# SPEAr320

Embedded MPU with ARM926 core
for factory automation and consumer applications

Preliminary data

Features

- ARM926EJ-S 333 MHz core
- High-performance 8-channel DMA
- Dynamic power-saving features
- Configurable peripheral functions multiplexed on 102 shared I/Os
- Memory:
  - 32 KB ROM and 8 KB internal SRAM
  - LPDDR-333/DDR2-666 external memory interface
  - SDIO/MMC card interface
  - Serial SPI Flash interface
  - Flexible static memory controller (FSMC) up to 16-bit data bus width, supporting NAND Flash
  - External memory interface (EMI) up to 16-bit data bus width, supporting NOR Flash and FPGAs
- Security
  - C3 Cryptographic accelerator
- Connectivity
  - 2 x USB 2.0 Host
  - 1 x USB 2.0 Device
  - 2 x fast Ethernet MII/SMII ports
  - 2 x CAN interface
  - 3 x SPI
  - 2 x I^2C
  - 1 x I^2S
  - 1 x fast IrDA interface
  - 3 x UART interface
  - 1 x standard parallel device port
- Peripherals supported
  - TFT/STN LCD controller (resolution up to 1024 x 768 and up to 24 bpp)
  - Touchscreen support
- Miscellaneous functions

LFBGA289 (15 x 15 x 1.7 mm)
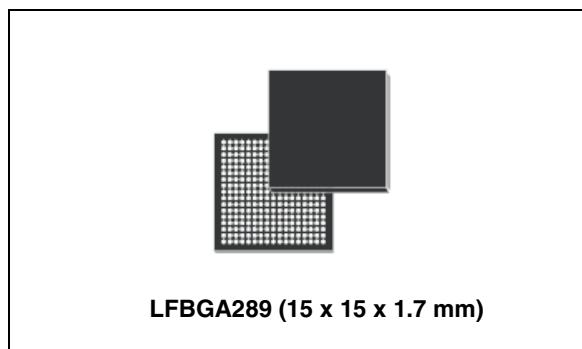
  - Integrated real time clock, watchdog, and system controller
  - 8-channel 10-bit ADC, 1 Msps
  - 4 x PWM timers
  - JPEG CODEC accelerator
  - 6 x 16-bit general purpose timers with and programmable prescaler, 4 capture inputs
  - Up to 102 GPIOs with interrupt capability

Applications

The SPEAr320 embedded MPU is configurable for a range industrial and consumer applications such as:

- Programmable logic controllers
- Factory automation
- Printers

Table 1.   Device summary

Order code	Temp range, °C	Package	Packing
SPEAR320-2	-40 to 85	LFBGA289 (15 x 15 mm, pitch 0.8 mm)	Tray

This is preliminary information on a new product now in development or undergoing evaluation. Details are subject to change without notice.

www.st.com

Contents

www.DataSheet4U.com

1 Description

The SPEAr320 is a member of the SPEAr family of embedded MPUs, optimized for industrial automation and consumer applications. It is based on the powerful ARM926EJ-S processor (up to 333 MHz), widely used in applications where high computation performance is required.

In addition, SPEAr320 has an MMU that allows virtual memory management -- making the system compliant with Linux operating system. It also offers 16 KB of data cache, 16 KB of instruction cache, JTAG and ETM (Embedded Trace Macrocell™) for debug operations.

A full set of peripherals allows the system to be used in many applications, some typical applications being factory automation, printer and consumer applications.

Figure 1. Functional block diagram
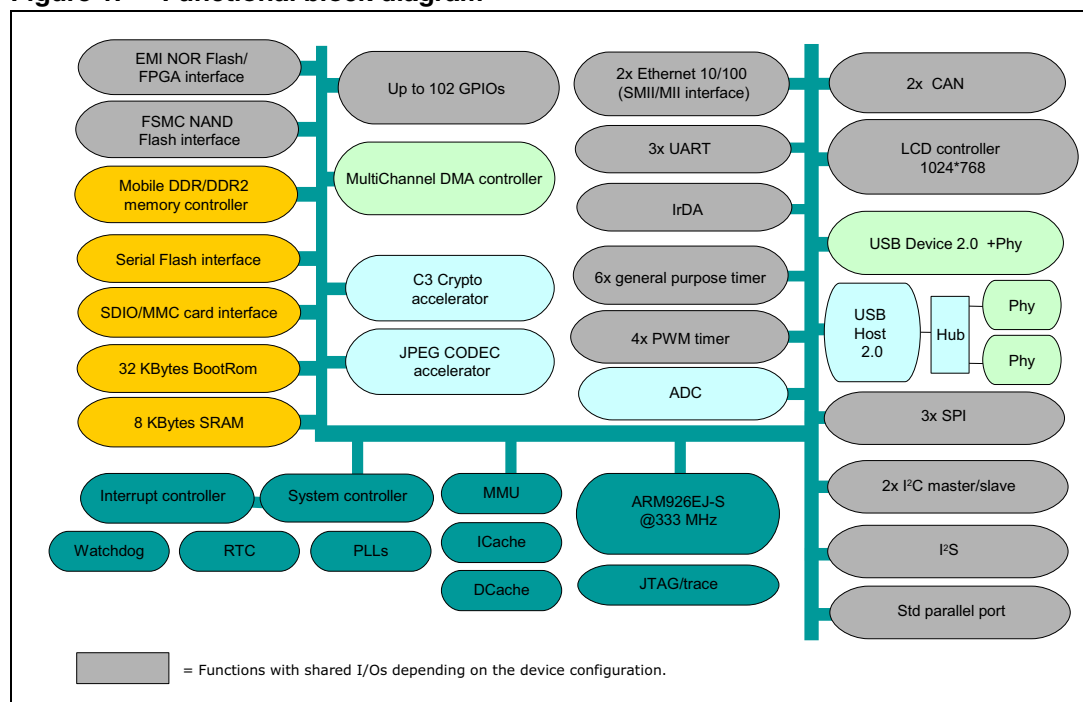
1.1 Main features

- ARM926EJ-S 32-bit RISC CPU, up to 333 MHz
  - 16 Kbytes of instruction cache, 16 Kbytes of data cache
  - 3 instruction sets: 32-bit for high performance, 16-bit (Thumb) for efficient code density, byte Java mode (Jazelle™) for direct execution of Java code.
  - Tightly Coupled Memory
  - AMBA bus interface
- 32-KByte on-chip BootRom
- 8-KByte on-chip SRAM
- External DRAM memory interface:
  - 8/16-bit (mobile DDR @ 166 MHz)
  - 8/16-bit (DDR2 @ 333 MHz)
- Serial memory interface
- SDIO interface supporting SPI, SD1, SFD4 and SD8 modes
- 8/16-bits NAND Flash controller (FSMC)
- External memory interface (EMI) for connecting NOR Flash or FPGAs
- Boot capability from NAND Flash, serial/parallel NOR Flash
- Boot and field upgrade capability from USB
- High performance 8-channel DMA controller
- 2x Ethernet MAC 10/100 Mbps with MII/SMII PHY interface
- Two USB2.0 host (high-full-low speed) with integrated PHY transceiver
- One USB2.0 device (high-full speed) with integrated PHY transceiver
- 2x CAN 2.0 interfaces
- Up to 102 GPIOs with interrupt capability
- Up to 4 PWM outputs
- 3x SPI master/slave (supporting Motorola, Texas instruments, National semiconductor protocols) up to 41.5 Mbps

- Standard Parallel Port (SPP device implementation)
- I^2S input-output for voice or modem interfaces
- 2x I^2C master/slave interface (slow- fast-high speed, up to 1.2Mb/s)
- 3x UART: UART1 with hardware flow control (up to 460.8 kbaud), UART2 and UART3 with software flow control (baud rate > 6 Mbps)
- ADC 10-bit, 1 Msps 8 inputs/1-bit DAC
- JPEG CODEC accelerator 1 clock/pixel
- Color LCD interface (up to 1024X768, 24-bits CLCD controller, TFT and STN panels)
- Touchscreen support
- C3 Crypto accelerator (DES/3DES/AES/SHA1)
- Advanced power saving features
  - Normal, Slow, Doze and Sleep modes CPU clock with software-programmable frequency
  - Enhanced dynamic power-domain management
  - Clock gating functionality

- – Low frequency operating mode
- – Automatic power saving controlled from application activity demands
- ● Vectored interrupt controller
- ● System and peripheral controller
  - – 3 pairs of 16-bit general purpose timers with programmable prescaler
  - – RTC with separate power supply allowing battery connection
  - – Watchdog timer
  - – Miscellaneous registers array for embedded MPU configuration
- ● Programmable PLL for CPU and system clocks
- ● JTAG IEEE 1149.1 boundary scan
- ● ETM functionality multiplexed on primary pins
- ● Supply voltages
  - – 1.2 V core, 1.8 V/2.5 V DDR, 2.5 V PLLs and 3.3 V I/Os
- ● Operating temperature: - 40 to 85 °C
- ● LFBGA289 (15 x 15 mm, pitch 0.8 mm)

2 Architecture overview

The SPEAr320 internal architecture is based on several shared subsystem logic blocks interconnected through a multilayer interconnection matrix.

The switch matrix structure allows different subsystem dataflow to be executed in parallel improving the core platform efficiency.

High performance master agents are directly interconnected with the memory controller reducing the memory access latency. The overall memory bandwidth assigned to each master port can be programmed and optimized through an internal efficient weighted round-robin arbitration mechanism.

Figure 2. Typical system architecture using SPEAr320
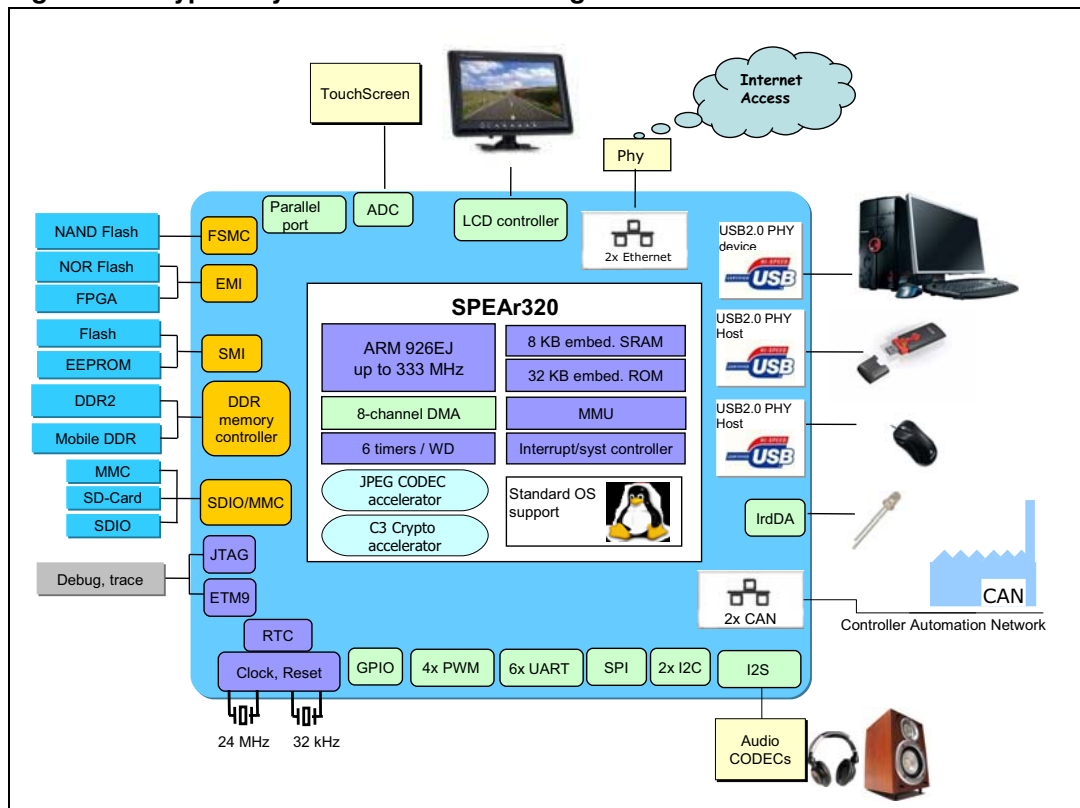
2.1 CPU ARM 926EJ-S

The core of the SPEAr320 is an ARM926EJ reduced instruction set computer (RISC) processor.

It 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.

The ARM CPU and is clocked at a frequency up to 333 MHz. It has a 16-Kbyte instruction cache, a 16-Kbyte data cache, and features a memory management unit (MMU) which makes it fully compliant with Linux and VxWorks operating systems.

It also includes an embedded trace module (ETM Medium+) for real-time CPU activity tracing and debugging. It supports 4-bit and 8-bit normal trace mode and 4-bit demultiplexed trace mode, with normal or half-rate clock.

2.2　Embedded memory units

- 32 Kbytes of BootROM
- 8 Kbytes of SRAM

2.3　Mobile DDR/DDR2 memory controller

SPEAr320 integrates a high performance multi-channel memory controller able to support low power Mobile DDR and DDR2 double data rate memory devices. The multi-port architecture ensures memory is shared efficiently among different high-bandwidth client modules.

It has 6 internal ports. One of them is reserved for register access during the controller initialization while the other five are used to access the external memory.

It also includes the physical layer (PHY) and DLLs for fine tuning the timing parameters to maximize the data valid windows at different frequencies.

2.4　Serial memory interface

SPEAr320 provides a serial memory interface (SMI), acting as an AHB slave interface (32-, 16- or 8-bit) to SPI-compatible off-chip memories.

These serial memories can be used either as data storage or for code execution.

Main features:

- Supports the following SPI-compatible Flash and EEPROM devices:
    - STMicroelectronics M25Pxxx, M45Pxxx
    - STMicroelectronics M95xxx, except M95040, M95020 and M95010
    - ATMEL AT25Fxx
    - YMC Y25Fxx
    - SST SST25LFxx
- Acts always as a SPI master and up to 2 SPI slave memory devices are supported (with separate chip select signals), with up to 16 MB address space each
- SMI clock signal (SMICLK) is generated by SMI (and input to all slaves) using a clock provided by the AHB bus
- SMICLK can be up to 50 MHz in fast read mode (or 20 MHz in normal mode). It can be controlled by a programmable 7-bit prescaler allowing up to 127 different clock frequencies.

2.5　External memory interface (EMI)

The EMI Controller provide a simple external memory interface that can be used for example to connect to NOR FLash memory or FPGA devices.

Main features:

● EMI bus master

● 16 and 8-bit transfers

● Can access 4 different peripherals using CS#, one at a time.

● Supports single asynchronous transfers.

● Supports peripherals which use Byte Lane procedure

2.6 Flexible static memory controller (FSMC)

SPEAr320 provides a Flexible Static Memory Controller (FSMC) which interfaces the AHB bus to external parallel NAND Flash memories.

● Provides an interface between AHB system bus and external NAND Flash memory devices.

● 8/16-bit wide data path

● FSMC performs only one access at a time and only one external device is accessed

● Supports little-endian and big-endian memory architectures

● AHB burst transfer handling to reduce access time to external devices

● Supplies an independent configuration for each memory bank

● Programmable timings to support a wide range of devices

   – Programmable wait states (up to 31)

   – Programmable bus turnaround cycles (up to 15)

   – Programmable output enable and write enable delays (up to 15)

● Independent chip select control for each memory bank

● Shares the address bus and the data bus with all the external peripherals

● Only chips selects are unique for each peripheral

● External asynchronous wait control

● Boot memory bank configurable at reset using external control pins

2.7 Multichannel DMA controller

Within its basic subsystem, SPEAr320 provides an DMA controller (DMAC) able to service up to 8 independent DMA channels for serial data transfers between single source and destination (i.e., memory-to-memory, memory-to-peripheral, peripheral to- memory, and peripheral-to-peripheral).

Each DMA channel can support a unidirectional transfer, with internal four-word FIFO per channel.

2.8 SMII/MII Ethernet controller

SPEAr320 features two Ethernet MACs, one supporting SMII and one supporting SMII and MII.

Each MAC channel has dedicated TX/RX signals while synchronization and clock signals are common for PHY connection.

The *Figure 3* shows the typical SMII configuration (a generic example with four ports):

Figure 3.   Typical SMII system
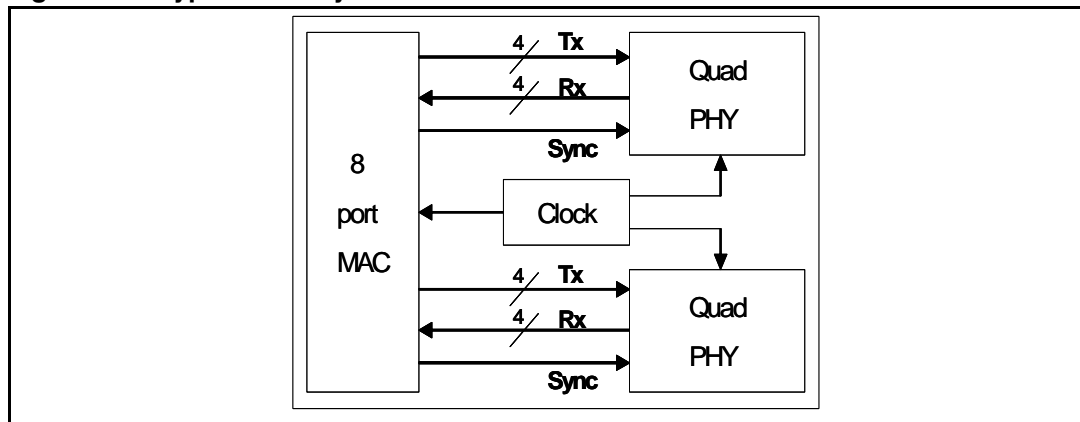
Each Ethernet port provides the following features:
- Compatible with IEEE Standard 802.3
- 10 and 100 Mbit/s operation
- Full and half duplex operation
- Statistics counter registers for RMON/MIB
- Interrupt generation to signal receive and transmit completion
- Automatic pad and CRC generation on transmitted frames
- Automatic discard of frames received with errors
- Address checking logic supports up to four specific 48-bit addresses
- Supports promiscuous mode where all valid received frames are copied to memory
- Hash matching of unicast and multicast destination addresses
- External address matching of received frames
- Physical layer management through MDIO interface

- Supports serial network interface operation
- Half duplex flow control by forcing collisions on incoming frames
- Full duplex flow control with recognition of incoming pause frames and hardware generation of transmitted pause frames
- Support for 802.1Q VLAN tagging with recognition of incoming VLAN and priority tagged frames
- Multiple buffers per receive and transmit frame
- Wake on LAN support
- Jumbo frames of up to 10240 bytes supported
- Configurable Endianess for the DMA Interface (AHB Master)

2.9 MII Ethernet controller

SPEAr320 provides an Ethernet MAC 10/100 Universal (commonly referred to as MAC-UNIV), enabling to transmit and receive data over Ethernet in compliance with the IEEE 802.3-2002 standard.

Main features:

● Supports the default Media Independent Interface (MII) defined in the IEEE 802.3 specifications.

● Supports 10/100 Mbps data transfer rates

● Local FIFO available (4 Kbyte RX, 2 Kbyte TX)

● Supports both half-duplex and full-duplex operation. In half-duplex operation, CSMA/CD protocol is provided

● Programmable frame length to support both standard and jumbo Ethernet frames with size up to 16 Kbytes

● 32/64/128-bit data transfer interface on system-side.

