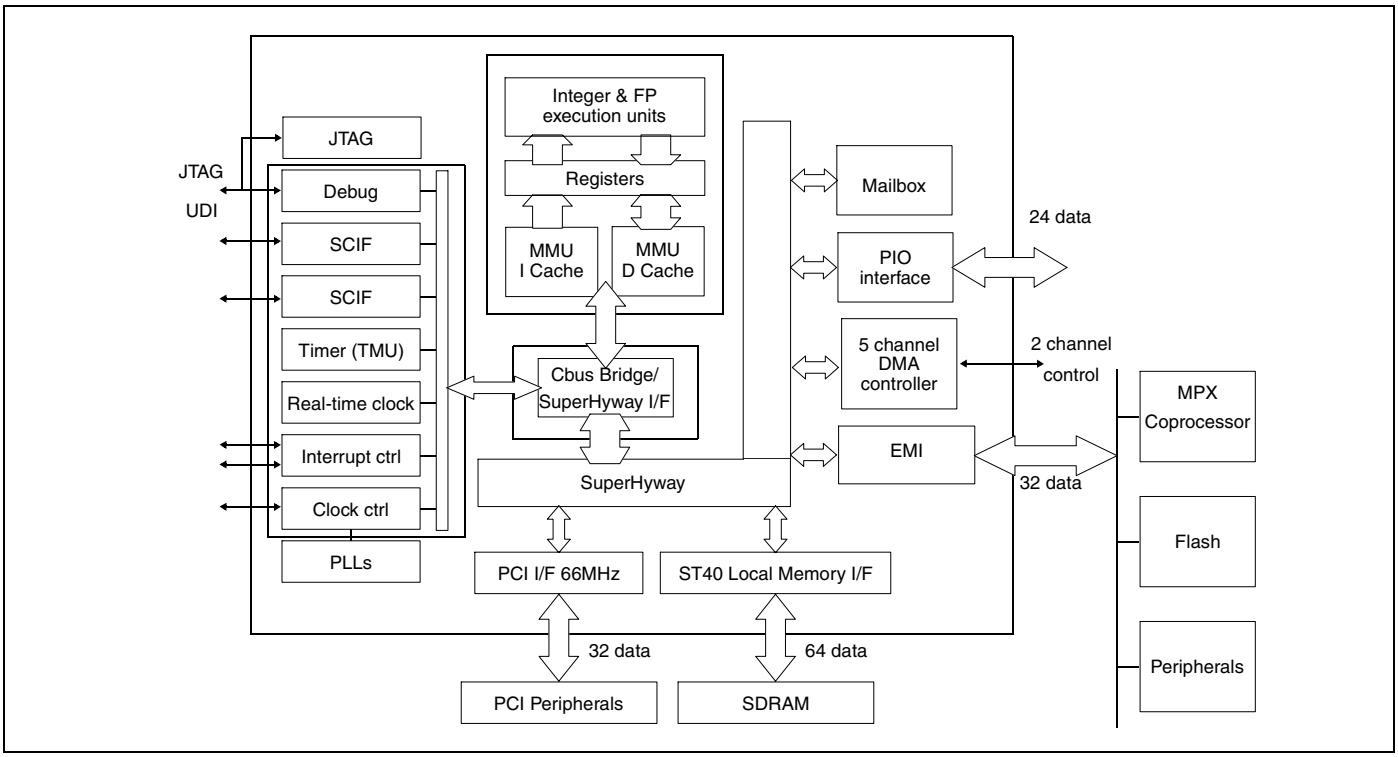
ST40RA

32-bit Embedded SuperH Device



Overview

The ST40RA is the first member of the ST40 family. Based on the SH-4, SuperH CPU core from SuperH Inc, the ST40RA is designed to work as a standalone device, or as part of a two chip solution for application specific systems.

Example applications the ST40RA is designed for include digital consumer, embedded communications, industrial and automotive. The high connectivity of the ST40 through its PCI bus and its dual memory uses makes it a versatile device, ideal for data-intensive and high performance applications.

System features

- **32-bit SuperH CPU**
  - 64-bit hardware FPU (1.16 GFLOPS)
  - 128-bit vector unit for matrix manipulations
  - up to 200MHz, 360 MIPS (DMIPS 1.1)
  - Up to 664 Mbytes/s CPU bandwidth
  - Direct mapped, on-chip, ICache (8 Kbytes) and DCache (16 Kbytes)
- **High-performance 5-channel DMA engine, supporting 1D or 2D block moves and linked lists**
- **SuperHyway internal interconnect**
  - High throughput, low latency, split transaction packet router
- **Memory protection and VM system support**
  - 64-entry unified TLB, 4-entry instruction TLB
  - 4 Gbytes address space
- **Standard ST40 peripherals**
  - 2 synchronous serial ports with FIFO (SCIF)
  - Timers and a real-time clock

IO devices

- Mailbox register for interprocessor communication
- Additional PIO

Bus interfaces

- **Local memory interface SDRAM & DDR SDRAM**
  - Up to 100 MHz (1.6 Gbytes/s peak throughput)
- **PCI interface - 32-bit, 66/33 MHz, 3.3 V**
- **Enhanced memory interface (EMI)**
  - 32-bit bus, up to 83 MHz, for attaching peripherals
  - High-speed, sync mode, burst flash ROM support
  - SDRAM support
  - MPX initiator and target interface
  - Programmable MPX bus arbiter

Contents

A Interconnect architecture

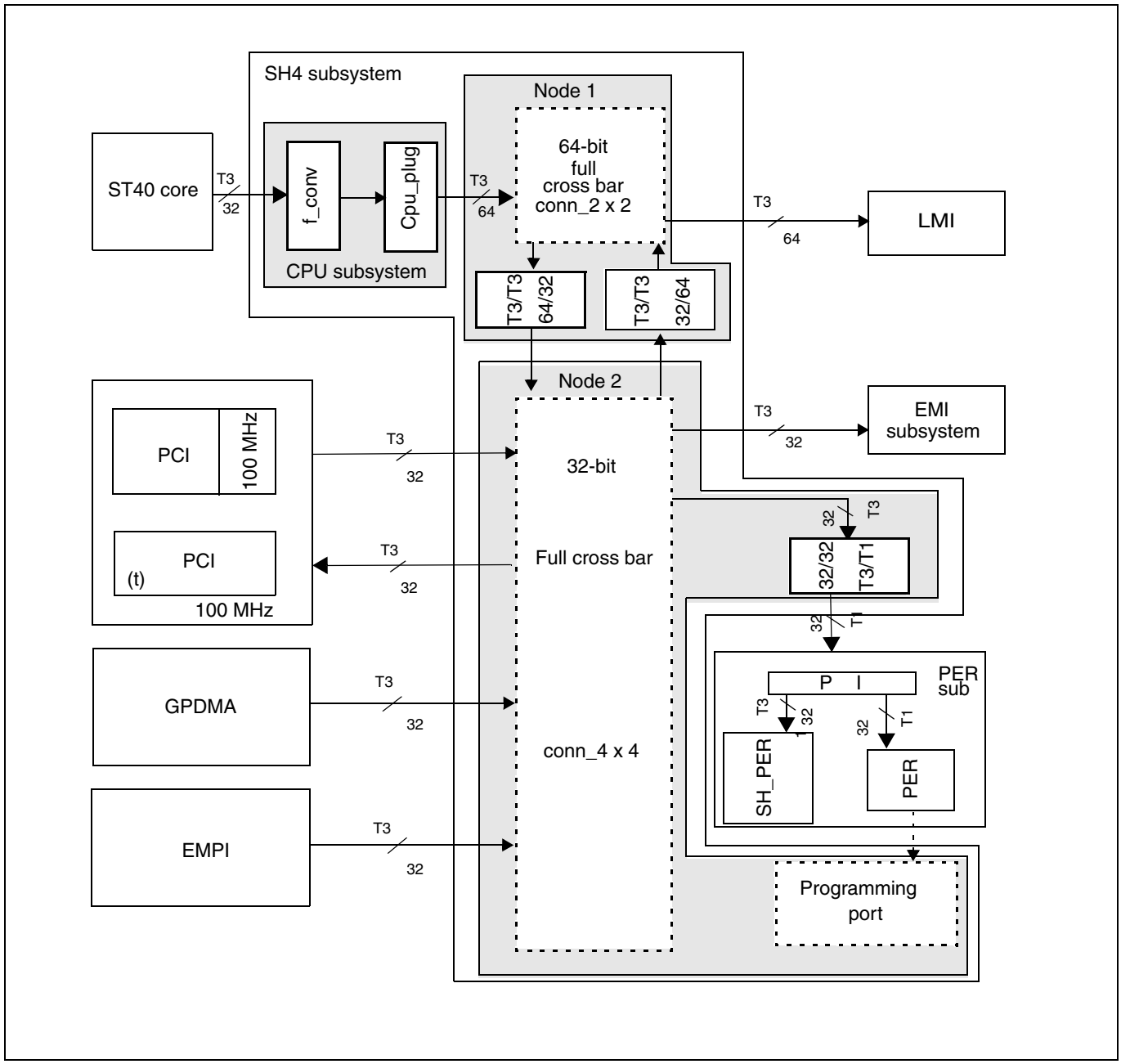
This detail is included for information only. It is not recommended to write to any of these registers, without prior consultation from ST, as it could cause the device to malfunction.

ST only guarantees correct operation of the device with the default register values. The register reset default values have been programmed to balance the system and give optimum system performance, so there is no need to modify them.

For details of other registers see the *ST40 System Architecture Manual*.

The internal architecture of the block is shown in *Figure 1*.

Figure 1: ST40RA interconnect architecture



A.1 Arbitration schemes

A.1.1 PCI arbiter: (CPU, GPDMA, PCI, EMPI)

The default configuration (after reset) for fixed priority mode has to be in the following priority order:

- CPU buffer,
- EMPI,
- GPDMA,
- PCI (PCI master request, although not expected, get served to avoid deadlock).

The priority orders have to be programmable and the latency checking algorithm can be enabled for GPDMA, PCI, EMPI.

A.1.2 EMI arbiter: (CPU buffer, GPDMA, PCI, EMPI)

The default configuration (after reset) for fixed priority mode has to be in the following priority order:

- CPU buffer,
- PCI,
- EMPI,
- GPDMA.

The priority order have to be programmable and the latency checking algorithm can be enabled for GPDMA, PCI, EMPI.

A.1.3 LMI 1 arbiter: (CPU, GPDMA, PCI, EMPI)

The default configuration (after reset) as to be to work fixed priority mode in the following priority order:

- CPU,
- GPDMA and PCI buffer.

The priority orders have to be programmable and the latency checking algorithm can be enabled for GPDMA, PCI, EMPI.

A.1.4 PER arbiter: (CPU, GPDMA, PCI, EMPI)

The default configuration (after reset) as to be to work fixed priority mode in the following priority order:

- CPU buffer,
- PCI,
- EMPI,
- GPDMA.

The priority order have to be programmable and the latency checking algorithm can be enabled for GPDMA, PCI, EMPI.

A.1.5 LMI2 arbiter: (CPU, GPDMA, PCI, EMPI)

The default configuration (after reset) as to be to work fixed priority mode in the following priority order:

- PCI,
- EMPI,
- GPDMA,
- CPU buffer (although the CPU requests are not supposed to go in that node to be send in the LMI, it has to be managed in order to avoid deadlock).

The priority order have to be programmable and the latency checking algorithm can be enabled for GPDMA, PCI, EMPI.

A.1.6 Return arbitration

The possibilities of the return arbitration are simpler than for the request arbitration. The arbiter is not programmable but a specific arbitration can be chosen when implementing it.

The arbitration mode chosen is the fixed priority. For each arbiter (one per initiator), the order is the following: LMI then other targets for the arbiters in node 1 and LMI, EMI, PCI, peripheral subsystem for the arbiters of node 2.

A.2 Interconnect registers

A summary of registers is given in *Table 1*. Addresses in the table are offset from the interconnect base address at 0x1B05 0000.

Table 1: Interconnect register summary

Address offset	Name	Function
0x010	LATENCY_LMI1_ENABLE	Enables or disables initiators latency counters, see *LMI1 arbiter on page 9*
0x018	LMI1_CPU_PRI	Defines priority for the CPU in the LMI1 arbiter, see *LMI1 arbiter on page 9*
0x020	LATENCY_LMI1_VALUE	Defines priority and latency value for the node 2 in the LMI1 arbiter, see *LMI1 arbiter on page 9*
0x110	LATENCY_LMI2_ENABLE	Enables or disables initiators latency counters, see *LMI2 arbiter on page 10*
0x118	LMI2_CPU_PRI	Defines priority for the CPU in the LMI2 arbiter, see *LMI2 arbiter on page 10*
0x120	LMI2_LATENCY_PCI	Defines priority and latency value for PCI initiator in the PCI arbiter, see *LMI2 arbiter on page 10*
0x128	LMI2_LATENCY_EMPI	Defines priority and latency value for EMPI initiator in the PCI arbiter, see *LMI2 arbiter on page 10*
0x130	LMI2_LATENCY_GPDMA	Defines priority and latency value for GPDMA initiator in the PCI arbiter, see *LMI2 arbiter on page 10*
0x210	LATENCY_EMI_ENABLE	Enables or disables initiators latency counters, see *EMI arbiter on page 11*
0x218	EMI_CPU_PRI	Defines priority for the CPU in the EMI arbiter, see *EMI arbiter on page 11*
0x220	EMI_LATENCY_PCI	Defines priority and latency value for PCI initiator in the EMI arbiter, see *EMI arbiter on page 11*
0x228	EMI_LATENCY_EMPI	Defines priority and latency value for EMPI initiator in the EMI arbiter, see *EMI arbiter on page 11*

Table 1: Interconnect register summary

Address offset	Name	Function
0x230	EMI_LATENCY_GPDMA	Defines priority and latency value for GPDMA initiator in the EMI arbiter, see *EMI arbiter on page 11*
0x310	LATENCY_PCI_ENABLE	Enables or disables initiators latency counters, see *PCI arbiter on page 12*
0x318	PCI_CPU_PRI	Defines priority for the CPU in the PCI arbiter, see *PCI arbiter on page 12*
0x320	PCI_LATENCY_PCI	Defines priority and latency value for PCI initiator in the PCI arbiter, see *PCI arbiter on page 12*
0x328	PCI_LATENCY_EMPI	Defines priority and latency value for EMPI initiator in the PCI arbiter, see *PCI arbiter on page 12*
0x330	PCI_LATENCY_GPDMA	Defines priority and latency value for GPDMA initiator in the PCI arbiter, see *PCI arbiter on page 12*
0x410	LATENCY_PER_ENABLE	Enables or disables initiators latency counters, see *Peripheral arbiter on page 13*
0x418	PER_CPU_PRI	Defines priority for the CPU in the peripheral arbiter, see *Peripheral arbiter on page 13*
0x420	PER_LATENCY_PCI	Defines priority and latency value for PCI initiator in the peripheral arbiter, see *Peripheral arbiter on page 13*
0x428	PER_LATENCY_EMPI	Defines priority and latency value for EMPI initiator in the peripheral arbiter, see *Peripheral arbiter on page 13*
0x430	PER_LATENCY_GPDMA	Defines priority and latency value for GPDMA initiator in the peripheral arbiter, see *Peripheral arbiter on page 13*

A.2.1 LMI1 arbiter

	LATENCY_LMI1_ENABLE	**LMI1 arbiter: enable latency counters**	**0x010**
0	Reserved	Reset: Always 0	
1	ENABLE_1	Enable latency check for node 2 Reset: 0	RW
[31:2]	Reserved	Reset: Always 0	

	LMI1_CPU_PRI	**LMI1 arbiter: CPU priority**	**0x018**
[3:0]	CPU_PRIORITY	Defines priority for CPU Reset: 0x1	RW
[31:4]	Reserved		

	LATENCY_LMI1_VALUE	**LMI1 arbiter: node 2 intitiator priority and latency**	**0x020**
[3:0]	NODE2_PRIORITY	Defines priority for node 2 initiators Reset: 0x0	RW
[15:4]	Reserved		

	LATENCY_LMI1_VALUE	LMI1 arbiter: node 2 intitiator priority and latency	0x020
[23:16]	NODE2_LATENCY	Defines maximum accepted latency for node 2 initiators Reset: 0x00	RW
[31:24]	Reserved		

A.2.2 LMI2 arbiter

	LATENCY_LMI2_ENABLE	LMI2 arbiter: enable latency counters	0x110
0	Reserved	Reset: Always 0	
1	ENABLE_PCI	Enable latency check for PCI Reset: 0	RW
2	ENABLE_EMPI	Enable latency check for EMPI Reset: 0	RW
3	ENABLE_GPDMA	Enable latency check for GPDMA Reset: 0	RW
[31:4]	Reserved	Reset: Always 0	

	LMI2_CPU_PRI	LMI2 arbiter: CPU priority	0x118
[3:0]	CPU_PRIORITY	Defines priority for CPU Reset: 0x0	RW
[31:4]	Reserved		

	LMI2_LATENCY_PCI	LMI2 arbiter: PCI intitiator priority and latency	0x120
[3:0]	PCI_PRIORITY	Defines priority for PCI Reset: 0x3	RW
[15:4]	Reserved		
[23:16]	PCI_LATENCY	Defines maximum accepted latency for PCI Reset: 0x00	RW
[31:24]	Reserved		

	LMI2_LATENCY_EMPI	LMI2 arbiter: EMPI intitiator priority and latency	0x128
[3:0]	EMPI_PRIORITY	Defines priority for EMPI Reset: 0x2	RW
[15:4]	Reserved		
[23:16]	EMPI_LATENCY	Defines maximum accepted latency for EMPI Reset: 0x00	RW
[31:24]	Reserved		

	LMI2_LATENCY_GPDMA	LMI2 arbiter: GPDMA intitiator priority and latency	0x130
[3:0]	GPDMA_PRIORITY	Defines priority for GPDMA Reset: 0x1	RW
[15:4]	Reserved		
[23:16]	GPDMA_LATENCY	Defines maximum accepted latency for GPDMA Reset: 0x00	RW
[31:24]	Reserved		

A.2.3 EMI arbiter

	LATENCY_EMI_ENABLE	EMI arbiter: enable latency counters	0x210
0	Reserved	Reset: Always 0	
1	ENABLE_PCI	Enable latency check for PCI Reset: 0	RW
2	ENABLE_EMPI	Enable latency check for EMPI Reset: 0	RW
3	ENABLE_GPDMA	Enable latency check for GPDMA Reset: 0	RW
[31:4]	Reserved	Reset: Always 0	

	EMI_CPU_PRI	EMI arbiter: CPU priority	0x218
[3:0]	CPU_PRIORITY	Defines priority for CPU Reset: 0x3	RW
[31:4]	Reserved		

	EMI_LATENCY_PCI	EMI arbiter: PCI intitiator priority and latency	0x220
[3:0]	PCI_PRIORITY	Defines priority for PCI Reset: 0x2	RW
[15:4]	Reserved		
[23:16]	PCI_LATENCY	Defines maximum accepted latency for PCI Reset: 0x00	RW
[31:24]	Reserved		

	EMI_LATENCY_EMPI	EMI arbiter: EMPI intitiator priority and latency	0x228
[3:0]	EMPI_PRIORITY	Defines priority for EMPI Reset: 0x1	RW
[15:4]	Reserved		

	EMI_LATENCY_EMPI	EMI arbiter: EMPI intitiator priority and latency	0x228
[23:16]	EMPI_LATENCY	Defines maximum accepted latency for EMPI Reset: 0x00	RW
[31:24]	Reserved		

	EMI_LATENCY_GPDMA	EMI arbiter: GPDMA intitiator priority and latency	0x230
[3:0]	GPDMA_PRIORITY	Defines priority for GPDMA Reset: 0x0	RW
[15:4]	Reserved		
[23:16]	GPDMA_LATENCY	Defines maximum accepted latency for GPDMA Reset: 0x00	RW
[31:24]	Reserved		

A.2.4 PCI arbiter

	LATENCY_PCI_ENABLE	PCI arbiter: enable latency counters	0x310
0	Reserved		
1	ENABLE_PCI	Enable latency check for PCI Reset: 0	RW
2	ENABLE_EMPI	Enable latency check for EMPI Reset: 0	RW
3	ENABLE_GPDMA	Enable latency check for GPDMA Reset: 0	RW
[31:4]	Reserved	Reset: Always 0	

	PCI_CPU_PRI	PCI arbiter: CPU priority	0x318
[3:0]	CPU_PRIORITY	Defines priority for CPU Reset: 0x3	RW
[31:4]	Reserved		

	PCI_LATENCY_PCI	PCI arbiter: PCI intitiator priority and latency	0x320
[3:0]	PCI_PRIORITY	Defines priority for PCI Reset: 0x0	RW
[15:4]	Reserved		
[23:16]	PCI_LATENCY	Defines maximum accepted latency for PCI Reset: 0x00	RW
[31:24]	Reserved		

	PCI_LATENCY_EMPI	PCI arbiter: EMPI intitiator priority and latency	0x328
[3:0]	EMPI_PRIORITY	Defines priority for EMPI Reset: 0x2	RW
[15:4]	Reserved		
[23:16]	EMPI_LATENCY	Defines maximum accepted latency for EMPI Reset: 0x00	RW
[31:24]	Reserved		

	PCI_LATENCY_GPDMA	PCI arbiter: GPDMA intitiator priority and latency	0x330
[3:0]	GPDMA_PRIORITY	Defines priority for GPDMA Reset: 0x1	RW
[15:4]	Reserved		
[23:16]	GPDMA_LATENCY	Defines maximum accepted latency for GPDMA Reset: 0x00	RW
[31:24]	Reserved		

A.2.5 Peripheral arbiter

	LATENCY_PER_ENABLE	Peripheral arbiter: enable latency counters	0x410
0	Reserved	Reset: Always 0	
1	ENABLE_PCI	Enable latency check for PCI Reset: 0	RW
2	ENABLE_EMPI	Enable latency check for EMPI Reset: 0	RW
3	ENABLE_GPDMA	Enable latency check for GPDMA Reset: 0	RW
[31:4]	Reserved	Reset: Always 0	

	PER_CPU_PRI	Peripheral arbiter: CPU priority	0x418
[3:0]	CPU_PRIORITY	Defines priority for CPU Reset: 0x3	RW
[31:4]	Reserved		

	PER_LATENCY_PCI	Peripheral arbiter: PCI intitiator priority and latency	0x420
[3:0]	PCI_PRIORITY	Defines priority for PCI Reset: 0x2	RW
[15:4]	Reserved		

	PER_LATENCY_PCI	Peripheral arbiter: PCI intitiator priority and latency	0x420
[23:16]	PCI_LATENCY	Defines maximum accepted latency for PCI Reset: 0x00	RW
[31:24]	Reserved		

	PER_LATENCY_EMPI	Peripheral arbiter: EMPI intitiator priority and latency	0x428
[3:0]	EMPI_PRIORITY	Defines priority for EMPI Reset: 0x1	RW
[15:4]	Reserved		
[23:16]	EMPI_LATENCY	Defines maximum accepted latency for EMPI Reset: 0x00	RW
[31:24]	Reserved		

	PER_LATENCY_GPDMA	Peripheral arbiter: GPDMA intitiator priority and latency	0x430
[3:0]	GPDMA_PRIORITY	Defines priority for GPDMA Reset: 0x0	RW
[15:4]	Reserved		
[23:16]	GPDMA_LATENCY	Defines maximum accepted latency for GPDMA Reset: 0x00	RW
[31:24]	Reserved		

B Implementation restrictions

B.1 ST40 CPU

B.1.1 tas.b

The atomicity of the **tas.b** instruction is only guaranteed for processes executing on the ST40 CPU core and should not be used to implement intermodule or interchip semaphores. Either use the mailbox functionality or an appropriate software algorithm for such semaphores.

B.1.2 Store queue power-down

The store queue is considered part of the general CPU and independent power-down of this block is not implemented.

B.1.3 UBC power-down

The UBC is considered part of the general CPU and independent power-down of this block is not implemented.

B.1.4 System standby

To enter and leave standby it is necessary for the CPU to power down the system including memory devices and then to enter standby by executing a **sleep** instruction. On leaving sleep and standby, it may be necessary for the CPU to power itself up and subsequently power up the system and its memory devices.

During the power-down and power-up sequences the main memory devices are not available. The CPU therefore preloads the appropriate code into the cache as part of the power sequencing.

B.2 PCI

B.2.1 Clocking

PCI internal clock loop back is not implemented. To use the internal PCI clock, the pads PCICLOCKOUT and PCICLOCKIN are connected to rollback the clock generator. Alternatively an external clock source may be used.

B.2.2 Type 2 configuration accesses

Configuration space accesses to devices across a PCI bridge are implemented as type 2 operations on the PCI bus. In this implementation such accesses must be broken into a sequence of byte operations. For example, access to a 32-bit register is through four single byte operations.

B.2.3 Software visible changes between STB1HC7 and ST40RAH8D

PCI PLL reprogramming required for H7 parts is no longer required for H8.

The PCI PLL register is renamed from PLLPCICR to CLKGENA.PLL2CR.

The register implementation for PCI MBAR mappings has changed between the STB1HC7 and ST40RAH8D implementations and software device drivers should reflect this.

