DATA SHEET

Generic device for portable multimedia applications
SAA7750-N1D

Preliminary Specification version 1.3
File under Integrated Circuits, <Handbook>

2002 Jan 21

CONTENTS

SAA7750-N1D

Generic device for portable multimedia applications

1   FEATURES

NOTE: this datasheet is for SAA7750EI version N1D onwards!!

1.1   Hardware Features

• Integrated ARM720T 32 bit RISC processor, capable of running at 72MHz.

• High performance 32-bits bus (AHB)

• Centralized address decoding for all AHB devices

• Four possible memory maps:
  – external boot
  – internal flash boot
  – internal ROM boot
  – normal operation

• Supports USB 1.1 compliant interface for down loading data from PC

• Support for flash-card applications:
  – Supports the Multi Media Card (MMC)
  – Supports Smart Media Card (EBI)
  – NAND FLASH (EBI)

• Memory interface (EBI) supporting a number of memory types like Static RAM, SDRAM, external Flash.
  The maximum bus frequency can be up to 48MHz.

• Integrated CD block decoder for CD-DA and MP3 CD applications

• UART + IrDA (IrDA is a new block on the N1D version)

• Integrated Master and Slave IIC interface

• Real-Time Clock (RTC)

• General-Purpose IO pins (28 pins)

• Integrated Remote Control interface

• Integrated LCD interface with 6800 / 8080 type interface

• Integrated 10 bits ADC with 8 selectable inputs (via analog multiplexer).

• Integrated SPDIF output interface

• Integrated IIS input and output interface

• Integrated stereo Audio Codec
  – Stereo Line input with Programmable Gain Amplifier (PGA)
  – Mono Microphone input with embedded Low Noise Amplifier (LNA) and Variable Gain Amplifier (VGA
  – stereo analog input with analog volume control (e.g. for tuner applications)
  – stereo line output
  – integrated stereo headphone driver which can be used in DC coupling (short circuit protection and detection build in).

1.2   General Features

• Integrated ARM720T 32 bit RISC processor

• Programmable architecture enables support of multiple audio decompression algorithms.

• Designed for applications that require long battery life

- Embedded 3Mbit (384kbyte) flash for Field upgradibility
- Embedded Audio Codec with headphone driver
- small footprint LFBGA208 package

1.3 Software features

- Audio Decoder support:
  - Supports MPEG 1 layer 3 and MPEG 2 layer 2.5 and layer 3 audio decoding (MP3), up to 320kbit/s , fixed and variable bitrate.
  - Supports Microsoft WMTA 4.0 decoding
  - Supports AAC-LC decoding
- Features on the audio codec:
  - Digital Automatic Gain Control (AGC) on the microphone input.
  - Programmable Gain Amplifier (PGA) for analog stereo line input
  - Volume control (incl. balance)
  - Bass-boost and Treble (left/right)
- DSP features:
  - UltraBass II
  - Incredible headphone
  - Infrapitch

2 GENERAL DESCRIPTION

The SAA7750 is an IC based on an embedded RISC processor in combination with a simple embedded DSP core for audio post-processing. The device is designed for hand-held applications like portable CD-DA/ MP3 players, memory card applications or other portable applications. The high level of integration, low power consumption and high processor performances make the SAA7750 very suitable for portable hand-held devices.

The SAA7750 is based on the powerful ARM720T CPU core, which is a full 32-bit RISC processor featuring the 16-bit Thumb instruction set for effective memory usage. The audio streaming and post-processing for the SAA7750 is handled by a separate audio co-processor DSP, which is a small, fast and powerful 24-bit Epics7A DSP core.

3 APPLICATIONS

- Portable Solid State Audio player
- Portable MP3 CD player
- Home audio applications
- Non-automotive Car applications
- Other portable applications like PDA

4 ORDERING INFORMATION

TYPE NUMBER	PACKAGE			
	NAME	DESCRIPTION		VERSION
SAA7750EL/N1	LFBGA208	low profile fine-pitch ball grid array package; 208 balls; body 15 x 15 x 1.2 mm.		SOT631-1

Generic device for portable
multimedia applications

5   BLOCK DIAGRAM
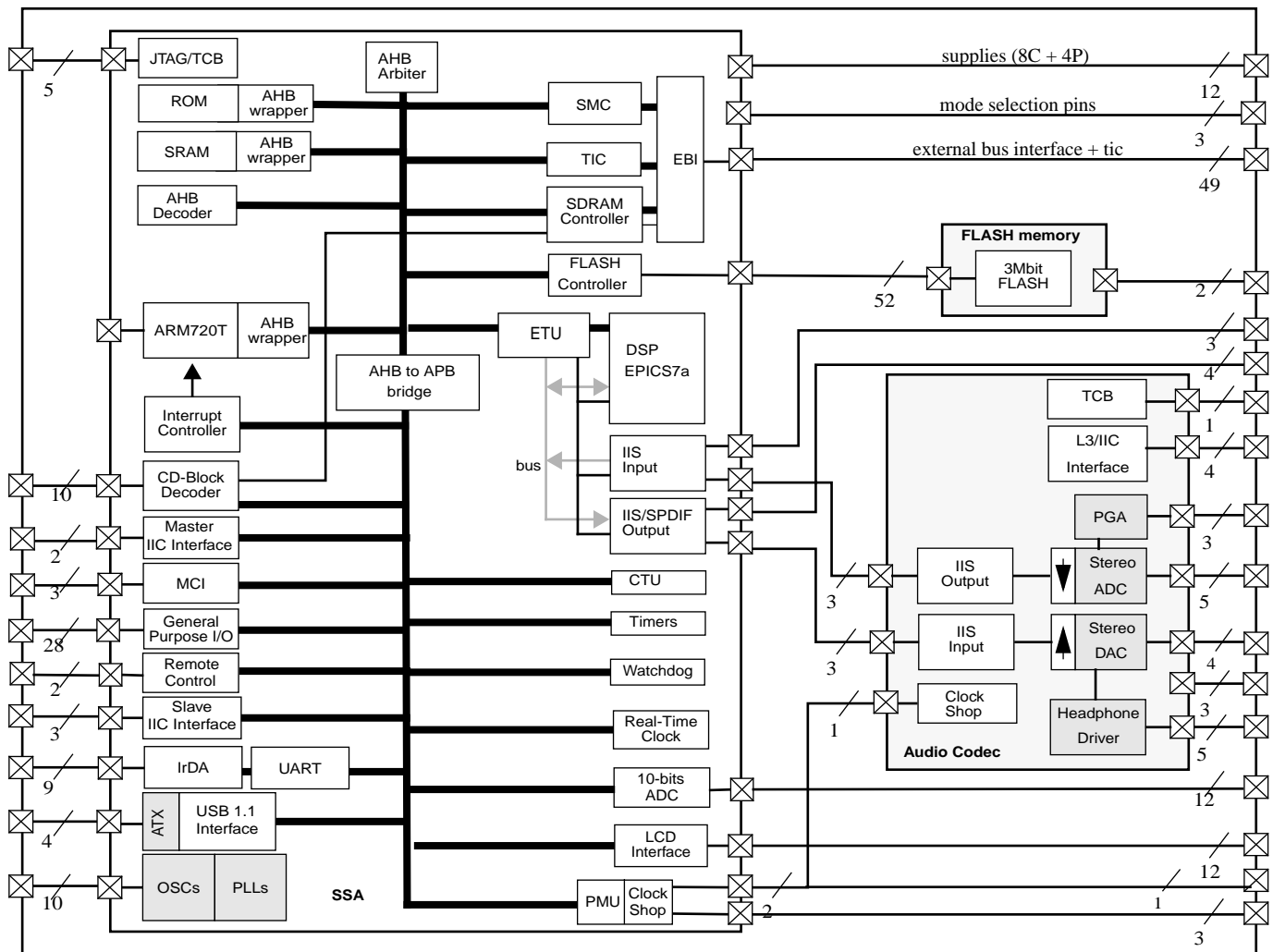
Fig. 1 Block diagram Solid State Audio 1

SAA7750-N1D

Generic device for portable
multimedia applications

6   PINNING

Table 1   Pin list SAA7750EL

SYMBOL[1]	LFBGA 208 PIN	DIGITAL I/O LEVEL	APPL. FUNC	PIN STATE AFTER RESET	DESCRIPTION
General Purpose Pins (fixed: 16 pins)					
GPIO<27>	A13	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<26>	A12	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<25>	B12	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<24>	A11	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<23>	B11	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<22>	A10	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<21>	B10	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<20>	A9	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<19>	B9	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<18>	A8	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<17>	B8	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<16>	A7	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<15>	F4	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<14>	G2	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<13>	F3	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<12>	G1	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<11>	F2	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<10>	F1	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<9>	D3	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<8>	E2	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<7>	D4	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<6>	E1	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<5>	D2	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<4>	D1	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<3>	C2	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<2>	C1	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<1>	B1	0-5 VDC tolerant	I/O	0	General Purpose IO pin
GPIO<0>	A1	0-5 VDC tolerant	I/O	0	General Purpose IO pin
Memory Card Interface (fixed: 3 pins)					
MCI_DAT	A4	0-5 VDC tolerant	I/O		Data input/Data output
MCI_CLK	A2		O		MCI clock output
MCI_CMD	B2	0-5 VDC tolerant	I/O		Command input/Command output
USB Interface (fixed: 4 pins)					
USB_DP	C17		A		Positive USB data line
USB_DM	D17		A		Negative USB data line
USB_CONNECT_N	D16		O		Soft connect output
USB_VUSB	C15	0-5 VDC tolerant	I		USB supply detection input
6 MHz oscillator (fixed: 4 pins)					
XTAL1I	P4		A		6MHz clock input
XTAL1O	R3		A		6MHz clock output

SAA7750-N1D

Generic device for portable multimedia applications

SYMBOL[1]	LFBGA 208 PIN	DIGITAL I/O LEVEL	APPL. FUNC	PIN STATE AFTER RESET	DESCRIPTION
VDDA1	R2				Analog supply Oscillator 1
VSSA1	R1				Analog ground Oscillator 1
32.768 kHz oscillator (fixed: 4 pins)					
XTAL2I	N4		A		32.768 kHz clock input
XTAL2O	P3		A		32.768 kHz clock output
VDDA2	P2				Analog supply Oscillator 2
VSSA2	P1				Analog ground Oscillator 2
Voltage Supply PLLs (fixed: 2 pins)					
VDDA3	N2				Analog supply PLLs
VSSA3	N1				Analog ground PLLs
PLL (fixed: 1 pin)					
CLKO1	F15		O	toggling	256fs clock output
LCD Interface (fixed: 13 pins)					
LCD_WE	K3		O		Write Enable
LCD_RW_WR	A16		O		6800 read/write select 8080 active 'high' write enable
LCD_E_RD	B15		O		6800 active 'low' enable 8080 active 'high' write enable
LCD_DB<0>	D14	0-5 VDC tolerant	I/O		Data input 0/Data output 0
LCD_DB<1>	B17	0-5 VDC tolerant	I/O		Data input 1/Data output 1
LCD_DB<2>	C14	0-5 VDC tolerant	I/O		Data input 2/Data output 2
LCD_DB<3>	C16	0-5 VDC tolerant	I/O		Data input 3/Data output 3
LCD_DB<4>	D13	0-5 VDC tolerant	I/O		Data input 4/Data output 4
LCD_DB<5>	A17	0-5 VDC tolerant	I/O		Data input 5/Data output 5/serial clock
LCD_DB<6>	C13	0-5 VDC tolerant	I/O		Data input 6/Data output 6/Serial data input
LCD_DB<7>	B16	0-5 VDC tolerant	I/O		Data input 7/Data output 7/Serial data output
LCD_CSB	C12		O		Chip Select (active low)
LCD_RS	D12		O		'high' Data register select 'low' Instruction register select
Parallel Port Interface (fixed: 18 pins) .. THIS FUNCTIONALITY HAS BEEN REMOVED!!					
10-bit ADC (fixed: 12pins)					
GPA<7>	B7		A		Analog General Purpose pin 7
GPA<6>	A6		A		Analog General Purpose pin 6
GPA<5>	B6		A		Analog General Purpose pin 5
GPA<4>	A5		A		Analog General Purpose pin 4
GPA<3>	B5		A		Analog General Purpose pin 3
GPA<2>	J3		A		Analog General Purpose pin 2
GPA<1>	M4		A		Analog General Purpose pin 1
GPA<0>	N3		A		Analog General Purpose pin 0
VREFP<1>	M3		A		10-bit ADC Reference voltage 1
VREFP<0>	L2		A		10-bit ADC Reference voltage 0
VDDA4	M2				Analog supply 10-bit ADC
VSSA4	M1				Analog ground 10-bit ADC
Remote Control (fixed: 2 pins)					
DO<0>	K1		O		Remote Control Data Output 0
DI<0>	K2	0-5 VDC tolerant	I		Remote Control Data Input 0
IIS input (fixed: 3 pins)					
BCKI1	J15	0-5 VDC tolerant	I		Bitclock input (external)

SAA7750-N1D

Generic device for portable multimedia applications

SYMBOL[1]	LFBGA 208 PIN	DIGITAL I/O LEVEL	APPL. FUNC	PIN STATE AFTER RESET	DESCRIPTION
WSI1	H15	0-5 VDC tolerant	I		Wordselect input (external)
DATAI1	G15	0-5 VDC tolerant	I		Serial data input (external)
IIS output (fixed: 3 pins)					
BCKO1	M14		O	Tri-state	Bitclock output (external)
WSO1	F16		O	Tri-state	Wordselect output (external)
DATAO1	E16		O	Output/Low	Serial data output (external)
SPDIF output (fixed: 1 pin)					
DATAO2_SPDIFO	E15		O		Serial data output (internal), SPDIF output
JTAG (fixed: 5 pins)					
JTAG_NTRST	K15	0-5 VDC tolerant	I		JTAG Reset Input
JTAG_TCK	U12	0-5 VDC tolerant	I		JTAG Clock Input
JTAG_TMS	K16	0-5 VDC tolerant	I		JTAG Mode Select Input
JTAG_TDI	T13	0-5 VDC tolerant	I		JTAG Data Input
JTAG_TDO	U13		O		JTAG Data Output
IIC slave Interface (fixed: 3 pins)					
SCL_SLAVE	P12	0-5 VDC tolerant	I		Serial clock IIC Slave
SDA_SLAVE	R12	0-5 VDC tolerant	I/O		Serial data IIC Slave
A0_SLAVE	T12	0-5 VDC tolerant	I		Address selection Slave
IIC master interface (fixed: 2 pins)					
SDA_MASTER	R13	0-5 VDC tolerant	I/O		IIC data I/O line (open drain output)/ UART Serial Data Input
SCL_MASTER	P13	0-5 VDC tolerant	I/O		IIC clock line output/ UART Serial Data Output
CD Block Decoder (fixed: 10 pins)					
CDB_CRQ_NERDY	C5	0-5 VDC tolerant	I		Communication request line/CD engine is ready to receive the next frame
CDB_NCRST_NHRDY	D5		O		CD engine reset line/Host is ready to receive the next frame
CDB_CLAB	C9	0-5 VDC tolerant	I		IIS/EIAJ input bit clock
CDB_DAAB	C7	0-5 VDC tolerant	I		IIS/EIAJ serial data
CDB_WSAB	C8	0-5 VDC tolerant	I		IIS/EIAJ word clock
CDB_EFAB	D9	0-5 VDC tolerant	I		IIS/EIAJ error flags
CDB_V4_SUB	D8	0-5 VDC tolerant	I		Versatile pin 4:single wire subcode/EIAJ subcode data bits
CDB_CFLAG_SBSY	D6	0-5 VDC tolerant	I		Absolute time sync/EIAJ subcode block sync
CDB_SFSY	D7	0-5 VDC tolerant	I		EIAJ subcode frame sync
CDB_RCK	C6		O		EIAJ subcode clock output
EBI (fixed: 49 pins)					
EBI_NCS<2>	G16		O		Chip Selected 2
EBI_NCS<1>	T10		O		Chip Selected 1
EBI_NCS<0>	U10		O		Chip Selected 0
EBI_SDNCS<0>	H3		O		External SDRAM selection1 and SDRAM selection0
EBI_WEN	J2		O		Write enable not
EBI_A<20>	J16		O		EBI address
EBI_A<19>	H16		O		EBI address
EBI_A<18>	F14		O		EBI address
EBI_A<17>	G14		O		EBI address
EBI_A<16>	H14		O		EBI address
EBI_A<15>	J14		O		EBI address
EBI_A<14>	R9		O		EBI address

SAA7750-N1D

Generic device for portable multimedia applications

SYMBOL[1]	LFBGA 208 PIN	DIGITAL I/O LEVEL	APPL. FUNC	PIN STATE AFTER RESET	DESCRIPTION
EBI_A<13>	T9		O		EBI address
EBI_A<12>	U9		O		EBI address
EBI_A<11>	R8		O		EBI address
EBI_A<10>	T8		O		EBI address
EBI_A<9>	U8		O		EBI address
EBI_A<8>	P11		O		EBI address
EBI_A<7>	R7		O		EBI address
EBI_A<6>	P10		O		EBI address
EBI_A<5>	U7		O		EBI address
EBI_A<4>	P9		O		EBI address
EBI_A<3>	T7		O		EBI address
EBI_A<2>	P8		O		EBI address
EBI_A<1>	R6		O		EBI address
EBI_A<0>	U6		O		EBI address
EBI_D<15>	T6	0-5 VDC tolerant	I/O		EBI data
EBI_D<14>	U5	0-5 VDC tolerant	I/O		EBI data
EBI_D<13>	T5	0-5 VDC tolerant	I/O		EBI data
EBI_D<12>	U4	0-5 VDC tolerant	I/O		EBI data
EBI_D<11>	T4	0-5 VDC tolerant	I/O		EBI data
EBI_D<10>	U3	0-5 VDC tolerant	I/O		EBI data
EBI_D<9>	T3	0-5 VDC tolerant	I/O		EBI data
EBI_D<8>	P7	0-5 VDC tolerant	I/O		EBI data
EBI_D<7>	U2	0-5 VDC tolerant	I/O		EBI data
EBI_D<6>	P6	0-5 VDC tolerant	I/O		EBI data
EBI_D<5>	U1	0-5 VDC tolerant	I/O		EBI data
EBI_D<4>	R5	0-5 VDC tolerant	I/O		EBI data
EBI_D<3>	T2	0-5 VDC tolerant	I/O		EBI data
EBI_D<2>	P5	0-5 VDC tolerant	I/O		EBI data
EBI_D<1>	T1	0-5 VDC tolerant	I/O		EBI data
EBI_D<0>	R4	0-5 VDC tolerant	I/O		EBI data
EBI_SDCLKOUT	J1		O		SDRAM clock
EBI_CKE<0>	H4		O		SDRAM clock enable
EBI_DQM<1>	T11		O		SDRAM data mask 1
EBI_DQM<0>	U11		O		SDRAM data mask 0
EBI_NRAS	R10		O		SDRAM row address strobe
EBI_NCAS	R11		O		SDRAM column address strobe
EBI_NOE	H2		O		EBI output enable
Test pins (3 pins)					
TEST_DAT<3>	B3	0-5 VDC tolerant	I/O		Data input/Data output
TEST_DAT<2>	A3	0-5 VDC tolerant	I/O		Data input/Data output
TEST_DAT<1>	B4	0-5 VDC tolerant	I/O		Data input/Data output
UART (fixed: 9 pins)					
UART_IO_NRI	E14	0-5 VDC tolerant	I		
UART_DIR_TX	D10		O		
UART_REQ_RX	C10	0-5 VDC tolerant	I		
UART_RST_NRTS	C11		O		
UART_CLK	D11	0-5 VDC tolerant	I/O		

