SHARC+ Dual Core DSP with ARM Cortex-A5

Preliminary Technical Data

ADSP-SC570/571/ADSP-21571

SYSTEM FEATURES

Dual enhanced SHARC+ high performance floating-point cores
- **Up to 450 MHz per SHARC+ core**
- **Up to 3M bits (384 kB) L1SRAM memory per core with parity (optional ability to configure as cache)**
- **32-bit, 40-bit, and 64-bit floating-point support**
- **32-bit fixed point**
- **Byte, short-word, word, long-word addressed**

ARM Cortex-A5 core
- **450 MHz/720 DMIPS with Neon/VFPv4-D16/Jazelle**
- **32 kB L1 instruction cache with parity/32 kB L1 data cache with parity**
- **256 kB L2 cache with parity**

Powerful DMA system

On-chip memory protection

Integrated safety features

17 mm × 17 mm 400-ball CSP BGA and 176-lead LQFP-EP, RoHS compliant

Low system power across automotive temperature range

MEMORY

Large on-chip L2 SRAM with ECC protection, up to 1MB

One L3 interface optimized for low system power, providing 16-bit interface to DDR3, DDR2 or LPDDR1 SDRAM devices

ADDITIONAL FEATURES

Security and Protection
- **Crypto hardware accelerators**
- **Fast secure boot with IP protection**
- **Support for TrustZone®**

Accelerators
- **FIR, IIR, offload engines**

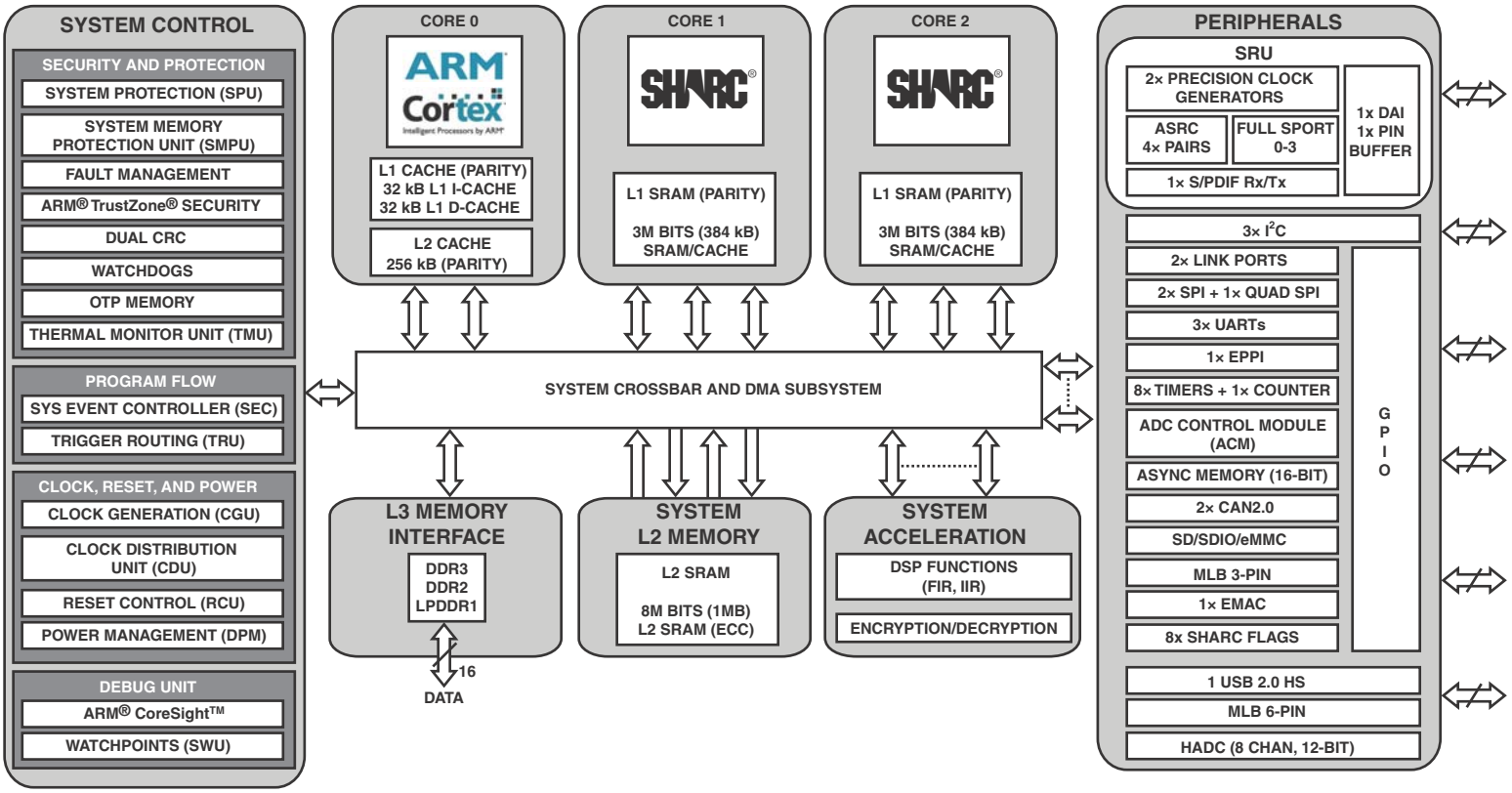


Figure 1. Processor Block Diagram

Rev. PrD **Document Feedback**

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106 U.S.A.
Tel: 781.329.4700 **Technical Support** **www.analog.com**

TABLE OF CONTENTS

General Description 3
ARM Cortex-A5 Processor 5
SHARC Processor 6
SHARC+ Core Architecture 8
System Infrastructure 10
System Memory Map 11
Security Features 13
Safety Features 14
Processor Peripherals 14
System Acceleration 19
System Design 19
System Debug 21
Development Tools 22
Additional Information 23
Related Signal Chains 23
Security Features Disclaimer 23
ADSP-SC57x/ADSP-2157x Detailed Signal Descriptions 24
400-Ball CSP_BGA Signal Descriptions 28
GPIO Multiplexing for 400-Ball CSP_BGA 35
176-Lead LQFP Signal Descriptions 38
GPIO Multiplexing for 176-Lead LQFP 43
ADSP-SC57x/ADSP-2157x Designer Quick Reference 45
Specifications 57
Operating Conditions 57
Electrical Characteristics 60
Absolute Maximum Ratings 63
ESD Sensitivity 64
Package Information 64
Timing Specifications 65
Environmental Conditions 122
ADSP-SC57x/ADSP-2157x 400-Ball BGA Ball Assignments 123
Numerical by Ball Number 123
Alphabetical by Pin Name 126
Configuration of the 400-Ball CSP_BGA 129
ADSP-SC57x/ADSP-2157x 176-Lead LQFP Lead Assignments 130
Numerical by Lead Number 130
Alphabetical by Pin Name 132
Configuration of the 176-Lead LQFP Lead Configuration 133
Outline Dimensions 134
Surface-Mount Design 135
Pre Release Products 136

REVISION HISTORY

6/2017—Revision PrC to Revision PrD

Removed products ADSP-SC572, ADSP-SC573, and ADSP-21573 from the title, header, and Processor Comparison tables. These products are released and have a final Rev 0 data sheet.

Removed engineering samples ADSP-SC573-BCZENG and ADSP-21573-BCZENG from the Pre Release Products table. These engineering samples are no longer available, as they have been released.

GENERAL DESCRIPTION

The ADSP-SC57x/ADSP-2157x processors are members of the SHARC® family of products. The ADSP-SC57x processor is based on the SHARC+ dual-core and the ARM® Cortex-A5™ core. The ADSP-SC57x/ADSP-2157x SHARC processors are members of the SIMD SHARC family of DSPs that feature Analog Devices Super Harvard Architecture. These 32-bit/40-bit/64-bit floating-point processors are optimized for high performance audio/floating-point applications with their large on-chip SRAM, multiple internal buses to eliminate I/O bottlenecks, and innovative digital audio interfaces (DAI). New enhancements to the SHARC+ core add cache enhancements, branch prediction, and other instruction set improvements—all while maintaining instruction set compatibility to previous SHARC products.

By integrating a rich set of industry-leading system peripherals and memory (see Table 1, Table 2, and Table 3), the ARM/SHARC processor is the platform of choice for next-generation applications that require RISC-like programmability, multimedia support, and leading-edge signal processing in one integrated package. These applications span a wide array of markets, from automotive and pro-audio to industrial-based applications that require high floating-point performance.

Table 1. Common Product Features

	ADSP-SC57x / ADSP-2157x
DAI (includes SRU)	1
Full SPORTs	1×4
S/PDIF Rx/Tx	1×1
ASRCs	1×4
Precision Clock Generators	1×2
Pin Buffers	1×20
I^2C (TWI)	3
Quad Data Bit SPI	1
Dual Data Bit SPI	2
CAN2.0	2
UARTs	3
Enhanced PPI: up to 16-bit on BGA, 12-bit on LQFP	1
GP Timer	8
GP Counter	1
Watchdog Timers	3
ADC Control Module	Yes
Hardware Accelerators	
FIR/IIR	Yes
Security Crypto Engine	Yes
Multi-channel 12-bit ADC	8-ch BGA; 4-ch LQFP

Table 2 provides feature comparison information for features that vary across the standard processors.

Table 3 provides feature comparison information for features that vary across the automotive processors.

Table 2. Processor Comparison[1]

Processor Feature		ADSP-SC570	ADSP-SC571	ADSP-21571
ARM Cortex-A5 (MHz max)		450/225	450	N/A
ARM Core L1 Cache (I, D kB)		32, 32	32, 32	N/A
ARM Core L2 Cache (kB)		256	256	N/A
SHARC+ Core1 (MHz max)		450	450	450
SHARC+ Core2 (MHz max)		N/A	450	450
SHARC L1 SRAM/core (kB)		1×384	2×384	2×384
System Memory	L2 SRAM (shared) (MB)	1	1	1
	DDR3/DDR2/LPDDR1 Controller (16-bit)	N/A	N/A	N/A
USB 2.0 HS + PHY (host/device/OTG)		N/A	N/A	N/A
EMAC Std/AVB + Timer IEEE-1588		10/100	10/100	N/A
SDIO/eMMC		N/A	N/A	N/A
Link Ports		1	1	1
GPIO Ports		Port A–D	Port A–D	Port A–D
GPIO + DAI Pins		64 + 20	64 + 20	64 + 20
Package Options		176-LQFP	176-LQFP	176-LQFP

[1] N/A denotes not applicable.

Table 3. Processor Comparison for Automotive[1]

Processor Feature		ADSP-SC570W	ADSP-SC571W	ADSP-21571W
ARM Cortex-A5 (MHz max)		450/225	450	N/A
ARM Core L1 Cache (I, D kB)		32, 32	32, 32	N/A
ARM Core L2 Cache (kB)		256	256	N/A
SHARC+ Core1 (MHz max)		450	450	450
SHARC+ Core2 (MHz max)		N/A	450	450
SHARC L1 SRAM/core (kB)		1×384	1×384	2×384
System Memory	L2 SRAM (shared) (MB)	1	1	1
	DDR3/DDR2/LPDDR1 Controller (16-bit)	N/A	N/A	N/A
USB 2.0 HS + PHY (host/device/OTG)		N/A	N/A	N/A
EMAC Std/AVB + Timer IEEE-1588		10/100	10/100	N/A
SDIO/eMMC		N/A	N/A	N/A
MLB		3-pin	3-pin	3-pin
Link Ports		1	1	1
GPIO Ports		Port A–D	Port A–D	Port A–D
GPIO + DAI Pins		64 + 20	64 + 20	64 + 20
Package Options		176-LQFP	176-LQFP	176-LQFP

[1] N/A denotes not applicable.

ARM CORTEX-A5 PROCESSOR

The ARM Cortex-A5 processor (Figure 2) is a high performance processor with the following features:

- Instruction and Data L1 cache units (32/32K bytes)
- In-order pipeline with dynamic branch prediction
- ARM, Thumb, and ThumbEE instruction set support
- TrustZone security extensions
- Harvard level 1 memory system with a Memory Management Unit (MMU)
- ARM v7 debug architecture
- Trace support through an Embedded Trace Macrocell (ETM) interface
- Extension: Vector Floating-point Unit (IEEE754) with trapless execution
- Extension: Media Processing Engine (MPE) with NEON™ technology
- Extension: Jazelle hardware acceleration

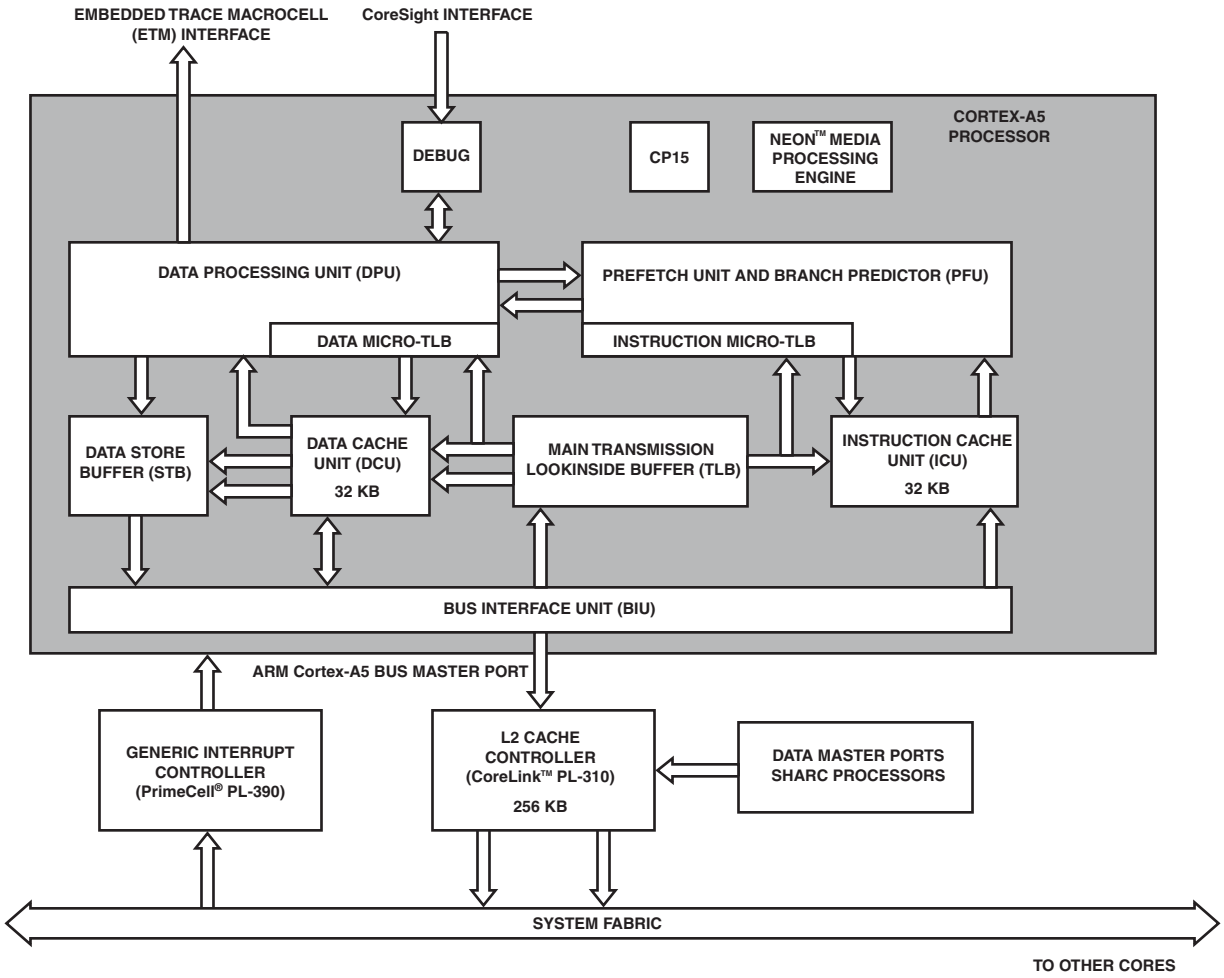


Figure 2. ARM Cortex A-5 Processor Block Diagram

Generic Interrupt Controller, PL390 (ADSP-SC57x only)

The Generic Interrupt Controller (GIC) is a centralized resource for supporting and managing interrupts. The ADSP-SC57x processor has a uniprocessor implementation of the GIC. The GIC splits logically into GICPORT0 (distributor block) and GICPORT1 (CPU interface block).

Generic Interrupt Controller Port0 (GICPORT0)

The GICPORT0 (distributor) performs interrupt prioritization and distribution to the GICPORT1 (CPU interface) blocks that connect to the processors in the system. It centralizes all interrupt sources, determines the priority of each interrupt, and forwards the interrupt with the highest priority to the interface, for priority masking and preemption handling.

Generic Interrupt Controller Port1 (GICPORT1)

GICPORT1 (CPU interface) block performs priority masking and preemption handling for a connected processor in the system. GICPORT1 supports 8 SGIs (software generated interrupts) and 212 SPIs (shared peripheral interrupts).

L2 Cache Controller, PL310 (ADSP-SC57x only)

The L2 cache controller PL310 (Figure 2) works efficiently with ARM processors that implement system fabric. The cache controller directly interfaces on the data and instruction interface. The internal pipelining of the cache controller is optimized to enable the processors to operate at the same clock frequency. The cache controller supports:

- Two read/write 64-bit slave ports for interfacing with data and instruction interfaces or for data between ARM and SHARC cores
- Two read/write 64-bit master ports for interfacing with the system fabric

SHARC PROCESSOR

As shown in Figure 3, the SHARC processor integrates a SHARC+ SIMD core, L1 memory crossbar, I/D cache controller, L1 memory blocks, and the master/slave ports. The SHARC+ SIMD core is shown in Figure 4.

The SHARC processor supports a modified Harvard architecture in combination with a hierarchical memory structure. Level 1 (L1) memories typically operate at the full processor speed with little or no latency.

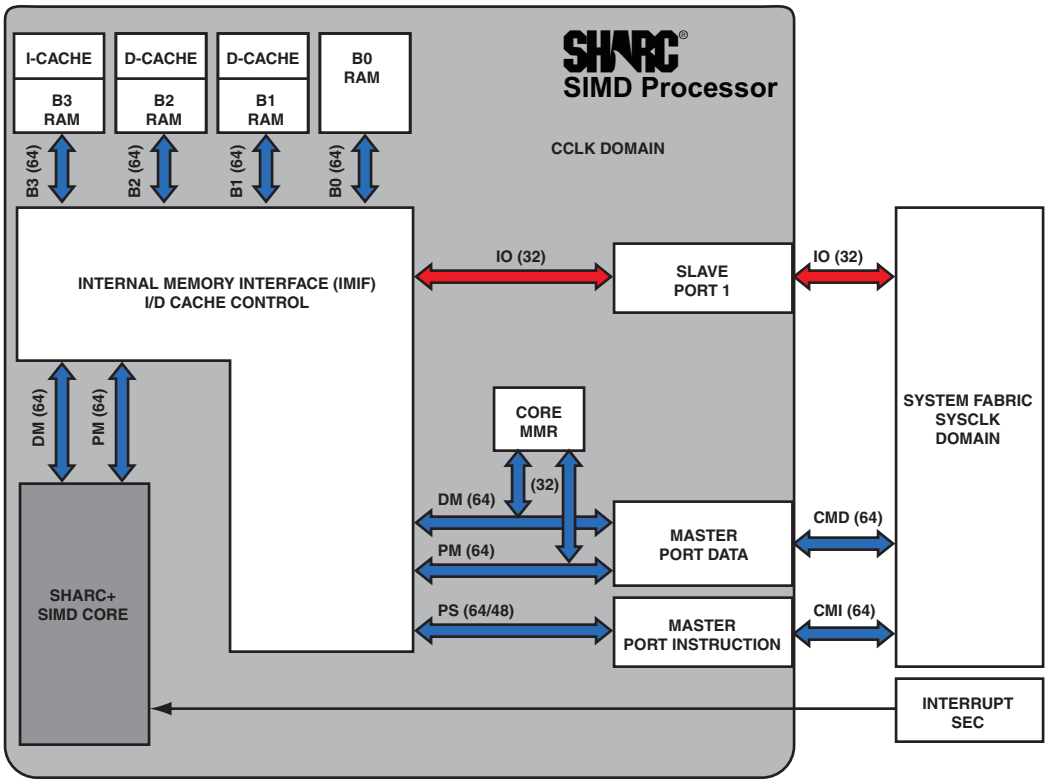


Figure 3. SHARC Processor Block Diagram

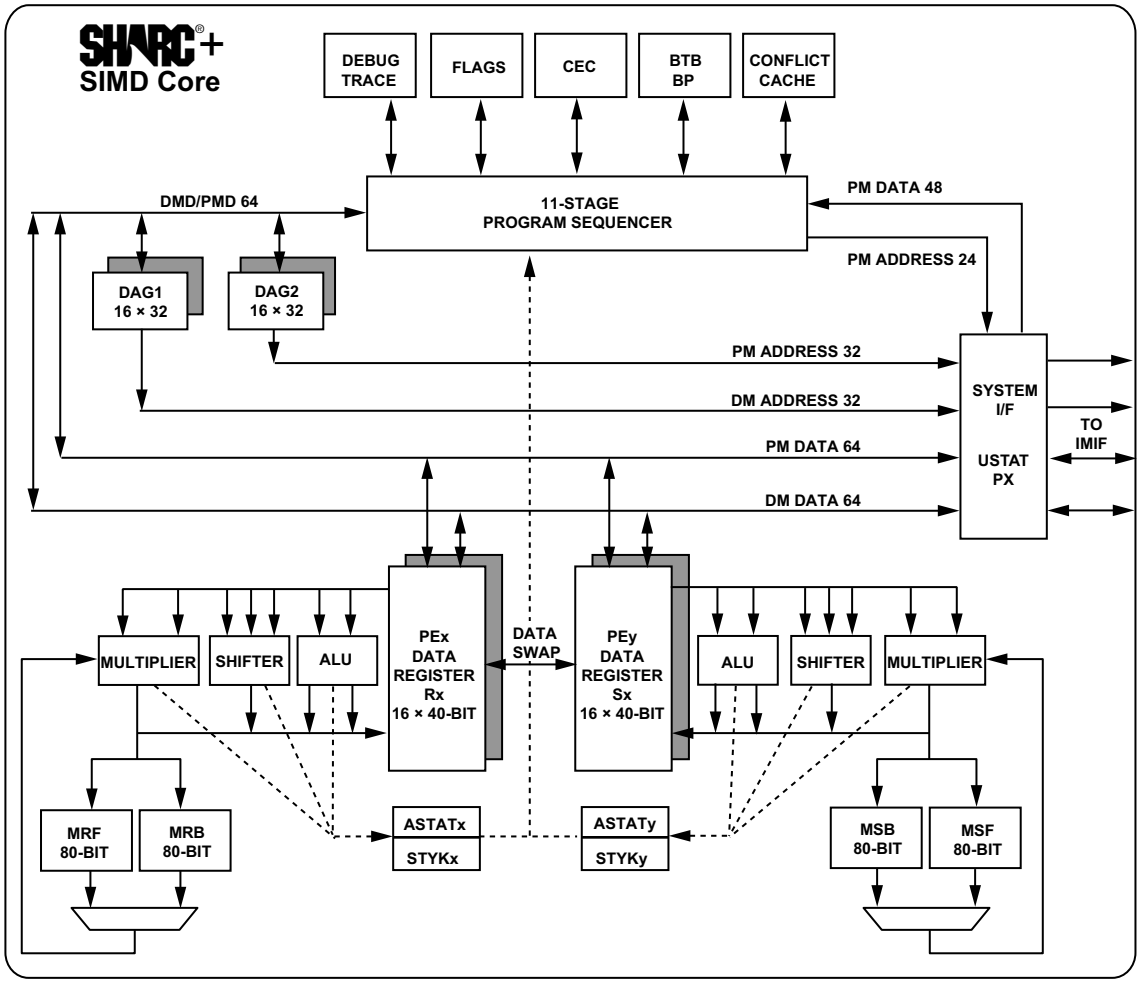


Figure 4. SHARC+ SIMD Core Block Diagram

L1 Memory

The ADSP-SC57x/ADSP-2157x memory structure is shown in Figure 5 on Page 8. Each SHARC+ core has a tightly coupled Level 1 (L1) SRAM of up to 3 Mbits. Each SHARC+ core can access code and data in a single cycle from this memory space. The ARM Cortex-A5 core can also access these memory spaces with multi-cycle accesses.

In the SHARC+ core private address space, both cores have their own L1 memory.

SHARC+ core MMR address space is 0x 0000 0000-0x0003 FFFF in Normal Word (32-bit). Each block can be configured for different combinations of code and data storage. Of the 3M bits SRAM, up to 1024K/512K bits can be configured for DM, PM, and instruction cache. Each memory block supports single cycle, independent accesses by the core processor and I/O processor. The memory architecture, in combination with its separate on-chip buses, allows two data transfers from the core and one from the I/O processor in a single cycle. The processor's SRAM can be configured as a maximum of 96K words of 32-bit data, 192K words of 16-bit data, 64K words of 48-bit instructions (or 40-bit data), or combinations of different word sizes up to 5 M bits. All of the memory can be accessed as 8-bit, 16-bit, 32-bit, 48-bit, or 64-bit words. A 16-bit floating-point storage format is supported that effectively doubles the amount of data that may be stored on chip.

Conversion between the 32-bit floating-point and 16-bit floating-point formats is performed in a single instruction. While each memory block can store combinations of code and data, accesses are most efficient when one block stores data using the DM bus for transfers, and the other block stores instructions and data using the PM bus for transfers.

Using the DM bus and PM buses, with one bus dedicated to a memory block, assures single-cycle execution with two data transfers. In this case, the instruction must be available in the cache. The system configuration is flexible, but a configuration is 512K bits DM, 128K bits PM, and 128K bits of cache, with the remaining L1 memory configured as SRAM is typical. Each addressable memory space outside the L1 memory can be accessed either directly or via cache.

The memory map in Table 4 on Page 11 gives the L1 memory address space.

In addition, there are multiple L1 memory blocks offering a configurable mix of SRAM and cache.

L1 Master and Slave Ports

Each SHARC+ core has two master ports and one slave port to/from the system fabric. One master port fetches instructions; the second master port drives data to the system world. For the slave port address, refer to the L1 memory address map.

L1 On-Chip Memory Bandwidth

The internal memory architecture allows programs to have four accesses at the same time to any of the four blocks (assuming no block conflicts). The total bandwidth is realized using the DMD and PMD buses (2 × 64-bits, CCLK speed and 2 × 32-bit, SYSCLK speed).

Instruction and Data Cache

The ADSP-SC57x/ADSP-2157x processors also include a traditional instruction cache (I-cache) and two data caches (PM/DM caches) with parity support for all caches. Together, these caches support one instruction access and two data accesses, over the DM and PM buses, per cycle. The cache controllers automatically manage the configured part of the L1 memory. The system can configure part of the L1 memory for automatic management by the cache controllers. The sizes of these caches are independently configurable from zero to a maximum of 128 kB each. The memory not managed by the cache controllers is directly addressable by the processors. The controllers ensure the data coherence between the two data caches. The caches provide user-controllable features such as locking (full as well as partial), range-bound invalidation, and flushing.

System Event Controller Input

The output of the SEC controller is forwarded to the Core Event Controller (CEC) to respond directly to all unmasked system based interrupts. It also supports nesting including various SEC interrupt channel arbitration options. For all SEC channels the processor automatically stacks the arithmetic status (ASTATx, and STATy) registers and mode (MODE1) registers in parallel with the interrupt servicing.

Core Memory-Mapped Registers (CMMR)

The core memory mapped registers control L1 I/D cache, BTB, L2 cache, parity error, system control, debug and monitor functions.

SHARC+ CORE ARCHITECTURE

The ADSP-SC57x/ADSP-2157x processors are code compatible at the assembly level with the ADSP-2148x, ADSP-2147x, ADSP-2146x, ADSP-2137x, ADSP-2136x, ADSP-2126x, ADSP-21160, and ADSP-21161, and with the first generation ADSP-2106x SHARC processors. The ADSP-SC57x/ADSP-2157x processors share architectural features with the ADSP-2126x, ADSP-2136x, ADSP-2137x, ADSP-214xx and ADSP-2116x SIMD SHARC processors, as shown in Figure 4 and detailed in the following sections.

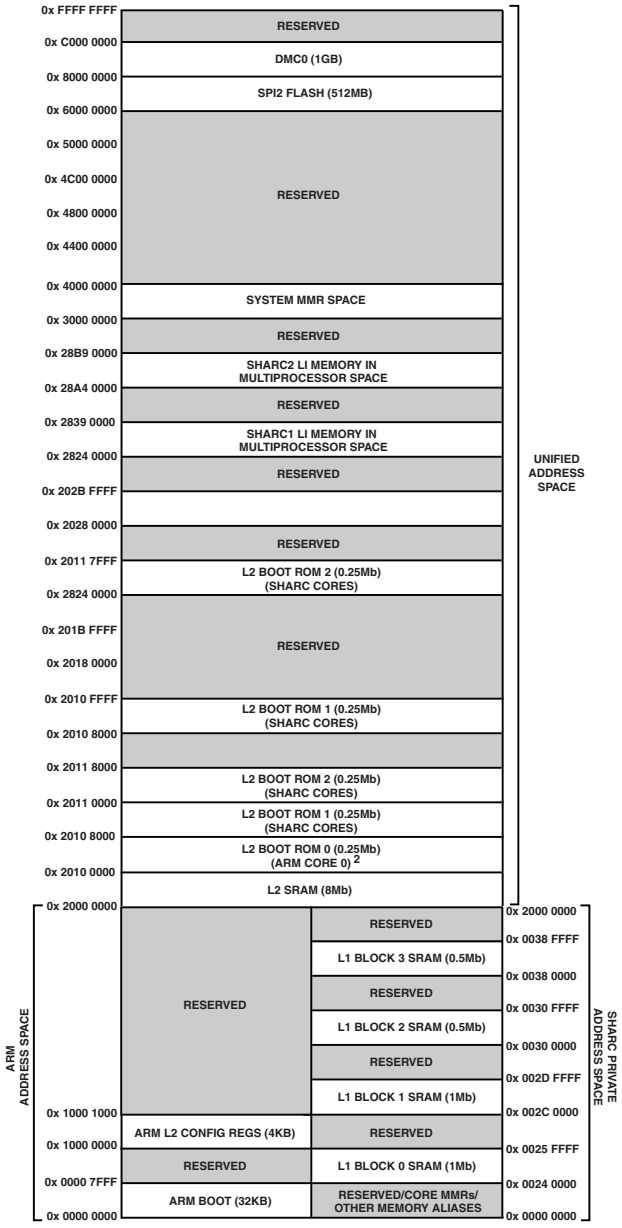


Figure 5. ADSP-SC57x/ADSP-2157x Memory Map

SIMD Computational Engine

The SHARC+ core contains two computational processing elements that operate as a single-instruction, multiple-data (SIMD) engine. The processing elements are referred to as PEx and PEy and each contains an ALU, multiplier, shifter, and register file. PEx is always active, and PEy is enabled by setting the PEYEN mode bit in the MODE1 register. SIMD mode allows the processors to execute the same instruction in both processing elements, but each processing element operates on different data. This architecture is efficient at executing math-

intensive DSP algorithms. In addition to all the features of its predecessors, the SHARC+ core also provides a new and simpler way to execute an instruction only on PEy.

SIMD mode also affects the way data transfers between memory and the processing elements because to sustain computational operation in the processing elements requires twice the data bandwidth. Therefore, entering SIMD mode also doubles the bandwidth between memory and the processing elements. When using the DAGs to transfer data in SIMD mode, two data values transfer with each memory or register file access.

Independent, Parallel Computation Units

Within each processing element is a set of pipelined computational units. The computational units consist of an arithmetic/logic unit (ALU), multiplier, and shifter. These units are arranged in parallel, maximizing computational throughput. These computational units support IEEE 32-bit single-precision floating-point, 40-bit extended-precision floating-point, IEEE 64-bit double-precision floating-point, and 32-bit fixed-point data formats.

Multifunction instruction set supports parallel execution of ALU and multiplier operations. In SIMD mode, the parallel ALU and multiplier operations occur in both processing elements per core.

All processing operations take one cycle to complete. For all floating-point operations, the processor takes two cycles to complete in case of data dependency. Double-precision floating-point data take two to six cycles to complete. The processor stalls for the appropriate number of cycles (interlocked pipeline plus data dependency check).

Core Timer

Each SHARC+ processor core also has its own dedicated timer. This extra timer is clocked by the internal processor clock and is typically used as a system tick clock for generating periodic operating system interrupts.

Data Register File

Each processing element contains a general-purpose data register file. The register files transfer data between the computation units and the data buses, and store intermediate results. These 10-port, 32-register (16 primary, 16 secondary) register files, combined with the enhanced Harvard architecture of the processor, allow unconstrained data flow between computation units and internal memory. The registers in PEx are referred to as R0-R15 and in PEy as S0-S15.

Context Switch

Many of the registers of the processor have secondary registers that can activate during interrupt servicing for a fast context switch. The data registers in the register file, the DAG registers, and the multiplier result registers all have secondary registers. The primary registers are active at reset, while control bits in a mode control register activate the secondary registers.

Universal Registers

General-purpose tasks use these registers. The USTAT (4) registers allow easy bit manipulations (Set, Clear, Toggle, Test, XOR) for all peripheral registers (control/status).

The data bus exchange register (PX) permits data to pass between the 64-bit PM data bus and the 64-bit DM data bus, or between the 40-bit register file and the PM/DM data bus. These registers contain hardware to handle the data width difference.

Data Address Generators With Zero-Overhead Hardware Circular Buffer Support

For indirect addressing and implementing circular data buffers in hardware, the processor uses the two data address generators (DAGs). Circular buffers allow efficient programming of delay lines and other data structures required in digital signal processing, and are commonly used in digital filters and Fourier transforms. The two DAGs of the processors contain sufficient registers to allow the creation of up to 32 circular buffers (16 primary register sets, 16 secondary). The DAGs automatically handle address pointer wraparound, reduce overhead, increase performance, and simplify implementation. Circular buffers can start and end at any memory location.

Flexible ISA Instruction Set

The 48-bit instruction word accommodates various parallel operations for concise programming. For example, the processors can conditionally execute a multiply, an add, and a subtract in both processing elements while branching and fetching up to four 32-bit values from memory—all in a single instruction. Additionally, the double precision floating-point instruction set is an addition to the SHARC+ core.

Variable Instruction Set Architecture (VISA)

In addition to supporting the standard 48-bit instructions (ISA) from previous SHARC processors, the SHARC+ core processors support new instructions of 16 and 32 bits for ADSP-214xx products. This feature, called Variable Instruction Set Architecture (VISA), drops redundant/unused bits within the 48-bit instruction to create more efficient and compact code. The program sequencer supports fetching these 16-bit and 32-bit instructions from both internal and external memories. VISA is not an operating mode, it is only address-dependent (refer to memory map ISA/VISA address spaces). Furthermore, it allows jumps between ISA and VISA instruction fetches.

Single-Cycle Fetch of Instructional Four Operands

The ADSP-SC57x/ADSP-2157x processors feature an enhanced Harvard architecture in which the data memory (DM) bus transfers data and the program memory (PM) bus transfers both instructions and data.

With its separate program and data memory buses and on-chip instruction conflict-cache, the processors can simultaneously fetch four operands (two over each data bus) and one instruction (from the conflict-cache), all in a single cycle.

Core Event Controller (CEC)

The SHARC+ core IVT generates various core interrupts (arithmetic and circular buffer instruction flow exceptions) and SEC events (debug/monitor, software). The core only responds to interrupts which are unmasked (IMASK register).

Instruction Conflict-Cache

The processors include a 32-entry instruction cache that enables three-bus operation for fetching an instruction and four data values. The cache is selective—only the instructions whose fetches conflict with PM bus data accesses cache. This cache allows full speed execution of core, looped operations such as digital filter multiply-accumulates, and FFT butterfly processing. The conflict cache serves for on-chip bus conflicts only.

Branch Target Buffer/Branch Predictor

Implementation of a hardware-based branch predictor (BP) and branch target buffer (BTB) reduce branch delay. The program sequencer supports efficient branching using this branch target buffer (BTB) for conditional and unconditional instructions.

Addressing Spaces

In addition to traditionally supported long-word, normal word, extended-precision word and short word addressing aliases, the processors support byte addressing for the data and instruction accesses. The enhanced ISA/VISA provides new instructions for accessing all sizes of data from byte space as well as for converting word addresses to byte and byte to word addresses.

Additional Features

The enhanced ISA/VISA of the ADSP-SC57x/ADSP-2157x processors also provides memory barrier instruction (sync) for data synchronization, exclusive data accesses support for multi-core data sharing, and exclusive data accesses to enable multiprocessor programming. To enhance the reliability of application, L1 data RAMs support parity error detection logic for every byte. Additionally, the processors detect illegal opcodes. Core interrupts flag both the errors. Master ports of the core also detect for failed external accesses.

SYSTEM INFRASTRUCTURE

The following sections describe the system infrastructure of the ADSP-SC57x/ADSP-2157x processors.

System L2 Memory

A system level L2 memory of 1MB is also available to both SHARC+ cores, the ARM Cortex-A5 core, and DMA channels. (See Table 5.) Memory accesses to this memory space are multi-cycle accesses by both the ARM and SHARC+ cores. The memory space is used for various cases including:

- ARM-to-SHARC+ core data sharing and inter-core communications
- Accelerator and peripheral source and destination memory to avoid having to access data in external memory
- A location for DMA descriptors
- Storage for additional data for either ARM or SHARC+ cores to avoid external memory latencies and reduce external memory bandwidth
- Storage for incoming Ethernet traffic to improve performance
- Storage for data coefficient tables cached by the SHARC+ core

See System Memory Protection Unit (SMPU) for options in limiting access by specific cores and DMA masters.

The ARM Cortex-A5 core has an L1 instruction and data cache, each of which is 32 kB in size. An L2 cache controller of 256 kB is also available. When enabling the caches, accesses to all other memory spaces (internal and external) go through the cache.

SHARC+ Core L1 Memory in Multiprocessor Space

The ARM Cortex-A5 core can access the L1 memory of the SHARC+ core. See Table 6 for the L1 memory address in multiprocessor space. The SHARC+ core can access the L1 memory of the other SHARC+ core in the multiprocessor space.

One-Time-Programmable Memory (OTP)

The processors feature 7K bits of One-Time-Programmable (OTP) memory which is memory-map accessible. This memory stores a unique chip identification and supports secure-boot and secure operation.

I/O Memory Space

The static memory controller (SMC) is programmed to control up to two blocks of external memories or memory-mapped devices, with flexible timing parameters. Each block occupies an 8K byte segment regardless of the size of the device used. Mapped I/Os also include SPI2 memory address space. See Table 7.

