DATA SHEET

SAA7146A
Multimedia bridge, high performance Scaler and PCI circuit (SPCI)

Product specification
File under Integrated Circuits, IC22

1998 Apr 09

Philips
Semiconductors

PHILIPS

Multimedia bridge, high performance Scaler and PCI circuit (SPCI)

SAA7146A

CONTENTS

Multimedia bridge, high performance Scaler and PCI circuit (SPCI)

SAA7146A

1 FEATURES

1.1 Video processing

- Full size, full speed video delivery to and from the frame buffer or virtual system memory enables various processing possibilities for any external PCI device
- Full bandwidth PCI-bus master write and read (up to 132 Mbytes/s)
- Virtual memory support (4 Mbytes per DMA channel)
- Processing of maximum 4095 active samples per line and maximum 4095 lines per frame
- Vanity picture (mirror) for video phone and video conferencing applications
- Video flip (upside down picture)
- Colour space conversion with gamma correction for different kinds of displays
- Chroma Key generation and utilization
- Pixel dithering for low resolution video output formats
- Brightness, contrast and saturation control
- Video and Vertical Blanking Interval (VBI) synchronized programming of internal registers with Register Programming Sequencer (RPS), ability to control two asynchronous data streams simultaneously
- Memory Management Unit (MMU) supports virtual demand paging memory management (Windows, Unix, etc.)
- Rectangular clipping of frame buffer areas minimizes PCI-bus load
- Random shape mask clipping protects selectable areas of frame buffer
- 3×128 Dword video FIFO with overflow detection and 'graceful' recovery.

1.2 Audio processing

- Time Slot List (TSL) processing for flexible control of audio frames up to 256 bits on 2 asynchronous bidirectional digital audio interfaces simultaneously (4 DMA channels)
- Video synchronous audio capture, e.g. for sound cards
- Various synchronization modes to support I^2S and other different audio and DSP data formats
- Audio input level monitoring enables peak control via software
- Programmable bit clock generation for master and slave applications.

1.3 Scaling

- Scaling of video pictures down to randomly sized windows (vertical down to 1 : 1024; horizontal down to 1 : 256)
- High Performance Scaler (HPS) offers two-dimensional, phase correct data processing for improved signal quality of scaled video data, especially for compression applications
- Horizontal and vertical FIR filters with up to 65 taps
- Horizontal upscaling (zoom) supports e.g. CCIR to square pixel conversion
- Additional Binary Ratio Scaler (BRS) supports CIF and QCIF formats, especially for video phone and video conferencing.

1.4 Interfacing

- Dual D1 (8-bit, CCIR 656) video I/O interface
- DMSD2 compatible (16-bit YUV) video input interface
- Supports various packed (pixel dithering) and planar video output formats
- Data Expansion Bus Interface (DEBI) for interfacing with e.g. MPEG or JPEG decoders with Intel (ISA like) and Motorola (68000 like) protocol style, capability for immediate and block mode (DMA) transfers with up to 23 Mbytes/s peak data rate
- 5 digital audio I/O ports
- 4 independent user configurable General Purpose I/O Ports (GPI/O) for interrupt and status processing
- PCI interface (release 2.1)
- I^2C-bus interface (bus master).

1.5 General

- Subsystem (board) vendor ID support for board identification via software driver
- Internal arbitration control
- Diagnostic support and event analysis
- Programmable Vertical Blanking Interval (VBI) data region for e.g. to support INTERCAST, teletext, closed caption and similar applications
- 3.3 V supply enables reduced power consumption, 5 V tolerant I/Os for 5 V PCI signalling environment.

2 GENERAL DESCRIPTION

The SAA7146A, Multimedia PCI-bridge, is a highly integrated circuit for DeskTop Video (DTV) applications. The device provides a number of interface ports that enable a wide variety of video and audio ICs to be connected to the PCI-bus thus supporting a number of video applications in a PC. One example of the application capabilities is shown in Fig.49.
Figure 1 shows the various interface ports and the main internal function blocks.

3 QUICK REFERENCE DATA

SYMBOL	PARAMETER	MIN.	TYP.	MAX.	UNIT
V_{DDD}	digital supply voltage	3.0	3.3	3.6	V
$I_{DDD(tot)}$	total digital supply current	–	400	–	mA
V_i; V_o	data input/output levels	TTL compatible			
f_{LLC}	LLC input clock frequency	–	–	32	MHz
f_{PCI}	PCI input clock frequency	–	–	33	MHz
f_{I2S}	I^2S input clock frequency	–	–	12.5	MHz
T_{amb}	operating ambient temperature	0	–	70	°C

4 ORDERING INFORMATION

TYPE NUMBER	PACKAGE		
	NAME	DESCRIPTION	VERSION
SAA7146AH	QFP160	plastic quad flat package; 160 leads (lead length 1.95 mm); body 28 × 28 × 3.4 mm; high stand-off height	SOT322-1
SAA7146AHZ	SQFP208	plastic shrink quad flat package; 208 leads (lead length 1.3 mm); body 28 × 28 × 3.4 mm	SOT316-1

5 BLOCK DIAGRAM

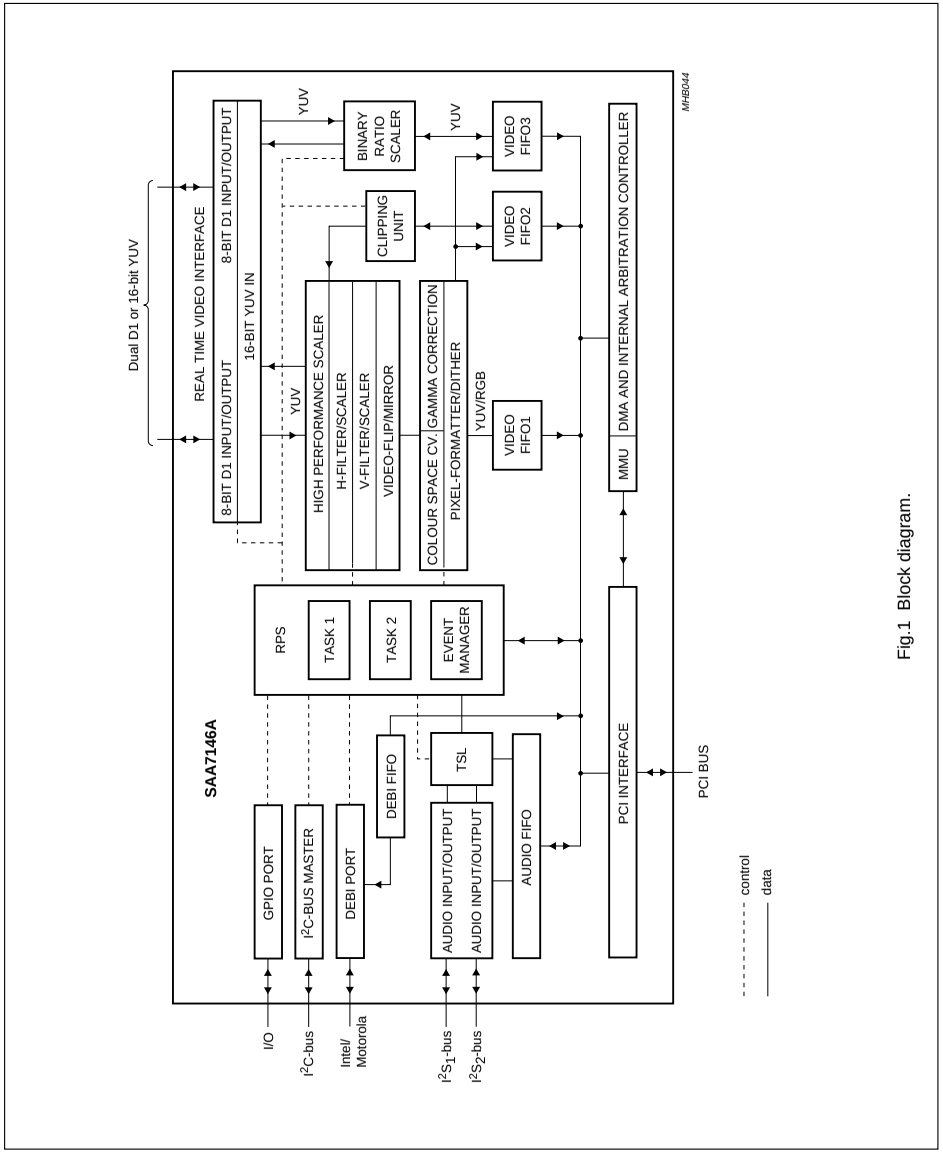


Fig.1 Block diagram.

6 PINNING

Pin description for QFP160

SYMBOL	PIN	STATUS	DESCRIPTION
D1_A0	1	I/O	bidirectional digital CCIR 656 D1 port A bit 0
D1_A1	2	I/O	bidirectional digital CCIR 656 D1 port A bit 1
D1_A2	3	I/O	bidirectional digital CCIR 656 D1 port A bit 2
D1_A3	4	I/O	bidirectional digital CCIR 656 D1 port A bit 3
V_{DDD1}	5	P	digital supply voltage 1 (3.3 V)
V_{SSD1}	6	P	digital ground 1
D1_A4	7	I/O	bidirectional digital CCIR 656 D1 port A bit 4
D1_A5	8	I/O	bidirectional digital CCIR 656 D1 port A bit 5
D1_A6	9	I/O	bidirectional digital CCIR 656 D1 port A bit 6
D1_A7	10	I/O	bidirectional digital CCIR 656 D1 port A bit 7
VS_A	11	I/O	bidirectional vertical sync signal port A
HS_A	12	I/O	bidirectional horizontal sync signal port A
LLC_A	13	I/O	bidirectional line-locked system clock port A
PXQ_A	14	I/O	bidirectional pixel qualifier signal to mark valid pixels port A; note 1
V_{DDD2}	15	P	digital supply voltage 2 (3.3 V)
V_{SSD2}	16	P	digital ground 2
$\overline{TRST}$	17	I	test reset input (JTAG pin must be set LOW for normal operation)
TMS	18	I	test mode select input (JTAG pin must be floating or set to HIGH during normal operation)
TCLK	19	I	test clock input (JTAG pin should be set LOW during normal operation)
TDO	20	O	test data output (JTAG pin not active during normal operation)
TDI	21	I	test data input (JTAG pin must be floating or set to HIGH during normal operation)
V_{DDD3}	22	P	digital supply voltage 3 (3.3 V)
V_{SSD3}	23	P	digital ground 3
INTA#	24	O	PCI interrupt line output (active LOW)
$\overline{RST}$	25	I	PCI global reset input (active LOW)
CLK	26	I	PCI clock input
GNT#	27	I	bus grant input signal, PCI arbitration signal (active LOW)
REQ#	28	O	bus request output signal, PCI arbitration signal (active LOW)
V_{DDD4}	29	P	digital supply voltage 4 (3.3 V)
V_{SSD4}	30	P	digital ground 4
AD PCI31	31	I/O	bidirectional PCI multiplexed address/data bit 31
AD PCI30	32	I/O	bidirectional PCI multiplexed address/data bit 30
AD PCI29	33	I/O	bidirectional PCI multiplexed address/data bit 29
AD PCI28	34	I/O	bidirectional PCI multiplexed address/data bit 28
V_{DDD5}	35	P	digital supply voltage 5 (3.3 V)
V_{SSD5}	36	P	digital ground 5
AD PCI27	37	I/O	bidirectional PCI multiplexed address/data bit 27

SYMBOL	PIN	STATUS	DESCRIPTION
AD PCI26	38	I/O	bidirectional PCI multiplexed address/data bit 26
AD PCI25	39	I/O	bidirectional PCI multiplexed address/data bit 25
AD PCI24	40	I/O	bidirectional PCI multiplexed address/data bit 24
C/BE# [3]	41	I/O	bidirectional PCI multiplexed bus command and byte enable 3 (active LOW)
IDSEL	42	I	PCI initialization device select signal input
AD PCI23	43	I/O	bidirectional PCI multiplexed address/data bit 23
AD PCI22	44	I/O	bidirectional PCI multiplexed address/data bit 22
AD PCI21	45	I/O	bidirectional PCI multiplexed address/data bit 21
AD PCI20	46	I/O	bidirectional PCI multiplexed address/data bit 20
V_{DDD6}	47	P	digital supply voltage 6 (3.3 V)
V_{SSD6}	48	P	digital ground 6
AD PCI19	49	I/O	bidirectional PCI multiplexed address/data bit 19
AD PCI18	50	I/O	bidirectional PCI multiplexed address/data bit 18
AD PCI17	51	I/O	bidirectional PCI multiplexed address/data bit 17
AD PCI16	52	I/O	bidirectional PCI multiplexed address/data bit 16
V_{DDD7}	53	P	digital supply voltage 7 (3.3 V)
V_{SSD7}	54	P	digital ground 7
C/BE# [2]	55	I/O	bidirectional PCI multiplexed bus command and byte enable 2 (active LOW)
FRAME#	56	I/O	bidirectional PCI cycle frame signal (active LOW)
IRDY#	57	I/O	bidirectional PCI initiator ready signal (active LOW)
TRDY#	58	I/O	bidirectional PCI target ready signal (active LOW)
DEVSEL#	59	I/O	bidirectional PCI device select signal (active LOW)
STOP#	60	I/O	bidirectional PCI stop signal (active LOW)
PERR#	61	O	PCI parity error signal output (active LOW)
PAR	62	I/O	bidirectional PCI parity signal
C/BE# [1]	63	I/O	bidirectional PCI-bus command and byte enable 1 (active LOW)
V_{DDD8}	64	P	digital supply voltage 8 (3.3 V)
V_{SSD8}	65	P	digital ground 8
AD PCI15	66	I/O	bidirectional PCI multiplexed address/data bit 15
AD PCI14	67	I/O	bidirectional PCI multiplexed address/data bit 14
AD PCI13	68	I/O	bidirectional PCI multiplexed address/data bit 13
AD PCI12	69	I/O	bidirectional PCI multiplexed address/data bit 12
V_{DDD9}	70	P	digital supply voltage 9 (3.3 V)
V_{SSD9}	71	P	digital ground 9
AD PCI11	72	I/O	bidirectional PCI multiplexed address/data bit 11
AD PCI10	73	I/O	bidirectional PCI multiplexed address/data bit 10
AD PCI9	74	I/O	bidirectional PCI multiplexed address/data bit 9
AD PCI8	75	I/O	bidirectional PCI multiplexed address/data bit 8
V_{DDD10}	76	P	digital supply voltage 10 (3.3 V)
V_{SSD10}	77	P	digital ground 10
C/BE# [0]	78	I/O	bidirectional PCI multiplexed bus command and byte enable 0 (active LOW)

SYMBOL	PIN	STATUS	DESCRIPTION
AD PCI7	79	I/O	bidirectional PCI multiplexed address/data bit 7
AD PCI6	80	I/O	bidirectional PCI multiplexed address/data bit 6
V_{SSD11}	81	P	digital ground 11
AD PCI5	82	I/O	bidirectional PCI multiplexed address/data bit 5
AD PCI4	83	I/O	bidirectional PCI multiplexed address/data bit 4
AD PCI3	84	I/O	bidirectional PCI multiplexed address/data bit 3
AD PCI2	85	I/O	bidirectional PCI multiplexed address/data bit 2
V_{DDD11}	86	P	digital supply voltage 11 (3.3 V)
V_{SSD12}	87	P	digital ground 12
AD PCI1	88	I/O	bidirectional PCI multiplexed address/data bit 1
AD PCI0	89	I/O	bidirectional PCI multiplexed address/data bit 0
V_{DDD12}	90	P	digital supply voltage 12 (3.3 V)
V_{SSD13}	91	P	digital ground 13
AD15	92	I/O	bidirectional DEBI multiplexed address data line bit 15
AD14	93	I/O	bidirectional DEBI multiplexed address data line bit 14
AD13	94	I/O	bidirectional DEBI multiplexed address data line bit 13
AD12	95	I/O	bidirectional DEBI multiplexed address data line bit 12
V_{DDD13}	96	P	digital supply voltage 13 (3.3 V)
V_{SSD14}	97	P	digital ground 14
AD11	98	I/O	bidirectional DEBI multiplexed address data line bit 11
AD10	99	I/O	bidirectional DEBI multiplexed address data line bit 10
AD9	100	I/O	bidirectional DEBI multiplexed address data line bit 9
AD8	101	I/O	bidirectional DEBI multiplexed address data line bit 8
V_{DDD14}	102	P	digital supply voltage 14 (3.3 V)
V_{SSD15}	103	P	digital ground 15
RWN_SBHE	104	O	DEBI data transfer control signal output (read write not/system byte high enable)
AS_ALE	105	O	DEBI address strobe and address latch enable output
LDS_RDN	106	O	lower data strobe/read not output
UDS_WRN	107	O	upper data strobe/write not output
DTACK_RDY	108	I	DEBI data transfer acknowledge or ready input
V_{DDD15}	109	P	digital supply voltage 15 (3.3 V)
V_{SSD16}	110	P	digital ground 16
AD0	111	I/O	bidirectional DEBI multiplexed address data line bit 0
AD1	112	I/O	bidirectional DEBI multiplexed address data line bit 1
AD2	113	I/O	bidirectional DEBI multiplexed address data line bit 2
AD3	114	I/O	bidirectional DEBI multiplexed address data line bit 3
V_{DDD16}	115	P	digital supply voltage 16 (3.3 V)
V_{SSD17}	116	P	digital ground 17
AD4	117	I/O	bidirectional DEBI multiplexed address data line bit 4
AD5	118	I/O	bidirectional DEBI multiplexed address data line bit 5

SYMBOL	PIN	STATUS	DESCRIPTION
AD6	119	I/O	bidirectional DEBI multiplexed address data line bit 6
AD7	120	I/O	bidirectional DEBI multiplexed address data line bit 7
WS0	121	I/O	bidirectional word select signal for audio interface A1
SD0	122	I/O	bidirectional serial data for audio interface A1
BCLK1	123	I/O	bidirectional bit clock for audio interface A1
WS1	124	O	word select output signal for audio interface A1/A2
SD1	125	I/O	bidirectional serial data for audio interface A1/A2
WS2	126	O	word select output signal for audio interface A1/A2
SD2	127	I/O	bidirectional serial data for audio interface A1/A2
V_{DDD17}	128	P	digital supply voltage 17 (3.3 V)
V_{SSD18}	129	P	digital ground 18
WS3	130	O	word select output signal for audio interface A1/A2
SD3	131	I/O	bidirectional serial data for audio interface A1/A2
BCLK2	132	I/O	bidirectional bit clock for audio interface A2
WS4	133	I/O	bidirectional word select signal for audio interface A2
SD4	134	I/O	bidirectional serial data for audio interface A2
ACLK	135	I	audio reference clock input signal
SCL	136	I/O	bidirectional I^2C-bus clock line
SDA	137	I/O	bidirectional I^2C-bus data line
V_{DDD18}	138	P	digital supply voltage 18 (3.3 V)
V_{DDI2C}	139	I	I^2C-bus voltage sense input; see note 3 of “Characteristics”
V_{SSD19}	140	P	digital ground 19
GPIO3	141	I/O	general purpose I/O signal 3
GPIO2	142	I/O	general purpose I/O signal 2
GPIO1	143	I/O	general purpose I/O signal 1
GPIO0	144	I/O	general purpose I/O signal 0
D1_B0	145	I/O	bidirectional digital CCIR 656 D1 port B bit 0
D1_B1	146	I/O	bidirectional digital CCIR 656 D1 port B bit 1
D1_B2	147	I/O	bidirectional digital CCIR 656 D1 port B bit 2
D1_B3	148	I/O	bidirectional digital CCIR 656 D1 port B bit 3
V_{DDD19}	149	P	digital supply voltage 19 (3.3 V)
V_{SSD20}	150	P	digital ground 20
D1_B4	151	I/O	bidirectional digital CCIR 656 D1 port B bit 4
D1_B5	152	I/O	bidirectional digital CCIR 656 D1 port B bit 5
D1_B6	153	I/O	bidirectional digital CCIR 656 D1 port B bit 6
D1_B7	154	I/O	bidirectional digital CCIR 656 D1 port B bit 7
V_{DDD20}	155	P	digital supply voltage 20 (3.3 V)
V_{SSD21}	156	P	digital ground 21
LLC_B	157	I/O	bidirectional line-locked system clock port B
VS_B	158	I/O	bidirectional vertical sync signal port B

SYMBOL	PIN	STATUS	DESCRIPTION
HS_B	159	I/O	bidirectional horizontal sync signal port B
PXQ_B	160	I/O	bidirectional pixel qualifier signal to mark valid pixels port B; note 2

Notes

1. For continuous CCIR 656 format at the D1_A port this pin must be set HIGH.
2. For continuous CCIR 656 format at the D1_B port this pin must be set HIGH.

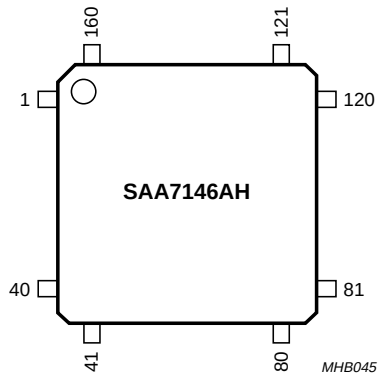


Fig.2 Pin configuration SAA7146AH (QFP160).

Pin description for SQFP208

SYMBOL	PIN	STATUS	DESCRIPTION
V_{SSD0}	1	P	digital ground 0
D1_A0	2	I/O	bidirectional digital CCIR 656 D1 port A bit 0
D1_A1	3	I/O	bidirectional digital CCIR 656 D1 port A bit 1
D1_A2	4	I/O	bidirectional digital CCIR 656 D1 port A bit 2
D1_A3	5	I/O	bidirectional digital CCIR 656 D1 port A bit 3
V_{DDD1}	6	P	digital supply voltage 1 (3.3 V)
n.c.	7	–	reserved pin; not connected internally
V_{SSD1}	8	P	digital ground 1
D1_A4	9	I/O	bidirectional digital CCIR 656 D1 port A bit 4
D1_A5	10	I/O	bidirectional digital CCIR 656 D1 port A bit 5
D1_A6	11	I/O	bidirectional digital CCIR 656 D1 port A bit 6
D1_A7	12	I/O	bidirectional digital CCIR 656 D1 port A bit 7
V_{DDD2}	13	P	digital supply voltage 2 (3.3 V)
n.c.	14	–	reserved pin; not connected internally
V_{SSD2}	15	P	digital ground 2
VS_A	16	I/O	bidirectional vertical sync signal port A
HS_A	17	I/O	bidirectional horizontal sync signal port A
LLC_A	18	I/O	bidirectional line-locked system clock port A
PXQ_A	19	I/O	bidirectional pixel qualifier signal to mark valid pixels port A; note 1
n.c.	20	–	reserved pin; do not connect
V_{DDD3}	21	P	digital supply voltage 3 (3.3 V)
n.c.	22	–	reserved pin; not connected internally
V_{SSD3}	23	P	digital ground 3
$\overline{TRST}$	24	I	test reset input (JTAG pin must be set LOW for normal operation)
TMS	25	I	test mode select input (JTAG pin must be floating or set to HIGH during normal operation)
TCLK	26	I	test clock input (JTAG pin should be set LOW during normal operation)
TDO	27	O	test data output (JTAG pin not active during normal operation)
TDI	28	I	test data input (JTAG pin must be floating or set to HIGH during normal operation)
V_{DDD4}	29	P	digital supply voltage 4 (3.3 V)
n.c.	30	–	reserved pin; not connected internally
V_{SSD4}	31	P	digital ground 4
INTA#	32	O	PCI interrupt line output (active LOW)
RST#	33	I	PCI global reset input (active LOW)
CLK	34	I	PCI clock input
GNT#	35	I	bus grant input signal input, PCI arbitration signal (active LOW)
REQ#	36	O	bus request output signal output, PCI arbitration signal (active LOW)
V_{DDD5}	37	P	digital supply voltage 5 (3.3 V)
n.c.	38	–	reserved pin; not connected internally

SYMBOL	PIN	STATUS	DESCRIPTION
V_{SSD5}	39	P	digital ground 5
AD PCI31	40	I/O	bidirectional PCI multiplexed address/data bit 31
AD PCI30	41	I/O	bidirectional PCI multiplexed address/data bit 30
AD PCI29	42	I/O	bidirectional PCI multiplexed address/data bit 29
AD PCI28	43	I/O	bidirectional PCI multiplexed address/data bit 28
V_{DDD6}	44	P	digital supply voltage 6 (3.3 V)
n.c.	45	–	reserved pin; not connected internally
V_{SSD6}	46	P	digital ground 6
AD PCI27	47	I/O	bidirectional PCI multiplexed address/data bit 27
AD PCI26	48	I/O	bidirectional PCI multiplexed address/data bit 26
AD PCI25	49	I/O	bidirectional PCI multiplexed address/data bit 25
AD PCI24	50	I/O	bidirectional PCI multiplexed address/data bit 24
V_{DDD7}	51	P	digital supply voltage 7 (3.3 V)
n.c.	52	–	reserved pin; do not connect
n.c.	53	–	reserved pin; not connected internally
V_{SSD7}	54	P	digital ground 7
C/BE# [3]	55	I/O	bidirectional PCI multiplexed bus command and byte enable 3 (active LOW)
IDSEL	56	I	PCI initialization device select input signal
AD PCI23	57	I/O	bidirectional PCI multiplexed address/data bit 23
AD PCI22	58	I/O	bidirectional PCI multiplexed address/data bit 22
AD PCI21	59	I/O	bidirectional PCI multiplexed address/data bit 21
AD PCI20	60	I/O	bidirectional PCI multiplexed address/data bit 20
n.c.	61	–	reserved pin; do not connect
n.c.	62	–	reserved pin; not connected internally
V_{SSD8}	63	P	digital ground 8
AD PCI19	64	I/O	bidirectional PCI multiplexed address/data bit 19
AD PCI18	65	I/O	bidirectional PCI multiplexed address/data bit 18
AD PCI17	66	I/O	bidirectional PCI multiplexed address/data bit 17
AD PCI16	67	I/O	bidirectional PCI multiplexed address/data bit 16
V_{DDD8}	68	P	digital supply voltage 8 (3.3 V)
n.c.	69	–	reserved pin; do not connect
V_{SSD9}	70	P	digital ground 9
C/BE# [2]	71	I/O	bidirectional PCI multiplexed bus command and byte enable 2 (active LOW)
FRAME#	72	I/O	bidirectional PCI cycle frame signal (active LOW)
IRDY#	73	I/O	bidirectional PCI initiator ready signal (active LOW)
TRDY#	74	I/O	bidirectional PCI target ready signal (active LOW)
V_{DDD9}	75	P	digital supply voltage 9 (3.3 V)
n.c.	76	–	reserved pin; do not connect
V_{SSD10}	77	P	digital ground 10
DEVSEL#	78	I/O	bidirectional PCI device select signal (active LOW)

SYMBOL	PIN	STATUS	DESCRIPTION
STOP#	79	I/O	bidirectional PCI stop signal (active LOW)
PERR#	80	O	PCI parity error output signal (active LOW)
n.c.	81	–	reserved pin; do not connect
PAR	82	I/O	bidirectional PCI parity signal
C/BE# [1]	83	I/O	bidirectional PCI-bus command and byte enable 1 (active LOW)
V_{DDD10}	84	P	digital supply voltage 10 (3.3 V)
n.c.	85	–	reserved pin; not connected internally
V_{SSD11}	86	P	digital ground 11
AD PCI15	87	I/O	bidirectional PCI multiplexed address/data bit 15
AD PCI14	88	I/O	bidirectional PCI multiplexed address/data bit 14
AD PCI13	89	I/O	bidirectional PCI multiplexed address/data bit 13
AD PCI12	90	I/O	bidirectional PCI multiplexed address/data bit 12
V_{DDD11}	91	P	digital supply voltage 11 (3.3 V)
n.c.	92	–	reserved pin; not connected internally
V_{SSD12}	93	P	digital ground 12
AD PCI11	94	I/O	bidirectional PCI multiplexed address/data bit 11
AD PCI10	95	I/O	bidirectional PCI multiplexed address/data bit 10
AD PCI9	96	I/O	bidirectional PCI multiplexed address/data bit 9
AD PCI8	97	I/O	bidirectional PCI multiplexed address/data bit 8
n.c.	98	–	reserved pin; do not connect
n.c.	99	–	reserved pin; not connected internally
V_{SSD13}	100	P	digital ground 13
C/BE# [0]	101	I/O	bidirectional PCI multiplexed bus command and byte enable (active LOW)
AD PCI7	102	I/O	bidirectional PCI multiplexed address/data bit 7
AD PCI6	103	I/O	bidirectional PCI multiplexed address/data bit 6
V_{DDD12}	104	P	digital supply voltage 12 (3.3 V)
n.c.	105	–	reserved pin; do not connect
n.c.	106	–	reserved pin; not connected internally
V_{SSD14}	107	P	digital ground 14
AD PCI5	108	I/O	bidirectional PCI multiplexed address/data bit 5
AD PCI4	109	I/O	bidirectional PCI multiplexed address/data bit 4
AD PCI3	110	I/O	bidirectional PCI multiplexed address/data bit 3
AD PCI2	111	I/O	bidirectional PCI multiplexed address/data bit 2
V_{DDD13}	112	P	digital supply voltage 13 (3.3 V)
n.c.	113	–	reserved pin; not connected internally
V_{SSD15}	114	P	digital ground 15
AD PCI1	115	I/O	bidirectional PCI multiplexed address/data bit 1
AD PCI0	116	I/O	bidirectional PCI multiplexed address/data bit 0
V_{DDD14}	117	P	digital supply voltage 14 (3.3 V)
n.c.	118	–	reserved pin; not connected internally
V_{SSD16}	119	P	digital ground 16

SYMBOL	PIN	STATUS	DESCRIPTION
AD15	120	I/O	bidirectional DEBI multiplexed address data line bit 15
AD14	121	I/O	bidirectional DEBI multiplexed address data line bit 14
AD13	122	I/O	bidirectional DEBI multiplexed address data line bit 13
AD12	123	I/O	bidirectional DEBI multiplexed address data line bit 12
V_{DDD15}	124	P	digital supply voltage 15 (3.3 V)
n.c.	125	–	reserved pin; not connected internally
V_{SSD17}	126	P	digital ground 17
AD11	127	I/O	bidirectional DEBI multiplexed address data line bit 11
AD10	128	I/O	bidirectional DEBI multiplexed address data line bit 10
AD9	129	I/O	bidirectional DEBI multiplexed address data line bit 9
AD8	130	I/O	bidirectional DEBI multiplexed address data line bit 8
V_{DDD16}	131	P	digital supply voltage 16 (3.3 V)
n.c.	132	–	reserved pin; not connected internally
V_{SSD18}	133	P	digital ground 18
RWN_SBHE	134	O	DEBI data transfer control output signal (read write not/system byte high enable)
AS_ALE	135	O	DEBI address strobe and address latch enable output
LDS_RDN	136	O	lower data strobe/read not output
UDS_WRN	137	O	upper data strobe/write not output
DTACK_RDY	138	I	DEBI data transfer acknowledge or ready input
V_{DDD17}	139	P	digital supply voltage 17 (3.3 V)
n.c.	140	–	reserved pin; not connected internally
V_{SSD19}	141	P	digital ground 19
AD0	142	I/O	bidirectional DEBI multiplexed address data line bit 0
AD1	143	I/O	bidirectional DEBI multiplexed address data line bit 1
AD2	144	I/O	bidirectional DEBI multiplexed address data line bit 2
AD3	145	I/O	bidirectional DEBI multiplexed address data line bit 3
V_{DDD18}	146	P	digital supply voltage 18 (3.3 V)
n.c.	147	–	reserved pin; not connected internally
V_{SSD20}	148	P	digital ground 20
AD4	149	I/O	bidirectional DEBI multiplexed address data line bit 4
AD5	150	I/O	bidirectional DEBI multiplexed address data line bit 5
AD6	151	I/O	bidirectional DEBI multiplexed address data line bit 6
AD7	152	I/O	bidirectional DEBI multiplexed address data line bit 7
n.c.	153	–	reserved pin; do not connect
n.c.	154	–	reserved pin; do not connect
V_{DDD19}	155	P	digital supply voltage 19 (3.3 V)
n.c.	156	–	reserved pin; not connected internally
n.c.	157	–	reserved pin; do not connect
V_{SSD21}	158	P	digital ground 21
WS0	159	I/O	bidirectional word select signal for audio interface A1
SD0	160	I/O	bidirectional serial data for audio interface A1

SYMBOL	PIN	STATUS	DESCRIPTION
BCLK1	161	I/O	bidirectional bit clock for audio interface A1
WS1	162	O	word select output signal for audio interface A1/A2
SD1	163	I/O	bidirectional serial data for audio interface A1/A2
WS2	164	O	word select output signal for audio interface A1/A2
SD2	165	I/O	bidirectional serial data for audio interface A1/A2
V_{DDD20}	166	P	digital supply voltage 20 (3.3 V)
n.c.	167	–	reserved pin; not connected internally
V_{SSD22}	168	P	digital ground 22
WS3	169	O	word select output signal for audio interface A1/A2
SD3	170	I/O	bidirectional serial data for audio interface A1/A2
BCLK2	171	I/O	bidirectional bit clock for audio interface A2
WS4	172	I/O	bidirectional word select signal for audio interface A2
SD4	173	I/O	bidirectional serial data for audio interface A2
ACLK	174	I	audio reference clock input signal
SCL	175	I/O	bidirectional I^2C-bus clock line
SDA	176	I/O	bidirectional I^2C-bus data line
V_{DDD21}	177	P	digital supply voltage 21 (3.3 V)
V_{DDI2C}	178	I	I^2C-bus voltage sense input; see note 3 of “Characteristics”
V_{SSD23}	179	P	digital ground 23
GPIO3	180	I/O	general purpose I/O signal 3
GPIO2	181	I/O	general purpose I/O signal 2
GPIO1	182	I/O	general purpose I/O signal 1
GPIO0	183	I/O	general purpose I/O signal 0
V_{DDD22}	184	P	digital supply voltage 22 (3.3 V)
n.c.	185	–	reserved pin; not connected internally
V_{SSD24}	186	P	digital ground 24
D1_B0	187	I/O	bidirectional digital CCIR 656 D1 port B bit 0
D1_B1	188	I/O	bidirectional digital CCIR 656 D1 port B bit 1
D1_B2	189	I/O	bidirectional digital CCIR 656 D1 port B bit 2
D1_B3	190	I/O	bidirectional digital CCIR 656 D1 port B bit 3
V_{DDD23}	191	P	digital supply voltage 23 (3.3 V)
n.c.	192	–	reserved pin; not connected internally
V_{SSD25}	193	P	digital ground 25
D1_B4	194	I/O	bidirectional digital CCIR 656 D1 port B bit 4
D1_B5	195	I/O	bidirectional digital CCIR 656 D1 port B bit 5
D1_B6	196	I/O	bidirectional digital CCIR 656 D1 port B bit 6
D1_B7	197	I/O	bidirectional digital CCIR 656 D1 port B bit 7
V_{DDD24}	198	P	digital supply voltage 24 (3.3 V)
n.c.	199	–	reserved pin; not connected internally
V_{SSD26}	200	P	digital ground 26
LLC_B	201	I/O	bidirectional line-locked system clock port B

SYMBOL	PIN	STATUS	DESCRIPTION
VS_B	202	I/O	bidirectional vertical sync signal port B
HS_B	203	I/O	bidirectional horizontal sync signal port B
PXQ_B	204	I/O	bidirectional pixel qualifier signal to mark valid pixels port B; note 2
n.c.	205	–	reserved pin; do not connect
V_{DDD25}	206	P	digital supply voltage 25 (3.3 V)
n.c.	207	–	reserved pin; not connected internally
n.c.	208	–	reserved pin; do not connect

Notes

1. For continuous CCIR 656 format at the D1_A port this pin must be set HIGH.
2. For continuous CCIR 656 format at the D1_B port this pin must be set HIGH.

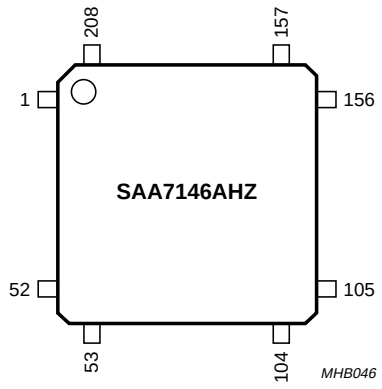


Fig.3 Pin configuration SAA7146AHZ (SQFP208).

7 FUNCTIONAL DESCRIPTION

This chapter provides information about the features realized with this device. First, a general, thus short, description of the functionality is given. The following sections deal with the single features in a detailed manner.

7.1 General

The Dual D1 (DD1) interface can be connected to digital video decoder ICs such as the SAA7110 and SAA7111A, digital video encoder such as the SAA7185B, video compression CODECs or to a D1 compatible connector, e.g. for interconnection to an external digital camera. The interface supports bidirectional full duplex two channel full D1 (CCIR 656), optionally with separate sync lines H/V, pixel qualifier signal and double pixel clock I/O, up to 32 MHz. It also supports a 16-bit parallel 'YUV bus' for interfacing to the SAA7110.

One of the two internal video processors of the SAA7146A is the two-dimensional High Performance Scaler (HPS). Phase accurate re-sampling by interpolation supports independent horizontal up and downscaling. In the horizontal direction the scaling process is performed in two functional blocks: integer decimation by window averaging (up to 65 tap), and phase linear interpolation (10 tap filter for luminance, 6 tap filter for chrominance). The vertical processing for downscaling either uses averaging over a window (up to 65 tap) or linear interpolation (2 tap). The scaling function can be used for random sized display windowing, for horizontal upscaling (zoom) or for conversion between various sample schemes such as CCIR or SQP. Incorporated with the HPS function is brightness, contrast and saturation control. Colour key generation is also established. The output of the HPS can be formatted in various RGB and YUV formats. Additionally, this output can be dithered for low bit rate formats. Packed formats as well as planar formats (YUV) are supported.

A second video channel (YUV 4 : 2 : 2 format) bypasses the HPS and connects the real time video interface with the PCI interface. This video bypass channel, using the second video processor Binary Ratio Scaler (BRS), is bidirectional and has means to convert from full size video (50 or 60 Hz) to Common Interchange Format (CIF), Quarter Common Interchange Format (QCIF) or Quarter Quarter Common Interchange Format (QQCIF) and vice versa (binary ratio 1, 2, 4, 8, $\frac{1}{2}$, $\frac{1}{4}$ and $\frac{1}{8}$ only). Multiple programmable VBI data and test signal regions can be bypassed without processing during each field.

The bidirectional digital audio serial interface is based on the I^2S-bus standard, but supports flexible programming for various data and timing formats.

Two independent interface circuits control audio data streaming of up to 2×128-bit frame width (bidirectional or simultaneous input/output). Five or more I^2S devices such as the SAA7360 and SAA7366 (ADC) and SAA7350 and SAA7351 (DAC) can be connected gluelessly.

The peripheral data port [Data Expansion Bus Interface (DEBI)] enables 8 or 16-bit parallel access for system set-up and programming of peripheral multimedia devices (behind SAA7146A), but is also highly capable to interface compressed MPEG/JPEG data of peripheral ICs with the PCI system. DEBI supports both Intel compatible (ISA-bus like) and Motorola (68000 style) compatible handshaking protocols with up to 23 Mbytes/s peak data rate. Besides the parallel port, there is also an I^2C-bus port to control peripheral ICs such as single-chip decoders SAA7110 and SAA7111A or as encoders such as SAA7185B and SAA7187 or as audio ICs.

The PCI interface has master read and master write capability. The video signal flows to and from the PCI and is controlled by three video DMA channels with a total FIFO capacity of 384 Dwords. The video DMA channel definition supports the typical video data structure (hierarchy) of pixels, lines, fields and frames. The audio signal flow is controlled by four audio DMA channels, each with 24 Dwords FIFO capacity. The DEBI port is connected to the PCI by single instruction direct access (immediate mode) and via a data DMA channel for streaming data (block mode) with 32 Dwords FIFO capacity. To improve PCI-bus efficiency, an arbiter schedules the access to PCI-bus for all local DMA channels.

The PCI interface of the SAA7146A supports virtual memory addressing for operating systems running virtual demand paging. The integrated Memory Management Unit (MMU) translates linear addressing to physical addresses using a page table inside the system memory provided by the software driver. The MMU supports up to 4 Mbytes of virtual address space per DMA channel.

The SAA7146A can change its programming sets using a Register Programming Sequencer (RPS) that works by itself on a user defined program controlled by internally supported real time events. The SAA7146A has two RPS machines to optimize flow control of e.g. an MPEG compressed data stream and real time video scaling control. The RPS programming is defined by an instruction list in the system main memory that consists of multiple RPS commands.

7.2 PCI interface

This section describes the interface of the SAA7146A to the PCI-bus. This includes the PCI modules, the DMA controls of the video, audio and data channels, the Memory Management Unit (MMU) and the Internal Arbitration Control (INTAC). The handling of the FIFOs and the corresponding errors are also described and a list of all DMA control registers is given.

7.2.1 PCI MODULES AND CONFIGURATION SPACE

The SAA7146A provides a PCI-bus interface having both slave and master capability. The master and the slave module fulfil the PCI local bus specification revision 2.1. They decode the C/BE# lines to provide a byte-wise access and support 32-bit transfers up to a maximum clock rate of 33 MHz. To increase bus performance, they are able to handle fast back-to-back transfers.

During normal operation the SAA7146A checks for parity errors and reports them via the PERR# pin. If an address parity error is detected the SAA7146A will not respond.

Using the SAA7146A as a slave, access is obtained only to the programmable registers and to its configuration space. Video, audio and other data of the SAA7146A reads/writes autonomously via the master interface (see Fig.4). The use of the PCI master module, i.e. which DMA channel gets access to the PCI-bus, is controlled by the INTAC (see Section 7.2.5).

The registers described in Table 1 are closely related to the PCI specification. It should be noted that Header type, Cache Line Size, BIST, Card bus CIS Pointer and Expansion ROM Base Address Registers are not implemented. All registers, which are not implemented are treated as read only with a value of zero. Some values are loaded after PCI reset via I^2C-bus from EEPROM with device address 1010000 (binary). This loading will take approximately 1 ms at 33 MHz PCI clock. If any device tries to read or write data from or to the SAA7146A during the loading phase after reset, the SAA7146A will disconnect with retry.

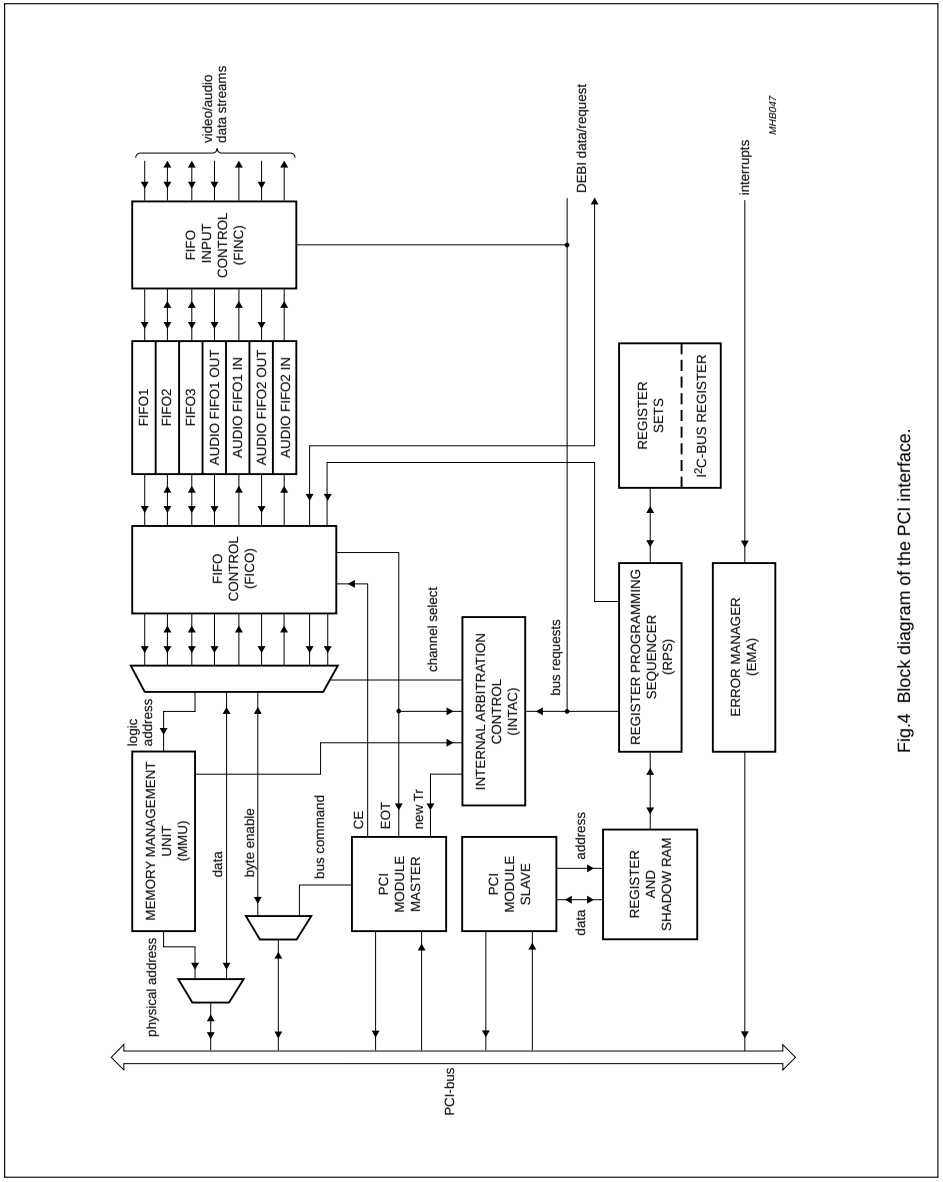


Fig.4 Block diagram of the PCI interface.