B.2.4 Error behavior

The implementation of local (PCI register) error handling is not fully implemented.

B.2.5 Master abort

When operating as a bus master, the PCI module is not guaranteed to have the value 0xFFFF FFFF following a master abort of a read cycle. The master abort may be detected using either the PCI module status and interrupt information supplied by the module.

B.3 EMI/EMPI

B.3.1 EMPI burst mode operation: ST40RA MPX target

MPX operations using the ST40RA as the target which lead to burst requests to memory (Read ahead, 8-, 16- and 32-byte read operations) have limited support.

MPX operations from the ST40RA as an initiator includes full support for all transfer sizes.

B.3.2 SDRAM initialization during boot from flash

During the SDRAM initialization sequence only internal EMI registers are accessible, it is therefore necessary to ensure the program required to execute the initialization sequence is placed in an alternate memory location such as the LMI or preloaded into the cache.

B.3.3 MPX boot

BOOTFROMMPX is not supported on this part.

B.4 Mailbox

B.4.1 Test and set functionality

This is not supported.

B.5 Power down

B.5.1 Module power-down sequencing

Whilst powering down using the associated registers for the ST40RA module, in general, software is responsible for ensuring the module is in a safe state before requesting module shutdown. For details refer to the appropriate documentation.

B.5.2 Accesses to modules in power-down state

Once a module is in power-down state, attempts to access that module may lead the system to hang.

B.6 PIO

B.6.1 PIO default functionality following reset

In the ST40 family device, the operational modes for these registers differ from the standard architecture definition and are shown in *Table 2*.

Table 2: PIO alternate function registers

PIO bit configuration	PIO output state	PIO.PC2	PIO.PC1	PIO.PC0
NonPIO function[a]	-	0	0	0
PIO bidirectional	Open drain	0	0	1
PIO output	Push-pull	0	1	0
PIO bidirectional	Open drain	0	1	1
PIO input	High impedance	1	0	0
PIO input	High impedance	1	0	1
Reserved	-	1	1	0
Reserved	-	1	1	1

a. State following reset

B.6.2 PCI/PIO alternate functions

The following PIO signals cannot be used when PCI is enabled even if the PCI implementation does not require the primary pin function.

Table 3: PCI/PIO alternate functions

Pin name	BPN		Architecture signal name	Pin function		Pin		
	Row	Col		Default	Alternate	Type	Dir	
NOTPREQ0	E	18	NOTPCI_REQ0	PCI external request for bus	PIO16	P8	I/O	I/O
NOTPREQ1	E	17	NOTPCI_REQ1	PCI external request for bus	PIO18	P8	I	I/O
NOTPREQ2	F	16	NOTPCI_REQ2	PCI external request for bus	PIO20	P8	I	I/O
NOTPREQ3	G	16	NOTPCI_REQ3	PCI external request for bus	PIO22 EMPIDREQ1	P8	I	I/O O

If PCI is disabled, the alternate functions may be used.

B.7 Interconnect

B.7.1 Memory bridge functionality

Ensure there is no traffic passing though the memory bridge when changing frequency.

Semisynchronous modes of operation are not supported.

B.7.2 Clock selection

The alternate CLOCKGENB clock is not supported for the LMI.

B.7.3 Pad drive control

Programmable drive strength control is not supported for DDR operation.

B.8 GPDMA

B.8.1 Linked list support

Decrementing transfers are not supported as part of link list transfer sequences

B.8.2 2-D transfers

2-D transfers fail if the following conditions are met.

1 Source or destination length is greater than 64 bytes.
2 Real transfer unit is less then 32 bytes.
3 The expression length = n * 64 + *tu* is true, where:
   - length is either SLENGTH or DLENGTH,
   - *tu* the real transfer unit of the first access of the second line,
   - $n > 0$.

B.8.3 Protocol signals

DACK and DRACK protocol signals have limited support.

1 Scope of this document

This document describes only those areas of the ST40RA that are device specific, for example the system address map. Information that is generic to the ST40 family of devices is contained in the ST40 documentation suite.

2 ST40 documentation suite

This document references a number of other generic ST40 documents that combined together form a complete datasheet.

CPU documentation

The SH-4 CPU core and its instruction set are documented in the *SH-4 CPU Core Architecture Manual*.

System documentation

Devices listed in the system address map, *Figure 5 on page 26* are documented in the *ST40 System Architecture Manual*:

- *Volume 1: System*, details the ST40 CPU and standard peripherals,
- *Volume 2: Bus Interfaces*, details the standard PCI, LMI and EMI bus interfaces.

3 ST40RA devices

Table 4: ST40RA device types

Device	CPU clock frequency	Temperature range		VDD Core
		Minimum	Maximum	
ST40RA150XHA	150 MHz	-40 °C	+85 °C	1.65V to 1.95V
ST40RA166XH1	166 MHz	0 °C	+70 °C	
ST40RA166XH6	166 MHz	-40 °C	+85 °C	
ST40RA200XH6	200 MHz	-40 °C	+85 °C	1.80V to 1.95V

4 Architecture

4.1 Overview

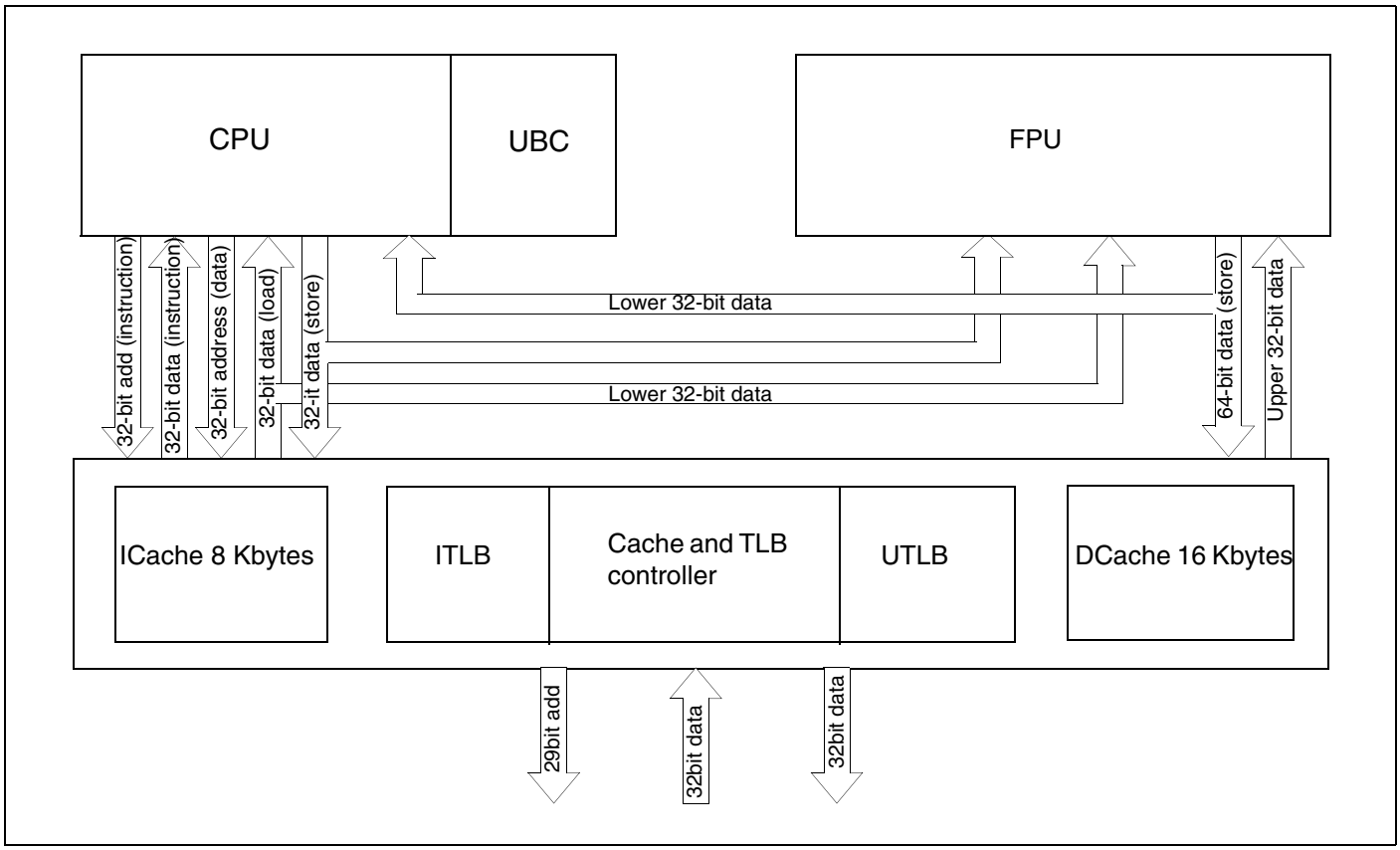
The ST40RA combines an SH-4, 32-bit microprocessor with a wide range of interfaces to external peripherals. This section briefly describes each of the features of the ST40RA.

4.2 ST40 system

4.2.1 SuperH ST40 SH-4 core

Figure 2 illustrates the system architecture of the ST40 SH-4 core. The following section briefly describes the features and performance of the core.

Figure 2: ST40 SH-4 core architecture



Central processing unit

The central processing unit is built around a 32-bit RISC, two-way superscalar architecture. Operating at 166 MHz it runs with high code density using fixed length 16-bit instructions. It has a load/store architecture, delayed branch instruction capability and an on-chip multiplier. It uses a five-stage pipeline.

Floating point unit/multiply and accumulate

The on-chip, floating point coprocessor executes single precision (32-bit) and double precision (64-bit) operations. It has a five-stage pipeline and supports IEEE754-compliant data types and exceptions. It has rounding modes: (round-to-nearest) and (round-to-zero), and handles denormalized numbers (truncation-to-zero) or interrupt generation for compliance with IEEE754. The floating point unit performs the following functions:

- **fmac** (multiply-and-accumulate), **fdiv** (divide),
- **fsqrt** (square root) instructions,
- 3-D graphics instructions (single-precision):
  - ➢ 4-dimensional vector conversion and matrix operations (**ftrv**): 4 cycles (pitch), 7 cycles (latency),
  - ➢ 4-dimensional vector (**fipr**) inner product: 1 cycle (pitch), 4 cycles (latency).

MMU configuration

There is 4 Gbytes virtual address space with 256 address space identifiers (8-bit ASIDs), supporting single virtual and multiple virtual memory modes. Page sizes are 1 Kbyte, 4 Kbytes, 64 Kbytes or 1 Mbyte. The MMU supports four-entry, fully associative ITLB for instructions and 64-entry fully associative UTLB for instructions and operands. Software-controlled replacement and random-counter replacement algorithms are also supported. The physical address space is 512 Mbytes (29-bit), see *Figure 3: System address organization on page 25*.

Cache

8 Kbytes of direct-mapped instruction cache are organized as 256 32-byte lines, and 16 Kbytes of direct-mapped operand cache are organized as 512 32-byte lines. RAM mode (8-Kbyte cache plus 8-Kbyte RAM) with selectable write method (copy back or write through) is supported. A single stage buffer for copy-back and a single stage buffer for write-through are available. The cache contents can be address mapped and there is a 32-byte two-entry store queue.

4.2.2 SuperHyway internal interconnect

The ST40RA uses the SuperHyway memory mapped packet router for on-chip intermodule communication. The interconnect supports a split transaction system allowing a nonblocking high throughput, low latency system to be built. There are separate request and response packet routers.

The ST40RA SuperHyway implementation is show in *Section 5.8: Memory bridge control on page 34*. The interconnect allows simultaneous requests between multiple modules and is able to ensure a very high data throughput with in many cases zero routing, arbitration and decode latencies.

4.2.3 Standard ST40 peripherals

Synchronous serial channel

There are two ST40 compatible full duplex communication channels (SCIF1, SCIF2). Asynchronous mode is supported. A separate 16-byte FIFO is provided for the transmitter and receiver.

Interrupt controller

The interrupt controller supports all of the on-chip peripheral module interrupts, and five external interrupts (NMI and IRL0 to IRL3). The priority can be set for each on-chip peripheral module interrupt. IRL0 to IRL3 are configured as four independent interrupts or encoded to provide 15 external interrupt levels.

Debug controller

Debugging is performed by break interrupts. There are two break channels. The address, data value, access type, and data size can all be set as break conditions. Sequential break functions are supported.

The user debug interface (UDI) contains a five-pin serial interface conforming to JTAG, IEEE Standard TAP and boundary scan architecture. The interface provides host access to the 1 Kbyte ASERAM for emulator firmware (accessible only in ASE mode).

Timers

The three-channel, auto-reload, 32-bit timer has an input capture function and a choice of seven counter input clocks.

Real-time clock

The built-in 32-kHz crystal oscillator has a maximum 1/256 second resolution. It has dynamically programmable operating frequencies and on-chip clock and calendar functions. It has two sleep modes and one standby mode.

Watchdog timer

The ST40RA has an 8-bit watchdog timer (WDT) with programmable clock ratio. The WDT is able to generate a power-on reset or a manual reset.

Programmable PLLs

The ST40RA has three programmable PLLs. The PLLs are configured by MODE pins at reset and then reconfigured by software to optimize system performance or reduce system power consumption.

General-purpose DMA controller

The five-channel physical address GPDMA controller has four general-purpose channels for memory-to-memory or memory-to-peripheral transfers, and one buffered multiplexed channel. Both 2-D block moves and linked lists are supported. Two sets of DMA handshake pins are available for use by external devices to support efficient transfer interdevice transfers via external interfaces such as the EMI MPX.

Parallel I/O module

24 bits of parallel I/O are provided from the ST40 compatible PIO. Each bit is programmable as an output or an input. “Input compare” generates an interrupt on any change of any input bit.

4.3 Bus interfaces

4.3.1 Local memory interface

The LMI supports 16-, 32- and 64-bit wide bus SDRAM and DDR SDRAM, at up to 100 MHz with a maximum address space of 112 Mbytes. Devices supported include two and four bank 16-, 64-, 128- and 256-Mbit technologies in x4, x8, x16 and x32 packages. The LMI pads are dual mode pads electrically compatible with LVTTL (for standard SDRAM) and SSTL_2 (for DDR SDRAM). For full detail of the configuration options of the LMI please see *ST40 System Architecture Manual, Volume 2: Bus Interfaces*.

4.3.2 PCI interface

The PCI interface complies to the *PCI v2.1* and *Power Management Interface V1.0* specifications. It is 32 bits wide and operates at 33 or 66 MHz. Master and target mode are supported. A PCI arbiter and clock generator is provided inside the ST40RA. For details on the configuration options for the PCI interface please see *ST40 System Architecture Manual, Volume 2: Bus Interfaces*.

4.3.3 EMI/MPX interface

The EMI/MPX interface contains the following blocks. For full details of the configuration options of the EMI please see the *ST40 System Architecture Manual, Volume 2: Bus Interfaces*.

EMI memory interface initiator

The EMI provides access to ROMs, SDRAM, memory mapped asynchronous external peripherals and synchronous MPX bus peripherals. The EMI supports burst mode flash ROM and MPX for memory-mapped device coupling. The ST40RA GPDMA unit accesses external devices and two sets of DMA channels control signals are provided for this purpose.

EMPI memory interface target

The EMPI is a synchronous MPX target that allows for an external MPX initiator to access the ST40RA internal memory space. The EMPI contains a general purpose control channel and four high performance channels each of which implements a write buffer and a pair of 32-byte read-ahead buffers able to optimize external device burst access to and from the ST40RA internal memory. These buffers can be associated with memory regions within the ST40RA and external DMA channels. Four sets of DMA handshake signals are provided to the EMPI to optimize long burst transfers between the ST40RA and external initiators like the STi5514.

MPX bus arbiter

The ST40RA has an internal programmable bus arbiter to optimize utilization of the MPX bus. The ST40RA MPX arbiter supports one external initiator and has programmable bus priority (ST40RA or external device), bus parking (ST40RA, external, idle or last user) and latency timers. The internal arbiter can be bypassed if an external arbiter supporting more initiators is required.

4.4 I/O devices

4.4.1 Mailbox

The ST40 and the external microprocessor communicate with each other and synchronize their activities using the memory-mapped mailbox. Processes generate interrupts to either CPU, and send and receive messages between the two CPUs. There are buffers for message queueing in both directions and interrupt bits can be set in each direction. Access to the mailbox from external devices is through the ST40RA EMPI or the PCI target interface.

4.5 Software

4.5.1 Development systems and software

The ST40RA supports application development, with a full range of debug features and an emulation mode (ASE). The ASE mode has a dedicated 1-Kbyte buffer for emulator firmware, supporting performance counters and branch trace. The ST40RA, with its memory management unit, supports standard operating systems including WindowsCE and Linux. The ST40 has a wide range of development support from ST and third parties, and efficiently runs applications written in C, C++ and Java.

ST's own tools include:

- C/C++ compilers,
- debugger,
- proprietary OS.

Third parties include:

- Microsoft: WindowsCE,
- Sun: JavaOS for consumers,
- WindRiver: VxWorks, Tornado tools,
- Linux,
- Insignia JVM,
- ANT browser.

4.5.2 Software compatibility

SH-4 core software

The ST40RA SH-4 core is binary code compatible with the Hitachi SH775x family.

Standard peripheral driver

The ST40 standard SCIF, timer, real-time clock and PIO are compatible with the ST40 SOC range of devices and the Hitachi SH775x family.

Bus interface driver

The PCI, LMI, and EMI interfaces are register compatible with the ST40 SOC range of devices.

The ST40RA contains an EMPI and MPX arbiter and MPX clock control unit which are additional to the bus interface components of the ST40 SOC range of devices.

I/O device driver

The Mailbox is a module with no ST legacy software.

5 System configuration

The ST40RA system address map has been designed to maintain compatibility with existing ST40 family devices and other STMicroelectronics devices.

The SH-4 core and core peripherals maintain compatibility with the ST40 SOC range of devices and Hitachi SH7750 wherever possible.

Devices listed in *Table 5: ST40RA system address map on page 26*, are documented in the *ST40 System Architecture Manual* as described in *Chapter 2: ST40 documentation suite on page 19*.

Coherency between the cache and external memory is assured by software. The ST40 CPU has cache control instructions which enable software to do this. Details of these instructions are given in the *ST40 CPU Core Architecture Manual*.

The ST40RA is run in little endian mode.

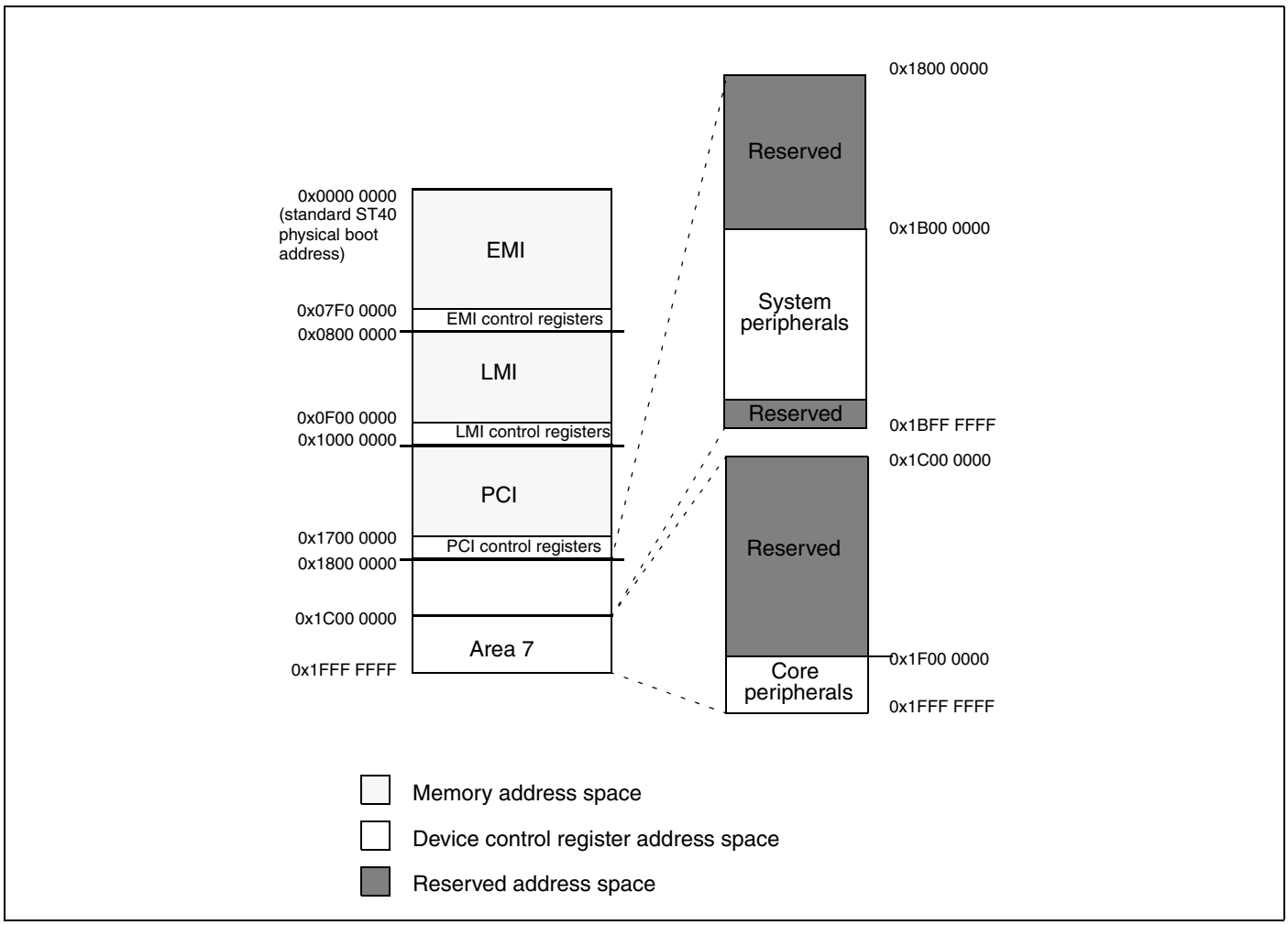
The ST40RA power on configuration is controlled by the MODE pins as defined in *Table 37: Mode selection pins for ST40RA on page 72*.

Subsystem configuration registers are usually found with the module register space. Other system level functions and the software register locations are shown in *Table 14: System configuration registers on page 36*.

5.1 System addresses

The ST40 family system address organization is shown in *Figure 3*.

Figure 3: System address organization



5.1.1 System address map

Table 5: ST40RA system address map

Module	Address[a]		Reference
	Base	Top	
Standard bus interfaces			*ST40 System Architecture Manual Volume 2: Bus Interfaces*
EMI (FMI)	0x0000 0000	0x07EF FFFF	
EMI control and buffer registers	0x07F0 0000	0x07FF FFFF	
LMI	0x0800 0000	0x0EFF FFFF	
LMI control registers	0x0F00 0000	0x0FFF FFFF	
PCI	0x1000 0000	0x16FF FFFF	
PCI control registers	0x1700 0000	0x17FF FFFF	
Reserved	0x1800 0000	0x1AFF FFFF	
ST40 core peripherals			*ST40 System Architecture Manual Volume 1: System*
DMAC	0x1B00 0000	0x1B00 FFFF	
PIO1	0x1B01 0000	0x1B01 FFFF	
PIO2	0x1B02 0000	0x1B02 FFFF	
PIO3	0x1B03 0000	0x1B03 FFFF	
CLOCKGEN	0x1B04 0000	0x1B04 FFFF	
Interconnect	0x1B05 0000	0x1B05 FFFF	
Reserved	0x1B06 0000	0x1B0F FFFF	
CLOCKGENB	0x1B10 0000	0x1B10 FFFF	
Reserved	0x1B11 0000	0x1B12 FFFF	
EMPI	0x1B13 0000	0x1B13 7FFF	*ST40 System Architecture Manual Volume 2: Bus Interfaces*
MPXARB	0x1B13 8000	0x1B13 FFFF	*ST40 System Architecture Manual Volume 2: Bus Interfaces*
ST40RA additional peripherals			*ST40 System Architecture Manual Volume 4: I/O Devices*
MailBox	0x1B15 0000	0x1B15 FFFF	
SYSCONF	0x1B19 0000	0x1B19 FFFF	
Reserved	0x1B1A 0000	0x1B1F FFFF	
Reserved for additional peripherals			
Reserved	0x1B20 0000	0x1B3F FFFF	
ST40 core peripherals			*ST40 System Architecture Manual Volume 1: System*
INTC2	0x1E08 0000	0x1E0F FFFF	

Table 5: ST40RA system address map

Module	Address[a]		Reference
	Base	Top	
Reserved: CPU only registers	0x1E10 0000	0x1FBF FFFF	
CPG	0x1FC0 0000	0x1FC7 9999	
RTC	0x1FC8 0000	0x1FCF FFFF	
INTC	0x1FD0 0000	0x1FD7 9999	
TMU	0x1FD8 0000	0x1FDF FFFF	
SCIF1	0x1FE0 0000	0x1FE7 9999	
SCIF2	0x1FE8 0000	0x1FEF FFFF	
EMU	0x1FF0 0000	0x1FF7 9999	
Reserved	0x1FF8 0000	0X1FFF FFFF	

a. For information about which address region to access for each module, see *SH-4 32-bit CPU Core Architecture, sections 2.5 and 3.4*.