SAA7750-N1D

Generic device for portable multimedia applications

SYMBOL[1]	LFBGA 208 PIN	DIGITAL I/O LEVEL	APPL. FUNC	PIN STATE AFTER RESET	DESCRIPTION
UART_NCTS	B14	0-5 VDC tolerant	I		
UART_NDCD	A15	0-5 VDC tolerant	I		
UART_NDSR	B13	0-5 VDC tolerant	I		
UART_NDTR	A14		O		
Mode Selection pins SAA7750 (fixed: 3 pins)					
MODE<2>	L16	0-5 VDC tolerant	I		
MODE<1>	M15	0-5 VDC tolerant	I		
MODE<0>	M16	0-5 VDC tolerant	I		
Wake-up input pin SAA7750 (fixed: 1 pin)					
WAKE_UP	L1	0-5 VDC tolerant	I		Wake up input pin
Reset input pin SAA7750 (fixed: 1 pin)					
NRESET_IN	L15	0-5 VDC tolerant	I		System Reset Input
Reset output pin SAA7750 (fixed: 1 pin)					
RESET_OUT	N16		O		Reset output
Reset input pin SSA Audio Codec (fixed: 1 pin)					
RESET	N15	0-5 VDC tolerant	I		Reset input pin with pull-down for creating Power-On-Reset
DAC SSA Audio Codec (fixed: 4 pins)					
VOUTL	P15		A		Analog left output pin
VOUTR	R16		A		Analog right output pin
VDDA(DA)	R17				Analog supply DAC
VSSA(DA)	P16				Analog ground DAC
Headphone Amplifier SSA Audio Codec (fixed: 5 pins)					
VOUTL(HP)	T17		A		Analog left output pin
VOUTR(HP)	U17		A		Analog right output pin
VREF(HP)	T16		A		Analog reference output pin
VDDA(HP)	R15				Analog supply Headphone Driver
VSSA(HP)	U16				Analog ground Headphone Driver
ADC SSA Audio Codec (fixed: 8 pins)					
VINL	M17		A		Left line input
VINR	K17		A		Right line input
VINM	H17		A		Microphone input
VADCP	G17		A		Positive Reference voltage ADC
VADCN	J17		A		Negative Reference voltage ADC
VREF	P17		A		Reference voltages ADC
VDDA(AD)	L17				Analog supply ADC
VSSA(AD)	N17				Analog ground ADC
Control pins SSA Audio Codec (fixed: 4 pins)					
L3CLOCK_SCL	P14	0-5 VDC tolerant	I		L3 Clock/IIC clock input
L3DATA_SDA	R14	0-5 VDC tolerant	I		L3 Data/IIC data input
L3MODE_A0	U15	0-5 VDC tolerant	I		L3 Mode/IIC address selection
SELECT_L3_IIC	T15	0-5 VDC tolerant	I		Select pin for L3 (LOW) or IIC control (HIGH)
Test pin SSA Audio Codec (fixed: 1 pin)					
TEST1	T14	0-5 VDC tolerant	I		Test control pin
Supplies SSA Audio Codec (fixed: 2 pins)					
VDDD(CODEC)	N14				Digital supply Audio Codec
VSSD (CODEC)	U14				Digital ground Audio Codec
Supplies SSA Flash memory (fixed: 2 pins)					

SYMBOL[1]	LFBGA 208 PIN	DIGITAL I/O LEVEL	APPL. FUNC	PIN STATE AFTER RESET	DESCRIPTION
VDDD(FLASH)	E17				Digital supply Flash
VSSD (FLASH)	F17				Digital ground Flash
Digital supplies SAA7750 (fixed: 8 pins)					
VDDI1	L4				Core supply SAA7750
VSSIS1	L3				Core ground and substrate SAA7750
VDDI2	G4				Core supply SAA7750
VSSI2	G3				Core ground SAA7750
VDDI3	E4				Core supply SAA7750
VSSI3	E3				Core ground SAA7750
VDDI4	C4				Core supply SAA7750
VSSI4	C3				Core ground SAA7750
Peripheral supplies SAA7750 (fixed: 4 pins)					
VDDE3V3	J4				Peripheral (I/O) supply SAA7750 (3.3V)
VSSE3V3	K4				Peripheral (I/O) ground SAA7750
VSSE3V3	H1				Peripheral (I/O) ground SAA7750
VDDE2V5	K14				Peripheral (I/O) supply SAA7750 (2.5V)
VSSE2V5	L14				Peripheral (I/O) ground SAA7750
Not connected pins (fixed: 2 pins)					
NC	D15				Not connected

1. Pin positions are fixed.

Table 2   : pinning diagram

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
A	GPIO<0>	MCI_CLK	TEST_DAT<2>	MCI_DAT<0>	GPA<4>	GPA<6>	GPIO<16>	GPIO<18>	GPIO<20>	GPIO<22>	GPIO<24>	GPIO<26>	GPIO<27>	UART_nDTR	UART_nDCD	RW_WR	DB<5>
B	GPIO<1>	MCI_CMD	TEST_DAT<3>	TEST_DAT<1>	GPA<3>	GPA<5>	GPA<7>	GPIO<17>	GPIO<19>	GPIO<21>	GPIO<23>	GPIO<25>	UART_nDSR	UART_nCTS	E_RD	DB<7>	DB<1>
C	GPIO<2>	GPIO<3>	VSSI4	VDDI4	CRQ_nERD	RCK	DAAB	WSAB	CLAB	REQ_RX	RTRST_nF	CSB	DB<6>	DB<2>	VUSB	DB<3>	USB_DPLUS
D	GPIO<4>	GPIO<5>	GPIO<9>	GPIO<7>	CRST_nHRQ	CFLAG_SBSF	SFSY	V4_SUB	EFAB	UART_DIR_T	UART_CLK	RS	DB<4>	DB<0>	N.C.	USB_CONNECT	USB_DMIN
E	GPIO<6>	GPIO<8>	VSSI3	VDDI3										UART_IO_nF	SPDIFO	DATAO1	VDD(F)
F	GPIO<10>	GPIO<11>	GPIO<13>	GPIO<15>										A<18>	CLK01	WSO1	VSS(F)
G	GPIO<12>	GPIO<14>	VSSI2	VDDI2										A<17>	DATAI1	nCS_2	VADCP
H	VSSE	nOE	SDnCS0	CKE0										A<16>	WSI1	A<19>	VINM
J	DCLKO	WEN	GPA<2>	VDDE3V3										A<15>	BCKI1	A<20>	VADCN
K	DO<0>	DI<0>	WE	WSSE3V3										VDDE2V5	JTAG_nTRST	JTAG_TMS	VINR
L	WAKE_UP	VREFP<0>	VSSI1	VDDI1										VSSE2V5	nRESET_IN	MODE<2>	VDDA(AD)
M	VSSA4	VDDA4	VREFP<1>	GPA<1>										BCK01	MODE<1>	JTAG_MODE<0>	VINL
N	VSSA3	VDDA3	GPA<0>	XTAL2i										VDDD	RESET	RESET_OUT	VSSA(AD)
P	VSSA2	VDDA2	XTAL2O	XTAL1I	D<2>	D<6>	D<8>	A<2>	A<4>	A<6>	A<8>	SCL_SLAVE	SCL_M	L3CLOCK	VOUTL	VSSA(DA)	VREF
R	VSSA1	VDDA1	XTAL1O	D<0>	D<4>	A<1>	A<7>	A<11>	A<14>	nRAS	nCAS	SDA_SLAVE	SDA_M	L3DATA	VDD(HP)	VOUTR	VDDA(DA)
T	D<1>	D<3>	D<9>	D<11>	D<13>	D<16>	A<3>	A<10>	A<13>	nCS<1>	DQM<1>	AO_SLAVE	JTAG_TDI	TEST1	SEL_L3_IIC	VREFHP	VOUTL-HP
U	D<5>	D<7>	D<10>	D<12>	D<14>	A<0>	A<5>	A<9>	A<12>	nCS<0>	DQM<0>	JTAG_TCK	JTAG_TDO	VSSD	L3MODE	VSS(HP)	VOUTR-HP
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17

Note: the pins which which have been changed between the version N1A, N1B, N1C and the final version N1D of the IC have been marked with **RED**. The pins which were changed and changed from digital to analog are marked with **BLUE**.

There is one pin (pin D15) which is left open.

7   HARDWARE DESCRIPTION SAA7750

7.1   ARM720T microcontroller

Quick reference:

- High performance low power ARM7TDMI based 32-bit RISC processor
- ARM 16-bit Thumb instruction set
- 8 KByte Unified Cache
- Memory Management Unit (MMU) giving full virtual memory and fast context switching support
- 32-bit register bank
- 32-bit ALU for RISC performance
- 32-bit shifter
- 32-bit addressing (no paging required above 64KByte)
- 32 x 8 DSP multiplier for signal processing
- Embedded ICE logic for debug
- Maximum ARM core clock frequency is 72MHz.

7.1.1   OVERVIEW

ARM720T is a general-purpose 32-bit microprocessor with 8KB cache, enlarged write buffer and Memory Management Unit (MMU) combined in a single core. The CPU within ARM720T is the ARM7TDMI. The ARM720T is software compatible with the ARM processor family.

ARM720T is a fully static part and has been designed to minimize power requirements. This makes it ideal for portable applications, where both these features are essential.

The ARM720T architecture is based on Reduced Instruction Set Computer (RISC) principles, and the instruction set and related decode mechanism are greatly simplified compared with micro programmed Complex Instruction Set Computers (CISC).

The MMU's mixed data and instruction cache, together with the write buffer, substantially raise the average execution speed and reduce the average amount of memory bandwidth required by the processor. This means that there is a minimal performance loss, when using 'slow' DRAM and 'slow' internal flash memory.

The memory interface has been designed to allow the performance potential to be realized without incurring high costs in the memory system. Speed-critical control signals are pipelined to allow system control functions to be implemented in standard low-power logic, and these control signals permit the exploitation of paged mode access offered by industry standard DRAMs.

7.1.2   BLOCK DIAGRAM

Fig. 2 ARM720T Block Diagram
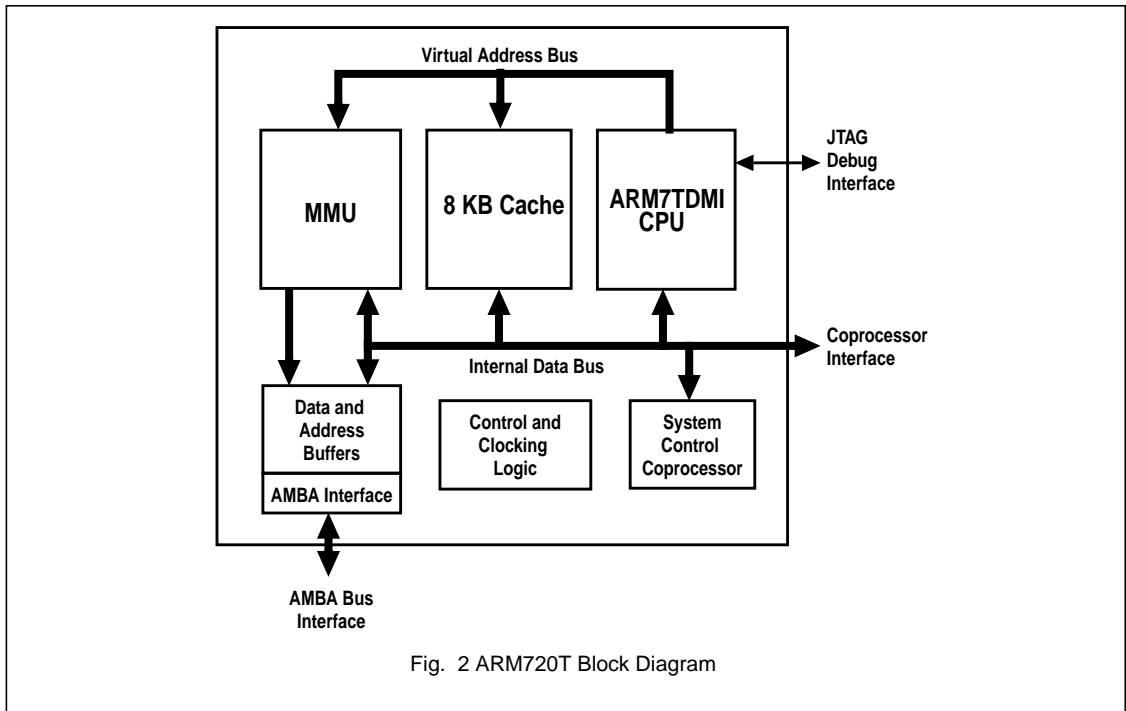
7.1.3   THE THUMB CONCEPT

The THUMB instruction set is a subset of the ARM instruction set. The THUMB is designed to increase the performance of ARM implementations that uses a 16-bit memory data bus, and may allow better code density than ARM instruction set.

The THUMB set's 16-bit instruction length allows it to approach twice the density of standard ARM code while retaining most of the ARM's performance advantage over a traditional 16-bit processor using 16-bit registers. This is possible because THUMB code operates on the same 32-bit register set as ARM code.

THUMB code is able to provide up to 65% of the code size of ARM, and 160% of the performance of an equivalent ARM processor connected to a 16-bit memory system.

Note: in standard operation accesses are 32-bit, only when executing via EBI the accesses are 16-bit.

7.2   Memory controllers

TThe SAA7750 offers transparent memory-mapped access for the processor to static memory and SDRAM devices.

Refer to the application notes on the memory controller.

7.2.1   OVERVIEW

The memory interface consists of an external bus interface (EBI) that handles all data and address interfacing from the SAA7750 to the outside world, and consists of two memory controllers. The first memory controller handles SRAM / ROM. This controller is also known as Static Memory Controller (SMC). The second memory controller handles SDRAM which is located externally. In Fig.3 on page 16 a block diagram of the SAA7750 memory interface is depicted.
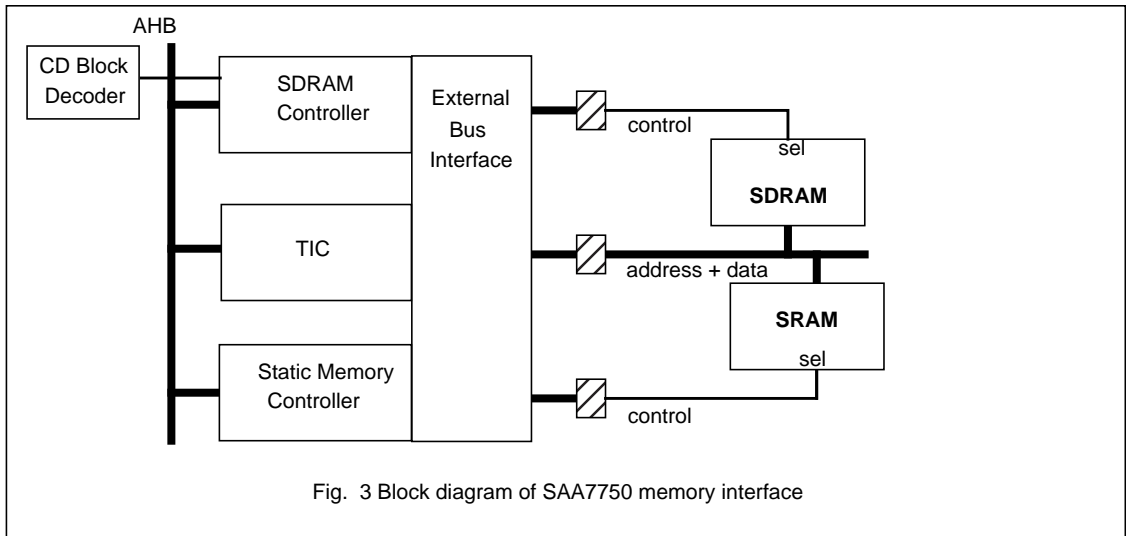

Fig. 3 Block diagram of SAA7750 memory interface

Note: the SDRAM and the SMC cannot be used together in one application since there is no arbitragion in the EBI to control the priority between the two blocks and the refresh of the SDRAM in that case.

7.2.2   Static Memory Controller

Within the memory interface the Static Memory Controller (SMC) is one the controllers which communicates with the External Bus Interface (EBI). This static memory controller can control up to four independent memory or expansion banks simultaneously. Those memories can be SRAM, ROM, FLASH or off-chip located peripherals. Each bank is 64 MByte, where the Static Memory Controller can handle all of the six main functions:

• Memory bank selection: support of 8 memory banks (64MByte each)

• Little Endian system

• Programmable wait states for read and write access:

  – 1...32 wait states for standard memory access

  – 0...15 wait states for burst mode reads from ROMs

• Supports sequential access burst reads of up to four consecutive locations in 8-, 16-, or 32-bit memories

• byte lane write control

• external bus interface

7.2.3   SDRAM Interface Controller

The SDRAM interface also known as Dynamic Memory Controller (DMC) has one port connection which is connected to the AHB system bus. This connection interfaces to the main SDRAM control engine and the External Bus Interface. The SDRAM control engine generates an efficient sequence of commands, to issue to the SDRAMs to transfer the requested data. A block diagram of the memory interface is depicted in Fig.3 on page 16 . The SDRAM controller provides the following features:

• Support for four banks of external SDRAM

• The width of each SDRAM bank can be either 8 or 16 bits

• Fast page-mode access support

• Byte, half word and word transaction support

• SDRAM refresh controller using CAS-before-RAS (CBR) refresh, hidden refresh or RAS-only refresh

• Auto pre-charge SDRAM accesses

• Shutdown mode where all SDRAM accesses (including refresh) are disabled. This state is compatible with self-refresh devices.

• Power down mode where all SDRAM accesses are disabled and all SDRAM control lines are driven low. This mode can be used to remove supply power from the SDRAM devices.

