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

ADV7613 Register Control Manual

INTRODUCTION

This manual describes the I²C control registers for the ADV7613. The Register Maps section of this reference manual provides detailed register tables for the ADV7613 register maps.

The Register Bit Descriptions section provides details about the controls present in each register.

FUNCTIONAL BLOCK DIAGRAM

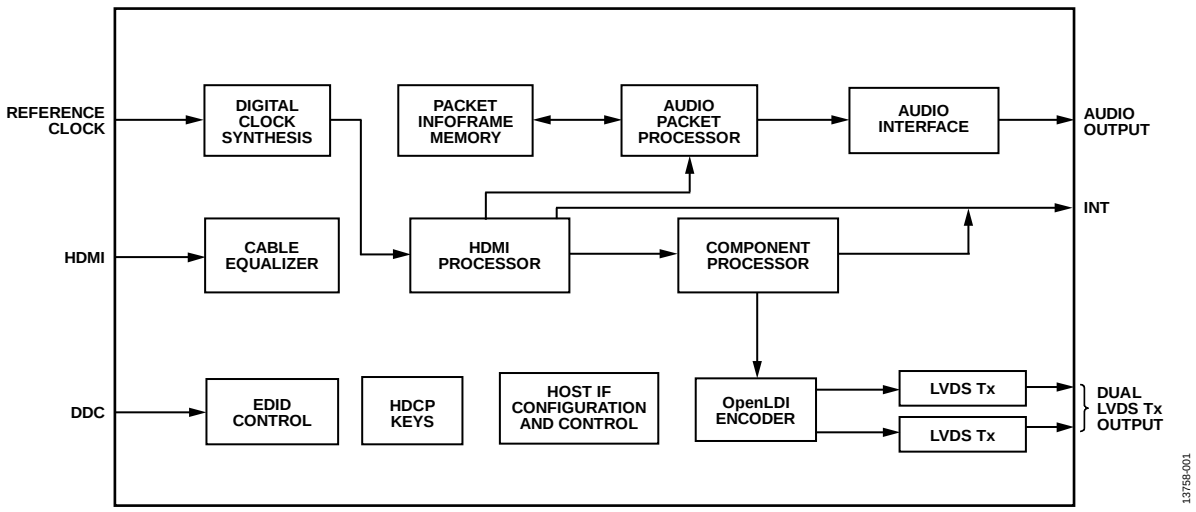

Figure 1.

TABLE OF CONTENTS

REVISION HISTORY

10/15—Revision 0: Initial Version

REGISTER MAPS

IO REGISTER MAP

Add is the register map I²C address, Def is the default value of the register, and Acc is the read/write access for the register (R means read only, R/W means read/write access, and SC means self clearing).

Table 1. ADV7613 IO Register Map

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x00	0x08	R/W			VID_STD[5]	VID_STD[4]	VID_STD[3]	VID_STD[2]	VID_STD[1]	VID_STD[0]
0x01	0x06	R/W		V_FREQ[2]	V_FREQ[1]	V_FREQ[0]	PRIM_MODE[3]	PRIM_MODE[2]	PRIM_MODE[1]	PRIM_MODE[0]
0x02	0xF0	R/W	INP_COLOR_SPACE[3]	INP_COLOR_SPACE[2]	INP_COLOR_SPACE[1]	INP_COLOR_SPACE[0]	ALT_GAMMA	OP_656_RANGE	RGB_OUT	ALT_DATA_SAT
0x03	0x00	R/W	OP_FORMAT_SEL[7]	OP_FORMAT_SEL[6]	OP_FORMAT_SEL[5]	OP_FORMAT_SEL[4]	OP_FORMAT_SEL[3]	OP_FORMAT_SEL[2]	OP_FORMAT_SEL[1]	OP_FORMAT_SEL[0]
0x04	0x62	R/W						XTAL_FREQ_SEL[1]	XTAL_FREQ_SEL[0]	
0x0B	0x44	R/W							CORE_PDN	XTAL_PDN
0x0C	0x62	R/W			POWER_DOWN			CP_PWRDN		PADS_PDN
0x12	0x00	R				CP_STDI_INTERLACED	CP_INTERLACED	CP_PROG_PARM_FOR_INT	CP_FORCE_INTERLACED	CP_NON_STD_VIDEO
0x15	0xBE	R/W				TRI_AUDIO				
0x20	0xF0	R/W	HPA_MAN_VALUE_A				HPA_TRISTATE_A			
0x21	0x00	R					HPA_STATUS_PORT_A			
0x3F	0x00	R							INTRQ_RAW	
0x40	0x20	R/W	INTRQ_DUR_SEL[1]	INTRQ_DUR_SEL[0]		STORE_UNMASKED_IRQS	EN_UMASK_RAW_INTRQ	MPU_STIM_INTRQ	INTRQ_OP_SEL[1]	INTRQ_OP_SEL[0]
0x41	0x30	R/W			CP_LOCK_UNLOCK_EDGE_SEL	STDI_DATA_VALID_EDGE_SEL				
0x42	0x00	R				STDI_DATA_VALID_RAW	CP_UNLOCK_RAW	CP_LOCK_RAW		
0x43	0x00	R				STDI_DATA_VALID_ST	CP_UNLOCK_ST	CP_LOCK_ST		
0x44	0x00	SC				STDI_DATA_VALID_CLR	CP_UNLOCK_CLR	CP_LOCK_CLR		
0x46	0x00	R/W				STDI_DATA_VALID_MB1	CP_UNLOCK_MB1	CP_LOCK_MB1		
0x47	0x00	R	MPU_STIM_INTRQ_RAW							
0x48	0x00	R	MPU_STIM_INTRQ_ST							
0x49	0x00	SC	MPU_STIM_INTRQ_CLR							
0x4B	0x00	R/W	MPU_STIM_INTRQ_MB1							
0x5B	0x00	R					CP_LOCK_CH1_RAW	CP_UNLOCK_CH1_RAW	STDI_DVALID_CH1_RAW	
0x5C	0x00	R					CP_LOCK_CH1_ST	CP_UNLOCK_CH1_ST	STDI_DVALID_CH1_ST	
0x5D	0x00	SC					CP_LOCK_CH1_CLR	CP_UNLOCK_CH1_CLR	STDI_DVALID_CH1_CLR	
0x5F	0x00	R/W					CP_LOCK_CH1_MB1	CP_UNLOCK_CH1_MB1	STDI_DVALID_CH1_MB1	
0x60	0x00	R	ISRC2_PCKT_RAW	ISRC1_PCKT_RAW	ACP_PCKT_RAW	VS_INFO_RAW	MS_INFO_RAW	SPD_INFO_RAW	AUDIO_INFO_RAW	AVI_INFO_RAW
0x61	0x00	R	ISRC2_PCKT_ST	ISRC1_PCKT_ST	ACP_PCKT_ST	VS_INFO_ST	MS_INFO_ST	SPD_INFO_ST	AUDIO_INFO_ST	AVI_INFO_ST

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x62	0x00	SC	ISRC2_PCKT_CLR	ISRC1_PCKT_CLR	ACP_PCKT_CLR	VS_INFO_CLR	MS_INFO_CLR	SPD_INFO_CLR	AUDIO_INFO_CLR	AVI_INFO_CLR
0x64	0x00	R/W	ISRC2_PCKT_MB1	ISRC1_PCKT_MB1	ACP_PCKT_MB1	VS_INFO_MB1	MS_INFO_MB1	SPD_INFO_MB1	AUDIO_INFO_MB1	AVI_INFO_MB1
0x65	0x00	R	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
0x66	0x00	R	CS_DATA_VALID_ST	INTERNAL_MUTE_ST	AV_MUTE_ST	AUDIO_CH_MD_ST	HDMI_MODE_ST	GEN_CTL_PCKT_ST	AUDIO_C_PCKT_ST	GAMUT_MDATA_ST
0x67	0x00	SC	CS_DATA_VALID_CLR	INTERNAL_MUTE_CLR	AV_MUTE_CLR	AUDIO_CH_MD_CLR	HDMI_MODE_CLR	GEN_CTL_PCKT_CLR	AUDIO_C_PCKT_CLR	GAMUT_MDATA_CLR
0x69	0x00	R/W	CS_DATA_VALID_MB1	INTERNAL_MUTE_MB1	AV_MUTE_MB1	AUDIO_CH_MD_MB1	HDMI_MODE_MB1	GEN_CTL_PCKT_MB1	AUDIO_C_PCKT_MB1	GAMUT_MDATA_MB1
0x6A	0x00	R		TMDSPLL_LCK_A_RAW		TMDS_CLK_A_RAW		VIDEO_3D_RAW	V_LOCKED_RAW	DE_REGEN_LCK_RAW
0x6B	0x00	R		TMDSPLL_LCK_A_ST		TMDS_CLK_A_ST		VIDEO_3D_ST	V_LOCKED_ST	DE_REGEN_LCK_ST
0x6C	0x00	SC		TMDSPLL_LCK_A_CLR		TMDS_CLK_A_CLR		VIDEO_3D_CLR	V_LOCKED_CLR	DE_REGEN_LCK_CLR
0x6E	0x00	R/W		TMDSPLL_LCK_A_MB1		TMDS_CLK_A_MB1		VIDEO_3D_MB1	V_LOCKED_MB1	DE_REGEN_LCK_MB1
0x6F	0x00	R						HDMI_ENCRPT_A_RAW		CABLE_DET_A_RAW
0x70	0x00	R						HDMI_ENCRPT_A_ST		CABLE_DET_A_ST
0x71	0x00	SC						HDMI_ENCRPT_A_CLR		CABLE_DET_A_CLR
0x73	0x00	R/W						HDMI_ENCRPT_A_MB1		CABLE_DET_A_MB1
0x79	0x00	R	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
0x7A	0x00	R	NEW_ISRC2_PCKT_ST	NEW_ISRC1_PCKT_ST	NEW_ACP_PCKT_ST	NEW_VS_INFO_ST	NEW_MS_INFO_ST	NEW_SPD_INFO_ST	NEW_AUDIO_INFO_ST	NEW_AVI_INFO_ST
0x7B	0x00	SC	NEW_ISRC2_PCKT_CLR	NEW_ISRC1_PCKT_CLR	NEW_ACP_PCKT_CLR	NEW_VS_INFO_CLR	NEW_MS_INFO_CLR	NEW_SPD_INFO_CLR	NEW_AUDIO_INFO_CLR	NEW_AVI_INFO_CLR
0x7D	0x00	R/W	NEW_ISRC2_PCKT_MB1	NEW_ISRC1_PCKT_MB1	NEW_ACP_PCKT_MB1	NEW_VS_INFO_MB1	NEW_MS_INFO_MB1	NEW_SPD_INFO_MB1	NEW_AUDIO_INFO_MB1	NEW_AVI_INFO_MB1
0x7E	0x00	R	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
0x7F	0x00	R	FIFO_NEAR_OVFL_ST	FIFO_UNDERFLO_ST	FIFO_OVERFLO_ST	CTS_PASS_THRSH_ST	CHANGE_N_ST	PACKET_ERROR_ST	AUDIO_PCKT_ERR_ST	NEW_GAMUT_MDATA_ST
0x80	0x00	SC	FIFO_NEAR_OVFL_CLR	FIFO_UNDERFLO_CLR	FIFO_OVERFLO_CLR	CTS_PASS_THRSH_CLR	CHANGE_N_CLR	PACKET_ERROR_CLR	AUDIO_PCKT_ERR_CLR	NEW_GAMUT_MDATA_CLR
0x82	0x00	R/W	FIFO_NEAR_OVFL_MB1	FIFO_UNDERFLO_MB1	FIFO_OVERFLO_MB1	CTS_PASS_THRSH_MB1	CHANGE_N_MB1	PACKET_ERROR_MB1	AUDIO_PCKT_ERR_MB1	NEW_GAMUT_MDATA_MB1
0x83	0x00	R	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
0x84	0x00	R	DEEP_COLOR_CHNG_ST	VCLK_CHNG_ST	AUDIO_MODE_CHNG_ST	PARITY_ERROR_ST	NEW_SAMP_RT_ST	AUDIO_FLT_LINE_ST	NEW_TMDS_FRQ_ST	FIFO_NEAR_UFLO_ST
0x85	0x00	SC	DEEP_COLOR_CHNG_CLR	VCLK_CHNG_CLR	AUDIO_MODE_CHNG_CLR	PARITY_ERROR_CLR	NEW_SAMP_RT_CLR	AUDIO_FLT_LINE_CLR	NEW_TMDS_FRQ_CLR	FIFO_NEAR_UFLO_CLR
0x87	0x00	R/W	DEEP_COLOR_CHNG_MB1	VCLK_CHNG_MB1	AUDIO_MODE_CHNG_MB1	PARITY_ERROR_MB1	NEW_SAMP_RT_MB1	AUDIO_FLT_LINE_MB1	NEW_TMDS_FRQ_MB1	FIFO_NEAR_UFLO_MB1
0x88	0x00	R	MS_INF_CKS_ERR_RAW	SPD_INF_CKS_ERR_RAW	AUD_INF_CKS_ERR_RAW	AVI_INF_CKS_ERR_RAW	RI_EXPIRED_B_RAW	RI_EXPIRED_A_RAW	AKSV_UPDATE_B_RAW	AKSV_UPDATE_A_RAW

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x89	0x00	R	MS_INF_CKS_ERR_ST	SPD_INF_CKS_ERR_ST	AUD_INF_CKS_ERR_ST	AVI_INF_CKS_ERR_ST	RI_EXPIRED_B_ST	RI_EXPIRED_A_ST	AKSV_UPDATE_B_ST	AKSV_UPDATE_A_ST
0x8A	0x00	SC	MS_INF_CKS_ERR_CLR	SPD_INF_CKS_ERR_CLR	AUD_INF_CKS_ERR_CLR	AVI_INF_CKS_ERR_CLR	RI_EXPIRED_B_CLR	RI_EXPIRED_A_CLR	AKSV_UPDATE_B_CLR	AKSV_UPDATE_A_CLR
0x8C	0x00	R/W	MS_INF_CKS_ERR_MB1	SPD_INF_CKS_ERR_MB1	AUD_INF_CKS_ERR_MB1	AVI_INF_CKS_ERR_MB1	RI_EXPIRED_B_MB1	RI_EXPIRED_A_MB1	AKSV_UPDATE_B_MB1	AKSV_UPDATE_A_MB1
0x8D	0x00	R								VS_INF_CKS_ERR_RAW
0x8E	0x00	R								VS_INF_CKS_ERR_ST
0x8F	0x00	SC								VS_INF_CKS_ERR_CLR
0x91	0x00	R/W								VS_INF_CKS_ERR_MB1
0x92	0x00	R			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
0x93	0x00	R			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
0x94	0x00	SC			CEC_RX_RDY2_CLR	CEC_RX_RDY1_CLR	CEC_RX_RDY0_CLR	CEC_TX_RETRY_TIMEOUT_CLR	CEC_TX_ARBITRATION_LOST_CLR	CEC_TX_READY_CLR
0x96	0x00	R/W			CEC_RX_RDY2_MB1	CEC_RX_RDY1_MB1	CEC_RX_RDY0_MB1	CEC_TX_RETRY_TIMEOUT_MB1	CEC_TX_ARBITRATION_LOST_MB1	CEC_TX_READY_MB1
0x97	0x00	R	CEC_INTERRUPT_BYTE[7]	CEC_INTERRUPT_BYTE[6]	CEC_INTERRUPT_BYTE[5]	CEC_INTERRUPT_BYTE[4]	CEC_INTERRUPT_BYTE[3]	CEC_INTERRUPT_BYTE[2]	CEC_INTERRUPT_BYTE[1]	CEC_INTERRUPT_BYTE[0]
0x98	0x00	R	CEC_INTERRUPT_BYTE_ST[7]	CEC_INTERRUPT_BYTE_ST[6]	CEC_INTERRUPT_BYTE_ST[5]	CEC_INTERRUPT_BYTE_ST[4]	CEC_INTERRUPT_BYTE_ST[3]	CEC_INTERRUPT_BYTE_ST[2]	CEC_INTERRUPT_BYTE_ST[1]	CEC_INTERRUPT_BYTE_ST[0]
0x99	0x00	SC	CEC_INTERRUPT_BYTE_CLR[7]	CEC_INTERRUPT_BYTE_CLR[6]	CEC_INTERRUPT_BYTE_CLR[5]	CEC_INTERRUPT_BYTE_CLR[4]	CEC_INTERRUPT_BYTE_CLR[3]	CEC_INTERRUPT_BYTE_CLR[2]	CEC_INTERRUPT_BYTE_CLR[1]	CEC_INTERRUPT_BYTE_CLR[0]
0x9A	0x00	R/W	CEC_INTERRUPT_BYTE_MB2[7]	CEC_INTERRUPT_BYTE_MB2[6]	CEC_INTERRUPT_BYTE_MB2[5]	CEC_INTERRUPT_BYTE_MB2[4]	CEC_INTERRUPT_BYTE_MB2[3]	CEC_INTERRUPT_BYTE_MB2[2]	CEC_INTERRUPT_BYTE_MB2[1]	CEC_INTERRUPT_BYTE_MB2[0]
0x9B	0x00	R/W	CEC_INTERRUPT_BYTE_MB1[7]	CEC_INTERRUPT_BYTE_MB1[6]	CEC_INTERRUPT_BYTE_MB1[5]	CEC_INTERRUPT_BYTE_MB1[4]	CEC_INTERRUPT_BYTE_MB1[3]	CEC_INTERRUPT_BYTE_MB1[2]	CEC_INTERRUPT_BYTE_MB1[1]	CEC_INTERRUPT_BYTE_MB1[0]
0xE0	0x00	R/W	DS_WITHOUT_FILTER							
0xE7	0x00	R/W			DPP_LUMA_HBW_SEL	DPP_CHROMA_LOW_EN				
0xE9	0x00	R/W	LVDS_TX_SLAVE_ADDR[6]	LVDS_TX_SLAVE_ADDR[5]	LVDS_TX_SLAVE_ADDR[4]	LVDS_TX_SLAVE_ADDR[3]	LVDS_TX_SLAVE_ADDR[2]	LVDS_TX_SLAVE_ADDR[1]	LVDS_TX_SLAVE_ADDR[0]	
0xEA	0x00	R	RD_INFO[15]	RD_INFO[14]	RD_INFO[13]	RD_INFO[12]	RD_INFO[11]	RD_INFO[10]	RD_INFO[9]	RD_INFO[8]
0xEB	0x00	R	RD_INFO[7]	RD_INFO[6]	RD_INFO[5]	RD_INFO[4]	RD_INFO[3]	RD_INFO[2]	RD_INFO[1]	RD_INFO[0]
0xF4	0x00	R/W	CEC_SLAVE_ADDR[6]	CEC_SLAVE_ADDR[5]	CEC_SLAVE_ADDR[4]	CEC_SLAVE_ADDR[3]	CEC_SLAVE_ADDR[2]	CEC_SLAVE_ADDR[1]	CEC_SLAVE_ADDR[0]	
0xF5	0x00	R/W	INFOFRAME_SLAVE_ADDR[6]	INFOFRAME_SLAVE_ADDR[5]	INFOFRAME_SLAVE_ADDR[4]	INFOFRAME_SLAVE_ADDR[3]	INFOFRAME_SLAVE_ADDR[2]	INFOFRAME_SLAVE_ADDR[1]	INFOFRAME_SLAVE_ADDR[0]	
0xF8	0x00	R/W	DPLL_SLAVE_ADDR[6]	DPLL_SLAVE_ADDR[5]	DPLL_SLAVE_ADDR[4]	DPLL_SLAVE_ADDR[3]	DPLL_SLAVE_ADDR[2]	DPLL_SLAVE_ADDR[1]	DPLL_SLAVE_ADDR[0]	
0xF9	0x00	R/W	KSV_SLAVE_ADDR[6]	KSV_SLAVE_ADDR[5]	KSV_SLAVE_ADDR[4]	KSV_SLAVE_ADDR[3]	KSV_SLAVE_ADDR[2]	KSV_SLAVE_ADDR[1]	KSV_SLAVE_ADDR[0]	
0xFA	0x00	R/W	EDID_SLAVE_ADDR[6]	EDID_SLAVE_ADDR[5]	EDID_SLAVE_ADDR[4]	EDID_SLAVE_ADDR[3]	EDID_SLAVE_ADDR[2]	EDID_SLAVE_ADDR[1]	EDID_SLAVE_ADDR[0]	

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xFB	0x00	R/W	HDMI_SLAVE_ ADDR[6]	HDMI_SLAVE_ ADDR[5]	HDMI_SLAVE_A DDR[4]	HDMI_SLAVE_A DDR[3]	HDMI_SLAVE_ ADDR[2]	HDMI_SLAVE_A DDR[1]	HDMI_SLAVE_ ADDR[0]	
0xFD	0x00	R/W	CP_SLAVE_ ADDR[6]	CP_SLAVE_ ADDR[5]	CP_SLAVE_ ADDR[4]	CP_SLAVE_ ADDR[3]	CP_SLAVE_ ADDR[2]	CP_SLAVE_ADD R[1]	CP_SLAVE_ ADDR[0]	
0xFF	0x00	SC	MAIN_RESET							

DPLL REGISTER MAP

Add is the register map I²C address, Def is the default value of the register, and Acc is the read/write access for the register (R means read only, and R/W means read/write access).

Table 2. ADV7613 DPLL Register Map

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xB5	0x01	R/W						MCLK_FS_N[2]	MCLK_FS_N[1]	MCLK_FS_N[0]

HDMI REGISTER MAP

Add is the register map I²C address, Def is the default value of the register, and Acc is the read/write access for the register (R means read only, R/W means read/write access, and SC means self clearing).

Table 3. ADV7613 HDMI Register Map

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x00	0x00	R/W	HDCP_A0	HDCP_ONLY_ MODE						
0x01	0x00	R/W				MUX_DSD_ OUT	OVR_AUTO_ MUX_DSD_ OUT	OVR_MUX_ HBR	MUX_HBR_ OUT	TERM_AUTO
0x03	0x18	R/W		I2SOUTMODE[1]	I2SOUTMODE[0]	I2SBITWIDTH[4]	I2SBITWIDTH[3]	I2SBITWIDTH[2]	I2SBITWIDTH[1]	I2SBITWIDTH[0]
0x04	0x00	R		AV_MUTE	HDCP_KEYS_ READ	HDCP_KEY_ ERROR	HDCP_RI_ EXPIRED		TMDS_PLL_ LOCKED	AUDIO_PLL_ LOCKED
0x05	0x00	R	HDMI_MODE	HDMI_ CONTENT_ ENCRYPTED	DVI_HSYNC_ POLARITY	DVI_VSYNC_ POLARITY	HDMI_PIXEL_ REPETITION[3]	HDMI_PIXEL_ REPETITION[2]	HDMI_PIXEL_ REPETITION[1]	HDMI_PIXEL_ REPETITION[0]
0x07	0x00	R	VERT_FILTER_ LOCKED	AUDIO_ CHANNEL_ MODE	DE_REGEN_ FILTER_ LOCKED	LINE_ WIDTH[12]	LINE_ WIDTH[11]	LINE_ WIDTH[10]	LINE_WIDTH[9]	LINE_WIDTH[8]
0x08	0x00	R	LINE_WIDTH[7]	LINE_WIDTH[6]	LINE_WIDTH[5]	LINE_WIDTH[4]	LINE_WIDTH[3]	LINE_WIDTH[2]	LINE_WIDTH[1]	LINE_WIDTH[0]
0x09	0x00	R				FIELD0_ HEIGHT[12]	FIELD0_ HEIGHT[11]	FIELD0_ HEIGHT[10]	FIELD0_ HEIGHT[9]	FIELD0_ HEIGHT[8]
0x0A	0x00	R	FIELD0_ HEIGHT[7]	FIELD0_ HEIGHT[6]	FIELD0_ HEIGHT[5]	FIELD0_ HEIGHT[4]	FIELD0_ HEIGHT[3]	FIELD0_ HEIGHT[2]	FIELD0_ HEIGHT[1]	FIELD0_ HEIGHT[0]
0x0B	0x00	R	DEEP_COLOR_ MODE[1]	DEEP_COLOR_M ODE[0]	HDMI_ INTERLACED	FIELD1_ HEIGHT[12]	FIELD1_ HEIGHT[11]	FIELD1_ HEIGHT[10]	FIELD1_ HEIGHT[9]	FIELD1_ HEIGHT[8]
0x0C	0x00	R	FIELD1_ HEIGHT[7]	FIELD1_ HEIGHT[6]	FIELD1_ HEIGHT[5]	FIELD1_ HEIGHT[4]	FIELD1_ HEIGHT[3]	FIELD1_ HEIGHT[2]	FIELD1_ HEIGHT[1]	FIELD1_ HEIGHT[0]
0x0D	0x04	R/W					FREQ- TOLERANCE[3]	FREQ- TOLERANCE[2]	FREQ- TOLERANCE[1]	FREQ- TOLERANCE[0]
0x0F	0x1F	R/W	MAN_AUDIO_ DL_BYPASS	AUDIO_ DELAY_LINE_ BYPASS		AUDIO_MUTE_ SPEED[4]	AUDIO_MUTE_ SPEED[3]	AUDIO_MUTE_ SPEED[2]	AUDIO_MUTE_ SPEED[1]	AUDIO_MUTE_ SPEED[0]
0x10	0x25	R/W			CTS_CHANGE_ THRESHOLD[5]	CTS_CHANGE_ THRESHOLD[4]	CTS_CHANGE_ THRESHOLD[3]	CTS_CHANGE_ THRESHOLD[2]	CTS_CHANGE_ THRESHOLD[1]	CTS_CHANGE_ THRESHOLD[0]
0x11	0x7D	R/W		AUDIO_FIFO_ ALMOST_ FULL_ THRESHOLD[6]	AUDIO_FIFO_ ALMOST_ FULL_ THRESHOLD[5]	AUDIO_FIFO_ ALMOST_ FULL_ THRESHOLD[4]	AUDIO_FIFO_ ALMOST_ FULL_ THRESHOLD[3]	AUDIO_FIFO_ ALMOST_ FULL_ THRESHOLD[2]	AUDIO_FIFO_ ALMOST_ FULL_ THRESHOLD[1]	AUDIO_FIFO_ ALMOST_ FULL_ THRESHOLD[0]
0x12	0x02	R/W		AUDIO_FIFO_ ALMOST_ EMPTY_ THRESHOLD[6]	AUDIO_FIFO_ ALMOST_ EMPTY_ THRESHOLD[5]	AUDIO_FIFO_ ALMOST_ EMPTY_ THRESHOLD[4]	AUDIO_FIFO_ ALMOST_ EMPTY_ THRESHOLD[3]	AUDIO_FIFO_ ALMOST_ EMPTY_ THRESHOLD[2]	AUDIO_FIFO_ ALMOST_ EMPTY_ THRESHOLD[1]	AUDIO_FIFO_ ALMOST_ EMPTY_ THRESHOLD[0]
0x13	0x7F	R/W		AC_MSK_ VCLK_CHNG	AC_MSK_ VPLL_UNLOCK		AC_MSK_ NEW_CTS	AC_MSK_ NEW_N	AC_MSK_ CHNG_PORT	AC_MSK_ VCLK_DET
0x14	0x3F	R/W			MT_MSK_ COMPRS_AUD	MT_MSK_AUD_ MODE_CHNG			MT_MSK_ PARITY_ERR	MT_MSK_ VCLK_CHNG
0x15	0xFF	R/W	MT_MSK_ APLL_UNLOCK	MT_MSK_ VPLL_UNLOCK	MT_MSK_ACR_ NOT_DET		MT_MSK_ FLATLINE_DET		MT_MSK_FIFO_ UNDERLFOW	MT_MSK_FIFO_ OVERFLOW
0x16	0xFF	R/W	MT_MSK_ AVMUTE	MT_MSK_NOT_ HDMIMODE	MT_MSK_ NEW_CTS	MT_MSK_ NEW_N	MT_MSK_ CHMODE_ CHNG	MT_MSK_ APCKT_ECC_ ERR	MT_MSK_ CHNG_PORT	MT_MSK_ VCLK_DET
0x18	0x00	R					HBR_AUDIO_ PCKT_DET	DST_AUDIO_ PCKT_DET	DSD_PACKET_D ET	AUDIO_ SAMPLE_PCKT_ DET
0x19	0x00	R						DST_DOUBLE		
0x1A	0x80	R/W		IGNORE_ PARITY_ERR		MUTE_AUDIO	WAIT_ UNMUTE[2]	WAIT_ UNMUTE[1]	WAIT_ UNMUTE[0]	NOT_AUTO_ UNMUTE
0x1B	0x18	R/W				DCFIFO_ RESET_ON_ LOCK	DCFIFO_KILL_ NOT_LOCKED	DCFIFO_KILL_ DIS		
0x1C	0x00	R					DCFIFO_ LOCKED	DCFIFO_ LEVEL[2]	DCFIFO_ LEVEL[1]	DCFIFO_ LEVEL[0]

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x1D	0x00	R/W			UP_ CONVERSION_ MODE					
0x1E	0x00	R			TOTAL_LINE_ WIDTH[13]	TOTAL_LINE_ WIDTH[12]	TOTAL_LINE_ WIDTH[11]	TOTAL_LINE_ WIDTH[10]	TOTAL_LINE_ WIDTH[9]	TOTAL_LINE_ WIDTH[8]
0x1F	0x00	R	TOTAL_LINE_ WIDTH[7]	TOTAL_LINE_ WIDTH[6]	TOTAL_LINE_ WIDTH[5]	TOTAL_LINE_ WIDTH[4]	TOTAL_LINE_ WIDTH[3]	TOTAL_LINE_ WIDTH[2]	TOTAL_LINE_ WIDTH[1]	TOTAL_LINE_ WIDTH[0]
0x20	0x00	R				HSYNC_FRONT _PORCH[12]	HSYNC_FRONT_ PORCH[11]	HSYNC_FRONT_ PORCH[10]	HSYNC_FRONT_ PORCH[9]	HSYNC_FRONT_ PORCH[8]
0x21	0x00	R	HSYNC_ FRONT_ PORCH[7]	HSYNC_ FRONT_ PORCH[6]	HSYNC_ FRONT_ PORCH[5]	HSYNC_ FRONT_ PORCH[4]	HSYNC_ FRONT_ PORCH[3]	HSYNC_ FRONT_ PORCH[2]	HSYNC_ FRONT_ PORCH[1]	HSYNC_ FRONT_ PORCH[0]
0x22	0x00	R				HSYNC_PULSE_ WIDTH[12]	HSYNC_PULSE_ WIDTH[11]	HSYNC_PULSE_ WIDTH[10]	HSYNC_PULSE_ WIDTH[9]	HSYNC_PULSE_ WIDTH[8]
0x23	0x00	R	HSYNC_PULSE_ WIDTH[7]	HSYNC_PULSE_ WIDTH[6]	HSYNC_PULSE_ WIDTH[5]	HSYNC_PULSE_ WIDTH[4]	HSYNC_PULSE_ WIDTH[3]	HSYNC_PULSE_ WIDTH[2]	HSYNC_PULSE_ WIDTH[1]	HSYNC_PULSE_ WIDTH[0]
0x24	0x00	R				HSYNC_BACK_ PORCH[12]	HSYNC_BACK_ PORCH[11]	HSYNC_BACK_ PORCH[10]	HSYNC_BACK_ PORCH[9]	HSYNC_BACK_ PORCH[8]
0x25	0x00	R	HSYNC_BACK_ PORCH[7]	HSYNC_BACK_ PORCH[6]	HSYNC_BACK_ PORCH[5]	HSYNC_BACK_ PORCH[4]	HSYNC_BACK_ PORCH[3]	HSYNC_BACK_ PORCH[2]	HSYNC_BACK_ PORCH[1]	HSYNC_BACK_ PORCH[0]
0x26	0x00	R			FIELD0_TOTAL_ HEIGHT[13]	FIELD0_TOTAL _HEIGHT[12]	FIELD0_TOTAL_ HEIGHT[11]	FIELD0_TOTAL_ HEIGHT[10]	FIELD0_TOTAL_ HEIGHT[9]	FIELD0_TOTAL_ HEIGHT[8]
0x27	0x00	R	FIELD0_TOTAL_ HEIGHT[7]	FIELD0_TOTAL_ HEIGHT[6]	FIELD0_TOTAL_ HEIGHT[5]	FIELD0_TOTAL _HEIGHT[4]	FIELD0_TOTAL_ HEIGHT[3]	FIELD0_TOTAL_ HEIGHT[2]	FIELD0_TOTAL_ HEIGHT[1]	FIELD0_TOTAL_ HEIGHT[0]
0x28	0x00	R			FIELD1_TOTAL_ HEIGHT[13]	FIELD1_TOTAL _HEIGHT[12]	FIELD1_TOTAL_ HEIGHT[11]	FIELD1_TOTAL_ HEIGHT[10]	FIELD1_TOTAL_ HEIGHT[9]	FIELD1_TOTAL_ HEIGHT[8]
0x29	0x00	R	FIELD1_TOTAL_ HEIGHT[7]	FIELD1_TOTAL_ HEIGHT[6]	FIELD1_TOTAL_ HEIGHT[5]	FIELD1_TOTAL _HEIGHT[4]	FIELD1_TOTAL_ HEIGHT[3]	FIELD1_TOTAL_ HEIGHT[2]	FIELD1_TOTAL_ HEIGHT[1]	FIELD1_TOTAL_ HEIGHT[0]
0x2A	0x00	R			FIELD0_VS_ FRONT_ PORCH[13]	FIELD0_VS_ FRONT_ PORCH[12]	FIELD0_VS_ FRONT_ PORCH[11]	FIELD0_VS_ FRONT_ PORCH[10]	FIELD0_VS_ FRONT_ PORCH[9]	FIELD0_VS_ FRONT_ PORCH[8]
0x2B	0x00	R	FIELD0_VS_ FRONT_ PORCH[7]	FIELD0_VS_ FRONT_ PORCH[6]	FIELD0_VS_ FRONT_ PORCH[5]	FIELD0_VS_ FRONT_ PORCH[4]	FIELD0_VS_ FRONT_ PORCH[3]	FIELD0_VS_ FRONT_ PORCH[2]	FIELD0_VS_ FRONT_ PORCH[1]	FIELD0_VS_ FRONT_ PORCH[0]
0x2C	0x00	R			FIELD1_VS_ FRONT_ PORCH[13]	FIELD1_VS_ FRONT_ PORCH[12]	FIELD1_VS_ FRONT_ PORCH[11]	FIELD1_VS_ FRONT_ PORCH[10]	FIELD1_VS_ FRONT_ PORCH[9]	FIELD1_VS_ FRONT_ PORCH[8]
0x2D	0x00	R	FIELD1_VS_ FRONT_ PORCH[7]	FIELD1_VS_ FRONT_ PORCH[6]	FIELD1_VS_ FRONT_ PORCH[5]	FIELD1_VS_ FRONT_ PORCH[4]	FIELD1_VS_ FRONT_ PORCH[3]	FIELD1_VS_ FRONT_ PORCH[2]	FIELD1_VS_ FRONT_ PORCH[1]	FIELD1_VS_ FRONT_ PORCH[0]
0x2E	0x00	R			FIELD0_VS_ PULSE_ WIDTH[13]	FIELD0_VS_ PULSE_ WIDTH[12]	FIELD0_VS_ PULSE_ WIDTH[11]	FIELD0_VS_ PULSE_ WIDTH[10]	FIELD0_VS_ PULSE_ WIDTH[9]	FIELD0_VS_ PULSE_ WIDTH[8]
0x2F	0x00	R	FIELD0_VS_ PULSE_ WIDTH[7]	FIELD0_VS_ PULSE_ WIDTH[6]	FIELD0_VS_ PULSE_ WIDTH[5]	FIELD0_VS_ PULSE_ WIDTH[4]	FIELD0_VS_ PULSE_ WIDTH[3]	FIELD0_VS_ PULSE_ WIDTH[2]	FIELD0_VS_ PULSE_ WIDTH[1]	FIELD0_VS_ PULSE_ WIDTH[0]
0x30	0x00	R			FIELD1_VS_ PULSE_ WIDTH[13]	FIELD1_VS_ PULSE_ WIDTH[12]	FIELD1_VS_ PULSE_ WIDTH[11]	FIELD1_VS_ PULSE_ WIDTH[10]	FIELD1_VS_ PULSE_ WIDTH[9]	FIELD1_VS_ PULSE_ WIDTH[8]
0x31	0x00	R	FIELD1_VS_ PULSE_ WIDTH[7]	FIELD1_VS_ PULSE_ WIDTH[6]	FIELD1_VS_ PULSE_ WIDTH[5]	FIELD1_VS_ PULSE_ WIDTH[4]	FIELD1_VS_ PULSE_ WIDTH[3]	FIELD1_VS_ PULSE_ WIDTH[2]	FIELD1_VS_ PULSE_ WIDTH[1]	FIELD1_VS_ PULSE_ WIDTH[0]
0x32	0x00	R			FIELD0_VS_ BACK_ PORCH[13]	FIELD0_VS_ BACK_ PORCH[12]	FIELD0_VS_ BACK_ PORCH[11]	FIELD0_VS_ BACK_ PORCH[10]	FIELD0_VS_ BACK_ PORCH[9]	FIELD0_VS_ BACK_ PORCH[8]
0x33	0x00	R	FIELD0_VS_ BACK_ PORCH[7]	FIELD0_VS_ BACK_ PORCH[6]	FIELD0_VS_ BACK_ PORCH[5]	FIELD0_VS_ BACK_ PORCH[4]	FIELD0_VS_ BACK_ PORCH[3]	FIELD0_VS_ BACK_ PORCH[2]	FIELD0_VS_ BACK_ PORCH[1]	FIELD0_VS_ BACK_ PORCH[0]
0x34	0x00	R			FIELD1_VS_ BACK_ PORCH[13]	FIELD1_VS_ BACK_ PORCH[12]	FIELD1_VS_ BACK_ PORCH[11]	FIELD1_VS_ BACK_ PORCH[10]	FIELD1_VS_ BACK_ PORCH[9]	FIELD1_VS_ BACK_ PORCH[8]
0x35	0x00	R	FIELD1_VS_ BACK_ PORCH[7]	FIELD1_VS_ BACK_ PORCH[6]	FIELD1_VS_ BACK_ PORCH[5]	FIELD1_VS_ BACK_ PORCH[4]	FIELD1_VS_ BACK_ PORCH[3]	FIELD1_VS_ BACK_ PORCH[2]	FIELD1_VS_ BACK_ PORCH[1]	FIELD1_VS_ BACK_ PORCH[0]
0x36	0x00	R	CS_DATA[7]	CS_DATA[6]	CS_DATA[5]	CS_DATA[4]	CS_DATA[3]	CS_DATA[2]	CS_DATA[1]	CS_DATA[0]

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x37	0x00	R	CS_DATA[15]	CS_DATA[14]	CS_DATA[13]	CS_DATA[12]	CS_DATA[11]	CS_DATA[10]	CS_DATA[9]	CS_DATA[8]
0x38	0x00	R	CS_DATA[23]	CS_DATA[22]	CS_DATA[21]	CS_DATA[20]	CS_DATA[19]	CS_DATA[18]	CS_DATA[17]	CS_DATA[16]
0x39	0x00	R	CS_DATA[31]	CS_DATA[30]	CS_DATA[29]	CS_DATA[28]	CS_DATA[27]	CS_DATA[26]	CS_DATA[25]	CS_DATA[24]
0x3A	0x00	R	CS_DATA[39]	CS_DATA[38]	CS_DATA[37]	CS_DATA[36]	CS_DATA[35]	CS_DATA[34]	CS_DATA[33]	CS_DATA[32]
0x3C	0x02	R/W				BYPASS_ AUDIO_ PASSTHRU				
0x40	0x00	R/W		OVERRIDE_ DEEP_COLOR_ MODE	DEEP_COLOR_ MODE_USER[1]	DEEP_COLOR_ MODE_USER[0]				
0x41	0x40	R/W				DEREP_N_ OVERRIDE	DEREP_N[3]	DEREP_N[2]	DEREP_N[1]	DEREP_N[0]
0x47	0x00	R/W						QZERO_ITC_ DIS	QZERO_RGB_ FULL	ALWAYS_ STORE_INF
0x48	0x00	R/W		DIS_CABLE_ DET_RST						
0x50	0x00	R/W				GAMUT_IRQ_ NEXT_FIELD			CS_ COPYRIGHT_ MANUAL	CS_ COPYRIGHT_ VALUE
0x51	0x00	R	TMDSFREQ[8]	TMDSFREQ[7]	TMDSFREQ[6]	TMDSFREQ[5]	TMDSFREQ[4]	TMDSFREQ[3]	TMDSFREQ[2]	TMDSFREQ[1]
0x52	0x00	R	TMDSFREQ[0]	TMDSFREQ_ FRAC[6]	TMDSFREQ_ FRAC[5]	TMDSFREQ_ FRAC[4]	TMDSFREQ_ FRAC[3]	TMDSFREQ_ FRAC[2]	TMDSFREQ_ FRAC[1]	TMDSFREQ_ FRAC[0]
0x53	0x00	R					HDMI_ COLORSPACE[3]	HDMI_ COLORSPACE[2]	HDMI_ COLORSPACE[1]	HDMI_ COLORSPACE[0]
0x56	0x58	R/W	FILT_5V_DET_ DIS	FILT_5V_DET_ TIMER[6]	FILT_5V_DET_ TIMER[5]	FILT_5V_DET_ TIMER[4]	FILT_5V_DET_ TIMER[3]	FILT_5V_DET_ TIMER[2]	FILT_5V_DET_ TIMER[1]	FILT_5V_DET_ TIMER[0]
0x5A	0x00	SC					HDCP_REPT_ EDID_RESET	DCFIFO_ RECENTER		FORCE_N_ UPDATE
0x5B	0x00	R	CTS[19]	CTS[18]	CTS[17]	CTS[16]	CTS[15]	CTS[14]	CTS[13]	CTS[12]
0x5C	0x00	R	CTS[11]	CTS[10]	CTS[9]	CTS[8]	CTS[7]	CTS[6]	CTS[5]	CTS[4]
0x5D	0x00	R	CTS[3]	CTS[2]	CTS[1]	CTS[0]	N[19]	N[18]	N[17]	N[16]
0x5E	0x00	R	N[15]	N[14]	N[13]	N[12]	N[11]	N[10]	N[9]	N[8]
0x5F	0x00	R	N[7]	N[6]	N[5]	N[4]	N[3]	N[2]	N[1]	N[0]
0x6C	0xA2	R/W	HPA_DELAY_ SEL[3]	HPA_DELAY_ SEL[2]	HPA_DELAY_ SEL[1]	HPA_DELAY_ SEL[0]	HPA_OVR_ TERM	HPA_AUTO_ INT_EDID[1]	HPA_AUTO_ INT_EDID[0]	HPA_MANUAL
0x6D	0x00	R/W	I2S_TDM_ MODE_ENABLE	I2S_SPDIF_ MAP_INV	I2S_SPDIF_ MAP_ROT[1]	I2S_SPDIF_ MAP_ROT[0]	DSD_MAP_INV	DSD_MAP_ ROT[2]	DSD_MAP_ ROT[1]	DSD_MAP_ ROT[0]
0x83	0xFF	R/W								CLOCK_TERMA_ DISABLE

REPEATER REGISTER MAP

Add is the register map I²C address, Def is the default value of the register, and Acc is the read/write access for the register (R means read only, and R/W means read/write access).

Table 4. ADV7613 Repeater Register Map

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x00	0x00	R	BKSV[7]	BKSV[6]	BKSV[5]	BKSV[4]	BKSV[3]	BKSV[2]	BKSV[1]	BKSV[0]
0x01	0x00	R	BKSV[15]	BKSV[14]	BKSV[13]	BKSV[12]	BKSV[11]	BKSV[10]	BKSV[9]	BKSV[8]
0x02	0x00	R	BKSV[23]	BKSV[22]	BKSV[21]	BKSV[20]	BKSV[19]	BKSV[18]	BKSV[17]	BKSV[16]
0x03	0x00	R	BKSV[31]	BKSV[30]	BKSV[29]	BKSV[28]	BKSV[27]	BKSV[26]	BKSV[25]	BKSV[24]
0x04	0x00	R	BKSV[39]	BKSV[38]	BKSV[37]	BKSV[36]	BKSV[35]	BKSV[34]	BKSV[33]	BKSV[32]
0x08	0x00	R	RI[7]	RI[6]	RI[5]	RI[4]	RI[3]	RI[2]	RI[1]	RI[0]
0x09	0x00	R	RI[15]	RI[14]	RI[13]	RI[12]	RI[11]	RI[10]	RI[9]	RI[8]
0x0A	0x00	R	PJ[7]	PJ[6]	PJ[5]	PJ[4]	PJ[3]	PJ[2]	PJ[1]	PJ[0]
0x10	0x00	R/W	AKSV[7]	AKSV[6]	AKSV[5]	AKSV[4]	AKSV[3]	AKSV[2]	AKSV[1]	AKSV[0]
0x11	0x00	R/W	AKSV[15]	AKSV[14]	AKSV[13]	AKSV[12]	AKSV[11]	AKSV[10]	AKSV[9]	AKSV[8]
0x12	0x00	R/W	AKSV[23]	AKSV[22]	AKSV[21]	AKSV[20]	AKSV[19]	AKSV[18]	AKSV[17]	AKSV[16]
0x13	0x00	R/W	AKSV[31]	AKSV[30]	AKSV[29]	AKSV[28]	AKSV[27]	AKSV[26]	AKSV[25]	AKSV[24]
0x14	0x00	R/W	AKSV[39]	AKSV[38]	AKSV[37]	AKSV[36]	AKSV[35]	AKSV[34]	AKSV[33]	AKSV[32]
0x15	0x00	R/W	AINFO[7]	AINFO[6]	AINFO[5]	AINFO[4]	AINFO[3]	AINFO[2]	AINFO[1]	AINFO[0]
0x18	0x00	R/W	AN[7]	AN[6]	AN[5]	AN[4]	AN[3]	AN[2]	AN[1]	AN[0]
0x19	0x00	R/W	AN[15]	AN[14]	AN[13]	AN[12]	AN[11]	AN[10]	AN[9]	AN[8]
0x1A	0x00	R/W	AN[23]	AN[22]	AN[21]	AN[20]	AN[19]	AN[18]	AN[17]	AN[16]
0x1B	0x00	R/W	AN[31]	AN[30]	AN[29]	AN[28]	AN[27]	AN[26]	AN[25]	AN[24]
0x1C	0x00	R/W	AN[39]	AN[38]	AN[37]	AN[36]	AN[35]	AN[34]	AN[33]	AN[32]
0x1D	0x00	R/W	AN[47]	AN[46]	AN[45]	AN[44]	AN[43]	AN[42]	AN[41]	AN[40]
0x1E	0x00	R/W	AN[55]	AN[54]	AN[53]	AN[52]	AN[51]	AN[50]	AN[49]	AN[48]
0x1F	0x00	R/W	AN[63]	AN[62]	AN[61]	AN[60]	AN[59]	AN[58]	AN[57]	AN[56]
0x20	0x00	R/W	SHA_A[7]	SHA_A[6]	SHA_A[5]	SHA_A[4]	SHA_A[3]	SHA_A[2]	SHA_A[1]	SHA_A[0]
0x21	0x00	R/W	SHA_A[15]	SHA_A[14]	SHA_A[13]	SHA_A[12]	SHA_A[11]	SHA_A[10]	SHA_A[9]	SHA_A[8]
0x22	0x00	R/W	SHA_A[23]	SHA_A[22]	SHA_A[21]	SHA_A[20]	SHA_A[19]	SHA_A[18]	SHA_A[17]	SHA_A[16]
0x23	0x00	R/W	SHA_A[31]	SHA_A[30]	SHA_A[29]	SHA_A[28]	SHA_A[27]	SHA_A[26]	SHA_A[25]	SHA_A[24]
0x40	0x83	R/W	BCAPS[7]	BCAPS[6]	BCAPS[5]	BCAPS[4]	BCAPS[3]	BCAPS[2]	BCAPS[1]	BCAPS[0]
0x41	0x00	R/W	BSTATUS[7]	BSTATUS[6]	BSTATUS[5]	BSTATUS[4]	BSTATUS[3]	BSTATUS[2]	BSTATUS[1]	BSTATUS[0]
0x42	0x00	R/W	BSTATUS[15]	BSTATUS[14]	BSTATUS[13]	BSTATUS[12]	BSTATUS[11]	BSTATUS[10]	BSTATUS[9]	BSTATUS[8]
0x70	0xC0	R/W	SPA_ LOCATION[7]	SPA_ LOCATION[6]	SPA_ LOCATION[5]	SPA_ LOCATION[4]	SPA_ LOCATION[3]	SPA_LOCATIO N[2]	SPA_ LOCATION[1]	SPA_ LOCATION[0]
0x71	0x00	R/W	KSV_LIST_ READY						SPA_ STORAGE_ MODE	SPA_ LOCATION_ MSB
0x72	0x00	R/W		EXT_EEPROM_ TRI						
0x73	0x00	R			VGA_EDID_ ENABLE_CPU					
0x74	0x00	R/W								EDID_A_ ENABLE
0x76	0x00	R								EDID_A_ ENABLE_CPU
0x78	0x00	SC								KSV_LIST_ READY_CLR_A
0x79	0x08	R/W	VGA_EDID_ ENABLE	KSV_MAP_ SELECT[2]	KSV_MAP_ SELECT[1]	KSV_MAP_ SELECT[0]	AUTO_HDCP_ MAP_ENABLE	HDCP_MAP_ SELECT[2]	HDCP_MAP_ SELECT[1]	HDCP_MAP_ SELECT[0]
0x7A	0x04	R/W							DISABLE_ AUTO_EDID	EDID_ SEGMENT_ POINTER
0x80	0x00	R/W	KSV_BYTE_0[7]	KSV_BYTE_0[6]	KSV_BYTE_0[5]	KSV_BYTE_0[4]	KSV_BYTE_0[3]	KSV_BYTE_0[2]	KSV_BYTE_0[1]	KSV_BYTE_0[0]
0x81	0x00	R/W	KSV_BYTE_1[7]	KSV_BYTE_1[6]	KSV_BYTE_1[5]	KSV_BYTE_1[4]	KSV_BYTE_1[3]	KSV_BYTE_1[2]	KSV_BYTE_1[1]	KSV_BYTE_1[0]
0x82	0x00	R/W	KSV_BYTE_2[7]	KSV_BYTE_2[6]	KSV_BYTE_2[5]	KSV_BYTE_2[4]	KSV_BYTE_2[3]	KSV_BYTE_2[2]	KSV_BYTE_2[1]	KSV_BYTE_2[0]
0x83	0x00	R/W	KSV_BYTE_3[7]	KSV_BYTE_3[6]	KSV_BYTE_3[5]	KSV_BYTE_3[4]	KSV_BYTE_3[3]	KSV_BYTE_3[2]	KSV_BYTE_3[1]	KSV_BYTE_3[0]