● A variety of flexible addresses filtering modes are supported

● A set of control and status registers (CSRs) to control GMAC core operation

● Native DMA with single-channel transmit and receive engines, providing 32/64/128-bit data transfers

● DMA implements dual-buffer (ring) or linked-list (chained) descriptor chaining

● An AHB slave acting as programming interface to access all CSRs, for both DMA and GMAC core subsystems

● An AHB master for data transfer to system memory

● 32-bit AHB master bus width, supporting 32, 64, and 128-bit wide data transactions

● It supports both big-endian and little-endian.

2.10 CAN controller

SPEAr320 has two CAN controllers for interfacing CAN 2.0 networks.

Main features:

● Supports CAN protocol version 2.0 part A and B

● Bit rates up to 1 MBit/s

● 32 Message Objects(136 X 32 Message RAM)

● Each Message Object has its own identifier mask

● Maskable interrupt

● Programmable loop-back mode for self-test operation

● Disabled Automatic Retransmission mode for Time Triggered CAN applications

2.11 USB2 host controller

SPEAr320 has two fully independent USB 2.0 hosts. Each consists of 5 major blocks:

● EHCI capable of managing high-speed transfers (HS mode, 480 Mbps)
● OHCI that manages the full and the low speed transfers (12 and 1.5 Mbps)
● Local 2-Kbyte FIFO
● Local DMA
● Integrated USB2 transceiver (PHY)

Both hosts can manage an external power switch, providing a control line to enable or disable the power, and an input line to sense any over-current condition detected by the external switch.

One host controller at time can perform high speed transfer.

2.12 USB2 device controller

Main features:

● Supports the 480 Mbps high-speed mode (HS) for USB 2.0, as well as the 12 Mbps full-speed (FS) and the low-speed (LS modes) for USB 1.1
● Supports 16 physical endpoints, configurable as different logical endpoints
● Integrated USB transceiver (PHY)
● Local 4 Kbyte FIFO shared among all the endpoints
● DMA mode and slave-only mode are supported
● In DMA mode, the UDC supports descriptor-based memory structures in application memory
● In both modes, an AHB slave is provided by UDC-AHB, acting as programming interface to access to memory-mapped control and status registers (CSRs)
● An AHB master for data transfer to system memory is provided, supporting 8, 16, and 32-bit wide data transactions on the AHB bus
● A USB plug (UPD) detects the connection of a cable.

2.13 CLCD controller

SPEAr320 has a color liquid crystal display controller (CLCDC) that provides all the necessary control signals to interface directly to a variety of color and monochrome LCD panels.

Main features:

● Resolution programmable up to 1024 x 768

● 16-bpp true-color non-palletized, for color STN and TFT

● 24-bpp true-color non-palletized, for color TFT

● Supports single and dual panel mono super twisted nematic (STN) displays with 4 or 8-bit interfaces

● Supports single and dual-panel color and monochrome STN displays

● Supports thin film transistor (TFT) color displays

● 15 gray-level mono, 3375 color STN, and 32 K color TFT support

● 1, 2, or 4 bits per pixel (bpp) palletized displays for mono STN

● 1, 2, 4 or 8-bpp palletized color displays for color STN and TFT

● Programmable timing for different display panels

● 256 entry, 16-bit palette RAM, arranged as a 128 x 32-bit RAM physically frame, line and pixel clock signals

● AC bias signal for STN and data enable signal for TFT panels patented gray scale algorithm

● Supports little and big-endian

2.14 GPIOs

A maximum of 102 GPIOs are available when part of the embedded or customizations IPs are not needed (see "Pin description" table).

Within its basic subsystem, SPEAr320 provides a base General Purpose Input/Output (GPIO) block (basGPIO). The base GPIO block provides 6 programmable inputs or outputs.
Each input/output can be controlled in two distinct modes:

● Software mode, through an APB interface.

● Hardware mode, through a hardware control interface.

Main features of the base GPIO block are:

● Six individually programmable input/output pins (default to input at reset)

● An APB slave acting as control interface in "software mode"

● Programmable interrupt generation capability on any number of pins.

● Hardware control capability of GPIO lines for different system configurations.

● Bit masking in both read and write operation through address lines.

Other GPIO blocks are present in the reconfigurable array subsystem (RAS).

2.15   Parallel port

Main features:

- Slave mode device interface for Standard parallel port host
- Supports unidirectional 8-bit data transfer from host to slave
- Supports 9th bit for parity/data/command etc.
- Maskable interrupts for data, device reset, auto line feed
- APB input clock frequency required is 83 MHz for acknowledgement timings
- Conforms to AMBA-APB specifications

2.16   SSP

SPEAr320 provides one synchronous serial port (SSP) block that offers a master or slave interface to enables synchronous serial communication with slave or master peripherals

Main features:

- Master or slave operation.
- Programmable clock bit rate and prescale.
- Separate transmit and receive first-in, first-out memory buffers, 16-bits wide, 8 locations deep.
- Programmable choice of interface operation:
    - SPI (Motorola)
    - Microwire (National Semiconductor)
    - TI synchronous serial.
- Programmable data frame size from 4 to 16-bits.
- Independent masking of transmit FIFO, receive FIFO, and receive overrun interrupts.
- Internal loopback test mode available.
- DMA interface

2.17   I2C

The SPEAr320 has 2 I2C interfaces;

Main features:

● Compliance to the I^{2}C bus specification (Philips)
● Supports three modes:
    – Standard (100 kbps)
    – Fast (400 kbps)
    – High-speed (3.4 Mbps)
● Clock synchronization
● Master and slave mode configuration possible
● Multi-master mode (bus arbitration)
● 7-bit or 10-bit addressing
● 7-bit or 10-bit combined format transfers
● Slave bulk transfer mode
● Ignores CBUS addresses (predessor to I2C that used to share the I2C bus)
● Transmit and receive buffers
● Interrupt or polled-mode operation
● handles bit and byte waiting at all bus speeds
● Digital filter for the received SDA and SCL lines
● Handles component parameters for configurable software driver support
● Supports APB data bus widths of 8, 16 and 32-bits.

2.18   I^{2}S audio block

The SPEAr320 has one I2S interface.

Main features:

● Conversion of AHB protocol to I^{2}S protocol.
● Supports 2.0 audio outputs (master mode only)
● Supports 32 (16L + 16R) and 64-bit (32L + 32R) of raw PCM data length
● 48 kHz audio sampling rate
● MIC/line-In recording (master/slave modes)

The the I2S audio blocks can be used to provide "Audio Play" and "Audio Record" capability. The Audio Play function works in master mode only but Audio Record can be used in both master and slave modes. In master mode, it outputs clock and WS signals in addition to stereo data. In slave mode, the clock and the WS signal has to be provided externally.

2.19   UARTs

The SPEAr320 has 2 UARTs featuring software flow control and 1 UART featuring hardware and/or software flow control.

2.19.1 UART0 with hardware flow control

Main features:

● Separate 16 x 8 (16 locations deep x 8-bit wide) transmit and 16 x 12 receive FIFOs to reduce CPU interrupts

● Speed up to 3 Mbps

● Hardware and/or software flow control

2.19.2 UART1 and UART2 with software flow control

Main features:

● Separate 16 x 8 (16 location deep x 8-bit wide) transmit and 16x12 receive FIFOs to reduce CPU interrupts

● Speed up to 5 Mbps.

2.20 JPEG CODEC

SPEAr320 provides a JPEG CODEC with header processing (JPGC), able to decode (or encode) image data contained in the SPEAr320 RAM, from the JPEG (or BMP) format to the BMP (or JPEG) format.

Main features:

● Compliance with the baseline JPEG standard (ISO/IEC 10918-1)

● Single-clock per pixel encoding/decoding

● Support for up to four channels of component color

● 8-bit/channel pixel depths

● Programmable quantization tables (up to four)

● Programmable Huffman tables (two AC and two DC)

● Programmable minimum coded unit (MCU)

● Configurable JPEG header processing

● Support for restart marker insertion

● Use of two DMA channels and of two 8 x 32-bits FIFO's (local to the JPEG) for efficient transferring and buffering of encoded/decoded data from/to the CODEC core.

2.21 Cryptographic co-processor (C3)

SPEAr320 has an embedded Channel Control Coprocessor (C3). C3 is a high-performance instruction driven DMA based co-processor. It executes instruction flows generated by the host processor. After it has been set-up by the host it runs in a completely autonomous way (DMA data in, data processing, DMA data out), until the completion of all the requested operations.

C3 has been used to accelerate the processing of cryptographic, security and network security applications. It can be used for other types of data intensive applications as well.

Hardware cryptographic co-processor features are listed below:

- Supported cryptographic algorithms:
  - Advanced encryption standard (AES) cipher in ECB, CBC, CTR modes.
  - Data encryption standard (DES) cipher in ECB and CBC modes.
  - SHA-1, HMAC-SHA-1, MD5, HMAC-MD5 digests.
- Instruction driven DMA based programmable engine.
- AHB master port for data access from/to system memory.
- AHB slave port for co-processor register accesses and initial engine-setup.
- The co-processor is fully autonomous (DMA input reading, cryptographic operation execution, DMA output writing) after being set up by the host processor.
- The co-processor executes programs written by the host in memory, it can execute an unlimited list of programs.
- The co-processor supports hardware chaining of cryptographic blocks for optimized execution of data-flow requiring multiple algorithms processing over the same set of data (for example encryption + hashing on the fly).

2.22   8-channel ADC

Main features:

- Successive approximation conversion method
- 10-bit resolution @ 1 Msps
- Hardware supporting up to 13.5 bits resolution at 8 ksps by oversampling and accumulation
- Eight analog input (AIN) channels, ranging from 0 to 2.5 V
- INL ± 1 LSB, DNL ± 1 LSB
- Programmable conversion speed, (min. conversion time is 1 µs)
- Programmable averaging of results from 1 (No averaging) up to 128
- Programmable auto scan for all the eight channels.

2.23   System controller

The System Controller provides an interface for controlling the operation of the overall system.

Main features:

- Power saving system mode control
- Crystal oscillator and PLL control
- Configuration of system response to interrupts
- Reset status capture and soft reset generation
- Watchdog and timer module clock enable

2.23.1 Power saving system mode control

Using three mode control bits, the system controller switch the SPEAr320 to any one of four different modes: DOZE, SLEEP, SLOW and NORMAL.

- **SLEEP mode**: In this mode the system clocks, HCLK and CLK, are disabled and the System Controller clock SCLK is driven by a low speed oscillator (nominally 32768 Hz). When either a FIQ or an IRQ interrupt is generated (through the VIC) the system enters DOZE mode. Additionally, the operating mode setting in the system control register automatically changes from SLEEP to DOZE.

- **DOZE mode**: In this mode the system clocks, HCLK and CLK, and the System Controller clock SCLK are driven by a low speed oscillator. The System Controller moves into SLEEP mode from DOZE mode only when none of the mode control bits are set and the processor is in Wait-for-interrupt state. If SLOW mode or NORMAL mode is required the system moves into the XTAL control transition state to initialize the crystal oscillator.

- **SLOW mode**: During this mode, both the system clocks and the System Controller clock are driven by the crystal oscillator. If NORMAL mode is selected, the system goes into the "PLL control" transition state. If neither the SLOW nor the NORMAL mode control bits are set, the system goes into the "Switch from XTAL" transition state.

- **NORMAL mode**: In NORMAL mode, both the system clocks and the System Controller clock are driven by the PLL output. If the NORMAL mode control bit is not set, then the system goes into the "Switch from PLL" transition state.

2.23.2 Clock and reset system

The clock system is a fully programmable block that generates all the clocks necessary to the chip.

The default operating clock frequencies are:

- Clock @ 333 MHz for the CPU.
- Clock @ 166 MHz for AHB bus and AHB peripherals.
- Clock @ 83 MHz for, APB bus and APB peripherals.
- Clock @ 333 MHz for DDR memory interface.

The default values give the maximum allowed clock frequencies. The clock frequencies are fully programmable through dedicated registers.

The clock system consists of 2 main parts: a multi clock generator block and two internal PLLs.

The multi clock generator block, takes a reference signal (which is usually delivered by the PLL), generates all clocks for the IPs of SPEAr320 according to dedicated programmable registers.

Each PLL uses an oscillator input of 24 MHz to generate a clock signal at a frequency corresponding at the highest of the group. This is the reference signal used by the multi clock generator block to obtain all the other requested clocks for the group. Its main feature is electromagnetic interference reduction capability.

The user can set up the PLL in order to modulate the VCO with a triangular wave. The resulting signal has a spectrum (and power) spread over a small programmable range of frequencies centered on F0 (the VCO frequency), obtaining minimum electromagnetic emissions. This method replaces all the other traditional methods of EMI reduction, such as

filtering, ferrite beads, chokes, adding power layers and ground planes to PCBs, metal shielding and so on. This gives the customer appreciable cost savings.

In sleep mode the SPEAr320 runs with the PLL disabled so the available frequency is 24 MHz or a sub-multiple (/2, /4, /8).

2.24 Vectored interrupt controller (VIC)

The VIC allows the OS interrupt handler to quickly dispatch interrupt service routines in response to peripheral interrupts. There are 32 interrupt lines and the VIC uses a separate bit position for each interrupt source. Software controls each request line to generate software interrupts.

2.25 General purpose timers

SPEAr320 provides 6 general purpose timers (GPTs) acting as APB slaves.

Each GPT consists of 2 channels, each one made up of a programmable 16-bit counter and a dedicated 8-bit timer clock prescaler. The programmable 8-bit prescaler performs a clock division by 1 up to 256, and different input frequencies can be chosen through configuration registers (a frequency range from 3.96 Hz to 48 MHz can be synthesized).

Two different modes of operation are available :
- Auto-reload mode, an interrupt source is activated, the counter is automatically cleared and then it restarts incrementing.
- Single-shot mode, an interrupt source is activated, the counter is stopped and the GPT is disabled.

2.26 PWM timers

SPEAr320 provides 4 PWM timers.

Main features:

- Prescaler to define the input clock frequency to each timer
- Programmable duty cycle from 0% to 100%
- Programmable pulse length
- APB slave interface for register programming

2.27 Watchdog timer

The ARM watchdog module consists of a 32-bit down counter with a programmable timeout interval that has the capability to generate an interrupt and a reset signal on timing out. The watchdog module is intended to be used to apply a reset to a system in the event of a software failure.

2.28   RTC oscillator

The RTC provides a 1-second resolution clock. This keeps time when the system is inactive and can be used to wake the system up when a programmed alarm time is reached. It has a clock trimming feature to compensate for the accuracy of the 32.768 kHz crystal and a secured time update.

Main features:

● Tme-of-day clock in 24 hour mode

● Calendar

● Alarm capability

● Isolation mode, allowing RTC to work even if power is not supplied to the rest of the device.

3 Pin description

The following tables describe the pinout of the SPEAr320 listed by functional block.

List of abbreviations:

PU = Pull Up

PD = Pull Down

3.1 Required external components

1. DDR_COMP_1V8: place an external 121 kΩ resistor between ball P4 and ball R4
2. USB_TX_RTUNE: connect an external 43.2 kΩ pull-down resistor to ball K5
3. DIGITAL_REXT: place an external 121 kΩ resistor between ball G4 and ball F4.

3.2 Dedicated pins

Table 2.   Master clock, RTC, Reset and 3.3 V comparator pin descriptions

Group	Signal name	Ball	Direction	Function	Pin type
Master Clock	MCLK_XI	P1	Input	24 MHz (typical) crystal in	Oscillator 2.5 V capable
	MCLK_XO	P2	Output	24 MHz (typical) crystal out	
RTC	RTC_XI	E2	Input	32 kHz crystal in	Oscillator 1V capable
	RTC_XO	E1	Output	32 kHz crystal out	
Reset	MRESET#	M17	Input	Main Reset	TTL Schmitt trigger input buffer, 3.3 V tolerant, PU
3.3 V Comp.	DIGITAL_REXT	G4	Output	Configuration	Analog, 3.3 V capable
	DIGITAL_GND_REX	F4	Power	Power	Power

Table 3.   Power supply pin description

Group	Signal name	Ball	Value
DIGITAL GROUND	GND	G6 G7 G8 G9 G10 G11 H6 H7 H8 H9 H10 H11 J6 J7 J8 J9 J10 J11 K6 K7 K8 K9 K10 K11 L6 L7 L8 L9 L10 M8 M9 M10	0 V
ANALOG GROUND	AGND	F2, G1, J2, L1, L3, L5, N2, N4, P3, R3, N12	0 V
I/O	V$_{DD}$3v3	F5 F6 F7 F10 F11 F12 G5 J12 K12 L12 M12	3.3 V

Table 3.   Power supply pin description

Group	Signal name	Ball	Value
CORE	V_{DD}	F8 F9 G12 H5 H12 J5 L11 M6 M7 M11	1.2 V
USB HOST0 PHY	HOST0_V_{DD}bc	L2	2.5 V
	HOST0_V_{DD}b3	K4	3.3 V
USB HOST1 PHY	HOST1_V_{DD}bc	K3	2.5 V
	HOST1_V_{DD}b3	J1	3.3 V
USB DEVICE PHY	DEVICE_V_{DD}bc	N1	2.5 V
	DEVICE_V_{DD}b3	N3	3.3 V
	HOST_V_{DD}bs	M3	1.2 V
OSCI (master clock)	MCLK_V_{DD}	R1	1.2 V
	MCLK_V_{DD}2v5	R2	2.5 V
PLL1	DITH1_AV_{DD}	G2	2.5 V
PLL2	DITH2_AV_{DD}	M4	2.5 V
DDR I/O	SSTL_V_{DD}e	M5 N5 N6 N7 N8 N9 N10 N11	1.8 V
ADC	ADC_AV_{DD}	N13	2.5 V
OSCI RTC	RTC_V_{DD}	F1	1.5 V

Table 4.   Debug pin descriptions

Group	Signal name	Ball	Direction	Function	Pin type
DEBUG	TEST_0	K16	Input	Test configuration ports. For functional mode, they have to be set to zero.	TTL input buffer, 3.3 V tolerant, PD
	TEST_1	K15			
	TEST_2	K14			
	TEST_3	K13			
	TEST_4	J15			
	BOOT_SEL	J14			
	nTRST	L16	Input	Test reset input	TTL Schmitt trigger input buffer, 3.3 V tolerant, PU
	TDO	L15	Output	Test data output	TTL output buffer, 3.3 V capable 4 mA
	TCK	L17	Input	Test clock	TTL Schmitt trigger input buffer, 3.3 V tolerant, PU
	TDI	L14	Input	Test data input	
	TMS	L13	Input	Test mode select	

Table 5.   Serial memory interface (SMI) pin description

Group	Signal name	Ball	Direction	Function	Pin type
SMI	SMI_DATAIN	M13	Input	Serial Flash input data	TTL Input Buffer 3.3 V tolerant, PU
	SMI_DATAOUT	M14	Output	Serial Flash output data	TTL output buffer 3.3 V capable 4 mA
	SMI_CLK	N17	I/O	Serial Flash clock	
	SMI_CS_0	M15	Output	Serial Flash chip select	
	SMI_CS_1	M16			

Table 6.   USB pin descriptions

Group	Signal name	Ball	Direction	Function	Pin type
USB DEV	DEV_DP	M1	I/O	USB Device D+	Bidirectional analog buffer 5 V tolerant
	DEV_DM	M2		USB Device D-	
	DEV_VBUS	G3	Input	USB Device VBUS	TTL input buffer 3.3 V tolerant, PD
	HOST1_DP	H1	I/O	USB HOST1 D+	Bidirectional analog buffer 5 V tolerant
	HOST1_DM	H2		USB HOST1 D-	

