SPEAR-09-H022

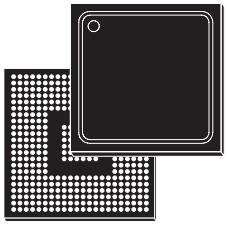
SPEAr™ Head200

ARM 926, 200K customizable eASIC™ gates, large IP portfolio SoC

PRELIMINARY DATA

Features

- ARM926EJ-S - f_{MAX} 266 MHz, 32 KI - 16 KD cache, 8 KI - KD TCM, ETM9 and JTAG interfaces
- 200K customizable equivalent ASIC gates (16K LUT equivalent) with 8 channels internal DMA high speed accelerator function and 112 dedicated general purpose I/Os
- Multilayer AMBA 2.0 compliant Bus with f_{MAX} 133 MHz
- Programmable internal clock generator with enhanced PLL function, specially optimized for E.M.I. reduction
- 16 KB single port SRAM embedded
- Dynamic RAM interface: 16 bit DDR, 32 / 16 bit SDRAM
- SPI interface connecting serial ROM and Flash devices
- 2 USB 2.0 Host independent ports with integrated PHYs
- USB 2.0 Device with integrated PHY
- Ethernet MAC 10/100 with MII management interface
- 3 independent UARTs up to 115 Kbps (Software Flow Control mode)
- I^2C Master mode - Fast and Slow speed
- 6 General Purpose I/Os



PBGA420

- ADC 8 bits, 230 Ksps, 16 analog input channels
- Real Time Clock
- WatchDog
- 4 General Purpose Timers
- Operating temperature: - 40 to 85 °C
- Package: PBGA 384+36 6R (23x23x1.81 mm)

Overview

SPEAr Head200 is a powerful digital engine belonging to SPEAr family, the innovative customizable System on Chips.

The device integrates an ARM core with a large set of proven IPs (Intellectual Properties) and a configurable logic block that allows very fast customization of unique and/or proprietary solutions, with low effort and low investment.

Optimized for embedded applications.

Order codes

Part number	Op. Temp. range, °C	Package	Packing
SPEAR-09-H022	-40 to 85	PBGA420 (23x23x1.81 mm)	Tray

Contents

List of tables

List of figures

1 Reference documentation

1. ARM926EJ-S - Technical Reference Manual
2. AMBA 2.0 Specification
3. EIA/JESD8-9 Specification
4. USB2.0 Specification
5. OCHI Specification
6. ECHI Specification
7. UTMI Specification
8. USB Specification
9. IEEE 802.3 Specification
10. I^2C - Bus Specification

2 Product Overview

SPEAr Head200 is a powerful System on Chip based on 110nm HCMOS and consists of 2 main parts: an ARM based architecture and an embedded customizable logic block.
The high performance ARM architecture frees the user from the task of developing a complete RISC system.
The customizable logic block allows user to design custom logic and special functions.
SPEAr Head200 is optimized for embedded applications and thanks to its high performance can be used for a wide range of different purposes.

Main blocks description:

1. CPU: ARM926EJ-S running at 266 MHz.
   It has:
   - MMU
   - 32 KB of instruction CACHE
   - 16 KB of data CACHE
   - 8 KB of instruction TCM (Tightly Coupled Memory)
   - 8 KB of data TCM
   - AMBA Bus interface
   - Coprocessor interface
   - JTAG
   - ETM9 (Embedded Trace Macro-cell) for debug; large size version.
2. Main Bus System: a complete AMBA Bus 2.0 subsystem connects different masters and slaves.
   The subsystem includes:
   - AHB Bus, for high performance devices
   - APB Bus, for low power / lower speed devices connectivity
   - Bus Matrix, for improving connection between the peripherals

   Bus System supports two masters: the ARM926EJ-S and the Customizable Logic block, connected to AHB Bus. All others blocks are slaves.
3. Clock and Reset System: fully programmable block with:
   - Separated set-up between clocks of AHB Bus and APB Bus peripherals
   - E.M.I. reduction mode, replacing all traditional drop methods for Electro-Magnetic Interference
   - Debug mode, compliant with ARM debug status
4. Interrupt Controller: the Interrupt Controller has 32 interrupt sources which are prioritized and vectorized.
5. On-chip memory: 4 independent static RAM cuts, 4 KB each, are available.
   They can be used on AHB Bus or directly by the custom logic.
6. Dynamic Memory Controller: it is a Multi-Port Memory Controller which is able to connect directly to memory sizes from 16 to 512 Mbits; the data size can be 8 or 16 bits for both DDR and SDRAM, also 32 bits for SDRAM. The external data bus can be maximum 32 bit wide at maximum clock frequency of 133 MHz and have up to 4 chip selects; the accessible memory is 256 MB.
   Internally it handles 7 ports supporting the following masters: AHB Bus, Bus Matrix, 2 USB 2.0 Hosts, USB 2.0 Device, Ethernet MAC, eASIC™ MacroCell.

The Multi-Port Memory Controller block has a programmable arbitration scheme and the transactions happen on a different layer from the main bus.

7. Serial Peripheral Interface: it allows a serial connection to ROM and Flash.
   The block is connected as a slave on the main AHB Bus, through the Bus Matrix.
   The default bus size is 32 bit wide and the accessible memory is 64 MB at a maximum speed of 50 MHz
8. USB 2.0 Hosts: these peripherals are compatible with USB 2.0 High-Speed specification. They can work simultaneously either in Full-Speed or in High-Speed mode.
   The peripherals have dedicated channels to the Multi-Port Memory Controller and 4 slave ports for CPU programming.
   The PHYs are embedded.
9. USB 2.0 Device: the peripheral is compatible with USB 2.0 High-Speed specifications.
   A dedicated channel connects the peripheral with the Multi-Port Memory Controller and registers.
   An USB-Plug Detector block is also available to verify the presence of the VBUS voltage.
   The port is provided with the following endpoints on the top of the endpoint 0:
   – 3 bulkin / bulkout endpoints
   – 2 isochronous endpoints

   The PHY is integrated.
10. Ethernet Media Access Control (MAC) 10/100: this peripheral is compatible with IEEE 802.3 standard and supports the MII management interface for the direct configuration of the external PHY.
    It is connected to the Multi-Port Memory Controller through a dedicated channel.
    The Ethernet controller and the configuration registers are accessible from the main AHB Bus.
11. ADC: 8 bit resolution, 230 Ksps (Kilo-sample per second), with 16 analog input channels. Connected to APB bus.
12. UART's: 3 independent interfaces, up to 115 Kbps each, support Software Flow Control.
    Connected to APB bus.
13. I²C supports Master mode protocol in Low and Full speed.
    Connected to APB bus.
14. 6 General Purpose I/O signals are available for user configuration.
    Connected to APB bus.
15. Embedded features: Real Time Clock, Watchdog, 4 General Purpose Timers.
    All blocks are interfaced with APB Bus.
16. Customizable Logic: it consists of an embedded macro where it is possible to map up to 200K equivalent ASIC gates. The same logic can be alternatively used to implement 32 KBytes of SRAM. Logic gates and RAM bits can be mixed in the same configuration so that processing elements, tightly coupled with embedded memories, can be easily implemented.
    The MacroCell has 2 dedicated buses, each of them connected with a 4 channel DMA in order to speed up the data flow with the main memories.
    8 interrupt lines and 112 dedicated general purpose I/Os are available.
    To allow a simple development of project, customizable logic can be emulated by an external FPGA, where customer can map his logic; FPGA is easy linkable and keeps the access to all on-chip and I/Os interfaces of the macro.

3 Features

3.1 CPU

It is an ARM926EJ-S RISC Processor

- ARM926EJ-S RISC Processor
- f_{MAX} 266 MHz (downward scalable)
- Virtual address support with MMU
- 32 KB instruction CACHE (4 way set associative)
- 16 KB data CACHE (4 way set associative)
- 8 KB instruction TCM
- 8 KB data TCM
- Coprocessor interface
- JTAG
- ETM9 (rev 2.2), large size FIFO

3.2 Internal bus structures

- Multilayer structure AMBA 2.0 compliant
- f_{MAX} 133 MHz
- High speed I/Os with embedded DMA function

3.3 Clock system

- Programmable clock generator
- PLL with E.M.I. reduction
- Low Jitter PLL for USB 2.0

3.4 Interrupt controller

- IRQ and FIQ interrupt generations
- Support up to 32 standard interrupts
- Support up to 16 vectored interrupts
- Software interrupt generation

3.5 Memory system

3.5.1 Memory on chip

16 KBytes single-port SRAM embedded in the eASIC™ MacroCell.

It can be used on AHB Bus or directly by the custom logic.

3.5.2 SPI

- 4 chip selects for asynchronous devices (ROM, Flash)
- Supports Normal mode 20 MHz and Fast mode 50 MHz
- AHB slave
- Accessible memory: 64 MB
- 8 / 16 / 32 bit widths
- Programmable wait states

3.5.3 Multi-port memory controller

- Maximum clock frequency 133 MHz
- Support up to 7 AHB master requests
- AHB slave
- Support for 8, 16 and 32 bit wide SDRAM
- Support for 8 and 16 bit wide DDR
- 4 chip selects
- Physical addressable memory up to 256 MB
- Memory clock tuning to match the timing of different memory vendors

3.6 High speed connectivity

3.6.1 USB 2.0 host

- 2 USB 2.0 Host controllers with their UTMI PHY port embedded
- High-Speed / Full-Speed / Low-Speed modes USB 2.0 complaint
- DMA FIFO
- 2 AHB slaves for configuration and FIFO access
- 2 AHB masters for data transfer

3.6.2 USB 2.0 device

- UDC 2.0 controller with embedded PHY
- High-Speed / Full-Speed / Low-Speed modes USB 2.0 complaint
- USB Self-Power mode
- DMA FIFO
- Master interface for DMA transfer to DRAM memory
- AHB slaves for: configuration, Plug autodetect
- Endpoints on the top of endpoint 0: 3 bulkin / bulkout, 2 isochronous

3.6.3 Ethernet 10/100

- MAC110 controller compliant with IEEE 802.3 standard
- Supporting MII 10/100 Mbits/s
- MII management protocol interface
- TX FIFO (512x36 Dual Port)
- RX FIFO (512x36 Dual Port)
- Master interface for DMA transfer to DRAM memory
- AHB slave for configuration

3.7 Low speed connectivity

3.7.1 UART

- Support for 8 bit serial data TX and RX
- Selectable 2 / 1 Stop bits
- Selectable Even, Odd and No Parity
- Parity, Overrun and Framing Error detector
- Max transfer rate: 115 Kbps

3.7.2 I^2C

- Standard I^2C mode (100 KHz) / Fast I^2C mode (400 KHz)
- Master interface only
- Master functions control all I^2C bus specific sequencing, protocol, arbitration and timing
- Detection of bus errors during transfers

3.8 General purpose I/Os

6 programmable GP I/Os

3.9 Analog to Digital Converter

- 8 bit resolutions
- 230 Ksps
- 16 analog input channels (0 - 3.3 V)
- INL ± 1 LSB
- DNL ± 0.5 LSB
- Programmable conversion speed - minimum conversion time 4.3 µs

3.10 Real time clock

- Real time clock-calendar (RTC)
- 14 digit (YYYY MM DD hh mm ss) precision
- Clocked by 32.768 KHz low power clock input
- Separated power supply (1.2 V)

3.11 Watchdog timer

- Programmable 16 bit Watchdog timer with reset output signal (more than 200 system clock period to initial peripheral devices)
- Programmable period 1 ~ 10 sec.
- For recovery from unexpected system Hang-up

3.12 General purpose timers

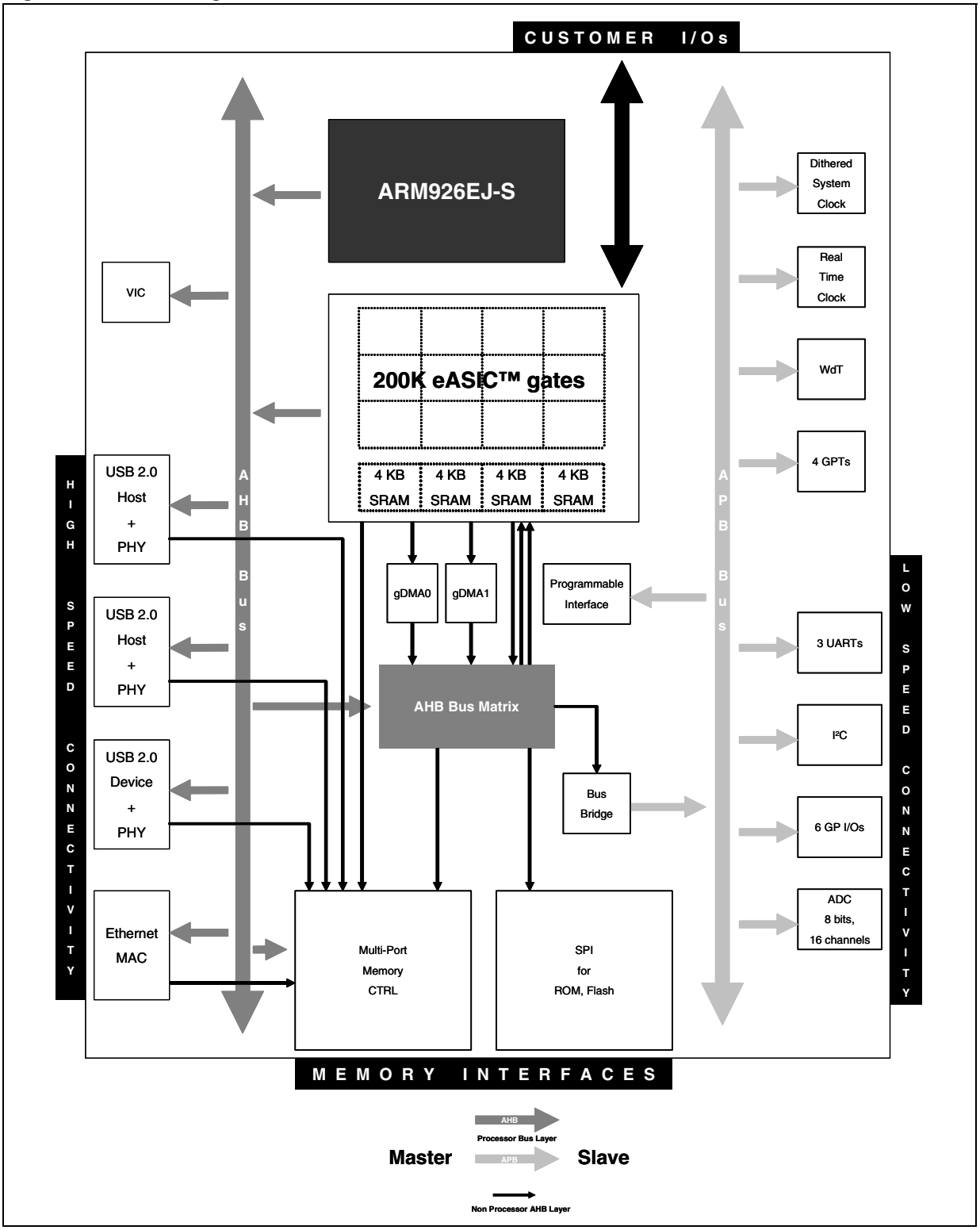
- Four 16 bit timers with 8 bit prescaler
- Frequency range: 3.96 Hz - 66.5 MHz
- Operating mode: Auto Reload and Single Shot

3.13 Customizable logic

- 200K equivalent ASIC gate (16K LUT equivalent) configurable either custom logic or 32 KBytes single-port SRAM or mixing logic and RAM
- 2 dedicated buses, each of them connected with a 4 channel DMA
- 8 interrupt lines (level type) available
- 112 dedicated GP I/Os
- Single VIA mask configurable interconnections
- Emulation by an external FPGA, keeping on-chip and I/O interfaces

4 Block diagram

Figure 1. Block diagram



5 Pin description

5.1 Functional pin groups

With reference to *Figure 20*. Package schematic -*Section 26*, here follows the pin list, sorted by their belonging IP. All supply and ground pins are classified as power signals and gathered in the *Table 2*.