7.2.4   Internal Memory Controller

The internal memory is made up of two units, a bank of SRAM and one bank of ROM. The internal memory controller allows the wait states of the SRAM and ROM to be controlled.

Embedded SRAM:

• 64KByte embedded SRAM (16K x 32)

• Supports byte, half-word and word access.

Embedded ROM:

• 256KByte embedded program ROM (64K x 32)

7.2.5   FLASH MEMORY CONTROLLER

The FLASH memory controller takes care of programming, erasing and reading the internal 384 KB FLASH memory.

7.2.5.1   FLASH reads

Any reads from the FLASH via the AHB will be handled automatically by the slave interface using the programmed number of wait states from the 'RdWaitCycles' of the FLASHWS register.
If the FLASH interface is in write mode when the AHB FLASH read is attempted then an abort will be generated. This means that the FLASH can't be programmed when the CPU is running code from FLASH. The default mode is FLASH read, with 8 wait states. Any writes to this area of the SSA memory map will generate an abort.

7.2.5.2   Erasing the FLASH block

Erasing (part) of the flash is necessary if the FLASH already contains data on the location that needs to be written. Erasing can be done in two ways: sector erase and mass erase.

7.2.5.3   Programming the FLASH block

Before writing words into the FLASH, ensure that the respective addresses are empty (erased). Writing to a non-empty address will result in invalid data on that location.

7.2.5.4   Operating conditions

Reading from the FLASH ROM can be done at any AHB speed. The number of programmed wait states is: 42 ns/ <HCLK cycle time>, rounded upwards. The number of CPU cycles for each read is 1+ <programmed WS>

The maximum bus speed for programming and erasing the FLASH is 48 Mhz;
A Mass erase takes 101 ms at this speed
A Sector erase takes 21 ms at this speed
Programming a word takes 41 us at this speed

The minimum bus speed for programming is 24 Mhz.Programming/erasing at 24 Mhz takes twice as long as programming at 48 Mhz.

The maximum time a sector can be accessed in write mode is 60 ms. If this is more, the data in this sector can be corrupted. Software must take care not to exceed this value. Using speed optimized code for FLASH-writes is recommended to keep programming time as short as possible.

7.3   Interrupt Controller

Refer to the application note "Interrupt handling SAA7750".

OVERVIEW

The interrupt controller has the following features:

1. Status information about the interrupt source.

2. Separate enabling and disabling of interrupt sources.

3. Polarity and mode (edge/level) controlled interrupt source.

4. Software programmable FIQ/IRQ interrupt source.

7.3.1   FUNCTIONAL DESCRIPTION

The interrupt controller provides a simple software interface to the interrupt system. Certain interrupt bits are defined for the basic functionality required in any system, while the remaining bits are available for use by other devices in any particular implementation.

The ARM720T processor within the SAA7750 supports two levels of interrupts:

1. FIQ (Fast Interrupt Request) for fast, low hardware latency interrupt handling.

2. IRQ (Interrupt Request) for more general interrupts.

For the FIQ only a single source should be in use at any particular time. This interrupt provides a true low-latency interrupt, because a single source ensures that the interrupt service routine may be executed directly without the need to determine the source of the interrupt. It also reduces the interrupt latency because the extra banked registers within

the ARM720T core, which are available for FIQ interrupts, may be used to maximum efficiency by preventing the need for a context save. For SAA7750 a multiple FIQ will be used to have more flexibility without a great loss of latency.

The IRQ interrupt controller uses a bit position for each different interrupt source. Bit positions are defined for sources like, communication channels, timers, clock, etc. shared in different IRQ registers.

The interrupt controller is not based on hardware priority or does not provide any kind of interrupt vectors, because these functions can be provided in the software.

7.4   Power Management Unit (PMU)

The Power Management and Reset Unit contains logic and registers used to support power management and to control the reset behaviour of SAA7750. The power management mode can be divided into:

- Operating mode
- Stand-by mode
- Power down mode

The Power Management Unit can take care off a proper power-up and power-down sequence under software control. All peripherals are controlled by the CPU. The CPU will request a power-down of a certain peripheral. After power-down the PMU will acknowledge the power-down request to the ARM.

To power-up a device, the ARM will request this to the PMU. The PMU takes care of the sequence and as soon as the peripheral is ready to receive data, the PMU will acknowledge the ARM for having a powered up peripheral.

If all peripherals are in powered down, including the ARM itself (power-down mode), the system can be wake-up via the PMU. As soon as an external interrupt occurs, a wake-up signal is asynchronously send to the PMU. This causes the PMU the enable all clocks of all peripherals and the clock of the CPU followed by generating an interrupt to the interrupt controller. The CPU will return to the last entered mode and all peripherals which aren't used in this specific mode can be disabled on request by the CPU.

7.4.1   FUNCTIONAL DESCRIPTION

The Power Management Unit can be divided into 5 main modules (Fig.4), clock generation module, register module, clock block, reset module and the power down module. The clock generation module serves all the derived clocks from the two master input clocks with internal PLL modules. All the outputs are controlled by the clock register module to enable or disable clocks in the main system. After selecting the clocks, the clock block takes care of hardwired overruling (enabling/disabling) which is only needed in testmode or evaluation modes. The reset module controls the resetting of blocks in the right way. The power down module controls the request/acknowledge mechanism to the CPU and can control the clock register as well, e.g. switching modules in a certain sequence. The register module takes care that the arm can write and read to the registers.
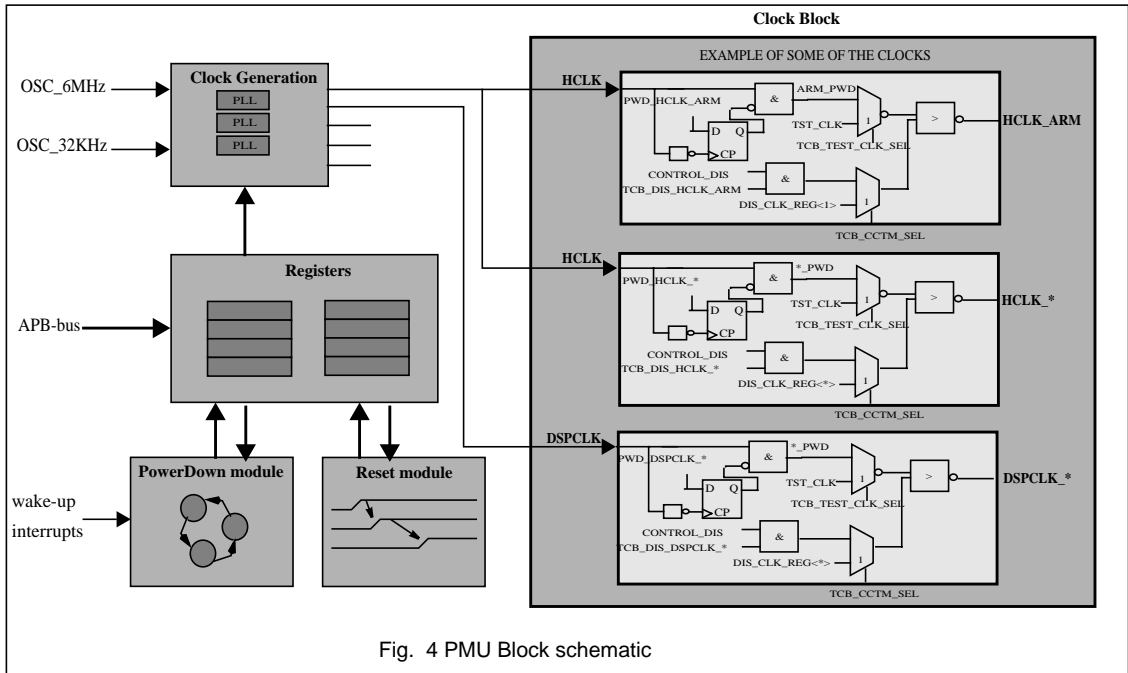
Fig. 4 PMU Block schematic

7.4.2 WAKE-UP BEHAVIOUR

If the system is setup properly, all the clocks can be shutdown. In this state the SAA7750 does consume minimum power. Only the RTC is running (if enabled). The asynchronous part of the GPIO can receive a wake-up signal. This will trigger the PMU to enable all clocks and to generate a wake-up interrupt to the ARM. The ARM should read the interrupt register to determine that there was a wake-up interrupt. Related to this interrupt, the ARM needs to read the GPIO interrupt register to determine who woke up the ARM and to handle the corresponding request. The following peripherals can wake-up the complete system:

- GPIO pins (15:0), and GPIO pins (20,21,22,25,26)
- IIC slave interface
- External wake-up pad
- RTC Alarm
- CD-Block decoder
- Uart
- Remote control
- USB interface

7.4.3 WATCHDOG BEHAVIOUR

The watchdog has the functionality of resetting the complete system due to an external disturbance. In that case, it is unknown which peripherals caused the lock, so the complete system needs a reset. In normal mode, the ARM will rewrite

the watchdog timer before it has timed out. As soon as the ARM is unable to rewrite the counter, the watchdog will request a reset to the PMU. The PMU will reset the complete system, including all the peripherals and switch on all the clocks. The power-on-reset (POR) bit will not be set and a watchdog request bit is set in the PMU. There is no interrupt generated, because the interrupt controller has been reset.

7.4.4   PAUSE BEHAVIOUR

The pause behaviour is implemented as part of the standby mode. By enabling this bit, it is possible to keep all the peripherals running and just prevent the ARM fetching instructions out of the memory. The pause can be retrieved by either pressing the hardware reset or if any peripheral generates an interrupt (and the interrupt is enabled). The interrupt controller will generate an IRQ or FIQ interrupt which is routed to both the ARM and PMU. The PMU will release the pause bit and the ARM will start with the corresponding interrupt handler.

7.5   Reset module

Using the MODE<2> and MODE<1> pins, the boot mode of the SAA7750 can be set accoring to the following settings:

Table 3

MODE<2>	MODE<1>	DESCRIPTION
0	0	Download mode => can be used for debugging
0	1	start executing from INTERNAL FLASH memory after initialisation and Remap according to the internal ROM boot code
1	0	start executing from EXTERNAL ROM memory after initialisation and Remap according to the internal ROM boot code
1	1	NO Remap will be done

7.6   Oscillators and clock generation

Refer to the application note "Clock and PLL settings in the SAA7750".

7.6.1   Overview clock generation module

The clock generation module contains logic for generating all clock signals required in SAA7750. The clock generation module consists of oscillators, PLL-based system clock synthesizers and dividers for generating several different clock signals required by the other internal modules and a clock multiplexer to select between the generated clock from the different inputs. The clock generation module is controlled by the PMU registers for enabling/disabling clocks, PLL settings and clock selection control.

7.6.2   Functional Description

The clock generator contains two oscillators, one oscillator of 32.768kHz (for real time clock) and one oscillator of 6 MHz. These oscillators are the base frequency for generating the rest of the main system frequencies.

Other system clocks will be generated by three PLL's:

- One MASTER PLL to generate the clock frequency of 96MHz/48MHz/24MHz/12MHz and 64MHz/32MHz/16MHz/8MHz/4MHz/2MHz/1MHz/500kHz/250kHz/125kHz/62.5kHz/31.25kHz

- One Audio PLL to generate the 256fs and 128fs clock with the sample frequency's 64kHzor 88.2kHz or 96 kHz. These frequency's can be divided by 1, 2, 4 or 8 to get other audio frequency(32kHz, 16kHz, 8kHz or 44.1kHz, 22.05kHz, 11.025kHz or 48kHz, 24kHz, 12kHz).

- One DSP PLL to generate other clock frequency for the ARM core or DSP core in the range from 8.5Mhz to 133MHz

The clock mux is a multiplexer which select depending on the control signals which of the input clock will be connected to the output clock.

The register block takes care that the ARM can control what the frequency of the ARM, DSP and audio part will be and what will be the source of the clock signals.

Note: the maximum speed of teh ARM core is 72MHz. The maximum frequency of the bus-clock and memory interface bus is up to 48MHz.

7.7   Multi Media Card Interface (MMC)

The Multimedia Card Interface (MCI) is an advanced microcontroller bus architecture (AMBA) compliant peripheral.

The multimedia card system provides communications and data storage, and consists of:

- A multimedia card stack. This can consist of up to 30 cards on a single physical bus.
- A multimedia card controller: This is the multimedia card master, and provides an interface between the system bus and the multimedia card bus.

The multimedia cards are grouped into three types according to their function:

- Read Only Memory(ROM) cards, containing a preprogrammed data.
- Read/Write(R/W) cards, used for mass storage.
- Input/Output(I/O) cards, used for communication.

The multimedia card system transfers commands and data using three signal lines:

- CLK: One bit transferred on both command and data lines with each clock cycle. The clock frequency varies between 0 MHz and 20 MHz.
- CMD: Bidirectional command channel that initializes a card and transfers commands. CMD has two operational modes, first mode 'open drain' for initialization and second mode 'push-pull' for command transfer.
- DAT: Bidirectional data channel, operating in push-pull mode.

7.7.1   Choice of flash memory cards

There are two different types of FLASH memory: NAND FLASH and NOR FLASH. The two FLASH types lend themselves to different applications. Basically NOR FLASH is a replacement for EPROM. NAND FLASH is a magnetic media replacement, particularly suited to serial data.

Today's FLASH cards are give in the table below.

Table 4   Available FLASH Cards.

FLASH CARD	VENDOR	VCC	CAPACITY	INTERFACE	SIZE	SAA7750
Solid State Floppy Disk Card (SSFDC) or SmartMedia Card	Toshiba & Samsung	2.7-3.6V and 5V	...32 MByte available 64 MByte Q2 1999	DOS file system ATA (22-pins)	37 x 45 x 0.76	EBI
Multi Media Card (MMC)	Hitachi Ltd. & Infineon Technologies (open standard)	2.7-3.6V	16 MByte available 32 MByte Q3 1999 64 MByte 2000 128 MByte 2001	SPI (7-pins)	32 x 24 x 1.40	MMC interface
Memory Stick (MS)	Sony (license needed)	2.7-3.6V	...8 MByte available 16 MByte Q2 1999	Serial (10-pins)	50 x 21.5 x 2.8	no
Compact Flash Card (CFC)	SanDisk Corp.	3.3V/5V tolerant	...64 MByte available	DOS file system ATA (50-pins)	42 x 36 x 3.3	no

The Pinning of the Multi Media Card is given in table 5

Table 5   Pinning of the Multi Media Card (3 pins)

PIN	SYMBOL	DESCRIPTION
A4	MCI_DAT	Data
A2	MCI_CLK	Clock
B2	MCI_CMD	Command/Response

7.8   10-bit ADC

Refer to the apllication note "the build-in 10 bits ADC of the SAA7750".

OVERVIEW

This section specifies the ADC interface, which can be used e.g. for observing battery voltage and/or scanning resistive key's. The interface can be divided into 2 main modules, a 10 bit A/D converter and an ADC controller/multiplexer. The A/D converter is a 10 bit successive approximation analog to digital converter.

The basic characteristics of the ADC interface module are:

- Eight analog input channels, selected by an analog multiplexer;
- Programmable ADC resolution from 2 to 10 bits;

  Converted digital values are stored in a 2 * 10 bits register;
- Maximum conversion rate is 400Ksamples/s in 10bits accuracy and 1500Ksamples/s in 2 bits accuracy mode.
- Single A/D conversion scan mode and continuous A/D conversion scan mode;
- Power down mode.

7.8.1   FUNCTIONAL DESCRIPTION

The ADC is able to convert on of its 8 inputs from analog to digital in 10 bits with a conversion rate of 400Ksampls/s . The resulution can be reduced till 2 bits and in that case the conversion speed can be increased to 1500Ksamples/s. The ADC is composed of an analog 8:1 multiplexer to select the input to convert. One 10 bits conversion requires 11 ADC clock cycles to complete. During the first cycle the selected input is sampled, in the next 10 cycles the sample is converted into 10 bits.

7.8.2   MULTI CHANNEL A/D CONVERSION SCAN

Associated to each analog input channel is a set of two 10 bits result registers for storage of A/D conversion result. It is programmable which channels are included and which channels are excluded from the A/D conversion scan process. The A/D conversion scan process can be started by software.

There are two scan modes, 'Continuous Scan' mode and 'Single Scan' mode:

- In 'Continuous Scan' mode, A/D conversion scans are carried out continuously: once one scan completed, the next one is started automatically.
- In 'Single Scan' mode, only a single conversion scan is carried out, the next scan must be started explicitly by software.

7.8.3   ADC RESOLUTION

The resolution within the AD conversion process is software programmable through ADC controller variables. The resolution can be adjust between 2 and 10 bits.

The conversion rate is computed as follows:

$$conversionrate = \frac{clockfrequency}{(resolution + 1)}$$

7.8.4   INTERRUPTS

The ADC interface implements one interrupt, a scan interrupt which indicates the completion of an A/D conversion scan process and the validity of the data in the result registers.

7.9   UART

The UART can be used for connecting a Modem, Blue tooth IC or a terminal emulator to the SAA7750 IC.

Overview:

- 16 word wide transmit and receive FIFO's
- Supports external modem peripheral
- Optional interface to external Philips Smartcard
- Build-in IrDA receiver

7.9.1   FUNCTIONAL DESCRIPTION

The receiver block, Rx, monitors the serial input line, SIN, for valid input. The Rx Shift Register (RSR) accepts valid characters via SIN. After a valid character is assembled in the RSR, it is passed to the Rx Buffer Register FIFO to await access by the CPU or host via the generic host interface.

The transmitter block, Tx, accepts data written by the CPU or host and buffers the data in the Tx Holding Register FIFO (THR). The Tx Shift Register (TSR) reads the data stored in the THR and assembles the data to transmit via the serial output pin, SOUT.

The Baud Rate Generator block, BRG, generates the timing enables used by the Tx block. The BRG clock input source is either the APB clock, PCLK, or the UART clock, UCLK. The main clock is divided down per the divisor specified in the DLL and DLM registers. This divided down clock is a 16x oversample clock, NBAUDOUT.

The modem interface contains registers MCR and MSR. This interface is responsible for handshaking between a modem peripheral and the UART.

The interrupt interface contains registers IER and IIR and controls the interrupt output pin, INTR. The interrupt interface receives several one clock wide enables from the Tx, Rx and modem blocks.

Status information from the Tx and Rx is stored in the LSR. Control information for the Tx and Rx is stored in the LCR.

The build-in IrDA block can be enabled or disabled. If disabled, the UART signals pass through this block unchanged . The build-in IrDA block operates over the entire range of 2.4kb/s up to 115.2kb/s.

7.10   General Purpose I/O

The General Purpose Input-Output (GPIO) module provides 16 external GPIO pins which can be independently programmed to be input or output. This means that each pin has a data input/output bit, a data direction bit and a value bit. The GPIOs can be used e.g. like push-buttons and detection switches.

