	-	33	-	576	-
280	1	326	1	34	-	34	-	34	-	577	-
281	1	327	1	35	-	35	-	35	-	578	-
282	1	328	1	36	-	36	-	36	-	579	-
283	4	329	2	37	-	37	-	37	-	580	-
284	7	330	1	38	-	38	-	38	-	581	-
285	1	331	1	39	-	39	-	39	-	582	4
286	6	332	3	40	-	40	-	40	-	583	-
287	6	333	1	41	4	41	-	41	-	584	-
288	6	334	1	42	-	42	-	42	-		
		335	7	43	-	43	4	43	-		
		336	1	44	-	44	-	44	-		
		337	0	45	-	45	-	45	-		

11.1.2 VDP Manual Configuration

Table 67 shows default standards decoded by the VDP. However decoded standard(s) on specific line(s) can be manually reconfigured - by using Manual Line Programming Registers (refer to Table 68). To select standard decoded on a particular line – write desired VBI standard code to a register corresponding to that specific line (VDP_MAN_LINE_x_y). VBI standard codes (VBI_DATA_STD) are 4-bit codes that are defined in Table 66.

For example:

To decode a VITC on line number 15 in 525i:

Write 0b0011 (VITC – see Table 66) to a VDP_MAN_LINE_10_30[7:4]

A zero value in the Manual Line Programming register means that the default VBI standard for that line will be decoded (as in Table 67).

Table 68: Details of Manual Line Programming Registers

Register Bit Names	Register Location	Line Numbers in which VBI Data is Inserted			
		525i	625i	1080i	525p, 625p, 720p
VDP_MAN_LINE_1_21[7:4]	0x64		6	6	6
VDP_MAN_LINE_2_22[7:4]	0x65		7	7	7
VDP_MAN_LINE_3_23[7:4]	0x66		8	8	8
VDP_MAN_LINE_4_24[7:4]	0x67		9	9	9
VDP_MAN_LINE_5_25[7:4]	0x68	10	10	10	10
VDP_MAN_LINE_6_26[7:4]	0x69	11	11	11	11
VDP_MAN_LINE_7_27[7:4]	0x6A	12	12	12	12
VDP_MAN_LINE_8_28[7:4]	0x6B	13	13	13	13
VDP_MAN_LINE_9_29[7:4]	0x6C	14	14	14	14
VDP_MAN_LINE_10_30[7:4]	0x6D	15	15	15	15
VDP_MAN_LINE_11_31[7:4]	0x6E	16	16	16	16
VDP_MAN_LINE_12_32[7:4]	0x6F	17	17	17	17
VDP_MAN_LINE_13_33[7:4]	0x70	18	18	18	18
VDP_MAN_LINE_14_34[7:4]	0x71	19	19	19	19
VDP_MAN_LINE_15_35[7:4]	0x72	20	20	20	20
VDP_MAN_LINE_16_36[7:4]	0x73	21	21	21 + Full Field	21
VDP_MAN_LINE_17_37[7:4]	0x74	22	22		22
VDP_MAN_LINE_18_38[7:4]	0x75	23	23		23
VDP_MAN_LINE_19_39[7:4]	0x76	24	24		24
VDP_MAN_LINE_20_40[7:4]	0x77	25 + Full Field	25 + Full Field		25
VDP_MAN_LINE_1_21[3:0]	0x64	272	318	569	26
VDP_MAN_LINE_2_22[3:0]	0x65	273	319	570	27
VDP_MAN_LINE_3_23[3:0]	0x66	274	320	571	28
VDP_MAN_LINE_4_24[3:0]	0x67	275	321	572	29
VDP_MAN_LINE_5_25[3:0]	0x68	276	322	573	30
VDP_MAN_LINE_6_26[3:0]	0x69	277	323	574	31
VDP_MAN_LINE_7_27[3:0]	0x6A	278	324	575	32
VDP_MAN_LINE_8_28[3:0]	0x6B	279	325	576	33
VDP_MAN_LINE_9_29[3:0]	0x6C	280	326	577	34
VDP_MAN_LINE_10_30[3:0]	0x6D	281	327	578	35
VDP_MAN_LINE_11_31[3:0]	0x6E	282	328	579	36
VDP_MAN_LINE_12_32[3:0]	0x6F	283	329	580	37
VDP_MAN_LINE_13_33[3:0]	0x70	284	330	581	38
VDP_MAN_LINE_14_34[3:0]	0x71	285	331	582	39
VDP_MAN_LINE_15_35[3:0]	0x72	286	332	583	40
VDP_MAN_LINE_16_36[3:0]	0x73	287	333	584 + Full Field	41
VDP_MAN_LINE_17_37[3:0]	0x74	288 + Full field	334		42
VDP_MAN_LINE_18_38[3:0]	0x75		335		43
VDP_MAN_LINE_19_39[3:0]	0x76		336		44
VDP_MAN_LINE_20_40[3:0]	0x77		337 + Full Field		45 + Full Frame

Note:

Full field/full frame detection (detection on lines other than VBI lines) of any standard can also be enabled. That detection can be enabled by writing code of desired standard (VBI_DATA_STD) into the respective registers. Table 68 and Table 69 indicate the manual programming registers that provide the full field/frame option. The VBI programmed standard assigned in those registers will be enabled for all the active lines (full field/frame).

Table 69: Details of Full Field/Frame Programming Registers

Video Standard	Register Bits to be Programmed for Odd Field (Frame for Progressive Input)	Register Bits to be Programmed for Even Field
525i	VDP_MAN_LINE_20_40[7:4]	VDP_MAN_LINE_17_37[7:4]
625i	VDP_MAN_LINE_20_40[7:4]	VDP_MAN_LINE_20_40[3:0]
1080i	VDP_MAN_LINE_16_36[7:4]	VDP_MAN_LINE_16_36[3:0]
525p/625p/720p	VDP_MAN_LINE_20_40[3:0]	n/a

11.2 TELETEXT SYSTEM IDENTIFICATION

VDP assumes that if teletext is present in a video channel, all the teletext lines will comply with a single standard system. By default, teletext B is decoded for PAL standards and teletext C is decoded for NTSC standards. To change default settings following controls should be used:

- VDP_TTXT_TYPE_MAN_EN
- VDP_TTXT_TYPE[1:0]

VDP_TTXT_TYPE_MAN_EN, Addr 48 (VDP), *Address 0x60[2]*

Enable for manual programming of Teletext decoding

Function

VDP_TTXT_TYPE_MAN_EN	Description
0 «	Manual programming of Teletext disabled
1	Manual programming of Teletext enabled

VDP_TTXT_TYPE[1:0], Addr 48 (VDP), *Address 0x60[1:0]*

Teletext Type Detected. These bits are functional only if VDP_TTXT_TYPE_MAN_EN is set to 1.

Function

VDP_TTXT_TYPE[1:0]	Description
00 «	ITU_BT.653-625/50-A - for PAL
01	ITU_BT.653-625/50-B(WST) - for PAL; ITU_BT.653-525/60-B - for NTSC
10	ITU_BT.653-625/50-C(WST) - for PAL; ITU_BT.653-525/60-C or EIA516(NABTS) - for NTSC
11	ITU_BT.653-625/50-D - for PAL; ITU_BT.653-525/60-D - for NTSC

11.3 VDP DECODED DATA READBACK REGISTERS

11.3.1 Teletext Readback Registers

Since teletext is a high data rate standard, decoded bytes are provided via ancillary data. A status bit is provided to indicate the teletext detection status.

VDP_STATUS_TTXT, Addr 48 (VDP), *Address 0x40[7] (Read Only)*

Teletext Detection Status Bit

Function

VDP_STATUS_TTXT	Description
0 «	Teletext not detected
1	Teletext detected

11.3.2 CGMS and WSS Readback Registers

CGMS and WSS convey the same type of information for different video standards. WSS is a 625i standard while CGMS is a 525i, 525p, 625p, 720p, and 1080i standard. Hence, the CGMS and WSS readback registers are shared. WSS is biphase coded and the VDP does a biphase decoding to produce the 14 raw WSS bits to be available in the CGMS and WSS VDP readback registers. A status bit is set when this data is available.

STATUS_CLEAR_WSS_CGMS, Addr 48 (VDP), *Address 0x78[2] (Self-Clearing)*

WSS or CGMS data status clear. Refreshes the WSS and CGMS readback registers

Function

STATUS_CLEAR_WSS_CGMS	Description
0 «	Do not refresh the WSS and CGMS readback registers
1	Refresh the WSS and CGMS readback registers

VDP_STATUS_WSS_CGMS, Addr 48 (VDP), *Address 0x40[2] (Read Only)*

WSS or CGMS Type A Data Detection Status Bit

Function

VDP_STATUS_WSS_CGMS	Description
0 «	WSS or CGMS Type A data not detected.
1	WSS or CGMS Type A data detected.

VDP_CGMS_WSS_DATA[23:0], Addr 48 (VDP), *Address 0x43[7:0]; Address 0x44[7:0]; Address 0x45[7:0] (Read Only)*

Decoded data for CGMS Type A and WSS

Function

VDP_CGMS_WSS_DATA[23:0]	Description
VDP_CGMS_WSS_DATA[23:0]	Decoded CGMS[23:0] data. VDP_CGMS_WSS_DATA[13:0] = Decoded WSS[13:0] data.

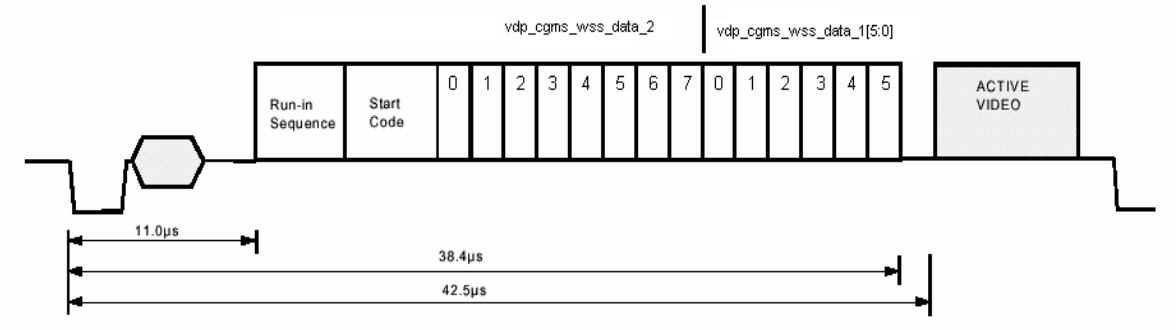


Figure 143: WSS (625i) Waveform

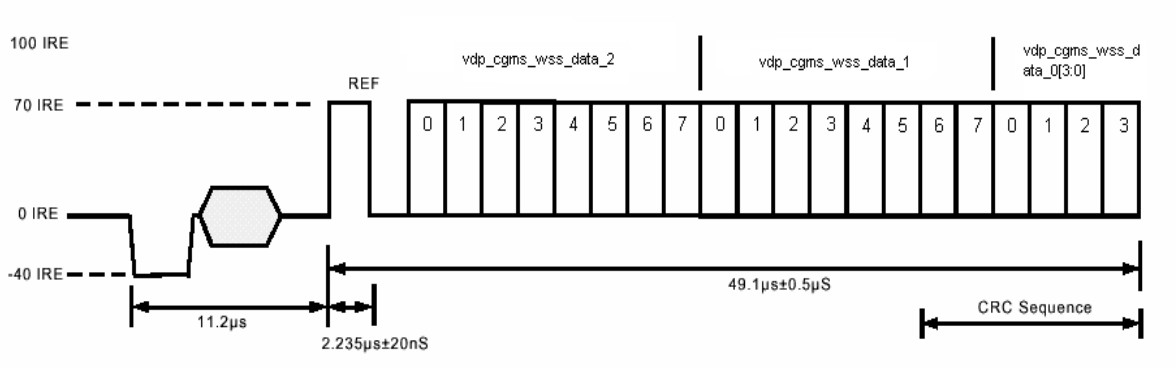


Figure 144: CGMS (525i) Waveform

11.3.3 Closed Captioning Readback Registers

Two bytes of decoded closed caption (CCAP) data are made available in the VDP readback registers. The VDP_STATUS_CCAP_EVEN_FIELD identifies the field from which the CCAP data was decoded.

STATUS_CLEAR_CCAP, Addr 48 (VDP), *Address 0x78[0] (Self-Clearing)*

Closed Caption data status clear. Refreshes the CCAP status register

Function

STATUS_CLEAR_CCAP	Description
0 «	Do not refresh the CCAP status registers
1	Refresh the CCAP status registers

VDP_STATUS_CCAP, Addr 48 (VDP), *Address 0x40[0] (Read Only)*

Closed Caption Data Detection Status Bit

Function

VDP_STATUS_CCAP	Description
0 «	Closed Caption data not detected
1	Closed Caption data detected

VDP_STATUS_CCAP_EVEN_FIELD, Addr 48 (VDP), *Address 0x40[1] (Read Only)*

Closed Caption data in even field Status Bit

Function

VDP_STATUS_CCAP_EVEN_FIELD	Description
0 «	Closed Caption data not detected in the even field.
1	Closed Caption data detected in the even field.

VDP_CCAP_DATA[15:8], Addr 48 (VDP), *Address 0x42[7:0] (Read Only)*

Byte 2 of Decoded Closed Caption data

Function

VDP_CCAP_DATA[15:8]	***Description***
xxxxxxxx	*Byte 2 of Decoded Closed Caption data.*

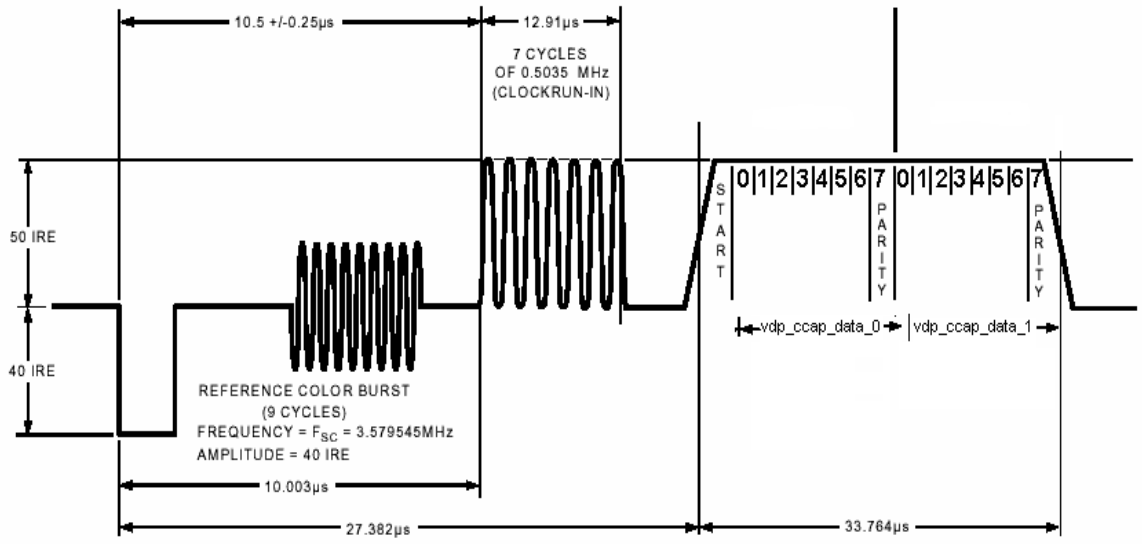


Figure 145: CCAP Waveform and Decoded Data Correlation

11.3.4 VITC Readback Registers

The decoded VITC data bytes are available in the VDP_VITC_DATA registers. Refer to Table 70. The VDP also calculates a CRC for the decoded data bytes and is made available in the VDP_VITC_CALC_CRC register.

VDP_VITC_CALC_CRC[7:0], Addr 48 (VDP), *Address 0x5E[7:0] (Read Only)*

Calculated CRC value for decoded VITC data.

Function

VDP_VITC_CALC_CRC[7:0]	Description
xxxxxxxx	Readback value

STATUS_CLEAR_VITC, Addr 48 (VDP), *Address 0x78[6] (Self-Clearing)*

VITC data status clear. Refreshes the VITC status register

Function

STATUS_CLEAR_VITC	Description
0 «	Do not refresh the VITC status registers
1	Refresh the VITC status registers

VDP_STATUS_VITC, Addr 48 (VDP), *Address 0x40[6] (Read Only)*

VITC Detection Status Bit

Function

VDP_STATUS_VITC	Description
0 «	VITC data not detected
1	VITC data detected

VDP_VITC_DATA[71:0], Addr 48 (VDP), *Address 0x5D[7:0]; Address 0x5C[7:0]; Address 0x5B[7:0]; Address 0x5A[7:0]; Address 0x59[7:0]; Address 0x58[7:0]; Address 0x57[7:0]; Address 0x56[7:0]; Address 0x55[7:0] (Read Only)*

Decoded VITC data readback

Table 70: VITC Readback Registers

VITC DATA BITS	Address in VDP map
VITC bits [9:2]	0x55[7:0]
VITC bits [19:12]	0x56[7:0]
VITC bits [29:22]	0x57[7:0]
VITC bits [39:32]	0x58[7:0]
VITC bits [49:42]	0x59[7:0]
VITC bits [59:52]	0x5A[7:0]
VITC bits [69:62]	0x5B[7:0]
VITC bits [79:72]	0x5C[7:0]
VITC bits [89:82]	0x5D[7:0]
VITC CRC	0x5E[7:0]

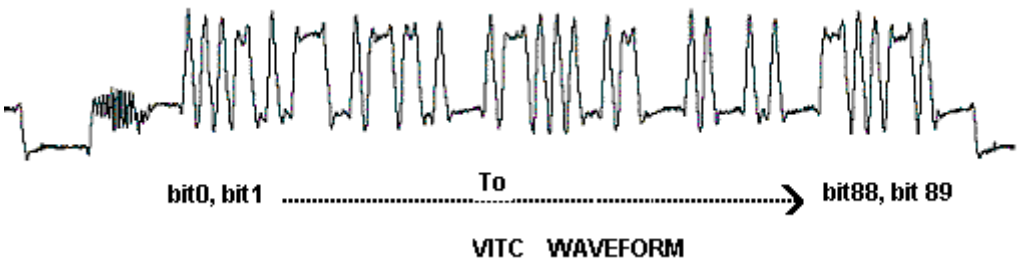


Figure 146: VITC Waveform and Decoded Data Correlation

11.3.5 VPS, PDC, UTC, Gemstar and CGMS Type B Readback Registers

The readback registers for VPS, PDC, and UTC are shared. Since Gemstar is a high data rate standard, it is available through the ancillary stream and the fast I^2C interface. However, for evaluation purposes, any one line of Gemstar is available through readback registers sharing the same register space as PDC, UTC, and VPS. Note that only *one* standard out of VPS, PDC, UTC, and Gemstar can be read back through the registers at a time.

To identify the data that should be made available in the readback registers, the user has to program GS_VPS_PDC_UTC_CGMSTB[2:0].

GS_VPS_PDC_UTC_CGMSTB[2:0], Addr 48 (VDP), *Address 0x9C[2:0]*

The readback registers for VPS, PDC, UTC and CGMS Type B are shared. These control bits identify which type of data is to be written to the shared registers.

Function

GS_VPS_PDC_UTC_CGMSTB[2:0]	Description
000 «	Gemstar 1x/2x
001	VPS
010	PDC
011	UTC
100	CGMS type B
101	Reserved
110	Reserved
111	Reserved

STATUS_CLEAR_GEMS_VPS, Addr 48 (VDP), *Address 0x78[4] (Self-Clearing)*

Gemstar or VPS data status clear. Refreshes the Gemstar and VPS status registers

Function

STATUS_CLEAR_GEMS_VPS	Description
0 «	Do not refresh the VPS status registers
1	Refresh the VPS readback registers

VDP_STATUS_GS_VPS_PDC_UTC_CGMSTB, Addr 48 (VDP), *Address 0x40[4] (Read Only)*

Gemstar, VPS, PDC, UTC, CGMS Type B Data Detection Status Bit

Function

VDP_STATUS_GS_VPS_PDC_UTC_CGMSTB	Description
0 «	Gemstar, VPS, PDC, UTC, CGMS Type B data not detected.
1	Gemstar, VPS, PDC, UTC, CGMS Type B data detected.

Table 71: VDP_GS_VPS_PDC_UTC_CGMSTB_DATA Readback Registers

GS PDC UTC Readback Registers VDP Map/SDP Map	Address
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[7:0]	0x47
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[15:8]	0x48
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[23:16]	0x49
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[31:24]	0x4A
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[39:32]	0x4B
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[47:40]	0x4C
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[55:48]	0x4D
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[63:56]	0x4E
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[71:64]	0x4F
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[79:72]	0x50
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[87:80]	0x51

GS PDC UTC Readback Registers VDP Map/SDP Map	Address
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[95:88]	0x52
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[103:96]	0x53

VPS

The VPS data bits are biphase decoded by the VDP. The decoded data is made available in both the ancillary stream and in the VDP readback registers. VPS decoded data is available in the VDP_GS_VPS_PDC_UTC_CGMSTB_DATA registers (refer to Table 71).

Gemstar

The Gemstar decoded data is made available in the ancillary stream and through the fast I²C port. For evaluation purposes only one line of Gemstar can be made available VDP readback registers. Gemstar must be selected via GS_VPS_PDC_UTC_CGMSTB[2:0]. VDP supports the autodetection of the Gemstar standard between Gemstar 1x or Gemstar 2x formats, and decodes accordingly. This autodetection feature must be enabled via the AUTO_DETECT_GEM bit and program the decoder to decode Gemstar 2x on the required lines through manual configuration of the VDP. The type of decoded Gemstar can be found out by observing the VDP_STATUS_GEMS_TYPE bit.

AUTO_DETECT_GEM, Addr 48 (VDP), *Address 0x61[4]*

Control for autodetection of the Gemstar type.

Function

AUTO_DETECT_GEM	Description
0	Disables autodetection of Gemstar type
1 «	Enables autodetection of Gemstar type

VDP_STATUS_GEMS_TYPE, Addr 48 (VDP), *Address 0x40[5] (Read Only)*

Gemstar Type Status Bit

Function

VDP_STATUS_GEMS_TYPE	Description
0 «	Gemstar 1X detected
1	Gemstar 2X detected

The Gemstar data that is available in the VDP readback register could be from any line of the input video on which Gemstar was decoded. If the user wants to read the Gemstar data on a particular video line, the user should use the manual configuration described in Section 11.1.2 and enable Gemstar decoding on only the required line.

PDC and UTC

PDC and UTC are data transmitted through teletext packet 8/30 format 2; and packet 8/30 format 1.

If PDC or UTC data is to be read through VDP readback registers, the corresponding teletext standard (WST – PAL System B) should be decoded by VDP.

The whole teletext decoded packet is output on the ancillary data stream and the user can look for the magazine number, row number, and 394designation code, and qualify the data as PDC/UTC.

If PDC/UTC packets are identified by the VDP, bytes 0 to 12 are updated to the VDP_GS_VPS_PDC_UTC_CGMSTB_DATA registers (refer to Table 72), and the VDP_STATUS_GEMS_VPS bit is set accordingly. The full packet data is also available as ancillary data.

CGMS Type B

The CGMS Type B data can be present in 525p, 720p, 1080i standards. The CGMS Type B standard has a total of 134 bits, (16 bytes, 6 bits). The decoded data is made available in both the Ancillary Stream and in the VDP readback registers. The CGMS Type B data is made available in VDP readback registers in a shared manner. The first 4 bytes are available in dedicated registers and the remaining bytes are available in registers in the share registers of VDP_GS_VPS_PDC_UTC_CGMSTB_DATA.

Table 72: CGMS Type B Readback Registers

CGMS Type B Readback Registers VDP Map	Address
VDP_CGMS_TYPEB_0	0x3C
VDP_CGMS_TYPEB_1	0x3D
VDP_CGMS_TYPEB_2	0x3E
VDP_CGMS_TYPEB_3	0x3F
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[7:0]	0x47
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[15:8]	0x48
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[23:16]	0x49
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[31:24]	0x4A
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[39:32]	0x4B
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[47:40]	0x4C
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[55:48]	0x4D
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[63:56]	0x4E
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[71:64]	0x4F
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[79:72]	0x50
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[87:80]	0x51
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[95:88]	0x52
VDP_GS_VPS_PDC_UTC_CGMSTB_DATA[103:96]	0x53

11.4 READBACK REGISTERS

I²C readback registers have separate registers for CCAP, CGMS, WSS, Gemstar, VPS, PDC/UTC, and VITC. The details of these registers and their access procedure are described in Section 11.5.

11.5 USER INTERFACE FOR I²C READBACK REGISTERS

VDP supports two types of I²C interfaces for the readback of decoded data:

1. Dedicated I²C registers (normal I²C bus)
2. Shared I²C (fast I²C bus)

Dedicated I²C readback registers have separate registers for CCAP, CGMS, WSS, GEMSTAR, VPS, PDC/UTC, and VITC, whereas the fast I²C space has a shared space of 45 registers that can be used for any or a combination of VBI data standards. The details of these

registers and their access procedure are described in this section.

11.5.1 VDP Register Readback Protocols

11.5.1.1 Data Available Updates

The VDP decodes all enabled VBI data standards in real time. Since the I²C access speed is much lower than the decoded rate, it is possible that when the registers are accessed, they are updated with data from the next line. In order to avoid this, the VDP block has a CLEAR control bit and an AVAILABLE status bit accompanying all the VDP readback registers.

Initially, the user has to clear the I²C readback register by writing 1 to the CLEAR bit (this control is self clearing). This resets the state of the AVAILABLE bit to LOW and indicates that the data in the associated readback registers are not valid. After the VDP decodes the next line of the corresponding VBI data, the decoded data is placed in the I²C readback register and the AVAILABLE bit is set to HIGH to indicate that valid data is now available.

Though the VDP will decode this VBI data, if present, in subsequent lines, the decoded data will **not** be updated to the readback registers until the CLEAR bit is set HIGH again. However, this data will be available through the 656 ancillary data packets.

Example I²C Readback Procedure:

The following tasks are performed to read one packet (line) of PDC data from the decoder.

1. Write "10" to GS_VPS_PDC_UTC_CGMSTB[2:0] to specify that PDC data has to be updated to I²C registers.
2. Set STATUS_CLEAR_GEMS_VPS to 1 to enable the update of I²C registers.
3. Poll the VDP_STATUS_GS_VPS_PDC_UTC_CGMSTB / GS_PDC_VPS_UTC_AVL_ST bit going HIGH to check the availability of the PDC packets.
4. Read the data bytes from the PDC I²C registers.
5. Repeat steps 1 to 4 to read another line or packet of data.
6. To read a packet of CC, CGMS, or WSS, steps 2, 3, and 4 only are required since they have dedicated registers.

11.5.2 Content Based Data Update

For certain standards like WSS, CGMS, Gemstar, PDC, UTC, and VPS, the information content in the transmitted signal remains the same over numerous lines but the user may want to be notified only when there is a change in the information content or loss of the information content. The user needs to enable the content based update for the required standard through the GS_VPS_PDC_UTC_CB_CHANGE and WSS_CGMS_CB_CHANGE bits. Thus, the AVAILABLE bit will show the availability of that standard only when there is a change in its content. AVAILABLE bits - are located in VDP_STATUS register.

The content based update also applies to loss of data at the lines where some data was present previously. For standards like VPS, Gemstar, CGMS, and WSS, if there is no data arrival in the next four lines programmed, the corresponding Available bit in the VDP_STATUS register is set to high and the content in the readback registers for that standard is set to zero. The user has to write high to the Clear bit so that if a valid line is decoded after some time so that the readback registers will be updated with the Available bit set to high.

If content based updating is enabled, the AVAILABLE bit is set to high (assuming the CLEAR bit was written) in the following cases:

- Data contents change
- There was some data being decoded and four lines with no data are detected

- There was no data being decoded and new data is being decoded

GS_VPS_PDC_UTC_CB_CHANGE, Addr 48 (VDP), *Address 0x9C[5]*

A control to allow content based updates of VPS, PDC and UTC data.

Function

GS_VPS_PDC_UTC_CB_CHANGE	Description
0	Disable content based update of VPS, PDC, UTC data
1 «	Enables content based update of UTC, PDC, UTC data

WSS_CGMS_CB_CHANGE, Addr 48 (VDP), *Address 0x9C[4]*

A control to allow content based updates of WSS and CGMS Type A data.

Function

WSS_CGMS_CB_CHANGE	Description
0 «	Disable content based update of WSS, CGMS Type A data
1	Enables content based update of WSS, CGMS Type A data

VDP_STATUS register (VDP Map, 0x40) consists of following bits:

- VDP_STATUS_TTXT
- VDP_STATUS_VITC
- VDP_STATUS_GEMS_TYPEVDP_STATUS_GS_VPS_PDC_UTC_CGMSTB
- VDP_STATUS_FAST_I2CVDP_STATUS_CCAP_EVEN_FIELD
- VDP_STATUS_CCAP

11.6 INTERRUPT BASED READING OF VDP READBACK REGISTERS

Some VDP status bits are also linked to the interrupt request controller so that the user does not have to poll the Available status bit. The user can configure the video decoder to trigger an interrupt request on an INTERRUPT pin in response to the valid data available in readback registers. This function is available for the following data types:

- **CGMS or WSS**
  The user can select between triggering an interrupt request each time sliced data is available or triggering an interrupt request only when the sliced data has changed.

- **Gemstar, PDC, VPS, or UTC**
  The user can select between triggering an interrupt request each time sliced data is available or triggering an interrupt request only when the sliced data has changed.

For more information on using the VDP interrupts, refer to Interrupt Section of this manual, and the ADV7842 Software Manual.

11.7 VDP ANCILLARY DATA OUTPUT

Reading the data back via I²C may not be feasible for VBI data standards with high data rates, for example, teletext. An alternative is to place the sliced data in a packet in the line blanking of the digital output CCIR656 stream. This is available for all standards sliced by the VDP module.

When VBI data is sliced on a given line, the corresponding ancillary data packet is placed immediately after the next EAV code that occurs at the output, that is, sliced data from multiple lines are not buffered up and then emitted in a burst. Note that the line number on which the packet is placed will differ from the line number on which the data was sliced due to the vertical delay through the comb filters.

The user can enable or disable the insertion of VDP decoded results into the 656 ancillary streams by using the ADF_EN bit

ADF_EN, Addr 48 (VDP), *Address 0x62[7]*

Controls insertion of VBI decoded data into the 656 datastream as ancillary data

Function

ADF_EN	Description
0 «	VBI decoded data not inserted into ancillary 656 stream
1	VBI decoded data inserted into ancillary 656 stream

Data Identification Word (DID) ADF_DID[4:0] and the Secondary Data Identification ADF_SDID[5:0] can be set by programming the following registers.

ADF_DID[4:0], Addr 48 (VDP), *Address 0x62[4:0]*

Control to specify the value of the DID sent in the ancillary stream with VBI decoded data

Function

ADF_DID[4:0]	Description
xxxxx	User specified DID sent in ancillary stream with VDP decoded data
10101 «	Default

ADF_SDID[5:0], Addr 48 (VDP), *Address 0x63[5:0]*

Control to specify the value of the SDID sent in the ancillary stream with VBI decoded data

Function

ADF_SDID[5:0]	Description
0x2A «	Default value

The VDP can output the ancillary data packets spread across the Y and C buses, or they can be duplicated on both the channels.

TOGGLE_ADF, Addr 48 (VDP), *Address 0x63[7]*

Control to specify how the ancillary data is placed in the luma and chroma datastreams

Function

TOGGLE_ADF	Description
0 «	Ancillary data packet is spread across the Y and C data streams
1	Ancillary data packet is duplicated across the Y and C data streams

11.8 NIBBLE OUTPUT MODE

The ancillary data packet sequence is explained in Table 73 and Table 74. Nibble mode is the default mode of output from the ancillary stream when the ancillary stream output is enabled. This format is in compliance with ITU-R BT.1364. Alternatively, the ancillary data packet can also be output in a byte mode. ADF_MODE[1:0] controls the packet mode.

ADF_MODE[1:0], Addr 48 (VDP), *Address 0x62[6:5]*

Control to set ADF mode for ancillary data

Function

ADF_MODE[1:0]	Description
00 «	Nibble mode
01	Byte mode, no code restrictions
10 = Byte mode, but 0x00 and 0xFF prevented (0x00	0x01) (0xFF->0xFE)
11	Reserved

The following definitions apply to the abbreviations in Table 73 and Table 74.

- **EF**
  Even Field signal. Indicates the field where this data packet was decoded.
- **EP**
  Even parity for bits B8 to B2. This means that the parity bit EP is set so there is an even number of 1s in bits in B8 to B2, including the parity bit B8.
- **CS**
  Checksum word. The CS word is used to increase confidence in the integrity of the ancillary data packet from the DID, SDID, and DC through the UDWs (User Data Words). It consists of 10 bits, a 9-bit calculated value, and B9 as the inverse of B8. The checksum value B8 to B0 is equal to the nine least significant bits of the total sum of the 9 least significant bits of the DID, SDID, DC words, and all UDWs in the packet. Prior to the start of the checksum count cycle, all checksum and carry bits are preset to zero. Any carry out resulting from the checksum count cycle is ignored.
- **$\overline{\text{EP}}$**
  The MSB B9 is the inverse even parity and this ensures that restricted codes 0x00 and 0xFF will not occur.
- **LCOUNT[11:0]**
  This number represents the line number from which the VBI data was decoded. The line number is as per the numbering system in ITU-R BT.470. The line number runs from 1 to 625 in a 625 line system. For PAL, the line number coming out with the ancillary stream is plus one of the line number in Table 68. This applies for both even and odd fields of PAL.
- **DC (Data Count)**
  The data count specifies the number of user data words in the ancillary stream for the standard. The *total number of user data words = 4 * Data Count*. Padding words can be introduced to make the total number of user data words divisible by four.
- **PADDING[1:0]**
  This value specifies how many padding words were added to the packet. Padding words are added to the end of packet (refer to the bottom of Table 73, Table 74) to make total number of user data words divisible by four. Total number of user data words that is divisible by 4 is a requirement of ITU-R BT.1364.

