

## Features

- DIOPSIS® Dual Core System Integrating an ARM926EJ-S™ ARM® Thumb® Processor Core and a mAgicV VLIW DSP of the Magic DSP™ family, optimized for Audio, Communication and Beam-forming Applications
- High Performance MagicV VLIW DSP
  - 1 GFLOPS - 1.6 Gops at 100 MHz
  - AHB Master Port, integrated DMA Engine and AHB Slave Port
  - Up to 10 Arithmetic Operations per Cycle (4 Multiply, 2 Add/subtract, 1 Add, 1 Subtract 40-bit Floating Point and 32-bit Integer) Allowing Single Cycle FFT Butterfly
  - Native Support for Complex Arithmetic and Vectorial SIMD Operations: One Complex Multiply with Dual Add/sub per Clock Cycle or Two Real Multiply and Two Add/sub or Simple Scalar Operations
  - 32-bit Integer and IEEE® 40-bit Extended Precision Floating Point Numeric Format
  - 16-port Data Register File: 256 Registers Organized in Two 128-register Banks
  - 5-issue predicated VLIW Architecture with Orthogonal ISA, Code Compression and Hardware Support for Code Efficient Software Pipeline Loops
  - 6 Accesses per Cycle Data Memory System (4 Accesses per Cycle for VLIW Operations + 2 Accesses per Cycle for DMA Transfers) supported by Flexible Addressing Capability
  - 2 Independent Address Generation Units Operating on a 64 Registers Address Register File Supporting Complex or Micro-Vectorial Accesses, and DSP features: Programmable Stride and Circular Buffers
  - 1.7 Mbits of On-chip SRAM:
  - 16 K x 40-bit Data Memory Locations (6 Memory Accesses per Cycle)
  - 8 K x 128-bit Dual Port Program Memory Location, Equivalent to ~50K DSP Assembler Instructions (typical) thanks to Code Compression and SW Pipelining
  - DMA Access to the External Program and Data Memory
  - Three Main Operating Modes: Run, Debug and Sleep Modes
  - User Mode and Privileged Interrupt Service Mode
  - Efficient Optimizing Assembler and C-Oriented Architecture: Allows Easy Exploitation of the Available Hardware Parallelism
- ARM926EJ-S ARM Thumb Processor
  - DSP instruction extensions
  - ARM Jazelle® Technology for Java® Acceleration
  - 16-KByte Data Cache, 16-KByte Instruction Cache, Write Buffer
  - 220MIPS at 200MHz
  - Memory Management Unit
  - EmbeddedICE™ In-circuit Emulation, Debug Communication Channel Support
- Efficient ARM - DSP Interface through AHB master and slave ports, Memory Mapped Registers and Ports, Interrupt Lines and Semaphores
- Additional Embedded Memories
  - 32-KByte of internal ROM, two-cycle access at maximum bus speed
  - 48-KByte of internal SRAM, single-cycle access at maximum processor or bus speed
- External Bus Interface (EBI)
  - Supports SDRAM, Static Memory, SmartMedia® and NAND Flash, CompactFlash®
- USB
  - USB 2.0 Full Speed (12 Mbits per second) Host Double Port



**DIOPSIS 940HF  
ARM926EJ-S PLUS  
ONE GFLOPS DSP**

**AT572D940HF**

**Preliminary**

**Summary**

**NOTE:** This is a summary document. The complete document is available under NDA. For more information, please contact your local Atmel sales office.

7010AS–DSP–07/07



- Dual On-chip Transceivers
- Integrated FIFOs and Dedicated DMA Channels
- USB 2.0 Full Speed (12 Mbits per second) Device Port
- On-chip Transceiver, 2-Kbyte Configurable Integrated FIFOs
- Two dedicated PDC channels
- Ethernet MAC 10/100
  - Reduced Media Independent Interface (RMII) to Physical Layer
  - Integrated DMA channel
- AHB bus Matrix
  - Seven Masters and Five Slaves Handled
  - Boot Mode Select Option
  - Remap Command
- System Controller (SYSC)
  - Reset Controller
  - Periodic Interval Timer, Watchdog and Real-Time Timer
- Power Management Controller (PMC)
  - Very Slow Clock (32768Hz) Operating Mode
  - Software Programmable Power Optimization Capabilities
  - 3 to 20 MHz On-chip Oscillator and two PLLs
  - Four Programmable External Clock Signals
- Advanced Interrupt Controller (AIC)
  - Individually Maskable, Eight-level Priority, Vectored Interrupt Sources
  - Three External Interrupt Sources and One Fast Interrupt Source, Spurious Interrupt Protected
- Three 32-bit Parallel Input/Output Controllers (PIO)
  - 96 Programmable I/O Lines Multiplexed with up to Two Peripheral I/Os
  - Input Change Interrupt Capability on Each I/O Line
  - Individually Programmable Open-drain, Pull-up resistor and Synchronous Output
- Twenty-three Peripheral Data Controller (PDC) Channels
- Debug Unit (DBGU)
  - 2-wire USART and support for Debug Communication Channel, Programmable ICE Access Prevention
  - Two dedicated PDC channels
- Four Synchronous Serial Controllers (SSC)
  - Two Independent Clock and Frame Sync Pair Signals for Each Receiver and Transmitter
  - I<sup>2</sup>S Analog Interface Support, Time Division Multiplex Support
  - High-speed Continuous Data Stream Capabilities with 32-bit Data Transfer
  - Two dedicated PDC channels for each SSC
- Three Universal Synchronous/Asynchronous Receiver Transmitters (USART)
  - Individual Baud Rate Generator, IrDA<sup>®</sup> Infrared Modulation/Demodulation
  - Support for ISO7816 T0/T1 Smart Card, Hardware and Software Handshaking, RS485 Support
  - Two dedicated PDC channels for each USART
- Two Master/Slave Serial Peripheral Interface (SPI)
  - 8- to 16-bit Programmable Data Length, Four External Peripheral Chip Selects
  - Two dedicated PDC channels for each SPI
- One Three-channel 16-bit Timer/Counters (TC)
  - Three External Clock Inputs, Two multi-purpose I/O Pins per Channel
  - Double PWM Generation, Capture/Waveform Mode, Up/Down Capability
- Two Two-Wire Interfaces (TWI)
  - Master Mode Support, All Two-wire Atmel EEPROM's Supported
- Two CAN Interfaces
  - Fully compliant with CAN 2.0 Part A and 2.0 Part B

- Multimedia Card Interface (MCI)
  - Automatic Protocol Control and Fast Automatic Data Transfers with PDMA, MMC and SDCard Compliant
- IEEE 1149.1 JTAG Boundary Scan on All Digital Pins
- Required Power Supplies:
  - 1.1V / 1.2V for VDDCORE and VDDOSC
  - 3.3V for VDDPLLA
  - 3.3V for VDDIOP (Peripheral I/Os) and for VDDIOM (Memory I/Os)
- Available in 324-ball CABGA Package

## 1. Description

DIOPSIS 940HF is a Dual CPU Processor integrating a mAgicV VLIW DSP and an ARM926EJ-S RISC MCU, plus a total of 370 Kbytes SRAM. The system combines the flexibility of the ARM926™ RISC controller with the very high performance of the DSP.

mAgicV is a high performance VLIW DSP of the Magic DSP family, delivering 1 Giga floating-point operations per second (GFLOPS) and 1.6 Gops at a clock rate of 100 MHz. It is equipped with an AHB master port and an AHB slave port for system-on-chip integration. It has 256 data registers, 64 address registers, 10 independent arithmetic operating units, 2 independent address generation units and a DMA engine. To sustain the internal parallelism, the data bandwidth among the Register File, the Operators and the Data Memory System, is 80 bytes/cycle. The Data Memory System is designed to transfer 28 bytes/cycle. For instance, mAgicV can produce one complete FFT butterfly per cycle by activating all the computing units. mAgicV operates on IEEE 754 40-bit extended precision floating-point and 32-bit integer numeric format for numerical computations, while internal memory accesses are supported by a powerful 16-bit MAGU (Multiple Address Generation Unit). It has also on-chip 16K x 40-bit 6-access/cycle data memory system and 8K x 128-bit dual port program memory locations. Efficient usage of the internal program memory is achieved through a general purpose code compression mechanism and software pipelining support of systematic loops.

A C-oriented architecture and an optimizing assembler ease the user from the burden of dealing with the parallelism of the processor resources and significantly simplifies the code development. A rich library of C-callable DSP routines is available.

The ARM926 embedded micro controller core is a member of the Advanced RISC Machines (ARM) family of general purpose 32-bit microprocessors, which offer high performance and very low power consumption. The ARM architecture is based on Reduced Instruction Set Computer (RISC) principles, and the instruction set and the related decode mechanism are much simpler than the micro programmed Complex Instruction Set Computers.

This simplicity results in a high instruction throughput and impressive real-time interrupt response. The ARM926 supports 16-bit Thumb subset of the most commonly used 32-bit instructions. These are expanded at run time with no degradation of the system performance. This gives 16-bit code density (saving memory area and cost) coupled with a 32-bit processor performance.

A rich set of peripherals and a 48 Kbytes internal memory provide a highly flexible and integrated system solution.

The ARM926EJ-S supports the Jazelle technology for Java acceleration.

## 2. Ball Configuration

**Table 2-1.** AT572D940HF Ball Assignment (I/O: 191 balls)

| Name         | Pin | Name       | Pin | Name            | Pin | Name   | Pin |
|--------------|-----|------------|-----|-----------------|-----|--------|-----|
| A0/NBS0      | B2  | D5         | K7  | NCS2            | B7  | PIOA27 | G9  |
| A1/NBS2/NWR2 | C2  | D6         | K5  | NCS3/SM_NCS     | E7  | PIOA28 | J9  |
| A2           | C1  | D7         | K1  | NRD/NOE/CF_NOE  | B6  | PIOA29 | A8  |
| A3           | D4  | D8         | K2  | NRST            | J17 | PIOA30 | D8  |
| A4           | D3  | D9         | K6  | NWR0/NWE/CF_NWE | C6  | PIOA31 | B8  |
| A5           | D1  | D10        | K8  | NWR1/NBS1/CFIOR | D6  | PIOB0  | U8  |
| A6           | E4  | D11        | L5  | NWR3/NBS3/CFIOW | G7  | PIOB1  | L9  |
| A7           | E3  | D12        | L1  | PIOA0           | F11 | PIOB2  | P9  |
| A8           | F6  | D13        | L2  | PIOA1           | C11 | PIOB3  | R9  |
| A9           | G6  | D14        | L4  | PIOA2           | A11 | PIOB4  | V9  |
| A10          | F3  | D15        | L7  | PIOA3           | B11 | PIOB5  | L10 |
| A11          | H8  | D16        | M3  | PIOA4           | H10 | PIOB6  | N10 |
| A12          | F2  | D17        | L8  | PIOA5           | G10 | PIOB7  | V10 |
| A13          | F1  | D18        | M4  | PIOA6           | D10 | PIOB8  | T10 |
| A14          | G3  | D19        | M5  | PIOA7           | B17 | PIOB9  | P10 |
| A15          | H7  | D20        | M6  | PIOA8           | A17 | PIOB10 | M10 |
| A16/SD_BA0   | G1  | D21        | N1  | PIOA9           | B16 | PIOB11 | N11 |
| A17/SD_BA1   | G2  | D22        | M7  | PIOA10          | A16 | PIOB12 | M11 |
| A18          | H6  | D23        | N4  | PIOA11          | C15 | PIOB13 | L11 |
| A19          | H3  | D24        | N5  | PIOA12          | H17 | PIOB14 | U12 |
| A20          | J8  | D25        | P1  | PIOA13          | V15 | PIOB15 | T12 |
| A21          | H2  | D26        | P3  | PIOA14          | U15 | PIOB16 | R12 |
| A_JCFG       | N16 | D27        | P4  | PIOA15          | V16 | PIOB17 | N12 |
| A_RTCK       | M17 | D28        | P5  | PIOA16          | T15 | PIOB18 | V13 |
| A_TCK        | N17 | D29        | R1  | PIOA17          | V17 | PIOB19 | U13 |
| A_TDI        | M14 | D30        | R2  | PIOA18          | T16 | PIOB20 | T13 |
| A_TDO        | M16 | D31        | R3  | PIOA19          | T17 | PIOB21 | P13 |
| A_TMS        | N15 | M_NTRST    | E16 | PIOA20          | U18 | PIOB22 | V14 |
| A_NTRST      | M13 | M_TCK      | F13 | PIOA21          | T18 | PIOB23 | R14 |
| D0           | H1  | M_TDI      | E15 | PIOA22          | R15 | PIOB24 | J10 |
| D1           | J7  | M_TDO      | E14 | PIOA23          | R18 | PIOB25 | H15 |
| D2           | J2  | M_TMS      | E17 | PIOA24          | H16 | PIOB26 | B12 |
| D3           | J1  | NCS0       | F7  | PIOA25          | B9  | PIOB27 | A12 |
| D4           | K9  | NCS1/SD_CS | A6  | PIOA26          | D9  | PIOB28 | F9  |