- A maximum of 28 General Purpose pins externally
- Each General Purpose pin has an interrupt which can be dynamically configured:
  - active high or low polarity
  - edge or level sensitive
  - masked or enabled

7.10.1   FUNCTIONAL DESCRIPTION
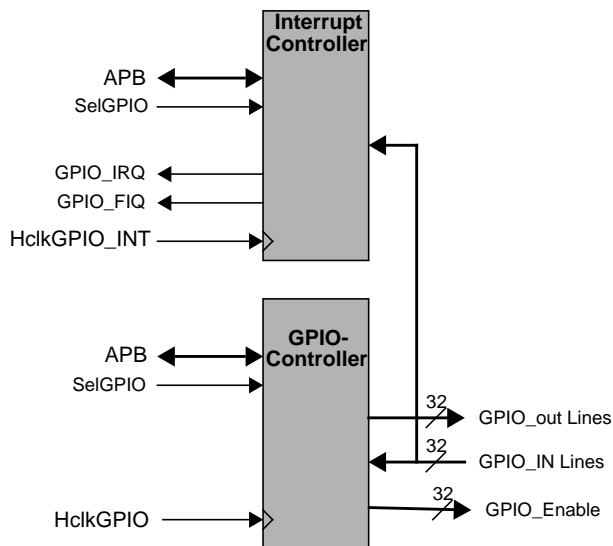
The GPIO module consists of two parts: The GPIO controller which functions as an input/output interface to the GPIO lines, and an interrupt controller which checks the GPIO-lines for level and/or edge changes and generates an interrupt to the CPU.

7.10.2 INTERRUPTS

Refer to the application note "Interrupt handling SAA7750".

Each GPIO line can be configured to generate interrupts either as an FIQ or an IRQ. They can be level or edge sensitive and either low/high active or Positive/negative edged.

The interrupt controller contains a raw status register where each GPIO line can be checked for an interrupt, independent of masking. It contains a Status register, which contains the values after masking.

7.11   Real Time Clock (RTC)

- Measures passage of time to maintain calendar and clock
- Uses external 32.768kHz crystal
- Counts seconds, minutes, hours, days, and years with leap year correction
- Counter increment interrupt
- Alarm clock interrupt

The Real-Time Clock (RTC) module consists of a counter which increments at a frequency of typically 32.768kHz. The RTC provide a set of counters to measure time during power on and power off operation. It is designed to use little power consumption in power down mode.

7.11.1 FUNCTIONAL DESCRIPTION

The RTC interfaces to a standard APB with either a unidirectional or bidirectional data bus. The data bus is 32-bits wide while the consolidated time registers are included to read all time counters with only three read operations.

The RTC uses a 32.768 kHz clock, that is divided down to a 1 Hz clock using a ripple counter. A ripple counter is used to minimize power during power down mode.

The counter clock consists of the exclusive-or of the 1 Hz clock and the counter write strobe. During a non-write operation the counters operate as a set of sequential counters clocked by the 1 Hz clock. During a write operation no event is allowed on the 1 Hz clock. To insure this condition is true the user should disable the 1 Hz clock by setting the clock enable bit (CR[0]) to zero before writing to the RTC.

Each counter has its count enable gated so that during a counter write operation no counter is increment by the clock pulse generated by the write strobe. Two of the counters have dynamic maximum values, the Day of Month counter and the Day of Year counter. These maximum values are determined via combinational logic whose inputs are the Year counter (for leap year) and Month counter.

For determining a leap year, the RTC does a simple bit comparison to see if the two lowest order bits of the year counter are zero. If true, then the RTC considers that year a leap year. A more accurate algorithm would prevent years evenly divisible by 100, but not evenly divisible by 400, from being leap years (the year 2000 is a leap year, but 2100 is not). The RTC considers all years evenly divisible by 4 as a leap year. This algorithm will be accurate until the year 2100.

7.11.2 INTERRUPTS

Interrupt generation is controlled through the Counter Increment Interrupt Register (CIIR), the alarm registers, and the Alarm Mask register (AMR). Interrupts are generated only by the transition into the interrupt state. Each bit in CIIR corresponds to one of the time counters. If CIIR is enabled for a particular counter, then every time the counter is increment an interrupt is generated. The alarm registers allow the user to specify a date and time for an interrupt to be generated. The AMR provides a mechanism to mask alarm compares. If all non-masked alarm registers match the value in their corresponding time counter, then an interrupt is generated.

7.11.3 POWER DOWN OPERATION

When the external signal pwr_up is active low the RTC goes into power down mode. In power down mode all bus interface inputs are gated. Besides the first element in the ripple counter, and the optional alarm clock sampling flip flop, all loads to the 32.768 KHz clock are gated to reduce power.

The user can optionally specify that the alarm compare interrupt output should remain active in power down mode to allow for a power-on timer. If this option is selected, the alarm registers are included in the low power section of the design. When powered down, the synchronizing clock for alarm comparison will be the 1 Hz clock. When powered up, the bus clock is used to synchronize the alarm.

7.12 Timers

- Two independent 32-bit timers
- Can be programmed to interrupt the processor
- Can operate in either free running or periodic timer mode

The timer block contains two fully independant timers, where each timer has its own clock and chip-select. Each timer is a 32 bit wide down-counter with selectable pre-scale. The pre-scaler allows either the system clock to be used directly, or the clock divided by 16 or 256 may be used. This is provided by 0, 4 or 8 stages of pre-scale. Two modes of operation are available, free-running and periodic timer. In periodic timer mode the counter will generate an interrupt at a constant interval. In free-running mode the timer will overflow after reaching its zero value and continue to count down from the maximum value.

Note: The timer speed depends on the system clock. The system clock can change depending the operation mode, which means that the timer can not be used as a real time clock.

7.12.1 FUNCTIONAL DESCRIPTION

The timer is loaded by writing to the Load register and then, if enabled, the timer will count down to zero. On reaching a count of zero an interrupt will be generated. The interrupt may be cleared by writing to the Clear register.

After reaching a zero count, if the timer is operating in free-running mode then the timer will continue to decrement from its maximum value. If periodic timer mode is selected then the timer will reload from the Load register and continue to decrement. In this mode the timer will effectively generate a periodic interrupt. The mode is selected by a bit in the Control register.

At any point the current timer value may be read from the Value register.

At any point the timer_load may be re-written. This will cause the timer to restart to the timer_load value.

the timer is enabled by a bit in the control register. At reset the timer will be disabled, the interrupt will be cleared and the Load register will be undefined. The mode and pre-scale value will also be undefined.

The timer clock is generated by a pre-scale unit. The timer clock may be the system clock, the system clock divided by 16, which is generated by 4 bits of pre-scale, or the system clock divided by 256, which is generated by a total of 8 bits of pre-scale.

7.12.2   INTERRUPTS

The timer is loaded by writing to the Load register and then, if enabled, the timer will count down to zero. On reaching a count of zero an interrupt will be generated. The interrupt may be cleared by writing to the Clear register

7.13   Watchdog Timer

The watchdog block is of similar design to the existing timer block, except that in stead of interrupting the CPU, it provides a reset request to the PMU and that it consists of only one timer.

Once the watchdog is enabled, it will monitor the programmed timeout period and generate a reset request when the period expires. In normal operation the watchdog is triggered periodically, resetting the watchdog counter and ensuring that no reset is generated. In the event of a software or hardware failure preventing the CPU from triggering the watchdog, the timeout period will be exceeded and a reset requested from the PMU/reset control logic.

The reset request allows the PMU to select a default set of clocks, and reset the CPU subsystem.

7.13.1   FUNCTIONAL DESCRIPTION

The functional description is the same as that of the timer block. Mind that the watchdog only contains one timer!

7.13.2   INTERRUPTS

The watchdog timer is loaded by writing to the Load register and then, if enabled, the timer will count down to zero. On reaching a count of zero an interrupt will be generated to the PMU.

7.14   IIC master Interface

The I^{2}C master module provides a serial interface that meets the I^{2}C bus specification and supports all transfer modes from and to the I^{2}C bus. It supports the following functionality:

- It supports both the normal mode (100 kHz SCL) and the fast mode (400 kHz SCL).
- It has word (32-bits) access from the CPU side.
- Interrupt generation on received or sent byte (and some special cases).
- It has two modes of operation: master transmitter and master receiver.
- 16 8bits word wide transmit and receive FIFO's

Important note: since 2 pins of the master IIC interface are shared with the pins of the CD-block decoder UART, the IIC master interface cannot be used in cases in which the CD-block decoder UART is used!

7.14.1   FUNCTIONAL DESCRIPTION

The main features of the I^{2}C-bus are:

- Serial clock synchronization allows devices with different bit rates to communicate via one serial bus

- Serial clock synchronization can be used as a handshake mechanism to suspend and resume serial transfer

- The I^{2}C bus may be used for test and diagnostic purpose

Two wires, SDA (serial data) and SCL (serial clock) carry information between devices connected to the I^{2}C bus. Each device can operate as either a transmitter or receiver, depending on the function of the device. In addition to transmitters and receivers, devices can also be considered as masters or slaves when performing data transfers. A master is the device which initiates a data transfer on the bus and generates the clock signals to permit that transfer. Any device addressed by a master is considered a slave.

Generation of clock signals on the I^{2}C bus is always the responsibility of the master device; each master generates its own clock signals when transferring data on the bus. Bus clock signals from a master can only be altered when they are stretched by a slow-slave device holding down the clock line, or by another master when arbitration occurs.

7.14.2   INTERRUPT

Active low signal indicates if an interrupt is pending. The reason for the interrupt is encoded in Status Register. There are several possible interrupt types: transfer completed, arbitration failure, missing acknowledge, need more data, Tx FIFO has room for more data, or data has been received.

7.15   IIC slave Interface

The I^{2}C Slave module provides a serial interface that meets the I^{2}C bus specification and supports all transfer modes from and to the I^{2}C bus. It supports the following functionality:

- It supports both the normal mode (100 kHz SCL) and the fast mode (400 kHz SCL).

- It has word (32-bits) access from the CPU side.

- Interrupt generation on received or sent byte (and some special cases).

- It has two modes of operation: slave transmitter and slave receiver.

7.15.1   FUNCTIONAL DESCRIPTION

The main features of the I^{2}C-bus are:
- Serial clock synchronization allows devices with different bit rates to communicate via one serial bus

- Serial clock synchronization can be used as a handshake mechanism to suspend and resume serial transfer

- The I^{2}C bus may be used for test and diagnostic purpose

Two wires, SDA (serial data) and SCL (serial clock) carry information between devices connected to the I^{2}C bus. Each device can operate as either a transmitter or receiver, depending on the function of the device. In addition to transmitters and receivers, devices can also be considered as masters or slaves when performing data transfers. A master is the device which initiates a data transfer on the bus and generates the clock signals to permit that transfer. Any device addresses by a master is considered a slave.

Generation of clock signals on the I^{2}C bus is always the responsibility of the master device; each master generates its own clock signals when transferring data on the bus. Bus clock signals from a master can only be altered when they are stretched by a slow-slave device holding down the clock line, or by another master when arbitration occurs.

7.15.2   INTERRUPT

Active low signal indicates if an interrupt is pending. The reason for the interrupt is encoded in Status Register. There
are several possible interrupt types: transfer completed, arbitration failure, missing acknowledge, need more data, Tx
FIFO has room for more data, or data has been received.

7.16   LCD Interface

The LCD interface contains logic to interface to a 6800/8080 compatible LCD controller. The LCD interface is compatible
with the 6800 bus standard and the 8080 bus standard, with one address pin (RS) for selecting the data or instruction
register.

The LCD interface contains a couple of options to delay the access on the 6800/8080 bus, if the specific controller
requires it.

7.16.1   INTERFACE

- 8/4 bit parallel interface mode: 6800-series, 8080-series
- Supports multiple frequencies for the 6800/8080 bus, to support high and low speed controllers
- Supports a maximum of 16 wait states on lcd-bus actions
- Supports polling the busy flag from LCD controller to off-load the CPU from polling
- Contains an 16 byte FIFO for sending control and data information to the LCD controller
- Contains a serial interface which uses the same FIFO for serial transmissions.
- Contains maskable interrupts.

7.16.2   SYSTEM INTERFACE

Table 6   Various modes of the LCD interface

PS	MI	IF	CSB	RS	RW_W R	E_RD	DB0-3	DB4-	DB5	DB6	DB7
Bus mode (L)	6800-ser ies (H)	8 bit (L)	CSB	RS	nR/W	E	DB0-3	DB4	DB5	DB6	DB7
		4 bit (H)	CSB	RS	nR/W	E	*	DB4	DB5	DB6	DB7
	8080-ser ies (L)	8 bit (L)	CSB	RS	nWR	nRD	DB0-3	DB4	DB5	DB6	DB7
		4 bit (H)	CSB	RS	nWR	nRD	*	DB4	DB5	DB6	DB7
Serial mode (H)	*	*	CSB	RS	*	*	*	*	SCL	SI	SO

Note:

1.   * Don't care ("High", "Low" or "Open")

PS = Parallel/Serial mode
CSB = Chip Select. Default low active
IF = 4 or 8-bit mode
MI = Motorola/Intel mode
RS = Register Select (also seen as A0)

E_RD = Enable / Read. Enable in 6800 mode, Read in 8080 mode.
RW_WR = ReadWrite / WRITE. Read/write in 6800 mode, Write in 8080 mode.
DB(7-0) = Data Bus.
SCL = Serial CLock.
SI = Serial Input.
SO = Serial Output.

7.16.3   RESETTING THE LCD CONTROLLER

Not all LCD controllers require a reset pin to reset the controller. In some cases a simple instruction to the controller is enough to perform the reset.

A GPIO pin, or maybe the system reset can be used to act as a reset signal for the LCD controller, if it requires a hardware reset pin.

7.16.4   OPERATIONAL MODES

The LCD_interface has three modes for outputting data: Byte-mode, 4-bit mode and serial-mode.

Byte mode:

At each shift of the FIFO, the last byte from the FIFO will be put on the data pins, and pin RS will indicate if the data is an instruction or data value.
In read mode the data on pins DB_IN 7-0 will be sampled by the LCD_interface.

4-bit mode:

At each shift of the FIFO, the last byte from the FIFO will be split, where the order depends on the 'MSB_first' from the control register.
When set to '1', bit 7-4 from the FIFO byte will be put first, or read first, at the data pins, and then bit 3-0.
When set to '0' bits 3-0 will be written or read first, and then bits 7-4.

7.16.5   SERIAL MODE:

At each shift of the FIFO the last FIFO byte will be split in 8 separate bits and be put on data pin 7, where the order depends on the 'MSB_first' from the control register.
When set to '0', then first bit 0 and last bit 7 will be written or read first, else the order is from 7 downto 0.
Signal RS is included for each 8 bits and indicates a instruction or data. Not all controllers require this signal in serial mode, but can be used if required.

7.16.6   LOOPBACK MODE

Setting the register 'LOOPBACK' of the CONTROL register to '1', will set the LCD interface in loopback-mode.

Internally, the LCD data output is connected to the LCD data input. The programmer can test correct behaviour of the LCD interface, by doing the following:

• Place the LCD interface in parallel, 8-bit mode

• Write a single byte to the LCD_DATA_BYTE register

• Write '0x01' to the LCD_READ_CMD register to request a bus read

• Poll the status bit, or wait for the 'valid' interrupt (if MASK is cleared)

• If valid, read the byte from LCD_DATA_BYTE register

• Compare this value with the written value

7.16.7   INTERRUPT

An interrupt is generated on the following occasions:

• When the FIFO is empty (LCD_FIFO_EMPTY).

• When the FIFO is half empty. (LCD_FIFO_HALF_EMPTY)

• When the FIFO is overrun. (LCD_FIFO_OVERRUN)

• When the requested instruction/data register is valid. (LCD_READ_VALID)

Any of these interrupts can be masked individually to keep them from generating an interrupt to the CPU, by using the LCD_INT_MASK register. The interrupts after masking can be read in the LCD_STATUS register. Writing a '1' in the mask register will mask the interrupt. The status of the interrupts without masking can be read in the 'LCD_INT_RAW' register

Clearing the interrupts:

An interrupt can be cleared by writing a '1' to the respectable bit in the LCD_INT_CLR register. If the interrupt has not been solved, for instance the FIFO is still empty, this will re-set the interrupt, when not masked.

7.17   Remote Control Interface

The remote control build into the SAA7750 is based on the discharge of a RC combination. Making different RC combinations, key's can be identified.

Key interface - Consists of DO and DI signals.

The key interface consists of an output pin, DO, which is a signal of high and low periods, similar to a clock.

Output pin, DO, when low, is used to discharge an RC network.

Input pin, DI, is sampled by the block to determine the time DI remains low.

NOTE: A sample is the time DI pin i

s low. Normally DO is used to discharge a RC network. DO going low will discharge the Capacitor and the time it takes to recharge is monitored by DI.

7.18   USB Interface

7.18.1   OVERVIEW

The SAA7750 USB interface can be used for:

- Down load bulk audio data (compressed) from a PC to the application with the SAA7750
- Download new firmwarde from a PC into the build-in program FLASH
- Up load speech or audio from a (analog) source to a PC

7.18.2   FUNCTIONAL DESCRIPTION

The USB interface for SAA7750 is a full speed USB interface (12Mbits/s) and is USB 1.1 compliant. It consist of an analog transceiver (ATX), and a Full Speed USB module (FS22).

- USB 1.1 compliant interface
- Supports bus-powered or self-powered operation (programmable)
- One full duplex control end points (8 bytes)
- Two full duplex interrupt end points (16 bytes)
- One full duplex bulk end point (64 bytes, double buffered)
- One full duplex isochronous end point (294 bytes, double buffered)
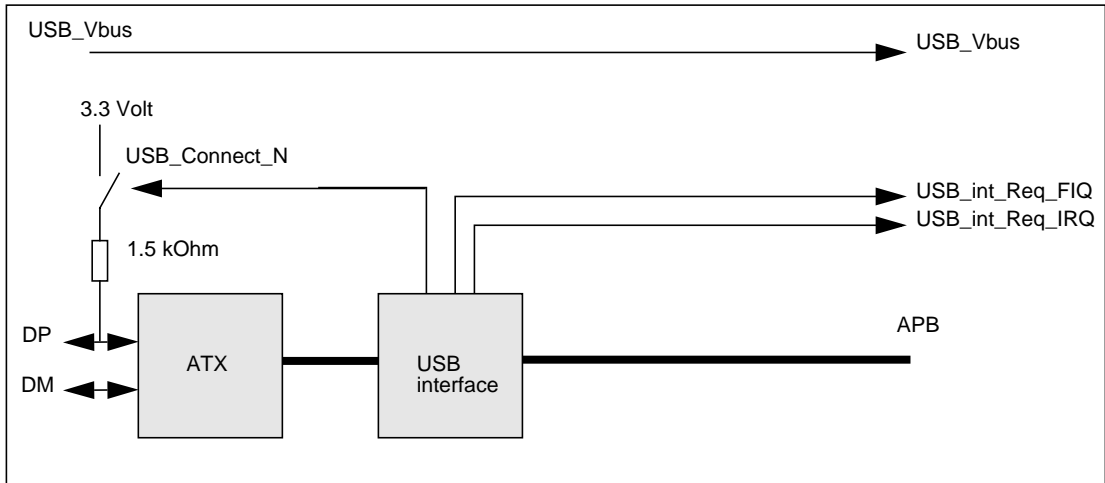
The USB_Connect_N line can be used in the situation where internal initialisation of the USB device is longer than the time needed according to USB specification:

"120ms after detection by the host of a USB device, the USB device should start responding to the transaction on the USB".

