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

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

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

Contact us

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

AS3524
Advanced Audio Processor System

1 Description

The AS3524 implements a highly flexible and fully integrated digital audio processor system combining strong calculating power and high performance interfaces commonly used within audio player systems.

Using advanced 0.13µm process technology and large on chip RAM leads to outstanding low power consumption of 0.3 mW/MHz for the ARM922T microcontroller core and 0.6 mW/MHz for the overall system measured with a typical MP3 player SW application.

Based on a powerful ARM9TDMI capable of performing up to 200MIPS it is suited to run MP3, AAC, WMA, OGG… decoders and encoders and, in addition, it can perform extensive user interfaces, motion graphics support, video playback and much more.

The AS3524 SOC (system-on-a-Chip) features dedicated high speed interfaces for ATA IDE, USB2.0 HS-OTG and SDRAM ensuring maximum performance for download, upload, and playback.

Furthermore interfaces for NAND flashes, MMC/SD cards and Memory Stick ensure most flexible system design possibilities. Hardware support for parallel interfaces lower the CPU load serving complex and/or colour user interfaces.

Additional serial high-speed data and control interfaces guarantee the connection to other peripherals and or processors in the system.

Two independently programmable PLLs generate the required frequencies for audio playback/recording, for the processor core and for the USB interface at the same time.

Key Features

1.1 Digital Core

Embedded 32-Bit RISC Controller

- ARM922TDMI RISC CPU
- 2.5Mbit on-chip RAM
- 1Mbit on chip ROM
- Clock speed max. 250MHz (200MIPS)
- Standard JTAG interface

USB 2.0 HS & OTG Interface

- Up to 480Mbit/s transfer speed
- USB 2.0 HS/FS physical inlcuding OTG support
- USB 2.0 HS/FS digital core including OTG host
- Dedicated dual port buffer RAM
- DMA bus master functionality

IDE Host Controller

- Supporting Ultra ATA 33/66/100/133 modes
- Programmable IO and Multi-word DMA capability
- Dedicated dual port buffer RAM
- DMA bus master functionality

External Memory Controller

- Dynamic memory interface
- Asynchronous static memory
- DMA bus master functionality

DMA Controller

- Single Master DMA controller
- 2 DMA channels possible at the same time
- 16 DMA requests supported

Interrupt Controller

- Support for 32 standard interrupts
- Support for 16 vectored IRQ interrupts

Audio Subsystem Interface

- Dedicated 2 wire serial control master
- I2S input and output with dual port buffer RAM

Nand Flash Interface

- 8 and 16bit flash support
- 3, 4 & 5 byte address support
- hardware ECC

MMC/SD Interface

- MMC/SD Card host for multiple card support
- 4 data line support for SD cards

MS / MS Pro Interface

- Dedicated dual port buffer RAM

Display Interface

- Serial and parallel controller supported
- On chip hardware acceleration

Synchronous Serial Interface

- Master and slave operation
- 8 and 16 bit support
- Several protocol standards supported

I2S Interface

- Input multiplexed with audio subsystem
- selectable SPDIF input conversion
- Dedicated dual port buffer RAM

2 Wire Serial Control Interface

- Master and slave operation
- Standard and fast mode support

General Purpose IO Interface

- 4x 8-bit ports

Multiple Boot Options

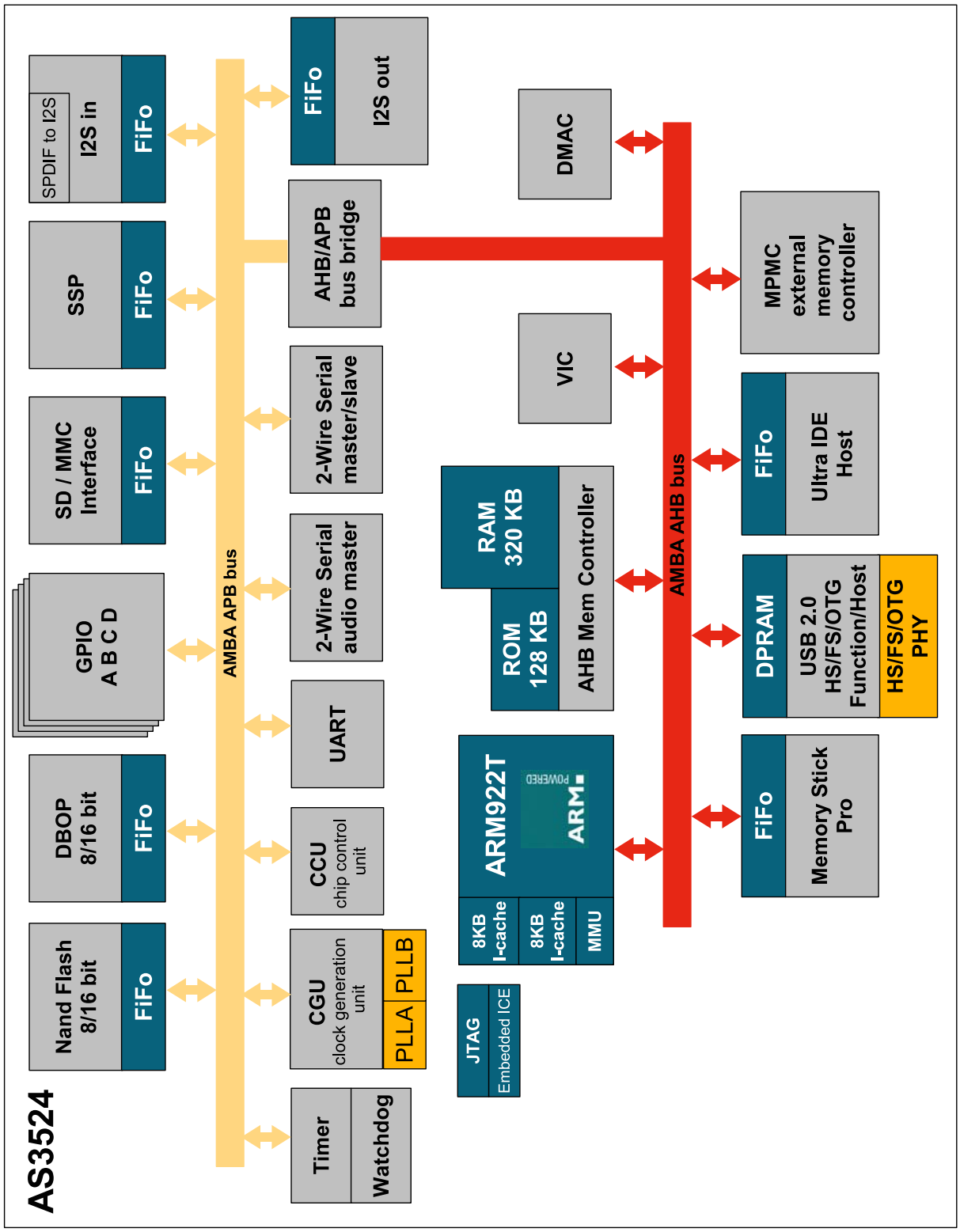
- Selection of internal ROM or external boot device
- Internal boot loader supporting boot from external NorFlash, NandFlash, IDE, SPI host
- Internal USB boot loader with USB promer supporting initial factory programming and firmware update