**Table 2-1.** AT572D940HF Ball Assignment (I/O: 191 balls) (Continued)

| Name   | Pin | Name   | Pin | Name    | Pin | Name     | Pin |
|--------|-----|--------|-----|---------|-----|----------|-----|
| PIOB29 | B10 | PIOC11 | L13 | PIOC25  | K15 | SD_NWE   | B4  |
| PIOB30 | A10 | PIOC12 | L18 | PIOC26  | K11 | TEST     | J18 |
| PIOB31 | A9  | PIOC13 | K12 | PIOC27  | K10 | USBD_DM  | N8  |
| PIOC0  | D15 | PIOC14 | H13 | PIOC28  | E12 | USBD_DP  | P8  |
| PIOC1  | D14 | PIOC15 | G17 | PIOC29  | D12 | USBHA_DM | R7  |
| PIOC2  | C14 | PIOC16 | G18 | PIOC30  | P16 | USBHA_DP | T7  |
| PIOC3  | D13 | PIOC17 | G14 | PIOC31  | P17 | USBHB_DM | U7  |
| PIOC4  | C13 | PIOC18 | F17 | PLL_RCA | U2  | USBHB_DP | V7  |
| PIOC5  | G12 | PIOC19 | H14 | PLL_RCB | P6  | XIN      | U5  |
| PIOC6  | F12 | PIOC20 | F16 | SD_A10  | A7  | XOUT     | V5  |
| PIOC7  | G13 | PIOC21 | E18 | SD_CK   | B5  | X32EN    | N7  |
| PIOC8  | F18 | PIOC22 | K14 | SD_CKE  | C5  | X32IN    | V2  |
| PIOC9  | M18 | PIOC23 | K16 | SD_NCAS | A4  | X32OUT   | V3  |
| PIOC10 | L12 | PIOC24 | K17 | SD_NRAS | D5  |          |     |

**Table 2-2.** AT572D940HF Ball Assignment (Power and Ground: 127 balls)

| Name    | Pin | Name    | Pin | Name     | Pin | Name    | Pin |
|---------|-----|---------|-----|----------|-----|---------|-----|
| VDDCORE | F4  | VDDIOM  | B3  | VDDIOP   | T9  | VDDPLLA | T3  |
| VDDCORE | J4  | VDDIOM  | E5  | VDDIOP   | V8  | GND     | D2  |
| VDDCORE | L6  | VDDIOM  | E1  | VDDIOP   | F14 | GND     | E2  |
| VDDCORE | T2  | VDDIOM  | G4  | VDDIOP   | G16 | GND     | F5  |
| VDDCORE | M9  | VDDIOM  | H4  | VDDIOP   | H18 | GND     | G5  |
| VDDCORE | P11 | VDDIOM  | J5  | VDDIOP   | J15 | GND     | H5  |
| VDDCORE | T14 | VDDIOM  | K3  | VDDIOP   | K13 | GND     | J6  |
| VDDCORE | N13 | VDDIOM  | M2  | VDDIOP   | L16 | GND     | J3  |
| VDDCORE | L15 | VDDIOM  | N3  | VDDIOP   | M12 | GND     | K4  |
| VDDCORE | J13 | VDDIOM  | P2  | VDDIOP   | N14 | GND     | L3  |
| VDDCORE | H11 | VDDIOMP | E9  | VDDIOP   | U17 | GND     | M1  |
| VDDCORE | D16 | VDDIOMP | G8  | VDDIOP   | P14 | GND     | N2  |
| VDDCORE | E13 | VDDIOP  | C10 | VDDIOP   | P12 | GND     | N6  |
| VDDCORE | H9  | VDDIOP  | D11 | VDDIOP   | U11 | GND     | R4  |
| VDDCORE | E8  | VDDIOP  | G11 | VDDIOP   | R10 | GND     | T1  |
| VDDCORE | A2  | VDDIOP  | A13 | VDDIOP   | V6  | GND     | T8  |
| VDDIOM  | D7  | VDDIOP  | A15 | VDDOSC32 | U4  | GND     | R8  |
| VDDIOM  | A5  | VDDIOP  | C16 | VDDOSCM  | R5  | GND     | N9  |

**Table 2-2.** AT572D940HF Ball Assignment (Power and Ground: 127 balls) (Continued)

| Name | Pin | Name | Pin | Name | Pin | Name     | Pin |
|------|-----|------|-----|------|-----|----------|-----|
| GND  | U10 | GND  | J14 | GND  | E10 | GND      | R16 |
| GND  | V11 | GND  | J12 | GND  | C9  | GND      | R17 |
| GND  | R11 | GND  | H12 | GND  | C8  | GND      | M15 |
| GND  | V12 | GND  | G15 | GND  | C7  | GND      | C18 |
| GND  | R13 | GND  | F15 | GND  | E6  | GND      | C17 |
| GND  | U14 | GND  | D18 | GND  | A3  | GND      | B18 |
| GND  | U16 | GND  | D17 | GND  | C4  | GND      | C3  |
| GND  | P15 | GND  | B15 | GND  | U6  | GND      | B1  |
| GND  | P18 | GND  | B14 | GND  | V4  | GND      | T6  |
| GND  | N18 | GND  | B13 | GND  | M8  | GND      | J11 |
| GND  | L14 | GND  | C12 | GND  | U9  | GNDOSC32 | T5  |
| GND  | J16 | GND  | E11 | GND  | T11 | GNDOSCM  | P7  |
| GND  | L17 | GND  | F8  | GND  | U1  | GNDPLLA  | U3  |
| GND  | K18 | GND  | F10 | GND  | A14 |          |     |

All pins not comprised in [Table 2-1](#) and [Table 2-2](#) are “not connected”.

## 2.1 Pin Name Conventions

Pin names are built using the following structure:

(functional block name) \_ (activity level) (line name) (bus index)

where:

functional block name = name of the related functional block (when not a global function)

activity level = “N” for low active lines; blank for high active lines

line name = name of the function of the pin line

bus index = number corresponding to the index when the pin line is an element of a bus

### 3. Pin Description

**Table 3-1.** AT572D940HF Pin Description

| Module   | Name                | Function                              | Type   | Active Level | Notes                                                                            |
|----------|---------------------|---------------------------------------|--------|--------------|----------------------------------------------------------------------------------|
| AIC      | EXT_IRQ0 - EXT_IRQ2 | External Interrupt Request            | bi-03  |              | input through PIO line                                                           |
| AIC      | M_MODE              | Interrupt Request from mAgicV         | bi-03  |              | output through PIO line                                                          |
| AIC      | M_SIRQ0 - M_SIRQ3   | Generic Interrupt Request from mAgicV | bi-03  |              | output through PIO line                                                          |
| A JTAG   | A_JCFG              | ARM JTAG / Chip Boundary Scan select  | in     |              | internal pull-down resistor (ARM JTAG selected)                                  |
| A JTAG   | A_RTCK              | ARM JTAG Returned Test Clock          | out-03 |              |                                                                                  |
| A JTAG   | A_TCK               | ARM JTAG Test Clock                   | in     |              | no pull-up resistor                                                              |
| A JTAG   | A_TDI               | ARM JTAG Test Data Input              | in     |              | no pull-up resistor                                                              |
| A JTAG   | A_TDO               | ARM JTAG Test Data Output             | out-03 |              |                                                                                  |
| A JTAG   | A_TMS               | ARM JTAG Test Mode Select             | in     |              | no pull-up resistor                                                              |
| CAN      | CAN0_RX             | CAN 0 bus Data in                     | bi-03  |              | input through PIO line                                                           |
| CAN      | CAN0_TX             | CAN 0 bus Data out                    | bi-03  |              | output through PIO line                                                          |
| CAN      | CAN1_RX             | CAN 1 bus Data in                     | bi-03  |              | input through PIO line                                                           |
| CAN      | CAN1_TX             | CAN 1 bus Data out                    | bi-03  |              | output through PIO line                                                          |
| CF Logic | CF_NCE1- CF_NCE2    | CompactFlash Chip Enable              | bi-03  | low          | output through PIO line                                                          |
| CF Logic | CF_NOE              | CompactFlash Output Enable            | out-03 | low          |                                                                                  |
| CF Logic | CF_NWE              | CompactFlash Write Enable             | out-03 | low          |                                                                                  |
| CF Logic | CF_NIOR             | CompactFlash IO Read                  | out-03 | low          |                                                                                  |
| CF Logic | CF_NIOW             | CompactFlash IO Write                 | out-03 | low          |                                                                                  |
| CF Logic | CF_RNW              | CompactFlash Read Not Write           | bi-03  |              | output through PIO line                                                          |
| CF Logic | CF_NCS0 - CF_NCS1   | CompactFlash Chip Select              | bi-03  | low          | output through PIO line                                                          |
| DBGU     | DBG_RXD             | Debug Serial Line Data in             | bi-03  |              | input through PIO line                                                           |
| DBGU     | DBG_TXD             | Debug Serial Line Data out            | bi-03  |              | output through PIO line                                                          |
| EBI      | A0 - A21            | Address Bus                           | out-03 |              | 0 at reset                                                                       |
| EBI      | A22 - A25           | Address Bus                           | bi-03  |              | output through PIO line<br>0 at reset                                            |
| EBI      | D0- D31             | Data Bus                              | bi-03  |              | Pulled-up input at reset                                                         |
| EBI      | NWAIT               | External Wait Signal                  |        | low          | input through PIO line                                                           |
| EBI      | BMS                 | Boot Memory Select                    | bi-03  |              | input through PIO line<br>1→ external boot selected<br>0→ internal boot selected |
| ETH      | E_RXER              | Ethernet RMI Receive Error            | bi-03  |              | input through PIO line                                                           |

**Table 3-1.** AT572D940HF Pin Description (Continued)

| Module | Name            | Function                               | Type   | Active Level | Notes                                                                          |
|--------|-----------------|----------------------------------------|--------|--------------|--------------------------------------------------------------------------------|
| ETH    | E_TXD0 - E_TXD1 | Ethernet RMII Transmit Data Bus        | bi-03  |              | output through PIO line                                                        |
| ETH    | E_TXEN          | Ethernet RMII Transmit Enable          | bi-03  |              | output through PIO line                                                        |
| ETH    | E_REFCK         | Ethernet RMII Reference Clock          | bi-03  |              | input through PIO line                                                         |
| ETH    | E_CRSDV         | Ethernet RMII Carrier Sense/Data Valid | bi-03  |              | input through PIO line                                                         |
| ETH    | E_RXD0 - E_RXD1 | Ethernet RMII Receive Data Bus         | bi-03  |              | input through PIO line                                                         |
| ETH    | E_FCE100        | Ethernet RMII Force 100 Mb/s operation | bi-03  | high         | output through PIO line                                                        |
| ETH    | E_MDIO          | Ethernet RMII PHY Management Data      | bi-03  |              | through PIO line                                                               |
| ETH    | E_MDCK          | Ethernet RMII PHY Management Clock     | bi-03  |              | output through PIO line                                                        |
| MCI    | MCCK            | Multimedia Card Clock                  | bi-03  |              | through PIO line                                                               |
| MCI    | MCCDA           | Multimedia Card Command                | bi-03  |              | through PIO line                                                               |
| MCI    | MCDA0-MCDA3     | Multimedia Card Data                   | bi-03  |              | through PIO line                                                               |
| M JTAG | M_NTRST         | mAgicV JTAG Test Reset                 | in     |              |                                                                                |
| M JTAG | M_TCK           | mAgicV JTAG Test Clock                 | in     |              | no pull-up resistor                                                            |
| M JTAG | M_TDI           | mAgicV JTAG Test Data Input            | in     |              | no pull-up resistor                                                            |
| M JTAG | M_TDO           | mAgicV JTAG Test Data Output           | out-03 |              |                                                                                |
| M JTAG | M_TMS           | mAgicV JTAG Test Mode Select           | in     |              | no pull-up resistor                                                            |
| OSC    | XIN             | Main Oscillator Quartz                 | in     |              |                                                                                |
| OSC    | XOUT            | Main Oscillator Quartz                 | out    |              |                                                                                |
| OSC    | X32IN           | Slow Clock Oscillator Quartz           | in     |              |                                                                                |
| OSC    | X32OUT          | Slow Clock Oscillator Quartz           | out    |              |                                                                                |
| OSC    | X32EN           | Slow Clock Oscillator Enable           | in     | high         | internal pull-up resistor (internal oscillator enabled)                        |
| PIO A  | PIOA0 - PIOA31  | Parallel Input/Output A                | bi-03  |              | general purpose programmable I/Os or peripheral I/Os; Pulled-up input at reset |
| PIO B  | PIOB0 - PIOB31  | Parallel Input/Output B                | bi-03  |              | general purpose programmable I/Os or peripheral I/Os; Pulled-up input at reset |
| PIO C  | PIOC0 - PIOC31  | Parallel Input/Output C                | bi-03  |              | general purpose programmable I/Os or peripheral I/Os; Pulled-up input at reset |
| PLL    | PLL_RCA         | PLL A Filter                           | in     |              |                                                                                |
| PLL    | PLL_RCB         | PLL B Filter                           | in     |              | to be left floating (test input)                                               |
| PMC    | A_CK            | ARM Clock                              | bi-03  |              | output through PIO line for test purpose                                       |

