*Lux*Sonor *DVDPC LS220*

Integrated DVD Decoder MPEG2/Dolby AC-3 PCI/Video Bus Version

1 Features

Video

- Real-time video decompression up to 720x576@25 fps or 720x480@30 fps
- MPEG2 ML/MP conforming to ISO-13818
  MPEG1 ML/MP conforming to ISO-11172
- B-frame compression to reduce memory
- I^2C interface supports Macrovision and close caption
- Software configurable glueless video port: VIP, VMI, AMC and LBP can all use the same connector
- OSD scaling for NTSC/PAL conversion

Audio

- 32 bit Audio DSP core using SonorDSP technology
- Dolby AC-3 5.1 channels of digital audio with optional downmixing to 2 channels
- MPEG1/2 audio decompression & linear PCM
- IEI-858 serial audio ports
- Integrated SP/DIF interface

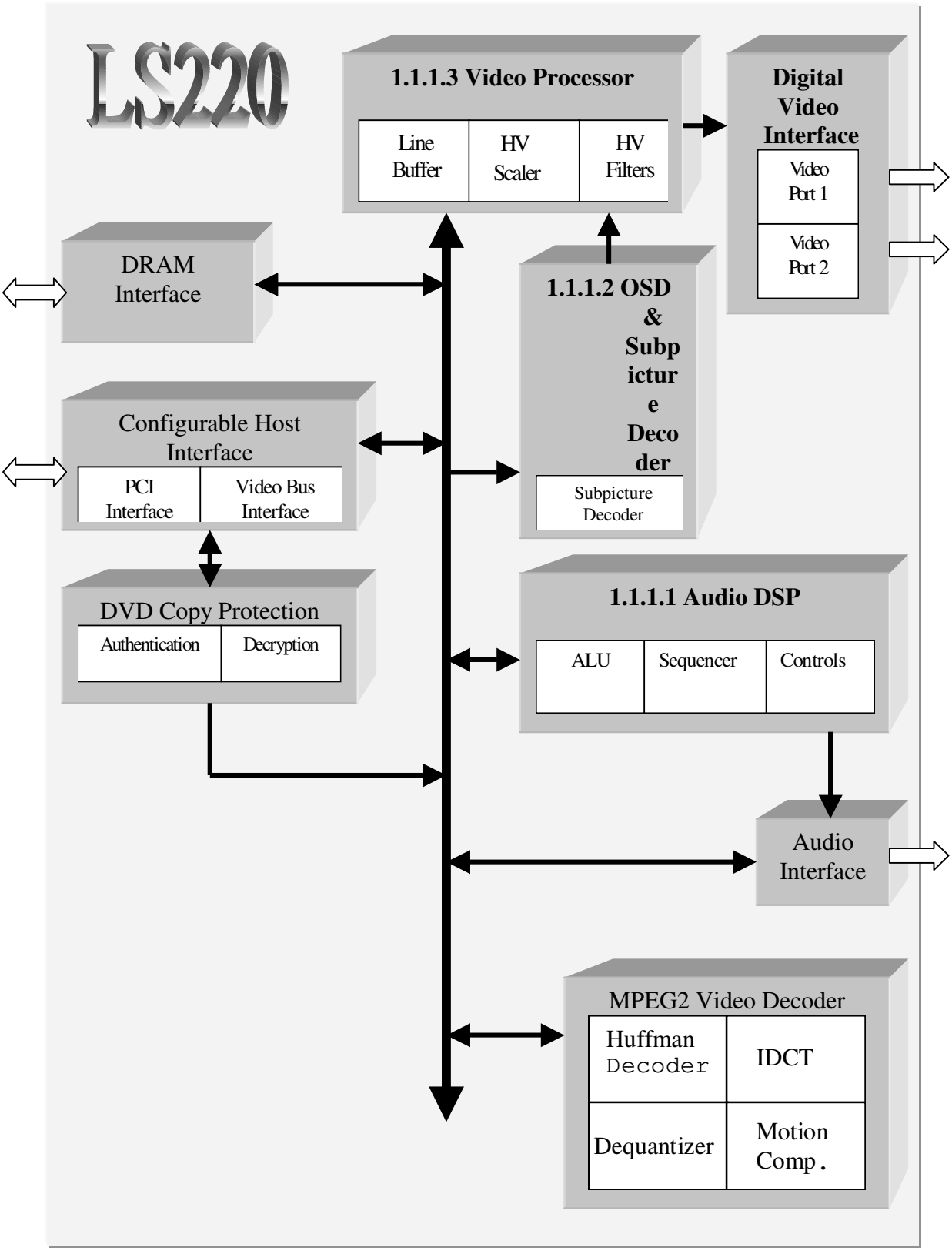
Other important features

- 2MB, 2.5MB or 3MB EDO DRAM (60ns)
- PCI interface can be disabled via jumpers which enables the Video Host Bus
- Six general purpose IO pins
- 208 pin PQFP
- Advanced 0.35micron CMOS technology

2 Table of Contents

1 Features 1
2 Table of Contents 2
3 LS220 Block diagram 3
4 DVDPC LS220 Functional Description 4
5 System Block Diagrams 7
6 Interface Buses: 8
7 LS220 Physical Pinout 9
7.1 LS220 Physical Pinout (continued) 10
7.2 LS220 Physical Pinout (continued) 11
8 Memory and Register Space 12
8.1 PCI Configuration Space 12
8.2 DRAM Space 14
8.3 LS220 Register Map 15
8.4 PCI Registers 21
8.5 DRAM Controller Registers 24
8.6 Audio DSP Registers 26
8.7 MPEG Video Decoder Registers 27
8.8 Synchronization Control Registers 34
8.9 PCM Audio Output Registers 41
8.10 Video Processor Registers 46
8.11 On-Screen Display Registers 58
8.12 DRAM Data Formats 60
9 Package Dimensions 67

3 LS220 Block diagram



4 DVDPC LS220 Functional Description

The DVDPC LS220 is a powerful and highly integrated DVD decoder solution for the PC market designed to perform all the DVD functions with a minimum PC software burden. The LS220 interfaces can be configured to support a wide set of protocols making it suitable for applications on motherboards, VGA cards, or standalone DVD or overlay cards. The LS220 is packaged in a 208-pin PQFP. Fabricated in advanced 0.35 micron technology, the LS220 has a 3.3V core and true 5V I/O's. Its primary function is to support DVD technology on the PC and comprises the following modules:

- MPEG2 Video Decoder
- AC-3/MPEG Audio Decoder
- Video Processor
- DVD Subpicture Processor
- DVD Copy Protection processor
- Interfaces:
  - Audio Interface
  - Digital Video interface
  - Configurable Host Interface
  - DRAM Interface

The MPEG2 Video Decoder an MPEG1 and MPEG2 video decoder optimized for minimum size while conforming to ISO 11172 and ISO 13818 standards and handles the bulk of MPEG video decompression with the following five submodules: Huffman Decoder, Inverse-Zig-Zag, Dequantizer, Inverse DCT and Pixel Generator. In addition to the standard MPEG2 functions, the LS220 contains a sophisticated error concealment function to reduce or eliminate visible artifacts that may be produced from errors or scratches on the DVD disk. The DVD standard imposes special requirements for seeks and long, frequent skips (due to multi-camera angle or parental controls), and the LS220 provides the functionality to support them. Additionally, the MPEG2 Video Decoder implements B-frame compression to reduce the amount of frame buffer memory needed.

The AC-3/MPEG Audio Decoder is composed of a proprietary programmable audio DSP core, called SonorDSP, and an audio PCM FIFO. These submodules are used to decompress audio streams and directly drive surround-sound serial outputs. The audio DSP can support MPEG layer 1 and layer 2, and Dolby AC-3 audio formats. Because the DSP uses external memory for program storage, additional algorithms can be added at no extra hardware cost. LuxSonor is working with leading 3D audio algorithm companies to supply a library of 3D sound software for the SonorDSP series. The DSP can also control the LS220 internal registers, using an arbiter shared with the host interface.

The Video Processor is responsible for advanced video processing functions to improve picture quality. These functions include horizontal and vertical filtering to minimize artifacts. The Video Processor provides frame to field and field to frame conversions for display of 24 frame

sequential or 60 field interlaced content on televisions or computer monitors. One of the most import features of the Video Processor is the support of scaling to convert NTSC to PAL size and vice-versa. The Video Processor also converts the frame rates from 25 to 30 frames per second or vice-versa to support the NTSC/PAL conversions.

The DVD Subpicture Decoder block provides the functions necessary to decode compressed DVD subpictures including run-length decoding. This block includes on-screen display (OSD) support for button and text controls.

The Synchronization Function is shared by dedicated hardware and the audio DSP, providing a flexible approach to synchronizing audio, video, and subpicture. The synchronization circuit can support PAL/NTSC conversion, and controls all the special DVD functions, such as seek, multi-angle and parental control.

The DVD Copy Protection Processor provides three functions required by the DVD standard. The first function is “Authentication”, which is used to verify that the DVD decoder board and the DVD-ROM drive are both licensed to support DVD copy protection. The second function is high speed (real time) “Decryption” that is used to decipher the video and audio data on an encrypted DVD disk. The final function is APS or Analog Protection System. The DVD disk may contain special information that must be passed to an external analog protection circuit (e.g. Macrovision) to prevent the transfer of a DVD title from a DVD disk to a VHS or 8mm tape.

The Digital Video Interface provides a flexible and powerful means of outputting digital video data to external devices in various formats. The basic output format is CCIR565/601/3 4:2:2 YUV. This format goes out through two 8-bit ports interleaving the luminance and chrominance data in a UYVY pattern. One of the 8-bit ports is normally used to control an NTSC/PAL encoder and pass the video data to it. This interface supports master and slave modes using an 8- or 16-bit interface, at CCIR-601 or 27MHz VGA (60 frames/sec) rates. The other 8-bit port is a special video bus that normally goes to a PC graphics controller. This second video port supports embedded CCIR656 control codes which are necessary for some graphics controllers. The second port supports a number of video buses including VIP, VMI, LPB and AMC using reconfigurable pins that allow a glueless interface using a single connector. Both video ports operate independently at 13.5Mpixels/sec which is the proper amount to support NTSC and PAL standards. The two video ports can also be combined together to drive a 16-bit video bus. In the 16-bit configuration, the combined ports can output the video data at 27Mpixels/sec non-interlaced to support YUV to RGB DAC applications. The combined 16-bit port also supports the standard 13.5Mpixels/sec interlaced mode.

The Audio Interface routes decoded PCM audio data to the audio ports. The LS220 can simultaneously output PCM data to the audio ports and to the Configurable Host Interface when in PCI mode. By providing this feature, the LS220 can output audio simultaneously to the PC sound system (e.g. sound card) and to an external device such as a television, VCR or amplifier. The Audio Interface supports audio sample sizes of 20, 18 or bits corresponding to Dolby AC-3 class A, B or C, respectively. The audio circuit includes an S/PDIF encoder which can output stereo PCM or 5.1 channel encoded AC-3 data to an external RS-422 line driver.

The Configurable Host Interface provides a very sophisticated method that directly supports many standard PC interfaces without any external glue logic. The Host Interface works in two primary modes: PCI and Video Bus. When set to PCI mode, the interface acts as a PCI 2.1 compatible slave which is used to program the LS220 and to transfer compressed data streams, including MPEG video and audio streams to the LS220 for processing. The LS220 can output PCM audio data through the Configurable Host Interface interface. This is particularly useful when the sound data must be sent to a wavetable audio card for mixing.

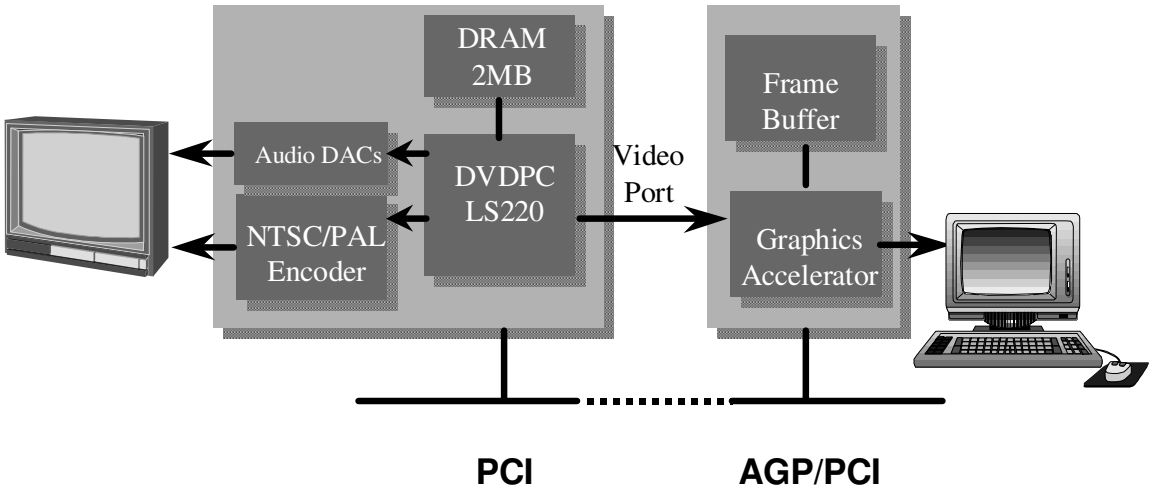
When programmed for Video Bus Host Mode, the Configurable Host Inteface supports the vast majority of video host bus standards including VMI, VIP, AMC and generic 8/16 bit synchronous and asynchronous modes, including DMA support.