Table 6.   USB pin descriptions (continued)

Group	Signal name	Ball	Direction	Function	Pin type
USB HOST	HOST1_VBUS	H3	Output	USBHOST1 VBUS	TTL output buffer 3.3 V capable, 4 mA
	HOST1_OVRC	J4	Input	USB Host1 Over-Current	TTL input buffer 3.3 V tolerant, PD
	HOST0_DP	K1	I/O	USB HOST0 D+	Bidirectional analog buffer 5 V tolerant
	HOST0_DM	K2		USB HOST0 D-	
	HOST0_VBUS	J3	Output	USB HOST0 VBUS	TTL output buffer 3.3 V capable, 4 mA
	HOST0_OVRC	H4	Input	USB Host0 Over-current	TTL Input Buffer 3.3 V tolerant, PD
	USB_TXRTUNE	K5	Output	Reference resistor	Analog
	USB_ANALOG_TEST	L4	Output	Analog Test Output	Analog

Table 7.   ADC pin description

Group	Signal name	Ball	Direction	Function	Pin type
ADC	AIN_0	N16	Input	ADC analog input channel	Analog buffer 2.5 V tolerant
	AIN_1	N15			
	AIN_2	P17			
	AIN_3	P16			
	AIN_4	P15			
	AIN_5	R17			
	AIN_6	R16			
	AIN_7	R15			
	ADC_VREFN	N14		ADC negative voltage reference	
	ADC_VREFP	P14		ADC positive voltage reference	

Table 8.   DDR pin description

Group	Signal name	Ball	Direction	Function	Pin type
DDR	DDR_ADD_0	T2	Output	Address Line	SSTL_2/SSTL_18
	DDR_ADD_1	T1			
	DDR_ADD_2	U1			
	DDR_ADD_3	U2			
	DDR_ADD_4	U3			
	DDR_ADD_5	U4			
	DDR_ADD_6	U5			
	DDR_ADD_7	T5			
	DDR_ADD_8	R5			
	DDR_ADD_9	P5			
	DDR_ADD_10	P6			
	DDR_ADD_11	R6			
	DDR_ADD_12	T6			
	DDR_ADD_13	U6			
	DDR_ADD_14	R7			
	DDR_BA_0	P7	Output	Bank select	
	DDR_BA_1	P8			
	DDR_BA_2	R8			
	DDR_RAS	U8	Output	Row Add. Strobe	
	DDR_CAS	T8	Output	Col. Add. Strobe	
	DDR_WE	T7	Output	Write enable	
	DDR_CLKEN	U7	Output	Clock enable	
	DDR_CLK_P	T9	Output	Differential clock	Differential SSTL_2/SSTL_18
	DDR_CLK_N	U9			

Table 8.   DDR pin description (continued)

Group	Signal name	Ball	Direction	Function	Pin type
DDR	DDR_CS_0	P9	Output	Chip Select	SSTL_2/SSTL_18
	DDR_CS_1	R9			
	DDR_ODT_0	T3	I/O	On-Die Termination Enable lines	
	DDR_ODT_1	T4			
	DDR_DATA_0	P11	I/O	Data Lines (Lower byte)	
	DDR_DATA_1	R11			
	DDR_DATA_2	T11			
	DDR_DATA_3	U11			
	DDR_DATA_4	T12			
	DDR_DATA_5	R12			
	DDR_DATA_6	P12			
	DDR_DATA_7	P13			
	DDR_DQS_0	U10	Output	Lower Data Strobe	Differential SSTL_2/SSTL_18
	DDR_nDQS_0	T10			
	DDR_DM_0	U12	Output	Lower Data Mask	SSTL_2/SSTL_18
	DDR_GATE_0	R10	I/O	Lower Gate Open	
	DDR_DATA_8	T17	I/O	Data Lines (Upper byte)	SSTL_2/SSTL_18
	DDR_DATA_9	T16			
	DDR_DATA_10	U17			
	DDR_DATA_11	U16			
	DDR_DATA_12	U14			
	DDR_DATA_13	U13			
	DDR_DATA_14	T13			
	DDR_DATA_15	R13			
	DDR_DQS_1	U15	I/O	Upper Data Strobe	Differential SSTL_2/SSTL_18
	DDR_nDQS_1	T15			
	DDR_DM_1	T14	I/O	Upper Data Mask	SSTL_2/SSTL_18
	DDR_GATE_1	R14		Upper Gate Open	
	DDR_VREF	P10	Input	Reference Voltage	Analog
	DDR_MEM_COMP_GND	R4	Power	Return for Ext. Resistors	Power
	DDR_MEM_COMP_REXT	P4	Power	Ext. Resistor	Analog
	DDR2_EN	J13	Input	Configuration	TTL Input Buffer 3.3 V Tolerant, PU

3.3 Shared I/O pins (PL_GPIOs)

The 98 PL_GPIO and 4 PL_CLK pins have the following characteristics:

- – Output buffer: TTL 3.3 V capable up to 10 mA
- – Input buffer: TTL, 3.3 V tolerant, selectable internal pull up/pull down (PU/PD)

Configuration modes

The following modes can be selected by programming the RAS control registers.

● Mode 1: SMII automation networking mode
● Mode 2: MII automation networking mode
● Mode 3: Expanded automation mode
● Mode 4: Printer mode

The peripherals available are shown in *Table 9: Available peripherals in each configuration mode*. Details of each PL_GPIO pin are given in *Table 10: PL_GPIO pin description on page 29*

Mode 1 is the default mode for SPEAr320.

Boot pins

The status of the boot pins is read at startup by the BootROM. Refer to the description of the Boot register in the SPEAr320 user manual.

Alternate functions

Other peripheral functions are listed in the Alternate Functions column of *Table 10: PL_GPIO pin description* and can be individually enabled/disabled via RAS control register 1. Refer to the user manual for the register descriptions.

Table 9.   Available peripherals in each configuration mode

Modes	FSMC NAND interface	EMI NOR interface	SDIO/MMC data lines	No. of Ethernet ports	CLCD	Touchscreen	No of CAN interfaces	Std. parallel portr	No. of PWM outputs	I2S	No. of I2C interfaces	No. of SPI interfaces	No. of UARTs
1	8-bit		8	2 SMII	1	1	2		4	1	1[1]	3[2]	3[3]
2	8-bit		8	2 MII[4]			2		4	1	1[1]	3[2]	3[3]
3	16-bit	16-bit		1 SMII			2		3		2[1]	1[2]	3[3]
4	8-bit		8	2 SMII				1	4			2[2]	3[3]

1. Assuming I2C0 alternate functions are enabled on PL5 and PL4 (see *Table 10*)

2. Assuming that SSP0alternate functions are enabled on PL9 thru PL6 (see *Table 10*)

3. Assuming that UART0 alternate functions are enabled on PL3 and PL3 and optionally on PL42-37 if hardware flow control is used (see *Table 10*)

4. Assuming that MII0 alternate functions are enabled on PL27 thru 10 (see *Table 10*)

Table 10. PL_GPIO pin description

PL / pin number	Boot pins	Alternate function (enabled by RAS register 1)	Configuration mode (enabled by RAS register 2)				Function in RAS GPIO mode	Function in debug trace mode (ETM)
			1	2	3	4		
97/H16			CLD0	MII1_TXCLK	EMI_A0	0	GPIO_97	ARM_TRACE_CLK
96/H15			CLD1	MII1_TXD0	EMI_A1	0	GPIO_96	ARM_TRACE_PKTA[0]
95/H14			CLD2	MII1_TXD1	EMI_A2	0	GPIO_95	ARM_TRACE_PKTA[1]
94/H13			CLD3	MII1_TXD2	EMI_A3	0	GPIO_94	ARM_TRACE_PKTA[2]
93/G17			CLD4	MII1_TXD3	EMI_A4	0	GPIO_93	ARM_TRACE_PKTA[3]
92/G16			CLD5	MII1_TXEN	EMI_A5	0	GPIO_92	ARM_TRACE_PKTB[0]
91/G15			CLD6	MII1_TXER	EMI_A6	0	GPIO_91	ARM_TRACE_PKTB[1]
90/G14			CLD7	MII1_RXCLK	EMI_A7	0	GPIO_90	ARM_TRACE_PKTB[2]
89/F17			CLD8	MII1_RXDV	EMI_A8	0	GPIO_89	ARM_TRACE_PKTB[3]
88/F16			CLD9	MII1_RXER	EMI_A9	0	GPIO_88	ARM_TRACE_SYNCA
87/G13			CLD10	MII1_RXD0	EMI_A10	0	GPIO_87	ARM_TRACE_SYNCB
86/E17			CLD11	MII1_RXD1	EMI_A11	0	GPIO_86	ARM_PIPESTATA[0]
85/F15			CLD12	MII1_RXD2	EMI_A12	SPP_DATA0	GPIO_85	ARM_PIPESTATA[1]
84/D17			CLD13	MII1_RXD3	EMI_A13	SPP_DATA1	GPIO_84	ARM_PIPESTATA[2]
83/E16			CLD14	MII1_COL	EMI_A14	SPP_DATA2	GPIO_83	ARM_PIPESTATB[0]
82/E15			CLD15	MII1_CRS	EMI_A15	SPP_DATA3	GPIO_82	ARM_PIPESTATB[1]
81/C17			CLD16	MII1_MDIO	EMI_A16	SPP_DATA4	GPIO_81	ARM_PIPESTATB[2]
80/D16			CLD17	MII1_MDC	EMI_A17	SPP_DATA5	GPIO_80	ARM_TRACE_PKTA[4]
79/F14			CLD18	0	EMI_A18	SPP_DATA6	GPIO_79	ARM_TRACE_PKTA[5]
78/D15			CLD19	0	EMI_A19	SPP_DATA7	GPIO_78	ARM_TRACE_PKTA[6]
77/B17			CLD20	0	EMI_A20	SPP_STRBn	GPIO_77	ARM_TRACE_PKTA[7]
76/F13			CLD21	0	EMI_A21	SPP_ACKn	GPIO_76	ARM_TRACE_PKTB[4]
75/E14			CLD22	0	EMI_A22	SPP_BUSY	GPIO_75	ARM_TRACE_PKTB[5]
74/C16			CLD23	0	EMI_A23	SPP_PERROR	GPIO_74	ARM_TRACE_PKTB[6]

Table 10. PL_GPIO pin description (continued)

PL / pin number	Boot pins	Alternate function (enabled by RAS register 1)	Configuration mode (enabled by RAS register 2)				Function in RAS GPIO mode	Function in debug trace mode (ETM)
			1	2	3	4		
73/A17			CLAC	0	EMI_D8/ FSMC_D8	SPP_SELECT	GPIO_73	ARM_TRACE_P KTB[7]
72/B16			CLFP	0	EMI_D9/ FSMC_D9	SPP_AUTOFDn	GPIO_72	
71/D14			CLLP	0	EMI_D10/ FSMC_D10	SPP_FAULTn	GPIO_71	
70/C15			CLLE	0	EMI_D11/ FSMC_D11	SPP_INITn	GPIO_70	
69/A16			CLPOWER	0	EMI_WAIT	SPP_SELINn	GPIO_69	
68/B15			FSMC_D0	FSMC_D0	EMI_D0/ FSMC_D0	FSMC_D0	GPIO_68	
67/C14			FSMC_D1	FSMC_D1	EMI_D1/ FSMC_D0	FSMC_D1	GPIO_67	
66/E13			FSMC_D2	FSMC_D2	EMI_D2/ FSMC_D2	FSMC_D2	GPIO_66	
65/B14			FSMC_D3	FSMC_D3	EMI_D3/ FSMC_D3	FSMC_D3	GPIO_65	
64/D13			FSMC_D4	FSMC_D4	EMI_D4/ FSMC_D4	FSMC_D4	GPIO_64	
63/C13			FSMC_D5	FSMC_D5	EMI_D5/ FSMC_D5	FSMC_D5	GPIO_63	
62/A15	H7		FSMC_D6	FSMC_D6	EMI_D6/ FSMC_D6	FSMC_D6	GPIO_62	
61/E12	H6		FSMC_D7	FSMC_D7	EMI_D7/ FSMC_D7	FSMC_D7	GPIO_61	
60/A14	H5		FSMC_ADDR_ LE	FSMC_ADDR_ LE	FSMC_ADDR_LE	FSMC_ADDR_L E	GPIO_60	
59/B13	H4		FSMC_WE	FSMC_WE	EMI_WE/ FSMC_WE	FSMC_WE	GPIO_59	
58/D12	H3		FSMC_RE	FSMC_RE	EMI_OE/ FSMC_RE	FSMC_RE	GPIO_58	
57/E11	H2		FSMC_CMD_ LE	FSMC_CMD_ LE	FSMC_CMD_LE	FSMC_CMD_LE	GPIO_57	
56/C12	H1		FSMC_RDY /BSY	FSMC_RDY/ BSY	FSMC_RDY/BSY	FSMC_RDY/ BSY	GPIO_56	
55/A13	H0		FSMC_CS0	FSMC_CS0	EMI_CE0/ FSMC_CS0	FSMC_CS0	GPIO_55	
54/E10	B3		FSMC_CS1	FSMC_CS1	EMI_CE1/ FSMC_CS1	FSMC_CS1	GPIO_54	
53/D11	B2		FSMC_CS2	FSMC_CS2	EMI_CE2/ FSMC_CS2	FSMC_CS2	GPIO_53	
52/B12	B1		FSMC_CS3	FSMC_CS3	EMI_CE_3/ FSMC_CS3	FSMC_CS3	GPIO_52	
51/D10	B0		SD_CD	SD_CD	EMI_BYTEN0	SD_CD	GPIO_51	
50/A12		TMR_CPTR4	SD_DAT7	SD_DAT7	EMI_BYTEN1	SD_DAT7	GPIO_50	
49/C11		TMR_CPTR3	SD_DAT6	SD_DAT6	EMI_D12/ FSMC_D12	SD_DAT6	GPIO_49	

Table 10.   PL_GPIO pin description (continued)

PL / pin number	Boot pins	Alternate function (enabled by RAS register 1)	Configuration mode (enabled by RAS register 2)				Function in RAS GPIO mode	Function in debug trace mode (ETM)
			1	2	3	4		
48/B11		TMR_CPTR2	SD_DAT5	SD_DAT5	EMI_D13/ FSMC_D13	SD_DAT5	GPIO_48	
47/C10		TMR_CPTR1	SD_DAT4	SD_DAT4	EMI_D14/ FSMC_D14	SD_DAT4	GPIO_47	
46/A11		TMR_CLK4	SD_DAT3	SD_DAT3	EMI_D15/ FSMC_D15	SD_DAT3	GPIO_46	
45/B10		TMR_CLK3	SD_DAT2	SD_DAT2	UART1_DCD	SD_DAT2	GPIO_45	
44/A10		TMR_CLK2	SD_DAT1	SD_DAT1	UART1_DSR	SD_DAT1	GPIO_44	
43/E9		TMR_CLK1	SD_DAT0	SD_DAT0	UART1_RTS	SD_DAT0	GPIO_43	
42/D9		UART0_DTR	I2S_RX	I2S_RX	0	0	GPIO_42	
41/C9		UART0_RI	I2S_TX	I2S_TX	0	0	GPIO_41	
40/B9		UART0_DSR	I2S_LR	I2S_LR	0	0	GPIO_40	
39/A9		UART0_DCD	I2S_CLK	I2S_CLK	0	0	GPIO_39	
38/A8		UART0_RTS	PWM0	PWM0	0	0	GPIO_38	
37/B8		UART0_CTS	PWM1	PWM1	0	0	GPIO_37	
36/C8		SSP0_CS4	TOUCH SCREEN X	0	UART1_CTS	UART1_CTS	GPIO_36	
35/D8		SSP0_CS3	I2S_AUDIO SAMPLE	0	UART1_DTR	UART1_DTR	GPIO_35	
34/E8		SSP0_CS2	SD_LED / PWM2	SD_LED / PWM2	UART1_RI	UART1_RI	GPIO_34	
33/E7		basGPIO5	CAN0_TX	CAN0_TX	CAN0_TX	UART1_DCD	GPIO_33	
32/D7		basGPIO4	CAN0_RX	CAN0_RX	CAN0_RX	UART1_DSR	GPIO_32	
31/C7		basGPIO3	CAN1_TX	CAN1_TX	CAN1_TX	UART1_RTS	GPIO_31	
30/B7		basGPIO2	CAN1_RX	CAN1_RX	CAN1_RX		GPIO_30	
29/A7		basGPIO1	UART1_TX	UART1_TX	UART1_TX	UART1_TX	GPIO_29	
28/A6		basGPIO0	UART1_RX	UART1_RX	UART1_RX	UART1_RX	GPIO_28	
27/B6		MII0_TXCLK	SMII0_TX	0	SMII0_TX	SMII0_TX	GPIO_27	
26/A5		MII0_TXD0	SMII0_RX	0	SMII0_RX	SMII0_RX	GPIO_26	
25/C6		MII0_TXD1	SMII1_TX	0	0	SMII1_TX	GPIO_25	
24/B5		MII0_TXD2	SMII1_RX	0	0	SMII1_RX	GPIO_24	
23/A4		MII0_TXD3	SMII_SYNC	0	SMII_SYNC	SMII_SYNC	GPIO_23	
22/D6		MII0_TXEN	SMII_CLKOUT	0	SMII_CLKOUT	SMII_CLKOUT	GPIO_22	
21/C5		MII0_TXER	SMII_CLKIN	0	SMII_CLKIN	SMII_CLKIN	GPIO_21	
20/B4		MII0_RXCLK	SSP1_MOSI	0	0	SSP1_MOSI	GPIO_20	
19/A3		MII0_RXDV	SSP1_CLK	0	0	SSP1_CLK	GPIO_19	
18/D5		MII0_RXER	SSP1_SS0	0	0	SSP1_SS0	GPIO_18	
17/C4		MII0_RXD0	SSP1_MISO	0	0	SSP1_MISO	GPIO_17	
16/E6		MII0_RXD1	SSP2_MOSI	0	0	0	GPIO_16	

Table 10.   PL_GPIO pin description (continued)

PL / pin number	Boot pins	Alternate function (enabled by RAS register 1)	Configuration mode (enabled by RAS register 2)				Function in RAS GPIO mode	Function in debug trace mode (ETM)
			1	2	3	4		
15/B3		MII0_RXD2	SSP2_CLK	0	PWM0	PWM0	GPIO_15	
14/A2		MII0_RXD3	SSP2_SS0	0	PWM1	PWM1	GPIO_14	
13/A1		MII0_COL	SSP2_MIDO	0	PWM2	PWM2	GPIO_13	
12/D4		MII0_CRS	PWM3	0	PWM3	PWM3	GPIO_12	
11/E5		MII0_MDC	SMII_MDIO	0	SMII_MDIO	SMII_MDIO	GPIO_11	
10/C3		MII0_MDIO	SMII_MDC	0	SMII_MDC	SMII_MDC	GPIO_10	
9/B2		SSP0_MOSI	0	0	0	0	GPIO_9	
8/C2		SSP0_CLK	0	0	0	0	GPIO_8	
7/D3		SSP0_SS0	0	0	0	0	GPIO_7	
6/B1		SSP0_MISO	0	0	0	0	GPIO_6	
5/D2		I2C0_SDA	0	0	0	0	GPIO_5	
4/C1		I2C0_SCL	0	0	0	0	GPIO_4	
3/D1		UART0_RX	0	0	0	0	GPIO_3	
2/E4		UART0_TX	0	0	0	0	GPIO_2	
1/E3		IrDA_RX	UART2_TX	UART2_TX	UART2_TX	UART2_TX	GPIO_1	
0/F3		IrDA_TX	UART2_RX	UART2_RX	UART2_RX	UART2_RX	GPIO_0	
CK1/K17		PL_CLK1	CLCP	0	I2C1_SDA	SD_LED	GPIO_98	
CK2/J17		PL_CLK2	SD_CLK	SD_CLK	I2C1_SCL	SD_CLK	GPIO_99	
CK3/J16		PL_CLK3	SD_WP	SD_WP	0	SD_WP	GPIO_100	
CK4/H17		PL_CLK4	SD_CMD	SD_CMD	0	SD_CMD	GPIO_101	