Table 1. Pin description by functional groups

Group	Signal Name	Ball	Direction	Function	Pin Type
ADC	AIN[0]	V20	Input	ADC analog input channel	Analog buffer, 3.3 V capable
	AIN[1]	V19			
	AIN[2]	V18			
	AIN[3]	V17			
	AIN[4]	V16			
	AIN[5]	T22			
	AIN[6]	T21			
	AIN[7]	T19			
	AIN[8]	P18			
	AIN[9]	N18			
	AIN[10]	M18			
	AIN[11]	L18			
	AIN[12]	U22			
	AIN[13]	U21			
	AIN[14]	U20			
	AIN[15]	U19			
	TEST_OUT	T20	Output	ADC output test pad	
DEBUG	TEST0	E22	Input	Test configuration port. For the functional mode they have to be set to 0	TTL input buffer, 3.3 V capable, with Pull Down
	TEST1	E21			
	TEST2	D22			
	TEST3	D21			
	PLL_BYPASS	H5	Input	Enable / disable PLL bypass	TTL Schmitt trigger input buffer, 3.3 V capable
eASIC	eASICGP_IO[0]	E1	I/O	eASIC general purpose IO	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability
	eASICGP_IO[1]	F2			

Table 1. Pin description by functional groups (continued)

Group	Signal Name	Ball	Direction	Function	Pin Type
eASIC	eASICGP_IO[2]	G3	I/O	eASIC general purpose IO	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability
	eASICGP_IO[3]	D1			
	eASICGP_IO[4]	E2			
	eASICGP_IO[5]	C1			
	eASICGP_IO[6]	F3			
	eASICGP_IO[7]	D2			
	eASICGP_IO[8]	B1			
	eASICGP_IO[9]	G4			
	eASICGP_IO[10]	E3			
	eASICGP_IO[11]	C2			
	eASICGP_IO[12]	A1			
	eASICGP_IO[13]	F4			
	eASICGP_IO[14]	D3			
	eASICGP_IO[15]	B2			
	eASICGP_IO[16]	A2			
	eASICGP_IO[17]	C3			
	eASICGP_IO[18]	E4			
	eASICGP_IO[19]	G5			
	eASICGP_IO[20]	B3			
	eASICGP_IO[21]	D4			
	eASICGP_IO[22]	F5			
	eASICGP_IO[23]	A3			
	eASICGP_IO[24]	E5			
	eASICGP_IO[25]	C4			
	eASICGP_IO[26]	A4			
	eASICGP_IO[27]	B4			
	eASICGP_IO[28]	C5			
	eASICGP_IO[29]	D5			
	eASICGP_IO[30]	B5			
	eASICGP_IO[31]	A5			
	eASICGP_IO[32]	E6			
	eASICGP_IO[33]	D6			
	eASICGP_IO[34]	B6			

Table 1. Pin description by functional groups (continued)

Group	Signal Name	Ball	Direction	Function	Pin Type
eASIC	eASICGP_IO[35]	C6	I/O	eASIC general purpose IO	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability
	eASICGP_IO[36]	A7			
	eASICGP_IO[37]	A6			
	eASICGP_IO[38]	C7			
	eASICGP_IO[39]	B7			
	eASICGP_IO[40]	E7			
	eASICGP_IO[41]	D7			
	eASICGP_IO[42]	E8			
	eASICGP_IO[43]	A8			
	eASICGP_IO[44]	B8			
	eASICGP_IO[45]	C8			
	eASICGP_IO[46]	D8			
	eASICGP_IO[47]	B9			
	eASICGP_IO[48]	A9			
	eASICGP_IO[49]	A10			
	eASICGP_IO[50]	C9			
	eASICGP_IO[51]	D9			
	eASICGP_IO[52]	B10			
	eASICGP_IO[53]	A11			
	eASICGP_IO[54]	E9			
	eASICGP_IO[55]	C10			
	eASICGP_IO[56]	B11			
	eASICGP_IO[57]	D10			
	eASICGP_IO[58]	A12			
	eASICGP_IO[59]	C11			
	eASICGP_IO[60]	B12			
	eASICGP_IO[61]	A13			
	eASICGP_IO[62]	E10			
	eASICGP_IO[63]	D11			
	eASICGP_IO[64]	C12			
	eASICGP_IO[65]	B13			

Table 1. Pin description by functional groups (continued)

Group	Signal Name	Ball	Direction	Function	Pin Type
eASIC	eASICGP_IO[66]	A14	I/O	eASIC general purpose IO	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability
	eASICGP_IO[67]	A15			
	eASICGP_IO[68]	B14			
	eASICGP_IO[69]	C13			
	eASICGP_IO[70]	D12			
	eASICGP_IO[71]	E11			
	eASICGP_IO[72]	A16			
	eASICGP_IO[73]	B15			
	eASICGP_IO[74]	C14			
	eASICGP_IO[75]	D13			
	eASICGP_IO[76]	A17			
	eASICGP_IO[77]	B16			
	eASICGP_IO[78]	E12			
	eASICGP_IO[79]	C15			
	eASICGP_IO[80]	A18			
	eASICGP_IO[81]	B17			
	eASICGP_IO[82]	D14			
	eASICGP_IO[83]	A19			
	eASICGP_IO[84]	C16			
	eASICGP_IO[85]	E13			
	eASICGP_IO[86]	B18			
	eASICGP_IO[87]	D15			
	eASICGP_IO[88]	C17			
	eASICGP_IO[89]	B19			
	eASICGP_IO[90]	C18			
	eASICGP_IO[91]	E14			
	eASICGP_IO[92]	D16			
	eASICGP_IO[93]	B21			
	eASICGP_IO[94]	C19			
	eASICGP_IO[95]	D17			
	eASICGP_IO[96]	E15			
	eASICGP_IO[97]	C20			
	eASICGP_IO[98]	D18			

Table 1. Pin description by functional groups (continued)

Group	Signal Name	Ball	Direction	Function	Pin Type
eASIC	eASICGP_IO[99]	E16	I/O	eASIC general purpose IO	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability
	eASICGP_IO[100]	D19			
	eASICGP_IO[101]	E17			
	eASICGP_IO[102]	D20			
	eASICGP_IO[103]	E18			
	eASICGP_IO[104]	E19			
	eASICGP_IO[105]	F18			
	eASICGP_IO[106]	E20			
	eASICGP_IO[107]	F20			
	eASICGP_IO[108]	F19			
	eASICGP_IO[109]	G19			
	eASICGP_IO[110]	G20			
	eASICGP_IO[111]	G18			
	eASIC_EXT_CLOCK	H18			
	eASIC_PI_CLOCK	R18		eAISC Program Interface out clock	TTL bidirectional buffer, 3.3 V capable, 8 mA drive capability
	eASIC_CLK	G2		eASIC output clock	
	CONFIG_DEVEL	A20	Input	External FPGA emulation mode	TTL input buffer, 3.3 V capable with Pull Down
Ethernet	TX_CLK	H19	Input	Ethernet input TX clock	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability, with Pull Down
	TXD[0]	J18	Output	Ethernet TX output data	
	TXD[1]	J19			
	TXD[2]	K18			
	TXD[3]	J20			
	TX_EN	J21		Ethernet TX enable	
	CRS	J22	Input	Carrier sense input	
	COL	K19		Collision detection input	
	RX_CLK	K20		Ethernet input RX clock	
	RXD[0]	K21		Ethernet RX input data	
	RXD[1]	K22			
	RXD[2]	L19			
	RXD[3]	L20	Input		
	RX_DV	L21	Input	Data valid on RX	
	RX_ER	L22	Input	Data error detected	

Table 1. Pin description by functional groups (continued)

Group	Signal Name	Ball	Direction	Function	Pin Type
Ethernet	MDC	M19	Output	Output timing reference for MDIO	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability, with Pull Down
	MDIO	M20	I/O	I/O data to PHY	
GPI/Os	GP_IO[0]	R22	I/O	General Purpose IO	TTL bidirectional buffer, 3.3 V capable, 8 mA drive capability
	GP_IO[1]	R21			
	GP_IO[2]	R20			
	GP_IO[3]	R19			
	GP_IO[4]	P22			
	GP_IO[5]	P21			
I^2C	SDA	M21	I/O	I^2C serial data	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability, with Pull Up
	SCL	M22			
JTAG	TDO	A21	Output	Jtag TDO	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability
	TDI	A22	Input	Jtag TDI	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability, with Pull Up
	TMS	B20	Input	Jtag TMS	
	RTCK	B22	Output	Jtag output clock	
	TCK	C21	Input	Jtag clock	
	nTRST	C22	Output	Jtag reset	
MASTER CLOCK	MCLK_in	T1	Input	12 MHz input crystal	Oscillator 3.3 V capable
	MCLK_out	U1	Output	12 MHz output crystal	
MASTER RESET	MRESET	H4	Input	Master reset	TTL Schmitt trigger input buffer, 3.3 V capable
MPMC	MPMCDATA[0]	AA12	I/O	DDR / SDRAM data	LVTTL / SSTL ClassII bidirectional buffer
	MPMCDATA[1]	Y12			
	MPMCDATA[2]	W12			
	MPMCDATA[3]	AB13			
	MPMCDATA[4]	AA13			
	MPMCDATA[5]	Y13			

Table 1. Pin description by functional groups (continued)

Group	Signal Name	Ball	Direction	Function	Pin Type
MPMC	MPMCDATA[6]	W13	I/O	DDR / SDRAM data	LVTTL / SSTL ClassII bidirectional buffer
	MPMCDATA[7]	AA14			
	MPMCDATA[8]	AA16			
	MPMCDATA[9]	AB18			
	MPMCDATA[10]	AB19			
	MPMCDATA[11]	AB20			
	MPMCDATA[12]	AB21			
	MPMCDATA[13]	AA21			
	MPMCDATA[14]	AB22			
	MPMCDATA[15]	AA22			
	MPMCDATA[16]	AA18	I/O	SDRAM data	TTL bidirectional buffer, 3.3 V capable, 8 mA drive capability
	MPMCDATA[17]	AA17			
	MPMCDATA[18]	Y22			
	MPMCDATA[19]	Y21			
	MPMCDATA[20]	Y20			
	MPMCDATA[21]	Y19			
	MPMCDATA[22]	Y18			
	MPMCDATA[23]	Y17			
	MPMCDATA[24]	Y16			
	MPMCDATA[25]	W22			
	MPMCDATA[26]	W21			
	MPMCDATA[27]	W20			
	MPMCDATA[28]	W19			
	MPMCDATA[29]	W18			
	MPMCDATA[30]	W17			
	MPMCDATA[31]	W16			
	MPMCADDROUT[0]	AB8	Output	DDR / SDRAM data	LVTTL / SSTL ClassII bidirectional buffer
	MPMCADDROUT[1]	AA8			
	MPMCADDROUT[2]	Y8			
	MPMCADDROUT[3]	W8			
	MPMCADDROUT[4]	AB9			
	MPMCADDROUT[5]	AA9			
	MPMCADDROUT[6]	Y9			
	MPMCADDROUT[7]	W9			
	MPMCADDROUT[8]	AB10			

Table 1. Pin description by functional groups (continued)

Group	Signal Name	Ball	Direction	Function	Pin Type
MPMC	MPMCADDROUT[9]	AA10	Output	DDR / SDRAM data	LVTTL / SSTL ClassII bidirectional buffer
	MPMCADDROUT[10]	Y10			
	MPMCADDROUT[11]	W10			
	MPMCADDROUT[12]	AB11			
	MPMCADDROUT[13]	AA11			
	MPMCADDROUT[14]	Y11			
	nMPMCDYCSOUT[0]	AB6		DDR / SDRAM chip select	
	nMPMCDYCSOUT[1]	AA6			
	nMPMCDYCSOUT[2]	Y6			
	nMPMCDYCSOUT[3]	W6			
	MPMCCKEOUT[0]	W11		DDR / SDRAM clock enable output	
	MPMCCKEOUT[1]	AB12			
	MPMCCLKOUT[0]	AB17		DDR / SDRAM output clock 1	LVTTL / SSTL ClassII bidirectional differential buffer
	nMPMCCLKOUT[0]	AB16		DDR / SDRAM output clock 1 neg.	
	MPMCCLKOUT[1]	AB15		DDR / SDRAM output clock 2	
	nMPMCCLKOUT[1]	AB14		DDR / SDRAM output clock 2 neg.	
	MPMCDQMOUT[0]	Y14		DDR / SDRAM data mask out	LVTTL / SSTL ClassII bidirectional buffer
	MPMCDQMOUT[1]	W15			
	MPMCDQMOUT[2]	AA19		SDRAM data mask out	TTL bidirectional buffer, 3.3 V capable, 8 mA drive capability
	MPMCDQMOUT[3]	AA20			
	MPMCDQS[0]	AA15		DDR data strobe	LVTTL / SSTL ClassII bidirectional buffer
	MPMCDQS[1]	Y15			
	nMPMCCASOUT	Y7		DDR / SDRAM CAS output strobe	
	nMPMCRASOUT	AA7			
	nMPMCWEOUT	AB7		DDR / SDRAM write enable	
	SSTL_VREF	W14	Input	Voltage reference SSTL / CMOS mode. This pin is used both as logic state and as power supply	Analog buffer, 3.3 V capable
RTC	RTCXO	AB5	Output	32 KHz output crystal	Oscillator 1.2 V capable
	RTCXI	AB4	Input	32 KHz input crystal	
SMI	SMINCS[0]	G22	Output	Serial Flash chip select	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability

Table 1. Pin description by functional groups (continued)