Table 73: Ancillary Data in Nibble Output Format

Byte	B9	B8	B7	B6	B5	B4	B3	B2	B1	B0	Description
0	0	0	0	0	0	0	0	0	0	0	Ancillary data preamble
1	1	1	1	1	1	1	1	1	1	1	
2	1	1	1	1	1	1	1	1	1	1	
3	$\overline{\text{EP}}$	EP		ADF_DID[4:0]					0	0	Data identification word (DID)

Byte	B9	B8	B7	B6	B5	B4	B3	B2	B1	B0	Description
4	$\overline{EP}$	EP	ADF_SDID[5:0]						0	0	Secondary data identification word (SDID)
5	$\overline{EP}$	EP	0	DC[4:0]					0	0	Data count
6	EP	EP	PADDING [1:0]		VBI_DATA_STD[3:0]				0	0	ID0 – user data word 1
7	$\overline{EP}$	EP	LCOUNT[11:6]						0	0	ID1 – user data word 2
8	$\overline{EP}$	EP	LCOUNT[5:0]						0	0	ID2 – user data word 3
9	$\overline{EP}$	EP	0	0	0	EF	VDP_TTXT_TYPE[1:0]		0	0	ID3 – user data word 4
10	$\overline{EP}$	EP	0	0	VBI_WORD_1[7:4]				0	0	User data word 5
11	$\overline{EP}$	EP	0	0	VBI_WORD_1[3:0]				0	0	User data word 6
12	$\overline{EP}$	EP	0	0	VBI_WORD_2[7:4]				0	0	User data word 7
13	$\overline{EP}$	EP	0	0	VBI_WORD_2[3:0]				0	0	User data word 8
14	$\overline{EP}$	EP	0	0	VBI_WORD_3[7:4]				0	0	User data word 9
-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-
n-3	1	0	0	0	0	0	0	0	0	0	(Pad 0x200. These padding words may or may not be present depending on ancillary data type.) User data word
n-2	1	0	0	0	0	0	0	0	0	0	
n-1	$\overline{B8}$	Checksum							0	0	

Table 74: Ancillary Data in Byte Output Format

Byte	B9	B8	B7	B6	B5	B4	B3	B2	B1	B0	Description
0	0	0	0	0	0	0	0	0	0	0	Ancillary data preamble
1	1	1	1	1	1	1	1	1	1	1	
2	1	1	1	1	1	1	1	1	1	1	
3	$\overline{EP}$	EP		ADF_DID[4:0]					0	0	DID – data identification word
4	$\overline{EP}$	EP	ADF_SDID[5:0]						0	0	SDID – secondary data identification word
5	$\overline{EP}$	EP	0	DC[4:0]					0	0	Data count
6	$\overline{EP}$	EP	PADDING[1:0]		VBI_DATA_STD[3:0]				0	0	ID0 – user data word 1
7	$\overline{EP}$	EP	LCOUNT[11:6]						0	0	ID1 – user data word 2
8	$\overline{EP}$	EP	LCOUNT[5:0]						0	0	ID2 – user data word 3
9	$\overline{EP}$	EP	0	0	0	EF	VDP_TTXT_TYPE[1:0]		0	0	ID3 – user data word 4
10	VBI_WORD_1[7:0]								0	0	User data word 5
11	VBI_WORD_2[7:0]								0	0	User data word 6
12	VBI_WORD_3[7:0]								0	0	User data word 7
13	VBI_WORD_4[7:0]								0	0	User data word 8
14	VBI_WORD_5[7:0]								0	0	User data word 9
-	-	-	-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-	-	-	-

Byte	B9	B8	B7	B6	B5	B4	B3	B2	B1	B0	Description
-	-	-	-	-	-	-	-	-	-	-	-
n-1	1	0	0	0	0	0	0	0	0	0	(Pad 0x200. These padding words may or may not be present depending on ancillary data type.) User data word
n-2	1	0	0	0	0	0	0	0	0	0	
n-1	$\overline{B8}$	CHECKSUM							0	0	

Note that this mode does **not** fully comply with ITU-R BT.1364.

11.9 STRUCTURE OF VBI WORDS IN ANCILLARY DATA STREAM

Each VBI data standard is split into a clock run in (CRI), a framing code (FC), and 'n' number of data bytes. The data packet in the ancillary stream includes only the FC and data bytes. The VBI_WORD_x in the ancillary data stream has the format described in Table 75.

Table 75: Structure of VBI Data Words in Ancillary Stream

VBI Word Number	Ancillary Data Composition	
VBI_WORD_1	FC0	FRAMING_CODE[23:16]
VBI_WORD_2	FC1	FRAMING_CODE [15:8]
VBI_WORD_3	FC2	FRAMING_CODE[7:0]
VBI_WORD_4	DB1	1st data byte
…	...	…
VBI_WORD_n+3	DBn	Last (nth) data byte

11.9.1 Framing Code

The length of the actual framing code depends on the VBI data standard. For uniformity, the length of the framing code reported in the ancillary data stream is always 24 bits. For standards with a lesser framing code length, the extra LSB bits are set to 0. The valid length of the framing code can be decoded from the VBI_DATA_STD bits available in ID0 (UDW 1). The framing code is always reported in the inverse transmission order. Table 76 shows the framing code and its valid length for VBI data standards supported by VDP.

Table 76: Framing Code Sequence for Different VBI Standards

VBI Standard	Framing Code Length (in Bits)	Error Free Framing Code Bits (in Order of Transmission)	Error Free Framing Code Given out by VDP (Reverse Order of Transmission)
TTXT_SYSTEM_A (625i)	8	1110_0111	1110_0111
TTXT_SYSTEM_B (625i)	8	1110_0100	0010_0111
TTXT_SYSTEM_B (525i)	8	1110_0100	0010_0111
TTXT_SYSTEM_C (625i and 525i)	8	1110_0111	1110_0111
TTXT_SYSTEM_D (625i and 525i)	8	1110_0101	1010_0111
VPS (625i)	16	1000_1010_10001_1001	1001_1001_0101_0001
VITC (525i and 625i)	1	0	0
WSS (625i)	24	0001_1110_0011_1100_0001_1111	1111_1000_0011_1100_0111_1000
GEMSTAR_1X (525i)	3	001	100
GEMSTAR_2X (525i)	11	1001_1101_101	101_1011_1001
CCAP (525i and 625i)	3	001	100

VBI Standard	Framing Code Length (in Bits)	Error Free Framing Code Bits (in Order of Transmission)	Error Free Framing Code Given out by VDP (Reverse Order of Transmission)
CGMS (525i)	1	0	0
CGMS (525p)	1	0	0
CGMS (720p)	1	0	0
CGMS (625p)	24	0001_1110_0011_1100_0001_1111	1111_1000_0011_1100_0111_1000
CGMS (1080i)	1	0	0
VITC (1080i)	1	0	0

Example:

The sequence of data in the VBI region is Bit 0 …. Bit 7 (Bit 0 comes first). The actual data will look reversed in the order of transmission (for example, 0x27 will look like 0xE4).

For teletext (B-WST), the framing code byte is 1110_0100_b ($E4_h$) (bits shown in the order of transmission). Thus, VBI_WORD_1 = 0x27, VBI_WORD _2 = 0x00, and VBI_WORD _3 = 0x00.

This VBI data translates into UDWs in the data stream as follows.

For the nibble mode:

UDW5[5:2] = 0010
UDW6[5:2] = 0111
UDW7[5:2] = 0000 (undefined bits made zeros)
UDW8[5:2] = 0000 (undefined bits made zeros)
UDW9[5:2] = 0000 (undefined bits made zeros)
UDW10[5:2] = 0000 (undefined bits made zeros)
(refer to Table 73)

For the byte mode:

UDW5[9:2] = 0010_0111
UDW6[9:2] = 0000_0000 (undefined bits made zeros)
UDW7[9:2] = 0000_0000 (undefined bits made zeros)
(refer to Table 74)

11.9.2 Data Bytes

The VBI_WORD_4 to VBI_WORD_n+3 contains the data words decoded by the VDP in the transmission order. The position of the bits in the bytes is in the inverse transmission order.

For example, closed caption has two user data bytes, as shown in Table 77. The data bytes in the ancillary data stream are as follows:

- VBI_WORD_4 = Byte 1[7:0]
- VBI_WORD_5 = Byte 2[7:0]

The number of VBI_WORDs for each VBI data standard and the total number of UDWs in the ancillary data stream are shown in Table 77.

Table 77: Total Number of User Data Words for Different VBI Standards

VBI Standard	ADF_MODE[1:0]	ID UDWs	Framing_code UDWs	VBI Data Words	Number of Padding Words	Total Number of UDWs
TTXT_SYSTEM_A (625i)	00 (nibble mode)	4	6	74	0	84
	01,10 (byte mode)	4	3	37	0	44
TTXT_SYSTEM_B (625i)	00 (nibble mode)	4	6	84	2	96
	01,10 (byte mode)	4	3	42	3	52
TTXT_SYSTEM_B (525i)	00 (nibble mode)	4	6	68	2	80
	01,10 (byte mode)	4	3	34	3	44
TTXT_SYSTEM_C (625i and 525i)	00 (nibble mode)	4	6	66	0	76
	01,10 (byte mode)	4	3	33	2	42
TTXT_SYSTEM_D (625iI and 525i)	00 (nibble mode)	4	6	68	2	80
	01,10 (byte mode)	4	3	34	3	44
VPS (625i)	00 (nibble mode)	4	6	26	0	36
	01,10 (byte mode)	4	3	13	0	20
VITC (525i and 625i)	00 (nibble mode)	4	6	18	0	28
	01,10 (byte mode)	4	3	9	0	16
WSS (625i)	00 (nibble mode)	4	6	4	2	16
	01,10 (byte mode)	4	3	2	3	12
GEMSTAR_1X (525i)	00 (nibble mode)	4	6	4	2	16
	01,10 (byte mode)	4	3	2	3	12
GEMSTAR_2X (525i)	00 (nibble mode)	4	6	8	2	20
	01,10 (byte mode)	4	3	4	1	12
CCAP (525i and 625i)	00 (nibble mode)	4	6	4	2	16
	01,10 (byte mode)	4	3	2	3	12
CGMS (525i)	00 (nibble mode)	4	6	6	0	16
	01,10 (byte mode)	4	3	3	2	12
CGMS(525p)	00 (nibble mode)	4	6	6	0	16
	01,10 (byte mode)	4	3	3	2	12
CGMS(625p)	00 (nibble mode)	4	6	4	2	16
	01,10 (byte mode)	4	3	2	3	12
CGMS(720p)	00 (nibble mode)	4	6	6	0	16
	01,10 (byte mode)	4	3	3	2	12
CGMS(1080i)	00 (nibble mode)	4	6	6	0	16
	01,10 (byte mode)	4	3	3	2	12
CGMS Type B	00 (nibble mode)	4	3	34	0	44
	01,10 (byte mode)	4	6	17	0	24
VITC(1080i)	00 (nibble mode)	4	6	18	0	28
	01,10 (byte mode)	4	3	9	0	16

11.10 FAST I²C READBACK REGISTERS

For all VBI data standards, the user can read back the decoded data on lines through the fast I2C interface. Users can select the standard(s) they want to read through fast I2C registers. A FIFO is used in VDP to facilitate storage of decoded data while data is being read through fast I2C. The data in FIFO is arranged in the following order:

- Type of standard (VBI_DATA_STD)
- Number of data bytes that follow (packet size)
- Decoded data bytes

After setting the required bits, the user can read back the decoded data from the registers addressed (0xC4 to 0xF0) of VDP map.

The first two bytes of the registers let the user know the type of standard for which the data is available and the number of data bytes to be read.

Table 78 explains the bits to be set for different VBI data that are available through the fast I²C registers.

Table 78: Configuration VBI Standard to be Output on Fast I²C

VDP_FAST_REG_CONF_CUS , VDP Map 0xC0[7]	VDP_FAST_REG_CONF_CUST, VDP Map 0xC0[6]	VDP_FAST_REG_CONF_CCAP, VDP Map 0xC0[5]	VDP_FAST_REG_CONF_GEM1X_2X, VDP Map 0xC0[4]	VDP_FAST_REG_CONF_CGMS_WSS, VDP Map 0xC0[3]	VDP_FAST_REG_CONF_VITC, VDP Map 0xC0[2]	VDP_FAST_REG_CONF_VPS_CGMSTB, VDP Map 0xC0[1]	VDP_FAST_REG_CONF_TTXT, VDP Map 0xC0[0]	Description
0	0	0	0	0	0	0	0	None (default)
0	0	0	0	0	0	0	1	Teletext
0	0	0	0	0	0	1	0	VPS
0	0	0	0	0	1	0	0	VITC
0	0	0	0	1	0	0	0	CGMS/WSS
0	0	0	1	0	0	0	0	Gemstar 1x/2x
0	0	1	0	0	0	0	0	CCAP
0	1	0	0	0	0	0	0	Custom Mode1
1	0	0	0	0	0	0	0	Custom Mode2

11.10.1 Enabling Fast I²C Port

By default, the fast I²C port on the ADV7842 is disabled.

To enable the fast I2C port:

- Select an SD PRIM_MODE and VID_STD setting
- Ensure that the VDP I²C address is set accordingly
- Set the SUBI2C_EN and VDP_ON_SUB_I2C bits in the IO Map

Access through the main I²C will be disabled when the fast I²C port is enabled.

SUBI2C_EN, Addr 40 (IO), *Address 0x1A[1]*

A control to enable the secondary I2C interface used for fast access to VBI data.

Function

SUBI2C_EN	Description
0	Disable secondary I2C interface
1 «	Enable secondary I2C interface

VDP_ON_SUB_I2C, Addr 40 (IO), *Address 0x1A[0]*

A control to allow VDP Map read access from secondary I2C interface.

Function

VDP_ON_SUB_I2C	Description
0 «	Disable VDP Map read access from sub I2C
1	Enable VDP Map read access from sub I2C

11.10.2 Fast I²C Readbacks

VDP_FAST_VBI_STD[3:0], Addr 48 (VDP), *Address 0xC2[3:0] (Read Only)*

Readback of the VBI standard in the fast I2C registers

Function

VDP_FAST_VBI_STD[3:0]	Description
0001	Teletext
0010	VPS
0011	VITC
0100	WSS / CGMS Type A
0101	Gemstar 1X
0110	Gemstar 2X
0111	CCAP
1000	CGMS Type B
1001	Reserved
1010	Reserved
1011	Reserved
1100	Reserved
1101	Custom 1
1110	Custom 2
1111	Reserved

VDP_FAST_PACKET_SIZE[7:0], Addr 48 (VDP), *Address 0xC3[7:0] (Read Only)*

Readback of the number of bytes contained in the fast I2C registers

Function

VDP_FAST_PACKET_SIZE[7:0]	Description
xxxxxxxx	Number of bytes contained in the fast I2C registers

VDP_STATUS_FAST_I2C, Addr 48 (VDP), *Address 0x40[3] (Read Only)*

Status of data availability in fast I2C regs

Function

VDP_STATUS_FAST_I2C	Description
0 «	Data is not available since last fast I2C read
1	Data is available since last fast I2C read

VDP_STATUS_CLEAR_FAST_I2C, Addr 48 (VDP), *Address 0x78[3] (Self-Clearing)*

Clears fast I2C status bit (VDP Map 0x40 [3])

Function

VDP_STATUS_CLEAR_FAST_I2C	Description
1	Clears status_fast_i2c bit

FAST_I2C data is located in VDP map (0xC4 – 0xF0):

Table 79: VDP_FAST_REG register location

Register Name	Address in VDP Map
VDP_FAST_REG00[7:0]	0xC4[7:0]
VDP_FAST_REG01[7:0]	0xC5[7:0]
VDP_FAST_REG02[7:0]	0xC6[7:0]
VDP_FAST_REG03[7:0]	0xC7[7:0]
VDP_FAST_REG04[7:0]	0xC8[7:0]
VDP_FAST_REG05[7:0]	0xC9[7:0]
VDP_FAST_REG06[7:0]	0xCA[7:0]
VDP_FAST_REG07[7:0]	0xCB[7:0]
VDP_FAST_REG08[7:0]	0xCC[7:0]
VDP_FAST_REG09[7:0]	0xCD[7:0]
VDP_FAST_REG10[7:0]	0xCE[7:0]
VDP_FAST_REG11[7:0]	0xCF[7:0]
VDP_FAST_REG12[7:0]	0xD0[7:0]
VDP_FAST_REG13[7:0]	0xD1[7:0]
VDP_FAST_REG14[7:0]	0xD2[7:0]
VDP_FAST_REG15[7:0]	0xD3[7:0]
VDP_FAST_REG16[7:0]	0xD4[7:0]
VDP_FAST_REG17[7:0]	0xD5[7:0]
VDP_FAST_REG18[7:0]	0xD6[7:0]
VDP_FAST_REG19[7:0]	0xD7[7:0]
VDP_FAST_REG20[7:0]	0xD8[7:0]
VDP_FAST_REG21[7:0]	0xD9[7:0]
VDP_FAST_REG22[7:0]	0xDA[7:0]
VDP_FAST_REG23[7:0]	0xDB[7:0]
VDP_FAST_REG24[7:0]	0xDC[7:0]
VDP_FAST_REG25[7:0]	0xDD[7:0]
VDP_FAST_REG26[7:0]	0xDE[7:0]
VDP_FAST_REG27[7:0]	0xDF[7:0]
VDP_FAST_REG28[7:0]	0xE0[7:0]
VDP_FAST_REG29[7:0]	0xE1[7:0]

Register Name	Address in VDP Map
VDP_FAST_REG30[7:0]	0xE2[7:0]
VDP_FAST_REG31[7:0]	0xE3[7:0]
VDP_FAST_REG32[7:0]	0xE4[7:0]
VDP_FAST_REG33[7:0]	0xE5[7:0]
VDP_FAST_REG34[7:0]	0xE6[7:0]
VDP_FAST_REG35[7:0]	0xE7[7:0]
VDP_FAST_REG36[7:0]	0xE8[7:0]
VDP_FAST_REG37[7:0]	0xE9[7:0]
VDP_FAST_REG38[7:0]	0xEA[7:0]
VDP_FAST_REG39[7:0]	0xEB[7:0]
VDP_FAST_REG40[7:0]	0xEC[7:0]
VDP_FAST_REG41[7:0]	0xED[7:0]
VDP_FAST_REG42[7:0]	0xEE[7:0]
VDP_FAST_REG43[7:0]	0xEF[7:0]
VDP_FAST_REG44[7:0]	0xF0[7:0]

12 MEMORY CONTROLLER

12.1 MEMORY REQUIREMENTS

The ADV7842 supports either a DDR or a SDR SD memory interface. Depending on the feature required, the appropriate memory device can be selected. The memory is required for 3D comb and frame TBC. Memory configuration options for the external SD RAM are provided in Table 80.

Table 80: Memory Configuration Options

Memory Size	Comb Only		TBC Only		Comb+TBC	
	NTSC	PAL	NTSC	PAL	NTSC	PAL
64 Mb SDR	Yes	Yes	Yes	Yes	No	No
128 Mb DDR	Yes	Yes	Yes	Yes	Yes	Yes
256 Mb DDR	Yes	Yes	Yes	Yes	Yes	Yes
512 Mb DDR	Yes	Yes	Yes	Yes	Yes	Yes

12.2 MEMORY SPECIFICATIONS (DDR SDRAM)

The ADV7842 supports a DDR SDRAM memory interface for 3D comb and frame synchronizer operations as follows:

- 128 Mb, 256, and 512 Mb DDR SDRAM memory are supported
- Required memory architecture is four banks of 2 Mb x16 (8M16)
- Speed grade of 133 MHz at CAS Latency (CL) 2.5 is required
- Examples of 128 Mb compatible memory include:
  - Micron MT46V16M16P-6T IT K
  - Hynix H5DU1262GTR-E3C
- 22 Ω series termination resistors are recommended for this configuration

Table 81: DDR SDRAM Layout Requirements

Decoder Pin	SDRAM Pin	Physical Requirement
SDRAM_A0 to SDRAM_A11	A0 to A11	Not critical, not longer than necessary
SDRAM_BA0 to SDRAM_BA1	BA0 to BA1	Not critical but not longer than necessary
SDRAM_CSN	CS#	Not critical but not longer than necessary
SDRAM_RASN	RAS#	Not critical but not longer than necessary
SDRAM_CASN	CAS#	Not critical but not longer than necessary
SDRAM_WEN	WE#	Not critical but not longer than necessary
SDRAM_LDM	LDM	Minimum skew among LDQS, UDQS, DQ15-DQ0, LDM, and UDM group
SDRAM_LDQS	LDQS	As short as possible and minimum skew among LDQS, UDQS, DQ15-DQ0, LDM,

Decoder Pin	SDRAM Pin	Physical Requirement
		and UDM group
SDRAM_DQ0 to SDRAM_DQ15	DQ0 to DQ15	Minimum skew among LDQS, UDQS, DQ15-DQ0, LDM, and UDM group
SDRAM_UDQS	UDQS	As short as possible and minimum skew among LDQS, UDQS, DQ15-DQ0, LDM, and UDM group
SDRAM_UDM	UDM	Minimum skew among LDQS, UDQS, DQ15-DQ0, LDM and UDM group
SDRAM_CKN	CK#	As short as possible and extremely tight skew between CK and CK# pair
SDRAM_CK	CK	As short as possible and extremely tight skew between CK and CK# pair
SDRAM_CKE	CKE	Not critical but not longer than necessary

12.3 MEMORY SPECIFICATIONS (SDR SDRAM)

The ADV7842 uses external SDR memory for 3-D comb or frame synchronizer operations as follows:

- 64 Mb SDR SDRAM memory requirement
- Required memory architecture is four banks of 1 Mb x16 (4M16)
- Speed grade of 133 MHz at CL 3 is required
- 22 Ω series termination resistors are recommended for this configuration

Table 82: Layout Requirements

Decoder Pin	SDRAM Pin	Physical Requirement
SDRAM_A0 to SDRAM_A11	A0 to A11	Not critical, no longer than necessary
SDRAM_BA0 to SDRAM_BA1	BA0 to BA1	Not critical, no longer than necessary
SDRAM_CSN	CS#	Not critical, no longer than necessary
SDRAM_RASN	RAS#	Not critical, no longer than necessary
SDRAM_CASN	CAS#	Not critical, no longer than necessary
SDRAM_WEN	WE#	Not critical, no longer than necessary
SDRAM_LDM	DQML	Minimum skew among DQ15-DQ0, LDM, and UDM group
SDRAM_LDQS	NC	
SDRAM_DQ0 to SDRAM_DQ15	DQ0 to DQ15	Minimum skew among LDQS, UDQS, DQ15-DQ0, LDM, and UDM group
SDRAM_UDQS	NC	
SDRAM_UDM	DQMH	Minimum skew among DQ15-DQ0, LDM, and UDM group
SDRAM_CKN	NC	
SDRAM_CK	CLK	As short as possible
SDRAM_CKE	CKE	Not critical, no longer than necessary

12.4 MEMORY CONTROLS

SDP_SDRAM_MEM, Addr 94 (SDP_IO), *Address 0x6F[0]*

A control to configure the memory controller for either SDR or DDR interface. Refer to the Hardware Manual for further memory interface configuration.

Function

SDP_SDRAM_MEM	Description
0 «	Select DDR external memory
1	Select SDR external memory

SDP_MEM_RESET, Addr 40 (IO), *Address 0xFF[2] (Self-Clearing)*

Memory interface reset

Function

SDP_MEM_RESET	Description
0 «	No function
1	Apply SDP Memory reset

SDP_MEM_SM_RESET, Addr 94 (SDP_IO), *Address 0x60[0] (Self-Clearing)*

A control to reset the memory controller state machine. This allows user to change parameters and re-initialize without full reset. This is a self clearing control.

Function

SDP_MEM_SM_RESET	Description
0 «	Don't reset memory controller state machine.
1	Reset memory controller state machine. (self clearing bit)

12.5 CONFIGURATION EXAMPLE

12.5.1 DDR SDRAM Memory

The ADV7842 memory controller is configured by default for a 128 Mb DDR memory module.
The ADV7842 can be configured as outlined in Table 83 or as in Table 84.

Table 83: DDR Recommended Configuration for 128 Mb Memory Size

Register Address	Value	Description
0x6F (SDP IO Map)	0x00	Sets up memory controller for DDR mode.
0x75 (SDP IO Map)	0x0A	Sets 128 Mb memory size.
0x60 (SDP IO Map)	0x01	Resets memory controller. The memory controller must be reset before changes take effect.
0x7A (SDP IO Map)	0xA5	Timing adjustment for memory interface.
0x7B (SDP IO Map)	0x8F	Timing adjustment for memory interface.

Table 84: DDR Recommended Configuration for 256 Mb Memory Size

Register Address	Value	Description
0x6F (SDP IO Map)	0x00	Sets up memory controller for DDR mode.
0x75 (SDP IO Map)	0x0A	Sets 256 Mb memory size.
0x60 (SDP IO Map)	0x01	Resets memory controller. The memory controller must be reset before changes take effect.
0x7A (SDP IO Map)	0xA5	Timing adjustment for memory interface.
0x7B (SDP IO Map)	0x8F	Timing adjustment for memory interface.

12.5.2 SDR SDRAM Memory

Table 85 outlines how the ADV7842 can be configured for a 64 Mb SDR memory module. The 64 Mb SDR memory module does not allow simultaneous operation of 3D comb or frame TBC. The 64 Mb SDR memory can only support these functions individually.

Table 85: SDR Configuration

Register Address	Value	Description
0x75 (SDP IO Map)	0x0A	Sets 64 Mb memory size and exit refresh time for SDR SDRAM memory.
0x74 (SDP IO Map)	0x00	Must be set to 0 for SDR SDRAM.
0x79 (SDP IO Map)	0x33	Sets CAS latency to 3 (depends on memory).
0x6F (SDP IO Map)	0x01	Sets up memory controller for SDR mode.
0x60 (SDP IO Map)	0x01	Resets memory controller. The memory controller must be reset before changes take effect.
0x7A (SDP IO Map)	0xA5	Timing adjustment for memory interface.
0x7B (SDP IO Map)	0x8F	Timing adjustment for memory interface.

13 CONSUMER ELECTRONICS CONTROL

The Consumer Electronics Control (CEC) module features the hardware required to behave as an initiator or a follower as per the specifications for a CEC device. The CEC module contains four main sections:

- Transmit section CEC_TX
- Receive section CEC_RX
- Clock generator section CEC_CLK_GEN
- Antiglitch filter section CEC_ANTI_GLITCH

The block diagram of the CEC module is shown in Figure 147.

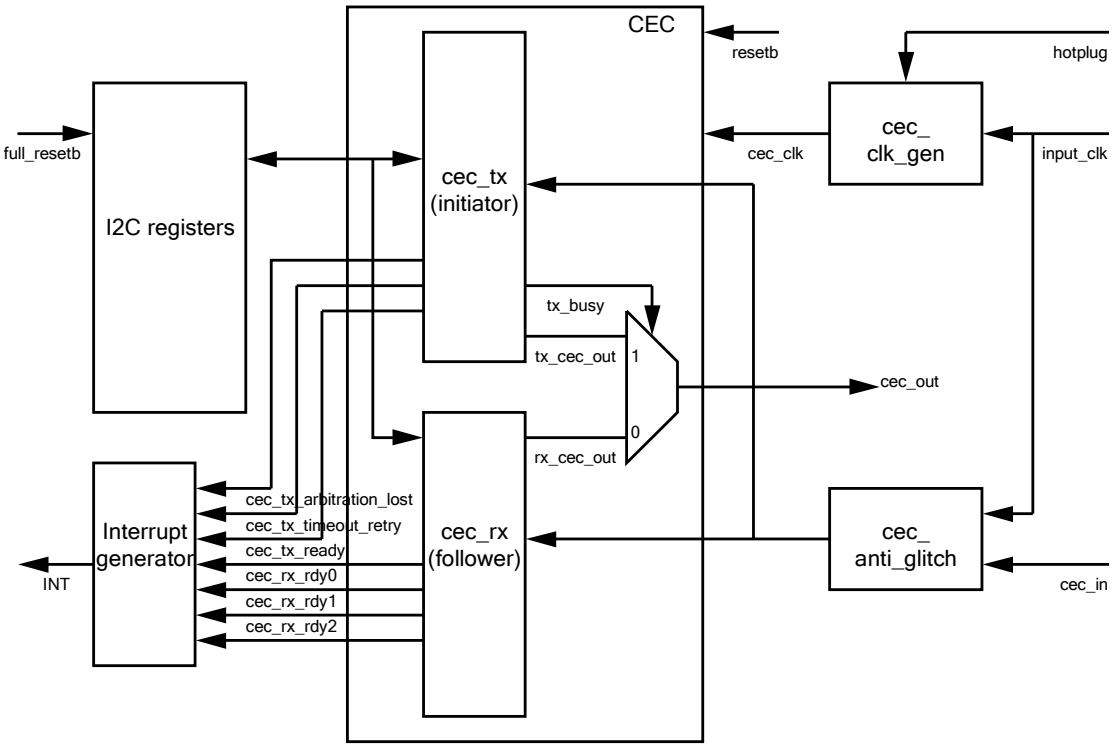


Figure 147: CEC Module Block Diagram

13.1 MAIN CONTROLS

This section describes the main controls for the CEC module.

CEC_POWER_UP, Addr 80 (CEC), *Address 0x2A[0]*

Power Mode of CEC module

Function

CEC_POWER_UP	Description
0 «	Power down the CEC module
1	Power up the CEC module

CEC_SOFT_RESET, Addr 80 (CEC), *Address 0x2C[0] (Self-Clearing)*

CEC module software reset.

Function

CEC_SOFT_RESET	Description
0 «	No function
1	Reset the CEC module

Note that the CEC_POWER_UP bit can be used to set the ADV7842 to power-down mode 1 (refer to Section **3.2.3.2**).

13.2 CEC TRANSMIT SECTION

The transmit section features the hardware required for the CEC module to act as an initiator. The host utilizes this section to transmit directly addressed messages or broadcast messages on the CEC bus. When the host wants to a send message to other CEC devices, it writes the message to the CEC outgoing message registers (refer to Table 86) and the message length register. Then, the host enables the transmission process by setting the CEC_TX_ENABLE bit to 1. When the message transmission is completed, or if an error occurs, the CEC transmitter section generates an interrupt (assuming the corresponding interrupt mask bits are set accordingly).

Table 86: CEC Outgoing Message Buffer Registers

Register Name	CEC Map Address	Description
CEC_TX_FRAME_HEADER[7:0]	0x00	Header of next outgoing message
CEC_TX_FRAME_DATA0[7:0]	0x01	Byte 0 of next outgoing message
CEC_TX_FRAME_DATA1[7:0]	0x02	Byte 1 of next outgoing message
CEC_TX_FRAME_DATA2[7:0]	0x03	Byte 2 of next outgoing message
CEC_TX_FRAME_DATA3[7:0]	0x04	Byte 3 of next outgoing message
CEC_TX_FRAME_DATA4[7:0]	0x05	Byte 4 of next outgoing message
CEC_TX_FRAME_DATA5[7:0]	0x06	Byte 5 of next outgoing message
CEC_TX_FRAME_DATA6[7:0]	0x07	Byte 6 of next outgoing message
CEC_TX_FRAME_DATA7[7:0]	0x08	Byte 7 of next outgoing message
CEC_TX_FRAME_DATA8[7:0]	0x09	Byte 8 of next outgoing message
CEC_TX_FRAME_DATA9[7:0]	0x0A	Byte 9 of next outgoing message
CEC_TX_FRAME_DATA10[7:0]	0x0B	Byte 10 of next outgoing message
CEC_TX_FRAME_DATA11[7:0]	0x0C	Byte 11 of next outgoing message
CEC_TX_FRAME_DATA12[7:0]	0x0D	Byte 12 of next outgoing message
CEC_TX_FRAME_DATA13[7:0]	0x0E	Byte 13 of next outgoing message
CEC_TX_FRAME_DATA14[7:0]	0x0F	Byte 14 of next outgoing message

CEC_TX_FRAME_LENGTH[4:0], Addr 80 (CEC), *Address 0x10[4:0]*

Message size of the transmitted frame. This is the number of byte in the outgoing message including the header.

Function

CEC_TX_FRAME_LENGTH[4:0]	Description
xxxxx	Total number of bytes (including header byte) to be sent

CEC_TX_ENABLE, Addr 80 (CEC), *Address 0x11[0]*

This bit enables the TX section. When set to 1 it initiates the start of transmission of the message in the outgoing message buffer. When the message transmission is completed this bit is automatically reset to 0. If it is manually set to 0 during a message transmission it may terminate the transmission depending on what stage of the transmission process has been reached. If the message transmission is sill in the 'signal free time' stage the message transmission will be terminated. If data transmission has begun then the transmission will continue until the message is fully sent, or until an error condition occurs.

Function

CEC_TX_ENABLE	Description
0 «	Transmission mode disabled
1	Transmission mode enabled and message transmission started

The ADV7842 features three status bits related to the transmission of CEC messages. The events that set these bits are mutually exclusive – i.e. only one of the three events can occur during any given message transmission.

- CEC_TX_READY_ST
- CEC_TX_ARBITRATION_LOST_ST
- CEC_TX_RETRY_TIMEOUT_ST

CEC_TX_READY_ST, Addr 40 (IO), *Address 0x93[0] (Read Only)*

Latched status of CEC_TX_READY_RAW signal. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit. When the CEC TX successfully sends the current message this bit is set. Once set this bit will remain high until the interrupt has been cleared via CEC_TX_READY_CLR.

Function

CEC_TX_READY_ST	Description
0 «	No change
1	Message transmitted successfully

CEC_TX_ARBITRATION_LOST_ST, Addr 40 (IO), *Address 0x93[1] (Read Only)*

Latched status of CEC_TX_ARBITRATION_LOST_RAW signal. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit. If the CEC TX loses arbitration while trying to send a message this bit is set. Once set this bit will remain high until the interrupt has been cleared via CEC_TX_ARBITRATION_LOST_CLR.

Function

CEC_TX_ARBITRATION_LOST_ST	Description
0 «	No change
1	The CEC TX has lost arbitration to another TX

CEC_TX_RETRY_TIMEOUT_ST, Addr 40 (IO), *Address 0x93[2] (Read Only)*

Latched status of CEC_TX_RETRY_TIMEOUT_RAW signal. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit. If the CEC TX fails to send the current message within the number of retry attempts specified by CEC_TX_RETRY this bit is set. Once set this bit will remain high until the interrupt has been cleared via CEC_TX_RETRY_TIMEOUT_CLR.

Function

CEC_TX_RETRY_TIMEOUT_ST	Description
0 «	No change
1	CEC TX has tried but failed to resend the current message for the number of times specified by CEC_TX_RETRY

CEC_TX_RETRY[2:0], Addr 80 (CEC), *Address 0x12[6:4]*

The number of times the CEC TX should try to retransmit the message if an error condition is encountered. Per the CEC spec this value should not be set to a value greater than 5.

Function

CEC_TX_RETRY[2:0]	Description
001 «	Try to retransmit the message 1 time if an error occurs
xxx	Try to retransmit the message xxx times if an error occurs

CEC_TX_NACK_COUNTER[3:0], Addr 80 (CEC), *Address 0x14[3:0] (Read Only)*

The number of times that the NACK error condition was encountered while trying to send the current message. This register is reset to 0b0000 when CEC_TX_ENABLE is set to 1.

Function

CEC_TX_NACK_COUNTER[3:0]	Description
0000 «	No error condition
XXXX	The number of times the NACK error condition was encountered

CEC_TX_LOWDRIVE_COUNTER[3:0], Addr 80 (CEC), *Address 0x14[7:4] (Read Only)*

The number of times that the LOWDRIVE error condition was encountered while trying to send the current message. This register is reset to 0b0000 when CEC_TX_ENABLE is set to 1.

Function

CEC_TX_LOWDRIVE_COUNTER[3:0]	Description
0000 «	No error condition
XXXX	The number of times the LOWDRIVE error condition was encountered

13.3 CEC RECEIVE SECTION

The receive section features the hardware required for the CEC module to act as a follower. Once the CEC module is powered up via the CEC_POWER_UP bit the CEC Rx section will immediately being monitoring the CEC bus for messages with the correct logical address(es). When the message reception is completed the CEC receive section generates an interrupt (assuming the corresponding interrupt mask bits are set accordingly).

The host can disable message reception while keeping the CEC module powered up by using the FORCE_NACK bit to not acknowledge received messages.

CEC_FORCE_NACK, Addr 80 (CEC), *Address 0x27[1]*

Force NO-ACK Control
Setting this bit forces the CEC controller not acknowledge any received messages.