SYSTEM MEMORY MAP

Table 4. L1 Block 0, 1, 2, and 3 SHARC Addressing Memory Map (Private Address Space)

Memory	Long Word (64 Bits)	Extended Precision/ ISA Code (48 Bits)	Normal Word (32 Bits)	Short Word/ VISA Code (16 Bits)	Byte Access (8 Bits)
L1 Block 0 SRAM (1 Mb)	0x00048000–0x0004BFFF	0x00090000–0x00095554	0x00090000–0x00097FFF	0x00120000–0x0012FFFF	0x00240000–0x0025FFFF
L1 Block 1 SRAM (1 Mb)	0x00058000–0x0005BFFF	0x000B0000–0x000B5554	0x000B0000–0x000B7FFF	0x00160000–0x0016FFFF	0x002C0000–0x002DFFFF
L1 Block 2 SRAM (0.5 Mb)	0x00060000–0x00061FFF	0x000C0000–0x000C2AA9	0x000C0000–0x000C3FFF	0x00180000–0x00187FFF	0x00300000–0x0030FFFF
L1 Block 3 SRAM (0.5 Mb)	0x00070000–0x00071FFF	0x000E0000–0x000E2AA9	0x000E0000–0x000E3FFF	0x001C0000–0x001C7FFF	0x00380000–0x0038FFFF

Table 5. L2 Memory Addressing Map

Memory	Byte Address Space ARM – Data Access and Instruction Fetch SHARC – Data Access	Normal Word Address Space for Data Access SHARC	Instruction Fetch VISA Address Space SHARC	Instruction Fetch ISA Address Space SHARC
L2 Boot-ROM0[1]	ARM: 0x00000000–0x00007FFF SHARC/DMA: 0x20100000–0x20107FFF	0x08040000–0x08041FFF	0x00B20000–0x00B23FFF	0x00580000–0x00581555
L2 RAM (8 Mb)	0x20000000–0x200FFFFF	0x08000000–0x0803FFFF	0x00B80000–0x00BFFFFF	0x005C0000–0x005EAAAA
Boot ROM1	0x20108000–0x2010FFFF	0x08042000–0x08043FFF	0x00B00000–0x00B03FFF	0x00500000–0x00501555
Boot ROM2	0x20110000–0x20117FFF	0x08044000–0x08045FFF	0x00B40000–0x00B43FFF	0x00540000–0x00541555

[1] From the ARM point of view, L2 Boot-ROM0 byte address space is 0x 0000 0000–0x 0000 7FFF.

Table 6. SHARC L1 Memory in Multiprocessor Space

		Memory Block	Byte Address Space for ARM and SHARC	Normal Word Address Space for SHARC
L1 Memory of SHARC1 in Multiprocessor Space	Address via Slave1 Port	Block0	0x28240000–0x2825FFFF	0x0A090000–0x0A097FFF
		Block1	0x282C0000–0x282DFFFF	0x0A0B0000–0x0A0B7FFF
		Block2	0x28300000–0x2830FFFF	0x0A0C0000–0x0A0C3FFF
		Block3	0x28380000–0x2838FFFF	0x0A0E0000–0x0A0E3FFF
L1 Memory of SHARC2 in Multiprocessor Space	Address via Slave1 Port	Block0	0x28A40000–0x28A70000	0x0A290000–0x0A297FFF
		Block1	0x28AC0000–0x28AF0000	0x0A2B0000–0x0A2B7FFF
		Block2	0x28B00000–0x28B20000	0x0A2C0000–0x0A2B3FFF
		Block3	0x28B80000–0x28BA0000	0x0A2E0000–0x0A2E3FFF

Table 7. Memory Map of Mapped I/Os

	Byte Address Space ARM – Data Access and Instruction Fetch SHARC – Data Access	Normal Word Address Space for Data Access SHARC	SHARC Core Instruction Fetch	
			VISA Space	ISA Space
SPI2 Memory (512 MB)	0x60000000–0x7FFFFFFF	0x04000000–0x07FFFFFF	0x00F80000–0x00FFFFFF	0x00780000–0x007FFFFF

Table 8. DMC Memory Map

	Byte Address Space ARM – Data Access and Instruction Fetch SHARC – Data Access	Normal Word Address Space for Data Access SHARC	SHARC Core Instruction Fetch	
			VISA Space	ISA Space
DMC0 – 1 GB	0x80000000–0xBFFFFFFF	0x10000000–0x17FFFFFF	0x00800000–0x00AFFFFF	0x00400000–0x004FFFFF

System Crossbars (SCB)

The system crossbars (SCB) are the fundamental building blocks of a switch-fabric style for (on-chip) system bus interconnection. The SCBs connect system bus masters to system bus slaves, providing concurrent data transfer between multiple bus masters and multiple bus slaves. A hierarchical model—built from multiple SCBs—provides a power and area efficient system interconnect, which satisfies the performance and flexibility requirements of a specific system.

The SCBs provide the following features:

- Highly efficient, pipelined bus transfer protocol for sustained throughput
- Full-duplex bus operation for flexibility and reduced latency
- Concurrent bus transfer support to allow multiple bus masters to access bus slaves simultaneously
- Protection model (privileged/secure) support for selective bus interconnect protection

Direct Memory Access (DMA)

The processors use direct memory access (DMA) to transfer data within memory spaces or between a memory space and a peripheral. The processors can specify data transfer operations and return to normal processing while the fully integrated DMA controller carries out the data transfers independent of processor activity.

DMA transfers can occur between memory and a peripheral or between one memory and another memory. Each memory-to-memory DMA stream uses two channels, where one channel is the source channel, and the second is the destination channel.

All DMAs can transport data to and from all on-chip and off-chip memories. Programs can use two types of DMA transfers, descriptor-based or register-based. Register-based DMA allows the processors to program DMA control registers directly to initiate a DMA transfer. On completion, the control registers automatically update with their original setup values for continuous transfer. Descriptor-based DMA transfers require a set of parameters stored within memory to initiate a DMA sequence. Descriptor-based DMA transfers allow multiple DMA sequences to be chained together. Program a DMA channel to set up and start another DMA transfer automatically after the current sequence completes.

The DMA engine supports the following DMA operations.

- A single linear buffer that stops on completion.
- A linear buffer with negative, positive or zero stride length.
- A circular, auto-refreshing buffer that interrupts when each buffer becomes full.
- A similar buffer that interrupts on fractional buffers (for example, 1/2, 1/4).
- 1D DMA – uses a set of identical ping-pong buffers defined by a linked ring of two-word descriptor sets, each containing a link pointer and an address.
- 1D DMA – uses a linked list of four word descriptor sets containing a link pointer, an address, a length, and a configuration.
- 2D DMA – uses an array of one-word descriptor sets, specifying only the base DMA address.
- 2D DMA – uses a linked list of multi-word descriptor sets, specifying everything.

Memory DMA (MDMA)

The processor supports various memory-to-memory DMA operations which include:

- Standard bandwidth MDMA channels with CRC protection (32-bit bus width, run on SYSCLK)
- Enhanced bandwidth MDMA channel (32-bit bus width, runs on SYSCLK)
- Maximum bandwidth MDMA channel (64-bit bus width, run on SYCLK)

Extended Memory DMA

Extended memory DMA supports various operating modes such as delay line (allows processor reads and writes to external delay line buffers and hence to external memory) with limited core interaction and scatter/gather DMA (writes to/from non-contiguous memory blocks).

CRC Protection

The CRC protection modules allow system software to calculate the signature of code or data or both in memory, the content of memory-mapped registers, or communication message objects periodically. Dedicated hardware circuitry compares the signature with pre calculated values and triggers appropriate fault events.

For example, every 100 ms the system software initiates the signature calculation of the entire memory contents and compares these contents with expected, pre calculated values. If a mismatch occurs, a fault condition is generated (through the processor core or the trigger routing unit).

The CRC is a hardware module based on a CRC32 engine that computes the CRC value of the 32-bit data words presented to it. The source channel of the memory-to-memory DMA (in memory scan mode) provides data. The data forwards optionally to the destination channel (memory transfer mode). The main features of the CRC peripheral are:

- Memory scan mode
- Memory transfer mode
- Data verify mode
- Data fill mode
- User-programmable CRC32 polynomial
- Bit/byte mirroring option (endianness)
- Fault/error interrupt mechanisms
- 1D and 2D fill block to initialize array with constants
- 32-bit CRC signature of a block of a memory or MMR block

Event Handling

The processors provide event handling that supports both nesting and prioritization. Nesting allows multiple event service routines to be active simultaneously. Prioritization ensures that servicing of a higher-priority event takes precedence over servicing of a lower-priority event. The processors provide support for five different types of events:

- Emulation – An emulation event causes the processors to enter emulation mode, allowing command and control of the processors through the JTAG interface.
- Reset – This event resets the processors.
- Exceptions – Events that occur synchronously to program flow (in other words, the exception is taken before the instruction is allowed to complete). Conditions triggered on the one side by the SHARC+ core, such as data alignment (SIMD/long word) or compute violations (fixed or floating point), and illegal instructions cause core exceptions. Conditions triggered on the other side by the SEC, such as ECC/parity/watchdog/system clock, cause system exceptions.
- Interrupts – Events that occur asynchronously to program flow. They are caused by input signals, timers, and other peripherals, as well as by an explicit software instruction.

System Event Controller (SEC)

Both SHARC+ cores feature a system event controller. System event controller features include the following:

- Comprehensive system event source management including interrupt enable, fault enable, priority, core mapping and source grouping
- Distributed programming model where each system event source control and all status fields are independent of all others
- Determinism where all system events have the same propagation delay and provide unique identification of a specific system event source
- Slave Control Port which provides access to all SEC registers for configuration, status, and interrupt/fault service model
- Global locking supports a register level protection model to prevent writes to “locked” registers
- Fault management including fault action configuration, time out, external indication, and system reset

Trigger Routing Unit (TRU)

The TRU provides system-level sequence control without core intervention. The TRU maps trigger masters (generators of triggers) to trigger slaves (receivers of triggers). Slave endpoints can be configured to respond to triggers in various ways. Common applications enabled by the TRU include:

- Automatically triggering the start of a DMA sequence after a sequence from another DMA channel completes
- Software triggering
- Synchronization of concurrent activities

SECURITY FEATURES

The following sections describe the security features of the ADSP-SC57x/ADSP-2157x processors.

ARM TrustZone

The ADSP-SC57x processors provide TrustZone® technology, which is tightly integrated into the Cortex-A5 processors. This technology enables a secure state which is also extended throughout the system through the AMBA® AXI™ buses and interconnect fabric.

Crypto Hardware Accelerators

The ADSP-SC57x/ADSP-2157x processors support standards-based hardware-accelerated encryption, decryption, authentication, and true random number generation.

Support for the following hardware-accelerated cryptographic ciphers include:

- AES in ECB, CBC, ICM and CTR modes with 128-bit, 192-bit, and 256-bit keys
- DES in ECB and CBC mode with 56-bit key
- 3DES in ECB and CBC mode with 3x 56-bit key
- ARC4 in stateful, stateless mode, up to 128-bit key

Support for the following hardware-accelerated hash functions include:

- SHA-1
- SHA-2 with 224-bit and 256-bit digest
- HMAC transforms for SHA-1 and SHA-2
- MD5

Public Key Accelerator is available to offload computation-intensive public key cryptography operations.

Both a hardware-based non-deterministic random number generator and pseudo-random number generator are available.

Secure boot is also available with 224-bit Elliptic Curve Digital Signatures ensuring integrity and authenticity of the boot stream. Optionally, ensuring confidentiality through AES-128 encryption is available.

CAUTION
This product includes security features that can be used to protect embedded nonvolatile memory contents and prevent execution of unauthorized code. When security is enabled on this device (either by the ordering party or the subsequent receiving parties), the ability of Analog Devices to conduct failure analysis on returned devices is limited. Contact Analog Devices for details on the failure analysis limitations for this device.

Employ Secure Debug to allow only trusted users to access the system with debug tools.

System Protection Unit (SPU)

The system protection unit (SPU) guards against accidental or unwanted access to an MMR space of the peripheral by providing a write-protection mechanism. The user is able to choose and configure the protected peripherals as well as configure which ones of the four system MMR masters (core, memory DMA, and CoreSight™ debug) the peripherals are guarded against.

The SPU is also part of the security infrastructure. Along with providing write-protection functionality, the SPU is employed to define which resources in the system are secure or non-secure and to block access to secure resources from non-secure masters.

System Memory Protection Unit (SMPU)

Synonymously, the system memory protection unit (SMPU) provides memory protection against read and/or write transactions to defined regions of memory. There are SMPU units in the ADSP-SC57x/ADSP-2157x processors for each memory space, except for SHARC L1 and SPI direct memory slave.

The SMPU is also part of the security infrastructure. It allows the user to not only protect against arbitrary read and/or write transactions, but it also allows regions of memory to be defined as secure and prevent non-secure masters from accessing those memory regions.

SAFETY FEATURES

The ADSP-SC57x/ADSP-2157x processors have been designed to support functional safety applications. While the level of safety is mainly dominated by the system concept, the following primitives are provided by the devices to build a robust safety concept.

Multi-Parity-Bit-Protected SHARC+ Core L1 Memories

In the SHARC+ core L1 memory space, whether SRAM or cache, each word is protected by multiple parity bits to detect the single event upsets that occur in all RAMs. The cache tags and BTB are also protected by parity.

Parity-Protected ARM L1Cache

In the Cortex-A5 L1 cache space, each word is protected by multiple parity bits to detect the single event upsets that occur in all RAMs. The cache tags are also protected by parity.

ECC-Protected L2 Memories

Error correcting codes (ECC) are used to correct single event upsets. The L2 memory is protected with a Single Error Correct-Double Error Detect (SEC-DED) code. By default ECC is enabled, but it can be disabled on a per-bank basis. Single-bit errors are transparently corrected. Dual-bit errors can issue a system event or fault if enabled. ECC protection is fully transparent to the user, even if L2 memory is read or written by 8-bit or 16-bit entities.

Parity-Protected Peripheral Memories

TBD.

CRC-Protected Memories

While parity bit and ECC protection mainly protect against random soft errors in L1 and L2 memory cells, the CRC engines can be used to protect against systematic errors (pointer errors) and static content (instruction code) of L1, L2 and even L3 memories (DDR2, LPDDR). The processors feature two CRC engines which are embedded in the memory-to-memory DMA controllers. CRC checksums can be calculated or compared on the fly during memory transfers, or one or multiple memory regions can be continuously scrubbed by a single DMA work unit as per DMA descriptor chain instructions. The CRC engine also protects data loaded during the boot process.

Signal Watchdogs

The eight general-purpose timers feature modes to monitor off-chip signals. The Watchdog Period mode monitors whether external signals toggle with a period within an expected range. The Watchdog Width mode monitors whether the pulse widths of external signals are within an expected range. Both modes help to detect undesired toggling (or lack thereof) of system-level signals.

System Event Controller (SEC)

Besides system events the SEC further supports fault management including fault action configuration as time out, internal indication by system interrupt or external indication via $\overline{\text{SYS\_FAULT}}$ pin, and system reset.

Memory Error Controller (MEC)

TBD.

PROCESSOR PERIPHERALS

The following sections describe the peripherals of the ADSP-SC57x/ADSP-2157x processors.

Dynamic Memory Controller (DMC)

The 16-bit DMC interfaces to:

- LPDDR1 (JESD209A) max freq 200 MHz DDRCLK (64M bit – 2G bit)
- DDR2 (JESD79-2E) max freq 400 MHz DDRCLK (256M bit – 4Gbit)
- DDR3 (JESD79-3E) max freq 450 MHz DDRCLK (512M bit – 8G bit)

See Table 8 for DMC memory map.

Digital Audio Interface (DAI)

The processor supports one DAI unit which is mirrored. The DAI provides the ability to connect various peripherals to any of the DAI pins (DAI0_PIN20–PIN01).

Programs make these connections using the signal routing unit (SRU), shown in Figure 1 on Page 1.

The SRU is a matrix routing unit (or group of multiplexers) that enables the peripherals provided by the DAI to be interconnected under software control. This allows easy use of the DAI associated peripherals for a much wider variety of applications by using a larger set of algorithms than is possible with nonconfigurable signal paths.

The DAI includes the peripherals described in the following sections. DAI pin buffers 20–19 can be used to change the polarity of the input signals. Signals of the peripherals belonging to different DAIs cannot be interconnected, with few exceptions.

The DAI pin buffers may also be used as GPIO pins. DAI input signals allow the triggering of interrupts on the rising edge, the falling edge, or both edges.

Serial Ports (SPORT)

The processors feature 4 synchronous full serial ports that provide an inexpensive interface to a wide variety of digital and mixed-signal peripheral devices such as Analog Devices' AD19xx/ADAU19xx family of audio codecs, ADCs, and DACs. The serial ports are made up of two data lines, a clock, and frame sync. The data lines can be programmed to either transmit or receive and each data line has a dedicated DMA channel.

An individual full SPORT module consists of two independently configurable SPORT halves with identical functionality. Two bidirectional data lines—Primary (0) and Secondary (1) per SPORT half, configurable as either transmitters or receivers. Therefore, each SPORT half can be configured for two transmitter or two receiver channels, permitting two unidirectional streams into or out of the same full SPORT. This bidirectional functionality provides greater flexibility for serial communications. For full-duplex configuration, one half SPORT provides two transmit signals, while the other half SPORT provides the two receive signals. The frame sync and clock are shared. The maximum SCLK is 1024xFS (TX master, RX slave) and 512xFS (TX slave, RX master).

Serial ports operate in six modes:

- Standard DSP serial mode
- Multichannel (TDM) mode
- I^2S mode
- Packed I^2S mode
- Left-justified mode
- Right-justified mode

Asynchronous Sample Rate Converter (ASRC)

The asynchronous sample rate converter (ASRC) contains four ASRC blocks, is the same core as that used in the AD1896 192 kHz stereo asynchronous sample rate converter, and provides up to 140 dB SNR. The ASRC block is used to perform synchronous or asynchronous sample rate conversion across independent stereo channels, without using internal processor resources. The SRC blocks can also be configured to operate together to convert multichannel audio data without phase mismatches. Finally, the ASRC can be used to clean up audio data from jittery clock sources such as the S/PDIF receiver.

S/PDIF-Compatible Digital Audio Receiver/Transmitter

The Sony/Philips Digital Interface (S/PDIF) is a standard audio data transfer format that allows the transfer of digital audio signals from one device to another without having to convert them to an analog signal. There is one S/PDIF Tx/Rx block on the processor. The digital audio interface carries three types of information; audio data, non audio data (compressed data) and timing information.

The S/PDIF interface supports one stereo channel or compressed audio streams. The S/PDIF transmitter and receiver is AES3-compliant. The S/PDIF receiver supports professional jitter standard.

The S/PDIF receiver/transmitter has no separate DMA channels. It receives audio data in serial format and converts it into a biphase encoded signal. The serial data input to the receiver/transmitter can be formatted as left-justified, I^2S or right-justified with word widths of 16, 18, 20, or 24 bits. The serial data, clock, and frame sync inputs to the S/PDIF receiver/transmitter are routed through the signal routing unit (SRU). They can come from a variety of sources, such as the SPORTs, external pins, and the precision clock generators (PCGs), and are controlled by the SRU control registers.

Precision Clock Generators (PCG)

The precision clock generators (PCG) consist of two units located in the DAI unit. The PCG can generate a pair of signals (clock and frame sync) derived from a clock input signal (CLKIN, SCLK0, DAI pin buffer). Both units are identical in functionality and operate independently of each other. The two signals generated by each unit are normally used as a serial bit clock/frame sync pair.

Parallel Peripheral Interface (PPI)

The processors provide a parallel peripheral interface (PPI) that supports data widths up to 16 bits for BGA and 8 bits for LQFP. The PPI supports direct connection to TFT LCD panels, parallel analog-to-digital and digital-to-analog converters, video encoders and decoders, image sensor modules and other general-purpose peripherals.

The following features are supported in the PPI module:

- Programmable data length: 8 bits, 10 bits, 12 bits, 14 bits, and 16 bits per clock.
- Various framed, non-framed, and general-purpose operating modes. Frame syncs can be generated internally or can be supplied by an external device.
- ITU-656 status word error detection and correction for ITU-656 receive modes and ITU-656 preamble and status word decode.
- Optional packing and unpacking of data to/from 32 bits from/to 8 bits, and 16 bits. If packing/unpacking is enabled, endianness can be configured to change the order of packing/unpacking of bytes/words.
- RGB888 can be converted to RGB666 or RGB565 for transmit modes.
- Various de-interleaving/interleaving modes for receiving/transmitting 4:2:2 YCrCb data.
- Configurable LCD data enable (DEN) output available on Frame Sync 3.

UART Ports

The processors provide three full-duplex universal asynchronous receiver/transmitter (UART) ports, which are fully compatible with PC-standard UARTs. Each UART port provides a simplified UART interface to other peripherals or hosts, supporting full-duplex, DMA-supported, asynchronous transfers of serial data. A UART port includes support for five to eight data bits, and none, even, or odd parity. Optionally, an additional address bit can be transferred to interrupt only addressed nodes in multi-drop bus (MDB) systems. A frame is terminated by a configurable number of stop bits.

The UART ports support automatic hardware flow control through the Clear To Send (CTS) input and Request To Send (RTS) output with programmable assertion FIFO levels.

To help support the Local Interconnect Network (LIN) protocols, a special command causes the transmitter to queue a break command of programmable bit length into the transmit buffer. Similarly, the number of stop bits can be extended by a programmable inter-frame space.

Serial Peripheral Interface (SPI) Ports

The processors have three industry-standard SPI-compatible ports that allow it to communicate with multiple SPI-compatible devices.

The baseline SPI peripheral is a synchronous, four-wire interface consisting of two data pins, one device select pin, and a gated clock pin. The two data pins allow full-duplex operation to other SPI-compatible devices. An additional two (optional) data pins are provided to support quad SPI operation. Enhanced modes of operation such as flow control, fast mode and dual-I/O mode (DIOM) are also supported. In addition, a direct memory access (DMA) mode allows for transferring several words with minimal CPU interaction.

With a range of configurable options, the SPI ports provide a glueless hardware interface with other SPI-compatible devices in master mode, slave mode, and multimaster environments. The SPI peripheral includes programmable baud rates, clock phase, and clock polarity. The peripheral can operate in a multimaster environment by interfacing with several other devices, acting as either a master device or a slave device. In a multimaster environment, the SPI peripheral uses open-drain outputs to avoid data bus contention. The flow control features enable slow slave devices to interface with fast master devices by providing an SPI Ready pin which flexibly controls the transfers.

The SPI port's baud rate and clock phase/polarities are programmable, and it has integrated DMA channels for both transmit and receive data streams.

Link Ports (LP)

Two 8-bit wide link ports can connect to the link ports of other DSPs or peripherals. Link ports are bidirectional ports having eight data lines, an acknowledge line, and a clock line.

ADC Control Module (ACM) Interface

The ADC control module (ACM) provides an interface that synchronizes the controls between the processors and an analog-to-digital converter (ADC). The analog-to-digital conversions are initiated by the processors, based on external or internal events.

The ACM allows for flexible scheduling of sampling instants and provides precise sampling signals to the ADC.

The ACM synchronizes the ADC conversion process, generating the ADC controls, the ADC conversion start signal, and other signals. The actual data acquisition from the ADC is done by an internal DAI routing of ACM with SPORT0.

The processors interface directly to many ADCs without any glue logic required.

Ethernet MAC (EMAC)

The processor features an EMAC—10/100/1000 Std/AVB Ethernet with precision time protocol IEEE-1588.

The processors can directly connect to a network by way of an embedded fast Ethernet media access controller (MAC) that supports 10-BaseT (10M bits/sec), 100-BaseT (100M bits/sec) and 1000-BaseT (1G bits/sec) operations.

Some standard features are:

- Support and MII/RMII/RGMII protocols for external PHYs. RGMII is supported only for BGA package
- Full duplex and half duplex modes
- Media access management (in half-duplex operation)
- Flow control
- Station management: generation of MDC/MDIO frames for read-write access to PHY registers

Some advanced features are:

- Automatic checksum computation of IP header and IP payload fields of RX frames
- Independent 32-bit descriptor-driven receive and transmit DMA channels
- Frame status delivery to memory through DMA, including frame completion semaphores for efficient buffer queue management in software
- TX DMA support for separate descriptors for MAC header and payload to eliminate buffer copy operations
- Convenient frame alignment modes
- 47 MAC management statistics counters with selectable clear-on-read behavior and programmable interrupts on half maximum value
- Advanced power management
- Magic packet detection and wakeup frame filtering
- Support for 802.3Q tagged VLAN frames
- Programmable MDC clock rate and preamble suppression

Audio Video Bridging (AVB) Support

The 10/100/1000 EMAC supports the following audio video (AVB) features:

- Separate channels or queues for AV data transfer in 100 Mbps and 1000 Mbps modes)
- IEEE 802.1-Qav specified credit-based shaper (CBS) algorithm for the additional transmit channels
- Configuring up to two additional channels (Channel 1 and Channel 2) on the transmit and receive paths for AV traffic. Channel 0 is available by default and carries the legacy best-effort Ethernet traffic on the transmit side
- Separate DMA, Tx FIFO, and Rx FIFO for AVB latency class
- Programmable control to route received VLAN tagged non-AV packets to channels or queues

Precision Time Protocol IEEE 1588 Support

The IEEE 1588 standard is a precision clock synchronization protocol for networked measurement and control systems. The processors include hardware support for IEEE 1588 with an integrated precision time protocol synchronization engine (PTP_TSYNC). This engine provides hardware assisted time stamping to improve the accuracy of clock synchronization between PTP nodes. The main features of the engine are:

- Support for both IEEE 1588-2002 and IEEE 1588-2008 protocol standards
- Hardware assisted time stamping capable of up to 12.5 ns resolution
- Lock adjustment
- Automatic detection of IPv4 and IPv6 packets, as well as PTP messages
- Multiple input clock sources (SCLK0, RGMII, RMII, MII clock, external clock)
- Programmable pulse per second (PPS) output
- Auxiliary snapshot to time stamp external events

Controller Area Network (CAN)

There are two CAN modules. A CAN controller implements the CAN 2.0B (active) protocol. This protocol is an asynchronous communications protocol used in both industrial and automotive control systems. The CAN protocol is well suited for control applications due to its capability to communicate reliably over a network. This is because the protocol incorporates CRC checking, message error tracking, and fault node confinement.

The CAN controller offers the following features:

- 32 mailboxes (8 receive only, 8 transmit only, 16 configurable for receive or transmit).
- Dedicated acceptance masks for each mailbox.
- Additional data filtering on first two bytes.
- Support for both the standard (11-bit) and extended (29-bit) identifier (ID) message formats.
- Support for remote frames.
- Active or passive network support.
- Interrupts, including: TX complete, RX complete, error and global.

An additional crystal is not required to supply the CAN clock, as the CAN clock is derived from a system clock through a programmable divider.

Timers

The processors include several timers which are described in the following sections.

General-Purpose Timers (TIMER)

There is one GP timer unit, and it provides eight general-purpose programmable timers. Each timer has an external pin that can be configured either as a pulse width modulator (PWM) or timer output, as an input to clock the timer, or as a mechanism for measuring pulse widths and periods of external events. These timers can be synchronized to an external clock input on the TIMER_TMRx pins, an external TIMER_CLK input pin, or to the internal SCLK0.

These timer units can be used in conjunction with the UARTs and the CAN controller to measure the width of the pulses in the data stream to provide a software auto-baud detect function for the respective serial channels.

The GP timers can generate interrupts to the processor core, providing periodic events for synchronization to either the system clock or to external signals. Timer events can also trigger other peripherals via the TRU (for instance, to signal a fault). Each timer may also be started and/or stopped by any TRU master without core intervention.

Watchdog Timer (WDT)

Three on-chip software watchdog timers can be used by the ARM and/or SHARC+ cores. A software watchdog can improve system availability by forcing the processors to a known state, via a general-purpose interrupt, or a fault, if the timer expires before being reset by software. The programmer initializes the count value of the timer, enables the appropriate interrupt, then enables the timer. Thereafter, the software must reload the counter before it counts down to zero from the programmed value. This protects the system from remaining in an unknown state where software that would normally reset the timer has stopped running due to an external noise condition or software error.

General-Purpose Counters (CNT)

A 32-bit counter is provided that can operate in general-purpose up/down count modes and can sense 2-bit quadrature or binary codes as typically emitted by industrial drives or manual thumbwheels. Count direction is either controlled by a level-sensitive input pin or by two edge detectors.

A third counter input can provide flexible zero marker support and can alternatively be used to input the push-button signal of thumbwheel devices. All three pins have a programmable debouncing circuit.

Internal signals forwarded to a GP timer enable this timer to measure the intervals between count events. Boundary registers enable auto-zero operation or simple system warning by interrupts when programmed count values are exceeded.

Housekeeping Analog-to-Digital Converter (HADC)

The HADC provides a general-purpose, multi-channel successive approximation A-to-D converter. It supports the following set of features:

- 12-bit ADC core (10-bit accuracy) with built in sample and hold
- 8 single-ended input channels for BGA; 4 single-ended input channels for LQFP
- Throughput rates up to 1 MSPS
- Single external reference with analog inputs between 0 and 3.3 V
- Selectable ADC clock frequency including the ability to program a pre-scaler
- Adaptable conversion type: allows single or continuous conversion with option of autoscan
- Auto sequencing capability with up to 8 “autoconversions” in a single session. Each conversion can be programmed to select any 1–8 input channels.
- 16 data registers (individually addressable) to store conversion values

USB 2.0 On-the-Go Dual-Role Device Controller (BGA only)

The USB supports HS/FS/LS USB2.0 OTG.

The USB 2.0 OTG dual-role device controller provides a low-cost connectivity solution for the growing adoption of this bus standard in industrial applications, as well as consumer mobile devices such as cell phones, digital still cameras, and MP3 players. The USB 2.0 controller allows these devices to transfer data using a point-to-point USB connection without the need for a PC host. The module can operate in a traditional USB peripheral-only mode as well as the host mode presented in the On-the-Go (OTG) supplement to the USB 2.0 specification.

The USB clock is provided through a dedicated external crystal or crystal oscillator.

The USB On-the-Go dual-role device controller includes a Phase Locked Loop with programmable multipliers to generate the necessary internal clocking frequency for USB.

Media Local Bus (MLB)

The automotive model has a MLB slave interface which allows the processors to function as a media local bus device. It includes support for both 3-pin and 6-pin media local bus protocols. MLB 3-pin supports speeds up to 1024 × FS and 6-pin up to 4096 × FS (48 kHz). MLB also supports up to 63 logical channels, with up to 124 bytes of data per media local bus frame.

The MLB interface supports MOST25/50/150 data rates and operates in slave mode only.

2-Wire Controller Interface (TWI)

The processors include three 2-wire interface (TWI) modules for providing a simple exchange method of control data between multiple devices. The TWI module is compatible with the widely used I^2C bus standard. The TWI module offers the capabilities of simultaneous master and slave operation and support for both 7-bit addressing and multimedia data arbitration. The TWI interface utilizes two pins for transferring clock (TWI_SCL) and data (TWI_SDA) and supports the protocol at speeds up to 400k bits/sec. The TWI interface pins are compatible with 5 V logic levels.

Additionally, the TWI module is fully compatible with serial camera control bus (SCCB) functionality for easier control of various CMOS camera sensor devices.

General-Purpose I/O (GPIO)

Each general-purpose port pin can be individually controlled by manipulation of the port control, status, and interrupt registers:

- GPIO direction control register – Specifies the direction of each individual GPIO pin as input or output.
- GPIO control and status registers – A “write one to modify” mechanism allows any combination of individual GPIO pins to be modified in a single instruction, without affecting the level of any other GPIO pins.

- GPIO interrupt mask registers – Allow each individual GPIO pin to function as an interrupt to the processors. GPIO pins defined as inputs can be configured to generate hardware interrupts, while output pins can be triggered by software interrupts.
- GPIO interrupt sensitivity registers – Specify whether individual pins are level- or edge-sensitive and specify—if edge-sensitive—whether just the rising edge or both the rising and falling edges of the signal are significant.

Pin Interrupts

Every port pin on the processors can request interrupts in either an edge-sensitive or a level-sensitive manner with programmable polarity. Interrupt functionality is decoupled from GPIO operation. Five system-level interrupt channels (PINT0–4) are reserved for this purpose. Each of these interrupt channels can manage up to 32 interrupt pins. The assignment from pin to interrupt is not performed on a pin-by-pin basis. Rather, groups of eight pins (half ports) can be flexibly assigned to interrupt channels.

Every pin interrupt channel features a special set of 32-bit memory-mapped registers that enable half-port assignment and interrupt management. This includes masking, identification, and clearing of requests. These registers also enable access to the respective pin states and use of the interrupt latches, regardless of whether the interrupt is masked or not. Most control registers feature multiple MMR address entries to write-one-to-set or write-one-to-clear them individually.

Mobile Storage Interface (MSI)

The mobile storage interface (MSI) controller acts as the host interface for multimedia cards (MMC), secure digital memory cards (SD), and secure digital input/output cards (SDIO). The MSI controller has the following features:

- Support for a single MMC, SD memory, SDIO card
- Support for 1-bit and 4-bit SD modes
- Support for 1-bit, 4-bit, and 8-bit MMC modes
- Support for eMMC 4.3 embedded NAND flash devices
- An 11-signal external interface with clock, command, optional interrupt, and up to 8 data lines
- Integrated DMA controller
- Card interface clock generation from CLKO9 from CDU
- SDIO interrupt and read wait features

SYSTEM ACCELERATION

The following sections describe the system acceleration blocks of the ADSP-SC57x/ADSP-2157x processors.

FIR Accelerator

The FIR (finite impulse response) accelerator consists of a 1024 word coefficient memory, a 1024 word deep delay line for the data, and four MAC units. A controller manages the accelerator. The FIR accelerator runs at the peripheral clock frequency. It can access all memory spaces and can run simultaneously with other accelerators.

IIR Accelerator

The IIR (infinite impulse response) accelerator consists of a 1440 word coefficient memory for storage of biquad coefficients, a data memory for storing the intermediate data, and one MAC unit. A controller manages the accelerator. The IIR accelerator runs at the peripheral clock frequency. It can access all memory spaces and can run simultaneously with other accelerators.

SYSTEM DESIGN

The following sections provide an introduction to system design options and power supply issues.

Clock Management

The processors provide three operating modes, each with a different performance/power profile. Control of clocking to each of the processor peripherals also reduces power consumption. The processors do not support any low power operation modes. Control of clocking to each of the processor peripherals can reduce the power consumption.

Reset Control Unit (RCU)

Reset is the initial state of the whole processor or the core and is the result of a hardware- or software-triggered event. In this state, all control registers are set to their default values and functional units are idle. Exiting a full system reset starts with the core being ready to boot.

The reset control unit (RCU) controls how all the functional units enter and exit reset. Differences in functional requirements and clocking constraints define how reset signals are generated. Programs must guarantee that none of the reset functions puts the system into an undefined state or causes resources to stall. This is particularly important when the core is reset (programs must ensure that there is no pending system activity involving the core when it is being reset).

From a system perspective reset is defined by both the reset target and the reset source as described below.

Target defined:

- System Reset – All functional units except the RCU are set to their default states.
- Hardware Reset – All functional units are set to their default states without exception. History is lost.
- Core-only Reset – Affects the core only. The system software should guarantee that the core, while in reset state, is not accessed by any bus master.

Source defined:

- System Reset – May be triggered by software (writing to the RCU_CTL register) or by another functional unit such as the dynamic power management (DPM) unit or any of the system event controller (SEC), trigger routing unit (TRU), or emulator inputs.

- Hardware Reset – The $\overline{\text{SYS\_HWRST}}$ input signal is asserted active (pulled down).
- Core-only Reset – Triggered by software.
- Trigger request (peripheral).

Clock Generation Unit (CGU)

The ADSP-SC57x/ADSP-2157x processors support two independent PLLs. Each PLL is part of a clock generation unit (CGU). (Refer to Figure 7 on Page 59.) Each CGU can either be driven externally by the same clock source or each can be driven by separate sources. This provides flexibility in determining the internal clocking frequencies for each clock domain.

Frequencies generated by each CGU are derived from a common multiplier with different divider values available for each output.

The clock generation unit (CGU) generates all on-chip clocks and synchronization signals. Multiplication factors are programmed to define the PLLCLK frequency.

Programmable values divide the PLLCLK frequency to generate the core clock (CCLK), the system clocks, the DDR1/2/3 clock (DCLK), and the output clock (OCLK). For more information on clocking, refer to the Clock Generation Unit (CGU) chapter of the hardware reference manual.

Writing to the CGU control registers does not affect the behavior of the PLL immediately. Registers are first programmed with a new value, and the PLL logic executes the changes so that it transitions smoothly from the current conditions to the new conditions.

System Crystal Oscillator and USB Crystal Oscillator

The processor can be clocked by an external crystal, (Figure 6) a sine wave input, or a buffered, shaped clock derived from an external clock oscillator. If an external clock is used, it should be a TTL compatible signal and must not be halted, changed, or operated below the specified frequency during normal operation. This signal is connected to the SYS_CLKIN pin of the processor. When an external clock is used, the SYS_XTAL pin must be left unconnected. Alternatively, because the processor includes an on-chip oscillator circuit, an external crystal may be used.

For fundamental frequency operation, use the circuit shown in Figure 6. A parallel-resonant, fundamental frequency, microprocessor grade crystal is connected across the SYS_CLKIN and SYS_XTAL pins. The on-chip resistance between SYS_CLKIN and the SYS_XTAL pin is in the 500 kΩ range. Further parallel resistors are typically not recommended.

The two capacitors and the series resistor shown in Figure 6 fine-tune phase and amplitude of the sine frequency. The capacitor and resistor values shown in Figure 6 are typical values only. The capacitor values are dependent upon the load capacitance recommendations of the crystal manufacturer and the PCB physical layout. The resistor value depends on the drive level specified by the crystal manufacturer. The user should verify the customized values based on careful investigations on multiple devices over the required temperature range.

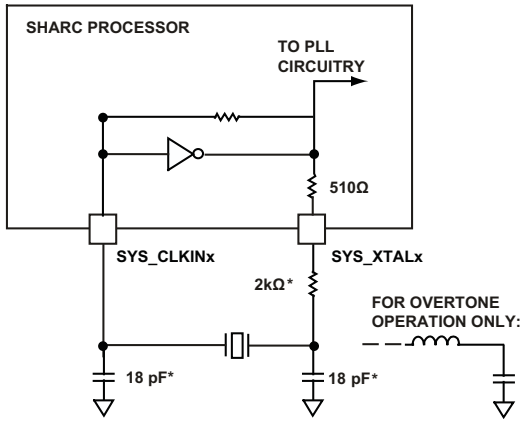


Figure 6. External Crystal Connection

A third-overtone crystal can be used for frequencies above 25 MHz. The circuit is then modified to ensure crystal operation only at the third overtone by adding a tuned inductor circuit as shown in Figure 6. A design procedure for third-overtone operation is discussed in detail in application note (EE-168) "Using Third Overtone Crystals with the ADSP-218x DSP" (www.analog.com/ee-168). The same recommendations may be used for the USB crystal oscillator.

Clock Distribution Unit (CDU)

The two CGUs each provide outputs which feed a clock distribution unit (CDU). The clock outputs CLKO0-9 are connected to various targets. For more information, refer to the Clock Distribution Unit (CDU) chapter of the hardware reference manual.