When operating in privilege mode, these registers should be accessed via the P2 region by adding an offset of 0xA000 0000, when operating in user mode, access should be via the U0 address.

5.2 System identifiers

- SH-4 core processor identity: 0x0100.
- SH-4 core processor version: 0x0541D.
- ST40RA-HC8 TAP identity: 05141041.
- ST40RA-HC8 PCI identity:
  - ➢ Vendor: 104A,
  - ➢ Device: 4000,
  - ➢ Revision ID: 0x01,
  - ➢ Class: 0x4 0000,
  - ➢ Subsystem ID: 0x0000.

5.3 Interrupt mapping

For full details on the interrupt controller see *ST40 System Architecture Manual Volume 1:System*.

The mapping of the CPU interrupts is described in *Section 5.3.1*, *Section 5.3.2* and *Section 5.3.3*.

Note: Some INTEVT codes are shown as reserved in Table 6 and therefore cannot be generated by this device.

5.3.1 ST40 core interrupt allocation

The allocation of core interrupts is as shown in *Table 6*.

Table 6: ST40 core interrupt allocation (page 1 of 2)

Interrupt source		INTEVT code	Interrupt priority		IPR bit numbers	Priority within IPR setting unit
			Value	Initial value		
NMI		0x1C0	16	-	-	-
IRL level encoding	IRL3–IRL0 = F	0x200	15	-	-	-
	IRL3–IRL0 = E	0x220	14	-	-	-
	IRL3–IRL0 = D	0x240	13	-	-	-
	IRL3–IRL0 = C	0x260	12	-	-	-
	IRL3–IRL0 = B	0x280	11	-	-	-
	IRL3–IRL0 = A	0x2A0	10	-	-	-
	IRL3–IRL0 = 9	0x2C0	9	-	-	-
	IRL3–IRL0 = 8	0x2E0	8	-	-	-
	IRL3–IRL0 = 7	0x300	7	-	-	-
	IRL3–IRL0 = 6	0x320	6	-	-	-
	IRL3–IRL0 = 5	0x340	5	-	-	-
	IRL3–IRL0 = 4	0x360	4	-	-	-
	IRL3–IRL0 = 3	0x380	3	-	-	-
	IRL3–IRL0 = 2	0x3A0	2	-	-	-
	IRL3–IRL0 = 1	0x3C0	1	-	-	-
IRL independent encoding	IRL0	0x240	15 to 0	13	IPRD[15:12]	-
	IRL1	0x2A0	15 to 0	10	IPRD[11:8]	-
	IRL2	0x300	15 to 0	7	IPRD[7:4]	-
	IRL3	0x360	15 to 0	4	IPRD[3:0]	-
H-UDI	H-UDI	0x600	15 to 0	0	IPRC[3:0]	-
TMU0	TUNI0	0x400	15 to 0	0	IPRA[15:12]	-
TMU1	TUNI1	0x420	0 to 15	0	IPRA[11:8]	-
TMU2	TUNI2	0x440	0 to 15	0	IPRA[7:4]	High
	TICPI2	0x460				Low

Table 6: ST40 core interrupt allocation (page 2 of 2)

Interrupt source		INTEVT code	Interrupt priority		IPR bit numbers	Priority within IPR setting unit
			Value	Initial value		
RTC	ATI	0x480	0 to 15	0	IPRA [3:0]	High to low
	PRI	0x4A0				
	CUI	0x4C0				
SCIF1	ERI	0x4E0	0 to 15	0	IPRB[7:4]	High to low
	RXI	0x500				
	BRI	0x520				
	TXI	0x540				
SCIF2	ERI	0x700	0 to 15	0	IPRC[7:4]	High to low
	RXI	0x720				
	BRI	0x740				
	TXI	0x760				
WDT	ITI	0x560	0 to 15	0	IPRB[15:12]	-

5.3.2 ST40 standard system interrupt allocation

Standard ST40 family interrupts are mapped as shown in *Table 7*.

Table 7: ST40 standard interrupt allocation

Interrupt source		INTEVT code	Interrupt priority		IPR bit numbers	Priority within IPR setting unit
			Value	Initial value		
PCI	PCI_SERR_INT	0xA00	0 to 15	0	INTPRI00[0:3] INTPRI00[7:4]	High to low
	PCI_ERR_INT	0xA20				High to low
	PCI_AD_INT	0xA40				
	PCI_PWR_DWN	0xA60				
	Reserved					
DMAC	DMA_INT0	0xB00	0 to 15	0	INTPRI00[11:8]	High to low
	DMA_INT1	0xB20				
	DMA_INT2	0xB40				
	DMA_INT3	0xB60				
	DMA_INT4	0xB80				
	Reserved					
	DMA_ERR	0xBC0				
PIO0	PIO0	0xC00	0 to 15	0	INTPRI00[15:12]	-
PIO1	PIO1	0xC80	0 to 15	0	INTPRI00[19:16]	-
PIO2	PIO2	0xD00	0 to 15	0	INTPRI00[23:20]	-

5.3.3 ST40RA I/O device interrupt allocation

Table 8: Mailbox and EMPI interrupt allocation

Interrupt source		INTEVT code	Interrupt priority		IPR bit numbers	Priority within IPR setting unit
			Value	Initial value		
Mailbox	MAILBOX	0x1000	0 to 15	0	INTPRI04[0:3]	High to low
	Reserved					
Reserved			0 to 15	0	INTPRI04[27:24]	High to low
EMPI	INV_ADDR	0x1380	0 to 15	0	INTPRI04[31:28]	High to low
	Reserved					

5.4 GPDMA channel mapping

For full details of the GPDMA controller see *ST40 System Architecture Manual Volume 1: System*.

The ST40RA general purpose DMA controller channel map is shown in *Table 9*.

Table 9: GPDMA request number allocation

Request number	Associated device	Protocol	Comment
0	External device 0	DREQ or DREQ/DRACK	The following pins are available for external peripherals: DREQ[0:1], DACK[0:1], DRAK[0:1].
1	External device 1	DREQ or DREQ/DRACK	
2 and 3	Reserved		
4	SCIF1 transmit	DREQ	This allow SCIF to memory and memory to SCIF transfer to be supported on any DMA channel.
5	SCIF1 received	DREQ	
6	SCIF2 transmit	DREQ	
7	SCIF2 receive	DREQ	
8	TMU	DREQ/DRACK	Typically used to trigger or pace memory transfers.
9 and 10	Reserved		
11	PCI1	DREQ or DREQ/DRACK	May be used to improve the efficiency of transfers to and from the PCI.
12	PCI2	DREQ or DREQ/DRACK	
13	PCI3	DREQ or DREQ/DRACK	
14	PCI4	DREQ or DREQ/DRACK	
15 to 31	Reserved		

5.5 EMI DACK mapping

For full details of the EMI bank address and bank type mappings refer to *ST40 System Architecture Manual Volume 2: Bus Interfaces.*

Two DACK strobes are supported in this implementation and are mapped as follows:

- **DACK[0]**: asserted when a transfer from GPDMA channel[1] occurs to an EMI bank configured as a MPX device,
- **DACK[1]**: asserted when a transfer from GPDMA channel[2] occurs to an EMI bank configured as a MPX device.

5.6 EMI address pin mapping

The data width of a connected device is 8, 16 or 32 bits wide. The 16-bit bank must use EDQM3 as address 1, the LSB address for the device and the 8-bit bank must use EDQM3 as address 1 and EDQM2 as address 0.

See the *ST40 System Architecture Manual, Volume 2: Bus Interfaces* for details of setting the device type and port size using the EMI configuration registers.

Table 10: Mapping the internal address lines of a connected device

Device type	Port size	Device address 25 to 2	Device address 1	Device address 0
SDRAM Peripheral SFlash™	32-bit	EADDR[25:2]	-	-
	16-bit	EADDR[24:2]	EDQM3	-
	8-bit	EADDR[23:2]	EDQM3	EDQM2
MPX	-	EADDR[25:2]	-	-

5.7 EMI pin to function relationship

Table 11: EMI pin functions

ST40RA EMI pin	Peripheral	SFlash	SDRAM	MPX	MPX/EMPI
EADDR[2:26]	MEM_ADDRESS	MEM_ADDRESS	MEM_ADDRESS	-	-
EADDR3	NOT_CS	-	-	CLK	CLK
EADDR4	NOT_OE	-	-	/CS	/CS
EADDR5	NOT_BE	-	-	/FRAME	/FRAME
EADDR6	MEM_DATA	-	-	/BS	/BS
EADDR7	(write)	-	-	/WE	/WE
EADDR8	MEM_DATA	-	-	I/O [31:0]	I/O [31:0]
EADDR9	(read)	-	-	I/O [63:61]	I/O [63:61]
EDATA[0:31]	MEM_DATA	MEM_DATA	MEM_DATA	MEM_DATA[31:0]	MEM_DATA[31:0]
ECLKOUT	-	-	SDRAMCLOCK	-	-
ECLKEN	-	-	CKEN	-	-
EDQM0	NOT_BE0	NOT_BE0	NOT_MEMBE0	-	-
EDQM1	NOT_BE1	NOT_BE1	NOT_MEMBE1	-	-
EDQM2	NOT_BE2	NOT_BE2	NOT_MEMBE2	-	-
EDQM3	NOT_BE3	NOT_BE3	NOT_MEMBE3	-	-
NOTECS0	NOT_CS0	NOT_CS0	NOT_SDRAMCS0	NOT_CS0	NOTEMPICS0
NOTECS1	NOT_CS1	NOT_CS1	NOT_SDRAMCS1	NOT_CS1	NOTEMPICS1
NOTECS2	NOT_CS2	NOT_CS2	NOT_SDRAMCS2	NOT_CS2	NOTEMPICS2
NOTECS3	NOT_CS3	NOT_CS3	NOT_SDRAMCS3	NOT_CS3	NOTEMPICS3
NOTECS4	NOT_CS4	NOT_CS4	-	NOT_CS4	NOTEMPICS4
NOTECS5	NOT_CS5	NOT_CS5	-	NOT_CS5	NOTEMPICS5
NOTERAS	-	NOT_ADDRVALID	NOT_MEMRAS	NOT_BS	NOT_BS
NOTECAS	NOT_OE	NOT_OE	NOT_MEMCAS	NOT_FRAME	NOT_FRAME
EWAIT	MEM_WAIT	MEM_WAIT	-	MEM_WAIT	MEM_WAIT
NOTEWE	READNOTWRITE	READNOTWRITE	READNOTWRITE	READNOTWRITE	READNOTWRITE
EPENDING[a]	RFSH_PENDING or ACC_PENDING (master) ACC_PENDING only (slave)	RFSH_PENDING or ACC_PENDING (master) ACC_PENDING only (slave)	RFSH_PENDING or ACC_PENDING (master) ACC_PENDING only (slave)	RFSH_PENDING or ACC_PENDING (master) ACC_PENDING only (slave)	RFSH_PENDING or ACC_PENDING (master) ACC_PENDING only (slave)
MCLKOUT	-	-	-	MPX clock	MPXCLOCK
NOTMREQ (slave)	EMI_HOLD_REQ	EMI_HOLD_REQ	EMI_HOLD_REQ	-	-
NOTMREQ (master)	EMI_BUS_REQ	EMI_BUS_REQ	EMI_BUS_REQ	MPX bus request	MPX bus request
NOTMACK (slave)	EMI_HOLD_ACK	EMI_HOLD_ACK	EMI_HOLD_ACK	-	-

Table 11: EMI pin functions

ST40RA EMI pin	Peripheral	SFlash	SDRAM	MPX	MPX/EMPI
NOTMACK (Master)	EMI_BUS_GRANT	EMI_BUS_GRANT	EMI_BUS_GRANT	MPX bus acknowledge	MPX bus acknowledge
FCLKOUT	-	FLASHCLOCK	-	-	-
NOTFBAA	-	Unconnected/ connected[b]	-	-	-
NOTESCS0	-	-	-	-	MBXINT
NOTESCS1	-	-	-	-	EMPIDREQ0
NOTESCS2	-	-	-	-	EMPIDRAK0

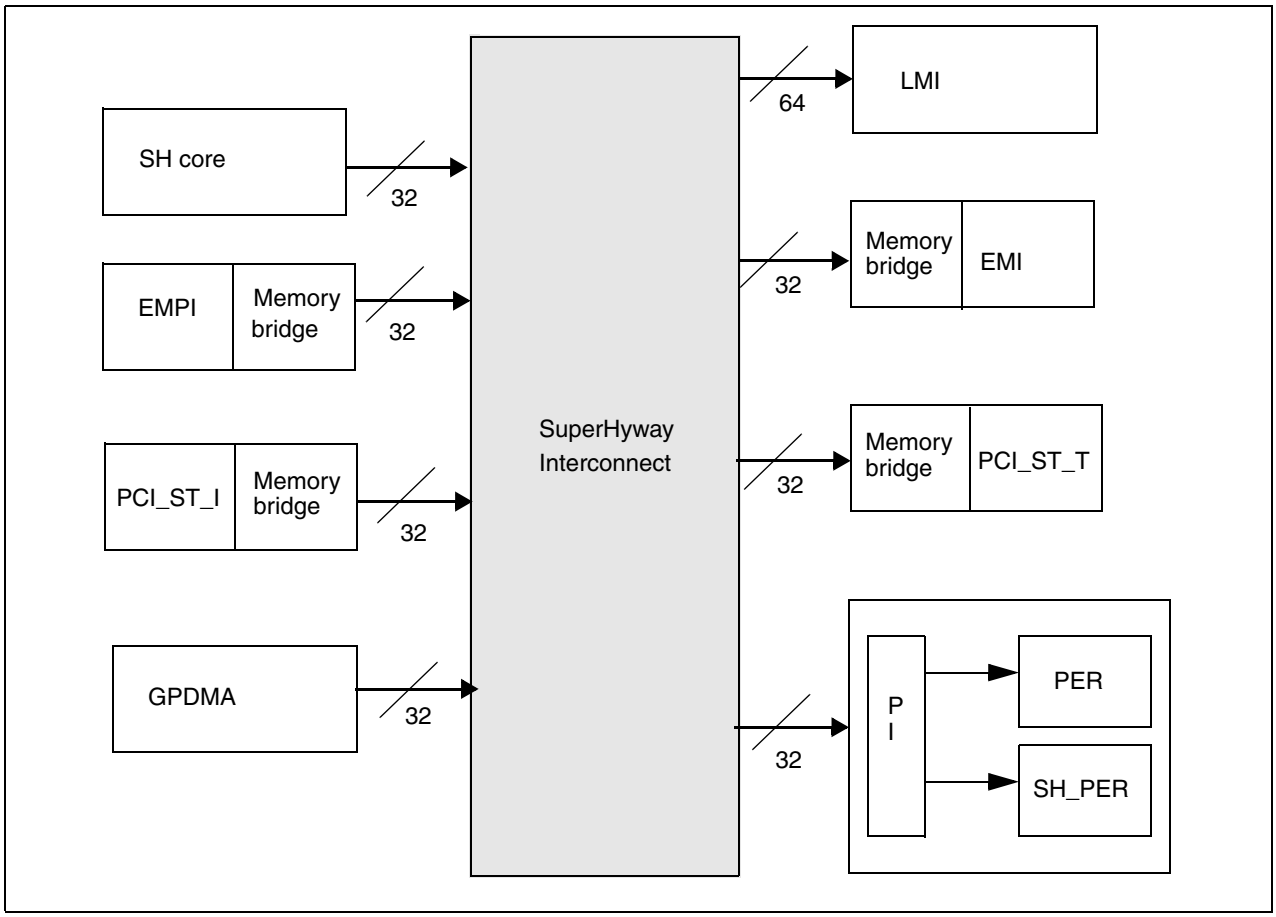
a. When the EMI is configured in master mode (MODE9 = H), and an external slave DMA asks for access to the bus (using NOTMACK or NOTMREQ), RFSH_PENDING and ACC_PENDING are used to signal that, while the external DMA request has been granted and the DMA is using the bus, a refresh time out occurred, or that the EMI has been asked for a new access. A bus arbiter, if present, can use this information to give back the bus to the EMI to allow a refresh operation, or improve bandwidth. When the EMI is in slave mode (MODE9 = L), RFSH_PENDING is always deasserted (so EPENDING = ACC_PENDING), and the pin is used to signal to the external bus arbiter that the EMI needs to use the bus.

b. NOTFBAA is an output of the ST40RA, and an input to the memory device. The pin must be left unconnected from the ST40RA side and tied low at the memory device side if the memory is an Intel or an STM part. It needs to be connected if the SFlash is an AMD.

5.8 Memory bridge control

The architecture of the SuperHyway interconnect is shown in *Figure 4*. Initiators are shown on the left, and targets are shown on the right of the interconnect. The bit width of the initiator and target ports are shown in the diagram.

Figure 4: ST40RA interconnect architecture



The ST40RA architecture requires seven memory bridges on clock change boundaries.

Table 12: Memory bridges

Memory bridge number	SuperHyway type	Subsystem
1	T3	EMI target
2	T3	EMPI initiator
3	T1	EMI_SS target
4	T2	Reserved
5	T2	Reserved
6	T3	PCI_ST_I
7	T3	PCI_ST_T

5.8.1 Memory bridge control signals

Each memory bridge has seven control signals as defined in *Table 13*.

Table 13: Memory bridge control signals

Bridge control bit field	Control name	Control function
1:0	MODE[1:0]	00: Sync (bypass) bridge 01: Semisync with no retime registers 10: Semisync with one retime register 11: Async with two retime registers
4:2	LATENCY[2:0]	Sets FIFO latency from 0 to 7 cycles.
5	SW_RESET	0: Software reset inactive 1: Software reset active
6	STROBE	The above control signals are latched in the bridge on the rising edge of this strobe bit

5.8.2 Memory bridge status

The memory bridge control signals are looped back to the ST40RA comms subsystem SYS_STAT1 register for test purposes. The format of this read-only register is shown in *Section 5.9.4.1: SYSCONF.SYS_STAT1. on page 39*.

5.8.3 Changing control of a memory bridge

At reset all these bridges are set to be synchronous. After reset and boot the function of these memory bridges can be changed. See *Section 5.9.4: SYSCONF registers on page 39*. The procedure for changing the control of a memory bridge is given below.

1. Ensure no initiators are accessing the subsystem the bridge is connected to and ensure the subsystem cannot initiate any requests to the SuperHyway.
2. Stop the clock to the subsystem.
3. Change the memory bridge configuration using the SYS_CONF.SYS_CON1 register as detailed in *Table 13*.
4. Restart the clock to the subsystem and reinitialize the system.

5.9 System configuration registers

Table 14 outlines the ST40RA system configuration registers.

Table 14: System configuration registers

Register	Module	Address offset	Type	Description
EMI.GENCFG	EMI	0x028	R/W	EMI general purpose configuration register, see *Section 5.9.1: EMI.GENCFG EMI general configuration on page 37*
LMI.COC	LMI	0x028	R/W	LMI clock and pad control register, see *Section 5.9.2: LMI.COC on page 38*
LMI.CIC	LMI	0x040	RO	LMI clock and pad status, see *Section 5.9.3: LMI.CIC on page 39*
SYS_STAT1	SYSCONF	0x040	RO	Memory bridge status, see *Section 5.9.4.1: SYSCONF.SYS_STAT1. on page 39*
SYSCONF.SYS_CON1	SYSCONF	0x010	R/W	System configuration register, see *Section 5.9.4.2: SYSCONF.SYS_CON1. on page 40*
SYSCONF.SYS_CON2	SYSCONF	0x018	R/W	System configuration register, see *Section 5.9.5: SYSCONF.SYS_CON2. on page 40*
SYSCONF.CNV_STATUS	SYSCONF	0x020	R/W	System configuration register, see *ST40 System Architecture Manual Volume 4: I/O Devices*
SYSCONF.CNV_SET	SYSCONF	0x028	R/W	System configuration register, see *ST40 System Architecture Manual Volume 4: I/O Devices*
SYSCONF.CNV_CLEAR	SYSCONF	0x030	R/W	System configuration register, see *ST40 System Architecture Manual Volume 4: I/O Devices*
SYSCONF.CNV_CONTROL	SYSCONF	0x038	R/W	System configuration register, see *ST40 System Architecture Manual Volume 4: I/O Devices*

5.9.1 EMI.GENCFG EMI general configuration

	EMI.GENCFG	EMI general configuration	0x0028
	The EMI provides a generic register to allow the configuration of the padlogic. ST40RA uses the bits detailed.		
0	SOFE	**Strobe positioning** Strobe on falling edge: 0: Disabled 1: Enabled Reset: 0	RW
[5:1]	SDPOS	**SDRAM bank location** 00001: Bank 0 00010: Bank 1 00011: Bank 2 00100: Bank 3 00101: Bank 4 00110: Bank 5 10001: Bank 0 to 1 10010: Bank 0 to 2 10011: Bank 0 to 3 10100: Bank 0 to 4 10101: Bank 0 to 5 10110: Bank 1 to 2 10111: Bank 1 to 3 11000: Bank 1 to 4 11001: Bank 1 to 5 11010: Bank 2 to 3 11011: Bank 2 to 4 11100: Bank 2 to 5 11101: Bank 3 to 4 11110: Bank 3 to 5 11111: Bank 4 to 5 Reset: 0	RW
6	EWPU	**Pull-up on EWAIT pin**[a] 0: Disabled 1: Enabled Reset: 0	RW
7	EAPU	**Pull-up enable on EADDR pins** 0: Disabled 1: Enabled Reset: 0	RW
[31:8]	Reserved	0: Ignored 1: Reserved Reset: Undefined	

a. If the EWAIT signal is set at the beginning of an access, and the data is to be set after the EWAIT is cleared, the parameters ACCESSTIMEREAD and LATCHPOINT in the EMI configuration registers must be set as follows:

ACCESSTIME > LATCHPOINT + 3.

See the *ST40 System Architecture Manual, Volume 2: Bus Interfaces* for details of setting the EMI configuration registers.

5.9.2 LMI.COC

	LMI.COC	LMI clock and pad control	0x028
	LMI.COC allows modification of the glue logic.		
0	DLY_SRC	**Delay line control source** 0: DLL provides delay line control 1: LMI.CFG[5:1] provides delay line control Reset: 0	RW
[5:1]	DLY_NUM	**Number of delays (~200ps each)** Reset: 0	RW
[7:6]	DLY_FRQ_RES	**External delay frequency resolution** Reset: 0	RW
19:8]	PLL_SETUP	**PLL setup** Reset: 0	RW
[21:20]	DLL_PRO_CON	**DLL programmer control** Reset: 0	RW
22	FRQ_RES_SRC	**Frequency resolution source of external delay** 0: DLL provides frequency resolution 1: LMI.CFG[7:6] provides frequency resolution Reset: 0	RW
23	PLL_SETUP	**PLL setup** Reset: 0	RW
24	DLL_PRO_SRC	**DLL programmer source** 0: Delay programmer block provides DLL programming 1: LMI.CFG[21:20] provides DLL programming Reset: 0	RW
[30:25]	Reserved		
31	DLL_ENB	**DLL enable** Reset: 0	RW

5.9.3 LMI.CIC

	LMI.CIC	LMI clock and pad status	0x040
	LMI.CIC reflects the status of the glue logic.		
[4:0]	DLY_STATE	**DLL delay state**	RO
5	DLL_LOCK	**DLL lock signal**	RO
6	PLL_LOCK	**PLL lock signal**	RO
[8:7]	DLL_STATE	**DLL state**	RO
[21:9]	PLL_SETUP_STATE	**PLL setup state**	RO
[24:22]	DLL_SETUP_STATE	**DLL setup state**	RO
[26:25]	DLL_BYPASS	**DLL bypass state**	RO
[31:27]	LMI_SETUP	**LMI.CFG setup for external delay**	RO

5.9.4 SYSCONF registers

All ST40 systems contain a number of general purpose configuration registers which may be used to configure system logic.

The definition of the general registers and their access functions is defined in the *ST40 System Architecture Manual.*

For ST40RA the bits within these registers have the following function.

5.9.4.1 SYSCONF.SYS_STAT1.

	SYS_STAT1	Memory bridge status	0x0040
[3:0]	Reserved		
[10:4]	STATUS1	**Status memory bridge 1**	RO
[17:11]	STATUS2	**Status memory bridge 2**	RO
[24:18]	STATUS3	**Status memory bridge 3**	RO
[31:25]	STATUS4	**Status memory bridge 4**	RO
[38:32]	STATUS5	**Status memory bridge 5**	RO
[45:39]	STATUS6	**Status memory bridge 6**	RO
[52:46]	STATUS7	**Status memory bridge 7**	RO
[63:53]	Reserved		

5.9.4.2 SYSCONF.SYS_CON1.

	SYSCONF.SYS_CON1	Memory bridge control	0x010
[3:0]	Reserved		RW
[10:4]	MB1	Memory bridge 1 control: EMI target	RW
[17:11]	MB2	Memory bridge 2 control: EMPI initiator	RW
[24:18]	MB3	Memory bridge 3 control: EMI_SS target	RW
[38:25]	Reserved		
[45:39]	MB6	Memory bridge 6 control: PCI initiator	RW
[52:46]	MB7	Memory bridge 7 control: PCI target	RW
[63:53]	Reserved		

Where the two clocks are sourced from independent PLLs the bridge must be put in asynchronous mode.