Table 1 Configuration space registers

ADDRESS (HEX)	NAME	BIT	TYPE	DESCRIPTION
00	Device ID	31 to 16	RO 7146H	SAA7146A
	Vendor ID	15 to 0	RO 1131H	Philips
04	Status Register	31	–	detected parity error
		29	–	received master abort
		28	–	received target abort
		26 and 25	RO 01	DEVSEL# timing medium
		24	–	data parity error detected
		23	RO 1	fast back-to-back capable
	Command Register	9	RW	fast back-to-back enable
		6	RW	parity error response
		2	RW	bus master enable
		1	RW	memory space
08	Class Code	31 to 8	RO 048000H	other multimedia device
	Revision ID	7 to 0	RO 01H	reading these 8 bits returns 01H
0C	Latency	15 to 8	RW	this register specifies, in units of PCI-bus clocks, the value of the latency timer for this PCI-bus master
10	Base Address Register	31 to 9	RW	this value must be added to the register offset to claim access to the programming registers; the lower 8 bits are forced to zero
		8 to 0	RO	
2C	Subsystem ID	31 to 16	RO	this value will be loaded after a PCI reset from external hardware using the I^2C-bus; the default value is 0000H
	Subsystem vendor ID	15 to 0	RO	this value will be loaded after a PCI reset from external hardware using the I^2C-bus; the default value is 0000H
3C	Max_Lat	31 to 24	RO	this value will be loaded after a PCI reset from external hardware using the I^2C-bus; the default value is 26H
	Min_Gnt	23 to 16	RO	this value will be loaded after a PCI reset from external hardware using the I^2C-bus; the default value is 0FH
	Interrupt Pin	15 to 8	RO 01H	The interrupt pin register tells which interrupt pin the device uses. This device uses interrupt pin INTA#. When these bits are read they return 01H.
	Interrupt Line	7 to 0	RW	the interrupt line register tells which input of the system interrupt controller the device's interrupt pin is connected to

7.2.2 VIDEO DMA CONTROL

The SAA7146A's DMA control is able to support up to three independent video targets or sources respectively. For this purpose it provides three video DMA channels. Each channel consists of a FIFO, a FIFO Input Control (FINC) placed on the video side of the FIFO, and a FIFO Control (FICO) placed on the PCI side of the FIFO. Channel 1 only supports the unidirectional data stream into the PCI memory. It is not able to read data from system memory. However, this access is possible using Channels 2 or 3. Table 2 surveys the possibilities and purposes of each video DMA channel.

Each FIFO, i.e. each DMA channel, has its own programming set including base address (doubled for odd and even fields), pitch, protection address, page table base address, several handling mode control bits and a transfer enable bit (TR_E). In addition, each channel has a threshold and a burst length definition for internal arbitration (see Table 6, Section 7.2.5).

To handle the reading modes FIFO 2 and FIFO 3 offer some additional registers: Number of Bytes per line (NumBytes), Number of Lines per field (NumLines) and the vertical scaling ratio (only FIFO 3, see Table 69). The programming sets could be reloaded after the previous job is done [Video Transfer Done (VTD)] to support several DMA targets per FIFO. The programming set currently used is loaded by the Register Programming Sequencer (RPS). If the RPS is not used, the registers could be rewritten each time, using the SAA7146A as a slave. But then the programmer must take care of the synchronization of these write accesses.

All registers needed for DMA control are described in Table 3, except the transfer enable bits, which are described in Table 10. The registers are accessed through PCI base address with appropriate offset (see Table 1).

Table 2 Size, direction and purpose of the video FIFOs and the associated DMA controls

FIFO	SIZE	DIRECTION	PURPOSE
FIFO 1	128 Dwords	write to PCI	FIFO 1 buffers data from the HPS output and writes into PCI memory. In planar mode FIFO 1 gets the Y data.
FIFO 2	128 Dwords	RW	**Planar mode**: FIFO 2 buffers U data provided by the HPS; the associated DMA control 2 sends it into the PCI memory.
			Clip mode: DMA control 2 reads clipping information (clip bit mask or rectangular overlay data) from the PCI system memory and buffers it in FIFO 2.
FIFO 3	128 Dwords	RW	**Planar mode**: FIFO 3 buffers V data provided by the HPS and writes it into the PCI memory.
			Chroma keying mode: FIFO 3 buffers chroma keying information and writes it into PCI memory.
			BRS mode: FIFO 3 buffers data provided by the BRS. DMA control 3 sends it into the PCI memory.
			Read mode: DMA control 3 reads video data from the PCI system memory (the same data up to four times to offer a simple upscaling algorithm) and buffers it in FIFO 3.

Table 3 Video DMA control registers

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
00	BaseOdd1	31 to 0	RW	**PCI base address for odd fields** of the upper (or lower if pitch is negative) left pixel of the transferred field
04	BaseEven1	31 to 0	RW	**PCI base address for even fields** of the upper (or lower if pitch is negative) left pixel of the transferred field
08	ProtAddr1	31 to 2	RW	**protection address**
	–	1 and 0	–	reserved
0C	Pitch1	31 to 0	RW	distance between the start addresses of two consecutive lines of a single field
10	Page1	31 to 12	RW	**base address of the page table** (see Section 7.2.4)
	ME1	11	RW	**mapping enable**; this bit enables the MMU
	–	10 to 8	–	reserved
	Limit1	7 to 4	RW	**interrupt limit**; defines the size of the memory range, that raise an interrupt, if its boundaries are passed
	PV1	3	RW	**protection violation** handling
	–	2	–	reserved
	Swap1	1 and 0	RW	**endian swapping** of all Dwords passing the FIFO 1: 00 = no swap 01 = two bytes swap (3210 to 2301) 10 = four bytes swap (3210 to 0123) 11 = reserved
14	NumLines1	27 to 16	RW	**Number of lines per field;** it defines the number of qualified lines to be processed by the HPS per field. This will cut off all the following input lines at the HPS input.
	NumBytes1	11 to 0	RW	**Number of pixels per line**; it defines the number of qualified pixels to be processed by the HPS per line. This will cut off all the following pixels at the HPS input.
18	BaseOdd2	31 to 0	RW	**PCI base address for odd fields** of the upper (or lower if top-down flip is selected) left pixel of the transferred field
1C	BaseEven2	31 to 0	RW	**PCI base address for even fields** of the upper (or lower if top-down flip is selected) left pixel of the transferred field
20	ProtAddr2	31 to 2	RW	**protection address**
	–	1 and 0	–	reserved
24	Pitch2	31 to 0	RW	distance between the start addresses of two consecutive lines of a field
28	Page2	31 to 12	RW	**base address of the page table** (see Section 7.2.4)
	ME2	11	RW	**mapping enable**; this bit enables the MMU
	–	10 to 8	–	reserved
	Limit2	7 to 4	RW	**interrupt limit**; defines the size of the memory range, that raise an interrupt, if its boundaries are passed
	PV2	3	RW	**protection violation** handling

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
28	RW2	2	RW	Specifies the data stream direction of FIFO 2. A logic 0 enables a write operation to the PCI memory. A logic 1 enables a read operation from the PCI memory.
	Swap2	1 and 0	RW	**endian swapping** of all Dwords passing the FIFO 2: 00 = no swap 01 = two byte swap (3210 to 2301) 10 = four byte swap (3210 to 0123) 11 = reserved
2C	NumLines2	27 to 16	RW	**Number of lines per field:** in read mode NumLines defines the number of lines to be read from system memory. A logic 0 specifies one line. In write mode this register is not used.
	NumBytes2	11 to 0	RW	**Number of bytes per line:** in read mode this defines the number of bytes per line to be read from system memory. A logic 0 specifies one byte. In write mode this register is not used.
30	BaseOdd3	31 to 0	RW	**PCI base address for odd fields** of the upper (or lower if top-down flip is selected) left pixel of the transferred field
34	BaseEven3	31 to 0	RW	**PCI base address for even fields** of the upper (or lower if top-down flip is selected) left pixel of the transferred field
38	ProtAddr3	31 to 2	RW	**protection address**
	–	1 and 0	–	reserved
3C	Pitch3	31 to 0	RW	distance between the start addresses of two consecutive lines of a field
40	Page3	31 to 12	RW	**base address of the page table** (see Section 7.2.4)
	ME3	11	RW	**mapping enable**; this bit enables the MMU
	–	10 to 8	–	reserved
	Limit3	7 to 4	RW	**interrupt limit**; defines the size of the memory range, that raise an interrupt, if its boundaries are passed
	PV3	3	RW	**protection violation** handling
	RW3	2	RW	Specifies the data stream direction of FIFO 3. A logic 0 enables a write operation to the PCI memory. A logic 1 enables a read operation from the PCI memory.
	Swap3	1 and 0	RW	**endian swapping** of all Dwords passing the FIFO 3: 00 = no swap 01 = two byte swap (3210 to 2301) 10 = four byte swap (3210 to 0123) 11 = reserved
44	NumLines3	27 to 16	RW	**Number of lines per field**: in read mode NumLines defines the number of lines to be read from system memory. A logic 0 specifies one line. In write mode it defines the number of qualified lines to be processed by the BRS per field. This will cut off all the following input-lines at the BRS input.
	NumBytes3	11 to 0	RW	**Number of bytes per line**: in read mode this defines the number of bytes per line to be read from system memory. A logic 0 specifies 1 byte. In write mode it defines the number of qualified bytes to be processed by the BRS per line. This will cut off all the following bytes at the BRS input.

The video channels provide 32 bits of data signals and 4 bits of Byte Enable (BE) signals, End-Of-Line (EOL), End-Of-Window (EOW), Begin-Of-Field (BOF), Line-Locked Clock (LLC), Odd/Even signal (OE) and a Valid Data (VD) signal. To start a video data transfer, e.g. via video DMA Channel 3, this channel must first be included in the internal arbitration scheme. This is achieved by setting the corresponding TR_E bit (see Table 10). If a TR_E bit is not set, the corresponding FIFO is reset.

In read mode, which is offered by Channels 2 and 3, the FICO requests a PCI transfer with the next BOF. Data is provided by the PCI master module. The FICO calculates the PCI address autonomously, starting with the base address of the corresponding field. Only the received data will be filled into the FIFO. FIFO 3 offers the possibility to read video information from PCI memory, e.g. from the frame buffer. This could be achieved by using the NumBytes and the NumLines register, which defines the size of the source picture, so that the DMA control is able to synchronize itself to the source frame. FIFO 2 does the same if reading clip information from memory.

To support the Binary Ratio Scaler (BRS) included in the SAA7146A, which only provides the possibility of horizontal upscaling, the DMA control 3 can be applied to perform line repetition by reading lines up to four times from PCI memory. This feature is controlled by the vertical scaling ratio in outbound mode (see Table 66). This ratio specifies the number of times each line should be read: 00 = only once, 01 = twice, and so on.

In the event of FIFO underflow, i.e. if the BRS or the clipping unit respectively tries to read data from the FIFO, even if the DMA control was not able to fill any data until that moment, the reading unit tries to synchronize itself to the outgoing data stream as soon as possible. In this way the reading of invalid data is minimized. If the clipping unit receives no data, it will disable the associated pixels. The behaviour of the BRS depends on the selected read mode which is described in Section 7.10.

In the event of FIFO overflow, i.e. if the scaler tries to transfer data although the FIFO is full, the FIFO input control locks the FIFO for the incoming data. During FIFO overflow the PCI address of the incoming data will be increased, over writing itself each time, if the scaler transfers data, which has been clipped, the same mechanism is used to improve PCI performance.

The SAA7146A is able to handle a negative pitch. With that, top-down-flip of the transmitted fields or frames is possible. A negative pitch (MSB = 1) leads to a different definition of the protection and the base address, as shown in Fig.5. If using negative pitch the first line starts at base address + pitch.

In ‘none-RPS’ mode the SAA7146A supports the displaying of interlaced video data by using the two different base addresses (BaseOdd and BaseEven) and vertical start phases (YPE6 to YPE0 and YPO6 to YPO0) for odd and even fields.

Using the protection address, system memory could be kept of from prohibited write accesses. If the PCI pointer of the current transfer reaches or exceeds the protection address, the SAA7146A stops this transfer and an interrupt is initiated. No interrupt is set if a protection violation occurs due to the programming that was done before the channel has been switched on. To prevent one field from being transferred into memory, set its base address (BaseOdd or BaseEven) to the same value as the protection address.

If the Protection Violation (PV) handling bit and the limit register are reset, the following data will be ignored until detection of the End-Of-Window (EOW) signal. In read mode the DMA control also waits for this signal, to start the next data transfer. If the PV bit is set, the input of the FIFO will be locked and the FIFO will be emptied. If the FIFO is empty the TR_E bit is reset. This feature could be used for a single capture mode, if the protection address is the same address as the last pixel in this field. With that, the SAA7146A will write one field into system memory and then stop.

If the limit register of any DMA channel (video, VBI data or audio) has a value other than ‘0000’ the continuous write mode is chosen. If the actual PCI address hits the protection address and the PV bit is zero, the FINC stops the current transfer, sets an interrupt and resets the actual address to the base address. Regarding this, the protection address could be used to define a memory space to which data is sent. The SAA7146A offers the possibility to monitor the filling level of this memory space. The limit register defines an address limit, which generates an interrupt if passed by the actual PCI address pointer. ‘0001’ means an interrupt will be generated if the lower 6 bits (64 bytes) of the PCI address are zero. ‘0010’ defines a limit of 128 bytes, ‘0011’ one of 256 bytes, and so on up to 1 Mbyte defined by ‘1111’. This interrupt range can be calculated as follows:

$$\text{Range} = 2^{(5 + \text{Limit})} \text{ bytes.}$$

The protection handling modes such as those selected by the PV bit and the contents of the limit register are shown in Table 4.

Table 4 Protection violation handling modes

LIMIT	PV	DESCRIPTION
0000	0	Lock input of FIFO and empty FIFO (only in write mode). Unlock FIFO and start next transfer using the base address at the detection of BOF.
$\overline{0000}$	0	Restart immediately at base address.
XXXX[1]	1	Lock input of FIFO, empty FIFO (only in write mode) and then reset TR_E bit. The next transfer starts with BOF using the corresponding base address, if the TR_E bit is set again. This setting is useful for single-shot, that means transferring only one frame of a video stream. Therefore the protection address has to be the same as the address of the last pixel of the field.

Note

1. X = don't care.

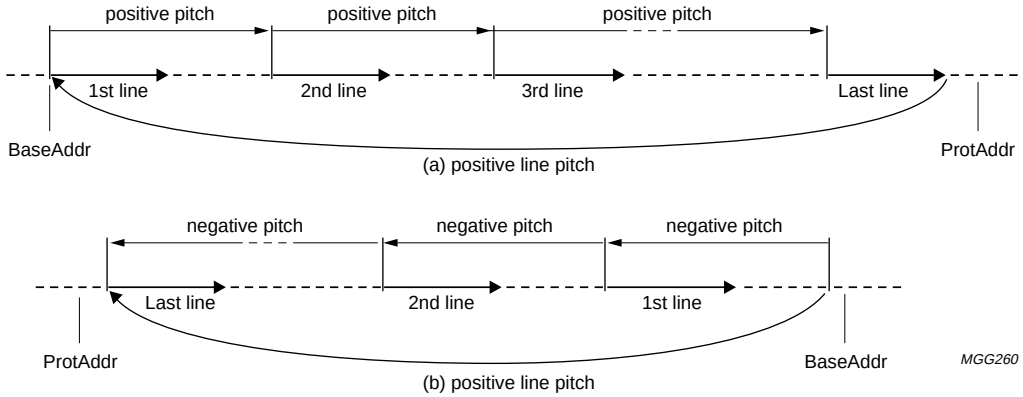


Fig.5 Handling of base and protection address using positive and negative line pitch.

7.2.3 AUDIO DMA CONTROL

The SAA7146A provides up to four audio DMA channels, each using a FIFO of 24 Dwords. Two channels are read only (A1_in and A2_in) and two channels are write only (A1_out and A2_out). Because audio represents a continuous data stream, which is neither line nor field dependent, the audio DMA control offers only one base address (BaseAxx) and no pitch register. For FIFO overflow and underflow the handling of these channels is done in the same way as the video DMA channels (see Section 7.2.2).

The protection violation handling differs only if the limit register and the PV bit are programmed to zero. The audio DMA channel does not wait for the EOF signal, like the video ones. It does not generate interrupts. The interrupt range specified by the limit register is defined in the same way as described in Section 7.2.2. The audio DMA channels try immediately to transfer data after setting the transfer enable bits. All registers for audio DMA control, which are the base address, the protection address and the control bits are listed in the following Table 5, except the input control bits (Burst, Threshold), which are listed in Table 6.

Table 5 Audio DMA control register

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
94	BaseA1_in	31 to 0	RW	**base address for audio input Channel 1**; this value specifies a byte address
98	ProtA1_in	31 to 2	RW	**protection address for audio input Channel 1**; this address could be used to specify a upper limit for audio access in memory space
	–	1 to 0	–	reserved
9C	PageA1_in	31 to 12	RW	**base address of the page table,** see Section 7.2.4.
	MEA1_in	11	RW	**mapping enable**; this bit enables the MMU
	–	10 to 8	–	reserved
	LimitA1_in	7 to 4	RW	**interrupt limit**; defines the size of the memory range, that generates interrupt, if its boundaries are passed
	PVA1_in	3	RW	**protection violation handling**
	–	2 to 0	–	reserved
A0	BaseA1_out	31 to 0	RW	**Base address for audio output Channel 1**; this value specifies a byte address. The lower two bits are forced to zero.
A4	ProtA1_out	31 to 2	RW	**protection address for audio output Channel 1**; this address could be used to specify a upper limit for audio access in memory space
	–	1 and 0	–	reserved
A8	PageA1_out	31 to 12	RW	**base address of the page table**, see Section 7.2.4.
	MEA1_out	11	RW	**mapping enable**; this bit enables the MMU
	–	10 to 8	–	reserved
	LimitA1_out	7 to 4	RW	**interrupt limit**; defines the size of the memory range, that generates an interrupt, if its boundaries are passed
	PVA1_out	3	RW	**protection violation handling**
	–	2 to 0	–	reserved

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
AC	BaseA2_in	31 to 0	RW	**Base address for audio input Channel 2**; this value specifies a byte address. The lower two bits are forced to zero.
B0	ProtA2_in	31 to 2	RW	**protection address for audio input Channel 2**; this address could be used to specify a upper limit for audio access in memory space
	–	1 and 0	–	reserve
B4	PageA2_in	31 to 12	RW	**base address of the page table**, see Section 7.2.4
	MEA2_in	11	RW	**mapping enable**; this bit enables the MMU
	–	10 to 8	–	reserved
	LimitA2_in	7 to 4	RW	**interrupt limit**; defines the size of the memory range, that raise an interrupt, if its boundaries are passed
	PVA2_in	3	RW	**protection violation handling**
	–	2 to 0	–	reserve
B8	BaseA2_out	31 to 0	RW	**Base address for audio output Channel 2**; this value specifies a byte address. The lower two bits are forced to zero.
BC	ProtA2_out	31 to 2	RW	**protection address for audio output Channel 2**; this address could be used to specify a upper limit for audio access in memory space
	–	1 and 0	–	reserved
C0	PageA2_out	31 to 12	RW	**base address of the page table**, see Section 7.2.4
	MEA2_out	11	RW	**mapping enable**; this bit enables the MMU
	–	10 to 8	–	reserved
	LimitA2_out	7 to 4	RW	**interrupt limit**; defines the size of the memory range, that raise an interrupt, if its boundaries are passed
	PVA2_out	3	RW	**protection violation handling**
	–	2 to 0	–	reserved

7.2.4 Memory Management Unit (MMU)

7.2.4.1 Introduction

To perform DMA transfers, physically continuous memory space is needed. However, operating systems such as Microsoft Windows are working with virtual demand paging, using a MMU to translate linear to physical addresses. Memory allocation is performed in the linear address space, resulting in fragmented memory in the physical address space. There is no way to allocate large buffers of physical, continuous memory, except reserving it during system start-up. Thus decreasing the system performance dramatically. To overcome this problem the SAA7146A contains a Memory Management Unit (MMU) as well. This MMU is able to handle memory fragmented to 4 kbyte pages, similar to the scheme used by the Intel 8086 processor family. The MMU can be bypassed to simplify transfers to non-paged memory such as the graphics adapter's frame buffer.

7.2.4.2 Memory allocation

The SAA7146A's MMU requires a special scheme for memory allocation. The following steps have to be performed:

- Allocation of n pages, each page being 4 kbytes of size, aligned to a 4 kbyte boundary
- Allocation of one extra page, to be used as page table
- Initialization of the page table.

Allocation of pages is done in physical address space. Operating systems implementing virtual memory provide services to allocate and free these pages.

The page table is stored in a separate page. This limits the linear address page to a size of 4 Mbytes and results in a 4 kbyte overhead. The page table is organized as an array of n Dwords, with each entry giving the physical address of one of the n pages of allocated memory. As pages are aligned to 4 kbytes, the lower 12 bits of each entry are fixed to zero.

7.2.4.3 Implementation

The SAA7146A has up to 8 DMA channels (3 video, 4 audio and 1 DEBI channel) for which the memory mapping is done. Each of them provides the linear address to (from) which it wants to send (read) data during the next transfer. Their register sets contain a page table base address (Pagexx) and a mapping enable bit (MExx). If MExx is set, mapping is enabled.

The MMU checks for each channel whether its address has been already translated. If translated, its request can pass to the Internal Arbitration Control (INTAC) managing the access to the PCI-bus. If not, the MMU starts a bus transfer to the page table. The page table entry address could be calculated from the channels PCI address and the page table base address, as shown in Fig.6. The upper 20 bits of the PCI address are replaced by the upper 20 bits of the according page address to generate the mapped PCI address.

If the PCI address crosses a 4 kbyte boundary during a transfer, the MMU stops this transfer and suppresses its request to the INTAC until it has renewed the page address, which means replacing the upper 20 bits of the current address. To reduce latency the SAA7146A will do a pre-fetch, i.e. it will always try to load the next page address in advance.

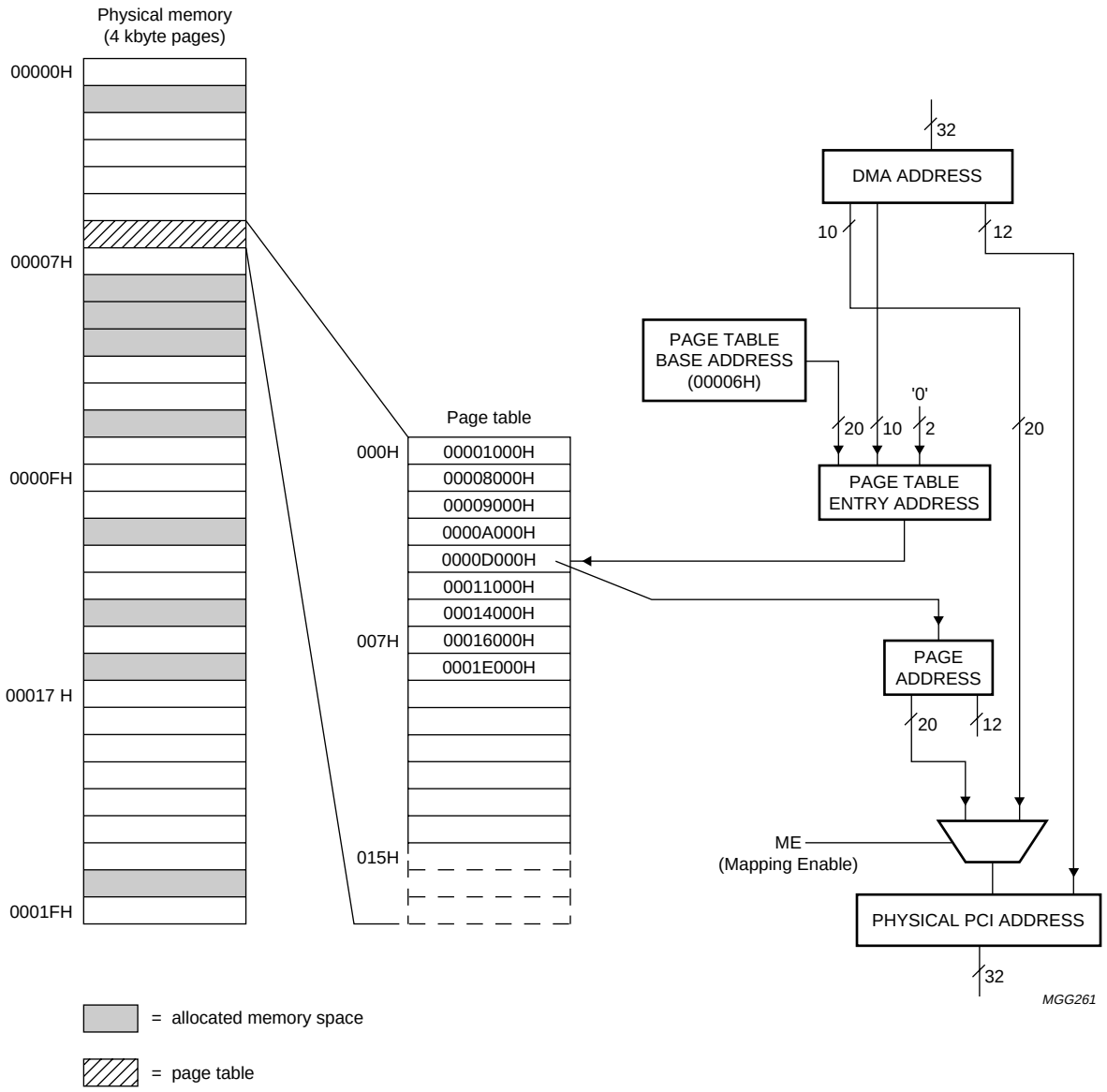


Fig.6 Memory Management Unit (MMU).

7.2.5 INTERNAL ARBITRATION CONTROL

The SAA7146A has up to three video DMA channels, four audio DMA channels and three other DMA channels (RPS, MMU and DEBI) each trying to get access to the PCI-bus. To handle this, an Internal Arbitration Control (INTAC) is needed. INTAC controls on the one hand the PCI-bus requests and on the other hand the order in which each DMA channel gets access to the bus.

The basic implementation of the internal arbitration control is a round-robin mechanism on the top, consisting of the RPS, the MMU and one of the eight data channels. Data channel arbitration is performed using a 'first come first serve' queue architecture, which may consist of up to eight entries.

Each data channel reaching a certain filling level of its FIFO defined by the threshold, is allowed to make an entry into the arbitration queue. The threshold defines the number of Dwords needed to start a sensible PCI transfer and must be small enough to avoid a loss of data due to an overflow regarding the PCI latency time. After each job (Video Transfer Done, VTD) the video channels have to be emptied and are allowed to fill an entry into the queue, even if they have not yet reached their threshold.

Concurrently to the entry the channel sets a bit which prohibits further entries to this channel. In the worst case, each data channel can have only one entry in the queue.

If each channel wants to access the bus, which means the queue is full, an order like the one shown below will be given.

- MMU
- RPS.

First entry of the data channel queue:

- MMU
- RPS.

Second entry of the data channel queue:

- MMU
- and so on.

If INTAC detects at least one DMA channel in the queue or an MMU or an RPS request, it signals the need for the bus by setting the REQ# signal on the PCI-bus. If the GNT# signal goes LOW, the SAA7146A is the owner of the bus and makes the PCI master module working with the first channel selected. The master module tries to transfer the number of Dwords defined in the Burst Register. For RPS the burst length is hardwired to four and for the MMU it is hardwired to two Dwords. After that the master module stops this transfer and starts a transfer using the next channel (due to the round-robin).

If a DMA channel gets its transfer stopped due to a retry, the arbitration control sets the corresponding retry flag. INTAC tries to end a retried transfer, even if this transfer gets stopped via the Transfer Enable bit (TR_E). For this reason the Transfer Enable bits are internally shadowed by INTAC. A transfer can only be stopped if it has no retry pending.

The Arbitration Control Registers (Burst and Threshold of DEBI, Video 1 to 3, Audio 1 to 4) are listed in Table 6.

Table 6 Arbitration control registers

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
48	BurstDebi	28 to 26	RW	PCI burst length of the DEBI DMA channel; see Table 7
	Burst3	20 to 18	RW	PCI burst length of video Channel 3; see Table 7
	Thresh3	17 to 16	RW	threshold of FIFO 3; see Table 8
	Burst2	12 to 10	RW	PCI burst length of video Channel 2; see Table 7
	Thresh2	9 to 8	RW	threshold of FIFO 2; see Table 8
	Burst1	4 to 2	RW	PCI burst length of video Channel 1; see Table 7
	Thresh1	1 and 0	RW	threshold of FIFO 1; see Table 8
4C	BurstA1_in	28 to 26	RW	PCI burst length of audio input Channel 1; see Table 7
	ThreshA1_in	25 to 24	RW	threshold of audio FIFO A1_in; see Table 8
	BurstA1_out	20 to 18	RW	PCI burst length of audio output Channel 1; see Table 7
	ThreshA1_out	17 and 16	RW	threshold of audio FIFO A1_out; see Table 8
	BurstA2_in	12 to 10	RW	PCI burst length of audio input Channel 2; see Table 7
	ThreshA2_in	9 and 8	RW	threshold of audio FIFO A2_in; see Table 8
	BurstA2_out	4 to 2	RW	PCI burst length of audio output Channel 2; see Table 7
	ThreshA2_out	1 and 0	RW	threshold of audio FIFO A2_out; see Table 8

Table 7 Burst length definition

VALUE	BURST LENGTH
000	1 Dword
001	2 Dwords
010	4 Dwords
011	8 Dwords
100	16 Dwords
101	32 Dwords
110	64 Dwords
111	128 Dwords

Table 8 Threshold definition

VALUE	WRITE MODE(1)		READ MODE(1)	
	VIDEO	AUDIO	VIDEO	AUDIO
00	4 Dwords of valid data	1 Dword of valid data	4 empty Dwords	1 empty Dword
01	8 Dwords of valid data	4 Dwords of valid data	8 empty Dwords	4 empty Dwords
10	16 Dwords of valid data	8 Dwords of valid data	16 empty Dwords	8 empty Dwords
11	32 Dwords of valid data	16 Dwords of valid data	32 empty Dwords	16 empty Dwords

Note

1. The threshold is reached, if the FIFO contains at least this number of Dwords.

7.2.6 STATUS INFORMATION OF THE PCI INTERFACE

Table 9 lists the status information that the PCI interface makes available to the user in addition to the interrupt sources that are described later. This information is read only.

Table 9 Status bits of the DMA control

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
120	VDP1	31 to 0	R	logical video DMA pointer of FIFO 1
124	VDP2	31 to 0	R	logical video DMA pointer of FIFO 2l
128	VDP3	31 to 0	R	logical video DMA pointer of FIFO 3
12C	ADP1	31 to 0	R	logical audio DMA pointer of audio output FIFO A1_out
130	ADP2	31 to 0	R	logical audio DMA pointer of audio input FIFO A1_in
134	ADP3	31 to 0	R	logical audio DMA pointer of audio output FIFO A2_out
138	ADP4	31 to 0	R	logical audio DMA pointer of audio input FIFO A2_in
13C	DDP	31 to 0	R	logical DEBI DMA pointer

7.3 Main control

7.3.1 GENERAL

The SAA7146A has two Dwords of general control to support quick enable/disable switching of any activity of the SAA7146A via direct access by the CPU. These main control Dwords are split in two parts. The upper parts have 16 bits of bit-mask to allow bit-selective write to the lower part which contains single bit enable/disable control of major interface functions of SAA7146A. If a certain bit position is masked with a logic 1 in the mask word (upper 2 bytes) during a write access, then the corresponding bit in the control word (lower 2 bytes) is changed according to the contents of the transmitted data. By that the CPU can easily switch on or off certain selected interfaces of the SAA7146A without checking the actual ‘remaining’ programming (enabling) of the other parts.

The programming of registers for the 3 Video DMA channels, both video processors (HPS, BRS) and for the interfaces DEBI and I^2C-bus is performed by an upload method. This is done to guarantee coherent programming data. During initiation of an upload operation from a shadow RAM each of the UPLD bits [10 to 0] (see Table 11) is assigned to a set of registers. If a logic 1 is written into a UPLD bit all dedicated shadow RAM registers containing changed data are uploaded into their working registers immediately. During a read cycle the UPLD bits give information on whether the shadow RAM contains changed data not yet uploaded into the working registers. The UPLD bits remain HIGH as long as the contents of the shadow RAM represents the current programming

Table 10 Main control register 1

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
Mask word				
FC	M15 to M00	31 to 16	RW	16-bit mask word for bit-selective writes to the control word; when read these bits always return logic 0
Control word				
FC	MRST_N	15	RW	**Master Reset Not:** this is the master reset for the SAA7146A. Writing a logic 0 to this bit will reset the SAA7146A to the same state as after a power-on reset. When read this bit always returns a logic 0.
	–	14	–	reserved: when read this bit always returns a logic 0
	ERPS1	13	RW	**Enable Register Program Sequencer Task 1**: if ERPS1 = 1, then any RPS Task 1 action is enabled. If ERPS1 = 0, then RPS Task 1 action does not fetch any more commands.
	ERPS0	12	RW	**Enable Register Program Sequencer Task 0**: if ERPS0 = 1, then any RPS Task 0 action is enabled. If ERPS0 = 0, then RPS Task 0 action does not fetch any more commands.
	EDP	11	RW	**Enable DEBI Port pins**: if EDP = 0, all pins of the DEBI port are set to 3-state. If EDP = 1, then the function of all pins at the DEBI port is as programmed via the DEBI registers.
	EVP	10	RW	**Enable Real Time Video Ports pins**: if EVP = 0, all 24 pins of the real time video interface (DD1 port) are 3-stated. If EVP = 1, then the function of all pins at the real time video interface (DD1 port) is as programmed by the scaler register; see Table 66.
	EAP	9	RW	**Enable Audio Port pins**: if EAP = 0, all 14 pins of the audio interface port are set to 3-state. If EAP = 1, then the function of all pins at the audio interface is as programmed in Section 7.16.3.
	EI2C	8	RW	**Enable I^2C Port pins**: if EI2C = 0, then both pins of the I^2C-bus interface port are set to 3-state. If EI2C = 1, then the I^2C-bus interface is enabled and will function as programmed in Section 7.17.2.
	TR_E_DEBI	7	RW	Transfer Enable bit of the DEBI.
	TR_E_1	6	RW	Transfer enable bit of video Channel 1: if set this channel is included in the internal arbitration scheme. If not set, this channel will be ignored and no transfer will start using this FIFO.
	TR_E_2	5	RW	Transfer Enable bit of video channel 2
	TR_E_3	4	RW	Transfer Enable bit of video channel 3
	TR_E_A2_OUT	3	RW	Transfer Enable bit of audio channel 2 out
	TR_E_A2_IN	2	RW	Transfer Enable bit of audio channel 2 in
	TR_E_A1_OUT	1	RW	Transfer Enable bit of audio channel 1 out
	TR_E_A1_IN	0	RW	Transfer Enable bit of audio channel 1 in

Table 11 Main control register 2

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
Mask word				
100	M15 to M00	31 to 16	RW	16-bit mask word for bit-selective writes to the control word; when read this bits always returns logic 0
Control word				
100	RPS_SIG4	15	RW	RPS Signal 4
	RPS_SIG3	14	RW	RPS Signal 3
	RPS_SIG2	13	RW	RPS Signal 2
	RPS_SIG1	12	RW	RPS Signal 1
	RPS_SIG0	11	RW	RPS Signal 0
	UPLD_D1_B	10	RW	Upload ‘Video DATA stream handling at port D1_B (54H)’; see Table 68. To upload ‘Initial setting of Dual D1 Interface (50H)’, this bit and bit 9 must be set; see Table 66.
	UPLD_D1_A	9	RW	Upload ‘Video DATA stream handling at port D1_A (54H)’; see Table 67. To upload ‘Initial setting of Dual D1 Interface (50H)’, this bit and bit 10 must be set; see Table 66.
	UPLD_BRS	8	RW	Upload ‘BRS Control Register (58H)’; see Table 69.
	–	7	–	Reserved; when read this bit always returns a logic 0.
	UPLD_HPS_H	6	RW	Upload ‘HPS Horizontal prescale (68H)’; see Table 79.
				Upload ‘HPS Horizontal fine-scale (6CH)’; see Table 81.
				Upload ‘BCS control (70H)’; see Table 82.
	UPLD_HPS_V	5	RW	Upload ‘HPS control (5CH)’; see Table 71. Upload ‘HPS Vertical scale (60H)’; see Table 72. Upload ‘HPS Vertical scale and gain (64H)’; see Table 73. Upload ‘Chroma Key range (74H)’; see Table 86. Upload ‘HPS Outputs and Formats (78H)’; see Table 87. Upload ‘Clip control (78H)’; see Table 89.
	UPLD_DMA3	4	RW	Upload ‘Video DMA3 registers’; 30H, 34H, 38H, 3CH, 40H, 44H and 48H (20 to 16).
	UPLD_DMA2	3	RW	Upload ‘Video DMA2 registers’; 18H, 1CH, 20H, 24H, 28H, 2CH and 48H (12 to 8).
	UPLD_DMA1	2	RW	Upload ‘Video DMA1 registers’; 00H, 04H, 08H, 0CH, 10H, 14H and 48H (4 to 0).
	UPLD_DEBI	1	RW	Upload ‘DEBI registers’; 88H, 7CH, 80H, 84H and 48H (28 to 26).
	UPLD_IIC	0	RW	Upload ‘I^2C-bus registers’; (8CH and 90H).

7.4 Register Programming Sequencer (RPS)

The RPS is used as an additional method to program or read the registers of the SAA7146A. Its main function is programming the registers on demand without delay via the interrupt handler of the host system.

Because different applications of the SAA7146A can run independently on and asynchronously to each other the RPS is capable of running two parallel tasks. Both tasks are completely equal to each other and each has its own set of registers (RPS address, RPS page, HBI threshold and RPS time out value). Each task can be separately enabled by setting its related ERPSx bit in the Main control register 1 (see Table 10). To allow communication between both tasks and the CPU there are five signals which can be set or reset from both tasks (see Table 11). The programming of a task is defined by an instruction list in the system main memory that consists of RPS commands. The operation of the RPS is initiated on command by setting the ERPS bit of the desired task in the Main control register 1.

The processing of RPS can be controlled by a sequence of wait commands on special events. Furthermore the program flow can be controlled via conditional jumps related to the communication with the host setting semaphores or special internal interrupts.

7.4.1 Reset

During a reset the ERPSx (Enable RPS of task ‘x’) bits in the Main control register 1 (see Table 10) of the SAA7146A are cleared so that an RPS task has to be explicitly started.

7.4.2 Event description

Table 12 shows the events available during the execution of an RPS program. The execution can for example wait on these events to become true. In general these events are set if a rising edge of the corresponding signal occurs and are cleared if a falling edge of the signal occurs.
If signals are logic HIGH after the reset and no rising edge occurs the corresponding event (available in an RPS program execution) will not be set.

Table 12 Description of events

EVENT	DESCRIPTION
IICD	**IIC Done:** Done flag of the I^2C-bus
DEBID	**DEBI Done:** Done flag of DEBI; see note 1
O_FID_A; O_FID_B	**Field Identification signal:** for an odd field dependent on sync detection at Port A/Port B
E_FID_A; E_FID_B	**Field Identification signal**: for an even field dependent on sync-detection at Port A/Port B
HS	**HPS Source**: wait for processing of source line before line addressed by SLCT is done
HT	**HPS Target**: wait for processing of target line before line addressed by TLCT is done
VBI_A; VBI_B	**Vertical Blanking Indicator** at Port A/Port B: for details on this signal see Table 90
BRS_DONE	**Inactive BRS data path**: for details on this signal see Table 90
HPS_DONE	**Inactive HPS data path between two windows**: for details on this signal see Table 90
HPS_LINE_DONE	**Inactive HPS data path between two lines**: for details on this signal see Table 90
VTD1; VTD2; VTD3	**Video Transfer Done**: video DMA 1, video DMA 2 or video DMA 3 has transferred a complete window and is ready to be reprogrammed
GPIO0	**General Purpose I/O 0**: this bit reflects the status of the GPIO pin 0
GPIO1	**General Purpose I/O 1**: this bit reflects the status of the GPIO pin 1
GPIO2	**General Purpose I/O 2**: this bit reflects the status of the GPIO pin 2
GPIO3	**General Purpose I/O 3**: this bit reflects the status of the GPIO pin 3
SIGx	**General purpose signal x**: for intertask and RPS to CPU communication or program flow control. ‘x’ can take a value within the range 0 to 4

Note

1. If an RPS program is used to make DEBI transfer consecutive data blocks employ the following commands: LOAD REGISTER, CLEAR SIGNAL, UPLOAD and PAUSE. Before uploading the register contents the DEBI_DONE flag of a former transfer has to be cleared. With this, the following PAUSE command waits correctly for DEBI_DONE of the just started DEBI block transfer.

7.4.3 COMMAND LIST

An instruction list of an RPS task is built in the system memory by the device driver. This list is made up of command sequences; each command being at least one Dword long. The first Dword of a command consists of the instruction code (4-bit) and a command specific part (28 bits). Commands longer than one Dword contain data in the additional Dwords.

Table 13 Command Dword

D31 to D28	D27 to D0
Instruction code	command specific

7.4.4 THE INSTRUCTION CODE

The instruction code identifies one of the following commands (see bits 31 to 28 of Tables 14 to 29).

7.4.4.1 PAUSE

The PAUSE command is a one Dword command. This command contains in the command specific part the events to wait for; see Tables 14 and 15. The execution of the RPS task is delayed until the condition addressed via the events becomes true or a time out occurs.

To control the time a PAUSE command stays in the wait state, it is possible to set a RPS time out value. This value specifies after how many PCI clocks and/or V_syncs a time out will be asserted. When it occurs the RPS_TO bits in the PSR (see Table 38) is set and if enabled an interrupt will be generated. However, the RPS will stop this task.

The OAN bit specifies if the condition in bits 25 to 0 is an AND (OAN set to 0) or if the condition is an OR (OAN set to 1). If the INV bit is set this command will wait for the condition to become false.

7.4.4.2 UPLOAD

The UPLOAD command is a one Dword-command. This command contains in the command specific part the sections to be uploaded from the shadow RAM to the working registers, see Tables 16 and 17.

If the UPLOAD command finds a bit of a section set it uploads the corresponding registers from the shadow RAM to the working registers. This is done for registers with changed shadow RAM values only.

7.4.4.3 CHECK-LATE

The CHECK_LATE command is a one Dword-command. This command contains in the command specific part the events to check and if necessary to wait for, as shown in Tables 18 and 19. The execution of the RPS task is delayed until the condition addressed via the events becomes true, or a time out occurs and the upload is performed.

The OAN bit specifies if the condition in bits 25 to 0 is an AND (OAN set to 0) or if the condition is an OR (OAN set to 1). If the INV bit is set this command will wait for the condition to become false.

If the CHECK_LATE command finds that the wait condition is already true the RPS-LATE is set. Otherwise it waits for the condition as the PAUSE command. A time out behaviour such as described for the PAUSE command is also supplied.

7.4.4.4 CLR_SIGNAL

The CLR_SIGNAL Command clears the selected signals. This will not affect the real status bits of the SAA7146A. Only a copy of this bit related to the RPS will be cleared. It will be set again via a SET_SIGNAL command or when the real status will be set due to normal processing. The CLR_SIGNAL format is shown in Tables 20 and 21.

7.4.4.5 NOP

The NOP command consists of one Dword and has the instruction code 0000. All bits of the command specific part have to be set to zero. This command is a special case of the CLR_SIGNAL command!

7.4.4.6 SET_SIGNAL

The SET_SIGNAL command sets the selected signals. If one of the SAA7146A status related signals is selected to be set, it will not affect the real status bit of the SAA7146A. Only a copy of this bit related to the RPS, will be set. The SET_SIGNAL format is shown in Tables 22 and 23.

7.4.4.7 INTERRUPT

The INTERRUPT command will set the RPS_I bit of the task in the Interrupt status register (see Table 41) if it is executed and the condition described by the event flags is true. The execution of RPS continues. The format of the Interrupt command is shown in Tables 24 and 25.

The OAN bit specifies if the condition in bits 25 to 0 is an AND (OAN set to 0) or if the condition is an OR (OAN set to 1). If the INV bit is set this command will wait for the condition to become false.

7.4.4.8 STOP

The STOP command will terminate the RPS execution and reset the ERPS-bit. The command specific part of the STOP command is like the INTERRUPT command. If the addressed event is true the STOP will be executed otherwise the execution will continue with the next command. If no event is addressed the STOP will be executed unconditionally. The format of the STOP command is shown in Tables 26 and 27.

The OAN bit specifies if the condition in bits 25 to 0 is an AND (OAN set to 0) or if the condition is an OR (OAN set to 1). If the INV bit is set this command will wait for the condition to become false.