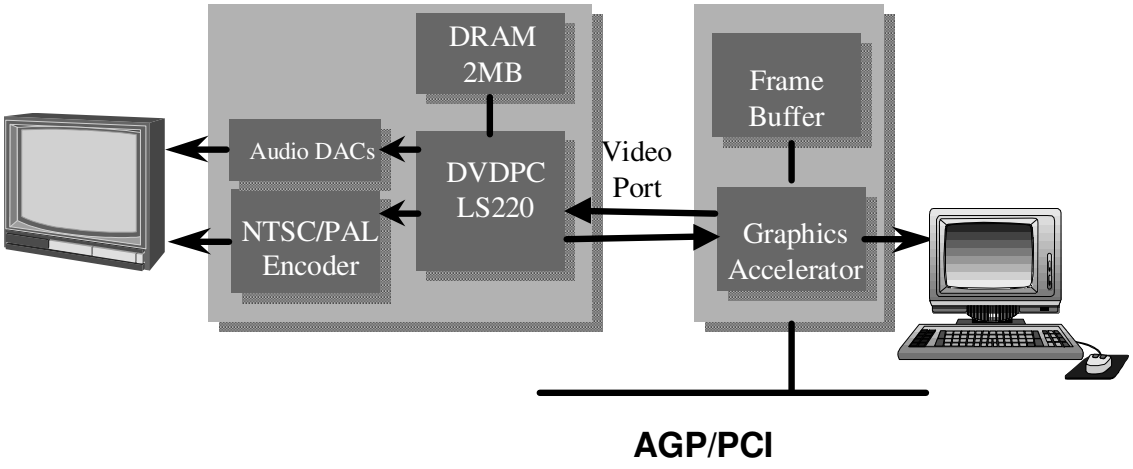
When the Configurable Host Bus interface is set to Video Bus mode, the LS220 is connected directly to the video host interface of a PC graphics controller, allowing a single PCI graphics card to include DVD functions using the LS220. In this configuration, the software drivers send the programming information and the compressed data streams to the graphics controller, which then passes the data to the LS220 through the Video Bus. The LS220 supports two video bus standards: VIP and VMI.

The DRAM Interface performs the EDO DRAM control and arbitration functions for all the other modules in the DVDPC LS220. The DRAM interface services and arbitrates a number of clients and stores their code and/or data within the local memory. This arbitration and scheduling guarantees the allocation of sufficient bandwidth to the various clients. The DRAM Interface supports standard 60ns EDO DRAMs through two bank types. The first bank is a high bandwidth bank that supports 64 bit wide transfers to and from memory. This bank must be exactly 2MB in size (generally 4 256k by 16 devices). The second bank is 16 bits wide and allows incremental memory expansion in ½ megabyte steps up to 3MB (i.e. 2.5MB and 3.0MB). The extra memory can be used for extra buffer space or additional program storage for new DSP algorithms.

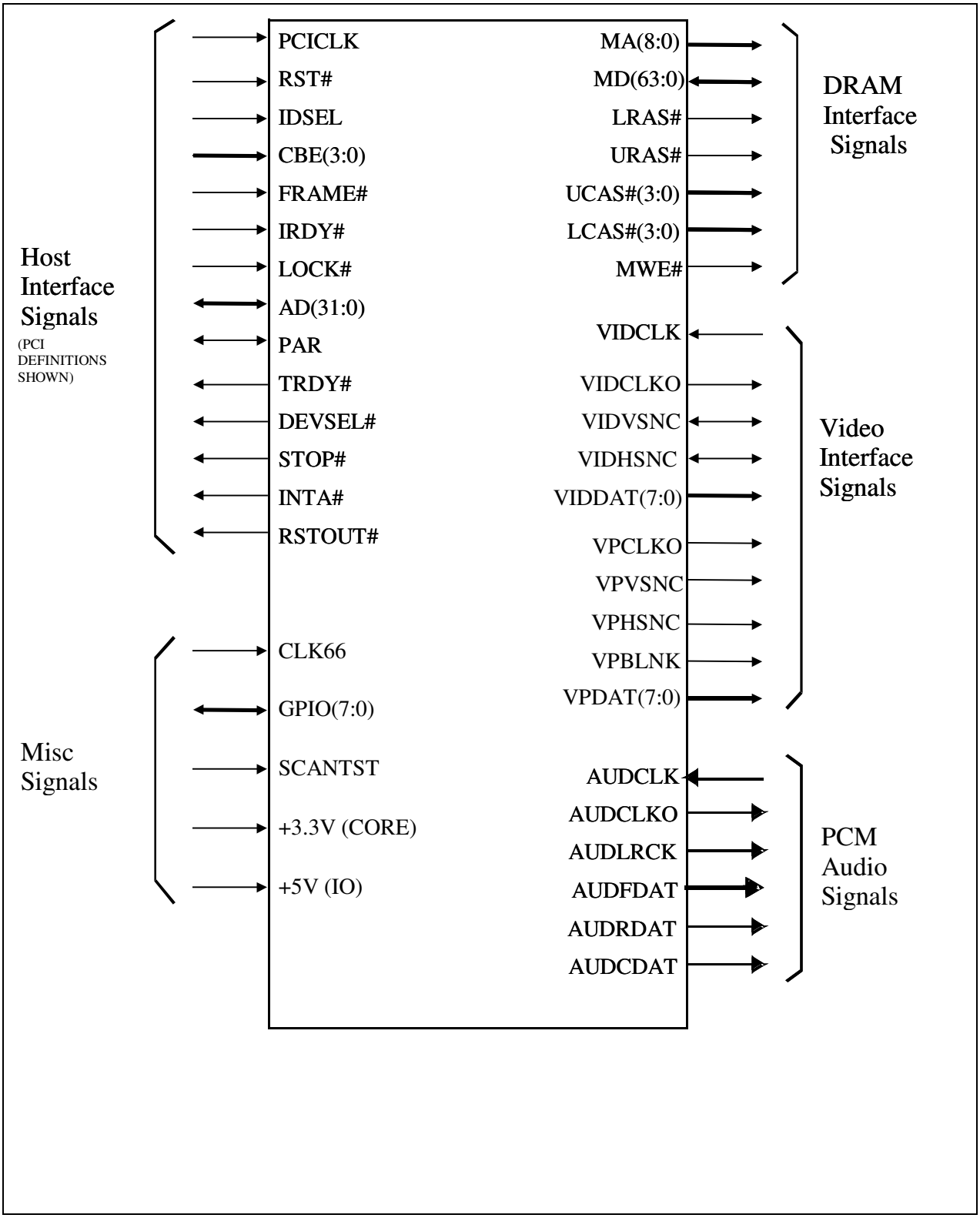
A set of eight general purpose I/O (GPIO) pins can be configured to support general or dedicated functions. Two pins can be configured as an I^2C master port to configure external hardware such as an analog copy protection circuit. Three other GPIO pins can be configured as a PCM input port, which can be used to receive encoded audio date from an external source.

5 System Block Diagrams





Interface Buses:



6 LS220 Physical Pinout

Pin	Main Function	Dir	Alt Function	Dir
1	GND		MFG_TST	I
2	GNDO	P		
3	VPORT_CK	O		
4	VCCO	P		
5	VCCO	P		
6	VPORT_D7	O	VDAT_15	O
7	VPORT_D6	O	VDAT_14	O
8	VPORT_D5	O	VDAT_13	O
9	VPORT_D4	O	VDAT_12	O
10	VPORT_D3	O	VDAT_11	O
11	VPORT_D2	O	VDAT_10	O
12	VPORT_D1	O	VDAT_9	O
13	VPORT_D0	O	VDAT_8	O
14	VMI_HSNC	O		
15	VMI_BLNK	O	LBP_HSNC	O
16	VMI_VSNC	O	LBP_VSNC	O
17	CLK27_O	O		
18	HSYNC	B		
19	VSYNC	B		
20	VDAT_7	O	HST_TYP3	I
21	VDAT_6	O	HST_TYP2	I
22	VDAT_5	O	HST_TYP1	I
23	VDAT_4	O	HST_TYP0	I
24	VDAT_3	O	DEV_ID3	I
25	VCCI	P		
26	VDAT_2	O	DEV_ID2	I
27	GNDI	P		
29	VDAT_1	O	DEV_ID1	I
30	VDAT_0	O	DEV_ID0	I
31	GPIO_7	B	IIC_CLK	B
32	GPIO_6	B	IIC_DAT	B
33	GPIO_5	B		I
34	GPIO_4	B		I
35	GPIO_3	B	SPDF_CLK	I
36	GPIO_2	B	SPDF_SNC	I
37	GPIO_1	B	SPDF_DAT	I
38	GPIO_O	B		
39	AUD_BCK	O		
40	AUD_LRCK	O		

Continued

Pin	Main Function	Dir	Alt Function	Dir
41	AUD_FDAT	O		
42	AUD_RDAT	O		
43	AUD_SDAT	O	SPDIF_O	O
44	RSTOUT_N	O		
45	INTA_L	O		
46	VCCI	P		
47	RST_N	I		
48	GNDI	P		
49	PCI_CLK	I		
50	VCCO	P		
51	AD_31	B		
52	GNDO	P		
53	AD_30	B		
54	AD_29	B		
55	AD_28	B		
56	AD_27	B		
57	AD_26	B		
58	AD_25	B		
59	GNDO	P		
60	AD_24	B		
61	CBE_3_N	B		
62	IDSEL	I		
63	AD_23	B	VMI_BHEN	
64	AD_22	B	VMI_RD	
65	AD_21	B	VMI_WR	
66	AD_20	B	VMI_CS	
67	GNDO	P		
68	AD_19	B	VMI_A3	
69	VCCI	P		
70	AD_18	B	VMI_A2	
71	AD_17	B	VMI_A1	
72	AD_16	B	VMI_A0	
73	CBE_2_N	B		
74	FRAME_N	B	VIP_HCTL	B
75	IRDY_N	B	VIP_AD0	B
76	TRDY_N	B		
77	GNDO	P		
78	GNDI	P		
79	DEVSEL_N	B		

6.1 LS220 Physical Pinout (continued)

Pin	Main Function	Dir	Alt Function	Dir
81	STOP_N	B		
82	LOCK_N	I		
83	PAR	B	VIP_AD1	B
84	CBE_1_N	B		
85	AD_15	B	VMI_D15	B
86	AD_14	B	VMI_D14	B
87	AD_13	B	VMI_D13	B
89	AD_12	B	VMI_D12	B
91	AD_11	B	VMI_D11	B
92	AD_10	B	VMI_D10	B
93	AD_9	B	VMI_D9	B
94	AD_8	B	VMI_D8	B
95	CBE_0_N	B		
96	AD_7	B	VMI_D7	B
98	AD_6	B	VMI_D6	B
99	AD_5	B	VMI_D5	B
100	AD_4	B	VMI_D4	B
101	AD_3	B	VMI_D3	B
102	AD_2	B	VMI_D2	B
103	AD_1	B	VMI_D1	B
105	AD_0	B	VMI_D0	B
106	VCCO	P		
107	MAIN_CLK	I		
104	GNDO	P		
105	CLK27_I	I		
106	GNDI	P		
107	AUD_CLK	I		
108	VCCI	P		
109	M_D_63	B		
110	M_D_62	B		
111	M_D_61	B		
112	M_D_60	B		
113	M_D_59	B		
114	M_D_58	B		
115	M_D_57	B		
116	M_D_56	B		
117	LCAS3_L	O		
118	UCAS3_L	O		
119	M_D_55	B		
120	M_D_54	B		