**Table 3-1.** AT572D940HF Pin Description (Continued)

| Module   | Name                  | Function                       | Type   | Active Level | Notes                                                                     |
|----------|-----------------------|--------------------------------|--------|--------------|---------------------------------------------------------------------------|
| PMC      | M_CK                  | mAgicV Clock                   | bi-03  |              | output through PIO line for test purpose                                  |
| PMC      | P_CK0-P_CK3           | Programmable Clock             | bi-03  |              | output through PIO line                                                   |
| SDRAMC   | SDCK                  | SDRAM Clock Output             | out-03 |              |                                                                           |
| SDRAMC   | SD_CKE                | SDRAM Clock Enable             | out-04 | high         |                                                                           |
| SDRAMC   | SD_NCS                | SDRAM Chip Select              | out-03 | low          |                                                                           |
| SDRAMC   | SD_BA0 - SD_BA1       | SDRAM Bank Select              | out-03 |              |                                                                           |
| SDRAMC   | SD_NWE                | SDRAM Write Enable             | out-04 | low          |                                                                           |
| SDRAMC   | SD_NRAS - SD_NCAS     | Row and Column Address Strobe  | out-04 | low          |                                                                           |
| SDRAMC   | SD_A10                | SDRAM Bus Address bit 10       | out-04 |              |                                                                           |
| SMC      | NCS0 - NCS3           | Chip Select Signal             | out-03 | low          | 1 at reset;                                                               |
| SMC      | NCS4 - NCS7           | Chip Select Signal             | bi-03  | low          | 1 at reset<br>output through PIO line                                     |
| SMC      | NWR0 - NWR3           | Write Signal                   | out-03 | low          | 1 at reset                                                                |
| SMC      | NOE                   | Output Enable                  | out-03 | low          | 1 at reset                                                                |
| SMC      | NRD                   | Read Signal                    | out-03 | low          | 1 at reset                                                                |
| SMC      | NWE                   | Write Enable                   | out-03 | low          | 1 at reset                                                                |
| SMC      | NBS0 - NBS3           | Byte Select                    | out-03 | low          | 1 at reset                                                                |
| SM Logic | SM_NOE                | SmartMedia Output Enable       | bi-03  | low          | output through PIO line                                                   |
| SM Logic | SM_NWE                | SmartMedia Write Enable        | bi-03  | low          | output through PIO line                                                   |
| SPI      | SPI0_MOSI             | SPI 0 Master Out/Slave In data | bi-03  |              | through PIO line<br>SPI SLV → data input<br>SPI MST → data output         |
| SPI      | SPI0_MISO             | SPI 0 Master In/Slave Out data | bi-03  |              | through PIO line<br>SPI SLV → data output<br>SPI MST → data input         |
| SPI      | SPI0_NCS0             | SPI 0 Input/Output Chip select | out-03 | low          | through PIO line<br>SPI SLV → CS Input<br>SPI MST → CS 0 Output           |
| SPI      | SPI0_NCS1 - SPI0_NCS3 | SPI 0 Output Chip Selects      | bi-03  | low          | output through PIO line<br>SPI SLV → n.a.<br>SPI MST → CS 3, 2, 1 Outputs |
| SPI      | SPI0_CK               | SPI 0 Serial clock             | bi-03  |              | through PIO line<br>SPI SLV → clock input<br>SPI MST → clock output       |

**Table 3-1.** AT572D940HF Pin Description (Continued)

| Module | Name                     | Function                                             | Type   | Active Level | Notes                                                                     |
|--------|--------------------------|------------------------------------------------------|--------|--------------|---------------------------------------------------------------------------|
| SPI    | SPI1_MOSI                | SPI 1 Master Out/Slave In data                       | bi-03  |              | through PIO line<br>SPI SLV → data input<br>SPI MST → data output         |
| SPI    | SPI1_MISO                | SPI 1 Master In/Slave Out data                       | bi-03  |              | through PIO line<br>SPI SLV → data output<br>SPI MST → data input         |
| SPI    | SPI1_NCS0                | SPI 1 Input/Output Chip select                       | out-03 | low          | through PIO line<br>SPI SLV → CS Input<br>SPI MST → CS 0 Output           |
| SPI    | SPI1_NCS1 -<br>SPI1_NCS3 | SPI 1 Output Chip Selects                            | bi-03  | low          | output through PIO line<br>SPI SLV → n.a.<br>SPI MST → CS 3, 2, 1 Outputs |
| SPI    | SPI1_CK                  | SPI 1 Serial clock                                   | bi-03  |              | through PIO line<br>SPI SLV → clock input<br>SPI MST → clock output       |
| SSC    | SSC0_TXD                 | Synchronous Serial Controller 0 Data Out             | bi-03  |              | output through PIO line                                                   |
| SSC    | SSC0_RXD                 | Synchronous Serial Controller 0 Data In              | bi-03  |              | input through PIO line                                                    |
| SSC    | SSC0_TF                  | Synchronous Serial Controller 0 Transmit Frame Clock | bi-03  |              | through PIO line                                                          |
| SSC    | SSC0_RF                  | Synchronous Serial Controller 0 Receive Frame Clock  | bi-03  |              | through PIO line                                                          |
| SSC    | SSC0_TK                  | Synchronous Serial Controller 0 Transmit Bit Clock   | bi-03  |              | through PIO line                                                          |
| SSC    | SSC0_RK                  | Synchronous Serial Controller 0 Receive Bit Clock    | bi-03  |              | through PIO line                                                          |
| SSC    | SSC1_TXD                 | Synchronous Serial Controller 1 Data Out             | bi-03  |              | output through PIO line                                                   |
| SSC    | SSC1_RXD                 | Synchronous Serial Controller 1 Data In              | bi-03  |              | input through PIO line                                                    |
| SSC    | SSC1_TF                  | Synchronous Serial Controller 1 Transmit Frame Clock | bi-03  |              | through PIO line                                                          |
| SSC    | SSC1_RF                  | Synchronous Serial Controller 1 Receive Frame Clock  | bi-03  |              | through PIO line                                                          |
| SSC    | SSC1_TK                  | Synchronous Serial Controller 1 Transmit Bit Clock   | bi-03  |              | through PIO line                                                          |
| SSC    | SSC1_RK                  | Synchronous Serial Controller 1 Receive Bit Clock    | bi-03  |              | through PIO line                                                          |
| SSC    | SSC2_TXD                 | Synchronous Serial Controller 2 Data Out             | bi-03  |              | output through PIO line                                                   |
| SSC    | SSC2_TF                  | Synchronous Serial Controller 2 Transmit Frame Clock | bi-03  |              | through PIO line                                                          |

**Table 3-1. AT572D940HF Pin Description (Continued)**

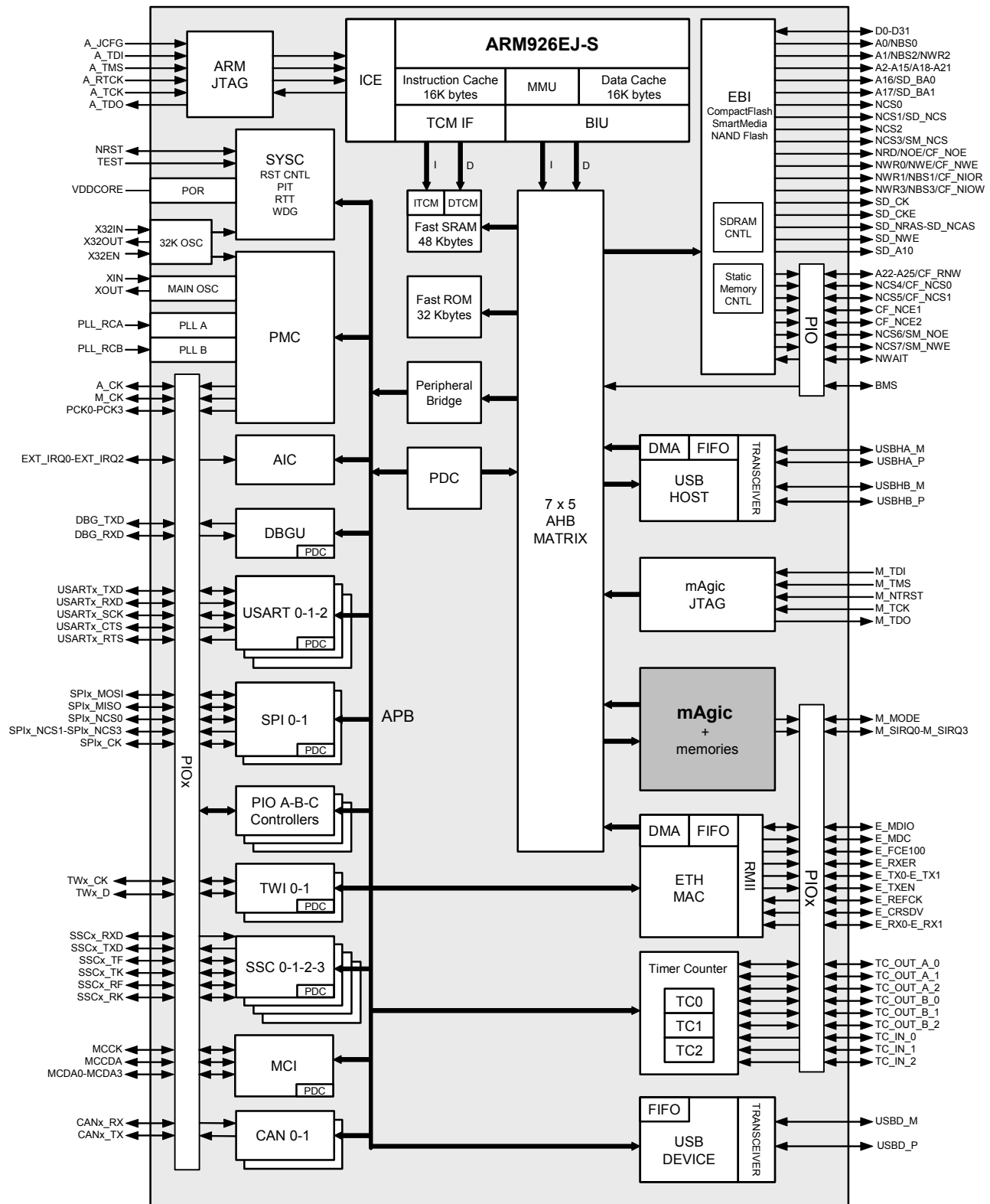
| Module | Name      | Function                                             | Type   | Active Level | Notes                                         |
|--------|-----------|------------------------------------------------------|--------|--------------|-----------------------------------------------|
| SSC    | SSC2_RF   | Synchronous Serial Controller 2 Receive Frame Clock  | bi-03  |              | through PIO line                              |
| SSC    | SSC2_TK   | Synchronous Serial Controller 2 Transmit Bit Clock   | bi-03  |              | through PIO line                              |
| SSC    | SSC2_RK   | Synchronous Serial Controller 2 Receive Bit Clock    | bi-03  |              | through PIO line                              |
| SSC    | SSC2_RXD  | Synchronous Serial Controller 2 Data In              | bi-03  |              | input through PIO line                        |
| SSC    | SSC3_TXD  | Synchronous Serial Controller 3 Data Out             | bi-03  |              | output through PIO line                       |
| SSC    | SSC3_RXD  | Synchronous Serial Controller 3 Data In              | bi-03  |              | input through PIO line                        |
| SSC    | SSC3_TF   | Synchronous Serial Controller 3 Transmit Frame Clock | bi-03  |              | through PIO line                              |
| SSC    | SSC3_RF   | Synchronous Serial Controller 3 Receive Frame Clock  | bi-03  |              | through PIO line                              |
| SSC    | SSC3_TK   | Synchronous Serial Controller 3 Transmit Bit Clock   | bi-03  |              | through PIO line                              |
| SSC    | SSC3_RK   | Synchronous Serial Controller 3 Receive Bit Clock    | bi-03  |              | through PIO line                              |
| SYSC   | NRST      | Chip Reset                                           | bi-03  | low          | open drain                                    |
| TC     | TC_OUT_A0 | Timer Counter A out 0                                | bi-03  |              | through PIO line                              |
| TC     | TC_OUT_A1 | Timer Counter A out 1                                | bi-03  |              | bidirectional through PIO line                |
| TC     | TC_OUT_A2 | Timer Counter A out 2                                | bi-03  |              | bidirectional through PIO line                |
| TC     | TC_OUT_B0 | Timer Counter B out 0                                | bi-03  |              | bidirectional through PIO line                |
| TC     | TC_OUT_B1 | Timer Counter B out 1                                | bi-03  |              | bidirectional through PIO line                |
| TC     | TC_OUT_B2 | Timer Counter B out 2                                | bi-03  |              | bidirectional through PIO line                |
| TC     | TC_IN_0   | Timer Counter in 0                                   | bi-03  |              | input through PIO line                        |
| TC     | TC_IN_1   | Timer Counter in 1                                   | bi-03  |              | input through PIO line                        |
| TC     | TC_IN_2   | Timer Counter in 2                                   | bi-03  |              | input through PIO line                        |
| TST    | TEST      | Test Mode Select                                     | in     | high         | pull-down resistor (Functional Mode selected) |
| TWI    | TW0_D     | Two Wire 0 Data                                      | bi-03  |              | bidirectional through PIO line                |
| TWI    | TW0_CK    | Two Wire 0 Clock                                     | bi-03  |              | bidirectional through PIO line                |
| TWI    | TW1_D     | Two Wire 1 Data                                      | bi-03  |              | bidirectional through PIO line                |
| TWI    | TW1_CK    | Two Wire 1 Clock                                     | bi-03  |              | bidirectional through PIO line                |
| USBD   | USBD_DM   | USB Device Port Data -                               | usb-bi |              |                                               |
| USBD   | USBD_DP   | USB Device Port Data +                               | usb-bi |              |                                               |
| USBH   | USBHA_DM  | USB Host Port A Data -                               | usb-bi |              |                                               |