Function

CEC_FORCE_NACK	Description
0 «	Acknowledge received messages
1	Do not acknowledge received messages

13.3.1 Logical Address Configuration

The host must set the destination logical address(es) that the CEC receive section will respond to. Up to three logical addresses can be enabled allowing support for multi-function devices such as DVD recorders with TV tuners which require multiple logical addresses. The logical address(es) are set via the following registers:

- CEC_LOGICAL_ADDRESS2[3:0] if CEC_LOGICAL_ADDRESS_MASK[2] is set to 1
- CEC_LOGICAL_ADDRESS1[3:0] if CEC_LOGICAL_ADDRESS_MASK[1] is set to 1
- CEC_LOGICAL_ADDRESS0[3:0] if CEC_LOGICAL_ADDRESS_MASK[0] is set to 1

CEC_LOGICAL_ADDRESS2[3:0], Addr 80 (CEC), *Address 0x29[3:0]*

Logical address 2 - this address must be enabled by setting CEC_LOGICAL_ADDRESS_MASK[2] to 1

Function

CEC_LOGICAL_ADDRESS2[3:0]	Description
1111 «	Default value
xxxx	User specified logical address

CEC_LOGICAL_ADDRESS_MASK_2, Addr 80 (CEC), *Address 0x27[6]*

Mask bit for logical address 2

Function

CEC_LOGICAL_ADDRESS_MASK_2	Description
0 «	Logical address 2 disabled
1	Logical address 2 enabled

CEC_LOGICAL_ADDRESS1[3:0], Addr 80 (CEC), *Address 0x28[7:4]*

Logical address 1 - this address must be enabled by setting CEC_LOGICAL_ADDRESS_MASK[1] to 1

Function

CEC_LOGICAL_ADDRESS1[3:0]	Description
1111 «	Default value
xxxx	User specified logical address

CEC_LOGICAL_ADDRESS_MASK_1, Addr 80 (CEC), *Address 0x27[5]*

Mask bit for logical address 1

Function

CEC_LOGICAL_ADDRESS_MASK_1	Description
0 «	Logical address 1 disabled
1	Logical address 1 enabled

CEC_LOGICAL_ADDRESS0[3:0], Addr 80 (CEC), *Address 0x28[3:0]*

Logical address 0 - this address must be enabled by setting CEC_LOGICAL_ADDRESS_MASK[0] to 1

Function

CEC_LOGICAL_ADDRESS0[3:0]	Description
1111 «	Default value
xxxx	User specified logical address

CEC_LOGICAL_ADDRESS_MASK_0, Addr 80 (CEC), *Address 0x27[4]*

Mask bit for logical address 0

Function

CEC_LOGICAL_ADDRESS_MASK_0	Description
0	Logical address 0 disabled
1 «	Logical address 0 enabled

13.3.2 Receive Buffers

The ADV7842 features three frame buffers which allow the receiver to receive up to three messages before the host processor needs to read a message out. When three messages have been received no further message reception is possible until the host reads at least one message.

Note that for backwards compatibility with previous generation ADI CEC-enabled parts, only one frame buffer is enabled by default. In this default mode, after a message is received, the host processor must read the message out before any further message reception is possible. The decision to use one or three messages buffers is controlled by the CEC_USE_ALL_BUFS bit.

CEC_USE_ALL_BUFS, Addr 80 (CEC), *Address 0x77[0]*

Control to enable supplementary receiver frame buffers.

Function

CEC_USE_ALL_BUFS	Description
0 «	Use only buffer 0 to store CEC frames
1	Use all 3 buffers to stores the CEC frames

For each of the frame buffers there is a corresponding two-bit time stamp and a raw flag as described below.

CEC_BUF0_TIMESTAMP[1:0], Addr 80 (CEC), *Address 0x53[1:0] (Read Only)*

Time stamp for frame stored in receiver frame buffer 0. This can be used to determine which frame should be read next from the receiver frame buffers.

Function

CEC_BUF0_TIMESTAMP [1:0]	Description
00 «	Invalid timestamp, no frame is available in this frame buffer
01	Of the frames currently buffered, this frame was the first to be received
10	Of the frames currently buffered, this frame was the second to be received
11	Of the frames currently buffered, this frame was the third to be received

CEC_BUF1_TIMESTAMP[1:0], Addr 80 (CEC), *Address 0x53[3:2] (Read Only)*

Time stamp for frame stored in receiver frame buffer 1. This can be used to determine which frame should be read next from the receiver frame buffers.

Function

CEC_BUF1_TIMESTAMP [1:0]	Description
00 «	Invalid timestamp, no frame is available in this frame buffer
01	Of the frames currently buffered, this frame was the first to be received
10	Of the frames currently buffered, this frame was the second to be received
11	Of the frames currently buffered, this frame was the third to be received

CEC_BUF2_TIMESTAMP[1:0], Addr 80 (CEC), *Address 0x53[5:4] (Read Only)*

Time stamp for frame stored in receiver frame buffer 2. This can be used to determine which frame should be read next from the receiver frame buffers.

Function

CEC_BUF2_TIMESTAMP [1:0]	Description
00 «	Invalid timestamp, no frame is available in this frame buffer
01	Of the frames currently buffered, this frame was the first to be received
10	Of the frames currently buffered, this frame was the second to be received
11	Of the frames currently buffered, this frame was the third to be received

CEC_RX_RDY0_ST, Addr 40 (IO), *Address 0x93[3] (Read Only)*

Latched status of CEC_RX_RDY0_RAW signal. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit. When a message has been received into buffer 0 this bit is set. Once set this bit will remain high until the interrupt has been cleared via CEC_RX_RDY0_CLR.

Function

CEC_RX_RDY0_ST	Description
0 «	No change
1	New CEC message received in buffer 0

CEC_RX_RDY1_ST, Addr 40 (IO), *Address 0x93[4] (Read Only)*

Latched status of CEC_RX_RDY1_RAW signal. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit. When a message has been received into buffer 1 this bit is set. Once set this bit will remain high until the interrupt has been cleared via CEC_RX_RDY0_CLR.

Function

CEC_RX_RDY1_ST	Description
0 «	No change
1	New CEC message received in buffer 1

CEC_RX_RDY2_ST, Addr 40 (IO), *Address 0x93[5] (Read Only)*

Latched status of CEC_RX_RDY2_RAW signal. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit. When a message has been received into buffer 2 this bit is set. Once set this bit will remain high until the interrupt has been cleared via CEC_RX_RDY0_CLR.

Function

CEC_RX_RDY2_ST	Description
0 «	No change
1	New CEC message received in buffer 2

When a message (other than a polling message) is received it is loaded into the first available frame buffer (starting with buffer 0) and a 2-bit time stamp is generated for that buffer. If the corresponding interrupt mask bit is set the status bit relating to that buffer is set and an interrupt is generated to alert the host processor to the fact that a message has been received.

When all three frame buffers are full the receive module can no longer receive CEC messages and will not acknowledge any new messages (other than polling messages). In the case that only one frame buffer is enabled (the default condition) then only one message can be received. In this case the received message is always available in buffer 0.

The host can read the receive buffers (refer to Table 87, Table 88 and Table 89) to get the messages that were addressed to the CEC receiver. The length of each received message is available in the corresponding frame length register.

Table 87: CEC Incoming Frame Buffer 0 Registers

Register Name	CEC Map Address	Description
CEC_BUF0_RX_FRAME_HEADER[7:0]	0x15	Header of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA0[7:0]	0x16	Byte 0 of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA1[7:0]	0x17	Byte 1 of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA2[7:0]	0x18	Byte 2 of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA3[7:0]	0x19	Byte 3 of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA4[7:0]	0x1A	Byte 4 of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA5[7:0]	0x1B	Byte 5 of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA6[7:0]	0x1C	Byte 6 of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA7[7:0]	0x1D	Byte 7 of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA8[7:0]	0x1E	Byte 8 of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA9[7:0]	0x1F	Byte 9 of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA10[7:0]	0x20	Byte 10 of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA11[7:0]	0x21	Byte 11 of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA12[7:0]	0x22	Byte 12 of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA13[7:0]	0x23	Byte 13 of message in frame buffer 0
CEC_BUF0_RX_FRAME_DATA14[7:0]	0x24	Byte 14 of message in frame buffer 0

CEC_BUF0_RX_FRAME_LENGTH[4:0], Addr 80 (CEC), *Address 0x25[4:0] (Read Only)*

Function

CEC_BUF0_RX_FRAME_LENGTH[4:0]	Description
xxxxx	The total number of bytes (including header byte) that were received into buffer 0

CEC_CLR_RX_RDY0, Addr 80 (CEC), *Address 0x2C[1] (Self-Clearing)*

Clear control for CEC_RX_RDY0

Function

CEC_CLR_RX_RDY0	Description
0 «	Retain the value of the CEC_RX_RDY0 flag
1	Clear the value of the CEC_RX_RDY0 flag

Table 88: CEC Incoming Frame Buffer 1 Registers

Register Name	CEC Map Address	Description
CEC_BUF1_RX_FRAME_HEADER[7:0]	0x54	Header of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA0[7:0]	0x55	Byte 0 of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA1[7:0]	0x56	Byte 1 of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA2[7:0]	0x57	Byte 2 of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA3[7:0]	0x58	Byte 3 of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA4[7:0]	0x59	Byte 4 of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA5[7:0]	0x5A	Byte 5 of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA6[7:0]	0x5B	Byte 6 of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA7[7:0]	0x5C	Byte 7 of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA8[7:0]	0x5D	Byte 8 of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA9[7:0]	0x5E	Byte 9 of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA10[7:0]	0x5F	Byte 10 of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA11[7:0]	0x60	Byte 11 of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA12[7:0]	0x61	Byte 12 of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA13[7:0]	0x62	Byte 13 of message in frame buffer 1
CEC_BUF1_RX_FRAME_DATA14[7:0]	0x63	Byte 14 of message in frame buffer 1

CEC_BUF1_RX_FRAME_LENGTH[4:0], Addr 80 (CEC), *Address 0x64[4:0] (Read Only)*

Function

CEC_BUF1_RX_FRAME_LENGTH[4:0]	Description
xxxxx	The total number of bytes (including header byte) that were received into buffer 1

CEC_CLR_RX_RDY1, Addr 80 (CEC), *Address 0x2C[2] (Self-Clearing)*

Clear control for CEC_RX_RDY1

Function

CEC_CLR_RX_RDY1	Description
0 «	Retain the value of the CEC_ RX_RDY1 flag
1	Clear the value of the CEC_RX_RDY1 flag

Table 89: CEC Incoming Frame Buffer 2 Registers

Register Name	CEC Map Address	Description
CEC_BUF2_RX_FRAME_HEADER[7:0]	0x65	Header of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA0[7:0]	0x66	Byte 0 of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA1[7:0]	0x67	Byte 1 of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA2[7:0]	0x68	Byte 2 of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA3[7:0]	0x69	Byte 3 of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA4[7:0]	0x6A	Byte 4 of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA5[7:0]	0x6B	Byte 5 of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA6[7:0]	0x6C	Byte 6 of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA7[7:0]	0x6D	Byte 7 of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA8[7:0]	0x6E	Byte 8 of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA9[7:0]	0x6F	Byte 9 of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA10[7:0]	0x70	Byte 10 of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA11[7:0]	0x71	Byte 11 of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA12[7:0]	0x72	Byte 12 of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA13[7:0]	0x73	Byte 13 of message in frame buffer 2
CEC_BUF2_RX_FRAME_DATA14[7:0]	0x74	Byte 14 of message in frame buffer 2

CEC_BUF2_RX_FRAME_LENGTH[4:0], Addr 80 (CEC), *Address 0x75[4:0] (Read Only)*

Function

CEC_BUF2_RX_FRAME_ LENGTH[4:0]	Description
xxxxx	The total number of bytes (including header byte) that were received into buffer 2

CEC_CLR_RX_RDY2, Addr 80 (CEC), *Address 0x2C[3] (Self-Clearing)*

Clear control for CEC_RX_RDY2

Function

CEC_CLR_RX_RDY2	Description
0 «	Retain the value of the CEC_RX_RDY2 flag
1	Clear the value of the CEC_RX_RDY2 flag

13.3.3 CEC Message Reception Overview

This section describes how messages are received and stored when only one frame buffer is enabled (default condition).

1. Initially the receive buffer (buffer 0) is empty.

2. A message is received and stored in receive buffer 0, and CEC_BUF0_TIMESTAMP is set to 0b01. If the corresponding interrupt mask bit is set CEC_RX_RDY0_ST goes high and an interrupt is generated to alert the host processor that a message has been received. No more messages can be received until the processor reads out the received message.

3. The host processor responds to the interrupt, or polls the CEC_BUF0_TIMESTAMP register and realizes a message has been received, and reads receive buffer 0. Once the message is read the processor sets CEC_RX_RDY0_CLR which resets the buffer 0 timestamp to 0b00 and will also clear the buffer 0 status bit (if applicable). The CEC module is now ready to receive the next incoming message.

This section describes how messages are received and stored, how the time stamps are generated, and what happens when the host reads a received message when all three frame buffers are enabled.

1. Initially all buffers are empty and all time stamps are 0b00.

2. A message is received and stored in receive buffer 0, and CEC_BUF0_TIMESTAMP is set to 0b01. If the corresponding interrupt mask bit is set CEC_RX_RDY0_ST goes high and an interrupt is generated to alert the host processor that a message has been received.

3. Another message is received and stored in receive buffer 1, and CEC_BUF1_TIMESTAMP is set to 0b10. If the corresponding interrupt mask bit is set CEC_RX_RDY1_ST goes high and an interrupt is generated to alert the host processor that a message has been received.

4. The host processor responds to the interrupts, or polls the timestamps and realizes that messages have been received, and reads the three time stamps to determine which receive buffer to read first. The buffer with the earliest time stamp should be read first, so in this example the processor should read receive buffer 0 first. Once the message has been read the processor sets CEC_RX_RDY0_CLR which resets the buffer 0 timestamp to 0b00 and will also clear the buffer 0 status bit (if applicable).

5. Another message is received. The receiver module checks to see which of the three buffers are available, starting with buffer 0. In this example buffer 0 has been read out already by the host processor and is available so the new message is stored in receive buffer 0. At this time the timestamp for receive buffer 1 is adjusted to 0b01 to show that it contains the first received message, and a timestamp of 0b10 is assigned to receive buffer 0 to show that it contains the second received message. If the corresponding interrupt mask bit is set the CEC_RX_RDY0_ST bit goes high and an interrupt is generated to alert the host processor that a message has been received.

6. Another message is received. This message is stored in receive buffer 2 (buffer 0 and buffer 1 are full). Time stamp 0b11 is assigned to receive buffer 2 to show that it contains an unread message that was the third to be received. If the corresponding interrupt mask bit is set the CEC_RX_RDY2_ST bit goes high and an interrupt is generated to alert the host processor that a message has been received. At this time all receive buffers are full and no more messages can be received until the processor reads at least one message.

7. The host processor responds to the interrupts, or polls the timestamps and realizes that messages have been received, and reads the three time stamps. The buffer with the earliest time stamp should be read first, therefore receive buffer 1 is read first, followed by receive buffer 0 and then receive buffer 2. Once the messages are read the processor sets CEC_RX_RDY0_CLR, CEC_RX_RDY1_CLR and CEC_RX_RDY2_CLR. The time stamps for all three buffers are reset to 0b00.

13.4 ANTIGLITCH FILTER MODULE

This module is used to remove any glitches on the CEC bus in order to make the CEC input signal cleaner before it enters the CEC module. The glitch filter is programmable through the CEC_GLITCH_FILTER_CTRL register. The register value specifies the minimum pulse width that will be passed through by the module. Any pulses with narrower widths will be rejected. There is a CEC_GLITCH_FILTER_CTRL + 1 number of clock delays introduced by the antiglitch filter.

CEC_GLITCH_FILTER_CTRL[5:0], Addr 80 (CEC), *Address 0x2B[5:0]*

The CEC input signal is sampled by the input clock (XTAL clock). CEC_GLITCH_FILTER_CTRL specifies the minimum pulse width requirement in input clock cycles. Pulses of widths less than the minimum specified width are considered glitches and will be removed by the filter.

Function

CEC_GLITCH_FILTER_CTRL[5:0]	Description
000000	Disable the glitch filter
000001	Filter out pulses with width less than 1 clock cycle
000010	Filter out pulses with width less than 2 clock cycles
	...
000111 «	Filter out pulses with width less than 7 clock cycles
	...
111111	Filter out pulses with width less than 63 clock cycles

13.5 TYPICAL OPERATION FLOW

This section describes the algorithm that should be implemented in the host processor controlling the CEC module.

13.5.1 Initializing CEC Module

Figure 148 shows the flow that can be implemented in the host processor controlling the ADV7842 in order to initialize the CEC module.

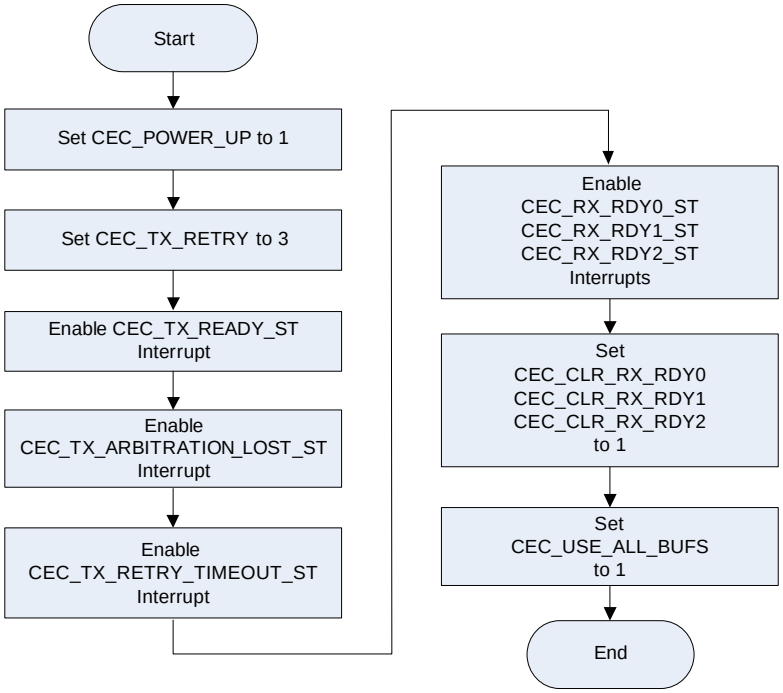


Figure 148: CEC Module Initialization

13.5.2 Using CEC Module as Initiator

Figure 149 shows the algorithm that can be implemented in the host processor controlling the ADV7842 in order to use the CEC module

as an initiator.

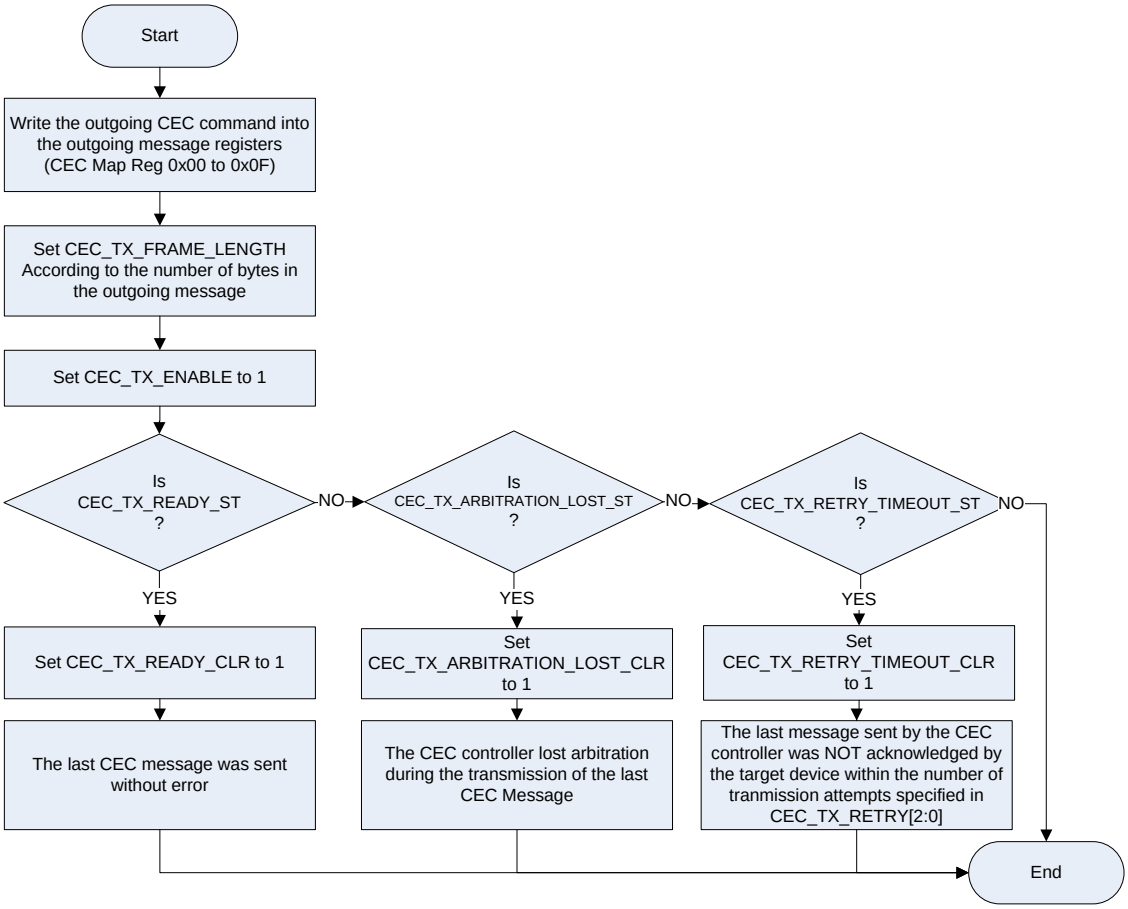


Figure 149: Using CEC Module as Initiator

13.5.3 Using CEC Module as Follower

Figure 150 shows the algorithm that can be implemented in the host processor controlling the ADV7842 in order to use the CEC module as a follower.

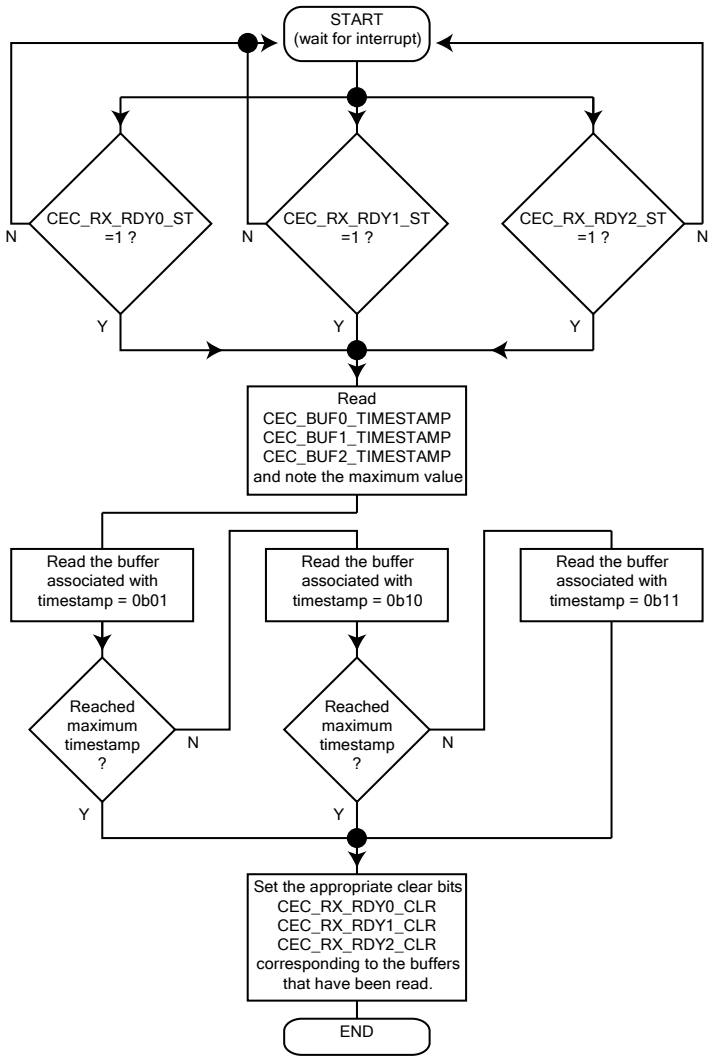


Figure 150: Using CEC Module as Follower

13.6 LOW POWER CEC MESSAGE MONITORING

The ADV7842 can be programmed to monitor the CEC line for messages which contain specific, user-programmable opcodes. These are referred to as "WAKE_OPCODEs" as they allow the system to go into a low power or sleep mode and be woken up when an opcode of interest is received, without the host processor having to check each received message.

The default values of the wake_opcode registers are detailed below. All of these registers can be overwritten as required by the host processor.

For each of the 8 WAKE_OPCODE registers there is a corresponding raw flag, a status bit and a clear bit. If one of the WAKE_OPCODEs is received the corresponding raw flag will go high for a brief period of time. If the appropriate interrupt mask bit is set the status bit will go high and remain high until cleared by the clear bit, and an interrupt will also be generated.

CEC_WAKE_OPCODE0[7:0], Addr 80 (CEC), *Address 0x78[7:0]*

CEC_WAKE_OPCODE0
This value can be set to a CEC opcode that requires a response. On receipt of this opcode the Rx generates an interrupt that can be used to alert the system that a CEC opcode of interest has been received and requires a response.

Function

CEC_WAKE_OPCODE0[7:0]	Description
01101101 «	POWER ON
xxxxxxxx	User specified OPCODE to respond to

CEC_WAKE_OPCODE1[7:0], Addr 80 (CEC), *Address 0x79[7:0]*

CEC_WAKE_OPCODE1
This value can be set to a CEC opcode that requires a response. On receipt of this opcode the Rx generates an interrupt that can be used to alert the system that a CEC opcode of interest has been received and requires a response.

Function

CEC_WAKE_OPCODE1[7:0]	Description
10001111 «	GIVE POWER STATUS
xxxxxxxx	User specified OPCODE to respond to

CEC_WAKE_OPCODE2[7:0], Addr 80 (CEC), *Address 0x7A[7:0]*

CEC_WAKE_OPCODE2
This value can be set to a CEC opcode that requires a response. On receipt of this opcode the Rx generates an interrupt that can be used to alert the system that a CEC opcode of interest has been received and requires a response.

Function

CEC_WAKE_OPCODE2[7:0]	Description
10000010 «	ACTIVE SOURCE
xxxxxxxx	User specified OPCODE to respond to

CEC_WAKE_OPCODE3[7:0], Addr 80 (CEC), *Address 0x7B[7:0]*

CEC_WAKE_OPCODE3
This value can be set to a CEC opcode that requires a response. On receipt of this opcode the Rx generates an interrupt that can be used to alert the system that a CEC opcode of interest has been received and requires a response.

Function

CEC_WAKE_OPCODE3[7:0]	Description
00000100 «	IMAGE VIEW ON
xxxxxxxx	User specified OPCODE to respond to

CEC_WAKE_OPCODE4[7:0], Addr 80 (CEC), *Address 0x7C[7:0]*

CEC_WAKE_OPCODE4
This value can be set to a CEC opcode that requires a response. On receipt of this opcode the Rx generates an interrupt that can be used to alert the system that a CEC opcode of interest has been received and requires a response.

Function

CEC_WAKE_OPCODE4[7:0]	Description
00001101 «	TEXT VIEW ON
xxxxxxxx	User specified OPCODE to respond to

CEC_WAKE_OPCODE5[7:0], Addr 80 (CEC), *Address 0x7D[7:0]*

CEC_WAKE_OPCODE5
This value can be set to a CEC opcode that requires a response. On receipt of this opcode the Rx generates an interrupt that can be used to alert the system that a CEC opcode of interest has been received and requires a response.

Function

CEC_WAKE_OPCODE5[7:0]	Description
01110000 «	SYSTEM AUDIO MODE REQUEST
xxxxxxxx	User specified OPCODE to respond to

CEC_WAKE_OPCODE6[7:0], Addr 80 (CEC), *Address 0x7E[7:0]*

CEC_WAKE_OPCODE6
This value can be set to a CEC opcode that requires a response. On receipt of this opcode the Rx generates an interrupt that can be used to alert the system that a CEC opcode of interest has been received and requires a response.

Function

CEC_WAKE_OPCODE6[7:0]	Description
01000010 «	DECK CONTROL
xxxxxxxx	User specified OPCODE to respond to

CEC_WAKE_OPCODE7[7:0], Addr 80 (CEC), *Address 0x7F[7:0]*

CEC_WAKE_OPCODE7
This value can be set to a CEC opcode that requires a response. On receipt of this opcode the Rx generates an interrupt that can be used to alert the system that a CEC opcode of interest has been received and requires a response.

Function

CEC_WAKE_OPCODE7[7:0]	Description
01000001 «	PLAY
xxxxxxxx	User specified OPCODE to respond to

14 AV.LINK BUS INTERFACE

The AV.link module features the hardware required to acts as an initiator or a follower as per the AV.link specification. When the AV.link controller is used, a host processor (e.g. an external processor controlling the ADV7842) is only required to initialize it, and then send or receive AV.link messages by accesses the message buffer via the AV.link map.

The AV.link module consists of four main sections:

- Transmit section AVLINK_TX
- Receive section AVLINK_RX
- Clock generator section AVLINK_CLK_GEN
- Anti-glitch filter section AVLINK_ANTI_GLITCH

The block diagram of the AV.link is shown in Figure 151.

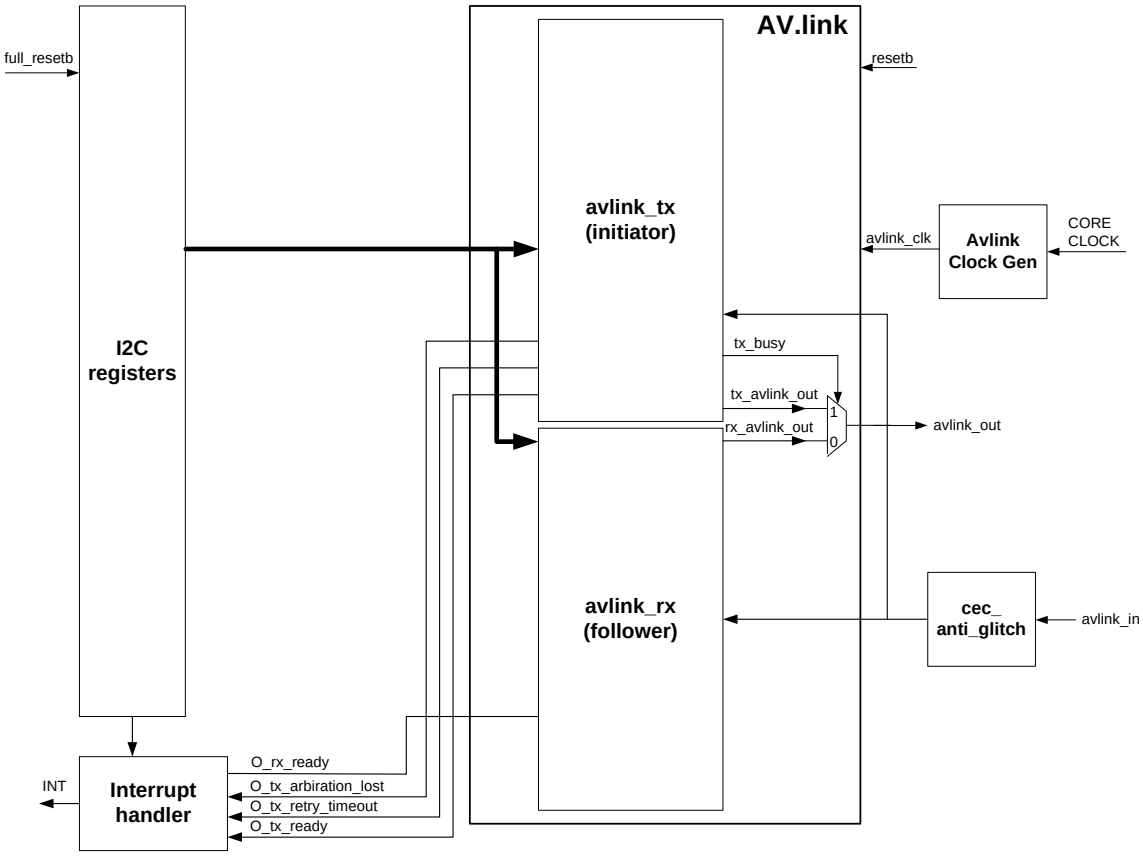


Figure 151: AV.link Module Block Diagram

14.1 MAIN CONTROLS

This section describes the main controls for the AV.link module.

AVL_AVLINK_POWER_UP, Addr 84 (AVLINK), *Address 0x27[0]*

Power Mode of AV.link module

Function

AVL_AVLINK_POWER_UP	Description
0 «	Power down the AV.link module
1	Power up the AV.link module

AVL_SOFT_RESET, Addr 84 (AVLINK), *Address 0x29[0] (Self-Clearing)*

AV.link module software reset.

Function

AVL_SOFT_RESET	Description
0 «	No function
1	Reset the AV.link module

Note that the AVL_AVLINK_POWER_UP bit can be used to set the ADV7842 to power-down mode 1 (refer to Section 3.2.3.2).

14.2 AV.LINK TRANSMIT SECTION

The transmit section features the hardware required for the AV.link module to act as an initiator. The host utilizes this section to transmit messages over the AV.link bus. When the host wants to send a message to other AV.link devices, it writes the message to the AV.link outgoing message registers (refer to Table 90) and other transmits related control registers such as the frame mode register and outgoing message length register. Then, the host then enables the transmission process by setting the AVL_TX_ENABLE bit to 1. When the message transmission is completed, or if an error occurs, the AV.link transmitter section generates an interrupt (assuming the corresponding interrupt mask bits are set accordingly).

Table 90: AV.link Outgoing Message Buffer Registers

Register Name	AVLINK Map Address	Description
AVL_TX_FRAME_HEADER[7:0]	0x01	Header of next outgoing message
AVL_TX_FRAME_DATA0[7:0]	0x02	Byte 0 of next outgoing message
AVL_TX_FRAME_DATA1[7:0]	0x03	Byte 1 of next outgoing message
AVL_TX_FRAME_DATA2[7:0]	0x04	Byte 2 of next outgoing message
AVL_TX_FRAME_DATA3[7:0]	0x05	Byte 3 of next outgoing message
AVL_TX_FRAME_DATA4[7:0]	0x06	Byte 4 of next outgoing message
AVL_TX_FRAME_DATA5[7:0]	0x07	Byte 5 of next outgoing message
AVL_TX_FRAME_DATA6[7:0]	0x08	Byte 6 of next outgoing message
AVL_TX_FRAME_DATA7[7:0]	0x09	Byte 7 of next outgoing message
AVL_TX_FRAME_DATA8[7:0]	0x0A	Byte 8 of next outgoing message
AVL_TX_FRAME_DATA9[7:0]	0x0B	Byte 9 of next outgoing message
AVL_TXFRAME_DATA10[7:0]	0x0C	Byte 10 of next outgoing message
AVL_TXFRAME_DATA11[7:0]	0x0D	Byte 11 of next outgoing message

AVL_TX_FRAME_MODE[1:0], Addr 84 (AVLINK), *Address 0x00[1:0]*

AV.link frame mode transmission.

Function

AVL_TX_FRAME_MODE[1:0]	Description
00 «	mode 1
01	mode 2
10	mode 3
11	Reserved

AVL_TX_FRAME_ECT, Addr 84 (AVLINK), *Address 0x0E[0]*

AV.link Extended Command Table (ECT) bit in the frame to be transmitted. The ECT bit is the ninth bit in the command block and is intended as an escape to an Extended Command Table for future use.

Function

AVL_TX_FRAME_ECT	Description
0 «	For future use
1	Specifies that the operand is specified in the present command table

AVL_TX_FRAME_LENGTH[7:0], Addr 84 (AVLINK), *Address 0x0F[7:0]*

Message size of the transmitted frame. This is the number of byte in the outgoing message including the header. Caters for a maximum frame size of 100 bits for mode 3.