Pin	Main Function	Dir	Alt Function	Dir
121	M_D_53	B		
122	M_D_52	B		
123	M_D_51	B		
124	M_D_50	B		
125	M_D_49	B		
126	VCCI	P		
127	M_D_48	B		
128	GNDI	P		
129	GNDO	P		
130	M_D_47	B		
131	M_D_46	B		
132	M_D_45	B		
133	M_D_44	B		
134	M_D_43	B		
135	M_D_42	B		
136	M_D_41	B		
137	M_D_40	B		
138	LCAS2_L	O		
139	UCAS2_L	O		
140	M_D_39	B		
141	M_D_38	B		
142	M_D_37	B		
143	M_D_36	B		
144	M_D_35	B		
145	M_D_34	B		
146	VCCO	P		
147	M_D_33	B		
148	GNDO	P		
149	GNDI	P		
150	M_D_32	B		
151	VCCI	P		
152	M_D_31	B		
153	M_D_30	B		
154	M_D_29	B		
155	M_D_28	B		
156	M_D_27	B		
157	M_D_26	B		
158	M_D_25	B		
159	M_D_24	B		
160	LCAS1_L	O		

Continued

6.2 LS220 Physical Pinout (continued)

Pin	Main Function	Dir	Alt Function	Dir
161	UCAS1_L	O		
162	M_D_23	B		
163	M_D_22	B		
164	M_D_21	B		
165	M_D_20	B		
166	M_D_19	B		
167	M_D_18	B		
168	GNDO	P		
169	M_D_17	B		
170	M_D_16	B		
171	M_D_15	B		
172	M_D_14	B		
173	M_D_13	B		
174	M_D_12	B		
175	M_D_11	B		
176	M_D_10	B		
177	M_D_9	B		
178	M_D_8	B		
179	LCAS0_L	O		
180	UCAS0_L	O		
181	M_D_7	B		
182	M_D_6	B		
183	VCCO	P		
184	M_D_5	B		
185	GNDO	P		
186	GNDI	P		
187	M_D_4	B		
188	VCCI	P		
189	M_D_3	B		
190	M_D_2	B		
191	M_D_1	B		
192	M_D_0	B		
193	M_A_8	O		
194	M_A_7	O		
195	M_A_6	O		
196	M_A_5	O		
197	M_A_4	O		
198	M_A_3	O		
199	M_A_2	O		
200	M_A_1	O		

Pin	Main Function	Dir	Alt Function	Dir
201	VCCI	P		
202	M_A_0	O		
203	GNDI	P		
204	GNDO	P		
205	WE_L	O		
206	URAS_L	O		
207	LRAS_L	O		
208	NC			

7 Memory and Register Space

The LS220 external DRAM memory and internal registers are mapped into a block of memory space on the PCI bus. External DRAM transfers support burst read/write and byte/word/doubleword transfer modes. Internal register transfers support burst and non-burst 32-bit transfers only. Expansion DRAM is only sixteen bits wide, and will operate slower than main DRAM. Table 3-1 lists the PCI address mapping before the LS220 is configured with a base address. Table 3-2 shows how the byte and 32-bit word addresses correspond to the 32-bit data bus.

Table 3-1 - PCI Interrupt Register Bit Assignments

Space	Size	Range (hex)
Registers / SRAMDebug	2 MBytes	00_0000 - 1F_FFFF
Main DRAM	2 MBytes	20_0000 - 3F_FFFF
Expansion DRAM 1	1/2 MBytes	40_0000 - 47_FFFF
Expansion DRAM 2	1/2 MBytes	48_0000 - 4F_FFFF

Table 3-2 - DRAM/PCI Byte Addresses

data bus bits:	31 24	23 16	15 8	7 0
Byte address 0	3	2	1	0
Byte address 4	7	6	5	4
Byte address 8	B	A	9	8

7.1 PCI Configuration Space

An additional address space is defined for the PCI configuration registers. Table 3-3 lists the default values for the PCI configuration registers, which fields are read only (RO), and for command/status, which fields are used (C, S) or hardwired to 0 or 1. For Device ID, the least significant four bits are set using external resistors on the board.

Table 3-3 - PCI Interrupt Register Bit Assignments

bits 31-24	bits 23-16	bits 15-8		bits 7-0	Address
Device ID 001X (RO)		Vendor ID 1287 (RO)			00h
Status SSSSS01010000000		Command 000000000C0000C0			04h
Class_Code 040000 (RO)				Revision ID 01 (RO)	08h
BIST 00 (RO)	Header Type 00 (RO)	Latency Timer (6)	0 0	Cache Line Size	0Ch
Memory Base Hi	Memory Base Lo 00000008 (RO)				10h
00000000 (RO)					14h
00000000 (RO)					18h

00000000 (RO)				1Ch
00000000 (RO)				20h
00000000 (RO)				24h
00000000 (RO)				28h
00000000 (RO)				2Ch
00000000 (RO)				30h
00000000 (RO)				34h
00000000 (RO)				38h
Max_Lat 00 (RO)	Min_Gnt 00 (RO)	Int_Pin 01 (RO)	Int_Line	3Ch

7.2 DRAM Space

The LS220 connects to 2Mbytes of external DRAM, expandable to 2.5Mbytes or 3Mbytes. The DRAM is used internally by the LS220, and is also accessable to the host via the PCI bus. The DRAM is used for the following:

- Output buffers to hold PCM Audio and Video frame data
- Input FIFOs to hold Audio, Video and Sync stream data written by the host
- On-screen Display Pixel Data and Display Control buffers written by the host
- Audio DSP core firmware and data storage area

The Video output frame buffers are located at a fixed location in memory space. All other DRAM mapping is flexible and is controlled by the host.

The Video input and two Sync input FIFOs are flexibly mapped into DRAM via host registers. The Sync input FIFOs are used for synchronizing MPEG video and Subpictures. The mapping is done by programming a start and end address for each region. Audio sync requires simlar FIFO area in DRAM, but this area is defined via protocal between PC software and DSP firmware.

The On Screen Display uses DRAM to store Pixel Data buffers and Display Control buffers for Subpictue displays, using a flexible mapping. Control of the On Screen Display is done by the host writing an address and timestamp to the Subpicture and Vertical Blanking Interval Sync FIFOs. The address points to the Display Control buffer to be used when the timestamp time is reached. The Display Control buffers contain a pointer to the start address in DRAM of the Pixel Data buffer. The Display Control buffers also contain fields the hardware uses to interpret and display the Pixel Data buffers. Display Control Buffers are derived from SP Display Control Commands in the DVD stream, either by the PC or the DSP. If the PC creates the Display Control Buffers, then it also needs to perform any required decryption.

7.3 LS220 Register Map

Table 3-5 lists the registers for the LS220, gives their addresses before base address configuration, and indicates whether the registers are read/write (R/W), read only (RO) or write only (WO).

Table 3-5 - LS220 Registers

Address	Type	Function	Register Name
000	RO	PCI Interface	VIP_PlugnPlay_0
004	RO	PCI Interface	VIP_PlugnPlay_1
008	RO	PCI Interface	VIP_PlugnPlay_2
010	R/W	PCI Interface	Host_Control
014	RO	PCI Interface	Interrupt_Status
018	WO	PCI Interface	Interrupt_Clear
01C	WO	PCI Interface	Interrupt_Set
020	R/W	PCI Interface	Interrupt_Mask
024	R/W	PCI Interface	Reserved
028	R/W	PCI Interface	Reserved
02C	R/W	PCI Interface	Reserved
030	R/W	PCI Interface	AVV_FIFO_Pointer
034	WO	PCI Interface	AVV_FIFO_Write_Data
038	WO	PCI Interface	Reserved
03C	WO	PCI Interface	Reserved
080	R/W	DRAM Controller	DRAM_Controller_Setup
084	R/W	DRAM Controller	DRAM_Controller_Debug
100	R/W	Audio DSP	DSP_Boot_Code_Start_Address
104	R/W	Audio DSP	DSP_Run_Enable
180	R/W	MPEG Vid. Decoder	MPEG_Video_Control
184	R/W	MPEG Vid. Decoder	MPEG_Video_Setup
188	R/W	MPEG Vid. Decoder	MPEG_Video_FIFO_Start_Address
18C	R/W	MPEG Vid. Decoder	MPEG_Video_FIFO_End_Address
190	R/O	MPEG Vid. Decoder	MPEG_Video_FIFO_Current_Address
194	WO	MPEG Vid. Decoder	MPEG_Video_FIFO_Force_Address
198	WO	MPEG Vid. Decoder	MPEG_Video_FIFO_Add_Bytes
19C	RO	MPEG Vid. Decoder	MPEG_Video_FIFO_Curr_Bytes
1A0	R/W	MPEG Vid.	MPEG_Video_FIFO_Interrupt_Bytes

		Decoder	
1A4	RO	MPEG Vid. Decoder	MPEG_Video_FIFO_Total_Bytes
1A8	RO	MPEG Vid. Decoder	MPEG_Video_Error_Status
1AC	R/W	MPEG Vid. Decoder	Macroblock Width
1B0	R/W	MPEG Vid. Decoder	Macroblock Height
1B8	RO	MPEG Vid. Decoder	MPEG_Video_Debug_1
1BC	RO	MPEG Vid. Decoder	MPEG_Video_Debug_2

Address	Type	Function	Register Name
200	R/W	Sync Control	Audio_Sync_Control
204	R/W	Sync Control	Video_Sync_Control
208	RO	Sync Control	Video_Sync_Status
20C	R/W	Sync Control	Video_Display _Waitline
210	R/W	Sync Control	Video_Decode_Frame_Period
214	R/W	Sync Control	System_Time_Clock_Interval
218	R/W	Sync Control	System_Time_Clock
21C	R/W	Sync Control	Timer_Bottom_Bits
220	R/W	Sync Control	Video_PTS_FIFO_DRAM_Start_Address
224	R/W	Sync Control	Video_PTS_FIFO_DRAM_End_Address
228	R/W	Sync Control	Video_PTS_FIFO_DRAM_Write_Address
22C	RO	Sync Control	Video_PTS_FIFO_DRAM_Read_Address
230	R/W	Sync Control	Subpict_PTS_FIFO_DRAM_Start_Address
234	R/W	Sync Control	Subpict_PTS_FIFO_DRAM_End_Address
238	R/W	Sync Control	Subpict_PTS_FIFO_DRAM_Write_Address
23C	RO	Sync Control	Subpict_PTS_FIFO_DRAM_Read_Address
240	R/W	Sync Control	Highlight Starting PTS
244	R/W	Sync Control	Highlight End PTS
248	R/W	Sync Control	Button End PTS
24C	RW	Sync Control	Highlight Control Information Address
250	R/W	Sync Control	Video_PTS
254	R/W	Sync Control	Audio_PTS
258	RO	Sync Control	Subpicture_PTS
25C	RO	Sync Control	Audio_Time
260	RO	Sync Control	Video_Sync_Debug
264	RO	Sync Control	DSP_Debug
268	RW	Sync Control	Frame Number
270	RO	Sync Control	DSP Interrupt Status
274	RW	Sync Control	DSP Interrupt control
278	RW	Sync Control	DSP Interrupt Set
27C	RW	Sync Control	DSP Interrupt Clear
280	R/W	PCM Output	PCM_Frequency_Generator_Control
284	R/W	PCM Output	PCM_Output_Control
288	R/W	PCM Output	PCM_FIFO_Start_Address
28C	R/W	PCM Output	PCM_FIFO_End_Address

290	R/W	PCM Output	PCM_FIFO_Interrupt_Address
294	RO	PCM Output	PCM_FIFO_Current_Address
298	R/W	PCM Output	SPDIF_Channel_Status_Word

300	R/W	Video Processor	Video_Processor_Control
304	R/W	Video Processor	Video_DRAM_Line_Length
308	R/W	Video Processor	Display_ActiveX
30C	R/W	Video Processor	Display_ActiveY
310	R/W	Video Processor	Blank_Color
314	R/W	Video Processor	Internal_HSync_Count
318	R/W	Video Processor	Internal_VSync_Count
31C	R/W	Video Processor	Horizontal_Y_Offset
320	R/W	Video Processor	Horizontal_UV_Offset
324	R/W	Video Processor	Vertical_Offset
328	R/W	Video Processor	Video_Horizontal_Scale
32C	R/W	Video Processor	Video_Vertical_Scale
330	WO	Video Processor	Gamma_Correction_18_00
334	WO	Video Processor	Gamma_Correction_38_20
338	WO	Video Processor	Gamma_Correction_58_40
33C	WO	Video Processor	Gamma_Correction_78_60
340	WO	Video Processor	Gamma_Correction_98_80
344	WO	Video Processor	Gamma_Correction_B8_A0
348	WO	Video Processor	Gamma_Correction_D8_C0
34C	WO	Video Processor	Gamma_Correction_F8_E0
350	R/W	Video Processor	GPIO_Control
354	R/W	Video Processor	I2C_ReadWrite_2Bytes
358	R/W	Video Processor	I2C_Write_3Bytes
35C	R/W	Video Processor	I2C_Write_4Bytes
364	R/W	Video Processor	I2C_Control
368	RO	Video Processor	I2C_Status
36C	RO	Video Processor	I2C_Read_Data
370	R/W	Video Processor	SPDIF_Input_Control
374	RO	Video Processor	SPDIF_Input_Data
380	R/W	On Screen Display	Subpicture_Color0
384	R/W	On Screen Display	Subpicture_Color1
388	R/W	On Screen Display	Subpicture_Color2
38C	R/W	On Screen Display	Subpicture_Color3
390	R/W	On Screen Display	Subpicture_Color4
394	R/W	On Screen Display	Subpicture_Color5
398	R/W	On Screen	Subpicture_Color6

		Display	
39C	R/W	On Screen Display	Subpicture_Color7
3A0	R/W	On Screen Display	Subpicture_Color8
3A4	R/W	On Screen Display	Subpicture_Color9
3A8	R/W	On Screen Display	Subpicture_Color10
3AC	R/W	On Screen Display	Subpicture_Color11
3B0	R/W	On Screen Display	Subpicture_Color12
3B4	R/W	On Screen Display	Subpicture_Color13
3B8	R/W	On Screen Display	Subpicture_Color14
3BC	R/W	On Screen Display	Subpicture_Color15
3D0	R/W	On Screen Display	On_Screen_Display_Control
3D4	R/W	On Screen Display	OSD_Offset
3D8	R/W	On Screen Display	OSD_Scale_Factor
3DC	R/W	On Screen Display	OSD Debugging

7.4 PCI Registers

7.4.1 VIP_PlugnPlay registers (000 - 008)

These read-only registers are the same as PCI configuration space registers 000-008.