2 Application

- Portable Digital Audio Player and Recorder
- Portable Digital Media Player
- PDA
- Smartphone

3 Block Diagram

Figure 1 AS3524 Block Diagram



Document Revisions

Revision	Chapter	Date	Owner	Description
0.1	all	9.3.2005	MMA	first preliminary version
0.2		31.3.2005	MMA	package drawing and pinout added
0.3		14.9.2005	PKM	marking description and top view added
1.0	all	8.5.2006	WSG	first release of document generated
1.1	5.1.6.2, 8 5.3.13.1	25.9.2006	WSG	added description for modified C22 bootloader added description for UART Baud rate settings
1.11	5.1.6.2	9.11.2006	WSG	corrected table headers for boot device selection and USB boot frequency settings
1.12	4.2.4	24.3.2010	WSG	Added chapter “Startup Sequence for Supply Voltages”
	4.1	24.3.2010	WSG	Added table for soldering conditions according to Pb-Free process

Related Documents

ARM922T Technical Reference Manual	DDI0184B_922T_TRM.pdf	http://www.arm.com
ARM9TDMI Technical Reference Manual	DDI0180A_9tdmi_trm.pdf	http://www.arm.com
PrimeCell™ MultiPort Memory Controller; PL172 Technical Reference Manual		http://www.arm.com
AMBA Specification (Rev 2.0)	IHI0011A_AMBA_SPEC.pdf	http://www.arm.com
PrimeCell™ Synchronous Serial Port; PL022 Technical Reference Manual		http://www.arm.com
PrimeCell™ General Purpose Input/Output; PL061 Technical Reference Manual		http://www.arm.com
PrimeCell™ Single Master DMA Controller; PL081 Technical Reference Manual		http://www.arm.com
PrimeCell™ Multimedia Card Interface; PL180 Technical Reference Manual		http://www.arm.com
PrimeCell™ Vectored Interrupt Controller; PL190 Technical Reference Manual		http://www.arm.com
CWda03 - SPDIF-AES/EBU TO I2S CONVERTER		http://www.coreworks.pt
TSMC TPZ013G3 Standard I/O Library Databook		http://www.tsmc.com
DesignWare USB 2.0 HI-SPEED ON-THE-GO Controller Subsystem		http://www.synopsys.com
DesignWare USB 2 PHY Hardmacro		http://www.synopsys.com
SMS2IP mem stick host controller		http://www.sony.com
ICON mem stick host con interface		http://www.sony.com
IDE host controller BK3710S		http://www.palmchip.com
AS352x USB MSC Boot Promer specification document	AS352X_USB_MSC_Boot_Promer_V07.doc	

4 Electrical Specifications

4.1 Absolute Maximum Ratings

Stresses beyond those listed under “Absolute Maximum Ratings“ may cause permanent damage to the device. These are stress ratings only. Functional operation of the device at these or any other conditions beyond those indicated under “Operating Conditions” is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. The device should be operated under recommended operating conditions.

Table 1 Absolute Maximum Ratings

Symbol	Parameter	Min	Max	Unit	Note
VDD_{peri}		-0.5	3.7	V	digital periphery supply voltage
VDD_{mem}		-0.5	3.7	V	digital IO supply for MPMC PADs
$USBVDDA33_T$		-0.5	3.7	V	USB analog supply transmit block to be connected to UVDD
$USBVDDA33_C$		-0.5	3.7	V	USB analog supply common block to be connected to UVDD
VDD_{core}		-0.5	1.68	V	digital core supply voltage
$VDD_{coreana}$		-0.5	1.68	V	core supply for critical blocks (1-TRAM)
$VDDA_{PLL}$		-0.5	1.68	V	core supply forPLLA, PLLB
V_{IN_5V}	5V pins	-0.5	7.0	V	Applicable for pins VBUS
V_{IN_VSS}	Voltage difference at VSS terminals	-0.5	0.5	V	Applicable for pins vss_core, vss_peri, vss_mem, usb_vssa33t, usb_vssa33c
I_{scr}	Input Current (latchup immunity)	-100	100	mA	Norm: JEDEC 17
ESD	Electrostatic Discharge HBM		+/-1	kV	Norm: MIL 883 E method 3015
ESD_USB	Electrostatic Discharge HBM for USB Pins		+/-2	kV	Norm: MIL 883 E method 3015 (Pins: usb_dp, usb_dm, usb_vbus)
P_t	Total Power Dissipation (all supplies and outputs)		1000	mW	for CTBGA180 package
T_{strg}	Storage Temperature	-55	125	°C	
H	Humidity non-condensing	5	85	%	

Table 2 Soldering Conditions

Symbol	Parameter	Min	Max	Unit	Note
T_c	Pb-Free Process – Classification Temperature		260	°C	According to IPC/JEDEC J-STD-020 (2), reflects moisture sensitivity level only
T_p	Pb-Free Process – Peak Temperature	255	260	°C	Peak temperature according to IPC/JEDEC J-STD-020 (2)
t_p	Time t_p		30	s	Time t_p within 5 °C of the specified classification temperature T_c according to IPC/JEDEC J-STD-020 (2)
D_{well}		30	45	s	Time above 217 °C

Note: (1) austriamicrosystems AG strongly recommends to use underfill.
(2) IPC/JEDEC J-STD-020: Moisture/Reflow Sensitivity Classification for Nonhermetic Solid State Surface Mount Devices.

4.2 Operating Conditions

4.2.1 Supply Voltages

Following supply voltages for the digital system are generated by internal LDOs.

Table 3 Operating conditions for internal generated supply voltages

Symbol	Parameter	Min	Max	Unit	Note
VDD_{peri}		3.0	3.6	V	digital periphery supply voltage
VDD_{mem}		1.75	3.4	V	digital IO supply for MPMC PADs
VDD_{core}		1.08	1.25	V	digital core supply voltage **see Note (1)**
$VDD_{coreana}$		1.08	1.25	V	core supply for critical blocks (1-TRAM)
$VDDA_{PLL}$		1.08	1.25	V	core supply forPLLA, PLLB
$USBVDDA33_{T}$		3.15	3.45	V	USB analog supply transmit block to be connected to UVDD
$USBVDDA33_{C}$		3.15	3.45	V	USB analog supply common block to be connected to UVDD
	Difference of Negative Supplies vss_peri, vss_core, vss_core_ana, vss_mem, vssa_pll, usb_vssa33c, usb_vssa33t,	-0.1	0.1	V	To achieve good performance, the negative supply terminals should be connected to low impedance ground plane.

Note(s) **(1) For the VDD_CORE supply, voltage scaling should be applied to optimize power consumption and CPU speed performance. For normal operation with fclk (CPU ARM-922T clock) frequencies below 200 MHz, CVDD (supply of VDD_CORE) can be set to a lower value of 1.10 V. Only for setting fclk of the CPU to clock frequencies above 200 MHz, the VDD_CORE supply voltage must be set to 1.20 V typical conditions.**

4.2.2 Operating Currents

Table 4 Supply currents

Symbol	Parameter	Typ	Max	Unit	Note
IDD_PERI_OP	Peripheral current	2	20	mA	
IDD_MEM_OP	External memory interface current	-	20	mA	(2)
IDD_CORE_OP	Digital core current	20	145	mA	(1), (2)
IDD_USBA33T_OP	USB transmitter current		30	mA	
IDD_USBA33C_OP	USB common blocks current		30	mA	

Notes (1) Typical condition for playback of MP3 music with 44.1 KHz / 128 kbit with 32Ω headphones. No external SDRAM connected. USB2.0 in standby.
(2) Maximum condition for ARM running at 250 MHz, AHB/APB bus and memory at 64 MHz, USB 2.0 in HS operation.