Power-Up

XTAL oscillations (SYS_CLKIN) start when power is applied to the V_{DD_EXT} pins. The rising edge of $\overline{\text{SYS\_HWRST}}$ starts on-chip PLL locking (PLL lock counter). The deassertion should apply only if all voltage supplies and SYS_CLKIN oscillations are valid (refer to Power-Up Reset Timing on Page 65).

Clock Out/External Clock

The SYS_CLKOUT output pin has programmable options to output divided-down versions of the on-chip clocks. By default, the SYS_CLKOUT pin drives a buffered version of the SYS_ CLKIN0 input. Refer to the Clock Distribution Unit (CDU) chapter in the hardware reference manual to change the default mapping of clocks.

Booting

The processors have several mechanisms for automatically loading internal and external memory after a reset. The boot mode is defined by the SYS_BMODE input pins dedicated for this purpose. There are two categories of boot modes. In master boot

mode, the processor actively loads data from serial memories. In slave boot modes, the processors receive data from external host devices.

The boot modes are shown in Table 9. These modes are implemented by the SYS_BMODE bits of the reset configuration register and are sampled during power-on resets and software-initiated resets.

In the ADSP-SC57x processors, the ARM Cortex-A5 (core 0) controls the boot process, including loading all internal and external memory. Likewise, in the ADSP-2157x processors, the SHARC+ (core 1) controls the boot function. The option for secure boot is available on all products.

Table 9. Boot Modes

SYS_BMODE Setting[1, 2]	Boot Mode
000	No boot
001	SPI2 Master
010	SPI2 Slave
011	UART0 Slave
100	Reserved
101	Reserved
110	Link0 Slave

[1] BMODE2 pin is applicable only for BGA package.
[2] Link0 slave boot is supported only on BGA package.

Thermal Monitoring Unit (TMU)

The thermal monitoring unit provides on-chip temp measurement which is important in applications that require substantial power consumption. The TMU is integrated into the processor die and digital infrastructure using an MMR-based system access to measure the die temperature variations in real-time.

TMU features include:

- On-chip temperature sensing
- Programmable over-temperature and under-temperature limits
- Programmable conversion rate
- Programmable clock source selection to run the sensor off an independent local clock
- Averaging feature available

Power Supplies

The processors have separate power supply connections for:

- Internal (V_{DD_INT})
- External (V_{DD_EXT})
- USB (V_{DD_USB})
- HADC (V_{DD_HADC})
- DMC (V_{DD_DMC})

All the power supplies should meet the specifications provided in Operating Conditions on Page 57. All the external supply pins must be connected to the same power supply.

Power Management

As shown in Table 10, the processor supports four different power domains, which maximizes flexibility while maintaining compliance with industry standards and conventions. There are no sequencing requirements for the various power domains, but all domains must be powered according to the appropriate Specifications table for processor operating conditions; even if the feature/peripheral is not used.

Table 10. Power Domains

Power Domain	V_{DD} Range
All internal logic	V_{DD_INT}
DDR3/DDR2/LPDDR	V_{DD_DMC}
USB	V_{DD_USB}
HADC	V_{DD_HADC}
All other I/O (includes SYS, JTAG, and Ports pins)	V_{DD_EXT}

The power dissipated by a processor is largely a function of its clock frequency and the square of the operating voltage. For example, reducing the clock frequency by 25% results in a 25% reduction in dynamic power dissipation.

Target Board JTAG Emulator Connector

Analog Devices DSP Tools product line of JTAG emulators uses the IEEE 1149.1 JTAG test access port of the processor to monitor and control the target board processor during emulation. Analog Devices DSP Tools product line of JTAG emulators provides emulation at full processor speed, allowing inspection and modification of memory, registers, and processor stacks. The processor's JTAG interface ensures that the emulator will not affect target system loading or timing.

For complete information on Analog Devices' SHARC DSP Tools product line of JTAG emulator operation, see the appropriate Emulator Hardware User's Guide.

SYSTEM DEBUG

The processors include various features that allow for easy system debug. These are described in the following sections.

System Watchpoint Unit (SWU)

The system watchpoint unit (SWU) is a single module which connects to a single system bus and provides for transaction monitoring. One SWU is attached to the bus going to each system slave. The SWU provides ports for all system bus address channel signals. Each SWU contains four match groups of registers with associated hardware. These four SWU match groups operate independently, but share common event (interrupt, trigger and others) outputs.

Debug Access Port (DAP)

DAP (debug access port) provides IEEE-1149.1 JTAG interface support through its JTAG debug. The DAP provides an optional instrumentation trace for both the core and system. It provides a trace stream that conforms to MIPI System Trace Protocol version 2(STPv2).

DEVELOPMENT TOOLS

Analog Devices supports its processors with a complete line of software and hardware development tools, including an integrated development environment (CrossCore® Embedded Studio), evaluation products, emulators, and a wide variety of software add-ins.

Integrated Development Environments (IDEs)

For C/C++ software writing and editing, code generation, and debug support, Analog Devices offers the CrossCore Embedded Studio IDE.

CrossCore Embedded Studio is based on the Eclipse™ framework. Supporting most Analog Devices processor families, it is the IDE of choice for processors, including multicore devices. CrossCore Embedded Studio seamlessly integrates available software add-ins to support real time operating systems, file systems, TCP/IP stacks, USB stacks, algorithmic software modules, and evaluation hardware board support packages. For more information visit www.analog.com/cces.

EZ-KIT Lite Evaluation Board

For processor evaluation, Analog Devices provides wide range of EZ-KIT Lite® evaluation boards. Including the processor and key peripherals, the evaluation board also supports on-chip emulation capabilities and other evaluation and development features. Also available are various EZ-Extenders®, which are daughter cards delivering additional specialized functionality, including audio and video processing. For more information visit www.analog.com and search on “ezkit” or “ezextender”.

EZ-KIT Lite Evaluation Kits

For a cost-effective way to learn more about developing with Analog Devices processors, Analog Devices offer a range of EZ-KIT Lite evaluation kits. Each evaluation kit includes an EZ-KIT Lite evaluation board, directions for downloading an evaluation version of the available IDE(s), a USB cable, and a power supply. The USB controller on the EZ-KIT Lite board connects to the USB port of the user’s PC, enabling the chosen IDE evaluation suite to emulate the on-board processor in-circuit. This permits the customer to download, execute, and debug programs for the EZ-KIT Lite system. It also supports in-circuit programming of the on-board Flash device to store user-specific boot code, enabling standalone operation. With the full version of CrossCore Embedded Studio installed (sold separately), engineers can develop software for supported EZ-KITs or any custom system utilizing supported Analog Devices processors.

Software Add-Ins for CrossCore Embedded Studio

Analog Devices offers software add-ins which seamlessly integrate with CrossCore Embedded Studio to extend its capabilities and reduce development time. Add-ins include board support packages for evaluation hardware, various middleware packages, and algorithmic modules. Documentation, help, configuration dialogs, and coding examples present in these add-ins are viewable through the CrossCore Embedded Studio IDE once the add-in is installed.

Board Support Packages for Evaluation Hardware

Software support for the EZ-KIT Lite evaluation boards and EZ-Extender daughter cards is provided by software add-ins called Board Support Packages (BSPs). The BSPs contain the required drivers, pertinent release notes, and select example code for the given evaluation hardware. A download link for a specific BSP is located on the web page for the associated EZ-KIT or EZ-Extender product. The link is found in the *Product Download* area of the product web page.

Middleware Packages

Analog Devices separately offers middleware add-ins such as real time operating systems, file systems, USB stacks, and TCP/IP stacks. For more information see the following web pages:

- www.analog.com/ucos2
- www.analog.com/ucos3
- www.analog.com/ucfs
- www.analog.com/ucusbd
- www.analog.com/ucusbh
- www.analog.com/lwip

Algorithmic Modules

To speed development, Analog Devices offers add-ins that perform popular audio and video processing algorithms. These are available for use with CrossCore Embedded Studio. For more information visit www.analog.com and search on “Blackfin software modules” or “SHARC software modules”.

Designing an Emulator-Compatible DSP Board (Target)

For embedded system test and debug, Analog Devices provides a family of emulators. On each JTAG DSP, Analog Devices supplies an IEEE 1149.1 JTAG Test Access Port (TAP). In-circuit emulation is facilitated by use of this JTAG interface. The emulator accesses the processor’s internal features via the processor’s TAP, allowing the developer to load code, set breakpoints, and view variables, memory, and registers. The processor must be halted to send data and commands, but once an operation is completed by the emulator, the DSP system is set to run at full speed with no impact on system timing. The emulators require the target board to include a header that supports connection of the DSP’s JTAG port to the emulator.

For details on target board design issues including mechanical layout, single processor connections, signal buffering, signal termination, and emulator pod logic, see (EE-68) “Analog Devices

JTAG Emulation Technical Reference" (www.analog.com/ee-68)This document is updated regularly to keep pace with improvements to emulator support.

ADDITIONAL INFORMATION

This data sheet provides a general overview of the ADSP-SC57x/ADSP-2157x architecture and functionality. For detailed information on the core architecture and instruction set, refer to the programming reference manual.

RELATED SIGNAL CHAINS

A signal chain is a series of signal-conditioning electronic components that receive input (data acquired from sampling either real-time phenomena or from stored data) in tandem, with the output of one portion of the chain supplying input to the next. Signal chains are often used in signal processing applications to gather and process data or to apply system controls based on analysis of real-time phenomena.

Analog Devices eases signal processing system development by providing signal processing components that are designed to work together well. A tool for viewing relationships between specific applications and related components is available on the www.analog.com website.

The application signal chains page in the Circuits from the Lab® site (www.analog.com\circuits) provides:

- Graphical circuit block diagram presentation of signal chains for a variety of circuit types and applications
- Drill down links for components in each chain to selection guides and application information
- Reference designs applying best practice design techniques

SECURITY FEATURES DISCLAIMER

To our knowledge, the Security Features, when used in accordance with the data sheet and hardware reference manual specifications, provide a secure method of implementing code and data safeguards. However, Analog Devices does not guarantee that this technology provides absolute security. ACCORDINGLY, ANALOG DEVICES HEREBY DISCLAIMS ANY AND ALL EXPRESS AND IMPLIED WARRANTIES THAT THE SECURITY FEATURES CANNOT BE BREACHED, COMPROMISED, OR OTHERWISE CIRCUMVENTED AND IN NO EVENT SHALL ANALOG DEVICES BE LIABLE FOR ANY LOSS, DAMAGE, DESTRUCTION, OR RELEASE OF DATA, INFORMATION, PHYSICAL PROPERTY, OR INTELLECTUAL PROPERTY.

ADSP-SC57X/ADSP-2157X DETAILED SIGNAL DESCRIPTIONS

Table 11 provides a detailed description of each pin.

Table 11. ADSP-SC57x/ADSP-2157x Detailed Signal Descriptions

Signal Name	Direction	Description
ACM_A[n]	Output	**ADC Control Signals.** Function varies by mode
ACM_T[n]	Input	**External Trigger n.** Input for external trigger events
C1_FLG[n]	Output	**SHARC Core 1 Flag Pin**
C2_FLG[n]	Output	**SHARC Core 2 Flag Pin**
CAN_RX	Input	**Receive.** Typically an external CAN transceiver's RX output
CAN_TX	Output	**Transmit.** Typically an external CAN transceiver's TX input
CNT_DG	Input	**Count Down and Gate.** Depending on the mode of operation this input acts either as a count down signal or a gate signal. Countdown: This input causes the GP counter to decrement. Gate: Stops the GP counter from incrementing or decrementing.
CNT_UD	Input	**Count Up and Direction.** Depending on the mode of operation this input acts either as a count up signal or a direction signal. Count up: This input causes the GP counter to increment. Direction: Selects whether the GP counter is incrementing or decrementing.
CNT_ZM	Input	**Count Zero Marker.** Input that connects to the zero marker output of a rotary device or detects the pressing of a pushbutton
DAI_PIN[nn]	InOut	**Pin n**
DMC_A[nn]	Output	**Address n.** Address bus
DMC_BA[n]	Output	**Bank Address Input n.** Defines which internal bank an ACTIVATE, READ, WRITE, or PRECHARGE command is being applied to on the dynamic memory. Also defines which mode registers (MR, EMR, EMR2, and/or EMR3) are loaded during the LOAD MODE REGISTER command.
$\overline{\text{DMC\_CAS}}$	Output	**Column Address Strobe.** Defines the operation for external dynamic memory to perform in conjunction with other DMC command signals. Connect to the CAS input of dynamic memory.
$\overline{\text{DMC\_CK}}$	Output	**Clock.** Outputs DCLK to external dynamic memory
DMC_CKE	Output	**Clock enable.** Active high clock enables. Connects to the dynamic memory's CKE input.
$\overline{\text{DMC\_CS[n]}}$	Output	**Chip Select n.** Commands are recognized by the memory only when this signal is asserted.
DMC_DQ[nn]	InOut	**Data n.** Bidirectional data bus
DMC_LDM	Output	**Data Mask for Lower Byte.** Mask for DMC_DQ07:DMC_DQ00 write data when driven high. Sampled on both edges of the data strobe by the dynamic memory.
$\overline{\text{DMC\_LDQS}}$	InOut	**Data Strobe for Lower Byte.** DMC_DQ07:DMC_DQ00 data strobe. Output with Write Data. Input with Read Data. May be single-ended or differential depending on register settings.
DMC_ODT	Output	**On-die termination.** Enables dynamic memory termination resistances when driven high (assuming the memory is properly configured). ODT is enabled/disabled regardless of read or write commands.
$\overline{\text{DMC\_RAS}}$	Output	**Row Address Strobe.** Defines the operation for external dynamic memory to perform in conjunction with other DMC command signals. Connect to the RAS input of dynamic memory.
$\overline{\text{DMC\_RESET}}$	Output	**Reset (DDR3 only)**
DMC_RZQ	InOut	**External calibration resistor connection**
DMC_UDM	Output	**Data Mask for Upper Byte.** Mask for DMC_DQ15:DMC_DQ08 write data when driven high. Sampled on both edges of the data strobe by the dynamic memory.
$\overline{\text{DMC\_UDQS}}$	InOut	**Data Strobe for Upper Byte.** DMC_DQ15:DMC_DQ08 data strobe. Output with Write Data. Input with Read Data. May be single-ended or differential depending on register settings.
DMC_VREF	Input	**Voltage Reference.** Connects to half of the VDD_DMC voltage
$\overline{\text{DMC\_WE}}$	Output	**Write Enable.** Defines the operation for external dynamic memory to perform in conjunction with other DMC command signals. Connect to the WEb input of dynamic memory.
ETH_COL	Input	**MII Collision detect.** Collision detect input signal valid only in MII

Table 11. ADSP-SC57x/ADSP-2157x Detailed Signal Descriptions (Continued)

Signal Name	Direction	Description
ETH_CRS	Input	**MII Carrier Sense.** Multiplexed on alternate clock cycles. CRS: Asserted by the PHY when either the transmit or receive medium is not idle. De-asserted when both are idle. RXDV: Asserted by the PHY when the data on RXDn is valid.
ETH_MDC	Output	**Management Channel Clock.** Clocks the MDC input of the PHY
ETH_MDIO	InOut	**Management Channel Serial Data.** Bidirectional data bus for PHY control
ETH_PTPAUXIN[n]	Input	**PTP Auxiliary Trigger Input.** Assert this signal to take an auxiliary snapshot of the time and store it in the auxiliary time stamp FIFO
ETH_PTPCLKIN[n]	Input	**PTP Clock Input.** Optional external PTP clock input
ETH_PTPPPS[n]	Output	**PTP Pulse-Per-Second Output.** When the Advanced Time Stamp feature is enabled this signal is asserted based on the PPS mode selected. Otherwise, PTPPPS is asserted every time the seconds counter is incremented.
ETH_RXCLK_REFCLK	InOut	**RXCLK (GigE) or REFCLK (10/100)**
ETH_RXCTL_RXDV	InOut	**RXCTL (GigE) or CRS (10/100)**
ETH_RXD[n]	Input	**Receive Data n.** Receive data bus
ETH_RXERR	Input	**Receive Error**
ETH_TXCLK	Input	**Reference Clock.** Externally supplied Ethernet clock
ETH_TXCTL_TXEN	InOut	**TXCTL (GigE) or TXEN (10/100)**
ETH_TXD[n]	Output	**Transmit Data n.** Transmit data bus
HADC_EOC_DOUT	Output	**End of Conversion / Serial Data Out.** Transitions high for one cycle of the HADC internal clock at the end of every conversion. Alternatively, HADC serial data out can be seen by setting the appropriate bit in HADC_CTL.
HADC_VIN[n]	Input	**Analog Input at channel n.** Analog voltage inputs for digital conversion
HADC_VREFN	Input	**Ground Reference for ADC.** Connect to an external voltage reference that meets data sheet specifications
HADC_VREFP	Input	**External Reference for ADC.** Connect to an external voltage reference that meets data sheet specifications
JTG_TCK	Input	**JTAG Clock.** JTAG test access port clock
JTG_TDI	Input	**JTAG Serial Data In.** JTAG test access port data input
JTG_TDO	Output	**JTAG Serial Data Out.** JTAG test access port data output
JTG_TMS	Input	**JTAG Mode Select.** JTAG test access port mode select
$\overline{\text{JTG\_TRST}}$	Input	**JTAG Reset.** JTAG test access port reset
LP_ACK	InOut	**Acknowledge.** Provides handshaking. When the link port is configured as a receiver, ACK is an output. When the link port is configured as a transmitter, ACK is an input.
LP_CLK	InOut	**Clock.** When the link port is configured as a receiver, CLK is an input. When the link port is configured as a transmitter, CLK is an output.
LP_D[n]	InOut	**Data n.** Data bus. Input when receiving, output when transmitting.
MLB_CLK	InOut	**Single-Ended Clock**
MLB_CLKN	InOut	**Differential Clock (–)**
MLB_CLKOUT	InOut	**Single-Ended Clock Out**
MLB_CLKP	InOut	**Differential Clock (+)**
MLB_DAT	InOut	**Single-Ended Data**
MLB_DATN	InOut	**Differential Data (–)**
MLB_DATP	InOut	**Differential Data (+)**
MLB_SIG	InOut	**Single-Ended Signal**
MLB_SIGN	InOut	**Differential Signal (–)**
MLB_SIGP	InOut	**Differential Signal (+)**
$\overline{\text{MSI\_CD}}$	Input	**Card Detect.** Connects to a pull-up resistor and to the card detect output of an SD socket

Table 11. ADSP-SC57x/ADSP-2157x Detailed Signal Descriptions (Continued)

Signal Name	Direction	Description
MSI_CLK	Output	**Clock.** Clock signal applied to the connected device from the MSI
MSI_CMD	InOut	**Command.** Used to send commands to and receive responses from the connected device
MSI_D[n]	InOut	**Data n.** Bidirectional data bus.
$\overline{\text{MSI\_INT}}$	Input	**eSDIO Interrupt Input.** Used only for eSDIO. Connects to an eSDIO card's interrupt output. An interrupt may be sampled even when the MSI clock to the card is switched off.
PPI_CLK	InOut	**Clock.** Input in external clock mode, output in internal clock mode
PPI_D[nn]	InOut	**Data n.** Bidirectional data bus
PPI_FS1	InOut	**Frame Sync 1 (HSYNC).** Behavior depends on EPPI mode. See the EPPI HRM chapter for more details.
PPI_FS2	InOut	**Frame Sync 2 (VSYNC).** Behavior depends on EPPI mode. See the EPPI HRM chapter for more details.
PPI_FS3	InOut	**Frame Sync 3 (FIELD).** Behavior depends on EPPI mode. See the EPPI HRM chapter for more details.
P_[nn]	InOut	**Position n.** General-purpose input/output. See the GP Ports chapter of the HRM for programming information.
SPI_CLK	InOut	**Clock.** Input in slave mode, output in master mode
SPI_D2	InOut	**Data 2.** Used to transfer serial data in quad mode. Open-drain when ODM mode is enabled.
SPI_D3	InOut	**Data 3.** Used to transfer serial data in quad mode. Open-drain when ODM mode is enabled.
SPI_MISO	InOut	**Master In, Slave Out.** Used to transfer serial data. Operates in the same direction as SPI_MOSI in dual and quad modes. Open-drain when ODM mode is enabled.
SPI_MOSI	InOut	**Master Out, Slave In.** Used to transfer serial data. Operates in the same direction as SPI_MISO in dual and quad modes. Open-drain when ODM mode is enabled.
SPI_RDY	InOut	**Ready.** Optional flow signal. Output in slave mode, input in master mode.
$\overline{\text{SPI\_SEL[n]}}$	Output	**Slave Select Output n.** Used in master mode to enable the desired slave
$\overline{\text{SPI\_SS}}$	Input	**Slave Select Input.** Slave mode: Acts as the slave select input. Master mode: Optionally serves as an error detection input for the SPI when there are multiple masters.
SPT_ACLK	InOut	**Channel A Clock.** Data and frame sync are driven/sampled with respect to this clock. This signal can be either internally or externally generated.
SPT_AD0	InOut	**Channel A Data 0.** Primary bidirectional data I/O. This signal can be configured as an output to transmit serial data, or as an input to receive serial data.
SPT_AD1	InOut	**Channel A Data 1.** Secondary bidirectional data I/O. This signal can be configured as an output to transmit serial data, or as an input to receive serial data.
SPT_AFS	InOut	**Channel A Frame Sync.** The frame sync pulse initiates shifting of serial data. This signal is either generated internally or externally.
SPT_ATDV	Output	**Channel A Transmit Data Valid.** This signal is optional and only active when SPORT is configured in multi-channel transmit mode. It is asserted during enabled slots.
SPT_BCLK	InOut	**Channel B Clock.** Data and frame sync are driven/sampled with respect to this clock. This signal can be either internally or externally generated.
SPT_BD0	InOut	**Channel B Data 0.** Primary bidirectional data I/O. This signal can be configured as an output to transmit serial data, or as an input to receive serial data.
SPT_BD1	InOut	**Channel B Data 1.** Secondary bidirectional data I/O. This signal can be configured as an output to transmit serial data, or as an input to receive serial data.
SPT_BFS	InOut	**Channel B Frame Sync.** The frame sync pulse initiates shifting of serial data. This signal is either generated internally or externally.
SPT_BTDV	Output	**Channel B Transmit Data Valid.** This signal is optional and only active when SPORT is configured in multi-channel transmit mode. It is asserted during enabled slots.
SYS_BMODE[n]	Input	**Boot Mode Control n.** Selects the boot mode of the processor
SYS_CLKIN0	Input	**Clock/Crystal Input**
SYS_CLKIN1	Input	**Clock/Crystal Input**
SYS_CLKOUT	Output	**Processor Clock Output.** Outputs internal clocks. Clocks may be divided down. See the CGU chapter of the HRM for more details.

Table 11. ADSP-SC57x/ADSP-2157x Detailed Signal Descriptions (Continued)

Signal Name	Direction	Description
SYS_FAULT	InOut	**Active-High Fault Output.** Indicates internal faults or senses external faults depending on the operating mode
$\overline{\text{SYS\_FAULT}}$	InOut	**Active-Low Fault Output.** Indicates internal faults or senses external faults depending on the operating mode
$\overline{\text{SYS\_HWRST}}$	Input	**Processor Hardware Reset Control.** Resets the device when asserted
$\overline{\text{SYS\_RESOUT}}$	Output	**Reset Output.** Indicates the device is in the reset state
SYS_XTAL0	Output	**Crystal Output**
SYS_XTAL1	Output	**Crystal Output**
TM_ACI[n]	Input	**Alternate Capture Input n.** Provides an additional input for WIDCAP, WATCHDOG, and PININT modes
TM_ACLK[n]	Input	**Alternate Clock n.** Provides an additional time base for use by an individual timer
TM_CLK	Input	**Clock.** Provides an additional global time base for use by all the GP timers
TM_TMR[n]	InOut	**Timer n.** Main input/output signal for each timer
TRACE_CLK	Output	**Trace Clock.** Clock output
TRACE_D[nn]	Output	**Trace Data n.** Unidirectional data bus
TWI_SCL	InOut	**Serial Clock.** Clock output when master, clock input when slave
TWI_SDA	InOut	**Serial Data.** Receives or transmits data
$\overline{\text{UART\_CTS}}$	Input	**Clear to Send.** Flow control signal
$\overline{\text{UART\_RTS}}$	Output	**Request to Send.** Flow control signal
$\overline{\text{UART\_RX}}$	Input	**Receive.** Receive input. Typically connects to a transceiver that meets the electrical requirements of the device being communicated with.
$\overline{\text{UART\_TX}}$	Output	**Transmit.** Transmit output. Typically connects to a transceiver that meets the electrical requirements of the device being communicated with.
USB_CLKIN	Input	**Clock/Crystal Input.** This clock input is multiplied by a PLL to form the USB clock. See data sheet specifications for frequency/tolerance information.
USB_DM	InOut	**Data –.** Bidirectional differential data line
USB_DP	InOut	**Data +.** Bidirectional differential data line
USB_ID	Input	**OTG ID.** Senses whether the controller is a host or device. This signal is pulled low when an A-type plug is sensed (signifying that the USB controller is the A device), but the input is high when a B-type plug is sensed (signifying that the USB controller is the B device).
USB_VBC	Output	**VBUS Control.** Controls an external voltage source to supply VBUS when in host mode. May be configured as open-drain. Polarity is configurable as well.
USB_VBUS	InOut	**Bus Voltage.** Connects to bus voltage in host and device modes
USB_XTAL	Output	**Crystal.** Drives an external crystal. Must be left unconnected if an external clock is driving USB_CLKIN.

400-BALL CSP_BGA SIGNAL DESCRIPTIONS

The processor's pin definitions are shown in Table 12. The columns in this table provide the following information:

- Signal Name: The Signal Name column in the table includes the signal name for every pin and (where applicable) the GPIO multiplexed pin function for every pin.
- Description: The Description column in the table provides a verbose (descriptive) name for the signal.
- General-Purpose Port: The Port column in the table shows whether or not the signal is multiplexed with other signals on a general-purpose I/O port pin.
- Pin Name: The Pin Name column in the table identifies the name of the package pin (at power on reset) on which the signal is located (if a single function pin) or is multiplexed (if a general-purpose I/O pin).

Table 12. ADSP-SC57x/ADSP-2157x 400-Ball CSP_BGA Signal Descriptions

Signal Name	Description	Port	Pin Name
ACM0_A0	ACM0 ADC Control Signals	F	PF_11
ACM0_A1	ACM0 ADC Control Signals	C	PC_14
ACM0_A2	ACM0 ADC Control Signals	C	PC_15
ACM0_A3	ACM0 ADC Control Signals	A	PA_14
ACM0_A4	ACM0 ADC Control Signals	B	PB_01
ACM0_T0	ACM0 External Trigger n	A	PA_15
C1_FLG0	SHARC Core 1 Flag Pin	E	PE_13
C1_FLG1	SHARC Core 1 Flag Pin	E	PE_01
C1_FLG2	SHARC Core 1 Flag Pin	F	PF_04
C1_FLG3	SHARC Core 1 Flag Pin	D	PD_06
C2_FLG0	SHARC Core 2 Flag Pin	B	PB_00
C2_FLG1	SHARC Core 2 Flag Pin	C	PC_14
C2_FLG2	SHARC Core 2 Flag Pin	F	PF_11
C2_FLG3	SHARC Core 2 Flag Pin	E	PE_15
CAN0_RX	CAN0 Receive	C	PC_12
CAN0_TX	CAN0 Transmit	C	PC_13
CAN1_RX	CAN1 Receive	C	PC_14
CAN1_TX	CAN1 Transmit	C	PC_15
CNT0_DG	CNT0 Count Down and Gate	D	PD_08
CNT0_UD	CNT0 Count Up and Direction	E	PE_13
CNT0_ZM	CNT0 Count Zero Marker	D	PD_07
DAI0_PIN01	DAI0 Pin 1	Not Muxed	DAI0_PIN01
DAI0_PIN02	DAI0 Pin 2	Not Muxed	DAI0_PIN02
DAI0_PIN03	DAI0 Pin 3	Not Muxed	DAI0_PIN03
DAI0_PIN04	DAI0 Pin 4	Not Muxed	DAI0_PIN04
DAI0_PIN05	DAI0 Pin 5	Not Muxed	DAI0_PIN05
DAI0_PIN06	DAI0 Pin 6	Not Muxed	DAI0_PIN06
DAI0_PIN07	DAI0 Pin 7	Not Muxed	DAI0_PIN07
DAI0_PIN08	DAI0 Pin 8	Not Muxed	DAI0_PIN08
DAI0_PIN09	DAI0 Pin 9	Not Muxed	DAI0_PIN09
DAI0_PIN10	DAI0 Pin 10	Not Muxed	DAI0_PIN10
DAI0_PIN11	DAI0 Pin 11	Not Muxed	DAI0_PIN11
DAI0_PIN12	DAI0 Pin 12	Not Muxed	DAI0_PIN12
DAI0_PIN13	DAI0 Pin 13	Not Muxed	DAI0_PIN13
DAI0_PIN14	DAI0 Pin 14	Not Muxed	DAI0_PIN14
DAI0_PIN15	DAI0 Pin 15	Not Muxed	DAI0_PIN15
DAI0_PIN16	DAI0 Pin 16	Not Muxed	DAI0_PIN16
DAI0_PIN17	DAI0 Pin 17	Not Muxed	DAI0_PIN17

Table 12. ADSP-SC57x/ADSP-2157x 400-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
DAI0_PIN18	DAI0 Pin 18	Not Muxed	DAI0_PIN18
DAI0_PIN19	DAI0 Pin 19	Not Muxed	DAI0_PIN19
DAI0_PIN20	DAI0 Pin 20	Not Muxed	DAI0_PIN20
DMC0_A00	DMC0 Address 0	Not Muxed	DMC0_A00
DMC0_A01	DMC0 Address 1	Not Muxed	DMC0_A01
DMC0_A02	DMC0 Address 2	Not Muxed	DMC0_A02
DMC0_A03	DMC0 Address 3	Not Muxed	DMC0_A03
DMC0_A04	DMC0 Address 4	Not Muxed	DMC0_A04
DMC0_A05	DMC0 Address 5	Not Muxed	DMC0_A05
DMC0_A06	DMC0 Address 6	Not Muxed	DMC0_A06
DMC0_A07	DMC0 Address 7	Not Muxed	DMC0_A07
DMC0_A08	DMC0 Address 8	Not Muxed	DMC0_A08
DMC0_A09	DMC0 Address 9	Not Muxed	DMC0_A09
DMC0_A10	DMC0 Address 10	Not Muxed	DMC0_A10
DMC0_A11	DMC0 Address 11	Not Muxed	DMC0_A11
DMC0_A12	DMC0 Address 12	Not Muxed	DMC0_A12
DMC0_A13	DMC0 Address 13	Not Muxed	DMC0_A13
DMC0_A14	DMC0 Address 14	Not Muxed	DMC0_A14
DMC0_A15	DMC0 Address 15	Not Muxed	DMC0_A15
DMC0_BA0	DMC0 Bank Address Input 0	Not Muxed	DMC0_BA0
DMC0_BA1	DMC0 Bank Address Input 1	Not Muxed	DMC0_BA1
DMC0_BA2	DMC0 Bank Address Input 2	Not Muxed	DMC0_BA2
$\overline{\text{DMC0\_CAS}}$	DMC0 Column Address Strobe	Not Muxed	$\overline{\text{DMC0\_CAS}}$
$\overline{\text{DMC0\_CK}}$	DMC0 Clock (complement)	Not Muxed	$\overline{\text{DMC0\_CK}}$
DMC0_CKE	DMC0 Clock enable	Not Muxed	DMC0_CKE
$\overline{\text{DMC0\_CS0}}$	DMC0 Chip Select 0	Not Muxed	$\overline{\text{DMC0\_CS0}}$
DMC0_DQ00	DMC0 Data 0	Not Muxed	DMC0_DQ00
DMC0_DQ01	DMC0 Data 1	Not Muxed	DMC0_DQ01
DMC0_DQ02	DMC0 Data 2	Not Muxed	DMC0_DQ02
DMC0_DQ03	DMC0 Data 3	Not Muxed	DMC0_DQ03
DMC0_DQ04	DMC0 Data 4	Not Muxed	DMC0_DQ04
DMC0_DQ05	DMC0 Data 5	Not Muxed	DMC0_DQ05
DMC0_DQ06	DMC0 Data 6	Not Muxed	DMC0_DQ06
DMC0_DQ07	DMC0 Data 7	Not Muxed	DMC0_DQ07
DMC0_DQ08	DMC0 Data 8	Not Muxed	DMC0_DQ08
DMC0_DQ09	DMC0 Data 9	Not Muxed	DMC0_DQ09
DMC0_DQ10	DMC0 Data 10	Not Muxed	DMC0_DQ10
DMC0_DQ11	DMC0 Data 11	Not Muxed	DMC0_DQ11
DMC0_DQ12	DMC0 Data 12	Not Muxed	DMC0_DQ12
DMC0_DQ13	DMC0 Data 13	Not Muxed	DMC0_DQ13
DMC0_DQ14	DMC0 Data 14	Not Muxed	DMC0_DQ14
DMC0_DQ15	DMC0 Data 15	Not Muxed	DMC0_DQ15
DMC0_LDM	DMC0 Data Mask for Lower Byte	Not Muxed	DMC0_LDM
$\overline{\text{DMC0\_LDQS}}$	DMC0 Data Strobe for Lower Byte (complement)	Not Muxed	$\overline{\text{DMC0\_LDQS}}$
DMC0_ODT	DMC0 On-die termination	Not Muxed	DMC0_ODT
$\overline{\text{DMC0\_RAS}}$	DMC0 Row Address Strobe	Not Muxed	$\overline{\text{DMC0\_RAS}}$
$\overline{\text{DMC0\_RESET}}$	DMC0 Reset (DDR3 only)	Not Muxed	$\overline{\text{DMC0\_RESET}}$
DMC0_RZQ	DMC0 External calibration resistor connection	Not Muxed	DMC0_RZQ

Table 12. ADSP-SC57x/ADSP-2157x 400-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
DMC0_UDM	DMC0 Data Mask for Upper Byte	Not Muxed	DMC0_UDM
DMC0_UDQS	DMC0 Data Strobe for Upper Byte (complement)	Not Muxed	DMC0_UDQS
DMC0_VREF	DMC0 Voltage Reference	Not Muxed	DMC0_VREF
DMC0_WE	DMC0 Write Enable	Not Muxed	DMC0_WE
ETH0_COL	EMAC0 MII Collision detect	C	PC_06
ETH0_CRS	EMAC0 Carrier Sense/RMII Receive Data Valid	B	PB_01
ETH0_MDC	EMAC0 Management Channel Clock	A	PA_11
ETH0_MDIO	EMAC0 Management Channel Serial Data	A	PA_10
ETH0_PTPAUXIN0	EMAC0 PTP Auxiliary Trigger Input 0	D	PD_14
ETH0_PTPAUXIN1	EMAC0 PTP Auxiliary Trigger Input 1	D	PD_15
ETH0_PTPAUXIN2	EMAC0 PTP Auxiliary Trigger Input 2	F	PF_06
ETH0_PTPAUXIN3	EMAC0 PTP Auxiliary Trigger Input 3	F	PF_07
ETH0_PTPCLKIN0	EMAC0 PTP Clock Input 0	F	PF_05
ETH0_PTPPPS0	EMAC0 PTP Pulse-Per-Second Output 0	A	PA_09
ETH0_PTPPPS1	EMAC0 PTP Pulse-Per-Second Output 1	D	PD_08
ETH0_PTPPPS2	EMAC0 PTP Pulse-Per-Second Output 2	E	PE_00
ETH0_PTPPPS3	EMAC0 PTP Pulse-Per-Second Output 3	E	PE_01
ETH0_RXCLK_REFCLK	EMAC0 RXCLK (GigE) or REFCLK (10/100)	B	PB_00
ETH0_RXCTL_RXDV	EMAC0 RXCTL (GigE) or CRS (10/100)	B	PB_01
ETH0_RXD0	EMAC0 Receive Data 0	A	PA_13
ETH0_RXD1	EMAC0 Receive Data 1	A	PA_12
ETH0_RXD2	EMAC0 Receive Data 2	A	PA_14
ETH0_RXD3	EMAC0 Receive Data 3	A	PA_15
ETH0_RXERR	EMAC0 Receive Error	B	PB_03
ETH0_TXCLK	EMAC0 Transmit Clock	B	PB_04
ETH0_TXCTL_TXEN	EMAC0 TXCTL (GigE) or TXEN (10/100)	B	PB_09
ETH0_TXD0	EMAC0 Transmit Data 0	B	PB_07
ETH0_TXD1	EMAC0 Transmit Data 1	B	PB_08
ETH0_TXD2	EMAC0 Transmit Data 2	B	PB_06
ETH0_TXD3	EMAC0 Transmit Data 3	B	PB_05
HADC0_EOC_DOUT	HADC0 End of Conversion / Serial Data Out	D	PD_09
HADC0_VIN0	HADC0 Analog Input at channel 0	Not Muxed	HADC0_VIN0
HADC0_VIN1	HADC0 Analog Input at channel 1	Not Muxed	HADC0_VIN1
HADC0_VIN2	HADC0 Analog Input at channel 2	Not Muxed	HADC0_VIN2
HADC0_VIN3	HADC0 Analog Input at channel 3	Not Muxed	HADC0_VIN3
HADC0_VIN4	HADC0 Analog Input at channel 4	Not Muxed	HADC0_VIN4
HADC0_VIN5	HADC0 Analog Input at channel 5	Not Muxed	HADC0_VIN5
HADC0_VIN6	HADC0 Analog Input at channel 6	Not Muxed	HADC0_VIN6
HADC0_VIN7	HADC0 Analog Input at channel 7	Not Muxed	HADC0_VIN7
HADC0_VREFN	HADC0 Ground Reference for ADC	Not Muxed	HADC0_VREFN
HADC0_VREFP	HADC0 External Reference for ADC	Not Muxed	HADC0_VREFP
JTG_TCK	JTAG Clock	Not Muxed	JTG_TCK
JTG_TDI	JTAG Serial Data In	Not Muxed	JTG_TDI
JTG_TDO	JTAG Serial Data Out	Not Muxed	JTG_TDO
JTG_TMS	JTAG Mode Select	Not Muxed	JTG_TMS
JTG_TRST	JTAG Reset	Not Muxed	JTG_TRST
LP0_ACK	LP0 Acknowledge	E	PE_03
LP0_CLK	LP0 Clock	E	PE_02