Function

AVL_TX_FRAME_LENGTH[7:0]	Description
xxxxxxxx	Total number of bytes (including header byte) to be sent

AVL_TX_ENABLE, Addr 84 (AVLINK), *Address 0x10[0]*

This bit enables the TX section. When set to 1 it initiates the start of transmission of the message in the outgoing message buffer. When the message transmission is completed this bit is automatically reset to 0. If it is manually set to 0 during a message transmission it may terminate the transmission depending on what stage of the transmission process has been reached. If the message transmission is sill in the 'signal free time' stage the message transmission will be terminated. If data transmission has begun then the transmission will continue until the message is fully sent, or until an error condition occurs.

Function

AVL_TX_ENABLE	Description
0 «	Transmitter mode disabled
1	Transmitter mode enabled and message transmission started

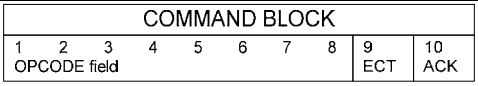


Figure 152: AV.link Command Block

The ADV7842 features a combination of raw flags and status bits related to the transmission of AV.link messages. The events that set these status bits are mutually exclusive – i.e. only one of the three events can occur during any given message transmission.

- AVLINK_TX_READY_RAW
- AVLINK_TX_ARBITRATION_LOST_ST
- AVLINK_TX_RETRY_TIMOUT_ST

AVLINK_TX_READY_RAW, Addr 40 (IO), *Address 0x4C[0] (Read Only)*

Raw status of AV.link Transmitter 'ready to send message' signal. This bit will be high whenever the TX is ready to send a message. This bit can be used as a "message sent" bit because it will return high when the current message has been sent.

Function

AVLINK_TX_READY_RAW	Description
0 «	AV.link TX is busy (not ready to send a message)
1	AV.link TX is ready to send a message

AVLINK_TX_ARBITRATION_LOST_ST, Addr 40 (IO), *Address 0x4D[1] (Read Only)*

Latched status of AVLINK_TX_ARBITRATION_LOST_RAW signal. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit. If the AV.link TX loses arbitration while trying to send a message this bit is set. Once set this bit will remain high until the interrupt has been cleared via AVLINK_TX_ARBITRATION_LOST_CLR.

Function

AVLINK_TX_ARBITRATION_LOST_ST	Description
0 «	No change
1	The AV.link TX lost arbitration to another TX

AVLINK_TX_RETRY_TIMEOUT_ST, Addr 40 (IO), *Address 0x4D[2] (Read Only)*

Latched status of AVLINK_TX_RETRY_TIMEOUT_RAW signal. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit. If the AV.link TX fails to send the current message within the number of retry attempts specified by AVL_TX_RETRY this bit is set. Once set this bit will remain high until the interrupt has been cleared via AVLINK_TX_RETRY_TIMEOUT_CLR.

Function

AVLINK_TX_RETRY_TIMEOUT_ST	Description
0 «	No change
1	AV.link TX has tried but failed to resend the current message for the number of times specified by AVL_TX_RETRY

AVL_TX_RETRY[2:0], Addr 84 (AVLINK), *Address 0x11[2:0]*

The number of times the AV.link TX should try to retransmit the message if an error condition is encountered.

Function

AVL_TX_RETRY[2:0]	Description
100 «	Try to retransmit the message 4 times if an error occurs
xxx	Try to retransmit the message xxx times if an error occurs

AVL_TX_NACK_COUNTER[3:0], Addr 84 (AVLINK), *Address 0x14[3:0] (Read Only)*

The number of times that the NACK error condition was encountered while trying to send the current message. This register is reset to 0b0000 when AVL_TX_ENABLE is set to 1.

Function

AVL_TX_NACK_COUNTER[3:0]	Description
0000 «	No error condition
xxxx	The number of times the NACK error condition was encountered

14.3 AV.LINK RECEIVER SECTION

The receive section features the hardware required for the AV.link module to act as a follower. Once the AV.link module is powered up via the AVL_AVLINK_POWER_UP bit and the Rx section is enabled by the AVL_RX_ENABLE bit the AV.link Rx section will immediately begin monitoring the AV.link bus for messages with the correct logical address(es). When the message reception is complete the AV.link receive section generates an interrupt (assuming the corresponding interrupt mask bits are set accordingly).

The host can disable message reception while keeping the AV.link module powered up by using the AVL_FORCE_NACK bit to not acknowledge received messages.

AVL_FORCE_NACK, Addr 84 (AVLINK), *Address 0x26[1]*

Force NO-ACK Control
Setting this bit forces the AV.link controller to not acknowledge any received messages.

Function

AVL_FORCE_NACK	Description
0 «	Acknowledge received messages
1	Do acknowledge received messages

14.3.1 Logical Address Configuration

The host must set the destination logical address(es) that the AV.link receive section will respond to. Up to three logical addresses can be enabled (applies to Mode 2 frames only). The logical addresses are set via the following registers:

- AVL_LOGICAL_ADDRESS0[3:0] if AVL_LOGICAL_ADDRESS_MASK[0] is set to 1
- AVL_LOGICAL_ADDRESS1[3:0] if AVL_LOGICAL_ADDRESS_MASK[1] is set to 1
- AVL_LOGICAL_ADDRESS2[3:0] if AVL_LOGICAL_ADDRESS_MASK[2] is set to 1

AVL_LOGICAL_ADDRESS2[3:0], Addr 84 (AVLINK), *Address 0x2B[3:0]*

Logical address 2 (mode 2). This address must be enabled by setting AVL_LOGICAL_ADDRESS_MASK[2] to 0b1

Function

AVL_LOGICAL_ADDRESS2[3:0]	Description
1111 «	Default value
xxxx	User specified logical address

AVL_LOGICAL_ADDRESS_MASK_2, Addr 84 (AVLINK), *Address 0x26[6]*

Mask bit for logical address 2

Function

AVL_LOGICAL_ADDRESS_MASK_2	Description
0 «	Logical address 2 disabled
1	Logical address 2 enabled

AVL_LOGICAL_ADDRESS1[3:0], Addr 84 (AVLINK), *Address 0x2A[7:4]*

Logical address 1 (mode 2). This address must be enabled by setting AVL_LOGICAL_ADDRESS_MASK[1] to 0b1

Function

AVL_LOGICAL_ADDRESS1[3:0]	Description
1111 «	Default value
xxxx	User specified logical address

AVL_LOGICAL_ADDRESS_MASK_1, Addr 84 (AVLINK), *Address 0x26[5]*

Mask bit for logical address 1

Function

AVL_LOGICAL_ADDRESS_MASK_1	Description
0 «	Logical address 1 disabled
1	Logical address 1 enabled

AVL_LOGICAL_ADDRESS0[3:0], Addr 84 (AVLINK), *Address 0x2A[3:0]*

Logical address 0 (mode 2). This address must be enabled by setting AVL_LOGICAL_ADDRESS_MASK[0] to 0b1

Function

AVL_LOGICAL_ADDRESS0[3:0]	Description
1111 «	Default value
xxxx	User specified logical address

AVL_LOGICAL_ADDRESS_MASK_0, Addr 84 (AVLINK), *Address 0x26[4]*

Mask bit for logical address 0

Function

AVL_LOGICAL_ADDRESS_MASK_0	Description
0	Logical address 0 disabled
1 «	Logical address 0 enabled

14.3.2 Message Reception

Once the logical addresses and logical address mask have been configured and the host is ready to accept messages, it informs the receive module of this fact by setting the AVL_RX_EN bit to 0b1. The AV.link receiver module then starts monitoring the AV.link bus and accepts correctly addressed incoming messages. When a message, other than a polling message, is completely received, the AV.link receiver section sets AVLINK_RX_READY_RAW high to indicate that the host can then read the received buffer to get the message that was addressed to the AV.link receiver. The AVL_RX_EN bit is reset to 0b0 at this time.

AVL_RX_ENABLE, Addr 84 (AVLINK), *Address 0x25[0]*

Receiver mode enable

Function

AVL_RX_ENABLE	Description
0 «	Receiver mode disabled
1	Receiver mode enabled

AVLINK_RX_READY_RAW, Addr 40 (IO), *Address 0x4C[3] (Read Only)*

Raw status of AV.link Receiver Ready signal.

Function

AVLINK_RX_READY_RAW	Description
0 «	No change
1	AV.link Rx has received a complete message which is ready to be read by the host.

Note: An interrupt can be enabled for AVLINK_RX_READY_RAW by setting the corresponding mask bit AVLINK_RX_READY_MB1 or AVLINK_RX_READY_MB2.

Table 91: AV.link Incoming Message Buffer Registers

Register Name	AVLINK Map Address	Description
AVL_RX_FRAME_HEADER[7:0]	0x16	Header of the last incoming message
AVL_RX_FRAME_DATA0[7:0]	0x17	Byte 0 of the last incoming message
AVL_RX_FRAME_DATA1[7:0]	0x18	Byte 1 of the last incoming message
AVL_RX_FRAME_DATA2[7:0]	0x19	Byte 2 of the last incoming message
AVL_RX_FRAME_DATA3[7:0]	0x1A	Byte 3 of the last incoming message
AVL_RX_FRAME_DATA4[7:0]	0x1B	Byte 4 of the last incoming message
AVL_RX_FRAME_DATA5[7:0]	0x1C	Byte 5 of the last incoming message
AVL_RX_FRAME_DATA6[7:0]	0x1D	Byte 6 of the last incoming message
AVL_RX_FRAME_DATA7[7:0]	0x1E	Byte 7 of the last incoming message
AVL_RX_FRAME_DATA8[7:0]	0x1F	Byte 8 of the last incoming message
AVL_RX_FRAME_DATA9[7:0]	0x20	Byte 9 of the last incoming message
AVL_RX_FRAME_DATA10[7:0]	0x21	Byte 10 of the last incoming message
AVL_RX_FRAME_DATA11[7:0]	0x22	Byte 11 of the last incoming message

AVL_RX_FRAME_MODE[1:0], Addr 84 (AVLINK), *Address 0x15[1:0] (Read Only)*

Frame mode of the received frame.

Function

AVL_RX_FRAME_MODE[1:0]	Description
00 «	mode 1
01	mode 2
10	mode 3
11	Reserved

AVL_RX_FRAME_ECT, Addr 84 (AVLINK), *Address 0x23[0] (Read Only)*

AV.link ECT bit in frame received

AVL_RX_FRAME_LENGTH[7:0], Addr 84 (AVLINK), *Address 0x24[7:0] (Read Only)*

Received message size (Number of data block + header)

14.4 AV.LINK ANTIGLITCH FILTER MODULE

This module is used to remove any glitches on the AV.link bus in order to make the AV.link input signal cleaner before it enters the AV.link module. The glitch filter is programmable through the AVL_GLITCH_FILTER_CTRL register. The register value specifies the minimum pulse width that will be passed through by the module. Any pulses with narrower widths will be rejected. There is a AVL_GLITCH_FILTER_CTRL + 1 number of clock delays introduced by the antiglitch filter.

AVL_GLITCH_FILTER_CTRL[5:0], Addr 84 (AVLINK), *Address 0x28[5:0]*

Glitch filter control for the AV.link input.
The AV.link input signal is sampled by the input clock (XTAL clock). AVL_GLITCH_FILTER_CTRL specifies the minimum pulse width requirement in input clock cycles. Pulses of widths less than the minimum specified width are considered glitches and will be removed by the filter.

Function

AVL_GLITCH_FILTER_CTRL[5:0]	Description
000000	Disable the glitch filter
000001	Filter out pulses with width less than 1 clock cycle
000010	Filter out pulses with width less than 2 clock cycles
	...
000111 «	Filter out pulses with width less than 7 clock cycles
	...
111111	Filter out pulses with width less than 63 clock cycles

14.5 FORMAT DESCRIPTION OF TRANSMITTED FRAMES

The host specifies the mode of the frame to be transmitted in the AVL_FRAME_MODE register, the header in the AVL_FRAME_HEADER register, and the data in the transmitted message payload registers (refer to Table 90). The bits that are transmitted for all three modes are described below.

Note that for all three modes, the transmission of the start sequence bits is handled by the AV.link controller itself, the host processor does not have to specify this separately.

Table 92: Start Sequence Table

Mode	Start Sequence (binary format)
1	1b0 (ESC)
2	2b10
3	3b110

14.5.1 Mode 1

The format of the mode 1 frame is shown in Figure 153. This mode 1 frame has no ACKs.

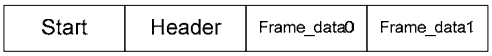


Figure 153: Mode 1 Frame Format

Table 93: Value Taken By Each Data Unit in Mode 1 Frame

Frame Type	**Value (Binary Format)**
Header	xxxbbbbb
Frame data 0	bbbbbbbb
Frame data 1	bbbbbbb

14.5.2 Mode 2

The format of the mode 2 frame is shown in Figure 154. Note that the AV.link controller automatically transmits 2b10 during the start sequence.

START SEQUENCE			HEADER				FRAME_DATA0		
Start	"1"	"0"	Source	Dest.	EOM	ACK	Opcode Field	ECT	ACK

Figure 154: Mode 2 Frame Format

Table 94: Value Taken by Each Data Unit in Mode 2 Frame

Frame Type	**Value (Binary Format)**
Header	bbbbbbbb
Frame data 0	bbbbbbbb

14.5.3 Mode 3

The format of the mode 3 frame is shown in Figure 155. Note that the AV.link module automatically transmits 3b110 during the start sequence.

START SEQUENCE		APPLICATION IDENTIFIER	APPLICATION			
Start	110	Header	Frame_data0	Frame_data1	. . .	Frame_data1

Figure 155: Mode 3 Frame Format

Table 95: Value Taken by Each Data Unit in Mode 3 Frame

Frame Type	**Value (Binary Format)**
Header	xxxxxbbb
Frame data 0	xxbbbbbb
Frame data 1	bbbbbbbb
Frame data 2	bbbbbbbb
Frame data 3	bbbbbbbb
Frame data 4	bbbbbbbb
Frame data 5	bbbbbbbb

Frame Type	Value (Binary Format)
Frame data 6	bbbbbbbb
Frame data 7	bbbbbbbb
Frame data 8	bbbbbbbb
Frame data 9	bbbbbbbb
Frame data 10	bbbbbbbb
Frame data 11	bbbbbbbb

The frame header stores the three bits of the application identifier. The frame data registers (refer to Table 90) store the data payload of the application block.

The AV.link specification mentions a maximum frame size of 100 bits. Excluding the three bits for the start sequence and the three bits for the Application ID, the remaining 94 bits of the application block are the frame data registers.

The host specifies the length of the frame in the AVL_TX_FRAME_LENGTH register. The frame length is defined as the number of frame data buffers that are required to be transmitted.

14.6 MODE DETECTION

The AV.link receiver module performs a mode detection of the incoming frame bits. After the Start bit is encountered, the receiver enters the receive state where the incoming bits are checked for the start sequence. The decoding starts with an invalid state 2b11 and, by the end of reception of the third bit, the decoding of the mode is expected to be over. Otherwise, the receiver returns to the invalid state and the message is not received.

```
mode=INVALID
if(bit_counter==0 andand sampled_avlink_in==0)
{
                          mode=1;
                          start_sequence_error=0;
}
else if(bit_counter==1 andand sampled_avlink_in==0)
{
                          mode=2;
                          start_sequence_error=0;
}
else if(bit_counter==2 andand sampled_avlink_in==0)
{
                          mode=3;
                          start_sequence_error=0;
}
else if(bit_counter>=3 andand mode==INVALID)
{
                          mode=INVALID;
                          start_sequence_error=1;
}
```

Figure 156: Pseudo C Code for Mode Detection

14.7 ESC/DIR BIT VALIDATION

The receiver performs optional ESC/DIR bit validation for the mode 0 header. This option is set by setting the AVL_MODE00_HEADER_VALIDATE bit in the memory. The validation is summarized in Figure 157.

```
* Check if "ONE" of /PAS and /NAS is low.
* Check if /DES is low.
```

```
* Check if DIR = (/PAS + /NAS) * /DES
```

Figure 157: Pseudo C Code for ESC/DIR Bit Validation

AVL_MODE00_HEADER_VALIDATE, Addr 84 (AVLINK), *Address 0x52[0]*

Enable the validation of header for mode 0 frames

Function

AVL_MODE00_HEADER_VALIDATE	Description
0 «	Disable validation of the header for mode 0 frames
1	Enable validation of the header for mode 0 frames

15 REGISTER ACCESS AND SERIAL PORTS DESCRIPTION

The ADV7842 has seven 2-wire serial (I²C compatible) ports:

- One main I²C port, SDA/SCL, which allows a system I²C master controller to control and configure the ADV7842
- Two DDC I²C ports for port A and port B which allow an HDMI host to access the internal E-EDID and the HDCP registers
- One fast I²C port that allows extracted VBI data to be read from the VDP block
- One I²C port that allows VGA E-EDID data to be read

15.1 MAIN I²C PORT

15.1.1 Register Access

The ADV7842 has twelve 256-byte maps that can be accessed via the main I²C ports, SDA and SCL. Each map has it own I²C address and acts as a standard slave device on the I²C bus.

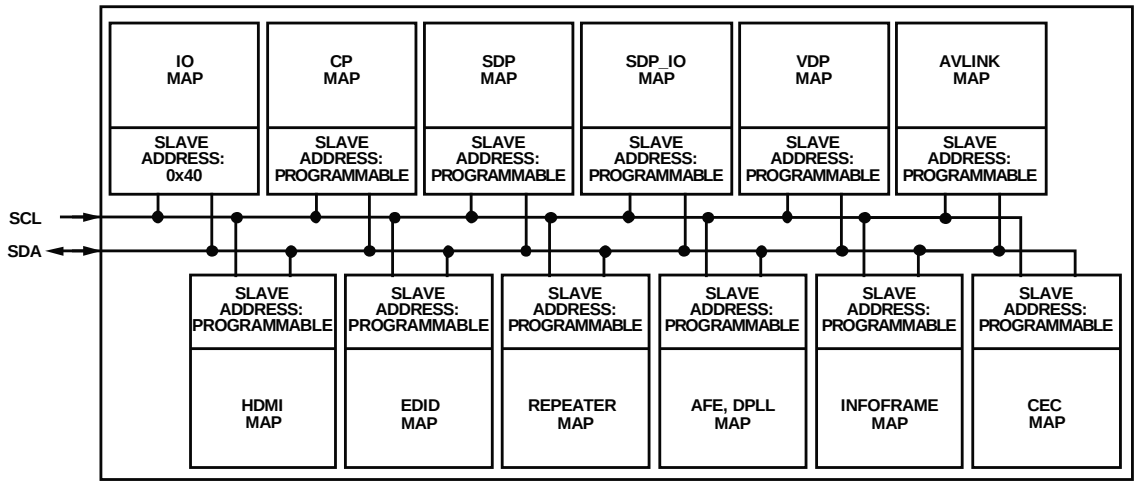


Figure 158: ADV7842 Register Map Access through Main I²C Port

Eleven out of the twelve maps have a programmable I²C address. This facilitates the integration of the ADV7842 in systems that have multiple slaves on the general I²C bus. The IO Map, which is non programmable, is accessible on initial powerup and following a reset. In order to access any other map, an appropriate I²C should be assigned using the registers in Table 96.

Table 96: Register Maps and I²C Addresses

Map	Default Address	Programmable Address	Location at which Address can be Programmed
IO Map	0x40	Not programmable	Not applicable
SDP_IO Map	0x00[9]	Programmable	IO Map register 0xF2
SDP Map	0x00	Programmable	IO Map register 0xF1
CP Map	0x00	Programmable	IO Map register 0xFD
VDP map	0x00	Programmable	IO Map register 0xFE
AFE Map	0x00	Programmable	IO Map register 0xF8
HDMI Map	0x00	Programmable	IO Map register 0xFB
Repeater Map	0x00	Programmable	IO Map register 0xF9
EDID Map	0x00	Programmable	IO Map register 0xFA
InfoFrame Map	0x00	Programmable	IO Map register 0xF5
CEC Map	0x00	Programmable	IO Map register 0xF4
AVLINK Map	0x00	Programmable	IO Map register 0xF3

15.1.2 Protocol for Main I²C Port

The system controller initiates a data transfer by establishing a start condition, defined by a high to low transition on SDA while SCL remains high. This transition indicates that an address/data stream will follow. All peripherals respond to the start condition and shift the next eight bits (7-bit address and R/W bit). The bits are transferred from MSB down to LSB. The peripheral that recognizes the transmitted address responds by pulling the data line low during the ninth clock pulse. This is known as an acknowledge bit. All other devices withdraw from the bus at this point and maintain an idle condition.

In the idle condition, the device monitors the SDA and SCLK lines for the start condition and the correct transmitted address. The R/W bit determines the direction of the data. A logic 0 on the LSB of the first byte means that the master will write information to the peripheral. A logic 1 on the LSB of the first byte means that the master will read information from the peripheral.

Each of the ADV7842 maps acts as a standard slave device on the bus. The data on the SDA pin is eight bits long, supporting the 7-bit addresses plus the R/W bit. It interprets the first byte as the map address and the second byte as the starting subaddress. The subaddresses auto increment, allowing data to be written to or read from the starting subaddress. A data transfer is always terminated by a stop condition. The user can also access any unique subaddress register on a one-by-one basis without having to update all the registers.

Stop and start conditions can be detected at any stage during the data transfer. If these conditions are asserted out of sequence with normal read and write operations, these cause an immediate jump to the idle condition. During a given SCLK high period the user should issue only one start condition, one stop condition, or a single stop condition followed by a single start condition. If an invalid subaddress is issued by the user, the ADV7842 does not issue an acknowledge and returns to the idle condition.

If the user exceeds the highest subaddress in auto increment mode, the following actions are taken:

- In read mode, the highest subaddress register contents continue to be output until the master device issues a no acknowledge. This indicates the end of a read. A no acknowledge condition is where the SDA line is not pulled low on the ninth pulse.
- In write mode, the data for the invalid byte is not loaded into any subaddress register. A no acknowledge is issued by the ADV7842 and the part returns to the idle condition.

9

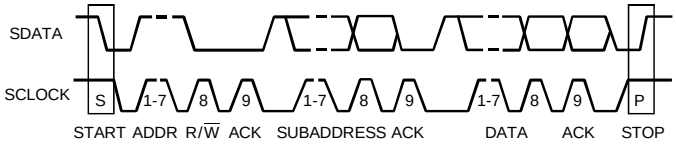


Figure 159: Bus Data Transfer

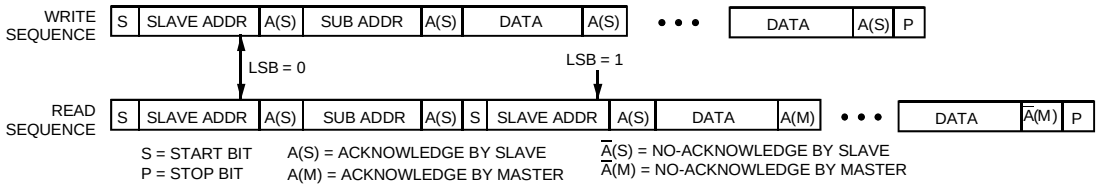


Figure 160: Read and Write Sequence

15.2 DDC PORTS

Two I²C ports, DDC port A and port B allow an HDMI host to access the internal E-EDID and the HDCP registers. Note that the DDC ports are 5 V tolerant, which simplifies the hardware between the HDMI connector and the ADV7842.

15.2.1 I²C Protocols for Access to the Internal E-EDID

An I²C master connected on a DDC port can access the internal E-EDID using the following protocol:

- Write sequence, as defined in Section 15.1.2
- Read sequence, as defined in Section 15.1.2
- Current address read sequence:
  Allows the master on the DDC port to read access internal E-EDID without specifying the subaddress that must be read. The ADV7842 stores an address counter for each DDC port that maintains the value of the subaddress that was last accessed. The address counter is incremented by one every time a read or a write access is requested on the DDC port.

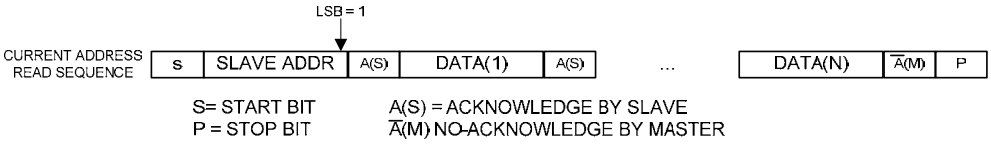


Figure 76: Current Address Read Sequence

15.2.2 I²C Protocols for Access to HDCP Registers

An I²C master connected on a DDC port can access the internal HDCP controller using the following protocol:

- Write sequence, as defined in Section 15.1.2
- Read sequence, as defined in Section 15.1.2
- Short read format, as defined in the High-bandwidth Digital Content Protection (HDCP) System Specifications

15.2.3 DDC Port A

The DDC lines of the HDMI port A comprise the DDCA_SCL and DDCA_SDA pins. An HDMI host connected to the DDC port A accesses the internal E-EDID at address 0xA0 in read only mode, and the HDCP registers at address 0x74 in read/write mode (refer to Figure 161). The internal E-EDID for port A is described in Section 8.9

Refer to the High-bandwidth Digital Content Protection (HDCP) System Specifications for detailed information on the HDCP registers.

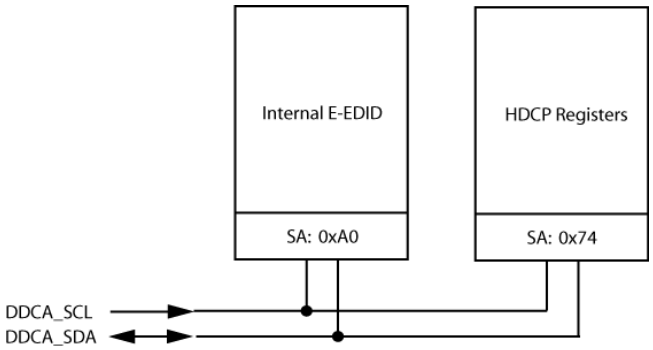


Figure 161: Internal E-EDID and HDCP Registers Access from Port A (SA = Slave Address)

15.2.4 DDC Port B

The DDC lines of the HDMI port B comprise the DDCB_SCL and DDCB_SDA pins. An HDMI host connected to the DDC port B accesses the internal E-EDID at address 0xA0 in read only mode, and the HDCP registers at address 0x74 in read/write mode (refer to Figure 162). The internal E-EDID for port B is described in Section 8.10.

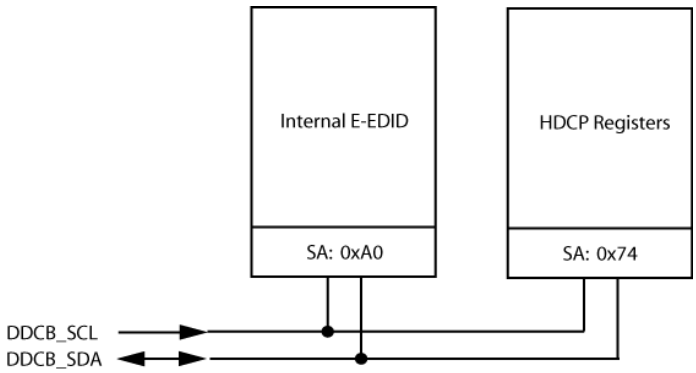


Figure 162: Internal E-EDID and HDCP Registers Access from Port B

Refer to the High-bandwidth Digital Content Protection (HDCP) System Specifications for detailed information on the HDCP registers.

16 INTERRUPTS

16.1 INTERRUPT ARCHITECTURE OVERVIEW

The ADV7842 interrupt architecture provides four different types of bits, namely

- Raw bits
- Status bits
- Interrupt mask bits
- Clear bits

Raw bits are defined as being either edge-sensitive or level-sensitive. The following example compares AVI_INFO_RAW and NEW_AVI_INFO_RAW to demonstrate the difference.

AVI_INFO_RAW, Addr 40 (IO), *Address 0x60[0] (Read Only)*

Raw status of AVI InfoFrame detected signal. This bit is set to one when an AVI InfoFrame is received and is reset to zero if no AVI InfoFrame is received for more than 7 VSyncs (on the eighth VSync leading edge following the last received AVI InfoFrame), after an HDMI packet detection reset or upon writing to AVI_PACKET_ID.

Function

AVI_INFO_RAW	Description
0 «	No AVI InfoFrame has been received within the last seven VSyncs or since the last HDMI packet detection reset
1	An AVI InfoFrame has been received within the last seven VSyncs

NEW_AVI_INFO_RAW, Addr 40 (IO), *Address 0x79[0] (Read Only)*

Status of the New AVI Infoframe interrupt signal. When set to 1 it indicates that an AVI Infoframe has been received with new contents. Once set this bit will remain high until the interrupt has been cleared via NEW_AVI_INFO_CLR.

Function

NEW_AVI_INFO_RAW	Description
0 «	No new AVI InfoFrame received
1	AVI InfoFrame with new content received

In the case of AVI_INFO_RAW this bit always represents the current status of whether or not the part is receiving AVI InfoFrames. It is not a latched bit and never requires to be cleared. This is the definition of a level-sensitive raw bit.

In the case of NEW_AVI_INFO_RAW the same strategy would not work. If the NEW_AVI_INFO_RAW bit were to behave in the same way as AVI_INFO_RAW it would go high at the instant the new InfoFrame was received, and would go low again some clock cycles afterwards. This is because a new InfoFrame is only new the instant it is received, and once received it is no longer new, so the event to set this bit only last for an instant and is then gone.

Having a raw bit that is only held high for an instant is not useful. Therefore, for these types of events, the raw bit is latched, and must be cleared by the corresponding clear bit. Accordingly, the raw bit does not truly represent the current status; instead it represents the status of an edge event that happened in the past. This is the definition of an edge-sensitive raw bit.

All raw bits, with the exceptions of INTRQ_RAW and INTRQ2_RAW, have corresponding status bits. The status bits always work in the

same manner whether the raw bit is edge or level sensitive. Status bits have the following characteristics

- A status bit must be enabled by setting either or both of the corresponding interrupt mask bits
- Status bits are always latched, and must be cleared by the corresponding clear bit.

When either of the interrupt mask bits for a given interrupt is set, if that raw bit changes state the corresponding status bit goes high and an interrupt is generated on the INT1 or INT2 pin, depending on which interrupt mask bit was set. The status bit must be cleared using the appropriate clear bit. The status bits, interrupt mask bits and clear bits for AVI_INFO and NEW_AVI_INFO are described below for completeness.

AVI_INFO_ST, Addr 40 (IO), *Address 0x61[0] (Read Only)*

Latched status of AVI_INFO_RAW signal. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit. Once set this bit will remain high until the interrupt has been cleared via AVI_INFO_CLR.

Function

AVI_INFO_ST	Description
0 «	AVI_INFO_RAW has not changed state
1	AVI_INFO_RAW has changed state

NEW_AVI_INFO_ST, Addr 40 (IO), *Address 0x7A[0] (Read Only)*

Latched status for the NEW_AVI_INFO_RAW. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit. Once set this bit will remain high until the interrupt has been cleared via NEW_AVI_INFO_CLR.

Function

NEW_AVI_INFO_ST	Description
0 «	NEW_AVI_INFO_RAW has not changed state
1	NEW_AVI_INFO_RAW has changed state

AVI_INFO_CLR, Addr 40 (IO), *Address 0x62[0] (Self-Clearing)*

Clear bit for AVI_INFO_RAW and AVI_INFO_ST bits.

Function

AVI_INFO_CLR	Description
0 «	No function
1	Clear AVI_INFO_RAW and AVI_INFO_ST

NEW_AVI_INFO_CLR, Addr 40 (IO), *Address 0x7B[0] (Self-Clearing)*

Clear bit for NEW_AVI_INFO_RAW and NEW_AVI_INFO_ST bits.

Function

NEW_AVI_INFO_CLR	Description
0 «	No function
1	Clear NEW_AVI_INFO_RAW and NEW_AVI_INFO_ST

AVI_INFO_MB1, Addr 40 (IO), *Address 0x64[0]*

INT1 interrupt mask for AVI Infoframe detection interrupt. When set an AVI Infoframe detection event will cause AVI_INFO_ST to be set and an interrupt will be generated on INT1.

Function

AVI_INFO_MB1	Description
0 «	Disables AVI Info frame detection interrupt for INT1
1	Enables AVI Info frame detection interrupt for INT1

AVI_INFO_MB2, Addr 40 (IO), *Address 0x63[0]*

INT2 interrupt mask for AVI Infoframe detection interrupt. When set an AVI Infoframe detection event will cause AVI_INFO_ST to be set and an interrupt will be generated on INT2.

Function

AVI_INFO_MB2	Description
0 «	Disables AVI Info frame detection interrupt for INT2
1	Enables AVI Info frame detection interrupt for INT2

NEW_AVI_INFO_MB1, Addr 40 (IO), *Address 0x7D[0]*

INT1 interrupt mask for New AVI Infoframe detection interrupt. When set a new AVI InfoFrame detection event will cause NEW_AVI_INFO_ST to be set and an interrupt will be generated on INT1.

Function

NEW_AVI_INFO_MB1	Description
0 «	Disable new AVI Infoframe interrupt for INT1
1	Enable new AVI Infoframe interrupt for INT1

NEW_AVI_INFO_MB2, Addr 40 (IO), *Address 0x7C[0]*

INT2 interrupt mask for New AVI Infoframe detection interrupt. When set a new AVI InfoFrame detection event will cause NEW_AVI_INFO_ST to be set and an interrupt will be generated on INT2.

Function

NEW_AVI_INFO_MB2	Description
0 «	Disables New SPD Infoframe interrupt for INT2
1	Enables New SPD Infoframe interrupt for INT2

The following diagrams and example timing figures show a graphical example of what has been described above.