7.4.4.9 JUMP

The JUMP command is a two Dword command. The second Dword contains the physical address at which the RPS will continue its execution. The address in the second Dword is directly transferred to the RPSAddr Register. The command specific part in the first Dword of the JUMP command is like the INTERRUPT command. If the addressed event is true the JUMP will be performed otherwise the execution will continue at the next command. If no event is addressed the JUMP will be unconditional. The format of the JUMP command is shown in Tables 28 and 29.

The OAN bit specifies if the condition in bits 25 to 0 is an AND (OAN set to 0) or if the condition is an OR (OAN set to 1). If the INV bit is set this command will wait for the condition to become false.

Table 14 PAUSE command format

D31 TO D28	D27	D26	D25	D24	D23	D22	D21	D20	D19	D18	D17	D16
0010	OAN	INV	SIG4	SIG3	SIG2	SIG1	SIG0	GPIO3	GPIO2	GPIO1	GPIO0	HT

Table 15 PAUSE command format (continued)

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
HS	O_FID_B	E_FID_B	O_FID_A	E_FID_A	VBI_A	VBI_B	BRS_DONE	–	HPS_ LINE_ DONE	HPS_DONE	VTD3	VTD2	VTD1	DEBID	IICD

Table 16 Upload command format

D31 TO D28	D25 TO D11	D10	D9	D8	D7	D6
0100	reserved	video data stream handling at Port D1_A (54H); see Table 68	video data stream handling at Port D1_B (54H); see Table 67	BRS control register (58H); see Table 69	reserved	horizontal-prescale (68H); see Table 79. horizontal fine-scale (6CH); see Table 81. BCS control (70H); see Table 82
		initial setting of Dual D1 Interface (50H)				

Table 17 Upload command format (continued)

D5	D4	D3	D2	D1	D0
HPS control (5CH); see Table 71. HPS vertical scale (60H); see Table 72. HPS vertical scale and gain (64H); see Table 73. Chroma key range (74H); see Table 86. HPS output and formats (78H); see Table 87. Clip control (78H); see Table 89.	video DMA3 (30H, 34H, 38H, 3CH, 40H, 44H, 48H); [20 to 16]	video DMA2 (18H, 1CH, 20H, 24H, 28H, 2CH, 48H); [12 to 8]	video DMA1 (00H, 04H, 08H, 0CH, 10H, 14H, 48H); [4 to 0]	DEBI (88H, 7CH, 80H, 84H, 48H); [28 to 26]	IIC (8CH, 90H)

Table 18 CHECK_LATE Command Dword format

D31 TO D28	D27	D26	D25	D24	D23	D22	D21	D20	D19	D18	D17	D16
0011	OAN	INV	SIG4	SIG3	SIG2	SIG1	SIG0	GPIO3	GPIO2	GPIO1	GPIO0	HT

Table 19 CHECK_LATE Command Dword format (continued)

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
HS	O_FID_B	E_FID_B	O_FID_A	E_FID_A	VBI_A	VBI_B	BRS_DONE	–	HPS_ LINE_ DONE	HPS_DONE	VTD3	VTD2	VTD1	DEBID	IICD

Table 20 CLR_SIGNAL Command format

D31 TO D28	D27 TO D26	D25	D24	D23	D22	D21	D20	D19	D18	D17	D16
0000	reserved	SIG4	SIG3	SIG2	SIG1	SIG0	GPIO3	GPIO2	GPIO1	GPIO0	HT

Table 21 CLR_SIGNAL Command format (continued)

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
HS	O_FID_B	E_FID_B	O_FID_A	E_FID_A	VBI_A	VBI_B	BRS_DONE	–	HPS_ LINE_ DONE	HPS_DONE	VTD3	VTD2	VTD1	DEBID	IICD

Table 22 SET_SIGNAL Command format

D31 TO D28	D27 TO D26	D25	D24	D23	D22	D21	D20	D19	D18	D17	D16
0001	reserved	SIG4	SIG3	SIG2	SIG1	SIG0	GPIO3	GPIO2	GPIO1	GPIO0	HT

Table 23 SET_SIGNAL Command format (continued)

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
HS	O_FID_B	E_FID_B	O_FID_A	E_FID_A	VBI_A	VBI_B	BRS_DONE	–	HPS_ LINE_ DONE	HPS_DONE	VTD3	VTD2	VTD1	DEBID	IICD

–

Table 24 INTERRUPT Command format

D31 TO D28	D27	D26	D25	D24	D23	D22	D21	D20	D19	D18	D17	D16
0110	OAN	INV	SIG4	SIG3	SIG2	SIG1	SIG0	GPIO3	GPIO2	GPIO1	GPIO0	HT

Table 25 INTERRUPT Command format (continued)

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
HS	O_FID_B	E_FID_B	O_FID_A	E_FID_A	VBI_A	VBI_B	BRS_DONE	–	HPS_ LINE_ DONE	HPS_DONE	VTD3	VTD2	VTD1	DEBID	IICD

Table 26 STOP Command format

D31 TO D28	D27	D26	D25	D24	D23	D22	D21	D20	D19	D18	D17	D16
0101	OAN	INV	SIG4	SIG3	SIG2	SIG1	SIG0	GPIO3	GPIO2	GPIO1	GPIO0	HT

Table 27 STOP Command format (continued)

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
HS	O_FID_B	E_FID_B	O_FID_A	E_FID_A	VBI_A	VBI_B	BRS_DONE	–	HPS_ LINE_ DONE	HPS_DONE	VTD3	VTD2	VTD1	DEBID	IICD

Table 28 JUMP Command format

D31 TO D28	D27	D26	D25	D24	D23	D22	D21	D20	D19	D18	D17	D16
1000	OAN	INV	SIG4	SIG3	SIG2	SIG1	SIG0	GPIO3	GPIO2	GPIO1	GPIO0	HT

Table 29 JUMP Command format (continued)

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
HS	O_FID_B	E_FID_B	O_FID_A	E_FID_A	VBI_A	VBI_B	BRS_DONE	–	HPS_ LINE_ DONE	HPS_DONE	VTD3	VTD2	VTD1	DEBID	IICD

7.4.4.10 LDREG and STREG

The Load Register (LDREG) command has a variable Dword count specified by the Block_length. It is at least two Dwords long and at maximum 256 Dwords.

The LDREG command interprets the following Dwords as data and writes it to the registers beginning at the specified register address (D6 to D0).

The Store Register (STREG) command is a two Dword command. It transfers the contents of the addressed (D6 to D0) SAA7146A register into PCI memory that is addressed by interpreting the contents of the next data Dword as the 32-bit target base address.

To perform STREG by two **different** tasks, a kind of arbitration with two semaphore signals is necessary.

The Block_length entry defines the number of data Dwords to be processed by these commands. This enables the access to multiple registers on following addresses within a single RPS command. The value specified must be at least one. If more than one Dword is accessed the register address is incremented each cycle. A value of zero is reserved and the command will be interpreted as NOP.

The register address defines the target register address in Dwords. If this address points to a non-existent register the RPS_RE (read error) bit for the actual task will be set and if enabled an interrupt will be generated. The command will be ignored and the execution of RPS continues.

All reserved bits should be written as zeros and should be ignored during read cycles.

Table 30 LDREG command format

D31 to D28	D27 to D16	D15 to D8	D7	D6 to D0
1001	reserved	Block_length	reserved	register address (register offset divided-by-4)

Table 31 STREG command format

D31 to D28	D27 to D16	D15 to D8	D7	D6 to D0
1010	reserved	Block_length	reserved	register address (register offset divided-by-4)

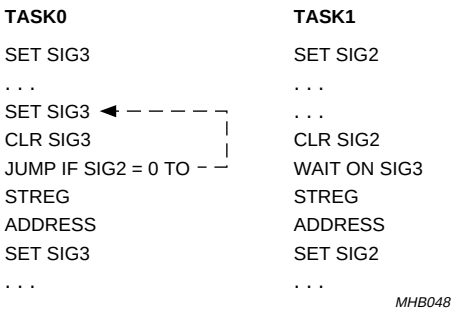


Fig.7 Possible solution employing two semaphore signals to perform STREG commands with two tasks.

7.4.4.11 MASKLOAD

The MASKLOAD command is a three Dword command. Its purpose is to modify only portions or selected bits of a SAA7146A register. The first Dword of the command contains the instruction code and specifies the register to be modified. The second Dword contains the mask and the third Dword contains the data to be written to the register through this mask. The mask works as follows: if a bit in the mask is set, the data from the third Dword at the corresponding bit position will be transferred to the register. If a bit in the mask is zero, the corresponding bit in the register will remain unchanged.

Table 32 MASKLOAD command first Dword

D31 to D28	D27 to D7	D6 to D0
1100	Reserved	register address (register offset divided-by-4)

7.4.5 OPERATION

The operation of the RPS is controlled by the enable bits in the main control register 1 (see Table 10). If one of these bits is set the related RPS task starts its execution with the command addressed by the task related RPS_ADDR register.

When a RPS task is switched on it immediately starts fetching its data via DMA, beginning at the actual address pointers location. Four Dwords are fetched at a time and loaded into an instruction queue. Operation continues to the end of the queue at the time the RPS DMA loads the next four Dwords in the RPS list.

To monitor the ongoing execution and the end of RPS there are status and interrupt bits for each task in the Primary Status Register (PSR) and the Secondary Status Register (SSR), see Tables 38 and 39.

7.4.6 RPS ADDRESS REGISTER

The start address of the RPS list of each task is defined in the RPS address register of the task. The start address must be Dword aligned.

During an RPS list execution this register works like a program counter. Since the RPS can write data into the main memory of the system a protection mechanism is implemented. There is a 4-kbyte page in the memory for each task in which the RPS tasks are allowed to write in. Every write access outside this page will cause an error and the RPS task will stop immediately. If the corresponding bit in the interrupt enable register is set, an interrupt will be generated. This protection mechanism can be disabled via the Enable RPS Page Register (ERPSPx) bit. This bit is located at bit 0 of the RPS page register. A zero enables page errors. This bit is set to 1 after a reset.

Table 33 RPS address register

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
104	RPS_ADDR0	31 to 2	RW	default value: 0
		1 and 0		00
108	RPS_ADDR1	31 to 2	RW	default value: 0
		1 and 0		00

Table 34 RPS page register

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
C4	RPS_PAGE0	31 to 12	RW	default value: 0
	–	11 to 1	–	reserved
	ERPSP0	0	RW	Enable RPS Page Register 0
C8	RPS_PAGE1	31 to 12	RW	default value: 0
	–	11 to 1	–	reserved
	ERPSP1	0	RW	Enable RPS Page Register 1

7.4.7 LINE COUNTER THRESHOLDS

For the events related to the line counters of the source and the target, (either HPS or BRS) there are two thresholds for each task in the HBI threshold register (see Table 35). The purpose of this register is to set the HS or HT event flag when the corresponding line counter has reached the threshold. These thresholds must be written before waiting on the event. A value of zero as threshold turns the HS or HT event on, for every line.

Table 35 HBI threshold register

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
CC	–	31 to 29	–	reserved
	TLCS0	28	RW	**Target Line Counter Select for Task 0**: this bit defines if the TLCT0 refers to the HPS (logic 0) or to the BRS (logic 1)
	TLCT0	27 to 16	RW	**Target Counter Threshold for Task 0**: specifies the threshold for the target line counter
	–	15 to 13	–	reserved
	SLCS0	12	RW	**Source Line Counter Select for Task 0**: the bit defines if the SLCT0 refers to the HPS (logic 0) or to the BRS (logic 1)
	SLCT0	11 to 0	RW	**Source Line Counter Threshold for Task 0**: specifies the threshold for the source line counter
D0	–	31 to 29	–	reserved
	TLCS1	28	RW	**Target Line Counter Select for Task 1**: this bit defines if the TLCT refers to the HPS (logic 0) or to the BRS (logic 1)
	TLCT1	27 to 16	RW	**Target Line Counter Threshold for Task 1**: specifies the threshold for the target line counter
	–	15 to 13	–	reserved
	SLCS1	12	RW	**Source Line Counter Select for Task 1**: this bit defines if the SLCT1 refers to the HPS (logic 0) or to the BRS (logic 1)
	SLCT1	11 to 0	RW	**Source Line Counter Threshold for Task 1**: specifies the threshold for the source line counter

7.4.8 RPS TIME OUT VALUE

These registers contain the values for the time out conditions of the PAUSE and CHECK_LATE commands for each task. If the selected counter value is zero, the time out generation is disabled.

Table 36 RPS time out value

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
D4	V_TO0	31	RW	these two bits determine how the RPS_TO0 is generated; see Table 37
	C_TO0	30	RW	
	V_ABN0	29	RW	this bit determines which port the V_sync for the time out check comes from: a logic 1 selects Port A; a logic 0 selects Port B
	–	28	–	reserved
	Vsync_Cnt0	27 to 24	RW	this is a 4-bit value which sets the V_sync time out between 1 and 15 V_syncs
	PCI_Cnt0	23 to 0	RW	this value specifies after how many PCI clocks a time out should be detected
D8	V_TO1	31	RW	these two bits determine how the RPS_TO1 is generated; see Table 37
	C_TO1	30	RW	
	V_ABN1	29	RW	this bit determines which port the V_sync for the time out check comes from: a logic 1 selects Port A; a logic 0 selects Port B
	–	28	–	reserved
	Vsync_Cnt1	27 to 24	RW	this is a 4-bit value which sets the V_sync time out between 1 and 15 V_syncs
	PCI_Cnt1	23 to 0	RW	this value specifies after how many PCI clocks a time out should be detected

Table 37 RPS_TOX generation

V_TOX	C_TOX	RPS_TOX GENERATED FORMAT
0	0	no time out check
0	1	PCI clock time out check
1	0	V_sync time out check
1	1	both time out checks

7.5 Status and interrupts

7.5.1 General

In order to control the SAA7146A, the status information is collected and stored in two status registers: Primary Status Register (PSR) and Secondary Status Register (SSR). These two registers follow a hierarchical approach because the PSR contains summed up information from the SSR. Interrupts can only be generated from the PSR and are enabled via the Interrupt Enable Register (IER). If an interrupt condition occurs and the interrupt is enabled, the corresponding bit in the Interrupt Status Register (ISR) is set. These bits can be cleared by writing a logic 1.

Both status registers are read only. Writing a logic 1 into any of the PSR bits causes the corresponding interrupt to be generated if enabled. Writing a logic 0 has no effect.

Table 38 Primary status register

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION	RESET
110	PPEF	31	R	**PCI Parity Error**: this bit is set when a PCI Parity Error occurs during any transfer other than ‘real time video data’. The bit in the ISR is set on the rising edge of this status bit.	ISR [31]
	PABO	30	R	**PCI Access Error**: this bit is set when the PCI interface starts an access, and has either a target or master abort. The bit in the ISR is set on the rising edge of this status bit.	ISR [30]
	PPED	29	R	**PCI Parity Errors on ‘real time Data’**: this bit is set when a parity error has occurred since the last Vsync or under RPS since the last wait.	–
	RPS_I1	28	R	**Interrupt issued by RPS command from Task 1**.	–
	RPS_I0	27	R	**Interrupt issued by RPS command from Task 0**.	–
	RPS_late1	26	R	**RPS Task 1 late**: this is set by the CHECK_LATE command. This bit is reset by starting a new RPS Task 1.	–
	RPS_late0	25	R	**RPS Task 0 late**: this is set by the CHECK_LATE command. This bit is reset by starting a new RPS Task 0.	–
	RPS_E1	24	R	**RPS_Error Task 1**: this bit reflects the status of the RPS error bits for Task 1 in the secondary status register (see Table 39). This bit is reset by starting a new RPS Task 1.	–
	RPS_E0	23	R	**RPS_Error Task 0**: this bit reflects the status of the RPS error bits for Task 0 in the secondary status register (see Table 39). This bit is reset by starting a new RPS Task 0.	–
	RPS_TO1	22	R	**RPS time out error in Task 1**: this bit is set when the RPS Task 1 stays longer than expected in the WAIT state. This bit is reset by starting a new RPS Task 1.	–

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION	RESET
110	RPS_TO0	21	R	**RPS time out error in Task 0**: this bit is set when the RPS Task 0 stays longer than expected in the WAIT state. This bit is reset by starting a new RPS Task 0.	–
	UPLD	20	R	**RPS in UPLOAD**: this bit is active while RPS uploads the working registers from the shadow RAM. The bit in the ISR is set on the falling edge of this status bit.	–
	DEBI_S	19	R	**DEBI Status**: this bit stays set as long as DEBI is processing or halted by an error. The bit in the ISR is set on the falling edge of this status bit, which indicates a 'DEBI Done'.	–
	DEBI_E	18	R	**DEBI Event**: this bit is set when one of the two DEBI event flags (DEBI_EF or DEBI_TO) in the SSR is set. This bit is reset when a new DEBI command starts. The reset value of DEBI_TO is a logic 1.	–
	IIC_S	17	R	**I^2C-bus Status**: this bit stays set as long as the I^2C-bus is transmitting data or halted by an error. The bit in the ISR is set on the falling edge of this status bit, which indicates an 'I^2C Done'.	–
	IIC_E	16	R	**I^2C-bus Error**: this bit gets set when one of the I^2C-bus status bits in the SSR is set. This bit is reset when a new I^2C-bus transfer starts.	–
	A2_in	15	R	**Audio input DMA2 protection**: this bit is set when the audio input DMA2 address generation exceeded an 'address boundary' or hit its 'limit' (protection address). It is reset with starting the DMA channel again.	–
	A2_out	14	R	**Audio output DMA2 protection**: this bit is set when the audio output DMA2 address generation exceeded an 'address boundary' or hit its 'limit' (protection address). It is reset with starting the DMA channel again.	–
	A1_in	13	R	**Audio input DMA1 protection**: this bit is set when the audio input DMA1 address generation exceeded an 'address boundary' or hit its 'limit' (protection address). It is reset with starting the DMA channel again.	–
	A1_out	12	R	**Audio output DMA1 protection**: this bit is set when the audio output DMA1 address generation exceeded an 'address boundary' or hit its 'limit' (protection address). It is reset with starting the DMA channel again.	–
	AFOU	11	R	**Audio FIFO Overflow/Underflow**: this bit gets set when one of the four audio FIFOs has an underflow or overflow.	–
	V_PE	10	R	**Video address Protection Error**: this bit is set when one of the video DMAs 1 to 3 has an address protection error during an active transmission.	–
	VFOU	9	R	**Video FIFO Overflow/Underflow**: this bit is set if any of the video FIFOs 1, 2 or 3 has an overflow or underflow.	–

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION	RESET
110	FIDA	8	R	**Field ID Port A**: via the FIDESA bits in the 'Initial setting of the Dual D1 Interface' (see Table 66), selected edge(s) of this signal will set the corresponding bit in the ISR when enabled.	–
	FIDB	7	R	**Field ID Port B**: via the FIDESB bits in the 'Initial setting of the Dual D1 Interface' (see Table 66), selected edge(s) of this signal will set the corresponding bit in the ISR when enabled.	–
	PIN3	6	R	**GPIO Pin 3**: this bit reflects the state of the general purpose pin 3. Via the GPIO register, selected edge(s) of this signal will set the corresponding bit in the ISR when enabled.	–
	PIN2	5	R	**GPIO Pin 2**: this bit reflects the state of the general purpose pin 2. Via the GPIO register, selected edge(s) of this signal will set the corresponding bit in the ISR when enabled.	–
	PIN1	4	R	**GPIO Pin 1**: this bit reflects the state of the general purpose pin 1. Via the GPIO register selected edge(s) of this signal will set the corresponding bit in the ISR when enabled.	–
	PIN0	3	R	**GPIO Pin 0**: this bit reflects the state of the general purpose pin 0. Via the GPIO register selected edge(s) of this signal will set the corresponding bit in the ISR when enabled.	–
	ECS	2	R	**Event Counter Status**: this bit reflects the status of the four (SSR) event counter status bits EC5S, EC4S, EC2S and EC1S.	–
	EC3S	1	R	**Event Counter 3 Status**: this bit is set when event counter 3 exceeds its threshold.	–
	EC0S	0	R	**Event Counter 0 Status**: this bit is set when event counter 0 exceeds its threshold.	–

Table 39 Secondary status register

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
114	PRQ	31	R	**PCI Request Pending**: this bit is set while the PCI has asserted its REQ# signal and has not received a GNT# yet
	PMA	30	R	**PCI master access**: this bit is active as long as the SAA7146A acts as a master on the PCI-bus
	RPS_RE1	29	R	**RPS Task 1 Register access Error**: this bit is set when the LDREG, STREG or MASKWRITE command tries to access a non-existing register. This bit is reset by writing a logic 1 to the RPS_E1 bit in the ISR or when a new RPS Task 1 is started.
	RPS_PE1	28	R	**RPS Task 1 Page Error**: this bit is set when the RPS Task 1 tries to write to an address outside the 4-kbyte page. This bit is reset by writing a logic 1 to the RPS_E1 bit in the ISR or when a new RPS Task 1 is started.
	RPS_A1	27	R	**RPS Task 1 Active**: this bit is set whenever RPS Task 1 is executing and not staying in a wait condition or uploading the working registers

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
114	RPS_RE0	26	R	**RPS Task 0 Register access Error**: this bit is set when the LDREG, STREG or MASKWRITE command tries to access a non-existing register. This bit is reset by writing a logic 1 to the RPS_E0 bit in the ISR or when a new RPS Task 0 is started.
	RPS_PE0	25	R	**RPS Task 0 Page Error**: this bit is set when the RPS Task 0 tries to write-access an address outside the 4-kbyte page. This bit is reset by writing a logic 1 to the RPS_E0 bit in the ISR or when a new RPS Task 0 is started.
	RPS_A0	24	R	**RPS Task 0 Active**: this bit is set whenever RPS Task 0 is executing and not staying in a wait condition or uploading the working registers
	DEBI_TO	23	R	**DEBI Time Out**: this bit is set when the TIMEOUT value was reached. This bit is reset by writing a logic 1 to the DEBI_E bit in the ISR. Reset value is a logic 1.
	DEBI_EF	22	R	**DEBI Format Error**: this bit indicates an illegal command to immediate transfer across a Dword boundary. This bit is reset by writing a logic 1 to the DEBI_E bit in the ISR.
	IIC_EA	21	R	**I^2C-bus Address Error**: this bit is set when there is no acknowledge after the device address. This bit is reset by writing a logic 1 to the IIC_E bit in the ISR or when a new I^2C-bus command starts.
	IIC_EW	20	R	**I^2C-bus Write data Error**: this bit is set when there is no acknowledge during the writing of the data byte(s). This bit is reset by writing a logic 1 to the IIC_E bit in the ISR or when a new I^2C-bus command starts.
	IIC_ER	19	R	**I^2C-bus Read data Error** This bit is set when there is no acknowledge during reading of the data byte(s). This bit is reset by writing a logic 1 to the IIC_E bit in the ISR or when a new I^2C-bus command starts.
	IIC_EL	18	R	**I^2C-bus Loss arbitration Error**: this bit is set when the I^2C-bus loses its arbitration. This bit is reset by writing a logic 1 to the IIC_E bit in the ISR or when a new I^2C-bus command starts.
	IIC_EF	17	R	**I^2C-bus Frame Error**: this bit is set when there is an invalid START/STOP condition since the last I^2C-bus command. This bit is reset by writing a logic 1 to the IIC_E bit in the ISR or when a new I^2C-bus command starts.
	V3P	16	R	**Video DMA 3 Protection error**: this bit is set when video DMA3 generates an address during an active transmission beyond its protection address. This bit is reset by writing a logic 1 to the V_PE bit in the ISR or by reloading the DMA base address.
	V2P	15	R	**Video DMA 2 Protection error**: this bit is set when video DMA2 generates an address during an active transmission beyond its protection address. This bit is reset by writing a logic 1 to the V_PE bit in the ISR or by reloading the DMA base address.
	V1P	14	R	**Video DMA 1 Protection error**: this bit is set when video DMA1 generates an address during an active transmission beyond its protection address. This bit is reset by writing a logic 1 to the V_PE bit in the ISR or by reloading the DMA base address.
	VF3	13	R	**Video FIFO 3 underflow/overflow**: this bit is set when the video FIFO 3 has an overflow/underflow. This bit is reset when reloading the DMA base address or by writing a logic 1 to the VFOU bit in the ISR.

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
114	VF2	12	R	**Video FIFO 2 underflow/overflow**: this bit is set when the video FIFO 2 has an overflow/underflow. This bit is reset when reloading the DMA base address or by writing a logic 1 to the VFOU bit in the ISR.
	VF1	11	R	**Video FIFO 1 overflow**: this bit is set when the video FIFO 1 has an overflow. This bit is reset when reloading the DMA base address or by writing a logic 1 to the VFOU bit in the ISR.
	AF2_in	10	R	**Audio input FIFO 2 underflow**: this bit is set when the audio input FIFO 2 has an underflow. This bit is reset by restarting the DMA channel or by writing a logic 1 to the AFOU bit in the ISR.
	AF2_out	9	R	**Audio output FIFO 2 overflow**: this bit is set when the audio output FIFO 2 has an overflow. This bit is reset by restarting the DMA channel or by writing a logic 1 to the AFOU bit in the ISR.
	AF1_in	8	R	**Audio input FIFO 1 underflow**: this bit is set when the audio input FIFO 1 has an underflow. This bit is reset by restarting the DMA channel or by writing a logic 1 to the AFOU bit in the ISR.
	AF1_out	7	R	**Audio output FIFO 1 overflow**: this bit is set when the audio output FIFO 1 has an overflow. This bit is reset by restarting the DMA channel or by writing a logic 1 to the AFOU bit in the ISR.
	–	6	–	reserved
	VGT	5	R	**Vertical Gate**: this bit reflects the vertical gate at the HPS output
	LNQG	4	R	**Line Qualifier Gate**: this bit reflects the horizontal gate at the HPS output
	EC5S	3	R	**Event Counter 5 Status**: this bit is set when the event counter 5 exceeds its threshold. This bit is reset by writing a logic 1 to the ECS bit in the ISR.
	EC4S	2	R	**Event Counter 4 Status**: this bit is set when event counter 4 exceeds its threshold. This bit is reset by writing a logic 1 to the ECS bit in the ISR.
	EC2S	1	R	**Event Counter 2 Status**: this bit is set when event counter 2 exceeds its threshold. This bit is reset by writing a logic 1 to the ECS bit in the ISR.
	EC1S	0	R	**Event Counter 1 Status**: this bit is set when event counter 1 exceeds its threshold. This bit is reset by writing a logic 1 to the ECS bit in the ISR.

Table 40 Interrupt enable register

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
DC	PPEF	31	RW	PCI Parity Error interrupt enable
	PABO	30	RW	PCI Access Error interrupt enable
	PPED	29	RW	PCI Parity Errors on ‘real time Data’ interrupt enable
	RPS_I1	28	RW	enables interrupts issued by RPS commands in Task 1
	RPS_I0	27	RW	enables interrupts issued by RPS commands in Task 0
	RPS_late1	26	RW	RPS Task 1 late interrupt enable
	RPS_late0	25	RW	RPS Task 0 late interrupt enable
	RPS_E1	24	RW	RPS_Error1 interrupt enable
	RPS_E0	23	RW	RPS_Error0 interrupt enable
	RPS_TO1	22	RW	RPS time out Task 1 interrupt enable
	RPS_TO0	21	RW	RPS time out Task 0 interrupt enable
	UPLD	20	RW	RPS Upload interrupt enable
	DEBI_S	19	RW	DEBI Status interrupt enable
	DEBI_E	18	RW	DEBI Error interrupt enable
	IIC_S	17	RW	I^2C Status interrupt enable
	IIC_E	16	RW	I^2C Error interrupt enable
	A2_in	15	RW	Audio input DMA2 protection interrupt enable
	A2_out	14	RW	Audio output DMA2 protection interrupt enable
	A1_in	13	RW	Audio input DMA1 protection interrupt enable
	A1_out	12	RW	Audio output DMA1 protection interrupt enable
	AFOU	11	RW	Audio FIFO Overflow/Underflow interrupt enable
	V_PE	10	RW	Video address Protection Error interrupt enable
	VFOU	9	RW	Video FIFO Overflow/Underflow interrupt enable
	FIDA	8	RW	Field ID port A interrupt enable
	FIDB	7	RW	Field ID port B interrupt enable
	PIN3	6	RW	GPIO Pin 3 interrupt enable
	PIN2	5	RW	GPIO Pin 2 interrupt enable
	PIN1	4	RW	GPIO Pin 1 interrupt enable
	PIN0	3	RW	GPIO Pin 0 interrupt enable
	ECS	2	RW	Event Counter 1, 2, 4 and 5 Status interrupt enable
	EC3S	1	RW	Event Counter 3 Status interrupt enable
	EC0S	0	RW	Event Counter 0 Status interrupt enable

Table 41 Interrupt status register

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
10C	PPEF	31	RW	PCI Parity Error interrupt status
	PABO	30	RW	PCI Access Error interrupt status
	PPED	29	RW	PCI Parity Errors on ‘real time Data’ interrupt status
	RPS_I1	28	RW	interrupt issued by RPS command from Task 1 interrupt status
	RPS_I0	27	RW	interrupt issued by RPS command from Task 0 interrupt status
	RPS_late1	26	RW	RPS Task 1 is late interrupt status
	RPS_late0	25	RW	RPS Task 0 is late interrupt status
	RPS_E1	24	RW	RPS_Error from Task 1 interrupt status
	RPS_E0	23	RW	RPS_Error from Task 0 interrupt status
	RPS_TO1	22	RW	RPS time out Task 1 interrupt status
	RPS_TO0	21	RW	RPS time out Task 0 interrupt status
	UPLD	20	RW	RPS Upload interrupt status
	DEBI_S	19	RW	DEBI Status interrupt status
	DEBI_E	18	RW	DEBI Error interrupt status
	IIC_S	17	RW	I^2C Status interrupt status
	IIC_E	16	RW	I^2C Error interrupt status
	A2_in	15	RW	Audio input DMA2 protection interrupt status
	A2_out	14	RW	Audio output DMA2 protection interrupt status
	A1_in	13	RW	Audio input DMA1 protection interrupt status
	A1_out	12	RW	Audio output DMA1 protection interrupt status
	AFOU	11	RW	Audio FIFO Overflow/Underflow interrupt status
	V_PE	10	RW	Video address Protection Error interrupt status
	VFOU	9	RW	Video FIFO Overflow/Underflow interrupt status
	FIDA	8	RW	Field ID port A interrupt status
	FIDB	7	RW	Field ID port B interrupt status
	PIN3	6	RW	GPIO Pin 3 interrupt status
	PIN2	5	RW	GPIO Pin 2 interrupt status
	PIN1	4	RW	GPIO Pin 1 interrupt status
	PIN0	3	RW	GPIO Pin 0 interrupt status
	ECS	2	RW	Event Counter 1, 2, 4 and 5 interrupt status
	EC3S	1	RW	Event Counter 3 interrupt status
	EC0S	0	RW	Event Counter 0 interrupt status

7.6 General Purpose Inputs/Outputs (GPIO)

7.6.1 General

The SAA7146A has four general purpose I/O pins. For example, they could be used to signal to other devices a power-down mode or to map an internal status bit to it, e.g. to detect a sync lost from the VBLK pin of the SAA7110.

Table 42 GPIO registers

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
E0	GPIO3	31 to 24	RW	GPIO3 control register
	GPIO2	23 to 16	RW	GPIO2 control register
	GPIO1	15 to 8	RW	GPIO1 control register
	GPIO0	7 to 0	RW	GPIO0 control register

Table 43 GPIO control register

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	DESCRIPTION
0	0	0	0	X	X	X	X	input, no interrupt condition
0	0	0	1	X	X	X	X	input, rising edge is interrupt condition
0	0	1	0	X	X	X	X	input, falling edge is interrupt condition
0	0	1	1	X	X	X	X	input, both edges are interrupt condition
0	1	X	0	X	X	X	X	output, fixed constant LOW
0	1	X	1	X	X	X	X	output, fixed constant HIGH
1	0	X	X	X	X	X	X	reserved
1	1	SBA[5]	SBA[4]	SBA[3]	SBA[2]	SBA[1]	SBA[0]	output, monitoring the selected status bits of PSR or SSR; see Table 48

7.7 Event counter

The event counters in the SAA7146A provide the possibility of obtaining a statistical look at the different interrupt sources. For this purpose six counters are implemented in two registers (EC1R and EC2R). Each register contains one 12-bit counter and two 10-bit counters. To be flexible in the information collected in the counters it is possible to map each status bit to any counter. This is done via the Event Counter Source Select Register (ECSSR). The four 10-bit counters and the two 12-bit counters are able to select one of the 64 possible sources (see Table 47). In addition to the counting, it is possible to generate interrupts via threshold values for the counters. These thresholds are kept in the two Event Threshold Registers (ET1R and ET2R). If a counter exceeds its threshold, it is reset to zero and the corresponding status bit is set.

Table 44 Event Counter set 1 Register (EC1R)

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
118	EC2 [9:0]	31 to 22	R	**Event Counter Two**: this is the second 10-bit counter
	EC1 [9:0]	21 to 12	R	**Event Counter One**: this is the first 10-bit counter
	EC0 [1:0]	11 to 0	R	**Event Counter Zero**: this is the first 12-bit counter

Table 45 Event Counter set 2 Register (EC2R)

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
11C	EC5 [9:0]	31 to 22	R	**Event Counter Five**: this is the fourth 10-bit counter
	EC4 [9:0]	21 to 12	R	**Event Counter Four**: this is the third 10-bit counter
	EC3 [11:0]	11 to 0	R	**Event Counter Three**: this is the second 12-bit counter

Table 46 Event Counter set 1 Source Select Register 1 (EC1SSR)

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
E4	–	31 to 24	–	reserved
	ECS2 [5:0]	23 to 18	RW	**Event Counter 2 Source**: this 6 bit value addresses one of the status bits
	ECEN2	17	RW	**Event Counter 2 Enable**: if this bit is set, event counter 2 is enabled
	ECCLR2	16	RW	**Event Counter 2 Clear**: writing a logic 1 to this bit will clear event counter 2
	ECS1 [5:0]	15 to 10	RW	**Event Counter 1 Source**: this 6 bit value addresses one of the status bits
	ECEN1	9	RW	**Event Counter 1 Enable**: if this bit is set event counter 1 is enabled
	ECCLR1	8	RW	**Event Counter 1 Clear**: writing a logic 1 to this bit will clear event counter 1
	ECS0 [5:0]	7 to 2	RW	**Event Counter 0 Source**: this 6 bit value addresses one of the status bits
	ECEN0	1	RW	**Event Counter 0 Enable**: if this bit is set event counter 0 is enabled
	ECCLR0	0	RW	**Event Counter 0 Clear**: writing a logic 1 to this bit will clear event counter 0

Table 47 Event Counter set 2 Source Select Register (EC2SSR)

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
E8	–	31 to 24	–	reserved
	ECS5 [5:0]	23 to 18	RW	**Event Counter 5 Source**: this 6 bit value addresses one of the status bits
	ECEN5	17	RW	**Event Counter 5 Enable**: if this bit is set the event counter 5 is enabled
	ECCLR5	16	RW	**Event Counter 5 Clear**: writing a logic 1 to this bit will clear event counter 5
	ECS4 [5:0]	15 to 10	RW	**Event Counter 4 Source**: this 6 bit value addresses one of the status bits
	ECEN4	9	RW	**Event Counter 4 Enable**: if this bit is set event counter 4 is enabled
	ECCLR4	8	RW	**Event Counter 4 Clear**: writing a logic 1 to this bit will clear event counter 4
	ECS3 [5:0]	7 to 2	RW	**Event Counter 3 Source**: this 6 bit value addresses one of the status bits
	ECEN3	1	RW	**Event Counter 3 Enable**: if this bit is set event counter 3 is enabled
	ECCLR3	0	RW	**Event Counter 3 Clear**: writing a logic 1 to this bit will clear event counter 3

Table 48 Status Bit Addresses (SBA)

ADDRESS (HEX)	STATUS BIT	EVENTS TO BE COUNTED
00	PPEF	number of PCI Parity errors
01	PABO	number of PCI Access errors
02	PPED	every PCI clock cycle with ‘data’ parity error
03	RPS_I1	number of RPS interrupts Task 1

ADDRESS (HEX)	STATUS BIT	EVENTS TO BE COUNTED
04	RPS_I0	number of RPS interrupts Task 0
05	RPS_LATE1	number of RPS late errors for Task 1
06	RPS_LATE0	number of RPS late errors for Task 0
07	RPS_E1	number of RPS errors for Task 1
08	RPS_E0	number of RPS errors for Task 0
09	RPS_TO1	number of time outs for RPS Task 1
0A	RPS_TO0	number of time outs for RPS Task 0
0B	UPLD	time for upload, in PCI clocks
0C	DEBI_S	time DEBI is busy, in PCI clocks
0D	DEBI_E	number of DEBI events in total
0E	IIC_S	time I^2C-bus is busy, in PCI clocks
0F	IIC_E	number of I^2C-bus errors in total
10	A2_in	number of protection hits
11	A2_out	number of protection hits
12	A1_in	number of protection hits
13	A1_out	number of protection hits
14	AFOU	number of audio FIFOs overflows/underflows in total
15	V_PE	number of video FIFO protection violations in total
16	VFOU	number of video FIFOs overflows/underflows in total
17	FIDA	number of odd/even fields on port A (defined via FIDESA)
18	FIDB	number of odd/even fields on port B (defined via FIDESB)
19	PIN3	number of active edges as defined in the GPIO registers; see Table 43
1A	PIN2	number of active edges as defined in the GPIO registers; see Table 43
1B	PIN1	number of active edges as defined in the GPIO registers; see Table 43
1C	PIN0	number of active edges as defined in the GPIO registers; see Table 43
1D	ECS	number of threshold overflows from EC1, EC2, EC4 and EC5 in total
1E	EC3S	number of threshold overflows of EC3S
1F	EC0S	number of threshold overflows of EC0S
20	PRQ	time from REQ# to GNT#, in PCI clocks
21	PMA	time in active master mode, in PCI clocks
22	RPS_RE1	number of RPS register access errors for Task 1
23	RPS_PE1	number of page errors for RPS Task 1
24	RPS_A1	time of RPS Task 1 busy, in PCI clocks
25	RPS_RE0	number of RPS register access errors for Task 0
26	RPS_PE0	number of page errors for RPS Task 0
27	RPS_A0	time of RPS Task 0 busy, in PCI clocks
28	DEBI_TO	number of DEBI time out events
29	DEBI_EF	number of format errors on DEBI port
2A	IIC_EA	number of address errors on the I^2C-bus
2B	IIC_EW	number of I^2C-bus write data errors

ADDRESS (HEX)	STATUS BIT	EVENTS TO BE COUNTED
2C	IIC_ER	number of I^2C-bus read data errors
2D	IIC_EL	number of arbitration losses on the I^2C-bus
2E	IIC_EF	number of I^2C-bus frame errors
2F	V3P	number of protection violations for video FIFO 3
30	V2P	number of protection violations for video FIFO 2
31	V1P	number of protection violations for video FIFO 1
32	VF3	number of missed Dwords
33	VF2	number of missed Dwords
34	VF1	number of missed Dwords
35	AF2_in	number of missed Dwords
36	AF2_out	number of missed Dwords
37	AF1_in	number of missed Dwords
38	AF1_out	number of missed Dwords
39	–	reserved
3A	VGT	number of V_syncs in acquisition of HPS
3B	LNQG	number of output lines
3C	EC5S	number of threshold overflows of EC5
3D	EC4S	number of threshold overflows of EC4
3E	EC2S	number of threshold overflows of EC2
3F	EC1S	number of threshold overflows of EC1

Table 49 Event Counter Threshold set 1 Register (ECT1R)

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
EC	ECT2 [9:0]	31 to 22	RW	**Event Counter 2 Threshold**: this is the threshold for the second 10-bit counter; see note 1
	ECT1 [9:0]	21 to 12	RW	**Event Counter 1 Threshold**: this is the threshold for the first 10-bit counter; see note 1
	ECT0 [11:0]	11 to 0	RW	**Event Counter 0 Threshold**: this is the threshold for the first 12-bit counter; see note 1

Note

1. Each of these threshold values shows the limit up to which the related counter will run before it sets its interrupt status bit.

Table 50 Event Counter Threshold set 2 Register (ECT2R)

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
F0	ECT6 [9:0]	31 to 22	RW	**Event Counter 5 Threshold**: this is the threshold for the fourth 10-bit counter; see note 1
	ECT5 [9:0]	21 to 12	RW	**Event Counter 4 Threshold**: this is the threshold for the third 10-bit counter; see note 1
	ECT4 [11:0]	11 to 0	RW	**Event Counter 3 Threshold**: this is the threshold for the second 12-bit counter; see note 1

Note

1. Each of these threshold values shows the limit up to which the related counter will run before it sets it interrupt status bit.

7.8 Video processing

7.8.1 The real time video interface

The real time video interface consists of two bidirectional 8-bit wide ports transporting colour difference samples and luminance samples in a byte sequential manner. Each of the two video ports (A and B) has its own clock pin, pixel qualifier and horizontal and vertical sync signal pin. The sync signal can be optionally coded in SAV and EAV codes according to the D1 standard (SMPTE125M or CCIR 656). The two 8-bit ports can be combined to form a single 16-bit wide YUV port to be compatible to the DMSD2 output format.

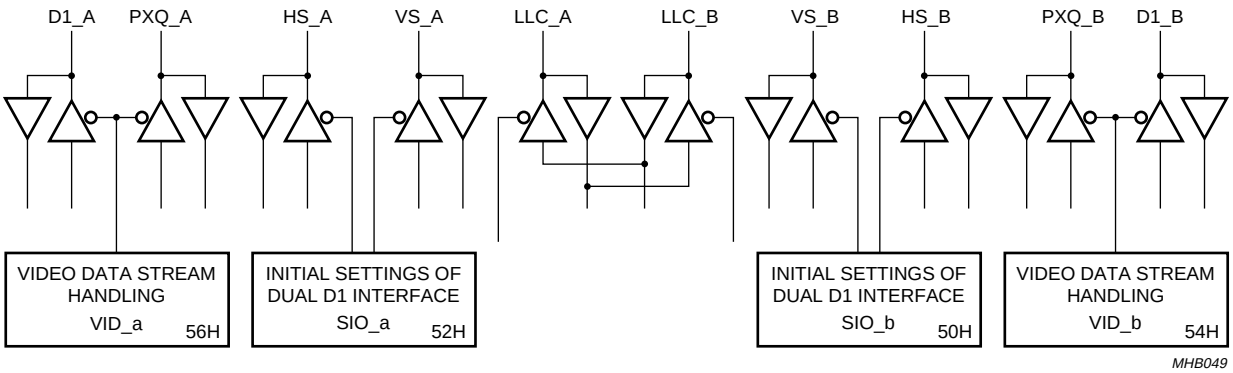


Fig.8 The real time video interface.

7.8.2 DD1: DUAL D1 (CCIR 656, SMPTE125M), I/O

7.8.2.1 Cb-Y-Cr-Y 8-bit wide stream

In this mode two video ports with YUV 4 : 2 : 2 sampling scheme are available. Each D1 port has an I/O capability and has a separate clock input and separate sync lines. In this format the pixel rate is equivalent to the clock rate LLC. The colour difference signal sample and luminance signal sample (straight binary) are byte-wise multiplexed into the same 8-bit wide data stream, with sequence and timing in accordance with CCIR 656 recommendation (respectively according to D1 for 60 Hz application). The incoming and scaled data are reformatted to 16-bit for the HPS data path and the corresponding reference signals are generated. A discontinuous data stream is supported by accepting or generating a pixel/byte qualifying signal (PXQ = 1: qualified pixel, PXQ = 0: invalid data, see Fig.9). The start condition for synchronizing to the correct Cb-Y-Cr-Y sequence is given by the selected horizontal reference signal. The sequence increments only with qualified bytes.

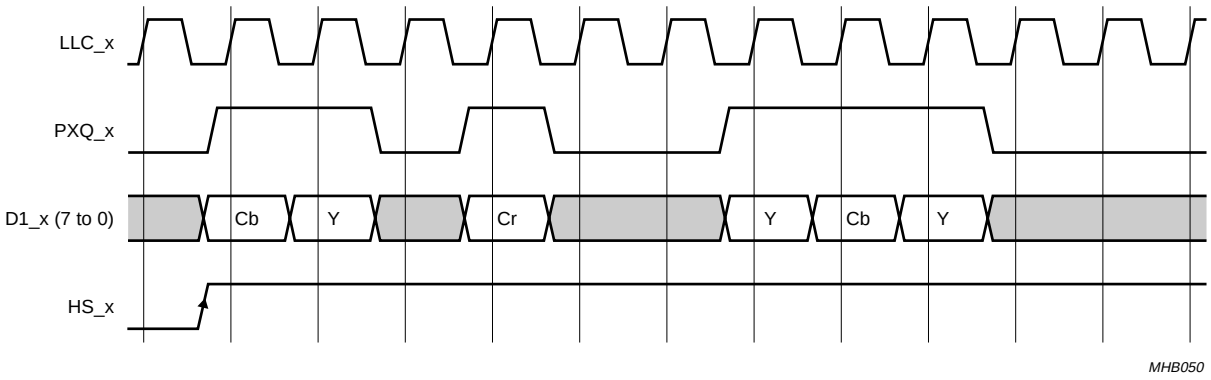


Fig.9 Timing of PXQ_x for serial 8-bit data input at the D1_x port.