5.9.5 SYSCONF.SYS_CON2.

	SYSCONF.SYS_CON2	Functional pin use and behavior	0x0018
The SYSCONF.SYS_CON2 register controls functional pin use and behavior			
8	LMI_MODE	**LMI pad type** 0: SSTL 1: LVTTL Reset: 0	RW
9	LMI_ENVREF	**Reference voltage source** 0: internally generated reference voltage 1: external reference voltage from VREF pins Reset: 0	RW
10	LMI_ECLK_BYPASS	**LMI control signal ECLK180 retime bypass** 0: ECLK180 flip flop not bypassed 1: ECLK180 flip flop is bypassed Reset: 0	RW
11	LMI_NOTCOMP25_EN	**Enable LMI 2.5 V compensation cell** 0: LMI 2.5 V compensation cell enabled 1: LMI 2.5 V compensation cell disabled Reset: 0	RW
12	LMI_COMP33_EN	**Enable LMI 3.3 V compensation cell** 0: LMI 2.5 V compensation cell enabled 1: LMI 2.5 V compensation cell disabled Reset: 0	RW

	SYSCONF.SYS_CON2	Functional pin use and behavior	0x0018
[13:14]	LMI_SDRAM_DATA_DRIVE	**SDRAM data and data strobe pad PROG 1:0 LVTTL OP drive strength** 00: 1x 01: 2x 10: 3x 11: 4x Reset: 0	RW
[15:16]	LMI_SDRAM_ADD_DRIVE	**LMI address and control pad PROG 1:0 LVTTL OP drive strength** 00: 1x 01: 2x 10: 3x 11: 4x Reset: 0	RW
[17:35]	Reserved		
36	EMPI_ENB[0]	**Enable EMPI channel 0 DREQ/DRACK/DRACK alternate function** 0: Disabled 1: NOTESCS1 remapped to EMPIDREQ0 NOTESCS2 remapped to EMPIDRAK0 EADDR26 remapped to EMPIDACK0 EADDR26 is only remapped when whilst the ST40RA is acting as a bus slave	RW
37	EMPI_ENB[1]	**Enable EMPI channel 1 DREQ/DRACK/DRACK alternate function** 0: Disabled 1: NOTPREQ3 remapped to EMPIDREQ1 NOTPGNT3 remapped to EMPIDRAK1 EADDR25 remapped to EMPIDACK0 EADDR25 is only remapped when whilst the ST40RA is acting as a bus slave	RW
38	EMPI_ENB[2]	**Enable EMPI channel 2 DREQ/DRACK/DRACK alternate function** 0: Disabled 1: DREQ0 remapped to EMPIDREQ2 DACK0 remapped to EMPIDACK2 DRAK0 remapped to EMPIDRAK2	RW
39	EMPI_ENB[3]	**Enable EMPI channel 2 DREQ/DRACK/DRACK alternate function** 0: Disabled 1: DREQ1 remapped to EMPIDREQ3 DACK1 remapped to EMPIDACK3 DRAK1 remapped to EMPIDRAK3	RW
40	MAILBOX_ENB	**Enable mailbox interrupt alternate function** 0:Disabled 1:NOTESC0 remapped to MBXINT	RW
[41:43]	Reserved		

	SYSCONF.SYS_CON2	Functional pin use and behavior	0x0018
[44:46]	EMPI_CS_ENB	**Enable EMPI chip selection alternate function** 000: NOTESC0 remapped to NOTEMPICS 001: NOTESC1 remapped to NOTEMPICS 010: NOTESC2 remapped to NOTEMPICS 011: NOTESC3 remapped to NOTEMPICS 100: NOTESC4 remapped to NOTEMPICS 101: NOTESC5 remapped to NOTEMPICS 110: Reserved 111: Disabled (value at reset)	RW
47	SEL_EXT_EMI_SLAVE	**Select EMI slave or master functionality** 0: EMI is bus master 1: EMI is bus slave	RW
[48:59]	Reserved		
[60:63]	PIO_CONF	**PIO_CONF**	RW

5.9.6 PIO alternate functions

The function of pads with PIO alternate functions are controlled by the PIO.PC0, PIO.PC1 and PIO.PC2 registers.

In the ST40RA device, the operational modes for these registers differ from the standard architecture definition and are shown in *Table 15*.

Table 15: PIO alternate function registers

PIO bit configuration	PIO output state	PIO.PC2	PIO.PC1	PIO.PC0
NonPIO function[a]	-	0	0	0
PIO bidirectional	Open drain	0	0	1
PIO output	Push-pull	0	1	0
PIO bidirectional	Open drain	0	1	1
PIO input	High impedance	1	0	0
PIO input	High impedance	1	0	1
Reserved	-	1	1	0
Reserved	-	1	1	1

a. State following reset

5.9.7 PCI.PERF register definition.

	PCI.PERF		0x0080
	PCI.PERF modifies the function of the PCI.		
[3:0]	DLY_PERRSAMPLE	**Parity error delay** Number of APP_CLOCK cycles after end of PCI that access master should wait to see if there is a parity error	RW
4	ENB_WRITEPOST	**Enable write posting in master**	RW
5	ENB_STBYBYPASS	**Enable standby bypass**	RW
[31:6]	Reserved		

6 Clock generation

The ST40 clock architecture has been organized to maintain compatibility across the ST40 family and allow additional flexibility to increase system performance where required. It includes a more diverse range of peripherals and provides low power use.

6.1 Clock domains and sources

Figure 5 shows possible clock domains for ST40RA clocks. The ST40RA implementation includes two CLOCKGEN macros, which supply up to three independent clock domains across the chip

Each PLL may be independently programmed to produce a clock at a specific frequency which is used to derive a series of related clocks which may be used by the system.

The clock domains mapping is shown in *Table 16*. The architecture of the ST40RA CLOCKGEN subsystem consists of two standard (ST40 family) CLOCKGEN units (CLOCKGENA and CLOCKGENB) and a CLOCKCON block. *Figure 6* shows the architecture of the ST40RA CLOCKGEN subsystem.

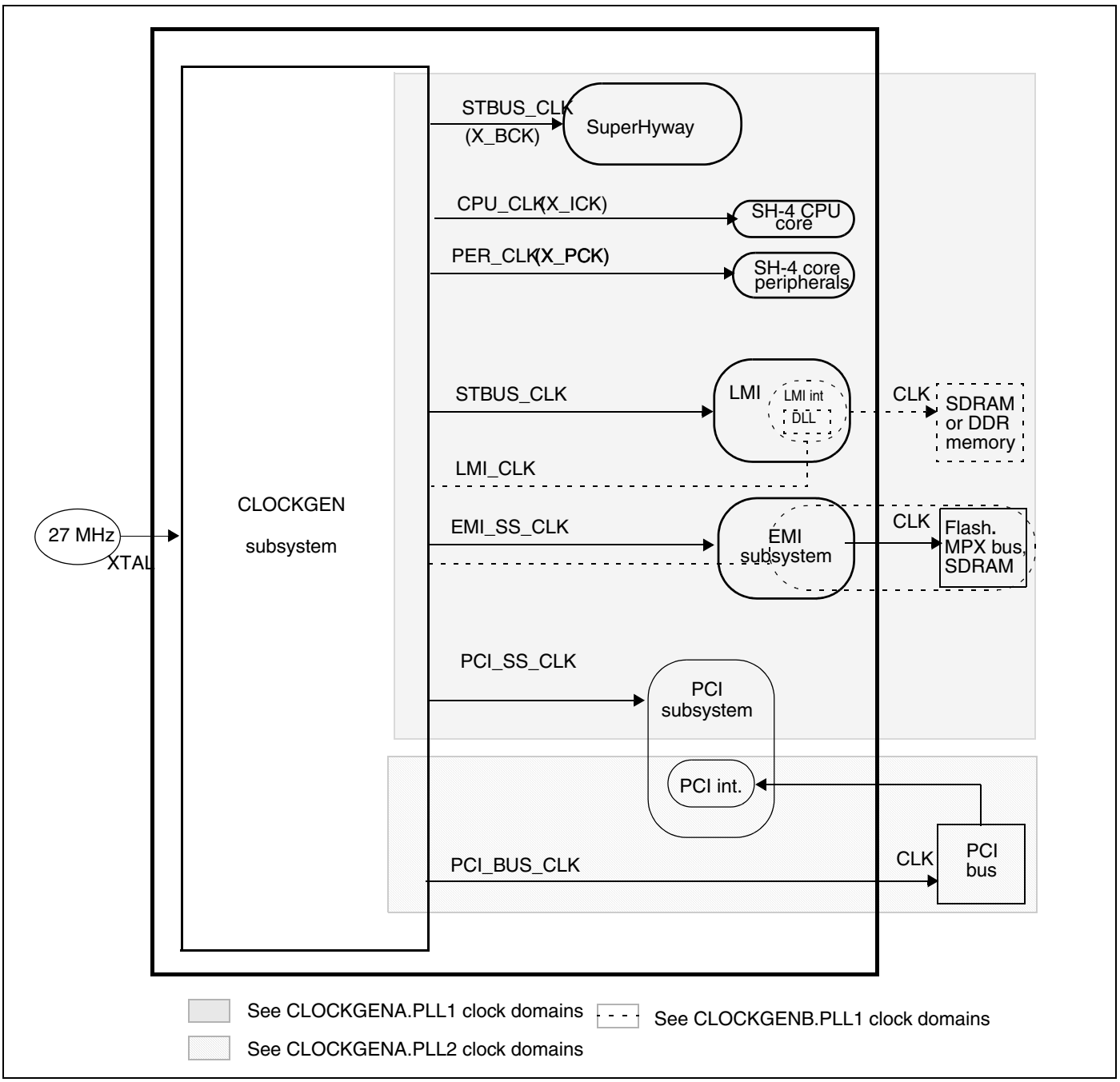


Figure 5: ST40RA clock domains

Table 16: Clock domains

Subsystem	Clock domain	Target frequencies (MHz)				Source[a]	Ratio
CPU core	CPU_CLK	200	166	150	133	CLOCKGEN_A11	1
SuperHyway	STBUS_CLK	-	111	100	88	CLOCKGEN_A12	2/3
		100	83	75	67		1/2
Peripherals	PER_CLK (CPU core PCK)	-	55	50	44	CLOCKGEN_A13	1/3
		50	42	38	33		1/4
PCI bus clock	PCI_BUS_CLK	33				CLOCKGEN_A21	1/16
		66				CLOCKGEN_A22	1/8
		25.14				CLOCKGEN_A23	1/21
		Disabled				CLOCKGEN_A24	-
PCI subsystem	PCI_SS_CLK	-	111	100	88	CLOCKGEN_A12	2/3
		100	83	75	67		1/2
		-	55	50	44	CLOCKGEN_A13	1/3
		50	42	38	33		1/4
Local memory interface (LMI)	LMI_CLK	133	111	100	88	CLOCKGEN_A14	2/3
		Reserved				CLOCKGEN_B11	1
EMI subsystem	EMI_CLK	50 to 100 MHz				CLOCKGEN_B12	1
		-	111	100	88	CLOCKGEN_A12	2/3
		100	83	75	67	CLOCKGEN_A14	1/2

a. Clock naming: CLOCKGEN_[CLOCKGEN label][PLL number][clock number]

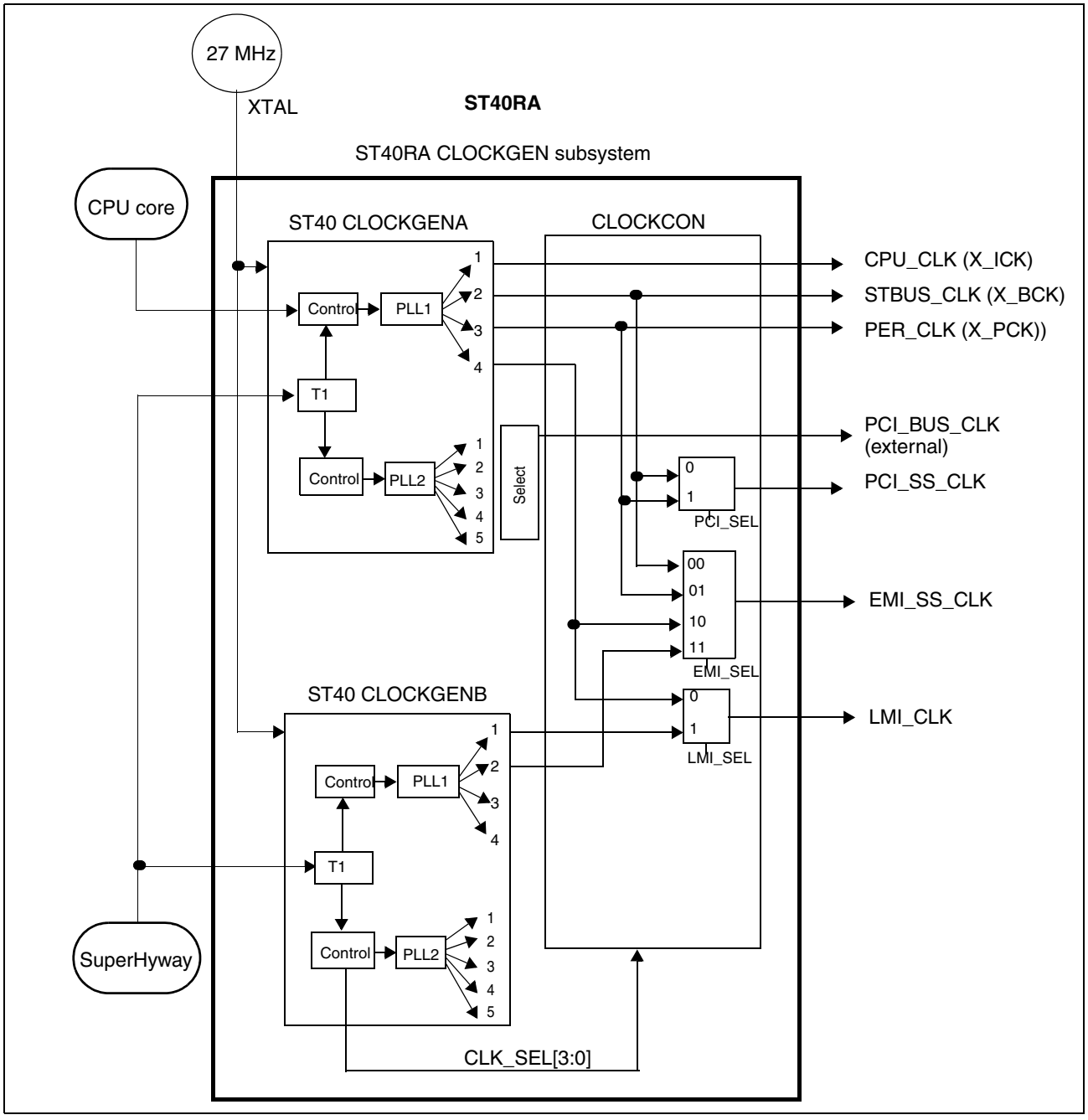
The sources for PCI_SS_CLK and EMI_SS_CLK, can be set using the PCI_SEL and EMI_SEL bits in the CLOCKGENB.CLK_SELCR register. See *Section 6.6.1: CLOCKGENB.CLK_SELCR register on page 52*.

If CLOCKGEN_A13 is used as PCI_SS_CLK source then the memory bridges 6 and 7 must be enabled. If CLOCKGEN_A12 is used, then the bridges may be placed in bypass mode. This is the recommended mode of operation.

If either CLOCKGEN_B12 or CLOCKGEN_A14 are used as the EMI_CLK, the memory bridges 1, 2 and 3 must be enabled. If CLOCKGEN_A12 is used, then the bridges may be placed in bypass. This is the recommended mode of operation.

See *Chapter 5.8: Memory bridge control on page 34*.

Figure 6: ST40RA CLOCKGEN subsystem



6.2 Recommended operating modes

Table 17: Supported operating frequencies

Mode for CLOCKGENA and CLOCKGENB		PLL frequency (MHz)		ST40RA clock domain frequencies (MHz)					
PLLA (mode)	PLLB (mode)	PLLA	PLLB	CPU_ CLK	STBUS_ CLK	PER_ CLK	LMI_ CLK	EMI_SS_ CLK	PCI_SS_ CLK
Recommended reset configuration									
0	-	200	-	100	50	25	50	50	50
Alternate reset configuration									
1	-	266	-	133	88	44	88	88	88
2	-	300	-	150	100	50	100	100	100
3	-	332	-	166	111	66	111	111	111
Recommended operating modes									
2	-	300	-	150	100	100	100	100	100
3	-	332	-	166	83	83	83	83	83
Low power configuration with clocks enabled (programmable after reset)									
A6 bypass	-	27	-	13.5	6.75	6.75	6.75	6.75	6.75

6.3 Clocks and registers at start up

Reset mode	Reset mode MODE[2:0]	CLOCKGENA .PLL1CR1 reset value	CLOCKGENA core frequency (PLL1)	$f_{PLL}/2$	CLK1 CPU_ CLK	CLK2 STBUS_ CLK	CLK3 PER_ CLK	CLK4 LMI_ CLK
0	000	0x7939 8612	200 MHz	100	1	1/2	1/4	1/2
1	001	0x7939 B112	266 MHz	133	1	2/3	1/3	2/3
2	010	0x7938 6412	300 MHz	150	1	2/3	1/3	2/3
3	011	0x7938 7B14	332 MHz	166	1	2/3	1/3	2/3
4	100	0x7938 8612	400 MHz	200	1	1/2	1/4	1/2
5	101	0x7938 A712	500 MHz	250	1	1/2	1/4	1/2
6	110	0x0938 0000	0 MHz	0	1	1/2	1/2	1/2
7	111	0x0939 8612	200 MHz	100	1/2	1/4	1/4	1/4

Table 18: CLOCKGENA PLL1 reset values

6.3.1 CLOCKGENA_2x PCI (PCI_DIV_BYPASS = 0)

Table 19: CLOCKGENA PLL2 reset values (PCI_DIV_BYPASS = 0)

Reset mode MODE[4:3]	Reset value	PLL2 frequency
00	0x7938 B012	528 MHz
01	0x7938 B012	528 MHz
10	0x7938 B012	528 MHz
11	0x0938 B012	0 MHz

6.3.2 Division ratios on CLOCKGENA_2x

Table 20: CLOCKGENA_PLL2 PCI reset division ratios.

Mode MODE[4:3]	Divide ratio selected	PCI_BUS_CLK freq.
00	8	66 MHz
01	16	33 MHz
10	21	25.14 MHz
11	-	0 MHz

6.4 Setting clock frequencies

Table 21 shows valid FRQCR ratios and the associated clock frequencies for derived clocks.

Table 21: Valid FRQCR values and their ratios

CLOCKGENA.FRQCR and CLOCKGENB.FRQCR		ST40RA codified ratios			Clock ratios		
Lower 9 bit	Available on start up	CPU_ CLK	BUS_ CLK	PER_ CLK	CPU_ CLK	BUS_ CLK	PER_ CLK
0x000		1	1	1/2	1	1	1/2
0x002				1/4	1	1	1/4
0x004				1/8	1	1	1/8
0x008	MODE6	1	1/2	1/2	1	1/2	1/2
0x00A	MODE[4:5]			1/4	1	1/2	1/4
0x00C			1/2	1/8	1	1/2	1/8
0x011			2/3	1/6	1	2/3	1/6
0x013	MODE[2:3]	1	2/3	1/3	1	2/3	1/3
0x01A	MODE0		1/2	1/4	1	1/2	1/4
0x01C				1/8	1	1/2	1/8
0x023	MODE1	1	2/3	1/3	1	2/3	1/3
0x02C			1/2	1/8	1	1/2	1/8
0x048		1/2	1/2	1/4	1	1	1/2
0x04A				1/6	1	1	1/3
0x04C				1/8	1	1	1/4
0x05A			1/3	1/6	1	2/3	1/6
0x05C							
0x063	MODE7	1/2	1/4	1/4	1	1/2	1/2
0x06C		1/2		1/8	1	1/2	1/4
0x091		1/3	1/3	1/6	1	1	1/2
0x093							
0x0A3			1/6		1	1/2	1/2
0x0DA		1/4	1/4	1/8	1	1	1/2
0x0DC							
0x0EC			1/8		1	1/2	1/2
0x123			1/4		1	1	1/2
0x16C			1/8		1	1/2	1/2

6.4.1 Programming the PLL output frequency

The three dividers used within the PLL are referred to as M (predivider), N (feedback divider) and P (postdivider) for brevity. Note that there is a divide-by-2 fixed prescaler before the feedback divider. The binary values applied to the programmable dividers, and the frequency of CLOCKIN controls the output frequency of the PLL macrocell:

$$F(clockout) = \frac{2 \times N}{M \times 2^P} \times F(clockin)$$

where the values of M, N and P must satisfy the following constraints:

- Divider limits: $1 \le M \le 255$, $1 \le N \le 255$, $0 \le P \le 5$,
- Phase comparator limits: $1MHz \le \frac{F(clockin)}{M} \le 2MHz$,
- VCO limit: $200MHz \le \left(\frac{2 \times N}{M}\right) \times F(clockin) \le 622MHz$,
- M divider limit: $F(clockin) \le 200MHż$.

For example, if 300 MHz from an input clock of 33 MHz is to be generated, the values of M, N and P are worked out as below.

1 The phase comparator must operate between 1 MHz and 2 MHz, so choose M = 22 (for 1.5 MHz operation).

2 The VCO needs to run between 200 MHz and 622 MHz. It could be run at 300 MHz directly (which takes a little less current), or at 600 MHz then divide by 2 to ensure an exact 50% duty cycle. In this example 600 MHz is chosen so N = 200.

3 The postdivider then needs to be a divide by 2. This is programmed in powers of 2, so P = 1.

The P divider changes value without glitching of the output clock.

6.4.2 Changing clock frequency

The clock frequencies are changed in two ways.

- Change the core PLL frequencies.

  The PLL must be stopped, the control register reconfigured with the new settings, and the PLL restarted at the new frequency.
- Change the frequency division ratio of the clock domains.

  The control registers are changed dynamically and the new frequencies are effective immediately.

6.4.3 Changing the core PLL frequencies

This procedure applies to either CLOCKGENA or CLOCKGENB and to PLL1 or PLL2.

1 Stop the PLL. The CLOCKGENA.PLL1CR2.STBPLLENSEL register selects whether the PLL is enabled by the CLOCKGENA.PLL1CR2.STBPLLEN or the CPG.FRQCR.PLL1EN register.

2 Reconfigure the PLL. Set the CLOCKGENA.PLL1CR1 register to one of the supported configurations on the datasheet.

3 Restart the PLL, following the procedure described in the *ST40 System Architecture Volume 1: System*.

6.4.4 Changing the frequency division ratio

The frequency division ratio is selected by changing the CPG.FRQCR register for PLL1 or the CLOCKGENA.PLL2_MUXCR register for PLL2. This change is immediately effective.

6.5 Power management

The power management unit (PMU) is responsible for clock startup and shutdown for each of the on-chip modules. Power is conserved by powering down those modules which are not in use, or even the CPU itself.

The PMU is operated using three banks of registers as follows:

- **CPG:** controls the power-down mode of the CPU and the power-down states of the legacy on-chip peripherals,
- **CLOCKGENA** and **CLOCKGENB:** control the power-down states of the other on-chip peripherals.

6.5.1 CPU low-power modes

The CPU can be put into sleep or standby modes. In sleep mode the CPU is halted while the on-chip peripherals continue to operate. In standby mode all the on-chip peripherals are stopped along with the CPU. In addition, the on-chip peripherals can be independently stopped.

Power down is initiated with the **sleep** instruction and the power down mode is selected with bit 7 of the CPG.STBCR register. If the bit is set, the CPU enters standby mode on the next **sleep** instruction, and if unset it enters sleep mode.

6.5.2 Module low-power modes

Modules are powered down in two ways, depending on whether the module is a ST40 legacy peripheral (controlled by the CPG register bank) or a ST40RA peripheral (controlled by the CLOCKGEN register banks).

A module controlled by the CPG register bank has its clock stopped when the corresponding bit in the CPG.STBCR or CPG.STBCR2 register is set. The clock is started again when the bit is cleared.

To request the power down of a module controlled by the CLOCKGENA or CLOCKGENB register bank, 1 is written to the corresponding bit in the STBREQCR_SET register. When the module has completed its power down sequence and its clock has been stopped, the corresponding bit in the STBACKCR register is set. To restart the module, 1 is written to the corresponding bit in the STBREQCR_CLR register.

Note: The modules governed by the CLOCKGENB register bank do not support hardware-only power down and require software interaction to maintain data coherency before making a request to stop the module clock.