7.4.2 Host_Control register (010)

Software_Chip_Reset Register

Bit	Name	Function	Default
0	Soft_Reset	1 = reset the entire chip	0
1	External_Reset	1 = forces reset output pin low	0
2	Vid_Clockout_Disable	1 = turns off the 27Mhz video clock output pin	0
3	Sleep	1 = puts LS220 in sleep mode	1
4	Host_DMA_EnableH		0
5	Asynchronous_DMA		0
7:6	FIFO_Sector_Select	00=Video, 01=Audio, 10=Subpicture, 11=none	0
10:8	FIFO_Burst_Thresh(2:0)	Number of words before FIFO is bursted into DRAM	2
14:11	FIFO_Fullness	Number of words in the AVV FIFO (read only)	

7.4.3 Interrupt registers

Interrupts are controlled and monitored by the following registers: Interrupt_Status, Interrupt_Mask, Interrupt_Clear and Interrupt_Set. All four registers have the following identical bit assignments:

PCI Interrupt Register Bit Assignments

Bit	Name	Function
0	MPEG_Error	Decode error detected in MPEG Video Decoder
1	VID_Int	Video Processor end of active display area reached
2	DSP_Int	DSP Interrupt
3	MPEG_Vid_Input_FF	MPEG Video Decoder input FIFO fullness is above threshold
4	MPEG_Decode_Req	MPEG Video Decoder decode request **********
5	Y_FIFO_Overflow	Video processor Y FIFO overflowed
6	V_FIFO_Overflow	Video processor V FIFO overflowed
7	U_FIFO_Overflow	Video processor U FIFO overflowed
8	Video_Line_Reached	Video processor programmable line reached

9	PCM_Int	PCM output FIFO count reached
10	Internal_DSP_Int	Internal interrput generated by sync block to DSP block
11	I2C_Xfer_Done	I2C input or output transfer completed without errors
12	I2C_Xfer_Error	I2C input or output transfer was was terminated due to error
13	SPDIF_Input_Ready	S/PDIF input parallel word is ready for reading
14	PCI Master (reserved)	

Interrupt_Status register (014)
Read-Only register holds current interrupt states (1=interrupt present). The interrupt status bits are set by pulses on interrupt signals generated by internal blocks, and by writes to the Interrupt_Set register. The status bits are cleared by writes to the Interrupt_Clear register. If any bits are set then the INTA_L pin will be driven.

Interrupt_Clear (018) and Interrupt_Set (01C) registers
The Interrupt_Set register is a Write-Only register used to force interrupts for diagnostic purposes. Writing ones to bits of this register causes the corresponding bits of the Interrupt_Status register to be set. The Interrupt_Clear register is used to clear Interrupt_Status bits. Writing ones to bits of this register causes the corresponding bits of the Interrupt_Status register to be cleared. Internal interrupt set events take priority over interrupt clears. Both the Interrupt_Set and Interrupt_Clear are write-once registers.

Interrupt_Mask register (020)
Read-Write register determines which bits of the Interrupt_Status register may be set (1=use, 0=ignore).

7.4.4 AVV_FIFO_Pointer Register (030)

Sets the starting write address for tha AVV FIFO.

AVV_FIFO_Pointer Register (030)

Bit	Name	Function
21:0	Write_Address(22:0)	Byte address in DRAM (first byte is at 20_0000h)

7.4.5 AVV_FIFO_Write_Data Register (034)

Sets the starting write address for tha AVV FIFO.

AVV_FIFO_Write_Data Register (034)

Bit	Name	Function
31:0	Write_Data(31:0)	Data to be written into DRAM location pointed to by

		AVV_FIFO_Pointer

7.5 DRAM Controller Registers

DRAM_Controller_Setup Register (080)
DRAM_Controller_Debug Register (084)

The DRAM_Controller_Setup register is used to control DRAM refresh modes and arbitration priority. The DRAM Controller allows eight internal modules to access the DRAM by arbitrating between them using a programmable priority. The Video processor always has highest priority. Refresh has the next highest priority. The remaining modules are assigned a priority between 1 and 7 (1 is highest priority). Programming a priority of zero disables that module from requesting the DRAM. A second tier of priorities is defined to facilitate changing priorities. The register includes a bit for each of the seven arbitrating modules which, when set, drops the module's priority into the lower tier.

The DRC_60nsH_50nsL selects operation with 50 and 60 nsec DRAMs by controlling the RAS time before CAS. The Refresh_Rate bits set the number of 66mhz clocks between refresh requests. Each request refreshes four columns.

DRAM_Controller_Setup Register (080)

Bit	Name	Function	Default
2:0	MPEG_Decoder_Priority	DRAM Arbitration priority (range 1-7)	110
5:3	Pixel_Gen_Priority	" "	111
8:6	Audio_DSP_Priority	" "	100
11:9	PCM_Audio_Priority	" "	001
14:12	PCI_Priority	" "	011
17:15	Sync_Priority	" "	101
20:18	On_Scrn_Disp_Priority	" "	010
21	Lower_MPEG_Decoder	1= Drop priority into lower group.	0
22	Lower_Pixel_Gen	" "	0
23	Lower_Audio_DSP	" "	0
24	Lower_PCM_Audio	" "	0
25	Lower_PCI	" "	0
26	Lower_Sync	" "	0
27	Lower_On_Scrn_Disp	" "	0
28	DRC_60nsH_50nsL	0=50ns DRAMs, 1=60ns DRAMs	1
30:29	Refresh_Rate	00=53,247 clocks, 01 = 4,095 clocks, 10 = 3,327 clocks, 11= 255 clocks	00

The RAM_Controller_Debug register is used for debugging DSP code. This register accepts a DRAM byte address and will case an interrupt to the DSP when a write is done to that DRAM address.

DRAM_Controller_Debug Register (088)

Bit	Name	Function	Default
21:0	DRAM_Debug_Address(21:0)	Address to monitor for writes	0

7.6 Audio DSP Registers

7.6.1 DSP Run Control Registers

DSP_Boot_Code_Start_Address (R/W) (100)
DSP_Run_Enable (R/W) (104)

The boot code is the location in DRAM of the start of the DSP firmware. The address given points to a 16-bit boundary in DRAM (address bit 0 is assumed zero). The not used bits in this register should be masked off when reading.

DSP_Boot_Code_Start_Address register

Bit	Name	Function
15:0	Boot_Address	DRAM boot code start address (byte address 16:1)
31:16	not used	

DSP_Run is a bit, cleared upon chip reset, which must be set for the DSP to start executing it's firmware. This bit should not be set until a valid boot code start address is programmed. The DSP_Run bit is written in bit 0, and read back in bit 31. The remaining not used bits should be masked off when reading.

DSP_Run_Enable register

Bit	Name	Function
0	DSP_Run (write)	1 = enable DSP operations (write only)
30:1	not used	
31	DSP_Run (read)	1 = DSP operations enabled (read only)

7.6.2 DSP Internal Register port

The DSP has an internal register port that connects to the Sync block and PCM output block. This register set provides 5 addresses which may be written by the DSP to write the Audio_PTS, STC_Run_H and PCM_Audio_Run_H fields to the sync block, and write the top of DRAM FIFO address to the PCM block (via the sync block). A single read register (decoded over all addresses) provides status to the DSP. The current PCM Audio DRAM FIFO read pointer goes to the DSP for FIFO management. A bit indicates whether the host controls writing the PCM top of DRAM pointer value. A bit indicates whether an Audio PTS has been written by the DSP which has not yet been used to update the STC. Four other bits are used to manage the dual-source control over the STC and PCM output.

DSP Internal Register Port registers

Address	Bits	Mode	Function
0	0	Write	Audio_PTS(0)
1	15:0	Write	Audio_PTS(16:1)

2	15:0	Write	Audio_PTS(32:17) and signals a new PTS has been loaded
3	0	Write	STC_Run_H
3	1	Write	PCM_Audio_Run_H
4	9:0	Write	pointer to top of PCM DRAM FIFO (byte address 12:3)
ALL	9:0	Read	Current PCM DRAM FIFO read address (byte address 12:3)
ALL	10	Read	1 = PCM addresses are under PC control
ALL	11	Read	PCM_RunH_StopL (current PCM state programmed from the PC)
ALL	12	Read	DSP_PCM_Kill_Enable (1 = DSP allowed to stop PCM Output)
ALL	13	Read	STC_RunH_StopL (current STC state programmed from the PC)
ALL	14	Read	DSP_STC_Kill_Enable (1 = DSP allowed to stop STC)
ALL	15	Read	Audio_PTS_Pending (1=valid PTS pending)

7.7 MPEG Video Decoder Registers

The MPEG Video decoder reads host-written video streams from a DRAM FIFO. A set of registers is used for FIFO management with the host. Additional registers are used for general control and status of the decoder.

FIFO management is done by a counter in the MPEG Video decoder. The host writes a register with the number of 64-byte blocks it has added to the FIFO, and the MPEG Decoder adds this number to its internal counter. The MPEG Decoder decrements the counter when it reads a block from the FIFO. The host can program the MPEG decoder to interrupt when its block count gets below a programmable threshold. The block counter is also readable by the host.

7.7.1 Video Decoder Control and Status registers

MPEG_Video_Control Register Bit Assignments (180)
MPEG_Video_Setup Register Bit Assignments (184)

The MPEG_Video_Decoder_Control register bits are used to reset, start and run the MPEG Video Decoder. The error control bits are used to program the error response of the Decoder.

MPEG_Video_Control Register Bit Assignments (180)
TBS

MPEG_Video_Setup Register Bit Assignments (184)
TBS

7.7.2

7.7.3 Video Input FIFO Location registers

MPEG_Video_FIFO_Start_Address (R/W) (188)
MPEG_Video_FIFO_End_Address (R/W) (18C)
MPEG_Video_FIFO_Current_Address (RO) (190)
MPEG_Video_FIFO_Force_Address (WO) (194)
MPEG_Video_FIFO_Add_Bytes (WO) (198)
MPEG_Video_FIFO_Curr_Bytes (RO) (19C)
MPEG_Video_FIFO_Interrupt_Bytes (R/W) (1A0)
MPEG_Video_FIFO_Total_Bytes (RO) (1A4)

The DRAM range occupied by the Video input FIFO is specified by the MPEG_Video_FIFO_Start_Address and MPEG_Video_FIFO_End_Address registers, which give the locations of the start of the first and last 64-byte blocks of the FIFO. The current DRAM read position is provided in the read-only MPEG_Video_FIFO_Current_Address register. The addresses specified in these registers are aligned on 64-byte boundaries, so address bits 5:0 are assumed zero. Since the upper limit of the DRAM is typically at 256K, the upper three bits will normally be zero for all fields in these registers.

The read-only MPEG_Video_FIFO_Curr_Bytes register reports the MPEG Decoder's internal block counter. The MPEG_Video_FIFO_Add_Blocks register is a write-only (write once) register the host uses to add a number to the MPEG Decoder's internal block counter. The Interrupt_Bytes register programs a number of bytes on which the decoder will generate an interrupt. The decoder will interrupt when it reads the FIFO down to this number of bytes, or when it reads the FIFO down to zero bytes. The MPEG_Video_FIFO_Total_Bytes register is a read-only register reporting the total number of bytes read by the Decoder since reset, and is used mainly for sync.