Table 12. ADSP-SC57x/ADSP-2157x 400-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
LP0_D0	LP0 Data 0	E	PE_04
LP0_D1	LP0 Data 1	E	PE_05
LP0_D2	LP0 Data 2	E	PE_06
LP0_D3	LP0 Data 3	E	PE_07
LP0_D4	LP0 Data 4	E	PE_08
LP0_D5	LP0 Data 5	E	PE_09
LP0_D6	LP0 Data 6	E	PE_10
LP0_D7	LP0 Data 7	E	PE_11
LP1_ACK	LP1 Acknowledge	B	PB_01
LP1_CLK	LP1 Clock	B	PB_03
LP1_D0	LP1 Data 0	D	PD_10
LP1_D1	LP1 Data 1	D	PD_11
LP1_D2	LP1 Data 2	D	PD_12
LP1_D3	LP1 Data 3	D	PD_13
LP1_D4	LP1 Data 4	D	PD_14
LP1_D5	LP1 Data 5	D	PD_15
LP1_D6	LP1 Data 6	A	PA_09
LP1_D7	LP1 Data 7	D	PD_09
MLB0_CLK	MLB0 Single-Ended Clock	B	PB_06
MLB0_CLKN	MLB0 Differential Clock (-)	Not Muxed	MLB0_CLKN
MLB0_CLKOUT	MLB0 Single-Ended Clock Out	B	PB_03
MLB0_CLKP	MLB0 Differential Clock (+)	Not Muxed	MLB0_CLKP
MLB0_DAT	MLB0 Single-Ended Data	B	PB_04
MLB0_DATN	MLB0 Differential Data (-)	Not Muxed	MLB0_DATN
MLB0_DATP	MLB0 Differential Data (+)	Not Muxed	MLB0_DATP
MLB0_SIG	MLB0 Single-Ended Signal	B	PB_05
MLB0_SIGN	MLB0 Differential Signal (-)	Not Muxed	MLB0_SIGN
MLB0_SIGP	MLB0 Differential Signal (+)	Not Muxed	MLB0_SIGP
$\overline{\text{MSI0\_CD}}$	MSI0 Card Detect	C	PC_12
MSI0_CLK	MSI0 Clock	F	PF_04
MSI0_CMD	MSI0 Command	F	PF_07
MSI0_D0	MSI0 Data 0	E	PE_12
MSI0_D1	MSI0 Data 1	E	PE_13
MSI0_D2	MSI0 Data 2	E	PE_14
MSI0_D3	MSI0 Data 3	E	PE_15
MSI0_D4	MSI0 Data 4	F	PF_00
MSI0_D5	MSI0 Data 5	F	PF_01
MSI0_D6	MSI0 Data 6	F	PF_02
MSI0_D7	MSI0 Data 7	F	PF_03
$\overline{\text{MSI0\_INT}}$	MSI0 eSDIO Interrupt Input	C	PC_13
PPI0_CLK	EPPI0 Clock	C	PC_11
PPI0_D00	EPPI0 Data 0	D	PD_10
PPI0_D01	EPPI0 Data 1	D	PD_11
PPI0_D02	EPPI0 Data 2	D	PD_12
PPI0_D03	EPPI0 Data 3	D	PD_13
PPI0_D04	EPPI0 Data 4	D	PD_14
PPI0_D05	EPPI0 Data 5	D	PD_15
PPI0_D06	EPPI0 Data 6	C	PC_05

Table 12. ADSP-SC57x/ADSP-2157x 400-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
PPI0_D07	EPPI0 Data 7	D	PD_09
PPI0_D08	EPPI0 Data 8	C	PC_01
PPI0_D09	EPPI0 Data 9	C	PC_02
PPI0_D10	EPPI0 Data 10	C	PC_03
PPI0_D11	EPPI0 Data 11	C	PC_04
PPI0_D12	EPPI0 Data 12	E	PE_00
PPI0_D13	EPPI0 Data 13	C	PC_07
PPI0_D14	EPPI0 Data 14	C	PC_08
PPI0_D15	EPPI0 Data 15	E	PE_01
PPI0_FS1	EPPI0 Frame Sync 1 (HSYNC)	C	PC_14
PPI0_FS2	EPPI0 Frame Sync 2 (VSYNC)	C	PC_15
PPI0_FS3	EPPI0 Frame Sync 3 (FIELD)	C	PC_06
SPI0_CLK	SPI0 Clock	C	PC_01
SPI0_MISO	SPI0 Master In, Slave Out	C	PC_02
SPI0_MOSI	SPI0 Master Out, Slave In	C	PC_03
SPI0_RDY	SPI0 Ready	C	PC_05
$\overline{\text{SPI0\_SEL1}}$	SPI0 Slave Select Output 1	C	PC_04
$\overline{\text{SPI0\_SEL2}}$	SPI0 Slave Select Output 2	C	PC_05
$\overline{\text{SPI0\_SEL3}}$	SPI0 Slave Select Output 3	C	PC_06
$\overline{\text{SPI0\_SEL4}}$	SPI0 Slave Select Output 4	A	PA_09
$\overline{\text{SPI0\_SEL5}}$	SPI0 Slave Select Output 5	F	PF_05
$\overline{\text{SPI0\_SEL6}}$	SPI0 Slave Select Output 6	F	PF_04
$\overline{\text{SPI0\_SEL7}}$	SPI0 Slave Select Output 7	D	PD_05
$\overline{\text{SPI0\_SS}}$	SPI0 Slave Select Input	C	PC_04
SPI1_CLK	SPI1 Clock	C	PC_07
SPI1_MISO	SPI1 Master In, Slave Out	C	PC_08
SPI1_MOSI	SPI1 Master Out, Slave In	C	PC_09
SPI1_RDY	SPI1 Ready	C	PC_11
$\overline{\text{SPI1\_SEL1}}$	SPI1 Slave Select Output 1	C	PC_10
$\overline{\text{SPI1\_SEL2}}$	SPI1 Slave Select Output 2	C	PC_11
$\overline{\text{SPI1\_SEL3}}$	SPI1 Slave Select Output 3	F	PF_11
$\overline{\text{SPI1\_SEL4}}$	SPI1 Slave Select Output 4	A	PA_14
$\overline{\text{SPI1\_SEL5}}$	SPI1 Slave Select Output 5	B	PB_02
$\overline{\text{SPI1\_SEL6}}$	SPI1 Slave Select Output 6	D	PD_07
$\overline{\text{SPI1\_SEL7}}$	SPI1 Slave Select Output 7	D	PD_06
$\overline{\text{SPI1\_SS}}$	SPI1 Slave Select Input	C	PC_10
SPI2_CLK	SPI2 Clock	B	PB_14
SPI2_D2	SPI2 Data 2	B	PB_12
SPI2_D3	SPI2 Data 3	B	PB_13
SPI2_MISO	SPI2 Master In, Slave Out	B	PB_10
SPI2_MOSI	SPI2 Master Out, Slave In	B	PB_11
SPI2_RDY	SPI2 Ready	C	PC_00
$\overline{\text{SPI2\_SEL1}}$	SPI2 Slave Select Output 1	B	PB_15
$\overline{\text{SPI2\_SEL2}}$	SPI2 Slave Select Output 2	F	PF_10
$\overline{\text{SPI2\_SEL3}}$	SPI2 Slave Select Output 3	C	PC_00
$\overline{\text{SPI2\_SEL4}}$	SPI2 Slave Select Output 4	D	PD_08
$\overline{\text{SPI2\_SEL5}}$	SPI2 Slave Select Output 5	A	PA_15
$\overline{\text{SPI2\_SEL6}}$	SPI2 Slave Select Output n	A	PA_10

Table 12. ADSP-SC57x/ADSP-2157x 400-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
SPI2_SEL7	SPI2 Slave Select Output n	B	PB_07
SPI2_SS	SPI2 Slave Select Input	B	PB_15
SYS_BMODE0	Boot Mode Control n	Not Muxed	SYS_BMODE0
SYS_BMODE1	Boot Mode Control n	Not Muxed	SYS_BMODE1
SYS_BMODE2	Boot Mode Control n	Not Muxed	SYS_BMODE2
SYS_CLKIN0	Clock/Crystal Input	Not Muxed	SYS_CLKIN0
SYS_CLKIN1	Clock/Crystal Input	Not Muxed	SYS_CLKIN1
SYS_CLKOUT	Processor Clock Output	Not Muxed	SYS_CLKOUT
SYS_FAULT	Active-High Fault Output	Not Muxed	SYS_FAULT
SYS_FAULT	Active-Low Fault Output	Not Muxed	SYS_FAULT
SYS_HWRST	Processor Hardware Reset Control	Not Muxed	SYS_HWRST
SYS_RESOUT	Reset Output	Not Muxed	SYS_RESOUT
SYS_XTAL0	Crystal Output	Not Muxed	SYS_XTAL0
SYS_XTAL1	Crystal Output	Not Muxed	SYS_XTAL1
TM0_ACI0	TIMER0 Alternate Capture Input 0	F	PF_09
TM0_ACI1	TIMER0 Alternate Capture Input 1	F	PF_11
TM0_ACI2	TIMER0 Alternate Capture Input 2	C	PC_12
TM0_ACI3	TIMER0 Alternate Capture Input 3	C	PC_14
TM0_ACI4	TIMER0 Alternate Capture Input 4	C	PC_13
TM0_ACI5	TIMER0 Alternate Capture Input 5	Not Applicable	DAI0_PIN04[1]
TM0_ACI6	TIMER0 Alternate Capture Input 6	Not Applicable	DAI0_PIN19[1]
TM0_ACI7	TIMER0 Alternate Capture Input 7	Not Applicable	CNT0_TO
TM0_ACLK0	TIMER0 Alternate Clock 0	Not Applicable	SYS_CLKIN1
TM0_ACLK1	TIMER0 Alternate Clock 1	F	PF_06
TM0_ACLK2	TIMER0 Alternate Clock 2	C	PC_01
TM0_ACLK3	TIMER0 Alternate Clock 3	D	PD_09
TM0_ACLK4	TIMER0 Alternate Clock 4	E	PE_02
TM0_ACLK5	TIMER0 Alternate Clock 5	Not Applicable	DAI0_PIN03[1]
TM0_ACLK6	TIMER0 Alternate Clock 6	Not Applicable	DAI0_PIN20[1]
TM0_ACLK7	TIMER0 Alternate Clock 7	Not Applicable	SYS_CLKIN0
TM0_CLK	TIMER0 Clock	C	PC_03
TM0_TMR0	TIMER0 Timer 0	E	PE_12
TM0_TMR1	TIMER0 Timer 1	F	PF_05
TM0_TMR2	TIMER0 Timer 2	F	PF_07
TM0_TMR3	TIMER0 Timer 3	B	PB_01
TM0_TMR4	TIMER0 Timer 4	B	PB_03
TM0_TMR5	TIMER0 Timer 5	C	PC_15
TM0_TMR6	TIMER0 Timer 6	E	PE_14
TM0_TMR7	TIMER0 Timer 7	D	PD_07
TRACE0_CLK	TRACE0 Trace Clock	F	PF_06
TRACE0_D00	TRACE0 Trace Data 0	F	PF_00
TRACE0_D01	TRACE0 Trace Data	F	PF_01
TRACE0_D02	TRACE0 Trace Data	F	PF_02
TRACE0_D03	TRACE0 Trace Data 3	F	PF_03
TRACE0_D04	TRACE0 Trace Data	D	PD_10
TRACE0_D05	TRACE0 Trace Data	D	PD_11
TRACE0_D06	TRACE0 Trace Data	D	PD_12
TRACE0_D07	TRACE0 Trace Data 7	D	PD_13

Table 12. ADSP-SC57x/ADSP-2157x 400-Ball CSP_BGA Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
TWI0_SCL	TWI0 Serial Clock	Not Muxed	TWI0_SCL
TWI0_SDA	TWI0 Serial Data	Not Muxed	TWI0_SDA
TWI1_SCL	TWI1 Serial Clock	Not Muxed	TWI1_SCL
TWI1_SDA	TWI1 Serial Data	Not Muxed	TWI1_SDA
TWI2_SCL	TWI2 Serial Clock	Not Muxed	TWI2_SCL
TWI2_SDA	TWI2 Serial Data	Not Muxed	TWI2_SDA
$\overline{\text{UART0\_CTS}}$	UART0 Clear to Send	D	PD_06
$\overline{\text{UART0\_RTS}}$	UART0 Request to Send	D	PD_05
$\overline{\text{UART0\_RX}}$	UART0 Receive	F	PF_09
$\overline{\text{UART0\_TX}}$	UART0 Transmit	F	PF_08
$\overline{\text{UART1\_CTS}}$	UART1 Clear to Send	E	PE_14
$\overline{\text{UART1\_RTS}}$	UART1 Request to Send	E	PE_00
$\overline{\text{UART1\_RX}}$	UART1 Receive	F	PF_11
$\overline{\text{UART1\_TX}}$	UART1 Transmit	F	PF_10
$\overline{\text{UART2\_CTS}}$	UART2 Clear to Send	A	PA_11
$\overline{\text{UART2\_RTS}}$	UART2 Request to Send	A	PA_10
$\overline{\text{UART2\_RX}}$	UART2 Receive	C	PC_13
$\overline{\text{UART2\_TX}}$	UART2 Transmit	C	PC_12
USB0_CLKIN	USB0 Clock/Crystal Input	Not Muxed	USB_CLKIN
USB0_DM	USB0 Data -	Not Muxed	USB0_DM
USB0_DP	USB0 Data +	Not Muxed	USB0_DP
USB0_ID	USB0 OTG ID	Not Muxed	USB0_ID
USB0_VBC	USB0 VBUS Control	Not Muxed	USB0_VBC
USB0_VBUS	USB0 Bus Voltage	Not Muxed	USB0_VBUS
USB0_XTAL	USB0 Crystal	Not Muxed	USB_XTAL
VDD_EXT	External Voltage Domain	Not Muxed	VDD_EXT
VDD_INT	Internal Voltage Domain	Not Muxed	VDD_INT
VDD_DMC	DMC VVD	Not Muxed	VDD_DMC
VDD_HADC	HADC VVD	Not Muxed	VDD_HADC
VDD_USB	USB VVD	Not Muxed	VDD_USB

[1] Signal is routed to the DAI0_PINnn pin through the DAI0_PBnn pin buffers using the SRU.

GPIO MULTIPLEXING FOR 400-BALL CSP_BGA

Table 13 through Table 18 identify the pin functions that are multiplexed on the general-purpose I/O pins of the 400-ball CSP_BGA package.

Table 13. Signal Multiplexing for Port A

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PA_00					
PA_01					
PA_02					
PA_03					
PA_04					
PA_05					
PA_06					
PA_07					
PA_08					
PA_09	ETH0_PTPPPS0	LP1_D6	SPI0_SEL4		
PA_10	ETH0_MDIO	UART2_RTS	SPI2_SEL6		
PA_11	ETH0_MDC	UART2_CTS			
PA_12	ETH0_RXD1				
PA_13	ETH0_RXD0				
PA_14	ETH0_RXD2	ACM0_A3	SPI1_SEL4		
PA_15	ETH0_RXD3	ACM0_T0	SPI2_SEL5		

Table 14. Signal Multiplexing for Port B

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PB_00	ETH0_RXCLK_REFCLK	C2_FLG0			
PB_01	ETH0_CRS	ACM0_A4	LP1_ACK	TM0_TMR3	
PB_02	ETH0_RXCTL_RXDV		SPI1_SEL5		
PB_03	ETH0_RXERR	MLB0_CLKOUT	LP1_CLK	TM0_TMR4	
PB_04	ETH0_TXCLK	MLB0_DAT			
PB_05	ETH0_TXD3	MLB0_SIG			
PB_06	ETH0_TXD2	MLB0_CLK			
PB_07	ETH0_TXD0		SPI2_SEL7		
PB_08	ETH0_TXD1				
PB_09	ETH0_TXCTL_TXEN				
PB_10	SPI2_MISO				
PB_11	SPI2_MOSI				
PB_12	SPI2_D2				
PB_13	SPI2_D3				
PB_14	SPI2_CLK				
PB_15	SPI2_SEL1				SPI2_SS

Table 15. Signal Multiplexing for Port C

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PC_00	SPI2_SEL3	SPI2_RDY			
PC_01	SPI0_CLK	PPI0_D08			TM0_ACLK2
PC_02	SPI0_MISO	PPI0_D09			
PC_03	SPI0_MOSI	PPI0_D10			TM0_CLK
PC_04	SPI0_SEL1	PPI0_D11			SPI0_SS
PC_05	SPI0_SEL2	PPI0_D06	SPI0_RDY		
PC_06	SPI0_SEL3	ETH0_COL	PPI0_FS3		
PC_07	SPI1_CLK	PPI0_D13			
PC_08	SPI1_MISO	PPI0_D14			
PC_09	SPI1_MOSI				
PC_10	SPI1_SEL1				SPI1_SS
PC_11	SPI1_SEL2	PPI0_CLK	SPI1_RDY		TM0_ACLK4
PC_12	CAN0_RX	MSI0_CD	UART2_TX		TM0_ACI2
PC_13	CAN0_TX	MSI0_INT	UART2_RX		TM0_ACI4
PC_14	CAN1_RX	PPI0_FS1	ACM0_A1	C2_FLG1	TM0_ACI3
PC_15	CAN1_TX	PPI0_FS2	ACM0_A2	TM0_TMR5	

Table 16. Signal Multiplexing for Port D

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PD_00					
PD_01					
PD_02					
PD_03					
PD_04					
PD_05	SPI0_SEL7		UART0_RTS		
PD_06	SPI1_SEL7	C1_FLG3	UART0_CTS		
PD_07	SPI1_SEL6	CNT0_ZM	TM0_TMR7		
PD_08	ETH0_PTPPPS1	CNT0_DG	SPI2_SEL4		
PD_09	LP1_D7	PPI0_D07	HADC0_EOC_DOUT		TM0_ACLK3
PD_10	LP1_D0	PPI0_D00	TRACE0_D04		
PD_11	LP1_D1	PPI0_D01	TRACE0_D05		
PD_12	LP1_D2	PPI0_D02	TRACE0_D06		
PD_13	LP1_D3	PPI0_D03	TRACE0_D07		
PD_14	LP1_D4	PPI0_D04	ETH0_PTPAUXIN0		
PD_15	LP1_D5	PPI0_D05	ETH0_PTPAUXIN1		

Table 17. Signal Multiplexing for Port E

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PE_00	ETH0_PTPPPS2	PPI0_D12	UART1_RTS		
PE_01	ETH0_PTPPPS3	PPI0_D15	C1_FLG1		
PE_02	LP0_CLK				
PE_03	LP0_ACK				
PE_04	LP0_D0				
PE_05	LP0_D1				
PE_06	LP0_D2				
PE_07	LP0_D3				
PE_08	LP0_D4				
PE_09	LP0_D5				
PE_10	LP0_D6				
PE_11	LP0_D7				
PE_12	MSI0_D0		TM0_TMR0		
PE_13	MSI0_D1	C1_FLG0	CNT0_UD		
PE_14	MSI0_D2	UART1_CTS	TM0_TMR6		
PE_15	MSI0_D3	C2_FLG3			

Table 18. Signal Multiplexing for Port F

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PF_00	MSI0_D4	TRACE0_D00			
PF_01	MSI0_D5	TRACE0_D01			
PF_02	MSI0_D6	TRACE0_D02			
PF_03	MSI0_D7	TRACE0_D03			
PF_04	MSI0_CLK	C1_FLG2	SPI0_SEL6		
PF_05	ETH0_PTPCLKIN0	TM0_TMR1	SPI0_SEL5		
PF_06	ETH0_PTPAUXIN2	TRACE0_CLK			TM0_ACLK1
PF_07	ETH0_PTPAUXIN3	TM0_TMR2	MSI0_CMD		
PF_08	UART0_TX				
PF_09	UART0_RX				TM0_ACI0
PF_10	UART1_TX	SPI2_SEL2			
PF_11	UART1_RX	ACM0_A0	SPI1_SEL3	C2_FLG2	TM0_ACI1

The following alternate clock and alternate capture input signals are internally routed to specific system signals. The connections are fixed. If these modes are enabled, the specific system signals are automatically routed to the timer input signals. Table 19 shows the details of this routing.

Table 19. Internal Timer Signal Routing

Timer Input Signal	Internal Source
TM0_ACLK0[1]	SYS_CLKIN1
TM0_ACI5	DAI0_PB04_O
TM0_ACLK5	DAI0_PB03_O
TM0_ACI6	DAI0_PB20_O
TM0_ACLK6	DAI0_PB19_O
TM0_ACI7	CNT0_TO
TM0_ACLK7	SYS_CLKIN0

[1] Not applicable for LQFP package.

176-LEAD LQFP SIGNAL DESCRIPTIONS

The processor's pin definitions are shown Table 20. The columns in this table provide the following information:

- Signal Name: The Signal Name column in the table includes the signal name for every pin and (where applicable) the GPIO multiplexed pin function for every pin.
- Description: The Description column in the table provides a verbose (descriptive) name for the signal.
- General-Purpose Port: The Port column in the table shows whether or not the signal is multiplexed with other signals on a general-purpose I/O port pin.
- Pin Name: The Pin Name column in the table identifies the name of the package pin (at power on reset) on which the signal is located (if a single function pin) or is multiplexed (if a general-purpose I/O pin).

Table 20. ADSP-SC57x/ADSP-2157x 176-Lead LQFP Signal Descriptions

Signal Name	Description	Port	Pin Name
ACM0_A0	ACM0 ADC Control Signals	A	PA_08
ACM0_A1	ACM0 ADC Control Signals	C	PC_14
ACM0_A2	ACM0 ADC Control Signals	C	PC_15
ACM0_A3	ACM0 ADC Control Signals	A	PA_14
ACM0_A4	ACM0 ADC Control Signals	B	PB_01
ACM0_T0	ACM0 External Trigger n	A	PA_15
C1_FLG0	SHARC Core 1 Flag Pin	D	PD_00
C1_FLG1	SHARC Core 1 Flag Pin	D	PD_01
C1_FLG2	SHARC Core 1 Flag Pin	C	PC_09
C1_FLG3	SHARC Core 1 Flag Pin	D	PD_06
C2_FLG0	SHARC Core 2 Flag Pin	B	PB_00
C2_FLG1	SHARC Core 2 Flag Pin	C	PC_14
C2_FLG2	SHARC Core 2 Flag Pin	C	PC_15
C2_FLG3	SHARC Core 2 Flag Pin	D	PD_05
CAN0_RX	CAN0 Receive	C	PC_12
CAN0_TX	CAN0 Transmit	C	PC_13
CAN1_RX	CAN1 Receive	C	PC_14
CAN1_TX	CAN1 Transmit	C	PC_15
CNT0_DG	CNT0 Count Down and Gate	D	PD_08
CNT0_UD	CNT0 Count Up and Direction	D	PD_00
CNT0_ZM	CNT0 Count Zero Marker	D	PD_07
DAI0_PIN01	DAI0 Pin 1	Not Muxed	DAI0_PIN01
DAI0_PIN02	DAI0 Pin 2	Not Muxed	DAI0_PIN02
DAI0_PIN03	DAI0 Pin 3	Not Muxed	DAI0_PIN03
DAI0_PIN04	DAI0 Pin 4	Not Muxed	DAI0_PIN04
DAI0_PIN05	DAI0 Pin 5	Not Muxed	DAI0_PIN05
DAI0_PIN06	DAI0 Pin 6	Not Muxed	DAI0_PIN06
DAI0_PIN07	DAI0 Pin 7	Not Muxed	DAI0_PIN07
DAI0_PIN08	DAI0 Pin 8	Not Muxed	DAI0_PIN08
DAI0_PIN09	DAI0 Pin 9	Not Muxed	DAI0_PIN09
DAI0_PIN10	DAI0 Pin 10	Not Muxed	DAI0_PIN10
DAI0_PIN11	DAI0 Pin 11	Not Muxed	DAI0_PIN11
DAI0_PIN12	DAI0 Pin 12	Not Muxed	DAI0_PIN12
DAI0_PIN13	DAI0 Pin 13	Not Muxed	DAI0_PIN13
DAI0_PIN14	DAI0 Pin 14	Not Muxed	DAI0_PIN14
DAI0_PIN15	DAI0 Pin 15	Not Muxed	DAI0_PIN15
DAI0_PIN16	DAI0 Pin 16	Not Muxed	DAI0_PIN16

Table 20. ADSP-SC57x/ADSP-2157x 176-Lead LQFP Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
DAI0_PIN17	DAI0 Pin 17	Not Muxed	DAI0_PIN17
DAI0_PIN18	DAI0 Pin 18	Not Muxed	DAI0_PIN18
DAI0_PIN19	DAI0 Pin 19	Not Muxed	DAI0_PIN19
DAI0_PIN20	DAI0 Pin 20	Not Muxed	DAI0_PIN20
ETH0_COL	EMAC0 MII Collision detect	C	PC_06
ETH0_CRS	EMAC0 Carrier Sense/RMII Receive Data Valid	B	PB_01
ETH0_MDC	EMAC0 Management Channel Clock	A	PA_11
ETH0_MDIO	EMAC0 Management Channel Serial Data	A	PA_10
ETH0_PTPAUXIN0	EMAC0 PTP Auxiliary Trigger Input 0	D	PD_14
ETH0_PTPAUXIN1	EMAC0 PTP Auxiliary Trigger Input 1	D	PD_15
ETH0_PTPPPS0	EMAC0 PTP Pulse-Per-Second Output 0	A	PA_09
ETH0_PTPPPS1	EMAC0 PTP Pulse-Per-Second Output 1	D	PD_08
ETH0_RXCLK_REFCLK	EMAC0 RXCLK (GigE) or REFCLK (10/100)	B	PB_00
ETH0_RXCTL_RXDV	EMAC0 RXCTL (GigE) or CRS (10/100)	B	PB_01
ETH0_RXD0	EMAC0 Receive Data 0	A	PA_13
ETH0_RXD1	EMAC0 Receive Data 1	A	PA_12
ETH0_RXD2	EMAC0 Receive Data 2	A	PA_14
ETH0_RXD3	EMAC0 Receive Data 3	A	PA_15
ETH0_RXERR	EMAC0 Receive Error	B	PB_03
ETH0_TXCLK	EMAC0 Transmit Clock	B	PB_04
ETH0_TXCTL_TXEN	EMAC0 TXCTL (GigE) or TXEN (10/100)	B	PB_09
ETH0_TXD0	EMAC0 Transmit Data 0	B	PB_07
ETH0_TXD1	EMAC0 Transmit Data 1	B	PB_08
ETH0_TXD2	EMAC0 Transmit Data 2	B	PB_06
ETH0_TXD3	EMAC0 Transmit Data 3	B	PB_05
HADC0_EOC_DOUT	HADC0 End of Conversion / Serial Data Out	D	PD_09
HADC0_VIN0	HADC0 Analog Input at channel 0	Not Muxed	HADC0_VIN0
HADC0_VIN1	HADC0 Analog Input at channel 1	Not Muxed	HADC0_VIN1
HADC0_VIN2	HADC0 Analog Input at channel 2	Not Muxed	HADC0_VIN2
HADC0_VIN3	HADC0 Analog Input at channel 3	Not Muxed	HADC0_VIN3
HADC0_VREFN	HADC0 Ground Reference for ADC	Not Muxed	HADC0_VREFN
HADC0_VREFP	HADC0 External Reference for ADC	Not Muxed	HADC0_VREFP
JTG_TCK	JTAG Clock	Not Muxed	JTG_TCK
JTG_TDI	JTAG Serial Data In	Not Muxed	JTG_TDI
JTG_TDO	JTAG Serial Data Out	Not Muxed	JTG_TDO
JTG_TMS	JTAG Mode Select	Not Muxed	JTG_TMS
$\overline{\text{JTG\_TRST}}$	JTAG Reset	Not Muxed	$\overline{\text{JTG\_TRST}}$
LP1_ACK	LP1 Acknowledge	B	PB_01
LP1_CLK	LP1 Clock	B	PB_03
LP1_D0	LP1 Data 0	D	PD_10
LP1_D1	LP1 Data 1	D	PD_11
LP1_D2	LP1 Data 2	D	PD_12
LP1_D3	LP1 Data 3	D	PD_13
LP1_D4	LP1 Data 4	D	PD_14
LP1_D5	LP1 Data 5	D	PD_15
LP1_D6	LP1 Data 6	A	PA_09
LP1_D7	LP1 Data 7	D	PD_09
MLB0_CLK	MLB0 Single-Ended Clock	B	PB_06

Table 20. ADSP-SC57x/ADSP-2157x 176-Lead LQFP Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
MLB0_CLKOUT	MLB0 Single-Ended Clock Out	B	PB_03
MLB0_DAT	MLB0 Single-Ended Data	B	PB_04
MLB0_SIG	MLB0 Single-Ended Signal	B	PB_05
PPI0_CLK	EPPI0 Clock	C	PC_11
PPI0_D00	EPPI0 Data 0	D	PD_10
PPI0_D01	EPPI0 Data 1	D	PD_11
PPI0_D02	EPPI0 Data 2	D	PD_12
PPI0_D03	EPPI0 Data 3	D	PD_13
PPI0_D04	EPPI0 Data 4	D	PD_14
PPI0_D05	EPPI0 Data 5	D	PD_15
PPI0_D06	EPPI0 Data 6	C	PC_05
PPI0_D07	EPPI0 Data 7	D	PD_09
PPI0_D08	EPPI0 Data 8	C	PC_01
PPI0_D09	EPPI0 Data 9	C	PC_02
PPI0_D10	EPPI0 Data 10	C	PC_03
PPI0_D11	EPPI0 Data 11	C	PC_04
PPI0_FS1	EPPI0 Frame Sync 1 (HSYNC)	C	PC_14
PPI0_FS2	EPPI0 Frame Sync 2 (VSYNC)	C	PC_15
PPI0_FS3	EPPI0 Frame Sync 3 (FIELD)	C	PC_06
SPI0_CLK	SPI0 Clock	C	PC_01
SPI0_MISO	SPI0 Master In, Slave Out	C	PC_02
SPI0_MOSI	SPI0 Master Out, Slave In	C	PC_03
SPI0_RDY	SPI0 Ready	C	PC_05
$\overline{\text{SPI0\_SEL1}}$	SPI0 Slave Select Output 1	C	PC_04
$\overline{\text{SPI0\_SEL2}}$	SPI0 Slave Select Output 2	C	PC_05
$\overline{\text{SPI0\_SEL3}}$	SPI0 Slave Select Output 3	C	PC_06
$\overline{\text{SPI0\_SEL4}}$	SPI0 Slave Select Output 4	A	PA_09
$\overline{\text{SPI0\_SEL5}}$	SPI0 Slave Select Output 5	D	PD_03
$\overline{\text{SPI0\_SEL6}}$	SPI0 Slave Select Output 6	D	PD_04
$\overline{\text{SPI0\_SEL7}}$	SPI0 Slave Select Output 7	D	PD_05
$\overline{\text{SPI0\_SS}}$	SPI0 Slave Select Input	C	PC_04
SPI1_CLK	SPI1 Clock	C	PC_07
SPI1_MISO	SPI1 Master In, Slave Out	C	PC_08
SPI1_MOSI	SPI1 Master Out, Slave In	C	PC_09
SPI1_RDY	SPI1 Ready	C	PC_11
$\overline{\text{SPI1\_SEL1}}$	SPI1 Slave Select Output 1	C	PC_10
$\overline{\text{SPI1\_SEL2}}$	SPI1 Slave Select Output 2	C	PC_11
$\overline{\text{SPI1\_SEL3}}$	SPI1 Slave Select Output 3	A	PA_08
$\overline{\text{SPI1\_SEL4}}$	SPI1 Slave Select Output 4	A	PA_14
$\overline{\text{SPI1\_SEL5}}$	SPI1 Slave Select Output 5	B	PB_02
$\overline{\text{SPI1\_SEL6}}$	SPI1 Slave Select Output 6	D	PD_07
$\overline{\text{SPI1\_SEL7}}$	SPI1 Slave Select Output 7	D	PD_06
$\overline{\text{SPI1\_SS}}$	SPI1 Slave Select Input	C	PC_10
SPI2_CLK	SPI2 Clock	B	PB_14
SPI2_D2	SPI2 Data 2	B	PB_12
SPI2_D3	SPI2 Data 3	B	PB_13
SPI2_MISO	SPI2 Master In, Slave Out	B	PB_10
SPI2_MOSI	SPI2 Master Out, Slave In	B	PB_11

Table 20. ADSP-SC57x/ADSP-2157x 176-Lead LQFP Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
SPI2_RDY	SPI2 Ready	C	PC_00
SPI2_SEL1	SPI2 Slave Select Output 1	B	PB_15
SPI2_SEL2	SPI2 Slave Select Output 2	A	PA_07
SPI2_SEL3	SPI2 Slave Select Output 3	C	PC_00
SPI2_SEL4	SPI2 Slave Select Output 4	D	PD_08
SPI2_SEL5	SPI2 Slave Select Output 5	A	PA_15
SPI2_SEL6	SPI2 Slave Select Output n	A	PA_10
SPI2_SEL7	SPI2 Slave Select Output n	B	PB_07
SPI2_SS	SPI2 Slave Select Input	B	PB_15
SYS_BMODE0	Boot Mode Control n	Not Muxed	SYS_BMODE0
SYS_BMODE1	Boot Mode Control n	Not Muxed	SYS_BMODE1
SYS_CLKIN0	Clock/Crystal Input	Not Muxed	SYS_CLKIN0
SYS_CLKOUT	Processor Clock Output	Not Muxed	SYS_CLKOUT
SYS_FAULT	Active-High Fault Output	Not Muxed	SYS_FAULT
SYS_HWRST	Processor Hardware Reset Control	Not Muxed	SYS_HWRST
SYS_RESOUT	Reset Output	Not Muxed	SYS_RESOUT
SYS_XTAL0	Crystal Output	Not Muxed	SYS_XTAL0
TM0_ACI0	TIMER0 Alternate Capture Input 0	A	PA_06
TM0_ACI1	TIMER0 Alternate Capture Input 1	A	PA_08
TM0_ACI2	TIMER0 Alternate Capture Input 2	C	PC_12
TM0_ACI3	TIMER0 Alternate Capture Input 3	C	PC_14
TM0_ACI4	TIMER0 Alternate Capture Input 4	C	PC_13
TM0_ACI5	TIMER0 Alternate Capture Input 5	Not Applicable	DAI_PB04_O
TM0_ACI6	TIMER0 Alternate Capture Input 6	Not Applicable	DAI_PB19_O
TM0_ACI7	TIMER0 Alternate Capture Input 7	Not Applicable	CNT0_TO
TM0_ACLK1	TIMER0 Alternate Clock 1	A	PA_00
TM0_ACLK2	TIMER0 Alternate Clock 2	C	PC_01
TM0_ACLK3	TIMER0 Alternate Clock 3	D	PD_09
TM0_ACLK4	TIMER0 Alternate Clock 4	C	PC_11
TM0_ACLK5	TIMER0 Alternate Clock 5	Not Applicable	DAI_PB03_O
TM0_ACLK6	TIMER0 Alternate Clock 6	Not Applicable	DAI_PB20_O
TM0_ACLK7	TIMER0 Alternate Clock 7	Not Applicable	SYS_CLKIN0
TM0_CLK	TIMER0 Clock	C	PC_03
TM0_TMR0	TIMER0 Timer 0	D	PD_02
TM0_TMR1	TIMER0 Timer 1	D	PD_03
TM0_TMR2	TIMER0 Timer 2	D	PD_04
TM0_TMR3	TIMER0 Timer 3	B	PB_01
TM0_TMR4	TIMER0 Timer 4	B	PB_03
TM0_TMR5	TIMER0 Timer 5	C	PC_15
TM0_TMR7	TIMER0 Timer 7	D	PD_07
TRACE0_CLK	TRACE0 Trace Clock	A	PA_00
TRACE0_D00	TRACE0 Trace Data	A	PA_01
TRACE0_D01	TRACE0 Trace Data	A	PA_02
TRACE0_D02	TRACE0 Trace Data	A	PA_03
TRACE0_D03	TRACE0 Trace Data	A	PA_04
TRACE0_D04	TRACE0 Trace Data	D	PD_10
TRACE0_D05	TRACE0 Trace Data	D	PD_11
TRACE0_D06	TRACE0 Trace Data	D	PD_12

Table 20. ADSP-SC57x/ADSP-2157x 176-Lead LQFP Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
TRACE0_D07	TRACE0 Trace Data 7	D	PD_13
TWI0_SCL	TWI0 Serial Clock	Not Muxed	TWI0_SCL
TWI0_SDA	TWI0 Serial Data	Not Muxed	TWI0_SDA
TWI1_SCL	TWI1 Serial Clock	Not Muxed	TWI1_SCL
TWI1_SDA	TWI1 Serial Data	Not Muxed	TWI1_SDA
TWI2_SCL	TWI2 Serial Clock	Not Muxed	TWI2_SCL
TWI2_SDA	TWI2 Serial Data	Not Muxed	TWI2_SDA
$\overline{\text{UART0\_CTS}}$	UART0 Clear to Send	D	PD_06
$\overline{\text{UART0\_RTS}}$	UART0 Request to Send	D	PD_05
$\overline{\text{UART0\_RX}}$	UART0 Receive	A	PA_06
$\overline{\text{UART0\_TX}}$	UART0 Transmit	A	PA_05
$\overline{\text{UART1\_CTS}}$	UART1 Clear to Send	D	PD_01
$\overline{\text{UART1\_RTS}}$	UART1 Request to Send	D	PD_00
$\overline{\text{UART1\_RX}}$	UART1 Receive	A	PA_08
$\overline{\text{UART1\_TX}}$	UART1 Transmit	A	PA_07
$\overline{\text{UART2\_CTS}}$	UART2 Clear to Send	A	PA_11
$\overline{\text{UART2\_RTS}}$	UART2 Request to Send	A	PA_10
$\overline{\text{UART2\_RX}}$	UART2 Receive	C	PC_13
$\overline{\text{UART2\_TX}}$	UART2 Transmit	C	PC_12

GPIO MULTIPLEXING FOR 176-LEAD LQFP

Table 21 through Table 24 identify the pin functions that are multiplexed on the general-purpose I/O pins of the 176-lead LQFP package.