Group	Signal Name	Ball	Direction	Function	Pin Type
SMI	SMINCS[1]	G21		Serial Flash chip select	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability
	SMINCS[2]	F22			
	SMINCS[3]	F21			
	SMICLK	H20	Output	Serial Flash output clock	TTL bidirectional buffer, 3.3 V capable, 8 mA drive capability
	SMIDATAIN	H21	Input	Serial Flash data in	TTL bidirectional buffer, 3.3 V capable, 8 mA drive capability, with Pull Up
	SMIDATAOUT	H22	Output	Serial Flash data out	TTL bidirectional buffer, 3.3 V capable, 8 mA drive capability
UARTs	UART1_RXD	N19	Input	Uart1 RX data	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability, with Pull Down
	UART1_TXD	N20	Output	Uart1 TX data	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability
	UART2_RXD	N21	Input	Uart2 RX data	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability, with Pull Down
	UART2_TXD	N22	Output	Uart2 TX data	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability
	UART3_RXD	P19	Input	Uart3 RX data	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability, with Pull Down
	UART3_TXD	P20	Output	Uart3 TX data	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability

Table 1. Pin description by functional groups (continued)

Group	Signal Name	Ball	Direction	Function	Pin Type
USBs	DMNS	W1	I/O	D - port of USB device	Analog buffer, 5 V tolerant
	DPLS	V1	I/O	D + port of USB device	
	HOST1_DP	P1	I/O	D - port of USB host1	
	HOST1_DM	N1	I/O	D + port of USB host1	
	HOST2_DP	L1	I/O	D - port of USB host2	
	HOST2_DM	K1	I/O	D + port of USB host2	
	HOST1_VBUS	H2	Output	USB host1 VBUS signal	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability
	HOST2_VBUS	H1	Output	USB host2 VBUS signal	
	OVERCURH1	G1	I/O	USB host1 overcurrent	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability, with Pull Down
	OVERCURH2	F1	I/O	USB host2 overcurrent	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability
	VBUS	H3	I/O	USB device VBUS signal	TTL bidirectional buffer, 3.3 V capable, 4 mA drive capability, with Pull Down
	RREF	K5	Input	USB reference resistor	Analog buffer, 3.3 V capable

Table 2. Pins belonging to POWER group

Group	Signal Name	Ball	Function
POWER	vdde3v3	Note (1)	Digital 3.3 V power
	vdd	Note (2)	Digital 1.2 V power
	gnde	Note (3)	Digital ground
	vdd2v5	Note (4)	DDR / SDR digital 3.3 / 2.5V power
	thermal_gnd	Note (5)	Thermal Pad
	anavdd_3v3_adc	V22	Dedicated ADC 3.3 V power
	anagnd_3v3_adc	U18	Dedicated ADC ground
	VREFP_adc	V21	ADC positive reference Voltage
	VREFN_adc	T18	ADC pegative reference Voltage
	vdd_dith	W7	DDR / SDR dedicated digital PLL 3.3 V power
	vss_dith	V7	DRR / SDR dedicated digital PLL ground

Table 2. Pins belonging to POWER group (continued)

Group	Signal Name	Ball	Function
POWER	SSTL_VREF	W14	Voltage reference SSTL / CMOS mode. This pin is used both as logic state and as power supply
	vdd1v2_date_osci	AB2	1.2 V dedicated power for RTC
	vdd_date_osci	AA5	1.2 V dedicated power for RTC
	gnd_date_osci	AA4	Dedicated digital ground for RTC
	gnde_date_osci	AB3	Dedicated digital ground for RTC
	anavdd_3v3_pll1600	R2	Dedicated USB PLL analog 3.3 V power
	anagnd_3v3_pll1600	P4	Dedicated USB PLL analog ground
	digvdd_1v2_pll1600	R4	Dedicated USB PLL digital 1.2 V power
	diggnd_1v2_pll1600	U2	Dedicated USB PLL digital ground
	vddl_1v2_d	W3	Dedicated USB 1.2 V power
	vddb_1v2_d	U4	Dedicated USB 1.2 V power
	vddc_1v2_d	U3	Dedicated USB 1.2 V power
	vdd_usb	P2	Dedicated USB 1.2 V power
	vddc_1v2_h1	N5	Dedicated USB 1.2 V power
	vddb_1v2_h1	N3	Dedicated USB 1.2 V power
	vddl_1v2_h1	L4	Dedicated USB 1.2 V power
	vddc_1v2_h0	K4	Dedicated USB 1.2 V power
	vddb_1v2_h0	K3	Dedicated USB 1.2 V power
	vddl_1v2_h0	J4	Dedicated USB 1.2 V power
	vdd3_3v3_d	W4	Dedicated USB 3.3 V power
	vdde3v3_usb	P5	Dedicated USB 3.3 V power
	vdd3_3v3_h1	L3	Dedicated USB 3.3 V power
	vdd3_3v3_h0	J3	Dedicated USB 3.3 V power
	vssl_3v3_d	W5	Dedicated USB ground
	vssb_1v2_d	W2	Dedicated USB ground
	vssc_1v2_d	U5	Dedicated USB ground
	gnde_usb	R3	Dedicated USB ground
	gnd_usb	N4	Dedicated USB ground
	vssc_1v2_h1	N2	Dedicated USB ground
	vssb_1v2_h1	M2	Dedicated USB ground
	vssl_3v3_h0	J2	Dedicated USB ground
	vssl_3v3_h1	L2	Dedicated USB ground
	vssb_1v2_h0	J5	Dedicated USB ground
	vssc_1v2_h0	K2	Dedicated USB ground

1. Signal spread on the following balls: F7, F8, F9, F12, F13, F14, F17, G17 H6, J6, K17, L17, M6, N6, P6, P17, R17, U6, U16.
2. Signal spread on the following balls: F6, F10, F11, F15, F16, G6, H17, J17, K6, L6, M17, N17, R6, T6, T17, U17, V8, V15..
3. Signal spread on the following balls: J9 to J14, K9 to K14, L9 to L14, M9 to M14, N9 to N14, P9 to P14.
4. Signal spread on the following balls: U7 to U15
5. Signal spread on the following balls: L5, M3, M4, M5, R1, R5, T2 to T5.

5.2 Special I/Os

5.2.1 USB 2.0 Transceiver

SPEAr Head has three USB 2.0 UTMI + Multimode ATX transceivers. One transceiver will be used by the USB Device controller, and two will be used by the Hosts. These are all integrated into a single USB three-PHYs macro.

5.2.2 DRAM

Data and address buses of Multi-Port Memory Controller used to connect to the banks memory are constituted of programmable pins.

6 Memory map

Table 3. Memory map

START ADRESS	END ADDRESS	PERIPHERAL	NOTES
0x0000_0000	0x03FF_FFFF	Serial Memory	64 MB (on reset before the remap)
0x0000_0000	0x0FFF_FFFF	DRAM	256 MB
0x1000_0000	0x1000_07FF	USB Device	
0x1000_8000	0x1000_8FFF	USB Device Plug Detect	
0x1000_9000	0x1000_9FFF	USB Host 1 EHCI	
0x1000_A000	0x1000_AFFF	USB Host 1 OHCI	
0x1000_B000	0x1000_BFFF	Vectored Interrupt CTRL	
0x1000_C000	0x1000_CFFF	DRAM CTRL (MPMC)	
0x1000_D000	0x1000_DFFF	USB Host 2 EHCI	
0x1000_E000	0x1000_EFFF	USB Host 2 OHCI	
0x1000_F000	0x1000_F3FF	Serial Memory CTRL	
0x1000_F400	0x11FF_FFFF	Default Slave	
0x1200_0000	0x1200_0FFF	APB Control Status Register	
0x1200_1000	0x1200_1FFF	GPT 0 and GPT 1	
0x1200_2000	0x1200_2FFF	GPT 2 and GPT 3	
0x1200_3000	0x1200_3FFF	General Configuration Registers	
0x1200_4000	0x1200_4FFF	WdT	
0x1200_5000	0x1200_5FFF	RTC	
0x1200_6000	0x1200_6FFF	GPIO 0 → GPIO 5	
0x1200_7000	0x1200_7FFF	I²C	
0x1200_8000	0x1200_8FFF	UART 1	
0x1200_9000	0x1200_9FFF	UART 2	
0x1200_A000	0x1200_AFFF	UART 3	
0x1200_B000	0x1200_BFFF	ADC	
0x1200_C000	0x1200_CFFF	gDMA 1	
0x1200_D000	0x1200_DFFF	gDMA 2	
0x1200_E000	0x120F_FFFF	Default Slave	
0x1210_0000	0x12FF_FFFF	Default Slave	
0x1300_0000	0x1300_03FF	eASIC™ Programmable Interface	
0x1300_0400	0x13FF_FFFF	Default Slave	

Table 3. Memory map (continued)

START ADRESS	END ADDRESS	PERIPHERAL	NOTES
0x1400_0000	0x15FF_FFFF	Ethernet MAC	
0x1600_0000	0x19FF_FFFF	Serial Memory	64 MB (on reset after the remap)
0x1A00_0000	0x1FFF_FFFF	Default Slave	(maximum addressable size available for the set of the full master and the 2 slaves is 1.5 GB)
0x2000_0000	0x7FFF_FFFF	eASIC™ AHB SUBSYSTEM	

Write transactions on the APB Bus are all considered 32 bit wide unless otherwise stated. All the access to the Default Slave will cause an abort exception.

Memory map is repeated starting from the address 0x8000_0000.

7 Power on sequence

7.1 SPEAr Head200 software architecture

7.1.1 Boot process

Memory mapping

A major consideration in the design of an embedded ARM application is the layout of the memory map, in particular the memory that is situated at address 0x0. Following reset, the core starts to fetch instructions from 0x0, so there must be some executable code accessible from that address. In an embedded system, this requires ROM to be present, at least initially.

Serial Flash at 0x0

The SPEAr Head200 has been designed to use the REMAP concept into its AHB primary bus decoder. The decoder selection for the initial address range (first 64 MB) is conditioned with the content of an AHB remap register, which can be programmed by software at any time.

The Serial Flash memory space is accessible in the address range 0x9600_0000 - 0x99FF_FFFF (64 MB), with or without remap.

At reset, the Serial Flash memory is 'aliased' at 0x0, which means that the AHB decoder selects the Serial Flash space when accessing the address range 0x0000_0000 to 0x03FF_FFFF. Please note that DRAM is unreachable in this state.

Table 4. Memory mapping at reset (REMAP = 0)

ADDRESS RANGE	SIZE [MB]	DESCRIPTION
0x9600_0000 - 0x99FF_FFFF	64	Serial Flash
0x0000_0000 - 0x03FF_FFFF	64	Serial Flash (remap)

DRAM at 0x0

After reset, the boot program makes the remapping, so that the system will be able to access the complete 256 MB of logic memory space associated to DRAM in the range 0x0000_0000 - 0x0FFF_FFFF.

Table 5. Memory Mapping after reset after remapping

ADDRESS RANGE	SIZE [MB]	DESCRIPTION
0x9600_0000 - 0x99FF_FFFF	64	Serial Flash
0x0000_0000 - 0x0FFF_FFFF	256	DRAM

7.1.2 Booting sequence

A simple initial description of the boot process is showed in the following steps:

1. Power on to fetch the RESET vector at 0x0000_0000 (from the aliased-copy of Serial Flash).
2. Perform any critical CPU initialization at this time.
3. Load into the Program Counter (PC) the address of a routine that will be executed directly from the non-aliased mapping of Serial Flash (0x9600_0000 + addr_of_routine) and which main objectives are:
   - un-map the aliased-copy of the Serial Flash (set REMAP = 1)
   - copy the program text and data into DRAM.
4. Returning from this routine will set the Program Counter back to DRAM.

8 ARM926EJ-S

The processor is the powerful ARM926EJ-S, targeted for multi-tasking applications.

Belonging to ARM9 general purposes family microprocessor, it principally stands out for the Memory Management Unit, which provides virtually memory features, making it also compliant with WindowsCE, Linux and SymbianOS operating systems.

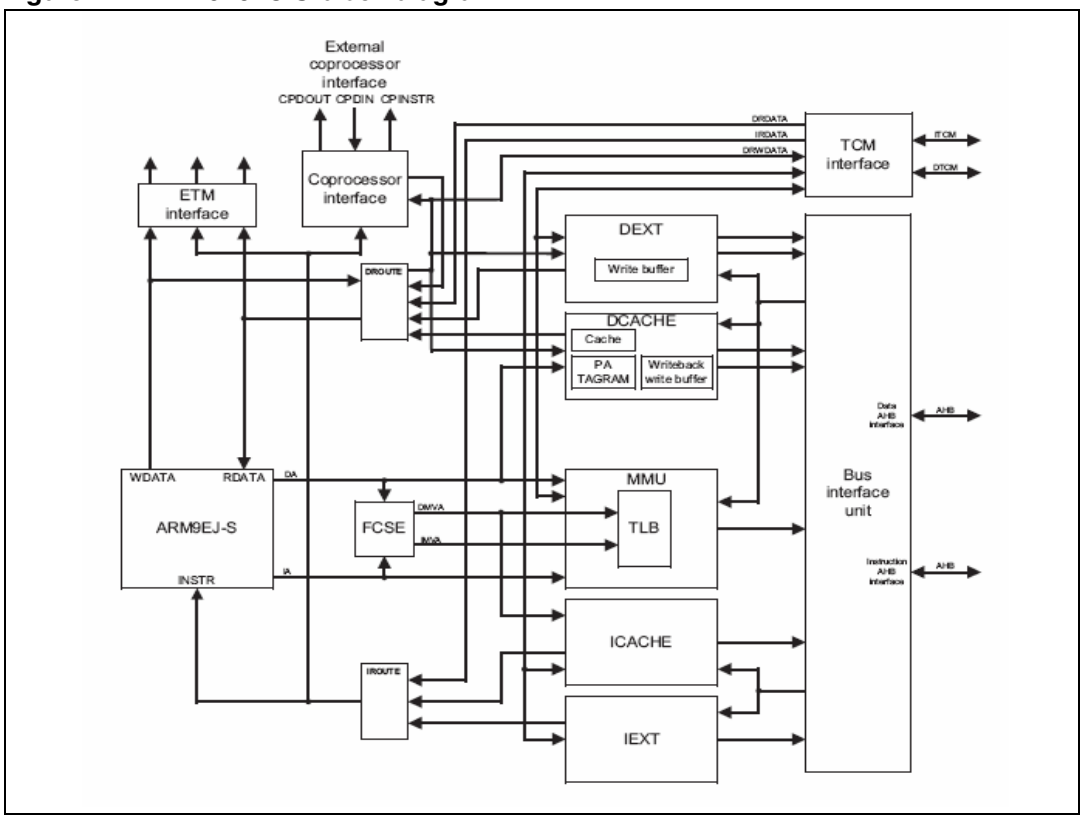
The ARM926EJ-S supports the 32 bit ARM and 16 bit Thumb instruction sets, enabling the user to trade off between high performance and high code density and includes features for efficient execution of Java byte codes.

Besides, it has the ARM debug architecture and includes logic to assist in software debug.

Its main features are:

- f_{MAX} 266 MHz (downward scalable)
- MMU
- 32 KB of instruction CACHE
- 16 KB of data CACHE
- 8 KB of instruction TCM (Tightly Coupled Memory)
- 8 KB of data TCM
- AMBA Bus interface
- Coprocessor interface
- JTAG
- ETM9 (Embedded Trace Macro-cell) for debug; large size version.