MPEG_Video_FIFO_Start_Address Register (188)

Bit	Name	Function	Default
5:0	Zeros	Lower part of start address	.
21:6	Address(21:6)	Upper part of Start Address of Video FIFO (in bytes)	.

MPEG_Video_FIFO_End_Address Register (18C)

Bit	Name	Function	Default
5:0	Zeros	Lower part of end address	.
21:6	Address(21:6)	Upper part of End Address+1 of Video FIFO (in bytes)	.

MPEG_Video_FIFO_Current_Address Register (190)

Bit	Name	Function	Default
5:0	Zeros	Lower part of current address	.
21:6	Address(21:6)	Upper part of Current Address in Video FIFO (in bytes, read only)	.

MPEG_Video_FIFO_Force_Address Register (194)

Bit	Name	Function	Default
5:0	Zeros	Lower part of force address	.
21:6	Address(21:6)	Upper part of Force Current Address in Video FIFO to this value (in bytes, write only)	.

MPEG_Video_FIFO_Add_Bytes Register (198)

Bit	Name	Function	Default
5:0	Zeros	Lower part of add bytes	.
17:6	Count(17:6)	Upper part of number of bytes to add	.

MPEG_Video_FIFO_Current_Bytes Register (19C)

Bit	Name	Function	Default
5:0	Zeros	Lower part of current bytes	.
17:6	Count(17:6)	Upper part of number current bytes (read only)	.

MPEG_Video_FIFO_Interrupt_Bytes Register (1A0)

Bit	Name	Function	Default
5:0	Zeros	Lower part of interrupt bytes	.
17:6	Count(17:6)	Upper part of byte count at which to interrupt (read only)	.

MPEG_Video_FIFO_Total_Bytes Register (1A4)

Bit	Name	Function	Default
23:0	Count(23:0)	Total number of bytes read since reset	.

7.7.4 Video Decoder Setup and Error Status registers

MPEG_Video_Error_Status (RO) (1A8)
Macroblock_Width (R/W) (1AC)
Macroblock_Height (R/W) (1B0)

MPEG_Video_Error_Status Register (1A8)

TBS

Macroblock_Width Register (1AC)

TBS

Macroblock_Height Register Bit Assignments (1B0)

TBS

7.7.5 Video Decoder Control and Status registers

MPEG_Video_Debug_1 (RO) (1B4)
MPEG_Video_Debug_2 (RO) (1B8)

MPEG_Video_Debug_1 Register (1B4)
TBS

MPEG_Video_Debug_2 Register (1B8)
TBS

7.8 Synchronization Control Registers

Control registers:
Audio control registers (200):

bits	name	read-write	function	default
0	Host_STC_RunH_StopL	RW	0=STC stop 1=STC run	0
1	Host_PCM_RunH_StopL	RW	0=PCM stop, 1=PCM run	0
2	STC_Clear_H	WO	Clear STC to 0, (pulse)	0
3	Host_Audio_Cntl_H	RW	1=enable host controls audio	0
4	Audio_Sync_Run_H	RW	1= enable audio sync	0
5	Host_Start_PCM_H	WO	1=Host initiate PCM running	0
6	No_Audio_Master_H	RW	1=no audio 0=audio present	0
7	Reset_SP_FIFO_H	WO	pulse to reset SP FIFO	0

video control registers (204)

bits	name	read-write	function	default
0	Video_RunH_StopL	RW	1=enable video control	0
1	Host_Vid_Decode_H	RW	1=enable host control video decode	0
2	Host_Vout_Control_H	RW	1=enable host control video out	0
3	Host_Vid_FIFO_Enable_H	RW	1=enable video PTS FIFO	0
4	Lock_Fields_H	RW	1=enable lock fields function	0
5	Host_VPTS_Control_H	RW	1=host writes Video PTS	0
6	Host_Continue_H	WO	send continue pulse to decoder	0
7	Host_Skip_H	WO	send skip pulse to decoder	0
8	Host_Decode_Request_H	WO	send internal decode request	0
9	Host_Vout_Request_H	WO	send internal video output	0
10	Host_TopField_Valid	RW	topfield is valid	0
11	Host_BottomField_Valid	RW	bottomfield is valid	0
13:12	Host_Decode_Base	RW	one of the three frame buffer	00
14	Host_TopField_First_H	RW	display top field first	0

15	Host_RepeatFirstField_H	RW	repeat first field	0
16	Host_ProgressiveFrame_H	RW	picture is progressive	0
17	Host_ProgressiveSeq_H	RW	all pictures in GOP is progressive	0
18	Host_Second_Disp_H	RW	displaying second field	0
19	Reset_Vid_FIFO_H	WO	reset video PTS FIFO	0
20	Host_Int_With_Line_H	RW	host interrupt with line reached	0

video sync status registers (208)

bits	name	read-write	function	default
0	MPGSnc_Req_H	RO	decode request from Mpeg decoder	0
2:1	MdcSnc_Dest_Base	RO	Decode destination base	0
3	MdcSnc_Back_Base	RO	it is always conjugate with Forw_Base	1
4	MdcSnc_Forw_Base	RO	Base for I/P pictures	0
5	Start_Video_Decode_H	RO	Decoding first picture	0
6	Second_Field_H	RO	Second field requests/decodes	0
7	Decode_Request_H	RO	decoder is decoding	0
8	Vid_LineReached_H	RO	video display reaches the line	0
9	Vid_DTS_Valid_H	RO	video DTS is valid	0
10	Vid_PTS_Valid_H	RO	STC reached value of video PTS	0
12:11	Mpg_PictureType	RO	00--not defined, 01--I-type 10-P-type, 11--B-type	00
14:13	Mpg_PictureStructure	RO	bit 0 -- top field valid bit 1 -- bottom valid	00
24:15	Mpg_TemporalReference	RO	temporal reference in the GOP	0
25	Mpg_ProgressiveSeqH	RO	all the picture is progressive	0
26	Mpg_ProgressiveFrame_H	RO	the picture is progressive	0
27	Mpg_RepeatFirstField_H	RO	useful when progressive frame	0
28	Mpg_TopFieldFirst_H	RO	1= display top field first 0= display bottom field first	0
29	Mpg_GroupStart_H	RO	First picture of the GOP	0
30	Mpg_ClosedGOP_H	RO	The first B-pciture is	0

			reconstructed from I-picture only	
31	Mpg_BrokenLink	RO	valid when closedGOP=0	0

Wait line (20C)

bits	name	read-write	function	default
6:0	Vid_WaitLine	RW	wait for the line to start decode	0
14:8	Vid_Current_Line	RO	Current line displayed	0
22:16	Vid_SecondDispLine	RW	used in control displaying field pictures.	0

Frame Period (210)

bits	name	read-write	function	default
13:0	Mpg_Frame_Per	RW	displaying time for two fields	0
29:16	Mpg_DTS_Sub	RW	difference between PTS and DTS	0

STC Interval (214)

bits	name	read-write	function	default
15:0	STC_Interval	RW	ratio of 90 Khz vs. system clock	2DD

System Time Clock(218)

bits	name	read-write	function	default
31:0	STC[31:0]	RW	STC value	0

Top bits (21C)

bits	name	read-write	function	default

The following registers(from 220--23C and 24C) have lower bits of zeros
Video PTS FIFO start address(220)

bits	name	read-write	function	default
21:3	Video_FIFO_Start_Adr	RW	start location of video PTS FIFO	0

Video PTS FIFO end address(224)

bits	name	read-write	function	default
21:3	Video_FIFO_End_Adr	RW	End location (inclusive) of video PTS FIFO	0

Video PTS FIFO write address(228)

bits	name	read-write	function	default
21:3	Video_FIFO_Write_Adr	RW	write pointer of video PTS FIFO	0

Video PTS FIFO read address(22C)

bits	name	read-write	function	default
21:2	Video_FIFO_Read_Adr	RO	read pointer of video PTS FIFO	0

Subpicture PTS FIFO start address(230)

bits	name	read-write	function	default
21:3	SP_FIFO_Start_Adr	RW	start location of SP PTS FIFO	0

Subpicture PTS FIFO end address(234)

bits	name	read-write	function	default
21:3	SP_FIFO_End_Adr	RW	End location (inclusive) of sub-picture PTS FIFO	0

Subpicture PTS FIFO write address(238)

bits	name	read-write	function	default
21:3	SP_FIFO_Write_Adr	RW	write pointer of SP PTS FIFO	0

Subpicture PTS FIFO read address(23C)

bits	name	read-write	function	default
21:2	SP_FIFO_Read_Adr	RO	read pointer of SP PTS FIFO	0

The PTSs associated with DVD(sub-pictures, highlights and bottons) are all 32-bit wide instead of 33 bits as defined in the MPEG-2 specification. In the comparison, the MSBs are treated as zero.

Highlight Start PTS(240)

bits	name	read-	function	default

		write		
31:0	HLI_Start_PTS	RW	highlight start PTS	0

Highlight End PTS(244)

bits	name	read-write	function	default
31:0	HLI_End_PTS	RW	highlight End PTS	0

Button End PTS(248)

bits	name	read-write	function	default
31:0	BTN_End_PTS	RW	Botton End PTS	0

Highlight Control Information Address(24C)

bits	name	read-write	function	default
21:1	HLI_DCI_Address	RW	Highlight Control Information Location in DRAM	0

Video PTS(250)

bits	name	read-write	function	default
31:0	Video_PTS	RW	Video PTS register	0

Audio PTS(254)

bits	name	read-write	function	default
31:0	Audio_PTS	RW	Audio PTS	0

Subpicture PTS(258)

bits	name	read-write	function	default
31:0	SP_PTS	RO	Subpicture PTS	0

Audio Time(25C)

bits	name	read-write	function	default
31:0	Audio_Time	RO	Audio Time used by DSP as reference to STC	0

Video Debug registers (260)

bits	name	read-write	function	default

31:0	Video_Debug	RO	Video debug	0
1:0	SncVid_Next_Frame	RO	video display buffer	0
2	SncVid_Src_ProgH_InterL	RO	video source progressive/interlaced	0
3	SncVid_Src_EvenH_OddL	RO	video source Even/Odd	0
5:4	Internal_Vid_Valid	RO	internal video valid	0
7:6	Internal_Vid_Base	RO	internal video base	0
8	Internal_ProgH_InterL	RO	internal progressive/interlace	1
9	Internal_TopFirst_H	RO	internal top field first	0
10	New_Output_Pending	RO	wait for new vsync to ouput internal video parameters	0
11	Vid_PTS_Good_H	RO	video PTS has been updated	0
12	Vid_PTS_Valid_H	RO	video PTS is valid	0
13	Vid_FIFO_Valid_H	RO	video FIFO contains valid data	0
14	Vid_FIFO_Request_H	RO	Requesting data to Dram	0
15	Second_Field_H	RO	Second field of a field picture	0
19:16	IP_Decode_Param	RO	Decode parameter for I/P	0
21:20	IP_Decode_Valid	RO		
23:22	IP_Decode_Base	RO		
27:24	Decode_Param	RO		
29:28	Decode_Valid	RO		
31:30	Decode_Base	RO		

OSD/DRC/VPTS debug registers (264)

bits	name	read-write	function	default
0	Internal_SP_Active_H	RO	internal subpicture active	0
1	Internal_HLI_Active_H	RO	internal highlight active	0
2	Internal_BTN_Active_H	RO	internal button active	0
4:3	SP_FIFO_State	RO		
6:5	Vid_FIFO_State	RO		
7	DRAM_Wd1H_Wd0L	RO	0=first dW, 1=second dW	0
8	DRAM_Active_H	RO	1=dram is reading data to snc	0
11	Vid_FIFO_Pending	RO		0
12	IPVid_FIFO_Pending	RO		0

DSP Interrupt Status(270)

bits	name	read-write	function	default

10:0	DSP_Int_Status	RO	DSP Interrupt Status	0
0	MpgReq_Int_H	RO	Mpeg decode interrupt (**)	0
1	VidLine_Int_H	RO	Video Line reached	0
2	VidDTS_Int_H	RO	DTS valid interrupt	0
3	VidPTS_Int_H	RO	Video PTS valid interrupt	0
4	AudPTS_Int_H	RO	Audio PTS valid interrupt	0
5	MpgErr_Int_H	RO	MPEG Decode Error Interrupt	0
6	AudPCM_Int_H	RO	First audio sample interrupt	0
7	ByteCnt_Int_H	RO	Byte count reached interrupt	0
8	UserData_Int_H	RO	User data request interrupt	0
9	VidDisp_Int_H	RO	video display interrupt	0
10	VidSkip_Int_H	RO	Mpeg skip request interrupt	0
11	I2C_Done_DspInt_H	RO		0
12	I2C_Error_DspInt_H	RO		0
13	SPDIF_In_DspInt_H	RO		0
14	FrameNum_Int_H	RO	Frame Number interrupt	

DSP Interrupt control(274)

bits	name	read-write	function	default
14:0	DSP_Int_Msk	RW	DSP interrupt mask (1=enable)	0
19:16	DSP_Int_Param	RW	used to program the MPEG decode request	0000
15	DSP_Int_EnH	RW	global on/off interrupt function	0

DSP Interrupt Set(278)

bits	name	read-write	function	default
14:0	DSP_Int_Set	RW	Set DSP Interrupt (Diag purpose)	0

DSP Interrupt Clear(27C)

bits	name	read-write	function	default
14:0	DSP_Int_Clear	RW	Clear DSP Interrupt	0

7.9 PCM Audio Output Registers

The PCM Audio block reads PCM audio data from the DRAM Audio Output FIFO and outputs up to 6 serial output audio channels, using an external audio clock or an internally generated programmable frequency clock. A S/PDIF interface can be used to output PCM output FIFO through a S/PDIF output shared by the Surround output pin.