Notes/legend for *Table 10*:

EMI_: External Memory interface (for NOR Flash or FPGA) signals

FSMC_: Flexible Static Memory Controller (for NAND Flash) signals

GPIO (General purpose I/O):

    basGPIO: Base GPIOs in the base subsystem

    GPIO_102 to GPIO_0: GPIOs in the RAS subsystem

MII0_ : MII signals from the MII controller in the high speed subsystem

MII1_: MII signals from the SMII controller in the RAS subsystem working in MII mode

SMII0_, **SMII1_**: Signals from the 2 SMII controllers in the RAS subsystem working in SMII mode

SPP_: Standard parallel port signals

SSP_: SPI interface signals from the synchronous serial peripheral

TMR_: General purpose timer signals

Note: 1   *Table 10 cells filled with '0' or '1' are unused and unless otherwise configured as Alternate function or GPIO, the corresponding pin is held at low or high level respectively by the internal logic.*

     2   *Pins shared by EMI and FSMC: Depending on the AHB address to be accessed the pins are used for EMI or FSMC transfers.*

Table shading:

FSMC	FSMC pins: NAND Flash
CLCD	Color LCD controller pins

4 Memory map

Table 11. SPEAr320 main memory map

Start address	End address	Peripheral	Description
0x0000.0000	0x3FFF.FFFF	External DRAM	Low power DDR or DDR2
0x4000.0000	0xBFFF.FFFF	-	RAS (See *Table 12*)
0xC000.0000	0xCFFF.FFFF	-	Reserved
0xD000.0000	0xD007.FFFF	UART0	
0xD008.0000	0xD00F.FFFF	ADC	
0xD010.0000	0xD017.FFFF	SPI0	
0xD018.0000	0xD01F.FFFF	I2C0	
0xD020.0000	0xD07F.FFFF	-	Reserved
0xD080.0000	0xD0FF.FFFF	JPEG CODEC	
0xD100.0000	0xD17F.FFFF	IrDA	
0xD180.0000	0xD1FF.FFFF	-	Reserved
0xD280.0000	0xD7FF.FFFF	SRAM	Static RAM shared memory (8 Kbytes)
0xD800.0000	0xE07F.FFFF	-	Reserved
0xE080.0000	0xE0FF.FFFF	Ethernet controller	MAC
0xE100.0000	0xE10F.FFFF	USB 2.0 device	FIFO
0xE110.0000	0xE11F.FFFF	USB 2.0 device	Configuration registers
0xE120.0000	0xE12F.FFFF	USB 2.0 device	Plug detect
0xE130.0000	0xE17F.FFFF	-	Reserved
0xE180.0000	0xE18F.FFFF	USB2.0 EHCI 0-1	
0xE190.0000	0xE19F.FFFF	USB2.0 OHCI 0	
0xE1A0.0000	0xE20F.FFFF	-	Reserved
0xE210.0000	0xE21F.FFFF	USB2.0 OHCI 1	
0xE220.0000	0xE27F.FFFF	-	Reserved
0xE280.0000	0xE28F.FFFF	ML USB ARB	Configuration register
0xE290.0000	0xE7FF.FFFF	-	Reserved
0xE800.0000	0xEFFF.FFFF	-	Reserved
0xF000.0000	0xF00F.FFFF	Timer0	
0xF010.0000	0xF10F.FFFF	-	Reserved
0xF110.0000	0xF11F.FFFF	ITC Primary	
0xF120.0000	0xF7FF.FFFF	-	Reserved
0xF800.0000	0xFBFF.FFFF	Serial Flash memory	
0xFC00.0000	0xFC1F.FFFF	Serial Flash controller	

Table 11.   SPEAr320 main memory map (continued)

Start address	End address	Peripheral	Description
0xFC20.0000	0xFC3F.FFFF	-	Reserved
0xFC40.0000	0xFC5F.FFFF	DMA controller	
0xFC60.0000	0xFC7F.FFFF	DRAM controller	
0xFC80.0000	0xFC87.FFFF	Timer 1	
0xFC88.0000	0xFC8F.FFFF	Watchdog timer	
0xFC90.0000	0xFC97.FFFF	Real time clock	
0xFC98.0000	0xFC9F.FFFF	basGPIO	
0xFCA0.0000	0xFCA7.FFFF	System controller	
0xFCA8.0000	0xFCAF.FFFF	Miscellaneous registers	
0xFCB0.0000	0xFCB7.FFFF	Timer 2	
0xFCB8.0000	0xFCFF.FFFF	-	Reserved
0xFD00.0000	0xFEFF.FFFF	-	Reserved
0xFF00.0000	0xFFFF.FFFF	BootROM	

Table 12.   Reconfigurable array subsystem (RAS) memory map

Start address	End address	Peripheral	Description
0x4000_0000	0x47FF_FFFF	EMI	
0x4800_0000	0x4BFF_FFFF	Reserved	
0x4C00_0000	0x5FFF_FFFF	FSMC	
0x6000_0000	0x6FFF_FFFF	I2S	
0x7000_0000	0x7FFF_FFFF	SDIO	
0x8000_0000	0x8000_3FFF	Boot memory	
0x8000_4000	0x8FFF_FFFF	Reserved	
0x9000_0000	0x9FFF_FFFF	CLCD	
0xA000_0000	0xA0FF_FFFF	Parallel port	
0xA100_0000 -	0xA1FF_FFFF	CAN0	
0xA200_0000 -	0xA2FF_FFFF	CAN1	
0xA300_0000 -	0xA3FF_FFFF	UART1	
0xA400_0000 -	0xA4FF_FFFF	UART23	
0xA500_0000	0xA5FF_FFFF	SPI1	
0xA600_0000	0xA6FF_FFFF	SPI2	
0xA700_0000	0xA7FF_FFFF	I2C1	
0xA800_0000	0xA8FF_FFFF	Quad PWM timer	
0xA900_0000	0xA9CF_FFFF	GPIO	
0xA9D0_0000	0xA9FF_FFFF	Reserved	

Table 12.   Reconfigurable array subsystem (RAS) memory map (continued)

Start address	End address	Peripheral	Description
0xAA00_0000	0xAAFF_FFFF	SMII0	
0xAB00_0000	0xABFF_FFFF	SMII1/MII	
0xAC00_0000	0xB2FF_FFFF	Reserved	
0xB300_0000	0xBFFF_FFFF	AHB interface	

5 Electrical characteristics

5.1 Absolute minimum and maximum ratings

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

The absolute minimum and maximum rating is the maximum stress that can be applied to a device without causing permanent damage. However, extended exposure to minimum/maximum ratings may affect long-term device reliability.

Table 13.   Absolute minimum and maximum ratings

Symbol	Parameter	Minimum value	Maximum value	Unit
V_{DD} 1.2	Supply voltage for the core	- 0.3	1.6	V
V_{DD} 3.3	Supply voltage for the I/Os	- 0.3	4.8	V
V_{DD} 2.5	Supply voltage for the analog blocks	- 0.3	4.8	V
V_{DD} 1.8	Supply voltage for the DRAM interface	- 0.3	4.8	V
T_J	Junction temperature	-40	125	°C
T_{STG}	Storage temperature	-55	150	°C

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

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

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:

$$P_D = K / (T_j + 273 \text{ °C})$$

And, solving first equations:

$$K = P_D \bullet (T_A + 273 \text{ °C}) + \Theta_{JA} \text{ x } P_D{}^2$$

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

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

5.2 Maximum power consumption

Note: *These values take into consideration the worst cases of process variation and voltage range and must be used to design the power supply section of the board.*

Table 14.   Maximum power consumption

Symbol	Description	Max	Unit
V_{DD} 1.2	Supply voltage for the core [1]	420	mA
V_{DD} 1.8	Supply voltage for the DRAM interface [2]	160	mA
V_{DD} 2.5	Supply voltage for the analog blocks	35	mA
V_{DD} 3.3	Supply voltage for the I/Os [3]	15	mA
P_D	Maximum power consumption	930	mW

1. Peak current with CPU at maximum speed in asynchronous mode with DDR at maximum speed.
2. Peak current with Linux memory test (50% write and 50% read) plus DMA reading memory.
3. With 30 logic channels connected to the device and simultaneously switching at 10 MHz.

The maximum current and power values listed above are not guaranteed to be the highest obtainable. These values are dependent on many factors including the type of applications running, clock rates, use of internal functional capabilities, external interface usage, case temperature, and the power supply voltages. Your specific application can produce significantly different results. 1.2 V current and power are primarily dependent on the applications running and the use of internal chip functions (DMA, USB, Ethernet, and so on).

3.3 V current and power are primarily dependent on the capacitive loading, frequency, and utilization of the external buses.

5.3 DC electrical characteristics

The recommended operating conditions are listed in the following table:

Table 15.   Recommended operating conditions

Symbol	Parameter	Min	Typ	Max	Unit
V_{DD} core	Supply voltage for the core	1.14	1.2	1.26	V
V_{DD} I/O	Supply voltage for the I/Os	3	3.3	3.6	V
V_{DD} PLL	PLL supply voltage	2.25	2.5	2.75	V
V_{DD} OSC	Oscillator supply voltage	2.25	2.5	2.75	V
V_{DD} 1.8	Supply voltage for DRAM interface	1.7	1.8	1.9	V

Table 15.   Recommended operating conditions (continued)

Symbol	Parameter	Min	Typ	Max	Unit
V_{DD} RTC	RTC supply voltage	1.2	1.5	1.8	V
T_A	Operating temperature	-40		85	°C

5.4   Overshoot and undershoot

This product can support the following values of overshoot and undershoot.

Table 16.   Overshoot and undershoot specifications

Parameter	3V3 I/Os	2V5 I/Os	1V8 I/Os
Amplitude	500 mV	500 mV	500 mV
Ratio of overshoot (or undershoot) duration with respect to pulse width	1/3	1/3	1/3

If the amplitude of the overshoot/undershoot increases (decreases), the ratio of Overshoot/Undershoot width to the pulse width decreases (increases). The formula relating the two is:

Amplitude of OS/US = 0.75*(1- ratio of OS (or US) duration with respect to pulse width)

Note: *The value of overshoot/undershoot should not exceed the value of 0.5 V. However, the duration of the overshoot/undershoot can be increased by decreasing its amplitude.*

5.5   General purpose I/O characteristics

The 3.3 V I/Os are compliant with JEDEC standard JESD8b

Table 17.   Low voltage TTL DC input specification (3 V< V_{DD} <3.6 V)

Symbol	Parameter	Min	Max	Unit
V_{IL}	Low level input voltage		0.8	V
V_{IH}	High level input voltage	2		V
V_{hyst}	Schmitt trigger hysteresis	300	800	mV

Table 18.   Low voltage TTL DC output specification (3 V< V_{DD} <3.6 V)

Symbol	Parameter	Test condition	Min	Max	Unit
V_{OL}	Low level output voltage	I_{OL}= X mA [1]		0.3	V
V_{OH}	High level output voltage	I_{OH}= -X mA [1]	V_{DD} - 0.3		V

1. For the max current value (X mA) refer to *Section 3: Pin description*.

Table 19.   Pull-up and pull-down characteristics

Symbol	Parameter	Test condition	Min	Max	Unit
R_{PU}	Equivalent pull-up resistance	$V_I = 0\ V$	29	67	$k\Omega$
R_{PD}	Equivalent pull-down resistance	$V_I = V_{DDE}3V3$	29	103	$k\Omega$

5.6   LPDDR and DDR2 pin characteristics

Table 20.   DC characteristics

Symbol	Parameter	Test condition	Min	Max	Unit
V_{IL}	Low level input voltage	SSTL2	-0.3	V_{REF}-0.15	V
		SSTL18	-0.3	V_{REF}-0.125	V
V_{IH}	High level input voltage	SSTL2	V_{REF}+0.15	$V_{DDE}2V5$+0.3	V
		SSTL18	V_{REF}+0.125	$V_{DDE}1V8$+0.3	V
V_{hyst}	Input voltage hysteresis		200		mV

Table 21.   Driver characteristics

Symbol	Parameter	Min	Typ	Max	Unit
R_O	Output impedance		45		Ω

Table 22.   On die termination

Symbol	Parameter	Min	Typ	Max	Unit
RT1*	Termination value of resistance for on die termination		75		Ω
RT2*	Termination value of resistance for on die termination		150		Ω

Table 23.   Reference voltage

Symbol	Parameter	Min	Typ	Max	Unit
V_{REFIN}	Voltage applied to core/pad	0.49 * V_{DDE}	0.500 * V_{DDE}	0.51 * V_{DDE}	V

5.6.1   DDR2 timing characteristics

The characterization timing is done considering an output load of 10 pF on all the DDR pads. The operating conditions are in worst case V = 0.90 V T_A = 125° C and in best case V=1.10 V T_A = 40° C.

DDR2 read cycle timings

Figure 4.   Read cycle waveforms
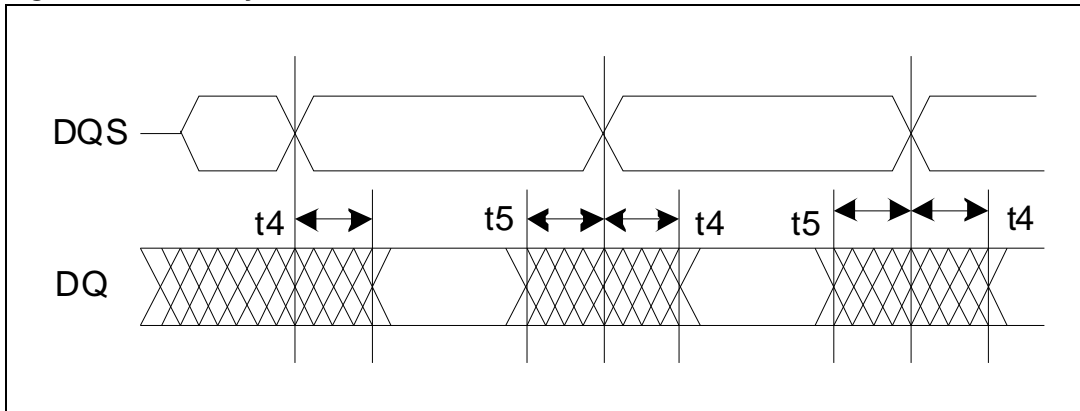
Figure 5.   Read cycle path
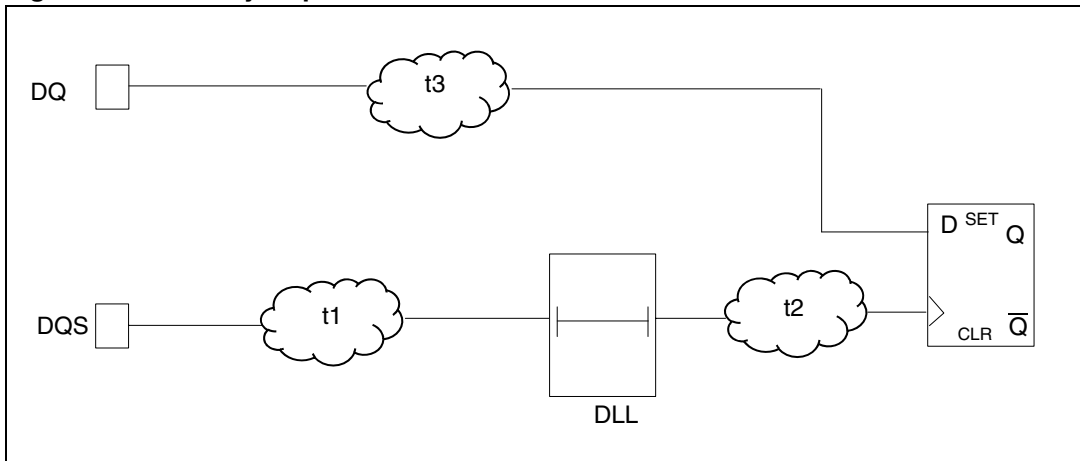
Table 24.   Read cycle timings

Frequency	t4 max	t5 max	t5 max
333 MHz	1.24 ns	-495 ps	-495 ps
266 MHz	1.43 ns	-306 ps	-306 ps
200 MHz	1.74 ns	4 ps	4 ps
166 MHz	2.00 ns	260 ps	260 ps
133 MHz	2.37 ns	634 ps	634 ps

DDR2 write cycle timings

Figure 6.   Write cycle waveforms
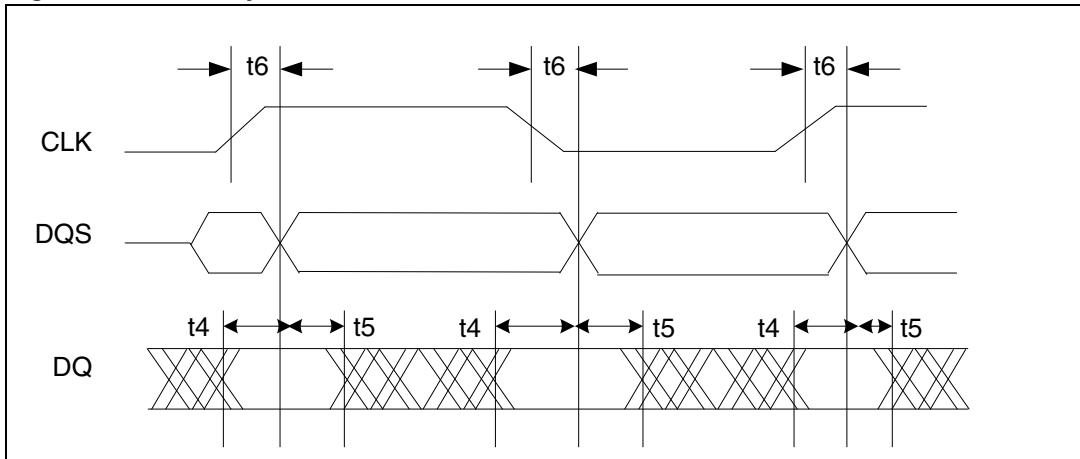
Figure 7.   Write cycle path
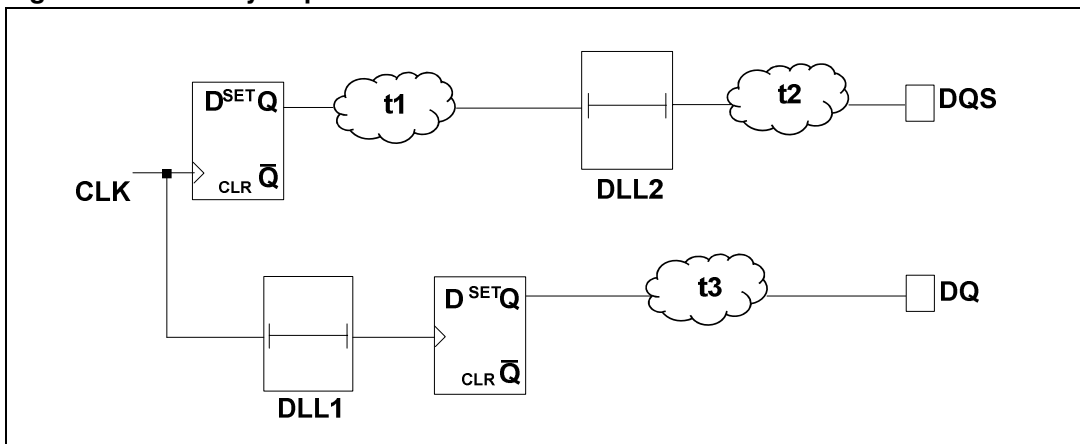
Table 25.   Write cycle timings