**Table 3-1.** AT572D940HF Pin Description (Continued)

| Module | Name       | Function                           | Type   | Active Level | Notes                                                    |
|--------|------------|------------------------------------|--------|--------------|----------------------------------------------------------|
| USBH   | USBHA_DP   | USB Host Port A Data +             | usb-bi |              |                                                          |
| USBH   | USBHB_DM   | USB Host Port B Data -             | usb-bi |              |                                                          |
| USBH   | USBHB_DP   | USB Host Port B Data +             | usb-bi |              |                                                          |
| USART  | USART0_RXD | USART 0 Data in                    | bi-03  |              | input through PIO line                                   |
| USART  | USART0_TXD | USART 0 Data out                   | bi-03  |              | bidirectional through PIO line                           |
| USART  | USART0_SCK | USART 0 Serial clock               | bi-03  |              | bidirectional through PIO line for synchronous mode only |
| USART  | USART0_CTS | USART 0 Clear to send              | bi-03  |              | input through PIO line                                   |
| USART  | USART0_RTS | USART 0 Request to send            | bi-03  |              | output through PIO line                                  |
| USART  | USART1_RXD | USART 1 Data in                    | bi-03  |              | input through PIO line                                   |
| USART  | USART1_TXD | USART 1 Data out                   | bi-03  |              | bidirectional through PIO line                           |
| USART  | USART1_SCK | USART 1 Serial clock               | bi-03  |              | bidirectional through PIO line for synchronous mode only |
| USART  | USART1_CTS | USART 1 Clear to send              | bi-03  |              | input through PIO line                                   |
| USART  | USART1_RTS | USART 1 Request to send            | bi-03  |              | output through PIO line                                  |
| USART  | USART2_RXD | USART 2 Data in                    | bi-03  |              | input through PIO line                                   |
| USART  | USART2_TXD | USART 2 Data out                   | bi-03  |              | bidirectional through PIO line                           |
| USART  | USART2_SCK | USART 2 Serial clock               | bi-03  |              | bidirectional through PIO line for synchronous mode only |
| USART  | USART2_CTS | USART 2 Clear to send              | bi-03  |              | input through PIO line                                   |
| USART  | USART2_RTS | USART 2 Request to send            | bi-03  |              | output through PIO line                                  |
| Power  | VDDCORE    | Core power supply                  | Power  |              | 1.1V / 1.2V                                              |
| Power  | VDDIOP     | Peripherals I/O Lines Power Supply | Power  |              | 3.3V                                                     |
| Power  | VDDIOM     | EBI I/O Lines Power Supply         | Power  |              | 3.3V                                                     |
| Power  | VDDOSC32   | 32KHz Oscillator Power Supply      | Power  |              | 1.1V / 1.2V                                              |
| Power  | VDDOSCM    | Main Oscillator PLLB Power Supply  | Power  |              | 1.1V / 1.2V                                              |
| Power  | VDDPLLA    | PLLA power supply                  | Power  |              | 3.3V                                                     |
| Ground | GND        | Core and IO Ground                 | Ground |              |                                                          |
| Ground | GNDOSC32   | 32KHz Oscillator Ground            | Ground |              |                                                          |
| Ground | GNDOSCM    | Main Oscillator PLLB Ground        | Ground |              |                                                          |
| Ground | GNDPLLA    | PLLA Ground                        | Ground |              |                                                          |

## 4. Block Diagram

Figure 4-1. AT572D940HF Architecture



## **5. Architectural Overview**

DIOPSIS 940 HF (also named D940HF) is a high performance dual-core processing platform for audio, communication and beam-forming applications, integrating a floating-point DSP (mAgicV VLIW DSP) and an ARM926EJ-S Reduced Instruction Set Computer (RISC). The D940HF is optimally suited for floating point applications with a significant need for complex domain computations like FFT and frequency domain phase-shift algorithms, requiring high dynamic range and maximum numerical precision.

The D940HF combines the flexibility of the ARM926 RISC controller with the very high performance of the DSP oriented VLIW architecture of mAgicV.

### **5.1 System management**

The availability of a standard RISC on-chip lowers software development effort for non critical and control segments of the application. ARM926 features an MMU for virtual memory and sophisticated memory protection, making it an ideal platform for operating systems such as WinCE or Linux. This leaves mAgicV fully available for the numerically intensive part of the application. The synchronization between the two processors can be either based on interrupts or on software polling on semaphores.

The ARM926 is the D940HF master processor. The bootstrap sequence of the D940HF starts from the bootstrap of the ARM926 from its internal ROM or external non-volatile memory. The ARM then boots mAgicV from a non-volatile memory. After bootstrap the D940HF can start its normal operations. The DSP side of many applications can be implemented on the D940HF by using only the internal memory. In fact, the program memory size of 8K by 128-bit coupled with the availability of the general purpose code compression and software pipelining of systematic loops, gives an equivalent on-chip program memory size of about 24K cycles, corresponding to ~50K DSP assembler instructions (typical).

### **5.2 AMBA Architecture**

The architecture is based on AMBA™ bus: the multilayer AHB matrix and the APB.

The AHB matrix consists of seven masters:

0. ARM926 Instruction
1. ARM926-Data
2. Peripheral Data Controller (PDC)
3. mAgicV
4. USB Host
5. Ethernet MAC 10/100
6. mAgicV JTAG

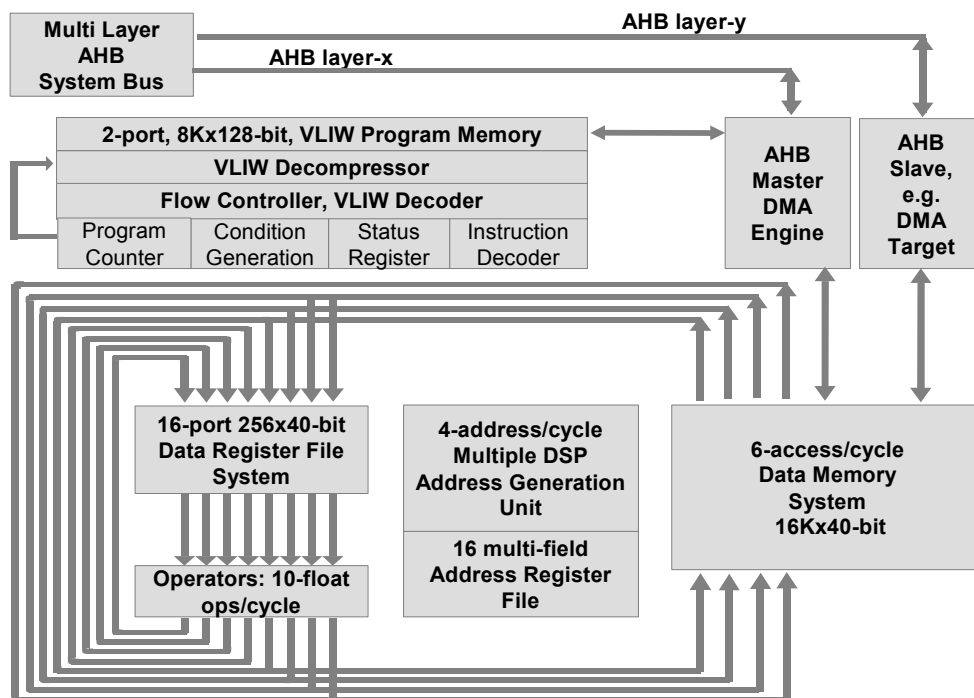
and of five slaves:

0. ARM926 SRAM
1. ARM926 ROM
2. mAgicV Registers and Memories + USB Host Registers
3. The External Bus Interface
4. The AHB-APB bridge

### 5.3 mAgicV VLIW DSP Processor

The mAgicV VLIW DSP is the numeric processor of the D940HF. It operates on IEEE 754 40-bit extended precision floating-point and 32-bit integer numeric format. The main components of the DSP subsystem are the core processor, the on-chip memories, the DMA engine and its AHB master and slave interfaces. The operators block, the register file, the multiple address generation unit and the program decoding and sequencing unit are the computing part of the core processor. A short description of each block is given in the following paragraphs.

**Figure 5-1.** mAgicV DSP Block Diagram



#### 5.3.1 RISC-like VLIW DSP

mAgicV is a Very Long Instruction Word engine, but from an user point of view, it works like a RISC machine by implementing triadic computing operations on data coming from the register file, and data move operations between the local memories and the register file. The operators are pipelined for maximum performance. The pipeline depth depends on the operator used. The scheduling and parallelism operations are automatically defined and managed at compile time by the assembler-optimizer, allowing efficient code execution. The architecture is designed for efficient C-language support.

#### 5.3.2 16-port, 256x40-bit Data Register File System

In order to provide optimal data bandwidth and to give the best support to the RISC-like programming model, mAgicV arithmetic computations are supported by a 16-ported, 256x40-bit entries, Data Register File System. The Data Register File can also be viewed as a complex 128-entry register file. It can be used as a complex register file (real + imaginary part), or as a dual register file for vectorial operations. When performing scalar instructions on the real domain, the register file can be used as an ordinary 256 register file. Both the odd and even sides of the register file are 9-ported (4-read ports and 4-write ports for computing/move operations + 1 port for independent debug access), making a total of 16 I/O ports available for the data

move to and from the operators block and the memory, plus the ports for the debug accesses. The total data bandwidth between the register file, the operators block and the data memory is 80 bytes per clock cycle, thus avoiding bottlenecks in the data flow inside the VLIW core.

The Operators block, the Data Register File, the Multiple Address Generation Unit and the Flow-Controller are the computing part of the core processor. The core is integrated with a 6-access/cycle, 16Kx40-bit on-chip Data Memory System and a 2-port, 8Kx128-bit on-chip VLIW Program Memory. The mAgicV VLIW DSP is equipped with an integrated AHB master and a DMA Engine plus an AHB Slave interface.

### 5.3.3 DSP Operators Block

The Operators Block contains the hardware that performs arithmetical operations. It works on 32-bit signed integers and IEEE 754 extended precision 40-bit floating-point data. The Operators Block is composed of four integer/floating point multipliers, an adder, a subtractor and two add-subtract integer/floating point units; moreover, it has two shift/logic units, a Min/Max operator and two seed generators for efficient division and inverse square root computation. The operators block is arranged in order to natively support complex arithmetic (single cycle complex multiply or multiply and add), fast FFT (single cycle butterfly computation) and vectorial computations (e.g. for Audio Stereo Channel support). The peak performance of mAgicV is achieved during single cycle FFT butterfly execution, when mAgicV delivers 10 floating-point operations per clock cycle.