7.8.2.2 YUV 16-bit parallel (DMSD2) stream

In this mode only the HPS data path is available since the BRS data path supports only 8-bit wide data streams. Colour difference signal and luminance signal (straight binary) are available in parallel on a 16-bit wide data stream. In this mode both D1 ports are inputs (see Fig.10). With this format the pixel rate is half the clock rate LLC. The start condition for synchronising the clock divider and/or the correct U-V sequence is given by the CREF signal, which must be connected to the same port as the colour difference signal.

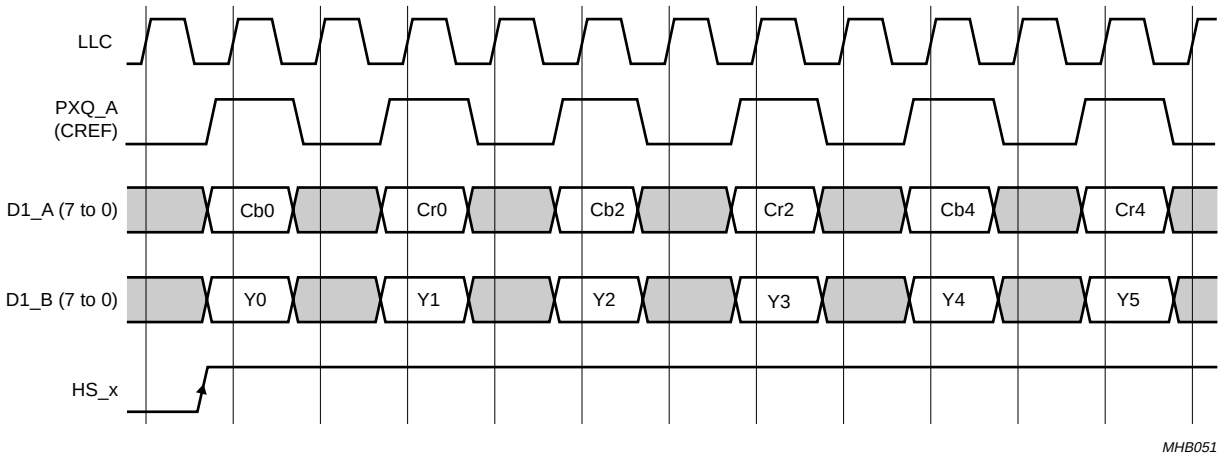


Fig.10 Timing of PXQ_x for 16-bit data input at the D1_x port.

7.8.3 VIDEO DATA FORMATS ON DD1

D1 (SMPTE125M, CCIR 656) as well as YUV16 represent both the same 4 : 2 : 2 sample scheme. Both formats, D1 and YUV16, are assumed to agree with the CCIR recommendation 601 coding:

Y = 16 = black, 0%

Y = 235 = white, 100% brightness

U,V = 128 = no colour, 0% saturation

U,V = 128 ±112 = full colour, 100% saturation.

Data path processing in HPS and BRS is not limited to this range and allows overshoots and uses ‘margins’ for processing. The reference values can be manipulated by the BCS processing in the HPS data path.

7.8.4 VIDEO TIMING REFERENCE CODES (SAV AND EAV)

There are two timing reference codes; one at the beginning of each video data block [Start of Active Video (SAV)] and one at the end of each video data block [End of Active Video (EAV)] as shown in Fig.11.

Each timing reference code consists of a four byte sequence in the following format: FF 00 00 XY. (values are expressed in hexadecimal notation: codes FF, 00 are reserved for use in timing reference codes). The first three bytes are a fixed preamble. The fourth byte contains information defining field identification, the state of field blanking and the state of line blanking. The assignment of bits within the timing reference code is given in Table 51.

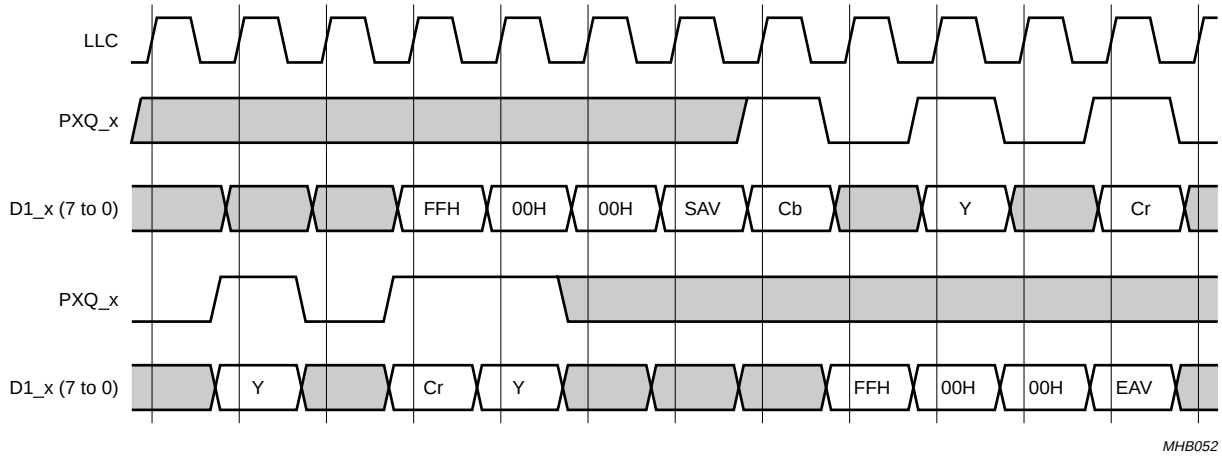


Fig.11 Timing of PXQ_x for CCIR 656 at the D1_x port.

Table 51 Video timing reference codes

BYTE	BIT NUMBER							
	7 (MSB)	**6**	**5**	**4**	**3**	**2**	**1**	**0 (LSB)**
First	1	1	1	1	1	1	1	1
Second	0	0	0	0	0	0	0	0
Third	0	0	0	0	0	0	0	0
Fourth	1	$F^{(1)}$	$V^{(2)}$	$H^{(3)}$	$P_3{}^{(4)}$	$P_2{}^{(4)}$	$P_1{}^{(4)}$	$P_0{}^{(4)}$

Notes

1. F = logic 0 during field 1 and logic 1 during field 2.
2. V = logic 0 elsewhere and logic 1 during field blanking.
3. H = logic 0 in SAV and logic 1 in EAV.
4. P_0, P_1, P_2 and P_3: protection bits (see Table 52).

Bits P_0, P_1, P_2 and P_3, have states dependent on the states of the bits F, V and H as shown in Table 52. At the receiver (SAA7146A) this arrangement permits one-bit errors to be corrected. If two-bit errors or up to four-bit errors occur, i. e. depending on uncoded protection bits, the circuit processes direct on the coded values. In this case the protection bits are ignored.

SAV and EAV are only decoded and removed from the signal stream (substituted with neighbouring first or last active video sample), if chosen this way. However, ‘single’ qualified codes of ‘00’ and/or ‘FF’ in the data stream, remain in the data stream and are processed as data.

Table 52 Protection bits

BIT NUMBER	FUNCTION							
	FIXED 1	F	V	H	P_3	P_2	P_1	P_0
7	1	1	1	1	1	1	1	1
6	0	0	0	0	1	1	0	1
5	0	0	1	1	0	0	1	1
4	0	1	0	1	0	1	0	1
3	0	1	1	0	0	1	1	0
2	0	1	0	1	1	0	1	0
1	0	0	1	1	1	1	0	0
0	0	1	1	0	1	0	0	1

7.8.5 Synchronization signals

Horizontal, vertical and frame synchronization signals are either carried beside the data stream on the extra sync pins of DD1 (one pair of sync pins per D1 channel) or are encoded as SAV and EAV in the 8-bit wide video signal stream. For the 16-bit wide YUV stream sync signals are always available on separate pins. For D1 video inputs the SAA7146A is programmed to determine where to recover the synchronization information (from the dedicated sync pins or from the encoded SAV and EAV codes in the data stream).

For D1 video outputs, the SAA7146A can be programmed to deliver synchronization information both in SAV and EAV codes as well as on the dedicated sync pins. Non-standard rastered video signals are supported by sync signals at the dedicated sync pins as well as via SAV and EAV codes. The number of clock cycles, pixels per line and lines per field can be non-standard. These number can range from 1 up to 4095.

The signal at the HS pin can perform the following functions:

- HS: input only, the rising edge is selected to act as timing reference
- HREF: input only, gated with CREF, the rising edge is selected as timing reference
- HGT: I/O, HIGH during active video
- ACT input only: HIGH during active video, inactive during horizontal and vertical blanking
- HGT and ACT: envelope all active pixels (there is no active pixel outside HGT or ACT), but may also include clock cycles marked as not valid pixels by means of PXQ.

The vertical sync signal can perform the following functions:

- VS: input only positive or negative, one edge is selected as timing reference:
  - If selected edge of VS and selected edge of HS are in phase, then begin 1st (odd) field
  - If selected edges of VS and HS are out of phase, then begin 2nd (even) field.
- V-DMSD: input only, falling (trailing) edge is timing reference:
  - If falling edge of V-DMSD is in high phase of HREF, then begin 1st (odd) field
  - If falling edge of V-DMSD is in low phase of HREF, then begin 2nd (even) field.
- VGT: I/O, HIGH during active video, (no holes for horizontal blanking)
- FS: input only, positive or negative, frame sync, (odd/even), (313/312, 263/262 lines) HIGH in one field, LOW in the other, changes on full line boundaries only.

7.8.6 Field detection

The fields are detected simultaneously at both D1 sync inputs. The results are available in two status registers.

Table 53 Field interval definitions for D1 (CCIR 656) SAV and EAV codes; note 1

DEFINITION	625 LINES	525 LINES
V-digital field blanking		
Field 1; start (V = 1)	Line 624	Line 1
Field 1; finish (V = 0)	Line 23	Line 10
Field 2; start (V = 1)	Line 311	Line 264
Field 2; finish (V = 0)	Line 336	Line 273
F-digital field identification		
Field 1; F = 0	Line 1	Line 4
Field 2; F = 1	Line 313	Line 266

Note

1. Signals F and V change state synchronously with the end of active video timing reference code at the beginning of the digital line.

7.8.6.1 Field detection control

Field detection modes:

- Direct mode: FLD signal detected from incoming H/V signals, for timing behaviour see Fig.11.
- Forced toggle: FLD signal regularly synchronized to source, but will never stay more than two fields with the same ID. The circuit expects to detect a field change with every vertical reference edge, if the field does not change (field error), the circuit change the field ID automatically. If the circuit switch to the wrong sequence i. e. at the beginning of processing, it will be synchronized after one second where no field error has occurred.
- Free toggle: FLD signal toggles with every vertical reference edge, independent of source FID.

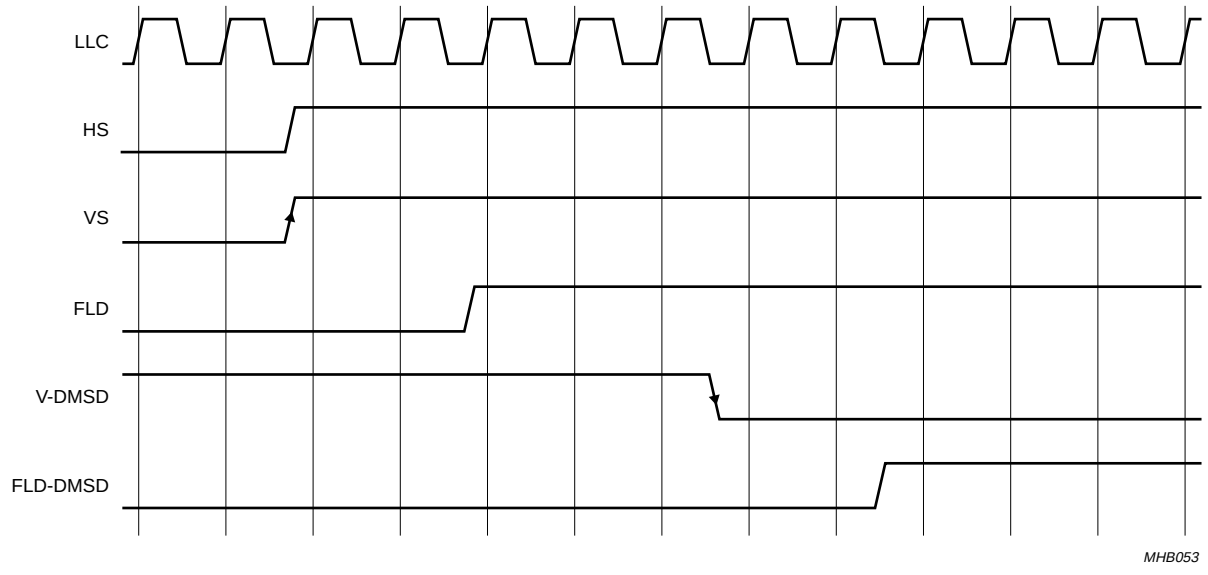


Fig.12 Timing of field detection EVEN-to-ODD for direct mode.

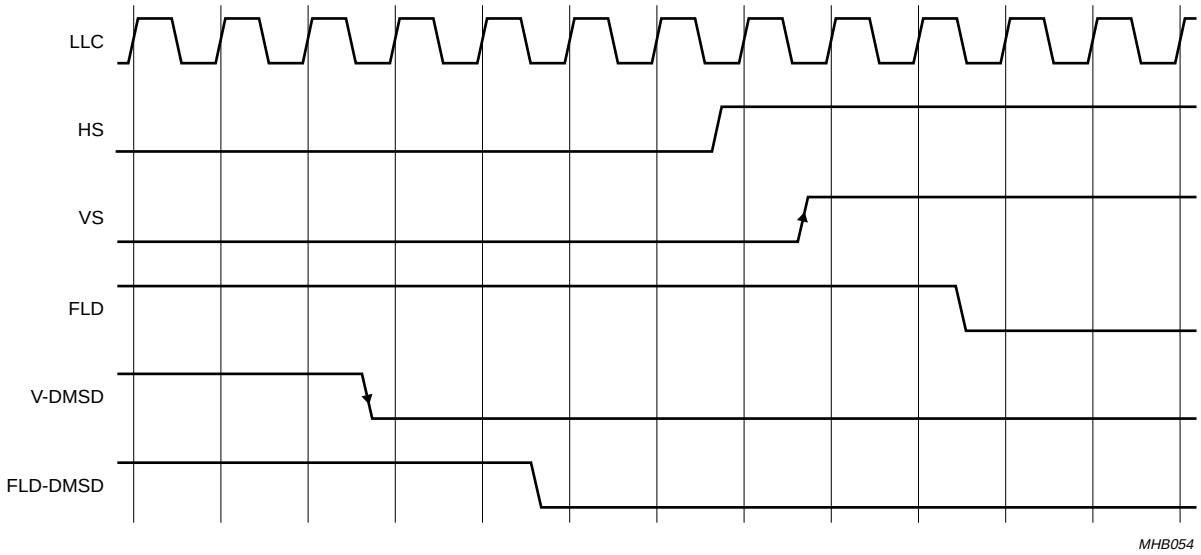
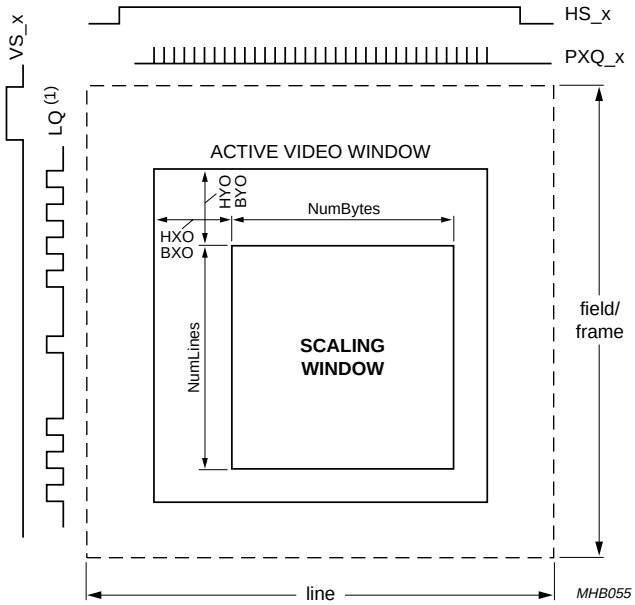


Fig.13 Timing of field detection ODD-to-EVEN for direct mode.

7.8.7 Acquisition control

The processing window for the scaling unit is defined in the acquisition control. The internal counters (one for the HPS and one for the BRS) receives programmable values for offset (HXO11 to HXO0, HYO11 to HYO0 and BXO9 to BXO0, BYO9 to BYO0) and length (NumLines, NumBytes). These counters are reset by the corresponding sync reference input signal. The horizontal counter increments in qualified pixels for the HPS and qualified bytes for the BRS, the vertical counter increments in qualified lines, i.e. lines containing at least one qualified pixel. In order to avoid programming dependent line drop effects, the horizontal offset value must not exceed the number of pixels per line. In order to avoid programming dependent field drop effects, the vertical offset value must not exceed the number of lines per field.

The acquisition provides the possibility to re-program the vertical offset after the previous job is done (EOW at the HPS and BRS is reached). Thus multiple windows can be opened during one field.



(1) LQ = qualified lines, i.e. lines containing at least one qualified pixel.

Fig.14 Reference signals for scaling window.

7.8.8 Comparison between CCIR 656 line and source line counter

This section describes how to choose the vertical offset and how to use the source line counter event for RPS programming for capturing the expected line.

The internal Source Line Counter (SLC) is reset by the selected edge of the vertical sync signal which is provided at port VS_x. The falling and rising edges of this signal are selected by the SYNC_X bits in the 'Initial settings DD1 Port Register' (offset = 50H). Consequently, the behaviour of the SLC depends on the connected vertical sync signal so that different offsets must be selected to capture the expected line. The active video begins in the CCIR 656 line 23 of the video signal; Table 54 lists the different offsets which must be selected to capture the expected line. The subsequent diagrams and tables illustrate the relationship between the different vertical sync signals of the PAL and NTSC standards, the ODD and EVEN field and the internal SLC.

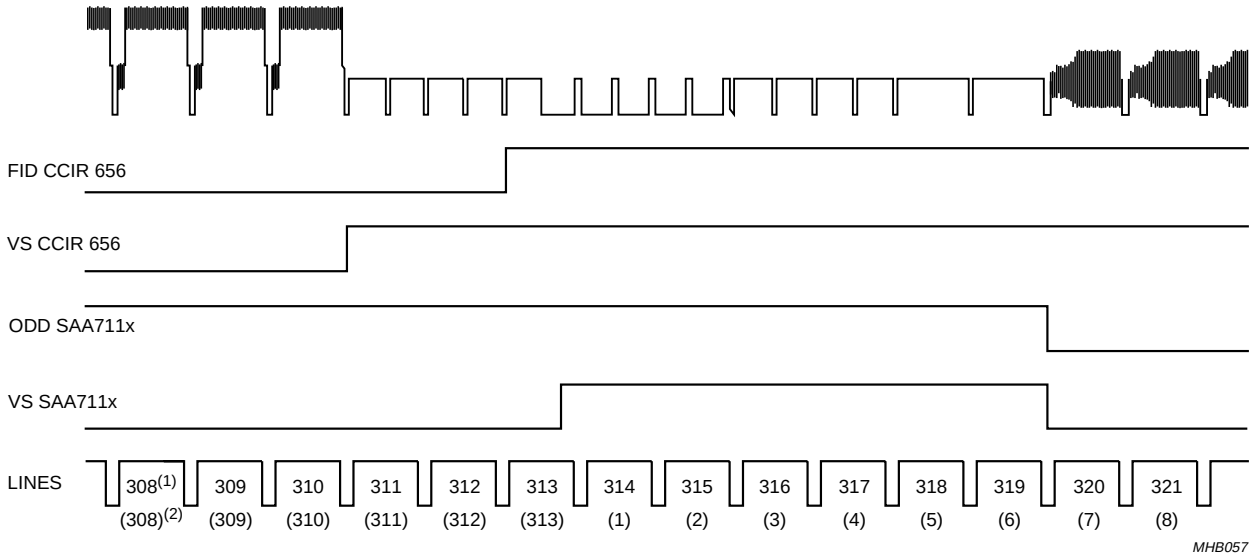
Table 54 Offsets to CCIR 656 line 23 depending on PAL or NTSC source (in compliance with Recommendation 601), ODD and EVEN field and select mode (see note 1)

PAL				NTSC			
SLC[2] SAV/EAV	**SLC ext.[3] FS**	**SLC[4] FALLING VS**	**SLC[5] RISING VS**	**SLC SAV/EAV**	**SLC ext. FS**	**SLC FALLING VS**	**SLC RISING VS**
24 (25) 18H (19H)	15 (16) 0FH (10H)	15H (16) 0FH (10H)	21 (22) 15H (16H)	22 (22) 16H (16H)	12 (13)) 0CH (0DH)	12 (13) 0CH (0DH)	18 (19) 12H (13H)

Notes

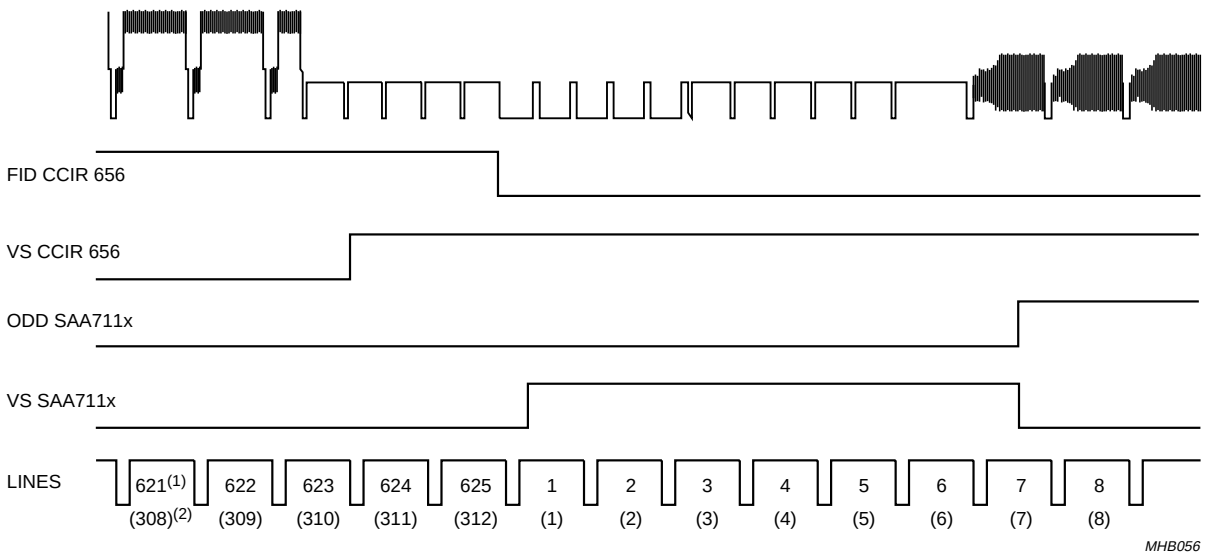
1. Line numbers in parenthesis refer to EVEN field counting.
2. Sync signal SLC with SAV/EAV detection (50H, SYNC_X = 7).
3. Sync signal SLC with external Field Identification Signal (50H, SYNC_X = 6).
4. Sync signal SLC with detection on the falling edge of the vertical sync signal (50H, SYNC_X = 1, 3 or 5).
5. Sync signal SLC with detection on the rising edge of the vertical sync signal. For Philips devices, the rising edge does not include field information, this information is only defined at the falling edge of the VS signal (50H, SYNC_X = 0, 2 or 4).

7.8.8.1 Video with PAL format



(1) The line numbers not in parenthesis refer to CCIR counting.
(2) The line numbers in parenthesis refer to single field counting.

Fig.15 Output timing of SAA711x, 50 Hz, lines 621 to 8.



(1) The line numbers not in parenthesis refer to CCIR counting.
(2) The line numbers in parenthesis refer to single field counting.

Fig.16 Output timing of SAA711x, 50 Hz, lines 308 to 321.

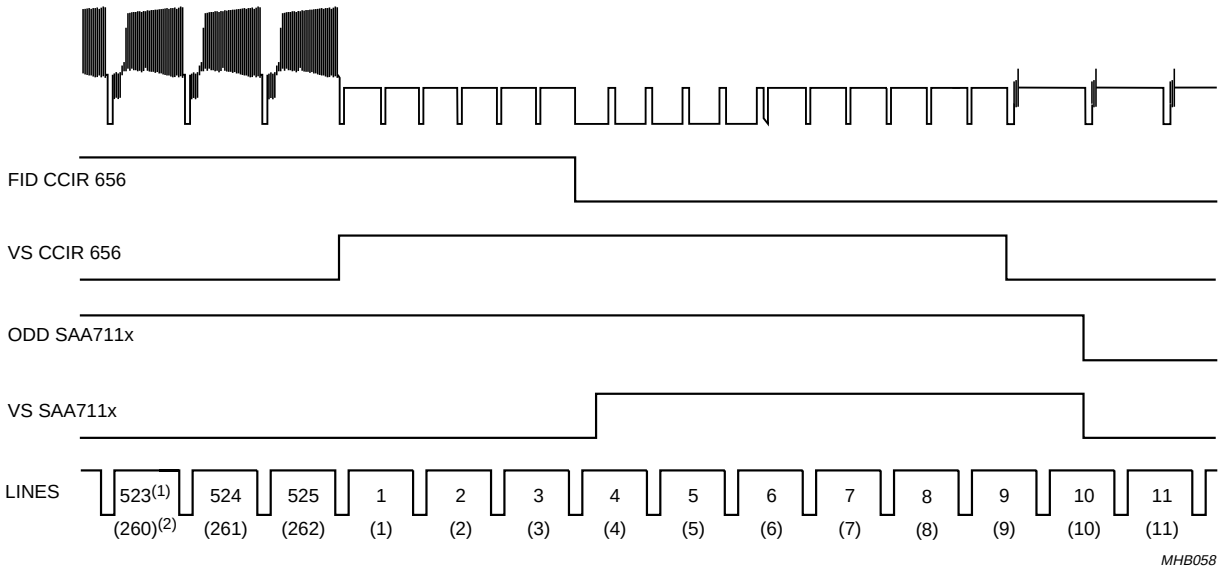
Table 55 Comparison between CCIR 656 lines and the SLC (note 1)

CCIR 656 (D1) LINE[2]	**(612) 310**	**(622) 311**	**(624) 312**	**(625) 313**	**1 (314)**	**2 (315)**	**3 (316)**	**4 (317)**	**5 (318)**	**6 (319)**	**7 (320)**	**8 (321)**	**9 (322)**	**10 (323)**	**11 (324)**	**12 (325)**	**13 (326)**	**14 (328)**	**15 (329)**
SLC[3] SAV/EAV	311 (313)	312 (1)	1 (2)	2 (3)	3 (4)	4 (5)	5 (6)	6 (7)	7 (8)	8 (9)	9 (10)	10 (11)	11 (12)	12 (13)	13 (14)	14 (15)	15 (16)	16 (17)	17 (18)
SLC ext. FS[4]	302 (304)	303 (305)	304 (306)	305 (307)	306 (308)	307 (309)	308 (310)	309 (311)	310 (312)	311 (313)	312 (1)	1 (2)	2 (3)	3 (4)	4 (5)	5 (6)	6 (7)	7 (8)	8 (9)
SLC falling VS[5]	302 (304)	303 (305)	304 (306)	305 (307)	306 (308)	307 (309)	308 (310)	309 (311)	310 (312)	311 (313)	312 (1)	1 (2)	2 (3)	3 (4)	4 (5)	5 (6)	6 (7)	7 (8)	8 (9)
SLC rising VS[6]	308 (310)	309 (311)	310 (312)	311 (313)	312 (1)	1 (2)	2 (3)	3 (4)	4 (5)	5 (6)	6 (7)	7 (8)	8 (9)	9 (10)	10 (11)	11 (12)	12 (13)	13 (14)	14 (15)

Notes

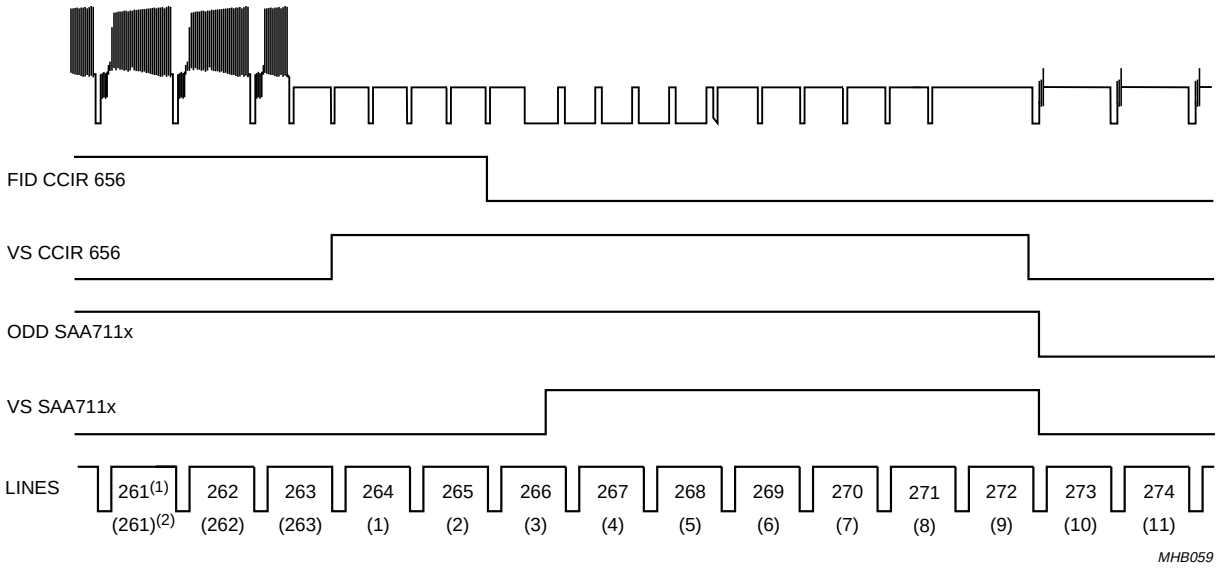
1. Line numbers in parenthesis refer to EVEN field counting.
2. CCIR 656 (D1) line.
3. SLC with SAV/EAV detection (50H, SYNC_X = 7).
4. SLC with external Field Identification Signal (50H, SYNC_X = 6).
5. SLC with detection on the falling edge of the vertical sync signal (50H, SYNC_X = 1, 3 or 5).
6. SLC with detection on the rising edge of the vertical sync signal. For Philips devices, the rising edge does not include field information, this information is only defined at the falling edge of the VS signal (50H, SYNC_X = 0, 2 or 4).

7.8.8.2 Video with NTSC format



(1) The line numbers not in parenthesis refer to CCIR counting.
(2) The line numbers in parenthesis refer to single field counting.

Fig.17 Output timing of SAA711x, 60 Hz, lines 523 to 11.



(1) The line numbers not in parenthesis refer to CCIR counting.
(2) The line numbers in parenthesis refer to single field counting.

Fig.18 Output timing of SAA711x, 60 Hz, lines 261 to 274.

Table 56 Comparison between CCIR 656 lines and SLC (note 1)

CCIR 656 (D1) LINE[2]	**(525) 262**	**1 (264)**	**2 (265)**	**3 (266)**	**4 (267)**	**5 (268)**	**6 (269)**	**7 (270)**	**8 (271)**	**9 (272)**	**10 (273)**	**11 (274)**	**12 (275)**	**13 (276)**	**14 (277)**	**15 (278)**	**16 (279)**	**17 (280)**	**18 (281)**
SLC SAV/EAV[3]	262 (263)	1 (1)	2 (2)	3 (3)	4 (4)	5 (5)	6 (6)	7 (7)	8 (8)	9 (9)	10 (10)	11 (11)	12 (12)	13 (13)	14 (14)	15 (15)	16 (16)	17 (17)	18 (18)
SLC ext. FS[4]	252 (251)	253 (252)	254 (253)	255 (254)	256 (255)	257 (256)	258 (257)	259 (258)	260 (259)	261 (260)	262 (261)	1 (262)	2 (263)	3 (1)	4 (2)	5 (3)	6 (4)	7 (5)	8 (6)
SLC falling VS[5]	252 (251)	253 (252)	254 (253)	255 (254)	256 (255)	257 (256)	258 (257)	259 (258)	260 (259)	261 (260)	262 (261)	1 (262)	2 (263)	3 (1)	4 (2)	5 (3)	6 (4)	7 (5)	8 (6)
SLC rising VS[6]	258 (260)	259 (261)	260 (262)	261 (263)	262 (1)	1 (2)	2 (3)	3 (4)	4 (5)	5 (6)	6 (7)	7 (8)	8 (9)	9 (10)	10 (11)	11 (12)	12 (13)	13 (14)	14 (15)

Notes

1. Line numbers in parenthesis refer to EVEN field counting.
2. CCIR 656 (D1) line.
3. SLC with SAV/EAV detection (50H, SYNC_X = 7).
4. SLC with external Field Identification Signal (50H, SYNC_X = 6).
5. SLC with detection on the falling edge of the vertical sync signal (50H, SYNC_X = 1, 3 or 5).
6. SLC with detection on the rising edge of the vertical sync signal. For Philips devices, the rising edge does not include field information, this information is only defined at the falling edge of the VS signal (50H, SYNC_X = 0, 2 or 4).

7.9 High Performance Scaler (HPS)

Depending on the selected port modes the incoming and scaled data are formatted/reformatted (8-bit or 16-bit), and the corresponding reference signals are generated. Based on these reference signals the active processing window is defined in a versatile way via programming.

The programming register can be loaded during the processing of the previous field, frame or line by RPS. In this way each D1 port gets processed in a field or frame alternating manner. If the incoming signals are not locked, then the acquisition is waiting for the new active video of the subsequent field. The corresponding fields are detected by a 'Field Detection'. To support asynchronous video processing in the two video paths, each D1 port has its own 'Field Detection'. The video signal source is also source for the qualify signal PXQ.

Before being processed in the central scaling unit the incoming data passes to the BCS control unit, where monitor control functions for adjusting Brightness and Contrast (luminance) as well as Saturation (chrominance) are implemented (BCS control). The horizontal scaling is carried out in two steps; a prefiltering (bandwidth limitation for initialising) and a horizontal fine scaling. Between them the vertical processing is performed.

7.9.1 BCS CONTROL

The parameters for brightness, contrast and saturation can be adjusted in the BCS control unit. The luminance signal can be controlled by the bits BRIG7 to BRIG0 and CONT6 to CONT0. The chrominance signal can be controlled by the bits SAT6 to SAT0.

Brightness control (BRIG7 to BRIG0):

- 00H; minimum offset
- 80H; CCIR level
- FFH; maximum offset.

Contrast control (CONT6 to CONT0):

- 00H; luminance off
- 40H; CCIR level
- 7FH; 1.9999 amplitude.

Saturation control (SAT6 to SAT0):

- 00H; colour off
- 40H; CCIR level
- 7FH; 1.9999 amplitude.

Limits: All resulting output values are limited to minimum (equals 0) and maximum (equals 255).

7.9.2 SCALING UNIT

The scaling to a randomly sized window is performed in three steps:

- Horizontal prescaling (bandwidth limitation for anti-aliasing, via FIR prefiltering and subsampling)
- Vertical scaling (generating phase interpolated or vertically low-passed lines)
- Horizontal phase scaling (phase correct scaling to the new geometric relations).

The scaling process generates a new pixel/clock qualifier sequence. There are restrictions in the combination of input sample rate and up or downscaling mode and scaling factor. The maximum resulting output sample rate at the DD1 port is ½LLC, because of compliance to the CCIR 656 format.

7.9.2.1 Horizontal prescaling

The incoming pixels in the selected range are pre-processed in the horizontal prescaler (first stage of the scaling unit). It consists of a FIR prefilter and a pixel collecting subsampler.

7.9.2.2 FIR prefilter

The video components Y, U and V are FIR pre-filtered to reduce the signal bandwidth according to the downscale for factors between 1 and ½, so that aliasing, due to signal bandwidth expansion, is reduced. The prefilter consists of 3 filter stages. The transfer functions are listed in the Section 7.12. The prefilter is controlled by the 'Scaler Register' bits PFY3 to PFY0 and PFUV3 to PFUV0 in the HPS horizontal prescale register (see Table 79).

Figures 19 and 20 show frequency response characteristics and the corresponding scaler register settings. The prefilter operates on YUV 4 : 4 : 4 data. As U and V are generated by simple chroma pixel doubling, the UV prefilter should also be used to generate the interpolated chroma values.

7.9.2.3 Subsampler

To improve the scaling performance for scales less than ½ down to icon size, a FIR filtering subsampler is available. It performs a subsampling of the incoming data by a factor of 1/N, where N = 1 to 64. This operation is controlled by XPSC, where N = XPSC + 1. Where NIP = number of input pixels/line and NOP = number of desired output pixels/line, the basic equation to calculate XPSC is:

XPSC = TRUNC [(NIP/NOP) – 1]

The subsampler collects a number of [XPSC + 2 – XACM] pixels to calculate a new subsampled output pixel. So a downscale dependent FIR filter is built with up to 65 taps which reduces anti-aliasing for small sizes. If XACM = 0, the collecting sequence overlaps which means that the last pixel of sequence M is also the first pixel of sequence M + 1.
To implement a real subsampler bypass, XACM has to be set to logic 1.

As the phase correct horizontal fine scaling is limited to a maximum downscale of $\frac{1}{4}$, this circuitry has to be used for downscales less than $\frac{1}{4}$ of the incoming pixel count.

To get unity gain at the subsamplers output for all subsampling ratios, the scaler register parameters CXY, CXUV and DCGX have to be used. In addition, this can be used to modify the FIR characteristic of the subsampler slightly. Table 57 illustrates examples for scaler register settings, depending on a given prescale ratio. Referring to Table 57(divider in column 'Weight Sum') it should be noted that an internal XPSC depending automatic prenormalization is valid for:

XPSC > 8, > 16, > 32, which reduces the input signal quantization. In addition it should be noted that for XPSC ≥ 15 the LSB of the CXY,CXUV parameter becomes valid.

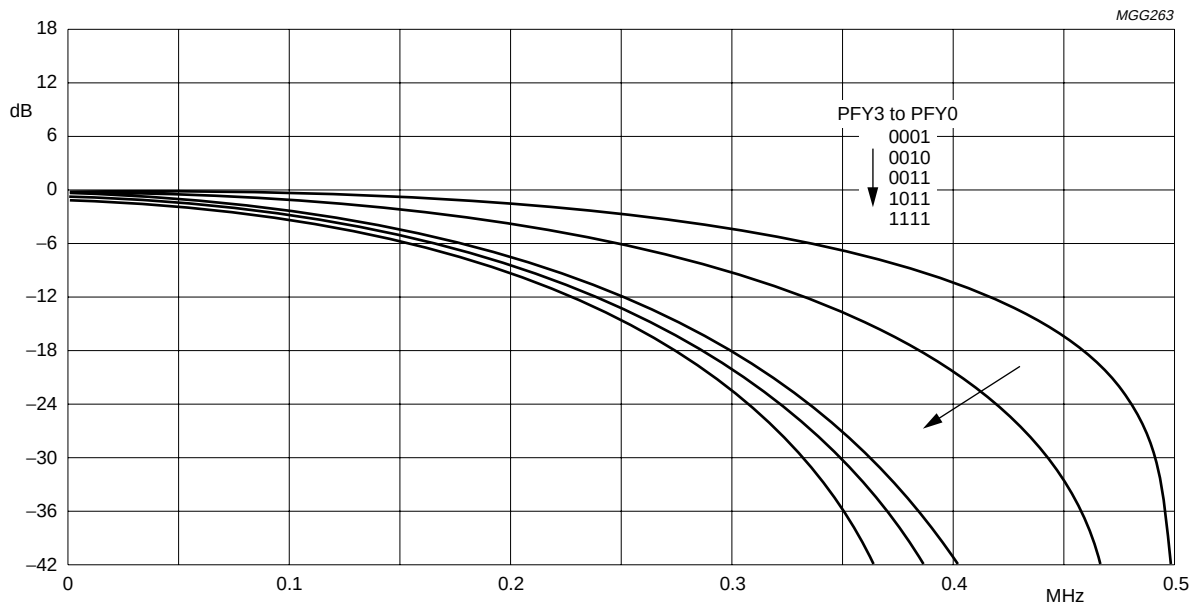


Fig.19 Luminance Prefilter: frequency response for miscellaneous register settings.

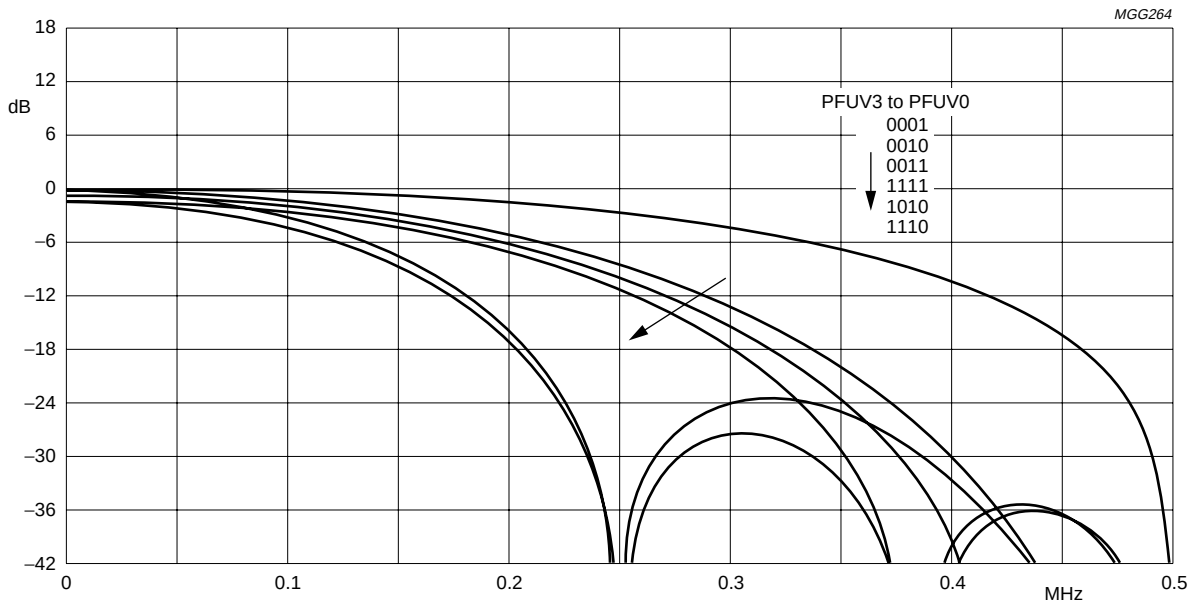


Fig.20 Chrominance Prefilter: frequency response for miscellaneous register settings.

Table 57 Horizontal prescaling and normalization

HORIZONTAL PRESCALER	XPSC	COEFFICIENT SEQUENCE (EXAMPLE)	CXY (LUMINANCE) CXUV (CHROMA)	WEIGHT SUM	DCGX	BCS (CONTR. | SAT.) = X/Y × 64
1	0	1-1	0 0	2	1	1
1/2	1	1-1-1	0 0	3	1	2/3
		1-2-1	0 2	4	2	1
1/3	2	1-1-1-1	0 0	4	2	1
1/4	3	1-1-1-1-1	0 0	5	2	4/5
		1-2-2-2-1	0 6	8	3	1
1/5	4	111 111	0 0	6	2	4/6
		121 121	0 2	8	3	1
		112 211	0 4	8	3	1
1/6	5	111 1 111	0 0	7	3	8/7
		111 2 111	0 8	8	3	1
1/7	6	1111 1111	0 0	8	3	1
1/8	7	1111 1 1111	0 0	9	3	8/9
		1222 2 2221	1 E	16	7	1

HORIZONTAL PRESCALER	XPSC	COEFFICIENT SEQUENCE (EXAMPLE)	CXY (LUMINANCE) CXUV (CHROMA)	WEIGHT SUM	DCGX	BCS (CONTR. | SAT.) = X/Y × 64
1/9	8	1111 1 1 1111	0 0	10/**2**	2	8/10
		1221 2 2 1221	1 6	16/**2**	3	1
		1122 2 2 2211	1 C	16/**2**	3	1
1/10	9	1111 1 1 1 1111	0 0	11/**2**	2	8/11
		1212 1 2 1 2121	2 A	16/**2**	3	1
		1112 2 2 2 2111	3 8	16/**2**	3	1
1/11	10	1111 11 11 1111	0 0	12/**2**	2	8/12
		1211 21 12 1121	1 2	16/**2**	3	1
		1111 22 22 1111	3 0	16/**2**	3	1
1/12	10	1111 11 1 11 1111	0 0	13/**2**	2	8/13
		1121 11 2 11 1211	4 4	16/**2**	3	1
		1111 12 2 21 1111	6 0	16/**2**	3	1
1/13	10	1111 111 111 1111	0 0	14/**2**	2	8/14
		1111 211 112 1111	1 0	16/**2**	3	1
		1111 112 211 1111	4 0	16/**2**	3	1
1/14	10	1111 111 1 111 1111	0 0	15/**2**	2	8/15
		1111 111 2 111 1111	8 0	16/**2**	3	1
1/15	14	1111 1111 1111 1111	0 0	16/**2**	3	1
1/16	15	1111 1111 1 1111 1111	–	17/**2**	3	16/17
		1222 2222 **2** 2222 2221	F **F**	32/**2**	7	1
1/17	16	1111 1111 1 1 1111 1111	0 0	18/**4**	2	16/18
		1222 2222 1 1 2222 2221	F E	32/**4**	3	1
		1222 2122 22 2212 2221	D F	32/**4**	3	1
1/18	17	1111 1111 1 1 1 1111 1111	0 0	19/**4**	2	16/19
		1222 1222 1 2 1 2221 2221	E E	32/**4**	3	1
		1222 2112 2 2 2 2112 2221	9 F	32/**4**	3	1
–	–	–	–	xx/**4**	–	–
1/33	32	1111...1111	0 0	34/**8**	2	–
–	–	–	–	xx/**8**	–	–
1/63	62	–	–	–	–	–
1/64	63	–	–	–	–	–

7.9.2.4 Vertical scaler

The vertical scaler performs the vertical downscaling of the input data stream to a randomly number of output lines. It can be used for input line lengths up to 768 pixels/line and has to be bypassed, if the input line length exceeds this pixel count.