Figure 2. ARM926EJ-S block diagram



9 Clock and reset system

9.1 Overview

The Clock System block is a fully programmable block able to generate all clocks necessary at the chip, except for USB 2.0 Host and Device controllers, which have a dedicated PLL.

The clocks, at default operative frequency, are:

- clock @ 266 MHz for ARM system
- clock @ 133 MHz for AHB Bus and AHB peripherals, eASIC MacroCell and Bus Bridge included
- clock @ 66.5 MHz for, APB Bus and APB peripherals, Bus Bridge included
- clock @ 20 MHz for eASIC Programmable Interface

The frequencies are the maximum allowed value and the user can modify them by programming dedicated registers.

The Clock System consists of 2 main parts: a Multi-Clock Generator block and an internal PLL.

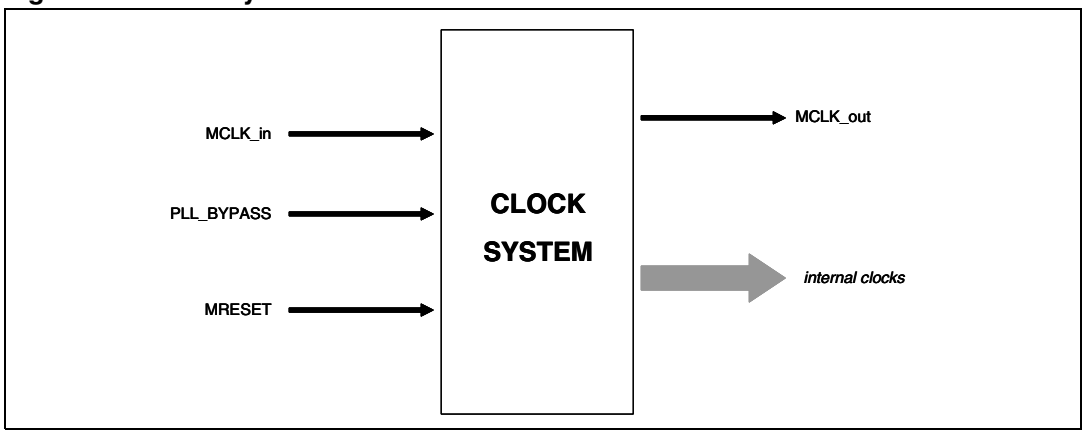
The Multi-Clock Generator block, starting from a reference signal (which generally is delivered from the PLL), generates all clocks for the IPs of SPEAr Head200 according to dedicated programmable registers.

The PLL, starting from the oscillator input of 12 MHz, generates a clock signal at a frequency corresponding at the highest of the chip, which is the reference signal used by the Multi-Clock Generator block to obtain all chip clocks. Its main features is the Electro-Magnetic Interference reduction capability: user has the possibility to set up the PLL in order to add a triangular wave to the VCO clock; the resulting signal will have the spectrum (and the power) spread on a small range (programmable) of frequencies centred on F0 (VCO Freq.), obtaining minimum electromagnetic emissions. This method replace all the other traditional methods of E.M.I. reduction, as filtering, ferrite beads, chokes, adding power layers and ground planets to PCBs, metal shielding etc., allowing sensible cost saving for customers.

There are 3 operating modes:

- Normal mode: the Clock System delivers signals at the operative frequency. The reference signal of the Multi-Clock Generator block is that generated by PLL
- Pseudo-Functional mode: in this mode PLL is bypassed and the reference signal is delivered from off-chip. The generated clocks are coherent with programmed registers.
  The purpose of Pseudo-Functional mode is let the rest of the chip properly work in case of PLL failure.
  This mode is set by assigning the following logic state to the Test pins:
  TEST0 → 1
  TEST1 → 0
  TEST2 → 0
  TEST3 → 0
- Debug mode: is the operating state when the ARM is in Debug mode. The block works as in Normal Mode but, APB peripheral clocks, eASIC clock and eASIC PI clock are gated

Figure 3. Clock system block interfaces



The I/O signals accessible from off-chip are listed in *Table 6* Clock System I/O off-chip interfaces:

Table 6. Clock system I/O off-chip interface

SIGNALS	DIRECTION	SIZE [bit]	DESCRIPTION
MCLK_in	Input	1	Oscillator input (12 MHz)
PLL_BYPASS	Input	1	External clock in Test mode
MCLK_out	Output	1	Oscillator output. It supplies the signal MCLK_in inverted
MRESET	Input	1	Asynchronous reset

9.2 Reset and PLL change parameters sequence

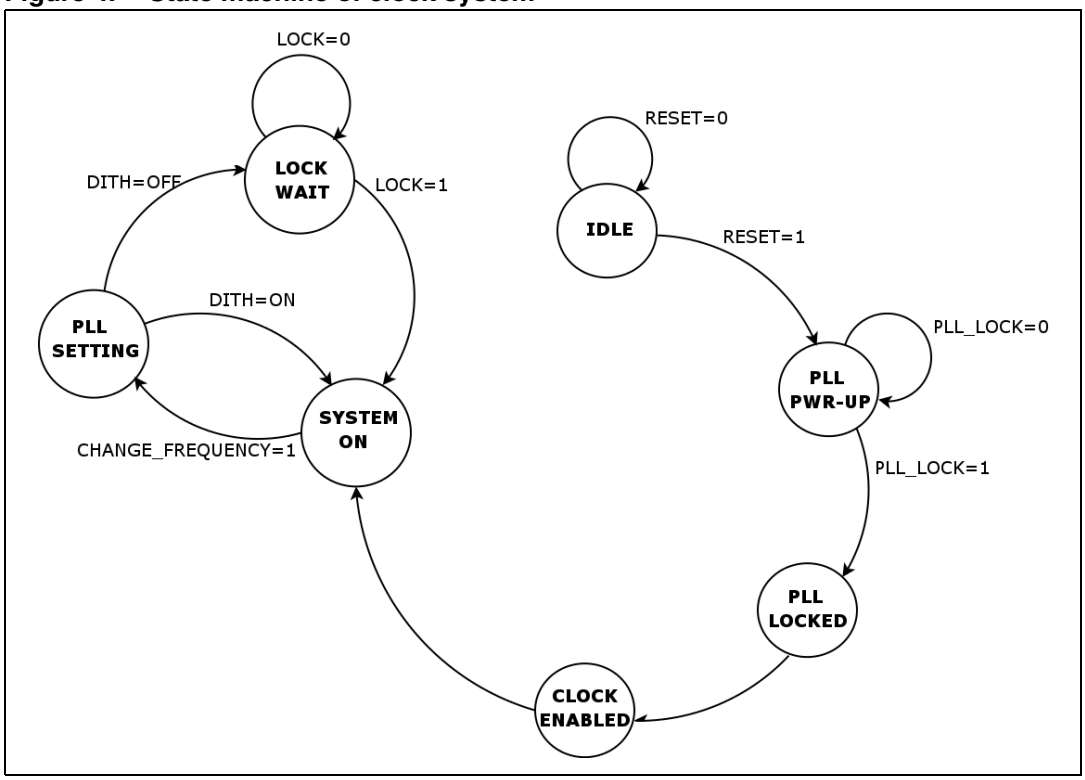
Figure 4 shows a simplified flow chart of clock system FSM.

The system remains in an IDLE state until RESET signal is asserted: RESET → 0.

When RESET = 0 the FSM change state and reaches the PLL PWR-UP state; when PLL locks (PLL_LOCK = 1), means that the PLL_OUT signal oscillates at 264 MHz and the PLL starts to work, so that the FSM advances in the next states.

In CLOCK ENABLED state the clocks can propagate in the system; then the FSM goes in SYSTEM ON state; it remains in this state in the normal chip working.

Figure 4. State machine of clock system



To reach at a clock frequency of 266 MHz, PLL had to be appropriately programmed because this frequency isn't an integer multiple of 12 MHz.

After this programming, the FSM stops all the clocks and exits from SYSTEM ON state proceeding in PLL SETTING state; here the new parameters are stored in the PLL.

If the PLL isn't in Dithered mode the FSM waits for PLL lock, going in LOCK WAIT state, and then will reach SYSTEM ON state when PLL locks.

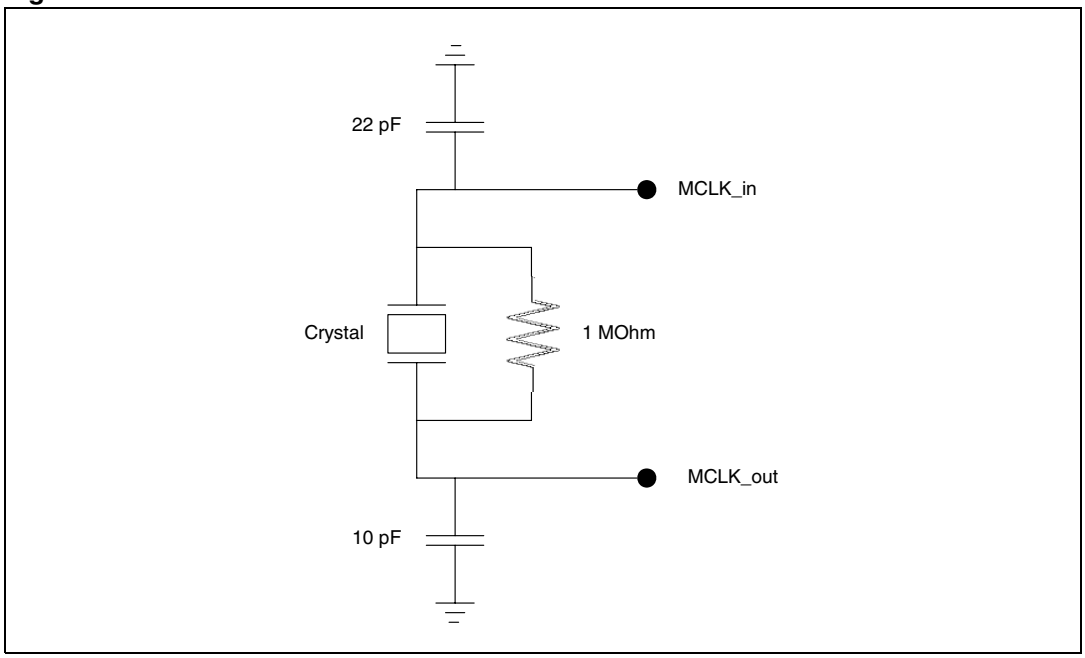
If the PLL is in Dithered mode, the lock signal loses his meaning and there's no need to wait for PLL lock, so the FSM jumps directly from PLL SETTING to SYSTEM ON.

When FSM is in SYSTEM ON, all clocks are enabled.

9.3 Crystal connection

The crystal will be connected to the chip as shown in the following schematic.

Figure 5. Oscillator board schematic



The chip will only be used with a crystal (not a ceramic resonator). The *Figure 6* Model for crystal details the model used and the *Table 7* Parameters for 12 MHz Crystal specifies the parameters for the crystal.

Figure 6. Model for crystal

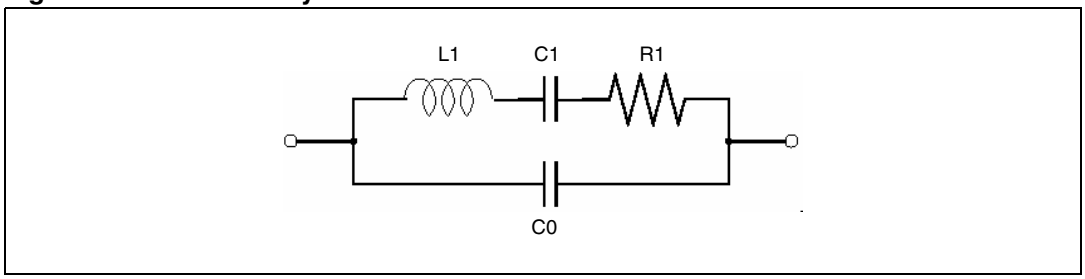


Table 7. Parameters for 12 MHz crystal

#	L1 (uH)	C1 (fF)	R1 (Ohm)	C0 pF)	Notes:
Model 1	6200	15.7	10	7	
Model 2	7500	13.2	15	3.3	
Model 3	470	94.6	14.5	11.6	
Model 4	17465	10.08	12.45	2.75	11.9965MHz Taitien

10 Vectored interrupt controller

10.1 Overview

The Vector Interrupt Controller provides a software interface to interrupt system, in order to determine the source that is requesting a service and where the service routing is loaded.

It supplies the starting address, or vector address, of the service routine corresponding to the highest priority requesting interrupt source.

In an ARM system 2 level of interrupt are available:

- Fast Interrupt Request (FIQ) for low latency interrupt handling
- Interrupt Request (IRQ) for standard interrupts

Generally, you only use a single FIQ source at a time to provide a true low-latency interrupt. This has the following benefits:

- You can execute the interrupt service routine directly without determining the source of the interrupt
- It reduces interrupt latency. You can use the banked registers available for FIQ interrupts more efficiently, because you do not require a context save

The interrupt inputs must be level sensitive, active HIGH, and held asserted until the interrupt service routine clears the interrupt. Edge-triggered interrupts are not compatible. The interrupt inputs do not have to be synchronous to AHB clock.

The main features of Vectored Interrupt Controller are:

- Compliance to AMBA Specification Rev. 2.0
- IRQ and FIQ interrupt generation
- AHB mapped for faster interrupt
- Hardware priority
- Support for 32 standard interrupts
- Support for 16 vectored interrupts
- Software interrupt generation
- Interrupt masking
- Interrupt request status.

Since 32 interrupts are supported, there are 32 interrupt input lines, coming from different sources. They are selected by a bit position and the software controls every line to generate software interrupts; it can generate 16 vectored interrupts. A vectored interrupt can generate only an IRQ interrupt.