Frequency	t4 max	t5 max	Unit
333 MHz	1.36	-1.55	ns
266 MHz	1.55	-1.36	ns
200 MHz	1.86	-1.05	ns
166 MHz	2.11	- 794	ns
133 MHz	2.49	-420	ns

DDR2I command timings

Figure 8.   Command waveforms
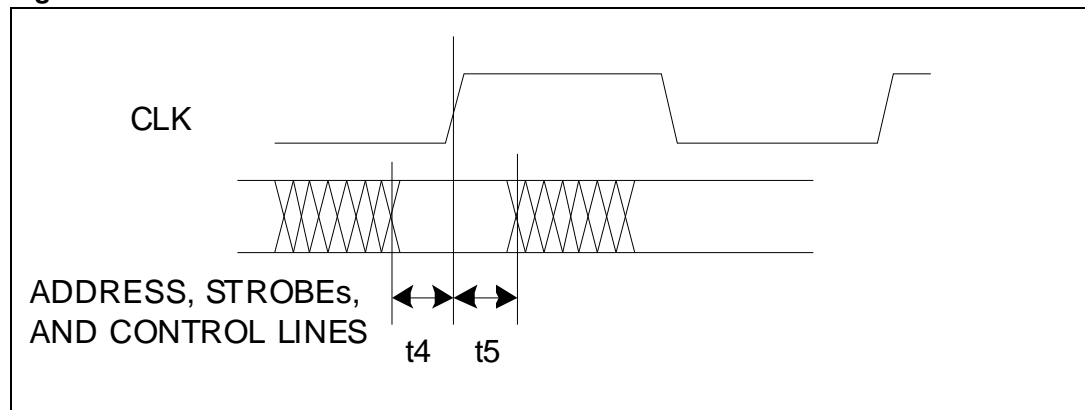
Figure 9.   Command path
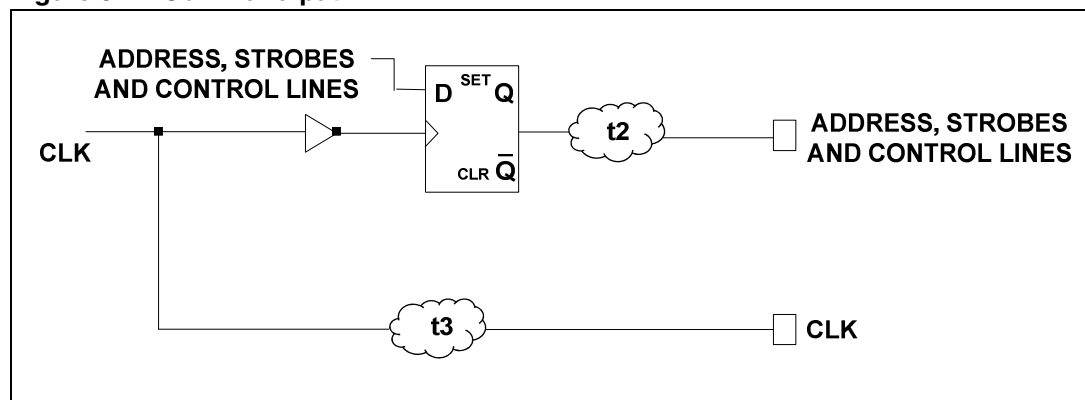
Table 26.   Command timings

Frequency	t4 max	t5 max	Unit
333 MHz	1.39	1.40	ns
266 MHz	1.77	1.78	ns
200 MHz	2.39	2.40	ns
166 MHz	2.90	2.91	ns
133 MHz	3.65	3.66	ns

5.7   CLCD timing characteristics

The characterization timing is done considering an output load of 10 pF on all the outputs.The operating conditions are in worst case V=0.90 V T=125 °C and in best case V =1.10 V T= 40° C.

The CLCD has a wide variety of configurations and setting and the parameters change accordingly. Two main scenarios will be considered, one with direct clock to output (166 MHz), setting BCD bit to '1', and the second one with the clock passing through a clock divider (83 MHz), setting BCD bit to '0'.

5.7.1   CLCD timing characteristics direct clock

Figure 10.   CLCD waveform with CLCP direct
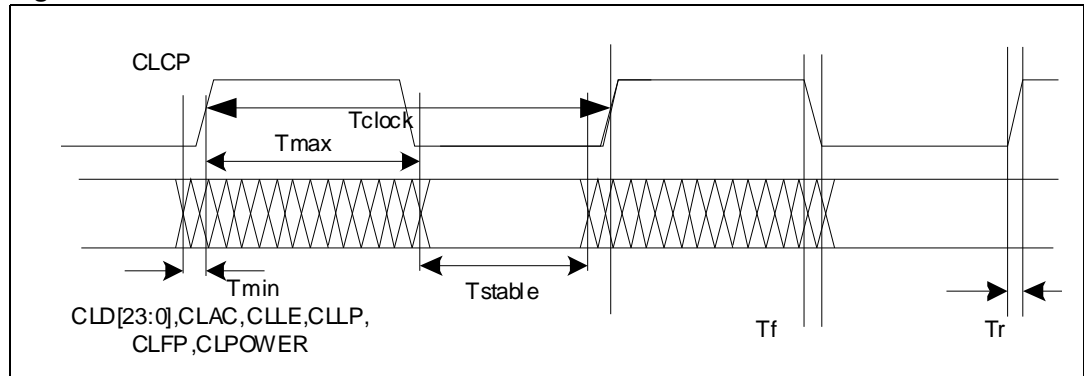
Figure 11.   CLCD block diagram with CLCP direct
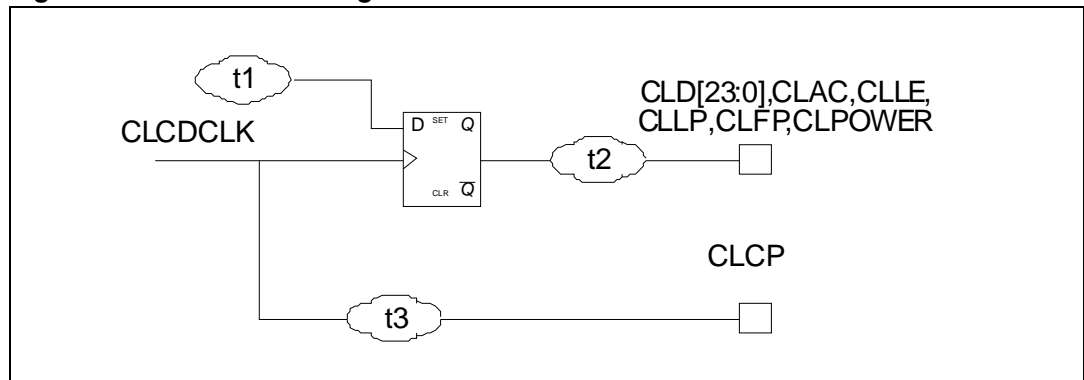
Table 27.   CLCD timings with CLCP direct

Parameter	Value	Frequency
t_{CLOCK} direct max (t_{CLOCK})	6 ns	166 MHz
t_{CLOCK} direct max rise (t_r)	0.81 ns	
t_{CLOCK} direct max (t_f)	0.87 ns	
t_{min}	-0.04 ns	
t_{max}	3.62 ns	
t_{STABLE}	2.34 ns	

www.DataSheet4U.com

Note: 1 $t_{STABLE} = t_{CLOCK}$ direct max - ($t_{max} + t_{min}$)

2   For t_{max} the maximum value is taken from the worst case and best case, while for t_{min} the minimum value is taken from the worst case and best case.

3   CLCP should be delayed by {t_{max} + [t_{CLOCK} direct max - ($t_{max} + t_{min}$)]/2} = 4.7915 ns

5.7.2 CLCD timing characteristics divided clock

Figure 12. **CLCD waveform with CLCP divided**
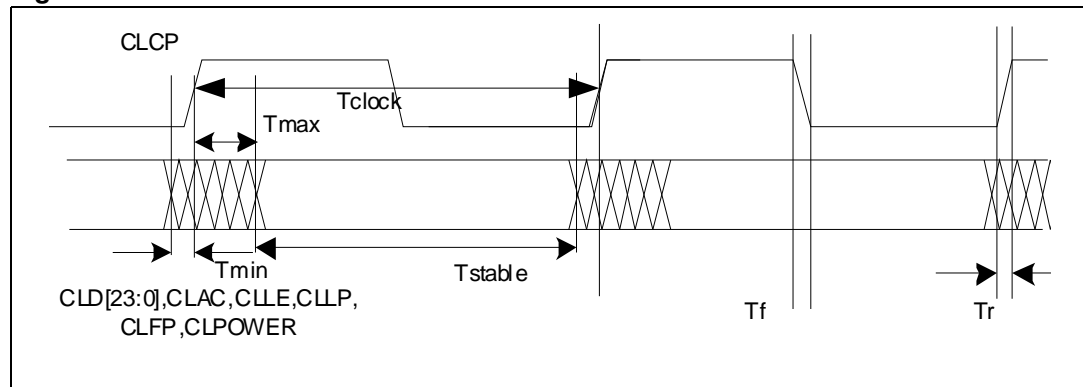
Figure 13. **CLCD block diagram with CLCP divided**
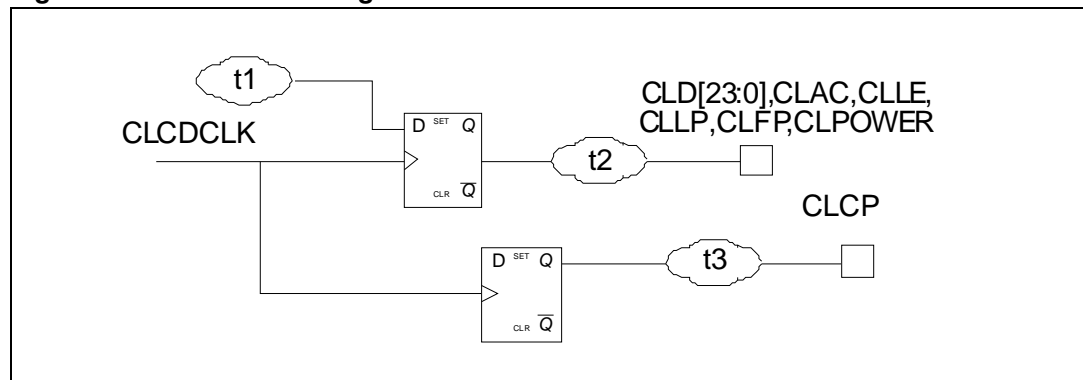
Table 28. **CLCD timings with CLCP divided**

Parameter	Value	Frequency
t_{CLOCK} divided max	12 ns	83.3 MHz
t_{CLOCK} divided max rise (t_r)	0.81 ns	
t_{CLOCK} divided max (t_f)	0.87 ns	
t_{min}	-0.49 ns	
t_{max}	2.38 ns	
t_{STABLE}	9.13 ns	

www.DataSheet4U.com

Note: 1 $t_{STABLE} = t_{CLOCK}$ direct max - (tmax + tmin)

2 For t_{max} the maximum value is taken from the worst case and for t_{min} the minimum value is taken from the best case.

3 CLCP should be delayed by $\{t_{max} + [t_{CLOCK}$ direct max - $(t_{max} + t_{min})]/2\}$ = 6.945 ns

5.8   I^{2}C timing characteristics

The characterization timing is done using primetime considering an output load of 10 pF on SCL and SDA. The operating conditions are V = 0.90 V, T_A=125° C in worst case and V =1. 10 V, T_A= 40° C in best case.

Figure 14.   I^{2}C output pins

D   Set   Q   SCL

HCLK

Clr $\overline{Q}$

D   Set   Q   SDA

Clr $\overline{Q}$

Figure 15.   I^{2}C input pins

SCL

D SET Q   D SET Q

CLR $\overline{Q}$ CLR $\overline{Q}$

HCLK

SDA

D SET Q   D SET Q

CLR $\overline{Q}$ CLR $\overline{Q}$
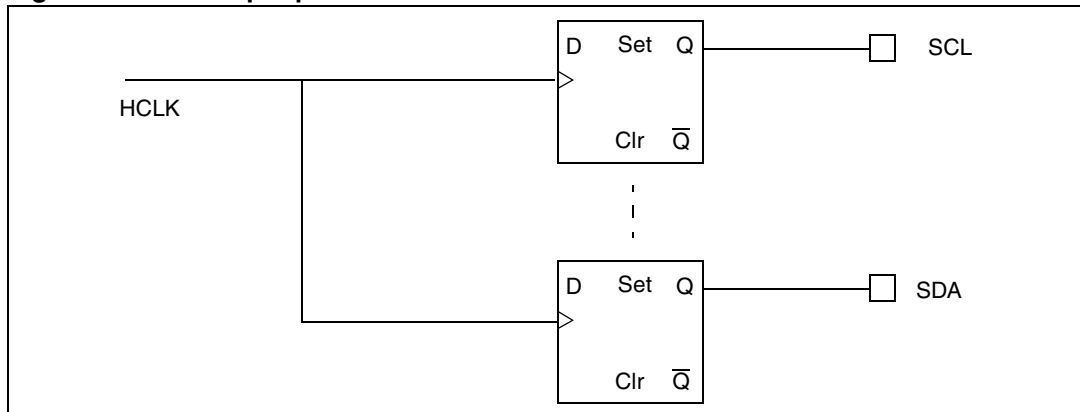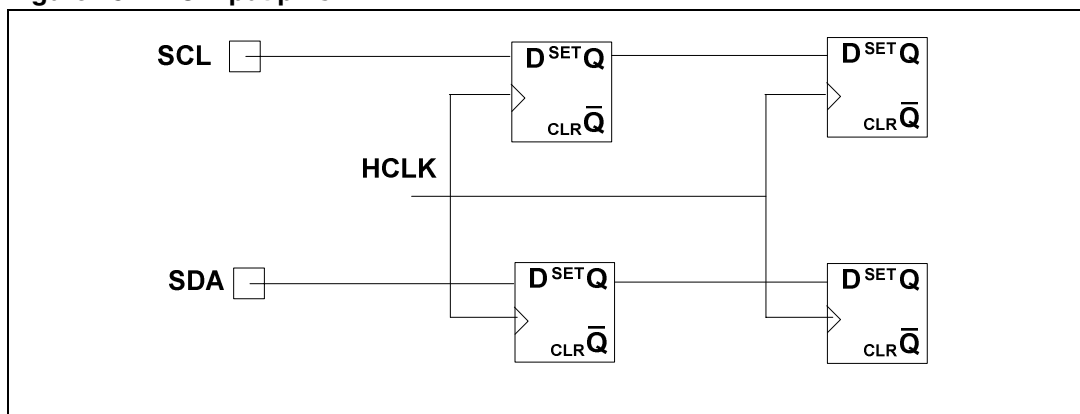
www.DataSheet4U.com

The flip-flops used to capture the incoming signals are re-synchronized with the AHB clock: so, no input delay calculation is required.

Table 29.   Output delays for I^{2}C signals

Parameter	Min	Max	Unit
$t_{HCLK^->SCLH}$	8.1067	11.8184	ns
$t_{HCLK^->SCLL}$	7.9874	12.6269	ns
$t_{HCLK^->SDAH}$	7.5274	11.2453	ns
$t_{HCLK^->SDAL}$	7.4081	12.0530	ns

Those values are referred to the common internal source clock which has a period of:

t_{HCLK} = 6 ns.

Figure 16.   Output signal waveforms for I²C signals
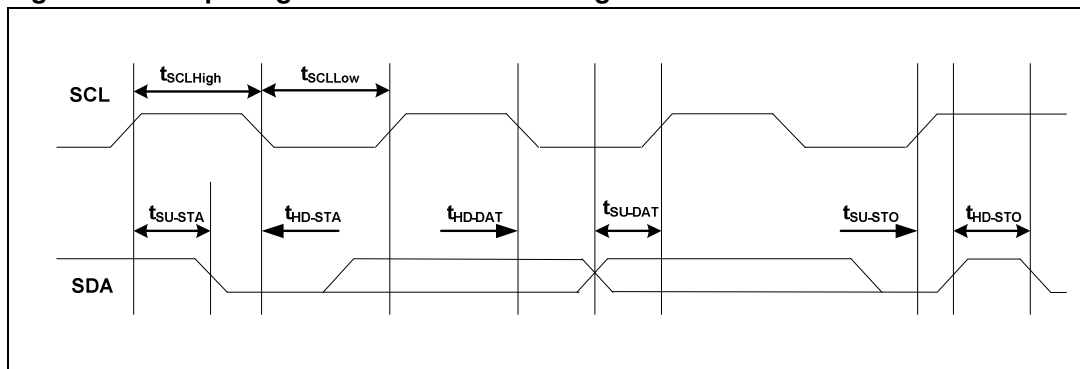
The timing of high and low level of SCL ($t_{SCLHigh}$ and t_{SCLLow}) are programmable.

Table 30.   Time characteristics for I²C in high-speed mode

Parameter	Min	Unit
t_{SU-STA}	157.5897	ns
t_{HD-STA}	325.9344	
t_{SU-DAT}	314.0537	
t_{HD-DAT}	0.7812	
t_{SU-STO}	637.709	
t_{HD-STO}	4742.1628	

Table 31.   Time characteristics for I²C in fast speed mode

Parameter	Min	Unit
t_{SU-STA}	637.5897	ns
t_{HD-STA}	602.169	
t_{SU-DAT}	1286.0537	
t_{HD-DAT}	0.7812	
t_{SU-STO}	637.709	
t_{HD-STO}	4742.1628	

Table 32.   Time characteristics for I²C in standard speed mode

Parameter	Min	Unit
t_{SU-STA}	4723.5897	ns
t_{HD-STA}	3991.9344	
t_{SU-DAT}	4676.0537	
t_{HD-DAT}	0.7812	
t_{SU-STO}	4027.709	
t_{HD-STO}	4742.1628	

Note: 1 *The timings shown in Figure 16. depend on the programmed value of $T_{SCLHigh}$ and T_{SCLLow}, so the values present in the three tables here above have been calculated using the minimum programmable values of :*

   IC_HS_SCL_HCNT=19 and IC_HS_SCL_LCNT=53 registers (for High-Speed mode);

   IC_FS_SCL_HCNT=99 and IC_FS_SCL_LCNT=215 registers (for Fast-Speed mode);

   IC_SS_SCL_HCNT=664 and IC_SS_SCL_LCNT=780 registers (for Standard-Speed mode).

