Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

DATA SHEET

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

Product specification
2004 Aug 25
Supersedes data of 1998 Apr 09

Philips
Semiconductors

PHILIPS

CONTENTS

1 FEATURES
1.1 Video processing
1.2 Audio processing
1.3 Scaling
1.4 Interfacing
1.5 General
2 GENERAL DESCRIPTION
3 QUICK REFERENCE DATA
4 ORDERING INFORMATION
5 BLOCK DIAGRAM
6 PINNING
7 FUNCTIONAL DESCRIPTION
7.1 General
7.2 PCI interface
7.3 Main control
7.4 Register Programming Sequencer (RPS)
7.5 Status and interrupts
7.6 General Purpose Inputs/Outputs (GPIO)
7.7 Event counter
7.8 Video processing
7.9 High Performance Scaler (HPS)
7.10 Binary Ratio Scaler (BRS)
7.11 Video data formats on the PCI-bus
7.12 Scaler register
7.13 Scaler event description
7.14 Clipping
7.15 Data Expansion Bus Interface (DEBI)
7.16 Audio interface
7.17 I^2C-bus interface
7.18 SAA7146A register tables
8 BOUNDARY SCAN TEST
8.1 Initialization of boundary scan circuit
8.2 Device identification codes
9 LIMITING VALUES
10 ELECTRICAL OPERATING CONDITIONS
11 CHARACTERISTICS
12 APPLICATION EXAMPLE
13 PACKAGE OUTLINE
14 SOLDERING
14.1 Introduction to soldering surface mount packages
14.2 Reflow soldering
14.3 Wave soldering
14.4 Manual soldering
14.5 Suitability of surface mount IC packages for wave and reflow soldering methods
15 DATA SHEET STATUS
16 DEFINITIONS
17 DISCLAIMERS
18 PURCHASE OF PHILIPS I^2C COMPONENTS

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-bus 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.48.
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-bus 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 1.6 mm); body 28 × 28 × 3.4 mm; high stand-off height	SOT322-2