For the vertical scaling there are two different modes:

- The ACCU mode (vertical accumulation) for scales down to icon size and
- The Linear Phase Interpolation (LPI) mode for scales between 1 and $\frac{1}{2}$.

7.9.2.5 ACCU mode (scaling factor range 1 to 1/1024; YACM = 1)

For vertical scales down to icon size the ACCU mode can be used. In this mode the parameter YSCI controls the scaling and the parameter YACL the vertical anti-aliasing filtering.

The output lines are generated by a scale-dependent variable averaging of (YACL + 2) input lines. In this way a vertical FIR filter is build for anti-aliasing, with up to maximum 65 taps.

YSCI defines the output line qualifier pattern and YACL defines the sequence length for the line averaging. For accurate processing the sequence has to fit into the qualifying pattern. In case of misprogramming YACL, unexpected line dropping occurs.

Where:

- N_{OL} = Number of Output Lines and
- N_{IL} = Number of Input Lines.

the YSCI (scaling increment), YACL (accumulation length; optimum: 1 line overlap) and YP (scaling start phase) have to be set according to the equations below, see Fig.21.

- YACL = TRUNC $[N_{IL}/N_{OL} - 1]$ accumulation sequence length; i.e. number of lines per sequence, that are not part of overlay region of neighbouring sequences (optimum: 1 line overlapped)
- YSCI = INT $[1024 \times (1 - N_{OL}/N_{IL})]$ scaling increment
- YPx = INT [YSCI/16] scaling start phase (fix; modified in LPI mode only).

In order to get a unity amplitude gain for all sequence lengths and to improve the vertical scaling performance, the accumulated lines can be weighted and the amplitude of the scaled output signal has to be renormalized. In the given example (see Fig.21), using the optimal weighting, the gain of a sequence results in 1 + 2 + 2 + 1 = 6. Renormalization (factor $\frac{1}{6}$) can be done

- By gain reduction using BCS control (brightness, contrast, saturation) down to $\frac{4}{6}$ and selecting factor $\frac{1}{4}$ for DCGY2 to DCGY0 which may result in a loss of signal quantization, or
- By gain emphasizing using BCS control up to $\frac{8}{6}$ and selecting factor $\frac{1}{8}$ for DCGY2 to DCGY0 which may result in a loss of signal detail due to limiting in the BCS control.

Normally, the weighting would be 2 + 2 + 2 + 2. In this case the gain can be renormalized simply with DCGY2 to DCGY0 = '010' (factor $\frac{1}{8}$). Table 58 gives examples for register settings depending on a given scale ratio.

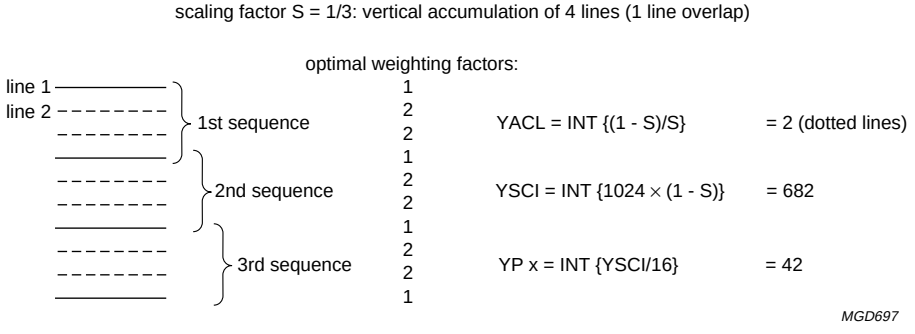


Fig.21 Example: vertical accumulation.

Table 58 Vertical scaling and normalization

VERTICAL SCALE RATIO	YACL	COEFFICIENT SEQUENCE (EXAMPLE)	CYA; CYB	WEIGHT SUM	DCGY	BCS (CONTR. | SAT.) = X/Y × 64
1 to 1/2 (0)	0	1-1	01, 00	2	0	1
1/2 to 1/3 (512)	1	1-1-1	03, 00	3	0	2/3
		1-2-1	01, 02	4	1	1
1/3 to 1/4 (683)	2	1-1-1-1	03, 00	4	1	1
1/4 to 1/5 (768)	3	1-1-1-1-1	07, 00	5	1	4/5
		1-2-2-2-1	01, 06	8	2	1
1/5 to 1/6 (820)	4	111 111	07, 00	6	1	4/6
		121 121	05, 02	8	2	1
		112 211	03, 04	8	2	1
1/6 to 1/7 (854)	5	111 1 111	0F, 00	7	2	8/7
		111 2 111	07, 08	8	2	1
1/7 to 1/8 (878)	6	1111 1111	0F, 00	8	2	1
1/8 to 1/9 (896)	7	1111 1 1111	1F, 00	9	2	8/9
		1222 2 2221	01, 1E	16	3	1
1/9 to 1/10 (911)	8	1111 1 1 1111	1F, 00	10	3	8/10
		2121 2 2 1212	09, 15	16	3	1
		1122 2 2 2211	03, 1C	16	3	1
1/10 to 1/11 (922)	9	1111 1 1 1 1111	3F, 00	11	2	8/11
		1212 1 2 1 2121	15, 2A	16	3	1
		1112 2 2 2 2111	07, 38	16	3	1
1/11 to 1/12 (931)	10	1111 11 11 1111	3F, 00	12	2	8/12
		1211 21 12 1121	2D, 12	16	3	1
		1111 22 22 1111	0F, 30	16	3	1
1/12 to 1/13 (939)	11	1111 11 1 11 1111	7F, 00	13	2	8/13
		1111 21 2 12 1111	2F, 50	16	3	1
		1121 11 2 11 1211	3B, 44	16	3	1
1/13 to 1/14 (946)	12	1111 111 111 1111	7F, 00	14	2	8/14
		1111 211 112 1111	6F, 10	16	3	1
		1111 112 211 1111	3F, 40	16	3	1
1/14 to 1/15 (951)	13	1111 111 1 111 1111	FF, 00	15	2	8/15
		1111 111 2 111 1111	7F, 80	16	3	1
1/15 to 1/16 (956)	14	1111 1111 1111 1111	FF, 00	16	3	1
1/16 to 1/17 (960)	15	1111 1111 1 1111 1111	FF, 00	17	3	16/17
		2122 2222 2 2222 2212	02, FD	32	4	1

VERTICAL SCALE RATIO	YACL	COEFFICIENT SEQUENCE (EXAMPLE)	CYA; CYB	WEIGHT SUM	DCGY	BCS (CONTR. | SAT.) = X/Y × 64
$^1/_{17}$ to $^1/_{18}$ (964)	16	1111 1111 1 1 1111 1111	FF, 00	18	3	$^{16}/_{18}$
		2212 2212 2 2 2122 2122	44, BB	32	4	1
		1222 2222 1 1 2222 2221	01, FE	32	4	1
...	...	...	...	...	...	...
$^1/_{23}$ to $^1/_{24}$ (980)	22	1111 2222 1111 1111 2222 1111	0F, F0	32	4	1
		1121 1212 1121 1211 2121 1211	AD, 52	32	4	1

7.9.2.6 LPI mode (scaling factor range 1 to $^1/_2$; register bit YACM = 0)

To preserve the signal quality for slight vertical downscales (scaling factors 1 to $^1/_2$) Linear Phase Interpolation (LPI) between consecutive lines is implemented to generate geometrically correct vertical output lines. Thus, the new geometric position between lines N and N + 1 can be calculated.

A new output line is calculated by weighting the samples ‘p’ of lines N and N + 1 with the normalized distance to the newly calculated position:

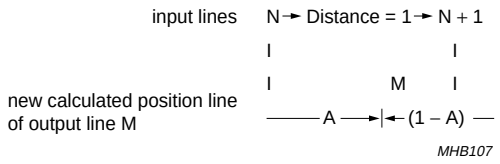


Fig.22 Calculation of output lines.

$p(M) = A \times p(N + 1) + (1 - A) \times p(N)$; where A = 0 to $^{63}/_{64}$.

With N_{OL} = Number of Output Lines and N_{IL} = Number of Input Lines the scaler register bits YSCI (scaling increment) and YP (scaling start phase) have to be set according to the following equations:

- YSCI = INT $[1024 \times (N_{IL}/(N_{OL} - 1))]$ scaling increment
- YPx = INT $[^{YSCI}/_{16}]$ scaling start phase (recommended value).

The vertical start phase offset is defined by $^{YP}/_{64}$ (YP = 0 to 64):

- YP = 0: offset = 0 geometrical position of 1st $line_{out}$ = 1st $line_{in}$
- YP = 64: offset = $^{64}/_{64}$ = 1 geometrical position of 1st $line_{out}$ = 2nd $line_{in}$.

Finally 3 special modes are to be emphasized:

1. **Bypass** (YSCI = 0, YP = 64); each $line_{out}$ is equivalent to corresponding $line_{in}$
2. **Low-pass** (YSCI = 0, YP < 64); e.g. YP = 32: average value of 2 lines ($1 + z^{-h}$ filter)
3. For processing of **interlaced input** signals the LPI mode must be used (ACCU mode would cause ‘line pairing’ problems). The scaling start phase for odd and even field have to be set to:

   $YP_{even} = ^3/_2 \times YP_{odd}$ (line 1 = odd)

In modes 1 and 2 the first input line is fed to the output (without processing), so that the number of output lines equals the number of input lines

7.9.2.7 Flip option (Mirror = 1)

For both vertical scaling modes there is a flip option ‘mirroring’ available for input lines with a maximum of 384 pixels. In the case of full screen pictures (e.g. 768 × 576) that have to be flipped, they first have to be downscaled to 384 pixel/line in the horizontal prescaling unit and after vertical processing (flipping) they may be rezoomed to the original 768 pixels/line in the following VPD.

It should be noted that, when using the flip option, the last input line can not be displayed at the output.

7.9.3 Horizontal phase scaling

In the phase correct Horizontal Phase Scaling (HPS) the pixels are calculated for the geometrically correct, orthogonal output pattern, down to $\frac{1}{4}$ of the prescaled pattern. A horizontal zooming feature is also supported. The maximum zooming factor is at least 2, even more dependent on input pattern and prescaling settings.

The phase scaling consists of a filter and an arithmetic structure which is able to generate a phase correct new pixel value almost without phase or amplitude artefacts. The required sample phase information is generated by a sample phase calculator, with an accuracy of $\frac{1}{64}$ of the pixel distance. The up/downscaling with this circuitry is controlled by the scaler register parameters XSCI and XP. As the fine scaling is restricted to downscales $>(\frac{1}{4}$ of the fine scalers input pixel count), XSCI is also a function of the prescaling parameter XPSC.

With N_{IP} = Number of Input Pixel/line (at DD1 input) and N_{OP} = Number of desired Output Pixels/line, XSCI is defined to:

$$XSCI = INT\ [(N_{IP}/N_{OP}) \times 1024/(XPSC + 1)]$$

The maximum value of XSCI = 4095. Zooming is performed for XSCI values less than 1024. The number of disqualified clock cycles between consecutive pixel qualifiers (at the fine scalers input) defines the maximum possible zoom factor. Consequently, zooming may also be a function of XPSC. It should be noted that if the zooming factor is greater than 2, some artefacts may occur at the end of the zoomed line.

7.9.4 Colour Space Matrix (CSM), dither and γ-correction

The scaled YUV output data is converted after interpolation into RGB data according to CCIR 601 recommendations. The CSM is bypassed in all YUV formats or monochrome modes.

The matrix equations considering the digital quantization are:

$$R = Y + 1.375V$$

$$G = Y - 0.703125V - 0.34375U$$

$$B = Y + 1.734375U.$$

A dither algorithm is implemented for error diffusion. ROM tables are implemented at the matrix output to provide anti-gamma correction of the RGB data. A curve for a gamma of 1.4 is implemented. The tables can be used to compensate gamma correction for linear data representation of RGB output data.

The ‘Chroma Signal Key’ generates an alpha signal used in several RGB formats. Therefore, the processed UV data amplitudes are compared with thresholds. A logic 1 is generated, if the amplitude is within the specified amplitude range. Otherwise a logic 0 is generated. Keying can be switched off by setting the lower limit higher than the upper limit!

7.10 Binary Ratio Scaler (BRS)

7.10.1 General description

The BRS is the second scaler in the SAA7146A. The BRS is supposed to support different encoder applications while the HPS is processing video data. The BRS does not support clipping.

The mainstream application of the BRS is to read data via PCI, e.g. a QCIF-formatted video data to proceed with horizontal and vertical upscaling to CIF-format and place it at the encoder’s disposal (normal playback mode).

To support CCIR encoder and square pixel encoder, an active video window as input for the BRS can be defined. It will prevent black pixels being displayed at the end of the line or at the bottom of the field.

The BRS supports only the YUV 4 : 2 : 2 video data format (see Section 7.11.2). The used DD1 I/O data format is 8-bit. The BRS uses video DMA Channel 3 (FIFO 3) which is only available, if the HPS is not in planar mode or writes back clip information.

Vertical upscaling is supported by means of repeated reading of the same line via PCI. Vertical downscaling is achieved by line dropping.

Horizontal downscaling is performed by an accumulating FIR filter. The downscaling is available for the inbound mode and the upscaling is available for the outbound mode (see Figs 23 and 24).

- Vertical ratios: 4, 2, 1, $\frac{1}{2}$ and $\frac{1}{4}$; select with BRS_V
- Horizontal ratios: 8, 4, 2, 1, $\frac{1}{2}$, $\frac{1}{4}$ and $\frac{1}{8}$; select with BRS_H.

If the data is sent from DD1 to PCI, the processing window for the BRS scaling unit is defined in the acquisition control (see Section 7.8.7).

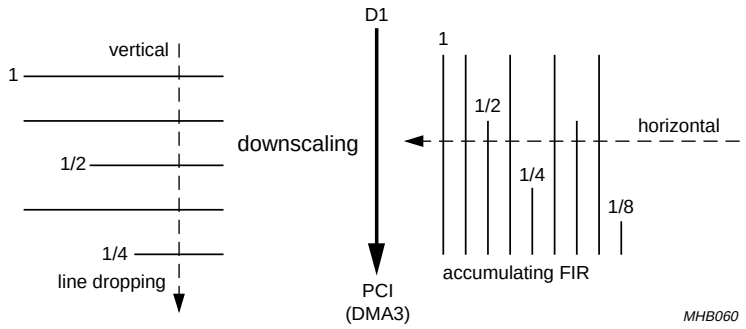


Fig.23 BRS inbound mode.

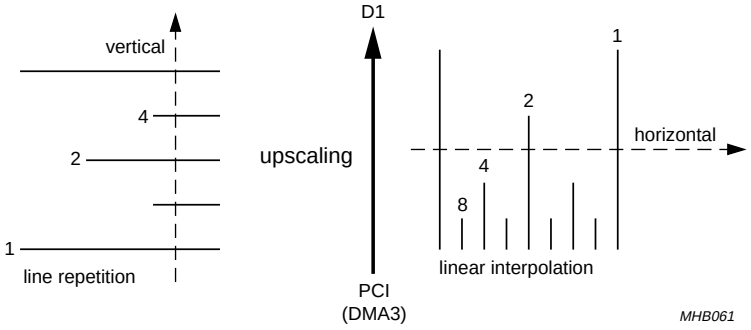


Fig.24 BRS outbound mode.

The PCI source data is defined by the base address (BaseOdd3 and BaseEven3), the distance between the start addresses of two consecutive lines of a field (Pitch3), the number of lines per field of the source frame (NumLines3) and the number of bytes per line of the source frame (NumByte3). The programmer must provide correct scaling settings to fulfil the target window requirements. The pitch has to be Dword aligned.

7.10.2 PLAYBACK MODE

The SAA7146A offers three different modes to support the playback mode for various systems. The Binary Ratio Scaler (BRS) inputs data from FIFO 3, therefore the DMA3 is in master read operation. The scaling result is passed to the DD1 output.

The following sections describe the three different modes: field memory mode, direct mode and line memory mode.

7.10.2.1 Field memory mode

In the field memory mode the SAA7146A takes a vertical sync signal as a timing reference signal. A reset signal for a field memory and a PXQ as write enable are generated within the circuit and both are sent to port A or port B. In this mode the pixel clock depends on the PCI load. The pixels are provided to the DD1 port with maximum $^1/_2$LLC (CCIR 656), the picture rate is restricted by the vertical timing reference. Since the transfer works without losing any data the pixel clock can be varied, therefore an external field memory is needed at the DD1 interface. The SAA7146A writes its data continuously to this memory. The video window size depends on the selected window size in the system memory, the frame buffer (Numlines, Numbytes, pitch and base address) and the selected scaling ratio.

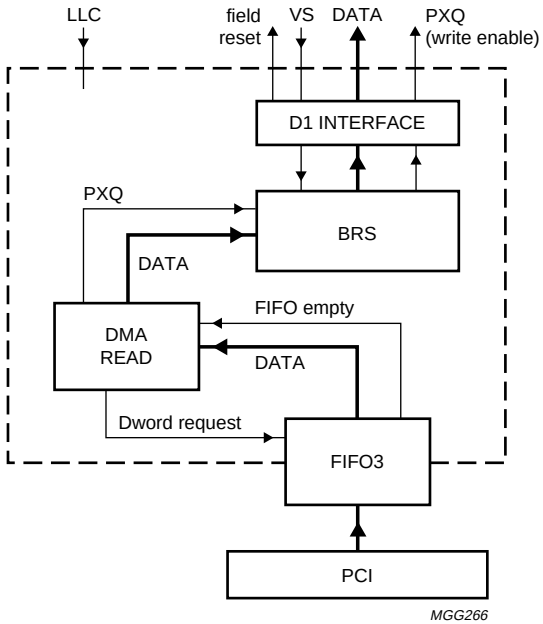


Fig.25 Sync and data path for field memory mode.

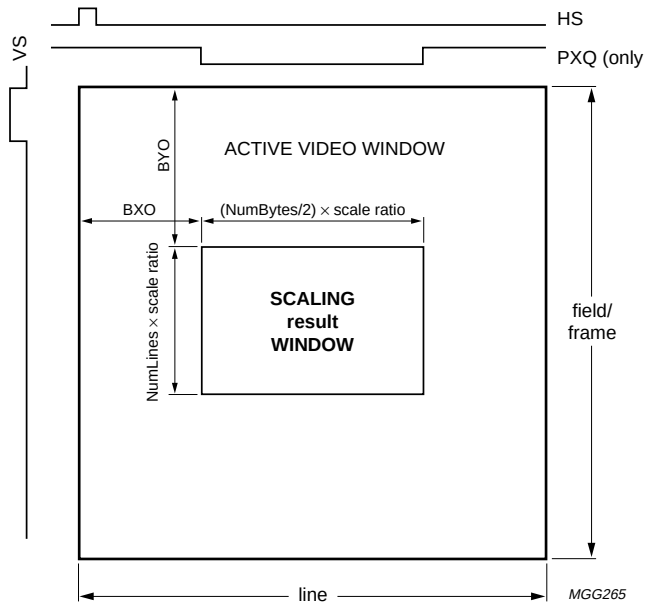


Fig.26 Reference signals for scaling window for direct and line memory mode.

7.10.2.2 Direct mode

The timing reference signals (VS, HS, LLC and FID) are taken from port A or port B. The BRS has to deliver pixel with pixel clock of ½LLC to the D1 port. To ensure that there are no dropouts, a simple underflow handling is performed by the DMA read module. If the PCI load is big and a FIFO underflow occurs, the DMA read module uses a grey value (10H for luminance, 80H for chrominance) or the last pixel as a substitute. The FIFO control counts the failed requests and removes the late values from the FIFO hoping to catch up for lost time to the end of a line. At the end of a line given by the external source the DMA tries to read the data of the new line. This time is defined by the horizontal offset (BXO) of the input acquisition, see Fig.26.

The PXQ can be used as KEY signal for the On Screen Display (OSD) data to support panning, if the video window has no full screen format.

7.10.2.3 Line memory mode

The timing reference signals (VS, HS, LLC and FID) are taken from port A or port B. The access time could be extended by using a line memory at the D1 interface. If a FIFO underflow occurs during the active processing, the DMA read unit waits for the next valid data hoping to catch up for the lost time during the horizontal blanking interval. The timing is retriggered by the H-sync and V-sync. Therefore it is possible, depending on the PCI load, that a line or a part of a line is read multiple from the line memory. The PXQ is used as a write enable signal (see Fig.27).

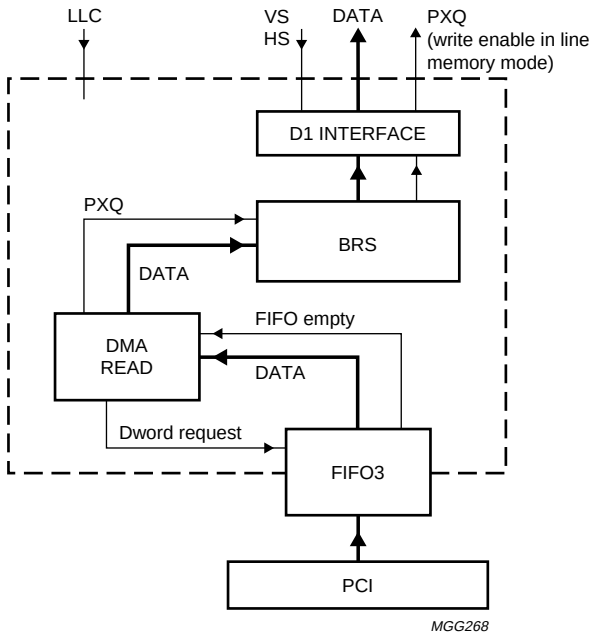


Fig.27 Sync and data path for direct and line memory mode.

7.10.3 VBI DATA INTERFACE

The SAA7146A transports VBI data (data during the Vertical Blanking Interval) or VBI test signals between real time world and the computer system. The data can be in YUV format, luminance Y only, encoded digital CVBS on luminance channel or a single bit stream of sliced data. 1 or 2 MSB of Y is utilized to carry ‘data bit’. PXQ pixel qualifier is used as ‘data clock’.

7.11 Video data formats on the PCI-bus

The big/little-endian is supported in the way that a 2 and a 4-byte swapping is possible. The data formats using 32 bits per pixel requires a 4-byte swap, whereas the data formats using 16 bits per pixel requires a 2-byte swap.

7.11.1 SCALER OUTPUT FORMATS (HPS)

7.11.1.1 RGB

RGB each defined as ‘full range’, all bits = 0 for black and all bits = 1 for white. All RGB formats are composed formats and use video FIFO 1 and video DMA Channel 1.

- **αRGB-32:** the αRGB format use a full byte for each colour component and one byte for the colour key information. The bytes are packed cyclicly into Dwords and uses one Dword per sample. ‘α’ can be the colour key bit in all 8 bits or read via FIFO 2 (see Section 7.11.1.3) and uses one entire Dword per sample (see Table 59).
- **RGB-24 packed:** The 24-bit RGB format use a full byte for each colour component. The bytes are packed cyclicly into Dwords, uses 0.75 Dwords per sample, i.e. 3 Dwords per 4 samples. The byte phase of the first sample on each line is defined by the 2 LSBs of the DMA base (see Table 60).

Table 59 αRGB-32 format

PACKING WITHIN 32-BIT Dword			
BIT 31 TO BIT 24	**BIT 23 TO BIT 16**	**BIT 15 TO BIT 8**	**BIT 7 TO BIT 0**
α/–	R	G	B
α/–	V	Y	U

Table 60 RGB-24 packed format

BUS CYCLE	PACKING WITHIN 32-BIT Dword			
	BIT 31 TO BIT 24	**BIT 23 TO BIT 16**	**BIT 15 TO BIT 8**	**BIT 7 TO BIT 0**
1	B_1	R_0	G_0	B_0
2	G_2	B_2	R_1	G_1
3	R_3	G_3	B_3	R_2

The following formats use two pixels per Dword and derive RGB from RGB-24 by truncation or by error diffusion dither. The byte phase of the first sample each line is defined by LSB + 1 of DMA base. 'α' is the colour key.

- RGB-16 (5 : 6 : 5): Red has 5 bits, Green has 6 bits, Blue has 5 bits
- RGB-15 (α : 5 : 5 : 5): α-bit, Red has 5 bits, Green has 5 bits, Blue has 5 bits
- RGB-15 (5 : 5 : α : 5): Red has 5 bits, Green has 5 bits, α bit, Blue has 5 bits.

Table 61 RGB-16 formats Dword

PACKING WITHIN 32-BIT Dword	
BIT 31 TO BIT 16	**BIT 15 TO BIT 0**
$Pixel_1$	$Pixel_0$

7.11.1.2 YUV

All YUV formats are based on CCIR coding:

Luminance Y in straight binary:

Black: Y = 16 of 256 linear coding

White: Y = 235 of 256 linear coding.

Colour difference signals UV in offset binary:

No colour: U = V = 128 of 256 steps

Full colour: U = V = 128 ±112 steps.

- **YUV 4 : 2 : 2:** U and V sampled co-sided with first Y sample (of 2 samples in-line). Byte phase of the first sample each line is defined by bit 0 and bit 1 of DMA base address (see Table 62).
- **YUV 4 : 1 : 1:** U and V sampled co-sided with first Y sample, (of 4 samples in-line), 8 samples are packed in 3 Dwords (see Table 63).

Table 62 YUV 4 : 2 : 2 format

PACKING WITHIN 32-BIT Dword			
BIT 31 TO BIT 24	**BIT 23 TO BIT 16**	**BIT 15 TO BIT 8**	**BIT 7 TO BIT 0**
Y_1	V_0	Y_0	U_0

Table 63 YUV 4 : 1 : 1 format

BUS CYCLE	PACKING WITHIN 32-BIT Dword			
	BIT 31 TO BIT 24	**BIT 23 TO BIT 16**	**BIT 15 TO BIT 8**	**BIT 7 TO BIT 0**
1	Y_1	V_0	Y_0	U_0
2	Y_3	V_4	Y_2	U_4
3	Y_7	V_6	Y_5	Y_4

The following formats are planar YUV formats and use the three video FIFOs and three video DMA Channels 1, 2 and 3. The byte phase of the first sample each line is defined by the 2 LSBs of every DMA base.

- YUV 4 : 4 : 4; U and V sampled with every Y sample
- YUV 4 : 2 : 2; U and V sampled co-sided with first Y sample (of 2 samples in-line)
- YUV 4 : 2 : 0; MPEG U and V sampled at upper left sample of 4 sample in square (2×2)
- YUV-9 video; U and V sampled at selected sample of 16 samples in-square (4×4)
- YUV1; YUYV, YUYV...
- YUV2; YYUU, YYVV...

7.11.1.3 8-bit formats

- **Y8G**; Only Y or inverted Y
- α**8**; 8-bit alpha information, to be master-read through FIFO 2 and merged into RGB-24 with alpha.

There are two pseudo CLUT formats, which derives its bits from RGB-24 or YUV-24 by truncation, or by error diffusion dither. The byte phase of the first sample each field is defined by 2 LSBs of DMA base.

- RGB-8 (3 : 3 : 2); Red has 3 bits, Green has 3 bits, Blue has 2 bits
- YUV8 (4 : 2 : 2); Y has 4 bits, U has 2 bits, V has 2 bits. Y = 0 doesn't exist, to handle 16-bit colour formats for pseudo CLUT. After dithering, $Y_{min} = 1$.

All 8-bit formats are packed formats, 4 samples go into one Dword. The byte phase of the first sample of each line is defined by the 2 LSBs of the DMA base. All except α8 use FIFO 1.

Table 64 8-bit formats

PACKING WITHIN 32-BIT Dword			
BIT 31 TO BIT 24	**BIT 23 TO BIT 16**	**BIT 15 TO BIT 8**	**BIT 7 TO BIT 0**
$pixel_3$	$pixel_2$	$pixel_1$	$pixel_0$

7.11.2 BINARY RATIO SCALER OUTPUT FORMATS

All YUV formats are based on CCIR coding:

Luminance Y in straight binary:

Black: Y = 16 of 256 linear coding

White: Y = 235 of 256 linear coding.

Colour difference signals UV in offset binary:

No colour: U = V = 128 of 256 steps

Full colour: U = V = 128 ±112 steps.

The following formats use video FIFO 3, DMA Channel 3 and are packed formats.

- YUV 4 : 2 : 2 U and V sampled co-sided with first Y sample (of 2 samples in-line). Byte phase of the first sample each line is defined by bit 0 and bit 1 of DMA base address.

Table 65 YUV 4 : 2 : 2 formats

PACKING WITHIN 32-BIT Dword			
BIT 31 TO BIT 24	**BIT 23 TO BIT 16**	**BIT 15 TO BIT 8**	**BIT 7 TO BIT 0**
Y_1	V_0	Y_0	U_0

7.11.2.1 VBI data formats

- Y8; uses only the Y portion of the data stream and packs four bytes in one Dword
- YUV 4 : 2 : 2; packs two pixel into one Dword, the order is Y_1, V_0, Y_0, U_0
- 1-bit format; the Y1 format is a 1-bit format which packs 32 times the most significant bit of luminance (Y) into one Dword, the first bit is bit 31 of the Dword
- 2-bit format; the Y2 format is a 2-bit format which packs 16 times the two most significant bits of luminance (Y) into one Dword, the first bit is bit 31 of the Dword.

7.12 Scaler register

7.12.1 INITIAL SETTING OF DUAL D1 INTERFACE

The initial settings of the Dual D1 interface contains all control bits of the scaler part which do not change during a cyclic processing of the video path. These control bits must be initialized at the beginning of the processing. The different upload conditions of the video path depend on these control bits. Changing these bits during the active processing can cause a valid UPLOAD.

Table 66 Initial setting of Dual D1 interface

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
50	LLC_A	31	RW	Line Locked Clock control for D1_A: 0: LLC_A set to input 1: LLC_A set to output, taken from LLC_B
	SIO_A	30 to 29	RW	Synchronization port_A configuration: 00: HS_A and VS_A are input (i.e. 3-state) 01: HS_A is output, HGT of HPS; VS_A is output, VGT of HPS 10: HS_A is output, RESET signal for a field memory VS_A is input, vertical sync signal for BRS this setting is needed for the field memory mode 11: HS_A is output, HGT of BRS; VS_A is output, VGT of BRS
	PVO_A	28	RW	Polarity of VS_A, if VS output: 0: direct from HPS or BRS, see SIO_A 1: inverted
	PHO_A	27	RW	Polarity of HS_A, if HS output is select by SIO_A: 0: direct from HPS or BRS, see SIO_A 1: inverted
50	SYNC_A	26 to 24	RW	Sync edge selection and field detection mode internal sync signals SyncA (Ha, Va, Fa) if: HS, VS are input: Ha/Va/Fa derived from pins HS, VS are output: HS/VS as select by SIO_A 000: Ha at rising edge of HS; Va at rising edge of VS; Fa = HS $\times$ VS-rising, directly 001: Ha at rising edge of HS; Va at falling edge of VS; Fa = Hs $\times$ VS-falling, directly 010: Ha at rising edge of HS; Va at rising edge of VS; Fa = HS $\times$ VS-falling, forced toggle 011: Ha at rising edge of HS; Va at falling edge of VS; Fa = HS $\times$ VS-falling, forced toggle 100: Ha at rising edge of HS; Va at rising edge of VS; Fa = free toggle 101: Ha at rising edge of HS; Va at falling edge of VS; Fa = free toggle 110: Ha at rising edge of HS; Va at rising and falling edge of Frame Sync at the VS pin; Fa = direct FS 111: Ha, Va, Fa; derived from SAV and EAV decoded from the data-stream at D1_A port. Not used if the MSB of HPSdatasel in Table 71 is set to logic 1
50	FIDESA	23 and 22	RW	Field identification port_A edge select (ODD is defined by FID = 1, EVEN is defined by FID = 0) 00: no interrupt condition 01: rising edge is interrupt condition 10: falling edge is interrupt condition 11: both edges are interrupt condition
	–	21 to 16	–	reserved

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
50	LLC_B	15	RW	Line Locked Clock control for D1_B: 0: LLC_B set to input 1: LLC_B set to output, taken from LLC_A
50	SIO_B	14 and 13	RW	Synchronization port_B configuration: 00: HS_B and VS_B are input (i.e. 3-state) 01: HS_B is output, HGT of HPS; VS_B is output, VGT of HPS 10: HS_B is output, RESET signal for a field memory; VS_B is input, vertical sync signal for BRS this setting is needed for the field memory mode 11: HS_B is output, HGT of BRS; VS_B is output, VGT of BRS
50	PVO_B	12	RW	Polarity of VS_B, if VS output: 0: direct from HPS or BRS, see SIO_B 1: inverted
50	PHO_B	11	RW	Polarity of HS_B, if HS output is select by SIO_B: 0: direct from HPS or BRS, see SIO_B 1: inverted
50	SYNC_B	10 to 8	RW	Sync edge selection and field detection mode internal sync signals SyncB (Hb, Vb and Fb) if: HS, VS are input: Hb/Vb/Fb derived from pins HS, VS are output: HS/VS as select by SIO_B 000: Hb at rising edge of HS; Vb at rising edge of VS; Fb = HS × VS-rising, directly 001: Hb at rising edge of HS; Vb at falling edge of VS; Fb = Hs × VS-falling, directly 010: Hb at rising edge of HS; Vb at rising edge of VS; Fb = HS × VS-falling, forced toggle 011: Hb at rising edge of HS; Vb at falling edge of VS; Fb = HS × VS-falling, forced toggle 100: Hb at rising edge of HS; Vb at rising edge of VS 101: Hb at rising edge of HS; Vb at falling edge of VS; Fb = free toggle 110: Hb at rising edge of HS; Vb at rising and falling edge of Frame Sync at the VS pin; Fb = direct FS 111: Hb, Vb and Fb derived from SAV and EAV decoded from the data-stream at D1_B port. Not used if the MSB of HPSdatasel in Table 71 is set to logic 1
50	FIDESB	7 and 6	RW	Field identification port_B edge select (ODD is defined by FID = 1, EVEN is defined by FID = 0) 00: no interrupt condition 01: rising edge is interrupt condition 10: falling edge is interrupt condition 11: both edges are interrupt condition
	–	5 to 0	–	reserved

7.12.2 VIDEO DATA STREAM HANDLING AT PORT D1_A

Table 67 Video data stream handling at port D1_A

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
54	VID_A	31 and 30	RW	Video data Port_A and PXQ_A select (PXQ goes always with data): 00: input, i.e. 3-state 01: reserved 10: output data stream is Y8C from BRS 11: output data stream is Y8C from HPS
	Y8C_A	29	RW	Y8C codes, if output Y8C only: 0: no SAV and EAV data in the video data output-stream 1: with SAV and EAV
	–	28 and 27	–	reserved
	PFID_A	26	RW	Polarity change of the field identification signal at Port_A: 0: as detected in the field detection 1: inverted
	–	25 to 16	–	reserved

7.12.3 VIDEO DATA STREAM HANDLING AT PORT D1_B

Table 68 Video data stream handling at port D1_B

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
54	VID_B	15 and 14	RW	Video data Port_B and PXQ_B select (PXQ goes always with data): 00: input, i.e. 3-state 01: reserved 10: output data stream is Y8C from BRS 11: output data stream is Y8C from HPS
	Y8C_B	13	RW	Y8C codes: 0: no SAV and EAV data in the video data output stream 1: with SAV and EAV
	–	12 and 11	–	reserved
	PFID_B	10	RW	Polarity change of the field identification signal at Port_B 0: as detected in the field detection 1: inverted
	–	9 to 0	–	reserved

7.12.4 BRS PROGRAMMING REGISTER

The BRS programming has in principle three modes:

1. **Inbound** and **downscaling**: the binary ratio scaler input multiplexer selects data from the Dual D1 real time video interface, Port A or B and 'normally' writes the result via FIFO 3 and DMA3 to PCI, if DMA3 is enabled in master write mode and not used for other purposes. Syncs including Field ID are taken from Port A or B (FID defines which base address is used in DMA3).
2. **Outbound** and **upscaling** in direct and line memory mode: the binary ratio scaler takes the data from FIFO 3. The DMA3 is in master read operation. The scaling result can be selected by the DD1 port output multiplexers. The timing reference signals (VS, HS, LLC and FID) are taken from Port A or B.
3. **Outbound** and **upscaling** in **field memory** mode: the binary ratio scaler takes the data from FIFO 3. The DMA3 is in master read operation. The scaling result can be selected by the DD1 port output multiplexers. The vertical sync signal is taken from the VS_A or VS_B port as timing reference signal. At the HS_A or HS_B port the SAA7146A generates a reset signal for each field. The PXQ is an output signal which is connected to the write enable port of the memory. If an interlaced source is selected (different base addresses for ODD and EVEN fields), the field detection must be set to 'free toggle' mode, due to the missing horizontal sync signal.

Table 69 BRS control register

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION	
				INBOUND	OUTBOUND
58	BRSdatasel and MODE	31 and 30	RW	**source select for BRS video data**:	
				00: video data stream from A 01: video data stream from B 10: reserved	11: read from DMA_3/FIFO 3
	BRSsyncsel	29	RW	**source select for BRS sync signals**:	
				0: take Ha, Va, Fa, LLC_A as select in the 'Initial setting of Dual D1 Interface'; see Table 66.	in direct and line memory mode the same setting as in the inbound mode is select
				1: take Hb, Vb, Fb, LLC_B as select in the 'Initial Setting of Dual D1 Interface'; see Table 66.	in field memory mode the horizontal sync port must set to output to get the a field RESET signal for a field memory
	BYO	28 to 19	RW	vertical offset, counted in lines, after selected vertical sync edge until data is captured from DD1	BYO defines a vertical offset, counted in lines, after selected vertical sync-edge until data is read from the FIFO. For field memory mode BYO must be 000H. The video window is selected by 'NumLines', 'NumBytes', 'pitch' and 'base address'.
	BRS_V	18 and 17	RW	vertical downscaling: 00: write every line to DMA3 01: write every 2nd line only 10: reserved 11: write every 4th line only	vertical upscaling: 00: regular read 01: read every line twice 10: reserved 11: read every line 4 times

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION	
				INBOUND	OUTBOUND
58	BXO	16 to 7	RW	horizontal offset, counted in qualified LLC cycles, after selected horizontal sync edge, till data is captured from DD1	BXO defines a horizontal offset, counted in LLC cycles, after selected horizontal sync edge till data is read from the FIFO
					in field memory mode the following offsets depending on the horizontal scaling ratio must be selected to guarantee the correct outrun behaviour of the scaler (see Table 70). The video window is select by ‘NumLines’, ‘NumBytes’, ‘pitch’ and ‘base address’
	BRS_H	6 to 4	RW	horizontal downscaling (see Section 7.10.1): 000: every pixel is captured 001: every 2nd pixel is captured 010: reserved 011: every 4th pixel is captured 100: reserved 101: reserved 110: reserved 111: every 8th pixel is captured	horizontal upscaling: 000: provide every sample once 001: provide every sample twice 010: reserved 011: provide every sample 4 times 100: reserved 101: reserved 110: reserved 111: provide every sample 8 times
	Read mode	3 and 2	RW	reserved	00: line memory mode 01: field memory mode 10: direct mode with pixel repetition for not qualified bytes. 11: direct with grey pixel (10H for luminance and 80H for chrominance values) for not qualified bytes.
	PCI format	1 and 0	RW	output format PCI side: 00: YUV 4 : 2 : 2 01: Y8, only luminance 10: Y2, 2 MSBs of Y only 11: Y1, 1 MSB of Y only	input format PCI side: 00: YUV 4 : 2 : 2 01: reserved 10: reserved 11: reserved

Table 70 Horizontal offset values for the field memory mode

RATIO	OFFSET
1	0CH
2	0CH
4	16H
8	2EH

7.12.5 HPS PROGRAMMING REGISTER

Table 71 HPS control register

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
5C	HPSdatasel	31 and 30	RW	**source select for HPS video data**: 00: input video stream for HPS is taken from Port_A 01 input video stream for HPS is taken from Port_B 10: Y-byte from Port_B, C-byte from Port_A (CREF must provide at Port_A) 11: Y-byte from Port_A, C-byte from Port_B (CREF must provide at Port_B)
	Mirror	29	RW	**left-right flip** (mirroring), e.g. for vanity picture: 0: regular processing 1: left-right flip, accessible only if XT (number of pixel after horizontal prescaling) is less than 384 pixels
	HPSsyncsel	28	RW	**source select for HPS sync-signals**: 0: take Ha, Va, Fa, LLC_A as selected in Table 66 1: take Hb, Vb, Fb, LLC_B, as selected in Table 66
	–	27 to 24	RW	reserved
	HYO	23 to 12	RW	vertical offset (start line) of HPS operation, counted in horizontal source/input events, after selected vertical sync edge
	–	11 to 0	RW	reserved

7.12.6 VERTICAL AND HORIZONTAL SCALING

Table 72 HPS, vertical scaling

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
60	YACM	31	RW	**Y (vertical) scaler Accumulation (calculation) Mode of vertical arithmetic**: 0: arithmetic operates as a linear phase interpolation (LPI) 1: arithmetic operates as accumulating FIR filter in vertical direction.
	YSCI	30 to 21	RW	**Y scaler increment for vertical downscaling**: YSCI = INT [1024 $\times$ (N_{IL}/N_{OL} – 1) for YACM = 0 $\rightarrow$ LPI YSCI = INT [1024 $\times$ (1 – N_{OL}/N_{IL})] for YACM = 1 $\rightarrow$ accumulation mode N_{IL} = number of qualified scaler input lines N_{OL} = number of output lines
	YACL	20 to 15	RW	**accumulation sequence Length of the Y (vertical) processing**: Defines vertical accumulation sequence length of input lines If accumulation FIR filter mode is selected YACM, YACL has to fit to the vertical scaling factor (defined by YSCI)
	YPO	14 to 8	RW	**vertical start phase for vertical scaling of the ODD field**: YPO = PHOL $\times$ 128 (PHOL represents a phase offset with values between logic 0 and logic 1, where the logic 1 represents a distance between two consecutive lines of the input pattern)
	YPE	7 to 1	RW	**vertical start phase for vertical scaling of the EVEN field**: YPE = PHOL $\times$ 128 (PHOL represents a phase offset with values between logic 0 and logic 1, where the logic 1 represents a distance between two consecutive lines of the input pattern).
	–	0	RW	reserved

Table 73 HPS, vertical scale and gain

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
64	PFY	31 to 28	RW	**prefilter selection for luminance component Y**: $H(z) = H1(z) \times H2(z) \times H3(z)$ $H1, H3 = 1 + z^{-1}$ $H2 = 1 + A \times z^{-1} + z^{-2}$ see Table 74
	PFUV	27 to 24	RW	**prefilter selection for colour difference signals UV**: $H(z) = H1(z) \times H2(z) \times H3(z)$ $H1 = 1 + z^{-1}$ $H2 = 1 + A \times z^{-1} + z^{-2}$ $H3 = 1 + z^{-2}$ see Table 75
	–	23 to 19	–	reserved
	DCGY	18 to 16	RW	**DC gain control of Y scaler**: Dependent on active coefficients and the sequence length, the amplitude gain has to be renormalized. Gain factor = 2 (DCGY + 1); see Table 76. The resulting factor is a function of CYi and DCGY. The resulting weight factor = 0 for CYAi = CYBi = 0 or CYAi = CYBi = 1 or DCGY >5 otherwise weight = weighting factor/gain factor; see Table 77.
	CYA	15 to 8	RW	Coefficient select for Y (vertical) processing in accumulation mode. For improvement of vertical filtering the accumulated lines can be weighted. Weighting factor = $2(2 \times CYBi + CYAi - 1)$; see Table 78.
	CYB	7 to 0	RW	

Table 74 Prefilter selection for luminance component Y

PFY1	PFY0	PFY1	PFY0	H1	H2	H3	A
X	X	0	0	bypass	bypass	bypass	X
X	X	0	1	active	bypass	bypass	X
X	X	1	0	active	bypass	active	X
0	0	1	1	active	active	active	2
0	1	1	1	active	bypass	bypass	$^{15}/_{16}$
1	0	1	1	active	bypass	active	$^{7}/_{8}$
1	1	1	1	active	active	active	$^{3}/_{4}$

Table 75 Prefilter selection for colour difference signals UV

PFY1	PFY0	PFY1	PFY0	H1	H2	H3	A
X	X	0	0	bypass	bypass	bypass	X
X	X	0	1	active	bypass	bypass	X
X	X	1	0	active	bypass	active	X
0	0	1	1	active	active	active	2
0	1	1	1	active	bypass	bypass	$^{15}/_{16}$
1	0	1	1	active	bypass	active	$^{7}/_{8}$
1	1	1	1	active	active	active	$^{3}/_{4}$

Table 76 DC gain control of Y scaler

DCGY2	DCGY1	DCGY0	DCGY	GAIN FACTOR
0	0	0	0	2
0	0	1	1	4
0	1	0	2	8
0	1	1	3	16
1	0	0	4	32
1	0	1	5	64
1	1	0	6	128
1	1	1	7	256

Table 77 Weight factor as a function of CYi and DCGY

CYi	DCGY							
	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	$^{1}/_{2}$	$^{1}/_{4}$	$^{1}/_{8}$	$^{1}/_{16}$	$^{1}/_{32}$	$^{1}/_{64}$	0	0
2	1	$^{1}/_{2}$	$^{1}/_{4}$	$^{1}/_{8}$	$^{1}/_{16}$	$^{1}/_{32}$	0	0
3	0	1	$^{1}/_{2}$	$^{1}/_{4}$	$^{1}/_{8}$	$^{1}/_{16}$	0	0

Table 78 Coefficient select for Y (vertical) processing in accumulation mode

CYBi	CYAi	CYi	WEIGHTING FACTOR
0	0	0	0
0	1	1	1
1	0	2	2
1	1	3	4

Table 79 HPS, horizontal prescaler

OFFSET (HEX)	NAME	BIT	TYP.	DESCRIPTION
68	–	31 and 30	–	reserved
	DCGX	29 to 27	RW	**DC gain control of X prescaler** (see Table 57): depending on the number of active coefficients '2' in the accumulation sequence and the sequence length, the output amplitude gain has to be set to as given in Table 80.
	–	26 to 24	–	reserved
	XPSC	23 to 18	RW	**prescaling factor of the X PreSCaler**: defines accumulation sequence length and subsampling factor of the input data stream: XPSC = TRUNC ($N_{IP}/N_{OP} - 1$) N_{OP} = number of prescaler output pixel N_{IP} = number of qualified scaler input pixel.
	XACM	17	RW	**X (horizontal) prescaler Accumulation Mode of accumulating FIR**: 0: accumulating operates overlapping 1: non overlapping accumulation (must be set to bypass the prescaler).
	–	16	–	reserved
	CXY	15 to 8	RW	**Coefficient select for X prescaler (luminance component Y)**: for DC gain compensation of prescaler the accumulated pixels can be weighted by '1' or '2'. CXYi defines a sequence of 8 bits, which control the coefficients: CXYi = 0: pixel weighted by '1' CXYi = 1: pixel weighted by '2'
	CXUV	7 to 0	RW	**Coefficient select for X prescaler (colour difference signals UV)**: for DC gain compensation of prescaler the accumulated pixels can be weighted by '1' or '2'. CXUVi defines a sequence of 8 bits, which control the coefficients: CXUVi = 0: pixel weighted by '1' CXUVi = 1: pixel weighted by '2'

Table 80 Selection of output gain

DCGX2	DCGX1	DCGX0	GAIN
0	0	0	1
0	0	1	$\frac{1}{2}$
0	1	0	$\frac{1}{4}$
0	1	1	$\frac{1}{8}$
1	0	0	$\frac{1}{2}$
1	0	1	$\frac{1}{4}$
1	1	0	$\frac{1}{8}$
1	1	1	$\frac{1}{16}$

Table 81 HPS, horizontal fine-scale

OFFSET (HEX)	NAME	BIT	TYP.	DESCRIPTION
6C	XIM	31	RW	**horizontal interpolation mode**: 0: normal mode, sample phase is calculated for every qualified sample 1: fixed phase, sample phase is fixed to the value set by XP
	XP	30 to 24	RW	**start phase for horizontal fine scaling XP = $PHO_P \times 128$** (PHOP represents a phase offset with values between '0' and '1', where the '1' represents a distance between two consecutive pixels of the input pattern)
	XSCI	23 to 12	RW	**X Scaler Increment for fine (phase correct) scaling in horizontal pixel phase arithmetic**: XSCI = INT [(N_{IP}/N_{OP}) $\times$ 1024/(SPSC + 1)] N_{OP} = number of output pixels N_{IP} = number of qualified scaler input pixels.
	HXO	11 to 0	RW	horizontal offset (horizontal start) of input source for HPS, counted in qualified pixels with PXQ, after selected horizontal sync edge.