### 5.3.4 6-port On-Chip Data Memory System

The Data Memory System of mAgicV contains 16K\*40-bit on-chip memory locations supporting up to 6 accesses/cycle. 4-accesses/cycle are reserved to the activities driven by the Multiple Address Generation unit of mAgicV: these accesses are reserved to the computing part of the core. 1 access/cycle is assigned to serve the DMA activity launched by the core itself, through mAgicV AHB master port. 1 additional access/cycle can be simultaneously requested by external devices through mAgicV AHB slave port (e.g. for data exchange with the interfaces of the ADC and the DAC converters). The Data Memory System is physically organized using two banks (assigned to even and odd addresses) of quadruple-port memories. The total bandwidth available is 28 bytes/cycle; for the computing part of the core it is 20 bytes per clock cycle, allowing full speed implementation of numerically intensive algorithms (e.g. complex FFT and FIR), plus 8 bytes/cycle assigned to the AHB master and slave interfaces.

### 5.3.5 Multiple DSP Address Generation Unit (MAGU)

The core can access vectorial and single data stored in the Data Memory. Accessing complex data is equivalent to accessing vectorial data (a pair of consecutive even and odd addresses pointing to the pair of banks). In vectorial mode, the Multiple Address Generation Unit (MAGU) is able to generate up to 4 addresses/cycle: two pairs of vectorial addresses, one to access the Data Memory System for reading a consecutive pair of memory locations and one address for writing a consecutive pair of memory locations. The MAGU can also generate any combination of two scalar accesses to the Data Memory System (Read-Read, Read-Write, Write-Write of any pair of single location accesses), or the combination of one vectorial access and one scalar access. The MAGU supports linear addressing and DSP oriented features like stride access and circular buffers. The address generation unit is supported by 16 multi field addressing registers each one composed of 4 16-bit individually addressable registers, for a total of 64 signed 16-bit integer registers. Registers named A0-A15 are used for the storage of pointers, while registers M0-M15 are for the 16-bit integer modifiers. For circular buffers, S0-S15 store the Start Addresses of the buffers, and L0-L15 are initialized with the circular buffer lengths. The MAGU

can also be used to perform 16-bit signed integer arithmetic operations in parallel with the activities of the operators block (40-bit floating point and 32 signed integer operations). The MAGU also performs the loop control computations needed to verify if the end of a loop is reached.

### 5.3.6 Flow Controller

The Flow Controller is dedicated to program address generation, conditioning, predication and software pipelining of systematic loops. The Program Address Generation Unit is devoted to control the correct Program Counter generation according to the program flow. It generates addresses for linear code execution as well as for non-sequential program flow. The Condition Generation Unit combines the flags generated by the operators and by the MAGU to produce complex conditions flags used to control the program execution. The Program Address Generation Unit also allows to perform conditioned and unconditioned branch instructions, loops, call to subroutines and return from subroutines.

### 5.3.7 Dual-Port On-Chip Program Memory

The Program Memory stores the VLIW program to be executed by mAgicV. It is 8K words by 128-bit dual port memory. One port is driven by the Flow Controller to fetch the compressed VLIW word. The other port is accessed by the DMA engine, supported by the AHB master interface, or by the external devices through mAgicV AHB slave port.

### 5.3.8 5 predicated VLIW Issues

At every cycle, a typical mAgicV VLIW instruction activates 5 issues named AGU0, AGU1, ADD, MULT and FLOW. The first two issues are associated to the pair of independent Address Generation Units in the MAGU. The third issue drives the Arithmetic Add/Subtract section of the Operators Block, the fourth drives the Multiplier section, and the last issue drives the Flow Controller. Each issue is predicated by a specific predication field, for conditional execution without pipeline breaking penalties. Using different instruction formats, the VLIW word can also contain initialization requests for the DMA engine, single cycle loading of multiple immediate values and other service instructions.

### 5.3.9 Software pipelining

Software pipelining of systematic loops is optimally supported by a dedicated engine which activates the VLIW issues only during the appropriate loop iterations. This mechanism is designed to reach optimal program memory usage of the DSP library and completes the general purpose Code Compression scheme.

### 5.3.10 Program Compression

The mAgicV VLIW architecture is natively designed for optimal program density. Moreover, a program compression scheme allows an average additional program compression between 2 and 3. Therefore, more than 10 issues are stored for each 128 bit program memory locations. A high Program Memory density is achieved thanks to the combined effect of Program Compression and Software Pipelining. The DSP side of many applications can be implemented on the D940HF using only the internal memory. In fact, the 8K by 128-bit program memory size provides, with code compression, ~50K DSP assembler instructions stored on-chip (typical). For DSP libraries, the density is even greater where software pipelining is activated. If the on-chip program memory is not large enough to contain the full DSP application, a DMA must be launched to refill the dual-port Program Memory. Thanks to the program compression, the program memory refill does not stall the activities of the DSP core.

## 5.3.11 mAgicV AHB master interface

mAgicV VLIW DSP is equipped with an AHB master which supports mAgicV DMA engine.

## 5.3.12 AHB DMA on Data Memory System

At every cycle, one port of the on-chip Data Memory System is reserved to fetch/store the activity driven by the DMA Engine. The DMA to the external memories or to the other devices mapped on the AHB System Bus is supported by mAgicV AHB master interface. The DMA engine can generate stride access to the external memory. The DMA transfers to and from the on-chip Memory can be executed in parallel with the full speed core instructions execution with zero-overhead and without the intervention of the core processor, except for initiating it.

## 5.3.13 AHB DMA on Program Memory

The on-chip Program Memory of mAgicV is a dual port. One port is reserved to the instruction fetch and the other to the DMA engine. In parallel with the activities of the core, a DMA can be activated between the external memories and the other devices mapped on the AHB System Bus.

## 5.3.14 mAgicV AHB slave interface

External AHB masters, like ARM and JTAG can access the memories and the registers of mAgicV DSP through mAgicV AHB slave interface. In Debug mode (see [Section 5.3.15.3](#) below) all the internal resources are memory mapped, while in run mode or sleep mode access restrictions apply (see [Section 5.3.15.1](#) and [Section 5.3.15.2](#) below). At every cycle, one port of the Data Memory System is reserved to read/store accesses performed through the AHB slave interface. Example of usage: data sampled by AD Converters can be written inside the mAgicV Data Memory in parallel to the DMA (through the master port) and the VLIW operations.

## 5.3.15 Operating Modes of mAgicV

mAgicV VLIW DSP can operate in three operating modes: Run mode, Sleep mode and Debug mode. The access allowed to the different resources through the AHB slave port depends on the status mode:

### 5.3.15.1 Run Mode

In Run Mode, a mAgicV VLIW program is under execution. mAgicV can access external resources through its AHB master interface. Control and status registers are visible. One port of the Data Memory System is accessible through the AHB Slave port.

### 5.3.15.2 Sleep Mode

In Sleep Mode, the AHB Master and Slave port and the DMA engine are still active. However, only “non-destructive access paths” are guaranteed through the AHB slave interface. Control and Status registers are active. Data and Address Registers are frozen (readable but not writable).

### 5.3.15.3 Debug Mode

In Debug Mode, mAgicV suspends its execution (if any) and debug paths are allowed. Data and Program memories are readable. Data and Address registers are readable. Pipeline registers are frozen. Any external master, like JTAG or the ARM can access the internal resources of mAgicV DSP for debug purpose. The ability of the ARM to access internal mAgicV resources in Debug Mode can be used for initialization and also for debugging purposes. By accessing the Command Register, the ARM can change the operating status of the DSP (Run/System Mode),

initiate DMA transactions, force single or multiple step execution, or simply read the DSP operating status.

#### **5.3.16 User/ Privileged Interrupt Mode**

During Run mode, mAgicV can execute either in User mode or in Privileged Interrupt Mode.

#### **5.3.17 ARM<->mAgicV Interrupts**

In order to allow a tight coupling between the operations of mAgicV and the ARM at run time, they can exchange synchronization signals, based on interrupts.

## **5.4 ARM926 Processor**

The ARM926 is a member of ARM9™ family of general purpose microprocessors. The ARM926 is targeted at multi-tasking applications where full memory management, high performance and low power are important.

The ARM926 supports the 32-bit ARM and 16-bit THUMB instruction sets, enabling the user to trade off between high performance and high code density. The ARM926 includes features for efficient execution of Java byte codes.

The ARM926 supports the ARM debug architecture and includes logic to assist both the hardware and the software debug.

The ARM926 provides an integer core that supports the DSP instruction set extension.

The ARM926 supports virtual memory addressing through its standard ARM v4 and v5 memory management unit (MMU).

The ARM926 provides two independent AHB master interfaces for data and instruction.

The ARM926 provides two independent Tightly Coupled Memory (TCM) interfaces.

The ARM926 implements ARM architecture version 5TEJ with 5 stage pipeline.

The ARM926 embeds 16-Kbyte Data Cache and 16-Kbyte Instruction Cache.

### **5.4.1 ARM Memories**

The ARM926 memories consist of:

- 32Kbyte ROM selectable as boot memory
- 48Kbyte Fast SRAM
  - Single Cycle Access at full bus speed
  - Supports ARM926EJ-S TCM interface at full processor speed
  - D-TCM and I-TCM programmable size

### **5.4.2 Arm Boot**

The system always boots at address 0x0. The memory layout can be configured with two parameters to ensure a maximum number of possibilities for booting.

REMAP allows the user to lay out the first internal SRAM bank to 0x0 to ease development. This is done by software once the system has booted for each Master of the Bus Matrix. When REMAP = 1, BMS is ignored. Refer to the Bus Matrix Section for more details.

When REMAP = 0, BMS allows the user, at ones convenience, to lay out the ROM or an external memory to 0x0. This is done via hardware at reset.

Note that Memory blocks not affected by these parameters can always be seen at their specified base addresses. The complete memory map is presented in [Table 5-1](#) to [Table 5-4](#).

The Bus Matrix manages a boot memory that depends on the level on the BMS pin at reset. The internal memory area mapped between address 0x0 and 0x000F FFFF is reserved for this purpose.

If BMS is detected at 1, the boot memory is the embedded ROM.

If BMS is detected at 0, the boot memory is the memory connected on the Chip Select 0 of the External Bus Interface.

#### 5.4.2.1 *BMS = 1, Boot on Embedded ROM*

The system boots using the Boot Program from the embedded ROM following the steps listed below:

Checks the presence of an SD card with a boot.bin file in the main dir:

If the file is found:

- Downloads the code in internal SRAM at 0x300000
- Executes Remap command
- Runs SD Boot code

If the file is not found, downloads the code from the SPI DataFlash®:

- Downloads the code in internal SRAM at 0x300000
- Checks the presence of a valid code on the first six word
- Executes Remap command
- Runs DataFlash Boot code

In case no valid program is detected in the external SPI DataFlash:

- Activates a Boot uploader enabling small monitor functionalities (read/write/run) interface with the SAM-BA™ application
- Performs an automatic detection of the communication link:  
Serial communication on a DBGU (XModem protocol)  
USB Device Port (CDC Protocol)

#### 5.4.2.2 *BMS = 0, Boot on External Memory*

- Boot on slow clock (32,768 Hz)
- Boot with the default configuration for the Static Memory Controller, byte select mode, 32-bit data bus, Read/Write controlled by Chip Select, allows boot on 32-bit non-volatile memory.

The customer-programmed software must perform a complete configuration.

To speed up the boot sequence when booting at 32 kHz EBI CS0 (BMS=0), the user must take the following steps:

1. Program the PMC (main oscillator enable or bypass mode).
2. Program and start the PLL.
3. Reprogram the SMC setup, cycle, hold, mode timings registers for CS0 to adapt them to the new clock Peripheral Data Controller (PDC).
4. Switch the main clock to the new value.

### 5.5 Peripheral Data Controller (PDC)

The PDC acting as an AHB master controls the data transfer between on chip peripherals: USARTs, SPIs, SSCs, MCI, DBGU, TWIs and the on- and off-chip memories. This leaves both the processors free of the overhead related to this function.

### 5.6 USB Host

The USB host acting as an AHB master controls the data exchange between the two USB host channels (port A and port B) and the ARM Internal RAM or the external memories.

The USB Host Port features:

- Compliance with Open HCI Rev 1.0 specification
- Compliance with USB V2.0 Full-speed and Low-speed Specification
- Supports both Low-speed 1.5 Mbps and Full-speed 12 Mbps USB devices
- Root hub integrated with two downstream USB ports
- Two embedded USB transceivers

## 5.7 Ethernet MAC 10/100

The Ethernet MAC acting as an AHB master controls the data exchange between the ethernet channel and the ARM Internal RAM or the external memories.

The Ethernet MAC is the hardware implementation of the MAC sub-layer OSI reference model between the physical layer (PHY) and the logical link layer (LLC). It controls the data exchange between a host and a PHY layer according to Ethernet IEEE 802.3u data frame format. The Ethernet MAC contains the required logic and transmits and receives FIFOs for the DMA management. In addition, it is interfaced through MDIO/MDC pins for the PHY layer management. The Ethernet MAC can transfer data through the Reduced Media Independent Interface (RMII).