In the case of standby mode or in the case of configuring the device to stopped clock, following current consumption is measured.

Table 5 Leakage currents

Symbol	Parameter	Typ	Max	Unit	Note
IDD_PERI_LEAK			4	mA	Including USBA33T, USBA33C
IDD_MEM_LEAK			800	µA	
IDD_CORE_LEAK			3	mA	@ $T_{ambient}$=25 ^{O} C
IDD_LEAK(VDDAPLL+COREANA)			1.5	mA	@ $T_{ambient}$=25 ^{O} C

4.2.3 Temperature Range

Table 6 Temperature Range

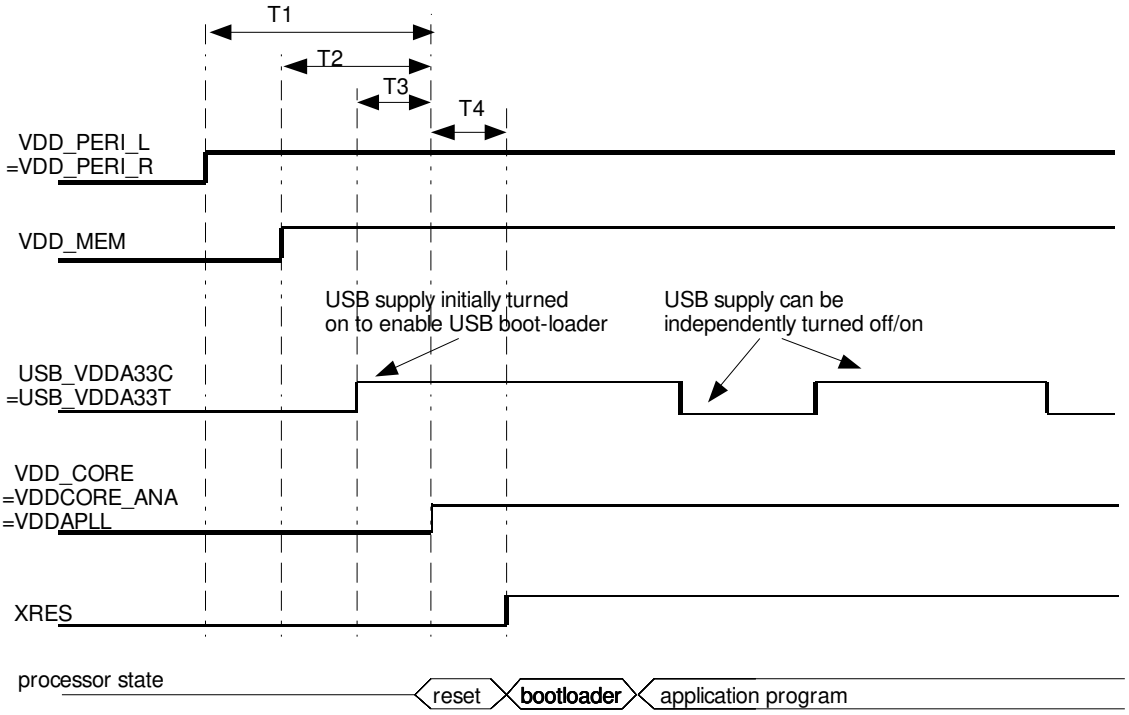
Symbol	Parameter	Min	Typ	Max	Unit	Note
T_{op}	Operating temperature range	0	25	85	°C	
T_j	Junction temperature range	0		110	°C	
R_{th}	Thermal Resistance		29		°C/W	

4.2.4 Start-up Sequence for Supply Voltages

To ensure correct start-up special timing constraints are given for circuit power-on. Also to ensure correct functionality of the USB Bootloader certain restrictions must be met.

For start-up it has to be guaranteed that VDD_CORE is not applied before all other peripheral supply voltages are switched on. The only exception are the USB suppies (USB_VDDA33C, USB_VDDA33T), which can be switched on and off independently of all other supplies.

Figure 2: AS3524 startup



Peripheral voltages can be started simultaneously, but because of transient currents during start-up flowing into decoupling caps it is suggested that these voltages are turned on in a serial sequence.

The USB supply voltage should be turned on at start-up to enable start of USB loader without any needed configuration for the AFE chip.

Symbol	Parameter	Min	Typ	Max	Unit
T1	delay between VDD_PERI and VDD_CORE startup	0.2	3		ms
T2	delay between VDD_MEM and VDD_CORE startup	0.2	2		ms
T3	delay between USB_VDD and VDD_CORE startup	0.2	1		ms
T4	delay between VDD_CORE startup and XRES release	0.01	1		ms

5 Detailed Functional Descriptions

5.1 ARM922-T Processor Core

5.1.1 General

The ARM922T macrocell is a high-performance 32-bit RISC integer processor combining an ARM9TDMI™ processor core with:

- 8KB instruction cache and 8 KB data cache
- Instruction and data Memory Management Unit (MMU)
- Write buffer with 16 data words and 4 addresses
- Advanced Microprocessor Bus Architecture (AMBA™) AHB interface

The ARM922T provides a high-performance processor solution for open systems requiring full virtual memory management and sophisticated memory protection. The ARM922T processor core is capable of running at 250 MHz. The ARM922T hard macrocell has a very low power consumption. The integrated cache helps to significantly reduce memory bandwith demands, improving performance and minimizing power consumption.

At 250 MHz the ARM922T comsumes as little as 65 mW, making it ideal for high-performance battery operated audio or video applications.

The ARM core and associated bus structures are configured for little endian byte order (compatible with Windows CE™ and Symbian™ OS).

Table 7 ARM 922T characteristics

Cache (I/D)	MMU	AHB	Thumb	mW/MHz	MHz
8KB / 8KB	yes	yes	yes	0.25 @ 1.2 V	250

Features

- 32-bit RISC architecture (ARMv4T)
- Harvard architecture with separated instruction (I) and data (D) caches with 8 KB each and 8-word line length
- Five stage pipeline (fetch, decode, execute, memory, write back) enabling high master clock speeds
- 32-bit ARM instruction set for maximum performance and flexibility
- 16-bit Thumb instruction set for increased code density
- Enhanced ARM architecture V4 MMU to provide translation and access permission checks for instruction and data addresses. With this MMU different operating systems (Windows CE, Symbian ...) can be implemented.
- Industry standard AMBA bus interface (AHB and APB)
- Hard-macro implementation
- The processor core clock frequency (FCLK) is programmable up to 250MHz and the ARM922 power consumption is directly proportional to this clock frequency FCLK