7.12.7 BCS

Table 82 BCS control

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
70	BRIG	31 to 24	RW	luminance brightness control; see Table 83
	CONT	23 to 16	RW	luminance contrast control; see Table 84
	–	15 to 8	–	reserved
	SATN	7 to 0	RW	chrominance saturation control; see Table 85

Table 83 Luminance brightness control

D7	D6	D5	D4	D3	D2	D1	D0	GAIN
1	1	1	1	1	1	1	1	255 (bright)
1	0	0	0	0	0	0	0	128 (CCIR level)
0	0	0	0	0	0	0	0	0 (dark)

Table 84 Luminance contrast control

D7	D6	D5	D4	D3	D2	D1	D0	GAIN
0	1	1	1	1	1	1	1	1.999 (max. contrast)
0	1	0	0	0	0	0	0	1 (CCIR level)
0	0	0	0	0	0	0	0	0 (luminance off)

Table 85 Chrominance saturation control

D7	D6	D5	D4	D3	D2	D1	D0	GAIN
0	1	1	1	1	1	1	1	1.999 (max. saturation)
0	1	0	0	0	0	0	0	1 (CCIR level)
0	0	0	0	0	0	0	0	0 (colour off)

7.12.8 CHROMA KEY

Table 86 Chroma key range

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
74	VL	31 to 24	RW	set lower limit V for chroma keying (8-bit; twos complement): 1000 0000: as maximum negative value = –128 signal level 0000 0000: limit = 0 0111 1111: as maximum positive value = +127 signal level
	VU	23 to 16	RW	set upper limit V for chroma keying (8-bit; twos complement): 1000 0000: as maximum negative value = –128 signal level 0000 0000: limit = 0 0111 1111: as maximum positive value = +127 signal level
	UL	15 to 8	RW	set lower limit U for chroma keying (8-bit; twos complement): 1000 0000: as maximum negative value = –128 signal level 0000 0000: limit = 0 0111 1111: as maximum positive value = +127 signal level
	UU	7 to 0	RW	set upper limit U for chroma-keying (8-bit; twos complement): 1000 0000: as maximum negative value = –128 signal level 0000 0000: limit = 0 0111 1111: as maximum positive value = +127 signal level

Table 87 HPS output and formats

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION	
78	matrix	31 and 30	RW	YUV to RGB conversion, gamma compensation	
				00: no YUV to RGB conversion	10: YUV to RGB conversion linear
				01: reserved	11: YUV to RGB conversion with compensation of gamma-pre-correction
	–	29 and 28	–	reserved	
	outformat	27 to 24	RW	output format, depends on matrix programming; see Table 88	
		23 and 22	RW	chroma line select for INDEO-9	
		21 and 20	RW	chroma pixel select for INDEO-9	
	SHIFT	17	RW	0: normal mode, all bytes stay MSB aligned 1: shift mode, ‘shift right’ of all bytes, fill MSB position with logic 0, this mode has meaning only for output formats defined in bytes, undefined result in non-byte modes	
	DITHER	16	RW	dither: applies only to formats with reduced bit resolution, (#) that derived from higher bit resolution formats: 1: dither is applied by ‘linear’ one-dimensional error diffusion 0: dither algorithm is not applied, just truncation	

Table 88 Output formats

CODE (HEX)	OUTPUT FORMAT	
0	YUV 4 : 2 : 2, (16 = 8 – 8), composed	RGB16 (5 : 6 : 5), composed, #
1	YUV 4 : 4 : 4, composed, ‘packed’	RGB24, composed, ‘packed’
2	reserved	αRGB32 (8 : 8 : 8 : 8), composed
3	YUV 4 : 1 : 1, composed	αRGB15 (1 : 5 : 5 : 5), composed, #
4	YUV2	RGαB15 (5 : 5 : 1 : 5), composed, #
5	reserved	reserved
6	Y8, monochrome	reserved
7	YUV8 (4 : 2 : 2), pseudo CLUT, #	RGB8 (3 : 3 : 2), pseudo CLUT, #
8	YUV 4 : 4 : 4, de-composed	reserved
9	YUV 4 : 2 : 2, de-composed	reserved
A	YUV 4 : 2 : 0 (2^2:1:1) MPEG, de-composed.	reserved
B	YUV9 (4^2:1:1) INDEO-9, de-composed.	reserved
C	reserved	reserved
D	Y1	reserved
E	Y2	reserved
F	YUV1	reserved

Table 89 Clip control

OFFSET	NAME	BIT	TYPE	DESCRIPTION
78H	–	15 to 10	–	reserved
	ClipCK	9 and 8	RW	**clipping by chroma key CK** (or MSB of α-8): OR-ed with other clip list or clip-bit mask, if ClipMode is enabled 00: chroma key CK is not used for clipping, CK → α 01: chroma key CK is not used for clipping, inverted CK → α 10: clipping, based on chroma key CK bit 11: clipping, based on chroma key CK bit inverted
	–	7	–	reserved
	ClipMode	6 to 4	RW	**clipping based on DMA2 read information**: OR-ed with chroma key CK, if ClipCK is enabled 000: no clipping based on DMA2 read, DMA2 can be used to write decomposed format U 001: no clipping based on DMA2 read; DMA2 reads 8-bit-α to substitute CK in α-formats 010: reserved 011: reserved 100: clipping, based on pixel clip list, rectangular overlays 101: clipping, based on pixel clip list, rectangular overlays, inverted 110: clipping, based on pixel clip bit mask 1-bit/pixel 111: clipping, based on pixel clip bit mask 1-bit/pixel, inverted
	RecInterl	3	RW	**select interlaced mode for rectangular overlays**: 0: normal mode 1: interlaced mode, this bit must be set if only one clip list for both fields is available. This function assumes that the ODD field is always above the EVEN field.
	–	2	–	reserved
	ClipOut	1 and 0	RW	**use of DMA3** to report (write) key of clip information back: 00: no clip output, DMA3 can be used to write decomposed format V, or to serve BRS, read or write 01: DMA3 writes chroma key information CK; 1-bit/pixel 10: DMA3 writes back (clip mask-CK); 1-bit /pixel 11: DMA3 writes applied pixel clipping back; 1-bit/pixel

7.13 Scaler event description

The RPS is controlled by the PAUSE command on special events. This section describes the video events. Because of these video events a defined time for an upload is given. Table 90 shows the UPLOAD handling for the scaler registers. For special applications it can also be useful to select other combinations. For this the termination of the UPLOAD must guarantee that the UPLOAD is completed before the processing restarts, e.g. with a new line or a new field. To avoid conflicts, e.g. change of vertical settings during vertical processing, the MASKWRITE command can be used to change single bits within a Dword. Each video event can force only one upload at a time. This means that the video event is cleared by the circuit as described below, if the corresponding upload has occurred.

Table 90 UPLOAD handling for the scaler registers

REGISTER	OFFSET (HEX)	VIDEO EVENT	DESCRIPTION
Initial setting of Dual D1 Interface	50	no video event	The 'initial settings of the Dual D1 interface' contains all control bits of the scaler part which do not change during a cyclic processing of the video path. These control bits must be initialized at the start of the processing. The different upload conditions of the video path depend on these control bits. Changing these bits during the cyclic processing can cause internal pulse signals which generate video events. These events may not fit into the sequence for the cyclic processing.
Video DATA stream handling at port D1_A	54	VBI_A	Vertical Blanking Indicator at VS_A port: the VBI is a V-pulse which depends on the selected edge of the vertical blanking interval. The edge is defined by the SYNC_A bits. The selected mode depends on the accepted sync signals. This register can be uploaded with this V-pulse.
Video DATA stream handling at port D1_B	54	VBI_B	Vertical Blanking Indicator at VS_B port: the VBI is a V-pulse which depends on the selected edge of the vertical blanking interval. The edge is defined by the SIO_B bits. The selected mode depends on the accepted sync signals. This register can be uploaded with this V-pulse.
BRS control register	58	BRS_DONE	Inactive BRS data path: in write mode the BRS data path is inactive from the falling edge of VGT at the output of the BRS which means that target line and target byte are reached to the start of the next field (V-pulse which triggered the BRS acquisition). For the read mode this register contains only initial settings which can not change during cyclic processing.
HPS control	5C	HPS_DONE	Inactive HPS data path between two video windows: the HPS data path is inactive from the falling edge of the VGT at the output of the HPS, indicating that target line and target byte are reached, to the start of the next window processing. V-pulse at the HPS acquisition input.
HPS vertical scale	60		
HPS vertical scale and gain	64		
Chroma key range	74		
HPS output and formats	78		
Clip control	78		
HPS, horizontal prescale	68	HPS_LINE_DONE	Inactive HPS data path between two lines: The HPS data path is inactive from the falling edge of the HGT at the output of the HPS, indicating that target byte are reached to the start of the next line processing. Rising edge of the HGT at the HPS acquisition output.
HPS, horizontal fine-scale	6C		
BCS control	70		

7.14 Clipping

The SAA7146A supports clipping in the HPS data path. Clipping can be achieved with the chroma key information or with clip data information coming via master read through FIFO 2. Both sources will be OR-ed and can be switched on/off or inverted individually. These settings are controlled by the registers ClipCK and ClipMode.

The information read via FIFO 2 can be used for clipping with rectangular overlays or for bit mask clipping. The overlay clipping supports up to 16 rectangular overlays using 64 Dwords. The bit mask clipping allows an arbitrary number of window clips of any size or shape. This mode needs one bit for every pixel.

Chroma or clip information can be written to system memory via FIFO 3. This is controlled by the ClipOut register. It is possible to combine the clip information with the inversion of the applied (foreground) chroma key. The result is a mask leaving the (background) area free. This mask can be read back in the next field to clip a different video stream to be placed into the same window as background (blue boxing).

It should be noted that planar output formats overrule the use of FIFO 2 and FIFO 3 for clipping. Only chroma clipping is available and no clip information can be written.

7.14.1 Bit mask clipping

The bit mask clipping will use one Dword as clip data for 32 pixels. The first bit of clip data is the MSB.

7.14.2 Rectangular overlay clipping

The rectangular overlay clipping is responsible for occluding rectangular overlay windows lying over a video window.

The rectangular clipping algorithm needs two lists; one for pixels and one for lines. Every list element in both lists contains a coordinate and display information for every overlay window. The 64 Dword FIFO 2 allows up to 16 overlay windows, each having two pixel list entries and two line list entries.

The rectangular overlay clipping can be used in interlaced or non-interlaced mode. This is controlled by the ‘RecInterl’ register bit. The overlay window coordinates are defined for the target window, independent of whether the video will be written interlaced or non-interlaced into the target window.

Every overlay window is defined by its top/left and bottom + 1/right + 1 coordinates. The coordinates are relative to the top left (0, 0) reference of the video window.

The overlay clipping combines the coordinates with display information which is a ‘overlay/no_overlay’ (1, 0) bit for each overlay window. A simple example, shown in Fig.28, illustrates the relationship between coordinates and display information.

In this example one overlay window ‘a’ (5, 1; 8, 3) is defined. Relevant coordinates for the algorithm are the coordinates where display information changes. At the top/left coordinates (5, 1) the display information will be set to 1 (‘overlay’). Therefore, first list entries are (5, 1) for the pixel list and (1, 1) for the line list. The overlay will end at the bottom/right coordinates plus one, e.g. at (9, 4) (8 + 1, 3 + 1). This will lead to the list entries (9, 0) for the pixel list and (4, 0) for the line list.

The central element of the rectangular overlay clipping combines the display information of lines and pixels held in the registers PIXEL_INFO and LINE_INFO. This unit will provide the ‘no_display’ information when both line and pixel display information are set to ‘no_display’. In the example shown in Fig.28, this will happen for the pixels 5 to 8 and for the lines 1 to 3.

If there is more than one overlay window, the window display information of all windows will be combined into one display information. If any of the display information of any window is indicated ‘no_display’ the actual pixel will not be displayed. This ensures that overlapping overlay windows will be handled by the hardware, since the video information will only be displayed when no window is lying over it. Since the overlapping information is only implicitly in the lists, the overlapping information need not be taken into account during the creation of the lists.

The main part of the algorithm is responsible for loading the display information registers PIXEL_INFO and LINE_INFO. Both will be initialized to ‘display’. LINE_INFO will be updated at the beginning of every line, when the line counter is equal to the LINE_NR in the line list. PIXEL_INFO will be updated when the pixel counter is equal to the PIXEL_NR in the pixel list. If there is no new information both registers will hold their old values.

Both line and pixel list have to be sorted from top to bottom or left to right coordinates and are not allowed to have two consecutive list elements with the same coordinate. In the example shown in Fig.29, the list entry with line coordinate 1 will hold the ‘display’ information of window ‘a’ and the ‘no_display’ information of window ‘c’, so two list elements with the same coordinate are merged into one. The last elements in the lists are characterized by the coordinate 0.

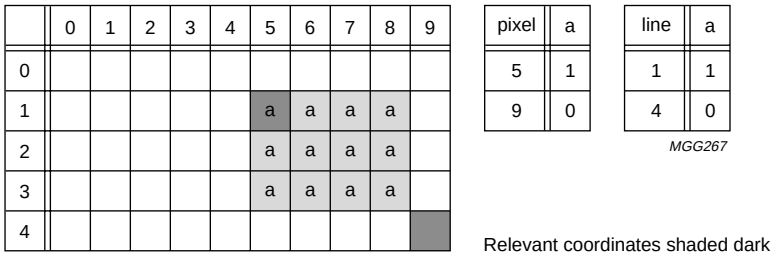


Fig.28 Example 1 - Rectangular overlay clipping.

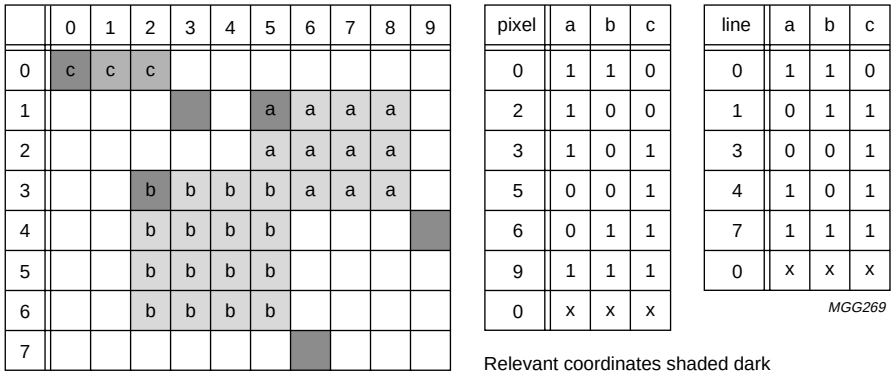


Fig.29 Example 2 - Rectangular overlay clipping.

7.14.2.1 Memory organization for rectangle overlay windows.

Every overlay window is defined by two corners with four coordinates. One Dword holds one 11-bit coordinate and 16-bit with the display information for up to 16 overlay windows.

Table 91 Dword organization for rectangular overlay windows

UNUSED	COORDINATE	DISPLAY INFO
bit 31 to bit 27 (5-bit)	bit 26 to bit 16 (11-bit)	bit 15 to bit 0 (16-bit)

The two lists, pixel list and line list, are interlocked in the 64 Dword memory. The pixel list is located at the even addresses, the line list at the odd addresses. This organization reduces the number of Dwords to be loaded, if there are less then 16 overlay windows. For example, 5 overlay windows need 20 Dwords for the coordinates and 2 Dwords as EOF marker.

7.14.2.2 Driver algorithm

Overlay window coordinates are relative to the video window and can range between 0, 0 and 2047. Relevant coordinates are top/left, bottom + 1/right + 1 of the overlay windows. If an overlay window has its bottom/right coordinates at the bottom/right of the video window its relevant coordinates bottom + 1/right + 1 would exceed the coordinate range and therefore do not have to be inserted into the lists.

- Build lists: build sorted lists of lines and pixels containing top/left, bottom + 1/right + 1 coordinates of every overlay window, without having consecutive list entries with the same coordinate. Every list will have an end of list entry with all coordinate bits set to zero. This EOL entry will follow the last entry. If there are 16 overlay windows and no double coordinates the lists are full and there is no last entry.
- Insert display information: for every relevant coordinate in both lists and for every overlay window, if the coordinate in the line/pixel list is in between the top/bottom or left/right coordinates of the overlay window then set the display information bit to 1 (‘display’). Otherwise, set the display information bit to 0 (‘no_display’).

7.15 Data Expansion Bus Interface (DEBI)

7.15.1 General description

The DEBI performs 16-bit parallel I/O in immediate (direct) transfer mode and block transfer mode. The immediate mode is used to transfer a byte, word or Dword to or from the target device. The block transfer mode offers the possibility to read or write up to 32 kbytes data blocks.

7.15.2 Features

- 8-bit and 16-bit slaves supported
- External interrupt supported, DMA suspend/resume function
- Byte, word and Dword transfers supported
- Slaves with or without handshake ability supported due to programmable cycle time
- Different endian types supported
- PCI DMA master transfer in block mode
- Optional address increment in block mode.

7.15.3 DEBI pins

There are 21 DEBI pins. Most of the control signals represent different functions with respect to the selected interface mode (Intel/ISA or Motorola/68 kbytes).

Table 92 DEBI pin list

PIN NAME	TYPE	DESCRIPTION
AD15 to AD0	input/output	multiplexed address and data lines
AS_ALE	output	address strobe/address latch enable
UDS_WRN	output	upper data strobe/write not
LDS_RDN	output	lower data strobe/read not
RWN_SBHE	output	read/write not/system byte HIGH enable
DTACK_RDY	input	data acknowledge/ready (should be pulled HIGH if not used)
GPIO3	input/output (used here as input only)	optional external interrupt input

7.15.4 FUNCTIONAL DESCRIPTION

An immediate access cycle consists of one address phase and one data phase. A block transfer with address increment enabled consists of several consecutive address/data phase couples. A block transfer with disabled address increment consists of one address phase followed by several data phases. The AS_ALE signal toggles only for a new address phase.The single bytes or words are assembled/disassembled to/from Dwords. This includes byte lane swapping since 8-bit devices use for data transfer AD7 to AD0 only, (AD15 to AD8 are used in address phase too, since all 16 AD lines are used for addressing.)

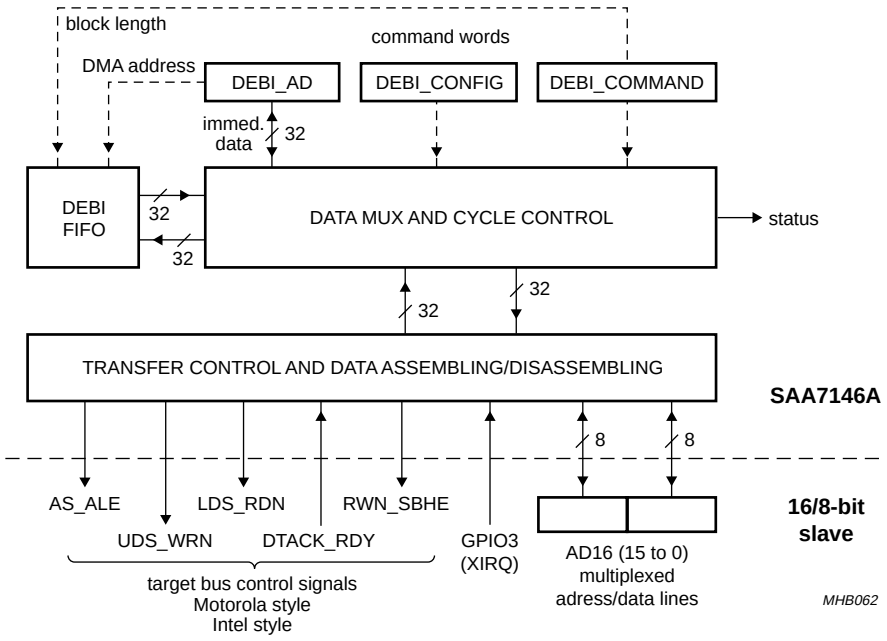


Fig.30 DEBI interface.

7.15.4.1 Target bus cycle in Intel mode

The SAA7146A starts a target transfer cycle by placing the target address on the multiplexed address/data lines (AD15 to AD0). The Address Latch Enable (ALE) is then asserted (set LOW) indicating that the address lines AD15 to AD0 and the SBHE signal are valid (the active LOW SBHE indicates data transfer on the high byte lane AD15 to AD8). After asserting ALE the AD lines are multiplexed for data transfer. Valid data on the AD lines is indicated by assertion of WRN in the write mode (data from SAA7146A to target), or by assertion of RDN in the read mode (data from target to SAA7146A). In the read mode, it is the responsibility of the target to place data on the AD lines as soon as possible following the assertion of RDN. If the target does not require wait states or handshake for data transfer, RDY should then be tied HIGH and the TIMEOUT value should be set to 0. If the target requires wait states, but still does not utilize handshake, then the TIMEOUT value can be increased. The width of both WRN and RDN pulses will be increased by 1 PCI cycle for each count in the TIMEOUT value. If the target is capable of handshake, to indicate when it is ready for data transfer, then the RDY signal can be used.

Since the SAA7146A will not evaluate the RDY signal until TIMEOUT + 1 PCI cycles have elapsed, it is recommended that TIMEOUT be set to a minimum value (usually 0) for maximum throughput. If the target is slow in responding to the RDN/WRN then TIMEOUT can be increase to allow the target time to de-assert RDY (pull to LOW level) for the current data cycle. Once the TIMEOUT + 1 number of PCI cycles have elapsed (from the assertion of RDN/WRN) the transfer control is in a ‘wait for RDY high’ state. The data transfer cycle will be ended when a TIMEOUT condition at RDY HIGH or a rising edge of RDY after TIMEOUT is detected. The cycle is ended by de-asserting ALE, SBHE and RDN/WRN.

It should be noted that in the INTEL mode the timer must be enabled (TIEN = 0). The TIMEOUT counter is used as ‘delay sampling RDY’ value to accommodate target reaction delay in generating a valid RDY signal. TIMEOUT is NOT used as an overall cycle watchdog timer (i.e.: to terminate the cycle if RDY fails to become de-asserted). The current cycle will not end and a new cycle will not start until RDY is asserted (HIGH).

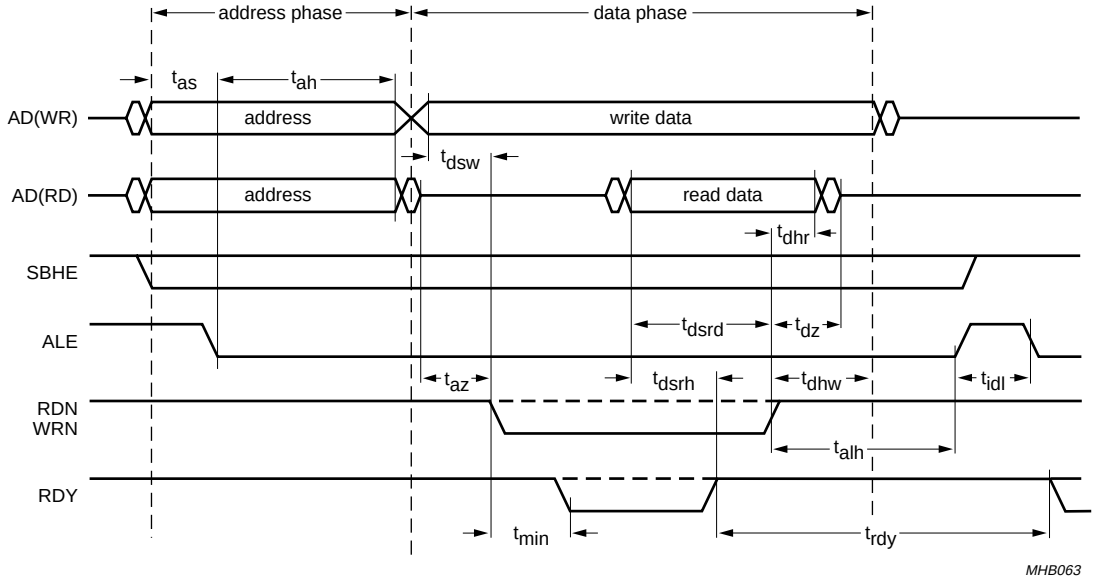


Fig.31 Intel style transfer.

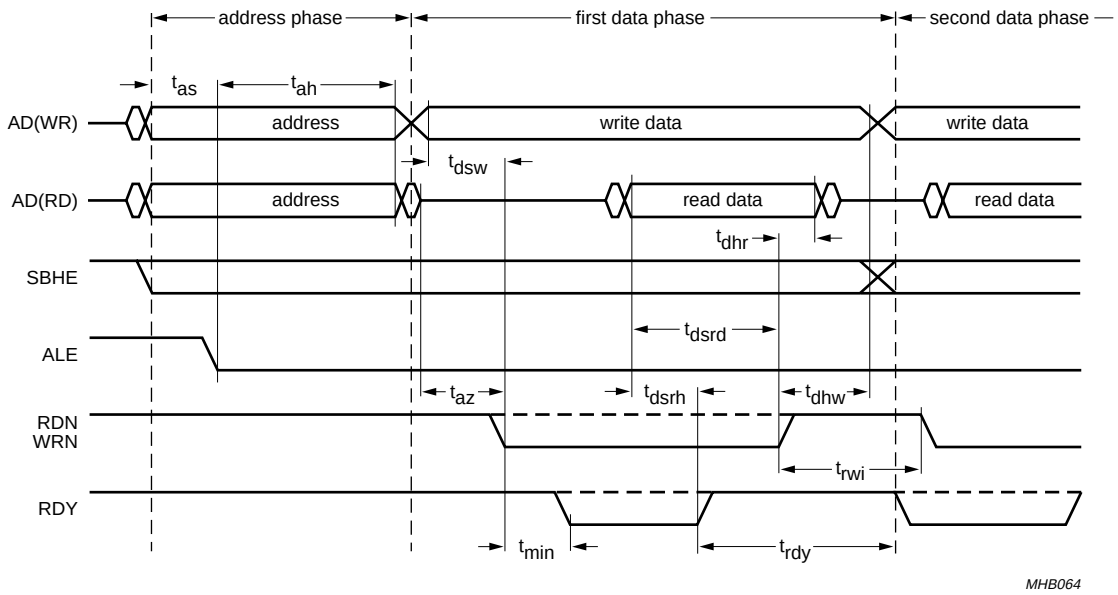


Fig.32 Intel style block transfer without address increment.

7.15.4.2 Target bus cycle in Motorola mode

The target transfer cycle starts with applying the target address onto the multiplexed address/data lines. By setting the Address Strobe (AS) to LOW it is indicated that the direction signal RWN (Read/Write Not) and the address are valid. The AS signal is usable as a address latch enable signal. After asserting AS LOW the address/data lines will change to the data transfer state. The indication of valid data in write mode or the request for data in read mode is done by transition of Upper Data Strobe (UDS) and/or Lower Data Strobe (LDS) to LOW.

Since the selection of the upper and lower bytes for transfer is done via LDS/UDS there is no need for decoding address line A0. Only AD15 to AD1 are needed for transmitting the (word-)address. Slaves with handshake ability have to drive DTACK LOW when they have placed valid data onto AD16 in read mode or when they have read their data in write mode. The cycle is ended when a TIMEOUT condition at inactive DTACK or a positive DTACK edge is detected. Then AS, LDS, UDS and RWN are reset to HIGH. A new cycle will not start before detection of resetting DTACK to HIGH.

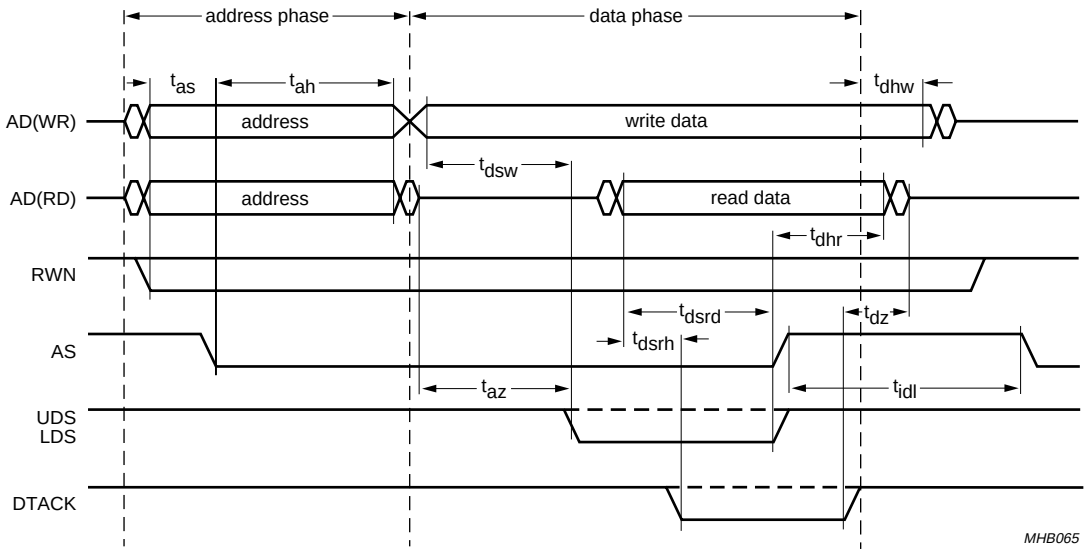


Fig.33 Motorola style transfer.

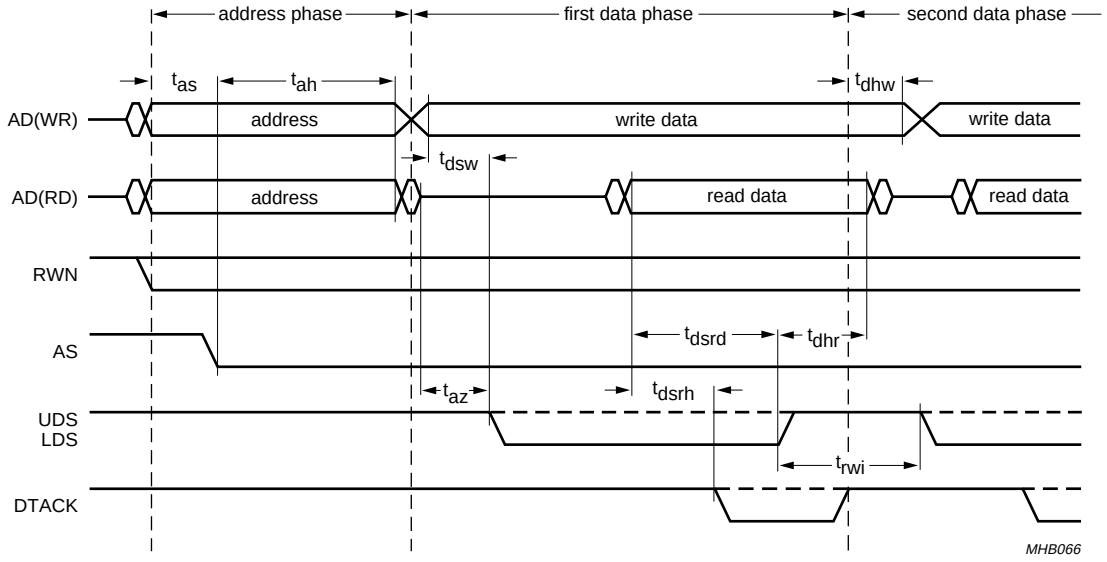


Fig.34 Motorola style block transfer without address increment.

Table 93 Timing parameters (t_{PCI}: PCI clock cycle time, minimum 30 ns)

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
t_{as}	address set-up time		$t_{PCI} - 15$	–	ns
t_{ah}	address hold time		$t_{PCI} - 10$	–	ns
t_{alh}	delay between de-asserting of RDN/WRN and ALE		$t_{PCI} - 15$	–	ns
t_{az}	address 3-state time before start of read command		$t_{PCI} - 10$	–	ns
t_{dhw}	write data output hold time		–3	–	ns
t_{dhr}	read data input hold time		0	–	ns
t_{dsw}	write data output set-up time		$t_{PCI} - 15$	–	ns
t_{dsrh}	read data input set-up time (relative to handshake edge)		$-0.7t_{PCI}$	–	ns
t_{dsrd}	read data input set-up time (dumb target, no handshake)		20	–	ns
t_{idl}	idle time before new transfer starts with AS_ALE (increment mode)	write access	$2t_{PCI} - 10$	–	ns
		read access	$3t_{PCI} - 10$	–	ns
t_{rwi}	idle time between two data access strobes in non-increment mode; this parameter depends on FAST mode enable	fast mode	$t_{PCI} - 10$	–	ns
		normal mode	$2t_{PCI} - 10$	–	ns
t_{rdy}	RDY assertion time (Intel mode)		$1.5t_{PCI}$	–	ns
t_{min}	delay from negative edge of RDN/WRN to de-assertion of RDY; TIMEOUT should be adjusted to TIMEOUT = t_{min} + 1	note 1	–	$(TIMEOUT - 0.5)t_{PCI}$	ns
t_{dz}	time until slave driven data lines have to go to 3-state, after read cycle is finished		–	$2.0\ t_{PCI}$	ns

Note

1. Only relevant to stretch access cycles in Intel mode. t_{min} can be negative, i.e. RDY can be set to LOW before falling edge of RDN/WRN.

7.15.4.3 Transfer configuration

When using 'dumb' targets (unable to handshake) or 'slow' targets (unable to pull DTACK_RDY immediately), the cycle length is adjusted by using a programmable cycle timer. At TIMEOUT in Motorola mode the transfer control gets into a defined state by finishing the cycle when a slave is hanging or not able to handshake. In Intel mode the transfer control waits for RDY = 1 after TIMEOUT, i.e. the timer reflects the RDY reaction time of the target. In any TIMEOUT case the Timer overflow Interrupt (TI) flag is set. The timer starts at the falling edge of UDS_WRN/LDS_RDN. For initiating a transfer the target address must be specified (16-bit, pointing to the first byte to transfer), the transfer direction (WRITE_n) and the BLOCKLENGTH that indicates how many bytes have to be transferred. For block transfer a 32-bit DMA start address (PCI) has to be specified in the DEBI_AD register. When the BLOCKLENGTH is 1 to 4 bytes the data is immediately transferred to/from the DEBI_AD register. Immediate transfer crossing a Dword boundary is not allowed. Such illegal transfer trials are reported by the Format Error bit (FE) in the status register. Immediate transfer starts with the least significant byte/word of the DEBI_AD register.

The following figures illustrate the protocol of the DEBI bus for Intel mode transfers. These figures contain no formal timing specification (see Table 93 for timing) but rather are intended to help in understanding the operation of the DEBI interface. The DEBI bus protocol operates in step with the PCI clock, so it is shown for reference at the bottom of these diagrams. At slower PCI clock rates, the DEBI transaction time is proportionally increased. It is not necessary to connect a PCI clock to the DEBI target system, since DEBI does not expect target read data or target driven handshake signals to be synchronous to PCI clock.

Figure 35 shows the non-incremental mode access in the fastest possible configuration (TIMEOUT = 0; FAST = 1). The overhead for this type of access is 2 PCI clock cycles for address phase plus 2 PCI clock cycles for each data phase. In this mode, the blocks are easily identified by the falling edge of ALE indicating a new target address can be latched.

Larger TIMEOUT values would lead to wider read/write pulses (pulse width = TIMEOUT + 1 [PCI clock cycles]). Disabling the FAST mode would force 2 PCI clock cycles Idle time between read/write strobes. It is not possible to adjust the address phase timing. It is also assumed in this figure that the RDY signal is not used (tied to HIGH level). Use of the RDY signal is allowed in this mode and further explained in the next example.

Figure 36 shows the incremental mode access. This mode will produce an address phase prior to each data phase, and as such has much lower bandwidth than the non-incremental mode. In the example shown, the RDY signal is also used, although that is not a requirement of this mode. The overhead for this type of access is 2 PCI cycles for address phase plus 2 PCI cycles for data transfer phase plus 3 PCI cycles for write (4 PCI cycles for read) Idle time between the data phase and the next address phase. In the example shown, since RDY was used with a TIMEOUT of 2, the resulting data phase was 4 PCI cycles, rather than the minimum of 2. In this example the RDY de-asserts within the same PCI clock cycle as RDN/WRN, which means RDY LOW is strobed by the DEBI interface 1 PCI clock cycle after setting RDN/WRN to LOW (parameter t_{min} = 1 PCI clock cycle). Due to this a TIMEOUT = t_{min} + 1 = 2 (or greater) is required for flexible access stretching, i.e. synchronizing the RDY and stretching the access until RDY is released to HIGH (see description of t_{min} in the timing parameters; Table 93). In difference to the example without RDY usage, increasing the value in TIMEOUT will NOT result in wider read/write strobes, as long as the TIMEOUT value does not exceed the RDY LOW phase by more than 1 PCI clock cycle. Enabling or disabling the FAST mode has no effect in incremental mode.

It should be noted that the minimum timing illustrated by these diagrams is not the sustainable data rate by the SAA7146A through the DEBI interface. PCI-bus latencies, FIFO fullness, target behaviour and other factors will affect the sustained data rate. For illustration purposes Table 94 provides indication of peak data rates in various DEBI configurations.

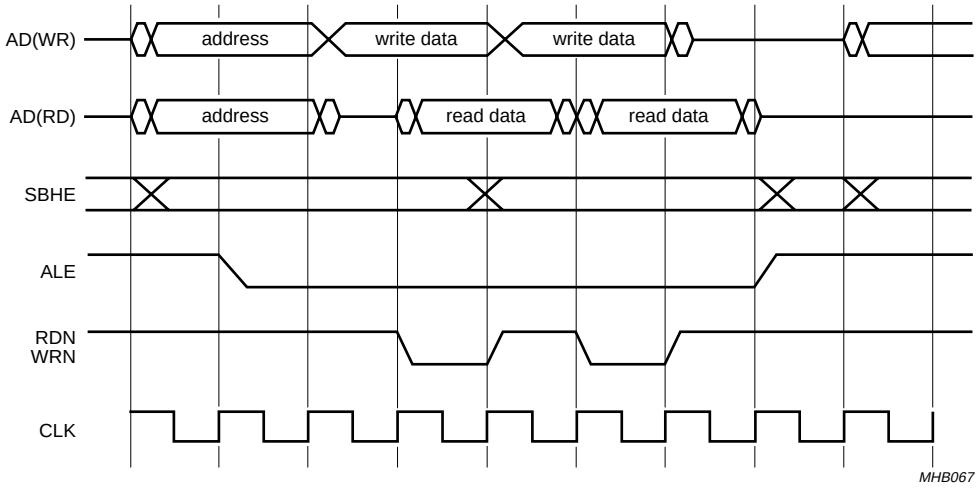


Fig.35 PCI clock related protocol scheme for non-increment Intel mode, no access stretching via RDY.

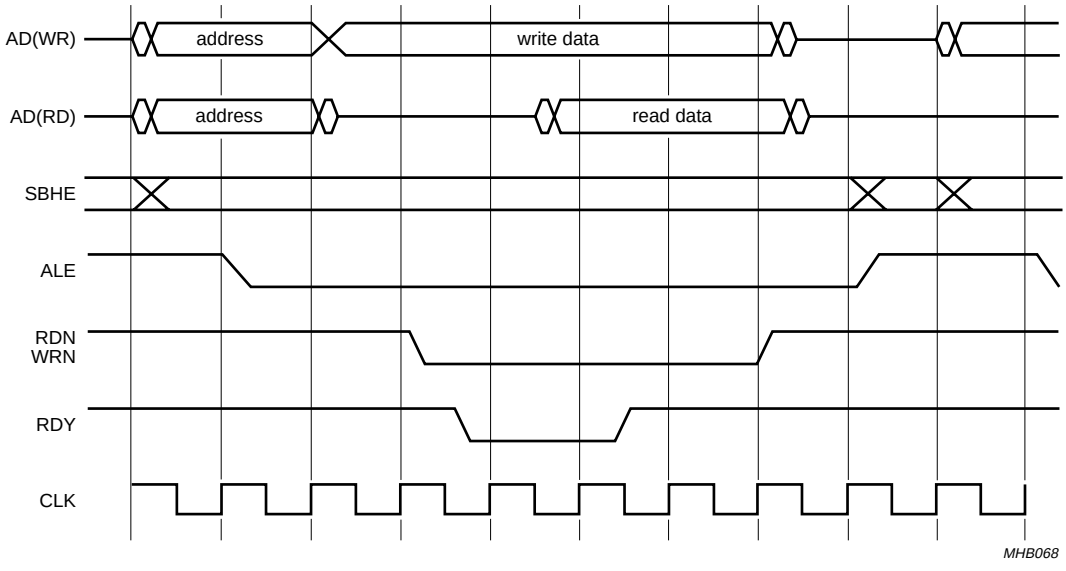


Fig.36 PCI clock related protocol scheme for Intel incremental mode, access stretching via RDY.