The aim of the interface reduction is to lower the pin count for a switch product that can be connected to multiple PHY interfaces. The characteristics specific to RMII mode are:

- Single clock at 50 MHz frequency
- Reduction of required control pins
- Reduction of data paths to di-bit (2-bit wide) by doubling clock frequency
- 10 Mbits/sec. and 100 Mbits/sec. data capability

## 5.8 mAgicV JTAG

The mAgicV-JTAG provides the JTAG interface to the mAgicV core. It converts JTAG commands coming from a JTAG probe into AHB cycles. Acting as an AHB master it can access all mAgicV memories and registers, thus allowing mAgicV debug software to control the core and its resources: to upload/download data and programs and to configure functional and debug registers.

## 5.9 External Bus Interface (EBI)

Each enabled AHB master can access the external memory resources through the EBI. The External Bus IF incorporates the Static Memory Controller (SMC) and Synchronous Dynamic RAM controller (SDRAMC).

The EBI features:

- Eight Chip Select Lines (four via PIO lines)
- 26-bit Address Bus (four msb via PIO lines)
- 32-bit Data Bus
- Multiple Access Modes supported
- Byte Write Lines
- Programmable Wait State Generation
- Programmable Data Float Time
- Slow clock mode supported

### 5.9.1 Static Memory Controller (SMC)

The SMC gives to the AHB enabled Hosts the capability to access to the following type of external memories: SRAM, Nor-Flash, EPROM, EEPROM.

The additional NAND LOGIC also provides the SMC with the capability to interface the Smart-Media removable non-volatile memory cards and the Nand FLASH memory chips.

The additional Compact Flash logic provides the SMC with the capability to interface the Compact Flash removable non-volatile memory cards.

### 5.9.2 Synchronous Dynamic RAM Controller (SDRAMC)

The SDRAMC provides the interface to an external 16-bit or 32-bit SDRAM device.

The page size supports ranges from 2048 to 8192 and the number of columns from 256 to 2048. It supports byte (8-bit), half-word (16-bit) and word (32-bit) accesses.

The SDRAMC supports a read or write burst length of one location. It does not support byte read/write bursts or half-word write bursts. It keeps track of the active row in each bank (avoiding precharge and active when, changing bank, the old row is accessed), thus maximizing SDRAM performance, e.g., the application may be placed in one bank and data in the other banks. So it is advisable to avoid accessing different rows in the same bank in order to optimize performance.

The maximum number of SDRAM locations that can be randomly accessed without penalty cycles (precharge, active) corresponds to the device row size x the number of banks. The SDRAMC can support row size up to 2048 locations and 4 banks: hence maximum 8K locations can be accessed without penalties. Anyway, typical SDRAM row size are 512/256 locations so maximum 2K/1K locations can be accessed without penalties.

## 5.10 Memory Mapping

The present section describes the memory mapping of ARM9System.

[Table 5-1](#) shows the D940HF global memory map:

**Table 5-1.** D940HF Global Memory Map

| Start Address | Size (MB) | masters                                            |                                                       |             |                |             |             |                |
|---------------|-----------|----------------------------------------------------|-------------------------------------------------------|-------------|----------------|-------------|-------------|----------------|
|               |           | ARM9-I mst # 0                                     | ARM9-D mst #1                                         | PDC mst # 2 | magicV mst # 3 | USB mst # 4 | ETH mst # 5 | m-JTAG mst # 6 |
| 0x0000 0000   | 256       | Internal Memories (See <a href="#">Table 5-3</a> ) |                                                       |             |                |             |             |                |
| 0x1000 0000   | 8 x 256   | External Memories (See <a href="#">Table 5-2</a> ) |                                                       |             |                |             |             |                |
| 0x9000 0000   | 6 x 256   | Undefined (Abort)                                  |                                                       |             |                |             |             |                |
| 0xF000 0000   | 256       |                                                    | Internal Peripherals (See <a href="#">Table 5-4</a> ) |             |                |             |             |                |

Table 5-2 shows the external memory mapping:

**Table 5-2.** External Memory Map

| Start Address | Size (MB) | masters                                 |              |            |               |            |            |               |
|---------------|-----------|-----------------------------------------|--------------|------------|---------------|------------|------------|---------------|
|               |           | ARM9-I mst #0                           | ARM-D mst #1 | PDC mst #2 | magicV mst #3 | USB mst #4 | ETH mst #5 | m-JTAG mst #6 |
| 0x1000 0000   | 256       | EBI CS0:                                |              |            |               |            |            |               |
| 0x2000 0000   | 256       | EBI CS1: SMC or SDRAMC                  |              |            |               |            |            |               |
| 0x3000 0000   | 256       | EBI CS2: SMC                            |              |            |               |            |            |               |
| 0x4000 0000   | 256       | EBI CS3: SMC (SmartMedia or NAND-Flash) |              |            |               |            |            |               |
| 0x5000 0000   | 256       | EBI CS4: SMC (Compact Flash slot 0)     |              |            |               |            |            |               |
| 0x6000 0000   | 256       | EBI CS5: SMC (Compact Flash slot 1)     |              |            |               |            |            |               |
| 0x7000 0000   | 256       | EBI CS6: SMC                            |              |            |               |            |            |               |
| 0x8000 0000   | 256       | EBI CS7: SMC                            |              |            |               |            |            |               |

Table 5-3 shows the internal memory map:

**Table 5-3.** Internal Memory Map

| Start Address | Size (MB) | masters        |          |          |                |          |          |             |                |             |             |                |
|---------------|-----------|----------------|----------|----------|----------------|----------|----------|-------------|----------------|-------------|-------------|----------------|
|               |           | ARM9-I mst # 0 |          |          | ARM9-D mst # 1 |          |          | PDC mst # 2 | magic V mst# 3 | USB mst # 4 | ETH mst # 5 | m-JTAG mst # 6 |
|               |           | REMAP=0        |          | REMAP=1  | REMAP=0        |          | REMAP=1  |             |                |             |             |                |
|               |           | BMS=1          | BMS=0    |          | BMS=1          | BMS=0    |          |             |                |             |             |                |
| 0x0000 0000   | 1         | IntROM         | EBI NCS0 | IntRAM C | IntROM         | EBI NCS0 | IntRAM C |             |                |             |             |                |
| 0x0010 0000   | 1         |                |          |          | I-TCM          |          |          |             |                |             |             |                |
| 0x0020 0000   | 1         |                |          |          | D-TCM          |          |          |             |                |             |             |                |
| 0x0030 0000   | 1         | ARM AHB MEM    |          |          |                |          |          |             |                |             |             |                |
| 0x0040 0000   | 1         | IntROM         |          |          |                |          |          |             |                |             |             |                |
| 0x0050 0000   | 1         |                |          |          | USB cfg        |          |          |             |                |             |             |                |
| 0x0060 0000   | 1         |                |          |          | magicV         |          |          | magic V     |                |             |             | magic V        |

**Table 5-4.** Internal Peripherals Map

| Start Address | Size (byte) | masters |        |            |        |     |     |        |
|---------------|-------------|---------|--------|------------|--------|-----|-----|--------|
|               |             | ARM9-I  | ARM9-D | PDC        | magicV | USB | ETH | m-JTAG |
| 0xF000 0000   | 40 x 16k    |         |        | reserved   |        |     |     |        |
| 0xFFFA 0000   | 16k         |         |        | TC 0, 1, 2 |        |     |     |        |
| 0xFFFA 4000   | 16k         |         |        | USB DEV    |        |     |     |        |
| 0xFFFA 8000   | 16k         |         |        | MCI        |        |     |     |        |
| 0xFFFA C000   | 16k         |         |        | TWI-0      |        |     |     |        |
| 0xFFFB 0000   | 16k         |         |        | USART-0    |        |     |     |        |
| 0xFFFB 4000   | 16k         |         |        | USART-1    |        |     |     |        |
| 0xFFFB 8000   | 16k         |         |        | USART-2    |        |     |     |        |
| 0xFFFB C000   | 16k         |         |        | SSC-0      |        |     |     |        |
| 0xFFFC 0000   | 16k         |         |        | SSC-1      |        |     |     |        |
| 0xFFFC 4000   | 16k         |         |        | SSC-2      |        |     |     |        |
| 0xFFFC 8000   | 16k         |         |        | SPI-0      |        |     |     |        |
| 0xFFFC C000   | 16k         |         |        | SPI-1      |        |     |     |        |
| 0xFFFD 0000   | 16k         |         |        | SSC-3      |        |     |     |        |
| 0xFFFD 4000   | 16k         |         |        | TWI-1      |        |     |     |        |
| 0xFFFD 8000   | 16k         |         |        | ETH CFG    |        |     |     |        |
| 0xFFFD C000   | 16k         |         |        | CAN-0      |        |     |     |        |
| 0xFFFE 0000   | 16k         |         |        | CAN-1      |        |     |     |        |
| 0xFFFE 4000   | 3 x 16k     |         |        | reserved   |        |     |     |        |
| 0xFFFF 0000   | 117 x 512   |         |        | reserved   |        |     |     |        |
| 0xFFFF EA00   | 512         |         |        | SDRAMC     |        |     |     |        |
| 0xFFFF EC00   | 512         |         |        | SMC        |        |     |     |        |
| 0xFFFF EE00   | 512         |         |        | HMATRIX    |        |     |     |        |
| 0xFFFF F000   | 512         |         |        | AIC        |        |     |     |        |
| 0xFFFF F200   | 512         |         |        | DBGU       |        |     |     |        |
| 0xFFFF F400   | 512         |         |        | PIO A      |        |     |     |        |
| 0xFFFF F600   | 512         |         |        | PIO B      |        |     |     |        |
| 0xFFFF F800   | 512         |         |        | PIO C      |        |     |     |        |
| 0xFFFF FA00   | 512         |         |        | reserved   |        |     |     |        |
| 0xFFFF FC00   | 256         |         |        | PMC        |        |     |     |        |
| 0xFFFF FD00   | 256         |         |        | SYSC       |        |     |     |        |
| 0xFFFF FE00   | 2 x 256     |         |        | reserved   |        |     |     |        |

## 5.11 APB peripherals

The D940HF provides a rich set of peripherals connected on the APB bus. All enabled AHB masters can access these peripherals through the AHB-APB bridge.

### 5.11.1 Peripheral ID

Table 5-5 defines the Peripheral Identifiers of the D940HF. A peripheral identifier is required for the control of the peripheral interrupt with the Advanced Interrupt Controller and for the control of the peripheral clock with the Power Management Controller.

**Table 5-5.** Peripheral ID

| Peripheral ID | Peripheral Clock Assignment | Host Clock Assignment |
|---------------|-----------------------------|-----------------------|
| 0             |                             |                       |
| 1             |                             |                       |
| 2             | PIO A                       |                       |
| 3             | PIO B                       |                       |
| 4             | PIO C                       |                       |
| 5             | ETH APB                     | ETH AHB               |
| 6             | USART-0                     |                       |
| 7             | USART-1                     |                       |
| 8             | USART-2                     |                       |
| 9             | MCI                         |                       |
| 10            | USB Device                  |                       |
| 11            | TWI-0                       |                       |
| 12            | SPI-0                       |                       |
| 13            | SPI-1                       |                       |
| 14            | SSC-0                       |                       |
| 15            | SSC-1                       |                       |
| 16            | SSC-2                       |                       |
| 17            | TIMER-0                     |                       |
| 18            | TIMER-1                     |                       |
| 19            | TIMER-2                     |                       |
| 20            |                             | USB HOST              |
| 21            | SSC-3                       |                       |
| 22            | TW1                         |                       |
| 23            | CAN-0                       |                       |
| 24            | CAN-1                       |                       |
| 25            |                             |                       |
| 26            |                             | MAGIC Core            |
| 27            |                             |                       |
| 28            |                             |                       |

**Table 5-5.** Peripheral ID (Continued)

| Peripheral ID | Peripheral Clock Assignment | Host Clock Assignment |
|---------------|-----------------------------|-----------------------|
| 29            |                             |                       |
| 30            |                             |                       |
| 31            |                             |                       |

### 5.11.2 Peripheral Multiplexing

The D940HF features three PIO controllers, PIOA, PIOB and PIOC, that multiplex the I/O lines of the peripheral set. Each PIO controller manages up to thirty-two lines. Each line can be assigned to one of the two peripheral functions, A or B. [Table 5-6](#) to [Table 5-8](#) define how the I/O lines of the peripherals A and B are multiplexed on the PIO Controllers. Note that some output only peripheral functions might be duplicated within the tables and are indicated with the suffix II and III.