A USB_Connect_N line can be used to switch the external pull-up resistor of 1.5kOhm under software control.

7.18.3   INTERRUPTS

There are two interrupts to the system:

- USB_int_req_FIQ
- USB_int_req_IRQ

7.18.3.1   USB_int_req_FIQ

This is the high priority interrupt to the system. The frame interrupt, Bulk OUT interrupt or Bulk IN interrupt can be routed to generate the FIQ. It is a must that this interrupt should have only one source at a time.

7.18.3.2   USB_int_req_IRQ

This is the low priority interrupt to the system. The data transfer for all end points other than the FIQ source is initiated through this interrupt. This interrupt has got multiple sources, sources different from the source that created the FIQ at that point in time.

7.18.3.3   Interrupt handling

When CPU gets FIQ interrupt it does not need to read the status register as there is only one source for it depending on the selection of the FIQ select register. In the service routine CPU has to clear the interrupt by writing '1' into bit position

'0' of intr_clear_register but when it gets the IRQ interrupt it has to read the Interrupt status register and understand which interrupt bit is set. The service routine has to clear the corresponding interrupt. If it is an interrupt from USB core (Bit 1 to Bit 8 of Status register) the clear interrupt command to the USB core must also be executed.

7.18.3.4 *Zero overhead operation*

To read the data without software overhead the CPU has to rely on the end_of_packet interrupt. The CPU can go on reading the receive data register and the read operation is to be terminated when the end of packet interrupt occurs. Hence the software does not have to track how many bytes it transferred. Still the number of bytes information is needed to remove the garbage bytes read.

7.19 CD Block Decoder

The CD decoder block enables the SSA chip to playback from an audio CD or an MP3 CD. Major functionality's of this block include various data interfaces, a minimal block decoder, a buffer manager and an SDRAM controller. External CD engine is controlled through the serial command interface (IIC **or** UART) and disc data comes in through the serial data interface in either IIS or EIAJ format and the subcode interface in either V4 or EIAJ format. The minimal decoder detects sync and frame address and performs necessary error detection and descrambling. The buffer manager maintains read and write pointers and stores data in the data buffer (SDRAM) through the SDRAM controller.

This report provides proposed features and functional descriptions of the CD decoder block. Register addresses are aligned to word (32-bit) boundary to facilitate accesses from the CPU.

7.19.1 FUNCTIONAL DESCRIPTION

7.19.1.1 *Features*

- Support of CD-DA mode, CD-ROM Yellow book and CD-ROM XA.
- CRC Q-subcode error detection.
- EDC C3 error detection to enable optional software correction through use of C2 error flag.
- Scratch pad random access area in SDRAM for use by the CPU.
- CD-DA seamless playback enables Constant-Angular-Velocity (CAV) drive compatibility
- I2C master[1] and UART command interfaces with CD engine. Configurable UART baud rate.
- Support of I2S/EIAJ with error flags.
- V4/EIAJ subcode interface.
- Hardware extraction of formatted Q-channel subcode.
- Support of 16Mbit or 64Mbit SDRAM chips with 8- or 16-bit data bus.
- Clock of the CD decoder block can be stopped and resumed to save power.
- Support of CD-TEXT mode.

Important note: the CD-block decoder has a PRIVITEpath to the SDRAM, having priority on the EBI over the ARM. This means that the CD block decoder can ALWAYS access the SDRAM. It also means that in all applications there MUST be an external SDRAM!

---

(1) An I2C slave also exists in the SSA system for connection to the user interface.
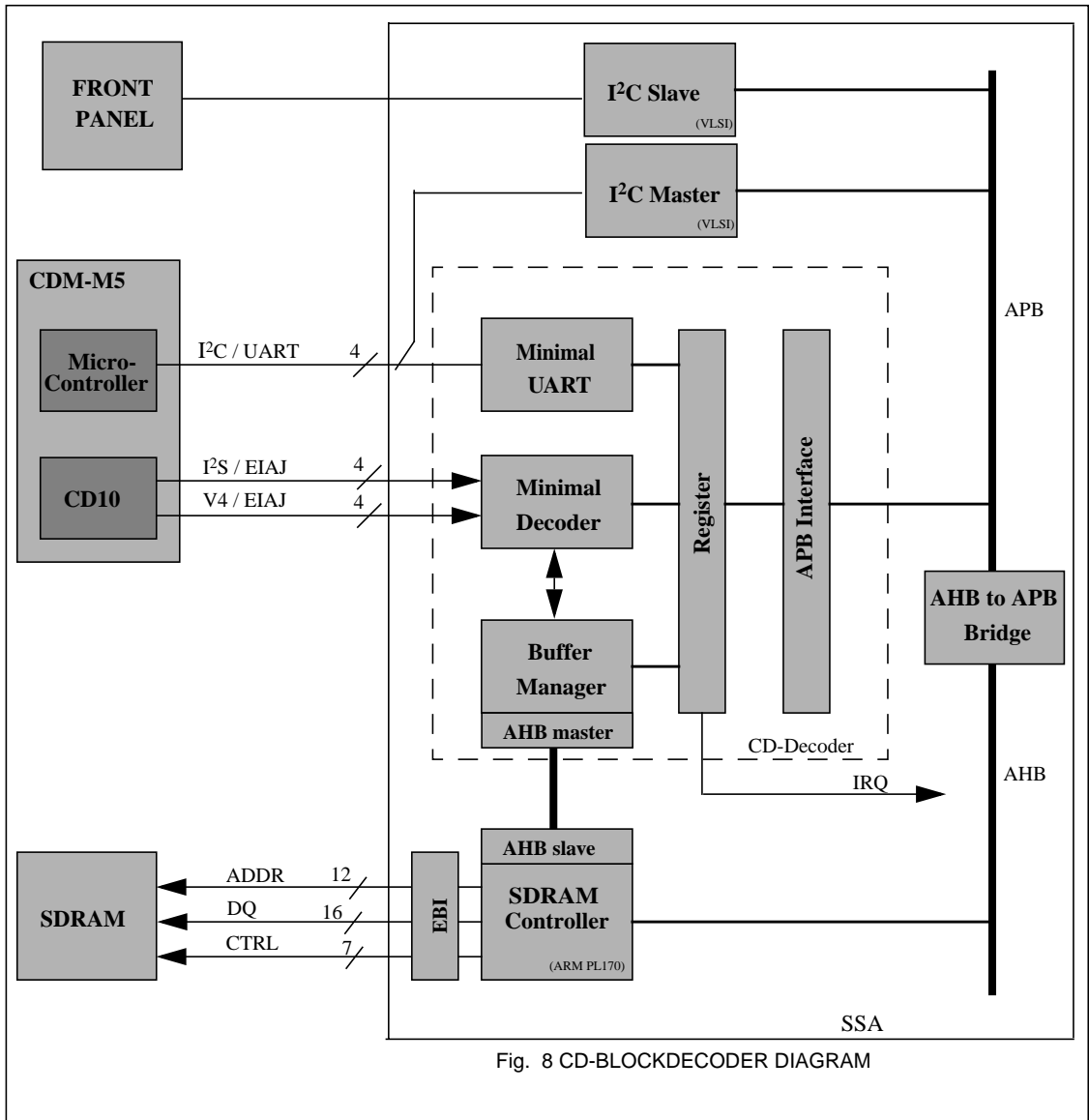
Block Diagram[1]

Fig. 8 CD-BLOCKDECODER DIAGRAM

7.19.2   INPUT/OUTPUT PIN FUNCTION

in table 7, the function of the pins of the CD blockdecoder are mentioned for both the IIC and the UART mode.

(1) CDM-M5 is a Philips CD engine that includes a microcontroller and a CD10 chip.

Table 7 Input/Output Pin Function[1]

PIN NR	MODE 1	MODE 1 PIN NAME	INPUT/ OUTPUT	FUNCTION
	MODE 2	MODE 2 PIN NAME		
R13	IIC	SDA	Input/Output	IIC data I/O line (open-drain output)
	UART	RXD	Input	UART Serial Data Input
P13	IIC	SLK	Input/Output	IIC clock line (open-drain output)
	UART	TXD	Output	UART Serial Data Output
C5	IIC	CRQ	Input	Communication request line
	UART	ENGINE_RDY	Input	CD engine is ready to receive the next frame
D5	IIC	CRST	Output	CD engine reset line
	UART	HOST_RDY	Output	Host is ready to receive the next frame
			Serial Data Interface	
C9	IIS or EIAJ	CLAB	Input	IIS/EIAJ input bit clock
C7	IIS or EIAJ	DAAB	Input	IIS/EIAJ serial data
C8	IIS or EIAJ	WSAB	Input	IIS/EIAJ word clock
D9	IIS or EIAJ	EFAB	Input	IIS/EIAJ error flags
			Subcode Interface	
D8	V4	V4	Input	Versatile pin 4: single wire subcode
	EIAJ	SUB	Input	EIAJ subcode data bits (3 wire)
D6	V4	CFLAG	Input	Absolute time sync
D7	V4	Not used		
	EIAJ	SFSY	Input	EIAJ subcode frame sync (3 wire)
C6	V4	Not used		
	EIAJ	RCK	Input/Output	EIAJ subcode clock input (3 wire)

7.19.3 I^2C Interface

This is the communication interface between the CD decoder block and the CD engine. The CD decoder block represents the I^2C master and communicates with the CD engine. The interface signals consist of the two wires used by I^2C bus-SDA (serial data) and SCL (serial clock), a communication request line $\overline{CRQ}$ and a reset line ($\overline{CRST}$). The $\overline{CRQ}$ line is used by the CD engine to signal a message is ready to be read out. The $\overline{CRST}$ line resets the CD engine. The high-low transition of CD engine insert line $\overline{SENS\_I}$ is used for the detection of CD insertion.

Important note: since 2 pins of the IIC master interface and the CD-block decoder UART are combined, the use of IIC master interface OR the UART are mutually exclusive!

(1) Pin name and function for different modes are listed for shared pins.

7.19.4   Standard Serial Interface UART

The UART serial port in the CD decoder block is a full duplex interface. For each frame, 11 bits are transmitted (through TXD) or received (through RXD): a start bit (logic 0), 8 data bits (LSB first), a parity bit and a stop bit (logic 1). Parity and baud rate can be configured through the UART_CTRL registers (see table 207). Both transmit and receive have a one-byte buffer that can be accessed through the UART_TX and the UART_RX registers, respectively. If enabled, an interrupt is generated at each byte transmitted or received.

Baud rate is controlled through BaudRate field of UART_CONF1 register. It is effective for both transmit and receive. The receive logic is able to tolerate a 5% baud-rate shift, provided that the HCLK frequency is correctly set in UART_CONF2 register.

All bits in UART_STATUS register can be cleared by writing a logic 1 to that bit, which also clears the interrupt associated with that bit.

7.19.5   Subcode Interface

There are two subcode interfaces:

1. One that conforms to "EIAJ CP-2401" (using SBSY, SFSY, RCK and SUB) and can be configured as a 3 wire interface. The interface formats are illustrated in Fig.9.
2. The Philips V4 format on V4 pin as illustrated in Fig.10. The subcode sync word is formed by a pause of 200 ms minimum at nominal speed, where all subcode channels are muted. Each subcode byte starts with a 1 at the P-channel bit position followed by 7 bits (Q to W), with P-channel bits discarded. The gap between two consecutive bytes can vary from 11.3 ms to 90 ms.

The byte alignment is found by searching for a minimum gap of 200 ms (at nominal speed) on the V4 input. The gap is counted against 12 rising edges of WSAB on the serial data interface (IIS or EIAJ). Once a start bit is detected, both rising and falling edges of WSAB, in conjunction with CLAB, are used to synchronize the sampling of subcode data.

The 96-byte subcode data in a subcode frame is buffered in the subcode area of a 3KB buffer block. In addition, the 96-bit Q-channel subcode is duplicated in a separate 12-byte area within the same block. The last 16 bits of the Q-channel subcode are used internally to perform CRC.

Fig. 9 EIAJ Subcode Interface Format

Fig. 10 Philips V4 Subcode Format
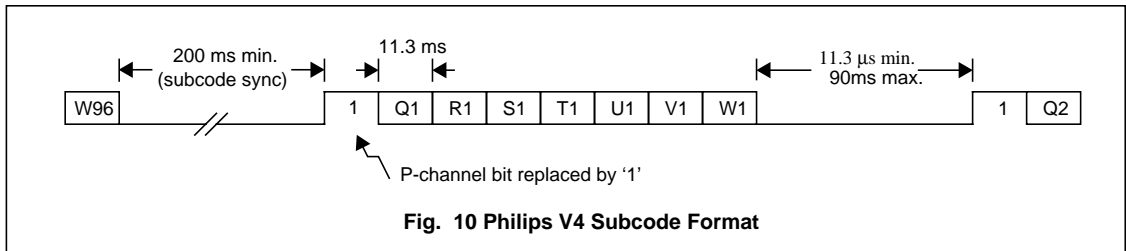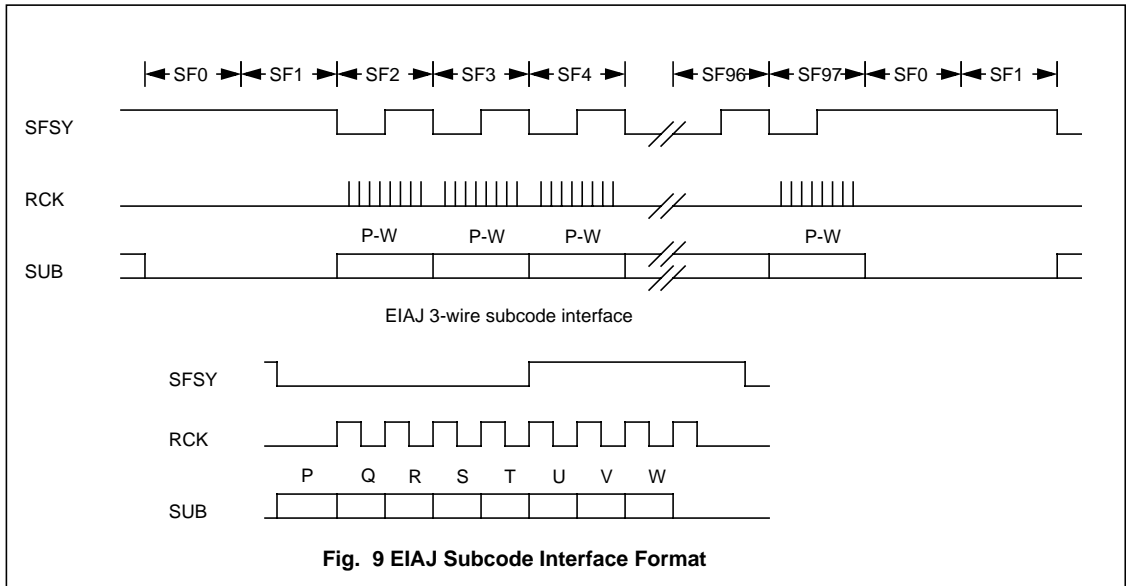
In audio mode, the first flag bit, F1, of the CFLAG input pulses for every block thus defines the block boundary. The flag is also called absolute time sync. The format of CFLAG is illustrated in Fig.11. Note that the audio sampling must always start from the left channel, which follows a falling edge of WSAB in IIS mode or a rising edge of WSAB in EIAJ mode.

Fig. 11 CFLAG Input Timing Diagram
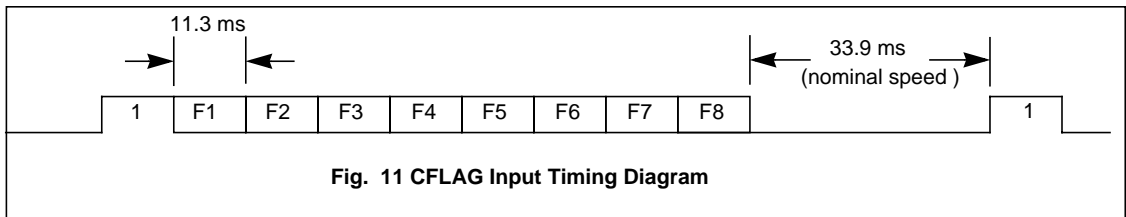
7.19.6 Serial Data Interface

The serial data interface can be switched between two modes: Philips I^{2}S and the EIAJ format. In each case, the serial data is transferred through a 3-wire interface: WSAB (word select), CLAB (serial clock) and DAAB (serial data). The polarity of CLAB can be inverted. The fourth line, EFAB, indicates the C2 error flags that associates with each byte. The error flags are stored in the data buffer and can be used for C3 error correction. For audio mode, EFAB has no meaning as concealment is already performed within the engine. The timing of I^{2}S and EIAJ is illustrated in Fig.12 on page 40 [1].
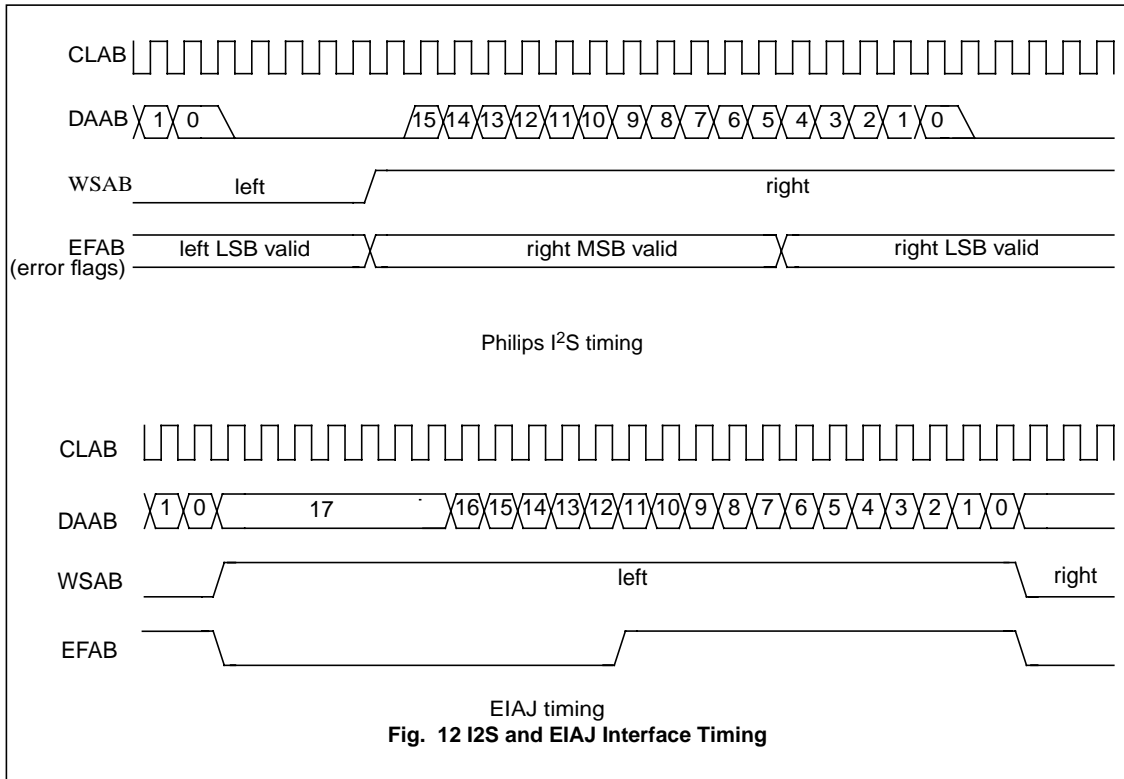

Philips I^{2}S timing

EIAJ timing
Fig. 12 I2S and EIAJ Interface Timing

7.19.7 Minimal Block Decoder

This block accepts data from the serial data interface (I^{2}S or EIAJ) and performs necessary word alignment, synchronization, data descrambling and error detection for the type of data being read.