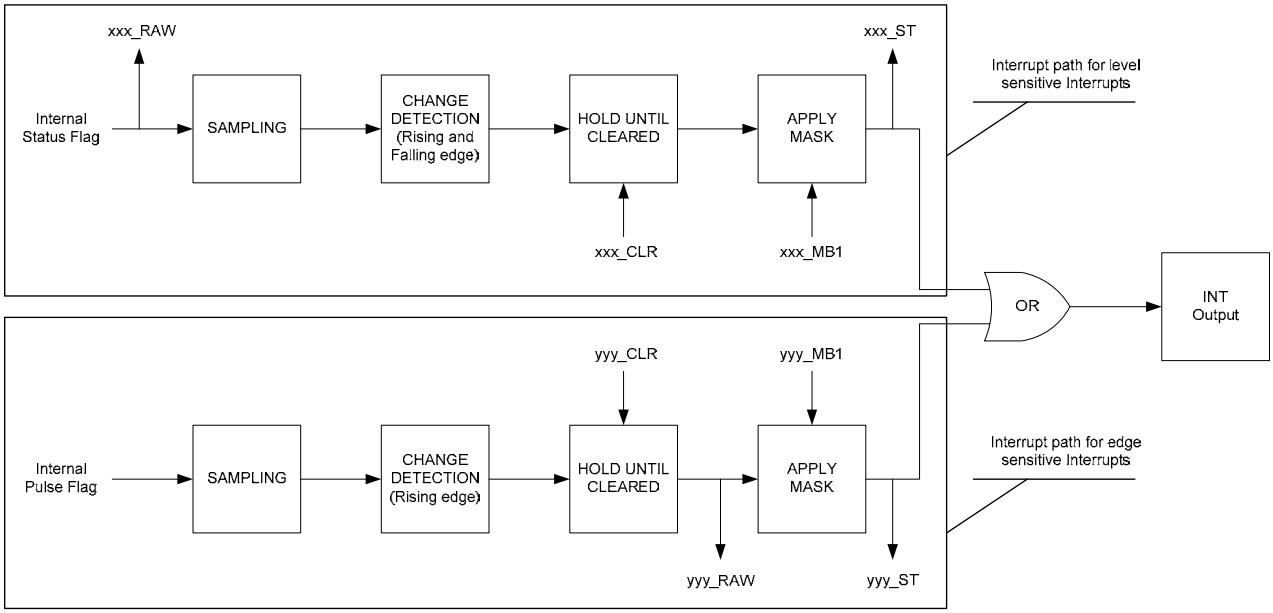


Figure 163: Level and Edge Sensitive Raw, Status and Interrupt Generation

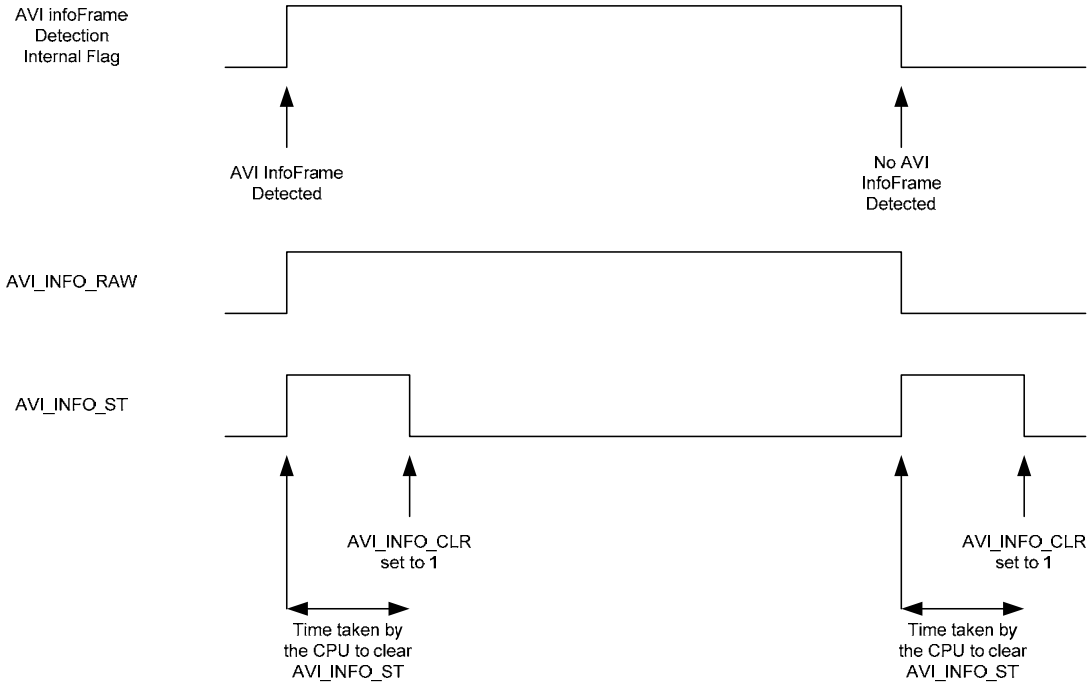


Figure 164: AVI_INFO_RAW and AVI_INFO_ST Timing

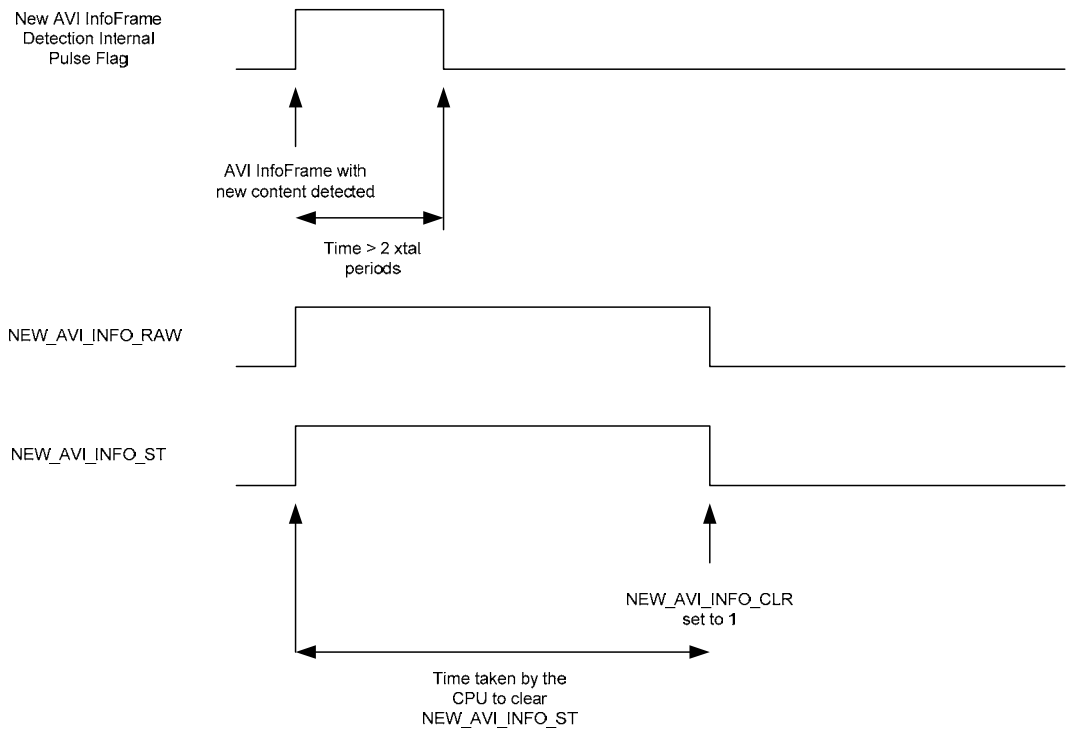


Figure 165: NEW_AVI_INFO_RAW and NEW_AVI_INFO_ST Timing

In this chapter all raw bits will be classified as being triggered by either level sensitive or edge sensitive events, with the following understanding of the terminology

Level sensitive events are events that are generally either high or low and which are not expected to change rapidly. The raw bit for level sensitive events is not latched and therefore always represents the true real-time status of the event in question.

Edge sensitive events are events that only exist for an instant. The raw bits for edge sensitive events are latched and therefore represent the occurrence of an edge sensitive event that happened in the past. Raw bits for edge sensitive events must be cleared by the corresponding clear bit.

16.2 INTERRUPT PINS

The ADV7842 features two dedicated interrupt pins, INT1 and INT2. INT1 is always enabled, but INT2 is disabled by default and must be enabled using the following I2C control.

INT2_EN, Addr 40 (IO), *Address 0x41[2]*

A control to enable INT2.

Function

INT2_EN	Description
0 «	Disable INT2
1	Enable INT2

16.2.1 Interrupt Duration

The interrupt duration can be programmed independently for INT1 and INT2. When an interrupt event occurs, the interrupt pin INT1 or INT2 becomes active with a programmable duration as described below.

INTRQ_DUR_SEL[1:0], Addr 40 (IO), *Address 0x40[7:6]*

A control to select the interrupt signal duration for the interrupt signal on INT1

Function

INTRQ_DUR_SEL[1:0]	Description
00 «	4 Xtal periods
01	16 Xtal periods
10	64 Xtal periods
11	Active until cleared

INTRQ2_DUR_SEL[1:0], Addr 40 (IO), *Address 0x41[7:6]*

A control to select the interrupt signal duration for the interrupt signal on INT2

Function

INTRQ2_DUR_SEL[1:0]	Description
00 «	4 Xtal periods
01	16 Xtal periods
10	64 Xtal periods
11	Active until cleared

16.2.2 Interrupt Drive Level

The drive level of INT1 and INT2 can be programmed independently for INT1 and INT2 as described below.

INTRQ_OP_SEL[1:0], Addr 40 (IO), *Address 0x40[1:0]*

Interrupt signal configuration control for INT1

Function

INTRQ_OP_SEL[1:0]	Description
00 «	Open drain
01	Drives low when active
10	Drives high when active
11	Disabled

INTRQ2_OP_SEL[1:0], Addr 40 (IO), *Address 0x41[1:0]*

Interrupt signal configuration control for INT2

Function

INTRQ2_OP_SEL[1:0]	Description
00 «	Open drain
01	Drives low when active
10	Drives high when active
11	Disabled

16.2.3 Interrupt Manual Assertion

It is possible to manually generate an interrupt on the INT1 and INT2 pins by setting MPU_STIM_INTRQ. This feature is designed for debug use and not intended for use in normal operation. The appropriate mask bit must be set to generate an interrupt at the pin.

MPU_STIM_INTRQ, Addr 40 (IO), *Address 0x40[2]*

Manual interrupt set control. This feature should be used for test purposes only. Note that the appropriate mask bit must be set to generate an interrupt at the pin

Function

MPU_STIM_INTRQ	Description
0 «	Disables manual interrupt mode
1	Enables manual interrupt mode

MPU_STIM_INTRQ_MB1, Addr 40 (IO), *Address 0x4B[7]*

INT1 interrupt mask for Manual forced interrupt signal. When set the Manual Forced interrupt will trigger the INT1 interrupt and MPU_STIM_INTRQ_ST will indicate the interrupt status.

Function

MPU_STIM_INTRQ_MB1	Description
0 «	Disables Manual forced interrupt for INT1
1	Enables Manual forced interrupt for INT1

MPU_STIM_INTRQ_MB2, Addr 40 (IO), *Address 0x4A[7]*

INT2 interrupt mask for Manual forced interrupt signal. When set the Manual Forced interrupt will trigger the INT2 interrupt and MPU_STIM_INTRQ_ST will indicate the interrupt status.

Function

MPU_STIM_INTRQ_MB2	Description
0 «	Disables Manual forced interrupt for INT2
1	Enables Manual forced interrupt for INT2

16.2.4 Multiple Interrupt Events

If an interrupt event occurs, and then a second interrupt event occurs before the system controller has cleared or masked the first interrupt event, the ADV7842 does not generate a second interrupt signal. The system controller should check all unmasked interrupt status bits as more than one may be active.

16.3 DESCRIPTION OF INTERRUPT BITS

This section lists all the raw bits in the IO map of the ADV7842 by category, and states whether the bit is an edge or level sensitive bit. A basic explanation for each bit is provided in the software manual and/or in the corresponding section of the hardware manual. For certain interrupts that require additional explanations, these are provided in the following sections of this chapter.

16.3.1 General Operation

- INTRQ_RAW (level sensitive event)
- INTRQ2_RAW (level sensitive event)

- AFE_INTERRUPT_RAW (edge sensitive event)
- MPU_STIM_INTRQ_RAW (edge sensitive event)

16.3.2 Analog/HDMI Video Mode

- SDP_STD_CHANGED_RAW (edge sensitive event)
- SDP_BURST_LOCKED_RAW (level sensitive event) [1]
- SDP_VIDEO_DETECTED_RAW (level sensitive event) [1]
- SDP_PROGRESSIVE_RAW (level sensitive event)
- SDP_SD_DET_RAW (level sensitive event) [1]
- SDP_PR_DET_RAW (level sensitive event) [1]
- SDP_50H_DET_RAW (level sensitive event)

- SSPD_RSLT_CHNGD_RAW (edge sensitive event)
- SSPD_RSLT_CHNGD_CH1_RAW (edge sensitive event)
- SSPD_RSLT_CHNGD_CH2_RAW (edge sensitive event)
- STDI_DATA_VALID_RAW (edge/level sensitive event .. programmable)
- STDI_DVALID_CH1_RAW (edge/level sensitive event .. programmable) [1]
- STDI_DVALID_CH2_RAW (edge/level sensitive event .. programmable) [1]

- CP_UNLOCK_RAW (edge/level sensitive event .. programmable) [1]
- CP_UNLOCK_CH1_RAW (edge/level sensitive event .. programmable) [1]
- CP_UNLOCK_CH2_RAW (edge/level sensitive event .. programmable) [1]
- CP_LOCK_RAW (edge/level sensitive event .. programmable) [1]
- CP_LOCK_CH1_RAW (edge/level sensitive event .. programmable) [1]
- CP_LOCK_CH2_RAW (edge/level sensitive event .. programmable) [1]

[1]Edge sensitive event on ADV7842 ES1

16.3.3 Macrovision Detection

The following raw bits are all related to Macrovision detection and are based on level sensitive events; it is therefore not necessary to clear these bits.

- MV_AGC_DET_RAW
- MV_PS_DET_RAW
- MV_CS_DET_RAW

16.3.4 VDP Operation

The following raw bits are all related to analog video VDP operation and are based on edge sensitive events; it is therefore necessary to clear these bits using the corresponding clear bit.

- CP_CGMS_CHNGD_RAW
- TTXT_AVL_RAW
- VITC_AVL_RAW
- GS_DATA_TYPE_RAW

- GS_PDC_VPS_UTC_AVL_RAW
- CGMS_WSS_AVL_RAW
- CCAP_EVEN_FIELD_RAW
- CCAP_AVL_RAW
- FASTI2C_DATA_RDY_RAW

16.3.5 CEC/AV.link

The following raw bits are all related to CEC and AV.link operation and are all edge sensitive events; it is therefore necessary to clear these bits.

- CEC_RX_RDY2_RAW
- CEC_RX_RDY1_RAW
- CEC_RX_RDY0_RAW
- CEC_TX_RETRY_TIMEOUT_RAW
- CEC_TX_ARBITRATION_LOST_RAW
- CEC_TX_READY_RAW
- CEC_INTERRUPT_BYTE[7:0]
- AVLINK_RX_READY_RAW
- AVLINK_TX_RETRY_TIMEOUT_RAW
- AVLINK_TX_ARBITRATION_LOST_RAW
- AVLINK_TX_READY_RAW

16.3.6 HDMI Only Mode

The following raw bits are all related to HDMI operation and are based on level sensitive events; it is therefore not necessary to clear these bits.

- ISRC2_PCKT_RAW
- ISRC1_PCKT_RAW
- ACP_PCKT_RAW
- VS_INFO_RAW
- MS_INFO_RAW
- SPD_INFO_RAW
- AUDIO_INFO_RAW
- AVI_INFO_RAW
- CS_DATA_VALID_RAW
- INTERNAL_MUTE_RAW
- AV_MUTE_RAW
- AUDIO_CH_MD_RAW
- HDMI_MODE_RAW
- GEN_CTL_PCKT_RAW
- AUDIO_C_PCKT_RAW
- GAMUT_MDATA_RAW
- TMDSPLL_LCK_A_RAW
- TMDSPLL_LCK_B_RAW

- TMDS_CLK_A_RAW
- TMDS_CLK_B_RAW
- HDMI_ENCRPT_A_RAW
- HDMI_ENCRPT_B_RAW
- CABLE_DET_A_RAW
- CABLE_DET_B_RAW
- V_LOCKED_RAW
- DE_REGEN_LCK_RAW

The following raw bits are all related to HDMI operation and are based on edge sensitive events; it is therefore necessary to clear these bits using the corresponding clear bit.

- NEW_ISRC2_PCKT_RAW
- NEW_ISRC1_PCKT_RAW
- NEW_ACP_PCKT_RAW
- NEW_VS_INFO_RAW
- NEW_MS_INFO_RAW
- NEW_SPD_INFO_RAW
- NEW_AUDIO_INFO_RAW
- NEW_AVI_INFO_RAW
- FIFO_NEAR_OVFL_RAW
- FIFO_UNDERFLO_RAW
- FIFO_OVERFLO_RAW
- CTS_PASS_THRSH_RAW
- CHANGE_N_RAW
- PACKET_ERROR_RAW
- AUDIO_PCKT_ERR_RAW
- NEW_GAMUT_MDATA_RAW
- DEEP_COLOR_CHNG_RAW
- VCLK_CHNG_RAW
- AUDIO_MODE_CHNG_RAW
- PARITY_ERROR_RAW
- NEW_SAMP_RT_RAW
- AUDIO_FLT_LINE_RAW
- NEW_TMDS_FRQ_RAW
- FIFO_NEAR_UFLO_RAW
- MS_INF_CKS_ERR_RAW
- SPD_INF_CKS_ERR_RAW
- AUD_INF_CKS_ERR_RAW
- AVI_INF_CKS_ERR_RAW
- AKSV_UPDATE_A_RAW
- AKSV_UPDATE_B_RAW
- BG_MEAS_DONE_RAW
- VS_INF_CKS_ERR_RAW

16.4 ADDITIONAL EXPLANATIONS

16.4.1 AFE_INTERRUPT_RAW

The AFE section contains the logic to slice 8 tri-level inputs. Each tri-level input has two independently programmable slice levels, and an interrupt associated with each slice level. Therefore the AFE section can generate 16 different interrupts related to the 8 tri-level inputs.

Each tri-level input has an interrupt mask control (TRIX_INT_MASK[1:0] where X = 1 to 8) which enables or disables interrupt generation when a tri-level input crosses its upper or lower slice level, a read-only status value (TRIX_INT_STATUS[1:0]) to indicate which tri-level input has generated an interrupt, and an interrupt clear control (TRIX_INT_CLEAR[1:0]) to clear latched interrupts. Real-time tri-level status can be obtained by reading TRIX_READBACK[1:0].

The 16 possible AFE interrupts are OR'd together and used to trigger the IO map AFE_INTERRUPT_RAW bit.

As an example, to generate in interrupt on INT2 when tri-level 6 crosses its upper slice level, firstly in the AFE map TRI6_INT_MASK[1:0] must be set to 0b10 to generate an interrupt when tri-level 6 input crosses its upper slice level, then in the IO map AFE_INTERRUPT_MB2 must be set to 0b1 to generate an interrupt on INT2 when AFE_INTERRUPT_RAW is set.

The following flow chart suggests a method to handle the AFE_INTERRUPT interrupt to avoid missing any interrupts that occur at the same time, or while servicing another interrupt.

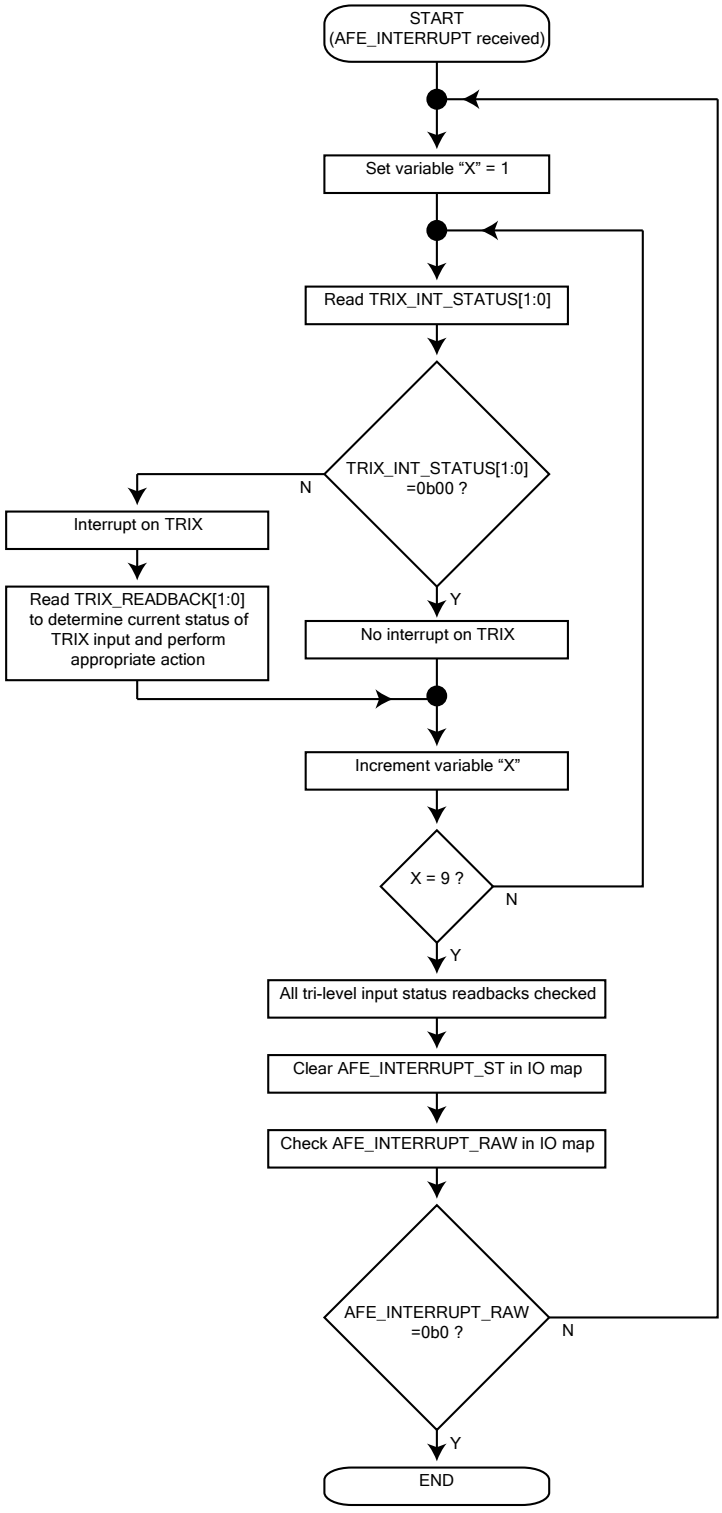


Figure 166: Suggested Method of Handling AFE_INTERRUPT

16.4.2 STDI_DATA_VALID_RAW

STDI_DATA_VALID_RAW is programmable as either an edge sensitive bit or a level sensitive bit using the following control. Note that this control also configures whether an interrupt is generated only on the rising edge of stdi_data_valid_raw, or on both edges.

STDI_DATA_VALID_EDGE_SEL, Addr 40 (IO), *Address 0x41[4]*

A control to configure the functionality of the STDI_DATA_VALID interrupt. The interrupt can be generated for the case when STDI changes to an STDI valid state. Alternatively it can be generated to indicate a change in STDI_VALID status.

Function

STDI_DATA_VALID_EDGE_SEL	Description
0	Generate interrupt for a LOW to HIGH change in STDI_VALID status
1 «	Generate interrupt for a LOW to HIGH or a HIGH to LOW change in STDI_VALID status

16.4.3 CP_LOCK, CP_UNLOCK

CP_UNLOCK_RAW is programmable as either an edge sensitive bit or a level sensitive bit using the following control. Note that this control also configures whether an interrupt is generated only on the rising edge of CP_UNLOCK_RAW, or on both edges.

CP_LOCK_UNLOCK_EDGE_SEL, Addr 40 (IO), *Address 0x41[5]*

A control to configure the functionality of the CP_LOCK and CP_UNLOCK interrupts. The interrupts can be generated when their respective status, CP_LOCK, CP_UNLOCK are valid. Or alternatively an interrupt can be generated when a change in their respective status occurs.

Function

CP_LOCK_UNLOCK_EDGE_SEL	Description
0	Generate interrupt for a LOW to HIGH change only for the CP_LOCK and CP_UNLOCK interrupts.
1 «	Generate interrupt for a LOW to HIGH or a HIGH to LOW change for the the CP_LOCK and CP_UNLOCK interrupts.

CP_UNLOCK_ST, Addr 40 (IO), *Address 0x43[3] (Read Only)*

Latched signal status of CP Unlock interrupt signal. Once set this bit will remain high until the interrupt has been cleared via CP_UNLOCK_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit.

Function

CP_UNLOCK_ST	Description
0 «	No CP UNLOCK interrupt event has occurred.
1	A CP UNLOCK interrupt event has occurred.

CP_LOCK_ST, Addr 40 (IO), *Address 0x43[2] (Read Only)*

Latched signal status of the CP Lock interrupt signal. Once set this bit will remain high until the interrupt has been cleared via CP_LOCK_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

CP_LOCK_ST	Description
0 «	No CP LOCK interrupt event has occurred.
1	A CP LOCK interrupt event has occurred.

16.4.4 Video 3D detection

The ADV7842 allows checking for 3D HDMI activity using following bits:

VIDEO_3D_RAW, Addr 40 (IO), *Address 0x74[2] (Read Only)*

Raw status of the Video 3D signal. This flag is set when the following requirements are met; a VS Infoframe is received with byte PB1, PB2 and PB3 set to 0x000C03 and the HDMI_VIDEO_FORMAT field in the VS Infoframe is set to 010b. This flag is cleared when a VS Infoframe with the appropriate requirements for 3D or when a VS Infoframe has not been received within 3 VSyncs.

Function

VIDEO_3D_RAW	Description
0 «	Video 3D not detected
1	Video 3D detected

16.4.5 HDMI Interrupts Validity Checking Process

All HDMI interrupts have a set of conditions that must be taken into account for validation in the display firmware. When the ADV7842 interrupts the display controller for an HDMI interrupt, the host must check that all validity conditions for that interrupt are met before processing that interrupt.

For simplicity, HDMI interrupts can be subdivided into three groups, as listed in the following sections.

16.4.5.1 Group 1 HDMI Interrupts

The interrupts listed in Table 97 are valid irrespective of the mode in which the ADV7842 is configured, that is:

- COMP mode (PRIM_MODE set to 0x01)
- GR mode (PRIM_MODE set to 0x02)
- HDMI mode (PRIM_MODE set to values 0x05 or 0x06)

Table 97: HDMI Interrupts Group 1

Interrupts
TMDS_CLK_A
TMDS_CLK_B
CABLE_DET_A
CABLE_DET_B

16.4.5.2 Group 2 HDMI Interrupts

The interrupts listed in Table 98 are valid on the condition that the ADV7842 is configured in HMDI mode.

Table 98: HDMI Interrupts Group 2

Interrupts

Interrupts
INTERNAL_MUTE
VIDEO_PLL_LCK
AKSV_UPDATE
V_LOCKED
DE_REGEN_LCK

16.4.5.3 Group 3 HDMI Interrupts

The interrupts listed in Table 99 are valid under the following conditions:

- ADV7842 is configured in HMDI mode
- TMDS_CLK_A_RAW is set to 1 if port A is the active HDMI port
- TMDS_CLK_B_RAW is set to 1 if port B is the active HDMI port
- VIDEO_PLL_LOCKED_RAW is set to 1

Table 99: HDMI Interrupts Group 3

Interrupts
ISRC2_PCKT
ISRC1_PCKT
ACP_PCKT
VS_INFO
MS_INFO
SPD_INFO
AUDIO_INFO
AVI_INFO
CS_DATA_VALID
AV_MUTE
AUDIO_CH_MD
AUDIO_MODE_CHANGE
GEN_CTL_PCKT
AUDIO_C_PCKT
GAMUT_MDATA
HDMI_MODE
HDMI_ENCRPT
NEW_ISRC2_INFO
NEW_ISRC1_INFO
NEW_ACP_INFO
NEW_VS_INFO
NEW_MS_INFO
NEW_SPD_INFO
NEW_AUDIO_INFO
NEW_AVI_INFO
FIFO_NEAR_OVFL
CTS_PASS_THRSH
CHANGE_N
PACKET_ERROR

Interrupts
AUDIO_PCKT_ERR
NEW_GAMUT_DATA
DEEP_COLOR_CHNG
VCLK_CHNG
PARRITY_ERROR
NEW_SAMP_RT
AUDIO_FLT_LINE
NEW_TMDS_FRQ
FIFO_NEAR_UFLO

16.4.6 Storing Masked Interrupts

STORE_UNMASKED_IRQS, Addr 40 (IO), *Address 0x40[4]*

STORE_MASKED_IRQS allows the HDMI status flags for any HDMI interrupt to be triggered regardless of whether the mask bits are set. This bit allows a HDMI interrupt to trigger and allows this interrupt to be read back through the corresponding status bit without triggering an interrupt on the interrupt pin. The status is stored until the clear bit is used to clear the status register and allows another interrupt to occur.

Function

STORE_UNMASKED_IRQS	Description
0 «	Does not allow x_ST flag of any HDMI interrupt to be set independently of mask bits
1	Allows x_ST flag of any HDMI interrupt to be set independently of mask bits

16.4.6.1 List of interrupt status registers

Trilevel Interrupt Status 1register consists of fields:

TRI1_INT_STATUS[1:0], Addr 4C (AFE), *Address 0x1B[7:6] (Read Only)*

Tri1 interrupt status

Function

TRI1_INT_STATUS[1:0]	Description
00 «	No signal change detected
01	Signal has crossed lower slice level
10	Signal has crossed upper slice level
11	Signal has crossed both slice levels

TRI2_INT_STATUS[1:0], Addr 4C (AFE), *Address 0x1B[5:4] (Read Only)*

Tri2 interrupt status

Function

TRI2_INT_STATUS[1:0]	Description
00 «	No signal change detected
01	Signal has crossed lower slice level
10	Signal has crossed upper slice level
11	Signal has crossed both slice levels

TRI3_INT_STATUS[1:0], Addr 4C (AFE), *Address 0x1B[3:2] (Read Only)*

Tri3 interrupt status

Function

TRI3_INT_STATUS[1:0]	Description
00 «	No signal change detected
01	Signal has crossed lower slice level
10	Signal has crossed upper slice level
11	Signal has crossed both slice levels

TRI4_INT_STATUS[1:0], Addr 4C (AFE), *Address 0x1B[1:0] (Read Only)*

Tri4 interrupt status

Function

TRI4_INT_STATUS[1:0]	Description
00 «	No signal change detected
01	Signal has crossed lower slice level
10	Signal has crossed upper slice level
11	Signal has crossed both slice levels

Trilevel Interrupt Status 2 register consists of fields:

TRI5_INT_STATUS[1:0], Addr 4C (AFE), *Address 0x1C[7:6] (Read Only)*

Tri5 interrupt status

Function

TRI5_INT_STATUS[1:0]	Description
00 «	No signal change detected
01	Signal has crossed lower slice level
10	Signal has crossed upper slice level
11	Signal has crossed both slice levels

TRI6_INT_STATUS[1:0], Addr 4C (AFE), *Address 0x1C[5:4] (Read Only)*

Tri6 interrupt status

Function

TRI6_INT_STATUS[1:0]	Description
00 «	No signal change detected
01	Signal has crossed lower slice level
10	Signal has crossed upper slice level
11	Signal has crossed both slice levels

TRI7_INT_STATUS[1:0], Addr 4C (AFE), *Address 0x1C[3:2] (Read Only)*

Tri7 interrupt status

Function

TRI7_INT_STATUS[1:0]	Description
00 «	No signal change detected
01	Signal has crossed lower slice level
10	Signal has crossed upper slice level
11	Signal has crossed both slice levels

TRI8_INT_STATUS[1:0], Addr 4C (AFE), *Address 0x1C[1:0] (Read Only)*

Tri8 interrupt status

Function

TRI8_INT_STATUS[1:0]	Description
00 «	No signal change detected
01	Signal has crossed lower slice level
10	Signal has crossed upper slice level
11	Signal has crossed both slice levels

INTERRUPT_STATUS_1 register consists of the following fields.

SSPD_RSLT_CHNGD_ST, Addr 40 (IO), *Address 0x43[7] (Read Only)*

Latched signal status of SSPD Result Changed interrupt signal. Once set this bit will remain high until the interrupt has been cleared via SSPD_RSLT_CHNGD_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

SSPD_RSLT_CHNGD_ST	Description
0 «	No SSPD result changed interrupt event occurred.
1	A SSPD result changed interrupt event has occurred..

MV_PS_DET_ST, Addr 40 (IO), *Address 0x43[6] (Read Only)*

Latched signal status of Macrovision Pseudo sync detected interrupt signal. Once set this bit will remain high until the interrupt has been cleared via MV_PS_DET_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

MV_PS_DET_ST	Description
0 «	No Macrovision pseudo sync detection interrupt event has occurred.
1	A Macrovision pseudo sync detected interrupt event has occurred.

STDI_DATA_VALID_ST, Addr 40 (IO), *Address 0x43[4] (Read Only)*

Latched signal status of STDI valid interrupt signal. Once set this bit will remain high until the interrupt has been cleared via STDI_DATA_VALID_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

STDI_DATA_VALID_ST	Description
0 «	No STDI valid interrupt has occurred.
1	A STDI valid interrupt has occurred.

AFE_INTERRUPT_ST, Addr 40 (IO), *Address 0x43[0] (Read Only)*

Latched signal status of the AFE interrupt signal. Once set this bit will remain high until the interrupt has been cleared via AFE_INTERRUPT_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

AFE_INTERRUPT_ST	Description
0 «	No AFE interrupt event has occurred.
1	AFE interrupt event has occurred.

INTERRUPT_STATUS_2 register consists of the following fields.

MPU_STIM_INTRQ_ST, Addr 40 (IO), *Address 0x48[7] (Read Only)*

Latched signal status of Manual Forced interrupt signal. Once set this bit will remain high until the interrupt has been cleared via MPU_STIM_INTRQ_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

MPU_STIM_INTRQ_ST	Description
0 «	Forced manual interrupt event has not occurred.
1	Force manual interrupt even has occurred.

MV_AGC_DET_ST, Addr 40 (IO), *Address 0x48[6] (Read Only)*

Latched signal status of Macrovision AGC detected interrupt signal. Once set this bit will remain high until the interrupt has been cleared via MV_AGC_DET_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

MV_AGC_DET_ST	Description
0 «	A Macrovision AGC detected interrupt event has not occurred.
1	A Macrovision AGC detected interrupt event has occurred.

MV_CS_DET_ST, Addr 40 (IO), *Address 0x48[5] (Read Only)*

Latched signal status of Macrovision Color-stripe detected interrupt signal. Once set this bit will remain high until the interrupt has been cleared via MV_CS_DET_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

MV_CS_DET_ST	Description
0 «	A Macrovision Color-stripe detected interrupt event has not occurred.
1	A Macrovision Color-stripe detected interrupt event has occurred.

CP_CGMS_CHNGD_ST, Addr 40 (IO), *Address 0x48[2] (Read Only)*

Latched signal status of CP CGMS Changed interrupt signal. Once set this bit will remain high until the interrupt has been cleared via CP_CGMS_CHNGD_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

CP_CGMS_CHNGD_ST	Description
0 «	A CGMS data changed interrupt event has not occurred.
1	A CGMS data changed interrupt event has occurred.

INTERRUPT_STATUS_3 register consists of the following fields.

AVLINK_RX_READY_ST, Addr 40 (IO), *Address 0x4D[3] (Read Only)*

Latched status of AV.link Receiver ready interrupt signal. Once set this bit will remain high until the interrupt has been cleared via AVLINK_RX_READY_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

AVLINK_RX_READY_ST	Description
0 «	AV.link message received interrupt has not occurred.
1	AV.link message received interrupt has occurred.

AVLINK_TX_READY_ST, Addr 40 (IO), *Address 0x4D[0] (Read Only)*

Latched status of AVLINK_TX_READY_RAW signal. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit. When the AV.link TX successfully sends the current message this bit is set. Once set this bit will remain high until the interrupt has been cleared via AVLINK_TX_READY_CLR.

Function

AVLINK_TX_READY_ST	Description
0 «	No change
1	Message transmitted successfully

INTERRUPT_STATUS_4 register consists of the following fields.

TTXT_AVL_ST, Addr 40 (IO), *Address 0x52[7] (Read Only)*

Latched status of Teletext data available interrupt signal. Once set this bit will remain high until the interrupt has been cleared via TTXT_AVL_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

TTXT_AVL_ST	Description
0 «	No Teletext data available interrupt event has occurred.
1	No Teletext data available interrupt event has occurred.

VITC_AVL_ST, Addr 40 (IO), *Address 0x52[6] (Read Only)*

Latched status of VITC data available interrupt signal. Once set this bit will remain high until the interrupt has been cleared via VITC_AVL_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

VITC_AVL_ST	Description
0 «	No VITC data available interrupt event has occurred.
1	A VITC data available interrupt event has occurred.

GS_DATA_TYPE_ST, Addr 40 (IO), *Address 0x52[5] (Read Only)*

Latched status of Gemstar type available interrupt signal. Once set this bit will remain high until the interrupt has been cleared via GS_DATA_TYPE_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

GS_DATA_TYPE_ST	Description
0 «	No Gemstar data type interrupt event has occurred.
1	A Gemstar data type interrupt event has occurred.