Table 94 Overview of peak data rates for non-increment (burst) block transfer configurations at 33 MHz PCI clock

PROTOCOL MODE	TIMEOUT VALUE	WORD WIDTH	FAST MODE	TRANSFER DIRECTION	TARGET SIDE PEAK DATA RATE	OVERALL PEAK DATA RATE(1)
Intel/Motorola(2)	0	16 bit	enabled	R/W	33 Mbytes/s	23.0 Mbytes/s
Intel/Motorola(2)	0	8 bit	enabled	R/W	16.5 Mbytes/s	13.5 Mbytes/s
Intel/Motorola	1	16 bit	enabled	R/W	22 Mbytes/s	17.0 Mbytes/s
Intel/Motorola	3	16 bit	enabled	R/W	13.2 Mbytes/s	11.2 Mbytes/s
Intel/Motorola(2)	0	16 bit	disabled	R/W	22 Mbytes/s	17.0 Mbytes/s
Intel/Motorola	1	16 bit	disabled	R/W	16.5 Mbytes/s	13.5 Mbytes/s

Note

1. These peak data rates could be reached for transfers with large BLOCKLENGTH settings, in a well performing PCI-bus system with low bus load and an appropriate target system without cycle stretching or interrupts.
2. No cycle stretching by RDY/DTACK possible.

It is possible to halt an actual block transfer by external interrupt. This is achieved by setting the XIRQ_EN bit in the DEBI_CONFIG register and asserting the GPIO3 pin input to LOW while an block transfer is active. If the XRESUME bit is set to 0, this will end the current block transfer within the next two Dwords. When XRESUME = 1 the transfer will go to a wait state, but the transfer operation will not end (DEBI_ACTIVE still asserted). When GPIO3 is de-asserted to HIGH the block transfer will resume. The contents of DEBI_AD and DEBI_COMMAND registers are steadily updated on actual address and block length values during block transfer. Due to this it is possible to abort the transfer, read back actual status, do other transfers and resume later with the saved information. It should be noted that after a Dword aligned read block transfer (i.e. if BLOCKLENGTH [1:0] + A16 [1:0] = 4 or A16 [1:0] = 0) the read back value of the DEBI_AD register points to the consecutive address of the just filled PCI memory range. After a Dword unaligned transfer the read back DEBI_AD value points 1 Dword further (it should be noted that this also effects the value of the remaining BLOCKLENGTH after interrupt; A16 target address read back is not effected by this). RPS is able to react on the GPIO3 pin events.The 16 AD lines are set to 3-state while DEBI is in XIRQ wait state (XRESUME enabled). To support target devices of different endian type the swap register has to be configured.

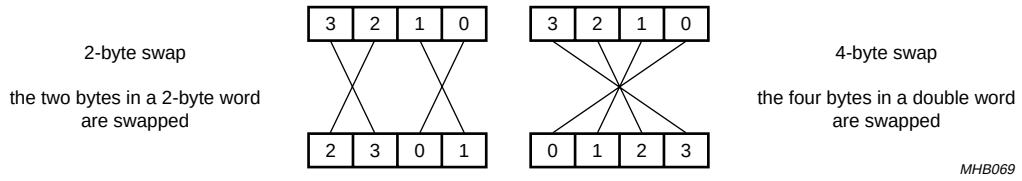


Fig.37 Endian swapping.

7.15.4.4 Command word description

To configure and initiate a transfer there are 3 PCI memory mapped command words. A DEBI register upload after writing to DEBI_COMMAND starts the transfer process.

Table 95 DEBI_CONFIG

OFFSET	NAME	BIT	TYPE	DESCRIPTION
7CH	XIRQ_EN	31	RW	enable external interrupt on GPIO3
	XRESUME	30	RW	resume block transfer when XIRQ was de-asserted
	–	29	–	reserved
	FAST	28	RW	enable fast mode (short t_{rwi} time)
	–	27 and 26	–	reserved
	TIMEOUT [3:0]	25 to 22	RW	timer set-up value (PCI clock cycles)
	SWAP	21 and 20	RW	endian swap type: 00: straight - don't swap 01: 2-byte swap 10: 4-byte swap 11: reserved
	SLAVE16	19	RW	indicates that slave is able to serve 16-bit cycles
	INCREMENT	18	RW	enables address increment for block transfer
	INTEL	17	RW	Intel style bus handshake if HIGH, else Motorola style
	TIEN	16	RW	timer enable (active LOW)
	–	15 to 0	–	reserved

Table 96 DEBI_COMMAND

OFFSET	NAME	BIT	TYPE	DESCRIPTION
80H	BLOCKLENGTH [14:0]	31 to 17	RW	BLOCKLENGTH > 4: block transfer length in bytes 4 ≥ BLOCKLENGTH > 0: immediate transfer 1 to 4 bytes BLOCKLENGTH = 0: reserved
	WRITE_N	16	RW	transfer direction (write if LOW)
	A16_IN	15 to 0	RW	slave target start address

Table 97 DEBI_PAGE

OFFSET	NAME	BIT	TYPE	DESCRIPTION
84H	DEBI_PAGE	31 to 12	RW	DEBI page table address (not used if PAGE_EN = 0)
	PAGE_EN	11	RW	enable address paging
	–	10 to 0	–	reserved

Table 98 DEBI_AD

OFFSET	NAME	BIT	TYPE	DESCRIPTION
88H	DEBI_AD	31 to 0	RW	data input/output in immediate mode or DMA start address for block transfer (Dword aligned, DEBI_AD [1:0] have to be set to logic 0)

7.16 Audio interface

7.16.1 General description

The SAA7146A has two independent audio interface circuits (A1 and A2) for serial input and output of digital audio data streams. The audio interface circuits are based on the I²S-bus standard but can be configured to several data and timing formats (with respect to framing, bit clock and synchronisation). LSB first (Sony) formats are not supported. Up to 5 audio devices with separate serial data lines and dedicated word select lines can be connected directly. The interface also supports devices that share one serial data line for multiple devices to transmit data in different time slots. A time slot consists of one (serial) byte. Each interface circuit supports up to 5 serial data lines and related framing signals. A1 and A2 have the same internal structure. They share the audio interface pins, i.e. the pins can be accessed and utilized by one or the other audio interface circuits at a time.

In order to support systems with asynchronous or mixed audio sampling rates (e.g. 48 and 44.1 kHz raster, or 48 kHz 2 × 16-bit stereo and 8 kHz 8-bit mono), the two audio interface circuits can run independently and even asynchronously regarding bit clock rate, sampling rate and framing (word select) signals. The two circuits can also be combined into one synchronous interface sharing bit clock and framing and sampling frequencies. Each audio interface has two FIFOs (one for input and one for output), and two associated DMA control circuits (one for master read and one for master write), to exchange data with any PCI address, e.g. main memory. The data structure and signal flow control is time slot oriented and also supports local feedback from input to output and from one timeslot to another time slot. A set of time slots can be looped into one ‘audio super frame’ containing up to 256 bits (32 time slots). The signal flow is defined per time slot and programmed by a time slot list.

7.16.2 Basics of I²S-bus specification

The I²S-bus transports digital audio (sound) signals serially between ICs and consists of three signals:

- A continuous bit clock BCLK (or SCK) with (n × 8) multiple of the audio sample frequency
- A serial data wire SD, transporting the data in serial bursts with MSB first
- A framing signal WS (Word Select) defining (synchronizing) the start of a serial data burst.

WS and BCLK signals are provided by the master device. The SAA7146A audio interfaces can be configured as master or slave.

A data receiver must latch the data on SD line with the rising edge of the bit clock BCLK. To satisfy the requirements regarding set-up and hold times more easily (and secure) it is recommended that the transmitter sends its data on the falling edge. Set-up and hold times are specified ‘parameterized’; i.e. as a percentage of the actual bit clock. The serial data starts one clock cycle after an edge of WS or synchronous to the edge. The word (burst) length of transmitter and receiver may be different. Missing LSBs are filled with zeros, excess LSBs are truncated. There are other formats for transmitting serial digital audio data between ICs which are slightly different but still very similar to the I²S. The data set-up and hold times or even the definition of an active clock edge may vary. The word select or framing signal can be ‘in sync’ with the MSB of the data burst instead of one clock cycle ahead. The Sony format is quite different, as it starts with LSB first.

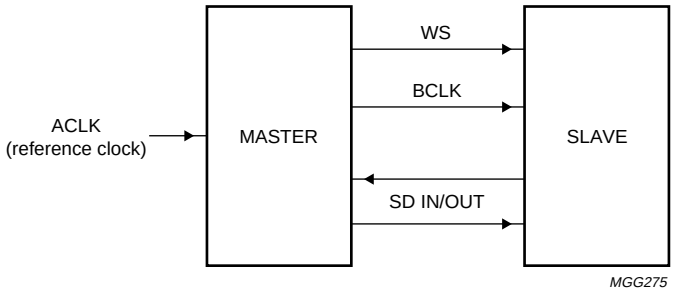


Fig.38 General application schematic of the audio interface.

7.16.3 AUDIO INTERFACE PINS

There are 14 audio interface pins. SD1 is output only for A1 and input only for A2, SD4 is output only for A2 and input only for A1. The other pins can be used by either one of the two interface circuits but only one at a time.

Table 99 Audio pin list

PIN	I/O	FUNCTION
ACLK	I	audio reference clock; multiple of serial bit clock, multiple of audio sampling; maximum frequency 25 MHz; has to be provided even in slave mode
BCLK1	I/O	bit clock 1
BCLK2	I/O	bit clock 2
SD0	I/O	serial data output for audio interface A1 or input for A2
SD1	I/O	serial data I/O for audio interface A1 or A2
SD2	I/O	serial data I/O for audio interface A1 or A2
SD3	I/O	serial data I/O for audio interface A1 or A2
SD4	I/O	serial data input for audio interface A1 or output for A2
WS0	I/O	word select line 0, as input it is used as super frame sync (trigger)
WS1	O	word select output line 1
WS2	O	word select output line 2
WS3	O	word select output line 3
WS4	I/O	word select line 4, as input it is used as super frame sync (trigger)

7.16.4 AUDIO INTERFACE CIRCUIT

Each of the two audio interface circuits of the SAA7146A consists of the following major functional parts:

- Output Dword (format) buffer (to load from FIFO)
- Output data selector (byte Mux 8 to 1)
- Parallel-to-serial converter
- Output pin selector (destination Mux)
- Serial data input selector (bit Mux 4 to 1)
- Serial-to-parallel converter (1 byte)
- Input Dword (format) buffer (to store into FIFO)
- Audio input FIFO and master write DMA
- Feedback (hand) buffer
- Master read DMA and audio output FIFO
- Bit clock selector or generator
- Time slot counter and word select signal generator.

7.16.4.1 Audio clock selection

The clock divider circuit offers 16 different clock stages. To transform a reference clock of 24.576 MHz to a bit clock for an 8 kHz and 8-bit sampling (just 8-bit serial), a clock division of 384 has to be selected. To transform a reference clock of 24.576 MHz to a bit clock for a 48 kHz sampling and 64-bit framing, a division of 8 has to be selected.

The bit clock is divided by 8, which defines a time slot corresponding to the time span of one byte in serial protocol.
The time slot counter gets a count pulse every time slot. It can be running free or can be triggered (reset) via an external word select signal (super frame sync).

Audio interface circuit A1 can be triggered by WS0, audio interface circuit A2 can be triggered by WS4. A time slot list processor generates word select output signals and the internal signals to control the signal flow per time slot. A time slot list contains up to 16 records, each 32 bits wide, supporting super frames with up to 32 time slots.

WS0 (or WS4) triggers the time slot generator and time slot counter directly 'in sync' or are one clock cycle ahead. The WS signals can be generated 'in sync' with the time slot (i.e. MSB of serial data) or 1-bit clock cycle ahead. Each of the two audio interface circuits A1 and A2 has its own independent timing generator. Extra control bits define which of the two timing generators drive which of the word select pins WS0 to WS4.

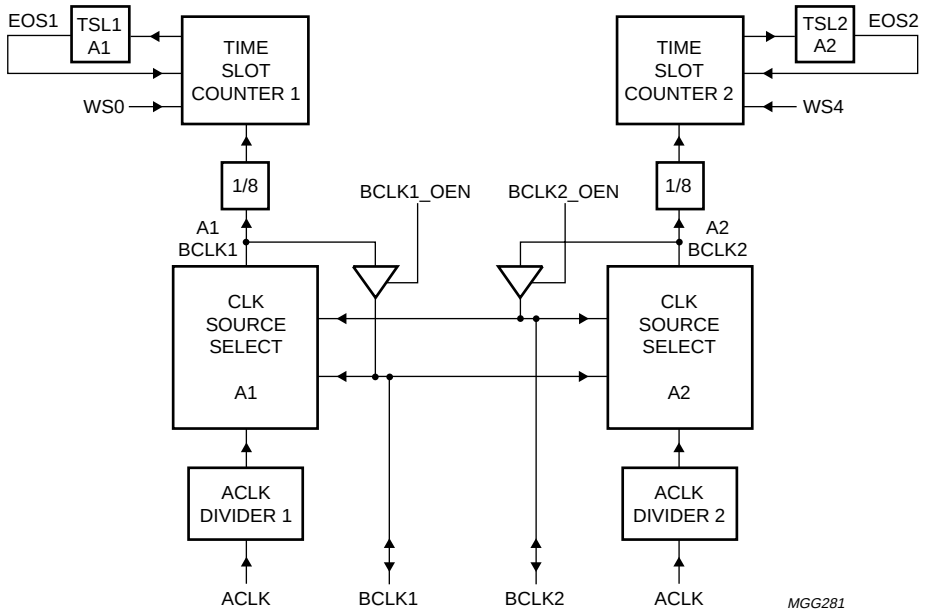


Fig.39 Audio clock control.

7.16.4.2 Audio data path

Figure 40 illustrates the audio data path. An input multiplexer selects serial data from one of four SD pins. A1 can select SD0 and the common serial data pins SD1, SD2 and SD3. A2 can select SD4 and the common serial data pins SD1, SD2 and SD3. A serial-to-parallel converter collects 8 bits to form a byte in a timeslot. At the end of the time slot this byte can be stored into a Dword buffer, and/or into the feedback buffer or can be thrown away. The first byte that is latched, is placed into the first byte place of the buffer, the second byte that is latched, is placed into the second byte place, etc.

Big-endian and little-endian stuffing is supported. If bytes are not latched into a certain buffer, the place pointer of the corresponding buffer is not incremented. The write (fill) pointer of the feedback buffer is reset to it's initial position with every start/restart of the super frame. Up to four bytes of the input data stream can be placed in the intermediate feedback buffer. They can be selected from the buffer to provide data to the output.

The feedback buffer is also read and write accessible via the PCI-bus. This allows reading of status information and writing of control information at specific positions in the audio frame. The write access is only possible when the interface is inactive. The samples of the various real world audio signal streams are byte or word interleaved in system memory and PCI address space. It is the responsibility of system/board designer and of software/programmer to produce a reasonable sample ordering, e.g. have a 16-bit sample on a word boundary and not crossing a Dword boundary.

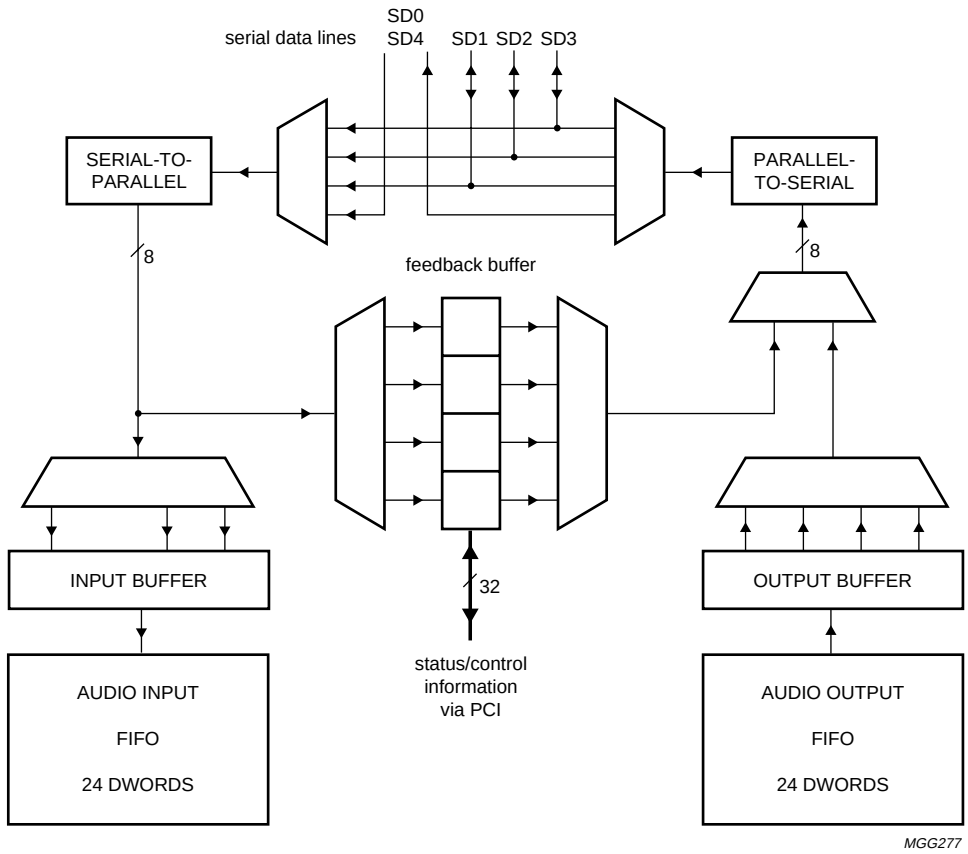


Fig.40 Audio data flow control in A1 and A2.

Table 100 Feedback buffers

OFFSET	NAME	BIT	TYPE	DESCRIPTION
144H	FB_BUFFER1	31 to 0	RW	feeds back audio data or stores status/control informations
148H	FB_BUFFER2	31 to 0	RW	feeds back audio data or stores status/control informations

Under control of the time slot list, a collected Dword is then stored into the input FIFO. The FIFO size is determined to 24 Dwords.

An audio sampling frequency of f_s = 48 kHz and n = 16 time slots in a super frame results in a maximum data load for the PCI from an audio capture DMA channel of 768 kbytes/s (the bit clock rate is 6144 kbit/s). That accounts for approximately 13 Dwords per regular video line time. To generate audio output signals, a master read DMA control fills the output FIFO. A Dword buffer is loaded from FIFO under control of the time slot list. The parallel-to-serial converter takes a byte as programmed in the time slot list from one of the 8 buffer places; 4 in the Dword buffer and 4 in the feedback buffer. The serial output is directed to one of the accessible SD pins. Positive and negative clock edge data transmission is supported by optional BCLK inversion.

Each record in the time slot list describes, how the bytes appearing on the port, are mapped to the Dword wide DMA channels, respectively to the feedback or input buffers. A time slot list record consists of 4 bytes. As up to 32 time slots are supported, the time slot list is comprised of 16 Dwords of programming for each audio interface circuit A1 or A2 which can be linked together.

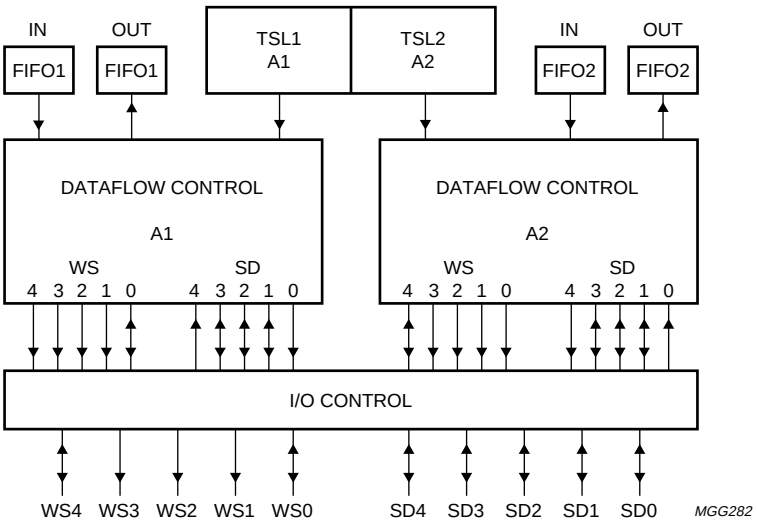


Fig.41 Audio data path components.

Each interface, A1 and A2, uses its own Time Slot List (TSL) when working independently of each other. The shaded areas are valid for combined processing of TSL1 and TSL2 only. In these modes, TSL1 or TSL2 are used interleaved or concatenated, to achieve one single TSL with up to 32 records. Both parts of the list control both interface circuits in parallel. All four DMA channels are available. The TSLs are write only.

Table 101 Time slot list structure

RECORD STRUCTURE		TSL1 16 DWORDS (OFFSET: 180 TO 1BCH)																TSL2 16 DWORDS (OFFSET: 1C0 TO 1FCH)															
BIT#	**NAME**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
31	WS0																																
30	WS1																																
29	WS2																																
28	WS3																																
27	WS4																																
26	reserved																																
25	DIS_A1 [1]																																
24	DIS_A1 [0]																																
23	SDW_A1																																
22	SIB_A1																																
21	SF_A1																																
20	LF_A1																																
19	BSEL_A1 [2]																																
18	BSEL_A1 [1]																																
17	BSEL_A1 [0]																																
16	DOD_A1 [1]																																
15	DOD_A1 [0]																																
14	LOW_A1																																
13	reserved																																
12	DIS_A2 [1]																																
11	DIS_A2 [0]																																
10	SDW_A2																																
9	SIB_A2																																
8	SF_A2																																
7	LF_A2																																
6	BSEL_A2 [2]																																
5	BSEL_A2 [1]																																
4	BSEL_A2 [0]																																
3	DOD_A2 [1]																																
2	DOD_A2 [0]																																
1	LOW_A2																																
0	EOS																																

Table 102 Time slot list bit functions

NAME	FUNCTION
WS0	defining pattern of word select signal output at WS0 pin; if WS0 pin is input and trigger, WS0 bit is meaningless
WS1	defining pattern of word select signal output at WS1 pin
WS2	defining pattern of word select signal output at WS2 pin
WS3	defining pattern of word select signal output at WS3 pin
WS4	defining pattern of word select signal output at WS4 pin: if WS4 pin is input and trigger, WS4 bit is meaningless
DIS_Ax [1:0]	select serial data input from: 00 : SD0 (for A2); SD4 (for A1) 01 : SD1 10 : SD2 11 : SD3
SDW_Ax	0: do not load this byte into the Dword buffer 1: load this byte into the Dword buffer, place into the next available position
SIB_Ax	0: do not load this byte into the intermediate feedback buffer 1: load this byte into the intermediate feedback buffer, place into the next available position
SF_Ax	0: do nothing 1: store Dword buffer into input FIFO, at the next available position
LF_Ax	0: do nothing 1: load next Dword from output FIFO into output Dword buffer
BSEL_Ax [2:0]	select byte for parallel-to-serial converter from output Dword buffer or from intermediate feedback buffer: 000: take byte 0 from output Dword buffer 001: take byte 1 from output Dword buffer 010: take byte 2 from output Dword buffer 011: take byte 3 from output Dword buffer 100: take byte 0 from intermediate feedback buffer 101: take byte 1 from intermediate feedback buffer 110: take byte 2 from intermediate feedback buffer 111: take byte 3 from intermediate feedback buffer
DOD_Ax [1:0]	Define on which SD pin the serial output data will appear: if both circuits attempt to drive the same SD pin in the same time slot, A1 gets preference. When a SD pin is not driven actively it is 3-stated. 00: at SD0 (for A1); at SD4 (for A2) 01: at SD1 10: at SD2 11: at SD3
LOW_Ax	Drive the SD pin which was driven in the previous time slot as output: for 7-bit clock cycles to active LOW and let then go to 3-state.
EOS	End Of Superframe: last record in time slot list, next time slot uses first record of the TSL (reset TSL pointer).

7.16.5 AUDIO CONFIGURATION

The configuration parameters are selected using two configuration registers, ACON1 and ACON2.

The ACON1 register is locally buffered. The download from the shadow register into the working register is performed when a DMA protection address is reached or immediately when both interfaces are not active (switched off, initial state).

Table 103 Audio Configuration Register 1 (ACON1)

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
F4	AUDIO_MODE [2:0]	31 to 29	RW	defines interface activation and combination
	MAXLEVEL [6:0]	28 to 22	RW	defines the maximum allowed absolute value for the most significant byte of an audio sample
	A1_SWAP	21	RW	defines if input (captured) data is stuffed in little-endian or big-endian format for A1 (4 byte swap if set)
	A2_SWAP	20	RW	defines if input (captured) data is stuffed in little-endian or big-endian format for A2 (4 byte swap if set)
	WS0_CTRL [1:0]	19 and 18	RW	function control for WS0 line
	WS0_SYNC [1:0]	17 and 16	RW	pulse position and width control for WS0 line
	WS1_CTRL [1:0]	15 and 14	RW	function control for WS1 line
	WS1_SYNC [1:0]	13 and 12	RW	pulse position and width control for WS1 line
	WS2_CTRL [1:0]	11 and 10	RW	function control for WS2 line
	WS2_SYNC [1:0]	9 and 8	RW	pulse position and width control for WS2 line
	WS3_CTRL [1:0]	7 and 6	RW	function control for WS3 line
	WS3_SYNC [1:0]	5 and 4	RW	pulse position and width control for WS3 line
	WS4_CTRL [1:0]	3 and 2	RW	function control for WS4 line
	WS4_SYNC [1:0]	1 and 0	RW	pulse position and width control for WS4 line

Table 104 Audio Configuration Register 2 (ACON2)

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
F8	A1_CLKSRC [4:0]	31 to 27	RW	defines the bit clock source for A1
	A2_CLKSRC [4:0]	26 to 22	RW	defines the bit clock source for A2
	INVERT_BCLK1	21	RW	input or output BCLK1 with inverted polarity
	INVERT_BCLK2	20	RW	input or output BCLK2 with inverted polarity
	BCLK1_OEN	19	RW	output enable BCLK1 (active LOW)
	BCLK2_OEN	18	RW	output enable BCLK2 (active LOW)

7.16.5.1 Audio mode control

There are 3 audio mode bits to select which TSL is active and how to synchronize and combine them. The first half of Table 105 supports asynchronous operation of A1 and A2, each following their own configuration bits and working independent of each other. In the second half of the table, the two interfaces work synchronously, according to the clock configuration setting of A1_CLKSRC. Some special cases are discussed below.

AUDIO_MODE [2:0] = 7: the two TSLs are chained to each other thus enabling a longer TSL, with up to 32 time slots to be built (SINGER format). There is only one common TSL pointer active which is able to reach the whole address range. The pointer synchronisation is done according to the A1 configuration. There is only one EOS bit needed.

AUDIO_MODE [2:0] = 5 $\leftrightarrow$ 6: both interfaces working synchronously on the same TSL half and with the same record. Synchronous change from one TSL to the other TSL when reaching the protection address. By that the TSL can be changed, by switching to the other TSL, without disturbing real time flow of audio streams. There is no time gap necessary for reprogramming a TSL. Reprogramming is performed by using the other TSL as shadow. The two TSL pointers are locked, i.e. each one pointing to its own list area, but resetting and incrementing synchronously.

Table 105 Audio_mode control

AUDIO_MODE [2:0]	AUDIO INTERFACE A1	AUDIO INTERFACE A2
0	off	off
1	processing TSL1; synchronisation according to A1 configuration	off
2	off	processing TSL2; synchronisation according to A2 configuration
3	processing TSL1; synchronisation according to A1 configuration	processing TSL2; synchronisation according to A2 configuration
4	processing TSL1; synchronisation according to A1 configuration	processing TSL2; TSL2 pointer is in sync (locked) to TSL1 pointer; TSL2 pointer = TSL1 pointer + 16
5	processing TSL1; synchronisation according to A1 configuration	processing TSL1 in common with A1; TSL2 pointer = TSL1 pointer
6	processing TSL2 in common with A2; TSL1 pointer = TSL2 pointer	processing TSL2; synchronisation according to A1 configuration
7	processing TSL1 and TSL2 in common with A2, synchronization according to A1 configuration; TSL1 pointer range is up to 32 records	processing TSL1 and TSL2 in common with A1; TSL2 pointer = TSL1 pointer

7.16.5.2 Audio input level monitoring

The audio input level monitoring feature allows the control of audio input levels without additional external hardware, by comparing the absolute value of the most significant byte of an audio sample to a programmable reference maximum level. The MAXLEVEL is defined by 7 bits, since serial audio data is transmitted in twos complement and the sign of the compared byte is not relevant for audio level control. Therefore, MAXLEVEL is programmable from 0 to 127. The twos complement value $|-128|$ is not reachable, but also not functionally needed. The comparison results are stored in the 32-bit level report register with one bit per time slot of TSL1 and TSL2, reporting whether there was a level violation in that time slot. The comparison runs all the time and the level report register is reset when it is read by software.

Table 106 Level report register

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
140	LEVEL_REPORT	31 to 0	R	stores the violation of MAXLEVEL for all 32 TSL records; reset to 0000H when read.

7.16.5.3 WS line controlling

The WSx_CTRL bits define which of the WS lines is output and controlled by which audio interface circuit (A1 or A2). WSx_SYNC defines the timing of WS signals.

Table 107 Static function control for word select lines

WSx_CTRL [1:0]	WS0 FUNCTION	WS1 FUNCTION	WS2 FUNCTION	WS3 FUNCTION	WS4 FUNCTION
00	3-state, input, rising edge resets TSL1 pointer	3-state	3-state	3-state	3-state, input, rising edge resets TSL2 pointer
01	output, controlled by TSL1	output, controlled by TSL1	output, controlled by TSL1	output, controlled by TSL1	output, controlled by TSL1
10	output, controlled by TSL2	output, controlled by TSL2	output, controlled by TSL2	output, controlled by TSL2	output, controlled by TSL2
11	output, active LOW	output, active LOW	output, active LOW	output, active LOW	output, active LOW

Table 108 Pulse width and position control

WSx_SYNC [1:0]	PULSE FUNCTION
00	I^2S style: WS goes active one bit clock cycle before MSB of time slot and stays active until LSB, i.e. one bit clock before MSB of next time slot
01	WS goes active in sync with MSB and stays active until next MSB, i.e. active in sync with current time slot
10	WS goes active one bit clock before MSB and stays active for one bit clock cycle, i.e. negative edge is in sync with beginning of time slot
11	SINGER style: WS goes active in sync with MSB and stays active for one bit clock cycle and for two bit clock cycles in first time slot of the superframe

7.16.5.4 Bit clock control

Specific to each audio interface, A1 or A2, is the programming of bit clock source.

Table 109 CLK source definition

AX_CLKSRC [4:0] (HEX)	A1 BIT CLOCK	A2 BIT CLOCK
1F to 13	reserved	reserved
12	ACLK divided-by-384	ACLK divided-by-384
11	ACLK divided-by-256	ACLK divided-by-256
10	ACLK divided-by-192	ACLK divided-by-192
0F	ACLK divided-by-128	ACLK divided-by-128
0E	ACLK divided-by-96	ACLK divided-by-96
0D	ACLK divided-by-64	ACLK divided-by-64
0C	ACLK divided-by-48	ACLK divided-by-48
0B	ACLK divided-by-32	ACLK divided-by-32
0A	ACLK divided-by-24	ACLK divided-by-24
09	ACLK divided-by-16	ACLK divided-by-16
08	ACLK divided-by-12	ACLK divided-by-12
07	ACLK divided-by-8	ACLK divided-by-8
06	ACLK divided-by-6	ACLK divided-by-6
05	ACLK divided-by-4	ACLK divided-by-4
04	ACLK divided-by-3	ACLK divided-by-3
03	ACLK divided-by-2	ACLK divided-by-2
02	ACLK	ACLK
01	BCLK2	BCLK1
00	BCLK1	BCLK2

7.16.6 SWITCHING AUDIO STREAMS

There are different levels of switching data streams on and off:

- DMA transfer enable; switching a DMA channel
- AUDIO_MODE; switching the two audio interfaces
- WSx_CTRL; switching a physical channel.

The AUDIO_MODE and WSx_CTRL programming is locally buffered and gets loaded when the corresponding DMA protection address is reached. When both interfaces are off, changes are loaded immediately.

If an audio interface is switched on, it will start working at TSL pointer reset. Disabling a DMA channel clears the corresponding FIFO and sets the DMA pointers to their base address: this is the initial state. It is recommended to enable the output DMA channels before activating the interface, since the output FIFO has to be filled with valid output data.

It is the responsibility of the software to configure data structures, TSL sequences and protection address in such a way, that they match each other.

7.17 I²C-bus interface

7.17.1 General description

The I²C-bus is a simple 2-wire bus for efficient inter-IC data exchange. Only two bus lines are required: a serial clock line (SCL) and a serial data line (SDA). It's a true multi-master bus including collision detection and arbitration to prevent data corruption if two or more masters simultaneously initiate data transfers. Serial clock synchronization allows devices with different bit rates to communicate via the same serial bus. The block diagram is shown in Fig.42.

7.17.2 Functional description

The I²C-bus performs byte oriented data transfers. Clock generation and bus control arbitration are controlled by hardware. The status register (IICSTA) reflects the status of the interface and the I²C-bus (see Table 110).
An interrupt after execution may be enabled optionally.
The bus clock generator supports clock rates from 5 to 400 kHz.

The I²C-bus interface is programmed through the transfer control register (IICTRF) which is shown in Table 112.
A write to this register starts the transfer sequence where up to 3 bytes are transferred: BYTE2, BYTE1 and BYTE0. Any of these 3 bytes may be disabled or enabled for use (as data byte or 7-bit address plus $\overline{RW}$ bit) in three I²C-bus protocol functions:

- START: start/restart and address device
- CONT: transfer data and continue
- STOP: transfer data and stop.

All bus operations are done via these three functions. The functional usage of each single byte is defined by the byte specific attribute information (see Table 113).

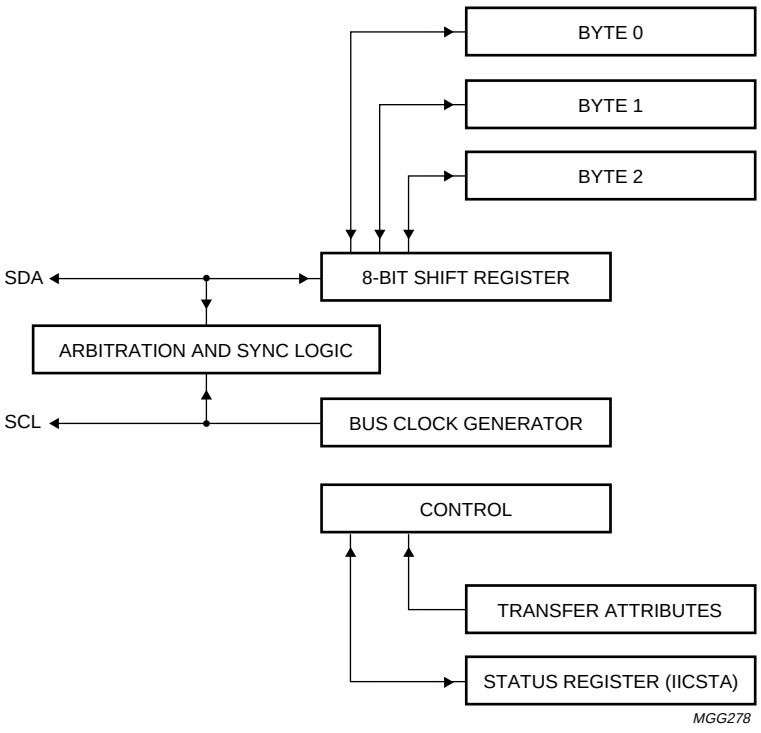


Fig.42 Block diagram of I²C-bus serial interface.

Table 110 Status register (IICSTA); note 1

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
90	IICCC [2:0]	10 to 8	RW	clock bit rate selection; see Table 111
	ABORT	7	RW	ABORT OPERATION: clears busy bit
	SPERR	6	RW	bus error due to invalid start/stop condition
	APERR	5	RW	NACK: error in address phase
	DTERR	4	RW	NACK: error in data transmission
	DRERR	3	RW	NACK: error when receiving data
	AL	2	RW	arbitration lost
	ERR	1	R	general error flag: has to be reset by clearing all error flags
	BUSY	0	R	operation ongoing

Note

1. The error bits have to be cleared, before a new command can be executed. This may be needed twice after using ABORT.

Table 111 Selection of I^2C-bus bit rate; note 1

IICCC2	IICCC1	IICCC0	BIT RATE(2)
1	0	1	PCI clock/6400
0	0	1	PCI clock/3200
1	0	0	PCI clock/480
1	1	0	PCI clock/320
1	1	1	PCI clock/240
0	0	0	PCI clock/120
0	1	0	PCI clock/80
0	1	1	PCI clock/60

Notes

1. Since the maximum width of spikes suppressed by the input filter depends on the PCI clock frequency, the appropriate timing parameter of 50 ns from the I^2C-bus specification is not fulfilled. Refer to the document for further details. This document may be ordered using the code 9398 393 40011.
2. The selected bit rate is the maximum bit rate and could be ‘stretched’ (slowed down) by slaves.

Table 112 Transfer control register (IICTRF)

OFFSET (HEX)	NAME	BIT	TYPE	DESCRIPTION
8C	BYTE2	31 to 24	RW	data/address register 2
	BYTE1	23 to 16	RW	data/address register 1
	BYTE0	15 to 8	RW	data/address register 0
	ATTR2 [1:0]	7 and 6	RW	attribute information for BYTE2
	ATTR1 [1:0]	5 and 4	RW	attribute information for BYTE1
	ATTR0 [1:0]	3 and 2	RW	attribute information for BYTE0
	ERR	1	RW	general error flag: has to be reset by clearing IICSTA
	BUSY	0	RW	operation ongoing

Table 113 ATTRx1 and ATTRx0; attribute information for BYTEx

ATTRx1	ATTRx0	SYMBOL	PROTOCOL FUNCTION				
1	1	START	start and address device, use BYTEx [7:1] as DA7 to D1 and BYTEx [0] as R/W bit				
			S	DA7 to DA1		R/W	A(1)
1	0	CONT	transfer D and continue, use BYTEx [7:0] as D7 to D0				
				D7 to D0			A(1)
0	1	STOP	transfer D and stop, use BYTEx as D7 to D0				
				D7 to D0			A/NA(1) P
0	0	NOP	no operation, don't use this byte				

Note

1. The generation of NA or A is performed by the controller hardware and is not user accessible.

7.17.2.1 Abbreviations used in Table 113

S: I^2C-bus START command

DA: 7-bit Device Address (BYTEx7 to BYTEx1)

R/W: Read/Write# bit (BYTEx0)

D: receive/transmit Data (BYTEx)

A: Acknowledge

NA: Negative Acknowledge (also used for identification of last master read data byte)

P: I^2C-bus STOP command.

7.17.2.2 Example

The protocol sequence for reading three bytes with subaddress access is illustrated in Fig.43. The procedure for this read operation is detailed below:

1. Address slave, write to IICTFR (see Fig.44):
   BYTE2 [7:1] = DA, BYTE2 [0] = 0 (write),
   ATTR2 = START
   BYTE1 = subaddress, ATTR1 = CONT
   BYTE0 [7:1] = DA, BYTE0 [0] = 1(read),
   ATTR0 = START
2. Wait until BUSY = 0
3. Check ERR bit, if it is inactive the slave target is successfully addressed
4. Transfer data, write attribute information to IICTFR (see Fig.45):
   BYTE2 = first received data byte, ATTR2 = CONT
   BYTE1 = second received data byte, ATTR1 = CONT
   BYTE0 = third received data byte, ATTR0 = STOP
5. Wait until BUSY = 0
6. Check ERR bit, if it is inactive IICTFR contains valid data.

Instead of checking the general error flag (ERR) after each single 3-byte sequence, it is possible to check the ERR at the end of the whole protocol sequence. During a bus cycle, the BUSY bit is set HIGH. At the end of a bus cycle an interrupt request is generated if enabled and BUSY is cleared if no error occurs. Writing to the IICTRF should not be done while the BUSY bit is active, otherwise the ERR flag will be set HIGH. If no transfer errors occur during the three transfer actions, the ERR bit will be set LOW. If an error occurs the ERR bit will be set HIGH and the BUSY bit stays HIGH. In this case the error and BUSY flags have to be cleared before starting a new operation.

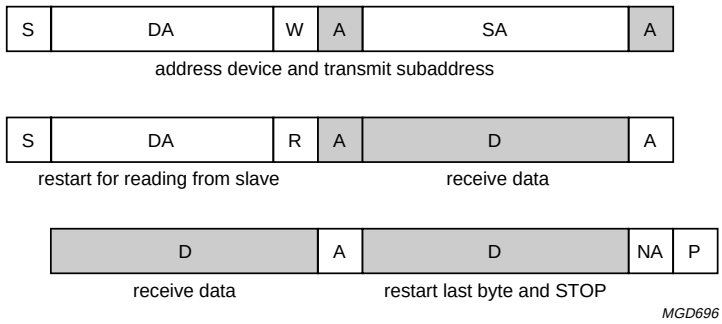


Fig.43 Protocol sequence for reading three bytes with subaddress access.

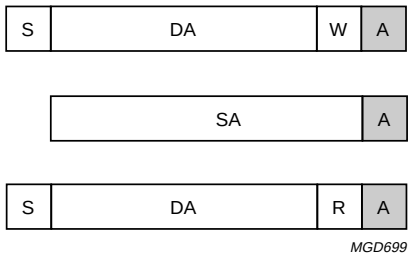


Fig.44 Address slave and write to IICTFR.

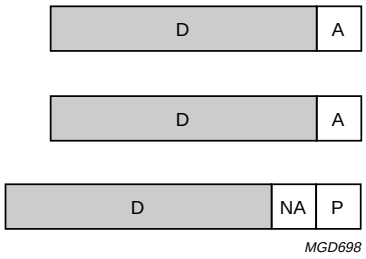


Fig.45 Transfer data and write attribute information to IICTFR.