5.1.2 Block Diagrams

Figure 3 ARM 922T Functional Block Diagram

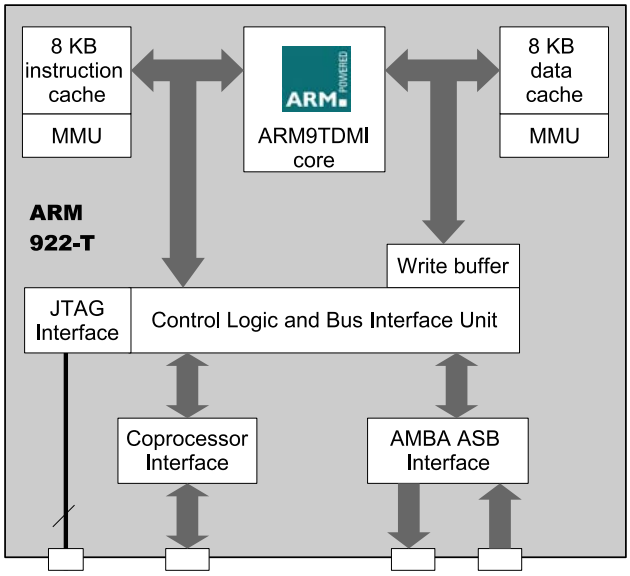
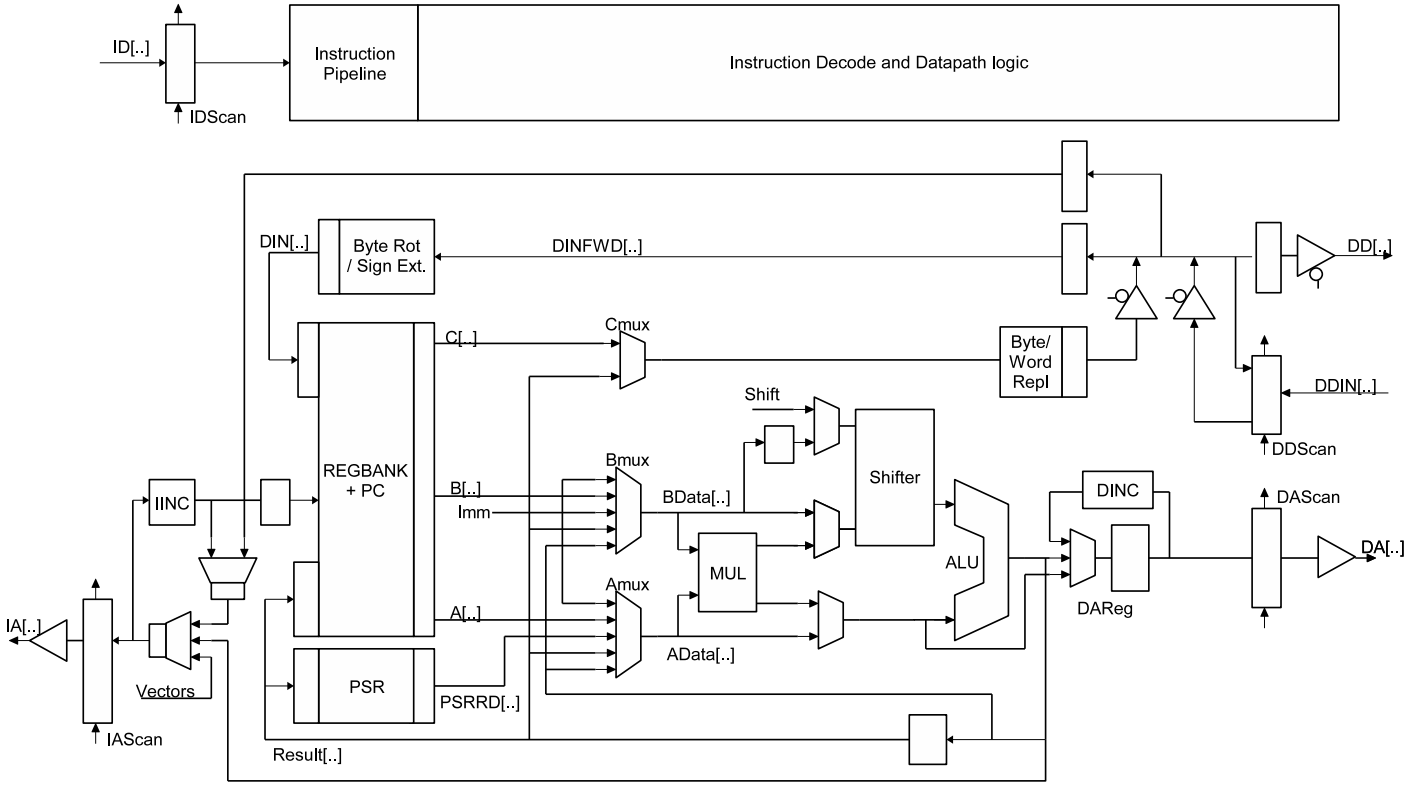


Figure 4 ARM9TDMI Functional Block Diagram



5.1.3 ARM922T Details

The ARM922T macrocell is based on the ARM9TDMI Harvard architecture processor core with an efficient five-stage pipeline. To reduce the effect of memory bandwidth and latency on performance, the ARM922T macrocell includes separate cachs and MMUs for both instructions and data. It also has a write buffer and physical address TAG RAM.

Caches

Two 8KB caches are implemented, one for instructions, the other for data, both with an 8-word line size. Separate buses connect each cache to the ARM9TDMI core permitting a 32 bit instruction to be fetched and fed into the Decode stage of the pipeline at the same time as a 32 bit data access for the memory stage of the pipeline.

Cache lock-down is provided to permit critical code sequences to be locked into the cache to ensure predictability for real-time code. The cache replacement algorithm can be selected by the operating system as either pseudo-random or round-robin. Both caches are 64-way set-associative. Lock-down operates on a per-way basis.

Write Buffer

The ARM922T macrocell also incorporates a 16-data, 4-address write buffer to avoid stalling the processor when writes to external memory are performed.

PA TAG RAM

The ARM922T macrocell implements a physical address TAG RAM (PA TAG RAM) to perform write-backs from the data cache.

The physical addresses of all the lines held in the data cache are stored by the PA TAG memory, removing the requirement for address translation when evicting a line from the cache.

MMU

The ARM922T macrocell implements an enhanced ARMv4 MMU to provide translation and access permission checks for the instruction and data address ports of the ARM9TDMI core.

The MMU features are:

- Standard ARMv4 MMU mapping sizes, domains, and access protection scheme
- Mapping sizes are 1 MB sections, 64 KB large pages, 4 KB small pages, and new 1KB tiny pages
- Access permissions for sections
- Access permissions for large pages and small pages can be specified separately for each quarter of the page (subpages)
- Access permissions for tiny pages
- 16 domains implemented in hardware
- 64-entry instruction Translation-Lookaside-Buffer (TLB) and 64-entry data TLB
- Hardware page table walks
- Round-robin replacement algorithm (also called cyclic)

Control Coprocessor (CP15)

The control coprocessor is provided for configuration of the caches, the write buffer, and other ARM922T options.

Eleven registers are available for program control:

- Register 1 controls system operation parameters including endianness, cache, and MMU enable
- Register 2 and 3 configure and control MMU functions
- Register 5 and 6 provide MMU status information
- Register 7 and 9 are used for cache maintenance operations
- Register 8 and 10 are used for MMU maintenance operations
- Register 13 is used for fast context switching
- Register 15 is used for test.

Debug Features

The ARM9TDMI processor core incorporates an EmbeddedICE unit and EmbeddedICE-RT logic permitting both software tasks and external debug hardware to

- Set hardware and software breakpoints
- Perform single-stepping
- Enable access to registers and memory

This functionality is implemented as a coprocessor and is accessible from hardware through the JTAG port.