Table 21. Signal Multiplexing for Port A

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PA_00	TRACE0_CLK				TM0_ACLK1
PA_01	TRACE0_D00				
PA_02	TRACE0_D01				
PA_03	TRACE0_D02				
PA_04	TRACE0_D03				
PA_05	UART0_TX				
PA_06	UART0_RX				TM0_ACI0
PA_07	UART1_TX	SPI2_SEL2			
PA_08	UART1_RX	ACM0_A0	SPI1_SEL3		TM0_ACI1
PA_09	ETH0_PTPPPS0	LP1_D6	SPI0_SEL4		
PA_10	ETH0_MDIO	UART2_RTS	SPI2_SEL6		
PA_11	ETH0_MDC	UART2_CTS			
PA_12	ETH0_RXD1				
PA_13	ETH0_RXD0				
PA_14	ETH0_RXD2	ACM0_A3	SPI1_SEL4		
PA_15	ETH0_RXD3	ACM0_T0	SPI2_SEL5		

Table 22. Signal Multiplexing for Port B

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PB_00	ETH0_RXCLK_REFCLK	C2_FLG0			
PB_01	ETH0_CRS	ACM0_A4	LP1_ACK	TM0_TMR3	
PB_02	ETH0_RXCTL_RXDV		SPI1_SEL5		
PB_03	ETH0_RXERR	MLB0_CLKOUT	LP1_CLK	TM0_TMR4	
PB_04	ETH0_TXCLK	MLB0_DAT			
PB_05	ETH0_TXD3	MLB0_SIG			
PB_06	ETH0_TXD2	MLB0_CLK			
PB_07	ETH0_TXD0		SPI2_SEL7		
PB_08	ETH0_TXD1				
PB_09	ETH0_TXCTL_TXEN				
PB_10	SPI2_MISO				
PB_11	SPI2_MOSI				
PB_12	SPI2_D2				
PB_13	SPI2_D3				
PB_14	SPI2_CLK				
PB_15	SPI2_SEL1				SPI2_SS

Table 23. Signal Multiplexing for Port C

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PC_00	SPI2_SEL3	SPI2_RDY			
PC_01	SPI0_CLK	PPI0_D08			TM0_ACLK2
PC_02	SPI0_MISO	PPI0_D09			
PC_03	SPI0_MOSI	PPI0_D10			TM0_CLK
PC_04	SPI0_SEL1	PPI0_D11			SPI0_SS
PC_05	SPI0_SEL2	PPI0_D06	SPI0_RDY		
PC_06	SPI0_SEL3	ETH0_COL	PPI0_FS3		
PC_07	SPI1_CLK				
PC_08	SPI1_MISO				
PC_09	SPI1_MOSI	C1_FLG2			
PC_10	SPI1_SEL1	C2_FLG2			SPI1_SS
PC_11	SPI1_SEL2	PPI0_CLK	SPI1_RDY		TM0_ACLK4
PC_12	CAN0_RX		UART2_TX		TM0_ACI2
PC_13	CAN0_TX		UART2_RX		TM0_ACI4
PC_14	CAN1_RX	PPI0_FS1	ACM0_A1	C2_FLG1	TM0_ACI3
PC_15	CAN1_TX	PPI0_FS2	ACM0_A2	TM0_TMR5	

Table 24. Signal Multiplexing for Port D

Signal Name	Multiplexed Function 0	Multiplexed Function 1	Multiplexed Function 2	Multiplexed Function 3	Multiplexed Function Input Tap
PD_00	C1_FLG0	UART1_RTS	CNT0_UD		
PD_01	C1_FLG1	UART1_CTS	TM0_TMR6		
PD_02	TM0_TMR0				
PD_03	TM0_TMR1	SPI0_SEL5			
PD_04	TM0_TMR2		SPI0_SEL6		
PD_05	SPI0_SEL7	C2_FLG3	UART0_RTS		
PD_06	SPI1_SEL7	C1_FLG3	UART0_CTS		
PD_07	SPI1_SEL6	CNT0_ZM	TM0_TMR7		
PD_08	ETH0_PTPPPS1	CNT0_DG	SPI2_SEL4		
PD_09	LP1_D7	PPI0_D07	HADC0_EOC_DOUT		TM0_ACLK3
PD_10	LP1_D0	PPI0_D00	TRACE0_D04		
PD_11	LP1_D1	PPI0_D01	TRACE0_D05		
PD_12	LP1_D2	PPI0_D02	TRACE0_D06		
PD_13	LP1_D3	PPI0_D03	TRACE0_D07		
PD_14	LP1_D4	PPI0_D04	ETH0_PTPAUXIN0		
PD_15	LP1_D5	PPI0_D05	ETH0_PTPAUXIN1		

ADSP-SC57x/ADSP-2157x DESIGNER QUICK REFERENCE

Table 25 provides a quick reference summary of pin related information for circuit board design. The columns in this table provide the following information:

- Signal Name: The Signal Name column in the table includes the signal name for every pin and (where applicable) the GPIO multiplexed pin function for every pin.
- Pin Type: The Type column in the table identifies the I/O type or supply type of the pin. The abbreviations used in this column are a (analog), s (supply), g (ground) and Input/Output/InOut.
- Driver Type: The Driver Type column in the table identifies the driver type used by the pin. The driver types are defined in the output drive currents section of this data sheet.
- Internal Termination: The Int Term column in the table specifies the termination present when the processor is not in the reset state.
- Reset Termination: The Reset Term column in the table specifies the termination present when the processor is in the reset state.
- Reset Drive: The Reset Drive column in the table specifies the active drive on the signal when the processor is in the reset state.
- Power Domain: The Power Domain column in the table specifies the power supply domain in which the signal resides.
- Description and Notes: The Description and Notes column in the table identifies any special requirements or characteristics for the signal. If no special requirements are listed the signal may be left unconnected if it is not used. Also, for multiplexed general-purpose I/O pins, this column identifies the functions available on the pin.

Table 25. ADSP-SC58x/ADSP-2158x Designer Quick Reference

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
DAI0_PIN01	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 1 Notes: No Notes
DAI0_PIN02	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 2 Notes: No Notes
DAI0_PIN03	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 3 Notes: No Notes
DAI0_PIN04	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 4 Notes: No Notes
DAI0_PIN05	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 5 Notes: No Notes
DAI0_PIN06	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 6 Notes: No Notes
DAI0_PIN07	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 7 Notes: No Notes
DAI0_PIN08	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 8 Notes: No Notes
DAI0_PIN09	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 9 Notes: No Notes
DAI0_PIN10	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 10 Notes: No Notes
DAI0_PIN11	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 11 Notes: No Notes
DAI0_PIN12	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 12 Notes: No Notes
DAI0_PIN13	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 13 Notes: No Notes
DAI0_PIN14	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 14 Notes: No Notes

Table 25. ADSP-SC58x/ADSP-2158x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
DAI0_PIN15	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 15 Notes: No Notes
DAI0_PIN16	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 16 Notes: No Notes
DAI0_PIN17	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 17 Notes: No Notes
DAI0_PIN18	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 18 Notes: No Notes
DAI0_PIN19	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 19 Notes: No Notes
DAI0_PIN20	InOut	A	PullDown	none	none	VDD_EXT	Desc: DAI0 Pin 20 Notes: No Notes
DMC0_A00	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 0 Notes: No Notes
DMC0_A01	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 1 Notes: No Notes
DMC0_A02	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 2 Notes: No Notes
DMC0_A03	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 3 Notes: No Notes
DMC0_A04	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 4 Notes: No Notes
DMC0_A05	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 5 Notes: No Notes
DMC0_A06	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 6 Notes: No Notes
DMC0_A07	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 7 Notes: No Notes
DMC0_A08	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 8 Notes: No Notes
DMC0_A09	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 9 Notes: No Notes
DMC0_A10	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 10 Notes: No Notes
DMC0_A11	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 11 Notes: No Notes
DMC0_A12	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 12 Notes: No Notes
DMC0_A13	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 13 Notes: No Notes
DMC0_A14	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 14 Notes: No Notes
DMC0_A15	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Address 15 Notes: No Notes
DMC0_BA0	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Bank Address Input 0 Notes: No Notes

Table 25. ADSP-SC58x/ADSP-2158x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
DMC0_BA1	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Bank Address Input 1 Notes: No Notes
DMC0_BA2	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Bank Address Input 2 Notes: No Notes
$\overline{\text{DMC0\_CAS}}$	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Column Address Strobe Notes: No Notes
DMC0_CK	Output	C	none	none	L	VDD_DMC	Desc: DMC0 Clock Notes: No Notes
DMC0_CKE	Output	B	none	none	L	VDD_DMC	Desc: DMC0 Clock enable Notes: No Notes
$\overline{\text{DMC0\_CK}}$	Output	C	none	none	L	VDD_DMC	Desc: DMC0 Clock (complement) Notes: No Notes
$\overline{\text{DMC0\_CS0}}$	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Chip Select 0 Notes: No Notes
DMC0_DQ00	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 0 Notes: No Notes
DMC0_DQ01	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 1 Notes: No Notes
DMC0_DQ02	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 2 Notes: No Notes
DMC0_DQ03	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 3 Notes: No Notes
DMC0_DQ04	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 4 Notes: No Notes
DMC0_DQ05	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 5 Notes: No Notes
DMC0_DQ06	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 6 Notes: No Notes
DMC0_DQ07	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 7 Notes: No Notes

Table 25. ADSP-SC58x/ADSP-2158x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
DMC0_DQ08	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 8 Notes: No Notes
DMC0_DQ09	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 9 Notes: No Notes
DMC0_DQ10	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 10 Notes: No Notes
DMC0_DQ11	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 11 Notes: No Notes
DMC0_DQ12	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 12 Notes: No Notes
DMC0_DQ13	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 13 Notes: No Notes
DMC0_DQ14	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 14 Notes: No Notes
DMC0_DQ15	InOut	B	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data 15 Notes: No Notes
DMC0_LDM	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Data Mask for Lower Byte Notes: No Notes
DMC0_LDQS	InOut	C	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data Strobe for Lower Byte Notes: External weak pull-down required in LPDDR mode.
$\overline{\text{DMC0\_LDQS}}$	InOut	C	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data Strobe for Lower Byte (complement) Notes: No Notes
DMC0_ODT	Output	B	none	none	none	VDD_DMC	Desc: DMC0 On-die termination Notes: No Notes
$\overline{\text{DMC0\_RAS}}$	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Row Address Strobe Notes: No Notes
$\overline{\text{DMC0\_RESET}}$	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Reset (DDR3 only) Notes: No Notes

Table 25. ADSP-SC58x/ADSP-2158x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
DMC0_RZQ	a	B	none	none	none	VDD_DMC	Desc: DMC0 External calibration resistor connection Notes: Applicable for DDR2 and DDR3 only. 34 Ohm external pull-down must be added.
DMC0_UDM	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Data Mask for Upper Byte Notes: No Notes
DMC0_UDQS	InOut	C	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data Strobe for Upper Byte Notes: External weak pull-down required in LPDDR mode.
$\overline{\text{DMC0\_UDQS}}$	InOut	C	Internal logic ensures that input signal does not float	none	none	VDD_DMC	Desc: DMC0 Data Strobe for Upper Byte (complement) Notes: No Notes
DMC0_VREF	a		none	none	none	VDD_DMC	Desc: DMC0 Voltage Reference Notes: No Notes
$\overline{\text{DMC0\_WE}}$	Output	B	none	none	none	VDD_DMC	Desc: DMC0 Write Enable Notes: No Notes
GND	g		none	none	none		Desc: Ground Notes: No Notes
HADC0_VIN0	a	NA	none	none	none	VDD_HADC	Desc: HADC0 Analog Input at channel 0 Notes: Connect to GND if not used.
HADC0_VIN1	a	NA	none	none	none	VDD_HADC	Desc: HADC0 Analog Input at channel 1 Notes: Connect to GND if not used.
HADC0_VIN2	a	NA	none	none	none	VDD_HADC	Desc: HADC0 Analog Input at channel 2 Notes: Connect to GND if not used.
HADC0_VIN3	a	NA	none	none	none	VDD_HADC	Desc: HADC0 Analog Input at channel 3 Notes: Connect to GND if not used.
HADC0_VIN4	a	NA	none	none	none	VDD_HADC	Desc: HADC0 Analog Input at channel 4 Notes: Connect to GND if not used.
HADC0_VIN5	a	NA	none	none	none	VDD_HADC	Desc: HADC0 Analog Input at channel 5 Notes: Connect to GND if not used.
HADC0_VIN6	a	NA	none	none	none	VDD_HADC	Desc: HADC0 Analog Input at channel 6 Notes: Connect to GND if not used.

Table 25. ADSP-SC58x/ADSP-2158x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
HADC0_VIN7	a	NA	none	none	none	VDD_HADC	Desc: HADC0 Analog Input at channel 7 Notes: Connect to GND if not used.
HADC0_VREFN	s	NA	none	none	none	VDD_HADC	Desc: HADC0 Ground Reference for ADC Notes: No Notes
HADC0_VREFP	s	NA	none	none	none	VDD_HADC	Desc: HADC0 External Reference for ADC Notes: No Notes
JTG_TCK	Input		PullUp	none	none	VDD_EXT	Desc: JTAG Clock Notes: No Notes
JTG_TDI	Input		PullUp	none	none	VDD_EXT	Desc: JTAG Serial Data In Notes: No Notes
JTG_TDO	Output	A	none	none	none	VDD_EXT	Desc: JTAG Serial Data Out Notes: No Notes
JTG_TMS	InOut	A	PullUp	none	none	VDD_EXT	Desc: JTAG Mode Select Notes: No Notes
$\overline{\text{JTG\_TRST}}$	Input		PullDown	none	none	VDD_EXT	Desc: JTAG Reset Notes: No Notes
MLB0_CLKN	Input	TBD	Internal logic ensures that input signal does not float	none	none	VDD_EXT	Desc: MLB0 Differential Clock (-) Notes: No Notes
MLB0_CLKP	Input	TBD	Internal logic ensures that input signal does not float	none	none	VDD_EXT	Desc: MLB0 Differential Clock (+) Notes: No Notes
MLB0_DATN	InOut	TBD	Internal logic ensures that input signal does not float	none	none	VDD_EXT	Desc: MLB0 Differential Data (-) Notes: No Notes
MLB0_DATP	InOut	TBD	Internal logic ensures that input signal does not float	none	none	VDD_EXT	Desc: MLB0 Differential Data (+) Notes: No Notes
MLB0_SIGN	InOut	TBD	Internal logic ensures that input signal does not float	none	none	VDD_EXT	Desc: MLB0 Differential Signal (-) Notes: No Notes
MLB0_SIGP	InOut	TBD	Internal logic ensures that input signal does not float	none	none	VDD_EXT	Desc: MLB0 Differential Signal (+) Notes: No Notes
PA_00	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 0 Notes: No Notes
PA_01	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 1 Notes: No Notes
PA_02	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 2 Notes: No Notes

Table 25. ADSP-SC58x/ADSP-2158x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
PA_03	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 3 Notes: No Notes
PA_04	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 4 Notes: No Notes
PA_05	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 5 Notes: No Notes
PA_06	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 6 Notes: No Notes
PA_07	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 7 Notes: No Notes
PA_08	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 8 Notes: No Notes
PA_09	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 9 Notes: No Notes
PA_10	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 10 Notes: No Notes
PA_11	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 11 Notes: No Notes
PA_12	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 12 Notes: No Notes
PA_13	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 13 Notes: No Notes
PA_14	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 14 Notes: No Notes
PA_15	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTA Position 15 Notes: No Notes
PB_00	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 0 Notes: No Notes
PB_01	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 1 Notes: No Notes
PB_02	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 2 Notes: No Notes
PB_03	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 3 Notes: No Notes
PB_04	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 4 Notes: No Notes
PB_05	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 5 Notes: No Notes
PB_06	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 6 Notes: No Notes
PB_07	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 7 Notes: No Notes
PB_08	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 8 Notes: No Notes
PB_09	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 9 Notes: No Notes
PB_10	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 10 Notes: No Notes

Table 25. ADSP-SC58x/ADSP-2158x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
PB_11	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 11 Notes: No Notes
PB_12	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 12 Notes: No Notes
PB_13	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 13 Notes: No Notes
PB_14	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 14 Notes: No Notes
PB_15	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTB Position 15 Notes: No Notes
PC_00	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 0 Notes: No Notes
PC_01	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 1 Notes: No Notes
PC_02	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 2 Notes: No Notes
PC_03	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 3 Notes: No Notes
PC_04	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 4 Notes: No Notes
PC_05	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 5 Notes: No Notes
PC_06	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 6 Notes: No Notes
PC_07	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 7 Notes: No Notes
PC_08	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 8 Notes: No Notes
PC_09	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 9 Notes: No Notes
PC_10	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 10 Notes: No Notes
PC_11	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 11 Notes: No Notes
PC_12	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 12 Notes: No Notes
PC_13	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 13 Notes: No Notes
PC_14	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 14 Notes: No Notes
PC_15	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTC Position 15 Notes: No Notes
PD_00	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 0 Notes: No Notes
PD_01	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 1 Notes: No Notes
PD_02	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 2 Notes: No Notes

Table 25. ADSP-SC58x/ADSP-2158x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
PD_03	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 3 Notes: No Notes
PD_04	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 4 Notes: No Notes
PD_05	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 5 Notes: No Notes
PD_06	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 6 Notes: No Notes
PD_07	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 7 Notes: No Notes
PD_08	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 8 Notes: No Notes
PD_09	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 9 Notes: No Notes
PD_10	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 10 Notes: No Notes
PD_11	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 11 Notes: No Notes
PD_12	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 12 Notes: No Notes
PD_13	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 13 Notes: No Notes
PD_14	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 14 Notes: No Notes
PD_15	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTD Position 15 Notes: No Notes
PE_00	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 0 Notes: No Notes
PE_01	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 1 Notes: No Notes
PE_02	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 2 Notes: No Notes
PE_03	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 3 Notes: No Notes
PE_04	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 4 Notes: No Notes
PE_05	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 5 Notes: No Notes
PE_06	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 6 Notes: No Notes
PE_07	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 7 Notes: No Notes
PE_08	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 8 Notes: No Notes
PE_09	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 9 Notes: No Notes
PE_10	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 10 Notes: No Notes

Table 25. ADSP-SC58x/ADSP-2158x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
PE_11	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTE Position 11 Notes: No Notes
PE_12	InOut	A	PullDown/Programmable PullUp	none	none	VDD_EXT	Desc: PORTE Position 12 Notes: No Notes
PE_13	InOut	A	PullDown/Programmable PullUp	none	none	VDD_EXT	Desc: PORTE Position 13 Notes: No Notes
PE_14	InOut	A	PullDown/Programmable PullUp	none	none	VDD_EXT	Desc: PORTE Position 14 Notes: No Notes
PE_15	InOut	A	PullDown/Programmable PullUp	none	none	VDD_EXT	Desc: PORTE Position 15 Notes: No Notes
PF_00	InOut	A	PullDown/Programmable PullUp	none	none	VDD_EXT	Desc: PORTF Position 0 Notes: No Notes
PF_01	InOut	A	PullDown/Programmable PullUp	none	none	VDD_EXT	Desc: PORTF Position 1 Notes: No Notes
PF_02	InOut	A	PullDown/Programmable PullUp	none	none	VDD_EXT	Desc: PORTF Position 2 Notes: No Notes
PF_03	InOut	A	PullDown/Programmable PullUp	none	none	VDD_EXT	Desc: PORTF Position 3 Notes: No Notes
PF_04	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTF Position 4 Notes: No Notes
PF_05	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTF Position 5 Notes: No Notes
PF_06	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTF Position 6 Notes: No Notes
PF_07	InOut	A	PullDown/Programmable PullUp	none	none	VDD_EXT	Desc: PORTF Position 7 Notes: No Notes
PF_08	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTF Position 8 Notes: No Notes
PF_09	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTF Position 9 Notes: No Notes
PF_10	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTF Position 10 Notes: No Notes
PF_11	InOut	A	PullDown	none	none	VDD_EXT	Desc: PORTF Position 11 Notes: No Notes
SYS_BMODE0	Input	NA	PullDown	none	none	VDD_EXT	Desc: Boot Mode Control n Notes: No Notes
SYS_BMODE1	Input	NA	PullDown	none	none	VDD_EXT	Desc: Boot Mode Control n Notes: No Notes
SYS_BMODE2	Input	NA	PullDown	none	none	VDD_EXT	Desc: Boot Mode Control n Notes: No Notes
SYS_CLKIN0	a	NA	none	none	none	VDD_EXT	Desc: Clock/Crystal Input Notes: No Notes
SYS_CLKIN1	a	NA	none	none	none	VDD_EXT	Desc: Clock/Crystal Input Notes: No Notes
SYS_CLKOUT	a	A	none	none	none		Desc: Processor Clock Output Notes: No Notes

Table 25. ADSP-SC58x/ADSP-2158x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
SYS_FAULT	InOut	A	none	none	none		Desc: Active-High Fault Output Notes: External pull-down required to keep signal in de-asserted state.
$\overline{\text{SYS\_FAULT}}$	InOut	A	none	none	none		Desc: Active-Low Fault Output Notes: External pull-up required to keep signal in de-asserted state.
$\overline{\text{SYS\_HWRST}}$	Input	NA	none	none	none	VDD_EXT	Desc: Processor Hardware Reset Control Notes: No Notes
$\overline{\text{SYS\_RESOUT}}$	Output	A	none	none	L	VDD_EXT	Desc: Reset Output Notes: No Notes
SYS_XTAL0	a	NA	none	none	none	VDD_EXT	Desc: Crystal Output Notes: No Notes
SYS_XTAL1	a	NA	none	none	none	VDD_EXT	Desc: Crystal Output Notes: No Notes
TWI0_SCL	InOut	D	none	none	none	VDD_EXT	Desc: TWI0 Serial Clock Notes: Add external pull-up if used. Can be pulled low when not used.
TWI0_SDA	InOut	D	none	none	none	VDD_EXT	Desc: TWI0 Serial Data Notes: Add external pull-up if used. Can be pulled low when not used.
TWI1_SCL	InOut	D	none	none	none	VDD_EXT	Desc: TWI1 Serial Clock Notes: Add external pull-up if used. Can be pulled low when not used.
TWI1_SDA	InOut	D	none	none	none	VDD_EXT	Desc: TWI1 Serial Data Notes: Add external pull-up if used. Can be pulled low when not used.
TWI2_SCL	InOut	D	none	none	none	VDD_EXT	Desc: TWI2 Serial Clock Notes: Add external pull-up if used. Can be pulled low when not used.
TWI2_SDA	InOut	D	none	none	none	VDD_EXT	Desc: TWI2 Serial Data Notes: Add external pull-up if used. Can be pulled low when not used.
USB0_DM	InOut	F	none	none	none	VDD_USB	Desc: USB0 Data - Notes: Add external pull-down if not used.
USB0_DP	InOut	F	none	none	none	VDD_USB	Desc: USB0 Data + Notes: Add external pull-down if not used.
USB0_ID	InOut		none	none	none	VDD_USB	Desc: USB0 OTG ID Notes: Connect to GND when USB is not used.

Table 25. ADSP-SC58x/ADSP-2158x Designer Quick Reference (Continued)

Signal Name	Type	Driver Type	Int Term	Reset Term	Reset Drive	Power Domain	Description and Notes
USB0_VBC	InOut	E	none	none	none	VDD_USB	Desc: USB0 VBUS Control Notes: Add external pull-down if not used.
USB0_VBUS	InOut	G	none	none	none	VDD_USB	Desc: USB0 Bus Voltage Notes: Connect to GND when USB is not used.
USB_CLKIN	a		none	none	none		Desc: USB0/USB1 Clock/Crystal Input Notes: Connect to GND when USB is not used.
USB_XTAL	a		none	none	none		Desc: USB0/USB1 Crystal Notes: No Notes
VDD_DMC	s		none	none	none		Desc: DMC VDD Notes: No Notes
VDD_EXT	s		none	none	none		Desc: External Voltage Domain Notes: No Notes
VDD_HADC	s		none	none	none		Desc: HADC VDD Notes: No Notes
VDD_INT	s		none	none	none		Desc: Internal Voltage Domain Notes: No Notes
VDD_USB	s		none	none	none		Desc: USB VDD Notes: Connect to VDD_EXT when USB is not used.

SPECIFICATIONS

For information about product specifications please contact your Analog Devices, Inc. representative.

OPERATING CONDITIONS

Parameter[1]		Conditions	Min	Nominal	Max	Unit
V_{DD_INT}	Internal (Core) Supply Voltage	CCLK ≤ 450 MHz	1.05	1.1	1.15	V
V_{DD_EXT}	External (I/O) Supply Voltage		3.13	3.3	3.47	V
V_{DD_HADC}	Analog Power Supply Voltage		3.13	3.3	3.47	V
V_{DD_DMC}[2]	DDR2/LPDDR Controller Supply Voltage		1.7	1.8	1.9	V
	DDR3 Controller Supply Voltage		1.425	1.5	1.575	V
V_{DD_USB}[3]	USB Supply Voltage		3.13	3.3	3.47	V
V_{DDR_VREF}	DDR2 Reference Voltage		0.49 × V_{DD_DMC}	0.50 × V_{DD_DMC}	0.51 × V_{DD_DMC}	V
V_{HADC_REF}[4]	HADC Reference Voltage		2.5	3.30	V_{DD_HADC}	V
V_{IH}[5]	High Level Input Voltage	V_{DD_EXT} = Maximum	2.0			V
V_{IL}[5]	Low Level Input Voltage	V_{DD_EXT} = Minimum			0.8	V
$V_{IL\_DDR2/3}$[6]	Low Level Input Voltage	V_{DD_DMC} = Minimum			V_{REF} – 0.25	V
$V_{IH\_DDR2/3}$[6]	High Level Input Voltage	V_{DD_DMC} = Maximum	V_{REF} + 0.25			V
V_{IL_LPDDR}[7]	Low Level Input Voltage	V_{DD_DMC} = Minimum			0.2 × V_{DD_DMC}	V
V_{IH_LPDDR}[7]	High Level Input Voltage	V_{DD_DMC} = Maximum	0.8 × V_{DD_DMC}			V
T_J	Junction Temperature 400-Ball CSP_BGA	$T_{AMBIENT}$ 0°C to +70°C	0		95	°C
T_J	Junction Temperature 400-Ball CSP_BGA	$T_{AMBIENT}$ –40°C to +100°C	–40		125	°C
T_J	Junction Temperature 176-Lead LQFP-EP	$T_{AMBIENT}$ 0°C to +70°C	0		90	°C
T_J	Junction Temperature 176-Lead LQFP-EP	$T_{AMBIENT}$ –40°C to +103°C	–40		125	°C
AUTOMOTIVE USE ONLY						
T_J[8]	Junction Temperature 400-Ball CSP_BGA (Automotive Grade)	$T_{AMBIENT}$ –40°C to +105°C	–40		130	°C
T_J[8]	Junction Temperature 176-Lead LQFP-EP (Automotive Grade)	$T_{AMBIENT}$ –40°C to +105°C	–40		126	°C

[1] Specifications subject to change without notice.
[2] Applies to DDR2/DDR3/LPDDR signals.
[3] If not used, V_{DD_USB} should be connected to 3.3V.
[4] V_{HADC_VREF} should always be less than V_{DD_HADC}.
[5] Parameter value applies to all input and bidirectional pins except all the TWI, DMC, USB, and MLB pins.
[6] This parameter applies to all DMC0/1 signals in DDR2/DDR3 mode. V_{REF} is the voltage applied to pin V_{REF_DMC}, nominally V_{DD_DMC}/2.
[7] This parameter applies to DMC0/1 signals in LPDDR mode.
[8] Automotive temperature grade product only. Contact Analog Devices for more information.

Table 26. TWI_VSEL Selections and V_{DD_EXT}/V_{BUSTWI}

	V_{DD_EXT} Nominal	V_{BUSTWI} Min	V_{BUSTWI} Nom	V_{BUSTWI} Max	Unit
TWI000[1]	3.30	3.13	3.30	3.47	V
TWI100	3.30	4.75	5.00	5.25	V

[1] Designs must comply with the V_{DD_EXT} and V_{BUSTWI} voltages specified for the default TWI_DT setting for correct JTAG boundary scan operation during reset.

Clock Related Operating Conditions

Table 27 describes the core clock, system clock, and peripheral clock timing requirements. The data presented in the tables applies to all speed grades except where expressly noted.

Table 27. Clock Operating Conditions

Parameter		Restriction	Min	Typ	Max	Unit
f_{CCLK}	Core Clock Frequency	$f_{CCLK} \geq f_{SYSCLK}$			450	MHz
f_{SYSCLK}	SYSCLK Frequency				225	MHz
f_{SCLK0}	SCLK0 Frequency[1]	$f_{SYSCLK} \geq f_{SCLK0}$	30		112.5	MHz
f_{SCLK1}	SCLK1 Frequency	$f_{SYSCLK} \geq f_{SCLK1}$			112.5	MHz
f_{DCLK}	LPDDR Clock Frequency				200	MHz
f_{DCLK}	DDR2 Clock Frequency				400	MHz
f_{DCLK}	DDR3 Clock Frequency				450	MHz
f_{OCLK}	Output Clock Frequency				TBD	MHz
$f_{SYS\_CLKOUTJ}$	SYS_CLKOUT Period Jitter[2, 3]			±1		%
$f_{PCLKPROG}$	Programmed PPI Clock When Transmitting Data and Frame Sync				56.25	MHz
$f_{PCLKPROG}$	Programmed PPI Clock When Receiving Data or Frame Sync				45	MHz
$f_{PCLKEXT}$	External PPI Clock When Receiving Data and Frame Sync[4, 5]	$f_{PCLKEXT} \leq f_{SCLK0}$			56.25	MHz
$f_{PCLKEXT}$	External PPI Clock Transmitting Data or Frame Sync[4, 5]	$f_{PCLKEXT} \leq f_{SCLK0}$			45	MHz
$f_{LCLKTPROG}$	Programmed Link Port Transmit Clock				112.5	MHz
$f_{LCLKREXT}$	External Link Port Receive Clock[4, 5]	$f_{LCLKEXT} \leq f_{SCLK0}$			112.5	MHz
$f_{SPTCLKPROG}$	Programmed SPT Clock When Transmitting Data and Frame Sync				56.25	MHz
$f_{SPTCLKPROG}$	Programmed SPT Clock When Receiving Data or Frame Sync				28.125	MHz
$f_{SPTCLKEXT}$	External SPT Clock When Receiving Data and Frame Sync[4, 5]	$f_{SPTCLKEXT} \leq f_{SCLK0}$			56.25	MHz
$f_{SPTCLKEXT}$	External SPT Clock Transmitting Data or Frame Sync[4, 5]	$f_{SPTCLKEXT} \leq f_{SCLK0}$			28.125	MHz
$f_{SPICLKPROG}$	Programmed SPI2 Clock When Transmitting Data				75	MHz
	Programmed SPI0, SPI1 Clock When Transmitting Data				56.25	MHz
$f_{SPICLKPROG}$	Programmed SPI Clock When Receiving Data				75	MHz
$f_{SPICLKEXT}$	External SPI2 Clock When Receiving Data[4, 5]	$f_{SPICLKEXT} \leq f_{SCLK1}$			75	MHz
	External SPI0, SPI1 Clock When Receiving Data[4, 5]	$f_{SPICLKEXT} \leq f_{SCLK0}$			56.25	MHz
$f_{SPICLKEXT}$	External SPI2 Clock When Transmitting Data[4, 5]	$f_{SPICLKEXT} \leq f_{SCLK1}$			45	MHz
	External SPI0, SPI1 Clock When Transmitting Data[4,5]	$f_{SPICLKEXT} \leq f_{SCLK0}$			56.25	MHz
$f_{ACLKPROG}$	Programmed ACM Clock				56.25	MHz

[1] The minimum frequency for SCLK0 applies only when the USB is used.

[2] SYS_CLKOUT jitter is dependent on the application system design including pin switching activity, board layout, and the jitter characteristics of the SYS_CLKIN source. Due to the dependency on these factors the measured jitter may be higher or lower than this typical specification for each end application.

[3] The value in the Typ field is the percentage of the SYS_CLKOUT period.

[4] The maximum achievable frequency for any peripheral in external clock mode is dependent on being able to meet the setup and hold times in the AC timing specifications section for that peripheral. Pay particular attention to setup and hold times for VDD_EXT = 1.8 V which may preclude the maximum frequency listed here.

[5] The peripheral external clock frequency must also be less than or equal to the f_{SCLK} (f_{SCLK0} or f_{SCLK1}) that clocks the peripheral.

Table 28. Phase-Locked Loop Operating Conditions

Parameter		Min	Max	Unit
f_{PLLCLK}	PLL Clock Frequency	250	900	MHz

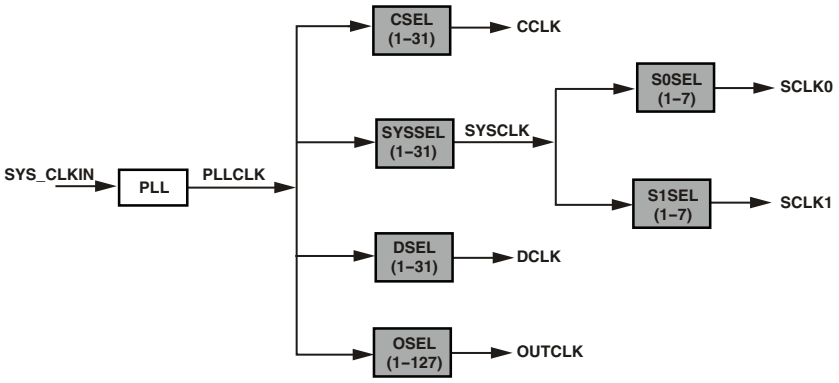


Figure 7. Clock Relationships and Divider Values

ELECTRICAL CHARACTERISTICS

Parameter[1]	Description	Conditions	450 MHz Min	450 MHz Typ	450 MHz Max	Unit
V_{OH}[2]	High Level Output Voltage	@ V_{DD_EXT} = Min, I_{OH} = –1.0 mA[3]	2.4			V
V_{OL}[2]	Low Level Output Voltage	@ V_{DD_EXT} = Min, I_{OL} = 1.0 mA[3]			0.4	V
V_{OH_DDR2}[4]	High Level Output Voltage for DDR2 DS = 40 ohm	@ V_{DD_DDR} = Min, I_{OH} = –7.0 mA	1.38			V
V_{OL_DDR2}[4]	Low Level Output Voltage for DDR2 DS = 40 ohm	@ V_{DD_DDR} = Min, I_{OL} = 7.0 mA			0.32	V
V_{OH_DDR2}[4]	High Level Output Voltage for DDR2 DS = 60 ohm	@ V_{DD_DDR} = Min, I_{OH} = –5.0 mA	1.38			V
V_{OL_DDR2}[4]	Low Level Output Voltage for DDR2 DS = 60 ohm	@ V_{DD_DDR} = Min, I_{OL} = 5.0 mA			0.32	V
V_{OH_DDR3}[5]	High Level Output Voltage for DDR3 DS = 40 ohm	@ V_{DD_DDR} = Min, I_{OH} = –7.0 mA	1.105			V
V_{OL_DDR3}[5]	Low Level Output Voltage for DDR3 DS = 60 ohm	@ V_{DD_DDR} = Min, I_{OL} = 5.0 mA			0.32	V
V_{OH_DDR3}[5]	High Level Output Voltage for DDR3 DS = 60 ohm	@ V_{DD_DDR} = Min, I_{OH} = –5.0 mA	1.105			V
V_{OL_DDR3}[5]	Low Level Output Voltage for DDR3 DS = 60 ohm	@ V_{DD_DDR} = Min, I_{OL} = 5.0 mA			0.32	V
V_{OH_LPDDR}[6]	High Level Output Voltage for LPDDR	@ V_{DD_DDR} = Min, I_{OH} = –9.0 mA	1.38			V
V_{OL_LPDDR}[6]	Low Level Output Voltage for LPDDR	@ V_{DD_DDR} = Min, I_{OL} = 9.0 mA			0.32	V
I_{IH}[7, 8]	High Level Input Current	@ V_{DD_EXT} = Max, V_{IN} = V_{DD_EXT} Max			10	µA
I_{IL}[7]	Low Level Input Current	@ V_{DD_EXT} = Max, V_{IN} = 0 V			10	µA
I_{IL_PU}[8]	Low Level Input Current Pull-up	@ V_{DD_EXT} = Max, V_{IN} = 0 V			200	µA
I_{IH_PD}[9]	High Level Input Current Pull-down	@ V_{DD_EXT} = Max, V_{IN} = 0 V			200	µA
I_{OZH}[10]	Three-State Leakage Current	@ V_{DD_EXT}/V_{DD_DDR} = Max, V_{IN} = V_{DD_EXT}/V_{DD_DDR} Max			10	µA
I_{OZL}[10]	Three-State Leakage Current	@ V_{DD_EXT}/V_{DD_DDR} = Max, V_{IN} = 0 V			10	µA
I_{DD_INT}[11]	Supply Current (Internal)	f_{CCLK} > 0 MHz			TBD	mA
C_{IN}[12]	Input Capacitance	T_{CASE} = 25°C			5	pF
I_{DD_IDLE}	V_{DD_INT} Current in Idle	f_{CCLK} = 450 MHz ASF_{SHARC1} = 0.27 ASF_{SHARC2} = 0.27 ASF_{A5} = 0.07 f_{SYSCLK} = 225 MHz $f_{SCLK0/1}$ = 112.5 MHz (Other clocks are disabled) No Peripheral or DMA activity T_J = 25°C V_{DD_INT} = 1.1 V		435		mA

Parameter[1]	Description	Conditions	450 MHz Min	450 MHz Typ	450 MHz Max	Unit
I_{DD_TYP}	V_{DD_INT} Current	f_{CCLK} = 450 MHz ASF_{SHARC1} = 1.0 ASF_{SHARC2} = 1.0 ASF_{A5} = 0.64 f_{SYSCLK} = 225 MHz $f_{SCLK0/1}$ = 112.5 MHz (Other clocks are disabled) FFT accelerator operating at $f_{SYSCLK/4}$ DMA data rate = 500 MB/s T_J = 25°C V_{DD_INT} = 1.1 V		1100		mA
I_{DD_INT}	V_{DD_INT} Current	f_{CCLK} > 0 MHz $f_{SCLK0/1} \geq 0$ MHz			See I_{DDINT_TOT} equation	mA

[1] Specifications subject to change without notice.
[2] Applies to all output and bidirectional pins except TWI, DMC, USB, PCIe, and MLB.
[3] See Output Drive Currents for typical drive current capabilities.
[4] Applies to all DMC output and bidirectional signals in DDR2 mode.
[5] Applies to all DMC output and bidirectional signals in DDR3 mode.
[6] Applies to all DMC output and bidirectional signals in LPDDR mode.
[7] Applies to input pins: SYS_BMODE0-2, SYS_CLKIN0, SYS_CLKIN1, $\overline{\text{SYS\_HWRST}}$, JTG_TDI, JTG_TMS, and USB0_CLKIN.
[8] Applies to input pins with internal pull-ups: JTG_TDI, JTG_TMS, and JTG_TCK.
[9] Applies to signals: JTAG_TRST, USB0_VBUS, USB1_VBUS.
[10] Applies to signals: PA0-15, PB0-15, PC0-15, PD0-15, PE0-15, PF0-15, PG0-5, DAI0_PINx, DMC0_DQx, DMC0_LDQS, DMC0_UDQS, $\overline{\text{DMC0\_LDQS}}$, $\overline{\text{DMC0\_UDQS}}$, SYS_FAULT, $\overline{\text{SYS\_FAULT}}$, JTG_TDO, USB0_ID, USBx_DM, USBx_DP, and USBx_VBC.
[11] See Engineer-to-Engineer Note EE-TBD "Estimating Power Dissipation for ADSP-215xx SHARC Processors" for further information.
[12] Applies to all signal pins.

Total Internal Power Dissipation

Total power dissipation has two components:

1. Static, including leakage current
2. Dynamic, due to transistor switching characteristics for each clock domain

Many operating conditions can also affect power dissipation, including temperature, voltage, operating frequency, and processor activity. The following equation describes the internal current consumption.

$I_{DDINT\_TOT} = I_{DDINT\_STATIC} + I_{DDINT\_CCLK\_SHARC1\_DYN} + I_{DDINT\_CCLK\_SHARC2\_DYN} + I_{DDINT\_CCLK\_A5\_DYN} + I_{DDINT\_DCLK\_DYN} + I_{DDINT\_SYSCLK\_DYN} + I_{DDINT\_SCLK0\_DYN} + I_{DDINT\_SCLK1\_DYN} + I_{DDINT\_OCLK\_DYN} + I_{DDINT\_ACCL\_DYN} + I_{DDINT\_USB\_DYN} + I_{DDINT\_MLB\_DYN} + I_{DDINT\_GIGE\_DYN} + I_{DDINT\_DMA\_DR\_DYN}$

I_{DDINT_STATIC} is the only item present that is part of the static power dissipation component. I_{DDINT_STATIC} is specified as a function of voltage (V_{DD_INT}) and temperature (see Table 29).