7.18 SAA7146A register tables

Table 114 Registers and offsets sorted by functional groups

OFFSET (HEX)	NAME	TYPE	RAM	READ VALUE AFTER RESET	CORRESPONDING UPLOAD BIT
00	BaseOdd1	RW	yes	undefined	video DMA1 upload
04	BaseEven1	RW	yes	undefined	
08	ProtAddr1	RW	yes	undefined	
0C	Pitch1	RW	yes	undefined	
10	BasePage1	RW	yes	undefined	
14	Num_Line_Byte1	RW	yes	undefined	
18	BaseOdd2	RW	yes	undefined	video DMA2 upload
1C	BaseEven2	RW	yes	undefined	
20	ProtAddr2	RW	yes	undefined	
24	Pitch2	RW	yes	undefined	
28	BasePage2	RW	yes	undefined	
2C	Num_Line_Byte2	RW	yes	undefined	
30	BaseOdd3	RW	yes	undefined	video DMA3 upload
34	BaseEven3	RW	yes	undefined	
38	ProtAddr3	RW	yes	undefined	
3C	Pitch3	RW	yes	undefined	
40	BasePage3	RW	yes	undefined	
44	Num_Line_Byte3	RW	yes	undefined	
94	BaseA1_in	RW	read	undefined	immediate write access
98	ProtA1_in	RW	read	undefined	
9C	PageA1_in	RW	read	undefined	
A0	BaseA1_out	RW	read	undefined	
A4	ProtA1_out	RW	read	undefined	
A8	PageA1_out	RW	read	undefined	
AC	BaseA2_in	RW	read	undefined	
B0	ProtA2_in	RW	read	undefined	
B4	PageA2_in	RW	read	undefined	
B8	BaseA2_out	RW	read	undefined	
BC	ProtA2_out	RW	read	undefined	
C0	PageA2_out	RW	read	undefined	
48	PCI_BT_V	RW	yes	undefined	video DMA1, 2 or 3 upload
4C	PCI_BT_A	RW	read	undefined	immediate write access
120	PCI_VDP1	R	no	00000000	
124	PCI_VDP2	R	no	00000000	
128	PCI_VDP3	R	no	00000000	
12C	PCI_ADP1	R	no	00000000	
130	PCI_ADP2	R	no	00000000	

OFFSET (HEX)	NAME	TYPE	RAM	READ VALUE AFTER RESET	CORRESPONDING UPLOAD BIT
134	PCI_ADP3	R	no	00000000	–
138	PCI_ADP4	R	no	00000000	
13C	PCI_DDP	R	no	00000000	
FC	MC1	RW	no	00000100	immediate access
100	MC2	RW	no	0000077F	
104	RPS_ADDR0	RW	no	00000000	
108	RPS_ADDR1	RW	no	00000000	
C4	RPS_PAGE0	RW	read	undefined	immediate write access
C8	RPS_PAGE1	RW	read	undefined	
CC	RPS_THRESH0	RW	read	undefined	
D0	RPS_THRESH1	RW	read	undefined	
D4	RPS_TOV0	RW	read	undefined	
D8	RPS_TOV1	RW	read	undefined	
110	PSR	R	no	undefined	–
114	SSR	R	no	undefined	
DC	IER	RW	read	undefined	immediate write access
10C	ISR	RW	no	00000000	immediate access
E0	GPIO_CTRL	RW	read	undefined	immediate write access
118	EC1R	R	no	00000000	–
11C	EC2R	R	no	00000000	
E4	EC1SSR	RW	read	undefined	immediate write access
E8	EC2SSR	RW	read	undefined	
EC	ECT1R	RW	read	undefined	
F0	ECT2R	RW	read	undefined	
50	initial settings DD1 port	RW	yes	undefined	D1 Interface upload
54	video DATA stream handling at port DD1	RW	yes	undefined	
58	BRS control Register	RW	yes	undefined	BRS upload
5C	HPS control	RW	yes	undefined	HPS section 1 upload
60	HPS vertical scale	RW	yes	undefined	HPS section 2 upload
64	HPS vertical scale and gain	RW	yes	undefined	
68	HPS horizontal prescale	RW	yes	undefined	HPS section 1 upload
6C	HPS horizontal fine-scale	RW	yes	undefined	
70	BCS control	RW	yes	undefined	HPS section 2 upload
74	chroma key range	RW	yes	undefined	
78	HPS output and formats clip control	RW	yes	undefined	
8C	IICTFR	RW	yes	undefined	I^2C-bus upload
90	IIC_STA	RW	yes	000004xx	

OFFSET (HEX)	NAME	TYPE	RAM	READ VALUE AFTER RESET	CORRESPONDING UPLOAD BIT
88	DEBI_AD	RW	yes	undefined	DEBI upload
7C	DEBI_CONFIG	RW	yes	undefined	
80	DEBI_COMMAND	RW	yes	undefined	
84	DEBI_PAGE	RW	yes	undefined	
F4	ACON1	RW	read	undefined	immediate write access
F8	ACON2	RW	read	undefined	
144	FB_BUFFER1	RW	no	00000000	immediate access
148	FB_BUFFER2	RW	no	00000000	
140	LEVEL_REP	R	no	00000000	–
180-1BC	audio time slot registers 1	W	no	no read back	immediate access
1C0-1FC	audio time slot registers 2	W	no	no read back	

Table 115 Registers and offsets sorted by address-offset

OFFSET (HEX)	NAME	TYPE	RAM	READ VALUE AFTER RESET	CORRESPONDING UPLOAD BIT
00	BaseOdd1	RW	yes	undefined	video DMA1 upload
04	BaseEven1	RW	yes	undefined	
08	ProtAddr1	RW	yes	undefined	
0C	Pitch1	RW	yes	undefined	
10	BasePage1	RW	yes	undefined	
14	Num_Line_Byte1	RW	yes	undefined	
18	BaseOdd2	RW	yes	undefined	video DMA2 upload
1C	BaseEven2	RW	yes	undefined	
20	ProtAddr2	RW	yes	undefined	
24	Pitch2	RW	yes	undefined	
28	BasePage2	RW	yes	undefined	
2C	Num_Line_Byte2	RW	yes	undefined	
30	BaseOdd3	RW	yes	undefined	video DMA3 upload
34	BaseEven3	RW	yes	undefined	
38	ProtAddr3	RW	yes	undefined	
3C	Pitch3	RW	yes	undefined	
40	BasePage3	RW	yes	undefined	
44	Num_Line_Byte3	RW	yes	undefined	
48	PCI_BT_V	RW	yes	undefined	video DMA 1 2 or 3 upload
4C	PCI_BT_A	RW	read	undefined	immediate write access
50	initial settings DD1 port	RW	yes	undefined	D1 Interface upload
54	video DATA stream handling at port DD1	RW	yes	undefined	
58	BRS control register	RW	yes	undefined	BRS upload
5C	HPS control	RW	yes	undefined	HPS section 1 upload

OFFSET (HEX)	NAME	TYPE	RAM	READ VALUE AFTER RESET	CORRESPONDING UPLOAD BIT
60	HPS, vertical scale	RW	yes	undefined	HPS section 2 upload
64	HPS, vertical scale and gain	RW	yes	undefined	
68	HPS horizontal prescale	RW	yes	undefined	HPS section 1 upload
6C	HPS horizontal fine-scale	RW	yes	undefined	
70	BCS control	RW	yes	undefined	HPS section 2 upload
74	chroma key range	RW	yes	undefined	
78	HPS output and formats clip control	RW	yes	undefined	
7C	DEBI_CONFIG	RW	yes	undefined	DEBI upload
80	DEBI_COMMAND	RW	yes	undefined	
84	DEBI_PAGE	RW	yes	undefined	
88	DEBI_AD	RW	yes	undefined	
8C	IICTFR	RW	yes	undefined	I^2C-bus upload
90	IIC_STA	RW	yes	000004xx	
94	BaseA1_in	RW	read	undefined	immediate write access
98	ProtA1_in	RW	read	undefined	
9C	PageA1_in	RW	read	undefined	
A0	BaseA1_out	RW	read	undefined	
A4	ProtA1_out	RW	read	undefined	
A8	PageA1_out	RW	read	undefined	
AC	BaseA2_in	RW	read	undefined	
B0	ProtA2_in	RW	read	undefined	
B4	PageA2_in	RW	read	undefined	
B8	BaseA2_out	RW	read	undefined	
BC	ProtA2_out	RW	read	undefined	
C0	PageA2_out	RW	read	undefined	
C4	RPS_PAGE0	RW	read	undefined	
C8	RPS_PAGE1	RW	read	undefined	
CC	RPS_THRESH0	RW	read	undefined	
D0	RPS_THRESH1	RW	read	undefined	
D4	RPS_TOV0	RW	read	undefined	
D8	RPS_TOV1	RW	read	undefined	
DC	IER	RW	read	undefined	
E0	GPIO_CTRL	RW	read	undefined	
E4	EC1SSR	RW	read	undefined	
E8	EC2SSR	RW	read	undefined	
EC	ECT1R	RW	read	undefined	
F0	ECT2R	RW	read	undefined	immediate write access
F4	ACON1	RW	read	undefined	
F8	ACON2	RW	read	undefined	

OFFSET (HEX)	NAME	TYPE	RAM	READ VALUE AFTER RESET	CORRESPONDING UPLOAD BIT
FC	MC1	RW	no	00000100	immediate access
100	MC2	RW	no	0000077F	
104	RPS_ADDR0	RW	no	00000000	
108	RPS_ADDR1	RW	no	00000000	
10C	ISR	RW	no	00000000	
110	PSR	R	no	undefined	–
114	SSR	R	no	undefined	
118	EC1R	R	no	00000000	
11C	EC2R	R	no	00000000	
120	PCI_VDP1	R	no	00000000	
124	PCI_VDP2	R	no	00000000	
128	PCI_VDP3	R	no	00000000	
12C	PCI_ADP1	R	no	00000000	
130	PCI_ADP2	R	no	00000000	
134	PCI_ADP3	R	no	00000000	
138	PCI_ADP4	R	no	00000000	
13C	PCI_DDP	R	no	00000000	
140	LEVEL_REP	R	no	00000000	
144	FB_BUFFER1	RW	no	00000000	immediate access
148	FB_BUFFER2	RW	no	00000000	
180-1BC	audio time slot registers 1	W	no	no read back	
1C0-1FC	audio time slot registers 2	W	no	no read back	

8 BOUNDARY SCAN TEST

The SAA7146A has built-in logic and 5 dedicated pins to support boundary scan testing which allows board testing without special hardware (nails). The SAA7146A follows the *"IEEE Std. 1149.1 - Standard Test Access Port and Boundary-Scan Architecture"* set by the Joint Test Action Group (JTAG) chaired by Philips.

The 5 special pins are Test Mode Select (TMS), Test Clock (TCK), Test Reset ($\overline{\text{TRST}}$), Test Data Input (TDI) and Test Data Output (TDO).

The Built-in Self Test (BST) functions BYPASS, EXTEST, SAMPLE, CLAMP and IDCODE are all supported (see Table 116). Details about the JTAG BST-TEST can be found in specification *"IEEE Std. 1149.1 - Standard Test Access Port and Boundary-Scan Architecture"*. A file containing the detailed Boundary Scan Description Language (BSDL) description of the SAA7146A is available on request.

Table 116 BST instructions supported by the SAA7146A

INSTRUCTION	DESCRIPTION
BYPASS	this mandatory instruction provides a minimum length serial path (1-bit) between TDI and TDO when no test operation of the component is required
EXTEST	this mandatory instruction allows testing of off-chip circuitry and board level interconnections
SAMPLE	This mandatory instruction can be used to take a sample of the inputs during normal operation of the component. It can also be used to preload data values into the latched outputs of the boundary scan register
CLAMP	This optional instruction is useful for testing when not all ICs have BST. This instruction addresses the bypass register while the boundary scan register is in external test mode
IDCODE	this optional instruction will provide information on the components manufacturer, part number and version number

8.1 Initialization of boundary scan circuit

The Test Access Port (TAP) controller of an IC should be in the reset state (TEST_LOGIC_RESET) when the IC is in functional mode. This reset state also forces the instruction register into a functional instruction such as IDCODE or BYPASS.

To solve the power-up reset, the standard specifies that the TAP controller will be forced asynchronously to the TEST_LOGIC_RESET state by setting the $\overline{\text{TRST}}$ pin LOW.

8.2 Device identification codes

A device identification register is specified in *"IEEE Std. 1149.1 - Standard Test Access Port and Boundary-Scan Architecture"*. It is a 32-bit register which contains fields for the specification of the IC manufacturer, the IC part number and the IC version number. Its biggest advantage is the possibility to check for the correct ICs mounted after production and determination of the version number of ICs during field service.

When the IDCODE instruction is loaded into the BST instruction register, the identification register will be connected between TDI and TDO of the IC. The identification register will load a component specific code during the CAPTURE_DATA_REGISTER state of the TAP controller and this code can subsequently be shifted out. At board level this code can be used to verify component manufacturer, type and version number. The device identification register contains 32 bits, numbered from 31 to 0, where bit 31 is the most significant bit (nearest to TDI) and bit 0 is the least significant bit (nearest to TDO); see Fig.46.

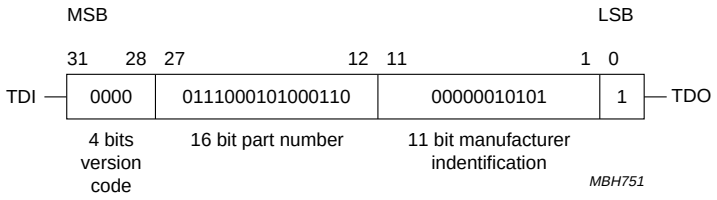


Fig.46 32 bits of identification code.

9 ELECTRICAL OPERATING CONDITIONS

- Operating time: the circuit is designed to be able to operate continuously
- Backup: no backup capability (standby) will be provided internally
- Handling: inputs and outputs are protected against electrostatic discharge in normal handling. However, to be totally safe, it is desirable to take normal handling precautions appropriate to handling MOS devices.

10 CHARACTERISTICS

V_{DDD} = 3.3 V for the internal core and for the I/O pad section. T_{amb} = 0 to 70 °C; unless otherwise specified.

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Supply						
V_{DDD}	digital supply voltage		3.0	3.3	3.6	V
I_{DDD}	digital supply current	video overlay RGB mode	–	400	–	mA
Data, clock and control inputs						
V_{IL}	LOW-level input voltage	clocks	–0.5	–	+0.6	V
		other outputs	–0.5	–	+0.8	V
V_{IH}	HIGH-level input voltage	clocks	2.4	–	5.5	V
		other inputs	2.0	–	5.5	V
I_{LI}	input leakage current	V_{IL} = 0 V	–	–	1	µA
C_i	input capacitance		–	–	10	pF
Data, clock and control outputs; note 1						
V_{OL}	LOW-level output voltage	clocks	0	–	0.6	V
		other outputs; note 2	0	–	0.6	V
V_{OH}	HIGH-level output voltage	clocks	2.6	–	V_{DDD}	V
		other outputs; note 2	2.4	–	V_{DDD}	V
I^2C-bus, SDA and SCL (pins 176 and 175)						
V_{IL}	LOW-level input voltage (V_{DDI2C} related input levels)	note 3	–0.5	–	$+0.3V_{DDI2C}$	V
V_{IH}	HIGH-level input voltage (V_{DDI2C} related input levels)	note 3	$0.7V_{DDI2C}$	–	V_{DDI2C} + 0,5	V
V_{hys}	hysteresis of Schmitt trigger inputs (V_{DDI2C} related input levels)	note 3	$0.05V_{DDI2C}$	–	–	V
V_{OL1}	LOW-level output voltage (open-drain or open-collector)	at 3 mA sink current	0	–	0.4	
V_{OL2}	LOW-level output voltage (open-drain or open-collector)	at 6 mA sink current	0	–	0.6	

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
$t_{o(f)}$	output fall time from V_{IHmin} to V_{ILmax} with a bus capacitance from 10 to 400 pF	note 4 up to 3 mA sink current at V_{OL1} up to 3 mA sink current at V_{OL2}	 $20 + 0.1C_b$ $20 + 0.1C_b$	 – –	 250 250	V
I_i	input current each I/O pin with an input voltage between 0.4 and $0.9V_{DDi2Cmax}$	note 3	$-10^{(5)}$	–	$+10^{(5)}$	µA
C_i	capacitance for each I/O pin		–	–	10	pF
Clock input timing (LLC_A and LLC_B); see Fig.47						
t_{LLC_A}, t_{LLC_B}	cycle time		31	–	45	ns
δ	duty factor	t_{LLCH}/t_{LLC}	40	50	60	%
t_r	rise time		–	–	5	ns
t_f	fall time		–	–	6	ns
Data and control input timing; see Fig.47						
t_{SU}	set-up time		6	–	–	ns
t_{HD}	hold time		3	–	–	ns
Clock output timing (LLC_A, LLC_B); see Fig.47						
C_L	output load capacitance		15	–	40	pF
t_{LLC_A}, t_{LLC_B}	cycle time		31	–	45	ns
δ	duty factor	t_{LLCH}/t_{LLC}	40	50	60	%
t_r	rise time	0.6 to 2.6 V	–	–	5	ns
t_f	fall time	2.6 to 0.6 V	–	–	5	ns
Data and control output timing; see Fig.47						
C_L	load capacitance		15	–	40	pF
t_{OH}	output hold time	C_L = 15 pF	4	–	–	ns
t_{pd}	propagation delay from positive edge of LLC_A, LLC_B	C_L = 40 pF	–	–	25	ns
PCI I/O signals						
DC SPECIFICATION						
V_{IH}	HIGH-level input voltage		2.0	–	5.75	V
V_{IL}	LOW-level input voltage		–0.5	–	+0.8	V
I_{LIH}	HIGH-level input leakage current	V_I = 2.7 V; note 1	–	–	70	µA
I_{LIL}	LOW-level input leakage current	V_I = 0.5 V; note 1	–	–	–70	µA
V_{OH}	HIGH-level output voltage	I_O = –2 mA	2.4	–	–	V
V_{OL}	LOW-level output voltage	I_O = 3 and 6 mA; note 6	–	–	0.55	V

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
C_i	input pin capacitance		–	–	10	pF
C_{CLK}	CLK pin capacitance		5	–	12	pF
C_{IDSEL}	IDSEL pin capacitance	note 7	–	–	8	pF
AC SPECIFICATION						
I_{OH}	switching current HIGH	$0 < V_o \leq 1.4$; note 8	–44	–	–	mA
		$1.4 < V_o < 2.4$; note 8	$-44 + \frac{(V_o - 1.4)}{0.024}$	–	–	mA
		$3.1 < V_o < V_{DDD}$; note 8	–	–	note 9	
	test point	$V_o = 3.1$ V; notes 9 and 10	–	–	–142	mA
I_{OL}	switching current LOW	$V_o > 2.2$ V; note 8	95	–	–	mA
		$2.2 > V_o > 0.55$; note 8	$V_o/0.023$	–	–	mA
		$0.71 > V_o > 0$; note 8	–	–	note 10	
	test point	$V_o = 0.71$ V; notes 9 and 10	–	–	206	mA
$t_{slew(r)}$	output rise slew rate	0.4 to 2.4 V; note 11	1	–	5	V/ns
$t_{slew(f)}$	output fall slew rate	2.4 to 0.4 V; note 11	1	–	5	V/ns
Timing parameters						
t_{val}	CLK to signal valid delay (bussed signals)	see note 12 and Fig.48	2	–	11	ns
$t_{val(ptp)}$	CLK to signal valid delay (point-to-point)	see note 12 and Fig.48	2	–	12	ns
t_{on}	float to active delay	see note 13 and Fig.48	2	–	–	ns
t_{off}	active to float delay	see note 13 and Fig.48	–	–	28	ns
t_{su}	input set-up time to CLK (bussed signal)	see note 12 and Fig.48	7	–	–	ns
$t_{su(ptp)}$	input set-up time to CLK (point-to-point)	see note 12 and Fig.48	10, 12	–	–	ns
t_h	input hold time from CLK	see Fig.48	0	–	–	ns
$t_{rst(CLK)}$	reset active time after CLK stable	note 14	100	–	–	µs
$t_{rst(off)}$	reset active to output float delay	notes 13, 14 and 15	–	–	40	ns

Notes

1. Input leakage currents include high-impedance output leakage for all bidirectional buffer with 3-state outputs.
2. Levels measured with load circuit: 1.2 kΩ at 3 V (TTL load) and C_L = 40 pF.
3. Voltage of the V_{DDI2C} sense pin is defined as V_{DDI2C} (4.75 : 5.0 : 5.25) (MIN. : TYP. : MAX.) for 5 V I^2C-bus devices and V_{DDI2C} (3.0 : 3.3 : 3.6) (MIN. : TYP. : MAX.) for 3 V I^2C-bus devices.
4. C_b = capacitance of one bus line measured in pF.
5. I/O pins of fast-mode devices must not obstruct the SDA and SCL lines if V_{DDI2C} is switched off.
6. FRAME#, TRDY#, IRDY#, DEVSEL# and STOP#.
7. Lower capacitance on this input only pin allows for non-resistive coupling to AD(xx).
8. Refer to the V/I curves in PCI specification. ‘Switching current high’ specifications are not relevant to INTA#, which are open-drain outputs.
9. $I_{OH} = 11.9 \times (V_o - 5.25) \times (V_o + 2.45)$ for $V_{DDD} > V_o > 3.1$ V.
10. $I_{OL} = 78.5 \times V_o \times (4.4 - V_o)$ for $0 V < V_o < 0.71$ V.
11. This parameter is to be interpreted as the cumulative edge rate across the specified range, rather than the instantaneous rate at any point within the transition range.
12. REQ# and GNT# are point-to-point signals, and have different output valid delay and input set-up times that do bussed signals. GNT# has a set-up time of 10 ns. REQ# has an output valid delay time of 12 ns. All other signals are bussed.
13. For purposes of active/float timing measurements the high-impedance or ‘off’ state is defined to be when the total current delivered through the component pin is less than or equal to the leakage current specification.
14. RST# is asserted and de-asserted asynchronously with respect to CLK.
15. All output drivers floated asynchronously when RST# is active.

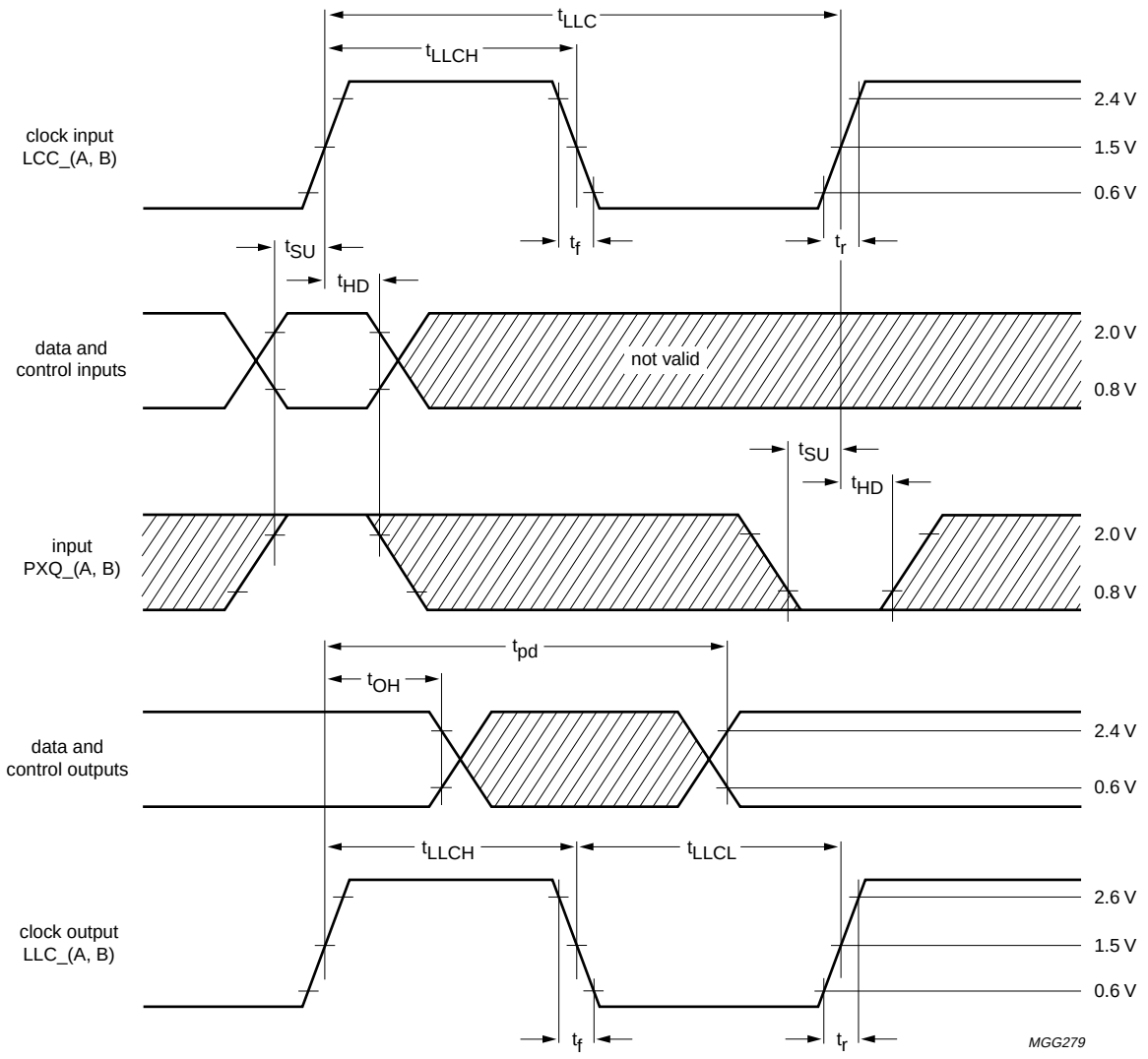


Fig.47 Clock/data timing.

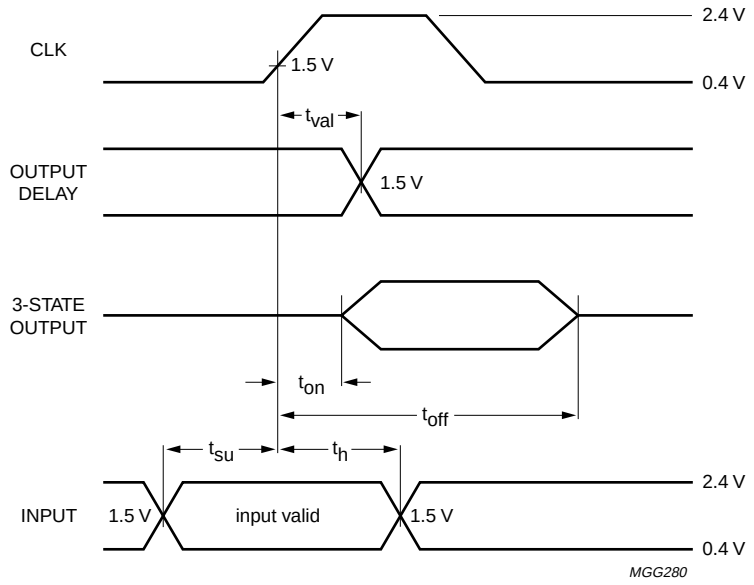


Fig.48 PCI I/O timing.

11 APPLICATION EXAMPLE

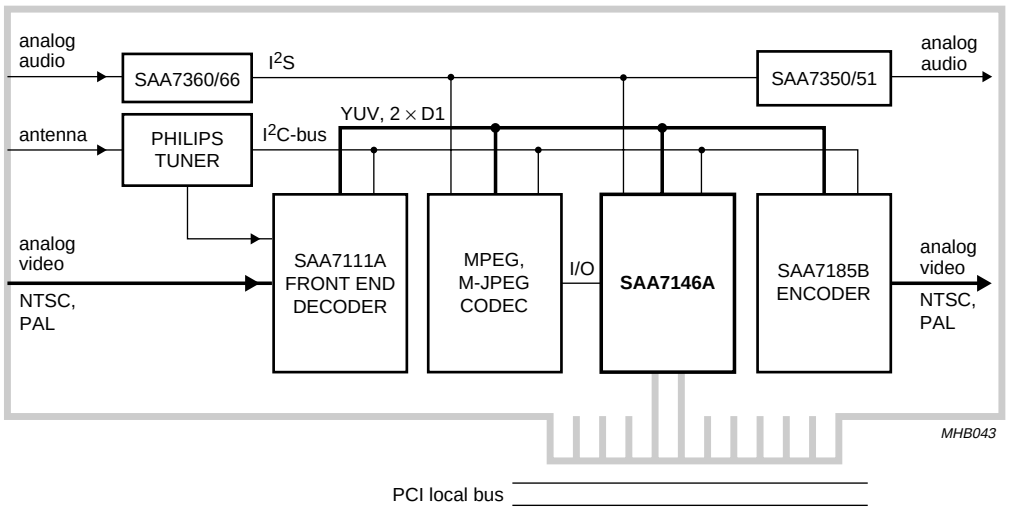
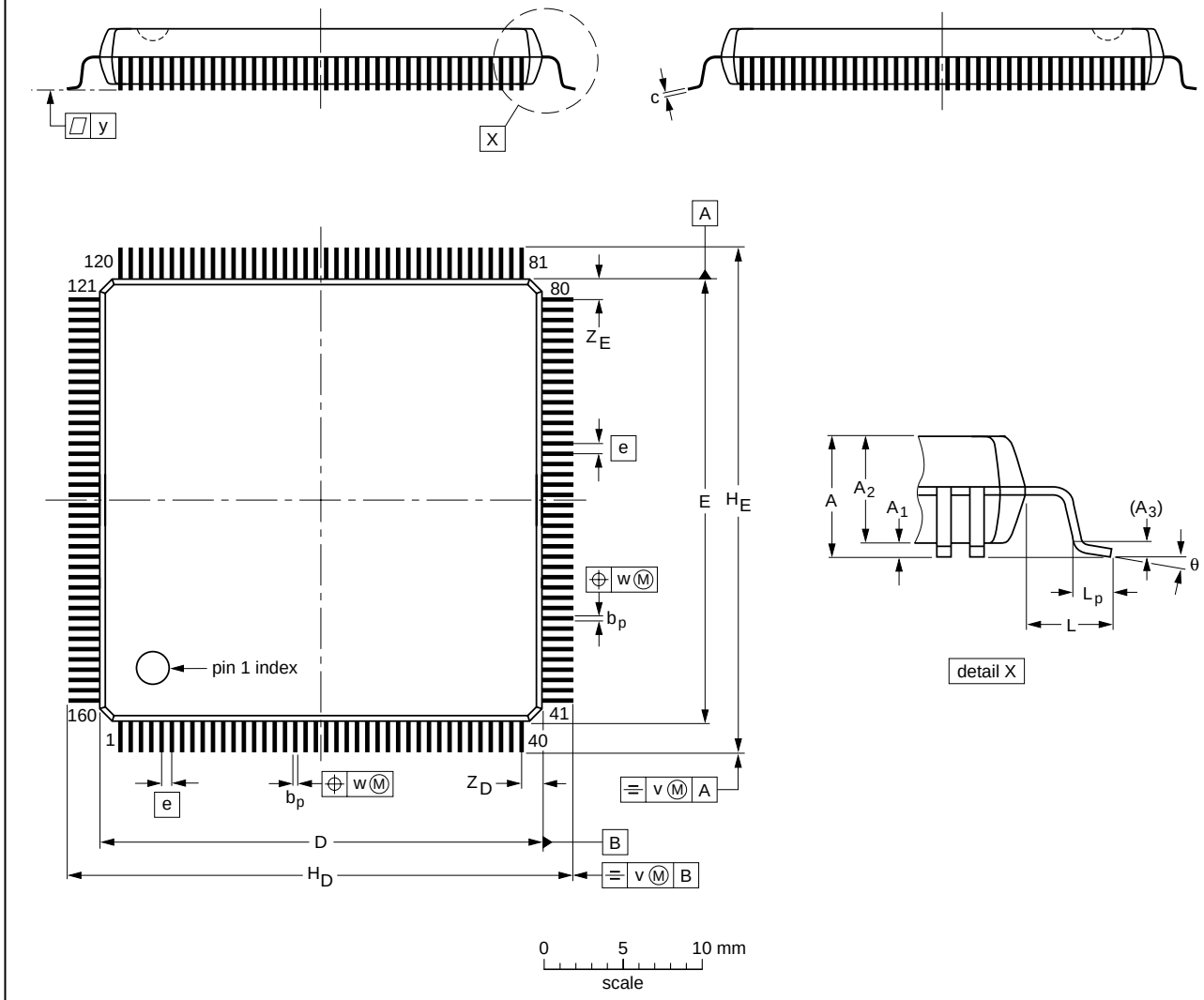


Fig.49 Application diagram.

12 PACKAGE OUTLINES

QFP160: plastic quad flat package;
160 leads (lead length 1.95 mm); body 28 x 28 x 3.4 mm; high stand-off height **SOT322-1**



DIMENSIONS (mm are the original dimensions)

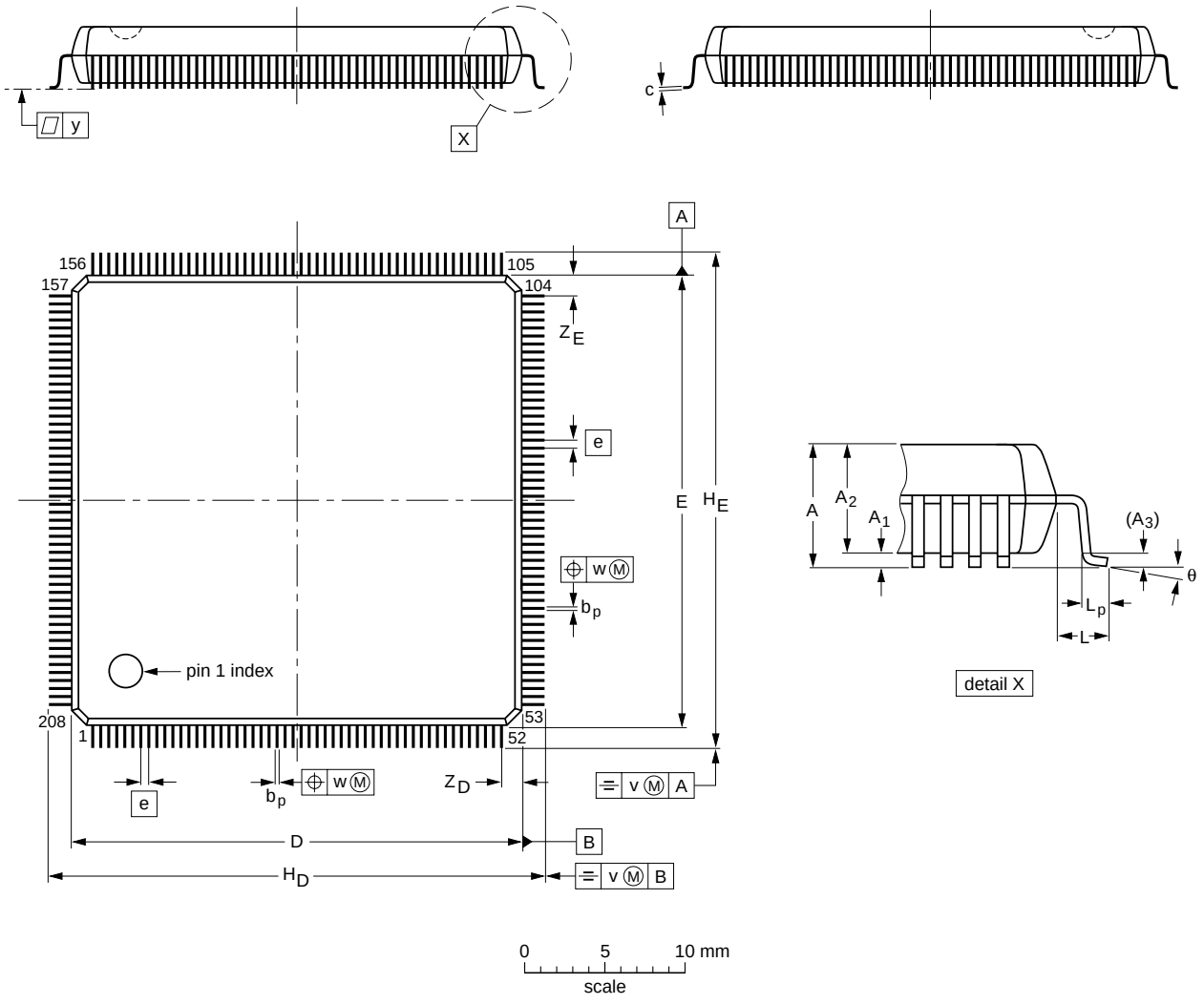
UNIT	A max.	A_1	A_2	A_3	b_p	c	$D^{(1)}$	$E^{(1)}$	e	H_D	H_E	L	L_p	v	w	y	$Z_D^{(1)}$	$Z_E^{(1)}$	θ
mm	3.95	0.40 0.25	3.70 3.15	0.25	0.40 0.25	0.23 0.13	28.1 27.9	28.1 27.9	0.65	32.2 31.6	32.2 31.6	1.95	1.1 0.7	0.3	0.15	0.1	1.5 1.1	1.5 1.1	8^o 0^o

Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT322-1		MO112DD1				~~95-02-04~~ 97-08-04

SQFP208: plastic shrink quad flat package;
208 leads (lead length 1.3 mm); body 28 x 28 x 3.4 mm **SOT316-1**



DIMENSIONS (mm are the original dimensions)

UNIT	A max.	A_1	A_2	A_3	b_p	c	$D^{(1)}$	$E^{(1)}$	e	H_D	H_E	L	L_p	v	w	y	$Z_D^{(1)}$	$Z_E^{(1)}$	θ
mm	4.10	0.40 0.25	3.70 3.15	0.25	0.25 0.13	0.23 0.13	28.1 27.9	28.1 27.9	0.5	30.9 30.3	30.9 30.3	1.3	0.70 0.45	0.1	0.1	0.075	1.45 1.05	1.45 1.05	8^o 0^o

Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT316-1						~~97-04-08~~ 97-08-01

13 SOLDERING

13.1 Introduction

There is no soldering method that is ideal for all IC packages. Wave soldering is often preferred when through-hole and surface mounted components are mixed on one printed-circuit board. However, wave soldering is not always suitable for surface mounted ICs, or for printed-circuits with high population densities. In these situations reflow soldering is often used.

This text gives a very brief insight to a complex technology. A more in-depth account of soldering ICs can be found in our *"IC Package Databook"* (order code 9398 652 90011).

13.2 Reflow soldering

Reflow soldering techniques are suitable for all QFP and SQFP packages.

The choice of heating method may be influenced by larger plastic QFP packages (44 leads, or more). If infrared or vapour phase heating is used and the large packages are not absolutely dry (less than 0.1% moisture content by weight), vaporization of the small amount of moisture in them can cause cracking of the plastic body. For more information, refer to the Drypack chapter in our *"Quality Reference Handbook"* (order code 9397 750 00192).

Reflow soldering requires solder paste (a suspension of fine solder particles, flux and binding agent) to be applied to the printed-circuit board by screen printing, stencilling or pressure-syringe dispensing before package placement.

Several methods exist for reflowing; for example, infrared/convection heating in a conveyor type oven. Throughput times (preheating, soldering and cooling) vary between 50 and 300 seconds depending on heating method. Typical reflow peak temperatures range from 215 to 250 °C.

13.3 Wave soldering

Wave soldering is **not recommended** for QFP packages and is **not suitable** for SQFP packages. This is because of the likelihood of solder bridging due to closely-spaced leads and the possibility of incomplete solder penetration in multi-lead devices.

CAUTION
Wave soldering is NOT applicable for all QFP packages with a pitch (e) equal or less than 0.5 mm.

If wave soldering cannot be avoided, for QFP packages with a pitch (e) larger than 0.5 mm, the following conditions must be observed:

- **A double-wave (a turbulent wave with high upward pressure followed by a smooth laminar wave) soldering technique should be used.**
- **The footprint must be at an angle of 45° to the board direction and must incorporate solder thieves downstream and at the side corners.**

During placement and before soldering, the package must be fixed with a droplet of adhesive. The adhesive can be applied by screen printing, pin transfer or syringe dispensing. The package can be soldered after the adhesive is cured.

Maximum permissible solder temperature is 260 °C, and maximum duration of package immersion in solder is 10 seconds, if cooled to less than 150 °C within 6 seconds. Typical dwell time is 4 seconds at 250 °C.

A mildly-activated flux will eliminate the need for removal of corrosive residues in most applications.

13.4 Repairing soldered joints

Fix the component by first soldering two diagonally-opposite end leads. Use only a low voltage soldering iron (less than 24 V) applied to the flat part of the lead. Contact time must be limited to 10 seconds at up to 300 °C. When using a dedicated tool, all other leads can be soldered in one operation within 2 to 5 seconds between 270 and 320 °C.

14 DEFINITIONS

Data sheet status	
Objective specification	This data sheet contains target or goal specifications for product development.
Preliminary specification	This data sheet contains preliminary data; supplementary data may be published later.
Product specification	This data sheet contains final product specifications.
Limiting values	
Limiting values given are in accordance with the Absolute Maximum Rating System (IEC 134). Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the Characteristics sections of the specification is not implied. Exposure to limiting values for extended periods may affect device reliability.	
Application information	
Where application information is given, it is advisory and does not form part of the specification.	

15 LIFE SUPPORT APPLICATIONS

These products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury. Philips customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Philips for any damages resulting from such improper use or sale.

16 PURCHASE OF PHILIPS I²C COMPONENTS

Purchase of Philips I²C components conveys a license under the Philips' I²C patent to use the components in the I²C system provided the system conforms to the I²C specification defined by Philips. This specification can be ordered using the code 9398 393 40011.

Philips Semiconductors – a worldwide company

Argentina: see South America

Australia: 34 Waterloo Road, NORTH RYDE, NSW 2113,
Tel. +61 2 9805 4455, Fax. +61 2 9805 4466

Austria: Computerstr. 6, A-1101 WIEN, P.O. Box 213, Tel. +43 160 1010,
Fax. +43 160 101 1210

Belarus: Hotel Minsk Business Center, Bld. 3, r. 1211, Volodarski Str. 6,
220050 MINSK, Tel. +375 172 200 733, Fax. +375 172 200 773

Belgium: see The Netherlands

Brazil: see South America

Bulgaria: Philips Bulgaria Ltd., Energoproject, 15th floor,
51 James Bourchier Blvd., 1407 SOFIA,
Tel. +359 2 689 211, Fax. +359 2 689 102

Canada: PHILIPS SEMICONDUCTORS/COMPONENTS,
Tel. +1 800 234 7381

China/Hong Kong: 501 Hong Kong Industrial Technology Centre,
72 Tat Chee Avenue, Kowloon Tong, HONG KONG,
Tel. +852 2319 7888, Fax. +852 2319 7700

Colombia: see South America

Czech Republic: see Austria

Denmark: Prags Boulevard 80, PB 1919, DK-2300 COPENHAGEN S,
Tel. +45 32 88 2636, Fax. +45 31 57 0044

Finland: Sinikalliontie 3, FIN-02630 ESPOO,
Tel. +358 9 615800, Fax. +358 9 61580920

France: 51 Rue Carnot, BP317, 92156 SURESNES Cedex,
Tel. +33 1 40 99 6161, Fax. +33 1 40 99 6427

Germany: Hammerbrookstraße 69, D-20097 HAMBURG,
Tel. +49 40 23 53 60, Fax. +49 40 23 536 300

Greece: No. 15, 25th March Street, GR 17778 TAVROS/ATHENS,
Tel. +30 1 4894 339/239, Fax. +30 1 4814 240

Hungary: see Austria

India: Philips INDIA Ltd, Band Box Building, 2nd floor,
254-D, Dr. Annie Besant Road, Worli, MUMBAI 400 025,
Tel. +91 22 493 8541, Fax. +91 22 493 0966

Indonesia: PT Philips Development Corporation, Semiconductors Division,
Gedung Philips, Jl. Buncit Raya Kav.99-100, JAKARTA 12510,
Tel. +62 21 794 0040 ext. 2501, Fax. +62 21 794 0080

Ireland: Newstead, Clonskeagh, DUBLIN 14,
Tel. +353 1 7640 000, Fax. +353 1 7640 200

Israel: RAPAC Electronics, 7 Kehilat Saloniki St, PO Box 18053,
TEL AVIV 61180, Tel. +972 3 645 0444, Fax. +972 3 649 1007

Italy: PHILIPS SEMICONDUCTORS, Piazza IV Novembre 3,
20124 MILANO, Tel. +39 2 6752 2531, Fax. +39 2 6752 2557

Japan: Philips Bldg 13-37, Kohnan 2-chome, Minato-ku, TOKYO 108,
Tel. +81 3 3740 5130, Fax. +81 3 3740 5077

Korea: Philips House, 260-199 Itaewon-dong, Yongsan-ku, SEOUL,
Tel. +82 2 709 1412, Fax. +82 2 709 1415

Malaysia: No. 76 Jalan Universiti, 46200 PETALING JAYA, SELANGOR,
Tel. +60 3 750 5214, Fax. +60 3 757 4880

Mexico: 5900 Gateway East, Suite 200, EL PASO, TEXAS 79905,
Tel. +9-5 800 234 7381

Middle East: see Italy

Netherlands: Postbus 90050, 5600 PB EINDHOVEN, Bldg. VB,
Tel. +31 40 27 82785, Fax. +31 40 27 88399

New Zealand: 2 Wagener Place, C.P.O. Box 1041, AUCKLAND,
Tel. +64 9 849 4160, Fax. +64 9 849 7811

Norway: Box 1, Manglerud 0612, OSLO,
Tel. +47 22 74 8000, Fax. +47 22 74 8341

Pakistan: see Singapore

Philippines: Philips Semiconductors Philippines Inc.,
106 Valero St. Salcedo Village, P.O. Box 2108 MCC, MAKATI,
Metro MANILA, Tel. +63 2 816 6380, Fax. +63 2 817 3474

Poland: Ul. Lukiska 10, PL 04-123 WARSZAWA,
Tel. +48 22 612 2831, Fax. +48 22 612 2327

Portugal: see Spain

Romania: see Italy

Russia: Philips Russia, Ul. Usatcheva 35A, 119048 MOSCOW,
Tel. +7 095 755 6918, Fax. +7 095 755 6919

Singapore: Lorong 1, Toa Payoh, SINGAPORE 319762,
Tel. +65 350 2538, Fax. +65 251 6500

Slovakia: see Austria

Slovenia: see Italy

South Africa: S.A. PHILIPS Pty Ltd., 195-215 Main Road Martindale,
2092 JOHANNESBURG, P.O. Box 7430 Johannesburg 2000,
Tel. +27 11 470 5911, Fax. +27 11 470 5494

South America: Al. Vicente Pinzon, 173, 6th floor,
04547-130 SÃO PAULO, SP, Brazil,
Tel. +55 11 821 2333, Fax. +55 11 821 2382

Spain: Balmes 22, 08007 BARCELONA,
Tel. +34 3 301 6312, Fax. +34 3 301 4107

Sweden: Kottbygatan 7, Akalla, S-16485 STOCKHOLM,
Tel. +46 8 5985 2000, Fax. +46 8 5985 2745

Switzerland: Allmendstrasse 140, CH-8027 ZÜRICH,
Tel. +41 1 488 2741 Fax. +41 1 488 3263

Taiwan: Philips Semiconductors, 6F, No. 96, Chien Kuo N. Rd., Sec. 1,
TAIPEI, Taiwan Tel. +886 2 2134 2865, Fax. +886 2 2134 2874

Thailand: PHILIPS ELECTRONICS (THAILAND) Ltd.,
209/2 Sanpavuth-Bangna Road Prakanong, BANGKOK 10260,
Tel. +66 2 745 4090, Fax. +66 2 398 0793

Turkey: Talatpasa Cad. No. 5, 80640 GÜLTEPE/ISTANBUL,
Tel. +90 212 279 2770, Fax. +90 212 282 6707

Ukraine: PHILIPS UKRAINE, 4 Patrice Lumumba str., Building B, Floor 7,
252042 KIEV, Tel. +380 44 264 2776, Fax. +380 44 268 0461

United Kingdom: Philips Semiconductors Ltd., 276 Bath Road, Hayes,
MIDDLESEX UB3 5BX, Tel. +44 181 730 5000, Fax. +44 181 754 8421

United States: 811 East Arques Avenue, SUNNYVALE, CA 94088-3409,
Tel. +1 800 234 7381

Uruguay: see South America

Vietnam: see Singapore

Yugoslavia: PHILIPS, Trg N. Pasica 5/v, 11000 BEOGRAD,
Tel. +381 11 625 344, Fax.+381 11 635 777

For all other countries apply to: Philips Semiconductors,
International Marketing & Sales Communications, Building BE-p, P.O. Box 218,
5600 MD EINDHOVEN, The Netherlands, Fax. +31 40 27 24825

Internet: http://www.semiconductors.philips.com

Printed in The Netherlands 655102/00/01/pp144 Date of release: 1998 Apr 09 Document order number: 9397 750 03115

Let's make things better.

Philips Semiconductors

PHILIPS