   These minimum values depend on the AHB clock frequency, which is 166 MHz.

2 *A device may internally require a hold time of at least 300 ns for the SDA signal (referred to the V_{IHmin} of the SCL signal) to bridge the undefined region of the falling edge of SCL (Please refer to the I^2C Bus Specification v3-0 Jun 2007). However, the SDA data hold time in the I^2C controller of SPEAr320 is one-clock cycle based (6 ns with the HCLK clock at 166 MHz). This time may be insufficient for some slave devices. A few slave devices may not receive the valid address due to the lack of SDA hold time and will not acknowledge even if the address is valid. If the SDA data hold time is insufficient, an error may occur.*

3 **Workaround:** *If a device needs more SDA data hold time than one clock cycle, an RC delay circuit is needed on the SDA line as illustrated in the following figure:*

Figure 17.   RC delay circuit
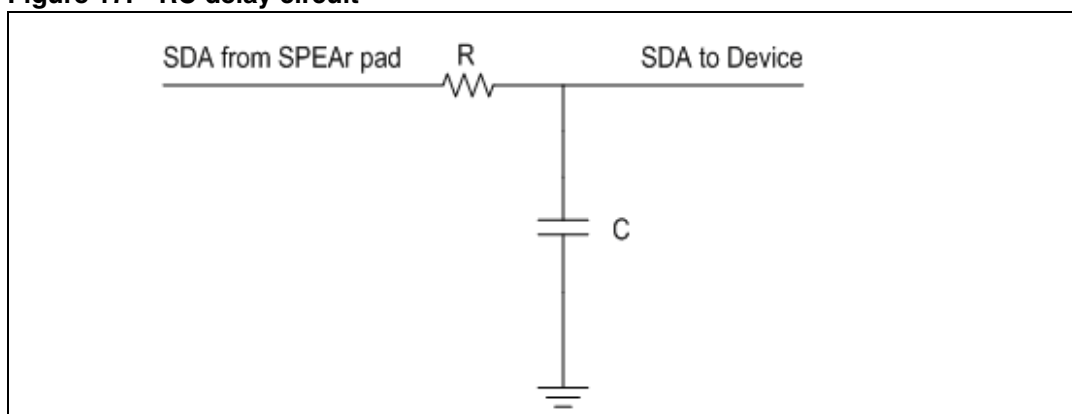
For example, R=K and C = 200 pF.

5.9   FSMC timing characteristics

The characterization timing is done using primetime considering an output load of 3pF on the data, 15pF on NF_CE, NF_RE and NF_WE and 10pF on NF_ALE and NF_CLE.

The operating conditions are V=0.90V, T=125°C in worst case and V=1.10V, T= 40°C in best case.

5.9.1   8-bit NAND Flash configuration

Figure 18.   Output pads for 8-bit NAND Flash configuration
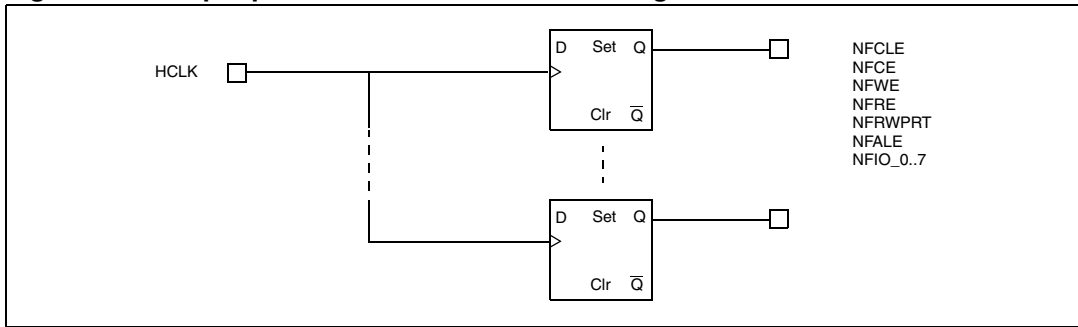
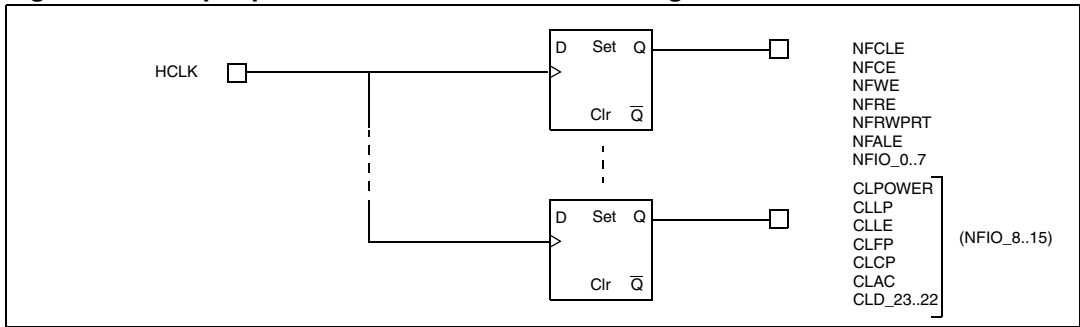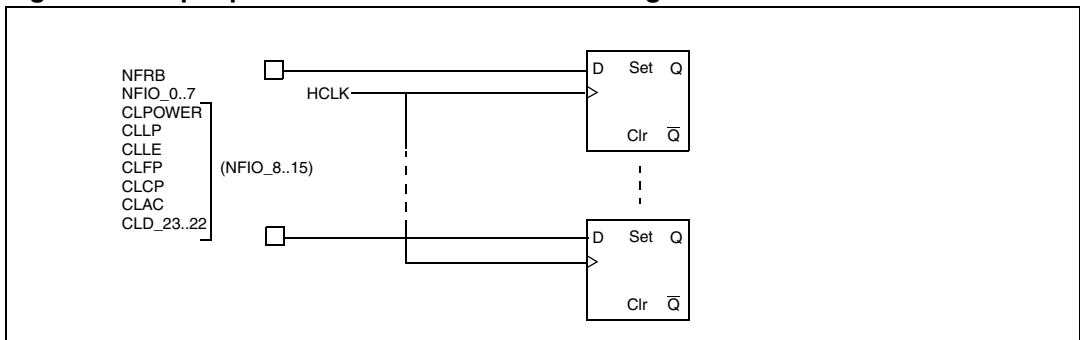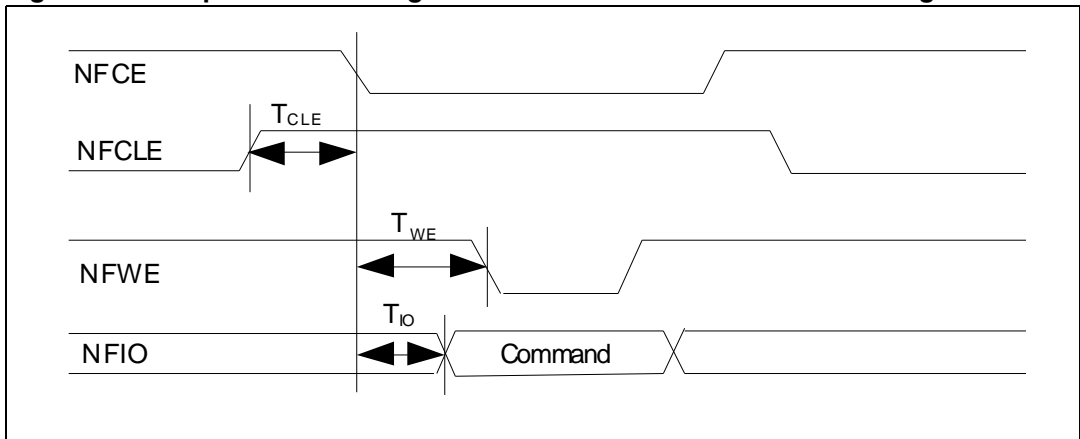**Figure 19.   Input pads for 8-bit NAND Flash configuration**
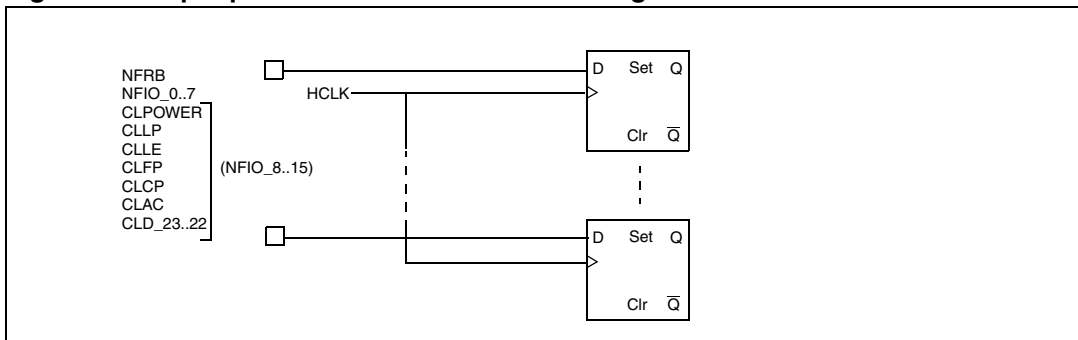
Figure 20.   Output command signal waveforms for 8-bit NAND Flash configuration
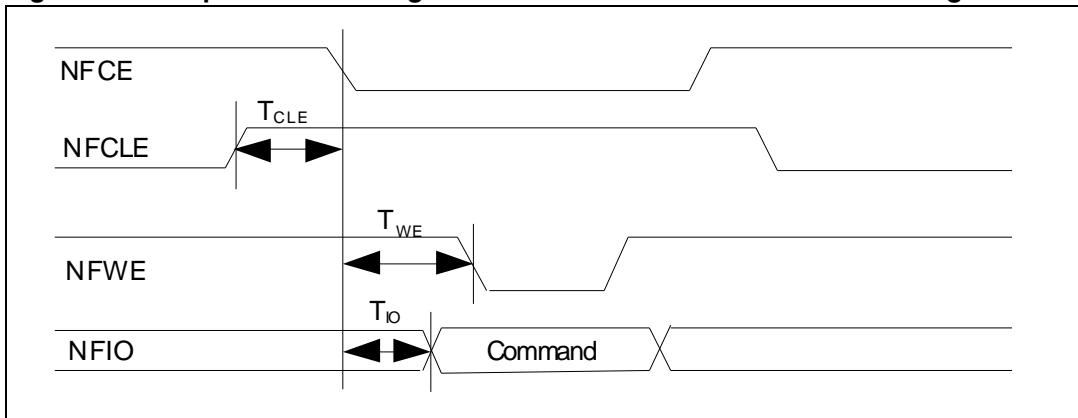
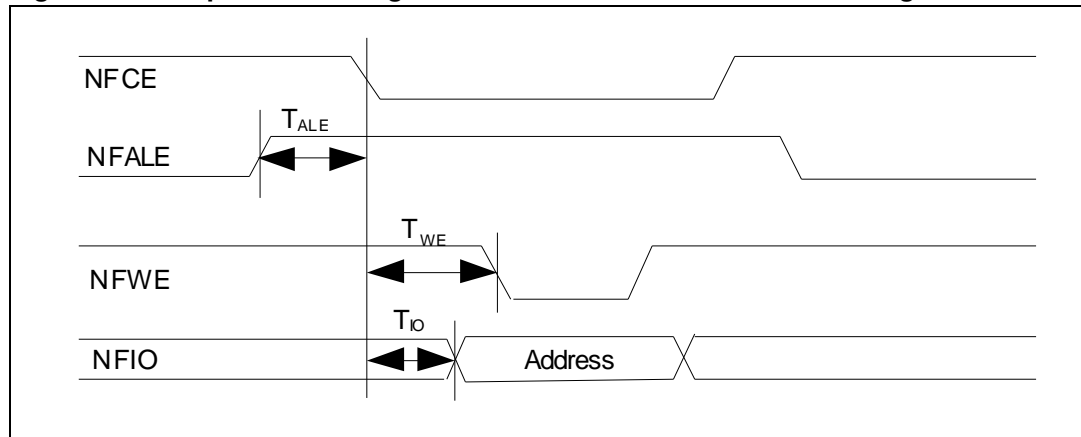**Figure 21. Output address signal waveforms for 8-bit NAND Flash configuration**
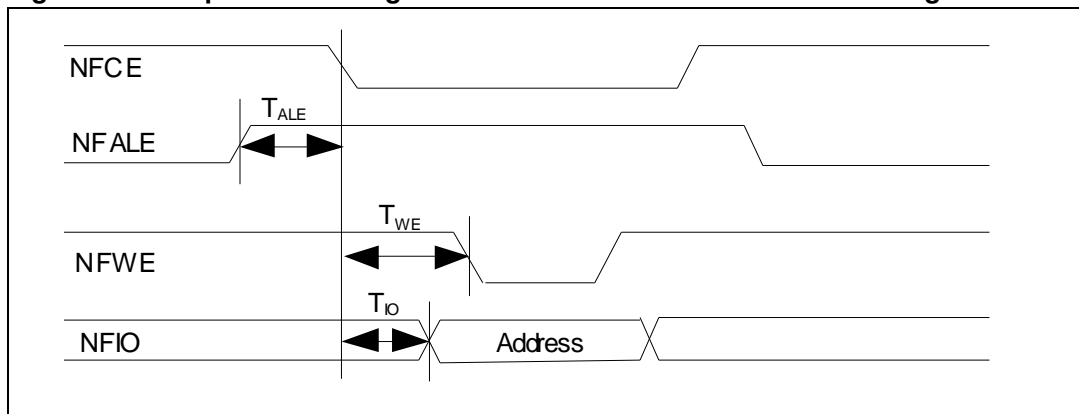
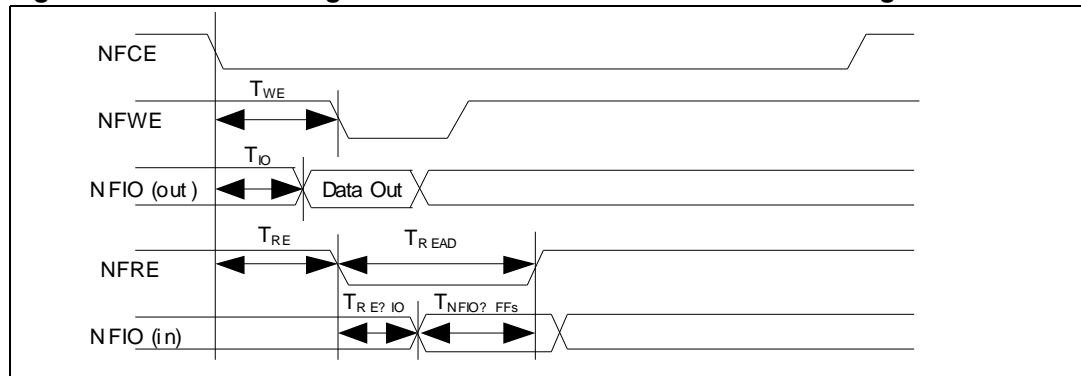**Figure 22. In/out data address signal waveforms for 8-bit NAND Flash configuration**
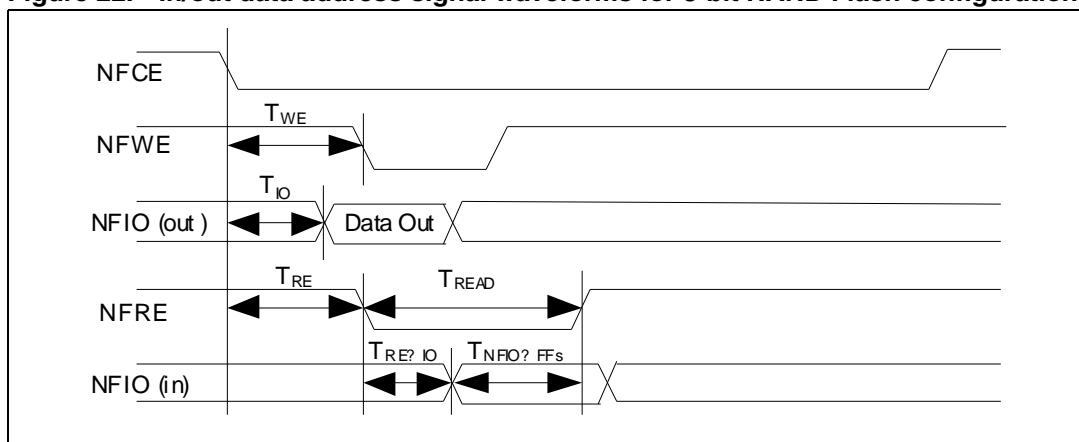
Table 33. Time characteristics for 8-bit NAND Flash configuration

Parameter	Min	Max
TCLE	-16.85 ns	-19.38 ns
TALE	-16.84 ns	-19.37 ns
TWE (s=1)	11.10 ns	13.04 ns
TRE (s=1)	11.18 ns	13.05 ns
TIO (h=1)	3.43 ns	8.86 ns

Note: Values in Table 33 are referred to the common internal source clock which has a period of THCLK = 6 ns.

5.9.2 16-bit NAND Flash configuration

Figure 23. Output pads for 16-bit NAND Flash configuration

Figure 24. Input pads for 16-bit NAND Flash configuration

Figure 25. Output command signal waveforms 16-bit NAND Flash configuration

Figure 26. Output address signal waveforms 16-bit NAND Flash configuration

Figure 27. In/out data signal waveforms for 16-bit NAND Flash configuration

Table 34. Time characteristics for 16-bit NAND Flash configuration

Parameter	MIN	MAX
TCLE	-16.85 ns	-19.38 ns
TALE	-16.84 ns	-19.37 ns
TWE (s=1)	11.10 ns	13.04 ns
TRE (s=1)	11.18 ns	13.05 ns
TIO (h=1)	3.27 ns	11.35 ns

www.DataSheet4U.com

Note: *Values in Table 34 are referred to the common internal source clock which has a period of THCLK = 6 ns.*

5.10 Ether MAC 10/100/1000 Mbps (GMAC-Univ) timing characteristics

The characterization timing is given for an output load of 5 pF on the GMII TX clock and 10 pF on the other pads. The operating conditions are in worst case V=0.90 V T=125° C and in best case V=1.10 V T= 40° C.