6.6 Clock generation registers

6.6.1 CLOCKGENB.CLK_SELCR register

	CLOCKGENB.CLK_SELCR	Clock source selection	0x0068
The CLKGENB.CLK_SELCR register controls the selection of clock domain clock sources			
0	LMI_SEL	**Reserved** Reset state: 0	
1	PCI_SEL	**Select PCI clock** 0: PCI_SS_CLK from CLOCKGENA_12 1: PCI_SS_CLK from CLOCKGENA_13 Reset state: 0	RW
[2:3]	EMI_SEL	**Select EMI clock** 00: EMI_SS_CLK from CLOCKGENA_12 01: EMI_SS_CLK from CLOCKGENA_13 10: EMI_SS_CLK from CLOCKGENA_14 11: EMI_SS_CLK from CLOCKGENB_12 Reset state: 00	RW
[4:7]	EXT_CLK_SEL	Not used Reset state: 0000	
[8:31]	Reserved	Reset state: 0	RW

6.6.2 CPG.STBCR register

	CPG.STBCR	Sleep or standby mode	0x0004
Select between sleep and standby modes when a sleep instruction is issued.			
0	MSTP0	**SCIF1 standby** 0: SCIF1 operates 1: SCIF1 clock stopped Reset state: 0	RW
1	MSTP1	**RTC standby** 0: RTC operates 1: RTC clock stopped Reset state: 0	RW
2	MSTP2	**TMU standby** 0: TMU operates 1: TMU clock stopped Reset state: 0	RW
3	MSTP3	**SCIF2 standby** 0: SCIF2 operates 1: SCIF2 clock stopped Reset state: 0	RW
4	MSTP4	Not used Reset state: 0	RW
5	PPU	**Peripheral module pull-up pin control** Controls the state of peripheral module related pins in the high impedance state 0: Peripheral module related pin pull-up resistors are on 1: Peripheral module related pin pull-up resistors are off Reset state: 0	RW
6	PHZ	**Peripheral module pin high impedance control** Controls the state of peripheral module related pins in standby mode 0: Peripheral module related pins are in normal state 1: Peripheral module related pins go to high impedance state Reset state: 0	RW
7	STBY	**Standby** 0: Transition to sleep mode on **sleep** instruction 1: Transition to standby mode on **sleep** instruction Reset state: 0	RW

6.6.3 CLOCKGENA.STBREQCR and CLOCKGENB.STBREQCR registers

	CLOCKGENA.STBREQCR CLOCKGENB.STBREQCR	Control power down requests	0x0018
This register gives direct access to the power down request register. Low power requests are made in the STBREQCR_SET register and cleared in the STBREQCR_CLR register.			
[0:7]	REQ[0:7]	**Power down requests for module [n]** Controls the power down state for module [n] Bit [n]: 0 Request module [n] to operate normally Bit [n]: 1 Request module [n] to power down Reset state: 0	RW
[8:31]	Reserved	0: No action 1: Undefined Reset state: Undefined	

6.6.4 CLOCKGENA.STBREQCR_SET and CLOCKGENB.STBREQCR_SET registers

	CLOCKGENA.STBREQCR_SET CLOCKGENB.STBREQCR_SET	Set power down requests	0x0020
This register sets a low power request.			
[0:7]	SET[0:7]	**Set power down request for module [n]** Sets the power down request state for module [n] Bit [n]: 0 No action Bit [n]: 1 Set power down request Reset state: 0	WO
[8:31]	Reserved	0: No action 1: Undefined Reset state: Undefined	

6.6.5 CLOCKGENA.STBREQCR_CLR and CLOCKGENB.STBREQCR_CLR register

	CLOCKGENA.STBREQCR_CLR CLOCKGENB.STBREQCR_CLR	Clear power down requests	0x0028
This register clears a low power request and recommences the clock supply to a module.			
[0:7]	CLR[0:7]	**Clear power down request for module [n]** Clears the power down request state for module [n] Bit [n]: 0 No action Bit [n]: 1 Clear power down request Reset state: 0	WO
[8:31]	Reserved	0: No action 1: Undefined Reset state: Undefined	

6.6.6 CLOCKGENA.STBACKCR and CLOCKGENB.STBACKCR register

	CLOCKGENA.STBACKCR CLOCKGENB.STBACKCR	Current module power status	0x0030
This register indicates the current module power status			
[0:7]	ACK[0:7]	**Power down status for module [n]** Indicates the current power down status of the module [n] Bit [n]: 0 Module [n] operating normally Bit [n]: 1 Module [n] powered down Reset state: 0	RO
[8:31]	Reserved	0: No action 1: Undefined Reset state: Undefined	

Table 22 defines the mapping of modules to bits in the STBREQ and STBACK registers.

Table 22: STBREQ and STBACK mapping for modules

Bit number	CLOCKGENA mapping	CLOCKGENB mapping
0	EMI	Reserved
1	LMI	Reserved
2	DMAC	Reserved
3	PCI	Reserved
4	PIO	Reserved
5	Reserved	Reserved
6	Reserved	PCI bus
7	Reserved	Reserved

7 Electrical specifications

7.1 DC absolute maximum ratings

Table 23: Absolute maximum ratings

Symbol	Parameter	Min	Max	Units	Notes
VDD_{CORE}	Core DC supply voltage		2.1	V	a, b
VDD_{IO}	I/O DC supply voltage		4.0	V	
VDD_{RTC}	RTC DC supply voltage		2.1	V	
V_{IO}	Voltage on input, output and bidirectional pins.	GND -0.6	VDD_{IO} + 0.6	V	
V_{IORTC}	Voltage on input pins on VDD_{RTC} supply (LPCLKIN, LPCLKOSC)	GND -0.6	VDD_{RTC} + 0.6	V	
VIO_{CLK}	Voltage on CLKIN and CLKOSC pins	GND -0.6	VDD_{CORE} + 0.6	V	
I_O	DC output current		25	mA	
T_S	Storage temperature (ambient)	-55	125	deg C	
T_A	Temperature under bias (ambient)	-55	125	deg C	

a. Stresses greater than those listed under *Table 23: Absolute maximum ratings* may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operating sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended period may effect reliability.

b. All I/O pins are 3.3 V tolerant except CLKIN, LPCLKIN, CLKOSC and LPCLKOSC.

7.1.1 Fmax clock domains

Table 24: Fmax clock domains

Function clock	ST40RA200XH6	ST40RA166XH6	ST40RA150XH6
CPU_CLK	200 MHz	166 MHz	150 MHz
STBUS_CLK	100 MHz	111 MHz	100 MHz
PER_CLK	50 MHz	55 MHz	50 MHz
LMI_CLK	133 MHz	100 MHz	100 MHz
EMI_SS_CLK	100 MHz	111 MHz	100 MHz
EMI_EXT	100 MHz	100 MHz	100 MHz
PCI_EXT	66 MHz	66 MHz	66 MHz

7.1.2 Operating conditions

Table 25: Operating conditions

Symbol	Parameter	Min	Typical	Max	Units	Notes
VDD_{CORE}	Core positive supply voltage (ST40RA166/ST40RA150)	1.65	1.80	1.95	V	a
	Core positive supply voltage (ST40RA200)	1.80	1.87	1.95		
VDD_{IO}	I/O positive supply voltage	3.0	3.3	3.6	V	a
VDD_{RTC}	RTC positive supply voltage	1.65	1.8	1.95	V	
VDD_{MM}	VDD mismatch			0.3	V	b
LV_{REF}		1.15	VDD_{LMI} / 2	1.35	V	
VDD_{LMI}		3.0 2.3	3.3 2.5	3.6 2.7	V V	c d
V_{IH}	LVTTL input logic 1 voltage	2.0		VDD + 0.6	V	
V_{IH1}	LVTTL input 1 logic voltage EMODE pins	2.4		VDD + 0.6	V	
V_{IL}	LVTTL input login 0 voltage	-0.5		0.8	V	
V_{IHs}	SSTT_2 input login 1 voltage	LV_{REF} + 0.18		VDD_{LMI} + 0.3	V	
V_{ILs}	SSTT_2 input login 0 voltage	-0.3		LVREF - 0.18	V	
V_{OH}	LVTTL output logic 1 voltage	2.4			V	e
V_{OL}	LVTTL output logic 0 voltage			0.4	V	c
V_{OHs}	SSTT_2 output logic 1 voltage	2.1			V	e
V_{OLs}	SSTT_2 output logic 0 voltage			0.3	V	
I_{IN}	Input current (input pin)			+-10	uA	f
I_{OZ}	Offstate digital output current			+- 50	uA	f
I_{WP}	Input weak pull-up or pull-down current	20	60	110	uA	d
C_{IN}	Input capacitance (input pins)			10	pF	
C_{IO}	Input capacitance (bidirectional pins)		7	15	pF	

a. Either the I/O ring (VDD_{IO}) or the core (VDD_{CORE}) may be powered up first.

b. VDDCORE - VDDRTC

c. When in SDRAM mode

d. When in DDR-SDRAM mode

e. For specified output loads see *Table 27*.

f. 0 <= VI <= VDD

Table 26: Power dissipation

	VDD_{CORE}		VDD_{IO}		Units
	Typical	Maximum	Typical	Maximum	
Operating	850	1150	250	350	mW[a]
Low power	5	10	25	50	mW

a. CPU 166 MHz (Mode 3)

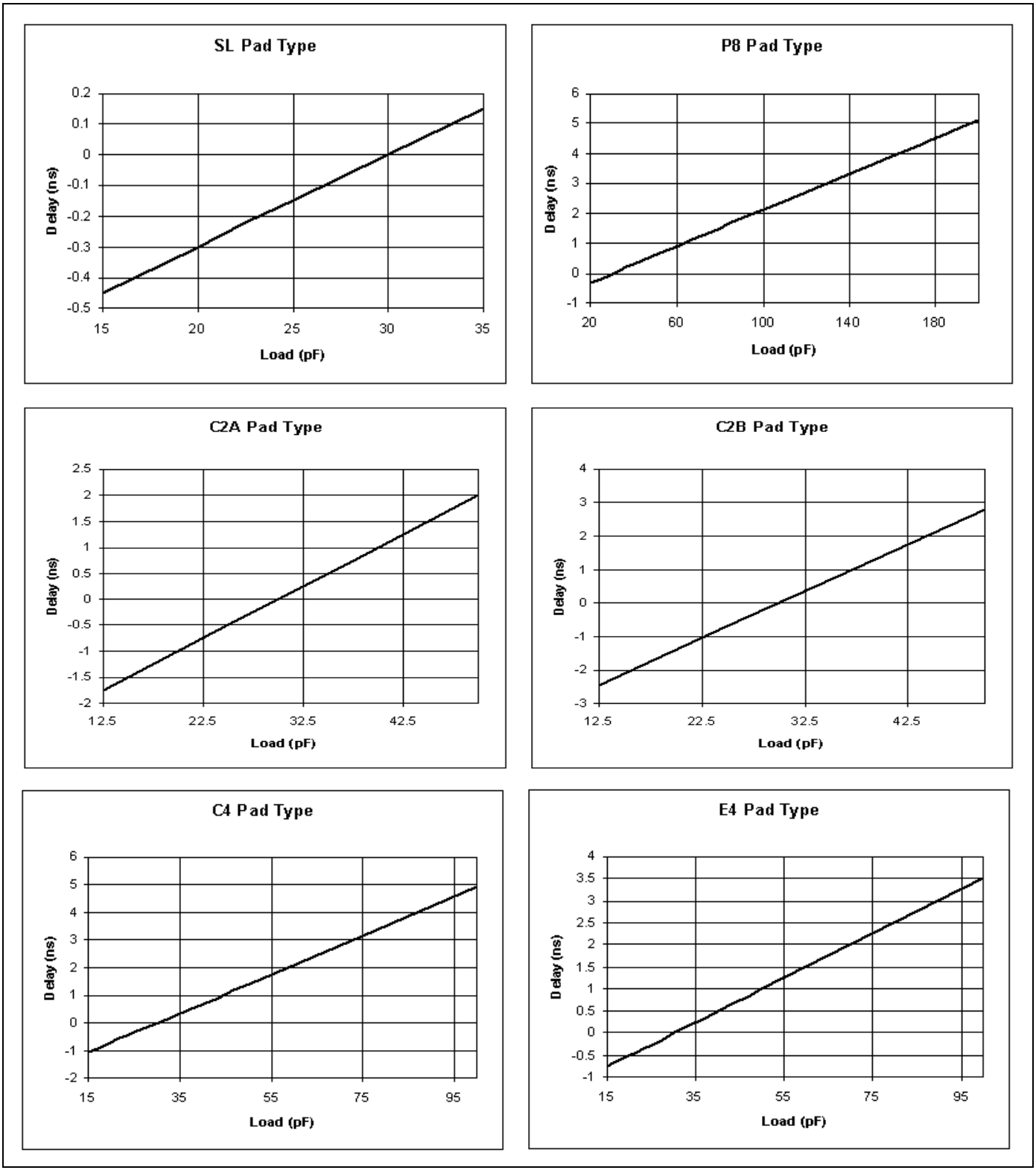
7.1.3 Pad specific output AC characteristics

Table 27: I/O maximum capacitive and DC loading

Pad type	Functional pin group	Maximum load (pf)	Drive (mA)	Notes
SL	LMI SDRAM/DDR	35	-	a
P8	PCI	200	8	
C2A		50	2	
C2B		50	2	
C4		100	4	
E4	EMI/MPX	100	4	

a. The SL pads are fully LVTTL and SSTL_2 compliant at maximum 35 pf load.

Figure 7: Pads characteristics



Note: 1.The SL pad type graph represents the maximum drive strength in the LVTTL mode.

7.2 Rise and fall times

Figure 8: Timings for C2A, C2B, E4 and C4 pad types

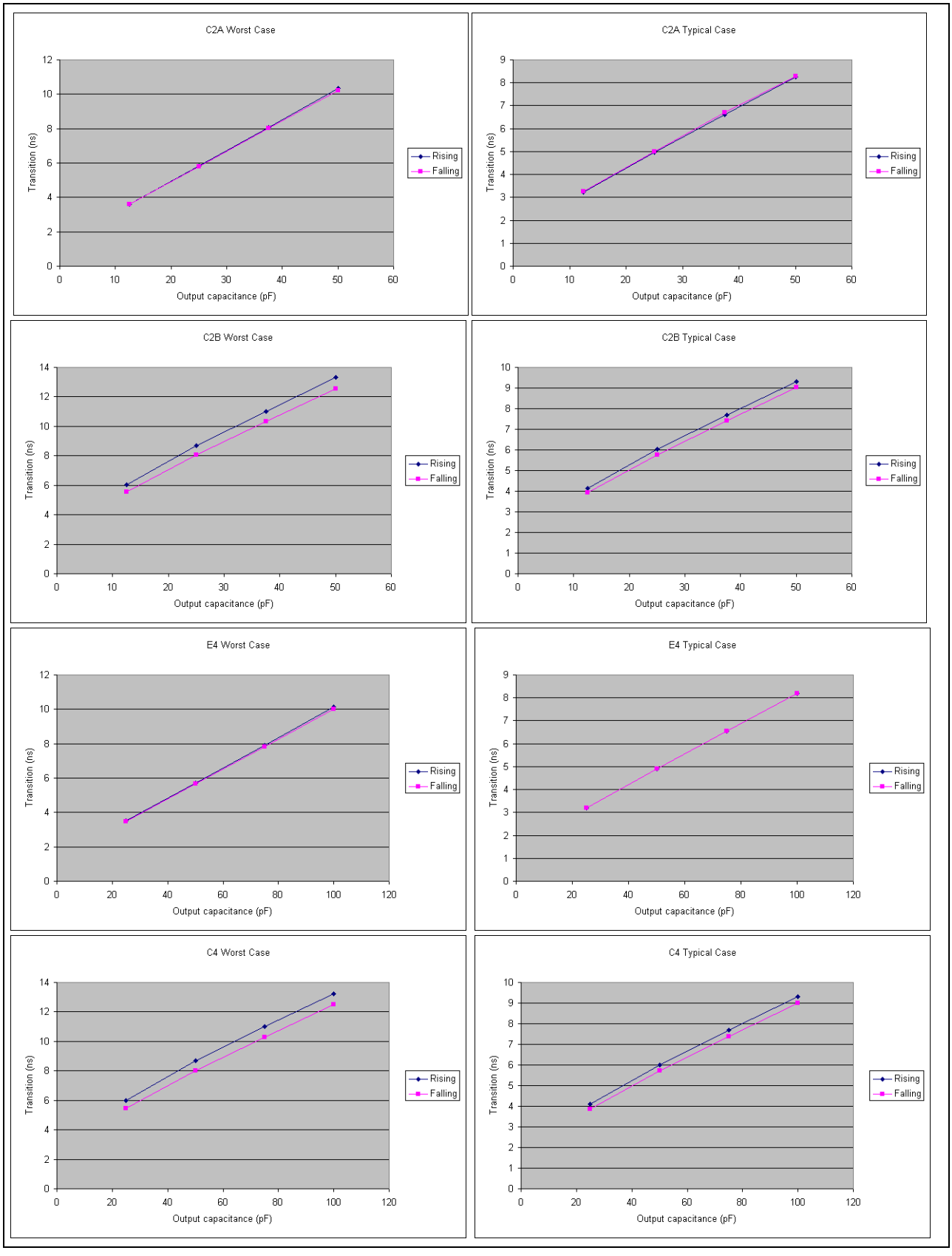


Figure 9: Timings for P8 and SL (LVTLL 00, 01 and 10) pad types

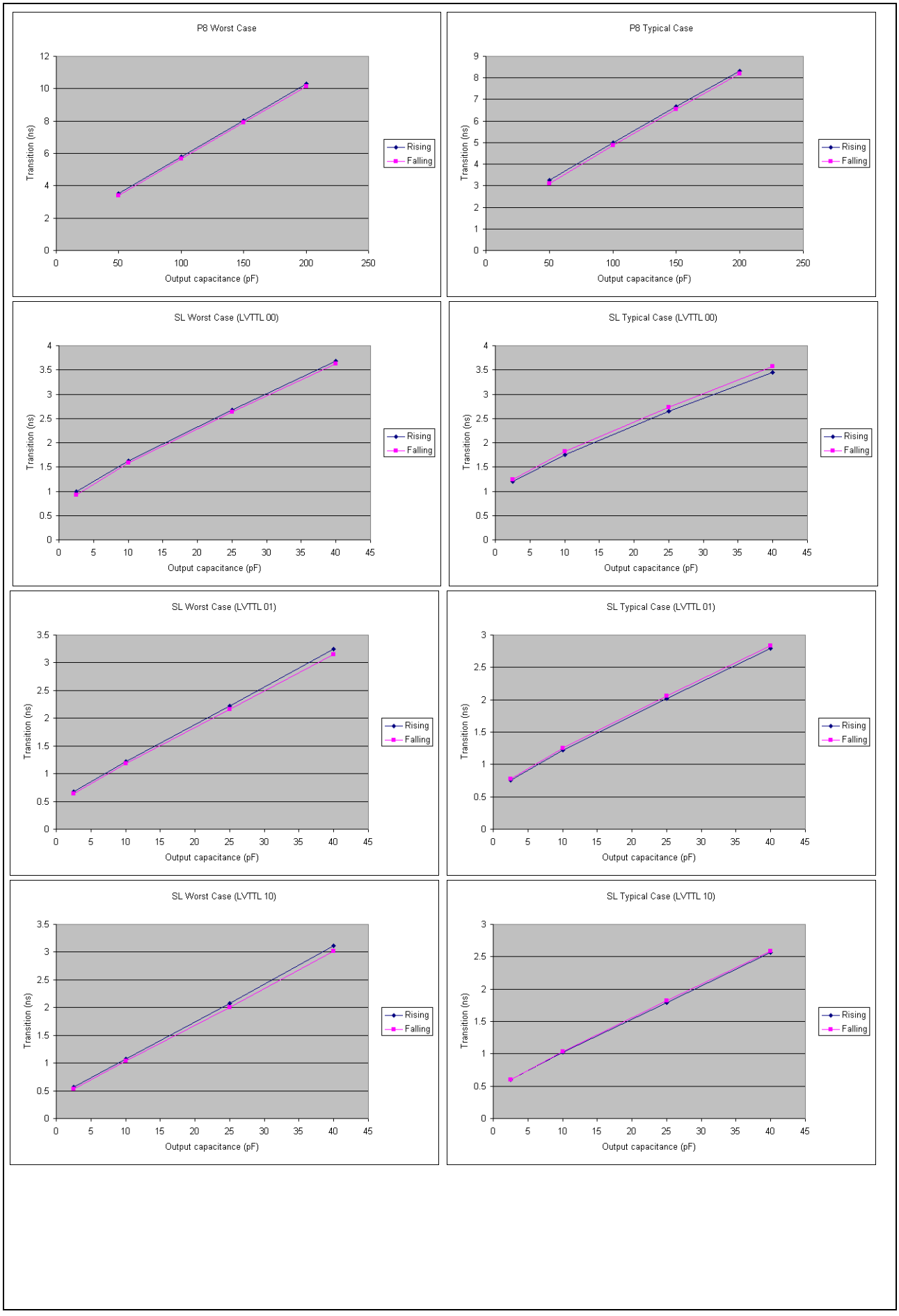
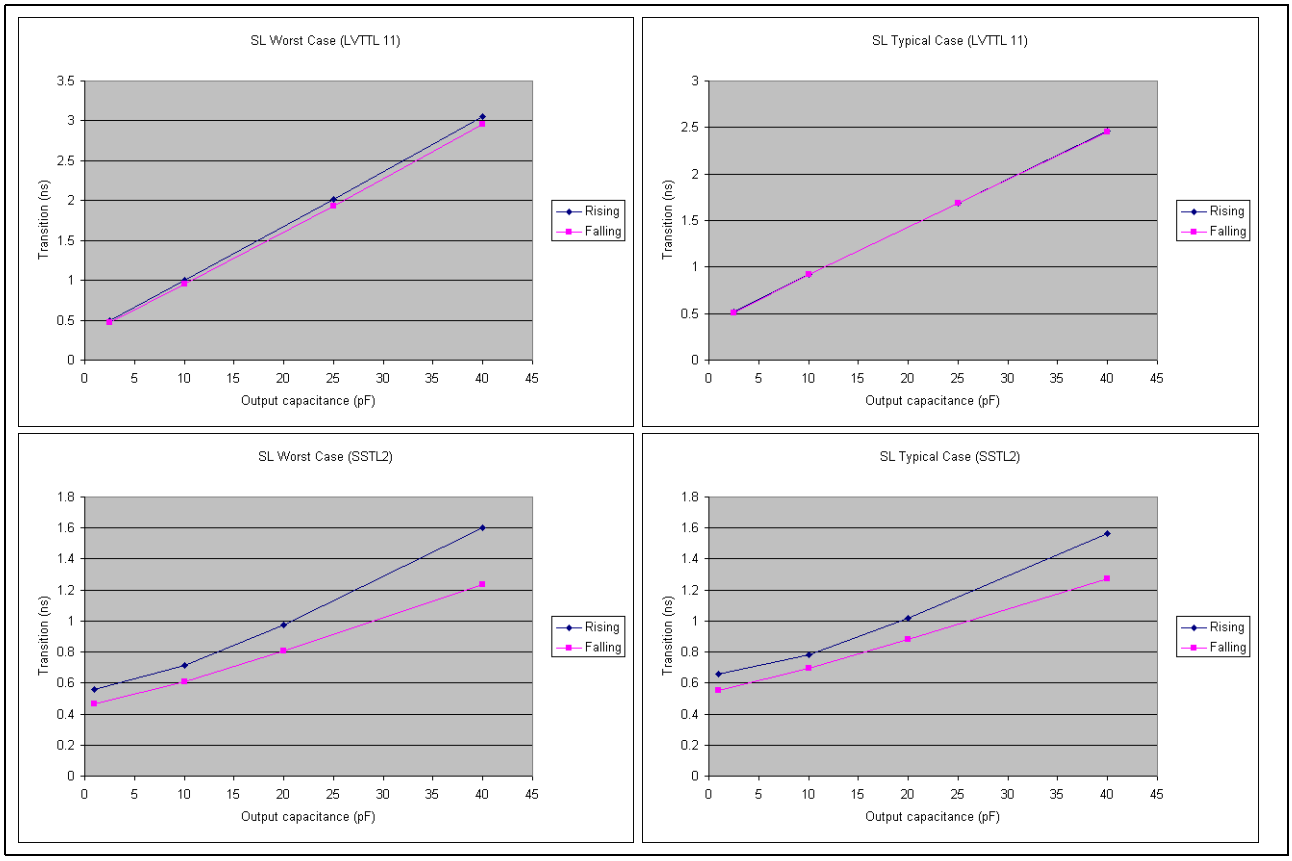


Figure 10: Timings for SL (LVTTL 11 and SSTL2) pad types



7.3 PCI interface AC specifications

Figure 11: PCI timings

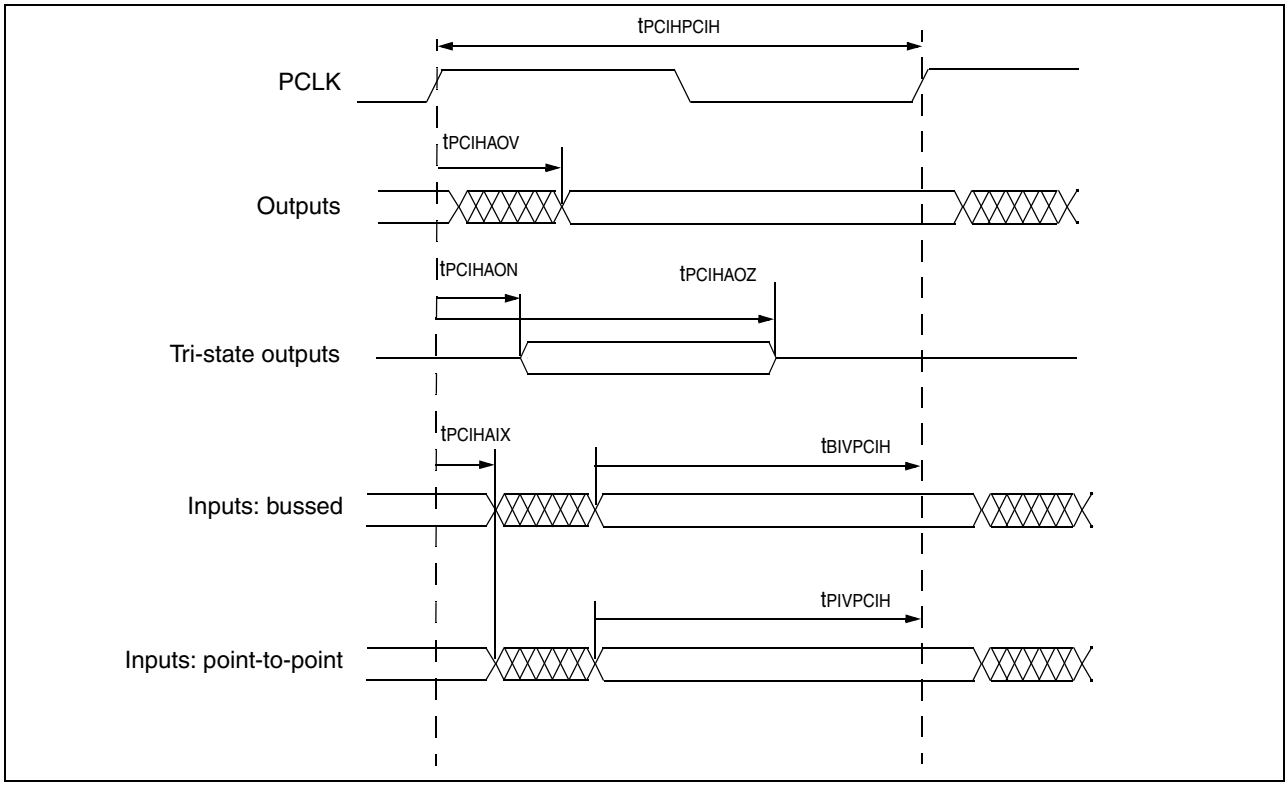


Table 28: PCI AC timings

Symbol	Parameter	Min	Max	Units	Note
tPCIHPCIH	PCI clock period	15		ns	a
tPCIHAOV	PCLK high to all PCI output signals valid	1	10	ns	a, b
tPCIHAOZ	PCLK high to all PCI outputs tri-state	2	14	ns	a
tPCIHAON	PCLK high to all PCI outputs on	2		ns	a
tBIVPCIH	Bused input signals valid to PCLK high	3		ns	c
tPIVPCIH	Point-to-point input signals valid to PCLK high	5		ns	b
tPCIHAIX	All PCI input signals hold after PCLK high	2		ns	

a. Specified with 30 pF load

b. Need to use 4 ns of the PCI propagation delay

c. NOTPREQ[0:3] and NOTPGNT[0:3] are point to point signals and have different input setup times to bussed signals. All other synchronous signals are bussed.