Full-speed, real-time execution of the processor is maintained until a breakpoint is hit.

At this point control is passed either to a software handler or to JTAG control.

5.1.4 ARM V4T Architecture

The ARM9TDMI processor core implements the ARMv4T Instruction Set Architecture (ISA). The ARMv4T ISA is a superset of the ARMv4 ISA with additional support for the Thumb 16-bit compressed instruction set.

Performance and Code Density

The ARM9TDMI core executes two instruction sets

- 32-bit ARM instruction set
- 16-bit Thumb instruction set

The ARM instruction set is designed so that a program can achieve maximum performance with the minimum number of instructions. Most ARM9TDMI instructions are executed in a single cycle.

The simpler Thumb instruction set offers much increased code density deducing code size and memory requirement.

Code can switch between the ARM and Thumb instruction sets on any procedure call.

ARM9TDMI Integer Pipeline Stages

The integer pipeline consists of five stages to maximize instruction throughput in the ARM9TDMI core:

- Fetch
- Decode and register read
- Execute shift and ALU operation, or address calculate, or multiply
- Memory access and multiply
- Write register

By using a five-stage pipeline, the ARM922T delivers a throughput approaching one instruction per cycle.

Registers

The ARM9TDMI processor core consists of a 32-bit datapath and associated conrol logic. This datapath contains 31 general-purpose registers, coupled to a full shifter, Arithmetic Logic Unit, and a multiplier. At any one time 16 registers are visible to the user. The remainder are mode-specific replacement registers (banked registers) used to speed up execution processing, and make nested exceptions possible.

Register 15 is the Program Counter (PC) that can be used in all instructions to reference data relative to the current instruction. R14 holds the return address after a subroutine call. R13 is used (by software convention) as a stack pointer.

Exeption Types/Modes

The ARM9TDMI core supports five types of exception, and a privileged processing mode for each type. The types of exceptions are:

- Fast interrupt (FIQ)
- Normal interrupt (IRQ)
- Memory aborts (used to implement memory protection or virtual memory)
- Attempted execution of an undefined instruction
- Software interrupts (SWIs)

All exceptions have banked registers for R14 and R13. After an exception, R14 holds the return address for exception processing. This address is used both to return after the exception is processed and to address the instruction that caused the exception.

R13 is banked across exception modes to provide each exception handler with a private stack pointer. The fast interrupt mode also banks registers 8 to 12 so that interrupt processing can begin without the need to save or restore these registers.

A seventh processing mode, System mode, uses the User mode registers. System mode runs tasks that require a privileged processor mode and enables them to invoke all classes of exceptions.

Status Registers

All other processor states are held in status registers. The current operating processor status is in the Current Program Status Register (CPSR). The CPSR holds:

- Four ALU flags (Negative, Zero, Carry, Overflow)
- An interrupt disable bit for each of the IRQ and FIQ interrupts
- A bit to indicate ARM or Thumb execution state
- Five bits to encode the current processor mode

All five exception modes also have a Saved Program Status Register (SPSR) that holds the CPSR of the task immediately before the exception occurred.

Conditional Execution

All ARM instructions can be executed conditionally and can optionally update the four condition code flags (Negative, Zero, Carry, and Overflow) according to their result. Fifteen conditions are implemented.

Classes of Instructions

The ARM and Thumb instruction sets can be divided into four broad classes of instruction:

- Data processing instructions
- Load and store instructions
- Branch instructions
- Coprocessor instructions

Data Processing Instructions

The data processing instructions operate on data held in general-purpose registers. Of the two source operands, one is always a register. The other has two basic forms:

- An immediate value
- A register value optionally shifted

If the operand is a shifted register, the shift can be an immediate value or the value of another register. Four types of shift can be specified. Most data processing instructions can perform a shift followed by a logical or arithmetic operation.

There are two classes of multiply instructions:

- Normal, 32 bit result
- Long, 64 bi resut variants.

Both types of multiply instruction can optionally perform an accumulate operation

Load and Store Instructions

There are two main types of laod and store instructions:

- Load or store the value of a single register
- Load or store multiple register values

Load and store single register instructions can transfer a 32-bit word, a 16-bit halfword, or an 8-bit byte between memory and a register. Byte and halfword loads can be automatically zero extended or sign extended as they are loaded. These instructions have three primary addressing modes:

- Offset
- Pre-indexed
- Post-indexed

The address is formed by adding an immediate, or register-based, positive, or negative offset to a base register. Register-based offsets can also be scaled with shift operations. Pre-indexed and post-indexed addressing modes update the base registers with the base plus offset calculation.

As the PC is a general-purpose register, a 32-bit balue can be loaded directly into the PC to perform a jump to any address in the 4GB memory space.

Load and store multiple instructions perform a block transfer of any number of the general purpose registers to, or from, memory. Four addressing modes are provided:

- Pre-increment addressing
- Post-increment addressing
- Pre-decrement addressing
- Post-decrement addressing

The base address is specified by a register value (that can be optionally updated after the transfer). As the subroutine return address and the PC values are in general-purpose registers, very efficient subroutine calls can be constructed.

Branch Instructions

As well as letting data processing or load instructions change control flow (by writing the PC) a standard branch instruction is provided with 24-bit signed offset, providing for forward and backward branches of up to 32 MB.

A branch with link (BL) instruction enables efficient subroutine calls. BL preserves the address of the instruction after the branch in R14 (Link register or LR). This lets a move instruction put the LR in to the PC and return to the instruction after the branch.

The branch and exchange (BX) instruction switches between ARM and Thumb instruction sets with the return address optionally preserving the operating mode of the calling subroutine.

Coprocessor Instructions

There are three types of coprocessor instructions:

- Coprocessor data processing instructions
- Coprocessor register transfer instructions
- Coprocessor data transfer instructions

5.1.5 JTAG Interface

The ARM933T debug interface is based on IEEE Std. 1149.1- 1990, standard test access port. The ARM922T contains hardware extensions for advanced debugging features. These are intended to ease the development of application software.

The debug extensions allow the core to be stopped by one of the following:

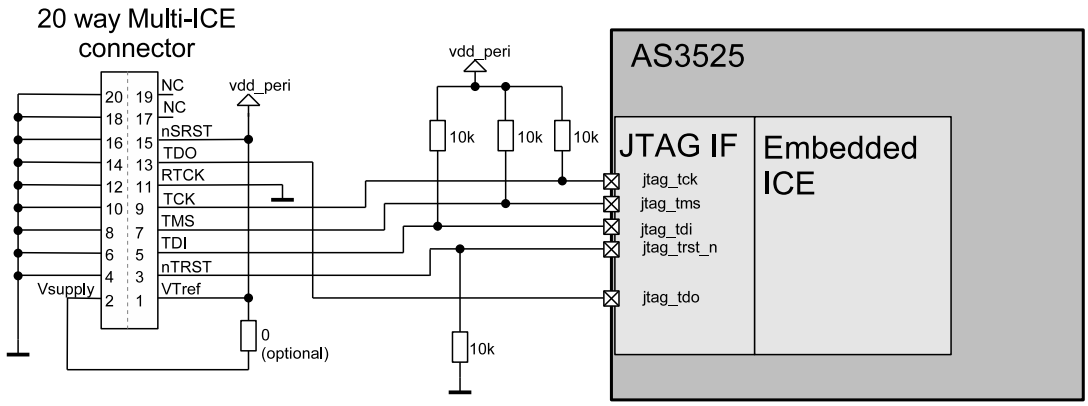
- A given instruction fetch (breakpoint)
- A data access (watchpoint)
- Asynchronously by a debug request

When this happens, the ARM922T is said to be in debug state. At this point, you can examine the internal state of the core and the external state of the system. When examination is complete, you can restore the core and system state and resume program execution.