There are 13 different items that contribute to the dynamic power dissipation. These components fall into four broad categories: application-dependent currents, clock currents, currents from high-speed peripheral operation, and data transmission currents.

Table 29. Static Current—I_{DDINT_STATIC} (mA)

	Voltage (V_{DD_INT})				
T_J (°C)	1.00	1.05	1.10	1.15	1.20
–40	3	4	5	6	8
–20	7	8	10	12	14
0	14	16	19	23	27
25	30	35	41	48	56
40	48	55	64	74	86
55	73	84	97	112	128
70	111	127	145	166	189
85	166	188	213	242	275
100	239	269	304	343	388
105	268	302	340	384	433
115	337	378	425	478	538
125	426	477	534	599	672
133	502	560	625	700	784

Application-Dependent Current

The application-dependent currents include the dynamic current in the core clock domain of the two SHARC+ cores and the ARM Cortex-A5 core and accelerator currents.

Core clock (CCLK) use is subject to an activity scaling factor (ASF) that represents application code running on the processor cores (Table 30 andTable 31). The ASF is combined with the CCLK frequency and V_{DD_INT} dependent data in Table 32 to calculate this portion.

$I_{DDINT\_CCLK\_SHARC1\_DYN}$ = Table 32 × ASF_{SHARC1}
$I_{DDINT\_CCLK\_SHARC2\_DYN}$ = Table 32 × ASF_{SHARC2}
$I_{DDINT\_CCLK\_A5\_DYN}$ = Table 33 × ASF_{A5}

Table 30. Activity Scaling Factors for SHARC+ Core1 and Core2 (ASF_{SHARC1} and ASF_{SHARC2})

I_{DDINT} Power Vector	ASF
$I_{DD-IDLE}$	0.27
I_{DD-NOP}	0.51
I_{DD-TYP_3070}	0.72
I_{DD-TYP_5050}	0.86
I_{DD-TYP_7030}	1.00
$I_{DD-PEAK\_100}$	1.13

Table 31. Activity Scaling Factors for ARM Cortex-A5 Core (ASF_{A5})

I_{DDINT} Power Vector	ASF
$I_{DD-IDLE}$	0.07
$I_{DD-DHRYSTONE}$	0.64
$I_{DD-PEAK}$	1.26

Table 32. CCLK Dynamic Current per SHARC+ Core (mA, with ASF = 1.00)

	Voltage (V_{DD_INT})				
f_{CCLK} (MHz)	1.000	1.050	1.100	1.150	1.200
450	288	302.4	316.8	331.2	345.6
400	256	268.8	281.6	294.4	307.2
350	224	235.2	246.4	257.6	268.8
300	192	201.6	211.2	220.8	230.4
250	160	168	176	184	192
200	128	134.4	140.8	147.2	153.6
150	96	100.8	105.6	110.4	115.2
100	64	67.2	70.4	73.6	76.8

Table 33. CCLK Dynamic Current per ARM Cortex-A5 Core (mA, with ASF = 1.00)

	Voltage (V_{DD_INT})				
f_{CCLK} (MHz)	1.000	1.050	1.100	1.150	1.200
450	49.5	52.0	54.5	56.9	59.4
400	44.0	46.2	48.4	50.6	52.8
350	38.5	40.4	42.4	44.3	46.2
300	33.0	34.7	36.3	38.0	39.6
250	27.5	28.9	30.3	31.6	33.0
200	22.0	23.1	24.2	25.3	26.4
150	16.5	17.3	18.2	19.0	19.8
100	11.0	11.6	12.1	12.7	13.2

The following equation is used to compute the power dissipation when the FFT accelerator is used:

$I_{DDINT\_ACCL\_DYN}$ (mA) = ASF_{ACCL} × f_{SYSCLK} (MHz) × V_{DD_INT} (V)

Table 34. Activity Scaling Factors for FFT Accelerator (ASF_{A5})

I_{DDINT} Power Vector	ASF_{ACCL}
Unused	0.0
$I_{DD-TYPP}$	0.4

Clock Current

The dynamic clock currents provide the total power dissipated by all transistors switching in the clock paths. The power dissipated by each clock domain is dependent on voltage (V_{DD_INT}), operating frequency and a unique scaling factor.

$I_{DDINT\_SYSCLK\_DYN}$ (mA) = 0.80 × f_{SYSCLK} (MHz) × V_{DD_INT} (V)

$I_{DDINT\_SCLK0\_DYN}$ (mA) = 0.40 × f_{SCLK0} (MHz) × V_{DD_INT} (V)

$I_{DDINT\_SCLK1\_DYN}$ (mA) = 0.04 × f_{SCLK1} (MHz) × V_{DD_INT} (V)

$I_{DDINT\_DCLK\_DYN}$ (mA) = 0.14 × f_{DCLK} (MHz) × V_{DD_INT} (V)

$I_{DDINT\_OCLK\_DYN}$ (mA) = 0.005 × f_{OCLK} (MHz) × V_{DD_INT} (V)

Current from High Speed Peripheral Operation

The following modules contribute significantly to power dissipation and a single term is added when they are used.

$I_{DDINT\_USB\_DYN}$ = 20 mA (if USB enabled)

$I_{DDINT\_MLB\_DYN}$ = 10 mA (if MLB 6-pin interface is enabled)

$I_{DDINT\_GIGE\_DYN}$ = 10 mA (if Gigabit Ethernet MAC controller is enabled)

Data Transmission Current

The data transmission current represents the power dissipated when transmitting data. This current is proportional to the data rate. Refer to the power calculator of this product to estimate $I_{DDINT\_DMA\_DR\_DYN}$ based on the bandwidth of the data transfer.

ABSOLUTE MAXIMUM RATINGS

Stresses at or above those listed in Table 35 may cause permanent damage to the product. This is a stress rating only; functional operation of the product at these or any other conditions above those indicated in the operational section of this specification is not implied. Operation beyond the maximum operating conditions for extended periods may affect product reliability.

Table 35. Absolute Maximum Ratings

Parameter	Rating
Internal (Core) Supply Voltage (V_{DD_INT})	–0.33 V to +1.26 V
External (I/O) Supply Voltage (V_{DD_EXT})	–0.33 V to +3.60 V
DDR2/LPDDR Controller Supply Voltage (V_{DD_DMC})	–0.33 V to +1.90 V
DDR3 Controller Supply Voltage (V_{DD_DMC})	–0.33 V to +1.60 V
USB PHY Supply Voltage (V_{DD_USB})	–0.33 V to +3.60 V
Real Time Clock Supply Voltage (V_{DD_RTC})	–0.33 V to +3.60 V
PCIe Transmit Supply Voltage ($V_{DD\_PCIE\_TX}$)	–0.33 V to +1.20 V
PCIe Receive Supply Voltage ($V_{DD\_PCIE\_RX}$)	–0.33 V to +1.20 V
PCIe Supply Voltage (V_{DD_PCIE})	–0.33 V to +3.60 V
HADC Supply Voltage (V_{DD_HADC})	–0.33 V to +3.60 V
HADC Reference Voltage (V_{HADC_REF})	–0.33 V to +3.60 V
DDR2/LPDDR Input Voltage	–0.33 V to +1.90 V
DDR3 Input Voltage	–0.33 V to +1.60 V
Input Voltage	–0.33 V to +3.60 V
Output Voltage Swing	–0.33 V to V_{DD_EXT} +0.5 V
Storage Temperature Range	–65°C to +150°C
Junction Temperature While Biased	133°C

Table 36. Max Duty Cycle for Input Transient Voltage[1, 2]

Maximum Duty Cycle (%)[2]	V_{IN} Min (V)[3]	V_{IN} Max (V)[3]
100	TBD	TBD
50	TBD	TBD
40	TBD	TBD
25	TBD	TBD
20	TBD	TBD
15	TBD	TBD
10	TBD	TBD

[1] Applies to all signal balls with the exception of SYS_CLKIN0, SYS_CLKIN1, SYS_XTAL0, SYS_XTAL1, and all the USB, TWI, PCI, and DMC0 signals.

[2] Applies only when VDD_EXT is within specifications. When VDD_EXT is outside specifications, the range is VDD_EXT ± 0.2 V.

[3] The individual values cannot be combined for analysis of a single instance of overshoot or undershoot. The worst case observed value must fall within one of the specified voltages, and the total duration of the overshoot or undershoot (exceeding the 100% case) must be less than or equal to the corresponding duty cycle.

ESD SENSITIVITY

ESD (electrostatic discharge) sensitive device. Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

PACKAGE INFORMATION

The information presented in Figure 8 and Table 37 provides details about the package branding for the processors. For a complete listing of product availability, see Pre Release Products on Page 136.

Figure 8. Product Information on Package[1]

[1] Exact brand may differ, depending on package type.

Table 37. Package Brand Information

Brand Key	Field Description
ADSP-SC570	Product Name[1]
t	Temperature Range
pp	Package Type
Z	RoHS Compliant Option
ccc	See Ordering Guide
vvvvvv.x	Assembly Lot Code
n.n	Silicon Revision
#	RoHS Compliant Designation
yyww	Date Code

[1] See available products in Pre Release Products.

TIMING SPECIFICATIONS

Specifications are subject to change without notice.

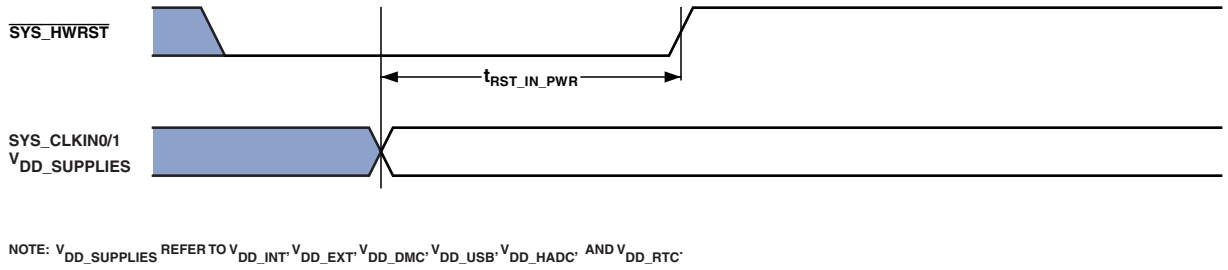
Power-Up Reset Timing

Table 38 and Figure 9 show the relationship between power supply startup and processor reset timing, related to the clock generation unit (CGU) and reset control unit (RCU).

In Figure 9, $V_{DD\_SUPPLIES}$ are V_{DD_INT}, V_{DD_EXT}, V_{DD_DMC}, V_{DD_USB}, V_{DD_HADC}, V_{DD_RTC}, $V_{DD\_PCI\_TX}$, $V_{DD\_PCI\_RX}$, and $V_{DD\_PCI\_CORE}$.

Table 38. Power-Up Reset Timing

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirement				
$t_{RST\_IN\_PWR}$	$\overline{\text{SYS\_HWRST}}$ Deasserted after $V_{DD\_SUPPLIES}$ (V_{DD_INT}, V_{DD_EXT}, V_{DD_DMC}, V_{DD_USB}, V_{DD_HADC}, V_{DD_RTC}, $V_{DD\_PCI\_TX}$, $V_{DD\_PCI\_RX}$, $V_{DD\_PCI\_CORE}$) and SYS_CLKIN are Stable and within Specification	$11 \times t_{CKIN}$		ns



NOTE: $V_{DD\_SUPPLIES}$ REFER TO V_{DD_INT}, V_{DD_EXT}, V_{DD_DMC}, V_{DD_USB}, V_{DD_HADC}, AND V_{DD_RTC}.

Figure 9. Power-Up Reset Timing

Clock and Reset Timing

Table 39 and Figure 10 describe clock and reset operations related to the clock generation unit (CGU) and reset control unit (RCU). Per the CCLK, SYSCLK, SCLK, DCLK, and OCLK timing specifications in Table 27 in Clock Related Operating Conditions on Page 58, combinations of SYS_CLKIN and clock multipliers must not select clock rates in excess of the processor's maximum instruction rate.

Table 39. Clock and Reset Timing

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
f_{CKIN}	SYS_CLKIN Frequency (Crystal)[1, 2, 3]	20	50	MHz
	SYS_CLKIN Frequency (External CLKIN)[1, 2, 3]	20	50	MHz
t_{CKINL}	CLKIN Low Pulse[1]	10		ns
t_{CKINH}	CLKIN High Pulse[1]	10		ns
t_{WRST}	$\overline{\text{RESET}}$ Asserted Pulse Width Low[4]	11 × t_{CKIN}		ns

[1] Applies to PLL bypass mode and PLL non bypass mode.
[2] The t_{CKIN} period (see Figure 10) equals $1/f_{CKIN}$.
[3] If the CGU_CTL.DF bit is set, the minimum f_{CKIN} specification is 40 MHz.
[4] Applies after power-up sequence is complete. See Table 38 and Figure 9 for power-up reset timing.

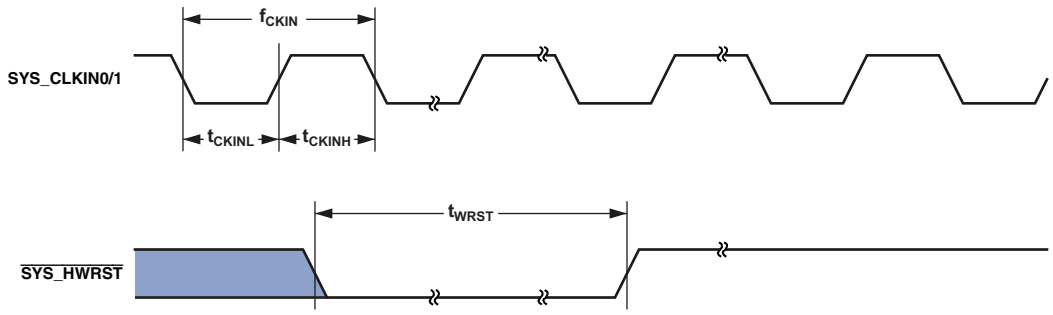


Figure 10. Clock and Reset Timing

DDR2 SDRAM Clock and Control Cycle Timing

Table 40 and Figure 11 show DDR2 SDRAM clock and control cycle timing, related to the dynamic memory controller (DMC).

Table 40. DDR2 SDRAM Clock and Control Cycle Timing, V_{DD_DMC0} Nominal 1.8 V

All specifications are based on simulation data and are subject to change without notice.				
		400 MHz[1]		
Parameter		**Min**	**Max**	**Unit**
Switching Characteristics				
t_{CK}	Clock Cycle Time (CL = 2 Not Supported)	2.5		ns
t_{CH}	Minimum Clock Pulse Width	0.48	0.52	t_{CK}
t_{CL}	Maximum Clock Pulse Width	0.48	0.52	t_{CK}
t_{IS}	Control/Address Setup Relative to DMC0_CK Rise	175		ps
t_{IH}	Control/Address Hold Relative to DMC0_CK Rise	250		ps

[1] In order to ensure proper operation of the DDR2, all the DDR2 guidelines must be strictly followed (see Engineer-to-Engineer Note EE-TBD).

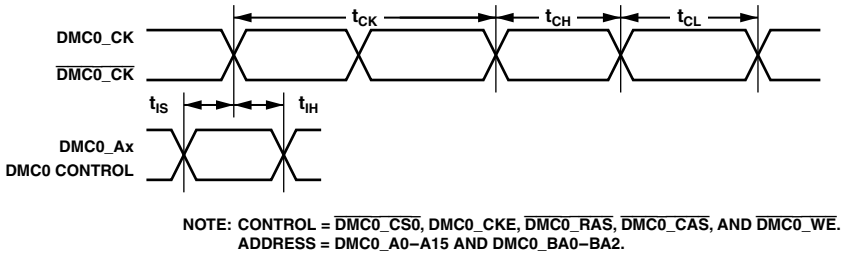


Figure 11. DDR2 SDRAM Clock and Control Cycle Timing

DDR2 SDRAM Read Cycle Timing

Table 41 and Figure 12 show DDR2 SDRAM read cycle timing, related to the dynamic memory controller (DMC).

Table 41. DDR2 SDRAM Read Cycle Timing, V_{DD_DMC0} Nominal 1.8 V

All specifications are based on simulation data and are subject to change without notice.				
		400 MHz[1]		
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{DQSQ}	DMC0_DQS-DMC0_DQ Skew for DMC0_DQS and Associated DMC0_DQ Signals		0.2	ns
t_{QH}	DMC0_DQ, DMC0_DQS Output Hold Time From DMC0_DQS	0.9		ns
t_{RPRE}	Read Preamble	0.9		t_{CK}
t_{RPST}	Read Postamble	0.4		t_{CK}

[1] In order to ensure proper operation of the DDR2, all the DDR2 guidelines must be strictly followed (see Engineer-to-Engineer Note EE-TBD).

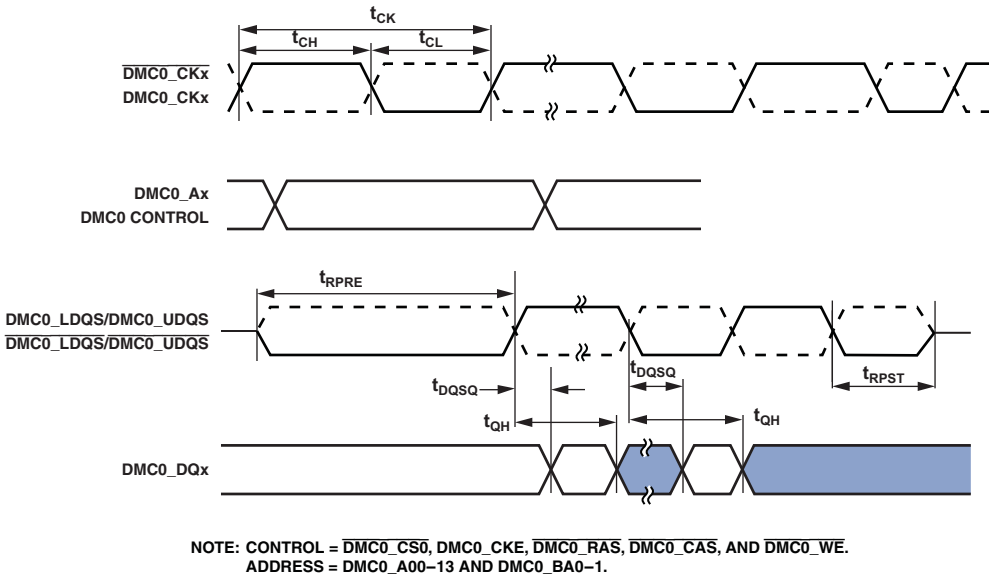


Figure 12. DDR2 SDRAM Controller Input AC Timing

DDR2 SDRAM Write Cycle Timing

Table 42 and Figure 13 show DDR2 SDRAM write cycle timing, related to the dynamic memory controller (DMC).

Table 42. DDR2 SDRAM Write Cycle Timing, V_{DD_DMC0} Nominal 1.8 V

All specifications are based on simulation data and are subject to change without notice.				
		400 MHz[1]		
Parameter		**Min**	**Max**	**Unit**
Switching Characteristics				
t_{DQSS}	DMC0_DQS Latching Rising Transitions to Associated Clock Edges[2]	–0.15	0.15	t_{CK}
t_{DS}	Last Data Valid to DMC0_DQS Delay	0.1		ns
t_{DH}	DMC0_DQS to First Data Invalid Delay	0.15		ns
t_{DSS}	DMC0_DQS Falling Edge to Clock Setup Time	0.2		t_{CK}
t_{DSH}	DMC0_DQS Falling Edge Hold Time From DMC0_CK	0.2		t_{CK}
t_{DQSH}	DMC0_DQS Input High Pulse Width	0.35		t_{CK}
t_{DQSL}	DMC0_DQS Input Low Pulse Width	0.35		t_{CK}
t_{WPRE}	Write Preamble	0.35		t_{CK}
t_{WPST}	Write Postamble	0.4		t_{CK}
t_{IPW}	Address and Control Output Pulse Width	0.6		t_{CK}
t_{DIPW}	DMC0_DQ and DMC0_DM Output Pulse Width	0.35		t_{CK}

[1] In order to ensure proper operation of the DDR2, all the DDR2 guidelines must be strictly followed (see Engineer-to-Engineer Note EE-TBD).
[2] Write command to first DMC0_DQS delay = WL × t_{CK} + t_{DQSS}.

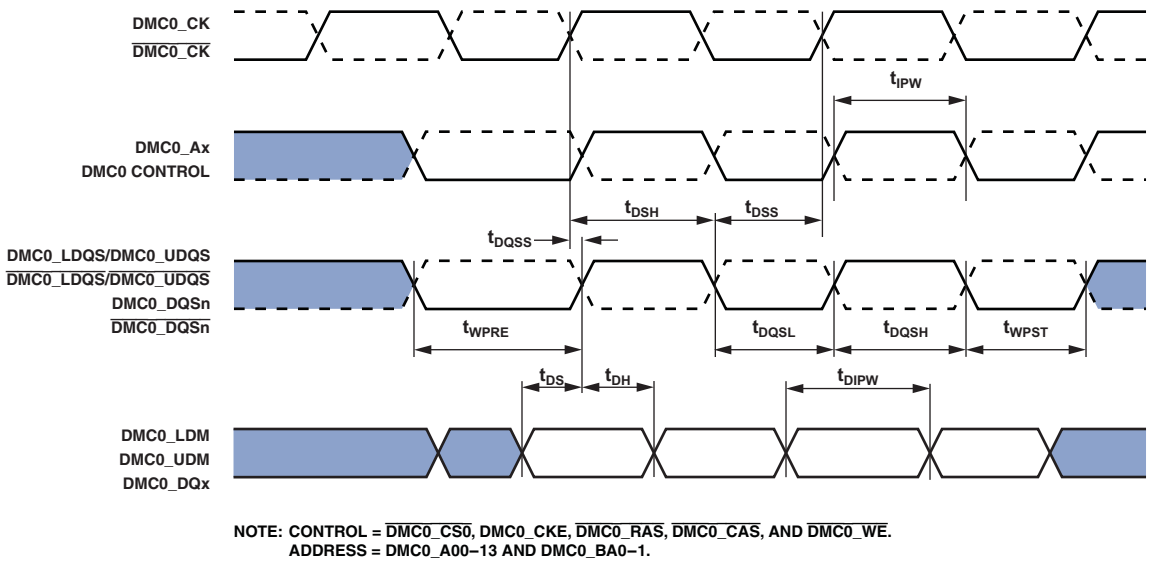


Figure 13. DDR2 SDRAM Controller Output AC Timing

Mobile DDR (LPDDR) SDRAM Clock and Control Cycle Timing

Table 43 and Figure 14 show mobile DDR SDRAM clock and control cycle timing, related to the dynamic memory controller (DMC).

Table 43. Mobile DDR SDRAM Clock and Control Cycle Timing, V_{DD_DMC0} Nominal 1.8 V

All specifications are based on simulation data and are subject to change without notice.				
		200 MHz[1]		
Parameter		**Min**	**Max**	**Unit**
Switching Characteristics				
t_{CK}	Clock Cycle Time (CL = 2 Not Supported)	5		ns
t_{CH}	Minimum Clock Pulse Width	0.45	0.55	t_{CK}
t_{CL}	Maximum Clock Pulse Width	0.45	0.55	t_{CK}
t_{IS}	Control/Address Setup Relative to DMC0_CK Rise	1		ns
t_{IH}	Control/Address Hold Relative to DMC0_CK Rise	1		ns

[1] In order to ensure proper operation of LPDDR, all the LPDDR guidelines must be strictly followed (see Engineer-to-Engineer Note EE-TBD).

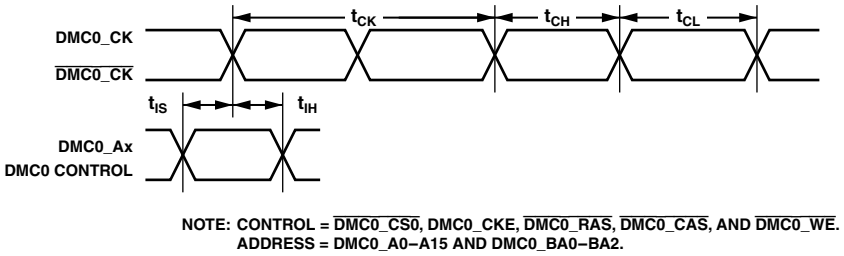


Figure 14. Mobile DDR SDRAM Clock and Control Cycle Timing

Mobile DDR SDRAM Read Cycle Timing

Table 44 and Figure 15 show mobile DDR SDRAM read cycle timing, related to the dynamic memory controller (DMC).

Table 44. Mobile DDR SDRAM Read Cycle Timing, V_{DD_DMC0} Nominal 1.8 V

All specifications are based on simulation data and are subject to change without notice.				
		200 MHz[1]		
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{QH}	DMC0_DQ, DMC0_DQS Output Hold Time From DMC0_DQS	1.75		ns
t_{DQSQ}	DMC0_DQS-DMC0_DQ Skew for DMC0_DQS and Associated DMC0_DQ Signals		0.4	ns
t_{RPRE}	Read Preamble	0.9	1.1	t_{CK}
t_{RPST}	Read Postamble	0.4	0.6	t_{CK}

[1] In order to ensure proper operation of LPDDR, all the LPDDR guidelines must be strictly followed (see Engineer-to-Engineer Note EE-TBD).

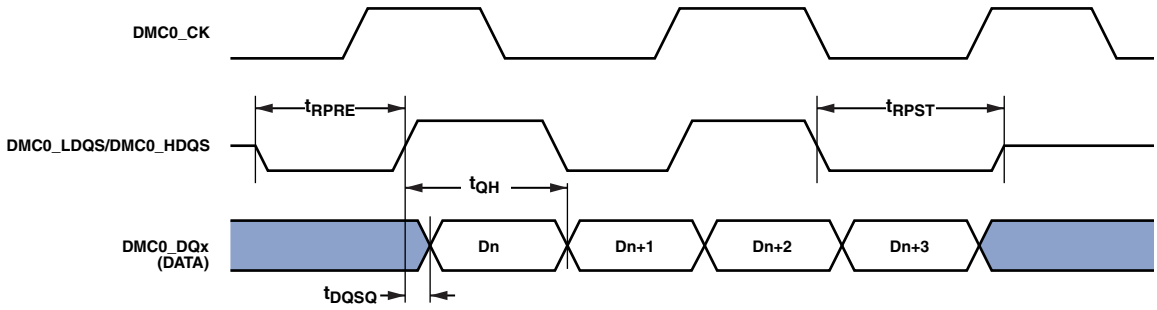


Figure 15. Mobile DDR SDRAM Controller Input AC Timing

Mobile DDR SDRAM Write Cycle Timing

Table 45 and Figure 16 show mobile DDR SDRAM write cycle timing, related to the dynamic memory controller (DMC).

Table 45. Mobile DDR SDRAM Write Cycle Timing, V_{DD_DMC0} Nominal 1.8 V

All specifications are based on simulation data and are subject to change without notice.				
		200 MHz[1]		
Parameter		**Min**	**Max**	**Unit**
Switching Characteristics				
t_{DQSS}[2]	DMC0_DQS Latching Rising Transitions to Associated Clock Edges	0.75	1.25	t_{CK}
t_{DS}	Last Data Valid to DMC0_DQS Delay (Slew > 1 V/ns)	0.48		ns
t_{DH}	DMC0_DQS to First Data Invalid Delay (Slew > 1 V/ns)	0.48		ns
t_{DSS}	DMC0_DQS Falling Edge to Clock Setup Time	0.2		t_{CK}
t_{DSH}	DMC0_DQS Falling Edge Hold Time From DMCx_CK	0.2		t_{CK}
t_{DQSH}	DMC0_DQS Input High Pulse Width	0.4		t_{CK}
t_{DQSL}	DMC0_DQS Input Low Pulse Width	0.4		t_{CK}
t_{WPRE}	Write Preamble	0.25		t_{CK}
t_{WPST}	Write Postamble	0.4		t_{CK}
t_{IPW}	Address and Control Output Pulse Width	2.3		ns
t_{DIPW}	DMC0_DQ and DMC0_DM Output Pulse Width	1.8		ns

[1] In order to ensure proper operation of LPDDR, all the LPDDR guidelines must be strictly followed (see Engineer-to-Engineer Note EE-TBD).
[2] Write command to first DMC0_DQS delay = WL × t_{CK} + t_{DQSS}.

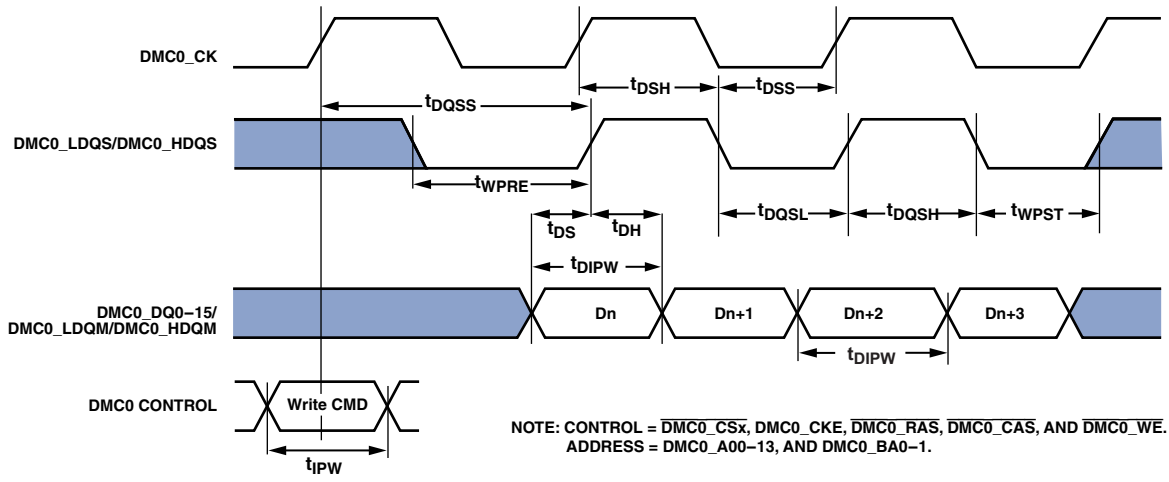


Figure 16. Mobile DDR SDRAM Controller Output AC Timing

DDR3 SDRAM Clock and Control Cycle Timing

Table 46 and Figure 17 show mobile DDR3 SDRAM clock and control cycle timing, related to the dynamic memory controller (DMC).

Table 46. DDR3 SDRAM Clock and Control Cycle Timing V_{DD_DMC0} Nominal 1.5 V

All specifications are based on simulation data and are subject to change without notice.				
		450 MHz[1]		
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{CK}	Clock Cycle Time (CL = 2 Not Supported)	2.22		ns
t_{CH}	Minimum Clock Pulse Width	0.47	0.53	t_{CK}
t_{CL}	Maximum Clock Pulse Width	0.47	0.53	t_{CK}
t_{IS}	Control/Address Setup Relative to DMC0_CK Rise	0.2		ns
t_{IH}	Control/Address Hold Relative to DMC0_CK Rise	0.275		ns

[1] In order to ensure proper operation of the DDR3, all the DDR3 guidelines must be strictly followed (see Engineer-to-Engineer Note EE-TBD).

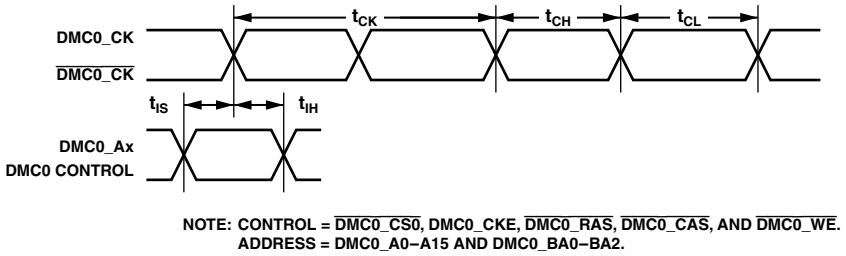


Figure 17. DDR3 SDRAM Clock and Control Cycle Timing

DDR3 SDRAM Read Cycle Timing

Table 47 and Figure 18 show mobile DDR3 SDRAM read cycle timing, related to the dynamic memory controller (DMC).

Table 47. DDR3 SDRAM Read Cycle Timing V_{DD_DMC0} Nominal 1.5 V

All specifications are based on simulation data and are subject to change without notice.				
		450 MHz[1]		
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{DQSQ}	DMC0_DQS-DMC0_DQ Skew for DMC0_DQS and Associated DMC0_DQ Signals		0.2	ns
t_{QH}	DMC0_DQ, DMC0_DQS Output Hold Time From DMC0_DQS	0.38		t_{CK}
t_{RPRE}	Read Preamble	0.9		t_{CK}
t_{RPST}	Read Postamble	0.3		t_{CK}

[1] In order to ensure proper operation of the DDR3, all the DDR3 guidelines must be strictly followed (see Engineer-to-Engineer Note EE-TBD).

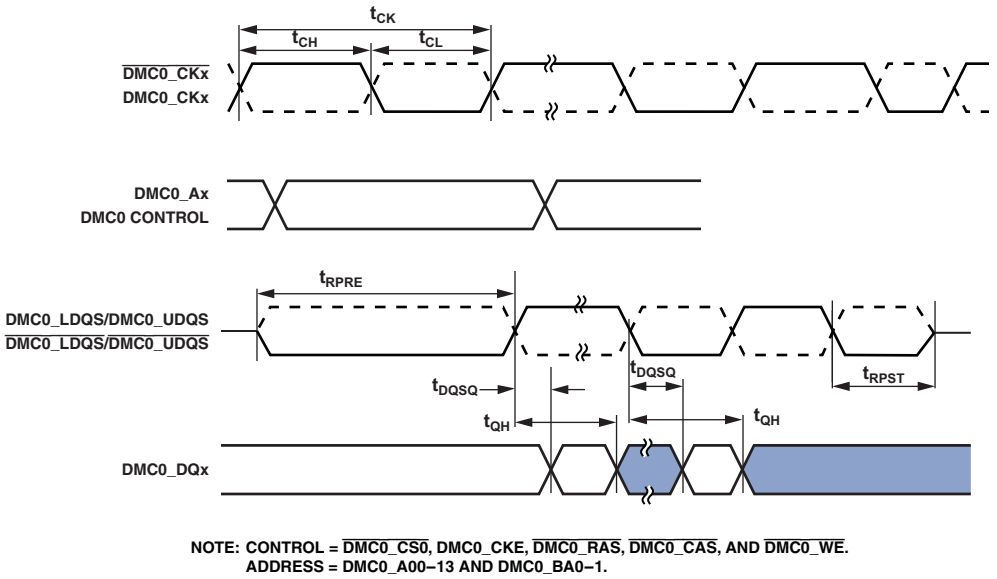


Figure 18. DDR3 SDRAM Controller Input AC Timing

DDR3 SDRAM Write Cycle Timing

Table 48 and Figure 19 show mobile DDR3 SDRAM output ac timing, related to the dynamic memory controller (DMC).

Table 48. DDR3 SDRAM Write Cycle Timing V_{DD_DMC0} Nominal 1.5 V

All specifications are based on simulation data and are subject to change without notice.				
		450 MHz[1]		
Parameter		**Min**	**Max**	**Unit**
Switching Characteristics				
t_{DQSS}	DMC0_DQS Latching Rising Transitions to Associated Clock Edges[2]	–0.25	0.25	t_{CK}
t_{DS}	Last Data Valid to DMC0_DQS Delay (Slew > 1 V/ns)	0.125		ns
t_{DH}	DMC0_DQS to First Data Invalid Delay (Slew > 1 V/ns)	0.150		ns
t_{DSS}	DMC0_DQS Falling Edge to Clock Setup Time	0.2		t_{CK}
t_{DSH}	DMC0_DQS Falling Edge Hold Time From DMC0_CK	0.2		t_{CK}
t_{DQSH}	DMC0_DQS Input High Pulse Width	0.45	0.55	t_{CK}
t_{DQSL}	DMC0_DQS Input Low Pulse Width	0.45	0.55	t_{CK}
t_{WPRE}	Write Preamble	0.9		t_{CK}
t_{WPST}	Write Postamble	0.3		t_{CK}
t_{IPW}	Address and Control Output Pulse Width	0.840		ns
t_{DIPW}	DMC0_DQ and DMC0_DM Output Pulse Width	0.550		ns

[1] In order to ensure proper operation of the DDR3, all the DDR3 guidelines must be strictly followed (see Engineer-to-Engineer Note EE-TBD).
[2] Write command to first DMC0_DQS delay = WL × t_{CK} + t_{DQSS}.

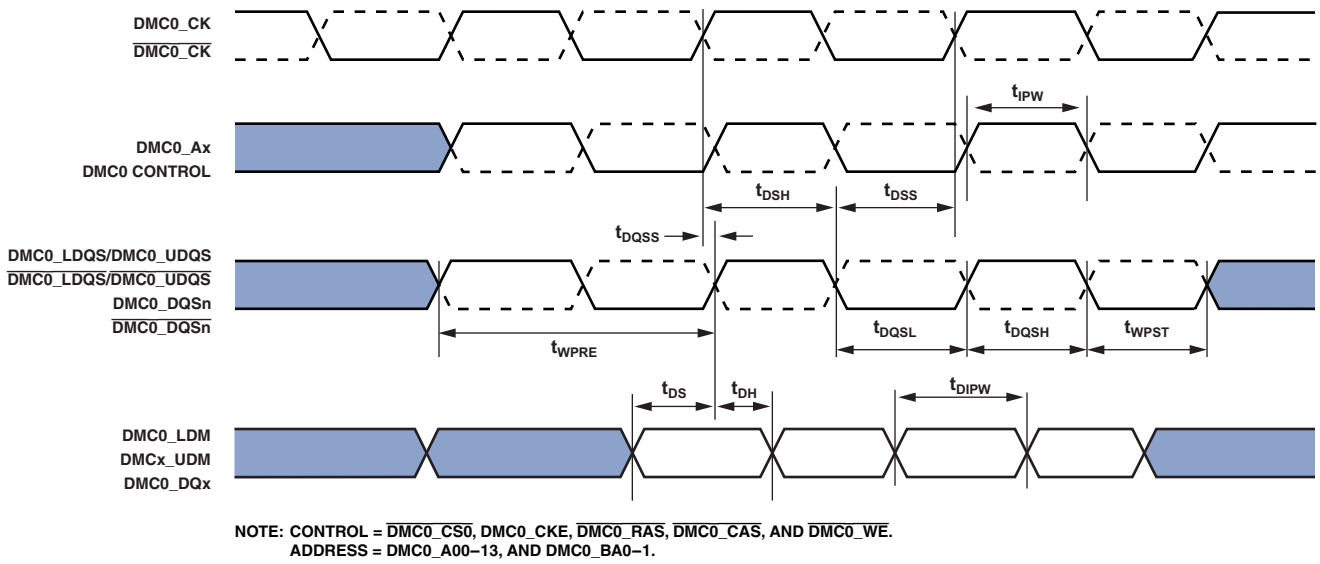


Figure 19. DDR3 SDRAM Controller Output AC Timing

Enhanced Parallel Peripheral Interface Timing

The following tables and figures describe enhanced parallel peripheral interface (EPPI) timing operations. The POLC bits in the EPPI_CTL register may be used to set the sampling/driving edges of the EPPI clock.