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x84	0x00	R/W	KSV_BYTE_4[7]	KSV_BYTE_4[6]	KSV_BYTE_4[5]	KSV_BYTE_4[4]	KSV_BYTE_4[3]	KSV_BYTE_4[2]	KSV_BYTE_4[1]	KSV_BYTE_4[0]
0x85	0x00	R/W	KSV_BYTE_5[7]	KSV_BYTE_5[6]	KSV_BYTE_5[5]	KSV_BYTE_5[4]	KSV_BYTE_5[3]	KSV_BYTE_5[2]	KSV_BYTE_5[1]	KSV_BYTE_5[0]
0x86	0x00	R/W	KSV_BYTE_6[7]	KSV_BYTE_6[6]	KSV_BYTE_6[5]	KSV_BYTE_6[4]	KSV_BYTE_6[3]	KSV_BYTE_6[2]	KSV_BYTE_6[1]	KSV_BYTE_6[0]
0x87	0x00	R/W	KSV_BYTE_7[7]	KSV_BYTE_7[6]	KSV_BYTE_7[5]	KSV_BYTE_7[4]	KSV_BYTE_7[3]	KSV_BYTE_7[2]	KSV_BYTE_7[1]	KSV_BYTE_7[0]
0x88	0x00	R/W	KSV_BYTE_8[7]	KSV_BYTE_8[6]	KSV_BYTE_8[5]	KSV_BYTE_8[4]	KSV_BYTE_8[3]	KSV_BYTE_8[2]	KSV_BYTE_8[1]	KSV_BYTE_8[0]
0x89	0x00	R/W	KSV_BYTE_9[7]	KSV_BYTE_9[6]	KSV_BYTE_9[5]	KSV_BYTE_9[4]	KSV_BYTE_9[3]	KSV_BYTE_9[2]	KSV_BYTE_9[1]	KSV_BYTE_9[0]
0x8A	0x00	R/W	KSV_BYTE_10[7]	KSV_BYTE_10[6]	KSV_BYTE_10[5]	KSV_BYTE_10[4]	KSV_BYTE_10[3]	KSV_BYTE_10[2]	KSV_BYTE_10[1]	KSV_BYTE_10[0]
0x8B	0x00	R/W	KSV_BYTE_11[7]	KSV_BYTE_11[6]	KSV_BYTE_11[5]	KSV_BYTE_11[4]	KSV_BYTE_11[3]	KSV_BYTE_11[2]	KSV_BYTE_11[1]	KSV_BYTE_11[0]
0x8C	0x00	R/W	KSV_BYTE_12[7]	KSV_BYTE_12[6]	KSV_BYTE_12[5]	KSV_BYTE_12[4]	KSV_BYTE_12[3]	KSV_BYTE_12[2]	KSV_BYTE_12[1]	KSV_BYTE_12[0]
0x8D	0x00	R/W	KSV_BYTE_13[7]	KSV_BYTE_13[6]	KSV_BYTE_13[5]	KSV_BYTE_13[4]	KSV_BYTE_13[3]	KSV_BYTE_13[2]	KSV_BYTE_13[1]	KSV_BYTE_13[0]
0x8E	0x00	R/W	KSV_BYTE_14[7]	KSV_BYTE_14[6]	KSV_BYTE_14[5]	KSV_BYTE_14[4]	KSV_BYTE_14[3]	KSV_BYTE_14[2]	KSV_BYTE_14[1]	KSV_BYTE_14[0]
0x8F	0x00	R/W	KSV_BYTE_15[7]	KSV_BYTE_15[6]	KSV_BYTE_15[5]	KSV_BYTE_15[4]	KSV_BYTE_15[3]	KSV_BYTE_15[2]	KSV_BYTE_15[1]	KSV_BYTE_15[0]
0x90	0x00	R/W	KSV_BYTE_16[7]	KSV_BYTE_16[6]	KSV_BYTE_16[5]	KSV_BYTE_16[4]	KSV_BYTE_16[3]	KSV_BYTE_16[2]	KSV_BYTE_16[1]	KSV_BYTE_16[0]
0x91	0x00	R/W	KSV_BYTE_17[7]	KSV_BYTE_17[6]	KSV_BYTE_17[5]	KSV_BYTE_17[4]	KSV_BYTE_17[3]	KSV_BYTE_17[2]	KSV_BYTE_17[1]	KSV_BYTE_17[0]
0x92	0x00	R/W	KSV_BYTE_18[7]	KSV_BYTE_18[6]	KSV_BYTE_18[5]	KSV_BYTE_18[4]	KSV_BYTE_18[3]	KSV_BYTE_18[2]	KSV_BYTE_18[1]	KSV_BYTE_18[0]
0x93	0x00	R/W	KSV_BYTE_19[7]	KSV_BYTE_19[6]	KSV_BYTE_19[5]	KSV_BYTE_19[4]	KSV_BYTE_19[3]	KSV_BYTE_19[2]	KSV_BYTE_19[1]	KSV_BYTE_19[0]
0x94	0x00	R/W	KSV_BYTE_20[7]	KSV_BYTE_20[6]	KSV_BYTE_20[5]	KSV_BYTE_20[4]	KSV_BYTE_20[3]	KSV_BYTE_20[2]	KSV_BYTE_20[1]	KSV_BYTE_20[0]
0x95	0x00	R/W	KSV_BYTE_21[7]	KSV_BYTE_21[6]	KSV_BYTE_21[5]	KSV_BYTE_21[4]	KSV_BYTE_21[3]	KSV_BYTE_21[2]	KSV_BYTE_21[1]	KSV_BYTE_21[0]
0x96	0x00	R/W	KSV_BYTE_22[7]	KSV_BYTE_22[6]	KSV_BYTE_22[5]	KSV_BYTE_22[4]	KSV_BYTE_22[3]	KSV_BYTE_22[2]	KSV_BYTE_22[1]	KSV_BYTE_22[0]
0x97	0x00	R/W	KSV_BYTE_23[7]	KSV_BYTE_23[6]	KSV_BYTE_23[5]	KSV_BYTE_23[4]	KSV_BYTE_23[3]	KSV_BYTE_23[2]	KSV_BYTE_23[1]	KSV_BYTE_23[0]
0x98	0x00	R/W	KSV_BYTE_24[7]	KSV_BYTE_24[6]	KSV_BYTE_24[5]	KSV_BYTE_24[4]	KSV_BYTE_24[3]	KSV_BYTE_24[2]	KSV_BYTE_24[1]	KSV_BYTE_24[0]
0x99	0x00	R/W	KSV_BYTE_25[7]	KSV_BYTE_25[6]	KSV_BYTE_25[5]	KSV_BYTE_25[4]	KSV_BYTE_25[3]	KSV_BYTE_25[2]	KSV_BYTE_25[1]	KSV_BYTE_25[0]
0x9A	0x00	R/W	KSV_BYTE_26[7]	KSV_BYTE_26[6]	KSV_BYTE_26[5]	KSV_BYTE_26[4]	KSV_BYTE_26[3]	KSV_BYTE_26[2]	KSV_BYTE_26[1]	KSV_BYTE_26[0]
0x9B	0x00	R/W	KSV_BYTE_27[7]	KSV_BYTE_27[6]	KSV_BYTE_27[5]	KSV_BYTE_27[4]	KSV_BYTE_27[3]	KSV_BYTE_27[2]	KSV_BYTE_27[1]	KSV_BYTE_27[0]
0x9C	0x00	R/W	KSV_BYTE_28[7]	KSV_BYTE_28[6]	KSV_BYTE_28[5]	KSV_BYTE_28[4]	KSV_BYTE_28[3]	KSV_BYTE_28[2]	KSV_BYTE_28[1]	KSV_BYTE_28[0]
0x9D	0x00	R/W	KSV_BYTE_29[7]	KSV_BYTE_29[6]	KSV_BYTE_29[5]	KSV_BYTE_29[4]	KSV_BYTE_29[3]	KSV_BYTE_29[2]	KSV_BYTE_29[1]	KSV_BYTE_29[0]
0x9E	0x00	R/W	KSV_BYTE_30[7]	KSV_BYTE_30[6]	KSV_BYTE_30[5]	KSV_BYTE_30[4]	KSV_BYTE_30[3]	KSV_BYTE_30[2]	KSV_BYTE_30[1]	KSV_BYTE_30[0]
0x9F	0x00	R/W	KSV_BYTE_31[7]	KSV_BYTE_31[6]	KSV_BYTE_31[5]	KSV_BYTE_31[4]	KSV_BYTE_31[3]	KSV_BYTE_31[2]	KSV_BYTE_31[1]	KSV_BYTE_31[0]
0xA0	0x00	R/W	KSV_BYTE_32[7]	KSV_BYTE_32[6]	KSV_BYTE_32[5]	KSV_BYTE_32[4]	KSV_BYTE_32[3]	KSV_BYTE_32[2]	KSV_BYTE_32[1]	KSV_BYTE_32[0]
0xA1	0x00	R/W	KSV_BYTE_33[7]	KSV_BYTE_33[6]	KSV_BYTE_33[5]	KSV_BYTE_33[4]	KSV_BYTE_33[3]	KSV_BYTE_33[2]	KSV_BYTE_33[1]	KSV_BYTE_33[0]
0xA2	0x00	R/W	KSV_BYTE_34[7]	KSV_BYTE_34[6]	KSV_BYTE_34[5]	KSV_BYTE_34[4]	KSV_BYTE_34[3]	KSV_BYTE_34[2]	KSV_BYTE_34[1]	KSV_BYTE_34[0]
0xA3	0x00	R/W	KSV_BYTE_35[7]	KSV_BYTE_35[6]	KSV_BYTE_35[5]	KSV_BYTE_35[4]	KSV_BYTE_35[3]	KSV_BYTE_35[2]	KSV_BYTE_35[1]	KSV_BYTE_35[0]
0xA4	0x00	R/W	KSV_BYTE_36[7]	KSV_BYTE_36[6]	KSV_BYTE_36[5]	KSV_BYTE_36[4]	KSV_BYTE_36[3]	KSV_BYTE_36[2]	KSV_BYTE_36[1]	KSV_BYTE_36[0]
0xA5	0x00	R/W	KSV_BYTE_37[7]	KSV_BYTE_37[6]	KSV_BYTE_37[5]	KSV_BYTE_37[4]	KSV_BYTE_37[3]	KSV_BYTE_37[2]	KSV_BYTE_37[1]	KSV_BYTE_37[0]
0xA6	0x00	R/W	KSV_BYTE_38[7]	KSV_BYTE_38[6]	KSV_BYTE_38[5]	KSV_BYTE_38[4]	KSV_BYTE_38[3]	KSV_BYTE_38[2]	KSV_BYTE_38[1]	KSV_BYTE_38[0]
0xA7	0x00	R/W	KSV_BYTE_39[7]	KSV_BYTE_39[6]	KSV_BYTE_39[5]	KSV_BYTE_39[4]	KSV_BYTE_39[3]	KSV_BYTE_39[2]	KSV_BYTE_39[1]	KSV_BYTE_39[0]
0xA8	0x00	R/W	KSV_BYTE_40[7]	KSV_BYTE_40[6]	KSV_BYTE_40[5]	KSV_BYTE_40[4]	KSV_BYTE_40[3]	KSV_BYTE_40[2]	KSV_BYTE_40[1]	KSV_BYTE_40[0]
0xA9	0x00	R/W	KSV_BYTE_41[7]	KSV_BYTE_41[6]	KSV_BYTE_41[5]	KSV_BYTE_41[4]	KSV_BYTE_41[3]	KSV_BYTE_41[2]	KSV_BYTE_41[1]	KSV_BYTE_41[0]
0xAA	0x00	R/W	KSV_BYTE_42[7]	KSV_BYTE_42[6]	KSV_BYTE_42[5]	KSV_BYTE_42[4]	KSV_BYTE_42[3]	KSV_BYTE_42[2]	KSV_BYTE_42[1]	KSV_BYTE_42[0]
0xAB	0x00	R/W	KSV_BYTE_43[7]	KSV_BYTE_43[6]	KSV_BYTE_43[5]	KSV_BYTE_43[4]	KSV_BYTE_43[3]	KSV_BYTE_43[2]	KSV_BYTE_43[1]	KSV_BYTE_43[0]
0xAC	0x00	R/W	KSV_BYTE_44[7]	KSV_BYTE_44[6]	KSV_BYTE_44[5]	KSV_BYTE_44[4]	KSV_BYTE_44[3]	KSV_BYTE_44[2]	KSV_BYTE_44[1]	KSV_BYTE_44[0]
0xAD	0x00	R/W	KSV_BYTE_45[7]	KSV_BYTE_45[6]	KSV_BYTE_45[5]	KSV_BYTE_45[4]	KSV_BYTE_45[3]	KSV_BYTE_45[2]	KSV_BYTE_45[1]	KSV_BYTE_45[0]
0xAE	0x00	R/W	KSV_BYTE_46[7]	KSV_BYTE_46[6]	KSV_BYTE_46[5]	KSV_BYTE_46[4]	KSV_BYTE_46[3]	KSV_BYTE_46[2]	KSV_BYTE_46[1]	KSV_BYTE_46[0]
0xAF	0x00	R/W	KSV_BYTE_47[7]	KSV_BYTE_47[6]	KSV_BYTE_47[5]	KSV_BYTE_47[4]	KSV_BYTE_47[3]	KSV_BYTE_47[2]	KSV_BYTE_47[1]	KSV_BYTE_47[0]
0xB0	0x00	R/W	KSV_BYTE_48[7]	KSV_BYTE_48[6]	KSV_BYTE_48[5]	KSV_BYTE_48[4]	KSV_BYTE_48[3]	KSV_BYTE_48[2]	KSV_BYTE_48[1]	KSV_BYTE_48[0]
0xB1	0x00	R/W	KSV_BYTE_49[7]	KSV_BYTE_49[6]	KSV_BYTE_49[5]	KSV_BYTE_49[4]	KSV_BYTE_49[3]	KSV_BYTE_49[2]	KSV_BYTE_49[1]	KSV_BYTE_49[0]
0xB2	0x00	R/W	KSV_BYTE_50[7]	KSV_BYTE_50[6]	KSV_BYTE_50[5]	KSV_BYTE_50[4]	KSV_BYTE_50[3]	KSV_BYTE_50[2]	KSV_BYTE_50[1]	KSV_BYTE_50[0]
0xB3	0x00	R/W	KSV_BYTE_51[7]	KSV_BYTE_51[6]	KSV_BYTE_51[5]	KSV_BYTE_51[4]	KSV_BYTE_51[3]	KSV_BYTE_51[2]	KSV_BYTE_51[1]	KSV_BYTE_51[0]
0xB4	0x00	R/W	KSV_BYTE_52[7]	KSV_BYTE_52[6]	KSV_BYTE_52[5]	KSV_BYTE_52[4]	KSV_BYTE_52[3]	KSV_BYTE_52[2]	KSV_BYTE_52[1]	KSV_BYTE_52[0]
0xB5	0x00	R/W	KSV_BYTE_53[7]	KSV_BYTE_53[6]	KSV_BYTE_53[5]	KSV_BYTE_53[4]	KSV_BYTE_53[3]	KSV_BYTE_53[2]	KSV_BYTE_53[1]	KSV_BYTE_53[0]
0xB6	0x00	R/W	KSV_BYTE_54[7]	KSV_BYTE_54[6]	KSV_BYTE_54[5]	KSV_BYTE_54[4]	KSV_BYTE_54[3]	KSV_BYTE_54[2]	KSV_BYTE_54[1]	KSV_BYTE_54[0]
0xB7	0x00	R/W	KSV_BYTE_55[7]	KSV_BYTE_55[6]	KSV_BYTE_55[5]	KSV_BYTE_55[4]	KSV_BYTE_55[3]	KSV_BYTE_55[2]	KSV_BYTE_55[1]	KSV_BYTE_55[0]
0xB8	0x00	R/W	KSV_BYTE_56[7]	KSV_BYTE_56[6]	KSV_BYTE_56[5]	KSV_BYTE_56[4]	KSV_BYTE_56[3]	KSV_BYTE_56[2]	KSV_BYTE_56[1]	KSV_BYTE_56[0]
0xB9	0x00	R/W	KSV_BYTE_57[7]	KSV_BYTE_57[6]	KSV_BYTE_57[5]	KSV_BYTE_57[4]	KSV_BYTE_57[3]	KSV_BYTE_57[2]	KSV_BYTE_57[1]	KSV_BYTE_57[0]
0xBA	0x00	R/W	KSV_BYTE_58[7]	KSV_BYTE_58[6]	KSV_BYTE_58[5]	KSV_BYTE_58[4]	KSV_BYTE_58[3]	KSV_BYTE_58[2]	KSV_BYTE_58[1]	KSV_BYTE_58[0]
0xBB	0x00	R/W	KSV_BYTE_59[7]	KSV_BYTE_59[6]	KSV_BYTE_59[5]	KSV_BYTE_59[4]	KSV_BYTE_59[3]	KSV_BYTE_59[2]	KSV_BYTE_59[1]	KSV_BYTE_59[0]

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xBC	0x00	R/W	KSV_BYTE_60[7]	KSV_BYTE_60[6]	KSV_BYTE_60[5]	KSV_BYTE_60[4]	KSV_BYTE_60[3]	KSV_BYTE_60[2]	KSV_BYTE_60[1]	KSV_BYTE_60[0]
0xBD	0x00	R/W	KSV_BYTE_61[7]	KSV_BYTE_61[6]	KSV_BYTE_61[5]	KSV_BYTE_61[4]	KSV_BYTE_61[3]	KSV_BYTE_61[2]	KSV_BYTE_61[1]	KSV_BYTE_61[0]
0xBE	0x00	R/W	KSV_BYTE_62[7]	KSV_BYTE_62[6]	KSV_BYTE_62[5]	KSV_BYTE_62[4]	KSV_BYTE_62[3]	KSV_BYTE_62[2]	KSV_BYTE_62[1]	KSV_BYTE_62[0]
0xBF	0x00	R/W	KSV_BYTE_63[7]	KSV_BYTE_63[6]	KSV_BYTE_63[5]	KSV_BYTE_63[4]	KSV_BYTE_63[3]	KSV_BYTE_63[2]	KSV_BYTE_63[1]	KSV_BYTE_63[0]
0xC0	0x00	R/W	KSV_BYTE_64[7]	KSV_BYTE_64[6]	KSV_BYTE_64[5]	KSV_BYTE_64[4]	KSV_BYTE_64[3]	KSV_BYTE_64[2]	KSV_BYTE_64[1]	KSV_BYTE_64[0]
0xC1	0x00	R/W	KSV_BYTE_65[7]	KSV_BYTE_65[6]	KSV_BYTE_65[5]	KSV_BYTE_65[4]	KSV_BYTE_65[3]	KSV_BYTE_65[2]	KSV_BYTE_65[1]	KSV_BYTE_65[0]
0xC2	0x00	R/W	KSV_BYTE_66[7]	KSV_BYTE_66[6]	KSV_BYTE_66[5]	KSV_BYTE_66[4]	KSV_BYTE_66[3]	KSV_BYTE_66[2]	KSV_BYTE_66[1]	KSV_BYTE_66[0]
0xC3	0x00	R/W	KSV_BYTE_67[7]	KSV_BYTE_67[6]	KSV_BYTE_67[5]	KSV_BYTE_67[4]	KSV_BYTE_67[3]	KSV_BYTE_67[2]	KSV_BYTE_67[1]	KSV_BYTE_67[0]
0xC4	0x00	R/W	KSV_BYTE_68[7]	KSV_BYTE_68[6]	KSV_BYTE_68[5]	KSV_BYTE_68[4]	KSV_BYTE_68[3]	KSV_BYTE_68[2]	KSV_BYTE_68[1]	KSV_BYTE_68[0]
0xC5	0x00	R/W	KSV_BYTE_69[7]	KSV_BYTE_69[6]	KSV_BYTE_69[5]	KSV_BYTE_69[4]	KSV_BYTE_69[3]	KSV_BYTE_69[2]	KSV_BYTE_69[1]	KSV_BYTE_69[0]
0xC6	0x00	R/W	KSV_BYTE_70[7]	KSV_BYTE_70[6]	KSV_BYTE_70[5]	KSV_BYTE_70[4]	KSV_BYTE_70[3]	KSV_BYTE_70[2]	KSV_BYTE_70[1]	KSV_BYTE_70[0]
0xC7	0x00	R/W	KSV_BYTE_71[7]	KSV_BYTE_71[6]	KSV_BYTE_71[5]	KSV_BYTE_71[4]	KSV_BYTE_71[3]	KSV_BYTE_71[2]	KSV_BYTE_71[1]	KSV_BYTE_71[0]
0xC8	0x00	R/W	KSV_BYTE_72[7]	KSV_BYTE_72[6]	KSV_BYTE_72[5]	KSV_BYTE_72[4]	KSV_BYTE_72[3]	KSV_BYTE_72[2]	KSV_BYTE_72[1]	KSV_BYTE_72[0]
0xC9	0x00	R/W	KSV_BYTE_73[7]	KSV_BYTE_73[6]	KSV_BYTE_73[5]	KSV_BYTE_73[4]	KSV_BYTE_73[3]	KSV_BYTE_73[2]	KSV_BYTE_73[1]	KSV_BYTE_73[0]
0xCA	0x00	R/W	KSV_BYTE_74[7]	KSV_BYTE_74[6]	KSV_BYTE_74[5]	KSV_BYTE_74[4]	KSV_BYTE_74[3]	KSV_BYTE_74[2]	KSV_BYTE_74[1]	KSV_BYTE_74[0]
0xCB	0x00	R/W	KSV_BYTE_75[7]	KSV_BYTE_75[6]	KSV_BYTE_75[5]	KSV_BYTE_75[4]	KSV_BYTE_75[3]	KSV_BYTE_75[2]	KSV_BYTE_75[1]	KSV_BYTE_75[0]
0xCC	0x00	R/W	KSV_BYTE_76[7]	KSV_BYTE_76[6]	KSV_BYTE_76[5]	KSV_BYTE_76[4]	KSV_BYTE_76[3]	KSV_BYTE_76[2]	KSV_BYTE_76[1]	KSV_BYTE_76[0]
0xCD	0x00	R/W	KSV_BYTE_77[7]	KSV_BYTE_77[6]	KSV_BYTE_77[5]	KSV_BYTE_77[4]	KSV_BYTE_77[3]	KSV_BYTE_77[2]	KSV_BYTE_77[1]	KSV_BYTE_77[0]
0xCE	0x00	R/W	KSV_BYTE_78[7]	KSV_BYTE_78[6]	KSV_BYTE_78[5]	KSV_BYTE_78[4]	KSV_BYTE_78[3]	KSV_BYTE_78[2]	KSV_BYTE_78[1]	KSV_BYTE_78[0]
0xCF	0x00	R/W	KSV_BYTE_79[7]	KSV_BYTE_79[6]	KSV_BYTE_79[5]	KSV_BYTE_79[4]	KSV_BYTE_79[3]	KSV_BYTE_79[2]	KSV_BYTE_79[1]	KSV_BYTE_79[0]
0xD0	0x00	R/W	KSV_BYTE_80[7]	KSV_BYTE_80[6]	KSV_BYTE_80[5]	KSV_BYTE_80[4]	KSV_BYTE_80[3]	KSV_BYTE_80[2]	KSV_BYTE_80[1]	KSV_BYTE_80[0]
0xD1	0x00	R/W	KSV_BYTE_81[7]	KSV_BYTE_81[6]	KSV_BYTE_81[5]	KSV_BYTE_81[4]	KSV_BYTE_81[3]	KSV_BYTE_81[2]	KSV_BYTE_81[1]	KSV_BYTE_81[0]
0xD2	0x00	R/W	KSV_BYTE_82[7]	KSV_BYTE_82[6]	KSV_BYTE_82[5]	KSV_BYTE_82[4]	KSV_BYTE_82[3]	KSV_BYTE_82[2]	KSV_BYTE_82[1]	KSV_BYTE_82[0]
0xD3	0x00	R/W	KSV_BYTE_83[7]	KSV_BYTE_83[6]	KSV_BYTE_83[5]	KSV_BYTE_83[4]	KSV_BYTE_83[3]	KSV_BYTE_83[2]	KSV_BYTE_83[1]	KSV_BYTE_83[0]
0xD4	0x00	R/W	KSV_BYTE_84[7]	KSV_BYTE_84[6]	KSV_BYTE_84[5]	KSV_BYTE_84[4]	KSV_BYTE_84[3]	KSV_BYTE_84[2]	KSV_BYTE_84[1]	KSV_BYTE_84[0]
0xD5	0x00	R/W	KSV_BYTE_85[7]	KSV_BYTE_85[6]	KSV_BYTE_85[5]	KSV_BYTE_85[4]	KSV_BYTE_85[3]	KSV_BYTE_85[2]	KSV_BYTE_85[1]	KSV_BYTE_85[0]
0xD6	0x00	R/W	KSV_BYTE_86[7]	KSV_BYTE_86[6]	KSV_BYTE_86[5]	KSV_BYTE_86[4]	KSV_BYTE_86[3]	KSV_BYTE_86[2]	KSV_BYTE_86[1]	KSV_BYTE_86[0]
0xD7	0x00	R/W	KSV_BYTE_87[7]	KSV_BYTE_87[6]	KSV_BYTE_87[5]	KSV_BYTE_87[4]	KSV_BYTE_87[3]	KSV_BYTE_87[2]	KSV_BYTE_87[1]	KSV_BYTE_87[0]
0xD8	0x00	R/W	KSV_BYTE_88[7]	KSV_BYTE_88[6]	KSV_BYTE_88[5]	KSV_BYTE_88[4]	KSV_BYTE_88[3]	KSV_BYTE_88[2]	KSV_BYTE_88[1]	KSV_BYTE_88[0]
0xD9	0x00	R/W	KSV_BYTE_89[7]	KSV_BYTE_89[6]	KSV_BYTE_89[5]	KSV_BYTE_89[4]	KSV_BYTE_89[3]	KSV_BYTE_89[2]	KSV_BYTE_89[1]	KSV_BYTE_89[0]
0xDA	0x00	R/W	KSV_BYTE_90[7]	KSV_BYTE_90[6]	KSV_BYTE_90[5]	KSV_BYTE_90[4]	KSV_BYTE_90[3]	KSV_BYTE_90[2]	KSV_BYTE_90[1]	KSV_BYTE_90[0]
0xDB	0x00	R/W	KSV_BYTE_91[7]	KSV_BYTE_91[6]	KSV_BYTE_91[5]	KSV_BYTE_91[4]	KSV_BYTE_91[3]	KSV_BYTE_91[2]	KSV_BYTE_91[1]	KSV_BYTE_91[0]
0xDC	0x00	R/W	KSV_BYTE_92[7]	KSV_BYTE_92[6]	KSV_BYTE_92[5]	KSV_BYTE_92[4]	KSV_BYTE_92[3]	KSV_BYTE_92[2]	KSV_BYTE_92[1]	KSV_BYTE_92[0]
0xDD	0x00	R/W	KSV_BYTE_93[7]	KSV_BYTE_93[6]	KSV_BYTE_93[5]	KSV_BYTE_93[4]	KSV_BYTE_93[3]	KSV_BYTE_93[2]	KSV_BYTE_93[1]	KSV_BYTE_93[0]
0xDE	0x00	R/W	KSV_BYTE_94[7]	KSV_BYTE_94[6]	KSV_BYTE_94[5]	KSV_BYTE_94[4]	KSV_BYTE_94[3]	KSV_BYTE_94[2]	KSV_BYTE_94[1]	KSV_BYTE_94[0]
0xDF	0x00	R/W	KSV_BYTE_95[7]	KSV_BYTE_95[6]	KSV_BYTE_95[5]	KSV_BYTE_95[4]	KSV_BYTE_95[3]	KSV_BYTE_95[2]	KSV_BYTE_95[1]	KSV_BYTE_95[0]
0xE0	0x00	R/W	KSV_BYTE_96[7]	KSV_BYTE_96[6]	KSV_BYTE_96[5]	KSV_BYTE_96[4]	KSV_BYTE_96[3]	KSV_BYTE_96[2]	KSV_BYTE_96[1]	KSV_BYTE_96[0]
0xE1	0x00	R/W	KSV_BYTE_97[7]	KSV_BYTE_97[6]	KSV_BYTE_97[5]	KSV_BYTE_97[4]	KSV_BYTE_97[3]	KSV_BYTE_97[2]	KSV_BYTE_97[1]	KSV_BYTE_97[0]
0xE2	0x00	R/W	KSV_BYTE_98[7]	KSV_BYTE_98[6]	KSV_BYTE_98[5]	KSV_BYTE_98[4]	KSV_BYTE_98[3]	KSV_BYTE_98[2]	KSV_BYTE_98[1]	KSV_BYTE_98[0]
0xE3	0x00	R/W	KSV_BYTE_99[7]	KSV_BYTE_99[6]	KSV_BYTE_99[5]	KSV_BYTE_99[4]	KSV_BYTE_99[3]	KSV_BYTE_99[2]	KSV_BYTE_99[1]	KSV_BYTE_99[0]
0xE4	0x00	R/W	KSV_BYTE_100[7]	KSV_BYTE_100[6]	KSV_BYTE_100[5]	KSV_BYTE_100[4]	KSV_BYTE_100[3]	KSV_BYTE_100[2]	KSV_BYTE_100[1]	KSV_BYTE_100[0]
0xE5	0x00	R/W	KSV_BYTE_101[7]	KSV_BYTE_101[6]	KSV_BYTE_101[5]	KSV_BYTE_101[4]	KSV_BYTE_101[3]	KSV_BYTE_101[2]	KSV_BYTE_101[1]	KSV_BYTE_101[0]
0xE6	0x00	R/W	KSV_BYTE_102[7]	KSV_BYTE_102[6]	KSV_BYTE_102[5]	KSV_BYTE_102[4]	KSV_BYTE_102[3]	KSV_BYTE_102[2]	KSV_BYTE_102[1]	KSV_BYTE_102[0]
0xE7	0x00	R/W	KSV_BYTE_103[7]	KSV_BYTE_103[6]	KSV_BYTE_103[5]	KSV_BYTE_103[4]	KSV_BYTE_103[3]	KSV_BYTE_103[2]	KSV_BYTE_103[1]	KSV_BYTE_103[0]
0xE8	0x00	R/W	KSV_BYTE_104[7]	KSV_BYTE_104[6]	KSV_BYTE_104[5]	KSV_BYTE_104[4]	KSV_BYTE_104[3]	KSV_BYTE_104[2]	KSV_BYTE_104[1]	KSV_BYTE_104[0]
0xE9	0x00	R/W	KSV_BYTE_105[7]	KSV_BYTE_105[6]	KSV_BYTE_105[5]	KSV_BYTE_105[4]	KSV_BYTE_105[3]	KSV_BYTE_105[2]	KSV_BYTE_105[1]	KSV_BYTE_105[0]
0xEA	0x00	R/W	KSV_BYTE_106[7]	KSV_BYTE_106[6]	KSV_BYTE_106[5]	KSV_BYTE_106[4]	KSV_BYTE_106[3]	KSV_BYTE_106[2]	KSV_BYTE_106[1]	KSV_BYTE_106[0]
0xEB	0x00	R/W	KSV_BYTE_107[7]	KSV_BYTE_107[6]	KSV_BYTE_107[5]	KSV_BYTE_107[4]	KSV_BYTE_107[3]	KSV_BYTE_107[2]	KSV_BYTE_107[1]	KSV_BYTE_107[0]
0xEC	0x00	R/W	KSV_BYTE_108[7]	KSV_BYTE_108[6]	KSV_BYTE_108[5]	KSV_BYTE_108[4]	KSV_BYTE_108[3]	KSV_BYTE_108[2]	KSV_BYTE_108[1]	KSV_BYTE_108[0]

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xED	0x00	R/W	KSV_BYTE_109[7]	KSV_BYTE_ 109[6]	KSV_BYTE_ 109[5]	KSV_BYTE_ 109[4]	KSV_BYTE_ 109[3]	KSV_BYTE_ 109[2]	KSV_BYTE_ 109[1]	KSV_BYTE_ 109[0]
0xEE	0x00	R/W	KSV_BYTE_110[7]	KSV_BYTE_ 110[6]	KSV_BYTE_ 110[5]	KSV_BYTE_ 110[4]	KSV_BYTE_ 110[3]	KSV_BYTE_ 110[2]	KSV_BYTE_ 110[1]	KSV_BYTE_ 110[0]
0xEF	0x00	R/W	KSV_BYTE_111[7]	KSV_BYTE_ 111[6]	KSV_BYTE_ 111[5]	KSV_BYTE_ 111[4]	KSV_BYTE_ 111[3]	KSV_BYTE_ 111[2]	KSV_BYTE_ 111[1]	KSV_BYTE_ 111[0]
0xF0	0x00	R/W	KSV_BYTE_112[7]	KSV_BYTE_ 112[6]	KSV_BYTE_ 112[5]	KSV_BYTE_ 112[4]	KSV_BYTE_ 112[3]	KSV_BYTE_ 112[2]	KSV_BYTE_ 112[1]	KSV_BYTE_ 112[0]
0xF1	0x00	R/W	KSV_BYTE_113[7]	KSV_BYTE_ 113[6]	KSV_BYTE_ 113[5]	KSV_BYTE_ 113[4]	KSV_BYTE_1 13[3]	KSV_BYTE_ 113[2]	KSV_BYTE_ 113[1]	KSV_BYTE _113[0]
0xF2	0x00	R/W	KSV_BYTE_114[7]	KSV_BYTE_ 114[6]	KSV_BYTE_ 114[5]	KSV_BYTE_ 114[4]	KSV_BYTE_ 114[3]	KSV_BYTE_ 114[2]	KSV_BYTE_ 114[1]	KSV_BYTE_ 114[0]
0xF3	0x00	R/W	KSV_BYTE_115[7]	KSV_BYTE_ 115[6]	KSV_BYTE_ 115[5]	KSV_BYTE_ 115[4]	KSV_BYTE_ 115[3]	KSV_BYTE_ 115[2]	KSV_BYTE_ 115[1]	KSV_BYTE_ 115[0]
0xF4	0x00	R/W	KSV_BYTE_116[7]	KSV_BYTE_ 116[6]	KSV_BYTE_ 116[5]	KSV_BYTE_ 116[4]	KSV_BYTE_ 116[3]	KSV_BYTE_ 116[2]	KSV_BYTE_ 116[1]	KSV_BYTE_ 116[0]
0xF5	0x00	R/W	KSV_BYTE_117[7]	KSV_BYTE_ 117[6]	KSV_BYTE_ 117[5]	KSV_BYTE_ 117[4]	KSV_BYTE_ 117[3]	KSV_BYTE_ 117[2]	KSV_BYTE_ 117[1]	KSV_BYTE_ 117[0]
0xF6	0x00	R/W	KSV_BYTE_118[7]	KSV_BYTE_ 118[6]	KSV_BYTE_ 118[5]	KSV_BYTE_ 118[4]	KSV_BYTE_ 118[3]	KSV_BYTE_ 118[2]	KSV_BYTE_ 118[1]	KSV_BYTE_ 118[0]
0xF7	0x00	R/W	KSV_BYTE_119[7]	KSV_BYTE_ 119[6]	KSV_BYTE_ 119[5]	KSV_BYTE_ 119[4]	KSV_BYTE_ 119[3]	KSV_BYTE_ 119[2]	KSV_BYTE_ 119[1]	KSV_BYTE_ 119[0]
0xF8	0x00	R/W	KSV_BYTE_120[7]	KSV_BYTE_ 120[6]	KSV_BYTE_ 120[5]	KSV_BYTE_ 120[4]	KSV_BYTE_ 120[3]	KSV_BYTE_ 120[2]	KSV_BYTE_ 120[1]	KSV_BYTE_ 120[0]
0xF9	0x00	R/W	KSV_BYTE_121[7]	KSV_BYTE_ 121[6]	KSV_BYTE_ 121[5]	KSV_BYTE_ 121[4]	KSV_BYTE_ 121[3]	KSV_BYTE_ 121[2]	KSV_BYTE_ 121[1]	KSV_BYTE_ 121[0]
0xFA	0x00	R/W	KSV_BYTE_122[7]	KSV_BYTE_ 122[6]	KSV_BYTE_ 122[5]	KSV_BYTE_ 122[4]	KSV_BYTE_ 122[3]	KSV_BYTE_ 122[2]	KSV_BYTE_ 122[1]	KSV_BYTE_ 122[0]
0xFB	0x00	R/W	KSV_BYTE_123[7]	KSV_BYTE_ 123[6]	KSV_BYTE_ 123[5]	KSV_BYTE_ 123[4]	KSV_BYTE_ 123[3]	KSV_BYTE_ 123[2]	KSV_BYTE_ 123[1]	KSV_BYTE_ 123[0]
0xFC	0x00	R/W	KSV_BYTE_124[7]	KSV_BYTE_ 124[6]	KSV_BYTE_ 124[5]	KSV_BYTE_ 124[4]	KSV_BYTE_ 124[3]	KSV_BYTE_ 124[2]	KSV_BYTE_ 124[1]	KSV_BYTE_ 124[0]
0xFD	0x00	R/W	KSV_BYTE_125[7]	KSV_BYTE_ 125[6]	KSV_BYTE_ 125[5]	KSV_BYTE_ 125[4]	KSV_BYTE_ 125[3]	KSV_BYTE_ 125[2]	KSV_BYTE_ 125[1]	KSV_BYTE_ 125[0]
0xFE	0x00	R/W	KSV_BYTE_126[7]	KSV_BYTE_ 126[6]	KSV_BYTE_ 126[5]	KSV_BYTE_ 126[4]	KSV_BYTE_ 126[3]	KSV_BYTE_ 126[2]	KSV_BYTE_ 126[1]	KSV_BYTE_ 126[0]
0xFF	0x00	R/W	KSV_BYTE_127[7]	KSV_BYTE_ 127[6]	KSV_BYTE_ 127[5]	KSV_BYTE_ 127[4]	KSV_BYTE_ 127[3]	KSV_BYTE_ 127[2]	KSV_BYTE_ 127[1]	KSV_BYTE_ 127[0]

INFOFRAME REGISTER MAP

Add is the register map I²C address, Def is the default value of the register, and Acc is the read/write access for the register (R means read only, and R/W means read/write access).

Table 5. ADV7613 InfoFrame Register Map

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x00	0x00	R	AVI_INF_PB[7]	AVI_INF_PB[6]	AVI_INF_PB[5]	AVI_INF_PB[4]	AVI_INF_PB[3]	AVI_INF_PB[2]	AVI_INF_PB[1]	AVI_INF_PB[0]
0x01	0x00	R	AVI_INF_PB[15]	AVI_INF_PB[14]	AVI_INF_PB[13]	AVI_INF_PB[12]	AVI_INF_PB[11]	AVI_INF_PB[10]	AVI_INF_PB[9]	AVI_INF_PB[8]
0x02	0x00	R	AVI_INF_PB[23]	AVI_INF_PB[22]	AVI_INF_PB[21]	AVI_INF_PB[20]	AVI_INF_PB[19]	AVI_INF_PB[18]	AVI_INF_PB[17]	AVI_INF_PB[16]
0x03	0x00	R	AVI_INF_PB[31]	AVI_INF_PB[30]	AVI_INF_PB[29]	AVI_INF_PB[28]	AVI_INF_PB[27]	AVI_INF_PB[26]	AVI_INF_PB[25]	AVI_INF_PB[24]
0x04	0x00	R	AVI_INF_PB[39]	AVI_INF_PB[38]	AVI_INF_PB[37]	AVI_INF_PB[36]	AVI_INF_PB[35]	AVI_INF_PB[34]	AVI_INF_PB[33]	AVI_INF_PB[32]
0x05	0x00	R	AVI_INF_PB[47]	AVI_INF_PB[46]	AVI_INF_PB[45]	AVI_INF_PB[44]	AVI_INF_PB[43]	AVI_INF_PB[42]	AVI_INF_PB[41]	AVI_INF_PB[40]
0x06	0x00	R	AVI_INF_PB[55]	AVI_INF_PB[54]	AVI_INF_PB[53]	AVI_INF_PB[52]	AVI_INF_PB[51]	AVI_INF_PB[50]	AVI_INF_PB[49]	AVI_INF_PB[48]
0x07	0x00	R	AVI_INF_PB[63]	AVI_INF_PB[62]	AVI_INF_PB[61]	AVI_INF_PB[60]	AVI_INF_PB[59]	AVI_INF_PB[58]	AVI_INF_PB[57]	AVI_INF_PB[56]
0x08	0x00	R	AVI_INF_PB[71]	AVI_INF_PB[70]	AVI_INF_PB[69]	AVI_INF_PB[68]	AVI_INF_PB[67]	AVI_INF_PB[66]	AVI_INF_PB[65]	AVI_INF_PB[64]
0x09	0x00	R	AVI_INF_PB[79]	AVI_INF_PB[78]	AVI_INF_PB[77]	AVI_INF_PB[76]	AVI_INF_PB[75]	AVI_INF_PB[74]	AVI_INF_PB[73]	AVI_INF_PB[72]
0x0A	0x00	R	AVI_INF_PB[87]	AVI_INF_PB[86]	AVI_INF_PB[85]	AVI_INF_PB[84]	AVI_INF_PB[83]	AVI_INF_PB[82]	AVI_INF_PB[81]	AVI_INF_PB[80]
0x0B	0x00	R	AVI_INF_PB[95]	AVI_INF_PB[94]	AVI_INF_PB[93]	AVI_INF_PB[92]	AVI_INF_PB[91]	AVI_INF_PB[90]	AVI_INF_PB[89]	AVI_INF_PB[88]
0x0C	0x00	R	AVI_INF_PB[103]	AVI_INF_PB[102]	AVI_INF_PB[101]	AVI_INF_PB[100]	AVI_INF_PB[99]	AVI_INF_PB[98]	AVI_INF_PB[97]	AVI_INF_PB[96]
0x0D	0x00	R	AVI_INF_PB[111]	AVI_INF_PB[110]	AVI_INF_PB[109]	AVI_INF_PB[108]	AVI_INF_PB[107]	AVI_INF_PB[106]	AVI_INF_PB[105]	AVI_INF_PB[104]
0x0E	0x00	R	AVI_INF_PB[119]	AVI_INF_PB[118]	AVI_INF_PB[117]	AVI_INF_PB[116]	AVI_INF_PB[115]	AVI_INF_PB[114]	AVI_INF_PB[113]	AVI_INF_PB[112]
0x0F	0x00	R	AVI_INF_PB[127]	AVI_INF_PB[126]	AVI_INF_PB[125]	AVI_INF_PB[124]	AVI_INF_PB[123]	AVI_INF_PB[122]	AVI_INF_PB[121]	AVI_INF_PB[120]
0x10	0x00	R	AVI_INF_PB[135]	AVI_INF_PB[134]	AVI_INF_PB[133]	AVI_INF_PB[132]	AVI_INF_PB[131]	AVI_INF_PB[130]	AVI_INF_PB[129]	AVI_INF_PB[128]
0x11	0x00	R	AVI_INF_PB[143]	AVI_INF_PB[142]	AVI_INF_PB[141]	AVI_INF_PB[140]	AVI_INF_PB[139]	AVI_INF_PB[138]	AVI_INF_PB[137]	AVI_INF_PB[136]
0x12	0x00	R	AVI_INF_PB[151]	AVI_INF_PB[150]	AVI_INF_PB[149]	AVI_INF_PB[148]	AVI_INF_PB[147]	AVI_INF_PB[146]	AVI_INF_PB[145]	AVI_INF_PB[144]
0x13	0x00	R	AVI_INF_PB[159]	AVI_INF_PB[158]	AVI_INF_PB[157]	AVI_INF_PB[156]	AVI_INF_PB[155]	AVI_INF_PB[154]	AVI_INF_PB[153]	AVI_INF_PB[152]
0x14	0x00	R	AVI_INF_PB[167]	AVI_INF_PB[166]	AVI_INF_PB[165]	AVI_INF_PB[164]	AVI_INF_PB[163]	AVI_INF_PB[162]	AVI_INF_PB[161]	AVI_INF_PB[160]
0x15	0x00	R	AVI_INF_PB[175]	AVI_INF_PB[174]	AVI_INF_PB[173]	AVI_INF_PB[172]	AVI_INF_PB[171]	AVI_INF_PB[170]	AVI_INF_PB[169]	AVI_INF_PB[168]
0x16	0x00	R	AVI_INF_PB[183]	AVI_INF_PB[182]	AVI_INF_PB[181]	AVI_INF_PB[180]	AVI_INF_PB[179]	AVI_INF_PB[178]	AVI_INF_PB[177]	AVI_INF_PB[176]
0x17	0x00	R	AVI_INF_PB[191]	AVI_INF_PB[190]	AVI_INF_PB[189]	AVI_INF_PB[188]	AVI_INF_PB[187]	AVI_INF_PB[186]	AVI_INF_PB[185]	AVI_INF_PB[184]
0x18	0x00	R	AVI_INF_PB[199]	AVI_INF_PB[198]	AVI_INF_PB[197]	AVI_INF_PB[196]	AVI_INF_PB[195]	AVI_INF_PB[194]	AVI_INF_PB[193]	AVI_INF_PB[192]
0x19	0x00	R	AVI_INF_PB[207]	AVI_INF_PB[206]	AVI_INF_PB[205]	AVI_INF_PB[204]	AVI_INF_PB[203]	AVI_INF_PB[202]	AVI_INF_PB[201]	AVI_INF_PB[200]
0x1A	0x00	R	AVI_INF_PB[215]	AVI_INF_PB[214]	AVI_INF_PB[213]	AVI_INF_PB[212]	AVI_INF_PB[211]	AVI_INF_PB[210]	AVI_INF_PB[209]	AVI_INF_PB[208]
0x1B	0x00	R	AVI_INF_PB[223]	AVI_INF_PB[222]	AVI_INF_PB[221]	AVI_INF_PB[220]	AVI_INF_PB[219]	AVI_INF _PB[218]	AVI_INF_PB[217]	AVI_INF_PB[216]
0x1C	0x00	R	AUD_INF_PB[7]	AUD_INF_PB[6]	AUD_INF_PB[5]	AUD_INF_PB[4]	AUD_INF_PB[3]	AUD_INF_PB[2]	AUD_INF_PB[1]	AUD_INF_PB[0]
0x1D	0x00	R	AUD_INF_PB[15]	AUD_INF_PB[14]	AUD_INF_PB[13]	AUD_INF_PB[12]	AUD_INF_PB[11]	AUD_INF_PB[10]	AUD_INF_PB[9]	AUD_INF_PB[8]
0x1E	0x00	R	AUD_INF_PB[23]	AUD_INF_PB[22]	AUD_INF_PB[21]	AUD_INF_PB[20]	AUD_INF_PB[19]	AUD_INF_PB[18]	AUD_INF_PB[17]	AUD_INF_PB[16]
0x1F	0x00	R	AUD_INF_PB[31]	AUD_INF_PB[30]	AUD_INF_PB[29]	AUD_INF_PB[28]	AUD_INF_PB[27]	AUD_INF_PB[26]	AUD_INF_PB[25]	AUD_INF_PB[24]
0x20	0x00	R	AUD_INF_PB[39]	AUD_INF_PB[38]	AUD_INF_PB[37]	AUD_INF_PB[36]	AUD_INF_PB[35]	AUD_INF_PB[34]	AUD_INF_PB[33]	AUD_INF_PB[32]
0x21	0x00	R	AUD_INF_PB[47]	AUD_INF_PB[46]	AUD_INF_PB[45]	AUD_INF_PB[44]	AUD_INF_PB[43]	AUD_INF_PB[42]	AUD_INF_PB[41]	AUD_INF_PB[40]

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x22	0x00	R	AUD_INF_PB[55]	AUD_INF_PB[54]	AUD_INF_PB[53]	AUD_INF_PB[52]	AUD_INF_PB[51]	AUD_INF_PB[50]	AUD_INF_PB[49]	AUD_INF_PB[48]
0x23	0x00	R	AUD_INF_PB[63]	AUD_INF_PB[62]	AUD_INF_PB[61]	AUD_INF_PB[60]	AUD_INF_PB[59]	AUD_INF_PB[58]	AUD_INF_PB[57]	AUD_INF_PB[56]
0x24	0x00	R	AUD_INF_PB[71]	AUD_INF_PB[70]	AUD_INF_PB[69]	AUD_INF_PB[68]	AUD_INF_PB[67]	AUD_INF_PB[66]	AUD_INF_PB[65]	AUD_INF_PB[64]
0x25	0x00	R	AUD_INF_PB[79]	AUD_INF_PB[78]	AUD_INF_PB[77]	AUD_INF_PB[76]	AUD_INF_PB[75]	AUD_INF_PB[74]	AUD_INF_PB[73]	AUD_INF_PB[72]
0x26	0x00	R	AUD_INF_PB[87]	AUD_INF_PB[86]	AUD_INF_PB[85]	AUD_INF_PB[84]	AUD_INF_PB[83]	AUD_INF_PB[82]	AUD_INF_PB[81]	AUD_INF_PB[80]
0x27	0x00	R	AUD_INF_PB[95]	AUD_INF_PB[94]	AUD_INF_PB[93]	AUD_INF_B[92]	AUD_INF_PB[91]	AUD_INF_PB[90]	AUD_INF_PB[89]	AUD_INF_PB[88]
0x28	0x00	R	AUD_INF_PB[103]	AUD_INF_PB[102]	AUD_INF_PB[101]	AUD_INF_PB[100]	AUD_INF_PB[99]	AUD_INF_PB[98]	AUD_INF_PB[97]	AUD_INF_PB[96]
0x29	0x00	R	AUD_INF_PB[111]	AUD_INF_PB[110]	AUD_INF_PB[109]	AUD_INF_PB[108]	AUD_INF_PB[107]	AUD_INF_PB[106]	AUD_INF_PB[105]	AUD_INF_PB[104]
0x2A	0x00	R	SPD_INF_PB[7]	SPD_INF_PB[6]	SPD_INF_PB[5]	SPD_INF_PB[4]	SPD_INF_PB[3]	SPD_INF_PB[2]	SPD_INF_PB[1]	SPD_INF_PB[0]
0x2B	0x00	R	SPD_INF_PB[15]	SPD_INF_PB[14]	SPD_INF_PB[13]	SPD_INF_PB[12]	SPD_INF_PB[11]	SPD_INF_PB[10]	SPD_INF_PB[9]	SPD_INF_PB[8]
0x2C	0x00	R	SPD_INF_PB[23]	SPD_INF_PB[22]	SPD_INF_PB[21]	SPD_INF_PB[20]	SPD_INF_PB[19]	SPD_INF_PB[18]	SPD_INF_PB[17]	SPD_INF_PB[16]
0x2D	0x00	R	SPD_INF_PB[31]	SPD_INF_PB[30]	SPD_INF_PB[29]	SPD_INF_PB[28]	SPD_INF_PB[27]	SPD_INF_PB[26]	SPD_INF_PB[25]	SPD_INF_PB[24]
0x2E	0x00	R	SPD_INF_PB[39]	SPD_INF_PB[38]	SPD_INF_PB[37]	SPD_INF_PB[36]	SPD_INF_PB[35]	SPD_INF_PB[34]	SPD_INF_PB[33]	SPD_INF_PB[32]
0x2F	0x00	R	SPD_INF_PB[47]	SPD_INF_PB[46]	SPD_INF_PB[45]	SPD_INF_PB[44]	SPD_INF_PB[43]	SPD_INF_PB[42]	SPD_INF_PB[41]	SPD_INF_PB[40]
0x30	0x00	R	SPD_INF_PB[55]	SPD_INF_PB[54]	SPD_INF_PB[53]	SPD_INF_PB[52]	SPD_INF_PB[51]	SPD_INF_PB[50]	SPD_INF_PB[49]	SPD_INF_PB[48]
0x31	0x00	R	SPD_INF_PB[63]	SPD_INF_PB[62]	SPD_INF_PB[61]	SPD_INF_PB[60]	SPD_INF_PB[59]	SPD_INF_PB[58]	SPD_INF_PB[57]	SPD_INF_PB[56]
0x32	0x00	R	SPD_INF_PB[71]	SPD_INF_PB[70]	SPD_INF_PB[69]	SPD_INF_PB[68]	SPD_INF_PB[67]	SPD_INF_PB[66]	SPD_INF_PB[65]	SPD_INF_PB[64]
0x33	0x00	R	SPD_INF_PB[79]	SPD_INF_PB[78]	SPD_INF_PB[77]	SPD_INF_PB[76]	SPD_INF_PB[75]	SPD_INF_PB[74]	SPD_INF_PB[73]	SPD_INF_PB[72]
0x34	0x00	R	SPD_INF_PB[87]	SPD_INF_PB[86]	SPD_INF_PB[85]	SPD_INF_PB[84]	SPD_INF_PB[83]	SPD_INF_PB[82]	SPD_INF_PB[81]	SPD_INF_PB[80]
0x35	0x00	R	SPD_INF_PB[95]	SPD_INF_PB[94]	SPD_INF_PB[93]	SPD_INF_PB[92]	SPD_INF_PB[91]	SPD_INF_PB[90]	SPD_INF_PB[89]	SPD_INF_PB[88]
0x36	0x00	R	SPD_INF_PB[103]	SPD_INF_PB[102]	SPD_INF_PB[101]	SPD_INF_PB[100]	SPD_INF_PB[99]	SPD_INF_PB[98]	SPD_INF_PB[97]	SPD_INF_PB[96]
0x37	0x00	R	SPD_INF_PB[111]	SPD_INF_PB[110]	SPD_INF_PB[109]	SPD_INF_PB[108]	SPD_INF_PB[107]	SPD_INF_PB[106]	SPD_INF_PB[105]	SPD_INF_PB[104]
0x38	0x00	R	SPD_INF_PB[119]	SPD_INF_PB[118]	SPD_INF_PB[117]	SPD_INF_PB[116]	SPD_INF_PB[115]	SPD_INF_PB[114]	SPD_INF_PB[113]	SPD_INF_PB[112]
0x39	0x00	R	SPD_INF_PB[127]	SPD_INF_PB[126]	SPD_INF_PB[125]	SPD_INF_PB[124]	SPD_INF_PB[123]	SPD_INF_PB[122]	SPD_INF_PB[121]	SPD_INF_PB[120]
0x3A	0x00	R	SPD_INF_PB[135]	SPD_INF_PB[134]	SPD_INF_PB[133]	SPD_INF_PB[132]	SPD_INF_PB[131]	SPD_INF_PB[130]	SPD_INF_PB[129]	SPD_INF_PB[128]
0x3B	0x00	R	SPD_INF_PB[143]	SPD_INF_PB[142]	SPD_INF_PB[141]	SPD_INF_PB[140]	SPD_INF_PB[139]	SPD_INF_PB[138]	SPD_INF_PB[137]	SPD_INF_PB[136]
0x3C	0x00	R	SPD_INF_PB[151]	SPD_INF_PB[150]	SPD_INF_PB[149]	SPD_INF_PB[148]	SPD_INF_PB[147]	SPD_INF_PB[146]	SPD_INF_PB[145]	SPD_INF_PB[144]
0x3D	0x00	R	SPD_INF_PB[159]	SPD_INF_PB[158]	SPD_INF_PB[157]	SPD_INF_PB[156]	SPD_INF_PB[155]	SPD_INF_PB[154]	SPD_INF_PB[153]	SPD_INF_PB[152]
0x3E	0x00	R	SPD_INF_PB[167]	SPD_INF_PB[166]	SPD_INF_PB[165]	SPD_INF_PB[164]	SPD_INF_PB[163]	SPD_INF_PB[162]	SPD_INF_PB[161]	SPD_INF_PB[160]
0x3F	0x00	R	SPD_INF_PB[175]	SPD_INF_PB[174]	SPD_INF_PB[173]	SPD_INF_PB[172]	SPD_INF_PB[171]	SPD_INF_PB[170]	SPD_INF_PB[169]	SPD_INF_PB[168]
0x40	0x00	R	SPD_INF_PB[183]	SPD_INF_PB[182]	SPD_INF_PB[181]	SPD_INF_PB[180]	SPD_INF_PB[179]	SPD_INF_PB[178]	SPD_INF_PB[177]	SPD_INF_PB[176]
0x41	0x00	R	SPD_INF_PB[191]	SPD_INF_PB[190]	SPD_INF_PB[189]	SPD_INF_PB[188]	SPD_INF_PB[187]	SPD_INF_PB[186]	SPD_INF_PB[185]	SPD_INF_PB[184]
0x42	0x00	R	SPD_INF_PB[199]	SPD_INF_PB[198]	SPD_INF_PB[197]	SPD_INF_PB[196]	SPD_INF_PB[195]	SPD_INF_PB[194]	SPD_INF_PB[193]	SPD_INF_PB[192]
0x43	0x00	R	SPD_INF_PB[207]	SPD_INF_PB[206]	SPD_INF_PB[205]	SPD_INF_PB[204]	SPD_INF_PB[203]	SPD_INF_PB[202]	SPD_INF_PB[201]	SPD_INF_PB[200]
0x44	0x00	R	SPD_INF_PB[215]	SPD_INF_PB[214]	SPD_INF_PB[213]	SPD_INF_PB[212]	SPD_INF_PB[211]	SPD_INF_PB[210]	SPD_INF_PB[209]	SPD_INF_PB[208]
0x45	0x00	R	SPD_INF_PB[223]	SPD_INF_PB[222]	SPD_INF_PB[221]	SPD_INF_PB[220]	SPD_INF_PB[219]	SPD_INF_PB[218]	SPD_INF_PB[217]	SPD_INF_PB[216]
0x46	0x00	R	MS_INF_PB[7]	MS_INF_PB[6]	MS_INF_PB[5]	MS_INF_PB[4]	MS_INF_PB[3]	MS_INF_PB[2]	MS_INF_PB[1]	MS_INF_PB[0]
0x47	0x00	R	MS_INF_PB[15]	MS_INF_PB[14]	MS_INF_PB[13]	MS_INF_PB[12]	MS_INF_PB[11]	MS_INF_PB[10]	MS_INF_PB[9]	MS_INF_PB[8]