7.9.1 PCM_Frequency_Generator_Control register (280)

The PCM_Frequncy_Generator_Control register, controls running and frequency selection of the internal audio frequency generator. One of 6 programmable frequencies are generated by dividing the main 66 Mhz clock. The Fgen_Run_Halt bit controls the run/stop state of the generator, without resetting the internal counters. If the Fgen_Pulse_at_Start bit is set, a transition from 0 to 1 on the Fgen_Run_Halt bit will cause an initial audio clock pulse to be generated. The Fgen_Clear bit forces a reset on the internal counters, and overrides the Fgen_Run_Halt bit.

PCM_Frequncy_Generator_Control Register (280)

Bit	Name	Function	Default
0	Fgen_Run_Halt	1= Run frequency generator, 0= stop	0
1	Fgen_Clear	1 = Clear frequency generator	0
2	Fgen_Pulse_at_Start	1 = Send a pulse upon new Run condtion	0
5:3	Fgen_Frequency_Select	000 = 64.0 Khz , 001 = 32.0 KHz 010 = 88.2 Khz , 011 = 44.1 KHz 100 = 96.0 Khz , 101 = 48.0 KHz	0

7.9.2 PCM_Output_Control register (284)

The PCM_Output_Control register, sets the operating mode of the PCM output and selects between the external audio clock pin and the internal frequency generator as the source of PCM output timing.

The register provides two methods of selecting the output mode (mono/stereo/4channel/6channel), depending on whether the PCM data in DRAM is packed or unpacked. For unpacked data, the PCM_Stereo_Mode field should be set to 11(6 channel), and the Enable_Front, Enable_Rear and Enable_Surround bits should beee set according to the source data in DRAM. For mono, the DSP must write the same data to both left and right channels. For packed data, the PCM_Stereo_Mode bits should be used to select the operating mode.

PCM_Output_Control Register (284)

Bit	Name	Function	Default
0	PCM_Run_Stop	1= Enables PCM output, 0= stop	0
1	Sample1H_Sample2L	0= two 16-bit samples per 32-bit DRAM word 1= one n-bit sample per 32-bit DRAM word	0
2	PackedH_UnpackedL	1= two channels in DRAM (packed) 0= six channels in DRAM (unpacked)	0
4:3	Sample_Size	00=16-bit, 01=18-bit, 10=20 bit, 11= 24 bit	0
5	External_Clock_H	0 = Use internal frequency generator 1 = Use external Audio clock input	0
6	External_24bitH_16bitL	Sets number of clocks per 16 bit output in external clock mode. 0 = 16 clocks, 1 = 24 clocks	0
7	LRCK_RightLow_H	Sets polarity of LRCK output pin 0 = Low first, high second 1 = High first, low second	0
8	LRCK_Disable	1=disables LRCK output when PCM is on	0
9	Reverse_Sample	1=outputs DRAM(31:16) first, DRAM(15:0) second 0= outputs DRAM(15:0) first, DRAM(31:16) second	0
10	Clockout_Polarity	1=data clocks out on falling edge 0=data clocks out on rising edge	0
11	Enable_Front	1=Front output enabled, 0=front shut off	0
12	Enable_Rear	1=Rear output enabled, 0=rear shut off	0
13	Enable_Surround	1=surround output enabled, 0= shut off	0
15:14	Front_Source_Select(1:0)	Selects what channels go out Front output: 00=Front, 01=Rear, 10=Surround	0
16	SPDIF_Enable	1=S/PDIF output from Surround output pin	0

18:17	SPDIF_Select(1:0)	Selects what channels go out S/DPIF output: 00=Front, 01=Rear, 10=Surround	0
19	SPDIF_Valid_Bit	Sets the value of the validity bit in the S/PDIF output	0

7.9.3 PCM_FIFO_Address registers

PCM_FIFO_Base_Start_Address (288)
PCM_FIFO_Base_End_Address (28C)
PCM_FIFO_Base_Interrupt_Address (290)
PCM_FIFO_Base_Current_Address (294) (RO)

These registers set the location of the PCM output FIFO in DRAM, allow the host to read the current address the PCM block is reading in the FIFO, and allows th host to program an interrput location in the PCM FIFO. The PCM block will generate an interrput when it reads this location.

The FIFO start address should start on a 64-byte boundary. The FIFO end address specifies the location of the first byte after the PCM FIFO. The offsets from the Start_Address given by the End_Address and Interrupt_Address should be in multiples of 24 bytes.

PCM_FIFO_Start_Address Register (288)

Bit	Name	Function	Default
5:0	zero		0
21:6	FIFO_Start_Address	PMC DRAM top of FIFO byte address (21:0)	0

PCM_FIFO_End_Address Register (28C)

Bit	Name	Function	Default
2:0	zero		0
21:3	FIFO_End_Address	PMC DRAM end of FIFO byte address (21:0)	0

PCM_FIFO_Interrupt_Address Register (290)

Bit	Name	Function	Default
2:0	zero		0
21:3	FIFO_Interrupt_Address	PMC DRAM interrupt byte address (21:0)	0

PCM_FIFO_Current_Address Register (294)

Bit	Name	Function	Default
2:0	zero		0
21:3	FIFO_Current_Address	PMC DRAM current byte address (21:0) (R/O)	0

7.9.4 SPDIF_Channel_Status register (298)

This register is used to program the 32-bit value to be output in the first 4 bytes of the 24-byte S/PDIF channel status word. The remaining 20 bytes will be output as zeros (commercial standard). The same channel status word will be output for both left and right sub-channels.

SPDIF_Channel_Status Register (298)

Bit	Name	Function	Default
31:0	SPDIF_Channel_Status		0

7.10 Video Processor Registers

The Video Processor reads pixel data from DRAM frame buffers and a variety of scaling operations to display pictures on interlaced TV's and non-interlaced monitors.

7.10.1 Video_Processor_Control register (300)

The Video_Processor_Control register controls the modes of the Video Processor. The display format can be set to interlaced (TV) or non-interlaced (monitor). Horizontal and vertical sync can come from the external HSYNC/VSYNC input pins, or can be generated internally using dot and line counters. Output signal polarities, tri-state enables and delays can be set. This register controls both the direct video output, and the VPM (VIP/AMC/VMI/LPB) output.

Video_Processor_Control Register Bit Assignments

Bit	Name	Function	Default
0	Enable_Video_Processor	1 = Enables Video Processor 0 = Syncs enabled but DRAM reads disabled and display is blanked	0
1	TVL_MonitorH	0 = Interlaced TV, 1 = Non-interlaced monitor	0
2	VPM_NonBlank_Polarity	0 = Generate NonBlank_Area_H 1 = Generate Blank_Area_H	0
4:3	Output_Mode(1:0)	00 = 656 YUV 01 = SAV/EAV 656 10 = Y 11 = Cb/Cr	00
6:5	SAV_EAV_Mode(1:0)	00 = use leasing edge of hsync 01 = use start/end X 10 = use NonBlank_Area_H 11 = use leading/trailing edge of hsync	00
8:7	YUV_Start(1:0)	00 = start with Cb 01 = start with Y0 10 = start with Cr 11 = start with Y1	0
9	Clock_Polarity	Active polarity of Video clock output pin	1
10	Hsync_Polarity	Active polarity of Video Hsync I/O pin	0
11	Vsync_Polarity	Active polarity of Video Vsync I/O pin	0
13:12	VPM_Output_Mode(1:0)	00 = 656 YUV 01 = SAV/EAV 656 10 = Y 11 = Cb/Cr	0
15:14	VPM_SAV_EAV_Mode(1:0)	00 = use leasing edge of hsync 01 = use start/end X 10 = use NonBlank_Area_H 11 = use leading/trailing edge of hsync	0
17:16	VPM_YUV_Start(1:0)	00 = start with Cb 01 = start with Y0 10 = start with Cr 11 = start with Y1	0
18	VPM_Clock_Polarity	Active polarity of VPM clock output pin	1
19	VPM_Hsync_Polarity	Active polarity of VPM Hsync output pin	0
20	VPM_Vsync_Polarity	Active polarity of VPM Vsync output pin	0
22:21	VPM_Sync_Delay(1:0)		0
24:23	VPM_Blank_Delay(1:0)		0

25	VPM_Clock_En	1=VPM Clock enabled, 0=tri-state	0
26	VPM_Control1_En	1=VMI control outputs enabled, 0=tri-state	0
27	VPM_Control2_En	1=VMI/LPB control outputs enabled, 0=tri-state	0
28	VPM_Data_En	1=VPM data output enabled, 0=tri-state	0
29	VPM_LpbH_VmiL	1=VPMin LBP mode, 0=VPM in VMI mode	0
31:30	Vid_Dly_8Bit_Start	number of Clock27's to delay VYUY... sequence	0

7.10.2 Video_DRAM_Line_Length register (304)

This register determines how many pixels the Video Processor is to read from DRAM for each line. Controlling the DRAM read size relieves the Video Processor of calculating it after scaling, ensures enough data is read for a line, and conseves DRAM bandwidth by preventing extra data from being read. The line length is specified as two fields: a number of 80-byte (y) blocks to be read, then a remainder (in 4 byte increments) to be read after the last 80-byte block. A CCIR clipper can be turned on which ensures the video YUV levels are clipped to within CCIR levels. Two special horizontal filter modes select 1/2X or 2X horizontal.

Video_DRAM_Line_Length Register Bit Assignments

Bit	Name	Function	Default
3:0	DRAM_Block_Count(3:0)	Number of 80-pixel (y) blocks to read	1111
8:4	Last_Read_Size(4:0)	Number of 4-pixel reads after last block	00000
19:16	Reserved		
24:20	Reserved		
25	CCIR_Clip	1 = Enables CCIR level clipper	0
26	Horiz_Half_Filter	1 = Enable Horizontal 1/2X filter	0
27	Horiz_TwoX_Filter	1 = Enable Horizontal 2X filter	0
28	IntSyncL_ExtSyncH	0 = Genterate VSYNC/HSYNC outputs 1 = Use VSYNC/HSYCN input pins	0
29	Gamma_Corr_EnH	1= Enable Gamma correction on VPM output	0

7.10.3 Display_ActiveX register (308)

This register determines the horizontal position the display will be turned on, using a 27MHz dot clock counter. The dot clock counter resets at the beginning of horizontal sync. The display turns on when the counter reaches the StartX value, and turns off when the EndX value is reached.

Display_ActiveX Register Bit Assignments

Bit	Name	Function	Default
10:0	StartX(10:0)	27MHz clock count at which display turns on	0
15:11	not used		
27:16	EndX(10:0)	27MHz clock count at which display turns off	0

7.10.4 Display_ActiveY register (30C)

This register determines the horizontal line numbers the display will be turned on. Line numbers count starting from zero. For interlaced displays, lines of both fields are counted as one line, and

each unit of StartY and EndY represents two lines on the display. In this case, the display turns on in the even field and turns off in the odd field.

Display_ActiveY Register Bit Assignments

Bit	Name	Function	Default
9:0	StartY(9:0)	Line number at which display turns on	0
15:10	not used		
25:16	EndY(9:0)	Line number at which display turns off	0

7.10.5 Blank_Color register (310)

This register determines the YUV color output by the Video Processor during blank periods.

Blank_Color Register Bit Assignments

Bit	Name	Function	Default
7:0	BlankV(7:0)	Blank V value	0
15:8	BlankU(7:0)	Blank U value	0
23:16	BlankY(7:0)	Blank Y value	0

7.10.6 Internal_Sync_Count registers

Internal_HSync_Count (314)
Internal_VSync_Count (318)

These registers determine the counts for generating internal horizontal and vertical sync. Vertical sync is generated by a counter which counts horizontal syncs. For both interlaced and non-interlaced modes, the Vertical_Count should be set to the number of lines in the total frame. For both sync's, a register value controls the length of the sync pulse, starting from the beginning of the horizontal or vertical period.