When internally generated, the programmed PPI clock ($f_{PCLKPROG}$) frequency in MHz is set by the following equation where VALUE is a field in the EPPI_CLKDIV register that can be set from 0 to 65535:

$$f_{PCLKPROG} = \frac{f_{SCLK0}}{(VALUE + 1)}$$

$$t_{PCLKPROG} = \frac{1}{f_{PCLKPROG}}$$

When externally generated the EPPI_CLK is called $f_{PCLKEXT}$:

$$t_{PCLKEXT} = \frac{1}{f_{PCLKEXT}}$$

Table 49. Enhanced Parallel Peripheral Interface—Internal Clock

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SFSPI}	External FS Setup Before EPPI_CLK	6.5		ns
t_{HFSPI}	External FS Hold After EPPI_CLK	0		ns
t_{SDRPI}	Receive Data Setup Before EPPI_CLK	6.5		ns
t_{HDRPI}	Receive Data Hold After EPPI_CLK	0		ns
t_{SFS3GI}	External FS3 Input Setup Before EPPI_CLK Fall Edge in Clock Gating Mode	14		ns
t_{HFS3GI}	External FS3 Input Hold Before EPPI_CLK Fall Edge in Clock Gating Mode	0		ns
Switching Characteristics				
t_{PCLKW}	EPPI_CLK Width[1]	$0.5 \times t_{PCLKPROG} - 1.5$		ns
t_{PCLK}	EPPI_CLK Period[1]	$t_{PCLKPROG} - 1.5$		ns
t_{DFSPI}	Internal FS Delay After EPPI_CLK		3.5	ns
t_{HOFSPI}	Internal FS Hold After EPPI_CLK	–0.5		ns
t_{DDTPI}	Transmit Data Delay After EPPI_CLK		3.5	ns
t_{HDTPI}	Transmit Data Hold After EPPI_CLK	–0.5		ns

[1] See Table 27 in Clock Related Operating Conditions on Page 58 for details on the minimum period that may be programmed for $t_{PCLKPROG}$.

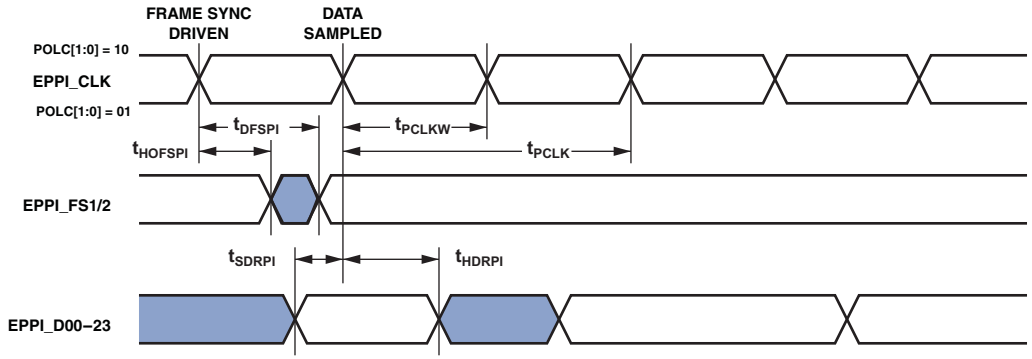


Figure 20. PPI Internal Clock GP Receive Mode with Internal Frame Sync Timing

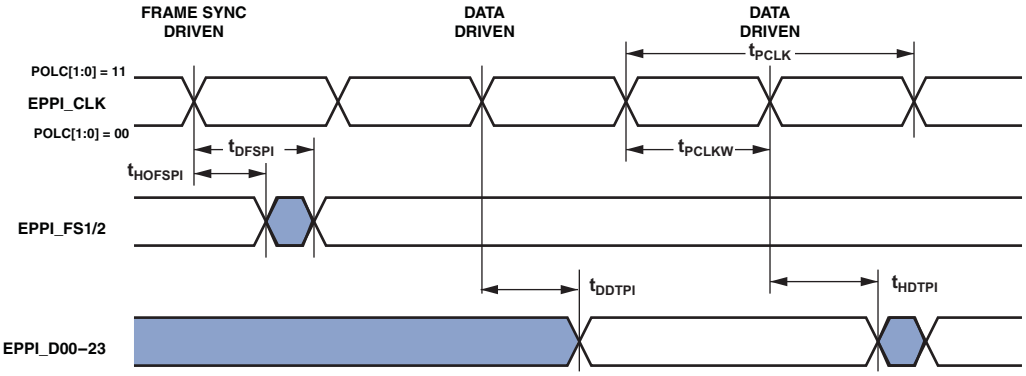


Figure 21. PPI Internal Clock GP Transmit Mode with Internal Frame Sync Timing

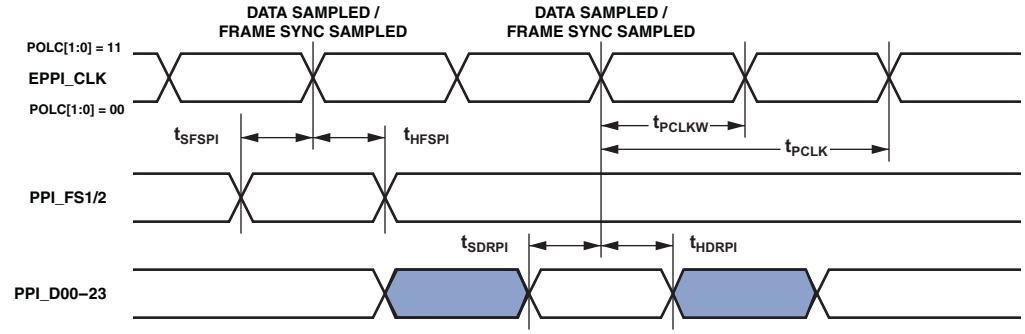


Figure 22. PPI Internal Clock GP Receive Mode with External Frame Sync Timing

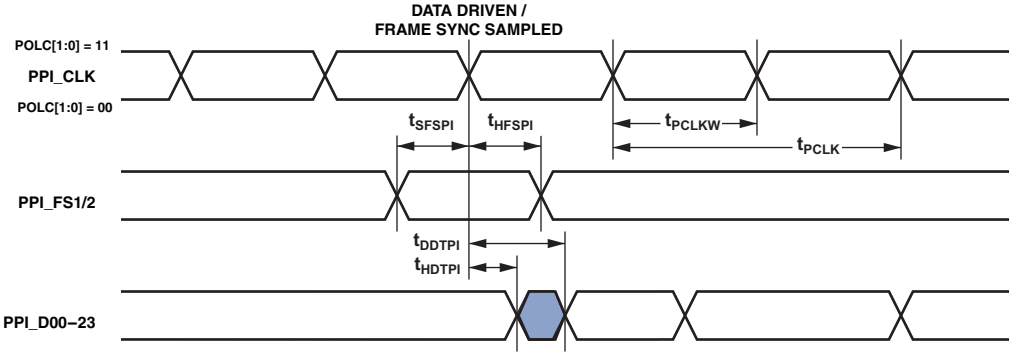


Figure 23. PPI Internal Clock GP Transmit Mode with External Frame Sync Timing

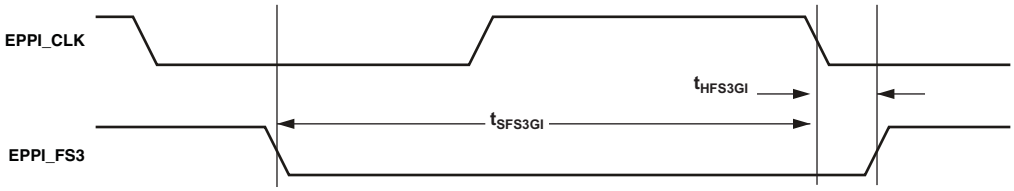


Figure 24. Clock Gating Mode with Internal Clock and External Frame Sync Timing

Table 50. Enhanced Parallel Peripheral Interface—External Clock

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{PCLKW}	EPPI_CLK Width[1]	0.5 × $t_{PCLKEXT}$ – 0.5		ns
t_{PCLK}	EPPI_CLK Period[1]	$t_{PCLKEXT}$ – 1		ns
t_{SFSPE}	External FS Setup Before EPPI_CLK	2		ns
t_{HFSPE}	External FS Hold After EPPI_CLK	3.7		ns
t_{SDRPE}	Receive Data Setup Before EPPI_CLK	2		ns
t_{HDRPE}	Receive Data Hold After EPPI_CLK	3.7		ns
Switching Characteristics				
t_{DFSPE}	Internal FS Delay After EPPI_CLK		15.3	ns
t_{HOFSPE}	Internal FS Hold After EPPI_CLK	2.4		ns
t_{DDTPE}	Transmit Data Delay After EPPI_CLK		15.3	ns
t_{HDTPE}	Transmit Data Hold After EPPI_CLK	2.4		ns

[1] This specification indicates the minimum instantaneous width or period that can be tolerated due to duty cycle variation or jitter on the external EPPI_CLK. For the external EPPI_CLK ideal maximum frequency see the $f_{PCLKEXT}$ specification in Table 27 in Clock Related Operating Conditions on Page 58.

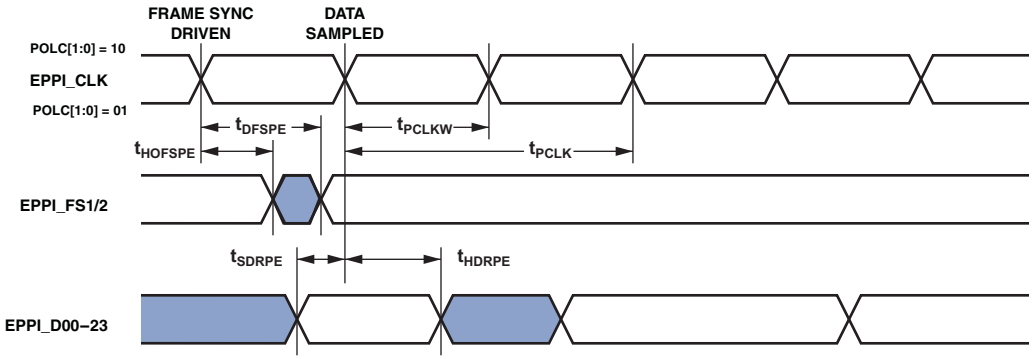


Figure 25. PPI External Clock GP Receive Mode with Internal Frame Sync Timing

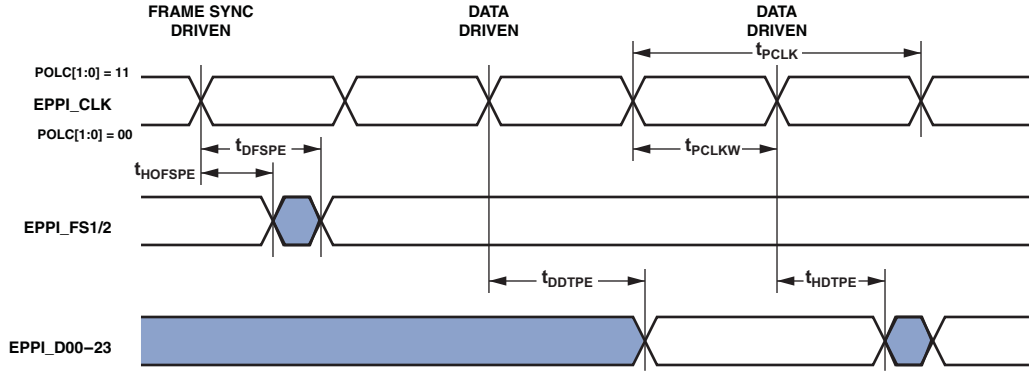


Figure 26. PPI External Clock GP Transmit Mode with Internal Frame Sync Timing

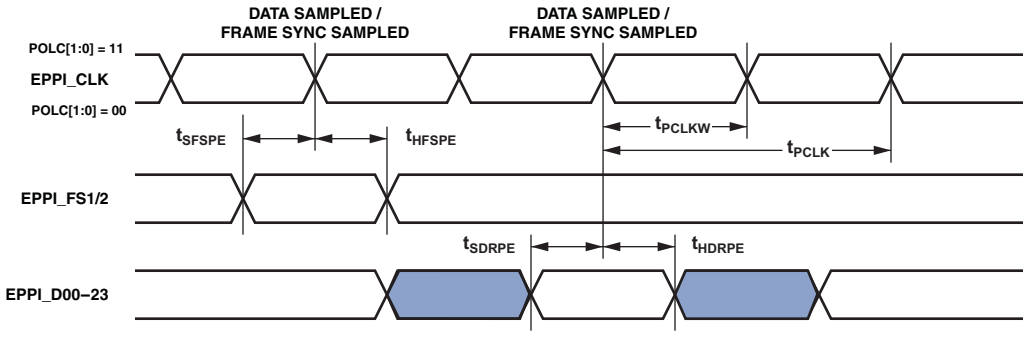


Figure 27. PPI External Clock GP Receive Mode with External Frame Sync Timing

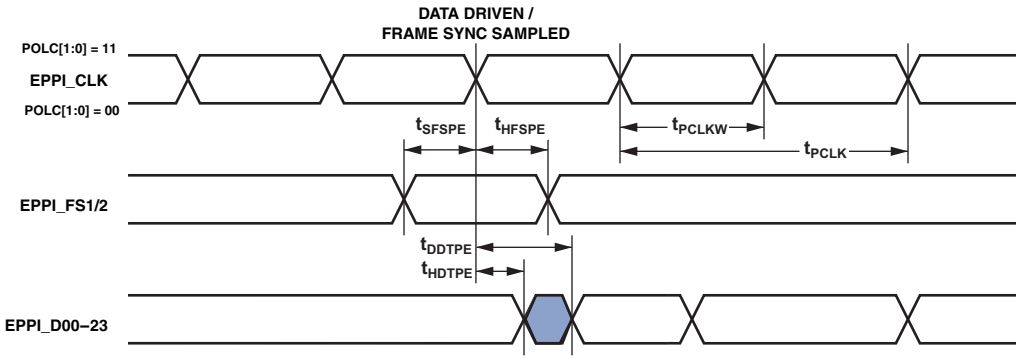


Figure 28. PPI External Clock GP Transmit Mode with External Frame Sync Timing

Link Ports

In link port (LP) receive mode, the link port clock is supplied externally and is called $f_{LCLKREXT}$:

$$t_{LCLKREXT} = \frac{1}{f_{LCLKREXT}}$$

In link port transmit mode, the programmed link port clock ($f_{LCLKTPROG}$) frequency in MHz is set by the following equation where VALUE is a field in the LP_DIV register that can be set from 1 to 255:

$$f_{LCLKTPROG} = \frac{f_{SCLKO}}{(VALUE \times 2)}$$

In the case where VALUE = 0, $f_{LCLKTPROG} = f_{SCLK0}$. For all settings of VALUE the following equation also holds:

$$t_{LCLKTPROG} = \frac{1}{f_{LCLKTPROG}}$$

Calculation of link receiver data setup and hold relative to link clock is required to determine the maximum allowable skew that can be introduced in the transmission path length difference between LPx_Dx (data) and LPx_CLK. Setup skew is the maximum delay that can be introduced in LPx_Dx relative to LPx_CLK: (setup skew = $t_{LCLKTWH}$ min – t_{DLDCH} – t_{SLDCL}). Hold skew is the maximum delay that can be introduced in LPx_CLK relative to LPx_Dx: (hold skew = $t_{LCLKTWL}$ min – t_{HLDCH} – t_{HLDCL}).

Table 51. Link Ports—Receive[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
$f_{LCLKREXT}$	LPx_CLK Frequency		150	MHz
t_{SLDCL}	Data Setup Before LPx_CLK Low	0.9		ns
t_{HLDCL}	Data Hold After LPx_CLK Low	1.4		ns
t_{LCLKEW}	LPx_CLK Period[2]	$t_{LCLKREXT}$ – 0.42		ns
$t_{LCLKRWL}$	LPx_CLK Width Low[2]	0.5 × $t_{LCLKREXT}$		ns
$t_{LCLKRWH}$	LPx_CLK Width High[2]	0.5 × $t_{LCLKREXT}$		ns
Switching Characteristic				
t_{DLALC}	LPx_ACK Low Delay After LPx_CLK Low[3]	1.5 × t_{SCLK0} + 4	2.5 × t_{SCLK0} + 12	ns

[1] Specifications apply to LP1 and LP2.
[2] This specification indicates the minimum instantaneous width or period that can be tolerated due to duty cycle variation or jitter on the external LPx_CLK. For the external LPx_CLK ideal maximum frequency see the $f_{LCLKTEXT}$ specification in Table 27 in Clock Related Operating Conditions on Page 58.
[3] LPx_ACK goes low with t_{DLALC} relative to rise of LPx_CLK after first byte, but does not go low if the receiver's link buffer is not about to fill.

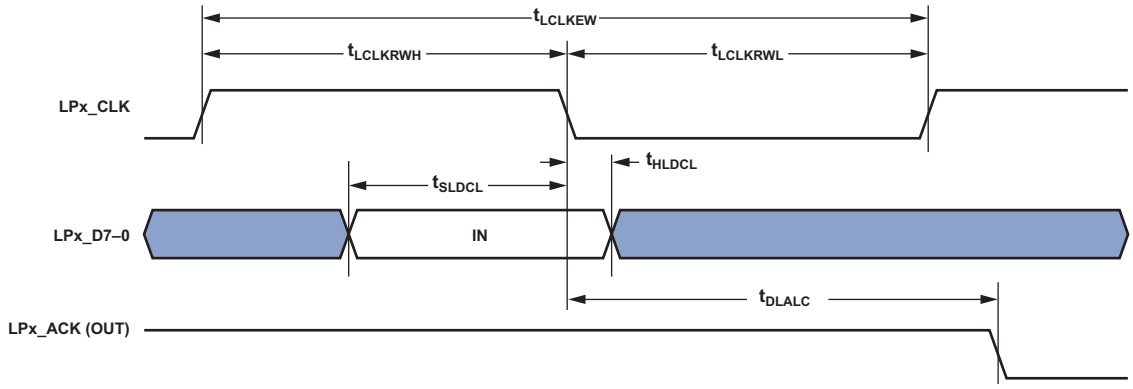
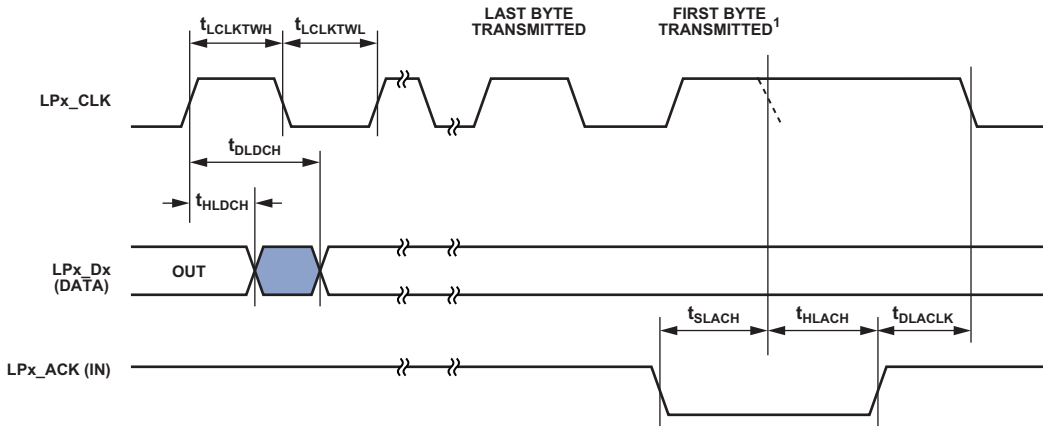


Figure 29. Link Ports—Receive

Table 52. Link Ports—Transmit[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SLACH}	LPx_ACK Setup Before LPx_CLK Low	$2 \times t_{SCLK0} + 13.5$		ns
t_{HLACH}	LPx_ACK Hold After LPx_CLK Low	–5.5		ns
Switching Characteristics				
t_{DLDCH}	Data Delay After LPx_CLK High		1.6	ns
t_{HLDCH}	Data Hold After LPx_CLK High	–0.8		ns
$t_{LCLKTWL}$[2]	LPx_CLK Width Low	$0.33 \times t_{LCLKTPROG}$	$0.6 \times t_{LCLKTPROG}$	ns
$t_{LCLKTWH}$[2]	LPx_CLK Width High	$0.45 \times t_{LCLKTPROG}$	$0.66 \times t_{LCLKTPROG}$	ns
t_{LCLKTW}[2]	LPx_CLK Period	$N \times t_{LCLKTPROG} - 0.5$		ns
t_{DLACLK}	LPx_CLK Low Delay After LPx_ACK High	$t_{SCLK0} + 4$	$2 \times t_{SCLK0} + 1 \times t_{LPCLK} + 10$	ns

[1] Specifications apply to LP1 and LP2
[2] See Table 27 in Clock Related Operating Conditions on Page 58 for details on the minimum period that may be programmed for $t_{LCLKTPROG}$.



NOTES
The t_{SLACH} and t_{HLACH} specifications apply only to the LPx_CLK falling edge. If these specifications are met, LPx_CLK would extend and the dotted LPx_CLK falling edge would not occur as shown. The position of the dotted falling edge can be calculated using the $t_{LCLKTWH}$ specification. $t_{LCLKTWH}$ Min should be used for t_{SLACH} and $t_{LCLKTWH}$ Max for t_{HLACH}.

Figure 30. Link Ports—Transmit

Serial Ports

To determine whether serial port (SPORT) communication is possible between two devices at clock speed n, the following specifications must be confirmed: 1) frame sync delay and frame sync setup and hold, 2) data delay and data setup and hold, and 3) serial clock (SPTx_CLK) width. In Figure 31 either the rising edge or the falling edge of SPTx_CLK (external or internal) can be used as the active sampling edge.

When externally generated, the SPORT clock is called $f_{SPTCLKEXT}$:

$$t_{SPTCLKEXT} = \frac{1}{f_{SPTCLKEXT}}$$

When internally generated, the programmed SPORT clock ($f_{SPTCLKPROG}$) frequency in MHz is set by the following equation where CLKDIV is a field in the SPORT_DIV register that can be set from 0 to 65535:

$$f_{SPTCLKPROG} = \frac{f_{SCLK0}}{(CLKDIV + 1)}$$

$$t_{SPTCLKPROG} = \frac{1}{f_{SPTCLKPROG}}$$

Table 53. Serial Ports—External Clock[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SFSE}	Frame Sync Setup Before SPTx_CLK (Externally Generated Frame Sync in either Transmit or Receive Mode)[2]	2		ns
t_{HFSE}	Frame Sync Hold After SPTx_CLK (Externally Generated Frame Sync in either Transmit or Receive Mode)[2]	2.7		ns
t_{SDRE}	Receive Data Setup Before Receive SPTx_CLK[2]	2		ns
t_{HDRE}	Receive Data Hold After SPTx_CLK[2]	2.7		ns
$t_{SPTCLKW}$	SPTx_CLK Width[3]	$0.5 \times t_{SPTCLKEXT} - 1.5$		ns
t_{SPTCLK}	SPTx_CLK Period[3]	$t_{SPTCLKEXT} - 1.5$		ns
Switching Characteristics				
t_{DFSE}	Frame Sync Delay After SPTx_CLK (Internally Generated Frame Sync in either Transmit or Receive Mode)[4]		14.5	ns
t_{HOFSE}	Frame Sync Hold After SPTx_CLK (Internally Generated Frame Sync in either Transmit or Receive Mode)[4]	2		ns
t_{DDTE}	Transmit Data Delay After Transmit SPTx_CLK[4]		14	ns
t_{HDTE}	Transmit Data Hold After Transmit SPTx_CLK[4]	2		ns

[1] Specifications apply to all eight SPORTs.
[2] Referenced to sample edge.
[3] This specification indicates the minimum instantaneous width or period that can be tolerated due to duty cycle variation or jitter on the external SPTx_CLK. For the external SPTx_CLK ideal maximum frequency see the $f_{SPTCLKEXT}$ specification in Table 27 in Clock Related Operating Conditions on Page 58.
[4] Referenced to drive edge.

Table 54. Serial Ports—Internal Clock[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SFSI}	Frame Sync Setup Before SPTx_CLK (Externally Generated Frame Sync in either Transmit or Receive Mode)[2]	12		ns
t_{HFSI}	Frame Sync Hold After SPTx_CLK (Externally Generated Frame Sync in either Transmit or Receive Mode)[2]	–0.5		ns
t_{SDRI}	Receive Data Setup Before SPTx_CLK[2]	3.4		ns
t_{HDRI}	Receive Data Hold After SPTx_CLK[2]	1.5		ns
Switching Characteristics				
t_{DFSI}	Frame Sync Delay After SPTx_CLK (Internally Generated Frame Sync in Transmit or Receive Mode)[3]		3.5	ns
t_{HOFSI}	Frame Sync Hold After SPTx_CLK (Internally Generated Frame Sync in Transmit or Receive Mode)[3]	–2.5		ns
t_{DDTI}	Transmit Data Delay After SPTx_CLK[3]		3.5	ns
t_{HDTI}	Transmit Data Hold After SPTx_CLK[3]	–2.5		ns
t_{SCLKIW}	SPTx_CLK Width[4]	$0.5 \times t_{SPTCLKPROG} - 1.5$		ns
t_{SPTCLK}	SPTx_CLK Period[4]	$t_{SPTCLKPROG} - 1.5$		ns

[1] Specifications apply to all eight SPORTs.
[2] Referenced to the sample edge.
[3] Referenced to drive edge.
[4] See Table 27 in Clock Related Operating Conditions on Page 58 for details on the minimum period that may be programmed for $t_{SPTCLKPROG}$.

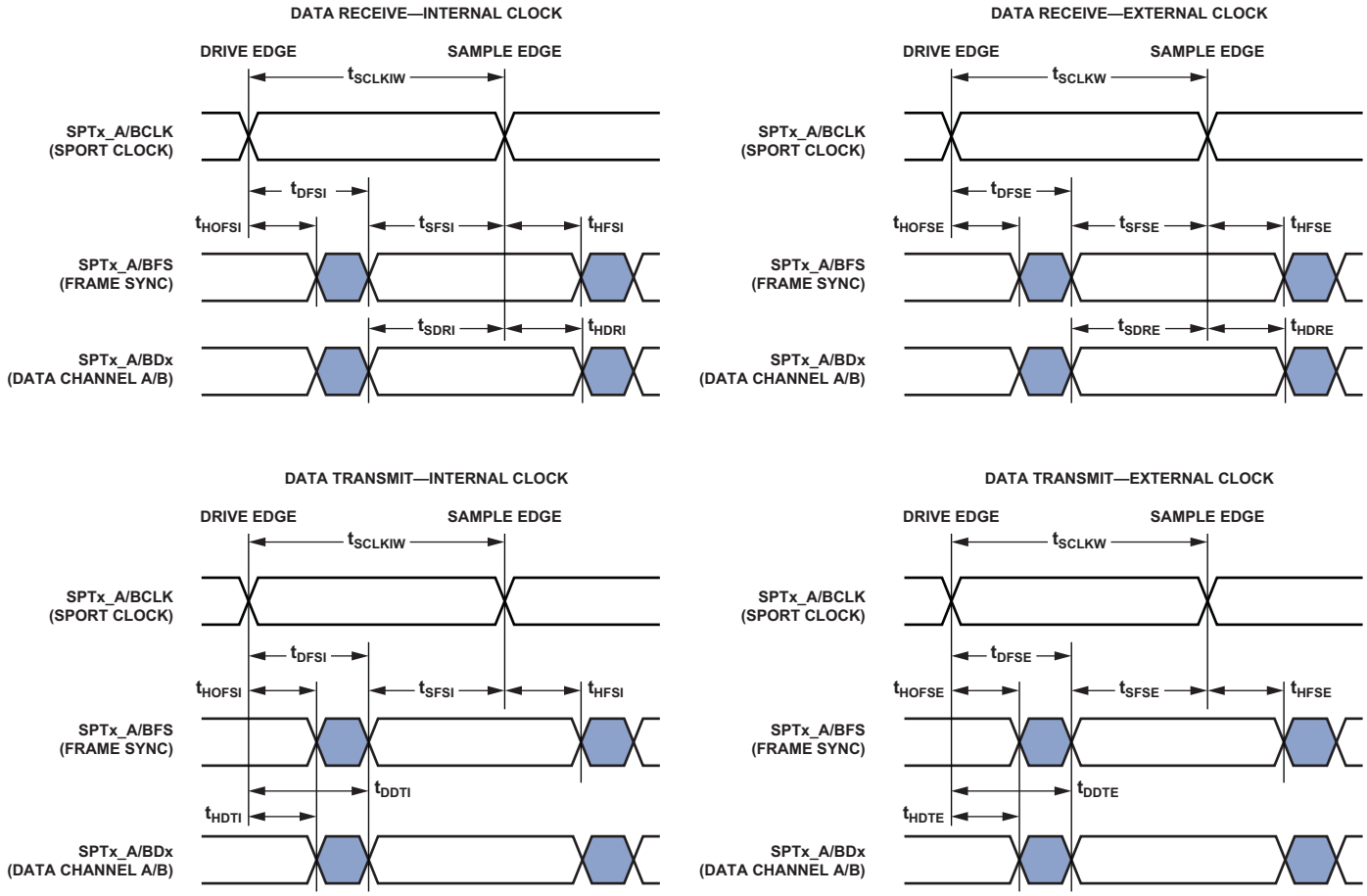


Figure 31. Serial Ports

Table 55. Serial Ports—Enable and Three-State[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Switching Characteristics				
t_{DDTEN}	Data Enable from External Transmit SPTx_CLK[2]	1		ns
t_{DDTTE}	Data Disable from External Transmit SPTx_CLK[2]		14	ns
t_{DDTIN}	Data Enable from Internal Transmit SPTx_CLK[2]	–2.5		ns
t_{DDTTI}	Data Disable from Internal Transmit SPTx_CLK[2]		2.8	ns

[1] Specifications apply to all eight SPORTs.
[2] Referenced to drive edge.

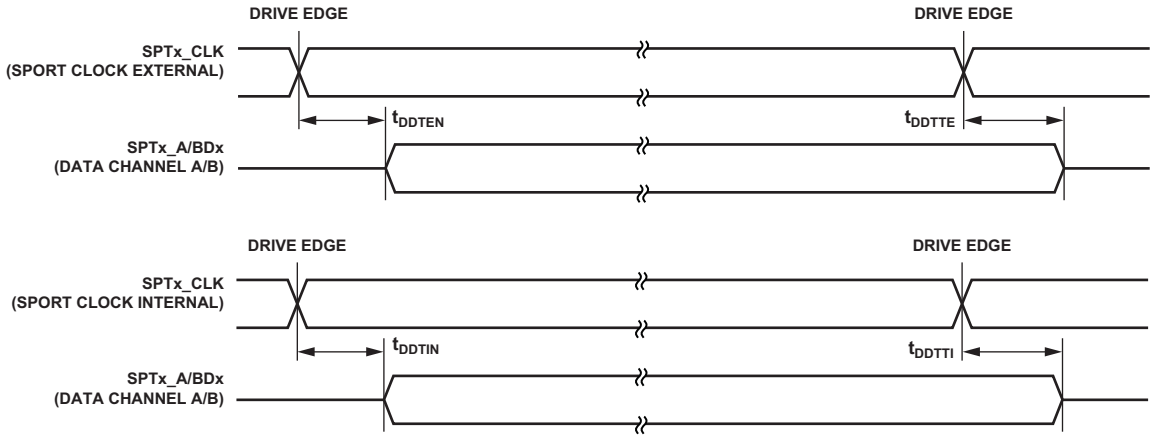


Figure 32. Serial Ports—Enable and Three-State

The SPTx_TDV output signal becomes active in SPORT multichannel mode. During transmit slots (enabled with active channel selection registers) the SPTx_TDV is asserted for communication with external devices.

Table 56. Serial Ports—TDV (Transmit Data Valid)[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Switching Characteristics				
t_{DRDVEN}	Data-Valid Enable Delay from Drive Edge of External Clock[2]	2		ns
t_{DFDVEN}	Data-Valid Disable Delay from Drive Edge of External Clock[2]		14	ns
t_{DRDVIN}	Data-Valid Enable Delay from Drive Edge of Internal Clock[2]	–2.5		ns
t_{DFDVIN}	Data-Valid Disable Delay from Drive Edge of Internal Clock[2]		3.5	ns

[1] Specifications apply to all eight SPORTs.
[2] Referenced to drive edge.

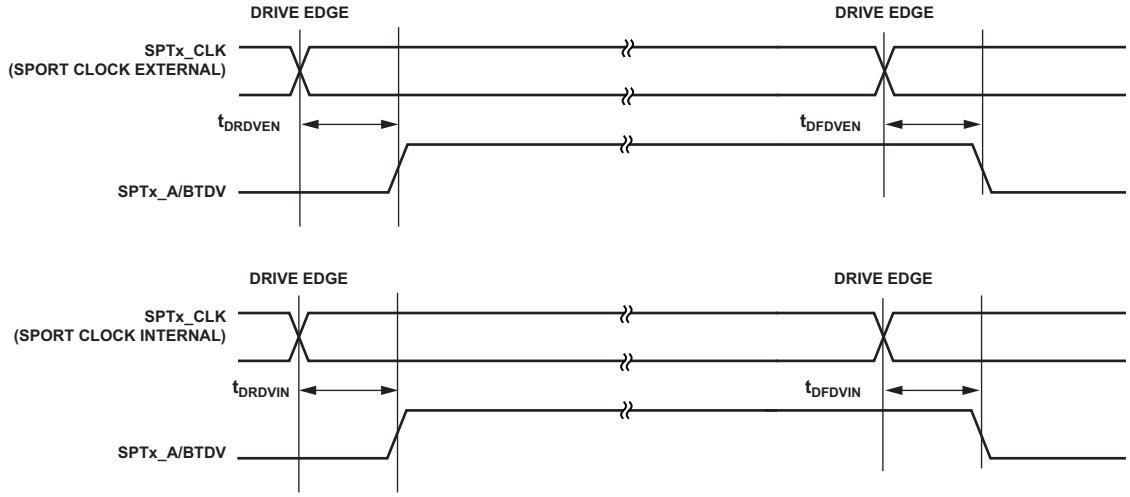


Figure 33. Serial Ports—Transmit Data Valid Internal and External Clock

Table 57. Serial Ports—External Late Frame Sync[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Switching Characteristics				
$t_{DDTLFSE}$	Data Delay from Late External Transmit Frame Sync or External Receive Frame Sync with MCE = 1, MFD = 0[2]		14	ns
$t_{DDTENFS}$	Data Enable for MCE = 1, MFD = 0[2]	0.5		ns

[1] Specifications apply to all eight SPORTs.
[2] The $t_{DDTLFSE}$ and $t_{DDTENFS}$ parameters apply to left-justified as well as standard serial mode, and MCE = 1, MFD = 0.

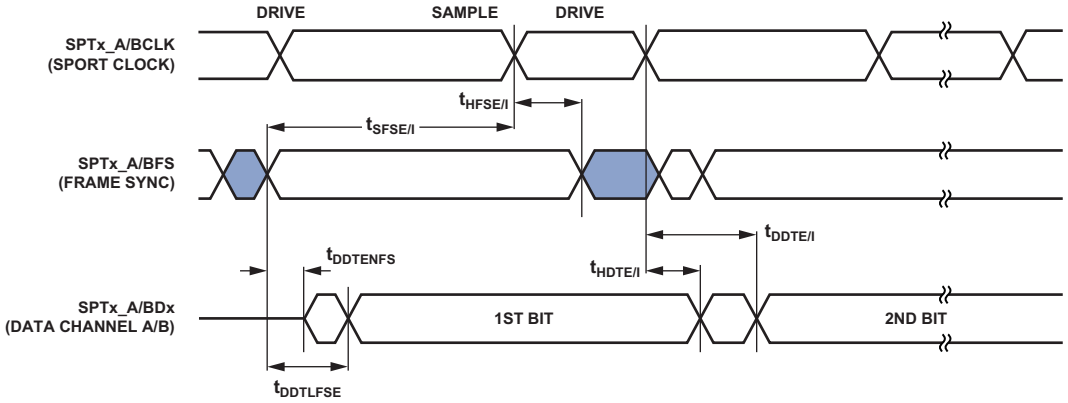


Figure 34. External Late Frame Sync

Sample Rate Converter—Serial Input Port

The ASRC input signals are routed from the DAI0_P20–1 pins using the SRU. Therefore, the timing specifications provided in Table 58 are valid at the DAI0_P20–1 pins.

Table 58. ASRC, Serial Input Port

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SRCSFS}[1]	Frame Sync Setup Before Serial Clock Rising Edge	4		ns
t_{SRCHFS}[1]	Frame Sync Hold After Serial Clock Rising Edge	5.5		ns
t_{SRCSD}[1]	Data Setup Before Serial Clock Rising Edge	4		ns
t_{SRCHD}[1]	Data Hold After Serial Clock Rising Edge	5.5		ns
$t_{SRCCLKW}$	Clock Width	$t_{SCLK0} - 1$		ns
t_{SRCCLK}	Clock Period	$2 \times t_{SCLK0}$		ns

[1] The serial clock, data, and frame sync signals can come from any of the DAI pins. The serial clock and frame sync signals can also come via PCG or SPORTs. PCG's input can be either CLKIN or any of the DAI pins.

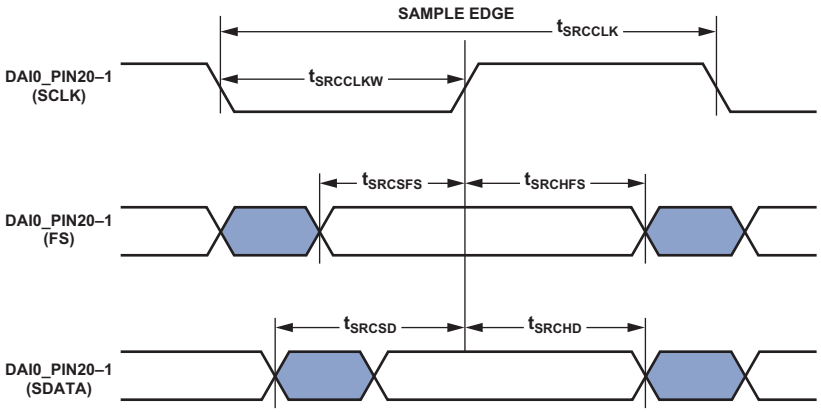


Figure 35. ASRC Serial Input Port Timing

Sample Rate Converter—Serial Output Port

For the serial output port, the frame sync is an input, and it should meet setup and hold times with regard to SCLK on the output port. The serial data output has a hold time and delay specification with regard to serial clock. Note that serial clock rising edge is the sampling edge, and the falling edge is the drive edge.