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x48	0x00	R	MS_INF_PB[23]	MS_INF_PB[22]	MS_INF_PB[21]	MS_INF_PB[20]	MS_INF_PB[19]	MS_INF_PB[18]	MS_INF_PB[17]	MS_INF_PB[16]
0x49	0x00	R	MS_INF_PB[31]	MS_INF_PB[30]	MS_INF_PB[29]	MS_INF_PB[28]	MS_INF_PB[27]	MS_INF_PB[26]	MS_INF_PB[25]	MS_INF_PB[24]
0x4A	0x00	R	MS_INF_PB[39]	MS_INF_PB[38]	MS_INF_PB[37]	MS_INF_PB[36]	MS_INF_PB[35]	MS_INF_PB[34]	MS_INF_PB[33]	MS_INF_PB[32]
0x4B	0x00	R	MS_INF_PB[47]	MS_INF_PB[46]	MS_INF_PB[45]	MS_INF_PB[44]	MS_INF_PB[43]	MS_INF_PB[42]	MS_INF_PB[41]	MS_INF_PB[40]
0x4C	0x00	R	MS_INF_PB[55]	MS_INF_PB[54]	MS_INF_PB[53]	MS_INF_PB[52]	MS_INF_PB[51]	MS_INF_PB[50]	MS_INF_PB[49]	MS_INF_PB[48]
0x4D	0x00	R	MS_INF_PB[63]	MS_INF_PB[62]	MS_INF_PB[61]	MS_INF_PB[60]	MS_INF_PB[59]	MS_INF_PB[58]	MS_INF_PB[57]	MS_INF_PB[56]
0x4E	0x00	R	MS_INF_PB[71]	MS_INF_PB[70]	MS_INF_PB[69]	MS_INF_PB[68]	MS_INF_PB[67]	MS_INF_PB[66]	MS_INF_PB[65]	MS_INF_PB[64]
0x4F	0x00	R	MS_INF_PB[79]	MS_INF_PB[78]	MS_INF_PB[77]	MS_INF_PB[76]	MS_INF_PB[75]	MS_INF_PB[74]	MS_INF_PB[73]	MS_INF_PB[72]
0x50	0x00	R	MS_INF_PB[87]	MS_INF_PB[86]	MS_INF_PB[85]	MS_INF_PB[84]	MS_INF_PB[83]	MS_INF_PB[82]	MS_INF_PB[81]	MS_INF_PB[80]
0x51	0x00	R	MS_INF_PB[95]	MS_INF_PB[94]	MS_INF_PB[93]	MS_INF_PB[92]	MS_INF_PB[91]	MS_INF_PB[90]	MS_INF_PB[89]	MS_INF_PB[88]
0x52	0x00	R	MS_INF_PB[103]	MS_INF_PB[102]	MS_INF_PB[101]	MS_INF_PB[100]	MS_INF_PB[99]	MS_INF_PB[98]	MS_INF_PB[97]	MS_INF_PB[96]
0x53	0x00	R	MS_INF_PB[111]	MS_INF_PB[110]	MS_INF_PB[109]	MS_INF_PB[108]	MS_INF_PB[107]	MS_INF_PB[106]	MS_INF_PB[105]	MS_INF_PB[104]
0x54	0x00	R	VS_INF_PB[7]	VS_INF_PB[6]	VS_INF_PB[5]	VS_INF_PB[4]	VS_INF_PB[3]	VS_INF_PB[2]	VS_INF_PB[1]	VS_INF_PB[0]
0x55	0x00	R	VS_INF_PB[15]	VS_INF_PB[14]	VS_INF_PB[13]	VS_INF_PB[12]	VS_INF_PB[11]	VS_INF_PB[10]	VS_INF_PB[9]	VS_INF_PB[8]
0x56	0x00	R	VS_INF_PB[23]	VS_INF_PB[22]	VS_INF_PB[21]	VS_INF_PB[20]	VS_INF_PB[19]	VS_INF_PB[18]	VS_INF_PB[17]	VS_INF_PB[16]
0x57	0x00	R	VS_INF_PB[31]	VS_INF_PB[30]	VS_INF_PB[29]	VS_INF_PB[28]	VS_INF_PB[27]	VS_INF_PB[26]	VS_INF_PB[25]	VS_INF_PB[24]
0x58	0x00	R	VS_INF_PB[39]	VS_INF_PB[38]	VS_INF_PB[37]	VS_INF_PB[36]	VS_INF_PB[35]	VS_INF_PB[34]	VS_INF_PB[33]	VS_INF_PB[32]
0x59	0x00	R	VS_INF_PB[47]	VS_INF_PB[46]	VS_INF_PB[45]	VS_INF_PB[44]	VS_INF_PB[43]	VS_INF_PB[42]	VS_INF_PB[41]	VS_INF_PB[40]
0x5A	0x00	R	VS_INF_PB[55]	VS_INF_PB[54]	VS_INF_PB[53]	VS_INF_PB[52]	VS_INF_PB[51]	VS_INF_PB[50]	VS_INF_PB[49]	VS_INF_PB[48]
0x5B	0x00	R	VS_INF_PB[63]	VS_INF_PB[62]	VS_INF_PB[61]	VS_INF_PB[60]	VS_INF_PB[59]	VS_INF_PB[58]	VS_INF_PB[57]	VS_INF_PB[56]
0x5C	0x00	R	VS_INF_PB[71]	VS_INF_PB[70]	VS_INF_PB[69]	VS_INF_PB[68]	VS_INF_PB[67]	VS_INF_PB[66]	VS_INF_PB[65]	VS_INF_PB[64]
0x5D	0x00	R	VS_INF_PB[79]	VS_INF_PB[78]	VS_INF_PB[77]	VS_INF_PB[76]	VS_INF_PB[75]	VS_INF_PB[74]	VS_INF_PB[73]	VS_INF_PB[72]
0x5E	0x00	R	VS_INF_PB[87]	VS_INF_PB[86]	VS_INF_PB[85]	VS_INF_PB[84]	VS_INF_PB[83]	VS_INF_PB[82]	VS_INF_PB[81]	VS_INF_PB[80]
0x5F	0x00	R	VS_INF_PB[95]	VS_INF_PB[94]	VS_INF_PB[93]	VS_INF_PB[92]	VS_INF_PB[91]	VS_INF_PB[90]	VS_INF_PB[89]	VS_INF_PB[88]
0x60	0x00	R	VS_INF_PB[103]	VS_INF_PB[102]	VS_INF_PB[101]	VS_INF_PB[100]	VS_INF_PB[99]	VS_INF_PB[98]	VS_INF_PB[97]	VS_INF_PB[96]
0x61	0x00	R	VS_INF_PB[111]	VS_INF_PB[110]	VS_INF_PB[109]	VS_INF_PB[108]	VS_INF_PB[107]	VS_INF_PB[106]	VS_INF_PB[105]	VS_INF_PB[104]
0x62	0x00	R	VS_INF_PB[119]	VS_INF_PB[118]	VS_INF_PB[117]	VS_INF_PB[116]	VS_INF_PB[115]	VS_INF_PB[114]	VS_INF_PB[113]	VS_INF_PB[112]
0x63	0x00	R	VS_INF_PB[127]	VS_INF_PB[126]	VS_INF_PB[125]	VS_INF_PB[124]	VS_INF_PB[123]	VS_INF_PB[122]	VS_INF_PB[121]	VS_INF_PB[120]
0x64	0x00	R	VS_INF_PB[135]	VS_INF_PB[134]	VS_INF_PB[133]	VS_INF_PB[132]	VS_INF_PB[131]	VS_INF_PB[130]	VS_INF_PB[129]	VS_INF_PB[128]
0x65	0x00	R	VS_INF_PB[143]	VS_INF_PB[142]	VS_INF_PB[141]	VS_INF_PB[140]	VS_INF_PB[139]	VS_INF_PB[138]	VS_INF_PB[137]	VS_INF_PB[136]
0x66	0x00	R	VS_INF_PB[151]	VS_INF_PB[150]	VS_INF_PB[149]	VS_INF_PB[148]	VS_INF_PB[147]	VS_INF_PB[146]	VS_INF_PB[145]	VS_INF_PB[144]
0x67	0x00	R	VS_INF_PB[159]	VS_INF_PB[158]	VS_INF_PB[157]	VS_INF_PB[156]	VS_INF_PB[155]	VS_INF_PB[154]	VS_INF_PB[153]	VS_INF_PB[152]
0x68	0x00	R	VS_INF_PB[167]	VS_INF_PB[166]	VS_INF_PB[165]	VS_INF_PB[164]	VS_INF_PB[163]	VS_INF_PB[162]	VS_INF_PB[161]	VS_INF_PB[160]
0x69	0x00	R	VS_INF_PB[175]	VS_INF_PB[174]	VS_INF_PB[173]	VS_INF_PB[172]	VS_INF_PB[171]	VS_INF_PB[170]	VS_INF_PB[169]	VS_INF_PB[168]
0x6A	0x00	R	VS_INF_PB[183]	VS_INF_PB[182]	VS_INF_PB[181]	VS_INF_PB[180]	VS_INF_PB[179]	VS_INF_PB[178]	VS_INF_PB[177]	VS_INF_PB[176]
0x6B	0x00	R	VS_INF_PB[191]	VS_INF_PB[190]	VS_INF_PB[189]	VS_INF_PB[188]	VS_INF_PB[187]	VS_INF_PB[186]	VS_INF_PB[185]	VS_INF_PB[184]
0x6C	0x00	R	VS_INF_PB[199]	VS_INF_PB[198]	VS_INF_PB[197]	VS_INF_PB[196]	VS_INF_PB[195]	VS_INF_PB[194]	VS_INF_PB[193]	VS_INF_PB[192]
0x6D	0x00	R	VS_INF_PB[207]	VS_INF_PB[206]	VS_INF_PB[205]	VS_INF_PB[204]	VS_INF_PB[203]	VS_INF_PB[202]	VS_INF_PB[201]	VS_INF_PB[200]
0x6E	0x00	R	VS_INF_PB[215]	VS_INF_PB[214]	VS_INF_PB[213]	VS_INF_PB[212]	VS_INF_PB[211]	VS_INF_PB[210]	VS_INF_PB[209]	VS_INF_PB[208]
0x6F	0x00	R	VS_INF_PB[223]	VS_INF_PB[222]	VS_INF_PB[221]	VS_INF_PB[220]	VS_INF_PB[219]	VS_INF_PB[218]	VS_INF_PB[217]	VS_INF_PB[216]
0x70	0x00	R	ACP_PB[7]	ACP_PB[6]	ACP_PB[5]	ACP_PB[4]	ACP_PB[3]	ACP_PB[2]	ACP_PB[1]	ACP_PB[0]
0x71	0x00	R	ACP_PB[15]	ACP_PB[14]	ACP_PB[13]	ACP_PB[12]	ACP_PB[11]	ACP_PB[10]	ACP_PB[9]	ACP_PB[8]
0x72	0x00	R	ACP_PB[23]	ACP_PB[22]	ACP_PB[21]	ACP_PB[20]	ACP_PB[19]	ACP_PB[18]	ACP_PB[17]	ACP_PB[16]
0x73	0x00	R	ACP_PB[31]	ACP_PB[30]	ACP_PB[29]	ACP_PB[28]	ACP_PB[27]	ACP_PB[26]	ACP_PB[25]	ACP_PB[24]
0x74	0x00	R	ACP_PB[39]	ACP_PB[38]	ACP_PB[37]	ACP_PB[36]	ACP_PB[35]	ACP_PB[34]	ACP_PB[33]	ACP_PB[32]
0x75	0x00	R	ACP_PB[47]	ACP_PB[46]	ACP_PB[45]	ACP_PB[44]	ACP_PB[43]	ACP_PB[42]	ACP_PB[41]	ACP_PB[40]
0x76	0x00	R	ACP_PB[55]	ACP_PB[54]	ACP_PB[53]	ACP_PB[52]	ACP_PB[51]	ACP_PB[50]	ACP_PB[49]	ACP_PB[48]
0x77	0x00	R	ACP_PB[63]	ACP_PB[62]	ACP_PB[61]	ACP_PB[60]	ACP_PB[59]	ACP_PB[58]	ACP_PB[57]	ACP_PB[56]
0x78	0x00	R	ACP_PB[71]	ACP_PB[70]	ACP_PB[69]	ACP_PB[68]	ACP_PB[67]	ACP_PB[66]	ACP_PB[65]	ACP_PB[64]
0x79	0x00	R	ACP_PB[79]	ACP_PB[78]	ACP_PB[77]	ACP_PB[76]	ACP_PB[75]	ACP_PB[74]	ACP_PB[73]	ACP_PB[72]
0x7A	0x00	R	ACP_PB[87]	ACP_PB[86]	ACP_PB[85]	ACP_PB[84]	ACP_PB[83]	ACP_PB[82]	ACP_PB[81]	ACP_PB[80]
0x7B	0x00	R	ACP_PB[95]	ACP_PB[94]	ACP_PB[93]	ACP_PB[92]	ACP_PB[91]	ACP_PB[90]	ACP_PB[89]	ACP_PB[88]
0x7C	0x00	R	ACP_PB[103]	ACP_PB[102]	ACP_PB[101]	ACP_PB[100]	ACP_PB[99]	ACP_PB[98]	ACP_PB[97]	ACP_PB[96]
0x7D	0x00	R	ACP_PB[111]	ACP_PB[110]	ACP_PB[109]	ACP_PB[108]	ACP_PB[107]	ACP_PB[106]	ACP_PB[105]	ACP_PB[104]
0x7E	0x00	R	ACP_PB[119]	ACP_PB[118]	ACP_PB[117]	ACP_PB[116]	ACP_PB[115]	ACP_PB[114]	ACP_PB[113]	ACP_PB[112]

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x7F	0x00	R	ACP_PB[127]	ACP_PB[126]	ACP_PB[125]	ACP_PB[124]	ACP_PB[123]	ACP_PB[122]	ACP_PB[121]	ACP_PB[120]
0x80	0x00	R	ACP_PB[135]	ACP_PB[134]	ACP_PB[133]	ACP_PB[132]	ACP_PB[131]	ACP_PB[130]	ACP_PB[129]	ACP_PB[128]
0x81	0x00	R	ACP_PB[143]	ACP_PB[142]	ACP_PB[141]	ACP_PB[140]	ACP_PB[139]	ACP_PB[138]	ACP_PB[137]	ACP_PB[136]
0x82	0x00	R	ACP_PB[151]	ACP_PB[150]	ACP_PB[149]	ACP_PB[148]	ACP_PB[147]	ACP_PB[146]	ACP_PB[145]	ACP_PB[144]
0x83	0x00	R	ACP_PB[159]	ACP_PB[158]	ACP_PB[157]	ACP_PB[156]	ACP_PB[155]	ACP_PB[154]	ACP_PB[153]	ACP_PB[152]
0x84	0x00	R	ACP_PB[167]	ACP_PB[166]	ACP_PB[165]	ACP_PB[164]	ACP_PB[163]	ACP_PB[162]	ACP_PB[161]	ACP_PB[160]
0x85	0x00	R	ACP_PB[175]	ACP_PB[174]	ACP_PB[173]	ACP_PB[172]	ACP_PB[171]	ACP_PB[170]	ACP_PB[169]	ACP_PB[168]
0x86	0x00	R	ACP_PB[183]	ACP_PB[182]	ACP_PB[181]	ACP_PB[180]	ACP_PB[179]	ACP_PB[178]	ACP_PB[177]	ACP_PB[176]
0x87	0x00	R	ACP_PB[191]	ACP_PB[190]	ACP_PB[189]	ACP_PB[188]	ACP_PB[187]	ACP_PB[186]	ACP_PB[185]	ACP_PB[184]
0x88	0x00	R	ACP_PB[199]	ACP_PB[198]	ACP_PB[197]	ACP_PB[196]	ACP_PB[195]	ACP_PB[194]	ACP_PB[193]	ACP_PB[192]
0x89	0x00	R	ACP_PB[207]	ACP_PB[206]	ACP_PB[205]	ACP_PB[204]	ACP_PB[203]	ACP_PB[202]	ACP_PB[201]	ACP_PB[200]
0x8A	0x00	R	ACP_PB[215]	ACP_PB[214]	ACP_PB[213]	ACP_PB[212]	ACP_PB[211]	ACP_PB[210]	ACP_PB[209]	ACP_PB[208]
0x8B	0x00	R	ACP_PB[223]	ACP_PB[222]	ACP_PB[221]	ACP_PB[220]	ACP_PB[219]	ACP_PB[218]	ACP_PB[217]	ACP_PB[216]
0x8C	0x00	R	ISRC1_PB[7]	ISRC1_PB[6]	ISRC1_PB[5]	ISRC1_PB[4]	ISRC1_PB[3]	ISRC1_PB[2]	ISRC1_PB[1]	ISRC1_PB[0]
0x8D	0x00	R	ISRC1_PB[15]	ISRC1_PB[14]	ISRC1_PB[13]	ISRC1_PB[12]	ISRC1_PB[11]	ISRC1_PB[10]	ISRC1_PB[9]	ISRC1_PB[8]
0x8E	0x00	R	ISRC1_PB[23]	ISRC1_PB[22]	ISRC1_PB[21]	ISRC1_PB[20]	ISRC1_PB[19]	ISRC1_PB[18]	ISRC1_PB[17]	ISRC1_PB[16]
0x8F	0x00	R	ISRC1_PB[31]	ISRC1_PB[30]	ISRC1_PB[29]	ISRC1_PB[28]	ISRC1_PB[27]	ISRC1_PB[26]	ISRC1_PB[25]	ISRC1_PB[24]
0x90	0x00	R	ISRC1_PB[39]	ISRC1_PB[38]	ISRC1_PB[37]	ISRC1_PB[36]	ISRC1_PB[35]	ISRC1_PB[34]	ISRC1_PB[33]	ISRC1_PB[32]
0x91	0x00	R	ISRC1_PB[47]	ISRC1_PB[46]	ISRC1_PB[45]	ISRC1_PB[44]	ISRC1_PB[43]	ISRC1_PB[42]	ISRC1_PB[41]	ISRC1_PB[40]
0x92	0x00	R	ISRC1_PB[55]	ISRC1_PB[54]	ISRC1_PB[53]	ISRC1_PB[52]	ISRC1_PB[51]	ISRC1_PB[50]	ISRC1_PB[49]	ISRC1_PB[48]
0x93	0x00	R	ISRC1_PB[63]	ISRC1_PB[62]	ISRC1_PB[61]	ISRC1_PB[60]	ISRC1_PB[59]	ISRC1_PB[58]	ISRC1_PB[57]	ISRC1_PB[56]
0x94	0x00	R	ISRC1_PB[71]	ISRC1_PB[70]	ISRC1_PB[69]	ISRC1_PB[68]	ISRC1_PB[67]	ISRC1_PB[66]	ISRC1_PB[65]	ISRC1_PB[64]
0x95	0x00	R	ISRC1_PB[79]	ISRC1_PB[78]	ISRC1_PB[77]	ISRC1_PB[76]	ISRC1_PB[75]	ISRC1_PB[74]	ISRC1_PB[73]	ISRC1_PB[72]
0x96	0x00	R	ISRC1_PB[87]	ISRC1_PB[86]	ISRC1_PB[85]	ISRC1_PB[84]	ISRC1_PB[83]	ISRC1_PB[82]	ISRC1_PB[81]	ISRC1_PB[80]
0x97	0x00	R	ISRC1_PB[95]	ISRC1_PB[94]	ISRC1_PB[93]	ISRC1_PB[92]	ISRC1_PB[91]	ISRC1_PB[90]	ISRC1_PB[89]	ISRC1_PB[88]
0x98	0x00	R	ISRC1_PB[103]	ISRC1_PB[102]	ISRC1_PB[101]	ISRC1_PB[100]	ISRC1_PB[99]	ISRC1_PB[98]	ISRC1_PB[97]	ISRC1_PB[96]
0x99	0x00	R	ISRC1_PB[111]	ISRC1_PB[110]	ISRC1_PB[109]	ISRC1_PB[108]	ISRC1_PB[107]	ISRC1_PB[106]	ISRC1_PB[105]	ISRC1_PB[104]
0x9A	0x00	R	ISRC1_PB[119]	ISRC1_PB[118]	ISRC1_PB[117]	ISRC1_PB[116]	ISRC1_PB[115]	ISRC1_PB[114]	ISRC1_PB[113]	ISRC1_PB[112]
0x9B	0x00	R	ISRC1_PB[127]	ISRC1_PB[126]	ISRC1_PB[125]	ISRC1_PB[124]	ISRC1_PB[123]	ISRC1_PB[122]	ISRC1_PB[121]	ISRC1_PB[120]
0x9C	0x00	R	ISRC1_PB[135]	ISRC1_PB[134]	ISRC1_PB[133]	ISRC1_PB[132]	ISRC1_PB[131]	ISRC1_PB[130]	ISRC1_PB[129]	ISRC1_PB[128]
0x9D	0x00	R	ISRC1_PB[143]	ISRC1_PB[142]	ISRC1_PB[141]	ISRC1_PB[140]	ISRC1_PB[139]	ISRC1_PB[138]	ISRC1_PB[137]	ISRC1_PB[136]
0x9E	0x00	R	ISRC1_PB[151]	ISRC1_PB[150]	ISRC1_PB[149]	ISRC1_PB[148]	ISRC1_PB[147]	ISRC1_PB[146]	ISRC1_PB[145]	ISRC1_PB[144]
0x9F	0x00	R	ISRC1_PB[159]	ISRC1_PB[158]	ISRC1_PB[157]	ISRC1_PB[156]	ISRC1_PB[155]	ISRC1_PB[154]	ISRC1_PB[153]	ISRC1_PB[152]
0xA0	0x00	R	ISRC1_PB[167]	ISRC1_PB[166]	ISRC1_PB[165]	ISRC1_PB[164]	ISRC1_PB[163]	ISRC1_PB[162]	ISRC1_PB[161]	ISRC1_PB[160]
0xA1	0x00	R	ISRC1_PB[175]	ISRC1_PB[174]	ISRC1_PB[173]	ISRC1_PB[172]	ISRC1_PB[171]	ISRC1_PB[170]	ISRC1_PB[169]	ISRC1_PB[168]
0xA2	0x00	R	ISRC1_PB[183]	ISRC1_PB[182]	ISRC1_PB[181]	ISRC1_PB[180]	ISRC1_PB[179]	ISRC1_PB[178]	ISRC1_PB[177]	ISRC1_PB[176]
0xA3	0x00	R	ISRC1_PB[191]	ISRC1_PB[190]	ISRC1_PB[189]	ISRC1_PB[188]	ISRC1_PB[187]	ISRC1_PB[186]	ISRC1_PB[185]	ISRC1_PB[184]
0xA4	0x00	R	ISRC1_PB[199]	ISRC1_PB[198]	ISRC1_PB[197]	ISRC1_PB[196]	ISRC1_PB[195]	ISRC1_PB[194]	ISRC1_PB[193]	ISRC1_PB[192]
0xA5	0x00	R	ISRC1_PB[207]	ISRC1_PB[206]	ISRC1_PB[205]	ISRC1_PB[204]	ISRC1_PB[203]	ISRC1_PB[202]	ISRC1_PB[201]	ISRC1_PB[200]
0xA6	0x00	R	ISRC1_PB[215]	ISRC1_PB[214]	ISRC1_PB[213]	ISRC1_PB[212]	ISRC1_PB[211]	ISRC1_PB[210]	ISRC1_PB[209]	ISRC1_PB[208]
0xA7	0x00	R	ISRC1_PB[223]	ISRC1_PB[222]	ISRC1_PB[221]	ISRC1_PB[220]	ISRC1_PB[219]	ISRC1_PB[218]	ISRC1_PB[217]	ISRC1_PB[216]
0xA8	0x00	R	ISRC2_PB[7]	ISRC2_PB[6]	ISRC2_PB[5]	ISRC2_PB[4]	ISRC2_PB[3]	ISRC2_PB[2]	ISRC2_PB[1]	ISRC2_PB[0]
0xA9	0x00	R	ISRC2_PB[15]	ISRC2_PB[14]	ISRC2_PB[13]	ISRC2_PB[12]	ISRC2_PB[11]	ISRC2_PB[10]	ISRC2_PB[9]	ISRC2_PB[8]
0xAA	0x00	R	ISRC2_PB[23]	ISRC2_PB[22]	ISRC2_PB[21]	ISRC2_PB[20]	ISRC2_PB[19]	ISRC2_PB[18]	ISRC2_PB[17]	ISRC2_PB[16]
0xAB	0x00	R	ISRC2_PB[31]	ISRC2_PB[30]	ISRC2_PB[29]	ISRC2_PB[28]	ISRC2_PB[27]	ISRC2_PB[26]	ISRC2_PB[25]	ISRC2_PB[24]
0xAC	0x00	R	ISRC2_PB[39]	ISRC2_PB[38]	ISRC2_PB[37]	ISRC2_PB[36]	ISRC2_PB[35]	ISRC2_PB[34]	ISRC2_PB[33]	ISRC2_PB[32]
0xAD	0x00	R	ISRC2_PB[47]	ISRC2_PB[46]	ISRC2_PB[45]	ISRC2_PB[44]	ISRC2_PB[43]	ISRC2_PB[42]	ISRC2_PB[41]	ISRC2_PB[40]
0xAE	0x00	R	ISRC2_PB[55]	ISRC2_PB[54]	ISRC2_PB[53]	ISRC2_PB[52]	ISRC2_PB[51]	ISRC2_PB[50]	ISRC2_PB[49]	ISRC2_PB[48]
0xAF	0x00	R	ISRC2_PB[63]	ISRC2_PB[62]	ISRC2_PB[61]	ISRC2_PB[60]	ISRC2_PB[59]	ISRC2_PB[58]	ISRC2_PB[57]	ISRC2_PB[56]
0xB0	0x00	R	ISRC2_PB[71]	ISRC2_PB[70]	ISRC2_PB[69]	ISRC2_PB[68]	ISRC2_PB[67]	ISRC2_PB[66]	ISRC2_PB[65]	ISRC2_PB[64]
0xB1	0x00	R	ISRC2_PB[79]	ISRC2_PB[78]	ISRC2_PB[77]	ISRC2_PB[76]	ISRC2_PB[75]	ISRC2_PB[74]	ISRC2_PB[73]	ISRC2_PB[72]
0xB2	0x00	R	ISRC2_PB[87]	ISRC2_PB[86]	ISRC2_PB[85]	ISRC2_PB[84]	ISRC2_PB[83]	ISRC2_PB[82]	ISRC2_PB[81]	ISRC2_PB[80]
0xB3	0x00	R	ISRC2_PB[95]	ISRC2_PB[94]	ISRC2_PB[93]	ISRC2_PB[92]	ISRC2_PB[91]	ISRC2_PB[90]	ISRC2_PB[89]	ISRC2_PB[88]
0xB4	0x00	R	ISRC2_PB[103]	ISRC2_PB[102]	ISRC2_PB[101]	ISRC2_PB[100]	ISRC2_PB[99]	ISRC2_PB[98]	ISRC2_PB[97]	ISRC2_PB[96]
0xB5	0x00	R	ISRC2_PB[111]	ISRC2_PB[110]	ISRC2_PB[109]	ISRC2_PB[108]	ISRC2_PB[107]	ISRC2_PB[106]	ISRC2_PB[105]	ISRC2_PB[104]
0xB6	0x00	R	ISRC2_PB[119]	ISRC2_PB[118]	ISRC2_PB[117]	ISRC2_PB[116]	ISRC2_PB[115]	ISRC2_PB[114]	ISRC2_PB[113]	ISRC2_PB[112]

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xB7	0x00	R	ISRC2_PB[127]	ISRC2_PB[126]	ISRC2_PB[125]	ISRC2_PB[124]	ISRC2_PB[123]	ISRC2_PB[122]	ISRC2_PB[121]	ISRC2_PB[120]
0xB8	0x00	R	ISRC2_PB[135]	ISRC2_PB[134]	ISRC2_PB[133]	ISRC2_PB[132]	ISRC2_PB[131]	ISRC2_PB[130]	ISRC2_PB[129]	ISRC2_PB[128]
0xB9	0x00	R	ISRC2_PB[143]	ISRC2_PB[142]	ISRC2_PB[141]	ISRC2_PB[140]	ISRC2_PB[139]	ISRC2_PB[138]	ISRC2_PB[137]	ISRC2_PB[136]
0xBA	0x00	R	ISRC2_PB[151]	ISRC2_PB[150]	ISRC2_PB[149]	ISRC2_PB[148]	ISRC2_PB[147]	ISRC2_PB[146]	ISRC2_PB[145]	ISRC2_PB[144]
0xBB	0x00	R	ISRC2_PB[159]	ISRC2_PB[158]	ISRC2_PB[157]	ISRC2_PB[156]	ISRC2_PB[155]	ISRC2_PB[154]	ISRC2_PB[153]	ISRC2_PB[152]
0xBC	0x00	R	ISRC2_PB[167]	ISRC2_PB[166]	ISRC2_PB[165]	ISRC2_PB[164]	ISRC2_PB[163]	ISRC2_PB[162]	ISRC2_PB[161]	ISRC2_PB[160]
0xBD	0x00	R	ISRC2_PB[175]	ISRC2_PB[174]	ISRC2_PB[173]	ISRC2_PB[172]	ISRC2_PB[171]	ISRC2_PB[170]	ISRC2_PB[169]	ISRC2_PB[168]
0xBE	0x00	R	ISRC2_PB[183]	ISRC2_PB[182]	ISRC2_PB[181]	ISRC2_PB[180]	ISRC2_PB[179]	ISRC2_PB[178]	ISRC2_PB[177]	ISRC2_PB[176]
0xBF	0x00	R	ISRC2_PB[191]	ISRC2_PB[190]	ISRC2_PB[189]	ISRC2_PB[188]	ISRC2_PB[187]	ISRC2_PB[186]	ISRC2_PB[185]	ISRC2_PB[184]
0xC0	0x00	R	ISRC2_PB[199]	ISRC2_PB[198]	ISRC2_PB[197]	ISRC2_PB[196]	ISRC2_PB[195]	ISRC2_PB[194]	ISRC2_PB[193]	ISRC2_PB[192]
0xC1	0x00	R	ISRC2_PB[207]	ISRC2_PB[206]	ISRC2_PB[205]	ISRC2_PB[204]	ISRC2_PB[203]	ISRC2_PB[202]	ISRC2_PB[201]	ISRC2_PB[200]
0xC2	0x00	R	ISRC2_PB[215]	ISRC2_PB[214]	ISRC2_PB[213]	ISRC2_PB[212]	ISRC2_PB[211]	ISRC2_PB[210]	ISRC2_PB[209]	ISRC2_PB[208]
0xC3	0x00	R	ISRC2_PB[223]	ISRC2_PB[222]	ISRC2_PB[221]	ISRC2_PB[220]	ISRC2_PB[219]	ISRC2_PB[218]	ISRC2_PB[217]	ISRC2_PB[216]
0xC4	0x00	R	GBD[7]	GBD[6]	GBD[5]	GBD[4]	GBD[3]	GBD[2]	GBD[1]	GBD[0]
0xC5	0x00	R	GBD[15]	GBD[14]	GBD[13]	GBD[12]	GBD[11]	GBD[10]	GBD[9]	GBD[8]
0xC6	0x00	R	GBD[23]	GBD[22]	GBD[21]	GBD[20]	GBD[19]	GBD[18]	GBD[17]	GBD[16]
0xC7	0x00	R	GBD[31]	GBD[30]	GBD[29]	GBD[28]	GBD[27]	GBD[26]	GBD[25]	GBD[24]
0xC8	0x00	R	GBD[39]	GBD[38]	GBD[37]	GBD[36]	GBD[35]	GBD[34]	GBD[33]	GBD[32]
0xC9	0x00	R	GBD[47]	GBD[46]	GBD[45]	GBD[44]	GBD[43]	GBD[42]	GBD[41]	GBD[40]
0xCA	0x00	R	GBD[55]	GBD[54]	GBD[53]	GBD[52]	GBD[51]	GBD[50]	GBD[49]	GBD[48]
0xCB	0x00	R	GBD[63]	GBD[62]	GBD[61]	GBD[60]	GBD[59]	GBD[58]	GBD[57]	GBD[56]
0xCC	0x00	R	GBD[71]	GBD[70]	GBD[69]	GBD[68]	GBD[67]	GBD[66]	GBD[65]	GBD[64]
0xCD	0x00	R	GBD[79]	GBD[78]	GBD[77]	GBD[76]	GBD[75]	GBD[74]	GBD[73]	GBD[72]
0xCE	0x00	R	GBD[87]	GBD[86]	GBD[85]	GBD[84]	GBD[83]	GBD[82]	GBD[81]	GBD[80]
0xCF	0x00	R	GBD[95]	GBD[94]	GBD[93]	GBD[92]	GBD[91]	GBD[90]	GBD[89]	GBD[88]
0xD0	0x00	R	GBD[103]	GBD[102]	GBD[101]	GBD[100]	GBD[99]	GBD[98]	GBD[97]	GBD[96]
0xD1	0x00	R	GBD[111]	GBD[110]	GBD[109]	GBD[108]	GBD[107]	GBD[106]	GBD[105]	GBD[104]
0xD2	0x00	R	GBD[119]	GBD[118]	GBD[117]	GBD[116]	GBD[115]	GBD[114]	GBD[113]	GBD[112]
0xD3	0x00	R	GBD[127]	GBD[126]	GBD[125]	GBD[124]	GBD[123]	GBD[122]	GBD[121]	GBD[120]
0xD4	0x00	R	GBD[135]	GBD[134]	GBD[133]	GBD[132]	GBD[131]	GBD[130]	GBD[129]	GBD[128]
0xD5	0x00	R	GBD[143]	GBD[142]	GBD[141]	GBD[140]	GBD[139]	GBD[138]	GBD[137]	GBD[136]
0xD6	0x00	R	GBD[151]	GBD[150]	GBD[149]	GBD[148]	GBD[147]	GBD[146]	GBD[145]	GBD[144]
0xD7	0x00	R	GBD[159]	GBD[158]	GBD[157]	GBD[156]	GBD[155]	GBD[154]	GBD[153]	GBD[152]
0xD8	0x00	R	GBD[167]	GBD[166]	GBD[165]	GBD[164]	GBD[163]	GBD[162]	GBD[161]	GBD[160]
0xD9	0x00	R	GBD[175]	GBD[174]	GBD[173]	GBD[172]	GBD[171]	GBD[170]	GBD[169]	GBD[168]
0xDA	0x00	R	GBD[183]	GBD[182]	GBD[181]	GBD[180]	GBD[179]	GBD[178]	GBD[177]	GBD[176]
0xDB	0x00	R	GBD[191]	GBD[190]	GBD[189]	GBD[188]	GBD[187]	GBD[186]	GBD[185]	GBD[184]
0xDC	0x00	R	GBD[199]	GBD[198]	GBD[197]	GBD[196]	GBD[195]	GBD[194]	GBD[193]	GBD[192]
0xDD	0x00	R	GBD[207]	GBD[206]	GBD[205]	GBD[204]	GBD[203]	GBD[202]	GBD[201]	GBD[200]
0xDE	0x00	R	GBD[215]	GBD[214]	GBD[213]	GBD[212]	GBD[211]	GBD[210]	GBD[209]	GBD[208]
0xDF	0x00	R	GBD[223]	GBD[222]	GBD[221]	GBD[220]	GBD[219]	GBD[218]	GBD[217]	GBD[216]
0xE0	0x82	R/W	AVI_PACKET_ ID[7]	AVI_PACKET_ ID[6]	AVI_PACKET_ ID[5]	AVI_PACKET_ ID[4]	AVI_PACKET_ ID[3]	AVI_PACKET_ ID[2]	AVI_PACKET_ ID[1]	AVI_PACKET_ ID[0]
0xE1	0x00	R	AVI_INF_ VERS[7]	AVI_INF_ VERS[6]	AVI_INF_ VERS[5]	AVI_INF_ VERS[4]	AVI_INF_ VERS[3]	AVI_INF_ VERS[2]	AVI_INF_ VERS[1]	AVI_INF_ VERS[0]
0xE2	0x00	R	AVI_INF_LEN[7]	AVI_INF_LEN[6]	AVI_INF_LEN[5]	AVI_INF_LEN[4]	AVI_INF_LEN[3]	AVI_INF_LEN[2]	AVI_INF_LEN[1]	AVI_INF_LEN[0]
0xE3	0x84	R/W	AUD_PACKET_ ID[7]	AUD_PACKET_ ID[6]	AUD_PACKET_ ID[5]	AUD_PACKET_ ID[4]	AUD_PACKET_ ID[3]	AUD_PACKET_ ID[2]	AUD_PACKET_ ID[1]	AUD_PACKET_ ID[0]
0xE4	0x00	R	AUD_INF_ VERS[7]	AUD_INF_ VERS[6]	AUD_INF_ VERS[5]	AUD_INF_ VERS[4]	AUD_INF_ VERS[3]	AUD_INF_ VERS[2]	AUD_INF_ VERS[1]	AUD_INF_ VERS[0]
0xE5	0x00	R	AUD_INF_ LEN[7]	AUD_INF_ LEN[6]	AUD_INF_ LEN[5]	AUD_INF_ LEN[4]	AUD_INF_ LEN[3]	AUD_INF_ LEN[2]	AUD_INF_ LEN[1]	AUD_INF_ LEN[0]
0xE6	0x83	R/W	SPD_PACKET_ ID[7]	SPD_PACKET_ ID[6]	SPD_PACKET_ ID[5]	SPD_PACKET_ ID[4]	SPD_PACKET_ ID[3]	SPD_PACKET_ ID[2]	SPD_PACKET_ ID[1]	SPD_PACKET_ ID[0]
0xE7	0x00	R	SPD_INF_ VERS[7]	SPD_INF_ VERS[6]	SPD_INF_ VERS[5]	SPD_INF_ VERS[4]	SPD_INF_ VERS[3]	SPD_INF_ VERS[2]	SPD_INF_ VERS[1]	SPD_INF_ VERS[0]
0xE8	0x00	R	SPD_INF_LEN[7]	SPD_INF_LEN[6]	SPD_INF_LEN[5]	SPD_INF_LEN[4]	SPD_INF_LEN[3]	SPD_INF_LEN[2]	SPD_INF_LEN[1]	SPD_INF_LEN[0]

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xE9	0x85	R/W	MS_PACKET_ ID[7]	MS_PACKET_ ID[6]	MS_PACKET_ ID[5]	MS_PACKET_ ID[4]	MS_PACKET_ ID[3]	MS_PACKET_ ID[2]	MS_PACKET_ ID[1]	MS_PACKET_ ID[0]
0xEA	0x00	R	MS_INF_ VERS[7]	MS_INF_ VERS[6]	MS_INF_ VERS[5]	MS_INF_ VERS[4]	MS_INF_ VERS[3]	MS_INF_ VERS[2]	MS_INF_ VERS[1]	MS_INF_ VERS[0]
0xEB	0x00	R	MS_INF_LEN[7]	MS_INF_LEN[6]	MS_INF_LEN[5]	MS_INF_LEN[4]	MS_INF_LEN[3]	MS_INF_LEN[2]	MS_INF_LEN[1]	MS_INF_LEN[0]
0xEC	0x81	R/W	VS_PACKET_ ID[7]	VS_PACKET_ ID[6]	VS_PACKET_ ID[5]	VS_PACKET_ ID[4]	VS_PACKET_ ID[3]	VS_PACKET_ ID[2]	VS_PACKET_ ID[1]	VS_PACKET_ ID[0]
0xED	0x00	R	VS_INF_VERS[7]	VS_INF_VERS[6]	VS_INF_VERS[5]	VS_INF_VERS[4]	VS_INF_VERS[3]	VS_INF_VERS[2]	VS_INF_VERS[1]	VS_INF_VERS[0]
0xEE	0x00	R	VS_INF_LEN[7]	VS_INF_LEN[6]	VS_INF_LEN[5]	VS_INF_LEN[4]	VS_INF_LEN[3]	VS_INF_LEN[2]	VS_INF_LEN[1]	VS_INF_LEN[0]
0xEF	0x04	R/W	ACP_PACKET_ ID[7]	ACP_PACKET_ ID[6]	ACP_PACKET_ ID[5]	ACP_PACKET_ ID[4]	ACP_PACKET_ ID[3]	ACP_PACKET_ ID[2]	ACP_PACKET_ ID[1]	ACP_PACKET_ ID[0]
0xF0	0x00	R	ACP_TYPE[7]	ACP_TYPE[6]	ACP_TYPE[5]	ACP_TYPE[4]	ACP_TYPE[3]	ACP_TYPE[2]	ACP_TYPE[1]	ACP_TYPE[0]
0xF1	0x00	R	ACP_ HEADER2[7]	ACP_ HEADER2[6]	ACP_ HEADER2[5]	ACP_ HEADER2[4]	ACP_ HEADER2[3]	ACP_ HEADER2[2]	ACP_ HEADER2[1]	ACP_ HEADER2[0]
0xF2	0x05	R/W	ISRC1_ PACKET_ID[7]	ISRC1_ PACKET_ID[6]	ISRC1_ PACKET_ID[5]	ISRC1_ PACKET_ID[4]	ISRC1_ PACKET_ID[3]	ISRC1_ PACKET_ID[2]	ISRC1_ PACKET_ID[1]	ISRC1_ PACKET_ID[0]
0xF3	0x00	R	ISRC1_ HEADER1[7]	ISRC1_ HEADER1[6]	ISRC1_ HEADER1[5]	ISRC1_ HEADER1[4]	ISRC1_ HEADER1[3]	ISRC1_ HEADER1[2]	ISRC1_ HEADER1[1]	ISRC1_ HEADER1[0]
0xF4	0x00	R	ISRC1_ HEADER2[7]	ISRC1_ HEADER2[6]	ISRC1_ HEADER2[5]	ISRC1_ HEADER2[4]	ISRC1_ HEADER2[3]	ISRC1_ HEADER2[2]	ISRC1_ HEADER2[1]	ISRC1_ HEADER2[0]
0xF5	0x06	R/W	ISRC2_ PACKET_ID[7]	ISRC2_ PACKET_ID[6]	ISRC2_ PACKET_ID[5]	ISRC2_ PACKET_ID[4]	ISRC2_ PACKET_ID[3]	ISRC2_ PACKET_ID[2]	ISRC2_ PACKET_ID[1]	ISRC2_ PACKET_ID[0]
0xF6	0x00	R	ISRC2_ HEADER1[7]	ISRC2_ HEADER1[6]	ISRC2_ HEADER1[5]	ISRC2_ HEADER1[4]	ISRC2_ HEADER1[3]	ISRC2_ HEADER1[2]	ISRC2_ HEADER1[1]	ISRC2_ HEADER1[0]
0xF7	0x00	R	ISRC2_ HEADER2[7]	ISRC2_ HEADER2[6]	ISRC2_ HEADER2[5]	ISRC2_ HEADER2[4]	ISRC2_ HEADER2[3]	ISRC2_ HEADER2[2]	ISRC2_ HEADER2[1]	ISRC2_ HEADER2[0]
0xF8	0x0A	R/W	GAMUT_ PACKET_ID[7]	GAMUT_ PACKET_ID[6]	GAMUT_ PACKET_ID[5]	GAMUT_ PACKET_ID[4]	GAMUT_ PACKET_ID[3]	GAMUT_ PACKET_ID[2]	GAMUT_ PACKET_ID[1]	GAMUT_ PACKET_ID[0]
0xF9	0x00	R	GAMUT_ HEADER1[7]	GAMUT_ HEADER1[6]	GAMUT_ HEADER1[5]	GAMUT_ HEADER1[4]	GAMUT_ HEADER1[3]	GAMUT_ HEADER1[2]	GAMUT_ HEADER1[1]	GAMUT_ HEADER1[0]
0xFA	0x00	R	GAMUT_ HEADER2[7]	GAMUT_ HEADER2[6]	GAMUT_ HEADER2[5]	GAMUT_ HEADER2[4]	GAMUT_ HEADER2[3]	GAMUT_ HEADER2[2]	GAMUT_ HEADER2[1]	GAMUT_ HEADER2[0]

CP REGISTER MAP

Add is the register map I²C address, Def is the default value of the register, and Acc is the read/write access for the register (R means read only, and R/W means read/write access).

Table 6. ADV7613 CP Register Map

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x0B	0x00	R	RB_CSC_SCALE[1]	RB_CSC_SCALE[0]		RB_A4[12]	RB_A4[11]	RB_A4[10]	RB_A4[9]	RB_A4[8]
0x0C	0x00	R	RB_A4[7]	RB_A4[6]	RB_A4[5]	RB_A4[4]	RB_A4[3]	RB_A4[2]	RB_A4[1]	RB_A4[0]
0x0D	0x00	R		RB_A3[12]	RB_A3[11]	RB_A3[10]	RB_A3[9]	RB_A3[8]	RB_A3[7]	RB_A3[6]
0x0E	0x00	R	RB_A3[5]	RB_A3[4]	RB_A3[3]	RB_A3[2]	RB_A3[1]	RB_A3[0]	RB_A2[12]	RB_A2[11]
0x0F	0x00	R	RB_A2[10]	RB_A2[9]	RB_A2[8]	RB_A2[7]	RB_A2[6]	RB_A2[5]	RB_A2[4]	RB_A2[3]
0x10	0x00	R	RB_A2[2]	RB_A2[1]	RB_A2[0]	RB_A1[12]	RB_A1[11]	RB_A1[10]	RB_A1[9]	RB_A1[8]
0x11	0x00	R	RB_A1[7]	RB_A1[6]	RB_A1[5]	RB_A1[4]	RB_A1[3]	RB_A1[2]	RB_A1[1]	RB_A1[0]
0x12	0x00	R				RB_B4[12]	RB_B4[11]	RB_B4[10]	RB_B4[9]	RB_B4[8]
0x13	0x00	R	RB_B4[7]	RB_B4[6]	RB_B4[5]	RB_B4[4]	RB_B4[3]	RB_B4[2]	RB_B4[1]	RB_B4[0]
0x14	0x00	R		RB_B3[12]	RB_B3[11]	RB_B3[10]	RB_B3[9]	RB_B3[8]	RB_B3[7]	RB_B3[6]
0x15	0x00	R	RB_B3[5]	RB_B3[4]	RB_B3[3]	RB_B3[2]	RB_B3[1]	RB_B3[0]	RB_B2[12]	RB_B2[11]
0x16	0x00	R	RB_B2[10]	RB_B2[9]	RB_B2[8]	RB_B2[7]	RB_B2[6]	RB_B2[5]	RB_B2[4]	RB_B2[3]
0x17	0x00	R	RB_B2[2]	RB_B2[1]	RB_B2[0]	RB_B1[12]	RB_B1[11]	RB_B1[10]	RB_B1[9]	RB_B1[8]
0x18	0x00	R	RB_B1[7]	RB_B1[6]	RB_B1[5]	RB_B1[4]	RB_B1[3]	RB_B1[2]	RB_B1[1]	RB_B1[0]
0x19	0x00	R				RB_C4[12]	RB_C4[11]	RB_C4[10]	RB_C4[9]	RB_C4[8]
0x1A	0x00	R	RB_C4[7]	RB_C4[6]	RB_C4[5]	RB_C4[4]	RB_C4[3]	RB_C4[2]	RB_C4[1]	RB_C4[0]
0x1B	0x00	R		RB_C3[12]	RB_C3[11]	RB_C3[10]	RB_C3[9]	RB_C3[8]	RB_C3[7]	RB_C3[6]
0x1C	0x00	R	RB_C3[5]	RB_C3[4]	RB_C3[3]	RB_C3[2]	RB_C3[1]	RB_C3[0]	RB_C2[12]	RB_C2[11]
0x1D	0x00	R	RB_C2[10]	RB_C2[9]	RB_C2[8]	RB_C2[7]	RB_C2[6]	RB_C2[5]	RB_C2[4]	RB_C2[3]
0x1E	0x00	R	RB_C2[2]	RB_C2[1]	RB_C2[0]	RB_C1[12]	RB_C1[11]	RB_C1[10]	RB_C1[9]	RB_C1[8]
0x1F	0x00	R	RB_C1[7]	RB_C1[6]	RB_C1[5]	RB_C1[4]	RB_C1[3]	RB_C1[2]	RB_C1[1]	RB_C1[0]
0x22	0x00	R/W				CP_START_HS[12]	CP_START_HS[11]	CP_START_HS[10]	CP_START_HS[9]	CP_START_HS[8]
0x23	0x00	R/W	CP_START_HS[7]	CP_START_HS[6]	CP_START_HS[5]	CP_START_HS[4]	CP_START_HS[3]	CP_START_HS[2]	CP_START_HS[1]	CP_START_HS[0]
0x24	0x00	R/W				CP_END_HS[12]	CP_END_HS[11]	CP_END_HS[10]	CP_END_HS[9]	CP_END_HS[8]
0x25	0x00	R/W	CP_END_HS[7]	CP_END_HS[6]	CP_END_HS[5]	CP_END_HS[4]	CP_END_HS[3]	CP_END_HS[2]	CP_END_HS[1]	CP_END_HS[0]
0x26	0x00	R/W				CP_START_SAV[12]	CP_START_SAV[11]	CP_START_SAV[10]	CP_START_SAV[9]	CP_START_SAV[8]
0x27	0x00	R/W	CP_START_SAV[7]	CP_START_SAV[6]	CP_START_SAV[5]	CP_START_SAV[4]	CP_START_SAV[3]	CP_START_SAV[2]	CP_START_SAV[1]	CP_START_SAV[0]
0x28	0x00	R/W				CP_START_EAV[12]	CP_START_EAV[11]	CP_START_EAV[10]	CP_START_EAV[9]	CP_START_EAV[8]
0x29	0x00	R/W	CP_START_EAV[7]	CP_START_EAV[6]	CP_START_EAV[5]	CP_START_EAV[4]	CP_START_EAV[3]	CP_START_EAV[2]	CP_START_EAV[1]	CP_START_EAV[0]
0x2A	0x00	R/W	CP_START_VBI_R[11]	CP_START_VBI_R[10]	CP_START_VBI_R[9]	CP_START_VBI_R[8]	CP_START_VBI_R[7]	CP_START_VBI_R[6]	CP_START_VBI_R[5]	CP_START_VBI_R[4]
0x2B	0x00	R/W	CP_START_VBI_R[3]	CP_START_VBI_R[2]	CP_START_VBI_R[1]	CP_START_VBI_R[0]	CP_END_VBI_R[11]	CP_END_VBI_R[10]	CP_END_VBI_R[9]	CP_END_VBI_R[8]
0x2C	0x00	R/W	CP_END_VBI_R[7]	CP_END_VBI_R[6]	CP_END_VBI_R[5]	CP_END_VBI_R[4]	CP_END_VBI_R[3]	CP_END_VBI_R[2]	CP_END_VBI_R[1]	CP_END_VBI_R[0]
0x2D	0x00	R/W	CP_START_VBI_EVEN_R[11]	CP_START_VBI_EVEN_R[10]	CP_START_VBI_EVEN_R[9]	CP_START_VBI_EVEN_R[8]	CP_START_VBI_EVEN_R[7]	CP_START_VBI_EVEN_R[6]	CP_START_VBI_EVEN_R[5]	CP_START_VBI_EVEN_R[4]
0x2E	0x00	R/W	CP_START_VBI_EVEN_R[3]	CP_START_VBI_EVEN_R[2]	CP_START_VBI_EVEN_R[1]	CP_START_VBI_EVEN_R[0]	CP_END_VBI_EVEN_R[11]	CP_END_VBI_EVEN_R[10]	CP_END_VBI_EVEN_R[9]	CP_END_VBI_EVEN_R[8]
0x2F	0x00	R/W	CP_END_VBI_EVEN_R[7]	CP_END_VBI_EVEN_R[6]	CP_END_VBI_EVEN_R[5]	CP_END_VBI_EVEN_R[4]	CP_END_VBI_EVEN_R[3]	CP_END_VBI_EVEN_R[2]	CP_END_VBI_EVEN_R[1]	CP_END_VBI_EVEN_R[0]
0x30	0x00	R/W	DE_V_START_R[3]	DE_V_START_R[2]	DE_V_START_R[1]	DE_V_START_R[0]	DE_V_END_R[3]	DE_V_END_R[2]	DE_V_END_R[1]	DE_V_END_R[0]
0x31	0x00	R/W	DE_V_START_EVEN_R[3]	DE_V_START_EVEN_R[2]	DE_V_START_EVEN_R[1]	DE_V_START_EVEN_R[0]	DE_V_END_EVEN_R[3]	DE_V_END_EVEN_R[2]	DE_V_END_EVEN_R[1]	DE_V_END_EVEN_R[0]