5.10.1   GMII Transmit timing specifications

Figure 28.   GMII TX waveforms
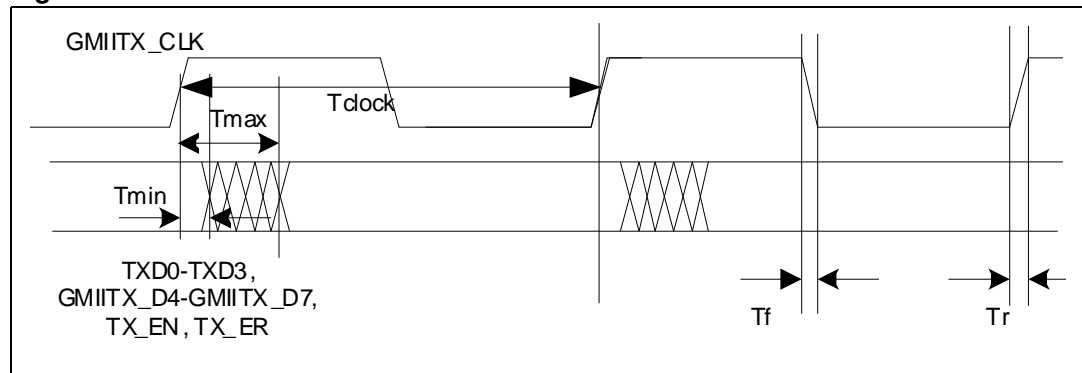
Figure 29.   Block diagram of GMII TX pins
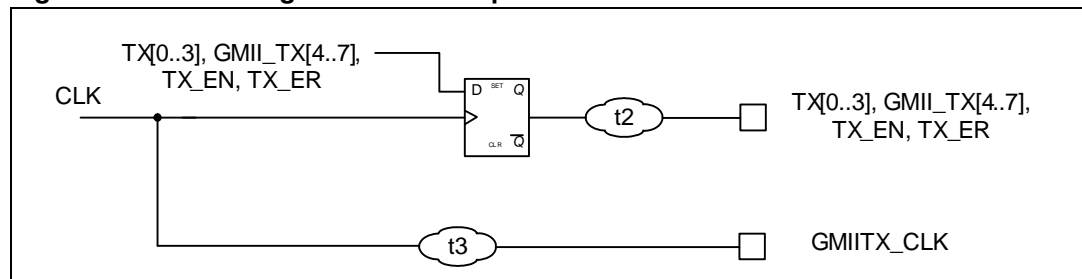
Table 35.   GMII TX timing

Parameter	Value using GMII [t_{CLK} period = 8 ns 125 MHz]
t_{rise} (t_r)	<1 ns
t_{fall} (t_f)	<1 ns
t_{max}= $t2_{max}$- $t3_{min}$	2.8 ns
t_{min}= $t2_{min}$- $t3_{max}$	0.4 ns
t_{SETUP}	5.19 ns

Note: *To calculate the t_{SETUP} value for the PHY you have to consider the next t_{CLK} rising edge, so you have to apply the following formula: $t_{SETUP} = t_{CLK} - t_{max}$*

## 5.10.2   MII transmit timing specifications

Figure 30.   MII TX waveforms
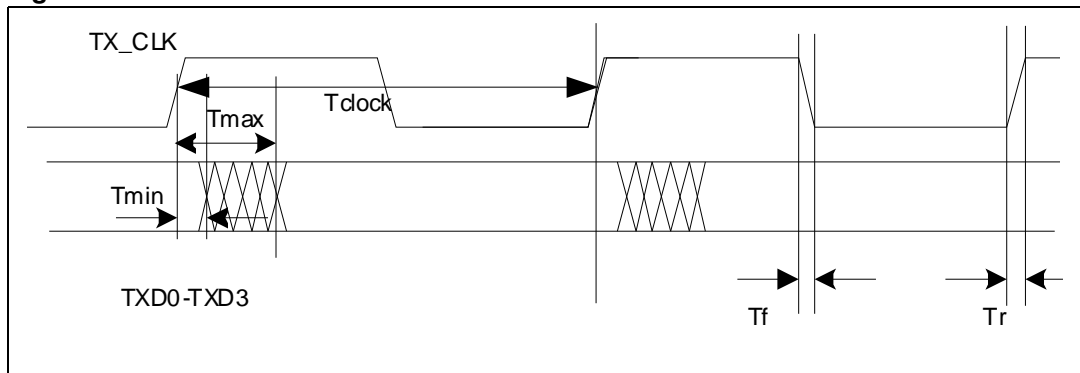
Figure 31.   Block diagram of MII TX pins
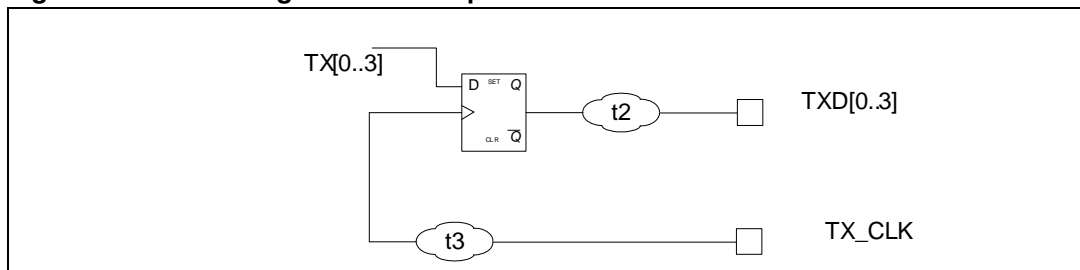
Table 36.   MII TX timings

Parameter	Value using MII 10 Mb [t_{CLK} period = 40 ns 25 MHz]	Value using MII 100 Mb [t_{CLK} period = 400 ns 2.5 MHz]
$t_{max} = t2_{max} - t3_{min}$	6.8 ns	6.8 ns
$t_{min} = t2_{min} - t3_{max}$	2.9 ns	2.9 ns
t_{SETUP}	33.2 ns	393.2 ns

Note: *To calculate the t_{SETUP} value for the PHY you have to consider the next t_{CLK} rising edge, so you have to apply the following formula: $t_{SETUP} = t_{CLK} - t_{max}$*

5.10.3 GMII-MII Receive timing specifications
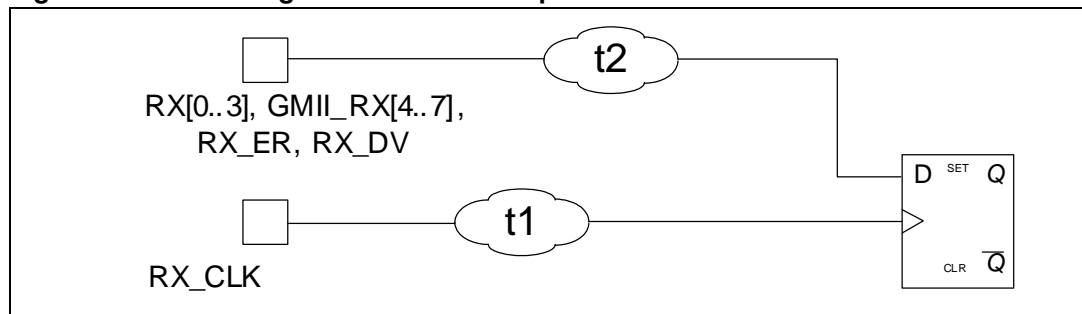
Figure 32.   GMII-MII RX waveforms
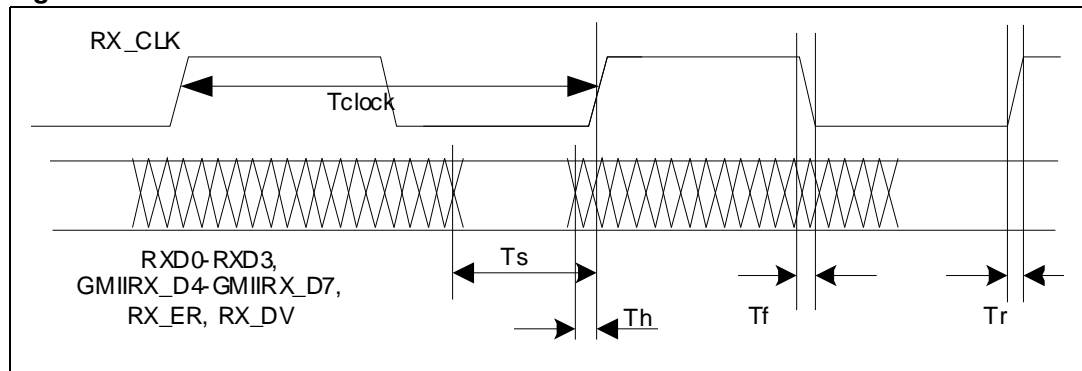
Figure 33.   Block diagram of GMII-MII RX pins

Note: *The input stage is the same for all the interfaces (GMII and MII10/100) so t_{SETUP} and t_{HOLD} values are equal in all the cases.*

The receive path is optimized for the GMII interface: this also ensures correct capture of data for the MII10/100 interface.

5.10.4 MDIO timing specifications

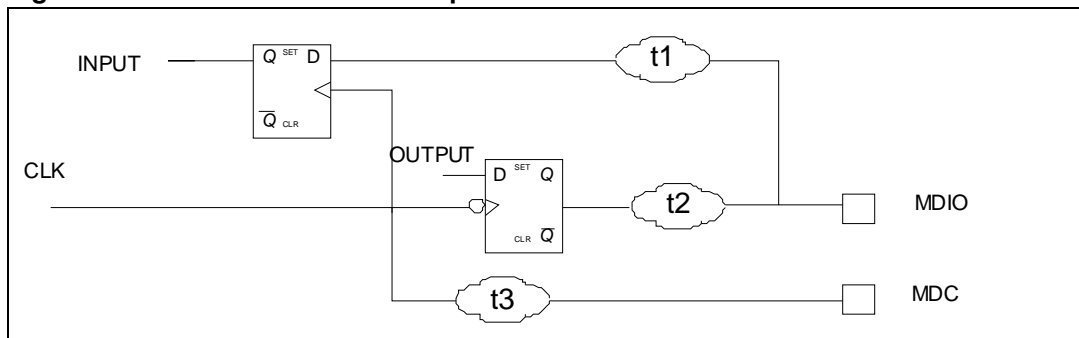**Figure 34.   MDC waveforms**
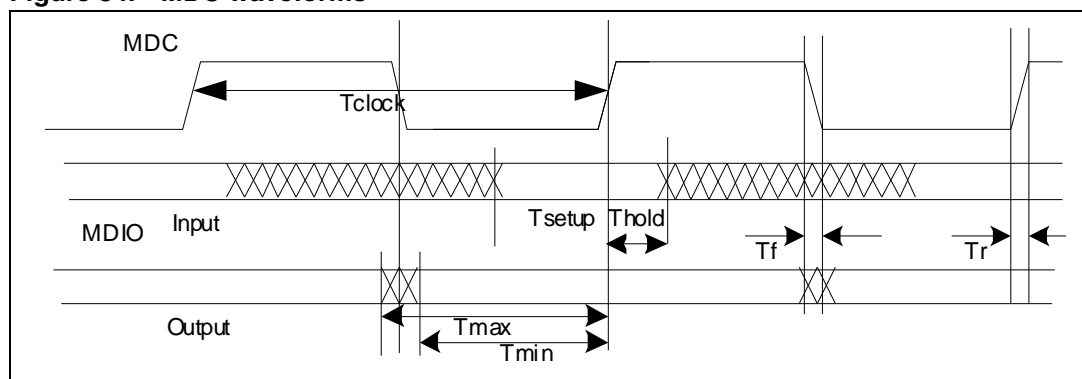
Figure 35. Paths from MDC/MDIO pads

Table 37. MDC/MDIO timing

Parameter	Value	Frequency
t_{CLK} period	614.4 ns	1.63 MHz
t_{CLK} fall (t_f)	1.18 ns	
t_{CLK} rise (t_r)	1.14 ns	
Output		
$t_{max} = \sim t_{CLK}/2$	307 ns	
$t_{min} = \sim t_{CLK}/2$	307 ns	
Input		
$t_{SETUPmax} = t1_{max} - t3_{min}$	6.88 ns	
$t_{HOLDmin} = t1_{min} - t3_{max}$	-1.54 ns	

Note: *When MDIO is used as output the data are launched on the falling edge of the clock as shown in Figure 34.*

5.11   SMI - Serial memory interface
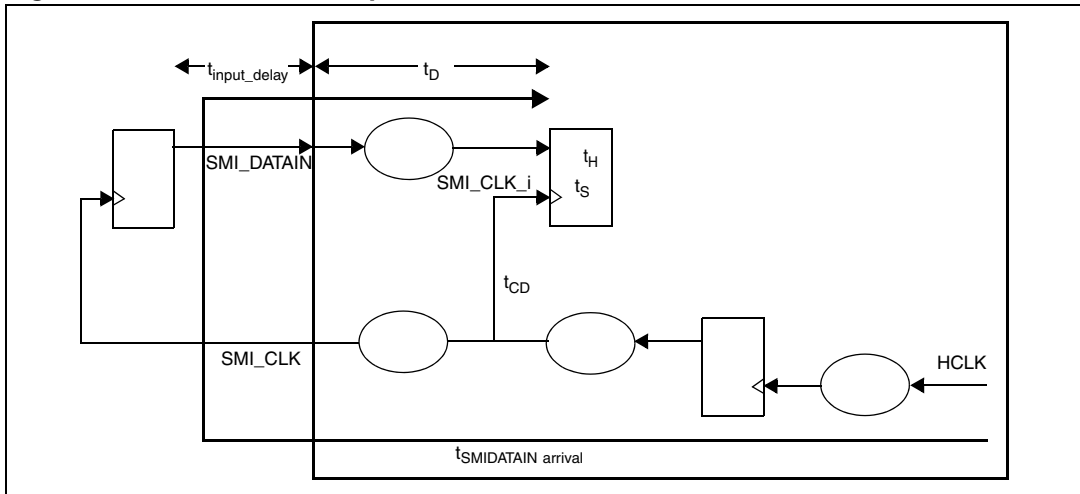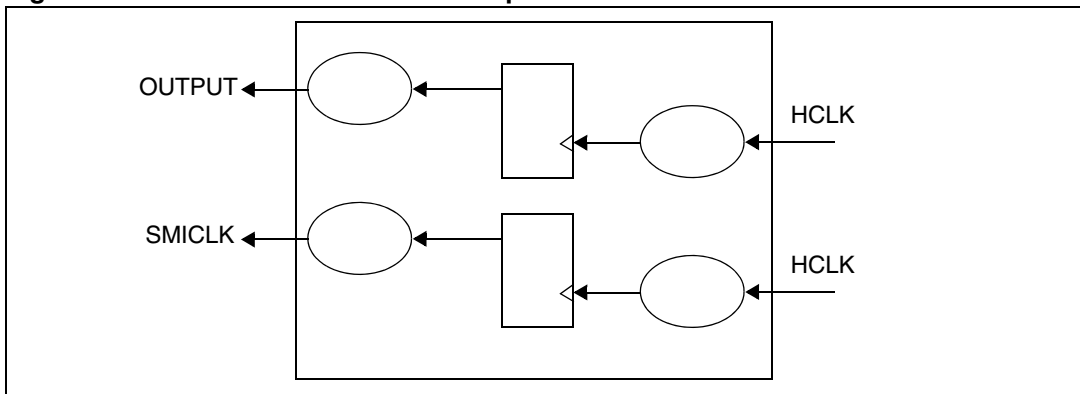
Figure 36.   SMIDATAIN data path

Table 38.   SMIDATAIN timings

Signal	Parameter	Value
SMI_DATAIN	t_{d_max}	$t_{SMIDATAIN\_arrival\_max} - t_{input\_delay}$
	t_{d_min}	$t_{SMIDATAIN\_arrival\_min} - t_{input\_delay}$
	t_{cd_min}	$t_{SMI\_CLK\_i\_arrival\_min}$
	t_{cd_max}	$t_{SMI\_CLK\_i\_arrival\_max}$
	t_{SETUP_max}	$t_s + t_{d\_max} - t_{cd\_min}$
	t_{HOLD_min}	$t_h - t_{d\_min} + t_{cd\_max}$

Figure 37.   SMIDATAOUT/SMICSn data paths
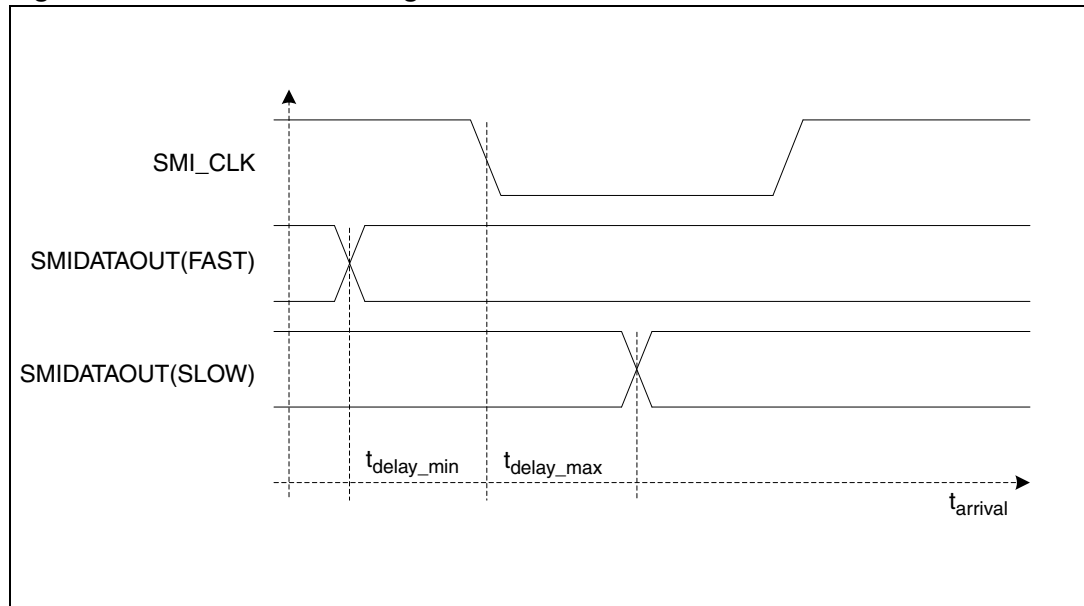
Figure 38. SMIDATAOUT timings

Table 39. SMIDATAIN timings

Signal	Parameter	Value
SMI_DATAOUT	t_{delay_max}	$t_{arrivalSMIDATAOUT\_max} - t_{arrival\_SMI\_CLK\_min}$
	t_{delay_min}	$t_{arrivalSMIDATAOUT\_min} - t_{arrival\_SMI\_CLK\_max}$

Figure 39. SMICSn fall timings
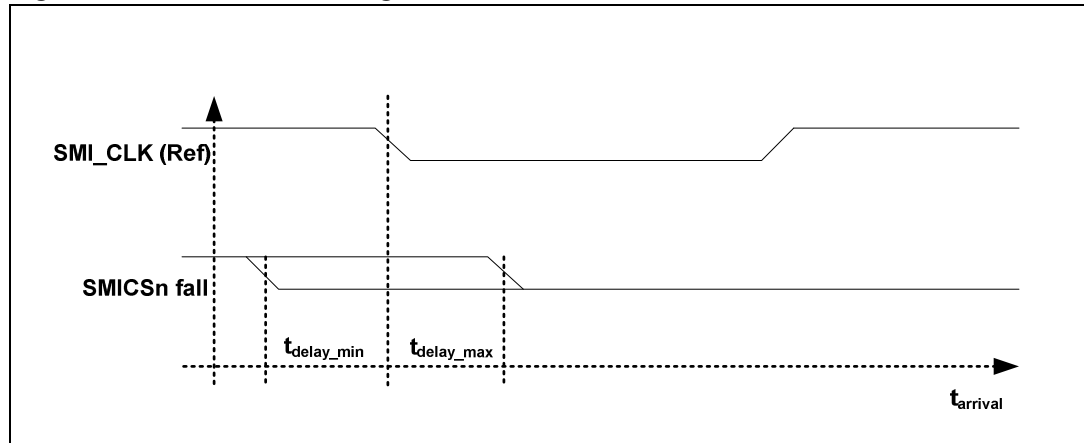
Table 40. SMICSn fall timings

Signal	Parameter	Value
SMI_CSn fall	t_{delay_max}	$t_{arrivalSMICSn\_max\_fall} - t_{arrival\_SMI\_CLK\_min\_fall}$
	t_{delay_min}	$t_{arrivalSMICSn\_min\_fall} - t_{arrival\_SMI\_CLK\_max\_fall}$