**Table 5-6.** PIO A Line Resource Mapping

| PIO A       | Periph INPUT A                          | Periph OUTPUT A     | Periph INPUT B                          | Periph OUTPUT B               |
|-------------|-----------------------------------------|---------------------|-----------------------------------------|-------------------------------|
| PIO A [0]   | SPI 0 bidir: MISO                       |                     |                                         | mAgicV output: M_SIRQ0        |
| PIO A [1]   | SPI 0 bidir: MOSI                       |                     |                                         | EBI: output: CFCE1 (III)      |
| PIO A [2]   | SPI 0 bidir: CLK                        |                     |                                         | EBI: output: CFCE2 (III)      |
| PIO A [3]   | SPI 0 bidir: CS0                        |                     |                                         | CAN 1: dout (III)             |
| PIO A [4]   |                                         | SPI 0 output: CS1   |                                         | mAgicV output: M_SIRQ2        |
| PIO A [5]   |                                         | SPI 0 output: CS2   | TIMER bidir: TIMER_OUT A0               |                               |
| PIO A [6]   |                                         | SPI 0 output: CS3   | TIMER bidir: TIMER_OUT B1               |                               |
| PIO A [7]   | USART 0 input: RXD                      |                     |                                         | DBGU output: DTXD(III)        |
| PIO A [8]   | USART 0 bidir: TXD                      |                     |                                         | PMC output: CKOUT 1           |
| PIO A [9]   | USART 0 input: CTS                      |                     |                                         | SPI 0 output: CS1 (III)       |
| PIO A [10]  |                                         | USART 0 output: RTS | TIMER input: TIMER_IN 1                 |                               |
| PIO A [11]  | USART 0 bidir: SCK                      |                     |                                         | SPI 0 output: CS2 (III)       |
| PIO A [12]  | AIC input: EXT_IRQ1<br>(also to mAgicV) |                     |                                         | USART 0 output: RTS (III)     |
| PIO A [13]  | ETH bidir MDIO                          |                     |                                         | mAgicV output: M_SIRQ1        |
| PIO A [14]  |                                         | ETH output MDC      | AIC input: EXT_IRQ2<br>(also to mAgicV) |                               |
| PIO A [15]  |                                         | ETH output: FCE100  | TIMER input: TIMER_IN 2                 |                               |
| PIO A [016] | ETH input: EREFCK                       |                     |                                         | PMC output: CKOUT 0           |
| PIO A [17]  | ETH input: ECRSDV                       |                     |                                         | EBI: output: NCS4/CFCS0 (III) |
| PIO A [18]  | ETH input: ERX0                         |                     |                                         | EBI: output: NCS5/CFCS1 (III) |
| PIO A [19]  | ETH input: ERX1                         |                     |                                         | EBI: output: NCS6 (III)       |
| PIO A [20]  | ETH input: ERXER                        |                     |                                         | EBI: output: NCS7 (III)       |
| PIO A [21]  |                                         | ETH output: ETX0    |                                         | TEST output: m_ck             |
| PIO A [22]  |                                         | ETH output: ETX1    |                                         | TEST output: a_ck             |

**Table 5-6.** PIO A Line Resource Mapping (Continued)

| PIO A      | Periph INPUT A   | Periph OUTPUT A           | Periph INPUT B            | Periph OUTPUT B              |
|------------|------------------|---------------------------|---------------------------|------------------------------|
| PIO A [23] |                  | ETH output: ETXEN         |                           | mAgicV output: M_SIRQ0 (III) |
| PIO A [24] | EBI input: BMS   |                           |                           | mAgicV output: M_SIRQ1 (III) |
| PIO A [25] | EBI input: NWAIT |                           |                           | USART 2 output: RTS (III)    |
| PIO A [26] |                  | EBI output:<br>NCS4/CFCS0 | TIMER bidir: TIMER_OUT A2 |                              |
| PIO A [27] |                  | EBI output:<br>NCS5/CFCS1 |                           | PMC output: CKOUT 2          |
| PIO A [28] |                  | EBI output: NCS6          |                           | EBI output: SMOE             |
| PIO A [29] |                  | EBI output: NCS7          |                           | EBI output: SMWE             |
| PIO A [30] |                  | EBI output: CFCE1         |                           | PMC output: CKOUT 3          |
| PIO A [31] |                  | EBI output: CFCE2         |                           | mAgicV output: M_SIRQ3       |

**Table 5-7.** PIO B Line Resource Mapping

| PIO B      | Periph INPUT A | Periph OUTPUT A | Periph INPUT B            | Periph OUTPUT B             |
|------------|----------------|-----------------|---------------------------|-----------------------------|
| PIO B [0]  | SSC: RD0       |                 |                           | SPI 0 output: CS3 (III)     |
| PIO B [1]  |                | SSC: TD0        | TIMER bidir: TIMER_OUT B0 |                             |
| PIO B [2]  | SSC: TF0       |                 |                           | PMC CKOUT 0 (II)            |
| PIO B [3]  | SSC: TK0       |                 |                           | CAN 0: dout (II)            |
| PIO B [4]  | SSC: RF0       |                 |                           | USART 0 RTS (II)            |
| PIO B [5]  | SSC: RK0       |                 |                           | mAgicV output: M_SIRQ1 (II) |
| PIO B [6]  | SSC: RD1       |                 |                           | CAN 0: dout (III)           |
| PIO B [7]  |                | SSC: TD1        | TIMER bidir: TIMER_OUT A1 |                             |
| PIO B [8]  | SSC: TF1       |                 |                           | PMC CKOUT 1 (II)            |
| PIO B [9]  | SSC: TK1       |                 |                           | SPI 1 output: CS1 (III)     |
| PIO B [10] | SSC: RF1       |                 |                           | USART 1 RTS (III)           |
| PIO B [11] | SSC: RK1       |                 |                           | EBI: A[22] (III)            |
| PIO B [12] | SSC: RD2       |                 |                           | EBI: A[23] (III)            |
| PIO B [13] |                | SSC: TD2        |                           | mAgicV output: M_SIRQ2 (II) |
| PIO B [14] | SSC: TF2       |                 |                           | EBI: A[24] (III)            |
| PIO B [15] | SSC: TK2       |                 |                           | SPI 0 output: CS3 (II)      |
| PIO B [16] | SSC: RF2       |                 |                           | ETH output: MDC (II)        |
| PIO B [17] | SSC: RK2       |                 |                           | ETH output: FCE100 (II)     |
| PIO B [18] | SSC: RD3       |                 |                           | EBI: A[25]-CFRNW (III)      |
| PIO B [19] |                | SSC: TD3        |                           | mAgicV output: M_SIRQ0 (II) |
| PIO B [20] | SSC: TF3       |                 |                           | ETH output: MDC (III)       |
| PIO B [21] | SSC: TK3       |                 |                           | ETH output: FCE100 (III)    |

**Table 5-7.** PIO B Line Resource Mapping (Continued)

| PIO B      | Periph INPUT A                          | Periph OUTPUT A  | Periph INPUT B | Periph OUTPUT B             |
|------------|-----------------------------------------|------------------|----------------|-----------------------------|
| PIO B [22] | SSC: RF3                                |                  |                | USART 1 RTS (II)            |
| PIO B [23] | SSC: RK3                                |                  |                | DBGU output: DTXD (II)      |
| PIO B [24] | TIMER input: TIMER_IN 0                 |                  |                | mAgicV output: M_MODE       |
| PIO B [25] | AIC input: EXT_IRQ0<br>(also to mAgicV) |                  |                | USART 2 RTS (II)            |
| PIO B [26] | CAN 0: din                              |                  |                | SPI 1 output: CS2 (III)     |
| PIO B [27] |                                         | CAN 0: dout      |                | mAgicV output: M_SIRQ3 (II) |
| PIO B [28] |                                         | EBI: A[22]       |                | SPI 0 output: CS1 (II)      |
| PIO B [29] |                                         | EBI: A[23]       |                | SPI 0 output: CS2 (II)      |
| PIO B [30] |                                         | EBI: A[24]       |                | PMC CKOUT 2 (II)            |
| PIO B [31] |                                         | EBI: A[25]-CFRNW |                | PMC CKOUT 3(II)             |

**Table 5-8.** PIO C Line Resource mapping

| PIO C      | Periph INPUT A             | Periph OUTPUT A   | Periph INPUT B | Periph OUTPUT B              |
|------------|----------------------------|-------------------|----------------|------------------------------|
| PIO C [0]  | SPI 1 bi-directional: MISO |                   |                | SSC: TD0 (II)                |
| PIO C [1]  | SPI 1 bi-directional: MOSI |                   |                | SSC: TD1 (II)                |
| PIO C [2]  | SPI 1 bi-directional: CLK  |                   |                | SSC: TD2 (II)                |
| PIO C [3]  | SPI 1 bi-directional: CS0  |                   |                | ETH output: ETX0 (II)        |
| PIO C [4]  |                            | SPI 1 output: CS1 |                | ETH output: ETX1 (II)        |
| PIO C [5]  |                            | SPI 1 output: CS2 |                | mAgicV output: M_SIRQ3 (III) |
| PIO C [6]  |                            | SPI 1 output: CS3 |                | EBI: output: SMOE (III)      |
| PIO C [7]  | TWI 0 bi-directional: TWD  |                   |                | SSC: TD0 (III)               |
| PIO C [8]  | TWI 0 bi-directional: TWCK |                   |                | SSC: TD1 (III)               |
| PIO C [9]  | USART 1 RXD                |                   |                | SSC: TD2 (III)               |
| PIO C [10] | USART 1 TXD                |                   |                | ETH output: ETX0 (III)       |
| PIO C [11] | USART 1 CTS                |                   |                | ETH output: ETX1 (III)       |
| PIO C [12] |                            | USART 1 RTS       |                | SPI 1 output: CS1 (II)       |
| PIO C [13] | USART 1 SCK                |                   |                | SSC: TD3 (II)                |
| PIO C [14] | USART 2 RXD                |                   |                | EBI: A[22] (II)              |
| PIO C [15] | USART 2 TXD                |                   |                | EBI: A[23] (II)              |
| PIO C [16] | USART 2 CTS                |                   |                | EBI: A[24] (II)              |
| PIO C [17] |                            | USART 2 RTS       |                | EBI: A[25]-CFRNW (II)        |
| PIO C [18] | USART 2 SCK                |                   |                | SPI 1 output: CS2 (II)       |
| PIO C [19] | TIMER bidir: TIMER_OUT B2  |                   |                | SPI 1 output: CS3 (II)       |

**Table 5-8.** PIO C Line Resource mapping (Continued)

| PIO C      | Periph INPUT A             | Periph OUTPUT A   | Periph INPUT B | Periph OUTPUT B              |
|------------|----------------------------|-------------------|----------------|------------------------------|
| PIO C [20] | TWI 1 bi-directional: TWD  |                   |                | SSC: TD3 (III)               |
| PIO C [21] | TWI 1 bi-directional: TWCK |                   |                | SPI 1 output: CS3 (III)      |
| PIO C [22] | MCI bidir: MCCK            |                   |                | CAN 1: dout (II)             |
| PIO C [23] | MCI bidir: MCCDA           |                   |                | mAgicV output: M_SIRQ2 (III) |
| PIO C [24] | MCI bidir: MCDA0           |                   |                | EBI: SMOE (II)               |
| PIO C [25] | MCI bidir: MCDA1           |                   |                | EBI: SMWE (II)               |
| PIO C [26] | MCI bidir: MCDA2           |                   |                | EBI: NCS4/CFCS0 (II)         |
| PIO C [27] | MCI bidir: MCDA3           |                   |                | EBI: NCS5/CFCS1 (II)         |
| PIO C [28] | CAN 1: din                 |                   |                | EBI: NCS6 (II)               |
| PIO C [29] |                            | CAN 1: dout       |                | EBI: NCS7 (II)               |
| PIO C [30] | DBGU input: DRXD           |                   |                | EBI: CFCE1 (II)              |
| PIO C [31] |                            | DBGU output: DTXD |                | EBI: CFCE2 (II)              |

### 5.11.3 System Controller (SYSC)

The SYSC includes the Reset Controller (RSTC) and the System Timers (SYST).

The RSTC manages all system resets: external devices reset, processors reset and peripheral reset.

The sources of reset can be: Power-On, Watch Dog, SW reset, External reset.

The SYST features:

- One 16-bit Period Interval Timer
- One 12-bit key-protected Watchdog Timer
- One 20-bit Free-running Real-time Timer

### 5.11.4 Power Management Controller (PMC)

The PMC features two clock sources: Slow Clock Oscillator (32.768 Hz) and Main Oscillator (8 to 20 MHz).

Two dividers, A and B, and two Phase Lock Loops, A and B, allow a wide range of frequencies to be generated from either the slow clock and/or the main clock.

The PMC provides dedicated clocks toward: ARM926, the AHB Matrix, mAgicV, mAgicV Memories, the USB, the Ethernet MAC and all Peripherals.