Normally, all control for debugging is done by running a debugger software (ARM AXD or ARM Realview Debugger) on a debug host PC. Connection to the chip is done by an ARM Multi-ICE interface, which connects either to the parallel port or the USB port of the debug host PC.

The connection to the multi-ICE interface is done via a 20 way connector and ribbon cable. Following diagram shows the signals connections to the ICE connector.

Figure 5 Interface connector to multi-ICE



5.1.6 Boot Concept

It can be selected if the system should boot either using the internal ROM (internal boot loader) or an external ROM/Flash (connected to the MPMC interface). XPC[0] is read within global chip reset to do the selection of either internal or external boot.

Table 8 Boot definitions for internal/external boot selection

XPC[0]	Booting Option
1	Internal ROM
0	External ROM/Flash

For the internal bootloader, two chip versions are available: C21 and C22. Version C22 has additional features and is fully backward compatible to C21.

5.1.6.1 Internal Bootloader Version C21

Within the internal ROM boot loader several options for booting can be selected:

- SSP IF - SPI master for ST serial flash types
- SSP IF - SPI slave
- NandFlash
- Debug UART diagnostics

All boot loader options of the internal bootloader are configured by XPC[3:1] pins. External pull-up or pull-down resistors should be used to configure the boot options.

Table 9 Boot definitions Chip version C21

XPC[3:1]		Boot Device
0	000	SPI master ST M25Pxx serial Nor Flash
1	001	reserved
2	010	SPI slave
3	011	NandFlash (SB/BB - autodetect)
4	100	NandFlash (SB/BB - autodetect)
5	101	UART / Command Line Interface without diagnostics
6	110	UART / Command Line Interface without diagnostics
7	111	UART / Command Line Interface with diagnostics

5.1.6.2 Internal Bootloader Version C22

For chip version C22 the boot loader is extended with two additional features

- IDE boot: direct boot from harddisk
- USB boot promer. In the case that a USB connection is present and either an update button is pressed or there is no bootable device, the USB promer is started (see Figure 6 Boot decision between normal boot and USB boot promer" for details). The USB boot promer allows update of the firmware by using an USB mass storage class device. This update can be used either for initial programming (factory programming) or as mechanism for an in-field firmware update.

The C22 boot loader is fully compatible to the C21 boot loader except for mode 4, where the previous NAF boot mode is replaced by IDE boot. For version C22, NAF boot is only available in mode 3. Refer to Table 10 Boot definitions Chip version C22" for details.

The update button is located between xpa[4] and xpa[0]. Within the key scan routine, xpa[4] is driven shortly to each logic level "0" and "1" and the value of xpa[0] is read back to sense a keypress of the update button.

For the USB promer, it is necessary that frequency settings defining the quarz crystal frequency are defined by the pins xpc[3:1]. For details refer to "Table 11 USB promer frequency settings". These settings are read at the beginning in the initialisation routine of the bootloader.

Figure 6 Boot decision between normal boot and USB boot promer

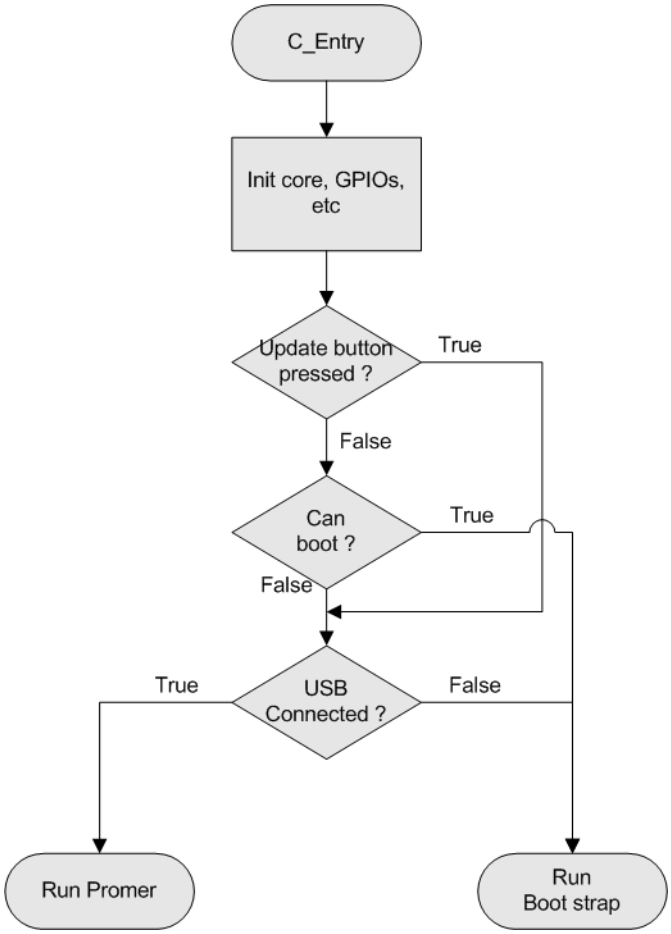


Table 10 Boot definitions Chip version C22

XPC[3:1]		Boot Device
0	000	SPI master ST M25Pxx serial Nor Flash
1	001	SPI master Atmel AT45DB011B serial Nor Flash
2	010	SPI slave
3	011	NandFlash
4	100	IDE
5	101	reserved for developers mode
6	110	UART / Command Line Interface without diagnostics
7	111	UART / Command Line Interface with diagnostics

Table 11 USB promer frequency settings

XPA[6:4]	USB promer frequency settings
000	24 MHz
001	20 MHz
010	13 MHz
011	12 MHz
100	10 MHz
others	reserved / defaults to 24 MHz

5.2 AHB Peripheral Blocks

ARM AHB ("advanced high-performance bus") is the new generation of AMBA bus, which is intended to address the requirements of high-performance synthesizable designs. AMBA AHB implements the features required for high performance, high clock frequency systems including:

- burst transfers
- split transactions
- single cycle bus master handover
- non-tristate implementation
- 32 bit bus width
- the clock frequency of the AHB can set by software up to 65MHz

5.2.1 2.5 MBIT RAM Main Memory

The memory subsystem consists of a RAM part and a ROM part.

Within the RAM memory subsystem, following functions are included:

- 1-TRAM controller with AHB bus slave interface
- 1-TRAM memory macros

5.2.1.1 1-TRAM Controller

The 1T RAM Controller is a slave interface connected to the AMBA AHB bus.

- slave AHB interface
- supports byte(8 bit), half-word(16 bit) and word(32 bit) read/write accesses
- 128-bit Line Buffer as temporary storage to reduce the number of memory accesses and optimise power consumption
- controls 5TSMC 1T-RAM instances

5.2.1.2 On-Chip 1T-RAM macro blocks

TSMC Emb1tRAM™ technology is a special kind of DRAM, which is implemented in a logic CMOS process. This innovative concept and design guarantees lowest power, high density, high performance and high yield advantages.