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x36	0x00	R/W								TEN_TO_ EIGHT_CONV
0x3A	0x80	R/W	CP_ CONTRAST[7]	CP_ CONTRAST[6]	CP_ CONTRAST[5]	CP_ CONTRAST[4]	CP_ CONTRAST[3]	CP_ CONTRAST[2]	CP_ CONTRAST[1]	CP_ CONTRAST[0]
0x3B	0x80	R/W	CP_ SATURATION[7]	CP_ SATURATION[6]	CP_ SATURATION[5]	CP_ SATURATION[4]	CP_ SATURATION[3]	CP_ SATURATION[2]	CP_ SATURATION[1]	CP_ SATURATION[0]
0x3C	0x00	R/W	CP_ BRIGHTNESS[7]	CP_ BRIGHTNESS[6]	CP_ BRIGHTNESS[5]	CP_ BRIGHTNESS[4]	CP_ BRIGHTNESS[3]	CP_ BRIGHTNESS[2]	CP_ BRIGHTNESS[1]	CP_ BRIGHTNESS[0]
0x3D	0x00	R/W	CP_HUE[7]	CP_HUE[6]	CP_HUE[5]	CP_HUE[4]	CP_HUE[3]	CP_HUE[2]	CP_HUE[1]	CP_HUE[0]
0x3E	0x00	R/W	VID_ADJ_EN		CP_UV_ ALIGN_SEL[1]	CP_UV_ALIGN_ SEL[0]	CP_UV_DVAL_ INV	CP_MODE_ GAIN_ADJ_EN	ALT_SAT_UV_ MAN	ALT_SAT_UV
0x40	0x5C	R/W	CP_MODE_ GAIN_ADJ[7]	CP_MODE_ GAIN_ADJ[6]	CP_MODE_ GAIN_ADJ[5]	CP_MODE_ GAIN_ADJ[4]	CP_MODE_ GAIN_ADJ[3]	CP_MODE_ GAIN_ADJ[2]	CP_MODE_ GAIN_ADJ[1]	CP_MODE_ GAIN_ADJ[0]
0x52	0x40	R/W	CSC_SCALE[1]	CSC_SCALE[0]		A4[12]	A4[11]	A4[10]	A4[9]	A4[8]
0x53	0x00	R/W	A4[7]	A4[6]	A4[5]	A4[4]	A4[3]	A4[2]	A4[1]	A4[0]
0x54	0x00	R/W		A3[12]	A3[11]	A3[10]	A3[9]	A3[8]	A3[7]	A3[6]
0x55	0x00	R/W	A3[5]	A3[4]	A3[3]	A3[2]	A3[1]	A3[0]	A2[12]	A2[11]
0x56	0x00	R/W	A2[10]	A2[9]	A2[8]	A2[7]	A2[6]	A2[5]	A2[4]	A2[3]
0x57	0x08	R/W	A2[2]	A2[1]	A2[0]	A1[12]	A1[11]	A1[10]	A1[9]	A1[8]
0x58	0x00	R/W	A1[7]	A1[6]	A1[5]	A1[4]	A1[3]	A1[2]	A1[1]	A1[0]
0x59	0x00	R/W				B4[12]	B4[11]	B4[10]	B4[9]	B4[8]
0x5A	0x00	R/W	B4[7]	B4[6]	B4[5]	B4[4]	B4[3]	B4[2]	B4[1]	B4[0]
0x5B	0x00	R/W		B3[12]	B3[11]	B3[10]	B3[9]	B3[8]	B3[7]	B3[6]
0x5C	0x01	R/W	B3[5]	B3[4]	B3[3]	B3[2]	B3[1]	B3[0]	B2[12]	B2[11]
0x5D	0x00	R/W	B2[10]	B2[9]	B2[8]	B2[7]	B2[6]	B2[5]	B2[4]	B2[3]
0x5E	0x00	R/W	B2[2]	B2[1]	B2[0]	B1[12]	B1[11]	B1[10]	B1[9]	B1[8]
0x5F	0x00	R/W	B1[7]	B1[6]	B1[5]	B1[4]	B1[3]	B1[2]	B1[1]	B1[0]
0x60	0x00	R/W				C4[12]	C4[11]	C4[10]	C4[9]	C4[8]
0x61	0x00	R/W	C4[7]	C4[6]	C4[5]	C4[4]	C4[3]	C4[2]	C4[1]	C4[0]
0x62	0x20	R/W		C3[12]	C3[11]	C3[10]	C3[9]	C3[8]	C3[7]	C3[6]
0x63	0x00	R/W	C3[5]	C3[4]	C3[3]	C3[2]	C3[1]	C3[0]	C2[12]	C2[11]
0x64	0x00	R/W	C2[10]	C2[9]	C2[8]	C2[7]	C2[6]	C2[5]	C2[4]	C2[3]
0x65	0x00	R/W	C2[2]	C2[1]	C2[0]	C1[12]	C1[11]	C1[10]	C1[9]	C1[8]
0x66	0x00	R/W	C1[7]	C1[6]	C1[5]	C1[4]	C1[3]	C1[2]	C1[1]	C1[0]
0x67	0x00	R/W			EMB_SYNC_ ON_ALL					
0x68	0xF0	R/W	CSC_COEFF_ SEL[3]	CSC_COEFF_ SEL[2]	CSC_COEFF_ SEL[1]	CSC_COEFF_ SEL[0]	CP_CHROMA_ LOW_EN			
0x69	0x04	R/W				MAN_CP_CSC_ EN		EIA_861_ COMPLIANCE		
0x6C	0x10	R/W	CLMP_A_MAN	CLMP_BC_ MAN	CLMP_FREEZE		CLMP_A[11]	CLMP_A[10]	CLMP_A[9]	CLMP_A[8]
0x6D	0x00	R/W	CLMP_A[7]	CLMP_A[6]	CLMP_A[5]	CLMP_A[4]	CLMP_A[3]	CLMP_A[2]	CLMP_A[1]	CLMP_A[0]
0x6E	0x00	R/W	CLMP_B[11]	CLMP_B[10]	CLMP_B[9]	CLMP_B[8]	CLMP_B[7]	CLMP_B[6]	CLMP_B[5]	CLMP_B[4]
0x6F	0x00	R/W	CLMP_B[3]	CLMP_B[2]	CLMP_B[1]	CLMP_B[0]	CLMP_C[11]	CLMP_C[10]	CLMP_C[9]	CLMP_C[8]
0x70	0x00	R/W	CLMP_C[7]	CLMP_C[6]	CLMP_C[5]	CLMP_C[4]	CLMP_C[3]	CLMP_C[2]	CLMP_C[1]	CLMP_C[0]
0x73	0x10	R/W	GAIN_MAN	AGC_MODE_ MAN	A_GAIN[9]	A_GAIN[8]	A_GAIN[7]	A_GAIN[6]	A_GAIN[5]	A_GAIN[4]
0x74	0x04	R/W	A_GAIN[3]	A_GAIN[2]	A_GAIN[1]	A_GAIN[0]	B_GAIN[9]	B_GAIN[8]	B_GAIN[7]	B_GAIN[6]
0x75	0x01	R/W	B_GAIN[5]	B_GAIN[4]	B_GAIN[3]	B_GAIN[2]	B_GAIN[1]	B_GAIN[0]	C_GAIN[9]	C_GAIN[8]
0x76	0x00	R/W	C_GAIN[7]	C_GAIN[6]	C_GAIN[5]	C_GAIN[4]	C_GAIN[3]	C_GAIN[2]	C_GAIN[1]	C_GAIN[0]
0x77	0xFF	R/W	CP_PREC[1]	CP_PREC[0]	A_OFFSET[9]	A_OFFSET[8]	A_OFFSET[7]	A_OFFSET[6]	A_OFFSET[5]	A_OFFSET[4]
0x78	0xFF	R/W	A_OFFSET[3]	A_OFFSET[2]	A_OFFSET[1]	A_OFFSET[0]	B_OFFSET[9]	B_OFFSET[8]	B_OFFSET[7]	B_OFFSET[6]
0x79	0xFF	R/W	B_OFFSET[5]	B_OFFSET[4]	B_OFFSET[3]	B_OFFSET[2]	B_OFFSET[1]	B_OFFSET[0]	C_OFFSET[9]	C_OFFSET[8]
0x7A	0xFF	R/W	C_OFFSET[7]	C_OFFSET[6]	C_OFFSET[5]	C_OFFSET[4]	C_OFFSET[3]	C_OFFSET[2]	C_OFFSET[1]	C_OFFSET[0]
0x7B	0x05	R/W	AV_INV_F	AV_INV_V				AV_POS_SEL		DE_WITH_ AVCODE
0x7C	0xC0	R/W	CP_INV_HS	CP_INV_VS		CP_INV_DE	START_HS[9]	START_HS[8]	END_HS[9]	END_HS[8]
0x7D	0x00	R/W	END_HS[7]	END_HS[6]	END_HS[5]	END_HS[4]	END_HS[3]	END_HS[2]	END_HS[1]	END_HS[0]

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x7E	0x00	R/W	START_HS[7]	START_HS[6]	START_HS[5]	START_HS[4]	START_HS[3]	START_HS[2]	START_HS[1]	START_HS[0]
0x7F	0x00	R/W	START_VS[3]	START_VS[2]	START_VS[1]	START_VS[0]	END_VS[3]	END_VS[2]	END_VS[1]	END_VS[0]
0x80	0x00	R/W	START_FE[3]	START_FE[2]	START_FE[1]	START_FE[0]	START_FO[3]	START_FO[2]	START_FO[1]	START_FO[0]
0x84	0x0C	R/W	CP_GAIN_ FILT[3]	CP_GAIN_ FILT[2]	CP_GAIN_ FILT[1]	CP_GAIN_ FILT[0]				
0x86	0x0B	R/W						CH1_TRIG_ STDI	CH1_STDI_ CONT	
0x88	0x00	R/W	DE_V_START_ EVEN[3]	DE_V_START_ EVEN[2]	DE_V_START_ EVEN[1]	DE_V_START_ EVEN[0]	DE_V_END_ EVEN[3]	DE_V_END_ EVEN[2]	DE_V_END_ EVEN[1]	DE_V_END_ EVEN[0]
0x89	0x00	R/W	START_VS_ EVEN[3]	START_VS_ EVEN[2]	START_VS_ EVEN[1]	START_VS_ EVEN[0]	END_VS_ EVEN[3]	END_VS_ EVEN[2]	END_VS_ EVEN[1]	END_VS_ EVEN[0]
0x8B	0x40	R/W					DE_H_START[9]	DE_H_START[8]	DE_H_END[9]	DE_H_END[8]
0x8C	0x00	R/W	DE_H_END[7]	DE_H_END[6]	DE_H_END[5]	DE_H_END[4]	DE_H_END[3]	DE_H_END[2]	DE_H_END[1]	DE_H_END[0]
0x8D	0x00	R/W	DE_H_START[7]	DE_H_START[6]	DE_H_START[5]	DE_H_START[4]	DE_H_START[3]	DE_H_START[2]	DE_H_START[1]	DE_H_START[0]
0x8E	0x00	R/W	DE_V_START[3]	DE_V_START[2]	DE_V_START[1]	DE_V_START[0]	DE_V_END[3]	DE_V_END[2]	DE_V_END[1]	DE_V_END[0]
0x8F	0x40	R/W						CH1_FR_LL[10]	CH1_FR_LL[9]	CH1_FR_LL[8]
0x90	0x00	R/W	CH1_FR_LL[7]	CH1_FR_LL[6]	CH1_FR_LL[5]	CH1_FR_LL[4]	CH1_FR_LL[3]	CH1_FR_LL[2]	CH1_FR_LL[1]	CH1_FR_LL[0]
0x91	0x40	R/W		Interlaced						
0x9A	0x00	R/W				CP_START_ VS[5]	CP_START_ VS[4]	CP_START_ VS[3]	CP_START_ VS[2]	CP_START_ VS[1]
0x9B	0x00	R/W	CP_START_ VS[0]	CP_END_VS[5]	CP_END_VS[4]	CP_END_VS[3]	CP_END_VS[2]	CP_END_VS[1]	CP_END_VS[0]	
0x9C	0x00	R/W	CP_START_VS_ EVEN[10]	CP_START_VS_ EVEN[9]	CP_START_VS_ EVEN[8]	CP_START_VS_E VEN[7]	CP_START_VS_ EVEN[6]	CP_START_VS_ EVEN[5]	CP_START_VS_ EVEN[4]	CP_START_VS_ EVEN[3]
0x9D	0x00	R/W	CP_START_VS_ EVEN[2]	CP_START_VS_ EVEN[1]	CP_START_VS_ EVEN[0]	CP_END_VS_ EVEN[10]	CP_END_VS_ EVEN[9]	CP_END_VS_ EVEN[8]	CP_END_VS_ EVEN[7]	CP_END_VS_ EVEN[6]
0x9E	0x00	R/W	CP_END_VS_ EVEN[5]	CP_END_VS_ EVEN[4]	CP_END_VS_ EVEN[3]	CP_END_VS_ EVEN[2]	CP_END_VS_ EVEN[1]	CP_END_VS_ EVEN[0]		
0x9F	0x00	R/W	CP_START_F_ ODD[10]	CP_START_F_ ODD[9]	CP_START_F_ ODD[8]	CP_START_F_ ODD[7]	CP_START_F_ ODD[6]	CP_START_F_ ODD[5]	CP_START_F_ ODD[4]	CP_START_F_ ODD[3]
0xA0	0x00	R/W	CP_START_F_ ODD[2]	CP_START_F_ ODD[1]	CP_START_F_ ODD[0]	CP_START_F_ EVEN[10]	CP_START_F_ EVEN[9]	CP_START_F_ EVEN[8]	CP_START_F_ EVEN[7]	CP_START_F_ EVEN[6]
0xA1	0x00	R/W	CP_START_F_ EVEN[5]	CP_START_F_ EVEN[4]	CP_START_F_ EVEN[3]	CP_START_F_ EVEN[2]	CP_START_F_ EVEN[1]	CP_START_F_ EVEN[0]		
0xA3	0x00	R					CH1_LCF[11]	CH1_LCF[10]	CH1_LCF[9]	CH1_LCF[8]
0xA4	0x00	R	CH1_LCF[7]	CH1_LCF[6]	CH1_LCF[5]	CH1_LCF[4]	CH1_LCF[3]	CH1_LCF[2]	CH1_LCF[1]	CH1_LCF[0]
0xA5	0x00	R/W	CP_START_ VBI[11]	CP_START_ VBI[10]	CP_START_ VBI[9]	CP_START_ VBI[8]	CP_START_ VBI[7]	CP_START_ VBI[6]	CP_START_ VBI[5]	CP_START_ VBI[4]
0xA6	0x00	R/W	CP_START_ VBI[3]	CP_START_ VBI[2]	CP_START_ VBI[1]	CP_START_ VBI[0]	CP_END_ VBI[11]	CP_END_ VBI[10]	CP_END_VBI[9]	CP_END_VBI[8]
0xA7	0x00	R/W	CP_END_VBI[7]	CP_END_VBI[6]	CP_END_VBI[5]	CP_END_VBI[4]	CP_END_VBI[3]	CP_END_VBI[2]	CP_END_VBI[1]	CP_END_VBI[0]
0xA8	0x00	R/W	CP_START_ VBI_EVEN[11]	CP_START_ VBI_EVEN[10]	CP_START_ VBI_EVEN[9]	CP_START_ VBI_EVEN[8]	CP_START_ VBI_EVEN[7]	CP_START_ VBI_EVEN[6]	CP_START_ VBI_EVEN[5]	CP_START_ VBI_EVEN[4]
0xA9	0x00	R/W	CP_START_ VBI_EVEN[3]	CP_START_ VBI_EVEN[2]	CP_START_ VBI_EVEN[1]	CP_START_ VBI_EVEN[0]	CP_END_VBI_ EVEN[11]	CP_END_VBI_ EVEN[10]	CP_END_VBI_ EVEN[9]	CP_END_VBI_ EVEN[8]
0xAA	0x00	R/W	CP_END_VBI_ EVEN[7]	CP_END_VBI_ EVEN[6]	CP_END_VBI_ EVEN[5]	CP_END_VBI_ EVEN[4]	CP_END_VBI_ EVEN[3]	CP_END_VBI_E VEN[2]	CP_END_VBI_ EVEN[1]	CP_END_VBI_ EVEN[0]
0xAB	0x00	R/W	CP_LCOUNT_ MAX[11]	CP_LCOUNT_ MAX[10]	CP_LCOUNT_ MAX[9]	CP_LCOUNT_ MAX[8]	CP_LCOUNT_ MAX[7]	CP_LCOUNT_ MAX[6]	CP_LCOUNT_ MAX[5]	CP_LCOUNT_ MAX[4]
0xAC	0x00	R/W	CP_LCOUNT_ MAX[3]	CP_LCOUNT_ MAX[2]	CP_LCOUNT_ MAX[1]	CP_LCOUNT_ MAX[0]				
0xB1	0x00	R	CH1_STDI_ DVALID	CH1_STDI_ INTLCD	CH1_BL[13]	CH1_BL[12]	CH1_BL[11]	CH1_BL[10]	CH1_BL[9]	CH1_BL[8]
0xB2	0x00	R	CH1_BL[7]	CH1_BL[6]	CH1_BL[5]	CH1_BL[4]	CH1_BL[3]	CH1_BL[2]	CH1_BL[1]	CH1_BL[0]
0xB3	0x00	R	CH1_LCVS[4]	CH1_LCVS[3]	CH1_LCVS[2]	CH1_LCVS[1]	CH1_LCVS[0]			
0xB8	0x00	R				CH1_FCL[12]	CH1_FCL[11]	CH1_FCL[10]	CH1_FCL[9]	CH1_FCL[8]
0xB9	0x00	R	CH1_FCL[7]	CH1_FCL[6]	CH1_FCL[5]	CH1_FCL[4]	CH1_FCL[3]	CH1_FCL[2]	CH1_FCL[1]	CH1_FCL[0]
0xBA	0x01	R/W							HDMI_FRUN_ MODE	HDMI_FRUN_ EN
0xBD	0x18	R/W				DPP_BYPASS_ EN				

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xBE	0x00	R/W	DLY_A	DLY_B	DLY_C				HCOUNT_ ALIGN_ADJ[4]	HCOUNT_ ALIGN_ADJ[3]
0xBF	0x12	R/W	HCOUNT_ ALIGN_ADJ[2]	HCOUNT_ ALIGN_ADJ[1]	HCOUNT_ ALIGN_ADJ[0]			CP_DEF_COL_ MAN_VAL	CP_DEF_COL_A UTO	CP_FORCE_ FREERUN
0xC0	0x00	R/W	DEF_COL_ CHA[7]	DEF_COL_ CHA[6]	DEF_COL_ CHA[5]	DEF_COL_ CHA[4]	DEF_COL_ CHA[3]	DEF_COL_ CHA[2]	DEF_COL_ CHA[1]	DEF_COL_ CHA[0]
0xC1	0x00	R/W	DEF_COL_ CHB[7]	DEF_COL_ CHB[6]	DEF_COL_ CHB[5]	DEF_COL_ CHB[4]	DEF_COL_ CHB[3]	DEF_COL_ CHB[2]	DEF_COL_ CHB[1]	DEF_COL_ CHB[0]
0xC2	0x00	R/W	DEF_COL_ CHC[7]	DEF_COL_ CHC[6]	DEF_COL_ CHC[5]	DEF_COL_ CHC[4]	DEF_COL_ CHC[3]	DEF_COL_ CHC[2]	DEF_COL_ CHC[1]	DEF_COL_ CHC[0]
0xC9	0x2C	R/W						SWAP_SPLIT_ AV		DIS_AUTO_ PARAM_BUFF
0xCB	0x60	R/W			AUTO_SL_ FILTER_ FREEZE_EN				HDMI_CP_ LOCK_ THRESHOLD[1]	HDMI_CP_ LOCK_ THRESHOLD[0]
0xE0	0x00	R		HDMI_CP_ AUTOPARM_ LOCKED	HDMI_ AUTOPARM_ STS[1]	HDMI_ AUTOPARM_ STS[0]			CP_AGC_ GAIN[9]	CP_AGC_ GAIN[8]
0xE1	0x00	R	CP_AGC_ GAIN[7]	CP_AGC_ GAIN[6]	CP_AGC_ GAIN[5]	CP_AGC_ GAIN[4]	CP_AGC_ GAIN[3]	CP_AGC_ GAIN[2]	CP_AGC_ GAIN[1]	CP_AGC_ GAIN[0]
0xF3	0xD4	R/W			CH1_FL_FR_ THRESHOLD[2]	CH1_FL_FR_ THRESHOLD[1]	CH1_FL_FR_ THRESHOLD[0]	CH1_F_RUN_ THR[2]	CH1_F_RUN_ THR[1]	CH1_F_RUN_ THR[0]
0xF4	0x00	R	CSC_COEFF_ SEL_RB[3]	CSC_COEFF_ SEL_RB[2]	CSC_COEFF_ SEL_RB[1]	CSC_COEFF_ SEL_RB[0]				
0xF5	0x00	R/W							BYPASS_STDI1_ LOCKING	
0xFF	0x00	R				CP_FREE_RUN				

CEC REGISTER MAP

Add is the register map I²C address, Def is the default value of the register, and Acc is the read/write access for the register (R means read only, and R/W means read/write access).

Table 7. ADV7613 CEC Register Map

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x00	0x00	R/W	CEC_TX_FRAME_HEADER[7]	CEC_TX_FRAME_HEADER[6]	CEC_TX_FRAME_HEADER[5]	CEC_TX_FRAME_HEADER[4]	CEC_TX_FRAME_HEADER[3]	CEC_TX_FRAME_HEADER[2]	CEC_TX_FRAME_HEADER[1]	CEC_TX_FRAME_HEADER[0]
0x01	0x00	R/W	CEC_TX_FRAME_DATA0[7]	CEC_TX_FRAME_DATA0[6]	CEC_TX_FRAME_DATA0[5]	CEC_TX_FRAME_DATA0[4]	CEC_TX_FRAME_DATA0[3]	CEC_TX_FRAME_DATA0[2]	CEC_TX_FRAME_DATA0[1]	CEC_TX_FRAME_DATA0[0]
0x02	0x00	R/W	CEC_TX_FRAME_DATA1[7]	CEC_TX_FRAME_DATA1[6]	CEC_TX_FRAME_DATA1[5]	CEC_TX_FRAME_DATA1[4]	CEC_TX_FRAME_DATA1[3]	CEC_TX_FRAME_DATA1[2]	CEC_TX_FRAME_DATA1[1]	CEC_TX_FRAME_DATA1[0]
0x03	0x00	R/W	CEC_TX_FRAME_DATA2[7]	CEC_TX_FRAME_DATA2[6]	CEC_TX_FRAME_DATA2[5]	CEC_TX_FRAME_DATA2[4]	CEC_TX_FRAME_DATA2[3]	CEC_TX_FRAME_DATA2[2]	CEC_TX_FRAME_DATA2[1]	CEC_TX_FRAME_DATA2[0]
0x04	0x00	R/W	CEC_TX_FRAME_DATA3[7]	CEC_TX_FRAME_DATA3[6]	CEC_TX_FRAME_DATA3[5]	CEC_TX_FRAME_DATA3[4]	CEC_TX_FRAME_DATA3[3]	CEC_TX_FRAME_DATA3[2]	CEC_TX_FRAME_DATA3[1]	CEC_TX_FRAME_DATA3[0]
0x05	0x00	R/W	CEC_TX_FRAME_DATA4[7]	CEC_TX_FRAME_DATA4[6]	CEC_TX_FRAME_DATA4[5]	CEC_TX_FRAME_DATA4[4]	CEC_TX_FRAME_DATA4[3]	CEC_TX_FRAME_DATA4[2]	CEC_TX_FRAME_DATA4[1]	CEC_TX_FRAME_DATA4[0]
0x06	0x00	R/W	CEC_TX_FRAME_DATA5[7]	CEC_TX_FRAME_DATA5[6]	CEC_TX_FRAME_DATA5[5]	CEC_TX_FRAME_DATA5[4]	CEC_TX_FRAME_DATA5[3]	CEC_TX_FRAME_DATA5[2]	CEC_TX_FRAME_DATA5[1]	CEC_TX_FRAME_DATA5[0]
0x07	0x00	R/W	CEC_TX_FRAME_DATA6[7]	CEC_TX_FRAME_DATA6[6]	CEC_TX_FRAME_DATA6[5]	CEC_TX_FRAME_DATA6[4]	CEC_TX_FRAME_DATA6[3]	CEC_TX_FRAME_DATA6[2]	CEC_TX_FRAME_DATA6[1]	CEC_TX_FRAME_DATA6[0]
0x08	0x00	R/W	CEC_TX_FRAME_DATA7[7]	CEC_TX_FRAME_DATA7[6]	CEC_TX_FRAME_DATA7[5]	CEC_TX_FRAME_DATA7[4]	CEC_TX_FRAME_DATA7[3]	CEC_TX_FRAME_DATA7[2]	CEC_TX_FRAME_DATA7[1]	CEC_TX_FRAME_DATA7[0]
0x09	0x00	R/W	CEC_TX_FRAME_DATA8[7]	CEC_TX_FRAME_DATA8[6]	CEC_TX_FRAME_DATA8[5]	CEC_TX_FRAME_DATA8[4]	CEC_TX_FRAME_DATA8[3]	CEC_TX_FRAME_DATA8[2]	CEC_TX_FRAME_DATA8[1]	CEC_TX_FRAME_DATA8[0]
0x0A	0x00	R/W	CEC_TX_FRAME_DATA9[7]	CEC_TX_FRAME_DATA9[6]	CEC_TX_FRAME_DATA9[5]	CEC_TX_FRAME_DATA9[4]	CEC_TX_FRAME_DATA9[3]	CEC_TX_FRAME_DATA9[2]	CEC_TX_FRAME_DATA9[1]	CEC_TX_FRAME_DATA9[0]
0x0B	0x00	R/W	CEC_TX_FRAME_DATA10[7]	CEC_TX_FRAME_DATA10[6]	CEC_TX_FRAME_DATA10[5]	CEC_TX_FRAME_DATA10[4]	CEC_TX_FRAME_DATA10[3]	CEC_TX_FRAME_DATA10[2]	CEC_TX_FRAME_DATA10[1]	CEC_TX_FRAME_DATA10[0]
0x0C	0x00	R/W	CEC_TX_FRAME_DATA11[7]	CEC_TX_FRAME_DATA11[6]	CEC_TX_FRAME_DATA11[5]	CEC_TX_FRAME_DATA11[4]	CEC_TX_FRAME_DATA11[3]	CEC_TX_FRAME_DATA11[2]	CEC_TX_FRAME_DATA11[1]	CEC_TX_FRAME_DATA11[0]
0x0D	0x00	R/W	CEC_TX_FRAME_DATA12[7]	CEC_TX_FRAME_DATA12[6]	CEC_TX_FRAME_DATA12[5]	CEC_TX_FRAME_DATA12[4]	CEC_TX_FRAME_DATA12[3]	CEC_TX_FRAME_DATA12[2]	CEC_TX_FRAME_DATA12[1]	CEC_TX_FRAME_DATA12[0]
0x0E	0x00	R/W	CEC_TX_FRAME_DATA13[7]	CEC_TX_FRAME_DATA13[6]	CEC_TX_FRAME_DATA13[5]	CEC_TX_FRAME_DATA13[4]	CEC_TX_FRAME_DATA13[3]	CEC_TX_FRAME_DATA13[2]	CEC_TX_FRAME_DATA13[1]	CEC_TX_FRAME_DATA13[0]
0x0F	0x00	R/W	CEC_TX_FRAME_DATA14[7]	CEC_TX_FRAME_DATA14[6]	CEC_TX_FRAME_DATA14[5]	CEC_TX_FRAME_DATA14[4]	CEC_TX_FRAME_DATA14[3]	CEC_TX_FRAME_DATA14[2]	CEC_TX_FRAME_DATA14[1]	CEC_TX_FRAME_DATA14[0]
0x10	0x00	R/W				CEC_TX_FRAME_LENGTH[4]	CEC_TX_FRAME_LENGTH[3]	CEC_TX_FRAME_LENGTH[2]	CEC_TX_FRAME_LENGTH[1]	CEC_TX_FRAME_LENGTH[0]
0x11	0x00	R/W								CEC_TX_ENABLE
0x12	0x13	R/W		CEC_TX_RETRY[2]	CEC_TX_RETRY[1]	CEC_TX_RETRY[0]	CEC_RETRY_SFT[3]	CEC_RETRY_SFT[2]	CEC_RETRY_SFT[1]	CEC_RETRY_SFT[0]
0x13	0x57	R/W	CEC_TX_SFT[3]	CEC_TX_SFT[2]	CEC_TX_SFT[1]	CEC_TX_SFT[0]	CEC_TX_SFT[3]	CEC_TX_SFT[2]	CEC_TX_SFT[1]	CEC_TX_SFT[0]
0x14	0x00	R	CEC_TX_LOWDRIVE_COUNTER[3]	CEC_TX_LOWDRIVE_COUNTER[2]	CEC_TX_LOWDRIVE_COUNTER[1]	CEC_TX_LOWDRIVE_COUNTER[0]	CEC_TX_NACK_COUNTER[3]	CEC_TX_NACK_COUNTER[2]	CEC_TX_NACK_COUNTER[1]	CEC_TX_NACK_COUNTER[0]

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x15	0x00	R	CEC_BUF0_RX_FRAME_HEADER[7]	CEC_BUF0_RX_FRAME_HEADER[6]	CEC_BUF0_RX_FRAME_HEADER[5]	CEC_BUF0_RX_FRAME_HEADER[4]	CEC_BUF0_RX_FRAME_HEADER[3]	CEC_BUF0_RX_FRAME_HEADER[2]	CEC_BUF0_RX_FRAME_HEADER[1]	CEC_BUF0_RX_FRAME_HEADER[0]
0x16	0x00	R	CEC_BUF0_RX_FRAME_DATA0[7]	CEC_BUF0_RX_FRAME_DATA0[6]	CEC_BUF0_RX_FRAME_DATA0[5]	CEC_BUF0_RX_FRAME_DATA0[4]	CEC_BUF0_RX_FRAME_DATA0[3]	CEC_BUF0_RX_FRAME_DATA0[2]	CEC_BUF0_RX_FRAME_DATA0[1]	CEC_BUF0_RX_FRAME_DATA0[0]
0x17	0x00	R	CEC_BUF0_RX_FRAME_DATA1[7]	CEC_BUF0_RX_FRAME_DATA1[6]	CEC_BUF0_RX_FRAME_DATA1[5]	CEC_BUF0_RX_FRAME_DATA1[4]	CEC_BUF0_RX_FRAME_DATA1[3]	CEC_BUF0_RX_FRAME_DATA1[2]	CEC_BUF0_RX_FRAME_DATA1[1]	CEC_BUF0_RX_FRAME_DATA1[0]
0x18	0x00	R	CEC_BUF0_RX_FRAME_DATA2[7]	CEC_BUF0_RX_FRAME_DATA2[6]	CEC_BUF0_RX_FRAME_DATA2[5]	CEC_BUF0_RX_FRAME_DATA2[4]	CEC_BUF0_RX_FRAME_DATA2[3]	CEC_BUF0_RX_FRAME_DATA2[2]	CEC_BUF0_RX_FRAME_DATA2[1]	CEC_BUF0_RX_FRAME_DATA2[0]
0x19	0x00	R	CEC_BUF0_RX_FRAME_DATA3[7]	CEC_BUF0_RX_FRAME_DATA3[6]	CEC_BUF0_RX_FRAME_DATA3[5]	CEC_BUF0_RX_FRAME_DATA3[4]	CEC_BUF0_RX_FRAME_DATA3[3]	CEC_BUF0_RX_FRAME_DATA3[2]	CEC_BUF0_RX_FRAME_DATA3[1]	CEC_BUF0_RX_FRAME_DATA3[0]
0x1A	0x00	R	CEC_BUF0_RX_FRAME_DATA4[7]	CEC_BUF0_RX_FRAME_DATA4[6]	CEC_BUF0_RX_FRAME_DATA4[5]	CEC_BUF0_RX_FRAME_DATA4[4]	CEC_BUF0_RX_FRAME_DATA4[3]	CEC_BUF0_RX_FRAME_DATA4[2]	CEC_BUF0_RX_FRAME_DATA4[1]	CEC_BUF0_RX_FRAME_DATA4[0]
0x1B	0x00	R	CEC_BUF0_RX_FRAME_DATA5[7]	CEC_BUF0_RX_FRAME_DATA5[6]	CEC_BUF0_RX_FRAME_DATA5[5]	CEC_BUF0_RX_FRAME_DATA5[4]	CEC_BUF0_RX_FRAME_DATA5[3]	CEC_BUF0_RX_FRAME_DATA5[2]	CEC_BUF0_RX_FRAME_DATA5[1]	CEC_BUF0_RX_FRAME_DATA5[0]
0x1C	0x00	R	CEC_BUF0_RX_FRAME_DATA6[7]	CEC_BUF0_RX_FRAME_DATA6[6]	CEC_BUF0_RX_FRAME_DATA6[5]	CEC_BUF0_RX_FRAME_DATA6[4]	CEC_BUF0_RX_FRAME_DATA6[3]	CEC_BUF0_RX_FRAME_DATA6[2]	CEC_BUF0_RX_FRAME_DATA6[1]	CEC_BUF0_RX_FRAME_DATA6[0]
0x1D	0x00	R	CEC_BUF0_RX_FRAME_DATA7[7]	CEC_BUF0_RX_FRAME_DATA7[6]	CEC_BUF0_RX_FRAME_DATA7[5]	CEC_BUF0_RX_FRAME_DATA7[4]	CEC_BUF0_RX_FRAME_DATA7[3]	CEC_BUF0_RX_FRAME_DATA7[2]	CEC_BUF0_RX_FRAME_DATA7[1]	CEC_BUF0_RX_FRAME_DATA7[0]
0x1E	0x00	R	CEC_BUF0_RX_FRAME_DATA8[7]	CEC_BUF0_RX_FRAME_DATA8[6]	CEC_BUF0_RX_FRAME_DATA8[5]	CEC_BUF0_RX_FRAME_DATA8[4]	CEC_BUF0_RX_FRAME_DATA8[3]	CEC_BUF0_RX_FRAME_DATA8[2]	CEC_BUF0_RX_FRAME_DATA8[1]	CEC_BUF0_RX_FRAME_DATA8[0]
0x1F	0x00	R	CEC_BUF0_RX_FRAME_DATA9[7]	CEC_BUF0_RX_FRAME_DATA9[6]	CEC_BUF0_RX_FRAME_DATA9[5]	CEC_BUF0_RX_FRAME_DATA9[4]	CEC_BUF0_RX_FRAME_DATA9[3]	CEC_BUF0_RX_FRAME_DATA9[2]	CEC_BUF0_RX_FRAME_DATA9[1]	CEC_BUF0_RX_FRAME_DATA9[0]
0x20	0x00	R	CEC_BUF0_RX_FRAME_DATA10[7]	CEC_BUF0_RX_FRAME_DATA10[6]	CEC_BUF0_RX_FRAME_DATA10[5]	CEC_BUF0_RX_FRAME_DATA10[4]	CEC_BUF0_RX_FRAME_DATA10[3]	CEC_BUF0_RX_FRAME_DATA10[2]	CEC_BUF0_RX_FRAME_DATA10[1]	CEC_BUF0_RX_FRAME_DATA10[0]
0x21	0x00	R	CEC_BUF0_RX_FRAME_DATA11[7]	CEC_BUF0_RX_FRAME_DATA11[6]	CEC_BUF0_RX_FRAME_DATA11[5]	CEC_BUF0_RX_FRAME_DATA11[4]	CEC_BUF0_RX_FRAME_DATA11[3]	CEC_BUF0_RX_FRAME_DATA11[2]	CEC_BUF0_RX_FRAME_DATA11[1]	CEC_BUF0_RX_FRAME_DATA11[0]
0x22	0x00	R	CEC_BUF0_RX_FRAME_DATA12[7]	CEC_BUF0_RX_FRAME_DATA12[6]	CEC_BUF0_RX_FRAME_DATA12[5]	CEC_BUF0_RX_FRAME_DATA12[4]	CEC_BUF0_RX_FRAME_DATA12[3]	CEC_BUF0_RX_FRAME_DATA12[2]	CEC_BUF0_RX_FRAME_DATA12[1]	CEC_BUF0_RX_FRAME_DATA12[0]
0x23	0x00	R	CEC_BUF0_RX_FRAME_DATA13[7]	CEC_BUF0_RX_FRAME_DATA13[6]	CEC_BUF0_RX_FRAME_DATA13[5]	CEC_BUF0_RX_FRAME_DATA13[4]	CEC_BUF0_RX_FRAME_DATA13[3]	CEC_BUF0_RX_FRAME_DATA13[2]	CEC_BUF0_RX_FRAME_DATA13[1]	CEC_BUF0_RX_FRAME_DATA13[0]
0x24	0x00	R	CEC_BUF0_RX_FRAME_DATA14[7]	CEC_BUF0_RX_FRAME_DATA14[6]	CEC_BUF0_RX_FRAME_DATA14[5]	CEC_BUF0_RX_FRAME_DATA14[4]	CEC_BUF0_RX_FRAME_DATA14[3]	CEC_BUF0_RX_FRAME_DATA14[2]	CEC_BUF0_RX_FRAME_DATA14[1]	CEC_BUF0_RX_FRAME_DATA14[0]
0x25	0x00	R				CEC_BUF0_RX_FRAME_LENGTH[4]	CEC_BUF0_RX_FRAME_LENGTH[3]	CEC_BUF0_RX_FRAME_LENGTH[2]	CEC_BUF0_RX_FRAME_LENGTH[1]	CEC_BUF0_RX_FRAME_LENGTH[0]
0x27	0x10	R/W		CEC_LOGICAL_ADDRESS_MASK[2]	CEC_LOGICAL_ADDRESS_MASK[1]	CEC_LOGICAL_ADDRESS_MASK[0]	CEC_ERROR_REPORT_MODE	CEC_ERROR_DET_MODE	CEC_FORCE_NACK	CEC_FORCE_IGNORE
0x28	0xFF	R/W	CEC_LOGICAL_ADDRESS1[3]	CEC_LOGICAL_ADDRESS1[2]	CEC_LOGICAL_ADDRESS1[1]	CEC_LOGICAL_ADDRESS1[0]	CEC_LOGICAL_ADDRESS0[3]	CEC_LOGICAL_ADDRESS0[2]	CEC_LOGICAL_ADDRESS0[1]	CEC_LOGICAL_ADDRESS0[0]
0x29	0x0F	R/W					CEC_LOGICAL_ADDRESS2[3]	CEC_LOGICAL_ADDRESS2[2]	CEC_LOGICAL_ADDRESS2[1]	CEC_LOGICAL_ADDRESS2[0]
0x2A	0x3E	R/W								CEC_POWER_UP
0x2B	0x07	R/W			CEC_GLITCH_FILTER_CTRL[5]	CEC_GLITCH_FILTER_CTRL[4]	CEC_GLITCH_FILTER_CTRL[3]	CEC_GLITCH_FILTER_CTRL[2]	CEC_GLITCH_FILTER_CTRL[1]	CEC_GLITCH_FILTER_CTRL[0]
0x2C	0x00	SC					CEC_CLR_RX_RDY2	CEC_CLR_RX_RDY1	CEC_CLR_RX_RDY0	CEC_SOFT_RESET
0x4C	0x00	R/W						CEC_DIS_AUTO_MODE		

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x53	0x00	R			CEC_BUF2_ TIMESTAMP[1]	CEC_BUF2_ TIMESTAMP[0]	CEC_BUF1_ TIMESTAMP[1]	CEC_BUF1_ TIMESTAMP[0]	CEC_BUF0_ TIMESTAMP[1]	CEC_BUF0_ TIMESTAMP[0]
0x54	0x00	R	CEC_BUF1_ RX_FRAME_ HEADER[7]	CEC_BUF1_RX_ FRAME_ HEADER[6]	CEC_BUF1_RX_ FRAME_ HEADER[5]	CEC_BUF1_ RX_FRAME_ HEADER[4]	CEC_BUF1_ RX_FRAME_ HEADER[3]	CEC_BUF1_ RX_FRAME_ HEADER[2]	CEC_BUF1_ RX_FRAME_ HEADER[1]	CEC_BUF1_ RX_FRAME_ HEADER[0]
0x55	0x00	R	CEC_BUF1_RX_ FRAME_ DATA0[7]	CEC_BUF1_RX_ FRAME_ DATA0[6]	CEC_BUF1_RX_ FRAME_ DATA0[5]	CEC_BUF1_RX_ FRAME_ DATA0[4]	CEC_BUF1_RX_ FRAME_ DATA0[3]	CEC_BUF1_RX_ FRAME_ DATA0[2]	CEC_BUF1_RX_ FRAME_ DATA0[1]	CEC_BUF1_RX_ FRAME_ DATA0[0]
0x56	0x00	R	CEC_BUF1_RX_ FRAME_ DATA1[7]	CEC_BUF1_RX_ FRAME_ DATA1[6]	CEC_BUF1_RX_ FRAME_ DATA1[5]	CEC_BUF1_RX_ FRAME_ DATA1[4]	CEC_BUF1_RX_ FRAME_ DATA1[3]	CEC_BUF1_RX_ FRAME_ DATA1[2]	CEC_BUF1_RX_ FRAME_ DATA1[1]	CEC_BUF1_RX_ FRAME_ DATA1[0]
0x57	0x00	R	CEC_BUF1_RX_ FRAME_ DATA2[7]	CEC_BUF1_RX_ FRAME_ DATA2[6]	CEC_BUF1_RX_ FRAME_ DATA2[5]	CEC_BUF1_RX_ FRAME_ DATA2[4]	CEC_BUF1_RX_ FRAME_ DATA2[3]	CEC_BUF1_RX_ FRAME_ DATA2[2]	CEC_BUF1_RX_ FRAME_ DATA2[1]	CEC_BUF1_RX_ FRAME_ DATA2[0]
0x58	0x00	R	CEC_BUF1_RX_ FRAME_ DATA3[7]	CEC_BUF1_RX_ FRAME_ DATA3[6]	CEC_BUF1_RX_ FRAME_ DATA3[5]	CEC_BUF1_RX_ FRAME_ DATA3[4]	CEC_BUF1_RX_ FRAME_ DATA3[3]	CEC_BUF1_RX_ FRAME_ DATA3[2]	CEC_BUF1_RX_ FRAME_ DATA3[1]	CEC_BUF1_RX_ FRAME_ DATA3[0]
0x59	0x00	R	CEC_BUF1_RX_ FRAME_ DATA4[7]	CEC_BUF1_RX_ FRAME_ DATA4[6]	CEC_BUF1_RX_ FRAME_ DATA4[5]	CEC_BUF1_RX_ FRAME_ DATA4[4]	CEC_BUF1_RX_ FRAME_ DATA4[3]	CEC_BUF1_RX_ FRAME_ DATA4[2]	CEC_BUF1_RX_ FRAME_ DATA4[1]	CEC_BUF1_RX_ FRAME_DATA4[0]
0x5A	0x00	R	CEC_BUF1_RX_ FRAME_ DATA5[7]	CEC_BUF1_RX_ FRAME_ DATA5[6]	CEC_BUF1_RX_ FRAME_ DATA5[5]	CEC_BUF1_RX_ FRAME_ DATA5[4]	CEC_BUF1_RX_ FRAME_ DATA5[3]	CEC_BUF1_RX_ FRAME_ DATA5[2]	CEC_BUF1_RX_ FRAME_ DATA5[1]	CEC_BUF1_RX_ FRAME_ DATA5[0]
0x5B	0x00	R	CEC_BUF1_RX_ FRAME_ DATA6[7]	CEC_BUF1_RX_ FRAME_ DATA6[6]	CEC_BUF1_RX_ FRAME_ DATA6[5]	CEC_BUF1_RX_ FRAME_ DATA6[4]	CEC_BUF1_RX_ FRAME_ DATA6[3]	CEC_BUF1_RX_ FRAME_ DATA6[2]	CEC_BUF1_RX_ FRAME_ DATA6[1]	CEC_BUF1_RX_ FRAME_ DATA6[0]
0x5C	0x00	R	CEC_BUF1_RX_ FRAME_ DATA7[7]	CEC_BUF1_RX_ FRAME_ DATA7[6]	CEC_BUF1_RX_ FRAME_ DATA7[5]	CEC_BUF1_RX_ FRAME_ DATA7[4]	CEC_BUF1_RX_ FRAME_ DATA7[3]	CEC_BUF1_RX_ FRAME_ DATA7[2]	CEC_BUF1_RX_ FRAME_ DATA7[1]	CEC_BUF1_RX_ FRAME_ DATA7[0]
0x5D	0x00	R	CEC_BUF1_RX_ FRAME_ DATA8[7]	CEC_BUF1_RX_ FRAME_ DATA8[6]	CEC_BUF1_RX_ FRAME_ DATA8[5]	CEC_BUF1_RX_ FRAME_ DATA8[4]	CEC_BUF1_RX_ FRAME_ DATA8[3]	CEC_BUF1_RX_ FRAME_ DATA8[2]	CEC_BUF1_RX_ FRAME_ DATA8[1]	CEC_BUF1_RX_ FRAME_ DATA8[0]
0x5E	0x00	R	CEC_BUF1_RX_ FRAME_ DATA9[7]	CEC_BUF1_RX_ FRAME_ DATA9[6]	CEC_BUF1_RX_ FRAME_ DATA9[5]	CEC_BUF1_RX_ FRAME_ DATA9[4]	CEC_BUF1_RX_ FRAME_ DATA9[3]	CEC_BUF1_RX_ FRAME_ DATA9[2]	CEC_BUF1_RX_ FRAME_ DATA9[1]	CEC_BUF1_RX_ FRAME_ DATA9[0]
0x5F	0x00	R	CEC_BUF1_ RX_FRAME_ DATA10[7]	CEC_BUF1_RX_ FRAME_ DATA10[6]	CEC_BUF1_RX_ FRAME_ DATA10[5]	CEC_BUF1_ RX_FRAME_ DATA10[4]	CEC_BUF1_RX_ FRAME_ DATA10[3]	CEC_BUF1_RX_ FRAME_ DATA10[2]	CEC_BUF1_RX_ FRAME_ DATA10[1]	CEC_BUF1_RX_ FRAME_ DATA10[0]
0x60	0x00	R	CEC_BUF1_ RX_FRAME_ DATA11[7]	CEC_BUF1_RX_ FRAME_ DATA11[6]	CEC_BUF1_RX_ FRAME_ DATA11[5]	CEC_BUF1_RX_ FRAME_ DATA11[4]	CEC_BUF1_RX_ FRAME_ DATA11[3]	CEC_BUF1_RX_ FRAME_ DATA11[2]	CEC_BUF1_RX_ FRAME_ DATA11[1]	CEC_BUF1_RX_ FRAME_ DATA11[0]
0x61	0x00	R	CEC_BUF1_ RX_FRAME_ DATA12[7]	CEC_BUF1_RX_ FRAME_ DATA12[6]	CEC_BUF1_RX_ FRAME_ DATA12[5]	CEC_BUF1_RX_ FRAME_ DATA12[4]	CEC_BUF1_RX_ FRAME_ DATA12[3]	CEC_BUF1_RX_ FRAME_ DATA12[2]	CEC_BUF1_RX_ FRAME_ DATA12[1]	CEC_BUF1_RX_ FRAME_ DATA12[0]
0x62	0x00	R	CEC_BUF1_ RX_FRAME_ DATA13[7]	CEC_BUF1_RX_ FRAME_ DATA13[6]	CEC_BUF1_RX_ FRAME_ DATA13[5]	CEC_BUF1_RX_ FRAME_ DATA13[4]	CEC_BUF1_RX_ FRAME_ DATA13[3]	CEC_BUF1_RX_ FRAME_ DATA13[2]	CEC_BUF1_RX_ FRAME_ DATA13[1]	CEC_BUF1_RX_ FRAME_ DATA13[0]
0x63	0x00	R	CEC_BUF1_ RX_FRAME_ DATA14[7]	CEC_BUF1_ RX_FRAME_ DATA14[6]	CEC_BUF1_RX_ FRAME_ DATA14[5]	CEC_BUF1_RX_ FRAME_ DATA14[4]	CEC_BUF1_RX_ FRAME_ DATA14[3]	CEC_BUF1_RX_ FRAME_ DATA14[2]	CEC_BUF1_RX_ FRAME_ DATA14[1]	CEC_BUF1_RX_ FRAME_ DATA14[0]
0x64	0x00	R				CEC_BUF1_RX_ FRAME_ LENGTH[4]	CEC_BUF1_RX_ FRAME_ LENGTH[3]	CEC_BUF1_RX_ FRAME_ LENGTH[2]	CEC_BUF1_RX_ FRAME_ LENGTH[1]	CEC_BUF1_RX_ FRAME_ LENGTH[0]
0x65	0x00	R	CEC_BUF2_ RX_FRAME_ HEADER[7]	CEC_BUF2_RX_ FRAME_ HEADER[6]	CEC_BUF2_RX_ FRAME_ HEADER[5]	CEC_BUF2_RX_ FRAME_ HEADER[4]	CEC_BUF2_RX_ FRAME_ HEADER[3]	CEC_BUF2_RX_ FRAME_ HEADER[2]	CEC_BUF2_RX_ FRAME_ HEADER[1]	CEC_BUF2_RX_ FRAME_ HEADER[0]
0x66	0x00	R	CEC_BUF2_RX_ FRAME_ DATA0[7]	CEC_BUF2_RX_ FRAME_ DATA0[6]	CEC_BUF2_RX_ FRAME_ DATA0[5]	CEC_BUF2_RX_ FRAME_ DATA0[4]	CEC_BUF2_RX_ FRAME_ DATA0[3]	CEC_BUF2_RX_ FRAME_ DATA0[2]	CEC_BUF2_RX_ FRAME_ DATA0[1]	CEC_BUF2_RX_ FRAME_ DATA0[0]
0x67	0x00	R	CEC_BUF2_RX_ FRAME_ DATA1[7]	CEC_BUF2_RX_ FRAME_ DATA1[6]	CEC_BUF2_RX_ FRAME_ DATA1[5]	CEC_BUF2_RX_ FRAME_ DATA1[4]	CEC_BUF2_RX_ FRAME_ DATA1[3]	CEC_BUF2_RX_ FRAME_ DATA1[2]	CEC_BUF2_RX_ FRAME_ DATA1[1]	CEC_BUF2_RX_ FRAME_ DATA1[0]
0x68	0x00	R	CEC_BUF2_RX_ FRAME_ DATA2[7]	CEC_BUF2_RX_ FRAME_ DATA2[6]	CEC_BUF2_RX_ FRAME_ DATA2[5]	CEC_BUF2_RX_ FRAME_ DATA2[4]	CEC_BUF2_RX_ FRAME_ DATA2[3]	CEC_BUF2_RX_ FRAME_ DATA2[2]	CEC_BUF2_RX_ FRAME_ DATA2[1]	CEC_BUF2_RX_ FRAME_ DATA2[0]

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x69	0x00	R	CEC_BUF2_RX_FRAME_DATA3[7]	CEC_BUF2_RX_FRAME_DATA3[6]	CEC_BUF2_RX_FRAME_DATA3[5]	CEC_BUF2_RX_FRAME_DATA3[4]	CEC_BUF2_RX_FRAME_DATA3[3]	CEC_BUF2_RX_FRAME_DATA3[2]	CEC_BUF2_RX_FRAME_DATA3[1]	CEC_BUF2_RX_FRAME_DATA3[0]
0x6A	0x00	R	CEC_BUF2_RX_FRAME_DATA4[7]	CEC_BUF2_RX_FRAME_DATA4[6]	CEC_BUF2_RX_FRAME_DATA4[5]	CEC_BUF2_RX_FRAME_DATA4[4]	CEC_BUF2_RX_FRAME_DATA4[3]	CEC_BUF2_RX_FRAME_DATA4[2]	CEC_BUF2_RX_FRAME_DATA4[1]	CEC_BUF2_RX_FRAME_DATA4[0]
0x6B	0x00	R	CEC_BUF2_RX_FRAME_DATA5[7]	CEC_BUF2_RX_FRAME_DATA5[6]	CEC_BUF2_RX_FRAME_DATA5[5]	CEC_BUF2_RX_FRAME_DATA5[4]	CEC_BUF2_RX_FRAME_DATA5[3]	CEC_BUF2_RX_FRAME_DATA5[2]	CEC_BUF2_RX_FRAME_DATA5[1]	CEC_BUF2_RX_FRAME_DATA5[0]
0x6C	0x00	R	CEC_BUF2_RX_FRAME_DATA6[7]	CEC_BUF2_RX_FRAME_DATA6[6]	CEC_BUF2_RX_FRAME_DATA6[5]	CEC_BUF2_RX_FRAME_DATA6[4]	CEC_BUF2_RX_FRAME_DATA6[3]	CEC_BUF2_RX_FRAME_DATA6[2]	CEC_BUF2_RX_FRAME_DATA6[1]	CEC_BUF2_RX_FRAME_DATA6[0]
0x6D	0x00	R	CEC_BUF2_RX_FRAME_DATA7[7]	CEC_BUF2_RX_FRAME_DATA7[6]	CEC_BUF2_RX_FRAME_DATA7[5]	CEC_BUF2_RX_FRAME_DATA7[4]	CEC_BUF2_RX_FRAME_DATA7[3]	CEC_BUF2_RX_FRAME_DATA7[2]	CEC_BUF2_RX_FRAME_DATA7[1]	CEC_BUF2_RX_FRAME_DATA7[0]
0x6E	0x00	R	CEC_BUF2_RX_FRAME_DATA8[7]	CEC_BUF2_RX_FRAME_DATA8[6]	CEC_BUF2_RX_FRAME_DATA8[5]	CEC_BUF2_RX_FRAME_DATA8[4]	CEC_BUF2_RX_FRAME_DATA8[3]	CEC_BUF2_RX_FRAME_DATA8[2]	CEC_BUF2_RX_FRAME_DATA8[1]	CEC_BUF2_RX_FRAME_DATA8[0]
0x6F	0x00	R	CEC_BUF2_RX_FRAME_DATA9[7]	CEC_BUF2_RX_FRAME_DATA9[6]	CEC_BUF2_RX_FRAME_DATA9[5]	CEC_BUF2_RX_FRAME_DATA9[4]	CEC_BUF2_RX_FRAME_DATA9[3]	CEC_BUF2_RX_FRAME_DATA9[2]	CEC_BUF2_RX_FRAME_DATA9[1]	CEC_BUF2_RX_FRAME_DATA9[0]
0x70	0x00	R	CEC_BUF2_RX_FRAME_DATA10[7]	CEC_BUF2_RX_FRAME_DATA10[6]	CEC_BUF2_RX_FRAME_DATA10[5]	CEC_BUF2_RX_FRAME_DATA10[4]	CEC_BUF2_RX_FRAME_DATA10[3]	CEC_BUF2_RX_FRAME_DATA10[2]	CEC_BUF2_RX_FRAME_DATA10[1]	CEC_BUF2_RX_FRAME_DATA10[0]
0x71	0x00	R	CEC_BUF2_RX_FRAME_DATA11[7]	CEC_BUF2_RX_FRAME_DATA11[6]	CEC_BUF2_RX_FRAME_DATA11[5]	CEC_BUF2_RX_FRAME_DATA11[4]	CEC_BUF2_RX_FRAME_DATA11[3]	CEC_BUF2_RX_FRAME_DATA11[2]	CEC_BUF2_RX_FRAME_DATA11[1]	CEC_BUF2_RX_FRAME_DATA11[0]
0x72	0x00	R	CEC_BUF2_RX_FRAME_DATA12[7]	CEC_BUF2_RX_FRAME_DATA12[6]	CEC_BUF2_RX_FRAME_DATA12[5]	CEC_BUF2_RX_FRAME_DATA12[4]	CEC_BUF2_RX_FRAME_DATA12[3]	CEC_BUF2_RX_FRAME_DATA12[2]	CEC_BUF2_RX_FRAME_DATA12[1]	CEC_BUF2_RX_FRAME_DATA12[0]
0x73	0x00	R	CEC_BUF2_RX_FRAME_DATA13[7]	CEC_BUF2_RX_FRAME_DATA13[6]	CEC_BUF2_RX_FRAME_DATA13[5]	CEC_BUF2_RX_FRAME_DATA13[4]	CEC_BUF2_RX_FRAME_DATA13[3]	CEC_BUF2_RX_FRAME_DATA13[2]	CEC_BUF2_RX_FRAME_DATA13[1]	CEC_BUF2_RX_FRAME_DATA13[0]
0x74	0x00	R	CEC_BUF2_RX_FRAME_DATA14[7]	CEC_BUF2_RX_FRAME_DATA14[6]	CEC_BUF2_RX_FRAME_DATA14[5]	CEC_BUF2_RX_FRAME_DATA14[4]	CEC_BUF2_RX_FRAME_DATA14[3]	CEC_BUF2_RX_FRAME_DATA14[2]	CEC_BUF2_RX_FRAME_DATA14[1]	CEC_BUF2_RX_FRAME_DATA14[0]
0x75	0x00	R				CEC_BUF2_RX_FRAME_LENGTH[4]	CEC_BUF2_RX_FRAME_LENGTH[3]	CEC_BUF2_RX_FRAME_LENGTH[2]	CEC_BUF2_RX_FRAME_LENGTH[1]	CEC_BUF2_RX_FRAME_LENGTH[0]
0x76	0x00	R						CEC_RX_RDY2	CEC_RX_RDY1	CEC_RX_RDY0
0x77	0x00	R/W								CEC_USE_ALL_BUFS
0x78	0x6D	R/W	CEC_WAKE_OPCODE0[7]	CEC_WAKE_OPCODE0[6]	CEC_WAKE_OPCODE0[5]	CEC_WAKE_OPCODE0[4]	CEC_WAKE_OPCODE0[3]	CEC_WAKE_OPCODE0[2]	CEC_WAKE_OPCODE0[1]	CEC_WAKE_OPCODE0[0]
0x79	0x8F	R/W	CEC_WAKE_OPCODE1[7]	CEC_WAKE_OPCODE1[6]	CEC_WAKE_OPCODE1[5]	CEC_WAKE_OPCODE1[4]	CEC_WAKE_OPCODE1[3]	CEC_WAKE_OPCODE1[2]	CEC_WAKE_OPCODE1[1]	CEC_WAKE_OPCODE1[0]
0x7A	0x82	R/W	CEC_WAKE_OPCODE2[7]	CEC_WAKE_OPCODE2[6]	CEC_WAKE_OPCODE2[5]	CEC_WAKE_OPCODE2[4]	CEC_WAKE_OPCODE2[3]	CEC_WAKE_OPCODE2[2]	CEC_WAKE_OPCODE2[1]	CEC_WAKE_OPCODE2[0]
0x7B	0x04	R/W	CEC_WAKE_OPCODE3[7]	CEC_WAKE_OPCODE3[6]	CEC_WAKE_OPCODE3[5]	CEC_WAKE_OPCODE3[4]	CEC_WAKE_OPCODE3[3]	CEC_WAKE_OPCODE3[2]	CEC_WAKE_OPCODE3[1]	CEC_WAKE_OPCODE3[0]
0x7C	0x0D	R/W	CEC_WAKE_OPCODE4[7]	CEC_WAKE_OPCODE4[6]	CEC_WAKE_OPCODE4[5]	CEC_WAKE_OPCODE4[4]	CEC_WAKE_OPCODE4[3]	CEC_WAKE_OPCODE4[2]	CEC_WAKE_OPCODE4[1]	CEC_WAKE_OPCODE4[0]
0x7D	0x70	R/W	CEC_WAKE_OPCODE5[7]	CEC_WAKE_OPCODE5[6]	CEC_WAKE_OPCODE5[5]	CEC_WAKE_OPCODE5[4]	CEC_WAKE_OPCODE5[3]	CEC_WAKE_OPCODE5[2]	CEC_WAKE_OPCODE5[1]	CEC_WAKE_OPCODE5[0]
0x7E	0x42	R/W	CEC_WAKE_OPCODE6[7]	CEC_WAKE_OPCODE6[6]	CEC_WAKE_OPCODE6[5]	CEC_WAKE_OPCODE6[4]	CEC_WAKE_OPCODE6[3]	CEC_WAKE_OPCODE6[2]	CEC_WAKE_OPCODE6[1]	CEC_WAKE_OPCODE6[0]
0x7F	0x41	R/W	CEC_WAKE_OPCODE7[7]	CEC_WAKE_OPCODE7[6]	CEC_WAKE_OPCODE7[5]	CEC_WAKE_OPCODE7[4]	CEC_WAKE_OPCODE7[3]	CEC_WAKE_OPCODE7[2]	CEC_WAKE_OPCODE7[1]	CEC_WAKE_OPCODE7[0]

OPENLDI Tx REGISTER MAP

Add is the register map I²C address, Def is the default value of the register, and Acc is the read/write access for the register (R means read only, and R/W means read/write access).