7.4 LMI interface (SDRAM) AC specifications

Figure 12: LMI SDRAM mode timings

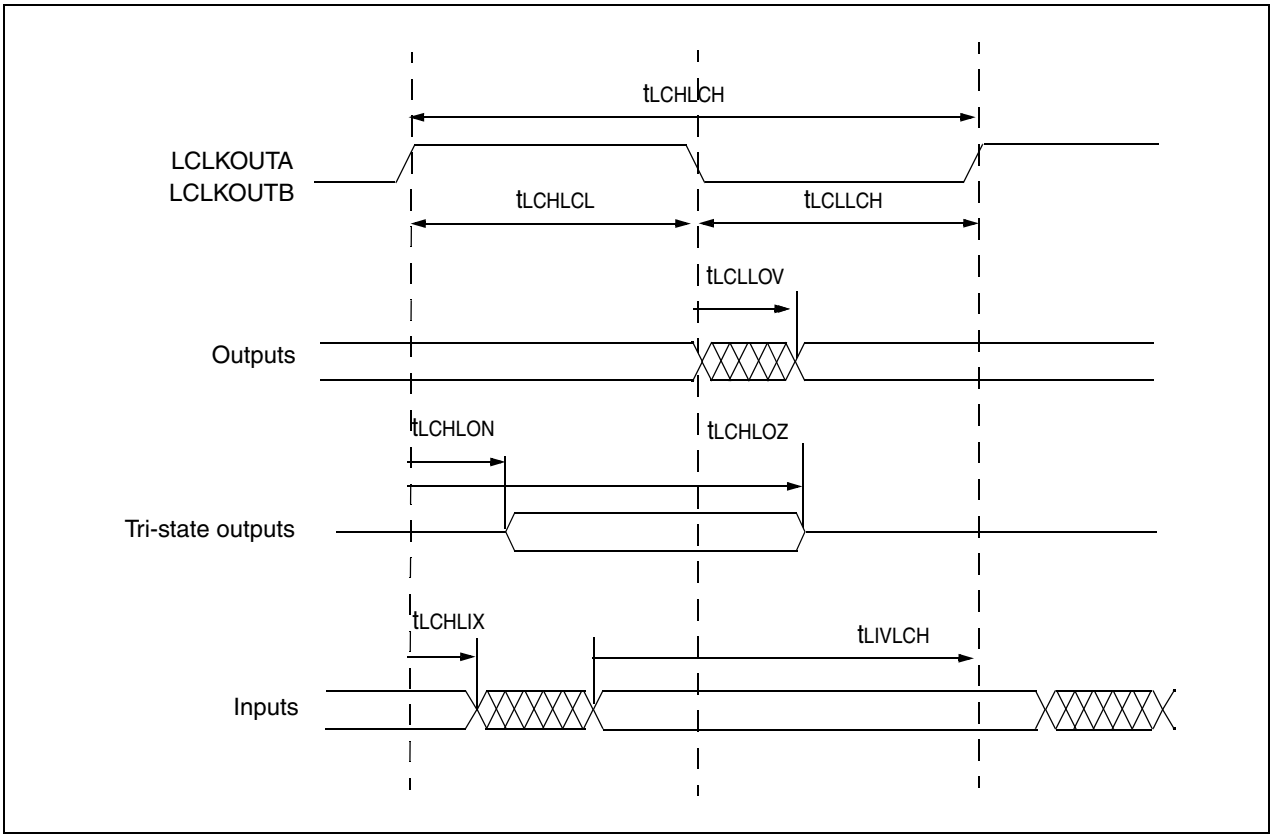


Table 29: LMI SDRAM AC timings

Symbol	Parameter	Min	Max	Units	Note
tLCHLCH	LMI clock period	10		ns	
tLCHLCL	LMI clock high time	0.45		tLCHLCH	
tLCLLCH	LMI clock low period	0.45		tLCHLCH	
tLCHLOV	LCLKOUT low to output signals valid	-2	2	ns	
tLCHLOZ	LCLKOUT high to outputs tri-state	0	2	ns	
tLCHLON	LCLKOUT high to outputs on	-2		ns	
tLIVLCH	Input signals valid to LCLKOUT high	2		ns	
tLCHLIX	Input signals hold after LCLKOUT high	2		ns	

7.5 LMI interface (DDR-SDRAM) AC specifications

Figure 13: LMI DDR mode timings

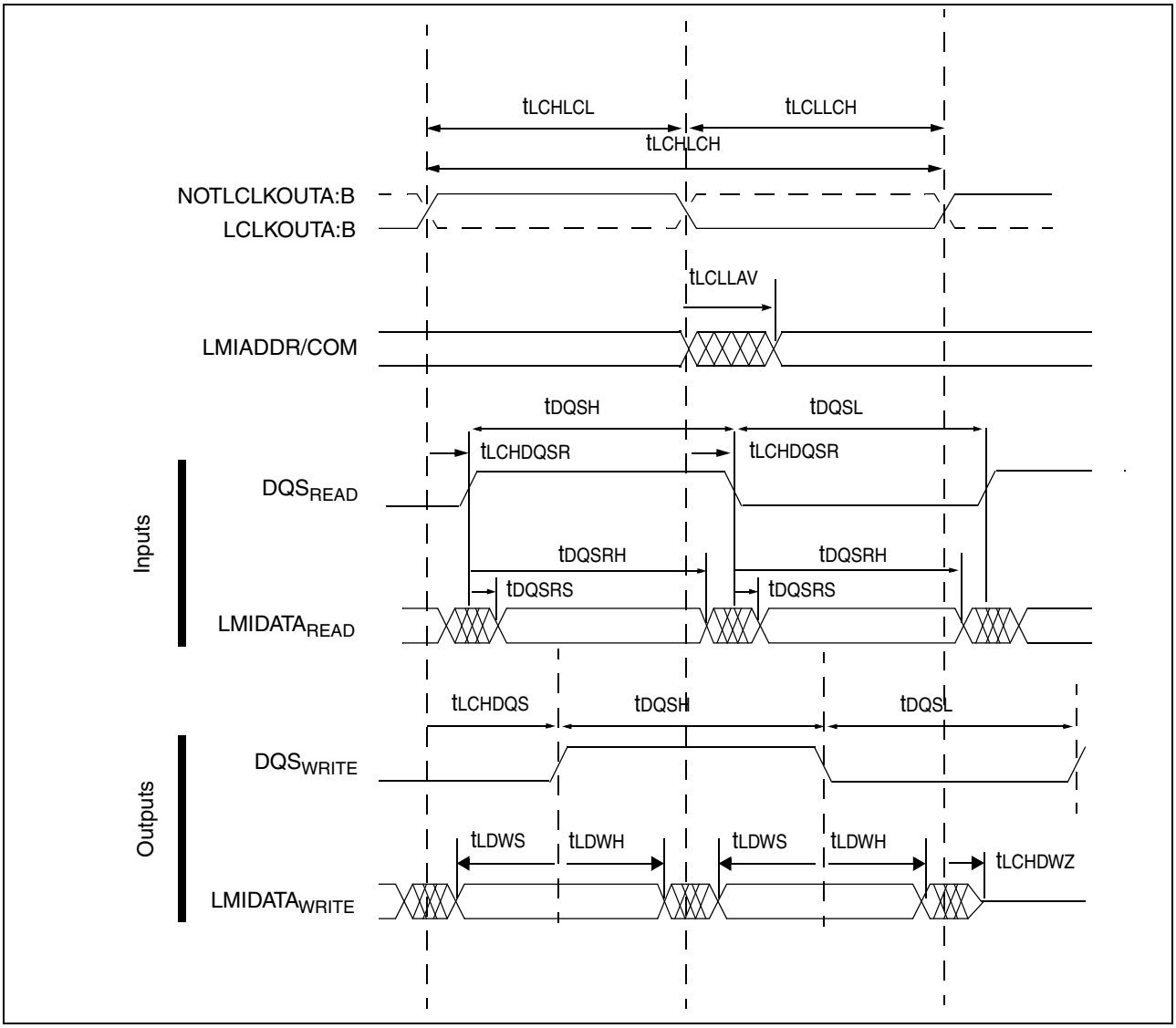


Table 30: LMI DDR-SDRAM AC timings

Symbol	Parameter	Min	Max	Units	Note
tLCHLCH	LMI clock period	10		ns	
tLCHLCL	LMI clock high time	0.45		tLCHLCH	

Table 30: LMI DDR-SDRAM AC timings

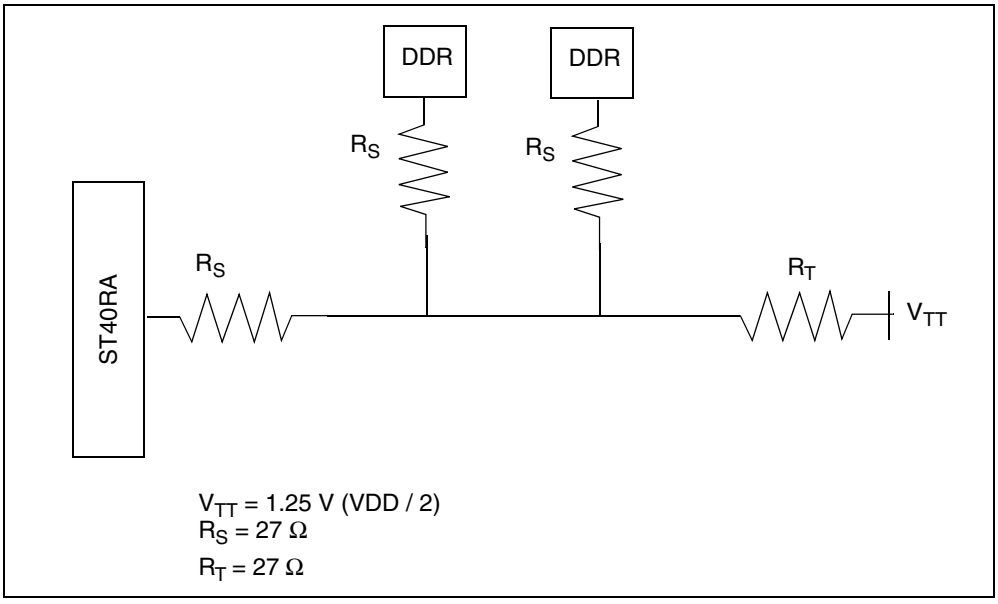
Symbol	Parameter	Min	Max	Units	Note
t_{LCLLCH}	LMI clock low period	0.45		t_{LCHLCH}	
t_{LCHLAV}	LCLKOUT low to address and command valid	-1.5	1.5	ns	
$t_{LCHDQSR}$	LCLKOUT high to read DQS edge	-1.5	1.5	ns	a
t_{DQSH}	DQS high	0.45		t_{LCHLCH}	
t_{DQSL}	DQS low	0.45		t_{LCHLCH}	
t_{DQSRS}	Read data setup for DQS edge	1 - t_{LCHLCH} / 4		ns	a
t_{DQSRH}	Read data hold for DQS edge	t_{LCHLCH} / 4 + 1		ns	a
t_{LCHDQS}	LCLKOUT high to write DQS	N * t_{LCHLCH} / 4 - 0.75	N * t_{LCHLCH} / 4 + 0.75	ns	
t_{LDWS}	Write data setup to DQS edge	N * t_{LCHLCH} / 4 - 0.75		ns	
t_{LDWH}	DQS edge to Write data invalid	N * t_{LCHLCH} / 4 + 0.75		ns	
t_{LCHDWZ}	LCLKOUT high to write data Z		2	ns	

a. Constraint placed on external system

7.6 DDR bus termination (SSTL_2)

The JEDEC specification for SSTL_2 and an application note from a DDR SDRAM manufacturer (DDR SDRAM Signaling Design Notes (MIcron Technology)) recommend the following layout to reduce signal reflections on the bus:

Figure 14: SSTL_2 bus termination



7.7 General purpose peripheral bus (EMI) AC specifications

Figure 15: EMI AC timings

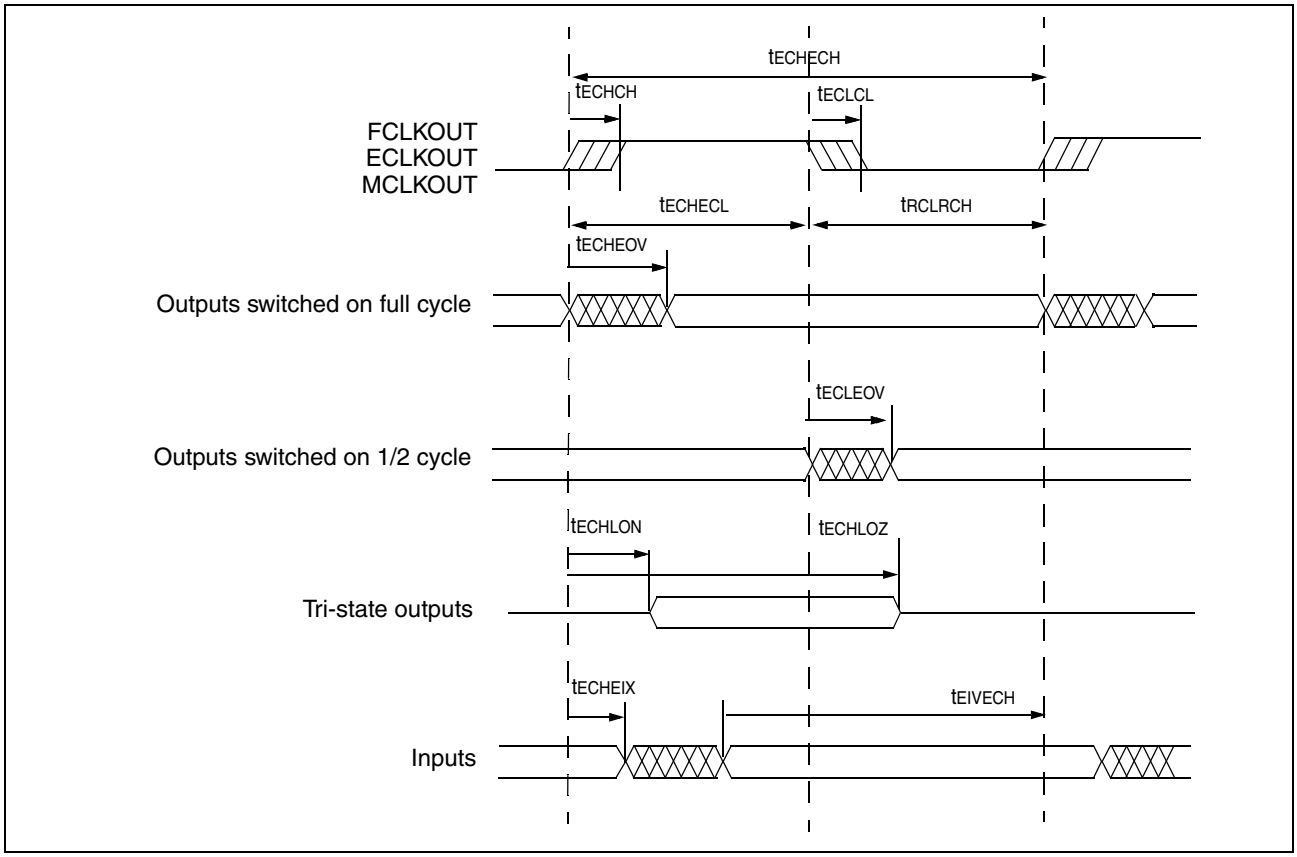


Table 31: EMI AC timings

Symbol	Parameter	Min	Max	Units	Note
tECHECH	EMI reference clock period	12		ns	a
tECHECL	EMI reference clock high time	4		ns	
tECLECH	EMI reference clock low period	4		ns	
tECHCH	EMI reference clock high to all clocks high	3	6	ns	
tECLCL	EMI reference clock low to all clocks low	3	6	ns	
tECHEOV	EMI reference clock high to output signals valid	0	2	ns	
tECLEOV	EMI reference clock low to output signals valid	0	2	ns	1
tECHEOZ	EMI reference clock high to outputs tri-state		4	ns	
tECHEON	EMI reference clock high to outputs on			ns	1
tEIVECH	Input signals valid to EMI reference clock high	4		ns	b
tECHEIX	Input signals hold after EMI reference clock high	2		ns	2

a. EMI reference clock is defined as the time when ECLKOUT, MCLKOUT and FCLKOUT are all valid.

b. Including EWAIT signal

7.8 PIO AC specifications

Reference clock in this case means the last transition of any PIO output signal within a bus, and hence is a virtual clock.

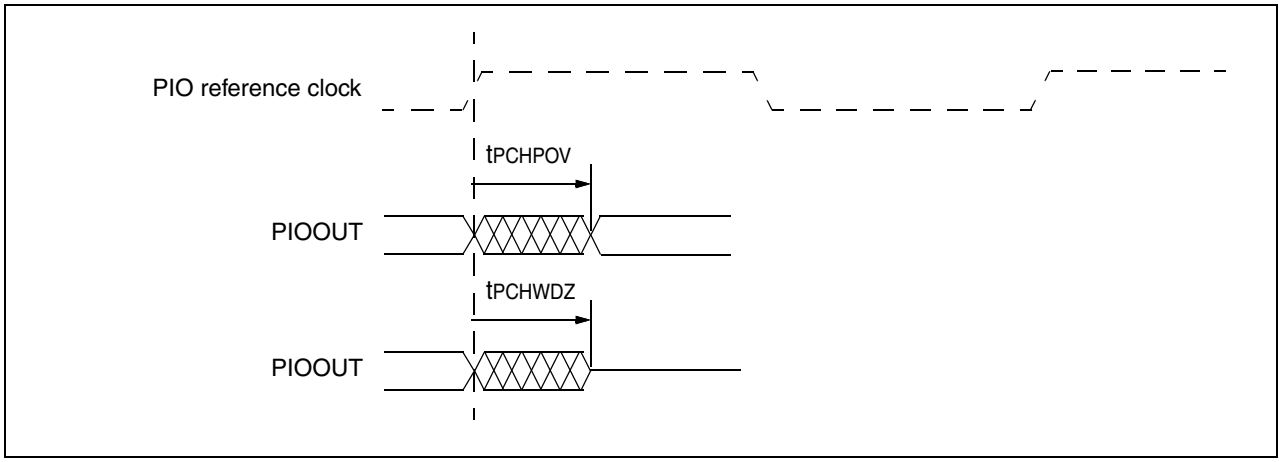
Table 32: PIO timings

Symbol	Parameter	PIO13:0		PIO23:14		Units	Note
		Min	Max	Min	Max		
t_{PCHPOV}	PIO reference clock high to PIO output valid	-5.5	1	-5.5	1	ns	a
t_{PCHWDZ}	PIO tri-state after PIO reference clock high	-5	5	-5	5	ns	1
t_{PIOr}	Output rise time	1	5	1	5	ns	
t_{PIOf}	Output fall time	1	5	1	5	ns	
t_{PIOr}	Input rise time		20		5	ns	b
t_{PIOf}	Input fall time		20		5	ns	2

a. No skew guarantee is made between the two separate PIO buses: PIO13:0 and PIO23:14

b. Loose input rise and fall times on PIO13:0 bus as these are schmitt trigger inputs.

Figure 16: PIO AC timings



7.9 System CLKIN AC specifications

The timings referenced in *Figure 17* refer to the case where CLKIN is directly clocked from an external source. In this case care should be taken that the total load on the CLOCKOSC output is <2pF.

Table 33: CLKIN timings

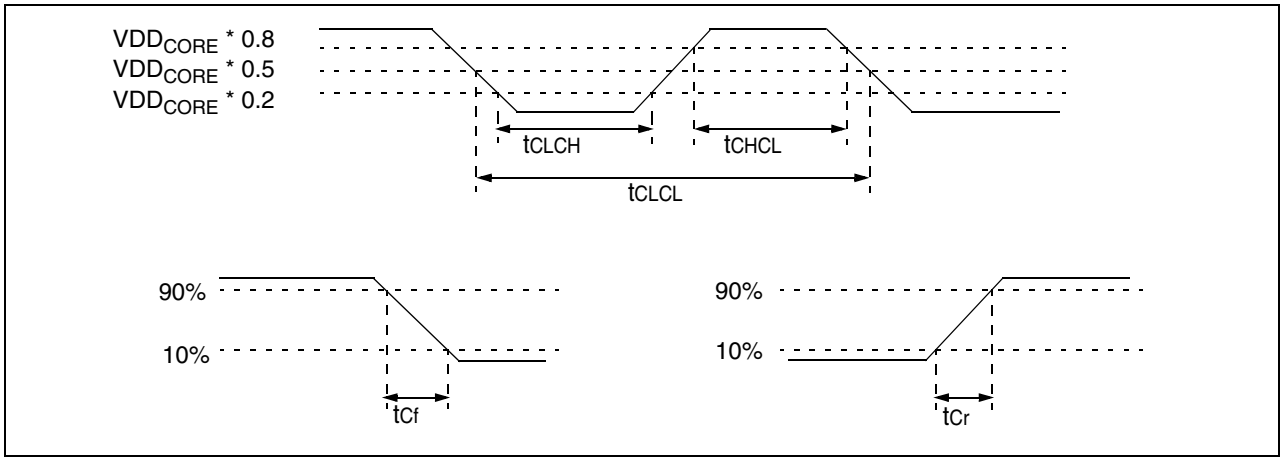
Symbol	Parameter	Min	Nom	Max	Units	Notes
tCLCH	CLKIN pulse width low	6			ns	
tCHCL	CLKIN pulse width high	6			ns	
tCLCL	CLKIN period		27		MHz	a
tCr	CLKIN rise time			10	ns	b, c
tCf	CLKIN fall time			10	ns	2, 3

a. Measured between corresponding points on consecutive falling edges.

b. When driven by an external clock.

c. Clock transitions must be monotonic within the range VIH to VIL.