Four CD formats are supported:

- The CD-DA format for audio playback. The MSF address is embedded in the Q-channel subcode. CRC is performed on the Q-channel data and MSF information is de interleaved. The audio data has no block structure, however, the F1 flag is used to divide the data stream into 2352-byte blocks to facilitate data buffering. Error flags are stored in the data buffer. Data descrambling must be disabled.

- The CD-ROM Yellow Book Mode 1 format. The data frame structure is illustrated in Fig.48. The C3 check bytes, including 172 P-parity bytes and 104 Q-parity bytes, along with the C2 error flags coming through EFAB line, are

(1) The IIS timing shown in Fig.12 is Philips 24-bit format, in which 24 CLAB cycles are contained within each half cycle of WSAB. There are other formats that are being used, such as 16-bit and 32-bit formats. However, for all formats, the 16 data bits are always aligned to the leading edge.

saved in the buffer and can be used for software C3 correction if desired. Data descrambling is performed before EDC. The four EDC bytes are generated from the sync pattern, the header, and the user data. EDC calculation is performed by the minimal decoder to validate these data fields.

- The CD-ROM XA Mode 2 Form 1. The data frame structure is illustrated in Fig.49. The form bit is included in both copies of sub-header. For each block, two form bits are extracted from each copy of sub-header and then compared. A mismatch is marked in the status field of the buffer block and also causes EDC failure since sub-header is used for EDC calculation. Data descrambling is performed before EDC. Note that EDC for this format does not cover the MSF address.

- The CD-ROM XA Mode 2 Form 2. As shown in Fig.50, compared with Form 1 data, Form 2 blocks do not contain C3 check bytes in exchange for more user data. The 4-byte EDC is not used for discs of this format. Descrambling and form bits extraction are performed in the same way as in Form 1. Sub-header mismatch is marked in the status field of the corresponding buffer block.

Format of an MP3 disc can be either CD-ROM Yellow Book Mode 1 or CD-ROM XA Mode 2 Form 1.
For CD-ROM modes, the EDC polynomial for a data frame is: $G(x) = X^{32}+X^{31}+X^{16}+X^{15}+X^4+X^3+X+1$. An EDC failure is indicated in the status field of the buffer block.

The 16-bit CRC of Q-subcode specifies a Cyclic Redundancy Check character computed over the CONTROL, ADR and DATA-Q fields. The field contains the inverted parity bits. The most significant bit of the CRC is in bit 81. The CRC generation polynomial is: $P(x) = X^{16}+X^{12}+X^5+1$.
A flywheel circuit is needed to interpolate a data frame sync if the sync pattern is not seen at the expected time. Some protection against spurious sync pattern is also needed by only allowing the sync pattern detector to operate within a small window which encompasses the expected time for the sync pattern. Address interpolation must be implemented.

Address interpolation is needed for Q-subcode frames as the MSF address information is only guaranteed to appear on 9 out of 10 consecutive frames while the other slots may contain UPC or MCN information.

If EDC is enabled in CD-ROM Yellow Book Mode 1 and CD-ROM XA Mode 2 Form 1, the block decoder will stop at any data block that fails EDC if EDCFailStop bit is set; BUF_WR_PTR and BlockCnt values will remain as they were when the last EDC-passed frame was buffered.

7.19.8   **CD TEXT MODE**

CD-TEXT data can be stored in the Lead-In Area and the Program Area and is read out from the disc through R-W subcode channels. The structure of CD-TEXT data is illustrated in Fig.13

Fig. 13 Structure of CD-TEXT data
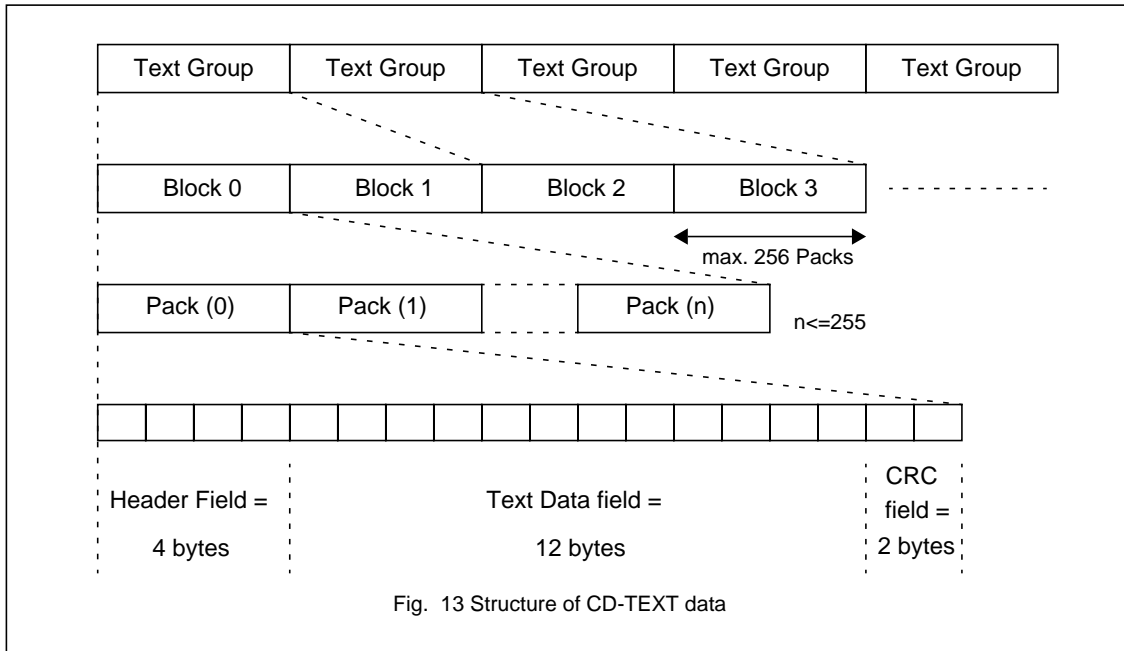
In CD-TEXT mode, R-W bits of subcode are de-interleaved (symbol to byte conversion) and saved into the buffer. Each Pack occupies a 20-byte buffer area, with 18 bytes of data and 2 bytes of status, in which only bit 0 of the last byte is used for CRC indication. CRC of CD-TEXT data uses the same algorithm as the CRC used in Q-channel subcode.

Note that in CD-TEXT mode, the Q-channel data can be accessed through Q_BUF registers.

7.19.9   Q-SUBCODE FRAME FORMAT

The general data format of Q-channel is illustrated in Fig.0. In all modes, the Q-channel subcode date can be read from registers Q_BUF_0 through Q_BUF_5, each register contains two subcode bytes. See table 212 for the organization of these registers. The Q_BUF registers are updated for each Q block sync detected and value of these registers belong to the last Q block.

Fig. 0 General Q-Channel Data Format
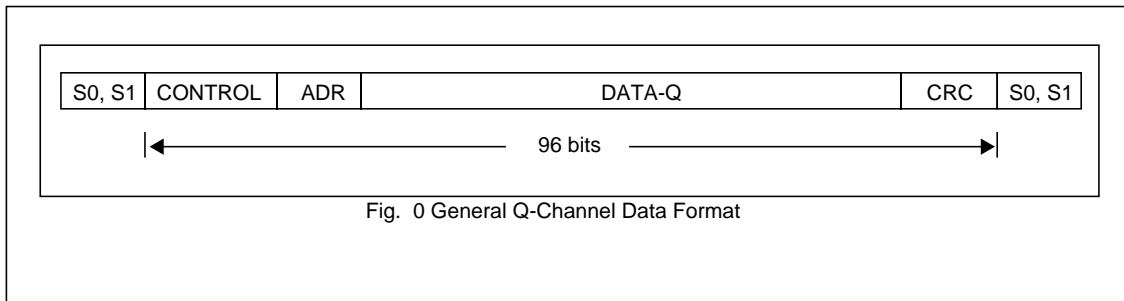
• CONTROL: 4 flag bits to define the kind of information in a track, MSB first.

- ADR: 4 control bits for DATA-Q, MSB first.
- DATA-Q: 72 data bits, MSB first.
- CRC: 16-bit CRC on CONTROL, ADR and DATA-Q. MSB first. On the disc the parity bits are inverted. The remainder has to be checked at zero.

7.20 Digital Signal Processor (EPICS7a)

In this chapter a description is given of the audio data flows between the ARm and the external world (with or without DSP intervention) and of the control data flow between the ARM and the DSP.

Fig. 15 AUDIO SUBSYSTEM
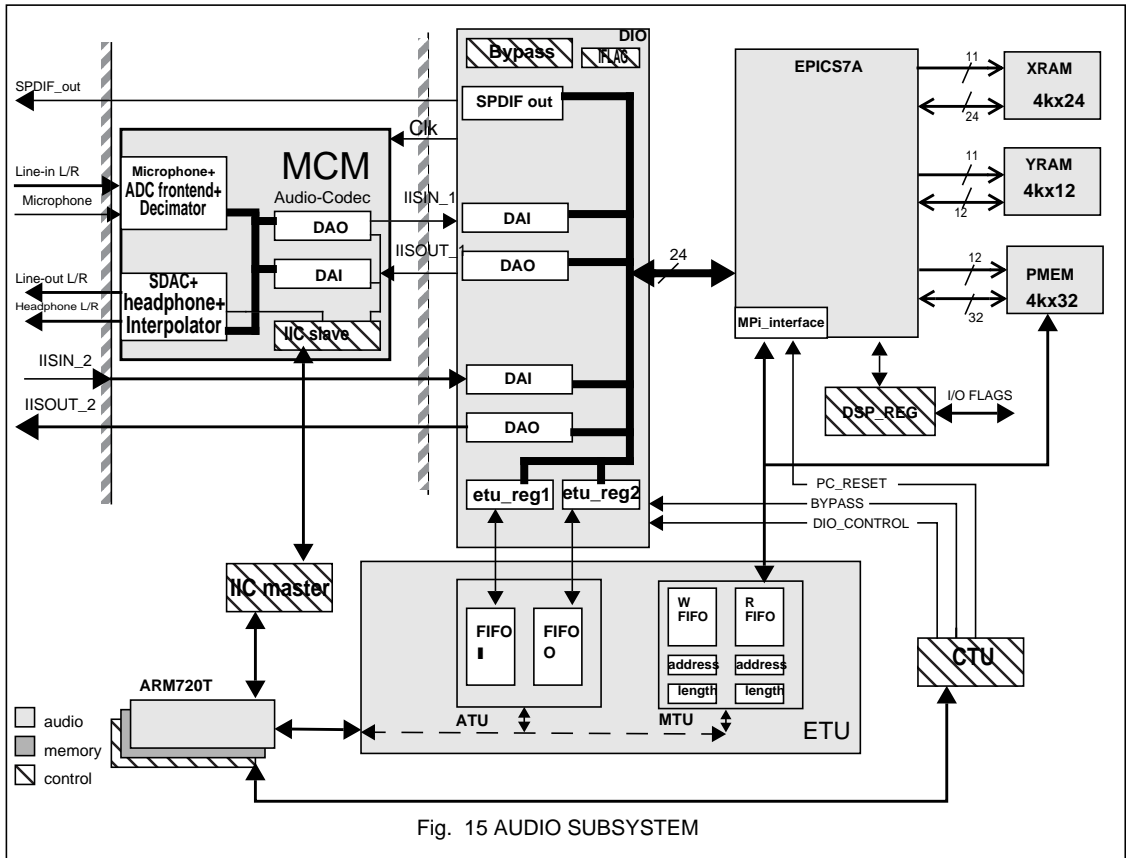
The DSP subsystem consists of the following main parts:

- 4K x 32 Program RAM (PRAM, field upgradable)

- 4K x 24 Data RAM (XRAM)

- 4K x 12 Coefficient RAM (YRAM, field upgradable)

- The EPICS Transfer Unit (ETU)

  The ETU consist out of an Audio Transfer Unit (ATU) and a Memory Transfer Unit (MTU). The ATU is for transferring the audio data between the CPU and the DIO. The MTU is for transferring data between the CPU and the DSP memories.

- The DSP

  The DSP (EPICS7a) is capable of processing all audio inputs and audio outputs of the DIO. The PMEM controls the ROM/RAM access for the EPICS7a. The PMEM also allows the ETU controller to access this memory.

- The Digital Input Output (DIO) module

The DIO consist of two IIS inputs (DAI) and two IIS outputs (DAO), connected to the Audio-codec Module. It contains a SPDIF_output including channel-, user- and validity bits for transferring audio as well as CD-trackdata.

A bypass part is included in the DIO to allow the option of bypassing the DSP and let the CPU take control of the DIO directly.

- The MPI interface

  This part interfaces the MTU to the correct DSP memories.

- The DSP registers (DSP_REG)

  The DSP registers indicates the SPDIF output status, user and validity bits and the selection bits of the DIO.

7.21 Digital Audio input and output

This part controls all digital inputs and outputs to and from the EPICS7A. All the audio streams are related to the same sampling frequency which is generated by the master digital input source. If no digital input source is available, the related signals are generated from the master system clock. All sources are feed to the DSP core via IO-addresses. The DSP will read these addresses and process the data. After processing, the data is sent back to a DSP IO-address which will go to a IIS output generator, a SPDIF channel output or to the CPU via the ATU. If the DSP is not used for audio-processing, the DSP can be powered-down and put in bypass-mode. In this case a selection can be made to feed any input to any output and the generator will generate the related signals.

The DIO system consist of the following blocks

- IIS input

  This part is an IIS input. It generates a parallel data signal for left and right and a newsam signal. Depending on the selection of the input the format will be IIS, LSB justified (16, 18, 20 and 24 bits) or MSB justified.

- IIS output

  This part will generate an IIS output data stream and depending on the setting it will be a IIS, LSB justified (16, 18, 20 or 24 bits) or MSB justified output format

- SPDIF Output, which supports:

  - Level II output

  - Support of 32kHz, 44.1kHz and 48kHz output frequencies.

  - Left and right channel status bits (40 bits per channel) can be set by the micro controller interface.

- BYPASS GEN

  This part defines if the audio inputs will be send to the DSP or if they will be send directly to an audio output

  – When bypass is disabled the 'YMU' signal and MODE signals to the LOGIC_block won't be changed by the BYPASS block

  – When bypass is enabled the 'YMU' signal to the LOGIC_block depends on the register settings of the BYPASS block. In these registers is stored which audio input is connected to which output. MODE and will always be '000000' in bypass mode.

8 HARDWARE DESCRIPTION BUILD-IN AUDIO CODEC

Refer to the application note "The SAA7750 build-in Audio Codec" and the "Application note: the Philips L3 interface".

General

- 2.4 to 3.6 V power supply
- 5V tolerant digital inputs (at 2.4 to 3.6V power supply)
- 24bits datapath for ADC and DAC datapath
- Selectable control via L3 micro-controller interface or I^{2}C interface control. Choice of 2 device addresses in L3 and I^{2}C mode.
- Supports sample frequencies from 8kHz to 55kHz for the ADC part, and 8kHz to 100kHz for the DAC part.
  This means from the ADC point of view no DVD audio (e.g. 96kHz audio) can be supported for the ADC part.
  For play-back 8kHz to 100kHz could be specified! = DVD playback is supported!
- Power management unit:
  – separate power control for ADC, AVC, DAC headphone driver and PLL.
  – analog blocks like ADC and PGA have a block to power down the bias circuits
  – when ADC and/or DAC are powered down, also the clocks to these blocks are stopped to save power.
- ADC part and DAC part can run at different frequencies (either system clock or WSPLL)
- ADC and PGA plus integrated high pass filter to cancel DC offset
- The decimation filter is equipped with a digital Automatic Gain Control.
- mono microphone input with Low Noise Amplifier (LNA) of 26dB and VGA (Variable Gain Control) with 0 to 30dB gain in 2dB steps.
- Integrated digital filter plus DAC
- separate single ended line output and one stereo Head Phone output, capable of driving 16Ω load. The headphone driver has build-in short circuit protection circuit with status bits which can be read out from the L3/IIC interface.
- Digital silence detection
- Easy application
- build-in plop prevention circuitry for the line out and headphone driver output
  Note: By default, when the IC is powered up, the complete chip will be in power down mode!

8.1 ADC front-end features

- ADC plus decimator can run at either PLL (regenerating the clock from WSI) or on SYSCLK.
- Stereo line in with PGA: gain range from 0dB till 24dB with 3dB steps
- Low Noise Amplifier with 29dB fixed gain for mono microphone input, including Variable Gain Amplifier with gain from 0dB till 30dB with 2dB steps
- Digital Left and Right independant volume control and mute in 0.5dB steps from +24dB gain till -63.5dB

8.2 DAC digital sound processing

- Separate digital logarithmic volume control for left and right channels via L3 or I^{2}C from 0dB down to -78dB in steps of 0.25dB.
- Digital tone control, bass boost and treble via L3 or I^{2}C

- Digital de-emphasis for 32, 44.1, 48 and 96kHz fs via L3 or I^{2}C

- cosine roll-off soft mute function

- Output signal polarity control via L3 or I^{2}C

- Digital mixer for mixing ADC output signal and digital serial input signal (in case they run at the same sampling frequency)

8.3   General description

The build-in audio Codec is a single chip stereo audio codec.