The interrupt priority is controlled by hardware and it is as follow:

1. FIQ interrupt
2. Vectored IRQ interrupt. The higher priority is 0; the lower is 15
3. Non vectored IRQ interrupt

10.2 Vector interrupt controller flow sequence

The following procedure shows the sequence for the vectored interrupt flow:

1. An interrupt occurs
2. The CPU branches to either the IRQ or FIQ Interrupt Vector
3. If the Interrupt is an IRQ, CPU read the VICVectAddr register and branch to the interrupt service routine. This can be done using a LDR PC instruction. Reading the VICVectAddr register updates the interrupt controllers hardware priority register
4. Stack the Workspace so that the IRQ interrupts can be re-enabled
5. Enable the IRQ interrupts so that a higher priority can be serviced
6. Execute the Interrupt Service Routine (ISR)
7. Clear the Requesting Interrupt in the peripheral, or write to the VICSoftIntClear register if the request was generated by a software interrupt
8. Disable the Interrupts and Restore the Workspace
9. Write to the VICVectAddr register. This clears the respective interrupt in the internal interrupt priority
10. Return from the interrupt. This re-enables the interrupts

10.3 Simple interrupt flow sequence

The following procedure shows the sequence for the simple interrupt flow:

1. An interrupt occurs
2. Branch to IRQ or FIQ interrupt vector
3. Branch to the Interrupt Handler
4. Interrogate the VICIRQStatus register to determine which source generated the interrupt, and prioritize the interrupts if there are multiple active interrupt sources. This takes a number of instructions to compute
5. Branch to the correct ISR
6. Execute the ISR
7. Clear the interrupt. If the request was generated by a software interrupt, the VICSoftIntClear register must be written too
8. Check the VICIRQStatus register to ensure that no other interrupt is active. If there is an active request go to Step 4
9. Return from Interrupt

10.4 Interrupt sources in SPEAr Head200

Table 8. Interrupt sources in SPEAr Head200

INTERRUPT LINE	SOURCE
0	eASIC0
1	eASIC1
2	eASIC2
3	eASIC3
4	SMI
5	RTC
6	USB HOST 1 – OHCI
7	USB HOST 2 – OHCI
8	USB HOST 1 – EHCI
9	USB HOST 2 – EHCI
10	USB DEVICE
11	MAC
12	I²C
13	GPT4
14	GPT3
15	gDMA1
16	gDMA0
17	GPT2
18	GPT1
19	UART2
20	UART1
21	UART0
22	ADC
23	Reserved
24	Reserved
25	Reserved
26	Reserved
27	Reserved
28	eASIC4
29	eASIC5
30	eASIC6
31	eASIC7

11 DMA controller block

SPEAr Head200 has 2 DMA Controllers used to transfer data between aASIC MacroCell and memory.

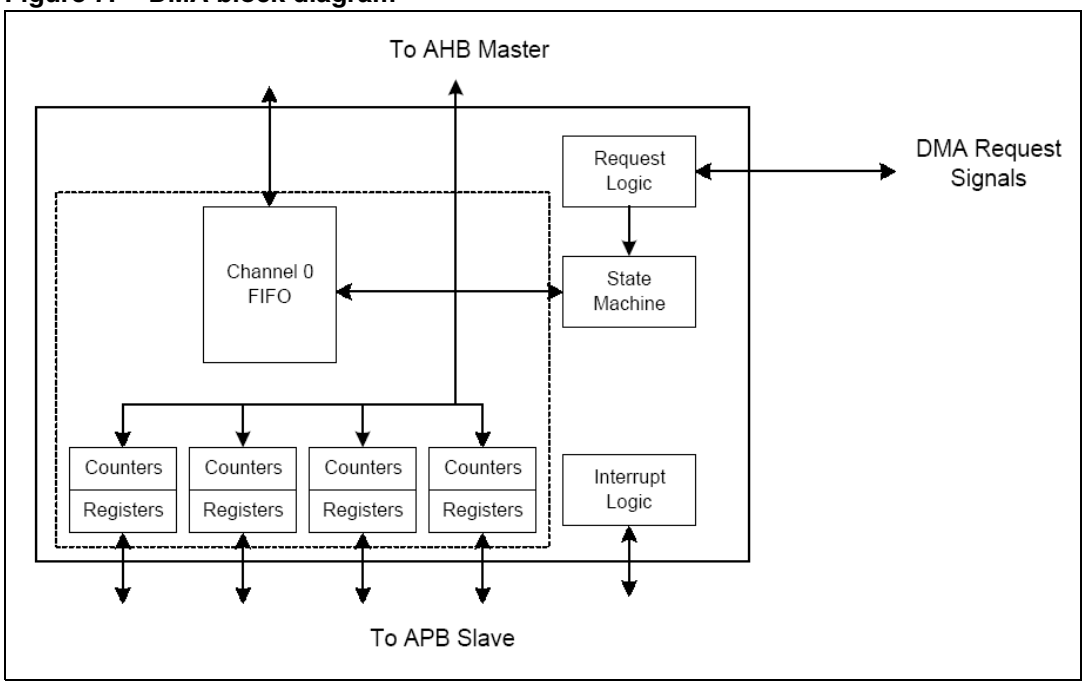
A DMA Controller can service up to 4 data streams at one time; a data transfer consists of a sequence of a DMA data packet transfers. There are two types of a data packet transfer

- one is from the source to the DMA Controller
- one other is from DMA Controller to the destination.

Each DMA Controller has an AHB Master interface to transfer data between DMA Controller and either a source or a destination, and has an APB Slave interface used to program its registers.

11.1 Functional description

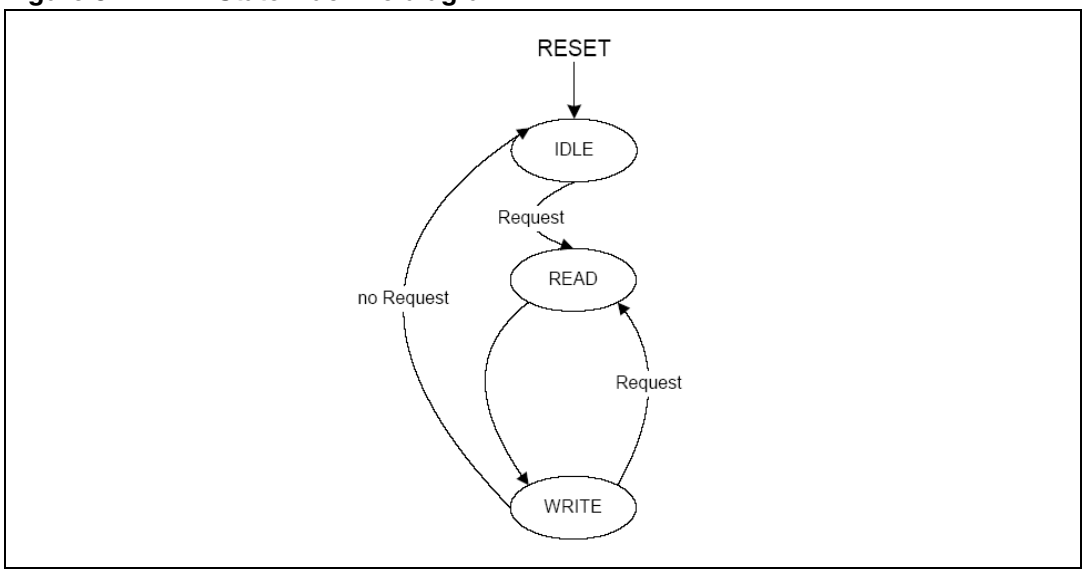
Figure 7. DMA block diagram



As a DMA requests are received, the Request Logic will arbiter between them and set the channel request signal. When a channel request is asserted, the State Machine starts a data transfer: first a data packet is transferred from a source to the DMA channel and then from the FIFO to the destination.

11.2 DMA control state machine

Figure 8. DMA State Machine diagram



The DMA control SM is always reset into the IDLE state.

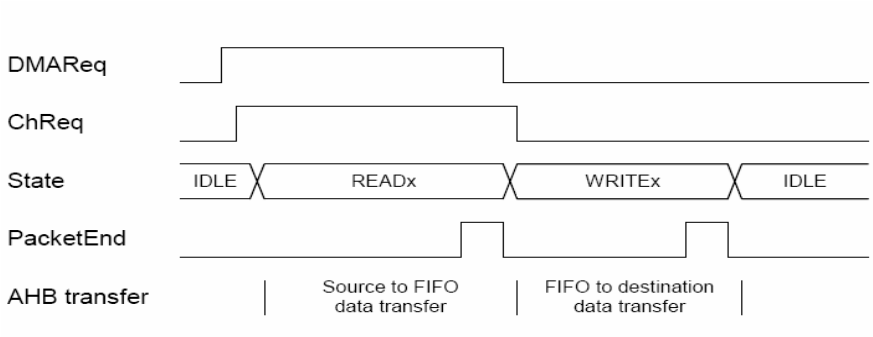
As a channel request is asserted, SM moves to READ state and the AHB Master will start a data packet transfer; SM selects appropriate source address.

When SM is in WRITE state, it selects the destination address and the data width from the Data Stream register and the AHB Master will transfer all data from the FIFO to the destination.

When the AHB Master has transferred the data packet, it asserts a Packend signal and the SM will move to the next state, which depends on the channel request signals.

The state transitions from the READ or WRITE states can occur only when a whole data packet has been transferred.

Figure 9. Data packet transfer



12 Multi-Port Memory Controller

12.1 Overview

The DRAM interface is controlled by the on-chip Multi-Port Memory Controller.

Its main features are:

- Supports for SDRAM up to 32 bit wide
- Supports for DDR up to 16 bit wide
- Maximum clock frequency 133 MHz
- 8 AHB port connections
- 4 chip selects
- Total addressable memory: 256 MB
- Maximum memory bank size: 64 MB
- Read and Write buffers to reduce latency
- Programmable timings

Table 9. Supported memory cuts

SIZE [MB]	BANK NUMBER	ROW LENGTH	COLUMN LENGTH
16 (2 MB x 8 bits)	2	11	9
16 (1 MB x 16 bits)	2	11	8
64 (8 MB x 8 bits)	4	12	9
64 (4 MB x 16 bits)	4	12	8
64 (2 MB x 32 bits)	4	11	8
128 (16 MB x 8 bits)	4	12	10
128 (8 MB x 16 bits)	4	12	9
128 (4 MB x 32 bits)	4	12	8
256 (32 MB x 8 bits)	4	13	10
256 (16 MB x 16 bits)	4	13	9
256 (8 MB x 32 bits)	4	13	8
512 (64 MB x 8 bits)	4	13	11
512 (32 MB x 16 bits)	4	13	10

Table 10. Multi-Port Memory Controller AHB port assignment

PORT	SIZE [bit]	PRIORITY	MASTER
0	32	Max	Bus Matrix
1	-		Reserved
2	32		eASIC™
3	32		USB 2.0 Device

Table 10. Multi-Port Memory Controller AHB port assignment (continued)

PORT	SIZE [bit]	PRIORITY	MASTER
4	32		USB 2.0 Host 1
5	32		USB 2.0 Host 2
6	32		Ethernet MAC
7	32	Min.	Main AHB System Bus

The table is compiled in decreasing order of priority.

The I/O interfaces accessible from off-chip are listed here:

Table 11. Multi-Port Memory Controller off-chip interfaces

SIGNAL	DIRECTION	SIZE [bit]	DESCRIPTION
MPMCDQS	Input	2	Data Strobe
MPMCDATA	Bidirectional	32	Read / write data
MPMCCLKOUT	Output	2	DRAM clock
nMPMCCLKOUT	Output	2	DRAM inverted clock
MPMCCKEOUT	Output	2	DRAM clock enable
MPMCDQMOUT	Output	4	Data mask
nMPMCRASOUT	Output	1	RAS (active low)
nMPMCCASOUT	Output	1	CAS (active low)
nMPMCWEOUT	Output	1	Write Enable (active low)
nMPMCDYCSOUT	Output	4	Chip Select (active low)
MPMCADDROUT	Output	15	Address
SSTL_VREF	Input	1	Voltage reference SSTL / CMOS mode: SSTL → 1.25 V CMOS → 0 V This pin is used both as logic state and as power supply.

12.2 MPMC DELAY LINES

As shown in *Figure 10*, there are 4 DLLsp.

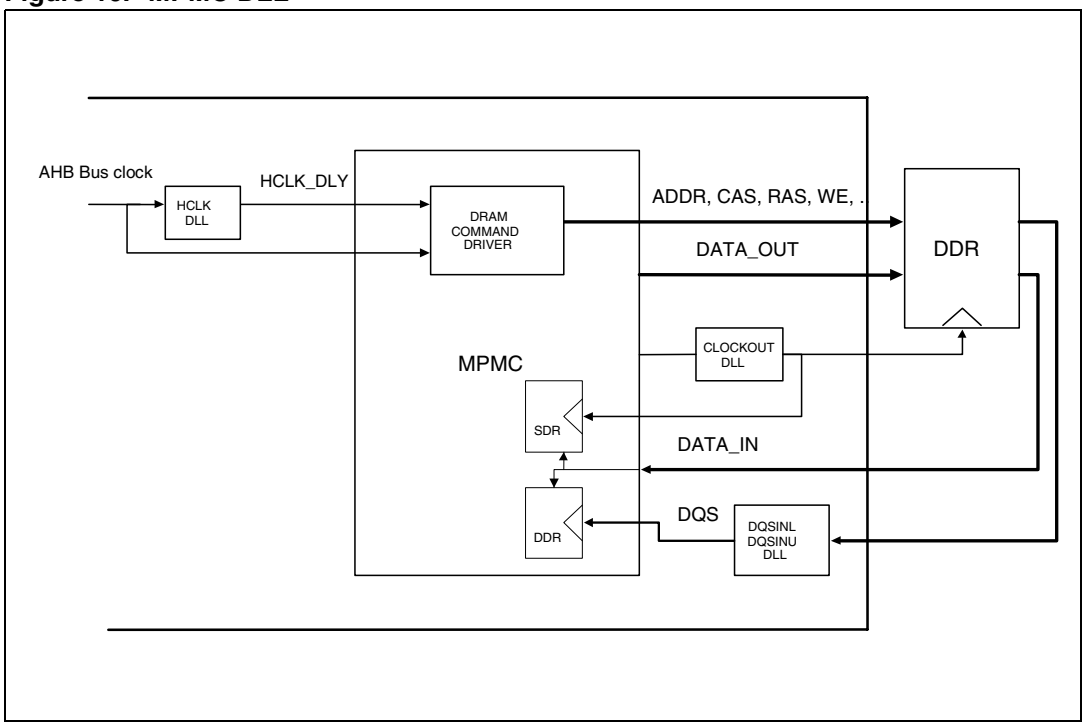
The CLOCKOUT delay line is used to tuner the clock driven from MPMC to external DRAM to match setup / hold constraints on external memory. This Delay Lines is used in the "clock delay methodology" suitable for SDRAM memory.

The AHB Bus Delay Lines delays the DRAM command signal (ADDR, CAS, RAS, ...) to capture easily read data from DRAM; this technique is called "command delay".

The DQSINL and DQSINH Delay Lines delay respectively the DQS0 (least 8 bit data strobe) and DQS1 (highest 8 bit of data strobe) signals coming from DDR.

Only the least 7 bits of these registers are significant because the Delay Lines programming parameter can vary from 0 to 127.

Figure 10. MPMC DLL



12.3 SSTLL PAD CONFIGURATION

The Stub Series-Terminated Logic (SSTL) interface standard is intended for high-speed memory interface applications and specifies switching characteristics such that operating frequencies up to 200 MHz are attainable.

The primary application for SSTL devices is to interface with SDRAMs.

In SPEAr Head200 SSTL pads are used for pins:

- MPMCDATA[15:0]
- MPMCADDROUT
- MPMCCLKOUT
- nMPMCCLKOUT

To enable the 2.5V SSTL mode the least significant bit has to be set to 0; when this bit is sets to 1 the 3.3V CMOS mode is enabled.
It is also possible to program the pad output impedance. Each pad has two configurable inputs to change output impedance: ZOUTPROGB and ZOUTPROGA.
Below the table to configure output impedance:

Table 12. Output impedance configuration

ZOUTPROGA	ZOUTPROGB	OUTPUT BUFFER IMPEDANCE
0	0	25Ω
0	1	35Ω
1	0	45Ω
1	1	55Ω

13 SPI memories

13.1 Overview

SPEAr Head200 supports the SPI memory devices Flash and EEPROM.