### 5.11.5 Advanced Interrupt Controller (AIC)

The AIC features:

- Controls the interrupt lines (nIRQ and nFIQ) of ARM926
- Thirty-two individually maskable and vectored interrupt sources
- Programmable Edge-triggered or Level-sensitive Internal Sources
- Programmable Positive/Negative Edge-triggered or High/Low Level sensitive

- 8-level Priority Controller
- Fast Forcing: allows redirection of any normal interrupt source on the nFIQ

#### **5.11.6 Parallel Input/Output (PIO)**

The three PIOs provide globally 96 programmable I/O Lines.

These lines are fully programmable through Set/Clear Registers or linked to one of the two peripheral functions.

Each I/O Line (assigned to a peripheral or used as a general purpose I/O) provides:

- Input change interrupt
- Glitch filter
- Multi-drive option enables driving in open drain
- Programmable pull up on each I/O line
- Pin data status register, supplies visibility of the level on the pin at any time

#### **5.11.7 Universal Synchronous Bus Device (USB)**

The USB Device provides communication services between an external host and D940HF. The USB device is connected to the APB through a FIFO.

The USB Device features:

- USB V2.0 full-speed compliant, 12 Mbits per second
- Embedded USB V2.0 full-speed transceiver
- Embedded dual-port RAM for endpoints
- Suspend/Resume logic
- Embedded Transceivers

#### **5.11.8 Timer Counter (TC)**

The TC consists of three 16-bit Timer Counter Channels providing a wide range of functions including:

- Frequency Measurement
- Event Counting
- Interval Measurement
- Pulse Generation
- Delay Timing
- Pulse Width Modulation
- Up/down Capabilities

Each channel is user-configurable and contains:

- Three external clock inputs
- Five internal clock inputs
- Two multi-purpose input/output signals

#### **5.11.9 Two Wire Interface (TWI)**

The D940HF provides two independent TWIs.

Each TWI interconnects components on a unique two-wire bus, made of one clock line and one data line which speeds of up to 400 Kbits per second, based on a byte oriented transfer format.

Each TWI is programmable as a master with sequential or single-byte access.

A configurable baud rate generator allows the output data rate to be adapted to a wide range of core clock frequencies.

## 5.11.10 Universal Synchronous Asynchronous Rx Tx (USART)

The D940HF provides three independent USARTs.

Each USART features:

- Synchronous and Asynchronous mode
- Programmable Baud Rate Generator (up to 115.2 Kbps in Asynchronous Mode and system clock frequency in Synchronous Mode)
- RS485 with driver control signal
- ISO7816, T = 0 or T = 1 Protocols for interfacing with smart cards
- IrDA modulation and demodulation
- PDC connection

## 5.11.11 Serial Synchronous Controller (SSC)

The D940HF provides four independent SSCs.

Each SSC provides a programmable serial synchronous communication link to be used in audio and telecom applications (CODECs in Master or Slave Modes, I2S, TDM Buses, Magnetic Card Reader, SPI, ...).

The PDC connection allows a direct data transfer between the CODECs and mAgicV data memory, ARM internal memory or external memories.

## 5.11.12 Serial Peripheral Interface (SPI)

The D940HF provides two independent SPIs.

Each SPI supports the communication with serial external devices such as DataFlash, ADCs, DACs, LCD Controllers, CAN Controllers and Sensors.

Four chip selects with external decoder support allow communication with up to 15 peripherals.

The PDC connection allows a direct data transfer between these serial devices and mAgicV data memory, ARM internal memory or external memories.

## 5.11.13 Debug Unit (DBGU)

The DBGU is a 2-wire UART dedicated to Debug Communication.

The DBGU TX and RX channels are associated with two PDC channels.

The Debug Unit also generates the Debug Communication Channel (DCC) signals provided by the In-circuit Emulator of the ARM processor visible to the software. These signals indicate the status of the DCC read and write registers and generate an interrupt to the ARM processor, allowing the handling of the DCC under interrupt control.

#### **5.11.14 Controller Area Network (CAN)**

The D940HF provides two independent CANs.

Each CAN is fully compliant with the CAN 2.0 Part A and 2.0 Part B.

The CAN supports bit/rate up to 1 Mbps.

#### **5.11.15 Multimedia Card Interface (MCI)**

The D940HF provides a MCI.

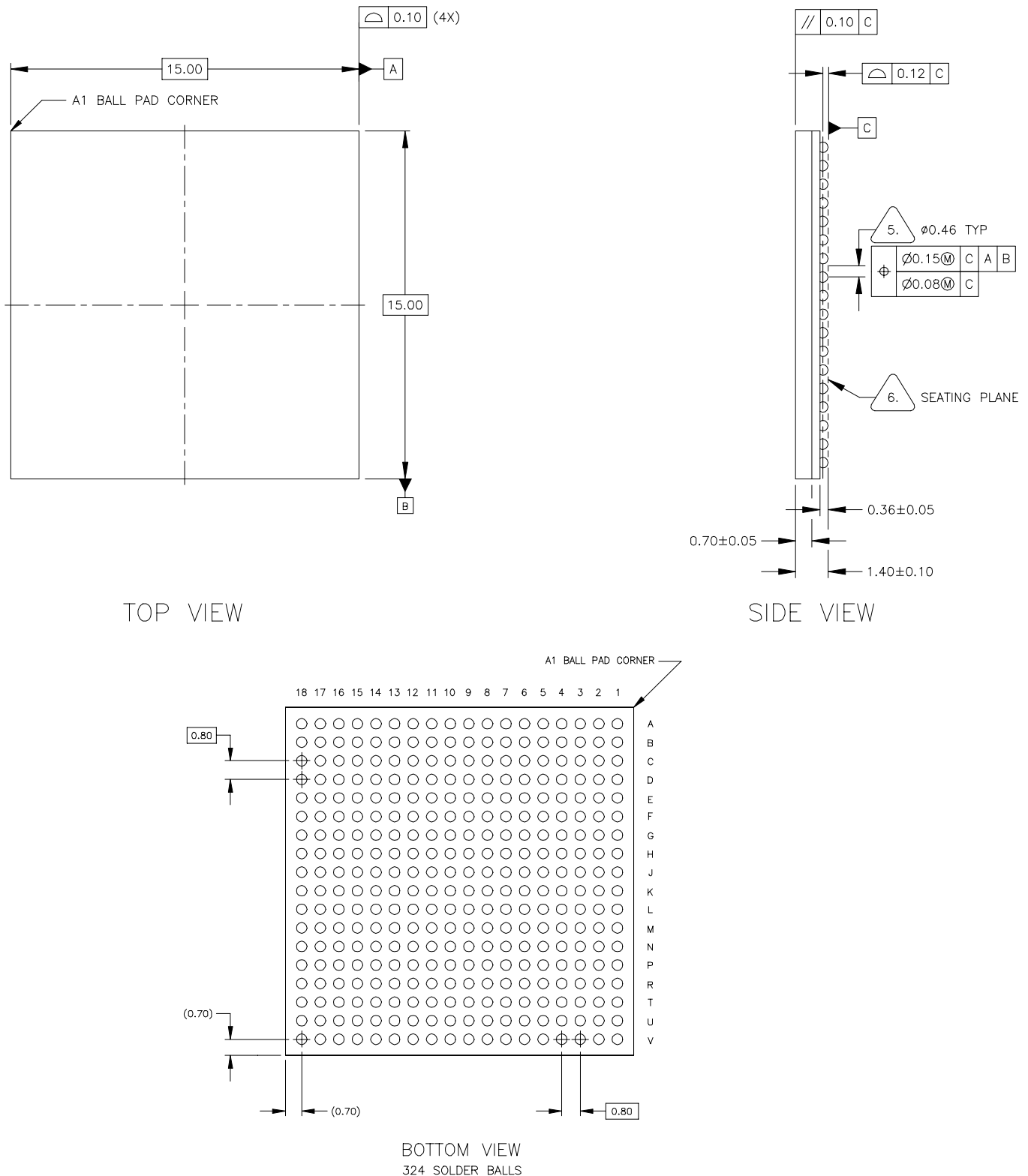
The MCI has two slots, each supporting:

- One slot for one MultiMedia Card bus (up to 30 cards) or
- One SD Memory Card

The PDC connection allows direct data transfer between these serial devices and mAgicV data memory, ARM internal memory or the external memories.

## 6. Mechanical Drawing

**Figure 6-1.** 324-ball CABGA Package Drawing (dimensions in mm)



## 7. Power Dissipation

The D940HF has six kinds of power supply pins:

- VDDCORE pins, which power the chip core (1.1V / 1.2V)
- VDDOSC32 pins, which power the 32KHz oscillator cell (1.1V / 1.2V)
- VDDOSCM pins, which power the main oscillator cell (1.1V / 1.2V)
- VDDIOM pins, which power the EBI I/O lines (3.3V)
- VDDIOP pins, which power the Peripheral I/O lines (3.3V)
- VDDPLLA pins, which power the PLLA cell (3.3V)

### 7.1 Power Consumption

The D940HF consumes about 2mA in typical conditions of static current VDDCORE.

For dynamic power consumption the D940HF consumes about 300mA in typical conditions at maximum working frequencies with a 20% toggling rate.

## 8. Ordering Guide

**Table 8-1.** Ordering Information

| Part Number    | Temp. Range   | Speed Grade (Max) | Operating Voltage                   | Package         | Notes                                 | Status                        |
|----------------|---------------|-------------------|-------------------------------------|-----------------|---------------------------------------|-------------------------------|
| AT572D940HF    | 0°C to 70°C   | 160 MHz           | 3.3V (I/O)<br>1.1V (core)           | CA324BGA (RoHS) | Full Peripheral Set                   | Sampling                      |
| AT572D940HF-CL | 0°C to 70°C   | 160 MHz           | 1.8V-2.5V-3.3V (I/O)<br>1.2V (core) | CA324BGA (RoHS) | Reduced Peripheral Set <sup>(1)</sup> | Contact:<br>diopsis@atmel.com |
| AT572D940HF-CJ | -40°C to 85°C | 200 MHz           | 1.8V-2.5V-3.3V (I/O)<br>1.2V (core) | CA324BGA (RoHS) | Full Peripheral Set                   | Contact:<br>diopsis@atmel.com |

1. Some peripherals are not accessible by the user in this low-cost version. Reduced Peripheral Set = Full Peripheral Set - 2 CANs - 3 SSCs - 1 SPI - 1 TWI - 2 USARTs. Consequently the related PIO lines can be used only as SW controlled PIO lines (not linked to any peripherals).

## 9. Revision History

| Doc. Rev. | Date  | Comments                   |
|-----------|-------|----------------------------|
| 7010AS    | 07/07 | • Initial document release |



## Headquarters

---

**Atmel Corporation**  
2325 Orchard Parkway  
San Jose, CA 95131  
USA  
Tel: 1(408) 441-0311  
Fax: 1(408) 487-2600

## International

---

**Atmel Asia**  
Room 1219  
Chinachem Golden Plaza  
77 Mody Road Tsimshatsui  
East Kowloon  
Hong Kong  
Tel: (852) 2721-9778  
Fax: (852) 2722-1369

**Atmel Europe**  
Le Krebs  
8, Rue Jean-Pierre Timbaud  
BP 309  
78054 Saint-Quentin-en-  
Yvelines Cedex  
France  
Tel: (33) 1-30-60-70-00  
Fax: (33) 1-30-60-71-11

**Atmel Japan**  
9F, Tonetsu Shinkawa Bldg.  
1-24-8 Shinkawa  
Chuo-ku, Tokyo 104-0033  
Japan  
Tel: (81) 3-3523-3551  
Fax: (81) 3-3523-7581

## Product Contact

---

**Web Site**  
[www.atmel.com](http://www.atmel.com)

**Technical Support**  
[diopsis@atmel.com](mailto:diopsis@atmel.com)

**Sales Contact**  
[www.atmel.com/contacts](http://www.atmel.com/contacts)

**Literature Requests**  
[www.atmel.com/literature](http://www.atmel.com/literature)

---

**Disclaimer:** The information in this document is provided in connection with Atmel products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Atmel products. **EXCEPT AS SET FORTH IN ATMEL'S TERMS AND CONDITIONS OF SALE LOCATED ON ATMEL'S WEB SITE, ATMEL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ATMEL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ATMEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.** Atmel makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Atmel does not make any commitment to update the information contained herein. Unless specifically provided otherwise, Atmel products are not suitable for, and shall not be used in, automotive applications. Atmel's products are not intended, authorized, or warranted for use as components in applications intended to support or sustain life.

© 2007 Atmel Corporation. All rights reserved. Atmel®, logo and combinations thereof, DIOPSIS®, DataFlash® and others are registered trademarks, Magic DSP™ and others are trademarks of Atmel Corporation or its subsidiaries. ARM®, Thumb® and others are the registered trademarks or trademarks of ARM Ltd. Other terms and product names may be trademarks of others.