Internal_HSync_Count Bit Assignments

Bit	Name	Function	Default
10:0	Horiz_Sync_Count(10:0)	Number of 27MHz clocks between HSYNC's	0
15:11	not used		0
23:16	Horiz_Sync_Pulse(7:0)	Number of 27MHz clocks long for HSYNC	0

Internal_VSync_Count Bit Assignments

Bit	Name	Function	Default
9:0	Vert_Sync_Count(9:0)	Number of lines in frame (must be odd if in interlaced mode)	0
15:10	not used		0
28:16	Vert_Sync_Pulse(12:0)	Number of dot clocks long for VSYNC	0

7.10.7 Source_XY_Offset registers

Horizontal_Y_Offset (31C)
Horizontal_UV_Offset (320)
Vertical_Offset (324)

These registers determine the the offsets in the DRAM frame buffer from which data is displayed. In the horizontal (X) direction, this allows panning the picture to 1/16 byte accuracy. Separate offsets control Y and UV horizontal offset. One byte of offset corresponds to one pixel for Y, and two pixels for UV. The Vertical_Offset register defines the starting line and fractional part of a line from the top of the frame buffer where output should start, and is used as the starting point for the vertical scaler.

Horizontal_Y_Offset and Horizontal_UV_Offset Registers

Bit	Name	Function	Default

3:0	Fract_XOffset(3:0)	Fractional part of X offset, in 1/16 bytes	0
13:4	Integer_XOffset(9:0)	Integer part of X offset, in bytes	0

Vertical_Offset Register

Bit	Name	Function	Default
3:0	Fract_YOffset(3:0)	Fractional part of Y offset, in lines	0
12:4	Integer_YOffseta(8:0)	Integer part of Y offset, in lines	0

7.10.8 Video Scaling registers

Video_Horizontal_Scale (328)
Video_Vertical_Scale (32C)

The Video_Horizontal_Scale register controls zoom-in of the picture in the X direction. The register represents a fractional number which is used as the denominator of the scale factor. For example, setting the register to 0800h gives a value of 0.5, which corresponds to a 2X zoom. The maximum value allowed is 1.0 (full size).

Video_Horizontal_Scale Register

Bit	Name	Function	Default
11:0	Frac_Horizontal_Scale	Fractional denominator of scale factor	
12	Int_Horizontal_Scale	Integer denominator of scale factor. If set to 1, fractional part must be zeros	

The Video_Vertical_Scale register controls zoom-in and zoom-out of the picture in the Y direction. The register represents a real number which is used as the denominator of the scale factor, with two bits to the left of the decimal point. The maximum value allowed is 2.0 (1/2 zoom).

Video_Vertical_Scale Register

Bit	Name	Function	Default
9:0	Frac_Vertical_Scale	Fractional denominator of scale factor	
11:10	Int_Vertical_Scale	Integer denominator of scale factor, maximum value is 2. If set to 2, fractional part must be zeros	

7.10.9 Gamma_Correction_Lookup registers (write-only)

Gamma_Correction_18_00 (330)
Gamma_Correction_38_20 (334)
Gamma_Correction_58_40 (338)
Gamma_Correction_78_60 (33C)
Gamma_Correction_98_80 (340)
Gamma_Correction_B8_A0 (344)
Gamma_Correction_D8_C0 (348)
Gamma_Correction_F8_E0 (34C)

These registers are used to write a 32-entry lookup table of 6-bit Gamma correction values. Table entries corrspond to every eighth value of luminance, starting with zero.

Bit	Name	Function	Default
5:0	Gamma0(5:0)	correction for first (smallest) luminance sample	
7:5	Not used		
13:8	Gamma0(5:0)	correction for second luminance sample	
15:14	Not used		
21:16	Gamma0(5:0)	correction for third luminance sample	
23:22	Not used		
29:24	Gamma0(5:0)	correction for fourth (largest) luminance sample	
31:30	Not used		

7.10.10GPIO_Control register (350)

Bit	Name	Function	Default
7:0	GPIO_In(7:0)	Read only	
15:8	GPIO_Out(7:0)	Read_Write	
23:16	GPIO_EnableH(7:0)	1= turn on tri-state driver for that pin	
24	DRAM_Debug_En	1= DRAM owner is output on GPIO(3:0)	
25	MPEG_Debug_En	1= MPEG state is output on GPIO(5:4)	
26	ISquaredC_En	1= ISquaredC controls GPIO(7:6)	

7.10.11ISquaredC registers

These registers control operation of an I^2C master using GPIO pins GPIO6 (SDATA) and GPIO7 (SCLK).

I2C_ReadWrite_2Bytes Register (354)

Bit	Name	Function	Default
15:8	Write_Data(7:0)	Used for write operations only	
23:16	Not used		
24	ReadH_WriteL	1=read from I2C Slave, 0=write to I2C slave	
31:25	Main_Address(6:0)	Read or Write main address	

I2C_Write_3Bytes Register (358)

Bit	Name	Function	Default
15:8	Write_Data(7:0)	Data written to (subaddress)	
23:16	Subaddress (7:0)		
24	(must be zero)	0=write to I2C slave	
31:25	Main_Address(6:0)	Read or Write main address	

I2C_Write_4Bytes Register (35C)

Bit	Name	Function	Default
7:0	Write_Data2(7:0)	Data written to (subaddress+1)	
15:8	Write_Data(7:0)	Data written to (subaddress)	
23:16	Subaddress (7:0)		
24	(must be zero)	0=write to I2C slave	
31:25	Main_Address(6:0)	Read or Write main address	

I2C_Control Register (364)

Bit	Name	Function	Default
9:0	I2C_Timeout	Number of I2C clocks to wait before aborting	
15:10	Not used		
23:16	Clock_Div	Divides 66mhz clock by this number to generate I2C clock	
30:24	Not used		
31	Reset_I2C	1=Reset I2C interface	

I2C_Status Register (368)

Bit	Name	Function
4:0	State	0000=IDLE, 0001=BUSY, 0010=SYNC1, 0011=SYNC2 0100=START, 0101=BYTE1, 0110=BYTE2R, 0111=BYTE2W, 1000=BYTE3W, 1001=BYTE4W, 1010=STOP, 1011=IRQOK, 1100=IRQNOTOK
7:5	Not used	
8	Timer_Status	1=Timer has timed out
9	Start_Status	1=Start not received
10	Bus_Arb_Status	1=Bus arbitration lost
11	Slave_ACK_Status	1=Slave has not acknowledged
12	Read_Transfer_Status	1=Read transfer not done
13	Write_Transfer_Status	1==Write transfer not done

I2C_Read_Data Register (36C)

Bit	Name	Function
7:0	Read_Data(7:0)	Data read from the slave

7.10.12SPDIF_Input_Control registers

These registers control inputting S/PDIF data using GPIO pins GPIO1 (DIN), GPIO2 (LRCK) and GPIO3 (SCLK).

SPDIF_Input_Control Register (370)

Bit	Name	Function	Default
0	Enable_SPDIF_In	1= Enables the SPDIF Input function	
1	LRCK_NewWord_Polarity	Sets the direction of an LRCK transition that triggers a 32-bit word to be parallel captured.	
2	LRCK_LSWord_MSWord	Sets which 16-bits of the 32-bit input word get loaded when LRCK rises.	
3	SPDIF_Interrupt_Status	1=A parallel word is available (read only).	
4	SPDIF_Overflow_Status	1=A second word was received while the Interrupt_Status bit was still set (read only).	
5	SPDIF_Interrupt_Clear	1=clears the Interrupt_Status and Overflow_Status bits (write-only).	

SPDIF_Input_Data Register (374)

Bit	Name	Function	Default

31:0	SPDIF_Data_In(31:0)		

7.11 On-Screen Display Registers

The On Screen Display function controls the output of Subpicture information to the display. The pixel buffer strores the Run Length Encoded pixel data, this data will be decoded internally to represent each pixel as two bits. The two-bit codes are mapped to different four bit codes using information in the control display buffer. The resulting 4-bit codes are used to index a 16-deep lookup table of YUV color .

7.11.1 Subpicture_Color registers

Subpicture Color0 - Subpicture Color15 (380 -3BC)

Sixteen color registers are used to hold the color pallate in YUV format for 16 possible subpicture colors.

Subpicture_Color Register Bit Assignments

Bit	Name	Function	Default
7:0	Color_V(7:0)	V Color	
15:8	Color_U(7:0)	U Color	
23:16	Color_Y(7:0)	Y Color	

7.11.2 Subpicture internal address registers

Subpicture DCI Address (3C0)

Subpicture DCI address Register Bit Assignments

Bit	Name	Function	Default
21:0	Internal SP_DCI address		

Subpicture HLI DCI Address (3C4)

Subpicture DCI address Register Bit Assignments

Bit	Name	Function	Default
21:0	Internal HLI_DCI address		

7.11.3 On_Screen_Display_Control register (3D0)

This register is used to enable/disable the display of subpictures and HLI.

On_Screen_Display_Control Register Bit Assignments

Bit	Name	Function	Default
0	Enable_Subpicture	1 = Displaying subpictures enabled	0

1	Reserved		0
2	Enable_HLI	1 = HLI exists 0 = No HLI	0
3	Activate the current HL	1 = Activate current HL, one shot Write - Only	
5	Test SP function	1 = Test SP function	
6	Test HLI function	1 = Test HLI function	

7.11.4 On Screen Display Scale/Offset registers

OSD_Offset (3D4)
OSD_Y_Scale_Factor_Offset (3D8)

The Onscreen display function performs scaling of its display, independant of the scaling of the Video Processor. The outputs of the two blocks are then blended by the Video Processor. Scaling in both X and Y dierections is done by a scale factor and an offset. The scale factor is a real number the Onscreen display uses to divide the pixel clock (for X) or line counter (for Y) to get a scale factor that matches that of the Video Processor. For both X and Y, the maximum scale factor is 2.0 (2X zoom out). The offset is the integer value, in scaled counts, from the start of the picture (horizontal sync for X, vertical sync for Y) to the start of the active display area (pixel 1 for X, line 1 for Y).

On Screen Display /Offset Register Bit Assignments

Bit	Name	Function	Default
9:0	X_Offset(9:0)	X offset count relative to Hsync	
23:16	Y_Offset(7:0)	Y offset count relative to Vsync	

On Screen Display Scale/Register Bit Assignments

Bit	Name	Function	Default
9:0	Frac_X_Scale_Factor(9:0)	Fractional part of X scale factor	
11:10	Int_X_Scale_Factor(1:0)	Integer part of X scale factor	
25:16	Frac_X_Scale_Factor(9:0)	Fractional part of Y scale factor	
27:26	Int_X_Scale_Factor(1:0)	Integer part of Y scale factor	

7.11.5 On Screen Debugging registers

OSD Debugging (3DC)

On Screen Display Debugging Register Bit Assignments

Bit	Name	Function	Default
2:0	Osd_State		
3	Frame_Start		
4	First_Line		
5	Other_Line		
6	SP_DCI_CYCLE		
7	SP_PIX_CYCLE		
8	HLI_Active_H		
9	SP_Active_H		
10	BTN_Active_H		
18:16	Osd27_State		

7.12 DRAM Data Formats

This sections specifies the format in DRAM for the following DRAM areas:

- video input stream FIFO
- audio input stream FIFO
- PTS sync FIFOs
- video output frame buffers
- PCM audio output buffers
- onscreen display pixel and display control buffers

This section uses byte addresses to specify DRAM addresses, using an address range of 20:0. The minimum address resolution in the LS220 is 2 bytes, so the internal address bus range is 19:0.

7.12.1 Video Input Stream FIFO

TBS

7.12.2 Audio Input Stream FIFO

TBS

7.12.3 PTS Sync FIFOs

The Sync function uses three FIFOs written by the host for timingkeeping (using audio as the time master) and to control the presentation of MPEG video, subpicture displays.
For all four FIFO types, each entry is 8 bytes long, containing a 33-bit presentation time stamp (PTS) and another field the hardware uses to track which presentation buffer it goes with. For audio and MPEG video, the PTS typically comes directly from the input stream, although the host may elect to modify these PTS's. For Subpictures, the PTS's are typically derived values by the host.

The audio sync FIFO contains a PTS and the frame number of the audio frame the PTS goes with.

	bit31			bit 0
Address n	PTS (32:1) (32 bits)			
Address n+4	not used (21 bits)		Frame_Number (9:0) (10 bits)	PTS(0) (1 bit)

The video sync FIFO contains a PTS and the byte count in the input stream where the PTS was found.

	bit31		bit 0
Address n	PTS (32:1) (32 bits)		
Address n+4	not used (1 bit)	Byte_Count (29:0) (30 bits)	PTS(0) (1 bit)

The subpicture sync FIFOs contain a PTS and the address in DRAM of the display control block the hardware should use for displaying at that PTS. Only address bits 16:2 out of address 20:0 are given since the display control buffers must be in the lower 256kbytes of DRAM and aligned on a 4-byte word boundary.