SPI controller, also called Serial Memory Interface (SMI), provides an AHB slave interface to SPI memories and allows CPU to use them as data storage or code execution.

Main features are

- SPI master type
- Up to 20 MHz clock speed in Standard Read mode and 50 MHz in Fast Read mode
- 4 chips select
- Up to 16 MBytes address space per bank
- Selectable 3 Byte addressing for Flash and 2 Byte addressing for EEPROM
- Programmable clock prescaler
- External memory boot mode capability
- 32, 16 or 8 bit AHB interface
- Interrupt request on write complete or software transfer complete

The compatible SPI memories are:

- STMicroelectronics M25Pxxx, M45Pxxx
- STMicroelectronics M95xxx except M95040, M95020 and M95010
- ATMEL AT25Fxx
- YMC Y25Fxx
- SST SST25LFxx

The I/O interfaces accessible from off-chip are listed here:

Figure 11. SPI Interfaces

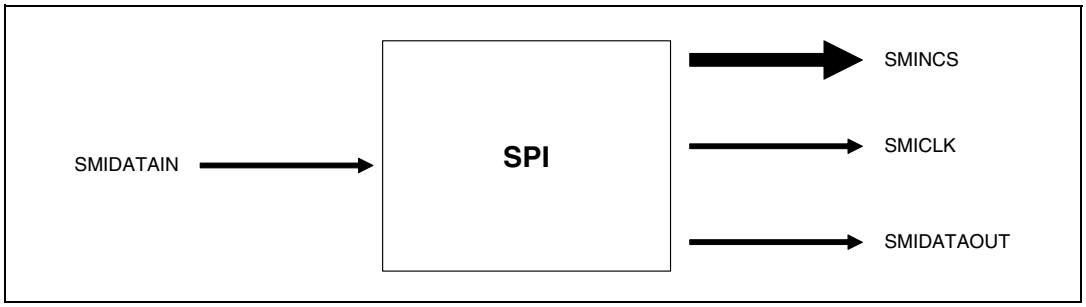


Table 13. SPI signal interfaces description

Signal	Direction	Size [bit]	Description
SMIDATAIN	Input	1	Memory output
SMIDATAOUT	Output	1	Memory input
SMICLK	Output	1	Memory clock
SMINCS	Output	4	Bankchip selects (active low)

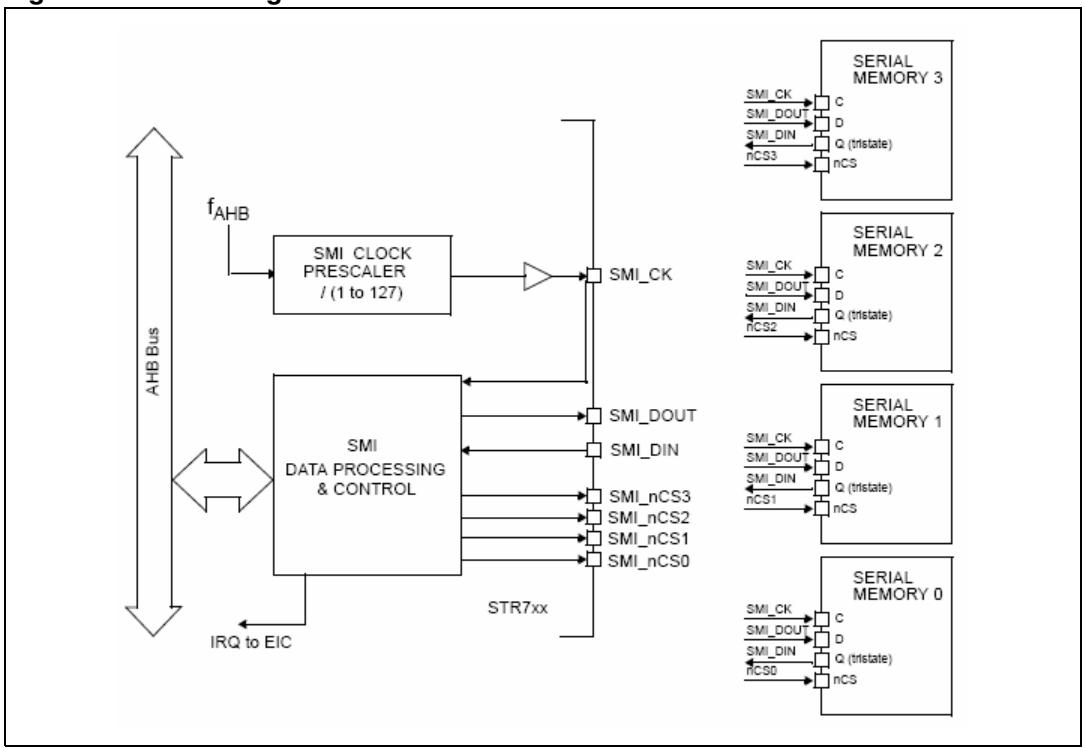
At power on the boot code is enabled from the static memory Bank0 by default; this has to be a Flash bank memory. Moreover, at power on, the memory clock signal is 19 MHz, the Release Deep Power-Down is 29 µs and the base address for external memories is 0.

13.2 SMI description

The main components of the SMI are two:

– The SMI CLOCK PRESCALER, which sets-up the memory clock
– The SMI DATA PROCESSING & CONTROL, which is the logic controlling the transfer of the data

Figure 12. Block diagram



13.2.1 Transfer rules

The following rules apply to the access from the AHB to the SPI Controller:

- Endianness is fixed to Little-Endian
- SPLIT / RETRY responses are not supported
- Bursts must not cross bank boundaries
- Size of data transfers for memories can be byte / half-word / word
- Size of data transfers for registers must be 32 bit wide
- Read Requests: all types of BURST are supported. Wrapping bursts take more time than incrementing bursts, as there is a break in the address increment
- Write Requests: wrapping bursts are not supported, and provoke an ERROR response on HRESP
- When BUSY transfer: the SPI Controller is held until busy is inactive

If instead of Flash memories are used EEPORMs, the address for a Read has to be ADDRESS + 1 while we want to read the one located at ADDRESS.

The communication protocol used is SPI in CPOL = 1 and CPHA = 1 mode.

The instructions supported are listed in Table 2. SMI Supported instructions.

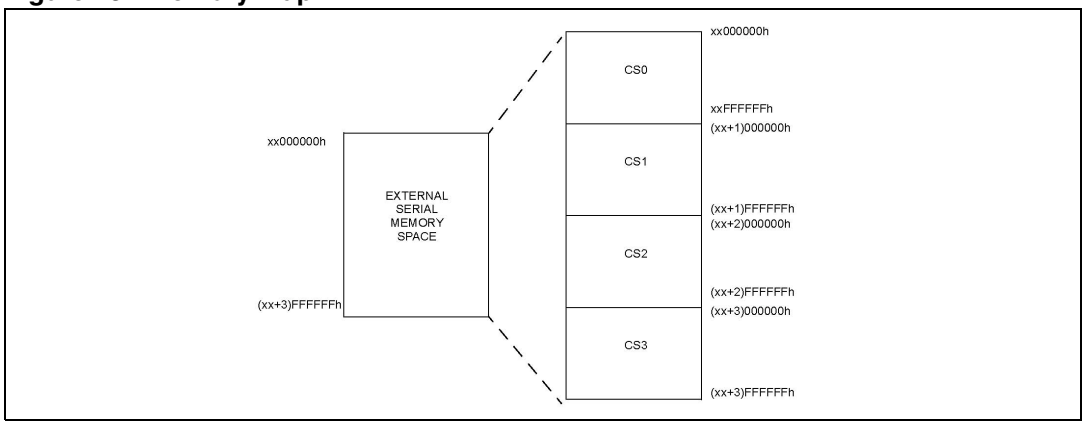
Table 14. SMI Supported instructions

OPCODE	DESCRIPTION
03	Read data bytes
0B	Read data high speed
05	Read status register
06	Write enable
02	Page program
AB	Release from deep power-down

13.2.2 Memory map

External memory is mapped in AHB address space as shown in *Figure 13*.

Figure 13. Memory map



where xx is 16, the address of SMI in the memory map is from 0x1600_0000 to 0x19F_FFFF

13.2.3 Operation mode

Two operation modes exist:

- Hardware mode is used to serve AHB read and write requests. This is the functional mode at reset.
- Software mode is used to serve no AHB requests., during normal working

In both cases SMI can work:

- or in Normal mode, up to a frequency of 20 MHz (19 Mhz at power on)
- either in Fast mode, in a range frequency between 20 MHz and 50 MHz

Hardware mode

At reset, the SMI operates in Hardware mode. In this mode, the transmit register and receive register must not be accessed. They are managed by the SMI State Machine and used to communicate with the external memory devices whenever an AHB master read or write to an address in external memory.

Software mode

In Software Mode, transmit register and receive register are accessible. Direct AHB transfers to/from external memories are not allowed.

Software mode is used to transfer any data or commands from the transmit register to external memory and to read data directly in the receive register. The transfer is started using a dedicated bit.

For example Software mode is used to erase Flash memory before writing. Erase cannot be managed in Hardware mode due to incompatibilities which exist between Flash devices from different vendors.

In Software mode, application code, being executed by the core, cannot be fetched from external memory. It must either reside in internal memory, or be previously loaded from external memory while the SMI is in Hardware mode.

13.2.4 Booting from external memory

SPEAr Head200 has an external boot from a Serial Flash at the Bank0 and the following command sequence is sent to it:

- Release from Deep Power-Down
- 29 µs delay
- Read of Status register
- Read of data bytes at memory start location

All other banks are disabled at reset and must be enabled by setting dedicated bits before they can be accessed.

14 Ethernet MAC 110

14.1 Overview

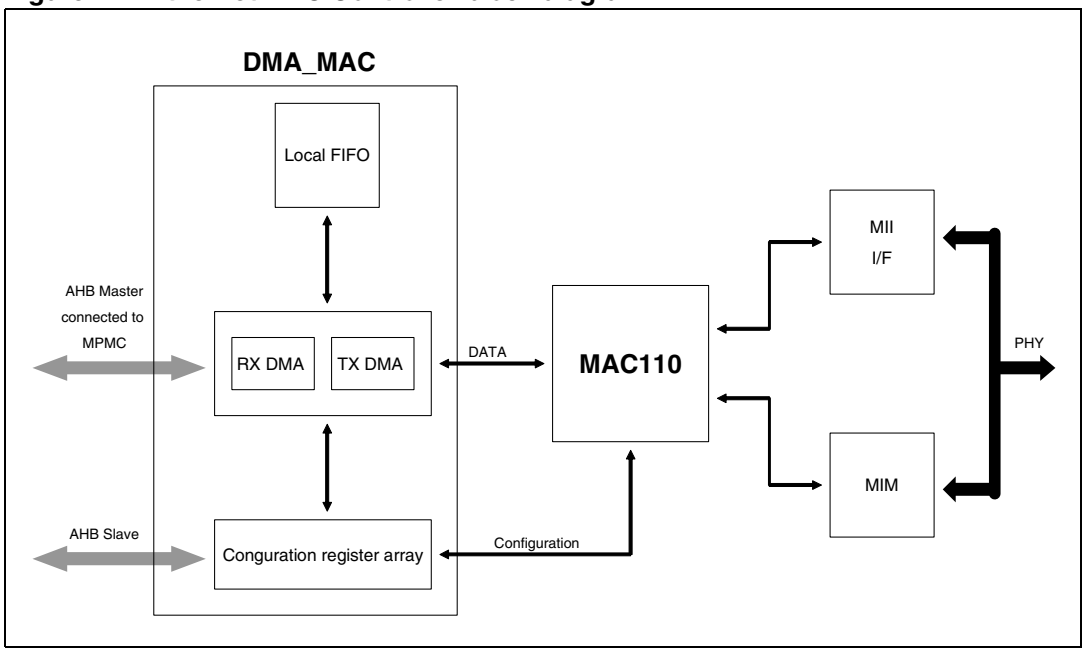
The Ethernet MAC controller is compliant with IEEE 802.3 specifications and provides the following features:

- Built-in DMA engine to manage Memory transfers
- Collision Detection in Half Duplex mode (CSMA/CD)
- Control Frames support in Full Duplex mode (IEEE 802.3)
- IEEE 802.3 Media Independent Interface (MII), Reduced Media Independent Interface (RMII) and General Purpose Serial Interface (GPSI)

The Ethernet MAC Controller is composed of two blocks:

- the DMA_MAC controller, which provides DMA facilities on top of the MAC110 block. It is able to support two types of operations:
  - write_type RX: data are moved from the MAC110 to a memory destination on the AMBA bus
  - read_type TX: data are moved from a memory source on the AMBA bus to the MAC110
- the MAC110 IP block: it implements the LAN CSMA/CD sublayer for the following families of systems: 10 Mb/s and 100 Mb/s of data rates for baseband and broadband systems

Figure 14. Ethernet MAC Controller block diagram



15 USB 2.0 Host

15.1 Overview

SPEAr Head has two fully independent USB 2.0 Hosts and each one is constituted with 2 main blocks:

- the USB2.0PHY that executes the serialization and the de-serialization and implements the transceiver for the USB line.
- the USB2.0 Host Controller (UHC). It is connected on AHB Bus and generates the commands for USB2.0PHY in UTMI+ interface.

15.1.1 USB2.0PHY

The USB2.0PHY is a hard macro included in SPEAr Head. It is designed using standard cells and custom cells. In this way has been possible to reach the max speed of USB: 480 Mbits/sec.

The macro is able to set his speed in LS / FS for USB 1.1 and in HS for USB 2.0.

15.1.2 UHC

The UHC is able to detect the USB speed configuration: USB 1.1 (LS / FS), USB 2.0 (HS) via UTMI+ interface.

When the speed is detected, the controller uses 2 sub-controllers: EHCI (Enhanced Host Controller Interface) for 2.0 configuration and OHCI (Open Host Controller Interface) for 1.1 configuration.

There is an AHB master and a slave for everyone of these controllers.

16 USB 2.0 Device

16.1 Overview

The USB 2.0 Device is constituted with 4 main blocks:

- USB2.0PHY that executes the serialization and the de-serialization and implements the transceiver for the USB line
- UDC, the USB 2.0 Device Controller. It is connected on AHB Bus and generates the commands for USB2.0PHY in UTMI+ interface
- a dedicated DMA macro to transfer data between the Device Controller and Multi-Port Memory Controller
- USB Plug Detect, which detects the connection of the device

16.1.1 USB2.0PHY

The USB2.0PHY is a hard macro that can reach the max speed HS of USB: 480 Mbits/Sec.

16.1.2 UDC

The UDC is able to detect the USB connection speed via UTMI+ interface.

There is an AHB master and two slaves.

The UDC contains 6 endpoints (0 control, 1 Bulk IN, 2 Bulk OUT, 3 ISO IN, 4 ISO OUT, 5 Interrupt IN)

The UDC contains 4 configurations.