ECC (Error Correction Code) technique is applied in the macro to dynamically correct errors caused by hard defects or soft errors. No fuses are needed because the conventional redundancy scheme is replaced with ECC design in the macro.

The macro can be operated at clock rate from 20 MHz up to maximum AHB bus clock frequency in flow through random access mode. In the product, one idle cycle for refresh is needed in every 32 clock cycles.

Total 5 macros with organisation of 4Kx128 = 64 KByte each are implemented. For the refresh, one master macro is generating the refresh clock (T1F4Kx128_PIFE) and four macros are connected serially in slave mode to the refresh clock (T1F4Kx128PIFES).

Features

- 20 Mhz to 65 Mhz operation speed
- Flow through random access
- Built-in error correction (ECC)
- 128-bit wide data bus
- Separated data in/out bus
- SRAM-style interface operation
- Built-in refresh controller with refresh clock generator

5.2.2 On-Chip ROM

5.2.2.1 ROM Controller

The ROM controller implements the AHB slave interface for accessing the ROM.

The ROM controller generates OK response for all reads and error response for all writes.

Access width is always 32 bits.

5.2.2.2 1MBIT ROM

128 KByte of on-chip mask-programmable ROM are included.

The ROM is metal mask programmable by a single mask change (VIA2).

The ROM contains the following firmware package

- Boot loader

5.2.2.3 ROM versions and chip versions

There are two versions of the chip with changed Bootloader functionality available

- Version C21: Bootloader supports basic function for boot from external Nor Flash (ST or ATMEL).
- Version C22: Bootloader supports extended boot functions

These two chip versions differ only in the ROM content.

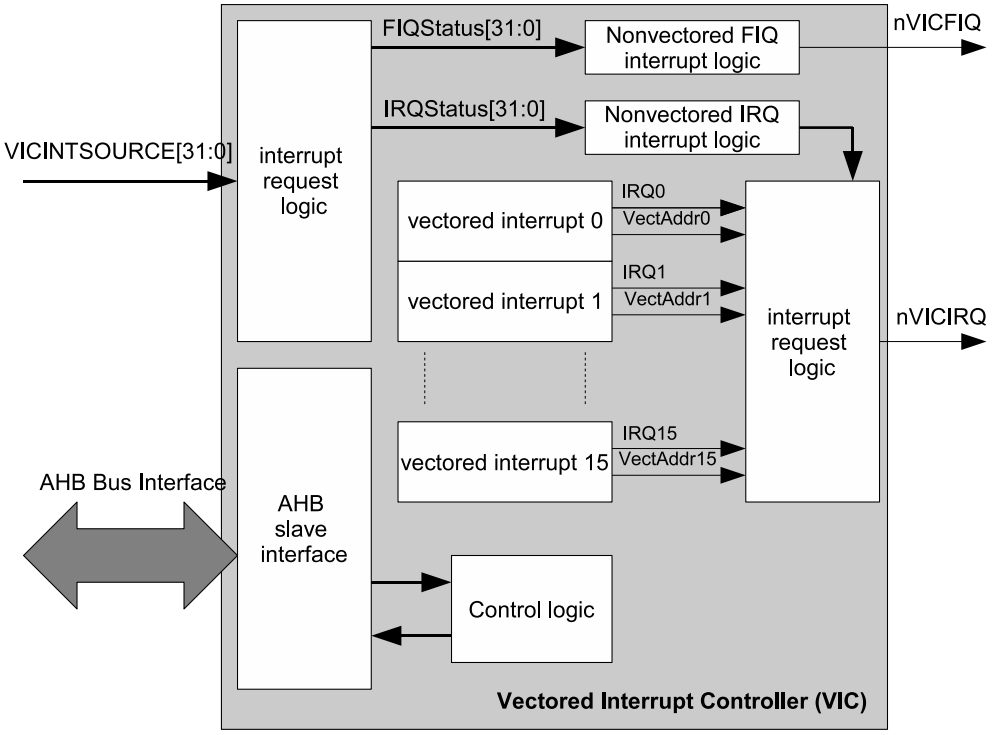
5.2.3 VIC – Vectored Interrupt Controller

The ARM PrimeCell™ PL190 “vectored interrupt controller” is included in the AHB system.

5.2.3.1 Features

- AMBA specification Rev 2.0 compliant
- support for 32 standard interrupts
- support for 16 vectored interrupts
- hardware interrupt priority
- IRQ and FIQ generation
- AHB mapped for fast interrupt response
- software interrupt generation
- test registers
- raw interrupt status
- interrupt request status
- interrupt masking
- privileged mode supportBlock Diagram

Figure 7 VIC Block Diagram



5.2.3.2 VIC Interrupt Sources

Table 12 VIC Interrupt Sources

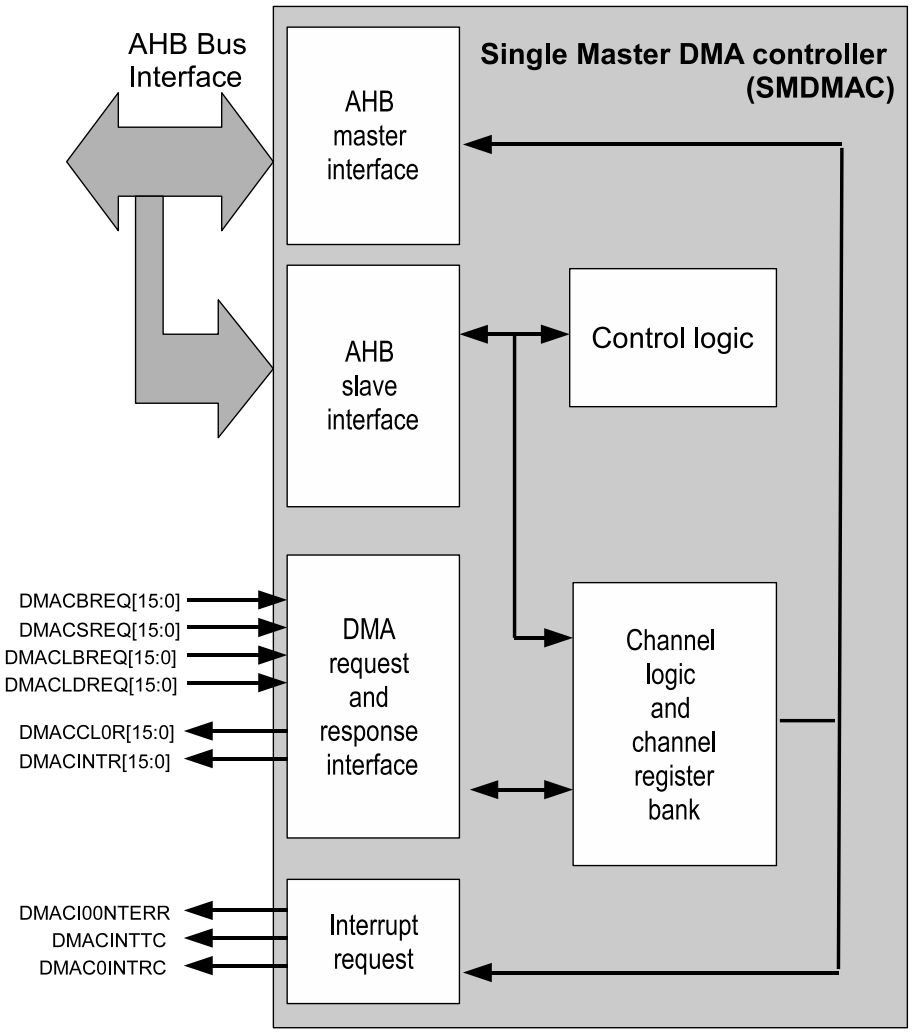
IRQ Source	Module	IRQ Source	Module
0	Watchdog	16	GPIO4 (XPD)
1	Timer 1	17	-
2	Timer 2	18	CGU
3	USB	19	Memory Stick
4	DMAC	20	DBOP
5	Nand Flash	21	-
6	IDE	22	-
7	MCI INTR0	23	-
8	MCI INTR1	24	-
9	AUDIO IRQ	25	-
10	SSP	26	-
11	I2C MS	27	-
12	I2C Audio	28	-
13	I2SIN	29	GPIO1 (XPA)
14	I2SOUT	30	GPIO2 (XPB)
15	UART	31	GPIO3 (XPC)