Table 8. ADV7613 OpenLDI Tx Register Map

Add	Def	Acc	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x40	0x02	R/W					TX_PLL_EN		TX_PDN	TX_MODE_ ITU656
0x44	0x00	R/W						TX_PLL_ GEAR[2]	TX_PLL_ GEAR[1]	TX_PLL_ GEAR[0]
0x4C	0x71	R/W		TX_OLDI_HS_ POL	TX_OLDI_VS_ POL	TX_OLDI_DE_ POL	TX_ENABLE_ NS_MAPPING	TX_656_ALL_ LANES_ENABLE	TX_OLDI_ BALANCED_ MODE	TX_COLOR_ MODE
0x4D	0xE3	R/W	TX_CLOCK_ LANE_CODE_ 656[7]	TX_CLOCK_ LANE_CODE_ 656[6]	TX_CLOCK_ LANE_CODE_ 656[5]	TX_CLOCK_ LANE_CODE_ 656[4]	TX_CLOCK_ LANE_CODE_ 656[3]	TX_CLOCK_ LANE_CODE_ 656[2]	TX_CLOCK_ LANE_CODE_ 656[1]	TX_CLOCK_ LANE_CODE_ 656[0]
0x4E	0x08	R/W							TX_INT_RES	TX_MUX_INT_ RES
0x4F	0x00	R						TX_PLL_LOCK_ DET	TX_DETECTED_ VS_POL	TX_DETECTED_ HS_POL

REGISTER BIT DESCRIPTIONS

IO REGISTER MAP BIT DESCRIPTIONS

Table 9. IO Register Map Bit Descriptions

Address	Name	Bits	Access	Description
0x00	VID_STD[5:0]	00**001000**	R/W	Sets the input video standards and oversampling mode. The configuration is dependent on PRIM_MODE[3:0]. 000010 = default value.
0x01	V_FREQ[2:0]	0**000**0110	R/W	A control to set the vertical frequency of HD component standards. 000 = 60 Hz. 001 = 50 Hz. 010 = 30 Hz. 011 = 25 Hz. 100 = 24 Hz. 101 = reserved. 110 = reserved. 111 = reserved.
0x01	PRIM_MODE[3:0]	0000**0110**	R/W	A control to select the primary mode of operation of the HDMI receiver (Rx). Use with VID_STD[5:0]. 0000 = reserved. 0001 = reserved. 0010 = reserved. 0011 = reserved. 0100 = reserved. 0101 = HDMI component. 0110 = HDMI graphics. 0111 to 1111 = reserved.
0x02	INP_COLOR_SPACE[3:0]	**1111**0000	R/W	A control to set the color space of the input video. Use in conjunction with ALT_GAMMA and RGB_OUT to configure the color space converter. A value of 4'b1111 selects the automatic setting of the input color space based on the primary mode and video standard settings. Setting 1000 to Setting 1110 are undefined. 0000 = forces RGB (range of 16 to 235) input. 0001 = forces RGB (range of 0 to 255) input. 0010 = forces YCrCb input (601 color space, range of 16 to 235). 0011 = forces YCrCb input (709 color space, range of 16 to 235). 0100 = forces XVYCC 601. 0101 = forces XVYCC 709. 0110 = forces YCrCb input (601 color space) (range of 0 to 255). 0111 = forces YCrCb input (709 color space) (range of 0 to 255). 1111 = in analog mode, input color space depends on the primary mode and video standard. In HDMI mode, the input color space depends on the color space reported by the HDMI block.
0x02	ALT_GAMMA	1111**0**000	R/W	A control to select the type of YPbPr color space conversion. Use this bit in conjunction with INP_COLOR_SPACE [3:0] and RGB_OUT. If ALT_GAMMA is set to 1 and RGB_OUT = 0, a color space conversion is applied to convert from 601 to 709 or 709 to 601. Valid only if RGB_OUT is set to 0. 0 = no conversion. 1 = YUV601 to YUV709 conversion applied if the input is YUV601. YUV709 to YUV601 conversion applied if the input is YUV709.

Address	Name	Bits	Access	Description
0x02	OP_656_RANGE	11110**0**00	R/W	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. 0 = enables the full output range (0 to 255). 1 = enables the limited output range (16 to 235).
0x02	RGB_OUT	111100**0**0	R/W	A control to select the 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. 0 = YPbPr color space output. 1 = RGB color space output.
0x02	ALT_DATA_SAT	1111000**0**	R/W	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. 0 = data saturator enabled or disabled according to the OP_656_RANGE setting. 1 = reverses the OP_656_RANGE setting to enable or disable the data saturator.
0x03	OP_FORMAT_SEL[7:0]	**00000000**	R/W	A control to select the data format configuration. 0x00 = reserved. 0x01 = reserved. 0x02 = reserved. 0x06 = reserved. 0x0A = reserved. 0x20 = reserved. 0x21 = reserved. 0x22 = reserved. 0x23 = reserved. 0x24 = reserved. 0x40 = reserved. 0x41 = reserved. 0x42 = 36-bit, 4:4:4 SDR Mode 0. 0x46 = reserved. 0x4C = reserved. 0x50 = reserved. 0x51 = reserved. 0x52 = reserved. 0x54 = reserved. 0x60 = reserved. 0x61 = reserved. 0x62 = reserved. 0x80 = reserved. 0x81 = reserved. 0x82 = reserved. 0x86 = reserved. 0x8A = reserved. 0x8D = reserved. 0x90 = reserved. 0x91 = reserved. 0x92 = reserved. 0x94 = reserved. 0x95 = reserved. 0x96 = reserved.

Address	Name	Bits	Access	Description
				0xC0 = reserved. 0xC1 = reserved. 0xC2 = reserved.
0x04	XTAL_FREQ_SEL[1:0]	01100**01**0	R/W	A control to set the crystal (XTAL) frequency used. 00 = reserved. 01 = 28.63636 MHz. 10 = reserved. 11 = reserved.
0x0B	CORE_PDN	0100010**0**	R/W	A power-down control for the DPP, CP core, and digital sections of the HDMI core. 0 = powers up the DPP, CP, and digital sections of the HDMI block. 1 = powers down the DPP, CP, and digital sections of the HDMI block. Standard identification (STDI) is still active when CORE_PDN is set to 1.
0x0B	XTAL_PDN	0100010**0**	R/W	A power-down control for the XTAL in the digital blocks. 0 = powers up the XTAL buffer to the digital core. 1 = powers down the XTAL buffer to the digital core.
0x0C	POWER_DOWN	01**1**00010	R/W	A control to enable power-down mode. This is the main I^2C power-down control. 0 = chip is operational. 1 = enables chip power-down.
0x0C	CP_PWRDN	01100**0**10	R/W	A power-down control for the CP core. 0 = powers up the clock to the CP core. 1 = powers down the clock to the CP core. The DPP and HDMI blocks are not affected by this bit.
0x0C	PADS_PDN	0110001**0**	R/W	A power-down control for the pads of the digital output pins. When enabled, the pads are tristated and the input path is disabled. This control applies to the INT pin. 0 = powers up the pads of the digital output pins. 1 = powers down the pads of the digital output pins.
0x12	CP_STDI_INTERLACED	000**0**0000	R	A readback to indicate the interlaced status of the currently selected STDI block that is applied to the CP core. 0 = selected STDI block has detected a progressive input. 1 = selected STDI block has detected an interlaced input.
0x12	CP_INTERLACED	0000**0**000	R	A readback to indicate the interlaced status of the CP core based on the configuration of the video standard and interlaced bit in the CP register map. 0 = CP core is processing the input as a progressive input. 1 = CP core is processing the input as an interlaced input.
0x12	CP_PROG_PARM_FOR_INT	00000**0**00	R	A readback to indicate whether the CP core is processing for the progressive standard, whereas the video standard and the interlaced bit in the CP register map are configured for an interlaced standard. 0 = CP core processing for a progressive standard, whereas the video standard and the interlaced bit are configured for an interlaced standard. 1 = CP core processing for a progressive standard, whereas the video standard and the interlaced bit are configured for an progressive standard
0x12	CP_FORCE_INTERLACED	000000**0**0	R	A readback to indicate forced interlaced status of the CP core based on the configuration of the video standard and interlaced bit in the CP register map. 0 = input is detected as interlaced, and the CP is programmed in an interlaced mode via VID_STD[5:0]. 1 = input is detected as progressive, and the CP is programmed in interlaced mode.

Address	Name	Bits	Access	Description
0x12	CP_NON_STD_VIDEO	0000000**0**	R	A control to indicate that the CP core has detected a nonstandard number of lines on the incoming video compared to the standard specified by VID_STD[5:0]. 0 = input has the same number of lines as that of the format programmed. 1 = input has a different number of lines to that of the format programmed.
0x15	TRI_AUDIO	101**1**1110	R/W	A control to tristate the audio output interface pins, AP0 to AP5. 0 = audio output pins active. 1 = tristate audio output pins.
0x20	HPA_MAN_VALUE_A	**1**1110000	R/W	A manual control for the value of HPA_A on Port A. Only valid if HPA_MANUAL is set to 1. 0 = 0 V applied to the HPA_A pin. 1 = high level applied to the HPA_A pin.
0x20	HPA_TRISTATE_A	1111**0**000	R/W	Tristate the HPA_A output pin for Port A. 0 = HPA_A pin active. 1 = tristate HPA_A pin.
0x21	HPA_STATUS_PORT_A	0000**0**000	R	Readback of HPA_A status for Port A. 0 = 5 V not applied to the HPA_A pin by the chip. 1 = 5 V applied to the HPA_A pin by the chip.
0x3F	INTRQ_RAW	000000**0**0	R	Status of the interrupt signal on the INT interrupt pin. If an interrupt event that has been enabled for the INT pin occurs, this bit is set to 1. Interrupts for INT are set via the Interrupt 1 mask bits. This bit remains set to 1 until the status for all interrupts enabled on INT are cleared. 0 = no interrupt on INT. 1 = an interrupt event for INT has occurred.
0x40	INTRQ_DUR_SEL[1:0]	**00**100000	R/W	A control to select the interrupt signal duration for the interrupt signal on INT. 00 = 4 XTAL periods. 01 = 16 XTAL periods. 10 = 64 XTAL periods. 11 = active until cleared.
0x40	STORE_UNMASKED_IRQS	001**0**0000	R/W	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 an 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. 0 = does not allow the x_ST flag of any HDMI interrupt to be set independently of mask bits. 1 = allows the x_ST flag of any HDMI interrupt to be set independently of mask bits.
0x40	EN_UMASK_RAW_INTRQ	0010**0**000	R/W	A control to apply the audio mute signal on the INT interrupt pin. 0 = does not output the audio mute signal on INT. 1 = outputs the audio mute signal on INT.
0x40	MPU_STIM_INTRQ	00100**0**00	R/W	Manual interrupt set control. This feature must be used for test purposes only. Note that the appropriate mask bit must be set to generate an interrupt at the pin. 0 = disables manual interrupt mode. 1 = enables manual interrupt mode.

Address	Name	Bits	Access	Description
0x40	INTRQ_OP_SEL[1:0]	00100000	R/W	Interrupt signal configuration control for INT. 00 = open drain. 01 = drives low when active. 10 = drives high when active. 11 = disabled.
0x41	CP_LOCK_UNLOCK_EDGE_SEL	00110000	R/W	A control to configure the functionality of the CP_LOCK and CP_UNLOCK interrupts. 0 = generates an interrupt for a low to high change in the CP_LOCK and CP_UNLOCK statuses for Channel 1. 1 = generates an interrupt for a low to high or a high to low change in the CP_LOCK and CP_UNLOCK statuses for Channel 1.
0x41	STDI_DATA_VALID_EDGE_SEL	00110000	R/W	A control to configure the functionality of the STDI_DATA_VALID interrupt. The interrupt can be generated when the STDI changes to an STDI valid state. Alternatively, it can be generated to indicate a change in the STDI_VALID status. 0 = generate interrupt for a low to high change in STDI_DATA_VALID status. 1 = generate interrupt for a low to high or a high to low change in STDI_DATA_VALID status.
0x42	STDI_DATA_VALID_RAW	00000000	R	The STDI_DATA_VALID interrupt can be either an edge sensitive or level sensitive interrupt, depending on the configuration of the STDI_DATA_VALID_EDGE_SEL bit. When STDI_DATA_VALID_EDGE_SEL is set to 1, it is a level sensitive interrupt and STDI_DATA_VALID_RAW is the raw signal status of the STDI data valid signal. When STDI_DATA_VALID_EDGE_SEL is set to 0, it is an edge sensitive interrupt and STDI_DATA_VALID_RAW is a sampled status of the STDI data valid signal following a change in the signal. When set, this bit remains high until it is cleared via STDI_DATA_VALID_CLR. 0 = STDI data is not valid. 1 = STDI data is valid.
0x42	CP_UNLOCK_RAW	00000000	R	Status of the CP_UNLOCK interrupt signal. When set to 1, it indicates a change in the unlock status of the CP core. When set, this bit remains high until it is cleared via CP_UNLOCK_CLR. 0 = CP is locked. 1 = CP is unlocked.
0x42	CP_LOCK_RAW	00000000	R	Status of the CP_LOCK interrupt signal. When set to 1, it indicates a change in the lock status of the CP core. When set, this bit remains high until it is cleared via CP_LOCK_CLR. 0 = CP is unlocked. 1 = CP is locked.
0x43	STDI_DATA_VALID_ST	00000000	R	Latched signal status of STDI valid interrupt signal. When set, this bit remains high until the interrupt is cleared via STDI_DATA_VALID_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no STDI valid interrupt has occurred. 1 = a STDI valid interrupt has occurred.
0x43	CP_UNLOCK_ST	00000000	R	Latched signal status of CP unlock interrupt signal. When set, this bit remains high until the interrupt is cleared via CP_UNLOCK_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no CP UNLOCK interrupt event has occurred. 1 = a CP UNLOCK interrupt event has occurred.

Address	Name	Bits	Access	Description
0x43	CP_LOCK_ST	00000**0**00	R	Latched signal status of the CP lock interrupt signal. When set, this bit remains high until the interrupt is cleared via CP_LOCK_ CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no CP LOCK interrupt event has occurred. 1 = a CP LOCK interrupt event has occurred.
0x44	STDI_DATA_VALID_CLR	000**0**0000	SC	Clear bit for the STDI data valid interrupt signal. 0 = does not clear the STDI_DATA_VALID_ST bit. 1 = clears the STDI_DATA_VALID_ST bit.
0x44	CP_UNLOCK_CLR	0000**0**000	SC	Clear bit for CP unlock interrupt signal. 0 = does not clear the CP_UNLOCK_ST bit. 1 = clears the CP_UNLOCK_ST bit.
0x44	CP_LOCK_CLR	00000**0**00	SC	Clear bit for CP Lock interrupt signal. 0 = does not clear the CP_LOCK_ST bit. 1 = clears the CP_LOCK_ST bit.
0x46	STDI_DATA_VALID_MB1	000**0**0000	R/W	INT interrupt mask for the STDI data valid interrupt. When set, the STDI data valid interrupt triggers the INT interrupt and STDI_DATA_ VALID_ST indicates the interrupt status. 0 = disables the STDI data valid interrupt for INT. 1 = enables the STDI data valid interrupt for INT.
0x46	CP_UNLOCK_MB1	0000**0**000	R/W	INT interrupt mask for CP Unlock interrupt. When set, the CP unlock interrupt triggers the INT interrupt and CP_UNLOCK_ST indicates the interrupt status. 0 = disable CP Unlock interrupt for INT. 1 = enable CP Unlock interrupt for INT.
0x46	CP_LOCK_MB1	00000**0**00	R/W	INT interrupt mask for the CP lock interrupt. When set, the CP lock interrupt triggers the INT interrupt and CP_LOCK_ST indicates the interrupt status. 0 = disable CP Lock interrupt for INT. 1 = enable CP Lock interrupt for INT.
0x47	MPU_STIM_INTRQ_RAW	**0**0000000	R	Raw status of the manual forced interrupt signal. 0 = manual forced interrupt is not applied. 1 = manual forced interrupt is applied.
0x48	MPU_STIM_INTRQ_ST	**0**0000000	R	Latched signal status of the manual forced interrupt signal. When set, this bit remains high until the interrupt is cleared via MPU_STIM_INTRQ_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = forced manual interrupt event has not occurred. 1 = force manual interrupt even has occurred.
0x49	MPU_STIM_INTRQ_CLR	**0**0000000	SC	Clear bit for manual forced interrupt signal. 0 = does not clear the MPU_STIM_INTRQ_ST bit. 1 = clears the MPU_STIM_INTRQ_ST bit.
0x4B	MPU_STIM_INTRQ_MB1	**0**0000000	R/W	INT interrupt mask for the manual forced interrupt signal. When set, the manual forced interrupt triggers the INT interrupt and MPU_ STIM_INTRQ_ST indicates the interrupt status. 0 = disables the manual forced interrupt for INT. 1 = enables the manual forced interrupt for INT.
0x5B	CP_LOCK_CH1_RAW	0000**0**000	R	0 = no change. 1 = Channel 1 input has changed from an unlocked state to a locked state.
0x5B	CP_UNLOCK_CH1_RAW	00000**0**00	R	0 = no change. 1 = Channel 1 CP input has changed from a locked state to an unlocked state.

Address	Name	Bits	Access	Description
0x5B	STDI_DVALID_CH1_RAW	000000**0**0	R	Raw status of the STDI data valid for Sync Channel 1 signal. 0 = STDI data is not valid for Sync Channel 1. 1 = STDI data is valid for Sync Channel 1.
0x5C	CP_LOCK_CH1_ST	0000**0**000	R	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.
0x5C	CP_UNLOCK_CH1_ST	00000**0**00	R	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.
0x5C	STDI_DVALID_CH1_ST	000000**0**0	R	Latched signal status of the STDI valid for Sync Channel 1 interrupt signal. When set, this bit remains high until the interrupt is cleared via STDI_DVALID_CH1_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no STDI valid for Sync Channel 1 interrupt has occurred. 1 = an STDI valid for Sync Channel 1 interrupt has occurred.
0x5D	CP_LOCK_CH1_CLR	0000**0**000	SC	0 = does not clear CP_LOCK_CH1_ST. 1 = clears CP_LOCK_CH1_ST.
0x5D	CP_UNLOCK_CH1_CLR	00000**0**00	SC	0 = does not clear CP_UNLOCK_CH1_ST. 1 = clears CP_UNLOCK_CH1_ST.
0x5D	STDI_DVALID_CH1_CLR	000000**0**0	SC	Clear bit for STDI data valid on Sync Channel 1 interrupt signal. 0 = does not clear STDI_DVALID_CH1_ST. 1 = clears STDI_DVALID_CH1_ST.
0x5F	CP_LOCK_CH1_MB1	0000**0**000	R/W	0 = masks CP_LOCK_CH1_ST. 1 = unmasks CP_LOCK_CH1_ST.
0x5F	CP_UNLOCK_CH1_MB1	00000**0**00	R/W	0 = masks CP_UNLOCK_CH1_ST. 1 = unmasks CP_UNLOCK_CH1_ST.
0x5F	STDI_DVALID_CH1_MB1	000000**0**0	R/W	INT interrupt mask for STDI data valid for sync Channel 1 interrupt. When set, the STDI data valid for Sync Channel 1 interrupt triggers the INT interrupt and STDI_DVALID_CH1_ST indicates the interrupt status. 0 = disables the STDI data valid for the Sync Channel 1 interrupt for INT. 1 = enables the STDI data valid for the Sync Channel 1 interrupt for INT.
0x60	ISRC2_PCKT_RAW	**0**0000000	R	Raw status signal of International Standard Recording Code 2 (ISRC2) packet detection signal. 0 = no ISRC2 packets received since the last HDMI packet detection reset. 1 = ISRC2 packets have been received. This bit resets to zero after an HDMI packet detection reset or upon writing to ISRC2_PACKET_ID[7:0].
0x60	ISRC1_PCKT_RAW	0**0**000000	R	Raw status signal of International Standard Recording Code 1 (ISRC1) packet detection signal. 0 = no ISRC1 packets received since the last HDMI packet detection reset. 1 = ISRC1 packets have been received. This bit resets to zero after an HDMI packet detection reset or upon writing to ISRC1_PACKET_ID[7:0].
0x60	ACP_PCKT_RAW	00**0**00000	R	Raw status signal of audio content protection (ACP) packet detection signal. 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 resets to zero after an HDMI packet detection reset or upon writing to ACP_PACKET_ID[7:0].

Address	Name	Bits	Access	Description
0x60	VS_INFO_RAW	000**0**0000	R	Raw status signal of vendor specific InfoFrame detection signal. 0 = no new vendor specific InfoFrame has been received since the last HDMI packet detection reset. 1 = a new vendor specific InfoFrame has been received. This bit resets to zero after an HDMI packet detection reset or upon writing to VS_PACKET_ID.
0x60	MS_INFO_RAW	0000**0**000	R	Raw status signal of MPEG source InfoFrame detection signal. 0 = no MPEG source InfoFrame received since the last HDMI packet detection reset. 1 = MPEG source InfoFrame received. This bit resets to zero after an HDMI packet detection reset or upon writing to MS_PACKET_ID[7:0].
0x60	SPD_INFO_RAW	00000**0**00	R	Raw status of source product descriptor (SPD) InfoFrame detected signal. 0 = no SPD InfoFrame received since the last HDMI packet detection reset. 1 = SPD InfoFrame received. This bit resets to zero after an HDMI packet detection reset or upon writing to SPD_PACKET_ID.
0x60	AUDIO_INFO_RAW	000000**0**0	R	Raw status of audio InfoFrame detected signal. 0 = no audio InfoFrame has been received within the last three vertical sync (VS) signals or since the last HDMI packet detection reset. 1 = an audio InfoFrame has been received within the last three VS signals. This bit resets to zero on the fourth VS leading edge following an audio InfoFrame, after an HDMI packet detection reset or upon writing to AUD_PACKET_ID.
0x60	AVI_INFO_RAW	0000000**0**	R	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 VS signals (on the eighth VS leading edge following the last received AVI InfoFrame), after an HDMI packet detection reset or upon writing to AVI_PACKET_ID. 0 = no AVI InfoFrame has been received within the last seven VS signals or since the last HDMI packet detection reset. 1 = an AVI InfoFrame has been received within the last seven VS signals.
0x61	ISRC2_PCKT_ST	**0**0000000	R	Latched status of ISRC2 packet detected interrupt signal. When set, this bit remains high until the interrupt is cleared via ISRC2_ INFO_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no interrupt generated from this register. 1 = ISRC2_PCKT_RAW has changed. An interrupt has been generated.
0x61	ISRC1_PCKT_ST	0**0**000000	R	Latched status of ISRC1 packet detected interrupt signal. When set, this bit remains high until the interrupt is cleared via ISRC1_ INFO_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no interrupt generated from this register. 1 = ISRC1_PCKT_RAW has changed. Interrupt has been generated.
0x61	ACP_PCKT_ST	00**0**00000	R	Latched status of the ACP packet detected interrupt signal. When set, this bit remains high until the interrupt is cleared via ACP_PCKT _CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no interrupt generated from this register. 1 = ACP_PCKT_RAW has changed. An interrupt has been generated.

Address	Name	Bits	Access	Description
0x61	VS_INFO_ST	000**0**0000	R	Latched status of the vendor specific InfoFrame detected interrupt signal. When set, this bit remains high until the interrupt is cleared via VS_INFO_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no interrupt generated from this register. 1 = VS_INFO_RAW has changed. An interrupt has been generated.
0x61	MS_INFO_ST	0000**0**000	R	Latched status of the MPEG source InfoFrame detected interrupt signal. When set, this bit remains high until the interrupt is cleared via MS_INFO_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no interrupt generated from this register. 1 = MS_INFO_RAW has changed. An interrupt has been generated.
0x61	SPD_INFO_ST	00000**0**00	R	Latched status of the SPD InfoFrame detected interrupt signal. When set, this bit remains high until the interrupt is cleared via SPD_ INFO_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no interrupt generated from this register. 1 = SPD_INFO_RAW has changed. An interrupt has been generated.
0x61	AUDIO_INFO_ST	000000**0**0	R	Latched status of the audio InfoFrame detected interrupt signal. When set, this bit remains high until the interrupt is cleared via AUDIO_INFO_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no interrupt generated from this register. 1 = AUDIO_INFO_RAW has changed. Interrupt has been generated.
0x61	AVI_INFO_ST	0000000**0**	R	Latched status of the AVI_INFO_RAW signal. This bit is only valid if enabled via the corresponding INT interrupt mask bit. When set, this bit remains high until the interrupt is cleared via AVI_INFO_CLR. 0 = AVI_INFO_RAW has not changed state. 1 = AVI_INFO_RAW has changed state.
0x62	ISRC2_PCKT_CLR	**0**0000000	SC	Clear bit for ISRC2 packet detection interrupt signal. 0 = does not clear ISRC2_PCKT_ST. 1 = clears ISRC2_PCKT_ST.
0x62	ISRC1_PCKT_CLR	0**0**000000	SC	Clear bit for the ISRC1 packet detection interrupt signal. 0 = does not clear ISRC1_PCKT_ST. 1 = clears ISRC1_PCKT_ST.
0x62	ACP_PCKT_CLR	00**0**00000	SC	Clear bit for the ACP packet detected interrupt signal. 0 = does not clear ACP_PCKT_ST. 1 = clears ACP_PCKT_ST.
0x62	VS_INFO_CLR	000**0**0000	SC	Clear bit for the vendor specific InfoFrame interrupt signal. 0 = does not clear VS_INFO_ST. 1 = clears VS_INFO_ST.
0x62	MS_INFO_CLR	0000**0**000	SC	Clear bit for the MPEG source InfoFrame interrupt signal. 0 = does not clear MS_INFO_ST. 1 = clears MS_INFO_ST.
0x62	SPD_INFO_CLR	00000**0**00	SC	Clear bit for the SPD InfoFrame interrupt signal. 0 = does not clear SPD_INFO_ST. 1 = clears SPD_INFO_ST.
0x62	AUDIO_INFO_CLR	000000**0**0	SC	Clear bit for the audio InfoFrame interrupt signal. 0 = does not clear AUDIO_INFO_ST. 1 = clears AUDIO_INFO_ST.

Address	Name	Bits	Access	Description
0x62	AVI_INFO_CLR	0000000**0**	SC	Clear bit for the AVI_INFO_RAW and AVI_INFO_ST bits. 0 = no function. 1 = clear AVI_INFO_RAW and AVI_INFO_ST.
0x64	ISRC2_PCKT_MB1	**0**0000000	R/W	INT interrupt mask for the ISRC2 InfoFrame detection interrupt. When set, the ISRC2 InfoFrame detection interrupt triggers the INT interrupt and ISRC2_PCKT_ST indicates the interrupt status. 0 = disables the ISRC2 packet detection interrupt for INT. 1 = enables the ISRC2 packet detection interrupt for INT.
0x64	ISRC1_PCKT_MB1	0**0**000000	R/W	INT interrupt mask for ISRC1 InfoFrame detection interrupt. When set, the ISRC1 InfoFrame detection interrupt triggers the INT interrupt and ISRC1_INFO_ST indicates the interrupt status. 0 = disables the ISRC1 InfoFrame detection interrupt for INT. 1 = enables the ISRC1 InfoFrame detection interrupt for INT.
0x64	ACP_PCKT_MB1	00**0**00000	R/W	INT interrupt mask for the ACP packet detection interrupt. When set, the ACP packet detection interrupt triggers the INT interrupt and ACP_INFO_ST indicates the interrupt status. 0 = disables the ACP InfoFrame detection interrupt for INT. 1 = enables the ACP InfoFrame detection interrupt for INT.
0x64	VS_INFO_MB1	000**0**0000	R/W	INT interrupt mask for the vendor specific InfoFrame detection interrupt. When set, the vendor specific InfoFrame detection interrupt triggers the INT interrupt and VS_INFO_ST indicates the interrupt status. 0 = disables the vendor specific InfoFrame detection interrupt for INT. 1 = enables the vendor specific InfoFrame detection interrupt for INT.
0x64	MS_INFO_MB1	0000**0**000	R/W	INT interrupt mask for the MPEG source InfoFrame detection interrupt. When set, the MPEG source InfoFrame detection interrupt triggers the INT interrupt and MS_INFO_ST indicates the interrupt status. 0 = disables the MPEG source InfoFrame detection interrupt for INT. 1 = enables the MPEG source InfoFrame detection interrupt for INT.
0x64	SPD_INFO_MB1	00000**0**00	R/W	INT interrupt mask for the SPD InfoFrame detection interrupt. When set, the SPD InfoFrame detection interrupt triggers the INT interrupt and SPD_INFO_ST indicates the interrupt status. 0 = disables the SPD InfoFrame detection interrupt for INT. 1 = enables the SPD InfoFrame detection interrupt for INT.
0x64	AUDIO_INFO_MB1	000000**0**0	R/W	INT interrupt mask for the audio InfoFrame detection interrupt. When set, the audio InfoFrame detection interrupt triggers the INT interrupt and AVI_INFO_ST indicates the interrupt status. 0 = disables the audio InfoFrame detection interrupt for INT. 1 = enables the audio InfoFrame detection interrupt for INT.
0x64	AVI_INFO_MB1	0000000**0**	R/W	INT interrupt mask for the AVI InfoFrame detection interrupt. When set, an AVI InfoFrame detection event causes AVI_INFO_ST to be set and an interrupt is generated on INT. 0 = disables the AVI InfoFrame detection interrupt for INT. 1 = enables the AVI InfoFrame detection interrupt for INT.
0x65	CS_DATA_VALID_RAW	**0**0000000	R	Raw status signal of the channel status data valid signal. 0 = channel status data is not valid. 1 = channel status data is valid.
0x65	INTERNAL_MUTE_RAW	0**0**000000	R	Raw status signal of the internal mute signal. 0 = audio is not muted. 1 = audio is muted.

Address	Name	Bits	Access	Description
0x65	AV_MUTE_RAW	00**0**00000	R	Raw status signal of the AV mute detection signal. 0 = no AV mute raw received since the last HDMI reset condition. 1 = AV mute received.
0x65	AUDIO_CH_MD_RAW	000**0**0000	R	Raw status signal indicating the layout value of the audio packets that were last received. 0 = the last audio packets received have a layout value of 1 (for example, 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 (for example, Layout 0 corresponds to 8-channel audio when audio sample packets are received).
0x65	HDMI_MODE_RAW	0000**0**000	R	Raw status signal of the HDMI mode signal. 0 = DVI is being received. 1 = HDMI is being received.
0x65	GEN_CTL_PCKT_RAW	00000**0**00	R	Raw status signal of the general control packet detection signal. 0 = no general control packets received since the last HDMI reset condition. 1 = general control packets received.
0x65	AUDIO_C_PCKT_RAW	000000**0**0	R	Raw status signal of the audio clock regeneration packet detection signal. 0 = no audio clock regeneration packets received since the last HDMI reset condition. 1 = audio clock regeneration packets received.
0x65	GAMUT_MDATA_RAW	0000000**0**	R	Raw status signal of gamut metadata packet detection signal. 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 resets to zero after an HDMI packet detection reset or upon writing to GAMUT_PACKET_ID[7:0].
0x66	CS_DATA_VALID_ST	**0**0000000	R	Latched status of the channel status data valid interrupt signal. When set, this bit remains high until the interrupt is cleared via ICS_ DATA_VALID_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 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.
0x66	INTERNAL_MUTE_ST	0**0**000000	R	Latched status of the internal mute interrupt signal. When set, this bit remains high until the interrupt is cleared via INTERNAL_ MUTE_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = INTERNAL_MUTE_RAW has not changed. An interrupt has not been generated. 1 = INTERNAL_MUTE_RAW has changed. An interrupt has been generated.
0x66	AV_MUTE_ST	00**0**00000	R	Latched status of the AV mute detected interrupt signal. When set, this bit remains high until the interrupt is cleared via AV_MUTE_ CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = AV_MUTE_RAW has not changed. An interrupt has not been generated. 1 = AV_MUTE_RAW has changed. An interrupt has been generated.

Address	Name	Bits	Access	Description
0x66	AUDIO_CH_MD_ST	000**0**0000	R	Latched status of the audio channel mode interrupt signal. When set, this bit remains high until the interrupt is cleared via AUDIO_CH_ MD_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 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.
0x66	HDMI_MODE_ST	0000**0**000	R	Latched status of the HDMI mode interrupt signal. When set, this bit remains high until the interrupt is cleared via HDMI_ MODE_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = HDMI_MODE_RAW has not changed. An interrupt has not been generated. 1 = HDMI_MODE_RAW has changed. An interrupt has been generated.
0x66	GEN_CTL_PCKT_ST	00000**0**00	R	Latched status of general control packet interrupt signal. When set, this bit remains high until the interrupt is cleared via GEN_CTL_ PCKT_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = GEN_CTL_PCKT_RAW has not changed. An interrupt has not been generated from this register. 1 = GEN_CTL_PCKT_RAW has changed. An interrupt has been generated from this register.
0x66	AUDIO_C_PCKT_ST	000000**0**0	R	Latched status of the audio clock regeneration packet interrupt signal. When set, this bit remains high until the interrupt is cleared via AUDIO_C_PCKT_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = AUDIO_C_PCKT_RAW has not changed. An interrupt has not been generated from this register. 1 = AUDIO_C_PCKT_RAW has changed. An interrupt has been generated from this register.
0x66	GAMUT_MDATA_ST	0000000**0**	R	Latched status of the gamut metadata packet detected interrupt signal. When set, this bit remains high until the interrupt is cleared via GAMUT_MDATA_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = GAMUT_MDATA_RAW has not changed. An interrupt has not been generated from this register. 1 = GAMUT_MDATA_RAW has changed. An interrupt has been generated from this register.
0x67	CS_DATA_VALID_CLR	**0**0000000	SC	Clear bit for the channel status data valid interrupt signal. 0 = does not clear CS_DATA_VALID_ST. 1 = clears CS_DATA_VALID_ST.
0x67	INTERNAL_MUTE_CLR	0**0**000000	SC	Clear bit for the internal mute interrupt signal. 0 = does not clear INTERNAL_MUTE_ST. 1 = clears INTERNAL_MUTE_ST.
0x67	AV_MUTE_CLR	00**0**00000	SC	Clear bit for the AV mute detected interrupt signal. 0 = does not clear AV_MUTE_ST. 1 = clears AV_MUTE_ST.
0x67	AUDIO_CH_MD_CLR	000**0**0000	SC	Clear bit for the audio channel mode interrupt signal. 0 = does not clear AUDIO_CH_MD_ST. 1 = clears AUDIO_CH_MD_ST.
0x67	HDMI_MODE_CLR	0000**0**000	SC	Clear bit for the HDMI mode interrupt signal. 0 = does not clear HDMI_MODE_ST. 1 = clears HDMI_MODE_ST.

Address	Name	Bits	Access	Description
0x67	GEN_CTL_PCKT_CLR	00000**0**00	SC	Clear bit for the general control packet detection interrupt signal. 0 = does not clear GEN_CTL_PCKT_ST. 1 = clears GEN_CTL_PCKT_ST.
0x67	AUDIO_C_PCKT_CLR	000000**0**0	SC	Clear bit for the audio clock regeneration packet detection interrupt signal. 0 = does not clear AUDIO_C_PCKT_ST. 1 = clears AUDIO_C_PCKT_ST.
0x67	GAMUT_MDATA_CLR	0000000**0**	SC	Clear bit for the gamut metadata packet detection interrupt signal. 0 = does not clear GAMUT_MDATA_ST. 1 = clears GAMUT_MDATA_ST.
0x69	CS_DATA_VALID_MB1	**0**0000000	R/W	INT interrupt mask for the channel status data valid interrupt. When set, the channel status data valid interrupt triggers the INT interrupt and CS_DATA_VALID_ST indicates the interrupt status. 0 = disables the channel status data valid interrupt for INT. 1 = enables the channel status data valid interrupt for INT.
0x69	INTERNAL_MUTE_MB1	0**0**000000	R/W	INT interrupt mask for the internal mute interrupt. When set, the internal mute interrupt triggers the INT interrupt and INTERNAL_ MUTE_ST indicates the interrupt status. 0 = disables the AV mute detected interrupt for INT. 1 = enables the AV mute detected interrupt for INT.
0x69	AV_MUTE_MB1	00**0**00000	R/W	INT interrupt mask for the AV mute detected interrupt. When set, the AV mute detected interrupt triggers the INT interrupt and AV_MUTE_ST indicates the interrupt status. 0 = disables the AV mute detected interrupt for INT. 1 = enables the AV mute detected interrupt for INT.
0x69	AUDIO_CH_MD_MB1	000**0**0000	R/W	INT interrupt mask for the audio channel mode interrupt. When set, the audio channel mode interrupt triggers the INT interrupt and AUDIO_CH_MD_ST indicates the interrupt status. 0 = disables the audio channel mode interrupt for INT. 1 = enables the audio channel mode interrupt for INT.
0x69	HDMI_MODE_MB1	0000**0**000	R/W	INT interrupt mask for the HDMI mode detection interrupt. When set, the HDMI mode interrupt triggers the INT interrupt and HDMI_MODE_ST indicates the interrupt status. 0 = disables the HDMI mode interrupt for INT. 1 = enables the HDMI mode interrupt for INT.
0x69	GEN_CTL_PCKT_MB1	00000**0**00	R/W	INT interrupt mask for the general control packet detection interrupt. When set, the general control packet detection interrupt triggers the INT interrupt and GEN_CTL_PCKT_ST indicates the interrupt status. 0 = disables the general control packet detection interrupt for INT. 1 = enables the general control packet detection interrupt for INT.
0x69	AUDIO_C_PCKT_MB1	000000**0**0	R/W	INT interrupt mask for the audio clock regeneration packet detection interrupt. When set, the audio clock regeneration packet detection interrupt triggers the INT interrupt and AUDIO_C_PCKT_ST indicates the interrupt status. 0 = disables the audio clock regeneration packet detection interrupt for INT. 1 = enables the audio clock regeneration packet detection interrupt for INT.

Address	Name	Bits	Access	Description
0x69	GAMUT_MDATA_MB1	0000000**0**	R/W	INT interrupt mask for the gamut metadata packet detection interrupt. When set, the gamut metadata packet detection interrupt triggers the INT interrupt and GAMUT_MDATA_PCKT_ST indicates the interrupt status. 0 = disables the gamut metadata packet detection interrupt for INT. 1 = enables the gamut metadata packet detection interrupt for INT.
0x6A	TMDSPLL_LCK_A_RAW	0**0**000000	R	A readback to indicate the raw status of the Port A TMDS PLL lock signal. 0 = TMDS PLL on Port A is not locked. 1 = TMDS PLL on Port A is locked to the incoming clock.
0x6A	TMDS_CLK_A_RAW	000**0**0000	R	Raw status of the Port A TMDS clock detection signal. 0 = no TMDS clock detected on Port A. 1 = TMDS clock detected on Port A.
0x6A	VIDEO_3D_RAW	00000**0**00	R	Raw status of the video 3D signal. 0 = video 3D not detected. 1 = video 3D detected.
0x6A	V_LOCKED_RAW	000000**0**0	R	Raw status of the VS filter locked signal. 0 = the VS filter has not locked and VS parameters are not valid. 1 = the VS filter has locked and VS parameters are valid.
0x6A	DE_REGEN_LCK_RAW	0000000**0**	R	Raw status of the DE regeneration lock signal. 0 = DE regeneration block has not been locked. 1 = DE regeneration block has been locked to the incoming DE signal.
0x6B	TMDSPLL_LCK_A_ST	0**0**000000	R	Latched status of the Port A TMDS PLL lock interrupt signal. When set, this bit remains high until the interrupt is cleared via TMDSPLL_LCK_A_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 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.
0x6B	TMDS_CLK_A_ST	000**0**0000	R	Latched status of the Port A TMDS clock detection interrupt signal. When set, this bit remains high until the interrupt is cleared via TMDS_CLK_A_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 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.
0x6B	VIDEO_3D_ST	00000**0**00	R	Latched status for the video 3D interrupt. When set, this bit remains high until the interrupt is cleared via VIDEO_3D_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = VIDEO_3D_RAW has not changed. An interrupt has not been generated. 1 = VIDEO_3D_RAW has changed. An interrupt has been generated.
0x6B	V_LOCKED_ST	000000**0**0	R	Latched status for the VS filter locked interrupt. When set, this bit remains high until the interrupt is cleared via V_LOCKED_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = V_LOCKED_RAW has not changed. An interrupt has not been generated. 1 = V_LOCKED_RAW has changed. An interrupt has been generated.

Address	Name	Bits	Access	Description
0x6B	DE_REGEN_LCK_ST	0000000**0**	R	Latched status for the DE regeneration lock interrupt signal. When set, this bit remains high until the interrupt is cleared via DE_REGEN_LCK_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 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.
0x6C	TMDSPLL_LCK_A_CLR	0**0**000000	SC	Clear bit for the Port A TMDS PLL lock interrupt signal. 0 = does not clear TMDSPLL_LCK_A_ST. 1 = clears TMDSPLL_LCK_A_ST.
0x6C	TMDS_CLK_A_CLR	000**0**0000	SC	Clear bit for the Port A TMDS clock detection interrupt signal. 0 = does not clear TMDS_CLK_A_ST. 1 = clears TMDS_CLK_A_ST.
0x6C	VIDEO_3D_CLR	00000**0**00	SC	Clear bit for the video 3D interrupt. 0 = does not clear VIDEO_3D_ST. 1 = clears VIDEO_3D_ST.
0x6C	V_LOCKED_CLR	000000**0**0	SC	Clear bit for the VS filter locked interrupt. 0 = does not clear V_LOCKED_ST. 1 = clears V_LOCKED_ST.
0x6C	DE_REGEN_LCK_CLR	0000000**0**	SC	Clear bit for the DE regeneration lock interrupt signal. 0 = does not clear DE_REGEN_LCK_ST. 1 = clears DE_REGEN_LCK_ST.
0x6E	TMDSPLL_LCK_A_MB1	0**0**000000	R/W	INT interrupt mask for the Port A TMDS PLL lock interrupt. When set, the Port A TMDS PLL lock interrupt triggers the INT interrupt and TMDSPLL_LCK_A_ST indicates the interrupt status. 0 = disables the Port A TMDSPLL lock interrupt for INT. 1 = enables the Port A TMDSPLL lock interrupt for INT.
0x6E	TMDS_CLK_A_MB1	000**0**0000	R/W	INT interrupt mask for the Port A TMDS clock detection interrupt. When set, the Port A TMDS clock detection interrupt triggers the INT interrupt and TMDS_CLK_A_ST indicates the interrupt status. 0 = disables the Port A TMDS clock detection interrupt for INT. 1 = enables the Port A TMDS clock detection interrupt for INT.
0x6E	VIDEO_3D_MB1	00000**0**00	R/W	INT interrupt mask for the video 3D interrupt. When set, the video 3D interrupt triggers the INT interrupt and VIDEO_3D_ST indicates the interrupt status. 0 = disables the video 3D interrupt on INT. 1 = enables the video 3D interrupt on INT.
0x6E	V_LOCKED_MB1	000000**0**0	R/W	INT interrupt mask for the VS filter locked interrupt. When set, the VS filter locked interrupt triggers the INT interrupt and V_LOCKED_ST indicates the interrupt status. 0 = disables the VS filter locked interrupt on INT. 1 = enables the VS filter locked interrupt on INT.
0x6E	DE_REGEN_LCK_MB1	0000000**0**	R/W	INT interrupt mask for the DE regeneration lock interrupt. When set, the DE regeneration lock interrupt triggers the INT interrupt and DE_REGEN_LCK_ST indicates the interrupt status. 0 = disables the DE regeneration lock interrupt on INT. 1 = enables the DE regeneration lock interrupt on INT.
0x6F	HDMI_ENCRPT_A_RAW	00000**0**00	R	Raw status of the Port A encryption detection signal. 0 = current frame in Port A is not encrypted. 1 = current frame in Port A is encrypted.

Address	Name	Bits	Access	Description
0x6F	CABLE_DET_A_RAW	0000000**0**	R	Raw status of the Port A 5 V cable detection signal. 0 = no cable detected on Port A. 1 = cable detected on Port A (high level on RXA_5V).
0x70	HDMI_ENCRPT_A_ST	00000**0**00	R	Latched status for the Port A encryption detection interrupt signal. When set, this bit remains high until the interrupt is cleared via HDMI_ENCRPT_A_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 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.
0x70	CABLE_DET_A_ST	0000000**0**	R	Latched status for the Port A 5 V cable detection interrupt signal. When set, this bit remains high until the interrupt is cleared via CABLE_DET_A_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = CABLE_DET_A_RAW has not changed. An interrupt has not been generated from this register. 1 = CABLE_DET_A_RAW has changed. An interrupt has been generated from this register.
0x71	HDMI_ENCRPT_A_CLR	00000**0**00	SC	Clear bit for the Port A encryption detection interrupt signal. 0 = does not clear HDMI_ENCRPT_A_ST. 1 = clears HDMI_ENCRPT_A_ST.
0x71	CABLE_DET_A_CLR	0000000**0**	SC	Clear bit for the Port A 5 V cable detection interrupt signal. 0 = does not clear CABLE_DET_A_ST. 1 = clears CABLE_DET_A_ST.
0x73	HDMI_ENCRPT_A_MB1	00000**0**00	R/W	INT interrupt mask for the Port A encryption detection interrupt. When set, the Port A encryption detection interrupt triggers the INT interrupt and HDMI_ENCRPT_A_ST indicates the interrupt status. 0 = disables the Port A HDMI encryption detection interrupt for INT. 1 = enables the Port A HDMI encryption detection interrupt for INT.
0x73	CABLE_DET_A_MB1	0000000**0**	R/W	INT interrupt mask for the Port A 5 V cable detection interrupt. When set, the Port A 5 V cable detection interrupt triggers the INT interrupt and CABLE_DET_A_ST indicates the interrupt status. 0 = disables the Port A 5 V cable detection interrupt for INT. 1 = enables the Port A 5 V cable detection interrupt for INT.
0x79	NEW_ISRC2_PCKT_RAW	**0**0000000	R	Status of the new ISRC2 interrupt signal. When set to 1, it indicates that an ISRC2 packet has been received with new content. When set, this bit remains high until it is cleared via NEW_ISRC2_PCKT_CLR. 0 = ISRC2 packet with no new content received. 1 = ISRC2 packet with new content received.
0x79	NEW_ISRC1_PCKT_RAW	0**0**000000	R	Status of the new ISRC1 interrupt signal. When set to 1, it indicates that an ISRC1 packet has been received with new content. When set, this bit remains high until it is cleared via NEW_ISRC1_PCKT_CLR. 0 = ISRC1 packet with no new content received. 1 = ISRC1 packet with new content received.
0x79	NEW_ACP_PCKT_RAW	00**0**00000	R	Status of the new ACP packet interrupt signal. When set to 1, it indicates that an ACP packet has been received with new content. When set, this bit remains high until it is cleared via NEW_ACP_PCKT_CLR. 0 = ACP packet with no new content received. 1 = ACP packet with new content received.

Address	Name	Bits	Access	Description
0x79	NEW_VS_INFO_RAW	000**0**0000	R	Status of the new vendor specific InfoFrame interrupt signal. When set to 1, it indicates that a vendor specific InfoFrame has been received with new content. When set, this bit remains high until it is cleared via NEW_VS_INFO_CLR. 0 = vendor specific packet with no new content received. 1 = vendor specific packet with new content received.
0x79	NEW_MS_INFO_RAW	0000**0**000	R	Status of the new MPEG source InfoFrame interrupt signal. When set to 1, it indicates that an MPEG source InfoFrame has been received with new content. When set, this bit remains high until it is cleared via NEW_MS_INFO_CLR. 0 = MPEG source InfoFrame with no new content received. 1 = MPEG source InfoFrame with new content received.
0x79	NEW_SPD_INFO_RAW	00000**0**00	R	Status of the new SPD packet interrupt signal. When set to 1, it indicates that a SPD packet has been received with new content. When set, this bit remains high until it is cleared via NEW_SPD_ INFO_CLR. 0 = SPD InfoFrame with no new content received. 1 = SPD InfoFrame with new content received.
0x79	NEW_AUDIO_INFO_RAW	000000**0**0	R	Status of the new audio InfoFrame interrupt signal. When set to 1, it indicates that an audio InfoFrame has been received with new content. When set, this bit remains high until it is cleared via NEW_AUDIO_INFO_CLR. 0 = no new audio InfoFrame received. 1 = audio InfoFrame with new content received.
0x79	NEW_AVI_INFO_RAW	0000000**0**	R	Status of the new AVI InfoFrame interrupt signal. When set to 1, it indicates that an AVI InfoFrame has been received with new content. When set, this bit remains high until the interrupt is cleared via NEW_AVI_INFO_CLR. 0 = AVI InfoFrame with no new content received. 1 = AVI InfoFrame with new content received.
0x7A	NEW_ISRC1_PCKT_ST	0**0**000000	R	Latched status for the new ISRC1 packet interrupt. When set, this bit remains high until the interrupt is cleared via NEW_ISRC1_PCKT_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = ISRC1 packet with no new content received. An interrupt has not been generated. 1 = ISRC1 packet with new content received. An interrupt has been generated.
0x7A	NEW_ACP_PCKT_ST	00**0**00000	R	Latched status for the new ACP packet interrupt. When set, this bit remains high until the interrupt is cleared via NEW_ACP_ PCKT_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = ACP packet received with no new content. An interrupt has not been generated. 1 = ACP packet with new content received. An interrupt has been generated.
0x7A	NEW_VS_INFO_ST	000**0**0000	R	Latched status for the new vendor specific InfoFrame interrupt. When set, this bit remains high until the interrupt is cleared via NEW_VS_INFO_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = vendor specific packet with no new content received. An interrupt has not been generated. 1 = vendor specific packet with new content received. An interrupt has been generated.