GS_PDC_VPS_UTC_AVL_ST, Addr 40 (IO), *Address 0x52[4] (Read Only)*

Latched status of Gemstar/ PDC/ VPS/UTC data available interrupt signal. Once set this bit will remain high until the interrupt has been cleared via GS_PDC_VPS_UTC_AVL_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

GS_PDC_VPS_UTC_AVL_ST	Description
0 «	No GemStar/PDC/VPS/UTC data available interrupt event has occurred.
1	A GemStar/PDC/VPS/UTC data available interrupt event has occurred.

FASTI2C_DATA_RDY_ST, Addr 40 (IO), *Address 0x52[3] (Read Only)*

Function

FASTI2C_DATA_RDY_ST	Description
0 «	Positive Edge not detected on FAST_I2C_DATA_RDY
1	Positive Edge detected on FAST_I2C_DATA_RDY

CGMS_WSS_AVL_ST, Addr 40 (IO), *Address 0x52[2] (Read Only)*

Latched status of CGMS/WSS data available interrupt signal. Once set this bit will remain high until the interrupt has been cleared via CGMS_WSS_AVL_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

CGMS_WSS_AVL_ST	Description
0 «	No CGMS/WSS data available interrupt event has occurred.
1	CGMS/WSS data available interrupt event has occurred.

CCAP_EVEN_FIELD_ST, Addr 40 (IO), *Address 0x52[1] (Read Only)*

Latched status of Closed captioning detected on even field interrupt signal. Once set this bit will remain high until the interrupt has been cleared via CCAP_EVEN_FIELD_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

CCAP_EVEN_FIELD_ST	Description
0 «	No closed captioning detected on even field interrupt event has occurred.
1	A closed captioning detected on even field interrupt event has occurred.

CCAP_AVL_ST, Addr 40 (IO), *Address 0x52[0] (Read Only)*

Latched status of Closed captioning data available interrupt signal. Once set this bit will remain high until the interrupt has been cleared via CCAP_AVL_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

CCAP_AVL_ST	Description
0 «	No Closed Captioning data available interrupt event has occurred.
1	A Closed Captioning data available interrupt event has occurred.

INTERRUPT_STATUS_5 register consists of the following fields.

SDP_PROGRESSIVE_ST, Addr 40 (IO), *Address 0x57[7] (Read Only)*

Function

SDP_PROGRESSIVE_ST	Description
0 «	No change. An interrupt has not been generated from this register.
1	ESDP_PROGRESSIVE_RAW has changed and generated an interrupt.

SDP_PR_DET_ST, Addr 40 (IO), *Address 0x57[6] (Read Only)*

Function

SDP_PR_DET_ST	Description
0 «	No change. An interrupt has not been generated from this register.
1	ESDP_PR_DET_RAW has changed and generated an interrupt.

SDP_SD_DET_ST, Addr 40 (IO), *Address 0x57[5] (Read Only)*

Function

SDP_SD_DET_ST	Description
0 «	No change. An interrupt has not been generated from this register.
1	ESDP_SD_DET_RAW has changed and generated an interrupt.

SDP_50HZ_DET_ST, Addr 40 (IO), *Address 0x57[4] (Read Only)*

Function

SDP_50HZ_DET_ST	Description
0 «	No change. An interrupt has not been generated from this register.
1	ESDP_50HZ_DET_RAW has changed and generated an interrupt.

INTERRUPT_STATUS_6 register consists of the following fields.

CP_LOCK_CH2_ST, Addr 40 (IO), *Address 0x5C[7] (Read Only)*

Function

CP_LOCK_CH2_ST	Description
0 «	No change. An interrupt has not been generated from this register.
1	Channel 2 CP input has caused the decoder to go from an unlocked state to a locked state

CP_UNLOCK_CH2_ST, Addr 40 (IO), *Address 0x5C[6] (Read Only)*

Function

CP_UNLOCK_CH2_ST	Description
0 «	No change. An interrupt has not been generated from this register.
1	CP input has caused the decoder to go from a locked state to an unlocked state

STDI_DVALID_CH2_ST, Addr 40 (IO), *Address 0x5C[5] (Read Only)*

Latched signal status of STDI valid for sync channel 2 interrupt signal. Once set this bit will remain high until the interrupt has been cleared via STDI_DATA_VALID_CH2_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

STDI_DVALID_CH2_ST	Description
0 «	No STDI valid for sync channel 2 interrupt has occurred.
1	A STDI valid for sync channel 2 interrupt has occurred.

CP_LOCK_CH1_ST, Addr 40 (IO), *Address 0x5C[3] (Read Only)*

Function

CP_LOCK_CH1_ST	Description
0 «	No change. An interrupt has not been generated from this register.
1	Channel 1 CP input has caused the decoder to go from an unlocked state to a locked state

CP_UNLOCK_CH1_ST, Addr 40 (IO), *Address 0x5C[2] (Read Only)*

Function

CP_UNLOCK_CH1_ST	Description
0 «	No change. An interrupt has not been generated from this register.
1	Channel 1 CP input has changed from a locked state to an unlocked state and has triggered an interrupt

STDI_DVALID_CH1_ST, Addr 40 (IO), *Address 0x5C[1] (Read Only)*

Latched signal status of STDI valid for sync channel 1 interrupt signal. Once set this bit will remain high until the interrupt has been cleared via STDI_DATA_VALID_CH1_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

STDI_DVALID_CH1_ST	Description
0 «	No STDI valid for sync channel 1 interrupt has occurred.
1	A STDI valid for sync channel 1 interrupt has occurred.

SSPD_RSLT_CHNGD_CH1_ST

HDMI Lvl INT Status 1 register consists of the following fields.

ISRC2_PCKT_ST, Addr 40 (IO), *Address 0x61[7] (Read Only)*

Latched status of ISRC2 Packet detected interrupt signal. Once set this bit will remain high until the interrupt has been cleared via ISRC2_INFO_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

ISRC2_PCKT_ST	Description
0 «	No interrupt generated from this register
1	ISRC2_PCKT_RAW has changed. Interrupt has been generated.

ISRC1_PCKT_ST, Addr 40 (IO), *Address 0x61[6] (Read Only)*

Latched status of ISRC1 Packet detected interrupt signal. Once set this bit will remain high until the interrupt has been cleared via ISRC1_INFO_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

ISRC1_PCKT_ST	Description
0 «	No interrupt generated from this register
1	ISRC1_PCKT_RAW has changed. Interrupt has been generated.

ACP_PCKT_ST, Addr 40 (IO), *Address 0x61[5] (Read Only)*

Latched status of Audio Content Protection Packet detected interrupt signal. Once set this bit will remain high until the interrupt has been cleared via ACP_INFO_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

ACP_PCKT_ST	Description
0 «	No interrupt generated from this register
1	ACP_PCKT_RAW has changed. Interrupt has been generated.

VS_INFO_ST, Addr 40 (IO), *Address 0x61[4] (Read Only)*

Latched status of Vendor Specific Infoframe detected interrupt signal. Once set this bit will remain high until the interrupt has been cleared via VS_INFO_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

VS_INFO_ST	Description
0 «	No interrupt generated from this register
1	VS_INFO_RAW has changed. Interrupt has been generated.

MS_INFO_ST, Addr 40 (IO), *Address 0x61[3] (Read Only)*

Latched status of MPEG Source Infoframe detected interrupt signal. Once set this bit will remain high until the interrupt has been cleared via MS_INFO_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

MS_INFO_ST	Description
0 «	No interrupt generated from this register
1	MS_INFO_RAW has changed. Interrupt has been generated.

SPD_INFO_ST, Addr 40 (IO), *Address 0x61[2] (Read Only)*

Latched status of SPD Infoframe detected interrupt signal. Once set this bit will remain high until the interrupt has been cleared via SPD_INFO_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

SPD_INFO_ST	Description
0 «	No interrupt generated from this register
1	SPD_INFO_RAW has changed. Interrupt has been generated.

AUDIO_INFO_ST, Addr 40 (IO), *Address 0x61[1] (Read Only)*

Latched status of Audio Infoframe detected interrupt signal. Once set this bit will remain high until the interrupt has been cleared via AUDIO_INFO_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

AUDIO_INFO_ST	Description
0 «	No interrupt generated from this register
1	AUDIO_INFO_RAW has changed. Interrupt has been generated.

HDMI Lvl INT Status 2 register consists of the following fields.

CS_DATA_VALID_ST, Addr 40 (IO), *Address 0x66[7] (Read Only)*

Latched status of Channel Status Data Valid interrupt signal. Once set this bit will remain high until the interrupt has been cleared via CS_DATA_VALID_CLR. This bit is only valid if enabled via the corresponding the INT1 or INT2 interrupt mask bit

Function

CS_DATA_VALID_ST	Description
0 «	CS_DATA_VALID_RAW has not changed. An interrupt has not been generated.
1	CS_DATA_VALID_RAW has changed. An interrupt has been generated.

INTERNAL_MUTE_ST, Addr 40 (IO), *Address 0x66[6] (Read Only)*

Latched status of Internal Mute interrupt signal. Once set this bit will remain high until the interrupt has been cleared via INTERNAL_MUTE_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

INTERNAL_MUTE_ST	Description
0 «	INTERNAL_MUTE_RAW has not changed. An interrupt has not been generated.
1	INTERNAL_MUTE_RAW has changed. An interrupt has been generated.

AV_MUTE_ST, Addr 40 (IO), *Address 0x66[5] (Read Only)*

Latched status of AV Mute detected interrupt signal. Once set this bit will remain high until the interrupt has been cleared via AV_MUTE_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

AV_MUTE_ST	Description
0 «	AV_MUTE_RAW has not changed. An interrupt has not been generated.
1	AV_MUTE_RAW has changed. An interrupt has been generated.

AUDIO_CH_MD_ST, Addr 40 (IO), *Address 0x66[4] (Read Only)*

Latched status of Audio Channel mode interrupt signal. Once set this bit will remain high until the interrupt has been cleared via AUDIO_CH_MD_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

AUDIO_CH_MD_ST	Description
0 «	AUDIO_CH_MD_RAW has not changed. An interrupt has not been generated.
1	AUDIO_MODE_CHNG_RAW has changed. An interrupt has been generated.

HDMI_MODE_ST, Addr 40 (IO), *Address 0x66[3] (Read Only)*

Latched status of HDMI Mode interrupt signal. Once set this bit will remain high until the interrupt has been cleared via HDMI_MODE_CLR. This bit is only valid if enabled via the corresponding the INT1 or INT2 interrupt mask bit

Function

HDMI_MODE_ST	Description
0 «	HDMI_MODE_RAW has not changed. An interrupt has not been generated.
1	HDMI_MODE_RAW has changed. An interrupt has been generated.

GEN_CTL_PCKT_ST, Addr 40 (IO), *Address 0x66[2] (Read Only)*

Latched status of General Control Packet interrupt signal. Once set this bit will remain high until the interrupt has been cleared via GEN_CTL_PCKT_CLR. This bit is only valid if enabled via the corresponding the INT1 or INT2 interrupt mask bit.

Function

GEN_CTL_PCKT_ST	Description
0 «	GEN_CTL_PCKT_RAW has not changed. Interrupt has not been generated from this register.
1	GEN_CTL_PCKT_RAW has changed. Interrupt has been generated from this register.

AUDIO_C_PCKT_ST, Addr 40 (IO), *Address 0x66[1] (Read Only)*

Latched status of Audio Clock Regeneration Packet interrupt signal. Once set this bit will remain high until the interrupt has been cleared via AUDIO_PCKT_CLR. This bit is only valid if enabled via the corresponding the INT1 or INT2 interrupt mask bit

Function

AUDIO_C_PCKT_ST	Description
0 «	AUDIO_C_PCKT_RAW has not changed. Interrupt has not been generated from this register
1	AUDIO_C_PCKT_RAW has changed. Interrupt has been generated from this register.

GAMUT_MDATA_ST, Addr 40 (IO), *Address 0x66[0] (Read Only)*

Latched status of Gamut Metadata Packet detected interrupt signal. Once set this bit will remain high until the interrupt has been cleared via GAMUT_MDATA_PCKT_CLR. This bit is only valid if enabled via the corresponding the INT1 or INT2 interrupt mask bit

Function

GAMUT_MDATA_ST	Description
0 «	GAMUT_MDATA_RAW has not changed. Interrupt has not been generated from this register
1	GAMUT_MDATA_RAW has changed. Interrupt has been generated from this register.

HDMI Lvl INT Status 3 register consists of the following fields.

TMDSPLL_LCK_A_ST, Addr 40 (IO), *Address 0x6B[5] (Read Only)*

Latched status of Port A TMDS PLL Lock interrupt signal. Once set this bit will remain high until the interrupt has been cleared via TMDSPLL_LCK_A_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

TMDSPLL_LCK_A_ST	Description
0 «	TMDSPLL_LCK_A_RAW has not changed. An interrupt has not been generated
1	TMDSPLL_LCK_A_RAW has changed. An interrupt has been generated.

TMDSPLL_LCK_B_ST, Addr 40 (IO), *Address 0x6B[4] (Read Only)*

Latched status of Port B TMDS PLL Lock interrupt signal. Once set this bit will remain high until the interrupt has been cleared via TMDSPLL_LCK_B_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

TMDSPLL_LCK_B_ST	Description
0 «	TMDSPLL_LCK_B_RAW has not changed. An interrupt has not been generated.
1	TMDSPLL_LCK_B_RAW has changed. An interrupt has been generated.

TMDS_CLK_A_ST, Addr 40 (IO), *Address 0x6B[1] (Read Only)*

Latched status of Port A TMDS Clock Detection interrupt signal .Once set this bit will remain high until the interrupt has been cleared via TMDS_CLK_A_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

TMDS_CLK_A_ST	Description
0 «	TMDS_CLK_A_RAW has not changed. An interrupt has not been generated.
1	TMDS_CLK_A_RAW has changed. An interrupt has been generated.

TMDS_CLK_B_ST, Addr 40 (IO), *Address 0x6B[0] (Read Only)*

Latched status of Port B TMDS Clock Detection interrupt signal .Once set this bit will remain high until the interrupt has been cleared via TMDS_CLK_B_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

TMDS_CLK_B_ST	Description
0 «	TMDS_CLK_B_RAW has not changed. An interrupt has not been generated.
1	TMDS_CLK_B_RAW has changed. An interrupt has been generated.

HDMI Lvl INT Status 4 register consists of the following fields.

HDMI_ENCRPT_A_ST, Addr 40 (IO), *Address 0x70[5] (Read Only)*

Latched status for Port A Encryption detection interrupt signal. Once set this bit will remain high until the interrupt has been cleared via HDMI_ENCRPT_A_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

HDMI_ENCRPT_A_ST	Description
0 «	HDMI_ENCRPT_A_RAW has not changed. An interrupt has not been generated
1	HDMI_ENCRPT_A_RAW has changed. An interrupt has been generated.

HDMI_ENCRPT_B_ST, Addr 40 (IO), *Address 0x70[4] (Read Only)*

Latched status for Port B Encryption detection interrupt signal. Once set this bit will remain high until the interrupt has been cleared via HDMI_ENCRPT_B_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

HDMI_ENCRPT_B_ST	Description
0 «	HDMI_ENCRPT_B_RAW has not changed. An interrupt has not been generated
1	HDMI_ENCRPT_B_RAW has changed. An interrupt has been generated.

CABLE_DET_A_ST, Addr 40 (IO), *Address 0x70[1] (Read Only)*

Latched status for Port A +5V cable detection interrupt signal. Once set this bit will remain high until the interrupt has been cleared via CABLE_DET_A_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

CABLE_DET_A_ST	Description
0 «	CABLE_DET_A_RAW has not changed. Interrupt has not been generated from this register.
1	CABLE_DET_A_RAW has changed. Interrupt has been generated from this register.

CABLE_DET_B_ST, Addr 40 (IO), *Address 0x70[0] (Read Only)*

Latched status for Port B +5V cable detection interrupt signal. Once set this bit will remain high until the interrupt has been cleared via CABLE_DET_B_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

CABLE_DET_B_ST	Description
0 «	CABLE_DET_B_RAW has not changed. Interrupt has not been generated from this register.
1	CABLE_DET_B_RAW has changed. Interrupt has been generated from this register.

HDMI Lvl INT Status 5 register consists of the following fields.

VIDEO_3D_ST, Addr 40 (IO), *Address 0x75[2] (Read Only)*

Latched status for the Video 3D interrupt. Once set this bit will remain high until the interrupt has been cleared via VIDEO_3D_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

VIDEO_3D_ST	Description
0 «	VIDEO_3D_RAW has not changed. An interrupt has not been generated.
1	VIDEO_3D_RAW has changed. An interrupt has been generated.

V_LOCKED_ST, Addr 40 (IO), *Address 0x75[1] (Read Only)*

Latched status for the Vertical Sync Filter Locked interrupt. Once set this bit will remain high until the interrupt has been cleared via V_LOCKED_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

V_LOCKED_ST	Description
0 «	V_LOCKED_RAW has not changed. An interrupt has not been generated.
1	V_LOCKED_RAW has changed. An interrupt has been generated.

DE_REGEN_LCK_ST, Addr 40 (IO), *Address 0x75[0] (Read Only)*

Latched status for DE Regeneration Lock interrupt signal. Once set this bit will remain high until the interrupt has been cleared via DE_REGEN_LCK_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

DE_REGEN_LCK_ST	Description
0 «	DE_REGEN_LCK_RAW has not changed. An interrupt has not been generated.
1	DE_REGEN_LCK_RAW has changed. An interrupt has been generated.

HDMI Edg INT Status 1 register consists of the following fields.

NEW_ISRC2_PCKT_ST, Addr 40 (IO), *Address 0x7A[7] (Read Only)*

Latched status for the New ISRC2 Packet interrupt. Once set this bit will remain high until the interrupt has been cleared via NEW_ISRC2_PCKT_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

NEW_ISRC2_PCKT_ST	Description
0 «	No new ISRC2 packet received. An interrupt has not been generated.
1	ISRC2 packet with new content received. An interrupt has been generated.

NEW_ISRC1_PCKT_ST, Addr 40 (IO), *Address 0x7A[6] (Read Only)*

Latched status for the New ISRC1 Packet interrupt. Once set this bit will remain high until the interrupt has been cleared via NEW_ISRC1_PCKT_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

NEW_ISRC1_PCKT_ST	Description
0 «	No new ISRC1 packet received. An interrupt has not been generated.
1	ISRC1 packet with new content received. An interrupt has been generated.

NEW_ACP_PCKT_ST, Addr 40 (IO), *Address 0x7A[5] (Read Only)*

Latched status for the New ACP Packet interrupt. Once set this bit will remain high until the interrupt has been cleared via NEW_ACP_PCKT_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

NEW_ACP_PCKT_ST	Description
0 «	No new ACP packet received. An interrupt has not been generated.
1	ACP packet with new content received. An interrupt has been generated.

NEW_VS_INFO_ST, Addr 40 (IO), *Address 0x7A[4] (Read Only)*

Latched status for the New Vendor Specific Infoframe interrupt. Once set this bit will remain high until the interrupt has been cleared via NEW_VS_INFO_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

NEW_VS_INFO_ST	Description
0 «	No new VS packet received. An interrupt has not been generated.
1	VS packet with new content received. An interrupt has been generated.

NEW_MS_INFO_ST, Addr 40 (IO), *Address 0x7A[3] (Read Only)*

Latched status for the New MPEG Source Infoframe interrupt. Once set this bit will remain high until the interrupt has been cleared via NEW_MS_INFO_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

NEW_MS_INFO_ST	Description
0 «	No new MPEG Source InfoFrame received. Interrupt has not been generated.
1	MPEG Source InfoFrame with new content received. Interrupt has been generated.

NEW_SPD_INFO_ST, Addr 40 (IO), *Address 0x7A[2] (Read Only)*

Latched status for the New Source Product Descriptor Infoframe interrupt. Once set this bit will remain high until the interrupt has been cleared via NEW_SPD_INFO_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

NEW_SPD_INFO_ST	Description
0 «	No new SPD InfoFrame received. Interrupt has not been generated.
1	SPD InfoFrame with new content received. Interrupt has been generated.

NEW_AUDIO_INFO_ST, Addr 40 (IO), *Address 0x7A[1] (Read Only)*

Latched status for the New Audio Infoframe interrupt. Once set this bit will remain high until the interrupt has been cleared via NEW_AUDIO_INFO_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

NEW_AUDIO_INFO_ST	Description
0 «	No new Audio InfoFrame received. Interrupt has not been generated.
1	Audio InfoFrame with new content received. Interrupt has been generated.

HDMI Edg INT Status 2 register consists of the following fields.

FIFO_NEAR_OVFL_ST, Addr 40 (IO), *Address 0x7F[7] (Read Only)*

Latched status for the Audio FIFO Near Overflow interrupt. Once set this bit will remain high until the interrupt has been cleared via FIFO_OVFL_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

FIFO_NEAR_OVFL_ST	Description
0 «	Audio FIFO has not reached high threshold
1	Audio FIFO has reached high threshold

FIFO_UNDERFLO_ST, Addr 40 (IO), *Address 0x7F[6] (Read Only)*

Latched status for the Audio FIFO Underflow interrupt. Once set this bit will remain high until the interrupt has been cleared via FIFO_UNDERFLO_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

FIFO_UNDERFLO_ST	Description
0 «	Audio FIFO has not underflowed
1	Audio FIFO has underflowed

FIFO_OVERFLO_ST, Addr 40 (IO), *Address 0x7F[5] (Read Only)*

Latched status for the Audio FIFO Overflow interrupt. Once set this bit will remain high until the interrupt has been cleared via FIFO_OVERFLO_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

FIFO_OVERFLO_ST	Description
0 «	Audio FIFO has not overflowed
1	Audio FIFO has overflowed

CTS_PASS_THRSH_ST, Addr 40 (IO), *Address 0x7F[4] (Read Only)*

Latched status for the ACR CTS Value Exceed Threshold interrupt. Once set this bit will remain high until the interrupt has been cleared via CTS_PASS_THRSH_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

CTS_PASS_THRSH_ST	Description
0 «	Audio clock regeneration CTS value has not passed the threshold
1	Audio clock regeneration CTS value has changed more than threshold

CHANGE_N_ST, Addr 40 (IO), *Address 0x7F[3] (Read Only)*

Latched status for the ACR N Value Changed interrupt. Once set this bit will remain high until the interrupt has been cleared via CHANGE_N_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

CHANGE_N_ST	Description
0 «	Audio clock regeneration N value has not changed
1	Audio clock regeneration N value has changed

PACKET_ERROR_ST, Addr 40 (IO), *Address 0x7F[2] (Read Only)*

Latched status for the Packet Error interrupt. Once set this bit will remain high until the interrupt has been cleared via PACKET_ERROR_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

PACKET_ERROR_ST	Description
0 «	No uncorrectable error detected in packet header. An interrupt has not been generated.
1	Uncorrectable error detected in an unknown packet (in packet header). An interrupt has been generated.

AUDIO_PCKT_ERR_ST, Addr 40 (IO), *Address 0x7F[1] (Read Only)*

Latched status for the Audio Packet Error interrupt. Once set this bit will remain high until the interrupt has been cleared via AUDIO_PCKT_ERR_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

AUDIO_PCKT_ERR_ST	Description
0 «	No uncorrectable error detected in audio packets. An interrupt has not been generated.
1	Uncorrectable error detected in an audio packet. An interrupt has been generated.

NEW_GAMUT_MDATA_ST, Addr 40 (IO), *Address 0x7F[0] (Read Only)*

Latched status for the New Gamut Metadata Packet interrupt. Once set this bit will remain high until the interrupt has been cleared via NEW_GAMUT_MDATA_PCKT_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

NEW_GAMUT_MDATA_ ST	Description
0 «	No new Gamut metadata packet received or no change has taken place. An interrupt has not been generated.
1	New Gamut metadata packet received. An interrupt has been generated.

HDMI Edg Status 3DEEP_COLOR_CHNG_ST, Addr 40 (IO), *Address 0x84[7] (Read Only)*

Latched status of Deep Color Mode Change Interrupt. Once set this bit will remain high until the interrupt has been cleared via DEEP_COLOR_CHNG_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

DEEP_COLOR_CHNG_ST	Description
0 «	Deep color mode has not changed
1	Change in deep color has been detected

VCLK_CHNG_ST, Addr 40 (IO), *Address 0x84[6] (Read Only)*

Latched status of Video Clock Change Interrupt. Once set this bit will remain high until the interrupt has been cleared via VCLK_CHNG_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

VCLK_CHNG_ST	Description
0 «	No irregular or missing pulse detected in TMDS clock
1	Irregular or missing pulses detected in TMDS clock

AUDIO_MODE_CHNG_ST, Addr 40 (IO), *Address 0x84[5] (Read Only)*

Latched status of Audio Mode Change Interrupt. Once set this bit will remain high until the interrupt has been cleared via AUDIO_MODE_CHNG_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

AUDIO_MODE_CHNG_ST	Description
0 «	Audio mode has not changed
1	Audio mode has changed. The following are considered Audio modes, No Audio, PCM, DSD or HBR

PARITY_ERROR_ST, Addr 40 (IO), *Address 0x84[4] (Read Only)*

Latched status of Parity Error Interrupt. Once set this bit will remain high until the interrupt has been cleared via PARITY_ERROR_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

PARITY_ERROR_ST	Description
0 «	No parity error detected in audio packets
1	Parity error detected in an audio packet

NEW_SAMP_RT_ST, Addr 40 (IO), *Address 0x84[3] (Read Only)*

Latched status of New Sampling Rate Interrupt. Once set this bit will remain high until the interrupt has been cleared via NEW_SAMP_RT_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

NEW_SAMP_RT_ST	Description
0 «	Sampling rate bits of the channel status data on audio channel 0 have not changed
1	Sampling rate bits of the channel status data on audio channel 0 have changed.

AUDIO_FLT_LINE_ST, Addr 40 (IO), *Address 0x84[2] (Read Only)*

Latched status of New TMDS Frequency Interrupt. Once set this bit will remain high until the interrupt has been cleared via NEW_TMDS_FREQ_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

AUDIO_FLT_LINE_ST	Description
0 «	Audio sample packet with flat line bit set has not been received
1	Audio sample packet with flat line bit set has been received

NEW_TMDS_FRQ_ST, Addr 40 (IO), *Address 0x84[1] (Read Only)*

Latched status of New TMDS Frequency Interrupt. Once set this bit will remain high until the interrupt has been cleared via NEW_TMDS_FREQ_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

NEW_TMDS_FRQ_ST	Description
0 «	TMDS frequency has not changed by more than tolerance
1	TMDS frequency has changed by more than tolerance

FIFO_NEAR_UFLO_ST, Addr 40 (IO), *Address 0x84[0] (Read Only)*

Latched status for the Audio FIFO near Underflow interrupt. Once set this bit will remain high until the interrupt has been cleared via FIFO_UFLO_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

FIFO_NEAR_UFLO_ST	Description
0 «	Audio FIFO has not reached low threshold
1	Audio FIFO has reached low threshold

HDMI Edg Status 4 register consists of the following fields.

MS_INF_CKS_ERR_ST, Addr 40 (IO), *Address 0x89[7] (Read Only)*

Latched status of MPEG Source Infoframe Checksum Error interrupt. Once set this bit will remain high until the interrupt has been cleared via MS_INF_CKS_ERR_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

MS_INF_CKS_ERR_ST	Description
0 «	No change in MPEG source infoframe checksum error
1	An MPEG source infoframe checksum error has triggered this interrupt

SPD_INF_CKS_ERR_ST, Addr 40 (IO), *Address 0x89[6] (Read Only)*

Latched status of SPD Infoframe Checksum Error interrupt. Once set this bit will remain high until the interrupt has been cleared via SPD_INF_CKS_ERR_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

SPD_INF_CKS_ERR_ST	Description
0 «	No change in SPD infoframe checksum error
1	An SPD infoframe checksum error has triggered this interrupt

AUD_INF_CKS_ERR_ST, Addr 40 (IO), *Address 0x89[5] (Read Only)*

Latched status of Audio Infoframe Checksum Error interrupt. Once set this bit will remain high until the interrupt has been cleared via AUDIO_INF_CKS_ERR_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

AUD_INF_CKS_ERR_ST	Description
0 «	No change in Audio infoframe checksum error
1	An Audio infoframe checksum error has triggered this interrupt

AVI_INF_CKS_ERR_ST, Addr 40 (IO), *Address 0x89[4] (Read Only)*

Latched status of AVI Infoframe Checksum Error interrupt. Once set this bit will remain high until the interrupt has been cleared via AVI_INF_CKS_ERR_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

AVI_INF_CKS_ERR_ST	Description
0 «	No change in AVI infoframe checksum error
1	An AVI infoframe checksum error has triggered this interrupt

AKSV_UPDATE_A_ST, Addr 40 (IO), *Address 0x89[1] (Read Only)*

Latched status of Port A AKSV Update Interrupt. Once set this bit will remain high until the interrupt has been cleared via AKSV_UPDATE_A_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

AKSV_UPDATE_A_ST	Description
0 «	AKSV_UPDATE_A_RAW has not changed. An interrupt has not been generated.
1	TAKSV_UPDATE_A_RAW has changed. An interrupt has been generated.

AKSV_UPDATE_B_ST, Addr 40 (IO), *Address 0x89[0] (Read Only)*

Latched status of Port B AKSV Update Interrupt. Once set this bit will remain high until the interrupt has been cleared via AKSV_UPDATE_A_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

AKSV_UPDATE_B_ST	Description
0 «	AKSV_UPDATE_B_RAW has not changed. An interrupt has not been generated.
1	TAKSV_UPDATE_B_RAW has changed. An interrupt has been generated.

HDMI Edg Status 5 register consists of the following fields.

BG_MEAS_DONE_ST, Addr 40 (IO), *Address 0x8E[1] (Read Only)*

Latched status of Background Port Measurement completed interrupt. Once set this bit will remain high until the interrupt has been cleared via BG_MEAS_DONE_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit.

Function

BG_MEAS_DONE_ST	Description
0 «	Measurements of TMDS frequency and video parameters of background port not finished or not requested.
1	Measurements of TMDS frequency and video parameters of background port are ready

VS_INF_CKS_ERR_ST, Addr 40 (IO), *Address 0x8E[0] (Read Only)*

Latched status of MPEG Source Infoframe Checksum Error interrupt. Once set this bit will remain high until the interrupt has been cleared via MS_INF_CKS_ERR_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit

Function

VS_INF_CKS_ERR_ST	Description
0 «	No change in VS infoframe checksum error
1	A VS infoframe checksum error has triggered this interrupt

CEC_STATUS1_INT_STATUS register consists of the following fields.

CEC_RX_RDY2_ST

CEC_RX_RDY1_ST
CEC_RX_RDY0_ST
CEC_TX_RETRY_TIMEOUT_ST
CEC_TX_ARBITRATION_LOST_ST
CEC_TX_READY_ST

CEC_STATUS2_INT_STATUS register consists of the following fields.

CEC_INT_WAKE_OPCODE7_ST, Addr 40 (IO), *Address 0x98[7] (Read Only)*

Latched status of CEC_WAKE_OPCODE7_RAW. Once set this bit will remain high until the interrupt has been cleared via CEC_INT_WAKE_OPCODE7_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit.

Function

CEC_INT_WAKE_OPCODE7_ST	Description
0 «	WAKE_OPCODE7 not received.
1	WAKE_OPCODE7 received.

CEC_INT_WAKE_OPCODE6_ST, Addr 40 (IO), *Address 0x98[6] (Read Only)*

Latched status of CEC_WAKE_OPCODE6_RAW. Once set this bit will remain high until the interrupt has been cleared via CEC_INT_WAKE_OPCODE6_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit.

Function

CEC_INT_WAKE_OPCODE6_ST	Description
0 «	WAKE_OPCODE6 not received.
1	WAKE_OPCODE6 received.

CEC_INT_WAKE_OPCODE5_ST, Addr 40 (IO), *Address 0x98[5] (Read Only)*

Latched status of CEC_WAKE_OPCODE5_RAW. Once set this bit will remain high until the interrupt has been cleared via CEC_INT_WAKE_OPCODE5_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit.

Function

CEC_INT_WAKE_OPCODE5_ST	Description
0 «	WAKE_OPCODE5 not received.
1	WAKE_OPCODE5 received.

CEC_INT_WAKE_OPCODE4_ST, Addr 40 (IO), *Address 0x98[4] (Read Only)*

Latched status of CEC_WAKE_OPCODE4_RAW. Once set this bit will remain high until the interrupt has been cleared via CEC_INT_WAKE_OPCODE4_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit.

Function

CEC_INT_WAKE_OPCODE4_ST	Description
0 «	WAKE_OPCODE4 not received.
1	WAKE_OPCODE4 received.

CEC_INT_WAKE_OPCODE3_ST, Addr 40 (IO), *Address 0x98[3] (Read Only)*

Latched status of CEC_WAKE_OPCODE3_RAW. Once set this bit will remain high until the interrupt has been cleared via CEC_INT_WAKE_OPCODE3_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit.

Function

CEC_INT_WAKE_OPCODE3_ST	Description
0 «	WAKE_OPCODE3 not received.
1	WAKE_OPCODE3 received.

CEC_INT_WAKE_OPCODE2_ST, Addr 40 (IO), *Address 0x98[2] (Read Only)*

Latched status of CEC_WAKE_OPCODE2_RAW. Once set this bit will remain high until the interrupt has been cleared via CEC_INT_WAKE_OPCODE2_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit.

Function

CEC_INT_WAKE_OPCODE2_ST	Description
0 «	WAKE_OPCODE2 not received.
1	WAKE_OPCODE2 received.

CEC_INT_WAKE_OPCODE1_ST, Addr 40 (IO), *Address 0x98[1] (Read Only)*

Latched status of CEC_WAKE_OPCODE1_RAW. Once set this bit will remain high until the interrupt has been cleared via CEC_INT_WAKE_OPCODE1_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit.

Function

CEC_INT_WAKE_OPCODE1_ST	Description
0 «	WAKE_OPCODE1 not received.
1	WAKE_OPCODE1 received.

CEC_INT_WAKE_OPCODE0_ST, Addr 40 (IO), *Address 0x98[0] (Read Only)*

Latched status of CEC_WAKE_OPCODE0_RAW. Once set this bit will remain high until the interrupt has been cleared via CEC_INT_WAKE_OPCODE0_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit.

Function

CEC_INT_WAKE_OPCODE0_ST	Description
0 «	WAKE_OPCODE0 not received.
1	WAKE_OPCODE0 received.

SDP_INTERRUPT_STATUS register consists of the following fields.

SDP_STD_CHANGED_ST, Addr 40 (IO), *Address 0x9D[3] (Read Only)*

Latched status for SDP Standard Changed interrupt signal. Once set this bit will remain high until the interrupt has been cleared via SDP_STD_CHANGED_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit.

Function

SDP_STD_CHANGED_ST	Description
0 «	No change. An interrupt has not been generated from this register.
1	SDP_STD_CHANGED_RAW has changed and generated an interrupt.

SDP_BURST_LOCKED_ST, Addr 40 (IO), *Address 0x9D[1] (Read Only)*

Latched status for SDP Burst Lock interrupt signal. Once set this bit will remain high until the interrupt has been cleared via SDP_BURST_LOCK_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit.

Function

SDP_BURST_LOCKED_ST	Description
0 «	No change. An interrupt has not been generated from this register.
1	SDP_BURST_LOCKED_RAW has changed and generated an interrupt.

SDP_VIDEO_DETECTED_ST, Addr 40 (IO), *Address 0x9D[0] (Read Only)*

Latched status for SDP Video Detected interrupt signal. Once set this bit will remain high until the interrupt has been cleared via SDP_VIDEO_DETECTED_CLR. This bit is only valid if enabled via the corresponding INT1 or INT2 interrupt mask bit.