5.2.4 SMDMAC - Single master DMAC

The ARM PrimeCell™ PL081 “SMDMAC single master DMA controller” is included in the AHB system.

- AMBA specification Rev 2.0 compliant
- two DMA channels. Each channel can support a unidirectional transfer
- provides 16 peripheral DMA request lines
- single DMA and burst DMA request signals. Each peripheral connected to the PrimeCell™ SMDMAC can assert either a burst DMA request or a single DMA request. The DMA burst size is set by programming the PrimeCell™ SMDMAC
- Memory-to-Memory, memory-to-peripheral, peripheral-to-memory and peripheral-to-peripheral transfers.
- Scatter or gather DMA is supported through the use of linked lists. This means that the source and destination areas do not need to occupy contiguous areas of memory
- Hardware DMA channel priority. Each DMA channel has a specific hardware priority. DMA channel 0 has the highest priority and channel 1 has the lowest priority. If requests from two channels become active at the same time the channel with the highest priority is serviced first.
- AHB slave DMA programming interface. The PrimeCell™ SMDMAC is programmed by writing to the DMA control registers over the AHB slave interface
- One AHB bus master for transferring data. This interface is used to transfer data when a DMA request goes active.

Figure 8 SMDMAC Block Diagram



5.2.4.1 DMAC Registers

Table 13 DMAC Registers

Register Name	Base Address	Offset	Note
DMAC_IntStatus	AS3525_DMAC_BASE	0x000	Interrupt status register
DMAC_IntTCStatus	AS3525_DMAC_BASE	0x004	Interrupt terminal count status register
DMAC_IntTCClear	AS3525_DMAC_BASE	0x008	Interrupt terminal count clear register
DMAC_IntErrorStatus	AS3525_DMAC_BASE	0x00C	Interrupt error status register
DMAC_IntErrorClear	AS3525_DMAC_BASE	0x010	Interrupt error clear register
DMAC_RawIntTCStatus	AS3525_DMAC_BASE	0x014	Raw interrupt terminal count status register
DMAC_RawIntErrorStatus	AS3525_DMAC_BASE	0x018	Raw interrupt error status register
DMAC_SoftBReq	AS3525_DMAC_BASE	0x020	Software burst request register
DMAC_SoftSReq	AS3525_DMAC_BASE	0x024	Software single request register
DMAC_SoftLBReq	AS3525_DMAC_BASE	0x028	Software last burst request register
DMAC_SoftSBReq	AS3525_DMAC_BASE	0x02C	Software last single request register
DMAC_Configuration	AS3525_DMAC_BASE	0x030	Configuration register
DMAC_Sync	AS3525_DMAC_BASE	0x034	Synchronisation register
DMAC_C0SrcAddr	AS3525_DMAC_BASE	0x100	Channel 0 source address
DMAC_C0DestAddr	AS3525_DMAC_BASE	0x104	Channel 0 destination address
DMAC_C0LLI	AS3525_DMAC_BASE	0x108	Channel 0 linked list item register
DMAC_C0Control	AS3525_DMAC_BASE	0x10C	Channel 0 control register
DMAC_C0Configuration	AS3525_DMAC_BASE	0x110	Channel 0 configuration register
DMAC_C1SrcAddr	AS3525_DMAC_BASE	0x120	Channel 1 source address
DMAC_C1DestAddr	AS3525_DMAC_BASE	0x124	Channel 1 destination address
DMAC_C1LLI	AS3525_DMAC_BASE	0x128	Channel 1 linked list item register
DMAC_C1Control	AS3525_DMAC_BASE	0x12C	Channel 1 control register
DMAC_C1Configuration	AS3525_DMAC_BASE	0x130	Channel 1 configuration register
DMAC_PeripheralId0	AS3525_DMAC_BASE	0xFE0	peripheral ID0 register
DMAC_PeripheralId1	AS3525_DMAC_BASE	0xFE4	peripheral ID1 register
DMAC_PeripheralId2	AS3525_DMAC_BASE	0xFE8	peripheral ID2 register
DMAC_PeripheralId3	AS3525_DMAC_BASE	0xFEC	peripheral ID3 register
DMAC_CellId0	AS3525_DMAC_BASE	0xFF0	peripheral cell ID0 register
DMAC_CellId1	AS3525_DMAC_BASE	0xFF4	peripheral cell ID1 register
DMAC_CellId2	AS3525_DMAC_BASE	0xFF8	peripheral cell ID2 register
DMAC_CellId3	AS3525_DMAC_BASE	0xFFC	peripheral cell ID3 register

5.2.5 Multi Port Memory Controller (MPMC)

The MPMC block is integrated into the AMBA system through AHB slave port.

The PrimeCell™ MPMC offers:

- AMBA 32-bit AHB compliance.
- Dynamic memory interface support including SDRAM and JEDEC low-power SDRAM
- Asynchronous static memory device support including RAM, ROM, and Flash, with or without asynchronous page mode.
- Low transaction latency.
- Read and write buffers to reduce latency and to improve performance.
- Single AHB interface for accessing external memory.
- 8-bit and 16-bit wide static memory support.
- 16-bit wide chip select SDRAM memory support.
- Static memory features include:
  - asynchronous page mode read
  - programmable wait states
  - bus turnaround delay
  - output enable, and write enable delays
  - extended wait
- Two chip selects for synchronous memory and two chip selects for static memory devices.
- Software controllable HCLK to MPMCCLKOUT ratio.
- Power-saving modes dynamically control SDRAM MPMCCKEOUT and MPMCCLKOUT.
- Dynamic memory self-refresh mode supported by software.
- Controller supports 2K, 4K, and 8K row address synchronous memory parts. That is typical 512MB, 256MB, 128MB, and 16Mb parts, with 8, 16 bits per device.
- Two reset domains enable dynamic memory contents to be preserved over a soft reset.
- A separate AHB interface to program the MPMC. This enables the PrimeCell™ MPMC registers to be situated in memory with other system peripheral registers.
- Locked AHB transactions supported.
- Support for all AHB burst types.

Figure 9 Multi Port Memory Controller Block Diagram