Figure 40.   SMICSn rise timings
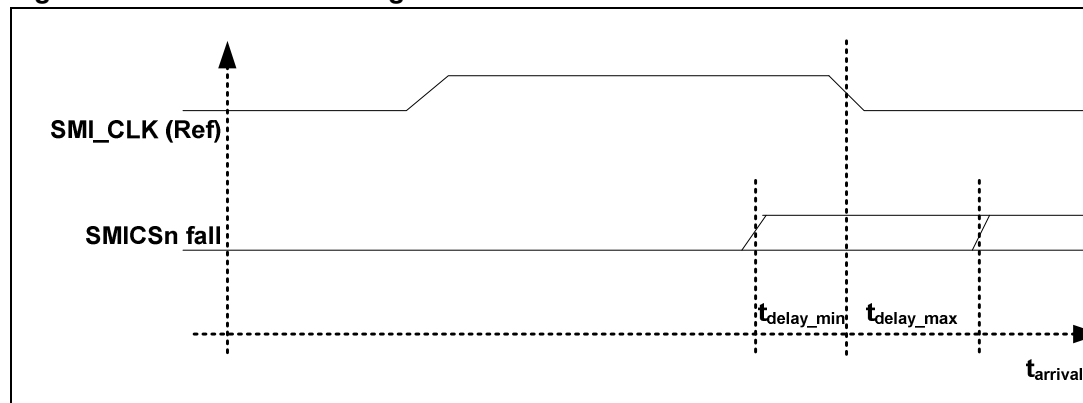
Table 41.   SMICSn rise timings

Signal	Parameter	Value
SMI_CSn rise	t_{delay_max}	$t_{arrivalSMICSn\_max\_rise} - t_{arrival\_SMI\_CLK\_min\_fall}$
	t_{delay_min}	$t_{arrivalSMICSn\_min\_rise} - t_{arrival\_SMI\_CLK\_max\_fall}$

Table 42.   Timing requirements for SMI

Parameter		Input setup-hold/output delay	
		Max	**Min**
SMI_CLK	Fall time	1.8209	1.4092
	Rise time	1.6320	1.1959
SMIDATAIN	Input setup time	8.27482	
	Input hold time	-2.595889	
SMIDATAOUT Output valid time		2.039774	
SMICS_0 Output valid time	fall	1.922779	
	rise	1.69768	
SMICS_1 Output valid time	fall	1.7898169	
	rise	1.638069	

5.12   SPI

This module provides a programmable length shift register which allows serial communication with other SPI devices through a 3 or 4 wire interface (SPI_SCK, MISO, MOSI and SPI_CSn).
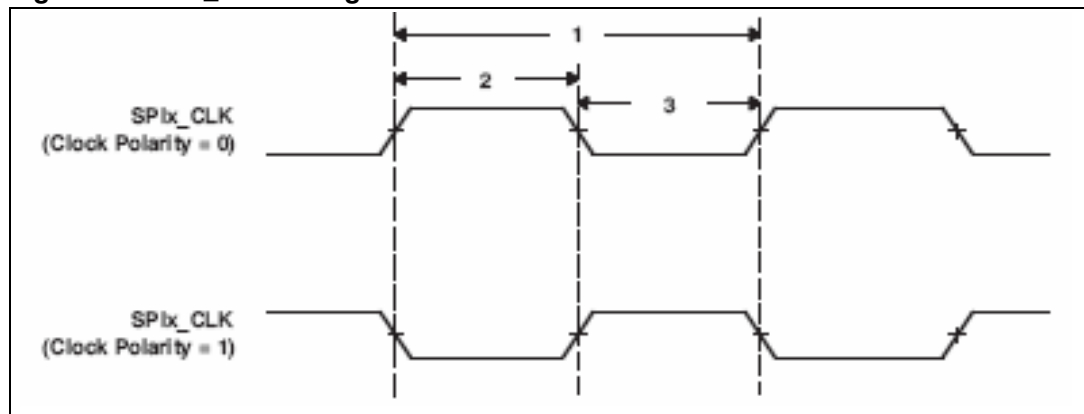
Figure 41. SPI_CLK timings

Table 43. SPI timing requirements (all modes)

No.	Parameters		Min	Max	Unit
1	$t_{c(CLK)}$	Cycle time, SPI_SCK	24		ns
2	$t_{w(CLKH)}$	Pulse duration, SPI_SCK high	$0.49*t_{c(CLK)}$ - $0.51*t_{c(CLK)}$		ns
3	$t_{w(CLKL)}$	Pulse duration, SPI_SCK low	$0.51t_{c(CLK)}$ - $0.49*t_{c(CLK)}$		ns

5.12.1 SPI master mode timings (Clock phase = 0)

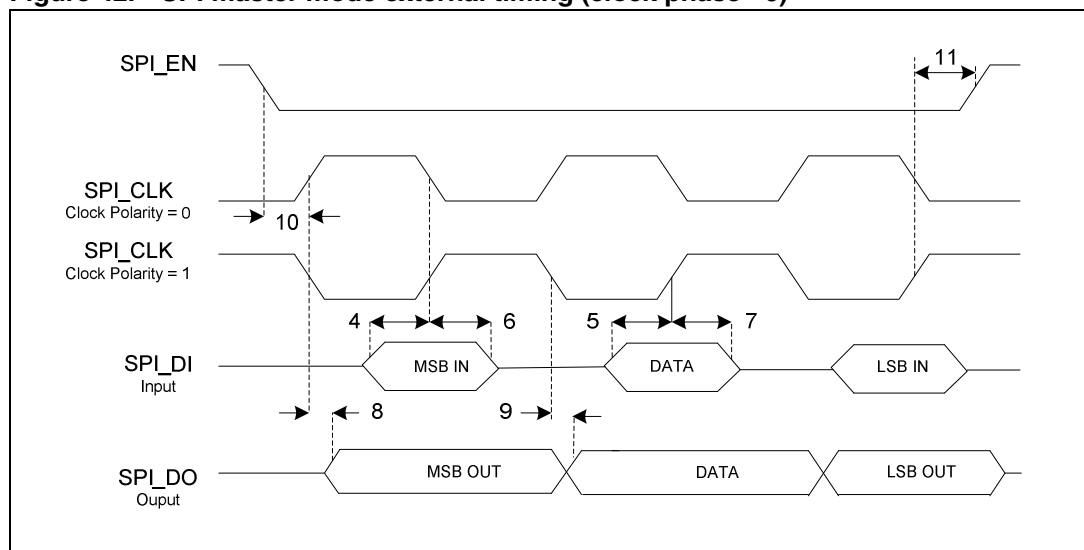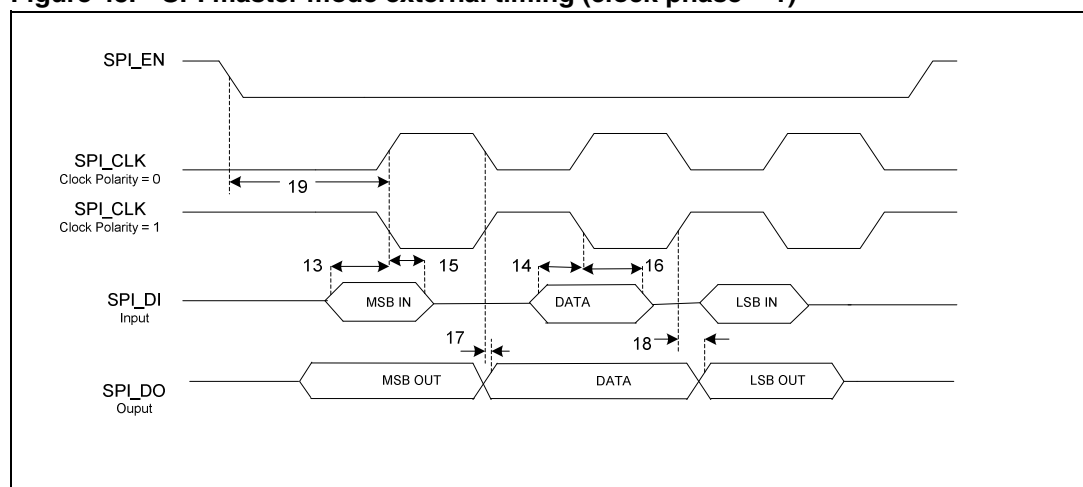**Figure 42. SPI master mode external timing (clock phase =0)**

Table 44.   Timing Requirements for SPI Master Mode [Clock Phase = 0]

No.		Parameters		Max.	Unit
4	$t_{su(DIV-CLKL)}$	Setup time, MISO (input) valid before SPI_SCK (output) falling edge	Clock Polarity = 0	11.832	ns
5	$t_{su(DIV-CLKH)}$	Setup time, MISO (input) valid before SPI_SCK (output) rising edge	Clock Polarity = 1	11.950	ns
6	$t_{h(CLKL-DIV)}$	Hold time, MISO (input)valid after SPI_SCK (output) falling edge	Clock Polarity = 0	-7.690	ns
7	$t_{h(CLKH-DIV)}$	Hold time, MISO (input) valid after SPI_SCK (output) rising edge	Clock Polarity = 1	-7.958	ns

Table 45.   Switching characteristics over recommen ded operating conditions for SPI master mode (clock phase = 0)

No.		Parameters		Max	Unit
8	$t_{d(CLKH-DOV)}$	Delay time, SPI_SCK (output) rising edge to MOSI (output) transition	Clock Polarity = 0	1.960	ns
9	$t_{d(CLKL-DOV)}$	Delay time, SPI_SCK (output) falling edge to MOSI (output) transition	Clock Polarity = 1	21.75	ns
10	$t_{d(ENL-CLKH/L)}$	Delay time, SPI_CSn (output) falling edge to first SPI_SCK (output) rising or falling edge		T	ns
11	$t_{d(CLKH/L-ENH)}$	Delay time, SPI_SCK (output) rising or falling edge to SPI_CSn (output) rising edge		T/2	ns

5.12.2   SPI master mode timings (Clock Phase = 1)

Figure 43.   SPI master mode external timing (clock phase = 1)

Table 46.   Timing requirements for SPI master mode (clock phase = 1)

No.		Parameters		Max	Unit
12	$t_{su(DIV\text{-}CLKL)}$	Setup time, MISO (input) valid before SPI_SCK (output) rising edge	Clock polarity = 0	11.950	ns
13	$t_{su(DIV\text{-}CLKH)}$	Setup time, MISO (input) valid before SPI_SCK (output) falling edge	Clock polarity = 1	11.832	ns
14	$t_{h(CLKL\text{-}DIV)}$	Hold time, MISO (input) valid after SPI_SCK (output) rising edge	Clock polarity = 0	-7.958	ns
15	$t_{h(CLKH\text{-}DIV)}$	Hold time, MISO (input) valid after SPI_SCK (output) falling edge	Clock polarity = 1	-7.690	ns

Table 47.   Switching characteristics over recommended oeprating conditions for SPI master mode (clock phase =1)

No.		Parameters		Max	Unit
16	$t_{d(CLKH\text{-}DOV)}$	Delay time, SPI_SCK (output) rising edge to MOSI (output) transition	Clock Polarity = 0	1.960	ns
17	$t_{d(CLKL\text{-}DOV)}$	Delay time, SPI_SCK (output) falling edge to MOSI (output) transition	Clock Polarity = 1	2.175	ns
18	$t_{d(ENL\text{-}CLKH/L)}$	Delay time, SPI_CSn (output) falling edge to first SPI_SCK (output) rising or falling edge		T/2	ns
19	td(CLKH/L-ENH)	Delay time, SPI_SCK (output) rising or falling edge to SPI_CSn (output) rising edge		T	ns

5.13   UART (Universal asynchronous receiver/transmitter)
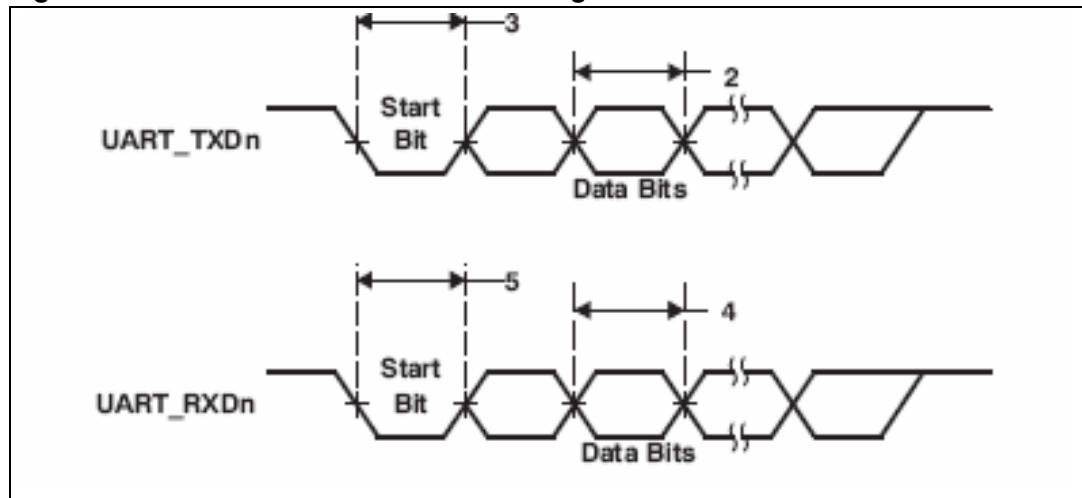
Figure 44.   UART transmit and receive timings

Table 48.   UART transmit timing characteristics

S.No.	Parameters	Min	Max	Unit
1	UART Maximum Baud Rate		3	Mbps
2	UART Pulse Duration Transmit Data (TxD)	$0.99B_{(1)}$	$B_{(1)}$	ns
3	UART Transmit Start Bit	$0.99B_{(1)}$	$B_{(1)}$	ns

Table 49.   UART receive timing characteristics

S.No.	Parameters	Min	Max	Units
4	UART Pulse Duration Receive Data (RxD)	$0.97B_{(1)}$	$1.06B_{(1)}$	ns
5	UART Receive Start Bit	$0.97B_{(1)}$	$1.06B_{(1)}$	ns

where (1) B = UART baud rate

5.14   Power up sequence

The only requirement is that the various power supplies reach the correct range in less than 10 ms.

5.15   Power on reset (MRESET)

The MRESET must remain active for at least 10 ms after all the power supplies are in the correct range and should become active in no more than 10 µs when one of the power supplies goes out of the correct range.

6   Package information

In order to meet environmental requirements, ST offers these devices in different grades of ECOPACK® packages, depending on their level of environmental compliance. ECOPACK® specifications, grade definitions and product status are available at: *www.st.com*. ECOPACK® is an ST trademark.

Table 50.   LFBGA289 (15 x 15 x 1.7 mm) mechanical data

Dim.	mm			inches		
	Min.	Typ.	Max.	Min.	Typ.	Max.
A			1.700			0.0669
A1	0.270			0.0106		
A2		0.985			0.0387	
A3		0.200			0.0078	
A4			0.800			0.0315
b	0.450	0.500	0.550	0.0177	0.0197	0.0217
D	14.850	15.000	15.150	0.5846	0.5906	0.5965
D1		12.800			0.5039	
E	14.850	15.000	15.150	0.5846	0.5906	0.5965
E1		12.800			0.5039	
e		0.800			0.0315	
F		1.100			0.0433	
ddd			0.200			0.0078
eee			0.150			0.0059
fff			0.080			0.0031

Figure 45. LFBGA289 package dimensions
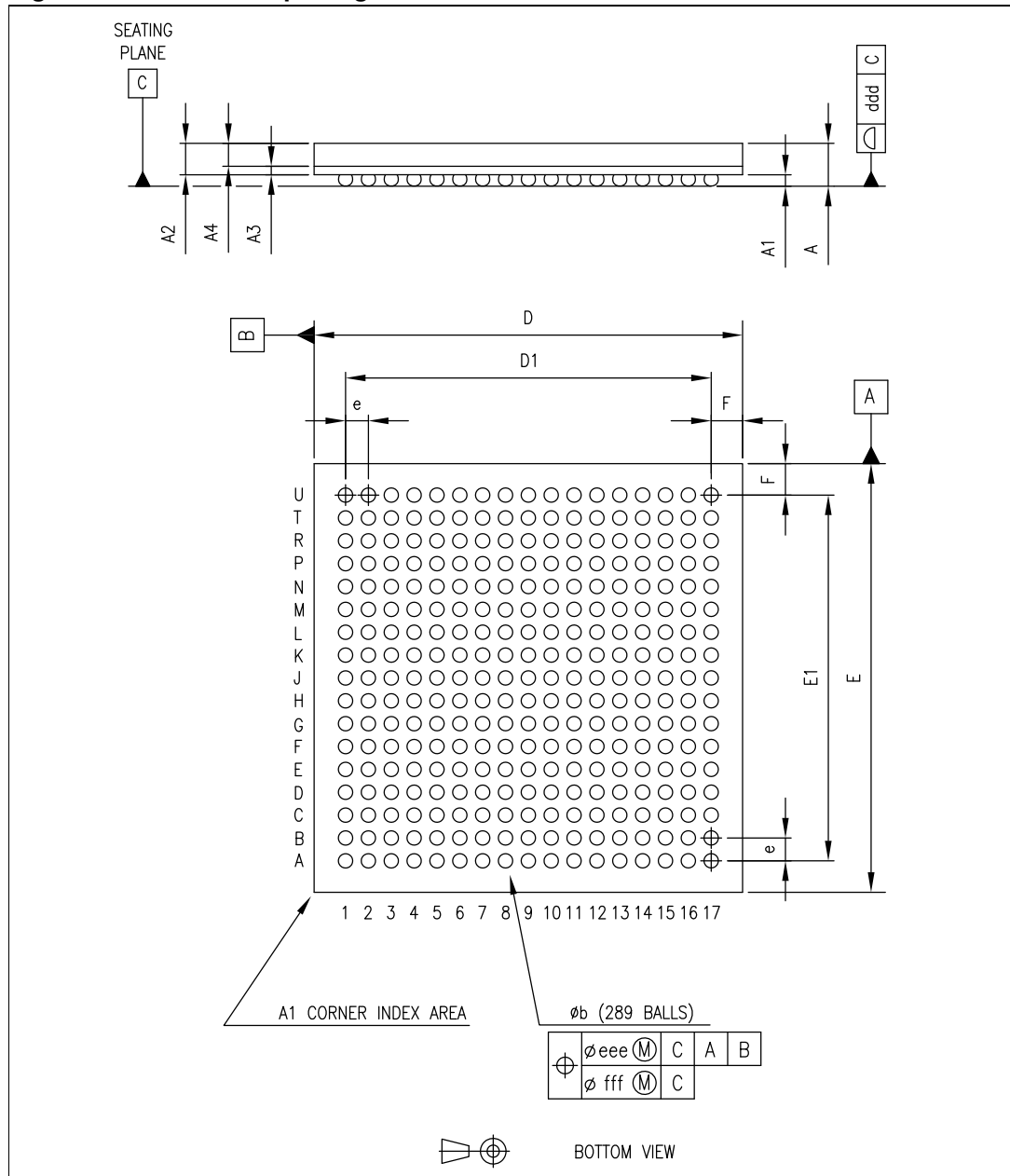
7   Revision history

Table 51.   Document revision history

Date	Revision	Changes
12-Nov-2009	1	Initial release.

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.

www.DataSheet4U.com

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.

STMicroelectronics group of companies

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

www.st.com