Function

SDP_VIDEO_DETECTED_ST	Description
0 «	No change. An interrupt has not been generated from this register.
1	SDP_VIDEO_DETECTED_RAW has changed and generated an interrupt.

16.4.7 Processing Analog Front End Interrupts

The analog front end (AFE) contains eight trilevel slicers. The inputs to trilevel slicers 1 to 4 are the trilevel signals received on pins TRI1 to TRI4 respectively. Pins HS_IN1/TRI5, VS_IN1/TRI6, HS_IN2/TRI7 and VS_IN2/TRI8 can be configured to receive trilevel signals and these inputs are sliced on slicers 5, 6, 7 and 8 respectively.

Each trilevel slicer has two digital outputs, one for each slice level. An edge on any of the outputs from the trilevel slicers will be captured as an interrupt. These interrupts are located in AFE Map, addresses 0x1B and 0x1C. Each interrupt signal has a separate mask and clear signal available to the user. These are located in the AFE Map, addresses 0x17 to 0x1A. Refer to the ADV7842 Software Manual, AFE Map section, for more details.

The 16 outputs from all eight trilevel slicers are OR'ed together and the result is captured as an interrupt in the IO Map. Figure 167 details the recommended method of processing the trilevel interrupts.

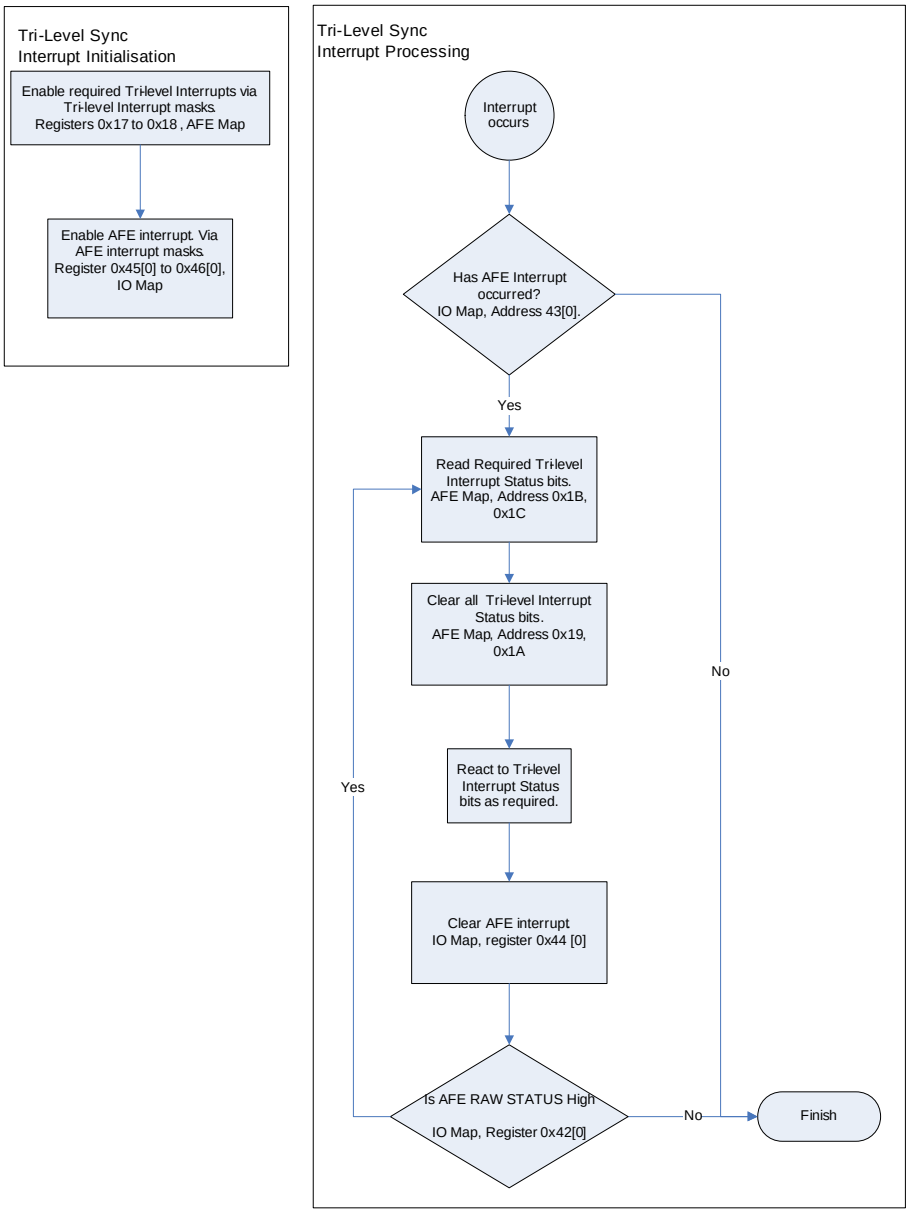


Figure 167 Processing Trilevel Interrupts

APPENDIX A

PCB LAYOUT RECOMMENDATIONS

The ADV7842 is a high precision, high speed, mixed signal device. It is important to have a well designed PCB board in order to achieve the maximum performance from the part. The following sections are a guide for designing a board using the ADV7842.

ANALOG INTERFACE INPUTS

The trace length running into the graphics inputs should be minimized. This is accomplished by placing the ADV7842 as close as possible to the graphic connector. Long input trace lengths are undesirable because they pick up noise from the board and other external sources.

The voltage divider 24 ohm/51 ohm, which acts as a 75 ohm termination (refer to Appendix B), should be placed as close as possible to the ADV7842 chip. Any additional trace length between the termination resistors and the input of the ADV7842 increases the magnitude of reflections, which corrupts the graphics signal. 75 ohm matched impedance traces should be used. Trace impedances other than 75 ohms also increase the chance of reflections.

The ADV7842 has high input bandwidth. While this is desirable for acquiring a high resolution PC graphics signal with fast edges, it means that it also captures high frequency noise that is present. Therefore, it is important to reduce the amount of noise that is coupled to the inputs. The designer should avoid running any digital traces near the analog inputs and ensure signal traces do not run too close together to avoid crosstalk.

The non graphics input should also receive care when being routed on the PCB. Again, track lengths should be kept to a minimum and 75 ohm traces impedances should be used where possible.

The following routing is strongly recommended:

- RGB Graphics – Ain 1, 2, 3
- Component – Ain 4, 5, 6
- SCART (RGB) – Ain 7, 8, 9 (CVBS-Ain10)
- CVBS – Ain11

POWER SUPPLY BYPASSING

It is recommended to bypass each power supply pin with a 0.1 uF and a 10 nF capacitor where possible. The fundamental idea is to have a bypass capacitor within about 0.5 cm of each power pin.

The bypass capacitors should be physically located between the power plane and the power pin. Current should flow from the power plane to the capacitor to the power pin. The power connection should not be made between the capacitor and the power pin. Generally, the best approach is to place a via underneath the 100 nF capacitor pads down to the power plane (refer to Figure 168).

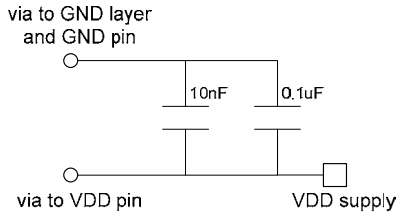


Figure 168: Recommended Power Supply Decoupling

It is particularly important to maintain low noise and good stability of the PVDD (the clock generator supply). Abrupt changes in the PVDD supply can result in similarly abrupt changes in sampling clock phase and frequency. This can be avoided by careful attention to regulation, filtering, and bypassing. It is highly desirable to provide separately regulated and heavily filtered supplies for each of the analog circuitry groups (AVDD, CVDD, TVDD, and PVDD).

Some graphic controllers use substantially different levels of power when active (during active picture time) and when idle (during horizontal and vertical synchronization periods). This can result in a measurable change in the voltage supplied to the analog supply regulator, which can in turn produce changes in the regulated analog supply voltage. This can be mitigated by regulating the analog supply, or at least PVDD, from a separate, cleaner, power source.

It is also recommended to use a single ground plane for the entire board. Repeatedly, experience has shown that the noise performance is the same or better with a single ground plane. Using multiple ground planes can be detrimental because each separate ground plane is smaller and long ground loops can result.

In some cases, using separate ground planes is unavoidable. For those cases, it is recommended to place at least a single ground plane under the ADV7842.It is important to place components wisely because the current loops are much longer when using split ground planes as the current takes the path of least resistance.

Example of a current loop:

Power plane => ADV7842 => digital output trace => digital data receiver => digital ground plane => analog ground plane.

Power Supply Sequencing

Power Up Sequence

The recommended power up sequence of the ADV7842 is as follows:

- 3.3 V supplies
- 2.5 V supply (applies only if using DDR memory)
- 1.8 V supplies

Notes:

- Reset should be held low while the supplies are being powered up
- 3.3 V supplies should be powered up first
- 2.5 V supply should be powered after the 3.3 V supplies are established but before the 1.8 V supplies
- 1.8 V supplies should be powered up last

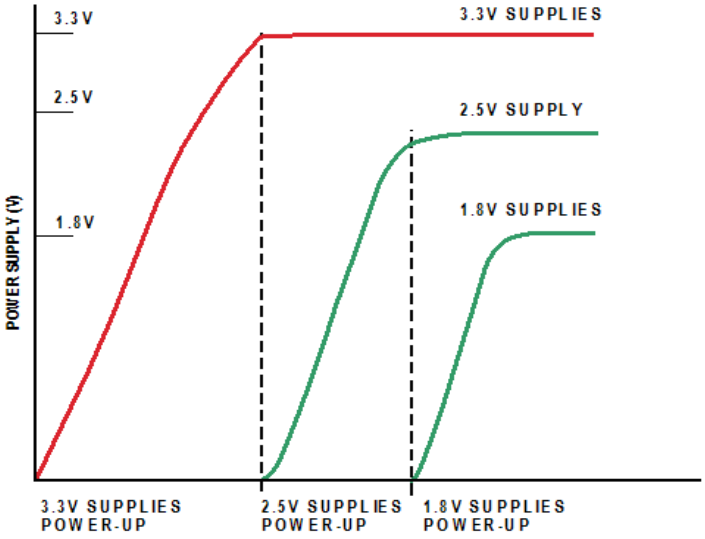


Figure 169: Recommended Power Up Sequence

Alternatively, the ADV7842 may be powered up as follows:

- All supplies asserted simultaneously

Note: In this case, care must be taken to ensure that a lower rated supply does not go above a higher rated supply level, as the supplies are being established.

16.4.7.1 Power Down Sequence

The ADV7842 supplies may be de asserted simultaneously as long as a higher rated supply does not go below a lower rated supply.

DIGITAL OUTPUTS (DATA AND CLOCKS)

The trace length that the digital outputs have to drive should be minimized. Longer traces have higher capacitance, which requires more current that can cause more internal digital noise. Shorter traces reduce the possibility of reflections.

Adding a series resistor of value between 50 to 200 ohms can suppress reflections, reduce EMI, and reduce the current spikes inside the ADV7842. If series resistors are used, they should be placed as close as possible to the ADV7842 pins and the trace impedance for these signals should match that of the termination resistors selected.

If possible, the capacitance that each of the digital outputs drives should be limited to is less than 15 pF. This can be accomplished easily by keeping traces short and by connecting the outputs to only one device. Loading the outputs with excessive capacitance increases the current transients inside the ADV7842, creating more digital noise on its power supplies.

DIGITAL INPUTS

The following digital inputs on the ADV7842 are 3.3 V inputs that are 5.0 V tolerant:

- HS_IN1/TRI5
- VS_IN1/TRI6
- HS_IN2/TRI7
- VS_IN2/TRI8
- DDCA_SCL
- DDCA_SDL
- DDCB_SCL
- DDCB_SDA
- VGA_SCL
- VGA_SDA

Any noise that gets onto the HS and VS inputs trace will add jitter to the system. Therefore, the trace length should be minimized; and digital or other high frequency traces should not be run near it.

XTAL AND LOAD CAP VALUE SELECTION

The ADV7842 uses a 28.6363MHz crystal. Figure 170 shows an example of a reference clock circuit for the ADV7842. Special care must be taken when using a crystal circuit to generate the reference clock for the ADV7842. Small variations in reference clock frequency can cause auto detection issues and impair the ADV7842 performance.

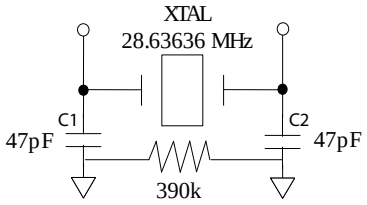


Figure 170: Crystal Circuit

These guidelines are followed to ensure correct operation:

- Use the correct frequency crystal, which is 28.6363 MHz. Tolerance should be 50 ppm or better.
- Know the C_{load} for the crystal part number selected. The value of capacitors C1 and C2 must be matched to the C_{load} for the specific crystal part number in the user's system.

  To find C1 and C2, use the following formula:

  $$C1 = C2 = 2(C_{load} - C_{stray}) - C_{pg}$$

  where C_{stray} is usually 2 to 3 pF, depending on board traces and C_{pg} (pin-to-ground-capacitance) is 4 pF for the ADV7842.

  Example:

  C_{load} = 30 pF, $C1$ = 50 pF, $C2$ = 50 pF (in this case, 47 pF is the nearest real-life cap value to 50 pF)

APPENDIX B

ADV7842 TYPICAL CONNECTION DIAGRAMS.

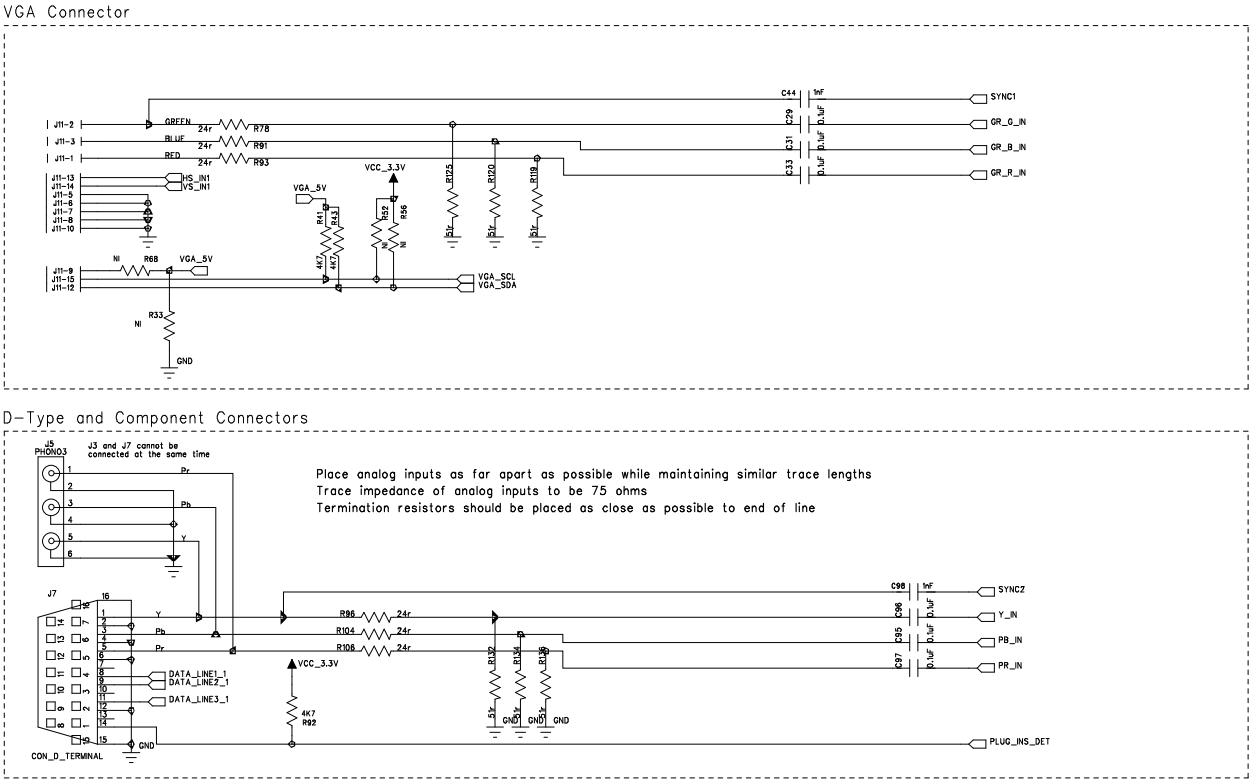


Figure 171: ADV7842 Analog Inputs (Graphics and Component)

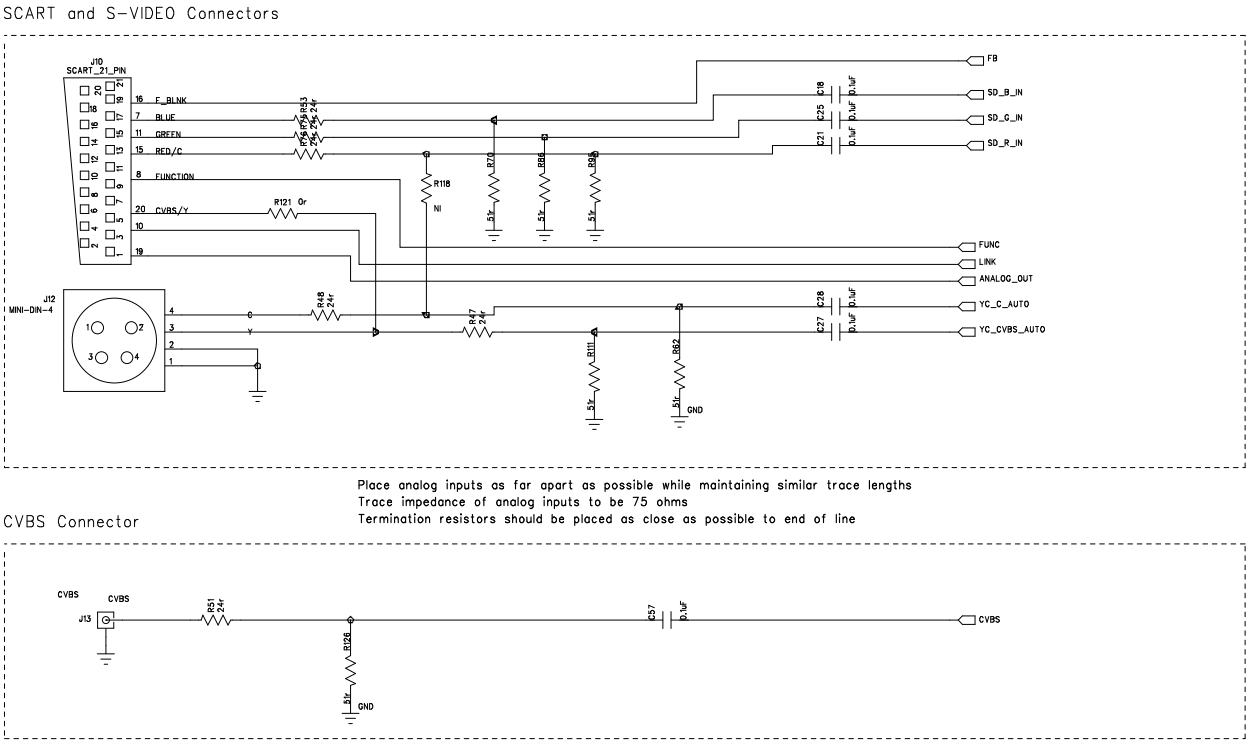


Figure 172: ADV7842 Analog Inputs (CVBS, S-Video and SCART)

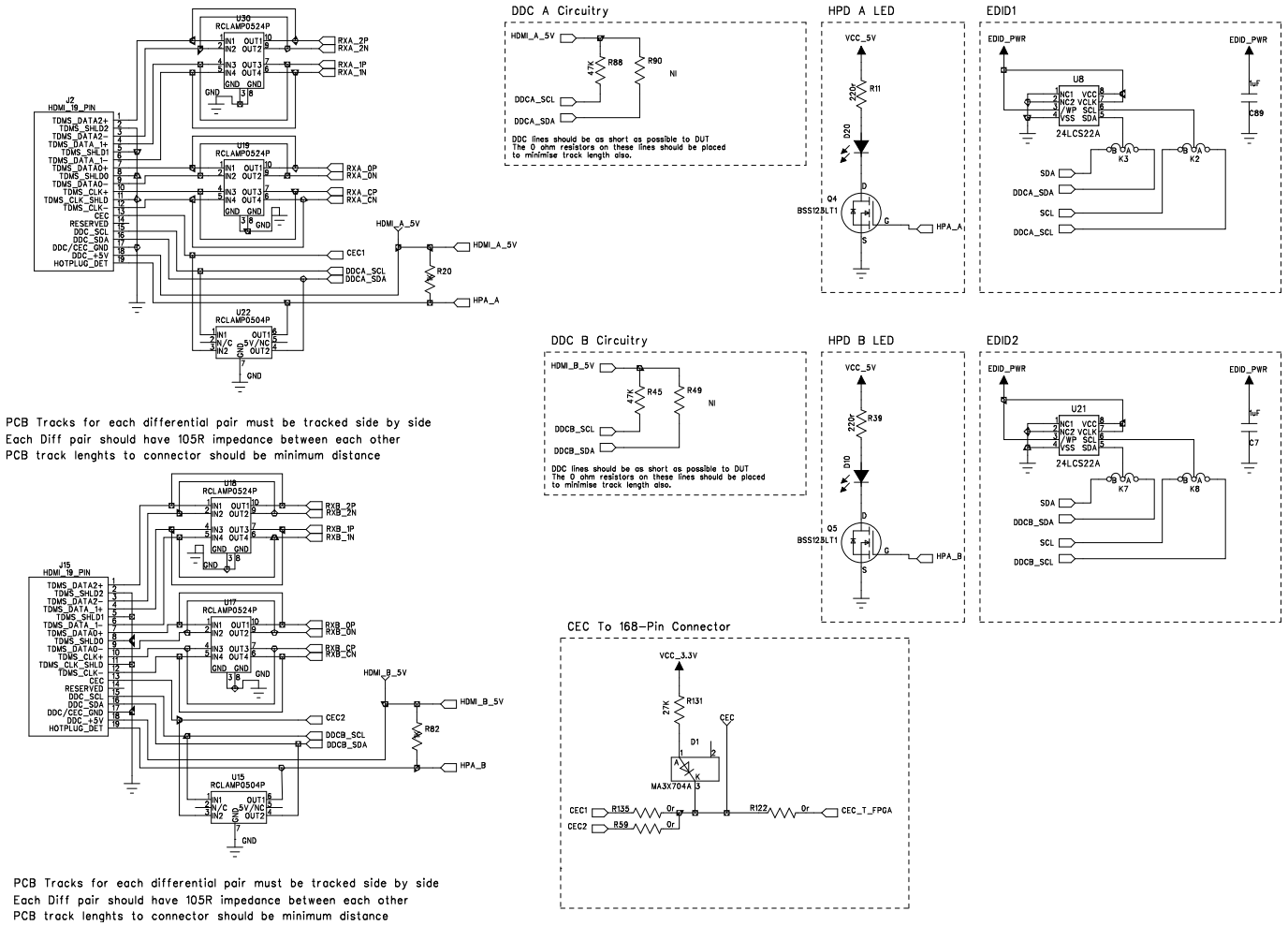


Figure 173: ADV7842 HDMI Inputs (Port A and Port B)