Address	Name	Bits	Access	Description
0x7A	NEW_MS_INFO_ST	0000**0**000	R	Latched status for the new MPEG source InfoFrame interrupt. When set, this bit remains high until the interrupt is cleared via NEW_MS_INFO_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = MPEG source InfoFrame with no new content received. An interrupt has not been generated. 1 = MPEG source InfoFrame with new content received. An interrupt has been generated.
0x7A	NEW_SPD_INFO_ST	00000**0**00	R	Latched status for the new SPD InfoFrame interrupt. When set, this bit remains high until the interrupt is cleared via NEW_SPD_INFO_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = SPD InfoFrame with no new content received. An interrupt has not been generated. 1 = SPD InfoFrame with new content received. An interrupt has been generated.
0x7A	NEW_AUDIO_INFO_ST	000000**0**0	R	Latched status for the new audio InfoFrame interrupt. When set, this bit remains high until the interrupt is cleared via NEW_AUDIO_INFO_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = audio InfoFrame with no new content received. An interrupt has not been generated. 1 = audio InfoFrame with new content received. An interrupt has been generated.
0x7A	NEW_AVI_INFO_ST	0000000**0**	R	Latched status for the NEW_AVI_INFO_RAW. This bit is only valid if enabled via the corresponding INT interrupt mask bit. When set, this bit remains high until the interrupt is cleared via NEW_AVI_ INFO_CLR. 0 = NEW_AVI_INFO_RAW has not changed state. 1 = NEW_AVI_INFO_RAW has changed state.
0x7B	NEW_ISRC2_PCKT_CLR	**0**0000000	SC	Clear bit for NEW_ISRC2_PCKT_RAW and NEW_ISRC2_PCKT_ST bits. 0 = no function. 1 = clear NEW_ISRC2_PCKT_RAW and NEW_ISRC2_PCKT_ST.
0x7B	NEW_ISRC1_PCKT_CLR	0**0**000000	SC	Clear bit for NEW_ISRC1_PCKT_RAW and NEW_ISCR1_PCKT_ST bits. 0 = no function. 1 = clear NEW_ISRC1_PCKT_RAW and NEW_ISRC1_PCKT_ST.
0x7B	NEW_ACP_PCKT_CLR	00**0**00000	SC	Clear bit for NEW_ACP_PCKT_RAW and NEW_ACP_PCKT_ST bits. 0 = no function. 1 = clear NEW_ACP_PCKT_RAW and NEW_ACP_PCKT_ST.
0x7B	NEW_VS_INFO_CLR	000**0**0000	SC	Clear bit for NEW_VS_INFO_RAW and NEW_VS_INFO_ST bits. 0 = no function. 1 = clear NEW_VS_INFO_RAW and NEW_VS_INFO_ST.
0x7B	NEW_MS_INFO_CLR	0000**0**000	SC	Clear bit for NEW_MS_INFO_RAW and NEW_MS_INFO_ST bits. 0 = no function. 1 = clear NEW_MS_INFO_RAW and NEW_MS_INFO_ST.
0x7B	NEW_SPD_INFO_CLR	00000**0**00	SC	Clear bit for NEW_SPD_INFO_RAW and NEW_SPD_INFO_ST bits. 0 = no function. 1 = clear NEW_SPD_INFO_RAW and NEW_SPD_INFO_ST.
0x7B	NEW_AUDIO_INFO_CLR	000000**0**0	SC	Clear bit for NEW_AUDIO_INFO_RAW and NEW_AUDIO_INFO_ST bits. 0 = no function. 1 = clear NEW_AUDIO_INFO_RAW and NEW_AUDIO_INFO_ST.

Address	Name	Bits	Access	Description
0x7B	NEW_AVI_INFO_CLR	0000000**0**	SC	Clear bit for NEW_AVI_INFO_RAW and NEW_AVI_INFO_ST bits. 0 = no function. 1 = clear NEW_AVI_INFO_RAW and NEW_AVI_INFO_ST.
0x7D	NEW_ISRC2_PCKT_MB1	**0**0000000	R/W	INT interrupt mask for new ISRC2 packet interrupt. When set, the new ISRC2 interrupt triggers the INT interrupt and NEW_ISRC2_ST indicates the interrupt status. 0 = disables the new ISRC2 packet interrupt for INT. 1 = enables the new ISRC2 packet interrupt for INT.
0x7D	NEW_ISRC1_PCKT_MB1	0**0**000000	R/W	INT interrupt mask for the new ISRC1 packet interrupt. When set, the new ISRC2 interrupt triggers the INT interrupt and NEW_ISRC1_PCKT_ST indicates the interrupt status. 0 = disables the new ISRC1 packet interrupt for INT. 1 = enables the new ISRC1 packet interrupt for INT.
0x7D	NEW_ACP_PCKT_MB1	00**0**00000	R/W	INT interrupt mask for the new ACP packet interrupt. When set, the new ACP interrupt triggers the INT interrupt and NEW_ACP_PCKT_ST indicates the interrupt status. 0 = disables the new ACP packet interrupt for INT. 1 = enables the new ACP packet interrupt for INT.
0x7D	NEW_VS_INFO_MB1	000**0**0000	R/W	INT interrupt mask for the new vendor specific InfoFrame interrupt. When set, the new vendor specific InfoFrame interrupt triggers the INT interrupt and NEW_VS_INFO_ST indicates the interrupt status. 0 = disables the new vendor specific InfoFrame interrupt for INT. 1 = enables the new vendor specific InfoFrame interrupt for INT.
0x7D	NEW_MS_INFO_MB1	0000**0**000	R/W	INT interrupt mask for the new MPEG source InfoFrame interrupt. When set, the new MPEG source InfoFrame interrupt triggers the INT interrupt and NEW_SPD_INFO_ST indicates the interrupt status. 0 = disables the new MS InfoFrame interrupt for INT. 1 = enables the new MS InfoFrame interrupt for INT.
0x7D	NEW_SPD_INFO_MB1	00000**0**00	R/W	INT interrupt mask for the new SPD InfoFrame interrupt. When set, the new SPD InfoFrame interrupt triggers the INT interrupt and NEW_ SPD_INFO_ST indicates the interrupt status. 0 = disables the new SPD InfoFrame interrupt for INT. 1 = enables the new SPD InfoFrame interrupt for INT.
0x7D	NEW_AUDIO_INFO_MB1	000000**0**0	R/W	INT interrupt mask for the new audio InfoFrame interrupt. When set, the new audio InfoFrame interrupt triggers the INT interrupt and NEW_AUDIO_INFO_ST indicates the interrupt status. 0 = disables the new audio InfoFrame interrupt for INT. 1 = enables the new audio InfoFrame interrupt for INT.
0x7D	NEW_AVI_INFO_MB1	0000000**0**	R/W	INT interrupt mask for new AVI InfoFrame detection interrupt. When set, a new AVI InfoFrame detection event causes NEW_AVI_ INFO_ST to be set and an interrupt is generated on INT. 0 = disable new AVI InfoFrame interrupt for INT. 1 = enable new AVI InfoFrame interrupt for INT.

Address	Name	Bits	Access	Description
0x7E	FIFO_NEAR_OVFL_RAW	**0**0000000	R	Status of the audio FIFO near overflow interrupt signal. When set to 1, it indicates the audio FIFO is near overflow when the number of FIFO registers containing stereo data is greater or equal to the value set in AUDIO_FIFO_ALMOST_FULL_THRESHOLD[5:0]. When set, this bit remains high until it is cleared via FIFO_NEAR_OVFL_CLR. 0 = the audio FIFO has not reached the high threshold defined in AUDIO_FIFO_ALMOST_FULL_THRESHOLD[5:0]. 1 = the audio FIFO has reached the high threshold defined in AUDIO_FIFO_ALMOST_FULL_THRESHOLD[5:0].
0x7E	FIFO_UNDERFLO_RAW	0**0**000000	R	Status of the 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. When set, this bit remains high until it is cleared via FIFO_UNDERFLO_CLR. 0 = the audio FIFO has not underflowed. 1 = the audio FIFO has underflowed.
0x7E	FIFO_OVERFLO_RAW	00**0**00000	R	Status of the 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. When set, this bit remains high until it is cleared via FIFO_OVERFLO_CLR. 0 = the audio FIFO has not overflowed. 1 = the audio FIFO has overflowed.
0x7E	CTS_PASS_THRSH_RAW	000**0**0000	R	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. When set, this bit remains high until it is cleared via CTS_PASS_THRSH_CLR. 0 = the audio clock regeneration CTS value has not passed the threshold. 1 = the audio clock regeneration CTS value has changed more than threshold.
0x7E	CHANGE_N_RAW	0000**0**000	R	Status of the ACR N value changed interrupt signal. When set to 1, it indicates the N value of the ACR packets has changed. When set, this bit remains high until it is cleared via CHANGE_N_CLR. 0 = the audio clock regeneration N value has not changed. 1 = the audio clock regeneration N value has changed.
0x7E	PACKET_ERROR_RAW	00000**0**00	R	Status of the packet error interrupt signal. When set to 1, it indicates that a packet has been received with an uncorrectable error correction code (ECC) error in either the header or body. When set, this bit remains high until it is cleared via PACKET_ERROR_CLR. 0 = no uncorrectable error detected in packet header. 1 = an uncorrectable error detected in an unknown packet (error in packet header).
0x7E	AUDIO_PCKT_ERR_RAW	000000**0**0	R	Status of the audio packet error interrupt signal. When set to 1, it indicates that an audio packet has been received with an uncorrectable error. When set, this bit remains high until it is cleared via AUDIO_PCKT_ERR_CLR. 0 = no uncorrectable error detected in an audio packet. 1 = an uncorrectable error detected in an audio packet.
0x7E	NEW_GAMUT_MDATA_RAW	0000000**0**	R	Status of the new gamut metadata packet interrupt signal. When set to 1, it indicates that a gamut metadata packet has been received with new content. When set, this bit remains high until it is cleared via NEW_GAMUT_MDATA_CLR. 0 = gamut metadata packet with no new content received or no change has taken place. 1 = gamut metadata packet with new content received that triggered this interrupt.

Address	Name	Bits	Access	Description
0x7F	FIFO_NEAR_OVFL_ST	**0**0000000	R	Latched status for the audio FIFO near overflow interrupt. When set, this bit remains high until the interrupt is cleared via FIFO_OVFL_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = the audio FIFO has not reached the high threshold. 1 = the audio FIFO has reached the high threshold.
0x7F	FIFO_UNDERFLO_ST	0**0**000000	R	Latched status for the audio FIFO underflow interrupt. When set, this bit remains high until the interrupt is cleared via FIFO_ UNDERFLO_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = the audio FIFO has not underflowed. 1 = the audio FIFO has underflowed.
0x7F	FIFO_OVERFLO_ST	00**0**00000	R	Latched status for the audio FIFO overflow interrupt. When set, this bit remains high until the interrupt is cleared via FIFO_ OVERFLO_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = the audio FIFO has not overflowed. 1 = the audio FIFO has overflowed.
0x7F	CTS_PASS_THRSH_ST	000**0**0000	R	Latched status for the ACR CTS value exceed threshold interrupt. When set, this bit remains high until the interrupt is cleared via CTS_PASS_THRSH_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = the audio clock regeneration CTS value has not passed the threshold. 1 = the audio clock regeneration CTS value has passed the threshold.
0x7F	CHANGE_N_ST	0000**0**000	R	Latched status for the ACR N value changed interrupt. When set, this bit remains high until the interrupt is cleared via CHANGE_N_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = the audio clock regeneration N value has not changed. 1 = the audio clock regeneration N value has changed.
0x7F	PACKET_ERROR_ST	00000**0**00	R	Latched status for the packet error interrupt. When set, this bit remains high until the interrupt is cleared via PACKET_ERROR_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 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.
0x7F	AUDIO_PCKT_ERR_ST	000000**0**0	R	Latched status for the audio packet error interrupt. When set, this bit remains high until the interrupt is cleared via AUDIO_PCKT_ERR_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 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.
0x7F	NEW_GAMUT_MDATA_ST	0000000**0**	R	Latched status for the new gamut metadata packet interrupt. When set, this bit remains high until the interrupt is cleared via NEW_GAMUT_MDATA_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 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.

Address	Name	Bits	Access	Description
0x80	FIFO_NEAR_OVFL_CLR	**0**0000000	SC	Clear bit for the audio FIFO near overflow interrupt. 0 = does not clear FIFO_NEAR_OVFL_ST. 1 = clears FIFO_NEAR_OVFL_ST.
0x80	FIFO_UNDERFLO_CLR	0**0**000000	SC	Clear bit for the audio FIFO underflow interrupt. 0 = does not clear FIFO_UNDERFLO_ST. 1 = clears FIFO_UNDERFLO_ST.
0x80	FIFO_OVERFLO_CLR	00**0**00000	SC	Clear bit for the audio FIFO overflow interrupt. 0 = does not clear FIFO_OVERFLO_ST. 1 = clears FIFO_OVERFLO_ST.
0x80	CTS_PASS_THRSH_CLR	000**0**0000	SC	Clear bit for the ACR CTS value exceed threshold interrupt. 0 = does not clear CTS_PASS_THRSH_ST. 1 = clears CTS_PASS_THRSH_ST.
0x80	CHANGE_N_CLR	0000**0**000	SC	Clear bit for the ACR N value changed interrupt. 0 = does not clear CHANGE_N_ST. 1 = clears CHANGE_N_ST.
0x80	PACKET_ERROR_CLR	00000**0**00	SC	Clear bit for the packet error interrupt. 0 = does not clear PACKET_ERROR_ST. 1 = clears PACKET_ERROR_ST.
0x80	AUDIO_PCKT_ERR_CLR	000000**0**0	SC	Clear bit for the audio packet error interrupt. 0 = does not clear AUDIO_PCKT_ERR_ST. 1 = clears AUDIO_PCKT_ERR_ST.
0x80	NEW_GAMUT_MDATA_CLR	0000000**0**	SC	Clear bit for the new gamut metadata packet interrupt. 0 = does not clear NEW_GAMUT_MDATA_ST. 1 = clears NEW_GAMUT_MDATA_ST.
0x82	FIFO_NEAR_OVFL_MB1	**0**0000000	R/W	INT interrupt mask for the audio FIFO near overflow interrupt. When this bit is set, the audio FIFO overflow interrupt triggers the INT interrupt and FIFO_NEAR_OVFL_ST indicates the interrupt status. 0 = disables the audio FIFO overflow interrupt on INT. 1 = enables the audio FIFO overflow interrupt on INT.
0x82	FIFO_UNDERFLO_MB1	0**0**000000	R/W	INT interrupt mask for the audio FIFO underflow interrupt. When this bit is set, the audio FIFO underflow interrupt triggers the INT interrupt and FIFO_UNDERFLO_ST indicates the interrupt status. 0 = disables the audio FIFO underflow interrupt on INT. 1 = enables the audio FIFO underflow interrupt on INT.
0x82	FIFO_OVERFLO_MB1	00**0**00000	R/W	INT interrupt mask for the audio FIFO overflow interrupt. When this bit is set, the audio FIFO overflow interrupt triggers the INT interrupt and FIFO_OVERFLO_ST indicates the interrupt status. 0 = disables the audio FIFO overflow interrupt on INT. 1 = enables the audio FIFO overflow interrupt on INT.
0x82	CTS_PASS_THRSH_MB1	000**0**0000	R/W	INT interrupt mask for the ACR CTS value exceed threshold interrupt. When set, the ACR CTS value exceed threshold interrupt triggers the INT interrupt and CTS_PASS_THRSH_ST indicates the interrupt status. 0 = disables the ACR CTS value exceeded threshold interrupt on INT. 1 = enables the ACR CTS value exceeded threshold interrupt on INT.
0x82	CHANGE_N_MB1	0000**0**000	R/W	INT interrupt mask for the ACR N value changed interrupt. When this bit is set, the ACR N value changed interrupt triggers the INT interrupt and CHANGE_N_ST indicates the interrupt status. 0 = disables the ACR N value changed interrupt for INT. 1 = enables the ACR N value changed interrupt for INT.

Address	Name	Bits	Access	Description
0x82	PACKET_ERROR_MB1	00000**0**00	R/W	INT interrupt mask for the packet error interrupt. When this bit is set, the audio packet error interrupt triggers the INT interrupt and PACKET_ERROR_ST indicates the interrupt status. 0 = disables the packet error interrupt for INT. 1 = enables the packet error interrupt for INT.
0x82	AUDIO_PCKT_ERR_MB1	000000**0**0	R/W	INT interrupt mask for the audio packet error interrupt. When this bit is set, the audio packet error interrupt triggers the INT interrupt and AUDIO_PCKT_ERR_ST indicates the interrupt status. 0 = disables the audio packet error interrupt for INT. 1 = enables the audio packet error interrupt for INT.
0x82	NEW_GAMUT_MDATA_MB1	0000000**0**	R/W	INT interrupt mask for the new gamut metadata packet interrupt. When set, the new gamut metadata packet interrupt triggers the INT interrupt and NEW_GAMUT_MDATA_ST indicates the interrupt status. 0 = disables the new gamut metadata InfoFrame interrupt for INT. 1 = enables the new SPD InfoFrame interrupt for INT.
0x83	DEEP_COLOR_CHNG_RAW	**0**0000000	R	Status of the Deep Color mode changed interrupt signal. When set to 1, it indicates a change in the Deep Color mode has been detected. When set, this bit remains high until it is cleared via DEEP_COLOR_CHNG_CLR. 0 = Deep Color mode has not changed. 1 = change in Deep Color triggered this interrupt.
0x83	VCLK_CHNG_RAW	0**0**000000	R	Status of the video clock changed interrupt signal. When set to 1, it indicates that irregular or missing pulses are detected in the TMDS clock. When set, this bit remains high until it is cleared via VCLK_CHNG_CLR. 0 = no irregular or missing pulse detected in the TMDS clock. 1 = irregular or missing pulses detected in the TMDS clock triggered this interrupt.
0x83	AUDIO_MODE_CHNG_RAW	00**0**00000	R	Status of the 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, Direct Stream Digital® (DSD) packet, high bit rate (HBR) packet, and direct stream transfer (DST) packet. When set, this bit remains high until it is cleared via AUDIO_MODE_CHNG_CLR. 0 = audio mode has not changed. 1 = audio mode has changed.
0x83	PARITY_ERROR_RAW	000**0**0000	R	Status of parity error interrupt signal. When set to 1, it indicates an audio sample packet has been received with parity error. When set, this bit remains high until it is cleared via PARITY_ERROR_CLR. 0 = no parity error detected in an audio packet. 1 = parity error has been detected in an audio packet.
0x83	NEW_SAMP_RT_RAW	0000**0**000	R	Status of new sampling rate interrupt signal. When set to 1, it indicates that audio sampling frequency field in the channel status data has changed. When set, this bit remains high until it is cleared via NEW_SAMP_RT_CLR. 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.

Address	Name	Bits	Access	Description
0x83	AUDIO_FLT_LINE_RAW	00000**0**00	R	Status of the audio flat line interrupt signal. When set to 1, it indicates audio sample packet has been received with the flat line bit set to 1. When set, this bit remains high until it is cleared via AUDIO_FLT_LINE_CLR. 0 = audio sample packet with the flat line bit set has not been received. 1 = audio sample packet with the flat line bit set has been received.
0x83	NEW_TMDS_FRQ_RAW	000000**0**0	R	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]. When set, this bit remains high until it is cleared via NEW_TMDS_FRQ_CLR. 0 = TMDS frequency has not changed by more than tolerance set in FREQTOLERANCE[3:0] in the HDMI register map. 1 = TMDS frequency has changed by more than tolerance set in FREQTOLERANCE[3:0] in the HDMI register map.
0x83	FIFO_NEAR_UFLO_RAW	0000000**0**	R	Status of the audio FIFO near underflow interrupt signal. When set to 1, it indicates the audio FIFO is near underflow when the number of FIFO registers containing stereo data is less or equal to value set in AUDIO_FIFO_ALMOST_EMPTY_THRESHOLD[5:0]. When set, this bit remains high until it is cleared via FIFO_NEAR_UFLO_CLR. 0 = audio FIFO has not reached the low threshold defined in AUDIO_FIFO_ALMOST_EMPTY_THRESHOLD[5:0]. 1 = audio FIFO has reached the low threshold defined in AUDIO_FIFO_ALMOST_EMPTY_THRESHOLD[5:0].
0x84	DEEP_COLOR_CHNG_ST	**0**0000000	R	Latched status of the Deep Color mode change interrupt. When set, this bit remains high until the interrupt is cleared via DEEP_COLOR_CHNG_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = Deep Color mode has not changed. 1 = change in Deep Color has been detected.
0x84	VCLK_CHNG_ST	0**0**000000	R	Latched status of the video clock change interrupt. When set, this bit remains high until the interrupt is cleared via VCLK_CHNG_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no irregular or missing pulse detected in the TMDS clock. 1 = irregular or missing pulses detected in the TMDS clock.
0x84	AUDIO_MODE_CHNG_ST	00**0**00000	R	Latched status of the audio mode change interrupt. When set, this bit remains high until the interrupt is cleared via AUDIO_MODE_CHNG_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = audio mode has not changed. 1 = audio mode has changed. The following are considered audio modes: no audio, pulse code modulation (PCM), DSD, HBR, and DST.
0x84	PARITY_ERROR_ST	000**0**0000	R	Latched status of the parity error interrupt. When set, this bit remains high until the interrupt is cleared via PARITY_ERROR_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no parity error detected in audio packets. 1 = parity error detected in an audio packet.

Address	Name	Bits	Access	Description
0x84	NEW_SAMP_RT_ST	0000**0**000	R	Latched status of the new sampling rate interrupt. When set, this bit remains high until the interrupt is cleared via NEW_SAMP_RT_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = the sampling rate bits of the channel status data on Audio Channel 0 have not changed. 1 = the sampling rate bits of the channel status data on Audio Channel 0 have changed.
0x84	AUDIO_FLT_LINE_ST	00000**0**00	R	Latched status of the audio flat line interrupt. When set, this bit remains high until the interrupt is cleared via AUDIO_FLT_LINE_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = an audio sample packet with the flat line bit set has not been received. 1 = an audio sample packet with the flat line bit set has been received.
0x84	NEW_TMDS_FRQ_ST	000000**0**0	R	Latched status of the new TMDS frequency interrupt. When set, this bit remains high until the interrupt is cleared via NEW_TMDS_FRQ_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = TMDS frequency has not changed by more than tolerance set in FREQTOLERANCE[3:0] in the HDMI register map. 1 = TMDS frequency has changed by more than tolerance set in FREQTOLERANCE[3:0] in the HDMI register map.
0x84	FIFO_NEAR_UFLO_ST	0000000**0**	R	Latched status for the audio FIFO near underflow interrupt. When set, this bit remains high until the interrupt is cleared via FIFO_NEAR_UFLO_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = an audio FIFO has not reached the low threshold. 1 = an audio FIFO has reached the low threshold.
0x85	DEEP_COLOR_CHNG_CLR	**0**0000000	SC	Clear bit for the Deep Color mode change interrupt. 0 = does not clear DEEP_COLOR_CHNG_ST. 1 = clears DEEP_COLOR_CHNG_ST.
0x85	VCLK_CHNG_CLR	0**0**000000	SC	Clear bit for the video clock change interrupt. 0 = does not clear VCLK_CHNG_ST. 1 = clears VCLK_CHNG_ST.
0x85	AUDIO_MODE_CHNG_CLR	00**0**00000	SC	Clear bit for the audio mode change interrupt. 0 = does not clear AUDIO_MODE_CHNG_ST. 1 = clears AUDIO_MODE_CHNG_ST.
0x85	PARITY_ERROR_CLR	000**0**0000	SC	Clear bit for the parity error interrupt. 0 = does not clear PARITY_ERROR_ST. 1 = clears PARITY_ERROR_ST.
0x85	NEW_SAMP_RT_CLR	0000**0**000	SC	Clear bit for the new sample rate interrupt. 0 = does not clear NEW_SAMP_RT_ST. 1 = clears NEW_SAMP_RT_ST.
0x85	AUDIO_FLT_LINE_CLR	00000**0**00	SC	Clear bit for the audio flat line interrupt. 0 = does not clear AUDIO_FLT_LINE_ST. 1 = clears AUDIO_FLT_LINE_ST.
0x85	NEW_TMDS_FRQ_CLR	000000**0**0	SC	Clear bit for the new TMDS frequency interrupt. 0 = does not clear NEW_TMDS_FRQ_ST. 1 = clears NEW_TMDS_FRQ_ST.
0x85	FIFO_NEAR_UFLO_CLR	0000000**0**	SC	Clear bit for the audio FIFO near underflow interrupt. 0 = does not clear FIFO_NEAR_UFLO_ST. 1 = clears FIFO_NEAR_UFLO_ST.

Address	Name	Bits	Access	Description
0x87	DEEP_COLOR_CHNG_MB1	**0**0000000	R/W	INT interrupt mask for the Deep Color mode changed interrupt. When this bit is set, the Deep Color mode changed interrupt triggers the INT interrupt and DEEP_COLOR_CHNG_ST indicates the interrupt status. 0 = disables the Deep Color mode changed interrupt on INT. 1 = enables the Deep Color mode changed interrupt on INT.
0x87	VCLK_CHNG_MB1	0**0**000000	R/W	INT interrupt mask for the video clock changed interrupt. When this bit is set, the video clock changed interrupt triggers the INT interrupt and VCLK_CHNG_ST indicates the interrupt status. 0 = disables the video clock changed interrupt on INT. 1 = enables the video clock changed interrupt on INT.
0x87	AUDIO_MODE_CHNG_MB1	00**0**00000	R/W	INT interrupt mask for the audio mode changed interrupt. When this bit is set, the audio mode changed interrupt triggers the INT interrupt and AUDIO_MODE_CHNG_ST indicates the interrupt status. 0 = disables the audio mode changed interrupt on INT. 1 = enables the audio mode changed interrupt on INT.
0x87	PARITY_ERROR_MB1	000**0**0000	R/W	INT interrupt mask for the parity error interrupt. When this bit is set, the parity error interrupt triggers the INT interrupt and PARITY_ERROR_ST indicates the interrupt status. 0 = disables the parity error interrupt on INT. 1 = enables the parity error interrupt on INT.
0x87	NEW_SAMP_RT_MB1	0000**0**000	R/W	INT interrupt mask for the new sample rate interrupt. When this bit is set, the new sample rate interrupt triggers the INT interrupt and NEW_SAMP_RT_ST indicates the interrupt status. 0 = disables the new sample rate interrupt on INT. 1 = enables the new sample rate interrupt on INT.
0x87	AUDIO_FLT_LINE_MB1	00000**0**00	R/W	INT interrupt mask for the audio flat line interrupt. When this bit is set, the audio flat line interrupt triggers the INT interrupt and AUDIO_FLT_LINE_ST indicates the interrupt status. 0 = disables the audio flat line interrupt on INT. 1 = enables the audio flat line interrupt on INT.
0x87	NEW_TMDS_FRQ_MB1	000000**0**0	R/W	INT interrupt mask for the new TMDS frequency interrupt. When this bit is set, the new TMDS frequency interrupt triggers the INT interrupt and NEW_TMDS_FRQ_ST indicates the interrupt status. 0 = disable new TMDS frequency interrupt on INT. 1 = enable new TMDS frequency interrupt on INT
0x87	FIFO_NEAR_UFLO_MB1	0000000**0**	R/W	INT interrupt mask for the audio FIFO near underflow interrupt. When this bit is set, the audio FIFO near underflow interrupt triggers the INT interrupt and FIFO_NEAR_UFLO_ST indicates the interrupt status. 0 = disable audio FIFO near underflow interrupt on INT. 1 = enable audio FIFO near underflow interrupt on INT.
0x88	MS_INF_CKS_ERR_RAW	**0**0000000	R	Status of the 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. When set, this bit remains high until it is cleared via MS_INF_CKS_ERR_CLR. 0 = no MPEG source InfoFrame checksum error has occurred. 1 = an MPEG source InfoFrame checksum error has occurred.
0x88	SPD_INF_CKS_ERR_RAW	0**0**000000	R	Status of the SPD InfoFrame checksum error interrupt signal. When set to 1, it indicates that a checksum error has been detected for an SPD InfoFrame. When set, this bit remains high until it is cleared via SPD_INF_CKS_ERR_CLR. 0 = no SPD InfoFrame checksum error has occurred. 1 = an SPD InfoFrame checksum error has occurred.

Address	Name	Bits	Access	Description
0x88	AUD_INF_CKS_ERR_RAW	00**0**00000	R	Status of the audio InfoFrame checksum error interrupt signal. When set to 1, it indicates that a checksum error has been detected for an audio InfoFrame. When set, this bit remains high until it is cleared via AUDIO_INF_CKS_ERR_CLR. 0 = no audio InfoFrame checksum error has occurred. 1 = an audio InfoFrame checksum error has occurred.
0x88	AVI_INF_CKS_ERR_RAW	000**0**0000	R	Status of the AVI InfoFrame checksum error interrupt signal. When set to 1, it indicates that a checksum error has been detected for an AVI InfoFrame. When set, this bit remains high until it is cleared via AVI_INF_CKS_ERR_CLR. 0 = no AVI InfoFrame checksum error has occurred. 1 = an AVI InfoFrame checksum error has occurred.
0x88	RI_EXPIRED_A_RAW	00000**0**00	R	Status of the Port A R_I expired interrupt signal. When set to 1, it indicates that HDCP cipher R_I value for Port A expired. When set, this bit remains high until it is cleared via RI_EXPIRED_A_CLR. 0 = no R_I expired on Port A. 1 = R_I expired on Port A.
0x88	AKSV_UPDATE_A_RAW	0000000**0**	R	Status of Port A AKSV update interrupt signal. When set to 1, it indicates that the transmitter (Tx) has written its AKSV into the HDCP registers for Port A. When set, this bit remains high until it is cleared via AKSV_UPDATE_A_CLR. 0 = no AKSV updates on Port A. 1 = detected a write access to the AKSV register on Port A.
0x89	MS_INF_CKS_ERR_ST	**0**0000000	R	Latched status of MPEG source InfoFrame checksum error interrupt. When set, this bit remains high until the interrupt is cleared via MS_INF_CKS_ERR_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no change in MPEG source InfoFrame checksum error. 1 = an MPEG source InfoFrame checksum error has triggered this interrupt.
0x89	SPD_INF_CKS_ERR_ST	0**0**000000	R	Latched status of the SPD InfoFrame checksum error interrupt. When set, this bit remains high until the interrupt is cleared via SPD_INF_CKS_ERR_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no change in the SPD InfoFrame checksum error. 1 = an SPD InfoFrame checksum error has triggered this interrupt.
0x89	AUD_INF_CKS_ERR_ST	00**0**00000	R	Latched status of the audio InfoFrame checksum error interrupt. When set, this bit remains high until the interrupt is cleared via AUDIO_INF_CKS_ERR_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no change in audio InfoFrame checksum error. 1 = an audio InfoFrame checksum error has triggered this interrupt.
0x89	AVI_INF_CKS_ERR_ST	000**0**0000	R	Latched status of the AVI InfoFrame checksum error interrupt. When set, this bit remains high until the interrupt is cleared via AVI_INF_CKS_ERR_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no change in AVI InfoFrame checksum error. 1 = an AVI InfoFrame checksum error has triggered this interrupt.
0x89	RI_EXPIRED_A_ST	00000**0**00	R	Latched status of the Port A R_I expired interrupt. When set, this bit remains high until the interrupt is cleared via RI_EXPIRED_A_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no R_I expired on Port A. 1 = R_I expired on Port A.

Address	Name	Bits	Access	Description
0x89	AKSV_UPDATE_A_ST	0000000**0**	R	Latched status of the Port A AKSV update interrupt. When set, this bit remains high until the interrupt is cleared via AKSV_ UPDATE_A_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no AKSV updates on Port A. 1 = detected a write access to the AKSV register on Port A.
0x8A	MS_INF_CKS_ERR_CLR	**0**0000000	SC	Clear bit for the MPEG source InfoFrame checksum error Interrupt. 0 = does not clear MS_INF_CKS_ERR_ST. 1 = clears MS_INF_CKS_ERR_ST.
0x8A	SPD_INF_CKS_ERR_CLR	0**0**000000	SC	Clear bit for the SPD InfoFrame checksum error interrupt. 0 = does not clear SPD_INF_CKS_ERR_ST. 1 = clears SPD_INF_CKS_ERR_ST.
0x8A	AUD_INF_CKS_ERR_CLR	00**0**00000	SC	Clear bit for the audio InfoFrame checksum error interrupt. 0 = does not clear AUD_INF_CKS_ERR_ST. 1 = clears AUD_INF_CKS_ERR_ST.
0x8A	AVI_INF_CKS_ERR_CLR	000**0**0000	SC	Clear bit for the AVI InfoFrame checksum error interrupt. 0 = does not clear AVI_INF_CKS_ERR_ST. 1 = clears AVI_INF_CKS_ERR_ST.
0x8A	RI_EXPIRED_A_CLR	00000**0**00	SC	Clear bit for the Port A R_I expired interrupt. 0 = does not clear RI_EXPIRED_A_ST. 1 = clears RI_EXPIRED_A_ST.
0x8A	AKSV_UPDATE_A_CLR	0000000**0**	SC	Clear bit for the Port A AKSV Update Interrupt. 0 = does not clear AKSV_UPDATE_A_ST. 1 = clears AKSV_UPDATE_A_ST.
0x8C	MS_INF_CKS_ERR_MB1	**0**0000000	R/W	INT interrupt mask for the MPEG source InfoFrame checksum error interrupt. When set, the MPEG source InfoFrame checksum error interrupt triggers the INT interrupt and MS_INF_CKS_ERR_ST indicates the interrupt status. 0 = disable SPD InfoFrame checksum error interrupt on INT. 1 = enable SPD InfoFrame checksum error interrupt on INT.
0x8C	SPD_INF_CKS_ERR_MB1	0**0**000000	R/W	INT interrupt mask for the SPD InfoFrame checksum error interrupt. When set, the SPD InfoFrame checksum error interrupt triggers the INT interrupt and SPD_INF_CKS_ERR_ST indicates the interrupt status. 0 = disable SPD InfoFrame checksum error interrupt on INT. 1 = enable SPD InfoFrame checksum error interrupt on INT.
0x8C	AUD_INF_CKS_ERR_MB1	00**0**00000	R/W	INT interrupt mask for the audio InfoFrame checksum error interrupt. When set, the audio InfoFrame checksum error interrupt triggers the INT interrupt and AUDIO_INF_CKS_ERR_ST indicates the interrupt status. 0 = disable audio InfoFrame checksum error interrupt on INT. 1 = enable audio InfoFrame checksum error interrupt on INT.
0x8C	AVI_INF_CKS_ERR_MB1	000**0**0000	R/W	INT interrupt mask for the AVI InfoFrame checksum error interrupt. When set, the AVI InfoFrame checksum error interrupt triggers the INT interrupt and AVI_INF_CKS_ERR_ST indicates the interrupt status. 0 = disable AVI InfoFrame checksum error interrupt on INT. 1 = enable AVI InfoFrame checksum error interrupt on INT.
0x8C	RI_EXPIRED_A_MB1	00000**0**00	R/W	INT interrupt mask for the Port A R_I expired interrupt. When set, the Port A R_I expired interrupt triggers the INT interrupt and RI_EXPIRED_A_ST indicates the interrupt status. 0 = disable Port A R_I expired interrupt on INT. 1 = enable Port A R_I expired interrupt on INT.

Address	Name	Bits	Access	Description
0x8C	AKSV_UPDATE_A_MB1	0000000**0**	R/W	INT interrupt mask for the Port A AKSV update interrupt. When set, the Port A AKSV update interrupt triggers the INT interrupt and AKSV_UPDATE_A_ST indicates the interrupt status. 0 = disable Port A AKSV Update interrupt on INT. 1 = enable Port A AKSV Update interrupt on INT.
0x8D	VS_INF_CKS_ERR_RAW	0000000**0**	R	Status of the 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. When set, this bit remains high until it is cleared via VS_INF_CKS_ERR_CLR. 0 = no vendor specific InfoFrame checksum error has occurred. 1 = a vendor specific InfoFrame checksum error has occurred.
0x8E	VS_INF_CKS_ERR_ST	0000000**0**	R	Latched status of the vendor specific InfoFrame checksum error interrupt. When set, this bit remains high until the interrupt is cleared via VS_INF_CKS_ERR_CLR. This bit is only valid if enabled via the corresponding INT interrupt mask bit. 0 = no change in vendor specific InfoFrame checksum error. 1 = a vendor specific InfoFrame checksum error has triggered this interrupt.
0x8F	VS_INF_CKS_ERR_CLR	0000000**0**	SC	Clear bit for the vendor specific InfoFrame checksum error interrupt. 0 = does not clear VS_INF_CKS_ERR_ST. 1 = clears VS_INF_CKS_ERR_ST.
0x91	VS_INF_CKS_ERR_MB1	0000000**0**	R/W	INT interrupt mask for the vendor specific InfoFrame checksum error interrupt. When set, the vendor specific InfoFrame checksum error interrupt triggers the INT interrupt and VS_INF_CKS_ERR_ST indicates the interrupt status. 0 = disable vendor specific checksum error interrupt on INT. 1 = enable vendor specific checksum error interrupt on INT.
0x92	CEC_RX_RDY2_RAW	00**0**00000	R	Raw status of the CEC Rx Buffer 2 ready signal. When set to 1, it indicates that a CEC frame has been received and is waiting to be read in Rx Frame Buffer 2. 0 = no change. 1 = CEC Rx Buffer 2 has received a complete message that is ready to be read by the host.
0x92	CEC_RX_RDY1_RAW	000**0**0000	R	Raw status of the CEC Rx Buffer 1 ready signal. When set to 1, it indicates that a CEC frame has been received and is waiting to be read in Rx Frame Buffer 1. 0 = no change. 1 = CEC Rx Buffer 1 has received a complete message that is ready to be read by the host.
0x92	CEC_RX_RDY0_RAW	0000**0**000	R	Raw status of the CEC Rx Buffer 0 ready signal. When set to 1, it indicates that a CEC frame has been received and is waiting to be read in Rx Frame Buffer 0. 0 = no change. 1 = CEC Rx Buffer 0 has received a complete message that is ready to be read by the host.
0x92	CEC_TX_RETRY_TIMEOUT_RAW	00000**0**00	R	Raw status of the CEC Tx retry timeout signal. 0 = no change. 1 = CEC Tx has retried to send the current message by the number of times specified in CEC_TX_RETRY[2:0] but it was unsuccessful every time.
0x92	CEC_TX_ARBITRATION_LOST_RAW	000000**0**0	R	Raw status of the CEC Tx arbitration lost signal. 0 = no change. 1 = CEC Tx has lost arbitration to another Tx.

Address	Name	Bits	Access	Description
0x92	CEC_TX_READY_RAW	0000000**0**	R	Raw status of the CEC Tx message sent signal. This bit goes high whenever the Tx has sent a message. 0 = no change. 1 = CEC Tx has sent the last outgoing message.
0x93	CEC_RX_RDY2_ST	00**0**00000	R	Latched status of the CEC_RX_RDY2_RAW signal. This bit is only valid if enabled via the corresponding INT interrupt mask bit. When a message has been received in Buffer 2, this bit is set. When set, this bit remains high until the interrupt is cleared via CEC_RX_RDY2_CLR. 0 = no change. 1 = new CEC message received in Buffer 2.
0x93	CEC_RX_RDY1_ST	000**0**0000	R	Latched status of the CEC_RX_RDY1_RAW signal. This bit is only valid if enabled via the corresponding INT interrupt mask bit. When a message has been received in Buffer 1, this bit is set. When set, this bit remains high until the interrupt is cleared via CEC_RX_RDY1_CLR. 0 = no change. 1 = new CEC message received in Buffer 1.
0x93	CEC_RX_RDY0_ST	0000**0**000	R	Latched status of the CEC_RX_RDY0_RAW signal. This bit is only valid if enabled via the corresponding INT interrupt mask bit. When a message has been received in Buffer 0, this bit is set. When set, this bit remains high until the interrupt is cleared via CEC_RX_RDY0_CLR. 0 = no change. 1 = new CEC message received in Buffer 0.
0x93	CEC_TX_RETRY_TIMEOUT_ST	00000**0**00	R	Latched status of the CEC_TX_RETRY_TIMEOUT_RAW signal. This bit is only valid if enabled via the corresponding INT interrupt mask bit. If the CEC Tx fails to send the current message within the number of retry attempts specified by CEC_TX_RETRY[2:0], this bit is set. When set, this bit remains high until the interrupt is cleared via CEC_TX_RETRY_TIMEOUT_CLR. 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[2:0].
0x93	CEC_TX_ARBITRATION_LOST_ST	000000**0**0	R	Latched status of the CEC_TX_ARBITRATION_LOST_RAW signal. This bit is only valid if enabled via the corresponding INT interrupt mask bit. If the CEC Tx loses arbitration while trying to send a message this bit is set. When set, this bit remains high until the interrupt is cleared via CEC_TX_ARBITRATION_LOST_CLR. 0 = no change. 1 = the CEC Tx has lost arbitration to another Tx.
0x93	CEC_TX_READY_ST	0000000**0**	R	Latched status of the CEC_TX_READY_RAW signal. This bit is only valid if enabled via the corresponding INT interrupt mask bit. When the CEC Tx successfully sends the current message this bit is set. When set, this bit remains high until the interrupt is cleared via CEC_TX_READY_CLR. 0 = no change. 1 = message transmitted successfully.
0x94	CEC_RX_RDY2_CLR	00**0**00000	SC	Clear bit for the CEC Rx Buffer 2 ready interrupt. 0 = does not clear CEC_RX_RDY2_ST. 1 = clears CEC_RX_RDY2_ST.
0x94	CEC_RX_RDY1_CLR	000**0**0000	SC	Clear bit for the CEC Rx Buffer 1 ready interrupt. 0 = does not clear CEC_RX_RDY1_ST. 1 = clears CEC_RX_RDY1_ST.
0x94	CEC_RX_RDY0_CLR	0000**0**000	SC	Clear bit for the CEC Rx Buffer 0 ready interrupt. 0 = does not clear CEC_RX_RDY0_ST. 1 = clears CEC_RX_RDY0_ST.

Address	Name	Bits	Access	Description
0x94	CEC_TX_RETRY_TIMEOUT_CLR	00000**0**00	SC	Clear bit for the CEC transmitter retry timeout interrupt. 0 = does not clear CEC_TX_RETRY_TIMEOUT_ST. 1 = clears CEC_TX_RETRY_TIMEOUT_ST
0x94	CEC_TX_ARBITRATION_LOST_CLR	000000**0**0	SC	Clear bit for the CEC transmitter arbitration lost interrupt. 0 = does not clear CEC_TX_ARBITRATION_LOST_ST. 1 = clears CEC_TX_ARBITRATION_LOST_ST.
0x94	CEC_TX_READY_CLR	0000000**0**	SC	Clear bit for the CEC transmitter ready interrupt. 0 = does not clear CEC_TX_READY_ST. 1 = clears CEC_TX_READY_ST.
0x96	CEC_RX_RDY2_MB1	00**0**00000	R/W	INT interrupt mask for the CEC Rx Buffer 2 ready interrupt. When set, the CEC Rx Buffer 2 ready interrupt triggers the INT interrupt and CEC_RX_RDY2_ST indicates the interrupt status. 0 = disables the CEC Rx Buffer 2 ready interrupt on INT. 1 = enables the CEC Rx Buffer 2 ready interrupt on INT.
0x96	CEC_RX_RDY1_MB1	000**0**0000	R/W	INT interrupt mask for the CEC Rx Buffer 2 ready interrupt. When set, the CEC Rx Buffer 2 ready interrupt triggers the INT interrupt and CEC_RX_RDY1_ST indicates the interrupt status. 0 = disables the CEC Rx Buffer 1 ready interrupt on INT. 1 = enables the CEC Rx Buffer 1 ready interrupt on INT.
0x96	CEC_RX_RDY0_MB1	0000**0**000	R/W	INT interrupt mask for the CEC Rx Buffer 0 ready interrupt. When set, the CEC Rx Buffer 0 ready interrupt triggers the INT interrupt and CEC_RX_RDY0_ST indicates the interrupt status. 0 = disables the CEC Rx Buffer 0 ready interrupt on INT. 1 = enables the CEC Rx Buffer 0 ready interrupt on INT.
0x96	CEC_TX_RETRY_TIMEOUT_MB1	00000**0**00	R/W	INT interrupt mask for the CEC transmitter retry timeout interrupt. When set, the CEC transmitter retry timeout interrupt triggers the INT interrupt and CEC_TX_RETRY_TIMEOUT_ST indicates the interrupt status. 0 = disables the CEC transmitter timeout retry interrupt on INT. 1 = enables the CEC transmitter timeout retry interrupt on INT.
0x96	CEC_TX_ARBITRATION_LOST_MB1	000000**0**0	R/W	INT interrupt mask for the CEC transmitter arbitration lost interrupt. When set, the CEC transmitter arbitration lost interrupt triggers the INT interrupt and CEC_TX_ARBIRATION_LOST_ST indicates the interrupt status. 0 = disables the CEC transmitter arbitration lost interrupt on INT. 1 = enables the CEC transmitter arbitration lost interrupt on INT.
0x96	CEC_TX_READY_MB1	0000000**0**	R/W	INT interrupt mask for the CEC transmitter ready interrupt. When set, the CEC transmitter ready interrupt triggers the INT interrupt and CEC_TX_READY_ST indicates the interrupt status. 0 = disables the CEC transmitter ready interrupt on INT. 1 = enables the CEC transmitter ready interrupt on INT.
0x97	CEC_INTERRUPT_BYTE[7:0]	**00000000**	R	One of the eight preprogrammed commands received. 00 = no change. 01 = Opcode 1 received. 02 = Opcode 2 received. 04 = Opcode 3 received. 08 = Opcode 4 received. 10 = Opcode 5 received. 20 = Opcode 6 received. 40 = Opcode 7 received. 80 = Opcode 8 received.
0x98	CEC_INTERRUPT_BYTE_ST[7:0]	**00000000**	R	0 = no change. 1 = one of eight opcodes received.

Address	Name	Bits	Access	Description
0x99	CEC_INTERRUPT_BYTE_CLR[7:0]	**00000000**	SC	0 = does not clear CEC_INTERRUPT_BYTE_ST. 1 = clears CEC_INTERRUPT_BYTE_ST.
0x9A	CEC_INTERRUPT_BYTE_MB2[7:0]	**00000000**	R/W	0 = masks CEC_INTERRUPT_BYTE_ST. 1 = unmasks CEC_INTERRUPT_BYTE_ST.
0x9B	CEC_INTERRUPT_BYTE_MB1[7:0]	**00000000**	R/W	0 = masks CEC_INTERRUPT_BYTE_ST. 1 = unmasks CEC_INTERRUPT_BYTE_ST.
0xE0	DS_WITHOUT_FILTER	**0**0000000	R/W	Disables the chroma filters on Channel B and Channel C while keeping the downsampler functional. 0 = filters and downsamples. 1 = downsamples only (no filtering).
0xE7	DPP_LUMA_HBW_SEL	00**0**00000	R/W	A control to select the DPP luma filter bandwidth for Stage 2 filters. 0 = select low bandwidth (0.44 f_S) higher stop-band attenuation. 1 = select high bandwidth (0.47 f_S) lower stop-band attenuation.
0xE7	DPP_CHROMA_LOW_EN	000**0**0000	R/W	A control to select DPP chroma filter bandwidth for Stage 2 filters. 0 = high bandwidth, sharp transition filter for Channel B/Channel C. 1 = soft filter with minimized ringing for Channel B/Channel C.
0xE9	LVDS_TX_SLAVE_ADDR[6:0]	**0000000**0	R/W	Programmable I²C slave address for the OpenLDI Tx register map.
0xEA, 0xEB	RD_INFO[15:0]	**00000000 00000000**	R	Chip revision code. 0x20C4 = ADV7613.
0xF4	CEC_SLAVE_ADDR[6:0]	**0000000**0	R/W	Programmable I²C slave address for the CEC register map.
0xF5	INFOFRAME_SLAVE_ADDR[6:0]	**0000000**0	R/W	Programmable I²C slave address for the InfoFrame register map.
0xF8	DPLL_SLAVE_ADDR[6:0]	**0000000**0	R/W	Programmable I²C slave address for the DPLL register map.
0xF9	KSV_SLAVE_ADDR[6:0]	**0000000**0	R/W	Programmable I²C slave address for the repeater register map.
0xFA	EDID_SLAVE_ADDR[6:0]	**0000000**0	R/W	Programmable I²C slave address for the EDID register map.
0xFB	HDMI_SLAVE_ADDR[6:0]	**0000000**0	R/W	Programmable I²C slave address for the HDMI register map.
0xFD	CP_SLAVE_ADDR[6:0]	**0000000**0	R/W	Programmable I²C slave address for the CP register map.
0xFF	MAIN_RESET	**0**0000000	SC	Main reset where all I²C registers are reset to their default values. 0 = normal operation. 1 = apply main I²C reset.

DPLL REGISTER MAP BIT DESCRIPTIONS

Table 10. DPLL Register Map Bit Descriptions

Address	Name	Bits	Access	Description
0xB5	MCLK_FS_N[2:0]	00000**001**	R/W	Selects the multiple of 128 f_S used for the MCLK output. 000 = 128 f_S. 001 = 256 f_S. 010 = 384 f_S. 011 = 512 f_S. 100 = 640 f_S. 101 = 768 f_S. 110 = not valid. 111 = not valid.

HDMI REGISTER MAP BIT DESCRIPTIONS

Table 11. HDMI Register Map Bit Descriptions

Address	Name	Bits	Access	Description
0x00	HDCP_A0	**0**0000000	R/W	A control to set the second LSB of the HDCP port I²C address. 0 = I²C address for the HDCP port is 0x74. Used for single-link mode or first receiver in dual-link mode. 1 = I²C address for the HDCP port is 0x76. Used only for a second receiver dual-link mode.
0x00	HDCP_ONLY_MODE	0**0**000000	R/W	A control 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. 0 = normal operation. 1 = HDCP only mode.
0x01	MUX_DSD_OUT	000**0**0000	R/W	An override control for the DSD output. 0 = override by outputting I²S data. 1 = override by outputting DSD/DST data.
0x01	OVR_AUTO_MUX_DSD_OUT	0000**0**000	R/W	DSD/DST override control. In automatic control, the DSD or I²S interface is selected according to the type of packet received. The DSD/DST interface is enabled if the device receives a DSD or DST audio sample packet. The I²S interface is enabled when the device receives audio sample packets or when no packet is received. In manual mode, MUX_DSD_OUT selects the output interface. 0 = automatic DSD/DST output control. 1 = override DSD/DST output control.
0x01	OVR_MUX_HBR	00000**0**00	R/W	A control to select the automatic or manual configuration for HBR outputs. The HBR outputs are automatically encoded as S/PDIF streams. In manual mode, MUX_HBR_OUT selects the audio output interface. 0 = automatic HBR output control. 1 = manual HBR output control.
0x01	MUX_HBR_OUT	000000**0**0	R/W	A control to manually select the audio output interface for HBR data. Valid when OVR_MUX_HBR is set to 1. 0 = override by outputting I²S data. 1 = override by outputting S/PDIF data.
0x01	TERM_AUTO	0000000**0**	R/W	This bit allows the user to select automatic or manual control of clock termination. 0 = disable termination automatic control. 1 = enable termination automatic control.
0x03	I2SOUTMODE[1:0]	0**00**11000	R/W	A control to configure the I²S output interface. 00 = I²S mode. 01 = right justified. 10 = left justified. 11 = raw S/PDIF (IEC 60958) mode.
0x03	I2SBITWIDTH[4:0]	000**11000**	R/W	A control to adjust the bit width for right justified mode on the I²S interface. 00000 = 0 bit. 00001 = 1 bit. 00010 = 2 bits. … 11000 = 24 bits. 11110 = 30 bits. 11111 = 31 bits.

Address	Name	Bits	Access	Description
0x04	AV_MUTE	0**0**000000	R	Readback of the AVMUTE status received in the last general control packet received. 0 = AVMUTE not set. 1 = AVMUTE set.
0x04	HDCP_KEYS_READ	00**0**00000	R	A readback to indicate a 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, that is, the HDCP keys and the KSV from the internal ROM are retrieved. 0 = HDCP keys and/or KSV has not yet been read. 1 = HDCP keys and/or KSV HDCP keys has been read.
0x04	HDCP_KEY_ERROR	000**0**0000	R	A readback to indicate whether 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. 0 = no error occurred while reading HDCP keys. 1 = HDCP keys read error.
0x04	HDCP_RI_EXPIRED	0000**0**000	R	Reads back high when a calculated R_I has not been read by the source TX on the active port. It remains high until the next AKSV update.
0x04	TMDS_PLL_LOCKED	000000**0**0	R	A readback to indicate whether the TMDS PLL is locked to the TMDS clock input to the selected HDMI port. 0 = the TMDS PLL is not locked. 1 = the TMDS PLL is locked to the TMDS clock input to the selected HDMI port.
0x04	AUDIO_PLL_LOCKED	0000000**0**	R	A readback to indicate the audio digital phase-locked loop (DPLL) lock status. 0 = the audio DPLL is not locked. 1 = the audio DPLL is locked.
0x05	HDMI_MODE	**0**0000000	R	A readback to indicate whether the stream processed by the HDMI core is a DVI or an HDMI stream. 0 = DVI mode detected. 1 = HDMI mode detected.
0x05	HDMI_CONTENT_ENCRYPTED	0**0**000000	R	A readback to indicate the use of HDCP encryption. 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.
0x05	DVI_HSYNC_POLARITY	00**0**00000	R	A readback to indicate the polarity of the horizontal sync (HS) signal encoded in the input stream. 0 = the HS signal is active low. 1 = the HS signal is active high.
0x05	DVI_VSYNC_POLARITY	000**0**0000	R	A readback to indicate the polarity of the VS signal encoded in the input stream. 0 = the VS signal is active low. 1 = the VS signal is active high.
0x05	HDMI_PIXEL_REPETITION[3:0]	0000**0000**	R	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. 0000 = 1×. 0001 = 2×. 0010 = 3×. 0011 = 4×. 0100 = 5×. 0101 = 6×. 0110 = 7×. 0111 = 8×.