Figure 17: CLKIN timings



7.10 Low power CLKIN AC specifications

The timings referenced in *Figure 18* refer to the case where CLKIN is directly clocked from an external source. In this case care should be taken that the total load on the LPCLKOSC output is <2pF.

Table 34: LPCLKIN timings

Symbol	Parameter	Min	Nom	Max	Units	Notes
tLCLLCL	LPCLKIN period		32.678		kHz	a, b
	LPCLKIN duty cycle	10	50	90	%	
tLCr	LPCLKIN rise time			10	ns	c, d
tLCf	LPCLKIN fall time			10	ns	3, 4

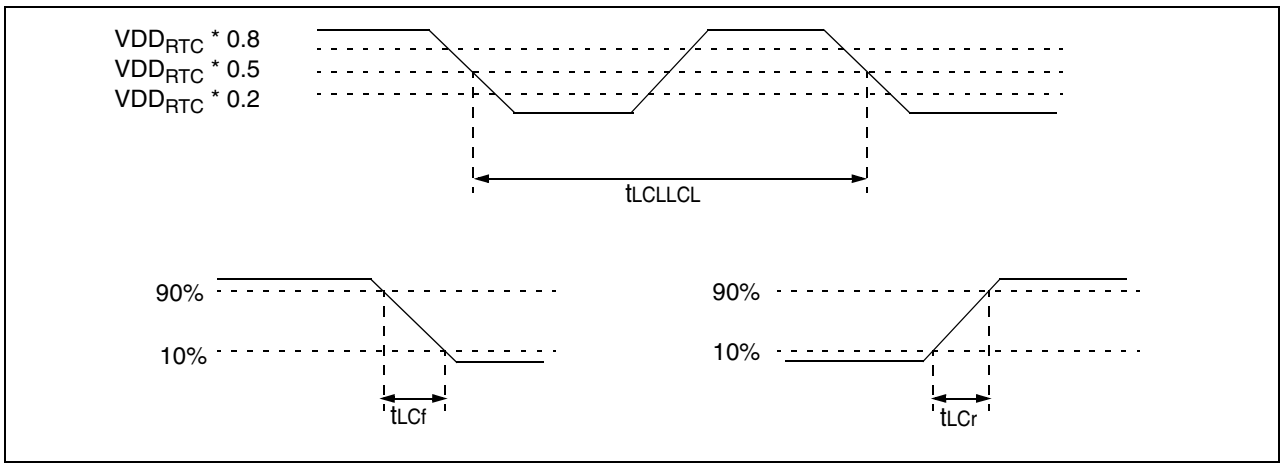
a. Measured between corresponding points on consecutive falling edges.

b. Variation of individual falling edges from their nominal times.

c. When driven by an external clock.

d. Transitions must be monotonic within the range VIH to VIL

Figure 18: CLKIN timings



7.11 UDI and IEEE 1149.1 TAP AC specifications

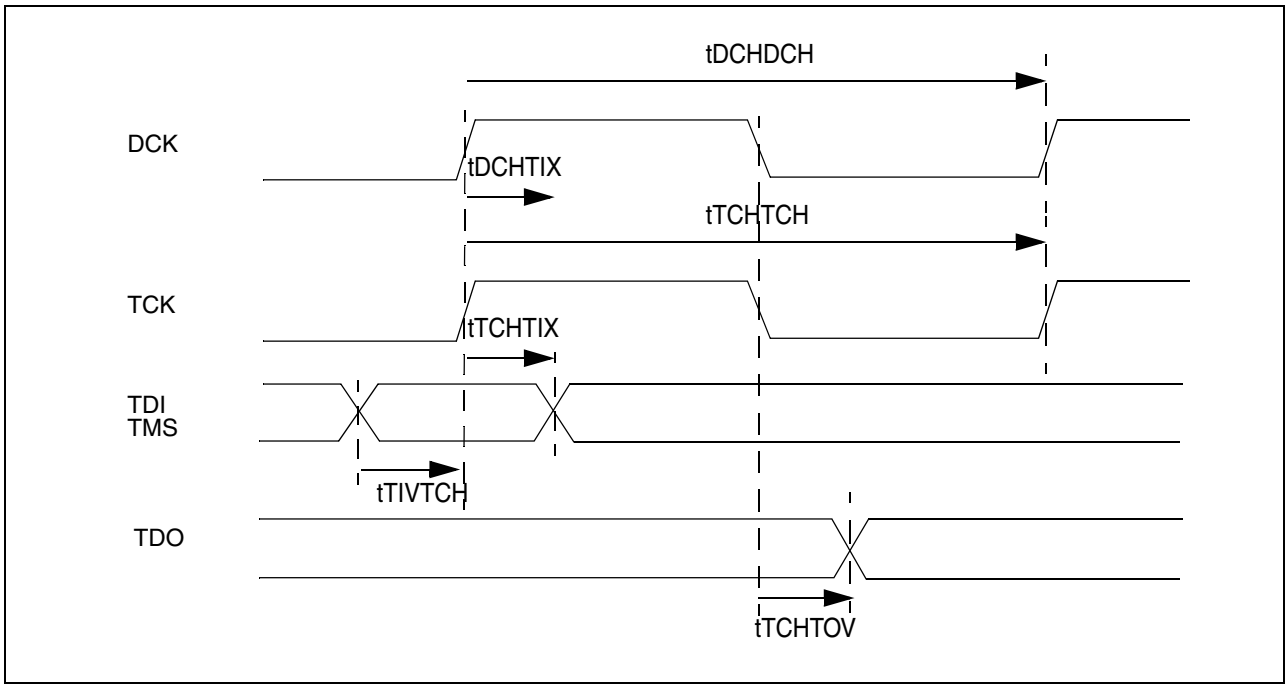
Table 35: TAP timings

Symbol	Parameter	Min	Nom	Max	Units	Notes
t_{TCHTCH}	TCK period	50			ns	a
t_{DCHDCH}	DCK period	50			ns	b
t_{TIVTCH}	TAP inputs setup to TCK/DCK high	5			ns	
t_{TCHTIX}	TAP input hold after TCK/DCK high	5			ns	
t_{TCHTOV}	TCK/DCK low to TAP output valid			10	ns	

a. During IEEE1149.1 drive board level manufacturing tests only TCK is active.

b. During application level diagnostics only DCLK is active.

Figure 19: UDI and IEEE TAP timings



8 Pin description

8.1 Function pin use selection

Full details of the functional pin sharing are found in *Section 8.3: PBGA 27 x 27 ballout on page 74*.

Table 36: ST40RA functional pin sharing summary

Functional pin group	Pins	Alternate use(s)	High-end interactive set-top box (with STi5514) example use
PCI request and grant	NOTPREQ[0:3] NOTPGNT[0:3] NOTPINTA	PIO[14:23]	PCI bus
PCI request and grant	NOTPREQ[2:3] NOTPGNT[2:3]	PIO[14:23] EMPIDREQ[0:1] EMPIDACK[0:1]	PCI bus
GPDMA handshake	DACK[0:1] DREQ[0:1] DRAQ[0:1]	PIO[8:13] EMPIDREQ[2:3] EMPIDACK[2:3] EMPIDRACK[2:3]	GPDMA
2 x SCIF	SCI2, CTS1 RXD0, RXD1 SCK0, SCK1 TXD0, TXD1	PIO[0:7]	2 x SCIF

8.2 Mode selection

During the power-on reset cycle a range of basic system configurations can be set up with resistive pull-ups or pull-downs. A detailed description of these selections is found in the relevant chapters of the *ST40 System Architecture Manual*.

See *Section 8.3: PBGA 27 x 27 ballout on page 74* for information on which pins these mode inputs have been placed on the ST40RA.

Table 37: Mode selection pins for ST40RA

Mode pin	Pin name	Architecture signal name	Block affected	Description	Notes
MODE2:0	EADDR2 EADDR3 EADDR4	MD2:0	CLOCKGEN	Set system clock operating mode	a
MODE4:3	EADDR5 EADDR6	MD4:3	CLOCKGEN	Set PCI clock operating mode	1
MODE5	EADDR7	MD5	CLOCKGEN	Set clock input source H: Crystal, L: External	
MODE6	EADDR8	MD6	CLOCKGEN	Set enable CKIO	

Table 37: Mode selection pins for ST40RA

Mode pin	Pin name	Architecture signal name	Block affected	Description	Notes
MODE7	EADDR9	MD7	EMISS	Enable MPX arbiter	
MODE8	EADDR10	MD8	System	Set endianness H: Little L: Big	
MODE9	EADDR11	MD9	EMI	Set EMI port H: Master L: Slave	b
MODE11:10	EADDR12 EADDR13	MD11:10	EMI	Set booting ROM bus size 00: Reserved 01: 32-bit 10: 16-bit 11: 8-bit	
MODE12	EADDR14	MD12	EMI	Enable NOP when accessing flash	c
MODE13	EADDR15	MD13	Reserved	Tie high	d
MODE14	EADDR16	MD14	PCI	PCI bridge mode H: Host L: Satellite	
MODE15	EADDR17	MD15	PCI	Reserved: PCI select clock H: External L: Internal	e
MODE16	EADDR18	MD16	-	Reserved: Tie high	f
MODE17	EADDR19	MD17	-		
MODE18	EADDR20	MD18	-		
MODE19	EADDR21	MD19	-		

a. See CLOCKGEN chapter of the *ST40 System Architecture Manual* for details.

b. ST40RA is always the clock master, providing EMI clocks to the system.

c. See EMI chapter of the *ST40 System Architecture Manual* for details.

d. reserved for enable retiming stage on EMI padlogic

e. PCI clock is selected externally on the board for ST40RA. The mode pin may be used for clock selection in future variants.

f. These mode pins are not used in current variants, however, they may be used to enable additional functionality in future variants

8.3 PBGA 27 x 27 ballout

This should be used in conjunction with *Figure 21: Package layout (viewed through package) on page 89*.

Pin name	Loc	Architecture signal name	Pin function		Pin	
			Default	Alternate	Type	Dir
LDATA0	A17	MD0	Memory data		SL	I/O
LDATA1	B17	MD1	Memory data		SL	I/O
LDATA2	A18	MD2	Memory data		SL	I/O
LDATA3	B18	MD3	Memory data		SL	I/O
LDATA4	A19	MD4	Memory data		SL	I/O
LDATA5	B19	MD5	Memory data		SL	I/O
LDATA6	A20	MD6	Memory data		SL	I/O
LDATA7	B20	MD7	Memory data		SL	I/O
LDATA8	A13	MD8	Memory data		SL	I/O
LDATA9	B13	MD9	Memory data		SL	I/O
LDATA10	A14	MD10	Memory data		SL	I/O
LDATA11	B14	MD11	Memory data		SL	I/O
LDATA12	A15	MD12	Memory data		SL	I/O
LDATA13	B15	MD13	Memory data		SL	I/O
LDATA14	A16	MD14	Memory data		SL	I/O
LDATA15	B16	MD15	Memory data		SL	I/O
LDATA16	A7	MD16	Memory data		SL	I/O
LDATA17	B7	MD17	Memory data		SL	I/O
LDATA18	A8	MD18	Memory data		SL	I/O
LDATA19	B8	MD19	Memory data		SL	I/O
LDATA20	A9	MD20	Memory data		SL	I/O
LDATA21	B9	MD21	Memory data		SL	I/O
LDATA22	A10	MD22	Memory data		SL	I/O
LDATA23	B10	MD23	Memory data		SL	I/O
LDATA24	A3	MD24	Memory data		SL	I/O
LDATA25	B3	MD25	Memory data		SL	I/O
LDATA26	A4	MD26	Memory data		SL	I/O
LDATA27	B4	MD27	Memory data		SL	I/O
LDATA28	A5	MD28	Memory data		SL	I/O
LDATA29	B5	MD29	Memory data		SL	I/O
LDATA30	A6	MD30	Memory data		SL	I/O

Table 38: PBGA ballout for ST40RA

Pin name	Loc	Architecture signal name	Pin function		Pin	
			Default	Alternate	Type	Dir
LDATA31	B6	MD31	Memory data		SL	I/O
LDATA32	F1	MD32	Memory data		SL	I/O
LDATA33	F2	MD33	Memory data		SL	I/O
LDATA34	E1	MD34	Memory data		SL	I/O
LDATA35	E2	MD35	Memory data		SL	I/O
LDATA36	D1	MD36	Memory data		SL	I/O
LDATA37	D2	MD37	Memory data		SL	I/O
LDATA38	C1	MD38	Memory data		SL	I/O
LDATA39	C2	MD39	Memory data		SL	I/O
LDATA40	K1	MD40	Memory data		SL	I/O
LDATA41	K2	MD41	Memory data		SL	I/O
LDATA42	J1	MD42	Memory data		SL	I/O
LDATA43	J2	MD43	Memory data		SL	I/O
LDATA44	H1	MD44	Memory data		SL	I/O
LDATA45	H2	MD45	Memory data		SL	I/O
LDATA46	G1	MD46	Memory data		SL	I/O
LDATA47	G2	MD47	Memory data		SL	I/O
LDATA48	T1	MD48	Memory data		SL	I/O
LDATA49	T2	MD49	Memory data		SL	I/O
LDATA50	R1	MD50	Memory data		SL	I/O
LDATA51	R2	MD51	Memory data		SL	I/O
LDATA52	P1	MD52	Memory data		SL	I/O
LDATA53	P2	MD53	Memory data		SL	I/O
LDATA54	N1	MD54	Memory data		SL	I/O
LDATA55	N2	MD55	Memory data		SL	I/O
LDATA56	Y1	MD56	Memory data		SL	I/O
LDATA57	Y2	MD57	Memory data		SL	I/O
LDATA58	W1	MD58	Memory data		SL	I/O
LDATA59	W2	MD59	Memory data		SL	I/O
LDATA60	V1	MD60	Memory data		SL	I/O
LDATA61	V2	MD61	Memory data		SL	I/O
LDATA62	U1	MD62	Memory data		SL	I/O
LDATA63	U2	MD63	Memory data		SL	I/O
LBANK0	J3	BA0	Mem bank address		SL	O

Table 38: PBGA ballout for ST40RA

Pin name	Loc	Architecture signal name	Pin function		Pin	
			Default	Alternate	Type	Dir
LBANK1	J4	BA1	Mem bank address		SL	O
LADDR0	G3	MA0	Memory page/column address		SL	O
LADDR1	G4	MA1	Memory page/column address		SL	O
LADDR2	G5	MA2	Memory page/column address		SL	O
LADDR3	F3	MA3	Memory page/column address		SL	O
LADDR4	F4	MA4	Memory page/column address		SL	O
LADDR5	F5	MA5	Memory page/column address		SL	O
LADDR6	E3	MA6	Memory page/column address		SL	O
LADDR7	E4	MA7	Memory page/column address		SL	O
LADDR8	E5	MA8	Memory page/column address		SL	O
LADDR9	D3	MA9	Memory page/column address		SL	O
LADDR10	D4	MA10	Memory page/column address		SL	O
LADDR11	D5	MA11	Memory page/column address		SL	O
LADDR12	C3	MA12	Memory page/column address		SL	O
LADDR13	C4	MA13	Memory page/column address		SL	O
LADDR14	C5	MA14	Memory page/column address		SL	O
LDQS0	C19	DQS0	DDR data strobe		SL	O
LDQS1	B12	DQS1	DDR data strobe		SL	O
LDQS2	A11	DQS2	DDR data strobe		SL	O
LDQS3	B2	DQS3	DDR data strobe		SL	O
LDQS4	B1	DQS4	DDR data strobe		SL	O
LDQS5	L2	DQS5	DDR data strobe		SL	O
LDQS6	M1	DQS6	DDR data strobe		SL	O
LDQS7	W3	DQS7	DDR data strobe		SL	O
LCLKOUTA	D8	MCLKOA	SDRAM clock output		SL	O
NOTLCLKOUTA	D7	NOTMCLKOA	SDRAM clock output		SL	O
LCLKOUTB	L3	MCLKOB	SDRAM clock output		SL	O
NOTLCLKOUTB	M3	NOTMCLKOB	SDRAM clock output		SL	O
LVREF	H5	VREF	DDR reference voltage		-	I
LDQM0	C20	DQM0	SDRAM data mask		SL	O
LDQM1	A12	DQM1	SDRAM data mask		SL	O
LDQM2	B11	DQM2	SDRAM data mask		SL	O
LDQM3	A2	DQM3	SDRAM data mask		SL	O
LDQM4	A1	DQM4	SDRAM data mask		SL	O

Table 38: PBGA ballout for ST40RA

Pin name	Loc	Architecture signal name	Pin function		Pin	
			Default	Alternate	Type	Dir
LDQM5	L1	DQM5	SDRAM data mask		SL	O
LDQM6	M2	DQM6	SDRAM data mask		SL	O
LDQM7	Y3	DQM7	SDRAM data mask		SL	O
NOTLCSA0	C9	NOTCSA0	Chip select A		SL	O
NOTLCSA1	D9	NOTCSA1	Chip select A		SL	O
NOTLCSB0	H3	NOTCSB0	Chip select B		SL	O
NOTLCSB1	H4	NOTCSB1	Chip select B		SL	O
NOTLRASA	C8	NOTRASA	Row add strobe A		SL	O
NOTLRASB	K4	NOTRASB	Row add strobe B		SL	O
NOTLCASA	C7	NOTCASA	Column add strobe A		SL	O
NOTLCASB	L4	NOTCASB	Column add strobe B		SL	O
NOTLWEA	D6	NOTWEA	Write enable A		SL	O
NOTLWEB	J5	NOTWEB	Write enable B		SL	O
LCLKEN0	C6	CKE0	Clock enable		SL	O
LCLKEN1	K3	CKE1	Clock enable		SL	O
PAD0	T17	PCI_AD0	PCI address and data		P8	I/O
PAD1	T18	PCI_AD1	PCI address and data		P8	I/O
PAD2	R19	PCI_AD2	PCI address and data		P8	I/O
PAD3	R20	PCI_AD3	PCI address and data		P8	I/O
PAD4	R17	PCI_AD4	PCI address and data		P8	I/O
PAD5	R18	PCI_AD5	PCI address and data		P8	I/O
PAD6	P19	PCI_AD6	PCI address and data		P8	I/O
PAD7	P20	PCI_AD7	PCI address and data		P8	I/O
PAD8	P17	PCI_AD8	PCI address and data		P8	I/O
PAD9	P18	PCI_AD9	PCI address and data		P8	I/O
PAD10	N19	PCI_AD10	PCI address and data		P8	I/O
PAD11	N20	PCI_AD11	PCI address and data		P8	I/O
PAD12	N17	PCI_AD12	PCI address and data		P8	I/O
PAD13	N18	PCI_AD13	PCI address and data		P8	I/O
PAD14	M19	PCI_AD14	PCI address and data		P8	I/O
PAD15	M20	PCI_AD15	PCI address and data		P8	I/O
PAD16	K17	PCI_AD16	PCI address and data		P8	I/O
PAD17	K18	PCI_AD17	PCI address and data		P8	I/O
PAD18	J19	PCI_AD18	PCI address and data		P8	I/O

Table 38: PBGA ballout for ST40RA

Pin name	Loc	Architecture signal name	Pin function		Pin		
			Default	Alternate	Type	Dir	
PAD19	J20	PCI_AD19	PCI address and data		P8	I/O	
PAD20	J17	PCI_AD20	PCI address and data		P8	I/O	
PAD21	J18	PCI_AD21	PCI address and data		P8	I/O	
PAD22	H19	PCI_AD22	PCI address and data		P8	I/O	
PAD23	H20	PCI_AD23	PCI address and data		P8	I/O	
PAD24	H17	PCI_AD24	PCI address and data		P8	I/O	
PAD25	H18	PCI_AD25	PCI address and data		P8	I/O	
PAD26	G19	PCI_AD26	PCI address and data		P8	I/O	
PAD27	G20	PCI_AD27	PCI address and data		P8	I/O	
PAD28	G17	PCI_AD28	PCI address and data		P8	I/O	
PAD29	G18	PCI_AD29	PCI address and data		P8	I/O	
PAD30	F17	PCI_AD30	PCI address and data		P8	I/O	
PAD31	F18	PCI_AD31	PCI address and data		P8	I/O	
NOTPCBE0	P16	PCI_C/BE0	PCI com and byte enable		P8	I/O	
NOTPCBE1	N16	PCI_C/BE1	PCI com and byte enable		P8	I/O	
NOTPCBE2	K16	PCI_C/BE2	PCI com and byte enable		P8	I/O	
NOTPCBE3	H16	PCI_C/BE3	PCI com and byte enable		P8	I/O	
PPAR	M16	PCI_PAR	Parity signal		P8	I/O	
NOTPFRAME	K19	NOTPCI_FRAME	PCI beginning access		P8	I/O	
NOTPIRDY	K20	NOTPCI_IRDY	PCI initiator ready		P8	I/O	
NOTPTRDY	L17	NOTPCI_TRDY	PCI target ready		P8	I/O	
NOTPSTOP	L19	NOTPCI_STOP	PCI req stop transfer		P8	I/O	
NOTPERR	M17	NOTPCI_PERR	PCI parity error		P8	I/O	
NOTPSERR	M18	NOTPCI_SERR	PCI system error		P8	I/O	
NOTPDEVSEL	L18	NOTPCI_DEVSEL	PCI device select		P8	I/O	
PIDSEL	J16	PCI_IDSEL	PCI initialization device		-	I/O	
NOTPRST	R16	NOTPCI_RST	PCI reset		P8	I/O	
NOTPLOCK	L20	NOTPLOCK	PCI exclusive access		P8	I	
PCLK	F19	PCI_CLK	PCI clock input		P8	I	
NOTPREQ0	E18	NOTPCI_REQ0	PCI external request for bus	PIO16	P8	I/O	I/O
NOTPREQ1	E17	NOTPCI_REQ1	PCI external request for bus	PIO18	P8	I	I/O
NOTPREQ2	F16	NOTPCI_REQ2	PCI external request for bus	PIO20	P8	I	I/O

Table 38: PBGA ballout for ST40RA

Pin name	Loc	Architecture signal name	Pin function		Pin		
			Default	Alternate	Type	Dir	
NOTPREQ3	G16	NOTPCI_REQ3	PCI external request for bus	PIO22 EMPIDREQ1	P8	I	I/O O
NOTPGNT0	D18	NOTPCI_GNT0	PCI grant external request	PIO17	P8	I/O	I/O
NOTPGNT1	D17	NOTPCI_GNT1	PCI grant external request	PIO19	P8	O	I/O
NOTPGNT2	E16	NOTPCI_GNT2	PCI grant external request	PIO21	P8	O	I/O
NOTPGNT3	D16	NOTPCI_GNT3	PCI grant external request	PIO23 EMPIDRAK1	P8	O	I/O I
PCLKOUT	F20	PCI_CLOCKOUT	PCI clock output	PIO14	P8	O	I/O
NOTPINTA	T19	NOTPCI_INTA	PCI interrupt request	PIO15	P8	I/O	I/O
DACK0	U19	DACK0	DMA bus acknowledge	PIO10 EMPIDACK2	C2A	O	I/O I
DRAK0	U18	DRACK0	DMA request acknowledge	PIO9 EMPIDRAK2	C2A	O	I/O I
DREQ0	V20	DREQ0	DMA transfer request	PIO8 EMPIDREQ2	C2A	I	I/O O
DACK1	U20	DACK1	DMA bus acknowledge	PIO13 EMPIDACK3	C2A	O	I/O I
DRAK1	T20	DRACK1	DMA request acknowledge	PIO12 EMPIDRAK3	C2A	O	I/O I
DREQ1	U17	DREQ1	DMA transfer request	PIO11 EMPIDREQ3	C2A	I	I/O O
SCI2	V19	RTS1/PIO7	SCI2 transmission request	PIO7	C2A	O	I/O
CTS1	V18	CTS1/PIO6	SCI2 transmission enabled	PIO6	C2A	O	I/O
RXD0	Y19	RXD0/PIO1	SCI receive data input	PIO1	C2A	I	I/O
RXD1	W20	RXD1/PIO4	SCI receive data input	PIO4	C2A	I	I/O
SCK0	Y18	SCK0/PIO0	SCI clock input	PIO0	C2A	I	I/O
SCK1	W18	SCK1/PIO3	SCI clock input	PIO3	C2A	I	I/O
TXD0	Y20	TXD0/PIO2	SCI transmit data output	PIO2	C2A	O	I/O
TXD1	W19	TXD1/PIO5	SCI transmit data output	PIO5	C2A	O	I/O
NOTRST	E14	NOTRESET	Power on reset		-	I	
IRL0	C10	IRL0	Interrupt request signal		-	I	
IRL1	C11	IRL1	Interrupt request signal		-	I	
IRL2	C12	IRL2	Interrupt request signal		-	I	
IRL3	D13	IRL3	Interrupt request signal		-	I	
NMI	C13	NMI	Nonmaskable interrupt		-	I	