The build-in audio Codec front-end is equipped with a stereo line input which has PGA control, and a mono microphone input with a Low Noise Amplifier (LNA) and a Variable Gain Control (VGA). The digital decimation filter is equipped with an AGC which can be used in case of voice-recording.

The DAC part is equipped with a stereo line out and a headphone driver output. The headphone driver is capable of driving a 16Ω load. The headphone driver is also capable of driving a headphone without the need for external DC decoupling capacitors, since the headphone can be connected to a reference pin on the chip. In addition, there is a built-in short circuit protection circuit for the headphone driver output which, in case of short circuit, limits the current through the OPAMPs and signals the event via L3/I^{2}C bits.

The build-in audio Codec also supports an application mode in which the codec itself is not running, but an analog signal, like coming from an FM tuner, can be controlled in gain and output via the headphone driver and line outputs.

The build-in audio Codec has sound processing features in playback mode, de-emphasis, volume, mute, bass boost and treble which can be controlled by micro-controller interface.
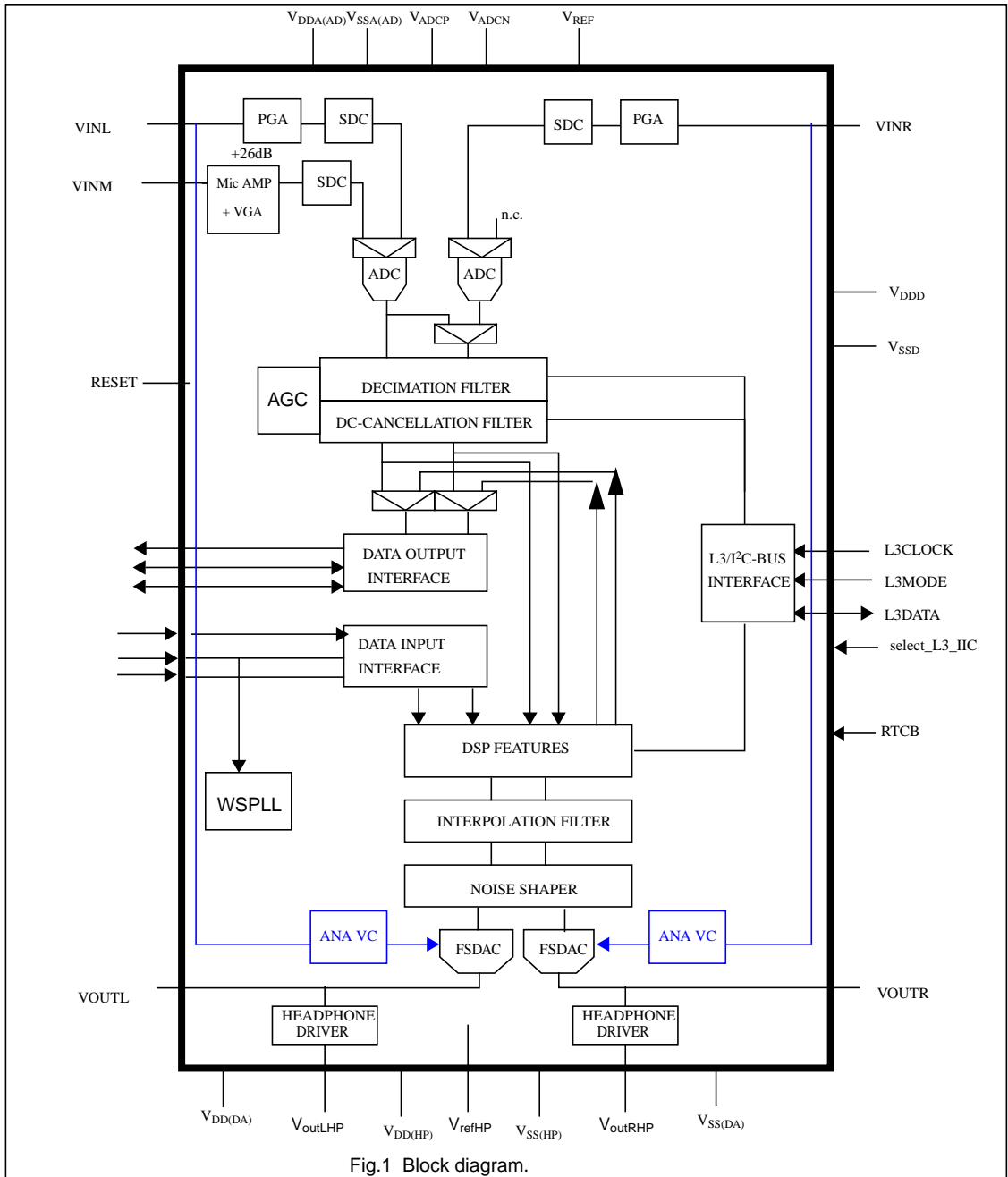
8.4 Block diagram

Fig.1 Block diagram.

9   HARDWARE DESCRIPTION FLASH

10 LIMITING VALUES

In accordance with the Absolute Maximum Rating System (IEC 134).

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
All digital I/Os						
V_I	DC input voltage range	note 1	−0.5	–	5.0	V
V_O	DC output voltage range		−0.5	–	3.6	V
I_O	output current	V_{DDE3V3} = 3.3 Volt		4		mA
Temperature values						
T_j	junction temperature		0	–	125	°C
T_{stg}	storage temperature		−55	–	+150	°C
T_{amb}	operating ambient temperature		-40	25	85	°C
Electrostatic handling						
V_{es}	electrostatic handling	HBM	−2000	–	+2000	V
		MM	−250	–	+250	V

Note

1.   All inputs are 5 Volt tolerant except for the USB pads.

11 THERMAL CHARACTERISTICS

SYMBOL	PARAMETER	CONDITIONS	VALUE	UNIT
R_{thj-a}	thermal resistance from junction to ambient	in free air	tbf	K/W

12 DC CHARACTERISTICS

$V_{DDE(3V3)}$ = 3.3 V; $V_{DDE(2V5)}$ = 2.5V; V_{DDI} = 1.8 V; T_{amb} = 25 °C; unless otherwise specified.

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Supply voltages						
V_{DDE3V3}	Peripheral (I/O) supply SAA7750		3.0	3.3	3.6	V
V_{DDE2V5}	Peripheral (I/O) supply SAA7750		2.2	2.5	2.8	V
V_{DDI1}	digital supply voltage 1 SAA7750		1.6	1.8	2.0	V
V_{DDI2}	digital supply voltage 2 SAA7750		1.6	1.8	2.0	V
V_{DDI3}	digital supply voltage 3 SAA7750		1.6	1.8	2.0	V
V_{DDI4}	digital supply voltage 4 SAA7750		1.6	1.8	2.0	V

SAA7750-N1D

Generic device for portable multimedia applications

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
V_{DDA1}	analog supply voltage 1, 6 MHz xtal supply voltage		1.6	1.8	2.0	V
V_{DDA2}	analog supply voltage 2, 32 kHz xtal supply voltage		1.6	1.8	2.0	V
V_{DDA3}	analog supply voltage 3, PLLs of SAA7750		1.6	1.8	2.0	V
V_{DDA4}	analog supply voltage 4, 10-bit ADC supply voltage		2.5	3.3	3.6	V
V_{DDDF}	digital supply voltage flash		2.2	2.5	2.8	V
V_{DDDC}	digital supply voltage codec		2.4	3.3	3.6	V
$V_{DDA(AD)}$	analog supply voltage ADC of codec		2.4	3.3	3.6	V
$V_{DDA(DA)}$	analog supply voltage DAC of codec		2.4	3.3	3.6	V
$V_{DDA(HP)}$	analog supply voltage Headphone Driver of codec		2.4	3.3	3.6	V
Supply currents (depend heavily on the application)						
I_{DDI1}	digital supply current 1 SAA7750		–	tbf	–	mA
I_{DDI2}	digital supply current 2 SAA7750		–	tbf	–	mA
I_{DDI3}	digital supply current 3 SAA7750		–	tbf	–	mA
I_{DDI4}	digital supply current 4 SAA7750		–	tbf	–	mA
I_{DDA1}	analog supply current 1, 6 MHz xtal	Oscillation	–	300	–	µA
		Power down	–	–	10	nA
I_{DDA2}	analog supply current 2, 32 kHz xtal	Oscillation	–	1.5	2.5	µA
		Power down	–	–	1	nA
I_{DDA3}	analog supply current 3, PLLs of SAA7750	Lock mode	–	3	-	mA
		Power down	–	–	3	µA
I_{DDA4}	analog supply current 4, 10-bit ADC	Normal mode	–	–	400	µA
		Power down	–	–	1	µA
I_{DDDF}	digital supply current flash	Normal mode	–	15	-	mA
		Power down	–	–	10	µA
I_{DDDC}	digital supply current codec	Playback mode	–	5.0	–	mA
		Recording mode	–	6.0	–	mA
		Full operational mode	–	10.0	–	mA
		Power down	–	4.0	–	µA

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
$I_{DDA(AD)}$	analog supply current ADC of codec	Speech recording mode	–	4.5	–	mA
		Audio recording mode	–	7.0	–	mA
		Speech + Audio recording mode	–	9.4	–	mA
		AVC only	–	3.15	–	mA
		Power down	–	2.0	–	µA
$I_{DDA(DA)}$	analog supply current DAC of codec	Normal mode	–	3.4	–	mA
		Power down	–	2.0	–	µA
$I_{DDA(HP)}$	analog supply current Headphone Driver of codec	Normal mode, no signal applied	–	3.5	–	mA
		Power down	–	2.0	–	µA
USB Interface (D+ and D−)						
V_{IH}	HIGH-level input voltage (driven)		2.0			V
V_{IHZ}	HIGH-level input voltage (floating)		2.7		3.6	V
V_{IL}	LOW-level input level				0.8	V
V_{DI}	differential input sensitivity		0.2			V
V_{CM}	differential common mode range		0.8		2.5	V
V_{OL}	LOW-level output voltage	R_L = 1.425 kΩ connected to V_{DD}	0.0		0.3	V
V_{OH}	HIGH-level output voltage (driven)	R_L = 14.25 kΩ connected to GND	2.8		3.6	V
V_{CRS}	output signal crossover voltage		1.3		2.0	V
I²C-bus (SDA and SCL)						
V_{IL}	LOW-level input voltage					V
V_{IH}	HIGH-level input voltage					V
V_{OL}	LOW-level output voltage					V
V_{OH}	HIGH-level output voltage					V
Digital input pins: 5V tolerant TTL compatible						
V_{IL}	LOW-level input voltage				$0.2*V_{DDE3V3}$	V
V_{IH}	HIGH-level input voltage		$0.8*V_{DDE3V3}$			V
$|I_{LI}|$	input leakage current				1.0	µA
Digital output pins						
V_{OL}	LOW-level output voltage				0.4	V
V_{OH}	HIGH-level output voltage		$0.85*V_{DDE3V3}$			V
10-bits ADC						
V_{in}	input voltage		V_{SSA4}	–	V_{refp}	V
V_{refp}	reference voltage		V_{SSA4}+2.0	–	V_{DDA4}	V

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
R_{refp}	input impedance V_{refp}		20	–	39	$k\Omega$
n	resolution		2	–	10	bits
INL	Integral non linearity				+/- 1	LSB
DNL	Differential non linearity				+/- 1	LSB
OSe	Offset error		-20	–	20	mV
FSe	Full Scale error		-20	–	20	mV
LNA+VGA build in audio Codec						
R_I	input resistance		–	12	–	$k\Omega$
C_I	input capacitance		–	24	-	pF
G	gain control range		20	–	50	dB
ΔG	gain control step size		–	2	–	dB
PGA SSA Codec						
R_I	input resistance		–	12	–	$k\Omega$
C_I	input capacitance		–	tbf	tbf	pF
G	gain control range		0	-	24	dB
ΔG	gain control step size		-	3	-	dB
ADC build in audio Codec						
V_{ref}	reference voltage	with respect to $V_{SSA(AD)}$	$0.45V_{DDA(AD)}$	$0.5V_{DDA(AD)}$	$0.55V_{DDA(AD)}$	V
V_{ADCP}	positive reference voltage of the audio ADC		–	$V_{DDA(AD)}$	–	V
V_{ADCN}	negative reference voltage of the audio ADC		–	$V_{SSA(AD)}$	–	V
Analog Volume Control build-in Codec						
G	gain control range		-48		+16.5	dB
ΔG_f	fine gain control step size		–	1.5	–	dB
ΔG_c	coarse gain control step size		–	6.0	–	dB
DAC build-in Codec						
$V_{OD(CM)}$	common mode output voltage		–	$0.5V_{DDA(DA)}$	–	V
$I_{o(max)}$	maximum output current		–	tbf	tbf	mA
R_{LD}	load resistance DAC		3	–	–	$k\Omega$
C_{LD}	load capacitance DAC		–	–	50	pF
Headphone Amplifier build-in Codec						
$V_{OH(CM)}$	common mode output voltage		–	$0.5V_{DDA(HP)}$	–	V
$R_{OH(VOUT)}$	output resistance at VOUTL (HP) and VOUTR(HP)		–	0.1	–	Ω

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
$I_{o(max)}$	maximum output current		–	tbf	tbf	mA
R_{LH}	load resistance Headphone Driver		16	–	–	Ω
C_{LH}	load capacitance Headphone Driver		–	–	tbf	pF

13 AC CHARACTERISTICS

$V_{DDE(3V3)}$ = 3.3 V; $V_{DDE(2V5)}$ = 2.5 V; V_{DDI} = 2.5 V; T_{amb} = 25 °C; unless otherwise specified.

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
10-bit ADC dynamic characteristics						
F_{smpl}	sampling rate		400 (10 bits)	–	1500 (2 bits)	KS/s
t_{conv}	conversion time		3 (2 bits)	–	11 (10 bits)	clk cycles
Oscillator 1						
f_{osc1}	oscillator frequency		–	6	–	MHz
α_{osc1}	duty cycle		–	50	–	%
$C_{i(XTAL1I)}$	parasitic input capacitance XTAL1a		tbf	tbf	tbf	pF
$C_{i(XTAL1O)}$	parasitic input capacitance XTAL2a		tbf	tbf	tbf	pF
t_{start}	start-up time			500		µs
P_{drive}	crystal level of drive		100		500	µW
Oscillator 2						
f_{osc2}	oscillator frequency		–	32.768	–	kHz
α_{osc2}	duty cycle		–	50	–	%
g_m	transconductance		tbf	tbf	tbf	mS
R_o	output resistance		tbf	tbf	tbf	kΩ
$C_{i(XTAL2I)}$	parasitic input capacitance XTAL1a		tbf	tbf	tbf	pF
$C_{i(XTAL2O)}$	parasitic input capacitance XTAL2a		tbf	tbf	tbf	pF
$t_{start, avg}$	average start-up time		–	4	–	ms
P_{drive}	crystal level of drive		0.5		1.5	µW
Analog-to-digital converter						
$V_{i(rms)}$	input voltage (RMS value)	0 dB setting	–	1.0	–	V
		3 dB setting	–	708	–	mV
		6 dB setting	–	501	–	mV
		9 dB setting	–	354	–	mV
		12 dB setting	–	252	–	mV
		15 dB setting	–	178	–	mV
		18 dB setting	–	125	–	mV
		21 dB setting	–	89	–	mV
		24 dB setting	–	63	–	mV
ΔV_i	unbalance between channels		–	<0.1	–	dB

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
$(THD + N)/S_{48}$	total harmonic distortion plus noise-to-signal ratio at f_s = 48 kHz	at 0 dB				
		0 dB setting	–	-85	–	dB
		3 dB setting	–	-85	–	dB
		6 dB setting	–	-85	–	dB
		9 dB setting	–	-85	–	dB
		12 dB setting	–	-84	–	dB
		15 dB setting	–	-83	–	dB
		18 dB setting	–	-82	–	dB
		21 dB setting	–	-80	–	dB
		24 dB setting	–	-78	–	dB
		at −60 dB; A-weighted				
		0 dB setting	–	-37	–	dB
		3 dB setting	–	-36	–	dB
		6 dB setting	–	-36	–	dB
		9 dB setting	–	-36	–	dB
		12 dB setting	–	-35	–	dB
		15 dB setting	–	-34	–	dB
		18 dB setting	–	-33	–	dB
		21 dB setting	–	-32	–	dB
		24 dB setting	–	-30	–	dB
α_s	channel separation		–	tbf	–	dB
S/N_{48}	signal-to-noise ratio at f_s = 48 kHz	V_i = 0 V; A-weighted	–	97	–	dB
LNA input plus analog-to-digital converter						
$V_{I(rms)}$	input voltage (rms value)		–	–	35	mV
$(THD+N)/S$	total harmonic distortion plus noise-to-signal ratio: f_s=48kHz	at 0dB	–	-74	–	dB
		at -60 dB; A-weighted	–	-25	–	dB
S/N	signal-to-noise: f_s=48kHz	V_I=0V; A-weighted	–	85	–	dB
α_{CS}	channel separation		–	70	–	dB
Digital-to-analog converter						
$V_{o(rms)}$	output voltage (RMS value)	at 0 dB (FS) digital input	–	0.9	–	V
ΔV_o	unbalance between channels		–	<0.1	–	dB
$(THD+N)/S_{48}$	total harmonic distortion-plus-noise to signal ratio at f_s = 48 kHz	at 0 dB	–	−85	–	dB
		at −60 dB; A-weighted	–	−40	–	dB
$(THD+N)/S_{96}$	total harmonic distortion-plus-noise to signal ratio at f_s = 96 kHz	at 0 dB	–	−80	–	dB
		at −60 dB; A-weighted	–	−37	–	dB
S/N_{48}	signal-to-noise ratio at f_s = 48 kHz	code = 0; A-weighted	–	100	–	dB

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
S/N$_{96}$	signal-to-noise at f_s = 96 kHz	code = 0; A-weighted	–	97	–	dB
α_{cs}	channel separation		–	90	–	dB
PSRR	power supply rejection ratio	f_{ripple} = 1 kHz; V_{ripple} = 30 mV (p-p)	–	60	–	dB
Headphone driver build in Codec						
P$_{O(rms)}$	output power	at 0 dB (FS) digital input, assuming 16Ω load	–	22	–	mW
(THD+N)/S	total harmonic distortion-plus-noise to signal ratio: f_s=48kHz	at 0 dB, 16Ω loaded	–	-65	–	dB
		at 0 dB, 5KΩ loaded	–	-85	–	
		at -60 dB; A-weighted	–	-35	–	dB
S/N	signal-to-noise; f_s=48kHz	code=0; A-weighted	–	95	–	dB
α_{CS}	channel separation	16Ω load, using V$_{REF(HP)}$, no DC decoupling capacitors	–	32	–	dB
		16Ω load, single-ended with DC decoupling capacitors (100uF typical), **note 1**	–	56	–	dB
Analog volume control (line in via ADC input, output on line-out and Headphone driver)						
V$_{I(rms)}$	input voltage (rms value)		–	150	–	mV
(THD+N)/S	total harmonic distortion-plus-noise to signal ratio at f_s=48kHz	at 0dB	–	-80	–	dB
		at -60 dB; A-weighted	–	-28	–	dB
S/N	signal-to-noise: f_s=48kHz	V$_I$=0V; A-weighted	–	87	–	dB
α_{CS}	channel separation		–	82	–	dB

Generic device for portable
multimedia applications

14 TIMING

$V_{DDD} = V_{DDA(ADC)} = V_{DDA(DAC)} = V_{DDA(HP)} = 2.7$ to 3.6 V; $T_{amb} = -20$ to +85 °C; all voltages referenced to ground; unless otherwise specified.