Address	Name	Bits	Access	Description
				1000 = 9×. 1001 = 10×. 1010 to 1111 = reserved.
0x07	VERT_FILTER_LOCKED	**0**0000000	R	Vertical filter lock status. Indicates whether the vertical filter is locked and vertical synchronization parameter measurements are valid for readback. 0 = vertical filter has not locked. 1 = vertical filter has locked.
0x07	AUDIO_CHANNEL_MODE	0**0**000000	R	Flags stereo or multichannel audio packets. Stereo packets may carry compressed multichannel audio. 0 = stereo audio (may be compressed multichannel). 1 = multichannel uncompressed audio detected (three to eight channels).
0x07	DE_REGEN_FILTER_LOCKED	00**0**00000	R	DE regeneration filter lock status. Indicates that the DE regeneration section has locked to the received DE signal and horizontal synchronization parameter measurements are valid for readback. 0 = DE regeneration not locked. 1 = DE regeneration locked to the incoming DE signal.
0x07, 0x08	LINE_WIDTH[12:0]	000**00000 00000000**	R	Line width is a horizontal synchronization measurement. This gives the number of active pixels in a line. This measurement is only valid when the DE regeneration filter is locked. XXXXXXXXXXX = total number of active pixels per line.
0x09, 0x0A	FIELD0_HEIGHT[12:0]	000**00000 00000000**	R	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. XXXXXXXXXXXX = the number of active lines in Field 0.
0x0B	DEEP_COLOR_MODE[1:0]	**00**000000	R	A readback of the Deep Color mode information extracted from the general control packet. 00 = 8 bits per channel. 01 = reserved. 10 = reserved. 11 = reserved.
0x0B	HDMI_INTERLACED	00**0**00000	R	HDMI input interlace status, a vertical filter measurement. 0 = progressive input. 1 = interlaced input
0x0B, 0x0C	FIELD1_HEIGHT[12:0]	000**00000 00000000**	R	Field 1 height is a vertical filter measurement. This readback gives the number of active lines in a 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. XXXXXXXXXXXX = the number of active lines in Field 1.
0x0D	FREQTOLERANCE[3:0]	0000**0100**	R/W	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. 0100 = default tolerance in MHz for new TMDS frequency detection. XXXX = tolerance in MHz for new TMDS frequency detection.
0x0F	MAN_AUDIO_DL_BYPASS	**0**0011111	R/W	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. 0 = the 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 the automatic bypass of the audio delay line. Audio delay line is applied depending on the AUDIO_DELAY_LINE_BYPASS control.

Address	Name	Bits	Access	Description
0x0F	AUDIO_DELAY_LINE_BYPASS	0**0**011111	R/W	Manual bypass control for the audio delay line. Only valid when MAN_AUDIO_DL_BYPASS is set to 1. 0 = enables the audio delay line. 1 = bypasses the audio delay line.
0x0F	AUDIO_MUTE_SPEED[4:0]	000**11111**	R/W	Number of samples between each volume change of 1.5 dB when muting and unmuting.
0x10	CTS_CHANGE_THRESHOLD[5:0]	00**100101**	R/W	Sets the tolerance for change in the CTS value. This tolerance is used for the audio mute mask, MT_MSK_NEW_CTS, the HDMI status bit, CTS_PASS_THRSH_RAW, and the HDMI interrupt status bit, CTS_PASS_THRSH_ST. This register controls the number of LSBs that the CTS can change before an audio mute, status change, or an interrupt is triggered. XXXXXX = tolerance of CTS value for CTS_PASS_THRSH_RAW and MT_MSK_NEW_CTS.
0x11	AUDIO_FIFO_ALMOST_FULL_THRESHOLD[6:0]	0**1111101**	R/W	Sets the threshold used for FIFO_NEAR_OVRFL_RAW. The FIFO_NEAR_OVRFL_ST interrupt is triggered when the audio FIFO reaches this level. 1111101 = default value.
0x12	AUDIO_FIFO_ALMOST_EMPTY_THRESHOLD[6:0]	0**0000010**	R/W	Sets the threshold used for FIFO_NEAR_UFLO_RAW. The FIFO_NEAR_UFLO_ST interrupt is triggered if audio FIFO goes below this level. 0000010 = default value.
0x13	AC_MSK_VCLK_CHNG	0**1**111111	R/W	Audio coast mask for a TMDS clock change. When this bit is set, the audio DPLL coasts if the TMDS clock has any irregular/missing pulses. 1 = audio DPLL coasts if the TMDS clock has any irregular/missing pulses. 0 = audio DPLL does not coast if the TMDS clock has any irregular/missing pulses.
0x13	AC_MSK_VPLL_UNLOCK	01**1**11111	R/W	Audio coast mask for the TMDS PLL unlock. When this bit is set, the audio DPLL coasts if the TMDS PLL unlocks. 1 = audio DPLL coasts if TMDS PLL unlocks. 0 = audio DPLL does not coast if TMDS PLL unlocks.
0x13	AC_MSK_NEW_CTS	0111**1**111	R/W	Audio coast mask for a new ACR CTS value. When this bit is set, the audio DPLL coasts if CTS changes by more than the threshold defined in CTS_CHANGE_THRESHOLD[5:0]. 1 = audio DPLL coasts if CTS changes by more than the threshold set in register CTS_CHANGE_THRESHOLD[5:0]. 0 = audio DPLL does not coast if CTS changes by more than the threshold set in register CTS_CHANGE_THRESHOLD[5:0].
0x13	AC_MSK_NEW_N	01111**1**11	R/W	Audio coast mask for a new ACR N value. When this bit is set, the audio DPLL coasts if the N value changes. 1 = audio DPLL coasts if a change in the N value occurs. 0 = audio DPLL does not coast if a change in the N value occurs.
0x13	AC_MSK_CHNG_PORT	011111**1**1	R/W	Audio coast mask for a HDMI port change. When this bit is set, the audio DPLL coasts if a change in the active port occurs. 1 = audio DPLL coasts if the active port is changed. 0 = audio DPLL does not coast if the active port is changed.
0x13	AC_MSK_VCLK_DET	0111111**1**	R/W	Audio coast mask for a TMDS clock detection. This bit sets the audio PLL to coast if no TMDS clock is detected on the active port. 1 = audio DPLL coasts if a TMDS clock is not detected on the active port. 0 = audio DPLL does not coast if a TMDS clock is not detected on the active port.
0x14	MT_MSK_COMPRS_AUD	00**1**11111	R/W	Audio mute mask for compressed audio. This bit sets the audio to mute if the audio received is in a compressed format. 1 = audio mute occurs if audio is received in compressed format.

Address	Name	Bits	Access	Description
0x14	MT_MSK_AUD_MODE_CHNG	001**1**1111	R/W	Audio mute mask for an audio mode change. When this bit is set, the audio mutes if the audio changes between any of the following formats: PCM, DSD, HBR, and DST. 1 = audio mute occurs if an audio changes between any of the following formats: PCM, DSD, HBR, and DST.
0x14	MT_MSK_PARITY_ERR	001111**1**1	R/W	Audio mute mask for a parity error. When this bit is set, the audio mutes if an audio sample packet is received with an incorrect parity bit. 1 = audio mute occurs if an audio sample packet is received with an incorrect parity bit.
0x14	MT_MSK_VCLK_CHNG	0011111**1**	R/W	Audio mute mask for a TMDS clock change. When this bit is set, the audio mutes if the TMDS clock has irregular/missing pulses. 1 = audio mute occurs if the TMDS clock has irregular/missing pulses.
0x15	MT_MSK_APLL_UNLOCK	**1**1111111	R/W	Audio mute mask for an audio PLL unlock. When this bit is set, the audio mutes if the audio PLL unlocks. 1 = audio mute occurs if the audio PLL unlocks.
0x15	MT_MSK_VPLL_UNLOCK	1**1**111111	R/W	Audio mute mask for a TMDS PLL unlock. When this bit is set, the audio mutes if the TMDS PLL unlocks. 1 = audio mute occurs if the TMDS PLL unlocks.
0x15	MT_MSK_ACR_NOT_DET	11**1**11111	R/W	Audio mute mask for an ACR packet. When this bit is set, the audio mutes if an ACR packet has not been received within one VS signal. 1 = audio mute occurs if an ACR packet has not been received within one VS signal.
0x15	MT_MSK_FLATLINE_DET	1111**1**111	R/W	Audio mute mask for a flat line bit. When this bit is set, the audio mutes if an audio packet is received with the flat line bit set. 1 = audio mute occurs if an audio packet is received with the flat line bit set.
0x15	MT_MSK_FIFO_UNDERLFOW	111111**1**1	R/W	Audio mute mask—FIFO underflow.
0x15	MT_MSK_FIFO_OVERFLOW	1111111**1**	R/W	Audio mute mask—FIFO overflow.
0x16	MT_MSK_AVMUTE	**1**1111111	R/W	Audio mute mask for AVMUTE. When this bit is set, the audio mutes if a general control packet is received with the SET_AVMUTE bit set. 1 = audio mute occurs if AVMUTE is set by a general control packet.
0x16	MT_MSK_NOT_HDMIMODE	1**1**111111	R/W	Audio mute mask for a nonHDMI input stream. When this bit is set, the audio mutes if the HDMI_MODE bit goes low. 1 = audio mute occurs if HDMI mode bit goes low.
0x16	MT_MSK_NEW_CTS	11**1**11111	R/W	Audio mute mask for a change of ACR CTS. When this bit is set, the audio mutes if the CTS changes by more than the specified threshold. CTS_CHANGE_THRESHOLD register sets this threshold. 1 = audio mute occurs if the CTS changes.
0x16	MT_MSK_NEW_N	111**1**1111	R/W	Audio mute mask for a new ACR N. When set, the audio mutes if there is a change in the N value. 1 = audio mute occurs if the N value changes
0x16	MT_MSK_CHMODE_CHNG	1111**1**111	R/W	Audio mute mask for an audio channel mode change. When this bit is set, the audio mutes if the channel mode changes between stereo and multichannel. 1 = audio mute occurs if the channel mode changes.
0x16	MT_MSK_APCKT_ECC_ERR	11111**1**11	R/W	Audio mute mask for audio packet ECC error. When this bit is set, the audio mutes if an uncorrectable error is detected in the audio packet by the ECC block. 1 = audio mute occurs if an uncorrectable error is detected in the audio packet by the ECC block.

Address	Name	Bits	Access	Description
0x16	MT_MSK_CHNG_PORT	111111**1**1	R/W	Audio mute mask for an HDMI port change. When this bit is set, the audio mutes if HDMI port selection is changed. 1 = audio mute occurs if HDMI port selection is changed.
0x16	MT_MSK_VCLK_DET	1111111**1**	R/W	Audio mute mask for a TMDS clock. When this bit is set, the audio mutes if a TMDS clock is not detected. 1 = audio mute occurs if a TMDS clock is not detected.
0x18	HBR_AUDIO_PCKT_DET	0000**0**000	R	HBR packet detection bit. This bit resets to zero on the 11th HS leading edge following an HBR packet if a subsequent HBR packet has not been detected. It also resets if an audio, DSD, or DST packet sample packet has been received or after an HDMI reset condition. 0 = no HBR audio packet received within the last 10 HS signals. 1 = HBR audio packet received within the last 10 HS signals.
0x18	DST_AUDIO_PCKT_DET	00000**0**00	R	DST audio packet detection bit. This bit resets to zero on the 11th HS leading edge following a DST packet if a subsequent DST has not been received. It also resets if an audio, DSD, or HBR packet sample packet has been received or after an HDMI reset condition. 0 = no DST packet received within the last 10 HS signals. 1 = DST packet received within the last 10 HS signals.
0x18	DSD_PACKET_DET	000000**0**0	R	DSD audio packet detection bit. This bit resets to zero on the 11th HS leading edge following a DSD packet or if an audio, DST, or HBR packet sample packet has been received or after an HDMI reset condition. 0 = no DSD packet received within the last 10 HS signals. 1 = DSD packet received within the last 10 HS signals.
0x18	AUDIO_SAMPLE_PCKT_DET	0000000**0**	R	Audio sample packet detection bit. This bit resets to zero on the 11th HS leading edge following an audio packet if a subsequent audio sample packet has not been received or if a DSD, DST, or HBR audio packet sample packet has been received. 0 = no L_PCM or IEC 61937 compressed audio sample packet received within the last 10 HS signals. 1 = L_PCM or IEC 61937 compressed audio sample packet received within the last 10 HS signals.
0x19	DST_DOUBLE	00000**0**00	R	A flag to indicate when the DST audio is double data rate. 0 = no DST double data rate audio detected. 1 = DST double data rate audio detected.
0x1A	IGNORE_PARITY_ERR	1**0**000000	R/W	A control to select the processing of audio samples even when they have a parity error. 0 = discards audio sample packet that have an invalid parity bit. 1 = processes audio sample packets that have an invalid parity bit.
0x1A	MUTE_AUDIO	100**0**0000	R/W	A control to force an internal mute independently of the mute mask conditions. 0 = audio in normal operation. 1 = force audio mute.
0x1A	WAIT_UNMUTE[2:0]	1000**000**0	R/W	A control to delay the audio unmute. When 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. 000 = disables/cancels delayed unmute. The audio unmutes directly after all mute conditions become inactive. 001 = unmutes 250 ms after all mute conditions become inactive. 010 = unmutes 500 ms after all mute conditions become inactive. 011 = unmutes 750 ms after all mute conditions become inactive. 100 = unmutes 1 sec after all.

Address	Name	Bits	Access	Description
0x1A	NOT_AUTO_UNMUTE	1000000**0**	R/W	A control to disable the automaticunmute feature. When this bit is 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. 0 = audio unmutes following a delay set by WAIT_UNMUTE after all mute conditions have become inactive. 1 = prevents audio from unmuting automatically.
0x1B	DCFIFO_RESET_ON_LOCK	000**1**1000	R/W	Enables the reset/recentering of video FIFO when the video PLL unlocks. 0 = do not reset on video PLL lock. 1 = resets FIFO on video PLL lock.
0x1B	DCFIFO_KILL_NOT_LOCKED	0001**1**000	R/W	DCFIFO_KILL_NOT_LOCKED controls whether or not the output of the video FIFO is set to zero when the video PLL is unlocked. 0 = FIFO data is output regardless of video PLL lock status. 1 = FIFO output is zeroed if video PLL is unlocked.
0x1B	DCFIFO_KILL_DIS	00011**0**00	R/W	The video FIFO output is zeroed if there is more than one resynchronization of the pointers within two FIFO cycles. This behavior can be disabled with this bit. 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.
0x1C	DCFIFO_LOCKED	0000**0**000	R	A readback to indicate if the video FIFO is locked. 0 = video FIFO is not locked. The video FIFO had to resynchronize between previous two VS signals. 1 = video FIFO is locked. The video FIFO did not have to resynchronize between previous two VS signals.
0x1C	DCFIFO_LEVEL[2:0]	00000**000**	R	A readback that indicates the distance between the read and write pointers. Overflow/underflow reads as Level 0. Ideal centered functionality reads as 0b100. 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.
0x1D	UP_CONVERSION_MODE	00**0**00000	R/W	A control to select the 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. 0 = Cr and Cb samples are repeated in their respective channel. 1 = interpolate Cr and Cb values.
0x1E, 0x1F	TOTAL_LINE_WIDTH[13:0]	00**000000 00000000**	R	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. XXXXXXXXXXXXX = total number of pixels per line.
0x20, 0x21	HSYNC_FRONT_PORCH[12:0]	000**00000 00000000**	R	HS 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. XXXXXXXXXXX = total number of pixels in the front porch.
0x22, 0x23	HSYNC_PULSE_WIDTH[12:0]	000**00000 00000000**	R	HS 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. XXXXXXXXXXX = total number of pixels in the HS pulse.

Address	Name	Bits	Access	Description
0x24, 0x25	HSYNC_BACK_PORCH[12:0]	000**00000 00000000**	R	HS 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. XXXXXXXXXX = total number of pixels in the back porch.
0x26, 0x27	FIELD0_TOTAL_HEIGHT[13:0]	00**000000 00000000**	R	Field 0 total height is a vertical synchronization measurement. This readback gives the total number of half lines in Field 0. This measurement is valid only when the vertical filter has locked. XXXXXXXXXXXXX = the total number of half lines in Field 0 (divide readback by 2 to obtain the number of lines).
0x28, 0x29	FIELD1_TOTAL_HEIGHT[13:0]	00**000000 00000000**	R	Field 1 total height is a vertical synchronization measurement. This readback gives the total number of half lines in Field 1. This measurement is valid only when the vertical filter has locked. Field 1 measurements are valid when HDMI_INTERLACED is set to 1. XXXXXXXXXXXXX = the total number of half lines in Field 1 (divide readback by 2 to obtain the number of lines).
0x2A, 0x2B	FIELD0_VS_FRONT_PORCH[13:0]	00**000000 00000000**	R	Field 0 VS front porch width is a vertical synchronization measurement. The unit of this measurement is half lines. This measurement is valid only when the vertical filter has locked. XXXXXXXXXXXXX = the total number of half lines in the VS front porch of Field 0 (divide readback by 2 to obtain the number of lines).
0x2C, 0x2D	FIELD1_VS_FRONT_PORCH[13:0]	00**000000 00000000**	R	Field 1 VS front porch width is a vertical synchronization measurement. The unit of this measurement is half lines. This measurement is valid only when the vertical filter has locked. Field 1 measurements are valid when HDMI_INTERLACED is set to 1. XXXXXXXXXXXXX = the total number of half lines in the VS front porch of Field 1 (divide readback by 2 to obtain the number of lines).
0x2E, 0x2F	FIELD0_VS_PULSE_WIDTH[13:0]	00**000000 00000000**	R	Field 0 VS width is a vertical synchronization measurement. The unit for this measurement is half lines. This measurement is valid only when the vertical filter has locked. XXXXXXXXXXXXX = the total number of half lines in the VS pulse of Field 0 (divide readback by 2 to obtain the number of lines).
0x30, 0x31	FIELD1_VS_PULSE_WIDTH[13:0]	00**000000 00000000**	R	Field 1 VS width is a vertical synchronization measurement. The unit for this measurement is half lines. This measurement is valid only when the vertical filter has locked. Field 1 measurements are valid when HDMI_INTERLACED is set to 1. XXXXXXXXXXXXX = the total number of half lines in the VS pulse of Field 1 (divide readback by 2 to obtain the number of lines)
0x32, 0x33	FIELD0_VS_BACK_PORCH[13:0]	00**000000 00000000**	R	Field 0 VS back porch width is a vertical synchronization measurement. The unit for this measurement is half lines. 00000000000000 = the total number of half lines in the VS back porch of Field 0 (divide readback by 2 to obtain the number of lines). XXXXXXXXXXXXX = the total number of half lines in the VS Back Porch of Field 0 (divide readback by 2 to obtain the number of lines).
0x34, 0x35	FIELD1_VS_BACK_PORCH[13:0]	00**000000 00000000**	R	Field 1 VS back porch width is a vertical synchronization measurement. The unit for this measurement is half lines. This measurement is valid only when the vertical filter has locked. Field 1 measurements are valid when HDMI_INTERLACED is set to 1. XXXXXXXXXXXXX = the number of half lines in the VS back porch of Field 1 (divide readback by 2 to obtain number of lines).

Address	Name	Bits	Access	Description
0x36, 0x37, 0x38, 0x39, 0x3A	CS_DATA[39:0]	**00000000 00000000 00000000 00000000 00000000**	R	Readback registers for the channel status data bits collected from Audio Channel 0.
0x3C	BYPASS_AUDIO_PASSTHRU	000**0**0010	R/W	Enable/disable for audio passthrough mode.
0x40	OVERRIDE_DEEP_COLOR_MODE	0**0**000000	R/W	A control to override the Deep Color mode. 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].
0x40	DEEP_COLOR_MODE_USER[1:0]	00**00**0000	R/W	A control to manually set the Deep Color mode. The value set in this register is effective when OVERRIDE_DEEP_COLOR_MODE is set to 1. 00 = 8 bits per channel. 01 = 10 bits per channel. 10 = 12 bits per channel. 11 = 16 bits per channel (not supported).
0x41	DEREP_N_OVERRIDE	0100**0**000	R/W	This control allows the user to override the pixel repetition factor. The ADV7613 then uses DEREP_N instead of HDMI_PIXEL_REPETITION[3:0] to discard video pixel data from the incoming HDMI stream. 0 = automatic detection and processing of the procession of pixel repeated modes using the AVI InfoFrame information. 1 = enables the manual setting of the pixel repetition factor as per DEREP_N[3:0].
0x41	DEREP_N[3:0]	0100**0000**	R/W	Sets the derepetition value if derepetition is overridden by setting DEREP_N_OVERRIDE. 0000 = DEREP_N + 1 indicates the pixel and clock discard factor. No derepetition is used. XXXX = DEREP_N + 1 indicates the pixel and clock discard factor. The derepetition value is used.
0x47	QZERO_ITC_DIS	00000**0**00	R/W	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. 0 = AVI InfoFrame ITC bit. 1 = manual RGB range as per QZERO_RGB_FULL.
0x47	QZERO_RGB_FULL	000000**0**0	R/W	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. 0 = RGB limited range when Q[1:0] = 00. 1 = RGB full when Q[1:0] = 00.
0x47	ALWAYS_STORE_INF	0000000**0**	R/W	A control to force InfoFrames with checksum errors to be stored. 0 = stores data from received InfoFrames only if their checksum is correct. 1 = always store the data from received InfoFrame regardless of their checksum.
0x48	DIS_CABLE_DET_RST	0**0**000000	R/W	This control disables the reset effects of cable detection. DIS_CABLE_DET_RST must be set to 1 if the 5 V pin is unused and left unconnected. 0 = resets the HDMI section if the 5 V input pin corresponding to the selected HDMI port (for example, RXA_5V for Port A) is inactive. 1 = do not use the 5 V input pin as the reset signal for the HDMI section.

Address	Name	Bits	Access	Description
0x50	GAMUT_IRQ_NEXT_FIELD	000**0**0000	R/W	A control to set the NEW_GAMUT_MDATA_RAW interrupt to detect when the new contents are applicable to the next field or to indicate that the gamut packet is new. This is done using header information of the gamut packet. 0 = interrupt flag indicates that the gamut packet is new. 1 = interrupt flag indicates that the gamut packet is to be applied to the next field.
0x50	CS_COPYRIGHT_MANUAL	000000**0**0	R/W	A control to select the automatic or manual setting of the copyright value of the channel status bit that is passed to the S/PDIF output. Manual control is set with the CS_COPYRIGHT_VALUE bit. 0 = automatic channel status (CS) copyright control. 1 = manual CS copyright control. The manual value is set by CS_COPYRIGHT_VALUE.
0x50	CS_COPYRIGHT_VALUE	0000000**0**	R/W	A control to set the CS copyright value when in manual configuration of the CS copyright bit that is passed to the S/PDIF output. 0 = copyright value of channel status bit is 0. Valid only when CS_COPYRIGHT_MANUAL is set to 1. 1 = copyright value of channel status bit is 1. Valid only when CS_COPYRIGHT_MANUAL is set to 1.
0x51, 0x52	TMDSFREQ[8:0]	**00000000** **0**0000000	R	This register provides a full precision integer TMDS frequency measurement. XXXXXXXXX = outputs a 9-bit TMDS frequency measurement in MHz.
0x52	TMDSFREQ_FRAC[6:0]	0**0000000**	R	A readback to indicate the fractional bits of measured frequency of PLL recovered TMDS clock. The unit is 1/128 MHz. XXXXXXX = outputs a 7-bit TMDS fractional frequency measurement in 1/128 MHz.
0x53	HDMI_COLORSPACE[3:0]	0000**0000**	R	A readback of the HDMI input color space decoded from several fields in the AVI InfoFrame. 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
0x56	FILT_5V_DET_DIS	**0**1011000	R/W	This bit is a control to disable the digital glitch filter on the HDMI 5 V detect signals. The filtered signals are used as interrupt flags and 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 42 MHz ± 10%. If the 5 V pins are not used and left unconnected, the 5 V detect circuitry must be disconnected from the HDMI reset signal by setting DIS_CABLE_DET_RST to 1. This avoids holding the HDMI section in reset. 0 = enabled. 1 = disabled.

Address	Name	Bits	Access	Description
0x56	FILT_5V_DET_TIMER[6:0]	0**1011000**	R/W	This bit is a control to set the timer for the digital glitch filter on the HDMI 5 V detect inputs. The unit of this parameter is two clock cycles of the ring oscillator (~47 ns). 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. 1011000 = approximately 4.2 µs. XXXXXXX = time duration of 5 V deglitch filter. The unit of this parameter is 2 clock cycles of the ring oscillator (~47 ns).
0x5A	HDCP_REPT_EDID_RESET	0000**0**000	SC	A reset control for the E-EDID/repeater controller. When asserted, it resets the E-EDID/repeater controller. 0 = normal operation. 1 = resets the E-EDID/repeater controller.
0x5A	DCFIFO_RECENTER	00000**0**00	SC	A reset to recenter the video FIFO. This is a self clearing bit. 0 = video FIFO normal operation. 1 = video FIFO to recenter.
0x5A	FORCE_N_UPDATE	0000000**0**	SC	A control to force an N and CTS value update to the audio DPLL. The audio DPLL regenerates the audio clock. 0 = no effect. 1 = forces an update on the N and CTS values for audio clock regeneration.
0x5B, 0x5C, 0x5D	CTS[19:0]	**00000000 00000000 0000**0000	R	A readback for the CTS value received in the HDMI data stream. 00000000000000000000 = default CTS value readback from HDMI stream. XXXXXXXXXXXXXXXXXXX = CTS value readback from HDMI stream.
0x5D, 0x5E, 0x5F	N[19:0]	0000**0000 00000000 00000000**	R	A readback for the N value received in the HDMI data stream. 00000000000000000000 = default N value readback from HDMI data stream. XXXXXXXXXXXXXXXXXXX = N value readback from HDMI data stream.
0x6C	HPA_DELAY_SEL[3:0]	**1010**0010	R/W	Sets a delay between 5 V detection and hot plug assertion on the HPA_A output pin, in increments of 100 ms per bit. 0000 = no delay. 0001 = 100 ms delay. 0010 = 200 ms delay. 1010 = 1 sec delay. 1111 = 1.5 sec delay.
0x6C	HPA_OVR_TERM	1010**0**010	R/W	A control to set termination control to be overridden by the hot plug assert (HPA) setting. When this bit is set, termination on a specific port is set according to the HPA status of that port. 0 = automatic or manual I^2C control of port termination. 1 = termination controls disabled and overridden by HPA controls.
0x6C	HPA_AUTO_INT_EDID[1:0]	101000**01**0	R/W	Selects the type of automatic control on the HPA_A output pin. This bit has no effect when HPA_MANUAL is set to 1. 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 deasserted low immediately after the internal E-EDID is deactivated for that port. 01 = the HPA of an HDMI port is asserted high following a programmable delay after the device detects an HDMI cable plug on that port. The HPA of an HDMI port is immediately deasserted after the device detects a cable disconnect on that HDMI port.

Address	Name	Bits	Access	Description
				10 = the HPA of an HDMI port is asserted high after two conditions have been met. The conditions are detailed as follows: the internal EDID is active for that port, and 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 deasserted after any of the following two conditions have been met: the internal EDID is deactivated for that port, and 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: the internal EDID is active for that port, the delayed version of the cable detect signal CABLE_DET_X_RAW for that port is high, and the user has set the manual HPA control for that port to 1 via the HPA_MAN_VALUE_A control. The HPA of an HDMI port is immediately deasserted after any of the following three conditions have been met: the internal EDID is deactivated for that port, the cable detect signal CABLE_DET_X_RAW for that port is low, and the user sets the manual HPD control for that port to 0 via the HPA_MAN_VALUE_A control.
0x6C	HPA_MANUAL	1010001**0**	R/W	Manual control enable for the hot plug assert output pin, HPA_A. By setting this bit, any automatic control of HPA_A is disabled. Manual control is determined by the HPA_MAN_VALUE_A. 0 = HPA takes its value based on HPA_AUTO_INT_EDID[1:0]. 1 = HPA takes its value from HPA_MAN_VALUE_A.
0x6D	I2S_TDM_MODE_ENABLE	**0**0000000	R/W	Enables I^2S TDM output mode, where all four stereo pairs are output through the I^2S0 signal. This mode can only be used in multichannel modes. Only the following f_S ratios for MCLKOUT are valid: 1, 2, and 4. 0 = disable TDM mode, each stereo pair comes out in an I^2S signal. 1 = enable TDM mode, all four stereo pairs are time multiplexed into I^2S0.
0x6D	I2S_SPDIF_MAP_INV	0**0**000000	R/W	A control to invert the arrangement of the I^2S or S/PDIF interface on the audio output port pins. The arrangement of the I^2S or S/PDIF interface on the audio output port pins is determined by I2S_SPDIF_MAP_ROT[1:0]. 0 = do not invert the arrangement of the I^2S or S/PDIF channels in audio output port pins. 1 = invert the arrangement of the I^2S or S/PDIF channels in audio output port pins.
0x6D	I2S_SPDIF_MAP_ROT[1:0]	00**00**0000	R/W	A control to select the arrangement of the I^2S or S/PDIF interface on the audio output port pins. 00 = I^2S0/SPDIF0 on AP1, I^2S1/SPDIF1 on AP2, I^2S2/SPDIF2 on AP3, I^2S3/SPDIF3 on AP4. 01 = I^2S3/SPDIF3 on AP1, I^2S0/SPDIF0 on AP2, I^2S1/SPDIF1 on AP3, I^2S2/SPDIF2 on AP4. 10 = I^2S2/SPDIF2 on AP1, I^2S3/SPDIF3 on AP2, I^2S0/SPDIF0 on AP3, I^2S1/SPDIF1 on AP4. 11 = I^2S1/SPDIF1 on AP1, I^2S2/SPDIF2 on AP2, I^2S3/SPDIF3 on AP3, I^2S0/SPDIF0 on AP4.
0x6D	DSD_MAP_INV	0000**0**000	R/W	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[2:0]. 0 = do not invert the arrangement of the DSD channels on the audio output port pins. 1 = invert the arrangement of the DSD channels on the audio output port pins.

Address	Name	Bits	Access	Description
0x6D	DSD_MAP_ROT[2:0]	00000**000**	R/W	A control to select the arrangement of the DSD interface on the audio output port pins. 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.
0x83	CLOCK_TERMA_DISABLE	1111111**1**	R/W	Disable clock termination on Port A. Can be used when TERM_AUTO is set to 0. 0 = enable Termination Port A. 1 = disable Termination Port A.

REPEATER REGISTER MAP BIT DESCRIPTIONS

Table 12. Repeater Register Map Bit Descriptions

Address	Name	Bits	Access	Description
0x00, 0x01, 0x02, 0x03, 0x04	BKSV[39:0]	**00000000 00000000 00000000 00000000 00000000**	R	The receiver key selection vector (BKSV) can be read back after the device has accessed the HDCP ROM. The following registers contain the BKSV read from the EEPROM. 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].
0x08, 0x09	RI[15:0]	**00000000 00000000**	R	R_I generated by the HDCP core.
0x0A	PJ[7:0]	**00000000**	R	Pj generated by the HDCP core.
0x10, 0x11, 0x12, 0x13, 0x14	AKSV[39:0]	**00000000 00000000 00000000 00000000 00000000**	R/W	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. 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]
0x15	AINFO[7:0]	**00000000**	R/W	AINFO written by Tx.
0x18, 0x19, 0x1A, 0x1B, 0x1C, 0x1D, 0x1E, 0x1F	AN[63:0]	**00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000**	R/W	AN written by Tx. 0x10[7:0] = AKSV[7:0]
0x20, 0x21, 0x22, 0x23	SHA_A[31:0]	**00000000 00000000 00000000 00000000**	R/W	SHA Hash Part A generated on chip. 0x11[7:0] = AKSV[15:8].
0x40	BCAPS[7:0]	**10000011**	R/W	This is the BCAPS register presented to the Tx attached to the active HDMI port. 10000011 = default BCAPS register value presented to the Tx. XXXXXXXX = BCAPS register value presented to the Tx.
0x41, 0x42	BSTATUS[15:0]	**00000000 00000000**	R/W	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. 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].
0x70	SPA_LOCATION[7:0]	**11000000**	R/W	This is the location in the E-EDID record where the source physical address (SPA) is located. 11000000 = default value. XXXXXXXX = location of source physical address in internal E-EDID of Port B, Port C, and Port D.

Address	Name	Bits	Access	Description
0x71	KSV_LIST_READY	**0**0000000	R/W	The system sets this bit to indicate that the KSV list has been read from the Tx ICs and written into the repeater map. The system must also set Bits[11:0] of BSTATUS before setting this bit. 0 = not ready. 1 = ready.
0x71	SPA_STORAGE_MODE	0000000**0**0	R/W	Selects how the SPA must be stored in the nonvolatile EEPROM. 0 = store only SPA for Port A. 1 = store SPA for Port A plus the upper nibble of SPA for the rest of the ports.
0x71	SPA_LOCATION_MSB	0000000**0**	R/W	Additional MSB of SPA_LOCATION (that is, SPA_LOCATION[8]) needed to point to SPAs stored in the second segment.
0x72	EXT_EEPROM_TRI	0**0**000000	R/W	Tristates the output pins to the external SPI EEPROM. 0 = SPI interface outputs enabled. 1 = SPI interface outputs tristated.
0x73	VGA_EDID_ENABLE_CPU	00**0**00000	R	Flags the internal EDID enable in the VGA port. 0 = disabled. 1 = enabled.
0x74	EDID_A_ENABLE	0000000**0**	R/W	Enables I^2C access to internal EDID RAM from DDC Port A. 0 = E-EDID for Port A disabled. 1 = E-EDID for Port A enabled.
0x76	EDID_A_ENABLE_CPU	0000000**0**	R	Flags the internal EDID enable on Port A. 0 = disabled. 1 = enabled.
0x78	KSV_LIST_READY_CLR_A	0000000**0**	SC	Clears the BCAPS KSV list ready bit in Port A.
0x79	VGA_EDID_ENABLE	**0**0001000	R/W	Enables the I^2C access to the internal EDID RAM for the VGA port. Note that enabling these bits disables access to the upper segment in the DDC ports.
0x79	KSV_MAP_SELECT[2:0]	0**000**1000	R/W	Selects which 128 bytes of the KSV list are accessed when reading or writing to Address 0x80 to Address 0xFF in this map. Values from 5 and greater are not valid.
0x79	AUTO_HDCP_MAP_ENABLE	0000**1**000	R/W	Selects which port is accessed for HDCP addresses: the HDMI active port or the one selected in HDCP_MAP_SELECT. 0 = HDCP data read from port given by HDCP_MAP_SELECT. 1 = HDCP data read from the active HDMI port.
0x79	HDCP_MAP_SELECT[2:0]	00001**000**	R/W	Selects which port is accessed for the HDCP addresses (Address 0x00 to Address 0x42 in the repeater register map). This only takes effect when AUTO_HDCP_MAN_ENABLE is 0. 000 = select Port A. 001 = reserved. 010 = reserved. 011 = reserved. 100 = reserved.
0x7A	DISABLE_AUTO_EDID	000001**0**0	R/W	Disables all automatic enables for the internal E-EDID. 0 = automatic enable of internal E-EDID on HDMI ports when the device comes out of Power-Down Mode 0. 1 = disable automatic enable of internal E-EDID on HDMI ports when the device comes out of Power-Down Mode 0.
0x7A	EDID_SEGMENT_POINTER	0000010**0**	R/W	Segment pointer for internal EDID in the main I^2C.
0x80	KSV_BYTE_0[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x81	KSV_BYTE_1[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].

Address	Name	Bits	Access	Description
0x82	KSV_BYTE_2[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x83	KSV_BYTE_3[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x84	KSV_BYTE_4[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x85	KSV_BYTE_5[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x86	KSV_BYTE_6[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x87	KSV_BYTE_7[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x88	KSV_BYTE_8[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x89	KSV_BYTE_9[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x8A	KSV_BYTE_10[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x8B	KSV_BYTE_11[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x8C	KSV_BYTE_12[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x8D	KSV_BYTE_13[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x8E	KSV_BYTE_14[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x8F	KSV_BYTE_15[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x90	KSV_BYTE_16[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x91	KSV_BYTE_17[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x92	KSV_BYTE_18[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x93	KSV_BYTE_19[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x94	KSV_BYTE_20[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].

Address	Name	Bits	Access	Description
0x95	KSV_BYTE_21[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x96	KSV_BYTE_22[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x97	KSV_BYTE_23[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x98	KSV_BYTE_24[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x99	KSV_BYTE_25[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x9A	KSV_BYTE_26[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x9B	KSV_BYTE_27[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x9C	KSV_BYTE_28[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x9D	KSV_BYTE_29[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x9E	KSV_BYTE_30[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0x9F	KSV_BYTE_31[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xA0	KSV_BYTE_32[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xA1	KSV_BYTE_33[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xA2	KSV_BYTE_34[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xA3	KSV_BYTE_35[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xA4	KSV_BYTE_36[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xA5	KSV_BYTE_37[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xA6	KSV_BYTE_38[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xA7	KSV_BYTE_39[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].

Address	Name	Bits	Access	Description
0xA8	KSV_BYTE_40[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xA9	KSV_BYTE_41[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xAA	KSV_BYTE_42[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xAB	KSV_BYTE_43[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xAC	KSV_BYTE_44[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xAD	KSV_BYTE_45[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xAE	KSV_BYTE_46[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xAF	KSV_BYTE_47[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xB0	KSV_BYTE_48[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xB1	KSV_BYTE_49[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xB2	KSV_BYTE_50[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xB3	KSV_BYTE_51[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xB4	KSV_BYTE_52[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xB5	KSV_BYTE_53[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xB6	KSV_BYTE_54[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xB7	KSV_BYTE_55[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xB8	KSV_BYTE_56[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xB9	KSV_BYTE_57[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xBA	KSV_BYTE_58[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].

Address	Name	Bits	Access	Description
0xBB	KSV_BYTE_59[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xBC	KSV_BYTE_60[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xBD	KSV_BYTE_61[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xBE	KSV_BYTE_62[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xBF	KSV_BYTE_63[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xC0	KSV_BYTE_64[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xC1	KSV_BYTE_65[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xC2	KSV_BYTE_66[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xC3	KSV_BYTE_67[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xC4	KSV_BYTE_68[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xC5	KSV_BYTE_69[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xC6	KSV_BYTE_70[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xC7	KSV_BYTE_71[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xC8	KSV_BYTE_72[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xC9	KSV_BYTE_73[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xCA	KSV_BYTE_74[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xCB	KSV_BYTE_75[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xCC	KSV_BYTE_76[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xCD	KSV_BYTE_77[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].

Address	Name	Bits	Access	Description
0xCE	KSV_BYTE_78[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xCF	KSV_BYTE_79[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xD0	KSV_BYTE_80[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xD1	KSV_BYTE_81[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xD2	KSV_BYTE_82[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xD3	KSV_BYTE_83[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xD4	KSV_BYTE_84[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xD5	KSV_BYTE_85[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xD6	KSV_BYTE_86[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xD7	KSV_BYTE_87[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xD8	KSV_BYTE_88[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xD9	KSV_BYTE_89[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xDA	KSV_BYTE_90[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xDB	KSV_BYTE_91[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xDC	KSV_BYTE_92[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xDD	KSV_BYTE_93[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xDE	KSV_BYTE_94[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xDF	KSV_BYTE_95[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xE0	KSV_BYTE_96[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].

Address	Name	Bits	Access	Description
0xE1	KSV_BYTE_97[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xE2	KSV_BYTE_98[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xE3	KSV_BYTE_99[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xE4	KSV_BYTE_100[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xE5	KSV_BYTE_101[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xE6	KSV_BYTE_102[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xE7	KSV_BYTE_103[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xE8	KSV_BYTE_104[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xE9	KSV_BYTE_105[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xEA	KSV_BYTE_106[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xEB	KSV_BYTE_107[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xEC	KSV_BYTE_108[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xED	KSV_BYTE_109[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xEE	KSV_BYTE_110[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xEF	KSV_BYTE_111[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xF0	KSV_BYTE_112[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xF1	KSV_BYTE_113[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xF2	KSV_BYTE_114[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xF3	KSV_BYTE_115[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].

Address	Name	Bits	Access	Description
0xF4	KSV_BYTE_116[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xF5	KSV_BYTE_117[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xF6	KSV_BYTE_118[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xF7	KSV_BYTE_119[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xF8	KSV_BYTE_120[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xF9	KSV_BYTE_121[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xFA	KSV_BYTE_122[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xFB	KSV_BYTE_123[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xFC	KSV_BYTE_124[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xFD	KSV_BYTE_125[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xFE	KSV_BYTE_126[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].
0xFF	KSV_BYTE_127[7:0]	**00000000**	R/W	This is a byte in the KSV list used for the HDCP repeater protocol. This may come from any of five 128-byte segments, controlled by KSV_MAP_SELECT[2:0].

INFOFRAME REGISTER MAP BIT DESCRIPTIONS

Table 13. InfoFrame Register Map Bit Descriptions

Address	Name	Bits	Access	Description
0x00, 0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0x07, 0x08, 0x09, 0x0A, 0x0B, 0x0C, 0x0D, 0x0E, 0x0F, 0x10, 0x11, 0x12, 0x13, 0x14, 0x15, 0x16, 0x17, 0x18, 0x19, 0x1A, 0x1B	AVI_INF_PB[223:0]	**00000000** **00000000**	R	AVI InfoFrame data.
0x1C, 0x1D, 0x1E, 0x1F, 0x20, 0x21, 0x22, 0x23, 0x24, 0x25, 0x26, 0x27, 0x28, 0x29	AUD_INF_PB[111:0]	**00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000**	R	Audio InfoFrame data.

Address	Name	Bits	Access	Description
0x2A, 0x2B, 0x2C, 0x2D, 0x2E, 0x2F, 0x30, 0x31, 0x32, 0x33, 0x34, 0x35, 0x36, 0x37, 0x38, 0x39, 0x3A, 0x3B, 0x3C, 0x3D, 0x3E, 0x3F, 0x40, 0x41, 0x42, 0x43, 0x44, 0x45	SPD_INF_PB[223:0]	**00000000** **00000000**	R	SPD InfoFrame data.
0x46, 0x47, 0x48, 0x49, 0x4A, 0x4B, 0x4C, 0x4D, 0x4E, 0x4F, 0x50, 0x51, 0x52, 0x53	MS_INF_PB[111:0]	**00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000** **00000000**	R	MPEG source InfoFrame data.

Address	Name	Bits	Access	Description
0x54, 0x55, 0x56, 0x57, 0x58, 0x59, 0x5A, 0x5B, 0x5C, 0x5D, 0x5E, 0x5F, 0x60, 0x61, 0x62, 0x63, 0x64, 0x65, 0x66, 0x67, 0x68, 0x69, 0x6A, 0x6B, 0x6C, 0x6D, 0x6E, 0x6F	VS_INF_PB[223:0]	**00000000** **00000000**	R	Vendor specific InfoFrame data.
0x70, 0x71, 0x72, 0x73, 0x74, 0x75, 0x76, 0x77, 0x78, 0x79, 0x7A, 0x7B, 0x7C, 0x7D, 0x7E, 0x7F, 0x80, 0x81, 0x82, 0x83, 0x84, 0x85, 0x86, 0x87, 0x88, 0x89, 0x8A, 0x8B	ACP_PB[223:0]	**00000000** **00000000**	R	ACP InfoFrame data.

Address	Name	Bits	Access	Description
0x8C, 0x8D, 0x8E, 0x8F, 0x90, 0x91, 0x92, 0x93, 0x94, 0x95, 0x96, 0x97, 0x98, 0x99, 0x9A, 0x9B, 0x9C, 0x9D, 0x9E, 0x9F, 0xA0, 0xA1, 0xA2, 0xA3, 0xA4, 0xA5, 0xA6, 0xA7	ISRC1_PB[223:0]	**00000000** **00000000**	R	ISRC1 InfoFrame data.
0xA8, 0xA9, 0xAA, 0xAB, 0xAC, 0xAD, 0xAE, 0xAF, 0xB0, 0xB1, 0xB2, 0xB3, 0xB4, 0xB5, 0xB6, 0xB7, 0xB8, 0xB9, 0xBA, 0xBB, 0xBC, 0xBD, 0xBE, 0xBF, 0xC0, 0xC1, 0xC2, 0xC3	ISRC2_PB[223:0]	**00000000** **00000000**	R	ISRC2 InfoFrame data.

Address	Name	Bits	Access	Description
0xC4, 0xC5, 0xC6, 0xC7, 0xC8, 0xC9, 0xCA, 0xCB, 0xCC, 0xCD, 0xCE, 0xCF, 0xD0, 0xD1, 0xD2, 0xD3, 0xD4, 0xD5, 0xD6, 0xD7, 0xD8, 0xD9, 0xDA, 0xDB, 0xDC, 0xDD, 0xDE, 0xDF	GBD[223:0]	**00000000** **00000000**	R	Gamut InfoFrame data.
0xE0	AVI_PACKET_ID[7:0]	**10000010**	R/W	AVI InfoFrame ID.
0xE1	AVI_INF_VERS[7:0]	**00000000**	R	AVI InfoFrame version.
0xE2	AVI_INF_LEN[7:0]	**00000000**	R	AVI InfoFrame length.
0xE3	AUD_PACKET_ID[7:0]	**10000100**	R/W	Audio InfoFrame ID.
0xE4	AUD_INF_VERS[7:0]	**00000000**	R	Audio InfoFrame version.
0xE5	AUD_INF_LEN[7:0]	**00000000**	R	Audio InfoFrame length.
0xE6	SPD_PACKET_ID[7:0]	**10000011**	R/W	SPD InfoFrame ID.
0xE7	SPD_INF_VERS[7:0]	**00000000**	R	SPD InfoFrame version.
0xE8	SPD_INF_LEN[7:0]	**00000000**	R	SPD InfoFrame length.
0xE9	MS_PACKET_ID[7:0]	**10000101**	R/W	MPEG source InfoFrame ID.
0xEA	MS_INF_VERS[7:0]	**00000000**	R	MPEG source InfoFrame version.
0xEB	MS_INF_LEN[7:0]	**00000000**	R	MPEG source InfoFrame length.
0xEC	VS_PACKET_ID[7:0]	**10000001**	R/W	Vendor specific InfoFrame ID.
0xED	VS_INF_VERS[7:0]	**00000000**	R	Vendor specific InfoFrame version.
0xEE	VS_INF_LEN[7:0]	**00000000**	R	Vendor specific InfoFrame length.
0xEF	ACP_PACKET_ID[7:0]	**00000100**	R/W	ACP packet ID.
0xF0	ACP_TYPE[7:0]	**00000000**	R	ACP Packet Header Byte 1 (ACP type).
0xF1	ACP_HEADER2[7:0]	**00000000**	R	ACP Packet Header Byte 2.
0xF2	ISRC1_PACKET_ID[7:0]	**00000101**	R/W	ISRC1 packet ID.
0xF3	ISRC1_HEADER1[7:0]	**00000000**	R	ISRC1 Packet Header Byte 1.
0xF4	ISRC1_HEADER2[7:0]	**00000000**	R	ISRC1 Packet Header Byte 2.
0xF5	ISRC2_PACKET_ID[7:0]	**00000110**	R/W	ISRC2 packet ID.
0xF6	ISRC2_HEADER1[7:0]	**00000000**	R	ISRC2 Packet Header Byte 1.
0xF7	ISRC2_HEADER2[7:0]	**00000000**	R	ISRC2 Packet Header Byte 2.
0xF8	GAMUT_PACKET_ID[7:0]	**00001010**	R/W	Gamut packet ID.
0xF9	GAMUT_HEADER1[7:0]	**00000000**	R	Gamut Packet Header Byte 1.
0xFA	GAMUT_HEADER2[7:0]	**00000000**	R	Gamut Packet Header Byte 2.

CP REGISTER MAP BIT DESCRIPTIONS

Table 14. CP Register Map Bit Descriptions

Address	Name	Bits	Access	Description
0x0B	RB_CSC_SCALE[1:0]	**00**000000	R	Readback of the CSC scale applied to the CSC coefficients. XX = readback value.
0x0B, 0x0C	RB_A4[12:0]	000**00000 00000000**	R	Readback of CSC Coefficient A4 modified by the video adjustment block. XXXXXXXXXXXXX = readback value.
0x0D, 0x0E	RB_A3[12:0]	0**0000000 000000**00	R	Readback of CSC Coefficient A3 modified by the video adjustment block. XXXXXXXXXXXXX = readback value.
0x0E, 0x0F, 0x10	RB_A2[12:0]	000000**00 00000000 000**00000	R	Readback of CSC Coefficient A2 modified by the video adjustment block. XXXXXXXXXXXXX = readback value.
0x10, 0x11	RB_A1[12:0]	000**00000 00000000**	R	Readback of CSC Coefficient A1 modified by the video adjustment block. XXXXXXXXXXXXX = readback value.
0x12, 0x13	RB_B4[12:0]	000**00000 00000000**	R	Readback of CSC Coefficient B4 modified by the video adjustment block. XXXXXXXXXXXXX = readback value.
0x14, 0x15	RB_B3[12:0]	0**0000000 000000**00	R	Readback of CSC Coefficient B3 modified by the video adjustment block. XXXXXXXXXXXXX = readback value.
0x15, 0x16, 0x17	RB_B2[12:0]	000000**00 00000000 000**00000	R	Readback of CSC Coefficient B2 modified by the video adjustment block. XXXXXXXXXXXXX = readback value.
0x17, 0x18	RB_B1[12:0]	000**00000 00000000**	R	Readback of CSC Coefficient B1 modified by the video adjustment block. XXXXXXXXXXXXX = readback value.
0x19, 0x1A	RB_C4[12:0]	000**00000 00000000**	R	Readback of CSC Coefficient C4 modified by the video adjustment block. XXXXXXXXXXXXX = readback value.
0x1B, 0x1C	RB_C3[12:0]	0**0000000 000000**00	R	Readback of CSC Coefficient C3 modified by the video adjustment block. XXXXXXXXXXXXX = readback value.
0x1C, 0x1D, 0x1E	RB_C2[12:0]	000000**00 00000000 000**00000	R	Readback of CSC Coefficient C2 modified by the video adjustment block. XXXXXXXXXXXXX = readback value.
0x1E, 0x1F	RB_C1[12:0]	000**00000 00000000**	R	Readback of CSC Coefficient C1 modified by the video adjustment block. XXXXXXXXXXXXX = readback value.
0x22, 0x23	CP_START_HS[12:0]	000**00000 00000000**	R/W	A control to set the position of the start of the HS output signal in the CP core in automatic graphics mode only. Programming of this parameter is optional and must only be performed when the device is set in automatic graphics mode. The value is unsigned. 0x0000 = default value.
0x24, 0x25	CP_END_HS[12:0]	000**00000 00000000**	R/W	A control to set the position of the end of the HS output signal in the CP core in automatic graphics mode only. Programming of this parameter is optional and must only be performed when the device is set in automatic graphics mode. The value is unsigned. 0x0000 = default value.

Address	Name	Bits	Access	Description
0x26, 0x27	CP_START_SAV[12:0]	000**00000 00000000**	R/W	Manual value for the start of active video (SAV) position. Sets the total number of pixels between the start of nonactive video and the SAV. Programming of this parameter is optional and must only be performed when the device is set in automatic graphics mode. The value is unsigned. 0x0000 = default value.
0x28, 0x29	CP_START_EAV[12:0]	000**00000 00000000**	R/W	Manual value for the end of active video (EAV) position. Sets the total number of pixels between the end of nonactive video and the EAV. Programming of this parameter is optional and must only be performed when the device is set in automatic graphics mode. The value is unsigned. 0x0000 = default value.
0x2A, 0x2B	CP_START_VBI_R[11:0]	**00000000 0000**0000	R/W	Manual value for the start of the VBI position of the extra blank region preceding the odd right field in 3D TV field alternative packing format. Normally not required to program because this parameter is calculated automatically from the input.
0x2B, 0x2C	CP_END_VBI_R[11:0]	0000**0000 00000000**	R/W	Manual value for the end of the VBI position of the extra blank region preceding the odd right field in 3D TV field alternative packing format. Normally not required to program because this parameter is calculated automatically from the input.
0x2D, 0x2E	CP_START_VBI_EVEN_R[11:0]	**00000000 0000**0000	R/W	Manual value for start of VBI position of the extra blank region preceding the even right field in 3D TV field alternative packing format. Normally not required to program because this parameter is calculated automatically from the input.
0x2E, 0x2F	CP_END_VBI_EVEN_R[11:0]	0000**0000 00000000**	R/W	Manual value for end of VBI position of the extra blank region preceding the even right field in 3D TV field alternative packing format. Normally not required to program because this parameter is calculated automatically from the input.
0x30	DE_V_START_R[3:0]	**0000**0000	R/W	A control to vary the position of the start of the extra VBI region between the left and right fields during the odd field in 3D TV field alternative packing format. This register stores a signed value represented in a twos complement format. The unit of DE_V_END_EVEN[9:0] is one pixel clock. Range = −8 lines to +7 lines.
0x30	DE_V_END_R[3:0]	0000**0000**	R/W	A control to vary the position of the end of the extra VBI region between the left and right fields during the odd field in 3D TV field alternative packing format. This register stores a signed value represented in a twos complement format. The unit of DE_V_END_EVEN[9:0] is one pixel clock. Range = −8 lines to +7 lines.
0x31	DE_V_START_EVEN_R[3:0]	**0000**0000	R/W	A control to vary the position of the start of the extra VBI region between the left and right fields during the even field in 3D TV field alternative packing format. This register stores a signed value represented in a twos complement format. The unit of DE_V_END_EVEN[9:0] is one pixel clock. Range = −8 lines to +7 lines.
0x31	DE_V_END_EVEN_R[3:0]	0000**0000**	R/W	A control to vary the position of the end of the extra VBI region between the left and right fields during the even field in 3D TV field alternative packing format. This register stores a signed value represented in a twos complement format. The unit of DE_V_END_EVEN[9:0] is one pixel clock. Range = −8 lines to +7 lines.

Address	Name	Bits	Access	Description
0x36	TEN_TO_EIGHT_CONV	0000000**0**	R/W	A control to indicate whether the precision of the data to be rounded and truncated to 8-bit has 10-bit precision. This control is for HDMI use only. 0 = if the input data has 12-bit precision, the output data has 12 bits, 10 bits, or 8 bits per channel. If the input data has 10-bit precision, the output data has 10 bits per channel. If the input data has 8-bit precision, the output data has 8 bits per channel. 1 = if the input data has 10-bit precision, the output data has 8 bits per channel.
0x3A	CP_CONTRAST[7:0]	**10000000**	R/W	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 represent the fractional part of the contrast value. The fractional part has a range of 0 to 0.99. This control is functional if VID_ADJ_EN is set to 1. 00000000 = contrast set to minimum. 10000000 = default. 11111111 = contrast set to maximum.
0x3B	CP_SATURATION[7:0]	**10000000**	R/W	A control to set the saturation. This field is a 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. 00000000 = saturation set to the minimum. 10000000 = default. 11111111 = saturation set to the maximum.
0x3C	CP_BRIGHTNESS[7:0]	**00000000**	R/W	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 of 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. 00000000 = the offset applied to the luma is 0. 01111111 = the offset applied to the luma is +508 decimal. This value corresponds to the brightness setting. 11111111 = the offset applied to the Luma is −512 decimal. This value corresponds to the darkest setting.
0x3D	CP_HUE[7:0]	**00000000**	R/W	A control to set the hue. This register represents an unsigned value that 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. 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.
0x3E	VID_ADJ_EN	**0**0000000	R/W	Video adjustment enable. This control selects whether the color controls feature is enabled. The color controls feature is configured via the CP_CONTRAST[7:0], CP_SATURATION[7:0], CP_BRIGHTNESS[7:0] and CP_HUE[7:0] parameters. The CP CSC must also be enabled for the color controls to be effective. 0 = disables the color controls. 1 = enables the color controls.