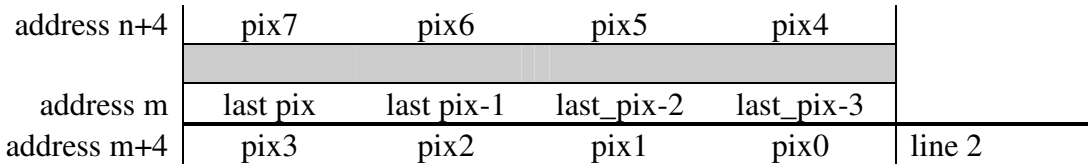
	bit31			bit0
Address n	PTS (32:1) (32 bits)			
Address n+4	not used (10 bits)	DRAM_Address (21:2) (20 bits)	Enable SP (1 bit)	Enable XY Group (1 bit)

7.12.4 Video Output Frame Buffers

Pixel data is stored in the buffer in display order. For interlaced source, the lines are stored interlaced (alternating even and odd lines). The number of bytes stored for a line is a multiple of macroblock size (16 bytes for Y / 8 bytes for U and V), so each line begins immediately after the previous line.

Video Frame Buffer DRAM Format

	bit31			bit 0	
address n	pix3	pix2	pix1	pix0	line 1


address n+4
pix7
pix6
pix5
pix4
address m
last pix
last pix-1
last_pix-2
last_pix-3
address m+4
pix3
pix2
pix1
pix0
line 2

7.12.5 PCM Audio Output Buffers

The PCM output format can be or 6-channel (front+rear+ambiance), 4-channel (front+rear) or 2-channel (stereo). For 4- and 2- channel modes, data can be written packed or unpacked. Packed data is shown in the figures. For unpacked data, the PCM output block skips the unused locations. For packed mono, identical data must be written to left and rear channels.

PCM Output buffer format for 6-channel mode
and unpacked 4- and 2-channel modes

	bit31 bit16	bit15 bit0
Address n	front left (15:0)	front right (15:0)
Address n+4	rear left (15:0)	rear right (15:0)
Address n+8	center (15:0)	ambiance (15:0)
Address n+12	(start of next buffer)	

PCM Output buffer format for 4-channel mode (packed)

	bit31 bit16	bit15 bit0
Address n	front left (15:0)	front right (15:0)
Address n+4	rear left (15:0)	rear right (15:0)
Address n+8	(start of next buffer)	

PCM Output buffer format for 2-channel mode (packed)

	bit31 bit16	bit15 bit0
Address n	front left (15:0)	front right (15:0)
Address n+4	(start of next buffer)	

7.12.6 Onscreen Display Buffers

The Onscreen display uses pixel display buffers and display control buffers in DRAM to display subpicturedata. Control and timing of the onscreen display is done by the host writing addresses and presentation timestamps to the Subpicture PTS sync FIFOs.

The pixel data buffers contain data which has been run-length encoded and will be decoded internally into two bits per pixel. The display control buffers contain information required to interpret and display the pixel data. The host may create any number of either type buffer, limited only by the DRAM area available. The host tells the hardware the starting address of the next display control buffer by writing it's address in the PTS sync FIFO. The hardware determines the start of the pixel data buffer using an address coded in the display control buffer.

The display control buffer for Subpicture contains the start/end X coordinates (in pixels) and Y coordinates (in lines) that define the position of the Subpicture on the display. The index field can be written by the host with a number used for tracking what buffer the hardware is using, as the hardware will make this filed available in a register when it starts reading this buffer. The Pix_Top_Adr and Pix_Bot_Adr fields define the start address(on byte boundary) of the pixel data buffer used for this display control buffer. The color and Contrast codes in the fifth word are used as the defaults for pixels not affected by subsequent color change information. Following the fifth word may be any number of color change sets. A color change set covers a range of lines, and may call out multiple color changes, to occur at defined X locations, on all lines in that range. Color change sets start on a word boundary.

Subpicture Display Control Buffer DRAM Format

	bit31			bit16 bit15				bit0
address n	filler (6b)	Start_Y(9:0) (10b)		filler (6b)	End_Y(9:0) (10b)			
address n+4	filler (6b)	Start_X(9:0) (10b)		filler (6b)	End_X(9:0) (10b)			
address n+8		Pix_Top_Adr(21:0) (22b)						
address n+12		Pix_Bot_Adr(21:0) (22b)						
address n+16	Color3 (4b)	Color2 (4b)	Color1 (4b)	Color0 (4b)	Cotra3 (4b)	Cotra2 (4b)	Cotra1 (4b)	Cotra0 (4b)
address m	Color_Change_Set 1							
address k								
address k+4	Color_Change_Set 2							

The color change sets defined in the Subpicture display control buffer contain the start and end line numbers over which they apply, a count of color changes in the set and, for each color change, an X position where the change takes affect and a new set of Color and Contrast values for that change. To limit hardware resources, the maximum number of color changes in a set is 8. Color changes are packed one after the other, except each new one starts on a 4-byte boundary. If unpacked storage is required for color change sets with less than 8 color changes, then the fill color changes at the end of the set should have an X position greater than the end of the line.

Subpicture Color Change Set DRAM Format

	bit31	bit16	bit15		bit0
address n	filler (6b)	Chg_Start_Line(9:0) (10b)	NChg(4b	Fill (2b)	Chg_End_Line(9:0)

					)		(10b)		
address n+4	Cotra3 (4b)	Cotra2 (4b)	Cotra1 (4b)	Cotra0 (4b)	Fill (6b)	Chg_X_Location(9:0) (10b)			first
address n+8	Fill (6b)	Chg_X_Location(9: 0) (10b)			Color3 (4b)	Color 2 (4b)	Color 1 (4b)	Color0 (4b)	change
address n+12	Color 3 (4b)	Color 2 (4b)	Color 1 (4b)	Color 0 (4b)	Cotra 3 (4b)	Cotra 2 (4b)	Cotra 1 (4b)	Cotra 0 (4b)	second
address n+16									change

7.12.7 HLI and Button Buffers

The Button is a rectangular area on the screen, which may be selected by the user. In general, Buttons are defined as a combination of Video, Sub-picture and Highlight Information in PCI. The basic feaure for Highlights are :

1. a maximun of 36 Highlights are permitted.
2. a maximun of 3 Button groups are permitted.
3. only one hightlight may be selected at a time .
4. the number of Highlight may be selected numerically .

The display control buffer for HLI contains the start/end X coordinates (in pixels) and Y coordinates (in lines) that define the position of the Current and Force Active HL on the display. The FOAC_BTNN and FOSL_BTNN select one of the 36 buttons to be displayed . The color and Contrast codes of selection and action states are used as the pixel attributes during selection and action period respectivelly.

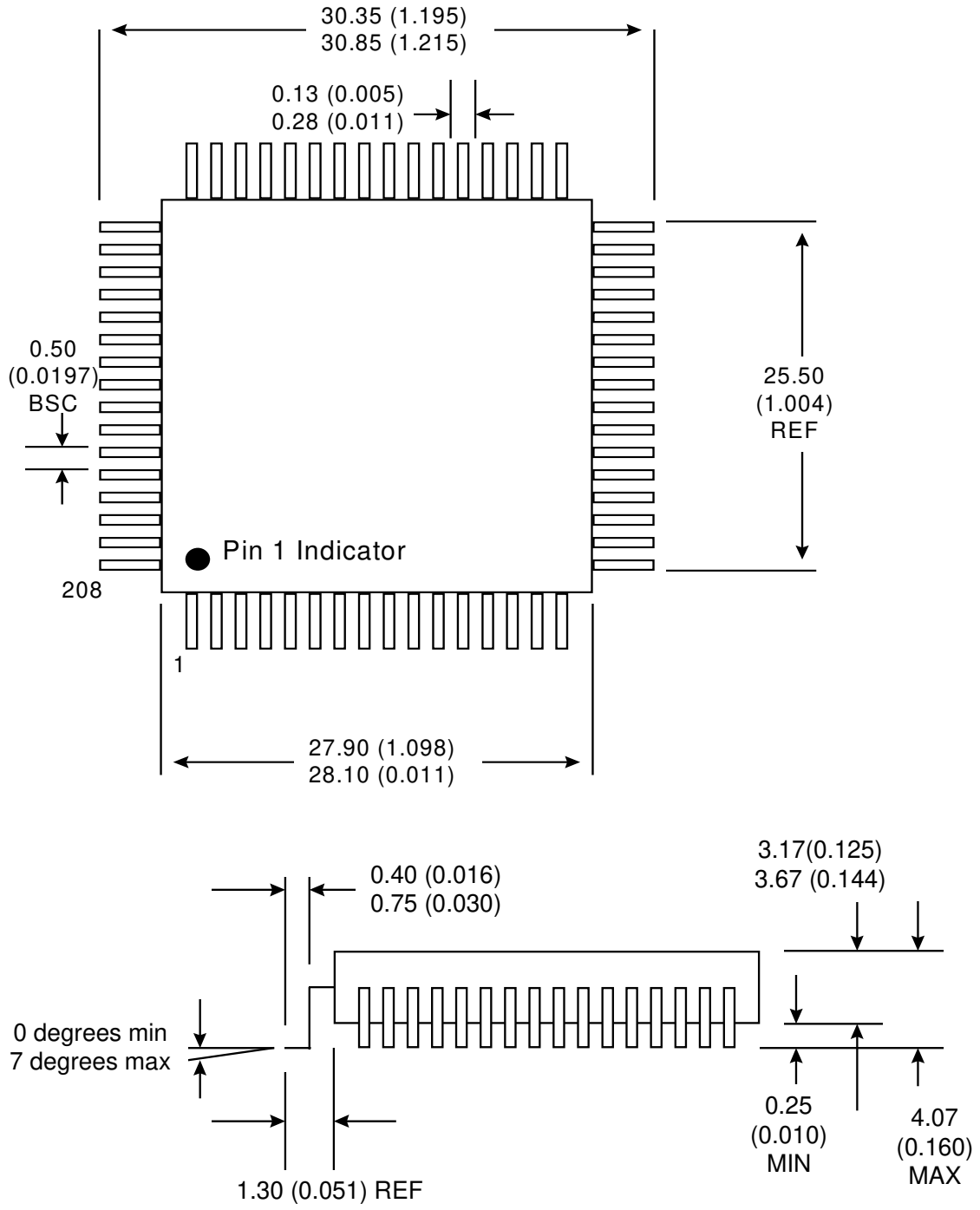
HL Display Control Buffer DRAM Format

	bit31		bit16	bit15		bit0
address n	fill (6b)		Start_Y (10b)	BTN_COLN (2b)	fill (4b)	End_Y (Current HL) (10b)
address n+4	Auto(1:0) (2b)	fill (4b)	Start_X(9:0) (10b)	fill (6b)		End_X (Current HL) (10b)
address 8	fill (6b)		Start_Y (10b)	BTN_COLN (2b)	fill (4b)	End_Y (F Active HL) (10b)

address n+12	Auto(1:0) (2b)	fill (4b)	Start_X(9:0) (10b)	fill (6b)	End_X (FActive HL) (10b)
address n+16	filler (10b)	FOAC_BTNN(5:0) (6b)		filler (10b)	FOSL_BTNN(5:0) (6b)
address n+20	Selection Color (16b)			Selection Cotrast (16b)	
address n+24	Action Color (16b)			Action Cotrast (16b)	

8 Package Dimensions

208 pin PQFP



1. Dimensions are in millimeters (inches), and standard dimension is millimeters.
2. Drawing is not to scale and number of pins is inexact.

Copyright Statement

Copyright 1996 LuxSonor, Inc. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopy, or otherwise, without the prior written consent of LuxSonor, Inc. The material in this document is for information only and is subject to change without notice. LuxSonor, Inc. reserves the right to make changes in the product design without reservation and without notice to its users.

Address and contact information

2 LuxSonor Offices http://www.LuxSonor.com

*Lux***Sonor**-USA
HEADQUARTERS
4110 Clipper Court
Fremont, CA 94538
Tel: (510) 683-4668
Fax: (510) 683-4669

*Lux***Sonor-**Hong Kong
Flat 7, 9/F., Ho Lik Centre,
64-66 Sha Tsui Road
Tsen Wan, N.T.,
Hong Kong
Tel: (852) 2414-2194
Fax: (852) 2405-3439

*Lux***Sonor-**Taiwan
Room C, 8F, No. 248, Sec. 3
Nan King E. Road
Taipei, Taiwan
Tel: (02) 7736350
Fax: (02) 7754610