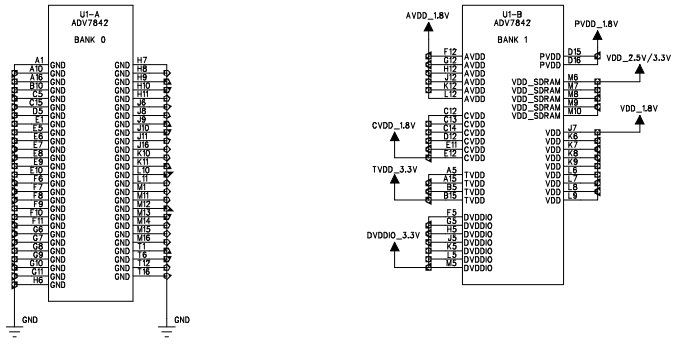


Figure 174: ADV7842 Power and Ground Connections

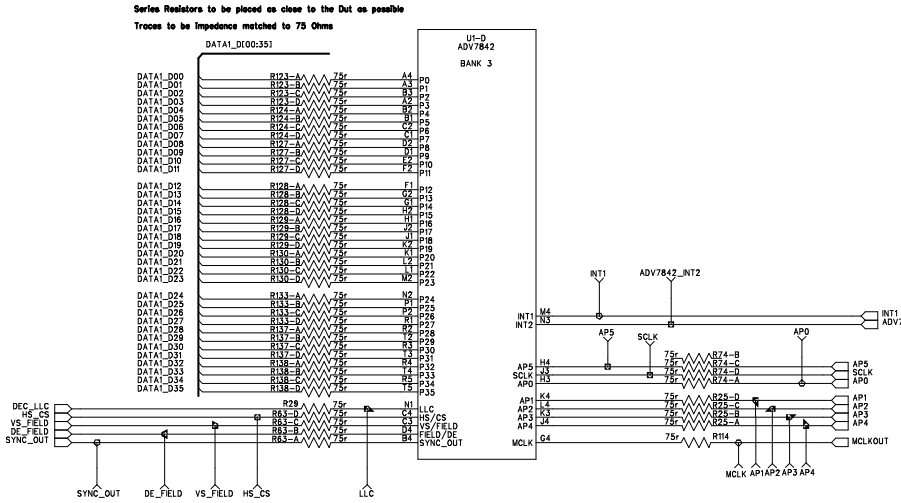


Figure 175: ADV7842 Pixel and Output Pins

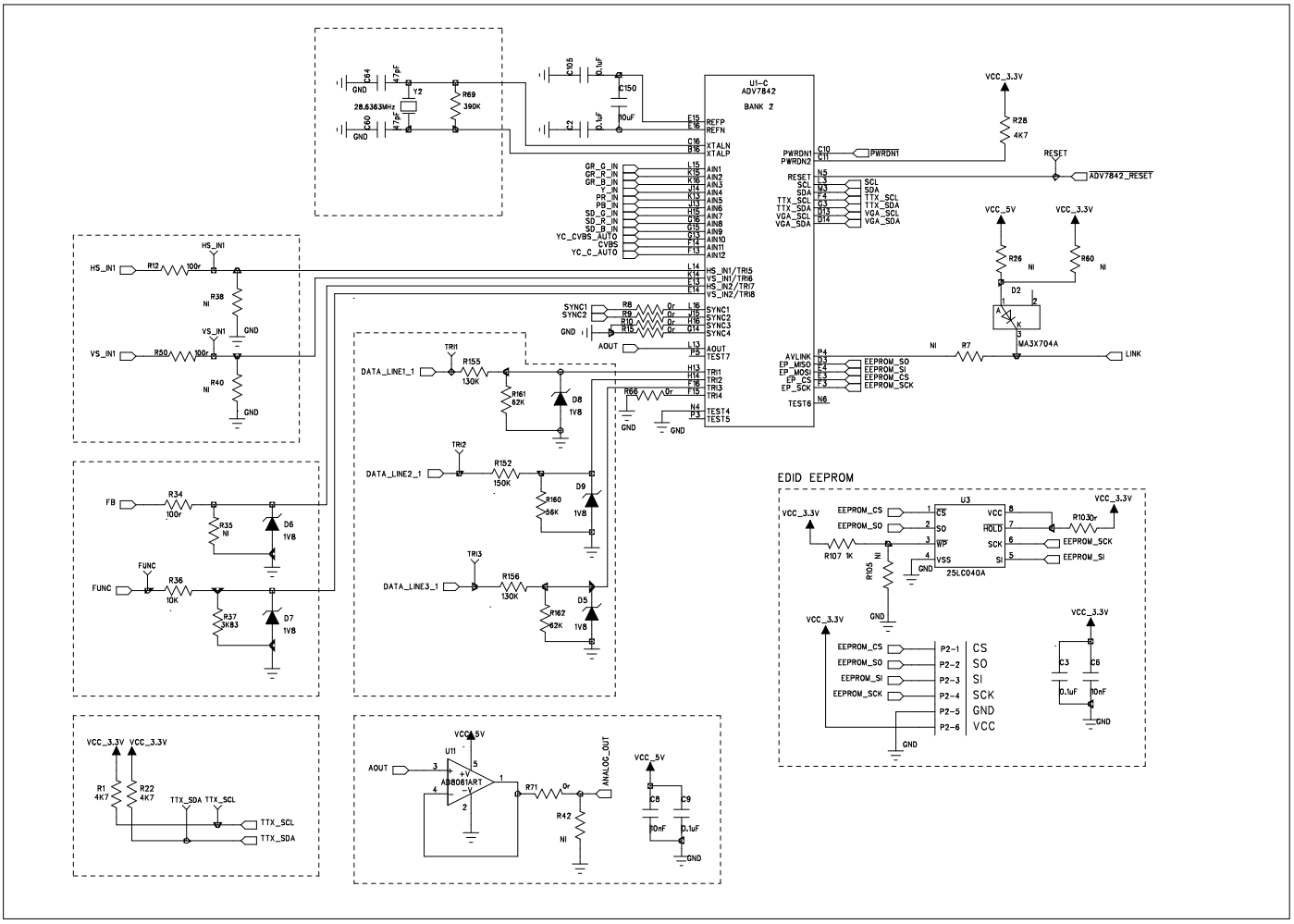


Figure 176: ADV7842 Input Pin Connections

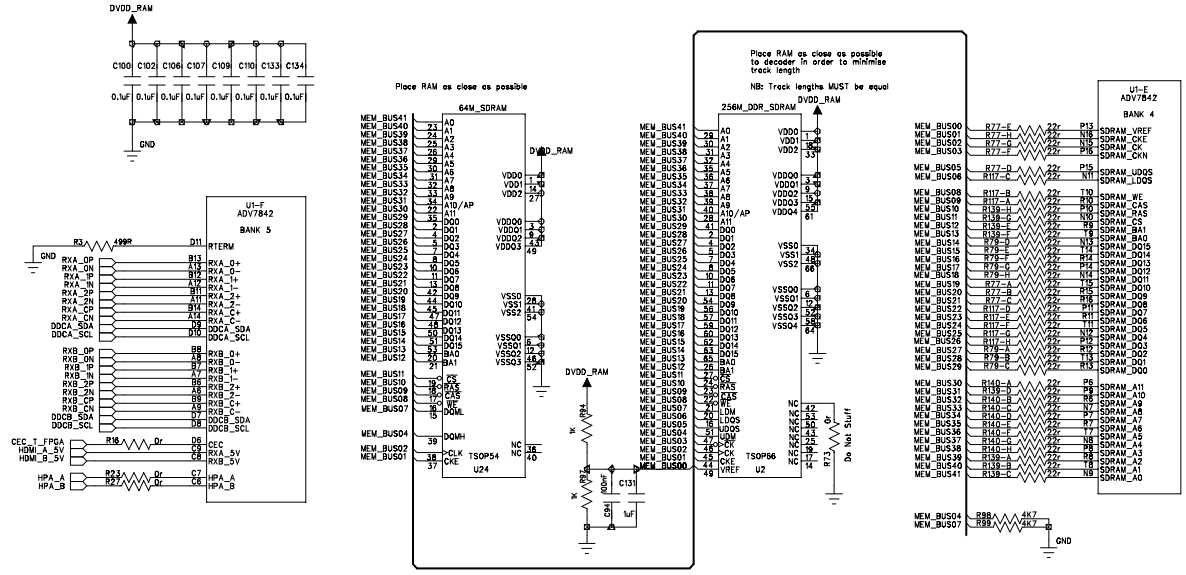


Figure 177: ADV7842 HDMI Input and Memory Connections

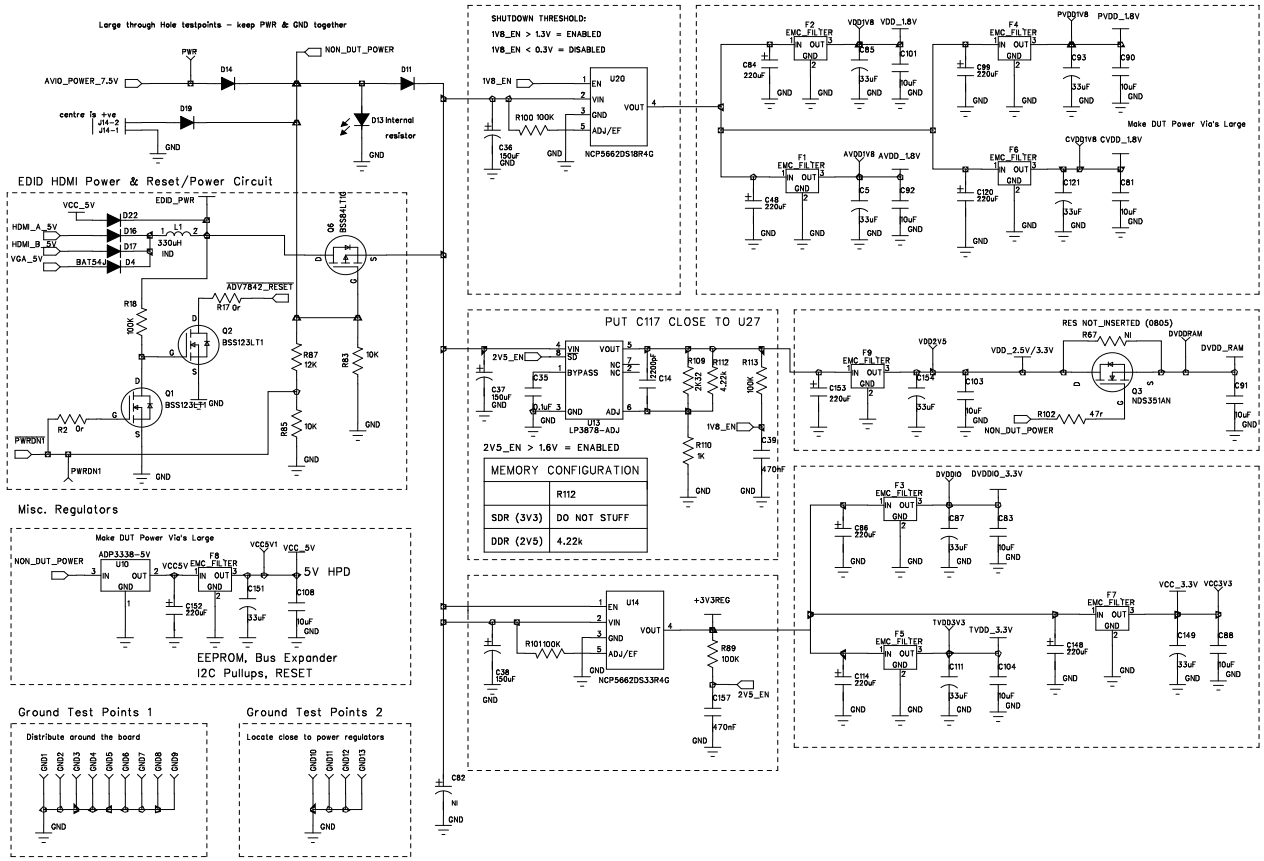


Figure 178: Example Power Supply Circuitry for ADV7842

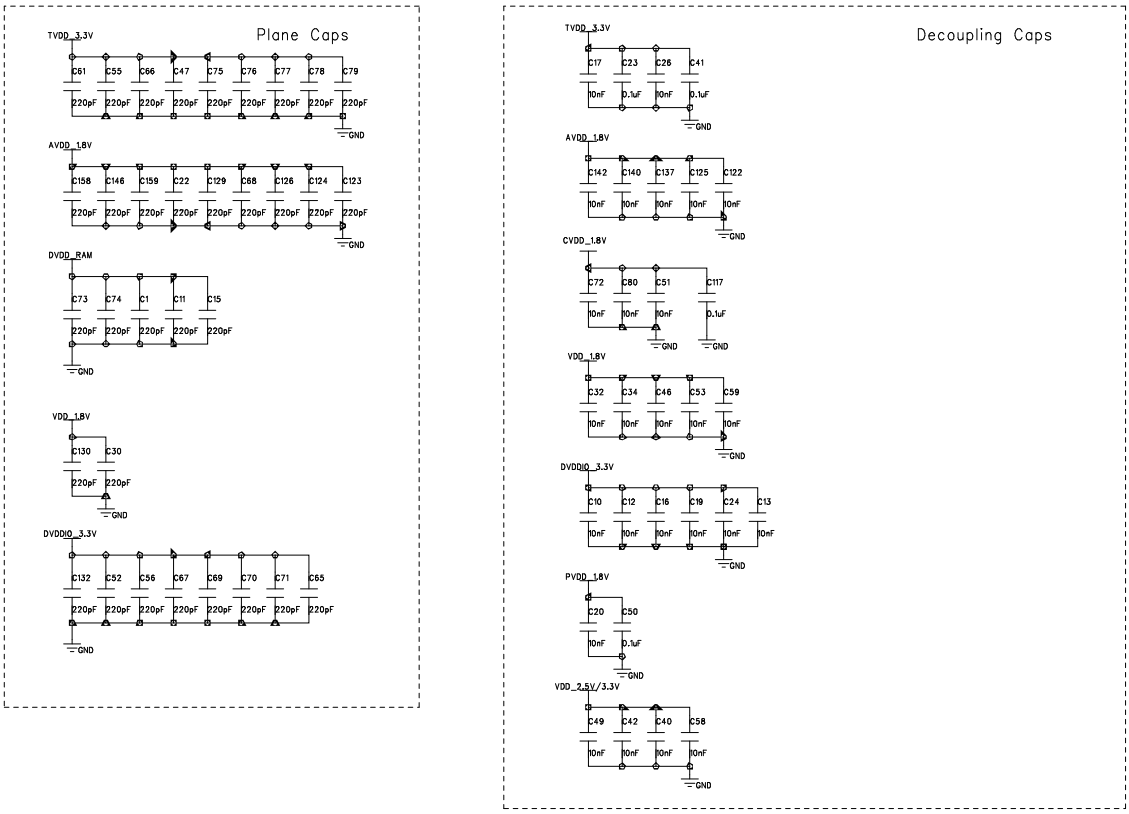


Figure 179: ADV7842 Power Supply decoupling Capacitors

APPENDIX C

RECOMMENDED UNUSED PIN CONFIGURATIONS

Table 100: Recommended Configuration of Unused Pins

Location	Mnemonic	Type	Recommended Configuration if Not Used
A1	GND	Ground	Ground.
A2	P3	Digital video output	Float this pin.
A3	P1	Digital video output	Float this pin.
A4	P0	Digital video output	Float this pin.
A5	TVDD	Power	This pin is always connected to Terminator supply voltage (3.3 V).
A6	RXB_2-	HDMI input	Float this pin
A7	RXB_1-	HDMI input	Float this pin
A8	RXB_0-	HDMI input	Float this pin
A9	RXB_C-	HDMI input	Float this pin
A10	GND	Ground	Ground.
A11	RXA_2-	HDMI input	Float this pin
A12	RXA_1-	HDMI input	Float this pin.
A13	RXA_0-	HDMI input	Float this pin
A14	RXA_C-	HDMI input	Float this pin
A15	TVDD	Power	This pin is always connected to Terminator supply voltage (3.3 V).
A16	GND	Ground	Ground.
B1	P5	Digital video output	Float this pin.
B2	P4	Digital video output	Float this pin.
B3	P2	Digital video output	Float this pin.
B4	SYNC_OUT	Miscellaneous digital	Float this pin
B5	TVDD	Power	This pin is always connected to Terminator supply voltage (3.3 V).
B6	RXB_2+	HDMI input	Float this pin
B7	RXB_1+	HDMI input	Float this pin
B8	RXB_0+	HDMI input	Float this pin
B9	RXB_C+	HDMI input	Float this pin
B10	GND	Ground	Ground.
B11	RXA_2+	HDMI input	Float this pin
B12	RXA_1+	HDMI input	Float this pin
B13	RXA_0+	HDMI input	Float this pin
B14	RXA_C+	HDMI input	Float this pin
B15	TVDD	Power	This pin is always connected to Terminator supply voltage (3.3 V).
B16	XTALP	Miscellaneous analog	This pin is always connected to 28.63636 MHz crystal.
C1	P7	Digital video output	Float this pin.
C2	P6	Digital video output	Float this pin.
C3	VS/FIELD	Digital video output	Float this pin.
C4	HS/CS	Digital video output	Float this pin
C5	GND	Ground	Ground.
C6	HPA_B	Miscellaneous digital	Float this pin
C7	HPA_A	Miscellaneous digital	Float this pin
C8	RXB_5V	HDMI input	This pin should be tied to 5 V.
C9	RXA_5V	HDMI input	This pin should be tied to 5 V.
C10	$\overline{\text{PWRDN1}}$	Miscellaneous digital	Tie to 3.3V via 4.7k resistor
C11	Test8	Test pin	Tie to 3.3V via 4.7k resistor
C12	CVDD	Power	This pin is always connected to Comparator supply voltage (1.8 V).
C13	CVDD	Power	This pin is always connected to Comparator supply voltage (1.8 V).
C14	CVDD	Power	This pin is always connected to Comparator supply voltage (1.8 V).

Location	Mnemonic	Type	Recommended Configuration if Not Used
C15	GND	Ground	Ground.
C16	XTALN	Miscellaneous analog	This pin is always connected to 28.63636 MHz crystal.
D1	P9	Digital video output	Float this pin.
D2	P8	Digital video output	Float this pin.
D3	EP_MISO	Digital output	Float this pin.
D4	FIELD/DE	Miscellaneous digital	Float this pin.
D5	GND	Ground	Ground.
D6	CEC	Digital input/output	Float this pin.
D7	DDCB_SDA	HDMI input	Float this pin
D8	DDCB_SCL	HDMI input	Connect this pin to ground via a 10k ohm resistor.
D9	DDCA_SDA	HDMI input	Float this pin
D10	DDCA_SCL	HDMI input	Connect this pin to ground via a 10k ohm resistor.
D11	RTERM	Miscellaneous analog	A 500 Ω resistor between this pin and GND should be always be used.
D12	CVDD	Power	This pin is always connected to Comparator supply voltage (1.8 V).
D13	VGA_SCL	Miscellaneous digital	Connect this pin to ground via a 10k ohm resistor.
D14	VGA_SDA	Miscellaneous digital	Float this pin
D15	PVDD	Power	This pin is always connected to PLL supply voltage (1.8 V).
D16	PVDD	Power	This pin is always connected to PLL supply voltage (1.8 V).
E1	GND	Ground	Ground.
E2	P10	Digital video output	Float this pin.
E3	$\overline{\text{EP\_CS}}$	Digital output	Float this pin.
E4	EP_MOSI	Digital input	Float this pin.
E5	GND	Ground	Ground.
E6	GND	Ground	Ground.
E7	GND	Ground	Ground.
E8	GND	Ground	Ground.
E9	GND	Ground	Ground.
E10	GND	Ground	Ground.
E11	CVDD	Power	This pin is always connected to Comparator supply voltage (1.8 V).
E12	CVDD	Power	This pin is always connected to Comparator supply voltage (1.8 V).
E13	HS_IN2/TRI7	Miscellaneous analog	Float this pin
E14	VS_IN2/TRI8	Miscellaneous analog	Float this pin.
E15	REFP	Miscellaneous analog	Always used as internal voltage reference output.
E16	REFN	Miscellaneous analog	Always as the internal voltage reference output.
F1	P12	Digital video output	Float this pin.
F2	P11	Digital video output	Float this pin.
F3	EP_SCK	Digital output	Float this pin.
F4	TTX_SCL	Miscellaneous digital	Connect this pin to ground via a 10k ohm resistor.
F5	DVDDIO	Power	This pin is always connected to the Digital IO supply voltage (3.3 V).
F6	GND	Ground	Ground.
F7	GND	Ground	Ground.
F8	GND	Ground	Ground.
F9	GND	Ground	Ground.
F10	GND	Ground	Ground.
F11	GND	Ground	Ground.
F12	AVDD	Power	This pin is always connected to Analog supply voltage (1.8 V).
F13	AIN12	Analog video input	Float this pin
F14	AIN11	Analog video input	Float this pin
F15	TRI4	Miscellaneous analog	Float this pin.
F16	TRI3	Miscellaneous analog	Float this pin.
G1	P14	Digital video output	Float this pin.

Location	Mnemonic	Type	Recommended Configuration if Not Used
G2	P13	Digital video output	Float this pin.
G3	TTX_SDA	Miscellaneous digital	Float this pin
G4	MCLK	Miscellaneous	Float this pin.
G5	DVDDIO	Power	This pin is always connected to the Digital IO supply voltage (3.3 V).
G6	GND	Ground	Ground.
G7	GND	Ground	Ground.
G8	GND	Ground	Ground.
G9	GND	Ground	Ground.
G10	GND	Ground	Ground.
G11	GND	Ground	Ground.
G12	AVDD	Power	This pin is always connected to Analog supply voltage (1.8 V).
G13	AIN10	Analog video input	Float this pin
G14	SYNC4	Miscellaneous analog	Float this pin
G15	AIN9	Analog video input	Float this pin
G16	AIN8	Analog video input	Float this pin
H1	P16	Digital video output	Float this pin.
H2	P15	Digital video output	Float this pin.
H3	AP0	Miscellaneous	Float this pin.
H4	AP5	Miscellaneous	Float this pin.
H5	DVDDIO	Power	This pin is always connected to the Digital IO supply voltage (3.3 V).
H6	GND	Ground	Ground.
H7	GND	Ground	Ground.
H8	GND	Ground	Ground.
H9	GND	Ground	Ground.
H10	GND	Ground	Ground.
H11	GND	Ground	Ground.
H12	AVDD	Power	This pin is always connected to Analog supply voltage (1.8 V).
H13	TRI1	Miscellaneous analog	Float this pin
H14	TRI2	Miscellaneous analog	Float this pin
H15	AIN7	Analog video input	Float this pin
H16	SYNC3	Miscellaneous analog	Float this pin
J1	P18	Digital video output	Float this pin.
J2	P17	Digital video output	Float this pin.
J3	SCLK	Miscellaneous digital	Float this pin.
J4	AP4	Miscellaneous	Float this pin.
J5	DVDDIO	Power	This pin is always connected to the Digital IO supply voltage (3.3 V).
J6	GND	Ground	Ground.
J7	VDD	Power	This pin is always connected to Digital core supply voltage (1.8 V).
J8	GND	Ground	Ground.
J9	GND	Ground	Ground.
J10	GND	Ground	Ground.
J11	GND	Ground	Ground.
J12	AVDD	Power	This pin is always connected to Analog supply voltage (1.8 V).
J13	AIN6	Analog video input	Float this pin
J14	AIN4	Analog video input	Float this pin
J15	SYNC2	Miscellaneous analog	Float this pin
J16	GND	Ground	Ground.
K1	P20	Digital video output	Float this pin.
K2	P19	Digital video output	Float this pin.
K3	AP3	Miscellaneous	Float this pin.
K4	AP1	Miscellaneous	Float this pin.
K5	DVDDIO	Power	This pin is always connected to the Digital IO supply voltage (3.3 V).

Location	Mnemonic	Type	Recommended Configuration if Not Used
K6	VDD	Power	This pin is always connected to Digital core supply voltage (1.8 V).
K7	VDD	Power	This pin is always connected to Digital core supply voltage (1.8 V).
K8	VDD	Power	This pin is always connected to Digital core supply voltage (1.8 V).
K9	VDD	Power	This pin is always connected to Digital core supply voltage (1.8 V).
K10	GND	Ground	Ground.
K11	GND	Ground	Ground.
K12	AVDD	Power	This pin is always connected to Analog supply voltage (1.8 V).
K13	AIN5	Analog video input	Float this pin
K14	VS_IN1/TRI6	Miscellaneous analog	Float this pin.
K15	AIN2	Analog video input	Float this pin
K16	AIN3	Analog video input	Float this pin
L1	P22	Digital video output	Float this pin.
L2	P21	Digital video output	Float this pin.
L3	SCL	Miscellaneous digital	This pin is always connected to the I2C clock line of a control processor.
L4	AP2	Miscellaneous	Float this pin.
L5	DVDDIO	Power	This pin is always connected to the Digital IO supply voltage (3.3 V).
L6	VDD	Power	This pin is always connected to Digital core supply voltage (1.8 V).
L7	VDD	Power	This pin is always connected to Digital core supply voltage (1.8 V).
L8	VDD	Power	This pin is always connected to Digital core supply voltage (1.8 V).
L9	VDD	Power	This pin is always connected to Digital core supply voltage (1.8 V).
L10	GND	Ground	Ground.
L11	GND	Ground	Ground.
L12	AVDD	Power	This pin is always connected to Analog supply voltage (1.8 V).
L13	AOUT	Analog monitor output	Float this pin.
L14	HS_IN1/TRI5	Miscellaneous analog	Float this pin
L15	AIN1	Analog video input	Float this pin
L16	SYNC1	Miscellaneous analog	Float this pin
M1	GND	Ground	Ground.
M2	P23	Digital video output	Float this pin.
M3	SDA	Miscellaneous digital	This pin is always connected to the I2C data line of a control processor.
M4	INT1	Miscellaneous digital	Float this pin.
M5	DVDDIO	Power	This pin is always connected to the Digital IO supply voltage (3.3 V).
M6	VDD_SDRAM	Power	Tie to DVDDIO 3.3V
M7	VDD_SDRAM	Power	Tie to DVDDIO 3.3V
M8	VDD_SDRAM	Power	Tie to DVDDIO 3.3V
M9	VDD_SDRAM	Power	Tie to DVDDIO 3.3V
M10	VDD_SDRAM	Power	Tie to DVDDIO 3.3V
M11	GND	Ground	Ground.
M12	GND	Ground	Ground.
M13	GND	Ground	Ground.
M14	GND	Ground	Ground.
M15	GND	Ground	Ground.
M16	GND	Ground	Ground.
N1	LLC	Digital video output	This pin is always connected to the pixel clock input
N2	P24	Digital video output	Float this pin.
N3	INT2	Miscellaneous digital	Float this pin.
N4	TEST4	Test	This pin should be tied to ground.
N5	$\overline{\text{RESET}}$	Miscellaneous digital	This level of this pin should be controlled by an external processor.
N6	TEST6	Input	Float this pin.
N7	SDRAM_A8	SDRAM interface	Float this pin.

Location	Mnemonic	Type	Recommended Configuration if Not Used
N8	SDRAM_A4	SDRAM interface	Float this pin.
N9	SDRAM_A0	SDRAM interface	Float this pin.
N10	$\overline{\text{SDRAM\_CS}}$	SDRAM interface	Float this pin.
N11	SDRAM_LDQS	SDRAM interface	Tied to ground via a 4k7 resistor.
N12	SDRAM_DQ4	SDRAM interface	Tied to ground via a 4k7 resistor.
N13	SDRAM_DQ15	SDRAM interface	Tied to ground via a 4k7 resistor.
N14	SDRAM_DQ11	SDRAM interface	Tied to ground via a 4k7 resistor.
N15	SDRAM_CK	SDRAM interface	Float this pin.
N16	SDRAM_CKE	SDRAM interface	Float this pin.
P1	P25	Digital video output	Float this pin.
P2	P26	Digital video output	Float this pin.
P3	TEST5	Test	Float this pin
P4	AVLINK	Digital input/output	Float this pin
P5	TEST7	Digital Output	Float this pin.
P6	SDRAM_A11	SDRAM interface	Float this pin.
P7	SDRAM_A7	SDRAM interface	Float this pin.
P8	SDRAM_A3	SDRAM interface	Float this pin.
P9	SDRAM_A10	SDRAM interface	Float this pin.
P10	$\overline{\text{SDRAM\_RAS}}$	SDRAM interface	Float this pin.
P11	SDRAM_DQ7	SDRAM interface	Tied to ground via a 4k7 resistor.
P12	SDRAM_DQ3	SDRAM interface	Tied to ground via a 4k7 resistor.
P13	SDRAM_VREF	SDRAM interface	Tied directly to ground.
P14	SDRAM_DQ12	SDRAM interface	Tied to ground via a 4k7 resistor.
P15	SDRAM_UDQS	SDRAM interface	Tied to ground via a 4k7 resistor.
P16	$\overline{\text{SDRAM\_CK}}$	SDRAM interface	Float this pin.
R1	P27	Digital video output	Float this pin.
R2	P28	Digital video output	Float this pin.
R3	P30	Digital video output	Float this pin.
R4	P32	Digital video output	Float this pin.
R5	P34	Digital video output	Float this pin.
R6	SDRAM_A9	SDRAM interface	Float this pin.
R7	SDRAM_A6	SDRAM interface	Float this pin.
R8	SDRAM_A2	SDRAM interface	Float this pin.
R9	SDRAM_BA1	SDRAM interface	Float this pin.
R10	$\overline{\text{SDRAM\_CAS}}$	SDRAM interface	Float this pin.
R11	SDRAM_DQ6	SDRAM interface	Tied to ground via a 4k7 resistor.
R12	SDRAM_DQ2	SDRAM interface	Tied to ground via a 4k7 resistor.
R13	SDRAM_DQ0	SDRAM interface	Tied to ground via a 4k7 resistor.
R14	SDRAM_DQ13	SDRAM interface	Tied to ground via a 4k7 resistor.
R15	SDRAM_DQ9	SDRAM interface	Tied to ground via a 4k7 resistor.
R16	SDRAM_DQ8	SDRAM interface	Tied to ground via a 4k7 resistor.
T1	GND	Ground	Ground.
T2	P29	Digital video output	Float this pin.
T3	P31	Digital video output	Float this pin.
T4	P33	Digital video output	Float this pin.
T5	P35	Digital video output	Float this pin.
T6	GND	Ground	Ground.
T7	SDRAM_A5	SDRAM interface	Float this pin.
T8	SDRAM_A1	SDRAM interface	Float this pin.
T9	SDRAM_BA0	SDRAM interface	Float this pin
T10	$\overline{\text{SDRAM\_WE}}$	SDRAM interface	Float this pin.
T11	SDRAM_DQ5	SDRAM interface	Tied to ground via a 4k7 resistor.

Location	Mnemonic	Type	Recommended Configuration if Not Used
T12	GND	Ground	Ground.
T13	SDRAM_DQ1	SDRAM interface	Tied to ground via a 4k7 resistor.
T14	SDRAM_DQ14	SDRAM interface	Tied to ground via a 4k7 resistor.
T15	SDRAM_DQ10	SDRAM interface	Tied to ground via a 4k7 resistor.
T16	GND	Ground	Ground.

APPENDIX D

PIXEL OUTPUT FORMATS

Table 101: SDR 4:2:2 Output Modes

	SDR 4:2:2				
OP_FORMAT_SEL[7:0]	**0x0**	**0x1**	**0x2**	**0x6**	**0x0A**
Pixel Output	**8-Bit SDR ITU-R BT.656 Mode 0**	**10-Bit SDR ITU-R BT.656 Mode 0**	**12-Bit SDR ITU-R BT.656 Mode 0**	**12-Bit SDR ITU-R BT.656 Mode 1**	**12-Bit SDR ITU-R BT.656 Mode 2**
P35	High-Z	High-Z	High-Z	High-Z	Y3, Cb3, Cr3
P34	High-Z	High-Z	High-Z	High-Z	Y2, Cb2, Cr2
P33	High-Z	High-Z	High-Z	High-Z	Y1, Cb1, Cr1
P32	High-Z	High-Z	High-Z	High-Z	Y0, Cb0, Cr0
P31	High-Z	High-Z	High-Z	High-Z	High-Z
P30	High-Z	High-Z	High-Z	High-Z	High-Z
P29	High-Z	High-Z	High-Z	Y1, Cb1, Cr1	High-Z
P28	High-Z	High-Z	High-Z	Y0, Cb0, Cr0	High-Z
P27	High-Z	High-Z	High-Z	High-Z	High-Z
P26	High-Z	High-Z	High-Z	High-Z	High-Z
P25	High-Z	High-Z	High-Z	High-Z	High-Z
P24	High-Z	High-Z	High-Z	High-Z	High-Z
P23	Y7, Cb7, Cr7	Y9, Cb9, Cr9	Y11, Cb11, Cr11	Y11, Cb11, Cr11	Y11, Cb11, Cr11
P22	Y6, Cb6, Cr6	Y8, Cb8, Cr8	Y10, Cb10, Cr10	Y10, Cb10, Cr10	Y10, Cb10, Cr10
P21	Y5, Cb5, Cr5	Y7, Cb7, Cr7	Y9, Cb9, Cr9	Y9, Cb9, Cr9	Y9, Cb9, Cr9
P20	Y4, Cb4, Cr4	Y6, Cb6, Cr6	Y8, Cb8, Cr8	Y8, Cb8, Cr8	Y8, Cb8, Cr8
P19	Y3, Cb3, Cr3	Y5, Cb5, Cr5	Y7, Cb7, Cr7	Y7, Cb7, Cr7	Y7, Cb7, Cr7
P18	Y2, Cb2, Cr2	Y4, Cb4, Cr4	Y6, Cb6, Cr6	Y6, Cb6, Cr6	Y6, Cb6, Cr6
P17	Y1, Cb1, Cr1	Y3, Cb3, Cr3	Y5, Cb5, Cr5	Y5, Cb5, Cr5	Y5, Cb5, Cr5
P16	Y0, Cb0, Cr0	Y2, Cb2, Cr2	Y4, Cb4, Cr4	Y4, Cb4, Cr4	Y4, Cb4, Cr4
P15	High-Z	Y1, Cb1, Cr1	Y3, Cb3, Cr3	Y3, Cb3, Cr3	High-Z
P14	High-Z	Y0, Cb0, Cr0	Y2, Cb2, Cr2	Y2, Cb2, Cr2	High-Z
P13	High-Z	High-Z	Y1, Cb1, Cr1	High-Z	High-Z
P12	High-Z	High-Z	Y0, Cb0, Cr0	High-Z	High-Z
P11	High-Z	High-Z	High-Z	High-Z	High-Z
P10	High-Z	High-Z	High-Z	High-Z	High-Z
P09	High-Z	High-Z	High-Z	High-Z	High-Z
P08	High-Z	High-Z	High-Z	High-Z	High-Z
P07	High-Z	High-Z	High-Z	High-Z	High-Z
P06	High-Z	High-Z	High-Z	High-Z	High-Z
P05	High-Z	High-Z	High-Z	High-Z	High-Z
P04	High-Z	High-Z	High-Z	High-Z	High-Z
P03	High-Z	High-Z	High-Z	High-Z	High-Z
P02	High-Z	High-Z	High-Z	High-Z	High-Z
P01	High-Z	High-Z	High-Z	High-Z	High-Z
P00	High-Z	High-Z	High-Z	High-Z	High-Z

Table 102: SDR 4:4:4 Output Modes

	SDR 4:4:4							
OP_FORMAT_SEL[7:0]	**0x40**	**0x41**	**0x42**	**0x46**	**0x4C**	**0x50**	**0x51**	**0x52**
Pixel Output	**24-Bit SDR 4:4:4 Mode 0**	**30-Bit SDR 4:4:4 Mode 0**	**36-Bit SDR 4:4:4 Mode 0**	**36-Bit SDR 4:4:4 Mode 1**	**24-Bit SDR 4:4:4 Mode 3**	**24-Bit SDR 4:4:4 Mode 4**	**30-Bit SDR 4:4:4 Mode 4**	**36-Bit SDR 4:4:4 Mode 4**
P35	R7	R9	R11	R9	High-Z	G3	G5	G7
P34	R6	R8	R10	R8	High-Z	G2	G4	G6
P33	R5	R7	R9	R7	R5	R5	R7	R9
P32	R4	R6	R8	R6	R4	R4	R6	R8
P31	R3	R5	R7	R5	R3	R3	R5	R7
P30	R2	R4	R6	R4	R2	R2	R4	R6
P29	R1	R3	R5	R3	R1	R1	R3	R5
P28	R0	R2	R4	R2	R0	R0	R2	R4
P27	High-Z	R1	R3	R1	High-Z	High-Z	High-Z	B1
P26	High-Z	R0	R2	R0	High-Z	High-Z	High-Z	G0
P25	High-Z	High-Z	R1	G7	High-Z	High-Z	High-Z	R0
P24	High-Z	High-Z	R0	G6	High-Z	High-Z	High-Z	R1
P23	G7	G9	G11	G5	G7	Z	B1	B3
P22	G6	G8	G10	G4	G6	Z	B0	B2
P21	G5	G7	G9	G3	G5	Z	G1	G3
P20	G4	G6	G8	G2	G4	Z	G0	G2
P19	G3	G5	G7	G1	G3	G7	G9	G11
P18	G2	G4	G6	G0	G2	G6	G8	G10
P17	G1	G3	G5	B11	G1	G5	G7	G9
P16	G0	G2	G4	B10	G0	G4	G6	G8
P15	High-Z	G1	G3	B9	R7	R7	R9	R11
P14	High-Z	G0	G2	B8	R6	R6	R8	R10
P13	High-Z	High-Z	G1	G11	High-Z	G0	G2	G4
P12	High-Z	High-Z	G0	G10	High-Z	G1	G3	G5
P11	B7	B9	B11	B7	B7	B7	B9	B11
P10	B6	B8	B10	B6	B6	B6	B8	B10
P09	B5	B7	B9	B5	B5	B5	B7	B9
P08	B4	B6	B8	B4	B4	B4	B6	B8
P07	B3	B5	B7	B3	B3	B3	B5	B7
P06	B2	B4	B6	B2	B2	B2	B4	B6
P05	B1	B3	B5	B1	B1	B1	B3	B5
P04	B0	B2	B4	B0	B0	B0	B2	B4
P03	High-Z	B1	B3	R11	High-Z	High-Z	High-Z	B0
P02	High-Z	B0	B2	R10	High-Z	High-Z	High-Z	G1
P01	High-Z	High-Z	B1	G9	High-Z	High-Z	R0	R2
P00	High-Z	High-Z	B0	G8	High-Z	High-Z	R1	R3

Table 103: DDR 4:2:2 Output Modes

OP_FORMAT_SEL[7:0]	DDR 4:2:2 Mode (Clock/2)					
	0x20		0x21		0x22	
	8-Bit DDR ITU-656 (Clock/2 Output) 4:2:2 Mode 0		10-Bit DDR ITU-656 (Clock/2 Output) 4:2:2 Mode 0		12-Bit DDR ITU-656 (Clock/2 Output) 4:2:2 Mode 0	
Pixel Output	**Clock Rise**	**Clock Fall**	**Clock Rise**	**Clock Fall**	**Clock Rise**	**Clock Fall**
P35	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P34	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P33	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P32	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P31	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P30	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P29	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P28	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P27	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P26	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P25	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P24	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P23	Cb7, Cr7	Y7	Cb9, Cr9	Y9	Cb11, Cr11	Y11
P22	Cb6, Cr6	Y6	Cb8, Cr8	Y8	Cb10, Cr10	Y10
P21	Cb5, Cr5	Y5	Cb7, Cr7	Y7	Cb9, Cr9	Y9
P20	Cb4, Cr4	Y4	Cb6, Cr6	Y6	Cb8, Cr8	Y8
P19	Cb3, Cr3	Y3	Cb5, Cr5	Y5	Cb7, Cr7	Y7
P18	Cb2, Cr2	Y2	Cb4, Cr4	Y4	Cb6, Cr6	Y6
P17	Cb1, Cr1	Y1	Cb3, Cr3	Y3	Cb5, Cr5	Y5
P16	Cb0, Cr0	Y0	Cb2, Cr2	Y2	Cb4, Cr4	Y4
P15	High-Z	High-Z	Cb1, Cr1	Y1	Cb3, Cr3	Y3
P14	High-Z	High-Z	Cb0, Cr0	Y0	Cb2, Cr2	Y2
P13	High-Z	High-Z	High-Z	High-Z	Cb1, Cr1	Y1
P12	High-Z	High-Z	High-Z	High-Z	Cb0, Cr0	Y0
P11	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P10	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P09	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P08	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P07	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P06	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P05	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P04	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P03	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P02	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P01	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z
P00	High-Z	High-Z	High-Z	High-Z	High-Z	High-Z

Table 104: Parallel Output Modes

	Parallel modes		
OP_FORMAT_SEL[7:0]	**0xC0**	**0xC1**	**0xC2**
Pixel Output	**8-Bit PAR Mode 0**	**10-Bit PAR Mode 0**	**12-Bit PAR Mode 0**
P35	Aux Y7,Cb7,Cr7	Aux Y9,Cb9,Cr9	Aux Y11,Cb11,Cr11
P34	Aux Y6,Cb6,Cr6	Aux Y8,Cb8,Cr8	Aux Y10,Cb10,Cr10
P33	Aux Y5,Cb5,Cr5	Aux Y7,Cb7,Cr7	Aux Y9,Cb9,Cr9
P32	Aux Y4,Cb4,Cr4	Aux Y6,Cb6,Cr6	Aux Y8,Cb8,Cr8
P31	Aux Y3,Cb3,Cr3	Aux Y5,Cb5,Cr5	Aux Y7,Cb7,Cr7
P30	Aux Y2,Cb2,Cr2	Aux Y4,Cb4,Cr4	Aux Y6,Cb6,Cr6
P29	Aux Y1,Cb1,Cr1	Aux Y3,Cb3,Cr3	Aux Y5,Cb5,Cr5
P28	Aux Y0,Cb0,Cr0	Aux Y2,Cb2,Cr2	Aux Y4,Cb4,Cr4
P27	High-Z	Aux Y1,Cb1,Cr1	Aux Y3,Cb3,Cr3
P26	High-Z	Aux Y0,Cb0,Cr0	Aux Y2,Cb2,Cr2
P25	High-Z	High-Z	Aux Y1,Cb1,Cr1
P24	High-Z	High-Z	Aux Y0,Cb0,Cr0
P23	Main Y7	Main Y9	Main Y11
P22	Main Y6	Main Y8	Main Y10
P21	Main Y5	Main Y7	Main Y9
P20	Main Y4	Main Y6	Main Y8
P19	Main Y3	Main Y5	Main Y7
P18	Main Y2	Main Y4	Main Y6
P17	Main Y1	Main Y3	Main Y5
P16	Main Y0	Main Y2	Main Y4
P15	High-Z	Main Y1	Main Y3
P14	High-Z	Main Y0	Main Y2
P13	High-Z	High-Z	Main Y1
P12	High-Z	High-Z	Main Y0
P11	Main Cb7,Cr7	Main Cb9,Cr9	Main Cb11,Cr11
P10	Main Cb6,Cr6	Main Cb8,Cr8	Main Cb10,Cr10
P09	Main Cb5,Cr5	Main Cb7,Cr7	Main Cb9,Cr9
P08	Main Cb4,Cr4	Main Cb6,Cr6	Main Cb8,Cr8
P07	Main Cb3,Cr3	Main Cb5,Cr5	Main Cb7,Cr7
P06	Main Cb2,Cr2	Main Cb4,Cr4	Main Cb6,Cr6
P05	Main Cb1,Cr1	Main Cb3,Cr3	Main Cb5,Cr5
P04	Main Cb0,Cr0	Main Cb2,Cr2	Main Cb4,Cr4
P03	High-Z	Main Cb1,Cr1	Main Cb3,Cr3
P02	High-Z	Main Cb0,Cr0	Main Cb2,Cr2
P01	High-Z	High-Z	Main Cb1,Cr1
P00	High-Z	High-Z	Main Cb0,Cr0

Table 105: DDR 4:4:4 Output Modes

	DDR 4:4:4 Mode (Clock/2)					
OP_FORMAT_SEL[7:0]	**0x60**		**0x61**		**0x62**	
	24-Bit DDR RGB (Clock/2 Output)		**30-Bit DDR RGB (Clock/2 Output)**		**36-Bit DDR RGB (Clock/2 Output)**	
Pixel Output	**Clock Rise**	**Clock Fall**	**Clock Rise**	**Clock Fall**	**Clock Rise**	**Clock Fall**
P35	R7-0	R7-1	R9-0	R9-1	R11-0	R11-1
P34	R6-0	R6-1	R8-0	R8-1	R10-0	R10-1
P33	R5-0	R5-1	R7-0	R7-1	R9-0	R9-1
P32	R4-0	R4-1	R6-0	R6-1	R8-0	R8-1
P31	R3-0	R3-1	R5-0	R5-1	R7-0	R7-1
P30	R2-0	R2-1	R4-0	R4-1	R6-0	R6-1
P29	R1-0	R1-1	R3-0	R3-1	R5-0	R5-1
P28	R0-0	R0-1	R2-0	R2-1	R4-0	R4-1
P27	High-Z	High-Z	R1-0	R1-1	R3-0	R3-1
P26	High-Z	High-Z	R0-0	R0-1	R2-0	R2-1
P25	High-Z	High-Z	High-Z	High-Z	R1-0	R1-1
P24	High-Z	High-Z	High-Z	High-Z	R0-0	R0-1
P23	G7-0	G7-1	G9-0	G9-1	G11-0	G11-1
P22	G6-0	G6-1	G8-0	G8-1	G10-0	G10-1
P21	G5-0	G5-1	G7-0	G7-1	G9-0	G9-1
P20	G4-0	G4-1	G6-0	G6-1	G8-0	G8-1
P19	G3-0	G3-1	G5-0	G5-1	G7-0	G7-1
P18	G2-0	G2-1	G4-0	G4-1	G6-0	G6-1
P17	G1-0	G1-1	G3-0	G3-1	G5-0	G5-1
P16	G0-0	G0-1	G2-0	G2-1	G4-0	G4-1
P15	High-Z	High-Z	G1-0	G1-1	G3-0	G3-1
P14	High-Z	High-Z	G0-0	G0-1	G2-0	G2-1
P13	High-Z	High-Z	High-Z	High-Z	G1-0	G1-1
P12	High-Z	High-Z	High-Z	High-Z	G0-0	G0-1
P11	B7-0	B7-1	B9-0	B9-1	B11-0	B11-1
P10	B6-0	B6-1	B8-0	B8-1	B10-0	B10-1
P09	B5-0	B5-1	B7-0	B7-1	B9-0	B9-1
P08	B4-0	B4-1	B6-0	B6-1	B8-0	B8-1
P07	B3-0	B3-1	B5-0	B5-1	B7-0	B7-1
P06	B2-0	B2-1	B4-0	B4-1	B6-0	B6-1
P05	B1-0	B1-1	B3-0	B3-1	B5-0	B5-1
P04	B0-0	B0-1	B2-0	B2-1	B4-0	B4-1
P03	High-Z	High-Z	B1-0	B1-1	B3-0	B3-1
P02	High-Z	High-Z	B0-0	B0-1	B2-0	B2-1
P01	High-Z	High-Z	High-Z	High-Z	B1-0	B1-1
P00	High-Z	High-Z	High-Z	High-Z	B0-0	B0-1

NOTES

NOTES

NOTES

I²C refers to a communications protocol originally developed by Philips Semiconductors (now NXP Semiconductors).

HDMI, the HDMI Logo, and High-Definition Multimedia Interface are trademarks or registered trademarks of HDMI Licensing LLC in the United States and other countries.

ESD Caution

ESD (electrostatic discharge) sensitive device. Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

UG09483-0-1/11(0)

www.analog.com