5 BLOCK DIAGRAM

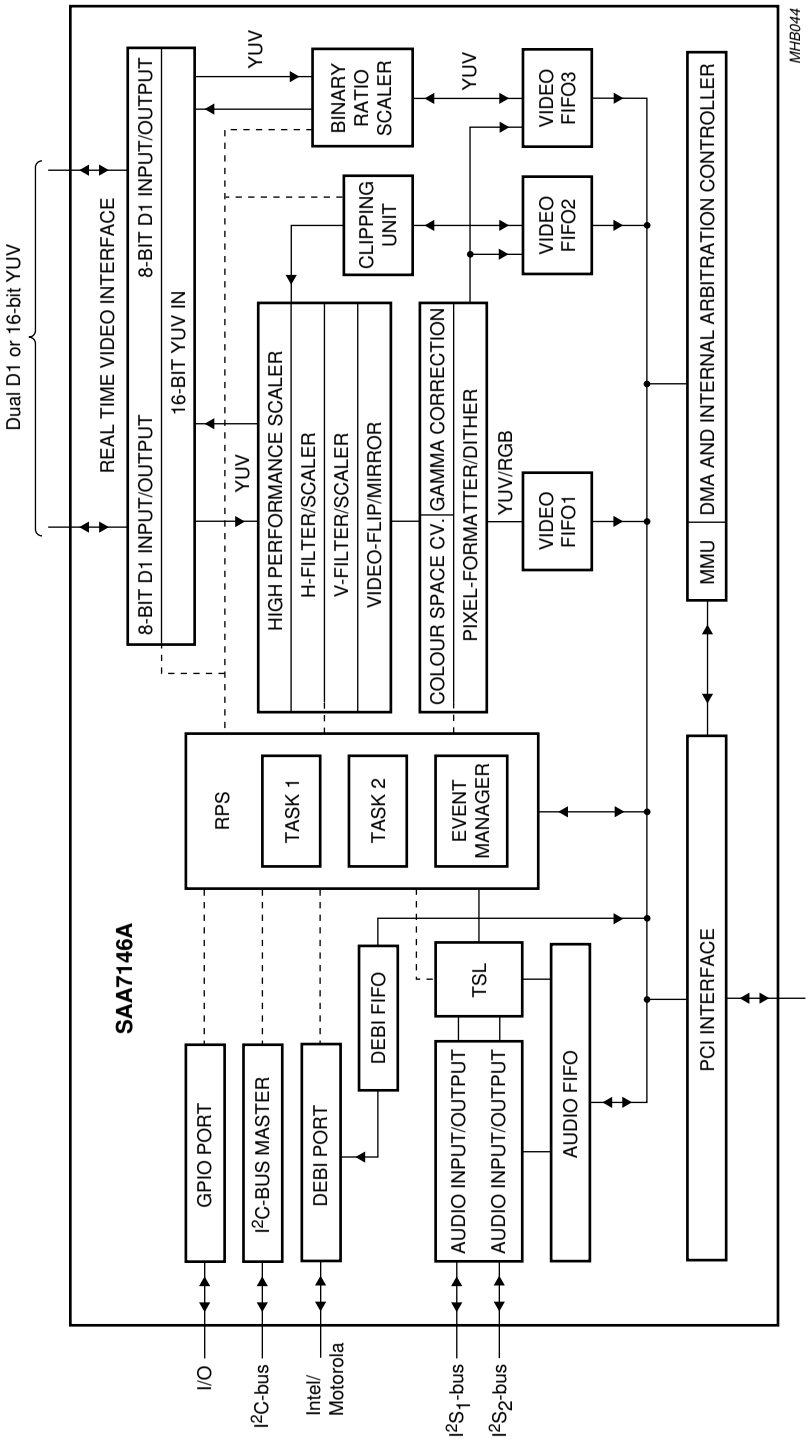

Fig.1 Block diagram.

6 PINNING

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
TRST_N	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)
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
AD_PCI26	38	I/O	bidirectional PCI multiplexed address/data bit 26
AD_PCI25	39	I/O	bidirectional PCI multiplexed address/data bit 25

SYMBOL	PIN	STATUS	DESCRIPTION
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 input signal
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 output signal (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)
AD_PCI7	79	I/O	bidirectional PCI multiplexed address/data bit 7
AD_PCI6	80	I/O	bidirectional PCI multiplexed address/data bit 6

SYMBOL	PIN	STATUS	DESCRIPTION
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 output signal (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
AD6	119	I/O	bidirectional DEBI multiplexed address data line bit 6
AD7	120	I/O	bidirectional DEBI multiplexed address data line bit 7

SYMBOL	PIN	STATUS	DESCRIPTION
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
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.

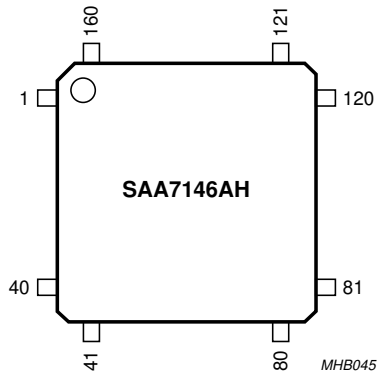

Fig.2 Pin configuration.

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 SAA7111A, SAA7113 and SAA7115 digital video encoder such as the SAA7128A and SAA7129A, 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.

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-bus devices such as the UDA1345, UDA1355 and UDA1380 (ADC and DAC) and UDA1334 (DAC) can be connected.

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 via the standard protocol external devices with speeds of up to 400 kbit/s.

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.3). 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.