Table 38: PBGA ballout for ST40RA

Pin name	Loc	Architecture signal name	Pin function		Pin		
			Default	Alternate	Type	Dir	
TMUCLK	E15	TCLK	RTC output clock	TMU input clock	C2B	I/O	I/O
LPCLKIN	E12	EXTAL2	RTC crystal resonator input: on VDD_{RTC} supply		-	I	
LPCLKOSC	E13	XTAL2	RTC crystal resonator output: on VDD_{RTC} supply		-	O	
VDDRTC	E11	VCCRTC	Real-time clock supply			I	
CLKIN	E20	CLKIN	System clock input: on VDD_{CORE} supply		-	I	
CLKOSC	D20	CLKOSC	Crystal resonator pin: on VDD_{CORE} supply		-	O	
AUXCLKOUT	E19	CKIO	Reference 27 MHz clock output		-	O	
STATUS0	C14	STATUS0	Processor operating status		-	O	
STATUS1	D14	STATUS1	Processor operating status		-	O	
AUDATA0	C18	AUDATA0	AUD bus command and data		-	O	
AUDATA1	C17	AUDATA1	AUD bus command and data		-	O	
AUDATA2	C16	AUDATA2	AUD bus command and data		-	O	
AUDATA3	C15	AUDATA3	AUD bus command and data		-	O	
AUDSYNC	D15	AUDSYNC	AUD command valid		-	O	
AUDCLK	D19	AUDCK	AUD clock output		-	O	
NOTASEBRK	E9	NOTASEBRK/ BRKACK	Dedicated emulator pin		C4	I/O	
DCLK	D11	DCK	Clock for udi		-	I	
TCK	D12	TCK	Test clock		-	I	
TMS	D10	TMS	Test mode		-	I	
NOTTRST	E7	TRST	Test reset		-	I	
TDI	E6	TDI	Test data input		-	I	
TDO	E8	TDO	Test data output		-	O	
EADDR2	V4	MA2	EMI external address	MODE0	E4	O	I
EADDR3	U4	MA3	EMI external address	MODE1	E4	O	I
EADDR4	V5	MA4	EMI external address	MODE2	E4	O	I
EADDR5	U5	MA5	EMI external address	MODE3	E4	O	I
EADDR6	U6	MA6	EMI external address	MODE4	E4	O	I
EADDR7	T6	MA7	EMI external address	MODE5	E4	O	I
EADDR8	U7	MA8	EMI external address	MODE6	E4	O	I
EADDR9	T7	MA9	EMI external address	MODE7	E4	O	I
EADDR10	U8	MA10	EMI external address	MODE8	E4	O	I
EADDR11	T8	MA11	EMI external address	MODE9	E4	O	I

Table 38: PBGA ballout for ST40RA

Pin name	Loc	Architecture signal name	Pin function		Pin		
			Default	Alternate	Type	Dir	
EADDR12	U9	MA12	EMI external address	MODE10	E4	O	I
EADDR13	T9	MA13	EMI external address	MODE11	E4	O	I
EADDR14	V11	MA14	EMI external address	MODE12	E4	O	I
EADDR15	U11	MA15	EMI external address	MODE13	E4	O	I
EADDR16	V12	MA16	EMI external address	MODE14	E4	O	I
EADDR17	U12	MA17	EMI external address	MODE15	E4	O	I
EADDR18	U13	MA18	EMI external address	MODE16	E4	O	I
EADDR19	U14	MA19	EMI external address	MODE17	E4	O	I
EADDR20	V15	MA20	EMI external address	MODE18	E4	O	I
EADDR21	U15	MA21	EMI external address	MODE19	E4	O	I
EADDR22	T15	MA22	EMI external address		E4	O	
EADDR23	V16	MA23	EMI external address		E4	O	
EADDR24	U16	MA24	EMI external address		E4	O	
EADDR25	T16	MA25	EMI external address	EMPIDACK1	E4	O	I
EADDR26	V17	MA26	EMI external address	EMPIDACK0	E4	O	I
EDATA0	W4	MD0	External data / MPX address		E4	I/O	
EDATA1	Y4	MD1	External data/MPX address		E4	I/O	
EDATA2	W5	MD2	External data/MPX address		E4	I/O	
EDATA3	Y5	MD3	External data/MPX address		E4	I/O	
EDATA4	V6	MD4	External data/MPX address		E4	I/O	
EDATA5	W6	MD5	External data/MPX address		E4	I/O	
EDATA6	Y6	MD6	External data/MPX address		E4	I/O	
EDATA7	V7	MD7	External data/MPX address		E4	I/O	
EDATA8	W7	MD8	External data/MPX address		E4	I/O	
EDATA9	Y7	MD9	External data/MPX address		E4	I/O	
EDATA10	V8	MD10	External data/MPX address		E4	I/O	
EDATA11	W8	MD11	External data/MPX address		E4	I/O	
EDATA12	Y8	MD12	External data/MPX address		E4	I/O	
EDATA13	V9	MD13	External data/MPX address		E4	I/O	
EDATA14	Y9	MD14	External data/MPX address		E4	I/O	
EDATA15	W9	MD15	External data/MPX address		E4	I/O	
EDATA16	W11	MD16	External data/MPX address		E4	I/O	
EDATA17	Y11	MD17	External data/MPX address		E4	I/O	
EDATA18	W12	MD18	External data/MPX address		E4	I/O	

Table 38: PBGA ballout for ST40RA

Pin name	Loc	Architecture signal name	Pin function		Pin		
			Default	Alternate	Type	Dir	
EDATA19	Y12	MD19	External data/MPX address		E4	I/O	
EDATA20	V13	MD20	External data/MPX address		E4	I/O	
EDATA21	W13	MD21	External data/MPX address		E4	I/O	
EDATA22	Y13	MD22	External data/MPX address		E4	I/O	
EDATA23	V14	MD23	External data/MPX address		E4	I/O	
EDATA24	W14	MD24	External data/MPX address		E4	I/O	
EDATA25	Y14	MD25	External data/MPX address		E4	I/O	
EDATA26	W15	MD26	External data/MPX address		E4	I/O	
EDATA27	Y15	MD27	External data/MPX address		E4	I/O	
EDATA28	W16	MD28	External data/MPX address		E4	I/O	
EDATA29	Y16	MD29	External data/MPX address		E4	I/O	
EDATA30	W17	MD30	External data/MPX address		E4	I/O	
EDATA31	Y17	MD31	External data/MPX address		E4	I/O	
ECLKOUT	W10	ECLKOUT	External clock for SDRAM		-	O	
ECLKEN	U10	ECLKEN	External clock enable		-	O	
EDQM0	N4	EBE_DQM0	External byte enables		-	I/O	
EDQM1	P4	EBE_DQM1	External byte enables		-	I/O	
EDQM2	P5	EBE_DQM2	External byte enables		-	I/O	
EDQM3	R5	EBE_DQM3	External byte enables		-	I/O	
NOTECS0	R4	NOTECS5	External chip select	One NOTECS[0:5] used for NOTEMPICS Selected via software	E4	O	I
NOTECS1	T4	NOTECS4	External chip select		E4	O	
NOTECS2	T5	NOTECS3	External chip select		E4	O	
NOTECS3	T12	NOTECS2	External chip select		E4	O	
NOTECS4	T13	NOTECS1	External chip select		E4	O	
NOTECS5	T14	NOTECS0	External chip select		E4	O	
NOTERAS	U3	NOTERAS	External raw add strobe	MSTART and FLBADDR	E4	O	I/O
NOTECAS	T3	NOTECAS	External column address strobe, MFRAME (MPX_FRAME) and EOE_N (EMI output enable signal)		E4	O	I/O
EWAIT	T10	EWAIT	External wait command (**notready**)		E4	I/O	
NOTEWE	V3	NOTEWR	External read not write		E4	I/O	
EPENDING	N3	EPENDING	EMI pending refresh or access		E4	O	
MCLKOUT	Y10	MCLKOUT	MPX clock		-	O	

Table 38: PBGA ballout for ST40RA

Pin name	Loc	Architecture signal name	Pin function		Pin	
			Default	Alternate	Type	Dir
NOTMREQ	R3	EMI_BUS_REQ or EMI_HOLD_ACK when EMI slave	MPX bus request		-	I/O
NOTMACK	P3	EMI_BUS_GRANT or EMI_HOLD_REQ when EMI slave	MPX bus acknowledge		-	I/O
FCLKOUT	V10	FCLKOUT	Flash clock		-	O
NOTFBAA	N5	-	Flash bus address advance		-	O
NOTESCS0	L5	-	Reserved tri-state	MBXINT	P8	O
NOTESCS1	M5	-	Reserved tri-state	EMPIDREQ0	P8	O
NOTESCS2	M4	-	Reserved tri-state	EMPIDRAK0	P8	I
GND	H8:N13		36 ball array for ground supply and heat dissipation			
VDDCORE	M6	VDDCORE				
VDDCORE	N6	VDDCORE				
VDDCORE	P6	VDDCORE				
VDDCORE	R6	VDDCORE				
VDDCORE	R7	VDDCORE				
VDDCORE	R8	VDDCORE				
VDDCORE	R9	VDDCORE				
VDDCORE	R10	VDDCORE				
VDDCORE	R11	VDDCORE				
VDDCORE	T11	VDDCORE				
VDDCORE	R12	VDDCORE				
VDDCORE	R13	VDDCORE				
VDDCORE	R14	VDDCORE				
VDDCORE	M15	VDDCORE				
VDDCORE	N15	VDDCORE				
VDDCORE	P15	VDDCORE				
VDDCORE	R15	VDDCORE				
VDDLMI	K5	VDDLMI				
VDDLMI	F6	VDDLMI				
VDDLMI	G6	VDDLMI				
VDDIO	H6	VDDIO				

Table 38: PBGA ballout for ST40RA

Pin name	Loc	Architecture signal name	Pin function		Pin		
			Default	Alternate	Type	Dir	
VDDLMI	J6	VDDLMI					
VDDIO	K6	VDDIO					
VDDLMI	L6	VDDLMI					
VDDIO	F7	VDDIO					
VDDLMI	F8	VDDLMI					
VDDIO	F9	VDDIO					
VDDIO	E10	VDDIO					
VDDLMI	F10	VDDLMI					
VDDIO	F11	VDDIO					
VDDIO	F12	VDDIO					
VDDIO	F13	VDDIO					
VDDLMI	F14	VDDLMI					
VDDLMI	F15	VDDLMI					
VDDIO	G15	VDDIO					
VDDIO	H15	VDDIO					
VDDIO	J15	VDDIO					
VDDIO	K15	VDDIO					
VDDIO	L15	VDDIO					
VDDIO	L16	VDDIO					

Table 38: PBGA ballout for ST40RA

8.4 Pin states

The following table shows the direction and state of the pins during and immediately after reset.

- **Z** indicates an output or I/O pin that has been tri-stated.
- **I** indicates an input or I/O pin in input modes (I/O buffer tri-stated).
- **1** indicates an output or I/O pin driving logical high.
- **0** indicates an output or I/O pin driving logical low.
- **X** indicates an output or I/O pin driving undefined data.
- **H** indicates a pin with weak internal pull-up enabled.
- **L** indicates a pin with weak internal pull-down enabled.

Table 39: Pin reset states for ST40RA

Pin names	Architecturally defined reset state		Implementation reset state during and after reset		
	Dir	During reset	Dir	During reset	Following reset
LMI system pins					
LDATA0:63	I/O	Z	I/O	Z	
LBANK0:1	O	X	I/O	11	
LADDR0:14	O	X	I/O	1...1	
LDQS0:7	I/O	Z	I/O	Z	
LCLKOUTA:B	O	1	I/O	X	
NOTLCLKOUTA:B	O	0	I/O	X	
LDQM0:7	O	X	I/O	X	
NOTLCSA/B0:1,	O	1	I/O	11	
NOTLRASA:B, NOTLCASA:B, NOTLWEA:B	O	1	I/O	1	
LCLKEN0:1	O	0	I/O	0	
PCI system pins					
PAD0:31	I/O	0	I/O	0	
NOTPCBE0:3	I/O	0	I/O	0	
PPAR	I/O	0	I/O	0	
NOTPFRAME	I/O	1	I/O	H	
NOT PIRDY	I/O	1	I/O	H	
NOTPTRDY	I/O	1	I/O	H	
NOTPSTOP	I/O	1	I/O	H	
NOTPERR	I/O	1	I/O	H	
NOTPSERR	I/O	1	I/O	H	
NOTPDEVSEL	I/O	1	I/O	H	
PIDSEL	I/O	0	I	0	
NOTPRST	I/O	0	I/O	0	

Table 39: Pin reset states for ST40RA

Pin names	Architecturally defined reset state		Implementation reset state during and after reset		
	Dir	During reset	Dir	During reset	Following reset
NOTPLOCK	I	-	I/O	H	
PCLK	I	-	I/O	Z	
NOTPREQ[0:3]	I	-	I/O	Z	
NOTPGNT[0:3]	O	1	I/O	1111	
PCLKOUT	O	Running	I/O	Running	
NOTPINTA	I/O	-	I/O	H	
GPDMA pins					
DACK0, DACK1	O	Z	I/O	0	
DRAK0, DRAK1	O	Z	I/O	0	
DREQ0, DREQ1	I	-	I/O	Z	
Serial communication interface with FIFO (SCIF) pins					
SCI2	I	-	I/O	H	
CTS1	O	Z	I/O	H	
RXD0, RXD1	I	-	I/O	H	
SCK0, SCK1	I	-	I/O	H	
TXD0, TXD1	O	Z	I/O	H	
Power, clocks and so on					
NOTRST	I	-	I	(0)	(1)
IRL0:3, NMI	I	-	I	H	
TMUCLK	I/O	-	I/O	H	
LPCLKIN	I	-	I	0	
CLKIN	I	-	I	Running	
LPCLKOSC, CLKOSC	O	Oscillator output	O	Running	
AUXCLKOUT	O	CLKIN	O	CLKIN	
STATUS1:0	O	11	O	11	00
AUDATA0:3	O	00	O	0000	
AUDSYNC	O	1	O	1	
AUDCLK	O	0	O	0	
NOTASEBRK	I	-	I/O	(1)	
DCLK, TCK, EADDR,TDI	I	-	I	(0)	
NOTTRST,	I	-	I	(0)	(1)
TDO	O	Z	O	Z	

Table 39: Pin reset states for ST40RA

Pin names	Architecturally defined reset state		Implementation reset state during and after reset		
	Dir	During reset	Dir	During reset	Following reset
EMI system pins					
EADDR[2:26][A]	O	Z	I/O	ZZZE740 (Mode 0)	0
EDATA[0:31]	I/O	Z	I/O	Z	
ECLKOUT, MCLKOUT, FCLKOUT	O	0	O	0	
ECLKEN	O	Z	O	Z	1
EDQM[0:3]	O	Z	O	Z	1111
NOTECS[0:5]	O	1	I/O	Z	111111
NOTERAS, NOTECAS, NOTEWE	I/O	1	I/O	Z	1
EWAIT	I/O	Z	I/O	Z	
EPENDING	O I	0 (MD7 = 0) Z (MD7 = 1)	I/O MD7 = 0	Z	0
NOTMREQ (EMI_HOLD_ACK when EMI slave)	I	-	I	Z	
NOTMACK (EMI_HOLD_REQ when EMI slave)	O	Z	O	Z	1
NOTFBAA	O	Z	O	Z	1
NOTESCS[0:2]	O	Z	I/O	Z	

a. The reset state of the EADDR bus is tri-state, the value given corresponds to a specific boot mode and shows the expected ties.

9 Package

Physical properties:

- 27 x 27 mm 372 plastic ball grid array (PBGA) (336 + 36 thermal ground balls),
- Typical power consumption <2 W,
- Substrate height: 0.56 mm,
- Total height: 2.33 mm,
- Cover + substrate: 1.73 mm.

Figure 20 and *Figure 21* are diagrams of the pin disposition on the package. .

Figure 20: 372-pin PBGA package

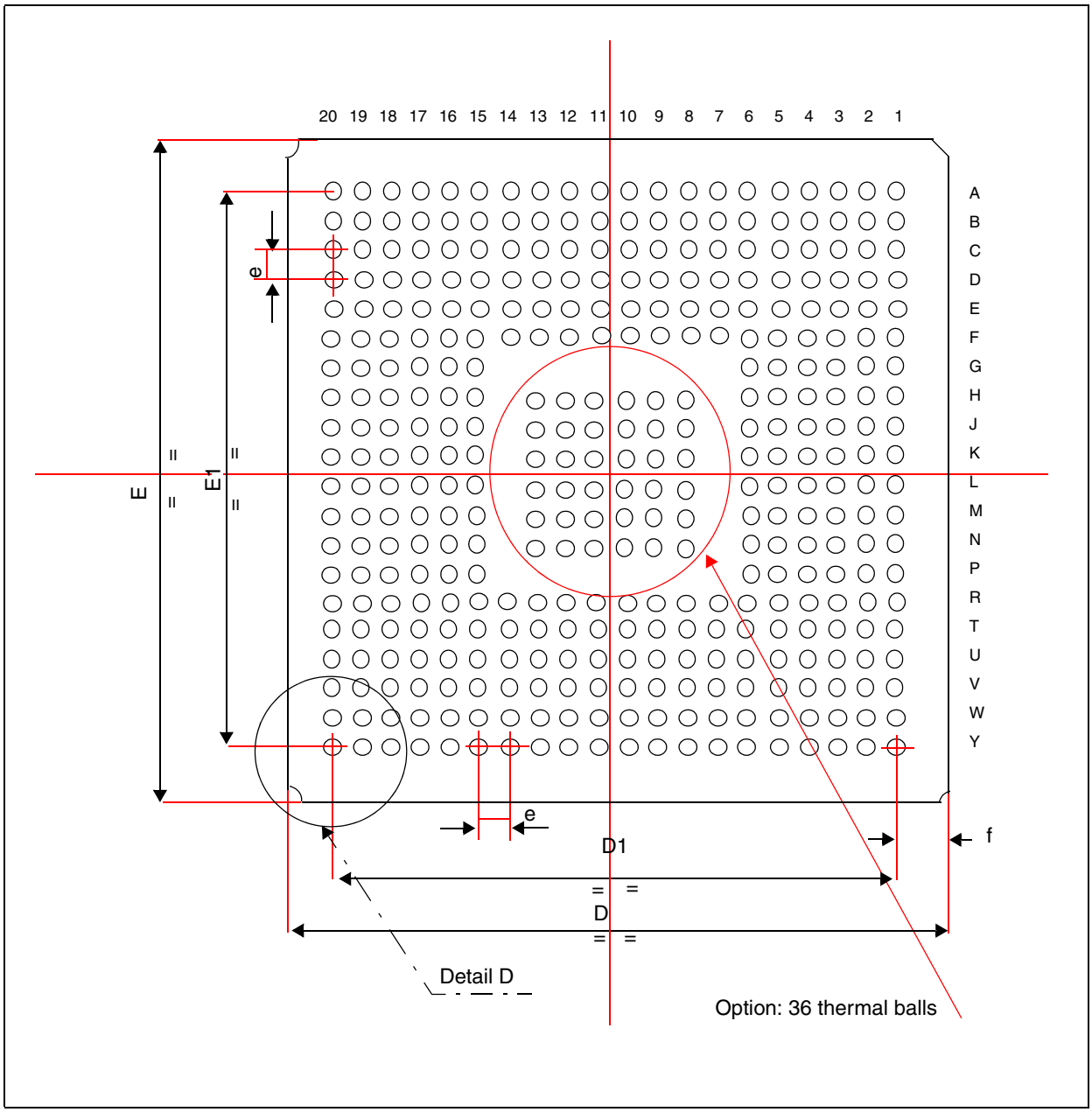


Figure 21: Package layout (viewed through package)

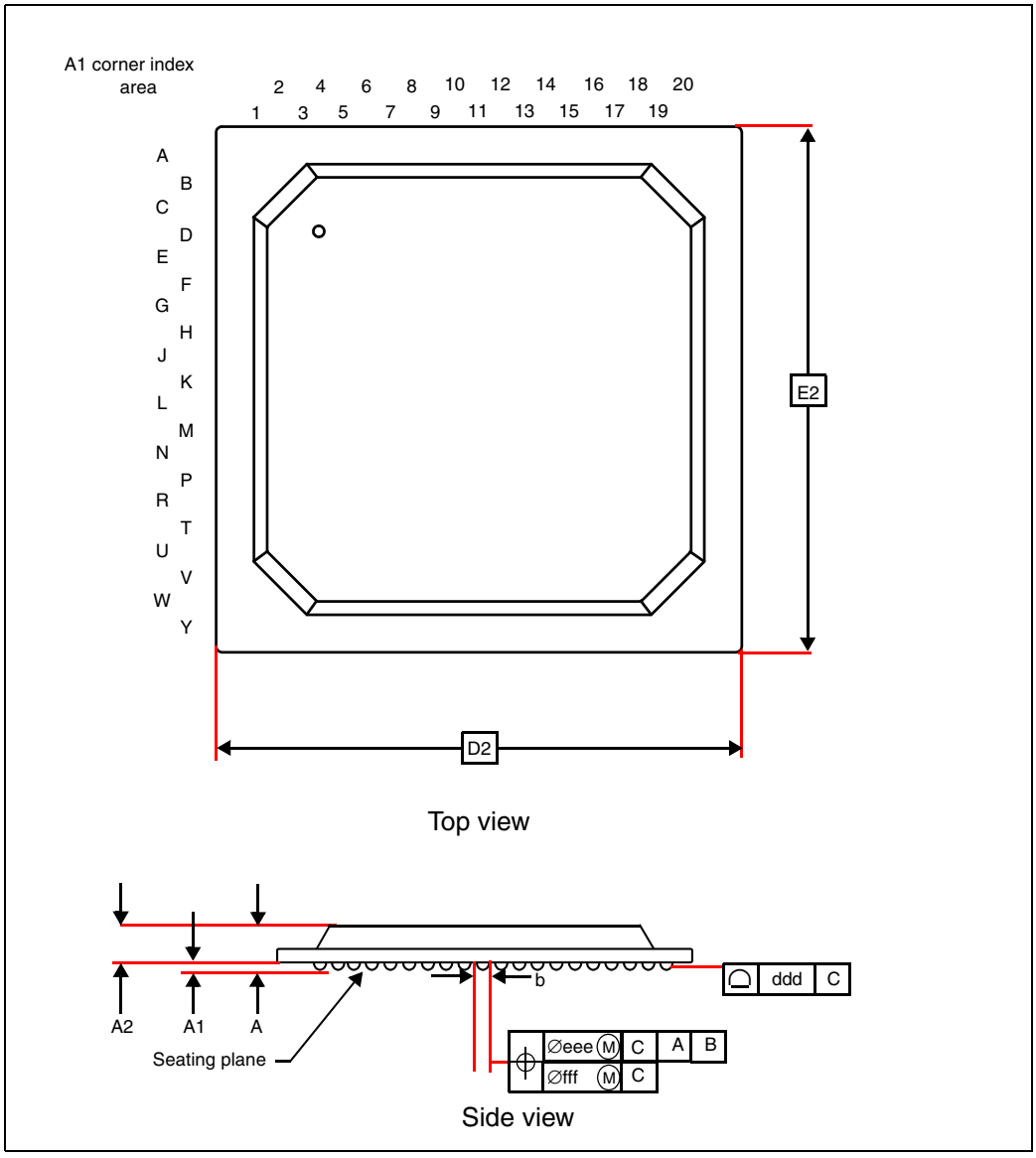


Table 40: Package dimensions

Ref	Dimensions						Description
	Databook (mm)			Drawing (mm)			
	Min	Typical	Max	Min	Typical	Max	
A			2.6			2.6	Overall thickness
A1	0.36			0.5		0.7	Ball height
A2			1.9	1.63		1.9	Body thickness
b	0.6	0.75	0.9	0.6	0.75	0.9	Ball diameter
D	26.8	27	27.2	26.8	27.0	27.2	Body size
D1		24.13			24.13		Ball footprint
E	26.8	27	27.2	26.8	27.0	27.2	Body size
E1		24.13			24.13		Ball footprint
e		1.27			1.27		Ball pitch
f		1.435			1.435		Ball to edge
. ddd			0.2			0.2	Co-planarity
. eee (3)		0.15			0.15		Cylindrical tolerance
. fff (4)		0.075			0.75		Cylindrical tolerance

Revision history

Date	Revision	Changes	
10-May-2005	2	Version number incremented from G (ADCS 7260755H) to 2 due to Internal Document Management System change Changed VDD_{core} range to 1.80V-1.95V (instead of 1.65V-1.95V) for ST40RA200 sales type in *Section 3 on page 19* and *Section 7.1.2 on page 57*	
13-Aug-2003	G	**4 Architecture** *Section 4.2.3: Standard ST40 peripherals on page 21*	New watchdog timer section
		5 System configuration *Section 5.7: EMI pin to function relationship on page 32*	New section
		7 Electrical specifications *Section 7.1.2: Operating conditions on page 57* *Section 7.3: PCI interface AC specifications on page 63* *Section 7.4: LMI interface (SDRAM) AC specifications on page 64* *Section 7.7: General purpose peripheral bus (EMI) AC specifications on page 67* *Section 7.8: PIO AC specifications on page 68* *Section 7.10: Low power CLKIN AC specifications on page 70*	IWP, LVREF updated, VIH1 defined tPCIHAIX changed tLCHLOV, tLIVLCH changed tECHCH, tECLCL, tECLEOV, tECHEOV changed tPCHPOV changed, tPIOf description changed tLCLLCL changed
		9 Package	New information