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
USB Interface driver characteristics D+ and D- (full-speed mode)						
t_{FR}	rise time	$C_L = 50$ pF	4	–	20	ns
t_{FF}	fall time	$C_L = 50$ pF	4	–	20	ns
t_{FRFM}	rise/fall time matching (t_{FR}/t_{FM})		90	–	111.11	%
Z_{DRV}	driver output resistance	steady-state drive	28	–	44	Ω
Serial interface input/output data timing (see Fig.2)						
f_{BCK}	bit clock frequency		–	–	$128f_s$	Hz
$T_{cy(BCK)}$	bit clock cycle time	$T_{cy(s)}$ = sample frequency cycle time	–	–	$^1/_{128}T_{cy(s)}$	s
t_{BCKH}	bit clock HIGH time		30	–	–	ns
t_{BCKL}	bit clock LOW time		30	–	–	ns
t_r	rise time		–	–	20	ns
t_f	fall time		–	–	20	ns
$t_{su(WS)}$	word select set-up time		10	–	–	ns
$t_{h(WS)}$	word select hold time		10	–	–	ns
$t_{su(DATAI)}$	data input set-up time		10	–	–	ns
$t_{h(DATAI)}$	data input hold time		10	–	–	ns
$t_{h(DATAO)}$	data output hold time		0	–	–	ns
$t_{d(DATAO-BCK)}$	data output to bit clock delay		–	–	30	ns
$t_{d(DATAO-WS)}$	data output to word select delay		–	–	30	ns
L3-bus interface timing (see Figs 3 and 4)						
t_r	rise time	note 2	–	–	10	ns/V
t_f	fall time	note 2	–	–	10	ns/V
$T_{cy(CLK)L3}$	L3CLOCK cycle time	note 3	500	–	–	ns
$t_{CLK(L3)H}$	L3CLOCK HIGH time		250	–	–	ns
$t_{CLK(L3)L}$	L3CLOCK LOW time		250	–	–	ns
$t_{su(L3)A}$	L3MODE set-up time in address mode		190	–	–	ns
$t_{h(L3)A}$	L3MODE hold time in address mode		190	–	–	ns
$t_{su(L3)D}$	L3MODE set-up time in data transfer mode		190	–	–	ns
$t_{h(L3)D}$	L3MODE hold time in data transfer mode		190	–	–	ns
$t_{stp(L3)}$	L3MODE stop time in data transfer mode		190	–	–	ns
$t_{su(L3)DA}$	L3DATA set-up time in address and data transfer mode		190	–	–	ns

SAA7750-N1D

Generic device for portable multimedia applications

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
$t_{h(L3)DA}$	L3DATA hold time in address and data transfer mode		30	–	–	ns
$t_{su(L3)R}$	L3DATA set-up time for read data		50	–	–	ns
$t_{h(L3)R}$	L3DATA hold time for read data		360	–	–	ns
$t_{en(L3)R}$	L3DATA enable time for read data		380	–	–	ns
$t_{dis(L3)R}$	L3DATA disable time for read data		50	–	–	ns
SDA and SCL lines (standard mode I^{2}C-bus) 100kHz mode						
f_{SCL}	SCL clock frequency		0	–	100	kHz
t_{LOW}	LOW period of the SCL clock		4.7	–	–	µs
t_{HIGH}	HIGH period of the SCL clock		4.0	–	–	µs
$t_{HD;STA}$	hold time (repeated) START condition		4.0	–	–	µs
$t_{SU;STA}$	set-up time for a repeated START condition		4.7	–	–	µs
$t_{SU;STO}$	set-up time for STOP condition		4.0	–	–	µs
t_{BUF}	bus free time between a STOP and START condition		4.7	–	–	µs
$t_{HD;DAT}$	data hold time		5.0	–	0.9	µs
$t_{SU;DAT}$	data set-up time		250	–	–	ns
t_r	rise time of both SDA and SCL signals		–	–	1000	ns
t_f	fall time of both SDA and SCL signals		–	–	300	ns
SDA and SCL lines (standard mode I2C-bus) 400kHz mode						
f_{SCL}	SCL clock frequency		0	–	400	kHz
t_{LOW}	SCL LOW time		1.3	–	–	µs
t_{HIGH}	SCL HIGH time		0.6	–	–	µs
t_r	rise time SDA and SCL	note 4	$20 + 0.1C_b$	–	300	ns
t_f	fall time SDA and SCL	note 4	$20 + 0.1C_b$	–	300	ns
$t_{HD;STA}$	hold time START condition	note 5	0.6	–	–	µs
$t_{SU;STA}$	set-up time repeated START		0.6	–	–	µs
$t_{SU;STO}$	set-up time STOP condition		0.6	–	–	µs
t_{BUF}	bus free time between a STOP and START condition		1.3	–	–	µs
$t_{SU;DAT}$	data set-up time		100	–	–	ns
$t_{HD;DAT}$	data hold time		0	–	–	µs
t_{SP}	pulse width of spikes	note 6	0	–	50	ns
C_b	capacitive load for each bus line		–	–	400	pF

Notes

1. The typical value of the timing is specified at 48 kHz sampling frequency.

2. In order to prevent digital noise interfering with the L3-bus communication, it is best to have the rise and fall times as small as possible.

3. When the sampling frequency is below 32 kHz, the L3CLOCK cycle must be limited to $\frac{1}{64fs}$ cycle.

4. C_b is the total capacitance of one bus line in pF. The maximum capacitive load for each bus line is 400 pF.

5. After this period, the first clock pulse is generated.
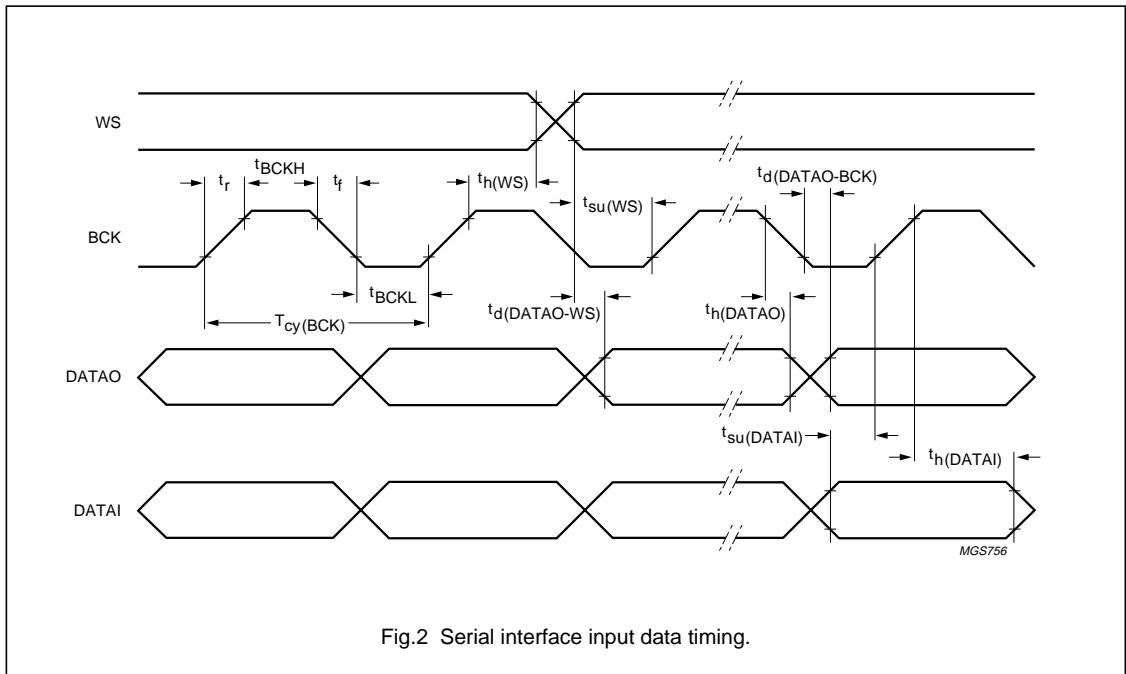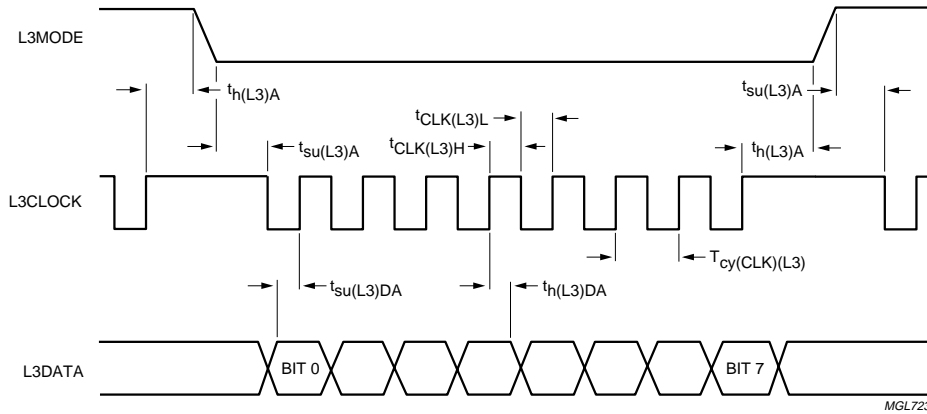
6. To be suppressed by the input filter.

Fig.2 Serial interface input data timing.

Fig.3 Timing of address mode.
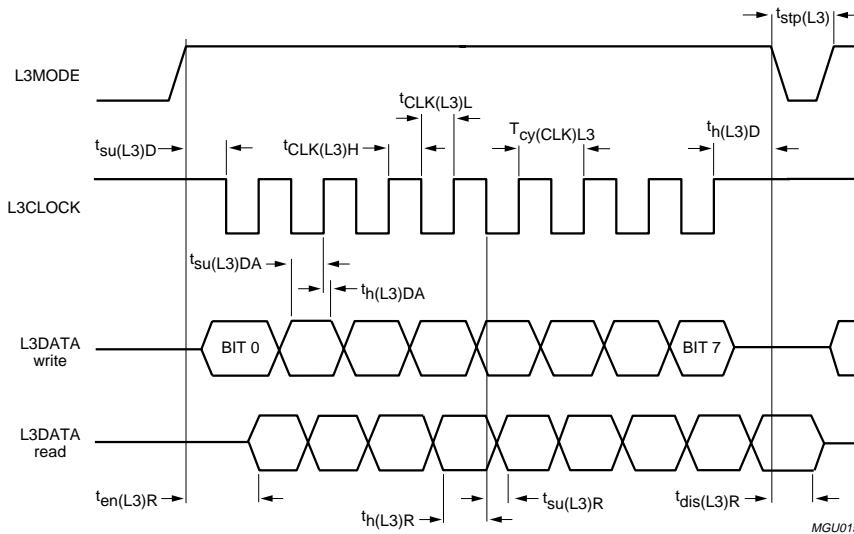
Fig.4 Timing of data transfer mode for write and read.
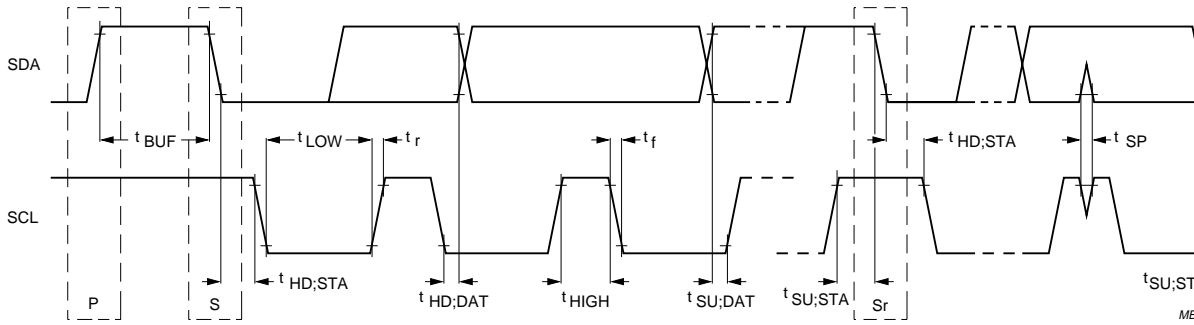
Fig.5 Timing of the I²C-bus transfer.
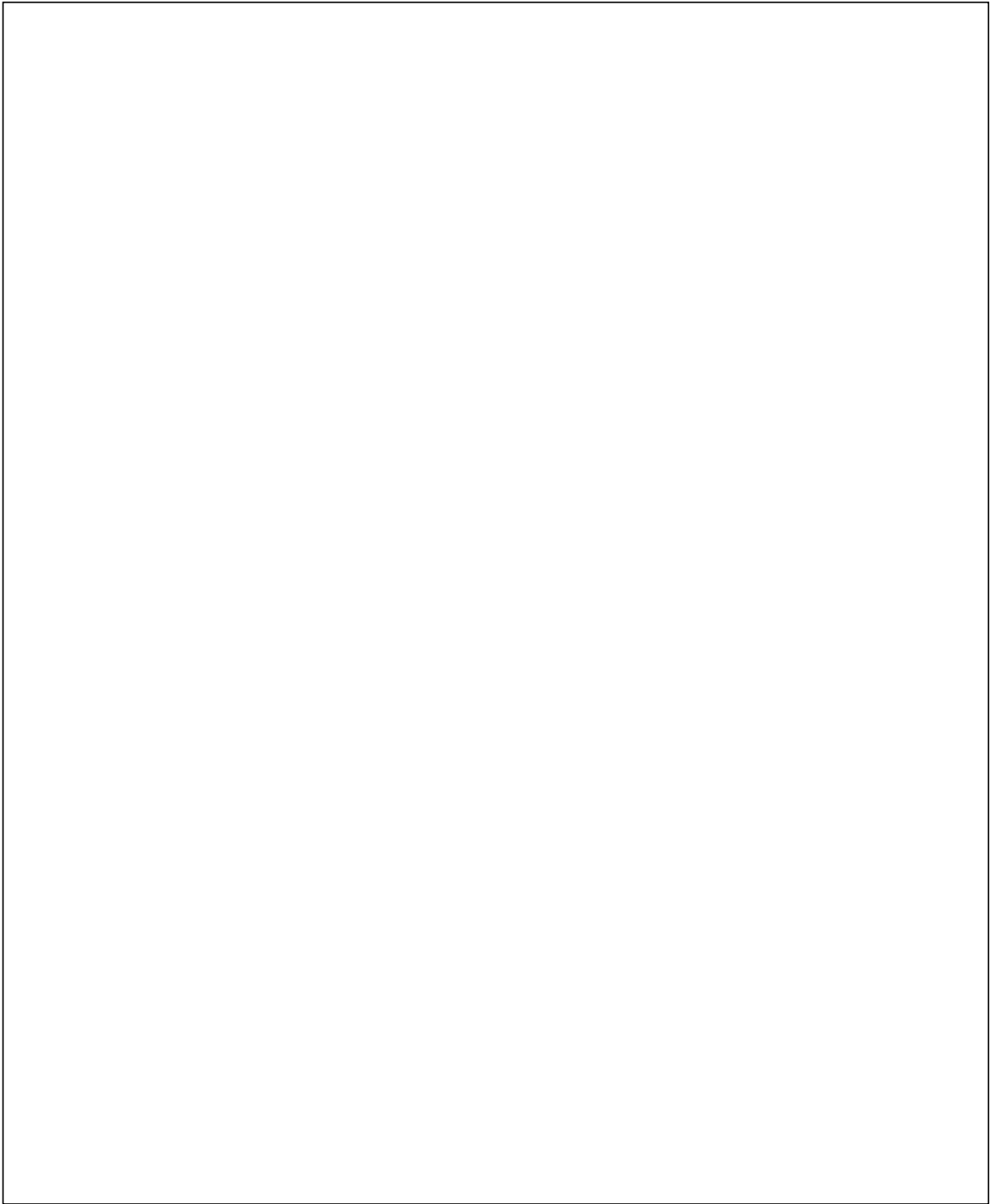
16 SOLDERING

Introduction

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

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

Reflow soldering

Reflow soldering techniques are suitable for all SSOP packages.

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

Several techniques exist for reflowing; for example, thermal conduction by heated belt. Dwell times vary between 50 and 300 seconds depending on heating method. Typical reflow temperatures range from 215 to 250 °C.

Preheating is necessary to dry the paste and evaporate the binding agent. Preheating duration: 45 minutes at 45 °C.

Wave soldering

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

If wave soldering cannot be avoided, the following conditions must be observed:

- **A double-wave (a turbulent wave with high upward pressure followed by a smooth laminar wave) soldering technique should be used.**
- **The longitudinal axis of the package footprint must be parallel to the solder flow and must incorporate solder thieves at the downstream end.**

Even with these conditions, only consider wave soldering SSOP packages that have a body width of 4.4 mm, that is SSOP16 (SOT369-1) or SSOP20 (SOT266-1).

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

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

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

Repairing soldered joints

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

17 DEFINITIONS

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

18 DISCLAIMERS

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

Right to make changes — Philips Semiconductors reserves the right to make changes, without notice, in the products, including circuits, standard cells, and/or software, described or contained herein in order to improve design and/or performance. Philips Semiconductors assumes no responsibility or liability for the use of any of these products, conveys no licence or title under any patent, copyright, or mask work right to these products, and makes no representations or warranties that these products are free from patent, copyright, or mask work right infringement, unless otherwise specified.

19 PURCHASE OF PHILIPS I^{2}C COMPONENTS

Purchase of Philips I^{2}C components conveys a license under the Philips' I^{2}C patent to use the components in the I^{2}C system provided the system conforms to the I^{2}C specification defined by Philips. This specification can be ordered using the code 9398 393 40011.