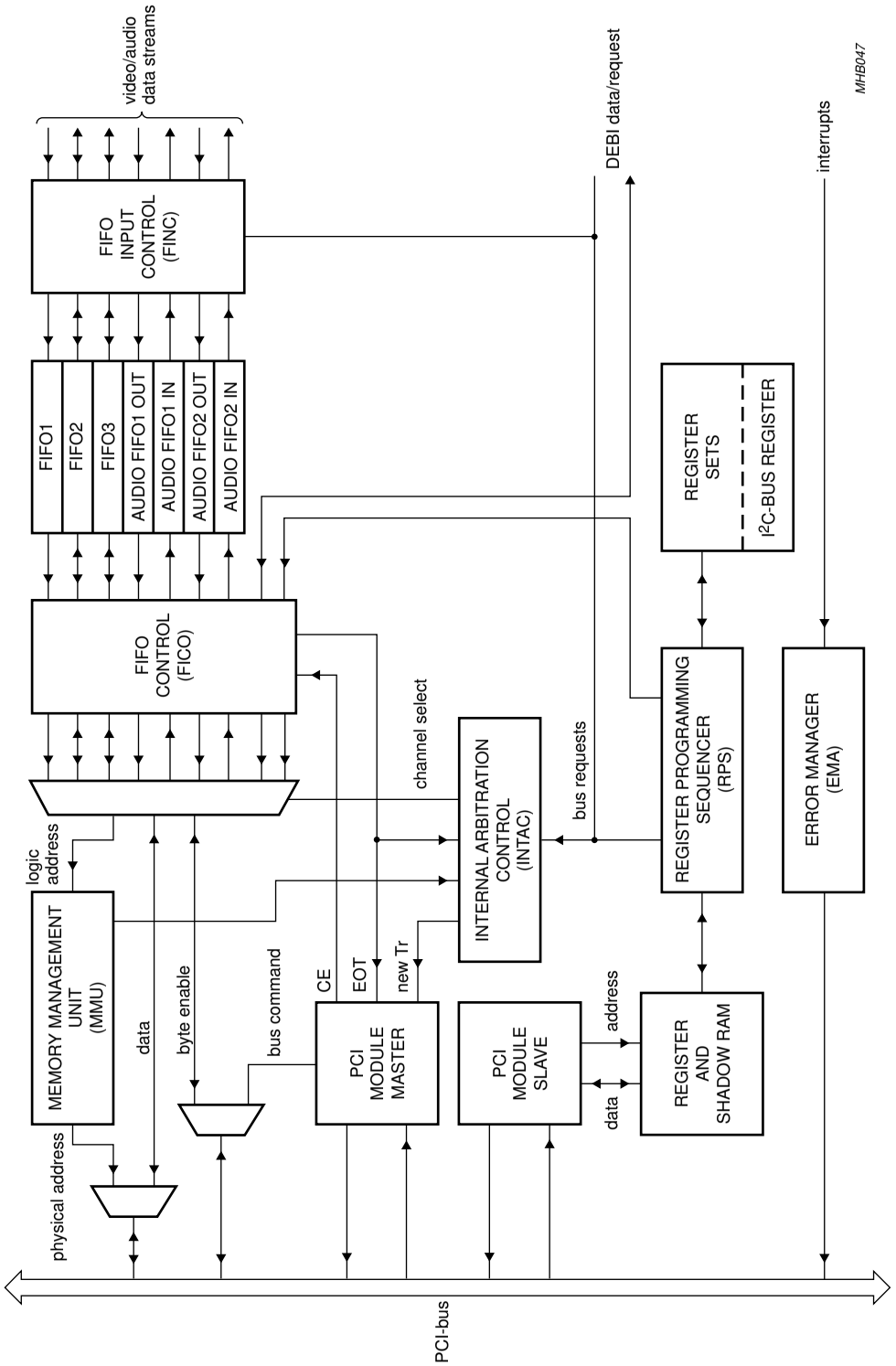

Fig.3 Block diagram of the PCI interface.

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 = 2-byte swap (3210 to 2301) 10 = 4-byte 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 = 2-byte swap (3210 to 2301) 10 = 4-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 = 2-byte swap (3210 to 2301) 10 = 4-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 one 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 69). 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.4. 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:

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

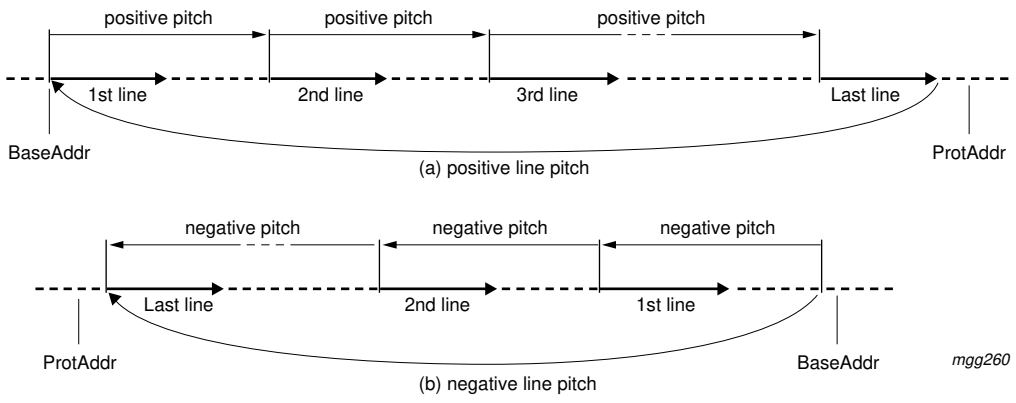

Fig.4 Handling of base and protection address using positive and negative line pitch.