16.1.3 DMA

Provided with a master interface on the AHB Bus, it manage data transfer between Device Controller CRSs and the FIFO embedded in the block.

16.1.4 USB plug detect

The Plug Detect detects when the Device is connected to an Host and is receiving the VBUS signal.

17 UART

UART provides a standard serial data communication with transmit and receive channels that can operate concurrently to handle a full-duplex operation.

Two internal FIFO for transmitted and received data, deep 16 and wide 8 bits, are present; these FIFO can be enabled or disabled through a register.

Interrupts are provided to control reception and transmission of serial data.

The clock for both transmit and receive channels is provided by an internal Baud-Rate generator that divides the AHB Bus clock by any divisor value from 1 to 255. The output clock frequency of baud generator is sixteen times the baud rate value.

The maximum speed achieved is 115 KBauds.

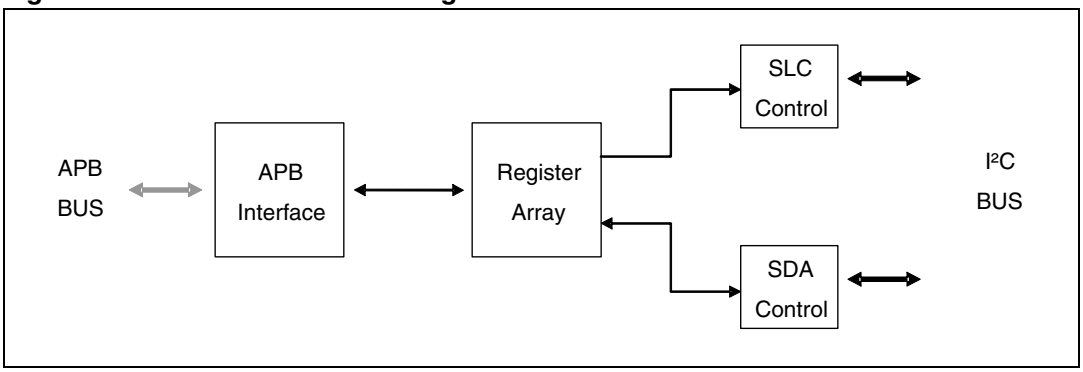
In SPEAr Head200 there are 3 UART's, APB Bus slaves.

18 I²C controller

18.1 Overview

The controller serves as an interface between the APB Bus and the serial I²C bus. It provides master functions, and controls all I²C bus-specific sequencing, protocol, arbitration and timing. Supported Standard (100 KHz) and Fast (400 KHz) I²C mode.

Figure 15. I²C Controller block diagram



Main features are:

- Parallel-bus APB / I2C protocol converter
- Standard I²C mode (100 KHz) / Fast I²C mode (400 KHz)
- Master interface (only).
- Detection of bus errors during transfers
- Control of all I²C bus-specific sequencing, protocol, arbitration and timing

In addition to receiving and transmitting data, this interface converts it from serial to parallel format and vice versa, using either an interrupt or a polled handshake. The interrupts can be enabled or disabled by software.

The interface is connected to the I²C bus by a data pin (SDA) and by a clock pin (SCL). SDA signal is synchronized by SCL signal.

18.2 Operating mode

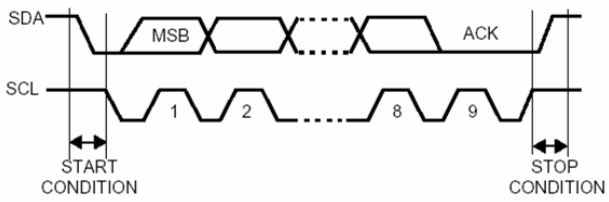
Communication flow

In Master mode, it initiates a data transfer and generates the clock signal. A serial data transfer always begins with a start condition and ends with a stop condition. Both start and stop conditions are generated by software.

The first byte following the start condition is the address byte; it is always transmitted in Master mode.

A 9th clock pulse follows the 8 clock cycles of a byte transfer, during which the receiver must send an acknowledge bit to the transmitter.

Figure 16. I²C timing



Acknowledge may be enabled and disabled by software.

The I²C interface address and / or general call address can be selected by software.

The speed of the I²C interface may be selected between Standard (0 - 100 KHz) and Fast (100 - 400 KHz).

SDA / SCL line control

Transmitter mode: the interface holds the clock line low before transmission to wait for the microcontroller to write the byte in the Data Register.

Receiver mode: the interface holds the clock line low after reception to wait for the microcontroller to read the byte in the Data Register.

The SCL frequency (FSCL) is controlled by a programmable clock divider which depends on the I²C bus mode.

When the I²C cell is enabled, the SDA and SCL ports must be configured as floating open-drain output or floating input. In this case, the value of the external pull-up resistance used depends on the application.

18.3 I²C functional description

Master Mode

The I²C clock is generated by the master peripheral.

The interface operates in Master mode through the generation of the Start condition: Start bit set to 1 in the control register and I²C not busy (Busy flag set to 0).

Once the Start condition is sent, if interrupts are enabled, an Event Flag bit and a Start bit are set by hardware. Then the master waits for a read of the register used to observe bus activity (SR1 register) followed by a write in the data register DR with the Slave address byte, holding the SCL line low (see *Figure 17* Transfer sequencing EV5). Then the slave address byte is sent to the SDA line via the internal shift register.

After completion of these transfers, the Event Flag bit is set by hardware with interrupt generation. Then the master waits for a read of the SR1 register followed by a write in the control register CR (for example set the Peripheral Enable bit), holding the SCL line low (see *Figure 17* Transfer sequencing EV6).

Next the Master must enter Receiver or Transmitter mode.

Master Receiver

Following the address transmission and after SR1 and CR registers have been accessed, the Master receives bytes from the SDA line into the DR register via the internal shift register. After each byte the interface generates in sequence:

1) Acknowledge pulse if the acknowledge bit ACK in the control register is set

2) Event Flag and the Byte Transfer Finish bits are set by hardware with an interrupt.

Then the interface waits for a read of the SR1 register followed by a read of the DR register, holding the SCL line low (see *Figure 17* Transfer sequencing EV7).

To close the communication: before reading the last byte from the DR register, set the Stop bit to generate the Stop condition.

In order to generate the non-acknowledge pulse after the last received data byte, the ACK bit must be cleared just before reading the second last data byte.

Master Transmitter

Following the address transmission and after SR1 register has been read, the Master sends bytes from the DR register to the SDA line via the internal shift register.

The master waits for a read of the SR1 register followed by a write in the DR register, holding the SCL line low (see *Figure 17* Transfer sequencing EV8).

When the acknowledge bit is received, the interface sets Event Flag and the Byte Transfer Finish bits with an interrupt.

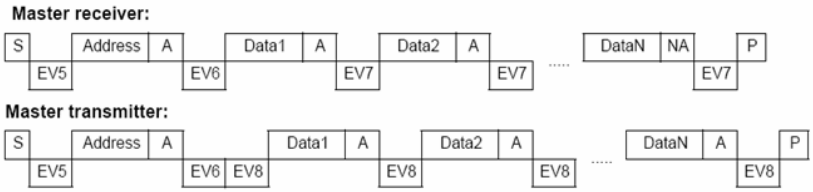
To close the communication: after writing the last byte to the DR register, set the Stop bit to generate the Stop condition.

Error Cases

- BERR: Detection of a Stop or a Start condition during a byte transfer. In this case, the Event Flag and BERR bits are set by hardware with an interrupt.
- AF: Detection of a non-acknowledge bit. In this case, the EVF and AF bits are set by hardware with an interrupt. To resume, set the Start or Stop bit.

In all these cases, the SCL line is not held low; however, the SDA line can remain low due to possible 0 bits transmitted last. It is then necessary to release both lines by software.

Figure 17. Transfer sequencing



Legend:

S=Start, P=Stop, A=Acknowledge, NA=Non-acknowledge

EVx=Event (with interrupt if ITE=1)

EV5: EVF=1, SB=1, cleared by reading SR1 register followed by writing DR register.

EV6: EVF=1, cleared by reading SR1 register followed by writing CR register (for example PE=1).

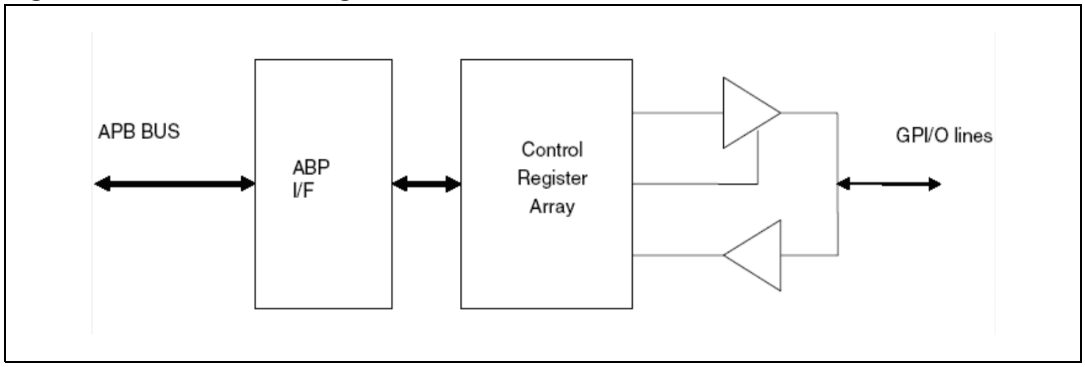
EV7: EVF=1, BTF=1, cleared by reading SR1 register followed by reading DR register.

EV8: EVF=1, BTF=1, cleared by reading SR1 register followed by writing DR register.

19 General purpose I/Os

The GPIO block consists of 6 General Purpose IOs which act as buffers between the IO pads and the processor core: data is stored in the GPIO block and can be written to and read from by the processor via the APB Bus.

Figure 18. GPIO block diagram



20 ADC

20.1 Overview

The ADC integrated in SPEAr Head200 is the ST-ADC8MUX16 and it is connected to APB Bus. It is a successive approximation ADC and its main features are:

- 8 bit resolutions
- 230 Ksps
- 16 analog input channels (0 - 3.3 V)
- INL ± 1 LSB
- DNL ± 0.5 LSB
- Programmable conversion speed - minimum conversion time 4.3 µs

For any ADC input channel the number of collected samples for the average can be 1 or 2's power, up to 128.

Positive and negative reference voltages are supply by dedicated pins:

- positive → VREFP_adc pin
- negative → VREFN_adc pin

20.2 Functional description

Conversion starts when enabling bit is set to 1; as it finishes, an interrupt signal is generated (a bit of conversion ready is set to 1). At this point the reading of the data could begin and when it finishes, conversion ready and the enabling bits becomes 0.

The signals accessible off-chip are listed in *Table 15* External pins of ADC macro.

Table 15. External pins of ADC macro

SIGNAL	DIRECTION	DESCRIPTION
AID[15:0]	Input	Analog channels

21 Real time clock

The Real Time Clock block implements 3 functions:

- time-of-day clock in 24 hour mode
- calendar
- alarm

Time and calendar value are stored in binary code decimal format.

The RTC provides also a self-isolation mode, which allows it working even if power isn't supplied to the rest of the device.

It is an APB Bus slave.

22 Watchdog timer

The WdT is based on a programmable 8 bit counter and generates a hot reset (single pulse) when it overflows. This reset will restart the ARM but the code will not be downloaded again. The timer should be cleared by the software before it overflows.

The counter is clocked by a slow signal coming from a 17 bit prescaler clocked by the APB clock. So that, as APB Bus has a frequency of 133 MHz, the elapsing time is 0.25 second.

The WdT is an APB Slave device.

23 General purpose timers

SPEAr Head200 has 4 GPTs, connected as APB Bus slaves.

A GPT is constituted by 2 channels and each one consists of a programmable 16 bit counter and a dedicated 8 bit timer clock prescaler. The programmable 8 bit prescaler unit performs a clock division by 1, 2, 4, 8, 16, 32, 64, 128, and 256, allowing a frequency range from 3.96 Hz to 48 MHz.

Two modes of operation are available for each GPT:

- Auto Reload Mode. When the timer is enabled, the counter is cleared and starts incrementing. When it reaches the compare register value, an interrupt source is activated, the counter first is automatically cleared and then restarts incrementing. The process is repeated until the timer is disabled.
- Single Shot Mode. When the timer is enabled, the counter is cleared and starts incrementing. When it reaches the compare register value, an interrupt source is activated, the counter stopped and the timer disabled.

The current timer counter value could be read from a register.

24 Customizable logic

24.1 Overview

The Customizable Logic consists of an embedded macro where it is possible embedding a custom project by mapping up to 200K equivalent ASIC gates (corresponding at 16K LUT).

The logic is interfaced with the rest of the system so that can be implemented:

- AHB sub-systems with masters and slaves (via 1 AHB full master, 1 AHB full slave, 1 AHB master lite, 2 AHB slave ports)
- AHB master lite connected to DRAM controller
- AHB memories (via AHB slave ports) implemented by configuring the logic cells as SRAM elements.
- I/O protocol handlers (via the 112 dedicated GPIO connections)
- 8 interrupt channels
- 8 DMA requests
- 4 independent SRAM data channels (via dedicated connection to on-chip 16 KByte SRAM)

All of the above configuration scenarios can be mixed together in the same user-defined logic.

24.2 Custom project development

There are 2 ways to develop a project to embed in SPEAr Head200: through SPEAr Head behavioral model or through external FPGA.

24.2.1 SPEAr Head behavioral model

In the first case ST provides behavioral model of the fixed architecture allowing the final user to verify custom logic. Verification procedure is the same as a standard ASIC flow

24.2.2 External FPGA

The custom project to design in the customizable logic can be implemented on an external FPGA, which emulates eASIC™ logic cells. The purpose of this characteristic is allowing the user to develop his project both under real-time constraints and compliant to eASIC™ MacroCell features.