Address	Name	Bits	Access	Description
0x3E	CP_UV_ALIGN_SEL[1:0]	00**00**0000	R/W	Alignment of the UV_DATA_SIGNAL internal signal generated by the CP core. The UV_DATA_SIGNAL signal is used to map U and V pixels data into one single signal when the device is configured to output a 4:2:2 digital video stream. 00 = the UV_DATA_SIGNAL is synchronized with the SAV. 01 = the UV_DATA_SIGNAL is synchronized with the leading edge of the HS signal. 10 = UV_DATA_SIGNAL is synchronized with the leading edge of the DE signal. 11 = the UV_DATA_SIGNAL is synchronized with the SAV.
0x3E	CP_UV_DVAL_INV	0000**0**000	R/W	This controls the polarity of the UV_DATA_VALID signal generated by the CP. The UV_DATA_VALID signal is used to map U and V pixels data into one single signal when the device is configured to output a 4:2:2 digital video stream. 0 = no change to UV_DATA_VALID signal. 1 = invert UV_DATA_VALID signal.
0x3E	CP_MODE_GAIN_ADJ_EN	00000**0**00	R/W	A control to enable pregain. 0 = the pregain block is bypassed. 1 = the pregain block is enabled.
0x3E	ALT_SAT_UV_MAN	000000**0**0	R/W	U and V saturation range control. 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.
0x3E	ALT_SAT_UV	0000000**0**	R/W	Cr and Cb saturation range. Refer to ALT_SAT_UV_MAN for additional information. 0 = the range of the saturators on the Cr and Cb channels is 15 to 235. 1 = the range of the saturators on the Cr and Cb channels is 16 to 240.
0x40	CP_MODE_GAIN_ADJ[7:0]	**01011100**	R/W	Pregain adjustment to compensate for the gain of the analog front end. This register stores a value in a 1.7 binary format. 0XXXXXXX = gain of (0 + (XXXXXXX ÷ 128)). 10000000 = default pregain (pregain of 1.0). 1XXXXXXX = gain of (1 + (XXXXXXX ÷ 128)).
0x52	CSC_SCALE[1:0]	**01**000000	R/W	A control to set the CSC coefficient scalar. 00 = CSC scalar set to 1. 01 = CSC scalar set to 2. 10 = reserved. Do not use. 11 = reserved. Do not use.
0x52, 0x53	A4[12:0]	010**00000 00000000**	R/W	CSC Coefficient A4. Contains the 13-bit A4 coefficient for the A channel. 0x0000 = default value.
0x54, 0x55	A3[12:0]	0**0000000 000000**00	R/W	CSC Coefficient A3. Contains the 13-bit A3 coefficient for the A channel. 0x0000 = default value.
0x55, 0x56, 0x57	A2[12:0]	000000**00 00000000 000**01000	R/W	CSC Coefficient A2. Contains the 13-bit A2 coefficient for the A channel. 0x0000 = default value.
0x57, 0x58	A1[12:0]	000**01000 00000000**	R/W	CSC Coefficient A1. Contains the 13-bit A1 coefficient for the A channel. 0x0800 = default value.
0x59, 0x5A	B4[12:0]	000**00000 00000000**	R/W	CSC Coefficient B4. Contains the 13-bit B4 coefficient for the B channel. 0x0000 = default value.

Address	Name	Bits	Access	Description
0x5B, 0x5C	B3[12:0]	0**0000000 000000**01	R/W	CSC Coefficient B3. Contains the 13-bit B3 coefficient for the B channel. 0x0000 = default value.
0x5C, 0x5D, 0x5E	B2[12:0]	000000**01 00000000 000**00000	R/W	CSC Coefficient B2. Contains the 13-bit B2 coefficient for the B channel. 0x0800 = default value.
0x5E, 0x5F	B1[12:0]	000**00000 00000000**	R/W	CSC Coefficient B1. Contains the 13-bit B1 coefficient for the B channel. 0x0000 = default value.
0x60, 0x61	C4[12:0]	000**00000 00000000**	R/W	CSC Coefficient C4. Contains the 13-bit C4 coefficient for the C channel. 0x0000 = default value.
0x62, 0x63	C3[12:0]	0**0100000 000000**00	R/W	CSC Coefficient C3. Contains the 13-bit C3 coefficient for the C channel. 0x0800 = default value.
0x63, 0x64, 0x65	C2[12:0]	000000**00 00000000 000**00000	R/W	CSC Coefficient C2. Contains the 13-bit C2 coefficient for the C channel. 0x0000 = default value.
0x65, 0x66	C1[12:0]	000**00000 00000000**	R/W	CSC Coefficient C1. Contains the 13-bit C1 coefficient for the C channel. 0x0000 = default value.
0x67	EMB_SYNC_ON_ALL	00**0**00000	R/W	A control to alter the gain computed by the AGC based on the presence of an embedded synchronization on Channel A, Channel B, and Channel C. Used only when the RGB input and RGB output with color controls are enabled. 0 = embedded synchronization is present only on the luma channel (that is, Channel A). 1 = all three input channels have embedded synchronization.
0x68	CSC_COEFF_SEL[3:0]	**1111**0000	R/W	A control to select the mode in which the CP CSC operates. 0000 = CP CSC configuration in manual mode. 1111 = CP CSC configured in automatic mode. All other values = reserved.
0x68	CP_CHROMA_LOW_EN	1111**0**000	R/W	Filter response control for the 444 to 422 chroma decimation filter. 0 = high bandwidth, sharp transition filter for Channel B/Channel C. 1 = soft filter with minimized ringing for Channel B/Channel C.
0x69	MAN_CP_CSC_EN	000**0**0100	R/W	A control to manually enable the CP CSC. By default, the CP CSC is automatically enabled when either a color space conversion or a video adjustment (hue, saturation, contrast, brightness) is determined to be required due to other I^2C settings. If MAN_CP_CSC_EN is set to 1, the CP CSC is forced into the enabled state. 0 = CP CSC is automatically enabled if required. For example, if either a color space conversion or a video adjustment (hue, saturation, contrast, brightness) is determined to be required due to other I^2C settings. 1 = manual override to force CP CSC to be enabled.
0x69	EIA_861_COMPLIANCE	00000**1**00	R/W	Control for compliance to CEA-861 for 525p. This bit sets the start of the VBI for the 525p standard only. 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 CEA-861 specification.
0x6C	CLMP_A_MAN	**0**0010000	R/W	Manual clamping enable for Channel A. 0 = ignore internal digital fine clamp loop result. Use CLMP_A[11:0]. 1 = use the digital fine clamp value determined by the on-chip clamp loop.

Address	Name	Bits	Access	Description
0x6C	CLMP_BC_MAN	0**0**010000	R/W	Manual clamping enable for Channel B and Channel C. 0 = ignore internal digital fine clamp loop result. Use CLMP_B[11:0] for Channel B and CLMP_C[11:0] for Channel C. 1 = use the digital fine clamp value determined by the on-chip clamp loop.
0x6C	CLMP_FREEZE	00**0**10000	R/W	Stops the digital fine clamp loops for Channel A, Channel B, and Channel C from updating. 0 = clamp value updated on every active video line. 1 = clamp loops are stopped and not updated.
0x6C, 0x6D	CLMP_A[11:0]	0001**0000 00000000**	R/W	Manual clamp value for Channel A. This field is an unsigned 12-bit value to be subtracted from the incoming video signal. The value programmed in this register is effective if the CLMP_A_MAN is set to 1. To change CLMP_A[11:0], Register 0x6C and Register 0x6D must be updated with the desired clamp value written to in this order and with no other I^2C access in between. 0x000 = minimum range. … 0xFFF = maximum range.
0x6E, 0x6F	CLMP_B[11:0]	**00000000 0000**0000	R/W	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 CLMP_B[11:0], Register 0x6E and Register 0x6F must be updated with the desired clamp value written to in this order and with no other I^2C access in between. 0x000 = minimum range. … 0xFFF = maximum range.
0x6F, 0x70	CLMP_C[11:0]	0000**0000 00000000**	R/W	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 CLMP_C[11:0], Register 0x6F and Register 0x70 must be updated with the desired clamp value written to in this order and with no other I^2C access in between. 0x000 = minimum range. … 0xFFF = maximum range.
0x73	GAIN_MAN	**0**0010000	R/W	Enables the gain factor to be set by the AGC or manually. 0 = AGC controls the gain for all three channels. 1 = manual gains are used for all three channels.
0x73	AGC_MODE_MAN	0**0**010000	R/W	A control to set how the gains for all three channels is configured. 0 = the gain is dependent on the type of input and OP_656_RANGE. 1 = gain operation controlled by GAIN_MAN.
0x73, 0x74	A_GAIN[9:0]	00**010000 0000**0100	R/W	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], Register 0x73 and Register 0x74 must be written to in this order with no I^2C access in between. 0x000 = gain of 0. 0x100 = unity gain. 0x3FF = gain of 3.99.

Address	Name	Bits	Access	Description
0x74, 0x75	B_GAIN[9:0]	0000**0100 000000**01	R/W	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], Register 0x74 and Register 0x75 must be written to in this order with no I²C access in between. 0x000 = gain of 0. 0x100 = unity gain. 0x3FF = gain of 3.99.
0x75, 0x76	C_GAIN[9:0]	000000**01 00000000**	R/W	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], Register 0x75 and Register 0x76 must be written to in this order with no I²C access in between. 0x000 = gain of 0. 0x100 = unity gain. 0x3FF = gain of 3.99.
0x77	CP_PREC[1:0]	**11**111111	R/W	A control to set the precision of the data output by the CP core for Channel A, Channel B, and Channel C. 00 = rounds and truncates data in Channel A, Channel B, and Channel C to 10-bit precision. 01 = rounds and truncates data in Channel A, Channel B, and Channel C to 12-bit precision. 10 = rounds and truncates data in Channel A, Channel B, and Channel C to 8-bit precision. 11 = rounds and truncates data in Channel A, Channel B, and Channel C to the precision set in OP_FORMAT_SEL[7:0].
0x77, 0x78	A_OFFSET[9:0]	11**111111 1111**1111	R/W	A control to set the manual offset for Channel A. This field stores an unsigned value. To change A_OFFSET[9:0], Register 0x77 and Register 0x78 must be written to in this order with no I²C access in between. 0x3FF = automatic offset to Channel A. Any other value = Channel A offset.
0x78, 0x79	B_OFFSET[9:0]	1111**1111 111111**11	R/W	A control to set the manual offset for Channel B. This field stores an unsigned value. To change B_OFFSET[9:0], Register 0x78 and Register 0x79 must be written to in this order with no I²C access in between. 0x3FF = automatic offset to Channel B. Any other value = Channel B offset.
0x79, 0x7A	C_OFFSET[9:0]	111111**11 11111111**	R/W	A control to set the manual offset for Channel C. This field stores an unsigned value. To change C_OFFSET[9:0], Register 0x79 and Register 0x7A must be written to in this order with no I²C access in between. 0x3FF = automatic offset to Channel C. Any other value = Channel C offset.
0x7B	AV_INV_F	**0**0000101	R/W	A control to invert the F bit in the AV codes. 0 = inserts the F bit with default polarity. 1 = inverts the F bit before inserting it into the AV code.
0x7B	AV_INV_V	0**0**000101	R/W	A control to invert the V bit in AV codes. 0 = do not invert the V bit polarity before inserting it into the AV code. 1 = invert the V bit polarity before inserting it into the AV code.
0x7B	AV_POS_SEL	00000**1**01	R/W	A control to select the AV codes position. 0 = SAV code at the HS falling edge and EAV code at the HS rising edge. 1 = uses predetermined (default) positions for the AV codes.
0x7B	DE_WITH_AVCODE	0000010**1**	R/W	A control to insert AV codes in relation to the DE output signal. 0 = AV codes locked to default values. The DE position can be moved independently of AV codes. 1 = inserted AV codes moves in relation to the DE position change.

Address	Name	Bits	Access	Description
0x7C	CP_INV_HS	**1**1000000	R/W	A control to set the polarity of the HS/composite sync signal output by the CP core. This control is not recommended for use. 0 = the CP outputs a HS/composite sync signal with negative polarity. 1 = the CP outputs a HS/composite sync signal with positive polarity.
0x7C,	CP_INV_VS	1**1**000000	R/W	A control to set the polarity of the VS output by the CP core. This control is not recommended for use. 0 = the CP outputs a VS signal with negative polarity. 1 = the CP outputs a VS signal with positive polarity.
0x7C	CP_INV_DE	110**0**0000	R/W	A control to set the polarity of the field/DE output by the CP core. This control is not recommended for use. 0 = the CP outputs a field/DE signal with negative polarity. 1 = the CP outputs a field/DE signal with positive polarity.
0x7C, 0x7E	START_HS[9:0]	1100**00**00 **00000000**	R/W	A control to shift the position of the leading edge of the HS signal output by the CP core. This register stores a signed value in a twos complement format. START_HS[9:0] is the number of pixel clocks by which the leading edge of the HS signal is shifted (for example, 0x3FF corresponds to a shift of one pixel clock from the active video, and 0x005 corresponds to a shift of five pixel clocks toward the active video). 0x000 = default value. 0x000 to 0x1FF = the leading edge of the HS signal is shifted toward the active video. 0x200 to 0x3FF = the leading edge of the HS signal is shifted from the active video.
0x7C, 0x7D	END_HS[9:0]	110000**00** **00000000**	R/W	A control to shift the position of the trailing edge of the HS signal output by the CP core. This register stores a signed value in a twos complement format. HS_END[9:0] is the number of pixel clock by which the leading edge of the HS signal is shifted (for example, 0x3FF corresponds to a shift of one pixel clock form the active video, 0x005 corresponds to a shift of five pixel clocks toward the active video). 0x000 = default value. 0x000 to 0x1FF = the trailing edge of the HS signal is shifted toward the active video. 0x200 to 0x3FF = the trailing edge of the HS signal is shifted from the active video.
0x7F	START_VS[3:0]	**0000**0000	R/W	A control to shift the position of the leading edge of the VS signal output by the CP core. This register stores a signed value in a twos complement format. START_VS[3:0] is the number of lines by which the leading edge of the VS signal is shifted (for example, 0x0F corresponds to a shift by one line toward the active video, and 0x01 corresponds to a shift of one line from the active video). 0x0 = default value. 0x0 to 0x7 = the leading edge of the VS signal is shifted toward the active video. 0x8 to 0xF = the leading edge of the VS signal is shifted away from the active video.

Address	Name	Bits	Access	Description
0x7F	END_VS[3:0]	0000**0000**	R/W	A control to shift the position of the trailing edge of the VS signal output by the CP core. This register stores a signed value in a twos complement format. SEND_VS[3:0] is the number of lines by which the trailing edge of the VS signal is shifted (for example, 0x0F corresponds to a shift of one line toward the active video, and 0x01 corresponds to a shift of one line from the active video). 0x0 = default value. 0x0 to 0x7 = the trailing edge of the VS signal is shifted toward the active video. 0x8 to 0xF = the trailing edge of the VS signal is shifted from the active video.
0x80	START_FE[3:0]	**0000**0000	R/W	A control to shift the position of the start of the even field edge of the field signal output by the CP core. This register stores a signed value in a twos complement format. START_FE[3:0] is the number of lines by which the start of the even fields edge of the field signal is shifted (for example, 0x0D corresponds to a shift of three lines toward the active video, and 0x05 corresponds to a shift of five lines from the active video). 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 from the active video.
0x80	START_FO[3:0]	0000**0000**	R/W	A control to shift the position of the start of the odd field edge of the field signal output by the CP core. This register stores a signed value in a twos complement format. START_FO[3:0] is the number of lines by which the start of the odd fields edge of the field signal is shifted (for example, 0x0D corresponds to a shift of three lines toward the active video, and 0x05 corresponds to a shift of five lines from the active video). 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 from the active video.
0x84	CP_GAIN_FILT[3:0]	**0000**1100	R/W	A control to set Coefficient A of the infinite impulse response (IIR) 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 an IIR filter with a transfer function of the form $Y[N] = (1 - A) \times Y[N - 1] + A \times X[N]$. 0000 = no filtering, that is, 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.

Address	Name	Bits	Access	Description
0x86	CH1_TRIG_STDI	00001**0**11	R/W	Trigger synchronization source and polarity detector for sync Channel 1 STDI. A 0 to 1 transition in this bit restarts the autosync detection algorithm. This is not a self clearing bit and must be set to 0 to prepare for the next trigger. 0 = default value. 1 = a 0 to 1 transition restarts the autosync detection algorithm.
0x86	CH1_STDI_CONT	000010**1**1	R/W	A control to select the Sync Channel 1 STDI mode of operation. 0 = Sync Channel 1 STDI block operates in single-shot mode. A 0 to 1 transition on CH1_TRIG_STDI triggers a measurement of the sync channel 1 STDI block. 1 = Sync Channel 1 STDI runs in continuous mode.
0x88	DE_V_START_EVEN[3:0]	**0000**0000	R/W	A control to vary the start position of the VBI region in the even field. This register stores a signed value represented in a twos complement format. The unit of DE_V_START_EVEN[3:0] is one pixel clock.
0x88	DE_V_END_EVEN[3:0]	0000**0000**	R/W	A control to vary the position of the end of the VBI region in the even field. This register stores a signed value represented in a twos complement format. The unit of DE_V_END_EVEN[9:0] is one pixel clock. Range = −8 lines to +7 lines.
0x89	START_VS_EVEN[3:0]	**0000**0000	R/W	A control to shift the position of the leading edge of the VS signal output by the CP core. This register stores a signed value in a twos complement format. START_VS_EVEN[3:0] is the number of lines by which the leading edge of the VS signal is shifted (for example, 0x0F corresponds to a shift by one line toward the active video, and 0x01 corresponds to a shift of one line from the active video). 0x0 to 0x7 = the leading edge of the even VS signal is shifted toward the active video. 0x8 to 0xF = the leading edge of the even VS signal is shifted from the active video.
0x89	END_VS_EVEN[3:0]	0000**0000**	R/W	A control to shift the position of the trailing edge of the VS signal output by the CP core. This register stores a signed value in a twos complement format. END_VS_EVEN[3:0] is the number of lines by which the trailing edge of the VS is shifted (for example, 0x0F corresponds to a shift of one line toward the active video, and 0x01 corresponds to a shift of one line from the active video). 0x0 to 0x7 = the trailing edge of the even VS signal is shifted toward the active video. 0x8 to 0xF = the trailing edge of the even VS signal is shifted from the active video.
0x8B, 0x8D	DE_H_START[9:0]	0100**00**00 **00000000**	R/W	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 twos complement format. The unit of DE_H_START[9:0] is one pixel clock. 0x200 = −512 pixels of shift. 0x3FF = −1 pixel of shift. 0x000 = default value (no shift). 0x001 = +1 pixel of shift. 0x1FF = +511 pixels.

Address	Name	Bits	Access	Description
0x8B, 0x8C	DE_H_END[9:0]	010000**00 00000000**	R/W	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 twos complement format. The unit of DE_H_END[9:0] is one pixel clock. 0x200 = −512 pixels of shift. 0x3FF = −1 pixel of shift. 0x000 = default value (no shift). 0x001 = +1 pixel of shift. 0x1FF = +511 pixels.
0x8E	DE_V_START[3:0]	**0000**0000	R/W	A control to vary the start position of the VBI region. This register stores a signed value represented in a twos complement format. The unit of DE_V_START[3:0] is one line. 1000 = −8 lines of shift. 1111 = −1 line of shift. 0000 = default. 0001 = +1 line of shift. 0111 = +7 lines of shift.
0x8E	DE_V_END[3:0]	0000**0000**	R/W	A control to vary the position of the end of the VBI region. This register stores a signed value represented in a twos complement format. The unit of DE_V_START[3:0] is one line. 1000 = −8 lines of shift. 1111 = −1 line of shift. 0000 = default. 0001 = +1 line of shift. 0111 = +7 lines of shift.
0x8F, 0x90	CH1_FR_LL[10:0]	01000**000 00000000**	R/W	Free run line length in number of crystal clock cycles in one line of video for Sync Channel 1 STDI. Program this register only for video standards that are not supported by PRIM_MODE[3:0] and VID_STD[5:0]. 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.
0x91	Interlaced	0**1**000000	R/W	Sets the interlaced/progressive mode of the incoming video processed through the CP core. 0 = video mode is progressive. 1 = video mode is interlaced.
0x9A, 0x9B	CP_START_VS[5:0]	000**00000 0**0000000	R/W	A control to set the position of the start of the VS output signal in the CP core in automatic graphics mode only. In the case of an interlaced signal, this register adjusts the odd VS signal. Programming of this parameter is optional and must only be performed when the device is set in automatic graphics mode. The value is unsigned. 000000 = default value.
0x9B	CP_END_VS[5:0]	0**000000**0	R/W	A control to set the position of the end of the VS output signal in the CP core in automatic graphics mode only. In the case of an interlaced signal, this register adjusts the odd VS signal. Programming of this parameter is optional and must only be performed when the device is set in automatic graphics mode. The value is unsigned. 000000 = default value.
0x9C, 0x9D	CP_START_VS_EVEN[10:0]	**00000000 000**00000	R/W	A control to set the position of the start of the even VS signal output signal in the CP core in automatic graphics mode only. Programming of this parameter is optional and must only be performed when the device is set in automatic graphics mode. The value is unsigned. 0x000 = default value.

Address	Name	Bits	Access	Description
0x9D, 0x9E	CP_END_VS_EVEN[10:0]	000**00000 000000**00	R/W	A control to set the position of the end of the even VS output signal in the CP core in automatic graphics mode only. Programming of this parameter is optional and must only be performed when the device is set in automatic graphics mode. The value is unsigned. 0x000 = default value.
0x9F, 0xA0	CP_START_F_ODD[10:0]	**00000000 000**00000	R/W	A control to set the position of the end of the odd field output signal in the CP core in automatic graphics mode only. Programming of this parameter is optional and must only be performed when the device is set in automatic graphics mode. The value is unsigned. 0x000 = default value.
0xA0, 0xA1	CP_START_F_EVEN[10:0]	000**00000 000000**00	R/W	A control to set the position of the end of the even field output signal in the CP core in automatic graphics mode only. Programming of this parameter is optional and must only be performed when the device is set in automatic graphics mode. The value is unsigned. 0x000 = default value.
0xA3, 0xA4	CH1_LCF[11:0]	0000**0000 00000000**	R	Readback for Sync Channel 1 line count in a field. This is the number of lines between two VS signals measured on Sync Channel 1. The readback from this field is valid if CH1_STDI_DVALID is high. XXXXXXXXXXX = readback value.
0xA5, 0xA6	CP_START_VBI[11:0]	**00000000 0000**0000	R/W	Manual value for the start of the 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 noninterlaced standard video, and the total number of lines at the start of the odd frame of interlaced standard video. Programming of this parameter is optional and must only be performed when the device is set in automatic graphics mode. 0x000 = default value.
0xA6, 0xA7	CP_END_VBI[11:0]	0000**0000 00000000**	R/W	Manual value for the end of the 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 noninterlaced standard video, and the total number of lines at the end of the odd frame of interlaced standard video. Programming of this parameter is optional and must only be performed when the device is set in automatic graphics mode. 0x000 = default value.
0xA8, 0xA9	CP_START_VBI_EVEN[11:0]	**00000000 0000**0000	R/W	Manual value for the start of the VBI region in the even fields. This is an unsigned value. This is the total number of lines at the start of the even frame of interlaced standard. Programming of this parameter is optional and must only be performed when the device is set in automatic graphics mode. 0x000 = default value.
0xA9, 0xAA	CP_END_VBI_EVEN[11:0]	0000**0000 00000000**	R/W	Manual value for the end of the VBI region position for the even fields. This is an unsigned value. This is the total number of lines at the end of the even frame of interlaced standard. Programming of this parameter is optional and must only be performed when the device is set in automatic graphics mode. 0x000 = default value.

Address	Name	Bits	Access	Description
0xAB, 0xAC	CP_LCOUNT_MAX[11:0]	**00000000 0000**0000	R/W	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] is set to 0x000. 0x000 = number of lines per frame used when deciding to enter free run is decoded from PRIM_MODE[3:0] and VID_STD[5:0] for Sync Channel 1. All other values = number of lines per frame used when deciding to enter free run is decoded from this register for Sync Channel 1.
0xB1	CH1_STDI_DVALID	**0**0000000	R	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. 0 = Sync Channel 1 STDI measurements are not valid. 1 = Sync Channel 1 STDI measurements are valid.
0xB1	CH1_STDI_INTLCD	0**0**000000	R	Interlaced vs. progressive mode detected by Sync Channel 1 STDI. The readback from this register is valid if CH1_STDI_DVALID is high. 0 = indicates a video signal on Sync Channel 1 with noninterlaced timing. 1 = indicates a signal on Sync Channel 1 with interlaced timing.
0xB1, 0xB2	CH1_BL[13:0]	00**000000 00000000**	R	A readback for the block length for Sync Channel 1. This is the number of crystal cycle cycles in a block of eight lines of incoming video. This readback is valid if CH1_STDI_DVALID is high. XXXXXXXXXXXXXX = readback value.
0xB3	CH1_LCVS[4:0]	**00000**000	R	A readback for the Sync Channel 1 Line Count in a VS period. Number of lines in a VS period measured on Sync Channel 1. The readback from this field is valid if CH1_STDI_DVALID is high. XXXXX = readback value.
0xB8, 0xB9	CH1_FCL[12:0]	000**00000 00000000**	R	A readback for the Sync Channel 1 field count length. This is the number of crystal clock cycles between successive VS signals 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. XXXXXXXXXXXXX = readback value.
0xBA	HDMI_FRUN_MODE	000000**0**1	R/W	A control to configure the free run feature in HDMI mode. 0 = HDMI Free Run Mode 0. The device 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 if the video resolution of HDMI stream processed by the device does not match the video resolution programmed in PRIM_MODE[3:0] and VID_STD[5:0].
0xBA	HDMI_FRUN_EN	0000000**1**	R/W	A control to enable free run in HDMI mode. 0 = disable the free run feature in HDMI mode. 1 = enable the free run feature in HDMI mode.
0xBD	DPP_BYPASS_EN	000**1**1000	R/W	Manual control to enable DPP block. 1 = DPP bypassed. 0 = DPP enabled.
0xBE	DLY_A	**0**0000000	R/W	A control to delay the data on Channel A by one pixel clock cycle. 1 = delay the data of Channel A by one pixel clock cycle. 0 = do not delay the data of Channel A.

Address	Name	Bits	Access	Description
0xBE	DLY_B	0**0**000000	R/W	A control to delay the data on Channel B by one pixel clock cycle. 1 = delay the data of Channel B by one pixel clock cycle. 0 = do not delay the data of Channel B.
0xBE	DLY_C	00**0**00000	R/W	A control to delay the data on Channel C by one pixel clock cycle. 1 = delay the data of Channel C by one pixel clock cycle. 0 = do not delay the data of Channel C.
0xBE, 0xBF	HCOUNT_ALIGN_ADJ[4:0]	000000**00** **000**10010	R/W	Manual adjustment for internally generated horizontal count 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. 00000 = default value.
0xBF	CP_DEF_COL_MAN_VAL	00010**0**10	R/W	A control to enable manual selection of the color used when the CP core free runs. 0 = uses default color blue. 1 = outputs default colors as given in DEF_COL_CHA, DEF_COL_CHB, and DEF_COL_CHC.
0xBF	CP_DEF_COL_AUTO	000100**1**0	R/W	A control to enable the insertion of the default color when the CP free runs. 0 = disable automatic insertion of default color. 1 = output default colors when the CP free runs.
0xBF	CP_FORCE_FREERUN	0001001**0**	R/W	A control to force the CP core to free run. 0 = do not force the CP core free run. 1 = force the CP core to free run.
0xC0	DEF_COL_CHA[7:0]	**00000000**	R/W	A control to set the default color for Channel A. To be used if CP_DEF_COL_MAN_VAL is 1. 0x00 = default value.
0xC1	DEF_COL_CHB[7:0]	**00000000**	R/W	A control to set the default color for Channel B. To be used if CP_DEF_COL_MAN_VAL is 1. 0x00 = default value.
0xC2	DEF_COL_CHC[7:0]	**00000000**	R/W	A control to set the default color for Channel C. To be used if CP_DEF_COL_MAN_VAL is 1. 0x00 = default value.
0xC9	SWAP_SPLIT_AV	00101**1**00	R/W	A control to swap the luma and chroma AV codes in DDR mode. 0 = swap the luma and chroma AV codes in DDR mode. 1 = do not swap the luma and chroma AV codes in DDR mode.
0xC9	DIS_AUTO_PARAM_BUFF	0010110**0**	R/W	A control to disable the buffering of the timing parameters used for free run in HDMI mode. 0 = buffer the last measured parameters in HDMI mode used to determine the video resolution, which the device free runs to. 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].
0xCB	AUTO_SL_FILTER_FREEZE_EN	01**1**00000	R/W	This bit determines whether the internally generated parameter for the position of the HS trailing edge is updated during the VBI region. This control is only intended for automatic graphics mode. It is recommended to leave AUTO_SL_FILTER_FREEZE_EN at the default setting. Unless AUTO_SL_FILTER_FREEZE_EN is left at the default setting, the device may generate an incorrect HS trailing edge position parameter if the input synchronization is embedded and has serration pulses. 0 = do not freeze the trailing edge position of the HS signal during the VBI region. 1 = freeze the trailing edge position of the HS signal during the VBI region.

Address	Name	Bits	Access	Description
0xCB	HDMI_CP_LOCK_THRESHOLD[1:0]	011000**00**	R/W	Locking time of filter used for buffering of timing parameters in HDMI mode. 00 = slowest locking time. 01 = medium locking time. 10 = fastest locking time. 11 = fixed step size of 0.5 pixel.
0xE0	HDMI_CP_AUTOPARM_LOCKED	0**0**000000	R	A readback to report the lock status of the parameter buffering in HDMI mode. 0 = the parameter buffering block has not locked to the synchronization signal from the HDMI core. 1 = the parameter buffering block has locked to the synchronization signal from the HDMI core.
0xE0	HDMI_AUTOPARM_STS[1:0]	00**00**0000	R	CP status for HDMI mode. 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.
0xE0, 0xE1	CP_AGC_GAIN[9:0]	000000**00** **00000000**	R	A readback value of the gain used on the data of Channel A. The value stored in this register is in a 1.9 binary format and is composed of one integer and nine fractional bits. XXXXXXXXXX = readback value of the gain.
0xF3	CH1_FL_FR_THRESHOLD[2:0]	11**010**100	R/W	Threshold for the difference between the input video field length and the internally stored standard to enter and exit free run. 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.
0xF3	CH1_F_RUN_THR[2:0]	11010**100**	R/W	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 enters free run mode. 000 = minimum difference to switch into free run is 2 XTAL clock cycles. Maximum difference to switch out of free run is 1 XTAL clock cycle. 001 = minimum difference to switch into free run is 256 XTAL clock cycles. Maximum difference to switch out of free run is 200 XTAL clock cycles. 010 = minimum difference to switch into free run is 128 XTAL clock cycles. Maximum difference to switch out of free run is 112 XTAL clock cycles.

Address	Name	Bits	Access	Description
				011 = minimum difference to switch into free run is 64 XTAL clock cycles. Maximum difference to switch out of free run is 48 XTAL clock cycles. 100 = minimum difference to switch into free run is 32 XTAL clock cycles. Maximum difference to switch out of free run is 24 XTAL clock cycles. 101 = minimum difference to switch into free run is 16 XTAL clock cycles. Maximum difference to switch out of free run is 12 XTAL clock cycles. 110 = minimum difference to switch into free run is 8 XTAL clock cycles. Maximum difference to switch out of free run is 6 XTAL clock cycles. 111 = minimum difference to switch into free run is 4 XTAL clock cycles. Maximum difference to switch out of free run is 3 XTAL clock cycles.
0xF4	CSC_COEFF_SEL_RB[3:0]	**0000**0000	R	Readback of the CP CSC conversion when configured in automatic mode. 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.
0xF5	BYPASS_STDI1_LOCKING	000000**0**0	R/W	Bypass the STDI lock for Sync Channel 1. 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, and CH1_LCVS from the Sync Channel 1 STDI as they are measured.
0xFF	CP_FREE_RUN	000**0**0000	R	Component processor free run status. 0 = the CP is not free running. 1 = the CP is free running.

CEC REGISTER MAP BIT DESCRIPTIONS

Table 15. CEC Register Map Bit Descriptions

Address	Name	Bits	Access	Description
0x00	CEC_TX_FRAME_HEADER[7:0]	**00000000**	R/W	Header block in the transmitted frame.
0x01	CEC_TX_FRAME_DATA0[7:0]	**00000000**	R/W	Opcode block in the transmitted frame.
0x02	CEC_TX_FRAME_DATA1[7:0]	**00000000**	R/W	Operand 1 in the transmitted frame.
0x03	CEC_TX_FRAME_DATA2[7:0]	**00000000**	R/W	Operand 2 in the transmitted frame.
0x04	CEC_TX_FRAME_DATA3[7:0]	**00000000**	R/W	Operand 3 in the transmitted frame.
0x05	CEC_TX_FRAME_DATA4[7:0]	**00000000**	R/W	Operand 4 in the transmitted frame.
0x06	CEC_TX_FRAME_DATA5[7:0]	**00000000**	R/W	Operand 5 in the transmitted frame.
0x07	CEC_TX_FRAME_DATA6[7:0]	**00000000**	R/W	Operand 6 in the transmitted frame.
0x08	CEC_TX_FRAME_DATA7[7:0]	**00000000**	R/W	Operand 7 in the transmitted frame.
0x09	CEC_TX_FRAME_DATA8[7:0]	**00000000**	R/W	Operand 8 in the transmitted frame.
0x0A	CEC_TX_FRAME_DATA9[7:0]	**00000000**	R/W	Operand 9 in the transmitted frame.
0x0B	CEC_TX_FRAME_DATA10[7:0]	**00000000**	R/W	Operand 10 in the transmitted frame.
0x0C	CEC_TX_FRAME_DATA11[7:0]	**00000000**	R/W	Operand 11 in the transmitted frame.
0x0D	CEC_TX_FRAME_DATA12[7:0]	**00000000**	R/W	Operand 12 in the transmitted frame.
0x0E	CEC_TX_FRAME_DATA13[7:0]	**00000000**	R/W	Operand 13 in the transmitted frame.
0x0F	CEC_TX_FRAME_DATA14[7:0]	**00000000**	R/W	Operand 14 in the transmitted frame.
0x10	CEC_TX_FRAME_LENGTH[4:0]	000**00000**	R/W	Message size of the transmitted frame. This is the number of bytes in the outgoing message, including the header. XXXXX = total number of bytes (including header byte) to be sent
0x11	CEC_TX_ENABLE	0000000**0**	R/W	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 still in the signal free time stage, the message transmission is terminated. If data transmission has begun, the transmission continues until the message is fully sent or until an error condition occurs. 0 = transmission mode disabled. 1 = transmission mode enabled and message transmission started.
0x12	CEC_TX_RETRY[2:0]	0**001**0011	R/W	The number of times the CEC Tx tries to retransmit the message if an error condition is encountered. Per the CEC specification, this value must not be set to a value greater than 5. 001 = try to retransmit the message 1 time if an error occurs. XXX = try to retransmit the message xxx times if an error occurs.
0x12	CEC_RETRY_SFT[3:0]	0001**0011**	R/W	Signal free time if the device is retrying for a transmission. This parameter must be set to a value equal to or greater than 3 and strictly less than 5.
0x13	CEC_TX_SFT[3:0]	**0101**0111	R/W	Signal free time if the device is a new initiator. This parameter must be set to a value equal to or greater than 5 and strictly less than 7.
0x13	CEC_TX_SFT[3:0]	0101**0111**	R/W	Signal free time if the device transmits a next frame immediately after its previous frame. This parameter must be set to a value equal to or greater than 7 and strictly less than 10.

Address	Name	Bits	Access	Description
0x14	CEC_TX_LOWDRIVE_COUNTER[3:0]	**0000**0000	R	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. 0000 = no error condition. XXXX = the number of times the LOWDRIVE error condition was encountered.
0x14	CEC_TX_NACK_COUNTER[3:0]	0000**0000**	R	The number of times that the no acknowledge 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. 0000 = no error condition. XXXX = the number of times the no acknowledge error condition was encountered.
0x15	CEC_BUF0_RX_FRAME_HEADER[7:0]	**00000000**	R	Header block of the received frame stored in Rx Frame Buffer 0.
0x16	CEC_BUF0_RX_FRAME_DATA0[7:0]	**00000000**	R	Opcode block of the received frame stored in Rx Frame Buffer 0.
0x17	CEC_BUF0_RX_FRAME_DATA1[7:0]	**00000000**	R	Operand 1 of the received frame stored in Rx Frame Buffer 0.
0x18	CEC_BUF0_RX_FRAME_DATA2[7:0]	**00000000**	R	Operand 2 of the received frame stored in Rx Frame Buffer 0.
0x19	CEC_BUF0_RX_FRAME_DATA3[7:0]	**00000000**	R	Operand 3 of the received frame in Rx Frame Buffer 0.
0x1A	CEC_BUF0_RX_FRAME_DATA4[7:0]	**00000000**	R	Operand 4 of the received frame stored in Rx Frame Buffer 0.
0x1B	CEC_BUF0_RX_FRAME_DATA5[7:0]	**00000000**	R	Operand 5 of the received frame stored in Rx Frame Buffer 0.
0x1C	CEC_BUF0_RX_FRAME_DATA6[7:0]	**00000000**	R	Operand 6 of the received frame stored in Rx Frame Buffer 0.
0x1D	CEC_BUF0_RX_FRAME_DATA7[7:0]	**00000000**	R	Operand 7 of the received frame stored in Rx Frame Buffer 0.
0x1E	CEC_BUF0_RX_FRAME_DATA8[7:0]	**00000000**	R	Operand 8 of the received frame stored in Rx Frame Buffer 0.
0x1F	CEC_BUF0_RX_FRAME_DATA9[7:0]	**00000000**	R	Operand 9 of the received frame stored in Rx Frame Buffer 0.
0x20	CEC_BUF0_RX_FRAME_DATA10[7:0]	**00000000**	R	Operand 10 of the received frame stored in Rx Frame Buffer 0.
0x21	CEC_BUF0_RX_FRAME_DATA11[7:0]	**00000000**	R	Operand 11 of the received frame stored in Rx Frame Buffer 0.
0x22	CEC_BUF0_RX_FRAME_DATA12[7:0]	**00000000**	R	Operand 12 of the received frame stored in Rx Frame Buffer 0.
0x23	CEC_BUF0_RX_FRAME_DATA13[7:0]	**00000000**	R	Operand 13 of the received frame stored in Rx Frame Buffer 0.
0x24	CEC_BUF0_RX_FRAME_DATA14[7:0]	**00000000**	R	Operand 14 of the received frame stored in Rx Frame Buffer 0.
0x25	CEC_BUF0_RX_FRAME_LENGTH[4:0]	000**00000**	R	XXXXX = the total number of bytes (including header byte) that were received in Rx Frame Buffer 0.
0x27	CEC_LOGICAL_ADDRESS_MASK[2:0]	0**001**0000	R/W	Logical Address mask of the CEC logical devices. Up to three logical devices are supported. When the mask bits are set for a particular logical device, the logical device is enabled and messages whose destination address matches that of the selected logical address is accepted. Bit 4 = mask bit for Logical Device 0. Bit 5 = mask bit for Logical Device 1. Bit 6 = mask bit for Logical Device 2.

Address	Name	Bits	Access	Description
0x27	CEC_ERROR_REPORT_MODE	0001**0**000	R/W	Error report mode. 0 = only report short bit period errors. 1 = report both short and long bit period errors.
0x27	CEC_ERROR_DET_MODE	00010**0**00	R/W	Error detection mode. 0 = if any short bit period error, except for start bit, is detected, the CEC controller immediately drives the CEC line low for 3.6 ms. 1 = if a short bit period is detected in the data block where the destination is the CEC section or a target CEC device, the CEC controller immediately drives the CEC line low for 3.6 ms.
0x27	CEC_FORCE_NACK	000100**0**0	R/W	Force the no acknowledge control. Setting this bit forces the CEC controller not acknowledge any received messages. 0 = acknowledge received messages. 1 = do not acknowledge received messages.
0x27	CEC_FORCE_IGNORE	00010000**0**	R/W	Force ignore control. Setting this bit forces the CEC controller to ignore any directly addressed messages. Normal operation must be kept for the broadcast message. 0 = do not ignore directly address messages. 1 = ignore any directly addressed message.
0x28	CEC_LOGICAL_ADDRESS1[3:0]	**1111**1111	R/W	Logical Address 1. This address must be enabled by setting CEC_LOGICAL_ADDRESS_MASK[1] to 1. 1111 = default value. XXXX = user specified logical address.
0x28	CEC_LOGICAL_ADDRESS0[3:0]	1111**1111**	R/W	Logical Address 0. This address must be enabled by setting CEC_LOGICAL_ADDRESS_MASK[0] to 1. 1111 = default value. XXXX = user specified logical address.
0x29	CEC_LOGICAL_ADDRESS2[3:0]	0000**1111**	R/W	Logical Address 2. This address must be enabled by setting CEC_LOGICAL_ADDRESS_MASK[2] to 1. 1111 = default value. XXXX = user specified logical address.
0x2A	CEC_POWER_UP	0011111**0**	R/W	Power mode of the CEC module. 0 = power down the CEC module. 1 = power up the CEC module.
0x2B	CEC_GLITCH_FILTER_CTRL[5:0]	00**000111**	R/W	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. Pulse widths less than the minimum specified width are considered glitches and are removed by the filter. 000000 = disable the glitch filter. 000001 = filter out pulses with a width less than 1 clock cycle. 000010 = filter out pulses with a width less than 2 clock cycles. … 000111 = filter out pulses with a width less than 7 clock cycles. … 111111 = filter out pulses with a width less than 63 clock cycles.
0x2C	CEC_CLR_RX_RDY2	0000**0**000	SC	Clear control for CEC_RX_RDY2. 0 = retain the value of the CEC_RX_RDY2 flag. 1 = clear the value of the CEC_RX_RDY2 flag.

Address	Name	Bits	Access	Description
0x2C	CEC_CLR_RX_RDY1	00000**0**00	SC	Clear control for CEC_RX_RDY1. 0 = retain the value of the CEC_RX_RDY1 flag. 1 = clear the value of the CEC_RX_RDY1 flag.
0x2C	CEC_CLR_RX_RDY0	000000**0**0	SC	Clear control for CEC_RX_RDY0. 0 = retain the value of the CEC_RX_RDY0 flag. 1 = clear the value of the CEC_RX_RDY0 flag.
0x2C	CEC_SOFT_RESET	0000000**0**	SC	CEC module software reset. 0 = no function. 1 = reset the CEC module.
0x4C	CEC_DIS_AUTO_MODE	00000**0**00	R/W	A control to disable the automatic CEC power-up feature when in chip power-down mode. 0 = automatic power-up feature enabled. 1 = automatic power-up feature disabled.
0x53	CEC_BUF2_TIMESTAMP[1:0]	00**00**0000	R	Time stamp for frame stored in Receiver Frame Buffer 2. This can be used to determine which frame must be read next from the receiver frame buffers. 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.
0x53	CEC_BUF1_TIMESTAMP[1:0]	0000**00**00	R	Time stamp for frame stored in Receiver Frame Buffer 1. This can be used to determine which frame must be read next from the receiver frame buffers. 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.
0x53	CEC_BUF0_TIMESTAMP[1:0]	000000**00**	R	Time stamp for frame stored in Receiver Frame Buffer 0. This can be used to determine which frame must be read next from the receiver frame buffers. 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.
0x54	CEC_BUF1_RX_FRAME_HEADER[7:0]	**00000000**	R	Header block of the received frame in Receiver Frame Buffer 1.
0x55	CEC_BUF1_RX_FRAME_DATA0[7:0]	**00000000**	R	Opcode block of the received frame in Receiver Frame Buffer 1.
0x56	CEC_BUF1_RX_FRAME_DATA1[7:0]	**00000000**	R	Operand 1 of the received frame in Rx Frame Buffer 1.
0x57	CEC_BUF1_RX_FRAME_DATA2[7:0]	**00000000**	R	Operand 2 of the received frame in Rx Frame Buffer 1.
0x58	CEC_BUF1_RX_FRAME_DATA3[7:0]	**00000000**	R	Operand 3 of the received frame in Rx Frame Buffer 1.
0x59	CEC_BUF1_RX_FRAME_DATA4[7:0]	**00000000**	R	Operand 4 of the received frame in Rx Frame Buffer 1.
0x5A	CEC_BUF1_RX_FRAME_DATA5[7:0]	**00000000**	R	Operand 5 of the received frame in Rx Frame Buffer 1.
0x5B	CEC_BUF1_RX_FRAME_DATA6[7:0]	**00000000**	R	Operand 6 of the received frame in Rx Frame Buffer 1.

Address	Name	Bits	Access	Description
0x5C	CEC_BUF1_RX_FRAME_DATA7[7:0]	**00000000**	R	Operand 7 of the received frame in Rx Frame Buffer 1.
0x5D	CEC_BUF1_RX_FRAME_DATA8[7:0]	**00000000**	R	Operand 8 of the received frame in Rx Frame Buffer 1.
0x5E	CEC_BUF1_RX_FRAME_DATA9[7:0]	**00000000**	R	Operand 9 of the received frame in Rx Frame Buffer 1.
0x5F	CEC_BUF1_RX_FRAME_DATA10[7:0]	**00000000**	R	Operand 10 of the received frame in Rx Frame Buffer 1.
0x60	CEC_BUF1_RX_FRAME_DATA11[7:0]	**00000000**	R	Operand 11 of the received frame in Rx Frame Buffer 1.
0x61	CEC_BUF1_RX_FRAME_DATA12[7:0]	**00000000**	R	Operand 12 of the received frame in Rx Frame Buffer 1.
0x62	CEC_BUF1_RX_FRAME_DATA13[7:0]	**00000000**	R	Operand 13 of the received frame in Rx Frame Buffer 1.
0x63	CEC_BUF1_RX_FRAME_DATA14[7:0]	**00000000**	R	Operand 14 of the received frame in Rx Frame Buffer 1.
0x64	CEC_BUF1_RX_FRAME_LENGTH[4:0]	000**00000**	R	XXXXX = the total number of bytes (including header byte) that were received in Rx Frame Buffer 1.
0x65	CEC_BUF2_RX_FRAME_HEADER[7:0]	**00000000**	R	Header block of the received frame in Rx Frame Buffer 2.
0x66	CEC_BUF2_RX_FRAME_DATA0[7:0]	**00000000**	R	Opcode block of the received frame in Rx Frame Buffer 2.
0x67	CEC_BUF2_RX_FRAME_DATA1[7:0]	**00000000**	R	Operand 1 of the received frame in Rx Frame Buffer 2.
0x68	CEC_BUF2_RX_FRAME_DATA2[7:0]	**00000000**	R	Operand 2 of the received frame in Rx Frame Buffer 2.
0x69	CEC_BUF2_RX_FRAME_DATA3[7:0]	**00000000**	R	Operand 3 of the received frame in Rx Frame Buffer 2.
0x6A	CEC_BUF2_RX_FRAME_DATA4[7:0]	**00000000**	R	Operand 4 of the received frame in Rx Frame Buffer 2.
0x6B	CEC_BUF2_RX_FRAME_DATA5[7:0]	**00000000**	R	Operand 5 of the received frame in Rx Frame Buffer 2.
0x6C	CEC_BUF2_RX_FRAME_DATA6[7:0]	**00000000**	R	Operand 6 of the received frame in Rx Frame Buffer 2.
0x6D	CEC_BUF2_RX_FRAME_DATA7[7:0]	**00000000**	R	Operand 7 of the received frame in Rx Frame Buffer 2.
0x6E	CEC_BUF2_RX_FRAME_DATA8[7:0]	**00000000**	R	Operand 8 of the received frame in Rx Frame Buffer 2.
0x6F	CEC_BUF2_RX_FRAME_DATA9[7:0]	**00000000**	R	Operand 9 of the received frame in Rx Frame Buffer 2.
0x70	CEC_BUF2_RX_FRAME_DATA10[7:0]	**00000000**	R	Operand 10 of the received frame in Rx Frame Buffer 2.
0x71	CEC_BUF2_RX_FRAME_DATA11[7:0]	**00000000**	R	Operand 11 of the received frame in Rx Frame Buffer 2.
0x72	CEC_BUF2_RX_FRAME_DATA12[7:0]	**00000000**	R	Operand 12 of the received frame in Rx Frame Buffer 2.
0x73	CEC_BUF2_RX_FRAME_DATA13[7:0]	**00000000**	R	Operand 13 of the received frame in Rx Frame Buffer 2.
0x74	CEC_BUF2_RX_FRAME_DATA14[7:0]	**00000000**	R	Operand 14 of the received frame in Rx Frame Buffer 2.
0x75	CEC_BUF2_RX_FRAME_LENGTH[4:0]	000**00000**	R	XXXXX = the total number of bytes (including header byte) that were received in Rx Frame Buffer 2.
0x76	CEC_RX_RDY2	00000**0**00	R	Flag that a CEC frame has been received and is waiting to be read in Rx Frame Buffer 2. This flag must be cleared via CEC_CLR_RX_RDY2 before another message can be received in Rx Frame Buffer 2. 0 = no CEC frame available in Rx Frame Buffer 2. 1 = a CEC frame is available in Rx Frame Buffer 2.
0x76	CEC_RX_RDY1	000000**0**0	R	Flag that a CEC frame has been received and is waiting to be read in Rx Frame Buffer 2. This flag must be cleared via CEC_CLR_RX_RDY1 before another message can be received in Rx Frame Buffer 1. 0 = no CEC frame available in Rx Frame Buffer 1. 1 = a CEC frame is available in Rx Frame Buffer 1.
0x76	CEC_RX_RDY0	0000000**0**	R	Flag that a CEC frame has been received and is waiting to be read in Rx Frame Buffer 2. This flag must be cleared via CEC_CLR_RX_RDY0 before another message can be received in Rx Frame Buffer 0. 0 = no CEC frame available in Rx Frame Buffer 0. 1 = a CEC frame is available in Rx Frame Buffer 0.
0x77	CEC_USE_ALL_BUFS	0000000**0**	R/W	Control to enable supplementary receiver frame buffers. 0 = use only Buffer 0 to store CEC frames. 1 = use all three buffers to stores the CEC frames.

Address	Name	Bits	Access	Description
0x78	CEC_WAKE_OPCODE0[7:0]	**01101101**	R/W	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. 01101101 = power on. XXXXXXXX = user specified opcode to respond to.
0x79	CEC_WAKE_OPCODE1[7:0]	**10001111**	R/W	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. 10001111 = give power status. XXXXXXXX = user specified opcode to respond to.
0x7A	CEC_WAKE_OPCODE2[7:0]	**10000010**	R/W	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. 10000010 = active source. XXXXXXXX = user specified opcode to respond to.
0x7B	CEC_WAKE_OPCODE3[7:0]	**00000100**	R/W	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. 00000100 = image view on. XXXXXXXX = user specified opcode to respond to
0x7C	CEC_WAKE_OPCODE4[7:0]	**00001101**	R/W	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. 00001101 = text view on. XXXXXXXX = user specified opcode to respond to.
0x7D	CEC_WAKE_OPCODE5[7:0]	**01110000**	R/W	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. 01110000 = system audio mode request. XXXXXXXX = user specified opcode to respond to
0x7E	CEC_WAKE_OPCODE6[7:0]	**01000010**	R/W	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. 01000010 = deck control. XXXXXXXX = user specified opcode to respond to
0x7F	CEC_WAKE_OPCODE7[7:0]	**01000001**	R/W	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. 01000001 = play. XXXXXXXX = user specified opcode to respond to.

OPENLDI Tx REGISTER MAP BIT DESCRIPTIONS

Table 16. OpenLDI Tx Register Map Bit Descriptions

Address	Name	Bits	Access	Description
0x40	TX_PLL_EN	0000**0**010	R/W	Enables the PLL in the OpenLDI transmitter. 0 = PLL disabled. 1 = PLL enabled.
0x40	TX_PDN	000000**1**0	R/W	Powers down the LVDS transmitter. 0 = powered up. 1 = powered down.
0x40	TX_MODE_ITU656	0000001**0**	R/W	Sets the Tx mode of operation: 7 bits for OLDI, 8 bits for ITU 656. 0 = 7 bits (OpenLDI). 1 = 8 bits (ITU 656).
0x44	TX_PLL_GEAR[2:0]	00000**000**	R/W	Sets the PLL gear to be used. 000 = 0 MHz to 200 MHz. 010 = 200 MHz to 300 MHz.
0x4C	TX_OLDI_HS_POL	0**1**110001	R/W	Sets the HS polarity for the output OpenLDI data.
0x4C	TX_OLDI_VS_POL	01**1**10001	R/W	Sets the VS polarity for the output OpenLDI data.
0x4C	TX_OLDI_DE_POL	011**1**0001	R/W	Sets the DE polarity for the output OpenLDI data.
0x4C	TX_ENABLE_NS_MAPPING	0111**0**001	R/W	Enables alternative bit mapping in the OpenLDI 8-bit nonbalanced modes.
0x4C	TX_656_ALL_LANES_ENABLE	01110**0**01	R/W	Enables all data lanes to carry ITU 656 output data
0x4C	TX_OLDI_BALANCED_MODE	011100**0**1	R/W	Enables dc balance in Open LDI mode. 0 = non-dc balanced. 1 = dc balanced.
0x4C	TX_COLOR_MODE	01110001	R/W	Selects the color depth to be encoded in Open LDI mode. 0 = 6 bits. 1 = 8 bits.
0x4D	TX_CLOCK_LANE_CODE_656[7:0]	**11100011**	R/W	Specifies the clock lane code to be sent in ITU 656 out mode.
0x4E	TX_INT_RES	0000100**0**	R/W	Sets the value for reserved bit control.
0x4E	TX_MUX_INT_RES	00001000	R/W	Enables programming of the reserved bit value in 8-bit color depth modes.
0x4F	TX_PLL_LOCK_DET	00000**0**00	R	Readback of the LVDS PLL lock detect.
0x4F	TX_DETECTED_VS_POL	000000**0**0	R	Readback of the detected VS signal polarity into OpenLDI Tx.
0x4F	TX_DETECTED_HS_POL	00000000	R	Readback of the detected HS signal polarity into OpenLDI Tx.

I²C refers to a communications protocol originally developed by Philips Semiconductors (now NXP Semiconductors).