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 an 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 generates 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 generates 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.5. 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.

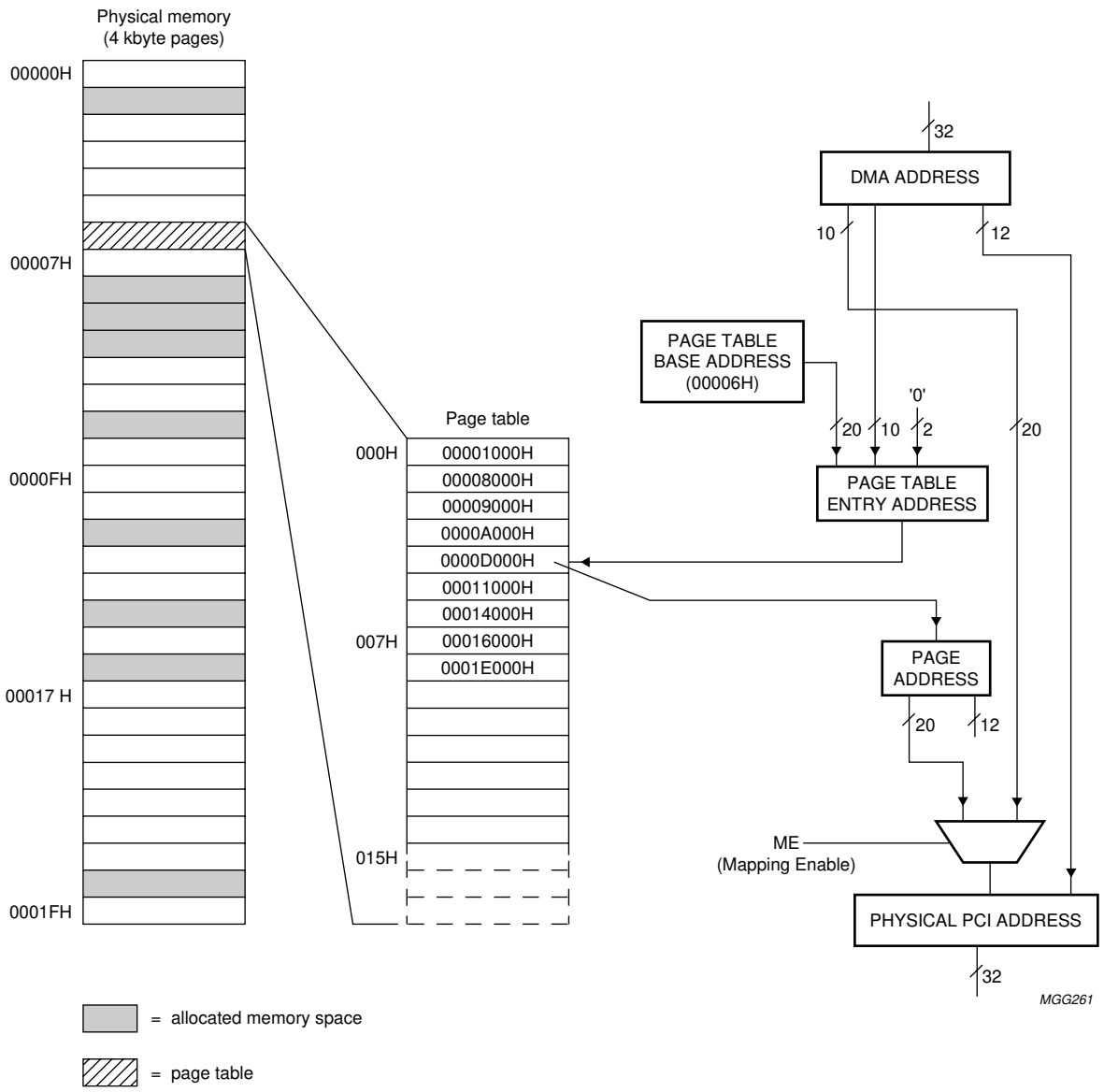

Fig.5 Memory Management Unit (MMU).

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.