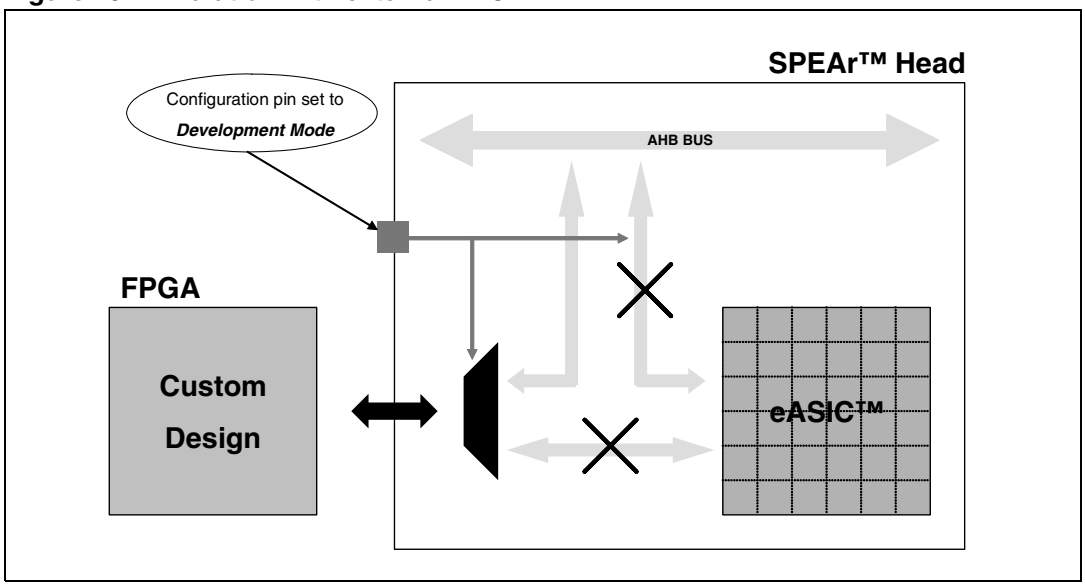
This mode is enabled by using the GPIO interface, which is internally configured to support full-master and full-slave AHB ports. *Figure 19* Emulation with external FPGA highlights the described behavior. In order to enable the "development mode", the configuration pin has to be set to state logic 1. After this configuration the logic implemented in the external FPGA

- can completely interact with the following scenarios:
  - AHB sub-systems with masters and slaves connected to the main system bus (full masters and full slaves peripherals)
  - I/O protocol handlers (via 112 dedicated FPGA I/Os)
  - 4 interrupt channels
  - 4 DMA requests

All the above scenarios can be mixed in the same FPGA configuration

- can be tested in order to verify the accordance with eASIC™ MacroCell features, by running the ARM926EJ-S software debugger on a PC connected to SPEAr Head200. Once this test has been completed successfully, then the user-defined logic is ready to be moved without any additional changes within the on-chip eASIC™ configurable logic.

Figure 19. Emulation with external FPGA



24.3 Customization process

The customization process requires two separate steps, executed at different times:

1. Programming layer fabrication (single VIA-mask).
2. Bitstream download.

The step 1 defines the interconnection between the customizable logic cells and is executed at fabrication level on top of the silicon wafers stored in the fab.

The step 2 defines the logic function for each customizable logic cell and is executed after that the system has been powered up by dedicated software routines running on the ARM926 microprocessor.

Both Bitstream and VIA-mask realize the user-defined customization for the entire device.

The eASIC™ mapping flow starts from the RTL description of the user-defined customization, with the purpose to generate the VIA-mask and configuration Bitstream.

24.4 Power on sequence

Once the system is powered-on, the eASIC™ logic has to be properly configured before its usage. In order to accomplish this task, two main operations have to be performed (both using dedicated software routines running on the ARM9 microprocessor):

1. Bitstream download
2. Startup of connection between the eASIC™ MacroCell and the rest of the device

Both steeps are driven by the a control register programmable via APB bus.

24.4.1 Bitstream download

The bitstream download operation is responsible for the eASIC™ logic initialization, since each configurable cell of the customizable logic is loaded with a data stream that represents the mapped logic function. Each operation of this download is performed by a dedicated software routine that read and writes data across the Control register reserved bits.

The bitstream is a 32 KByte data that is stored in the external non-volatile memory of the SPEAr™ device.

24.4.2 Connection startup

Once the eASIC™ logic is up and running due to the Bitstream initialization, next steep is its reset, in order to allow connections to the other IPs of the chip. The reset routine is activated by programming the Control register.

Last steep is the enabling of needed connections, by setting the Status register.

24.4.3 Programming interface

In order to achieve the Bitstream download and the reset routine, a dedicate logic has been embedded in the SPEAr Head200: the Programming Interface, which also includes the Control register.

25 Electrical characteristics

25.1 Absolute maximum ratings

This product contains devices to protect the inputs against damage due to high static voltages; however it is advisable to take normal precaution to avoid application of any voltage higher than the specified maximum rated voltages.

Table 16. Absolute maximum rating values

Symbol	Parameter	Value	Unit
VDD core	Supply voltage core	2.1	V
VDD I/O	Supply voltage I/O	6.4	V
VDD PLL	Supply voltage PLL	6.4	V
VDD SDR	Supply voltage SDRAM	6.4	V
VDD DDR	Supply voltage DDR	5.4	V
VDD RTC	Supply voltage RTC	2.1	V
Vi TTL	Input voltage TTL (3.3 and 5 V tolerant)	6.4	V
Vi SRAM	Input voltage SDRAM	6.4	V
Vi DDR	Input voltage DDR	5.4	V
Vi USBds	Input voltage USB (Host and Device) data signal interfaces	6.4	V
Vi USBrr	Input voltage USB reference resistor	6.4	V
Vi AN	Analog input voltage ADC	6.4	V
Tj	Junction temperature	-40 to 125	°C
Tstg	Storage temperature	-55 to 150	°C
VHBM	This device is compliant with Human Body Model JEDEC spec JESD22-A114C Class 2		
VCDM	This device is compliant with Charge Device Model JEDEC spec JESD22-C101C Class 2		

The average chip-junction temperature, Tj, can be calculated using the following equation:

$$T_j = T_A + (P_D \cdot \Theta_{JA})$$

where:

- T_A is the ambient temperature in °C
- Θ_{JA} is the package Junction-to-Ambient thermal resistance, which is 34 °C/W
- $P_D = P_{INT} + P_{PORT}$
  - P_{INT} is the chip internal power
  - P_{PORT} is the power dissipation on Input and Output pins; user determined

If P_{PORT} is neglected, an approximate relationship between P_D is:

$$P_D = K / (Tj + 273\ °C)$$

And, solving first equations:

$$K = P_D \cdot (T_A + 273\ °C) + \Theta_{JA} \times P_D^2$$

K is a constant for the particular, which can be determined through last equation by measuring P_D at equilibrium, for a know T_A

Using this value of K, the value of P_D and T_J can be obtained by solving first and second equation, iteratively for any value of T_A.

25.2 DC electrical characteristics

25.2.1 Supply voltage specifications

The recommended operating conditions are listed in the following table:

Table 17. Recommended operating conditions

Symbol	Description	Min.	Typ.	Max.	Unit
VDD core	Supply voltage core	1.14	1.2	1.26	V
VDD I/O	Supply voltage I/O	3	3.3	3.6	V
VDD PLL	Supply voltage PLL	3	3.3	3.6	V
VDD SDR	Supply voltage SDRAM	3	3.3	3.6	V
VDD DDR	Supply voltage DDR	2.3	2.5	2.7	V
VDD RTC	Supply voltage RTC	1.08	1.2	1.32	V
Top	Operating temperature	-40		85	°C

25.2.2 I/O voltage specifications

25.2.2.1 LVTTL I/O (compliant with EIA/JEDEC standard JESD8-B)

For LVTTL (3.3 V capable) pins, the allowed I/O voltages are:

Table 18. Low Voltage TTL DC input specification (3 < VDD < 3.6)

Symbol	Parameter	Test Condition	Min.	Max.	Unit
Vil	Low level input voltage			0.8	V
Vih	High level input voltage		2		V
Vhyst	Schmitt trigger hysteresis		0.495	0.620	V

Table 19. Low Voltage TTL DC output specification (3 < VDD < 3.6)

Symbol	Parameter	Test Condition	Min.	Max.	Unit
Vol	Low level output voltage	Iol = X mA *		0.15	V
Voh	High level output voltage	Ioh = X mA *	VDD - 0.15		V

* X is the source / sink current under worst case conditions and it is reflected in the name of the I/O cell according to the drive capability.

Table 20. Pull-up and Pull-down characteristics

Symbol	Parameter	Test Condition	Min.	Typ.	Max.	Unit
Ipu	Pull-up current	Vi = 0 V	40	60	110	µA
Ipd	Pull-down current	Vi = VDD	30	60	133	µA
Rup	Equivalent Pull-up resistance	Vi = 0 V	32	50	75	KOhm
Rpd	Equivalent Pull-down resistance	Vi = VDD	27	50	100	KOhm

25.2.2.2 LVCMOS/SSTL I/O

If the I/Os are set as LVCMOS (for SDRAM memories), the DC electrical characteristics are the following:

Table 21. LVCMOS DC input specification (3 < VDD < 3.6)

Symbol	Parameter	Test Condition	Min.	Max.	Unit
Vil	Low level input voltage	Vout ≥ Voh (min) or Vout ≤ Vol (max)	-0.3	0.8	V
Vih	High level input voltage		2	VDD+0.3	V
Iin	Input Current	Vin = 0 or Vin =VDD		±5	µA

Table 22. LVCMOS DC output specification (3 < VDD < 3.6)

Symbol	Parameter	Test Condition	Min.	Max.	Unit
Vol	Low level output voltage	VDD = min, Iol = 100 µA		0.2	V
Voh	High level output voltage	VDD = min, Ioh = -100 µA	VDD - 0.2		V

Instead when they are set as (for DDR memories), refer to following tables:

Table 23. DC input specification of bidirectional SSTL pins (2.3 < VDD DDR < 2.7)

Symbol	Parameter	Min.	Max.	Unit
Vil	Low level input voltage	-0.3	SSTL_VREF – 0.15	V
Vih	High level input voltage	SSTL_VREF + 0.15	VDD DDR – 0.15	V

Table 24. DC input specification of bidirectional differential SSTL pins (2.3 < VDD DDR < 2.7)

Symbol	Parameter	Min.	Max.	Unit
Vin	DC input signal voltage	-0.3	VDD DDR + 0.3	V
Vswing	DC differential input voltage	0.36	VDD DDR + 0.6	V

26 Package information

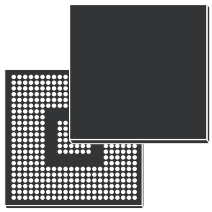
In order to meet environmental requirements, ST offers these devices in ECOPACK® packages. These packages have a Lead-free second level interconnect. The category of second Level Interconnect is marked on the package and on the inner box label, in compliance with JEDEC Standard JESD97. The maximum ratings related to soldering conditions are also marked on the inner box label. ECOPACK is an ST trademark.

ECOPACK specifications are available at: www.st.com.

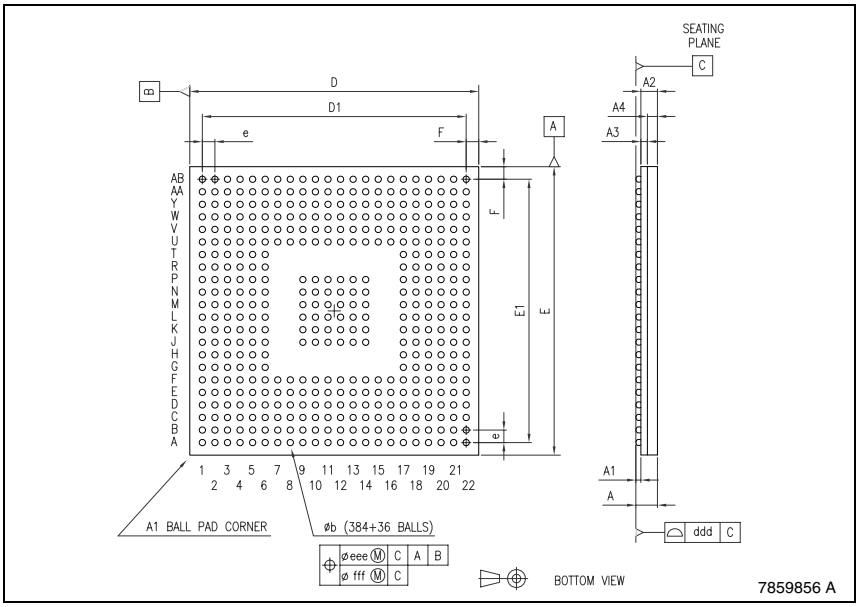
Figure 20. PBGA420 Mechanical Data & Package Dimensions

DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			1.81			0.0713
A1	0.27			0.0106		
A2		1.305			0.0514	
A3		0.52			0.0205	
A4		0.785			0.0309	
b	0.45	0.50	0.55	0.0177	0.0197	0.0217
D	22.80	23.00	23.20	0.8976	0.9055	0.9134
D1		21.00			0.8268	
E	22.80	23.00	23.20	0.8976	0.9055	0.9134
E1		21.00			0.8268	
e		1.00			0.0394	
F		1.00			0.0394	
ddd			0.20			0.0079
eee			0.25			0.0098
fff			0.10			0.0039

OUTLINE AND MECHANICAL DATA



PBGA420 (23x23x1.81mm)
Ball **G**rid **A**rray **P**ackage



27 Revision history

Table 25. Document revision history

Date	Revision	Changes
17-Oct-2005	1	Initial release.
01-Dec-2005	2	Changed the Part Number from SPEAR-09-H020 to SPEAR-09-H022. Modified/added some dates in the Chapter 7.
29-Mar-2006	3	Changed Block diagram & Pin out.
03-Aug-2006	4	Modified *Table 2: Pins belonging to POWER group*.
28-Sep-2006	5	Modified *Chapter 2: Product Overview*: point 9. Modified *Chapter 3.6.2: USB 2.0 device*. Modified *Table 2: Pins belonging to POWER group*: ball “AB2”. Modified *Section 16.1.2: UDC*. Modified *Table 16: Absolute maximum rating values*.

Please Read Carefully:

Information in this document is provided solely in connection with ST products. STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, modifications or improvements, to this document, and the products and services described herein at any time, without notice.

All ST products are sold pursuant to ST’s terms and conditions of sale.

Purchasers are solely responsible for the choice, selection and use of the ST products and services described herein, and ST assumes no liability whatsoever relating to the choice, selection or use of the ST products and services described herein.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted under this document. If any part of this document refers to any third party products or services it shall not be deemed a license grant by ST for the use of such third party products or services, or any intellectual property contained therein or considered as a warranty covering the use in any manner whatsoever of such third party products or services or any intellectual property contained therein.

UNLESS OTHERWISE SET FORTH IN ST’S TERMS AND CONDITIONS OF SALE ST DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY WITH RESPECT TO THE USE AND/OR SALE OF ST PRODUCTS INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (AND THEIR EQUIVALENTS UNDER THE LAWS OF ANY JURISDICTION), OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS EXPRESSLY APPROVED IN WRITING BY AN AUTHORIZED ST REPRESENTATIVE, ST PRODUCTS ARE NOT RECOMMENDED, AUTHORIZED OR WARRANTED FOR USE IN MILITARY, AIR CRAFT, SPACE, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS, NOR IN PRODUCTS OR SYSTEMS WHERE FAILURE OR MALFUNCTION MAY RESULT IN PERSONAL INJURY, DEATH, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. ST PRODUCTS WHICH ARE NOT SPECIFIED AS "AUTOMOTIVE GRADE" MAY ONLY BE USED IN AUTOMOTIVE APPLICATIONS AT USER’S OWN RISK.

Resale of ST products with provisions different from the statements and/or technical features set forth in this document shall immediately void any warranty granted by ST for the ST product or service described herein and shall not create or extend in any manner whatsoever, any liability of ST.

ST and the ST logo are trademarks or registered trademarks of ST in various countries.

Information in this document supersedes and replaces all information previously supplied.

The ST logo is a registered trademark of STMicroelectronics. All other names are the property of their respective owners.

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com