Table 59. ASRC, Serial Output Port

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SRCSFS}[1]	Frame Sync Setup Before Serial Clock Rising Edge	4		ns
t_{SRCHFS}[1]	Frame Sync Hold After Serial Clock Rising Edge	5.5		ns
$t_{SRCCLKW}$	Clock Width	$t_{SCLK0} - 1$		ns
t_{SRCCLK}	Clock Period	$2 \times t_{SCLK0}$		ns
Switching Characteristics				
t_{SRCTDD}[1]	Transmit Data Delay After Serial Clock Falling Edge		13	ns
t_{SRCTDH}[1]	Transmit Data Hold After Serial Clock Falling Edge	1		ns

[1] The serial clock, data, and frame sync signals can come from any of the DAI pins. The serial clock and frame sync signals can also come via PCG or SPORTs. PCG's input can be either CLKIN, SCLK0, or any of the DAI pins.

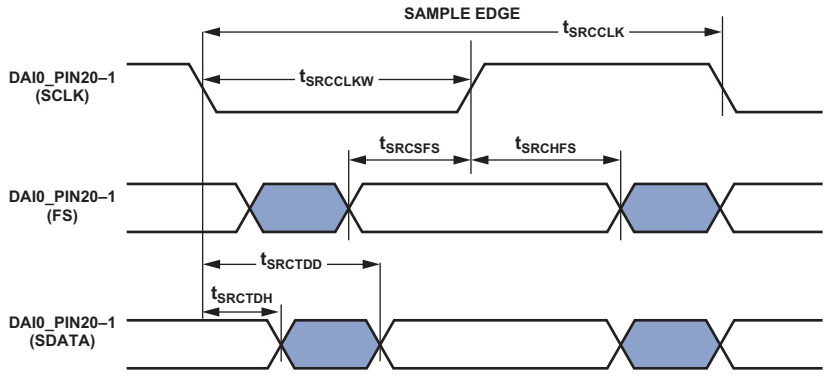


Figure 36. ASRC Serial Output Port Timing

Serial Peripheral Interface (SPI) Port—Master Timing

SPI0, SPI1 and SPI2

Table 60, Table 61, and Figure 37 describe serial peripheral interface (SPI) port master operations.

When internally generated, the programmed SPI clock ($f_{SPICLKPROG}$) frequency in MHz is set by the following equation where BAUD is a field in the SPIx_CLK register that can be set from 0 to 65535.

For SPI0, SPI1:

$$f_{SPICLKPROG} = \frac{f_{SCLK0}}{(BAUD + 1)}$$

For SPI2:

$$f_{SPICLKPROG} = \frac{f_{SCLK1}}{(BAUD + 1)}$$

$$t_{SPICLKPROG} = \frac{1}{f_{SPICLKPROG}}$$

Note that:

- In dual mode data transmit the SPIx_MISO signal is also an output.
- In quad mode data transmit the SPIx_MISO, SPIx_D2, and SPIx_D3 signals are also outputs.
- In dual mode data receive the SPIx_MOSI signal is also an input.
- In quad mode data receive the SPIx_MOSI, SPIx_D2, and SPIx_D3 signals are also inputs.
- Quad mode is supported by SPI2 only.

Table 60. Serial Peripheral Interface (SPI0, SPI1) Port—Master Timing[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SSPIDM}	Data Input Valid to SPIx_CLK Edge (Data Input Setup)	3.2		ns
t_{HSPIDM}	SPIx_CLK Sampling Edge to Data Input Invalid	1.2		ns
Switching Characteristics				
t_{SDSCIM}	$\overline{\text{SPIx\_SEL}}$ low to First SPI_CLK Edge for CPHA = 1[2]	[t_{SCLK0} – 2] or [18]		ns
	$\overline{\text{SPIx\_SEL}}$ low to First SPI_CLK Edge for CPHA = 0[2]	[1.5 × t_{SCLK0} – 2] or [13]		ns
t_{SPICHM}	SPIx_CLK High Period[3]	0.5 × $t_{SPICLKPROG}$ – 1		ns
t_{SPICLM}	SPIx_CLK Low Period[3]	0.5 × $t_{SPICLKPROG}$ – 1		ns
t_{SPICLK}	SPIx_CLK Period[3]	$t_{SPICLKPROG}$ – 1		ns
t_{HDSM}	Last SPIx_CLK Edge to $\overline{\text{SPIx\_SEL}}$ High for CPHA = 1[2]	[1.5 × t_{SCLK0} –2] or [13]		ns
	Last SPIx_CLK Edge to $\overline{\text{SPIx\_SEL}}$ High for CPHA = 0[2]	[t_{SCLK0} –2] or [18]		ns
t_{SPITDM}	Sequential Transfer Delay[2, 4]	[t_{SCLK0} – 1] or [19]		ns
$t_{DDSPIDM}$	SPIx_CLK Edge to Data Out Valid (Data Out Delay)		2.6	ns
$t_{HDSPIDM}$	SPIx_CLK Edge to Data Out Invalid (Data Out Hold)	–1.5		ns

[1] All specifications apply to SPI0 and SPI1 only.
[2] Whichever is greater.
[3] See Table 27 in Clock Related Operating Conditions on Page 58 for details on the minimum period that may be programmed for $t_{SPICLKPROG}$.
[4] Applies to sequential mode with STOP ≥ 1.

Table 61. Serial Peripheral Interface (SPI2) Port—Master Timing[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SSPIDM}	Data Input Valid to SPIx_CLK Edge (Data Input Setup)	3.2		ns
t_{HSPIDM}	SPIx_CLK Sampling Edge to Data Input Invalid	1.2		ns
Switching Characteristics				
t_{SDSCIM}	$\overline{\text{SPIx\_SEL}}$ low to First SPI_CLK Edge for CPHA = 1[2]	[t_{SCLK2} – 2] or [18]		ns
	$\overline{\text{SPIx\_SEL}}$ low to First SPI_CLK Edge for CPHA = 0[2]	[1.5 × t_{SCLK1} – 2] or [13]		ns
t_{SPICHM}	SPIx_CLK High Period[3]	0.5 × $t_{SPICLKPROG}$ – 1		ns
t_{SPICLM}	SPIx_CLK Low Period[3]	0.5 × $t_{SPICLKPROG}$ – 1		ns
t_{SPICLK}	SPIx_CLK Period[3]	$t_{SPICLKPROG}$ – 1		ns
t_{HDSM}	Last SPIx_CLK Edge to $\overline{\text{SPIx\_SEL}}$ High for CPHA = 1[2]	[1.5 × t_{SCLK1} –2] or [13]		ns
	Last SPIx_CLK Edge to $\overline{\text{SPIx\_SEL}}$ High for CPHA = 0[2]	[t_{SCLK1} –2] or [18]		ns
t_{SPITDM}	Sequential Transfer Delay[2, 4]	[t_{SCLK1} – 1] or [19]		ns
$t_{DDSPIDM}$	SPIx_CLK Edge to Data Out Valid (Data Out Delay)		2.6	ns
$t_{HDSPIDM}$	SPIx_CLK Edge to Data Out Invalid (Data Out Hold)	–1.5		ns

[1] All specifications apply to SPI2 only.
[2] Whichever is greater.
[3] See Table 27 in Clock Related Operating Conditions on Page 58 for details on the minimum period that may be programmed for $t_{SPICLKPROG}$.
[4] Applies to sequential mode with STOP ≥ 1.

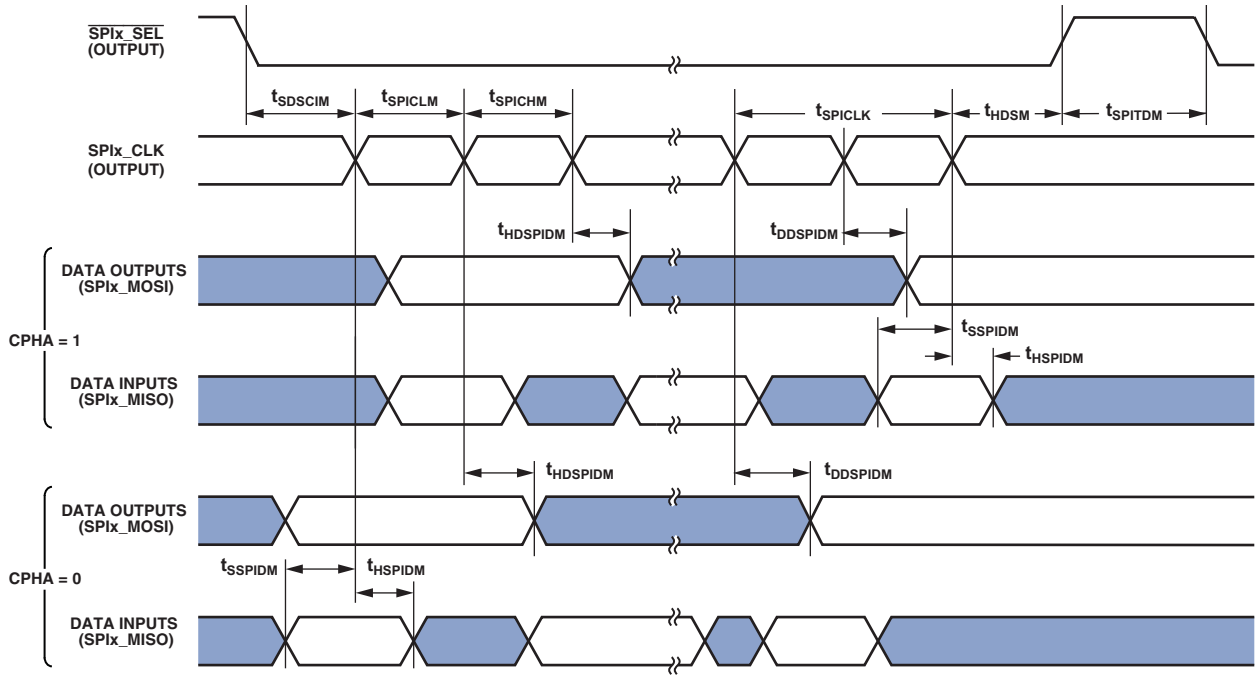


Figure 37. Serial Peripheral Interface (SPI) Port—Master Timing

Serial Peripheral Interface (SPI) Port—Slave Timing

Table 62 and Figure 38 describe serial peripheral interface (SPI) port slave operations. Note that:

- In dual mode data transmit the SPIx_MOSI signal is also an output.
- In quad mode data transmit the SPIx_MOSI, SPIx_D2, and SPIx_D3 signals are also outputs.
- In dual mode data receive the SPIx_MISO signal is also an input.
- In quad mode data receive the SPIx_MISO, SPIx_D2, and SPIx_D3 signals are also inputs.
- In SPI slave mode the SPI clock is supplied externally and is called $f_{SPICLKEXT}$:

$$t_{SPICLKEXT} = \frac{1}{f_{SPICLKEXT}}$$

- Quad mode is supported by SPI2 only.

Table 62. Serial Peripheral Interface (SPI) Port—Slave Timing[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SPICHS}	SPIx_CLK High Period[2]	$0.5 \times t_{SPICLKEXT} - 1$		ns
t_{SPICLS}	SPIx_CLK Low Period[2]	$0.5 \times t_{SPICLKEXT} - 1$		ns
t_{SPICLK}	SPIx_CLK Period[2]	$t_{SPICLKEXT} - 1$		ns
t_{HDS}	Last SPIx_CLK Edge to $\overline{\text{SPIx\_SS}}$ Not Asserted	5		ns
t_{SPITDS}	Sequential Transfer Delay	$t_{SPICLK} - 1$		ns
t_{SDSCI}	$\overline{\text{SPIx\_SS}}$ Assertion to First SPIx_CLK Edge	10.5		ns
t_{SSPID}	Data Input Valid to SPIx_CLK Edge (Data Input Setup)	2		ns
t_{HSPID}	SPIx_CLK Sampling Edge to Data Input Invalid	1.6		ns
Switching Characteristics				
t_{DSOE}	$\overline{\text{SPIx\_SS}}$ Assertion to Data Out Active	0	14	ns
t_{DSDHI}	$\overline{\text{SPIx\_SS}}$ Deassertion to Data High Impedance	0	12.5	ns
t_{DDSPID}	SPIx_CLK Edge to Data Out Valid (Data Out Delay)		14	ns
t_{HDSPID}	SPIx_CLK Edge to Data Out Invalid (Data Out Hold)	0		ns

[1] All specifications apply to all three SPIs.

[2] This specification indicates the minimum instantaneous width or period that can be tolerated due to duty cycle variation or jitter on the external SPIx_CLK. For the external SPIx_CLK ideal maximum frequency see the $f_{SPICLKTEXT}$ specification in Table 27 in Clock Related Operating Conditions on Page 58.

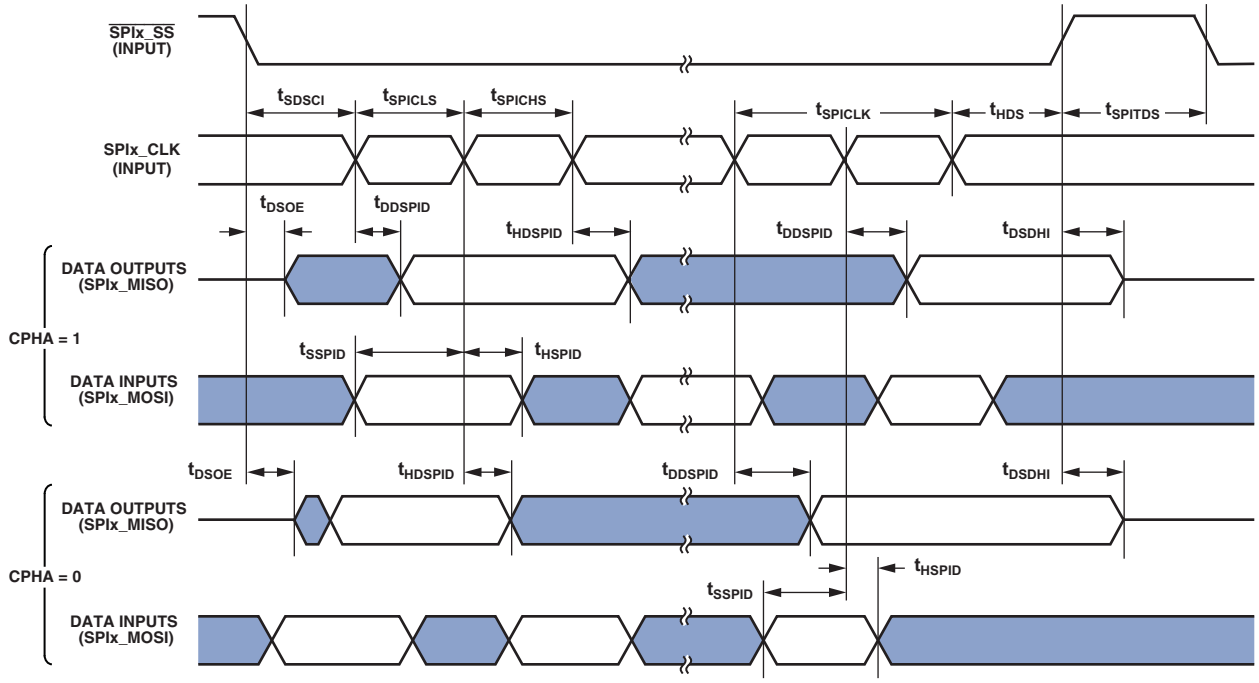


Figure 38. Serial Peripheral Interface (SPI) Port—Slave Timing

Serial Peripheral Interface (SPI) Port—SPIx_RDY Slave Timing

Table 63. SPI Port—SPIx_RDY Slave Timing[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Switching Characteristics				
$t_{DSPISCKRDYSR}$	SPIx_RDY Deassertion from Valid Input SPIx_CLK Edge in Slave Mode Receive	$3 \times t_{SCLK1}$	$4 \times t_{SCLK1} + 10$	ns
$t_{DSPISCKRDYST}$	SPIx_RDY Deassertion from Valid Input SPIx_CLK Edge in Slave Mode Transmit	$4 \times t_{SCLK1}$	$5 \times t_{SCLK1} + 10$	ns

[1] All specifications apply to all three SPIs.

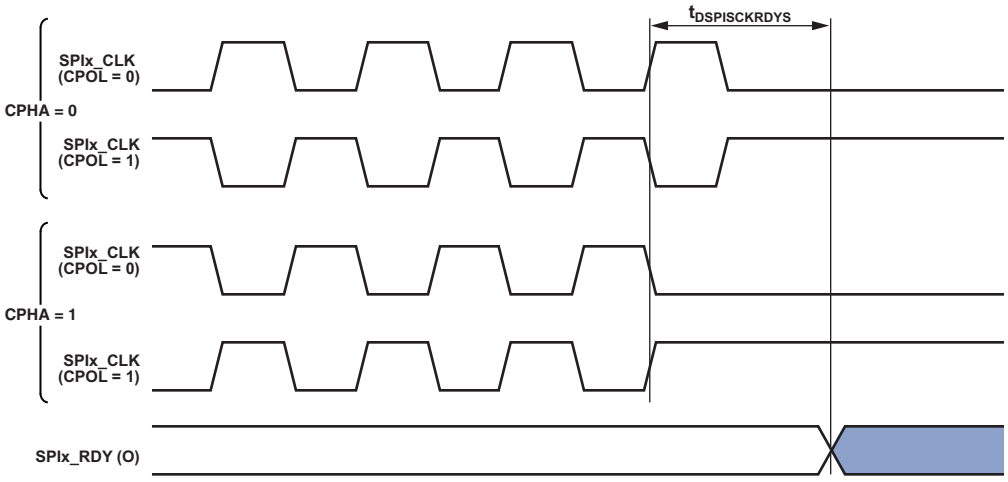


Figure 39. SPIx_RDY Deassertion from Valid Input SPIx_CLK Edge in Slave Mode Receive (FCCH = 0)

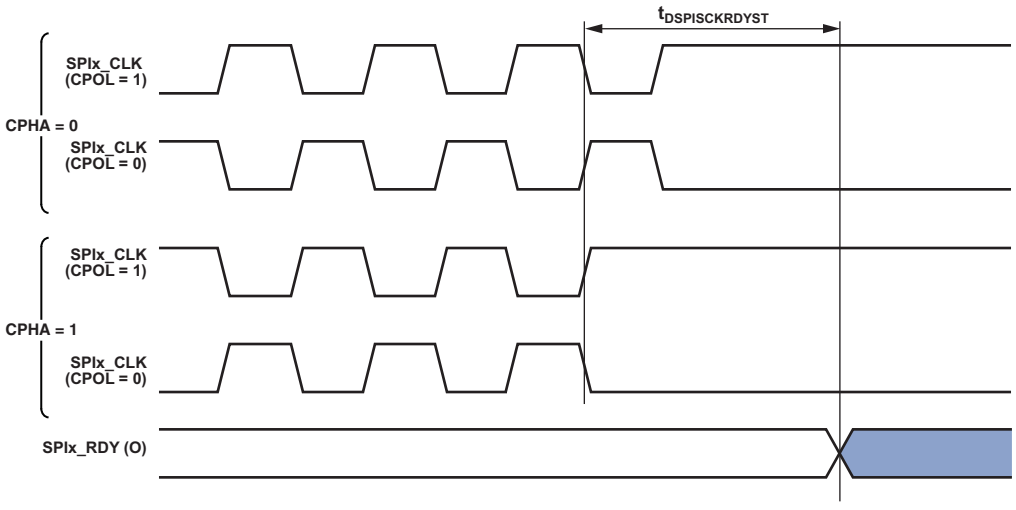


Figure 40. SPIx_RDY Deassertion from Valid Input SPIx_CLK Edge in Slave Mode Transmit (FCCH = 1)

Serial Peripheral Interface (SPI) Port—Open Drain Mode Timing

In Figure 41 and Figure 42, the outputs can be SPIx_MOSI SPIx_MISO, SPIx_D2, and/or SPIx_D3 depending on the mode of operation.

Table 64. SPI Port ODM Master Mode Timing[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Switching Characteristics				
$t_{HDSPIODMM}$	SPIx_CLK Edge to High Impedance from Data Out Valid	–1		ns
$t_{DDSPIODMM}$	SPIx_CLK Edge to Data Out Valid from High Impedance	–1	6	ns

[1] All specifications apply to all three SPIs.

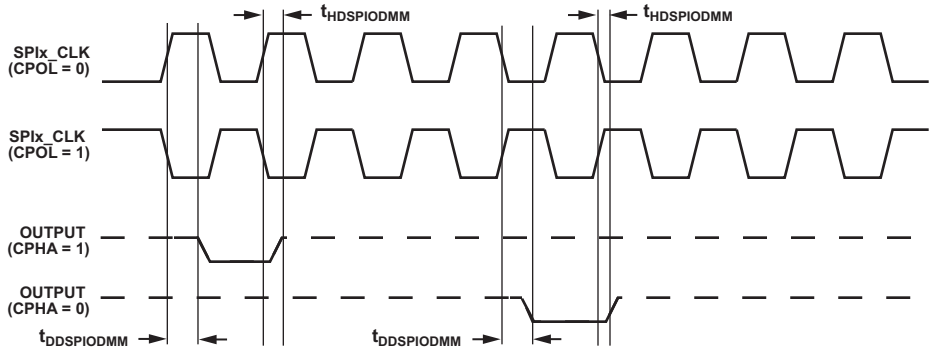


Figure 41. ODM Master

Table 65. SPI Port—ODM Slave Mode[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
$t_{HDSPIODMS}$	SPIx_CLK Edge to High Impedance from Data Out Valid	0		ns
$t_{DDSPIODMS}$	SPIx_CLK Edge to Data Out Valid from High Impedance		11	ns

[1] All specifications apply to all three SPIs.

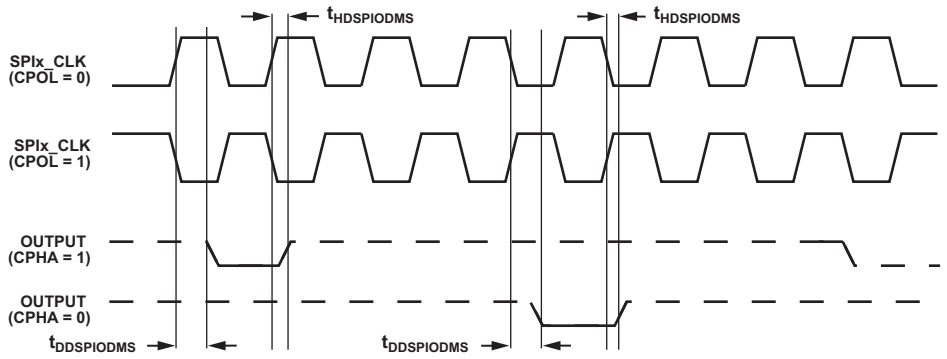


Figure 42. ODM Slave

Serial Peripheral Interface (SPI) Port—SPIx_RDY Master Timing

SPIx_RDY is used to provide flow control. The CPOL and CPHA bits are set in SPIx_CTL, while LEADX, LAGX, and STOP are in SPIx_DLY.

Table 66. SPI Port—SPIx_RDY Master Timing[1]

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
$t_{SRDYSCKM0}$	Minimum Setup Time for SPIx_RDY Deassertion in Master Mode Before Last Valid SPIx_CLK Edge of Valid Data Transfer to Block Subsequent Transfer with CPHA = 0	$(2 + 2 \times BAUD^2) \times t_{SCLK1} + 10$		ns
$t_{SRDYSCKM1}$	Minimum Setup Time for SPIx_RDY Deassertion in Master Mode Before Last Valid SPIx_CLK Edge of Valid Data Transfer to Block Subsequent Transfer with CPHA = 1	$(2 + 2 \times BAUD^2) \times t_{SCLK1} + 10$		ns
Switching Characteristic				
$t_{SRDYSCKM}$	Time Between Assertion of SPIx_RDY by Slave and First Edge of SPIx_CLK for New SPI Transfer with CPHA/CPOL = 0 and BAUD = 0 (STOP, LEAD, LAG = 0)	$4.5 \times t_{SCLK1}$	$5.5 \times t_{SCLK1} + 10$	ns
	Time Between Assertion of SPIx_RDY by Slave and First Edge of SPIx_CLK for New SPI Transfer with CPHA/CPOL = 1 and BAUD = 0 (STOP, LEAD, LAG = 0)	$4 \times t_{SCLK1}$	$5 \times t_{SCLK1} + 10$	ns
	Time Between Assertion of SPIx_RDY by Slave and First Edge of SPIx_CLK for New SPI Transfer with CPHA/CPOL = 0 and BAUD ≥ 1 (STOP, LEAD, LAG = 0)	$(1 + 1.5 \times BAUD^2) \times t_{SCLK1}$	$(2 + 2.5 \times BAUD^2) \times t_{SCLK1} + 10$	ns
	Time Between Assertion of SPIx_RDY by Slave and First Edge of SPIx_CLK for New SPI Transfer with CPHA/CPOL = 1 and BAUD ≥ 1 (STOP, LEAD, LAG = 0)	$(1 + 1 \times BAUD^2) \times t_{SCLK1}$	$(2 + 2 \times BAUD^2) \times t_{SCLK1} + 10$	ns

[1] All specifications apply to all three SPIs.
[2] BAUD value set using the SPIx_CLK.BAUD bits. BAUD value = SPIx_CLK.BAUD bits + 1.

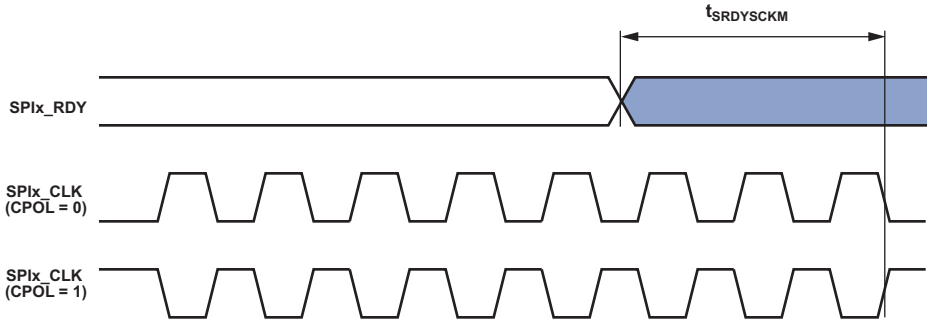


Figure 43. SPIx_RDY Setup Before SPIx_CLK with CPHA = 0

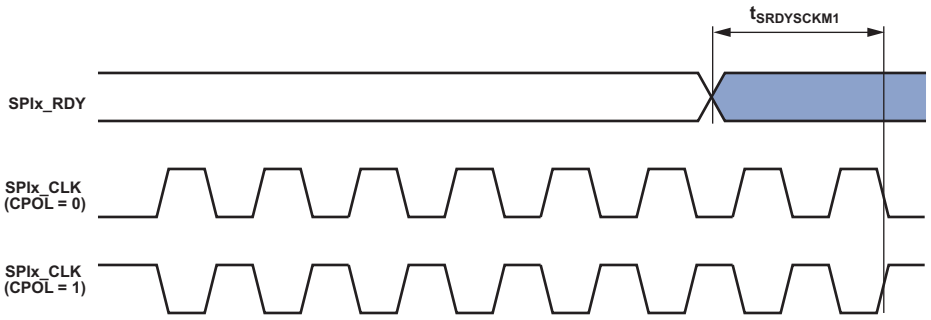


Figure 44. SPIx_RDY Setup Before SPIx_CLK with CPHA = 1

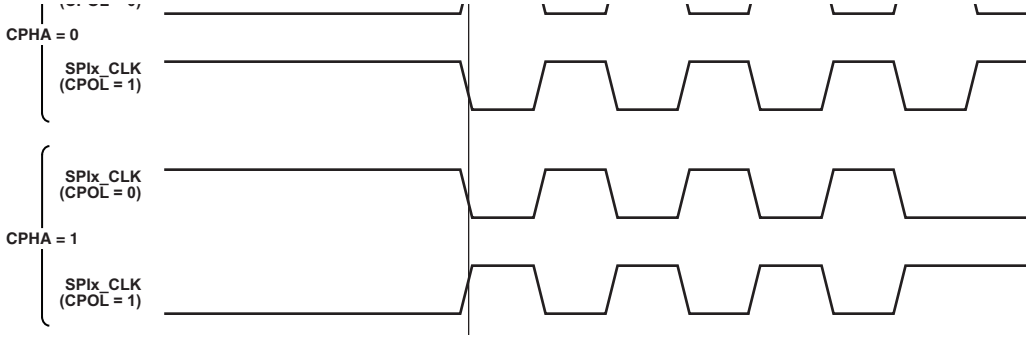


Figure 45. SPIx_CLK Switching Diagram after SPIx_RDY Assertion, CPHA = x

Precision Clock Generator (Direct Pin Routing)

This timing is only valid when the SRU is configured such that the precision clock generator (PCG) takes its inputs directly from the DAI pins (via pin buffers) and sends its outputs directly to the DAI pins. For the other cases, where the PCG's inputs and outputs are not directly routed to/from DAI pins (via pin buffers), there is no timing data available. All timing parameters and switching characteristics apply to external DAI pins (DAI0_PIN20-1).

Table 67. Precision Clock Generator (Direct Pin Routing)

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{PCGIP}	Input Clock Period	$t_{SCLK} \times 2$		ns
t_{STRIG}	PCG Trigger Setup Before Falling Edge of PCG Input Clock	4.5		ns
t_{HTRIG}	PCG Trigger Hold After Falling Edge of PCG Input Clock	3		ns
Switching Characteristics				
t_{DPCGIO}	PCG Output Clock and Frame Sync Active Edge Delay After PCG Input Clock	2.5	13.5	ns
$t_{DTRIGCLK}$	PCG Output Clock Delay After PCG Trigger	$2.5 + (2.5 \times t_{PCGIP})$	$13.5 + (2.5 \times t_{PCGIP})$	ns
$t_{DTRIGFS}$	PCG Frame Sync Delay After PCG Trigger	$2.5 + ((2.5 + D - PH) \times t_{PCGIP})$	$13.5 + ((2.5 + D - PH) \times t_{PCGIP})$	ns
t_{PCGOW}[1]	Output Clock Period	$2 \times t_{PCGIP} - 1$		ns
D = FSxDIV, PH = FSxPHASE. For more information, see the "Precision Clock Generators" chapter of the hardware reference manual.				

[1] Normal mode of operation.

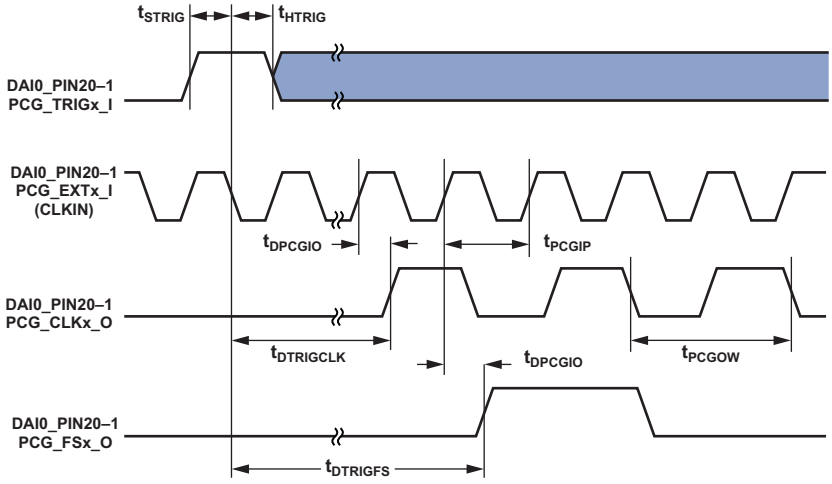


Figure 46. Precision Clock Generator (Direct Pin Routing)

General-Purpose IO Port Timing

Table 68 and Figure 47 describe I/O timing, related to the general-purpose ports (PORT).

Table 68. General-Purpose Port Timing

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirement				
t_{WFI}	General-Purpose Port Pin Input Pulse Width	$2 \times t_{SCLK0} - 1.5$		ns

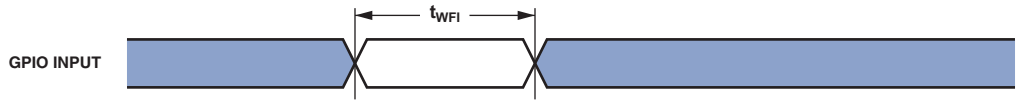


Figure 47. General-Purpose Port Timing

GPIO Timer Cycle Timing

Table 69, Table 70, and Figure 48 describe timer expired operations, related to the general-purpose timer (TIMER). The input signal is asynchronous in "width capture mode" and "external clock mode" and has an absolute maximum input frequency of ($f_{SCLK}/4$) MHz. The Width Value value is the timer period assigned in the TMx_TMRn_WIDTH register and can range from 1 to $2^{32} - 1$. Note that when externally generated, the TMR clock is called $f_{TMRCLKEXT}$:

$$t_{TMRCLKEXT} = \frac{1}{f_{TMRCLKEXT}}$$

Table 69. Timer Cycle Timing (Internal Mode)

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{WL}	Timer Pulse Width Input Low (Measured In SCLK Cycles)[1]	$2 \times t_{SCLK}$		ns
t_{WH}	Timer Pulse Width Input High (Measured In SCLK Cycles)[1]	$2 \times t_{SCLK}$		ns
Switching Characteristic				
t_{HTO}	Timer Pulse Width Output (Measured In SCLK Cycles)[2]	$t_{SCLK} \times$ WIDTH – 1.5	$t_{SCLK} \times$ WIDTH + 1.5	ns

[1] The minimum pulse width applies for TMx signals in width capture and external clock modes.
[2] WIDTH refers to the value in the TMRx_WIDTH register (it can vary from 1 to $2^{32} - 1$).

Table 70. Timer Cycle Timing (External Mode)

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{WL}	Timer Pulse Width Input Low (Measured In EXT_CLK Cycles)[1]	$2 \times t_{EXT\_CLK}$		ns
t_{WH}	Timer Pulse Width Input High (Measured In EXT_CLK Cycles)[1]	$2 \times t_{EXT\_CLK}$		ns
t_{EXT_CLK}	Timer External Clock Period[2]	$t_{TMRCLKEXT}$		ns
Switching Characteristic				
t_{HTO}	Timer Pulse Width Output (Measured In EXT_CLK Cycles)[3]	$t_{EXT\_CLK} \times$ WIDTH – 1.5	$t_{EXT\_CLK} \times$ WIDTH + 1.5	ns

[1] The minimum pulse width applies for TMx signals in width capture and external clock modes.
[2] This specification indicates the minimum instantaneous width or period that can be tolerated due to duty cycle variation or jitter on the external TMR_CLK. For the external TMR_CLK maximum frequency see the $f_{TMRCLKEXT}$ specification in Table 27 in Clock Related Operating Conditions.
[3] WIDTH refers to the value in the TMRx_WIDTH register (it can vary from 1 to $2^{32} - 1$).

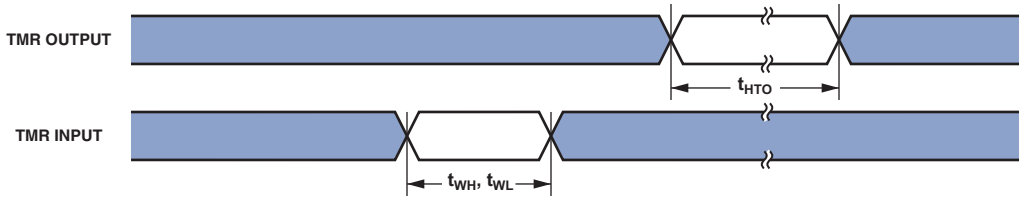


Figure 48. Timer Cycle Timing

DAI0 Pin to DAI0 Pin Direct Routing

Table 71 and Figure 49 describe I/O timing, related to the digital audio interface (DAI). For direct pin connections only (for example DAI0_PB01_I to DAI0_PB02_O).

Table 71. DAI/DAI Pin to Pin Routing

All specifications are based on simulation data and are subject to change without notice.			
Parameter	**Min**	**Max**	**Unit**
Timing Requirement			
t_{DPIO} Delay DAI Pin Input Valid to DAI Output Valid	1.5	12	ns

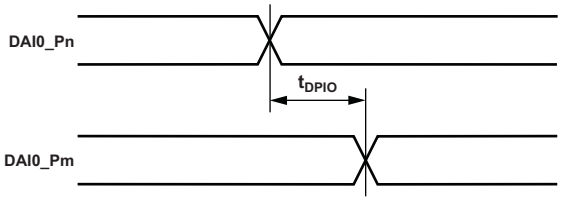


Figure 49. DAI Pin to Pin Direct Routing

Up/Down Counter/Rotary Encoder Timing

Table 72 and Figure 50 describe timing, related to the general-purpose counter (CNT).

Table 72. Up/Down Counter/Rotary Encoder Timing

All specifications are based on simulation data and are subject to change without notice.			
Parameter	**Min**	**Max**	**Unit**
Timing Requirement			
t_{WCOUNT} Up/Down Counter/Rotary Encoder Input Pulse Width	$2 \times t_{SCLK0}$		ns

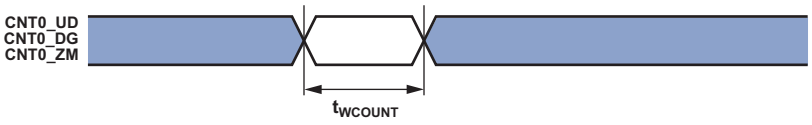


Figure 50. Up/Down Counter/Rotary Encoder Timing

ADC Controller Module (ACM) Timing

Table 73 and Figure 51 describe ADC control module (ACM) operations.

When internally generated, the programmed ACM clock ($f_{ACLKPROG}$) frequency in MHz is set by the following equation where CKDIV is a field in the ACM_TC0 register and ranges from 1 to 255:

$$f_{ACLKPROG} = \frac{f_{SCLK1}}{CKDIV + 1}$$

$$t_{ACLKPROG} = \frac{1}{f_{ACLKPROG}}$$

Setup cycles (SC) in Table 73 is also a field in the ACM_TC0 register and ranges from 0 to 4095. Hold Cycles (HC) is a field in the ACM_TC1 register that ranges from 0 to 15.

Table 73. ACM Timing

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SDR}	SPORT DRxPRI/DRxSEC Setup Before ACMx_CLK	3.5		ns
t_{HDR}	SPORT DRxPRI/DRxSEC Hold After ACMx_CLK	1.5		ns
Switching Characteristics				
t_{SCTLCS}	ACM Controls (ACMx_A[4:0]) Setup Before Assertion of $\overline{CS}$	(SC + 1) × t_{SCLK1} – 3		ns
t_{HCTLCS}	ACM Control (ACMx_A[4:0]) Hold After Deassertion of $\overline{CS}$	HC × $t_{ACLKPROG}$ – 1		ns
t_{ACLKW}	ACM Clock Pulse Width[1]	(0.5 × $t_{ACLKPROG}$) – 1.5		ns
t_{ACLK}	ACM Clock Period[1]	$t_{ACLKPROG}$ – 1.5		ns
$t_{HCSACLK}$	$\overline{CS}$ Hold to ACMx_CLK Edge	–2.5		ns
$t_{SCSACLK}$	$\overline{CS}$ Setup to ACMx_CLK Edge	$t_{ACLKPROG}$ – 3.5		ns

[1] See Table 27 in Clock Related Operating Conditions on Page 58 for details on the minimum period that may be programmed for $t_{ACLKPROG}$.

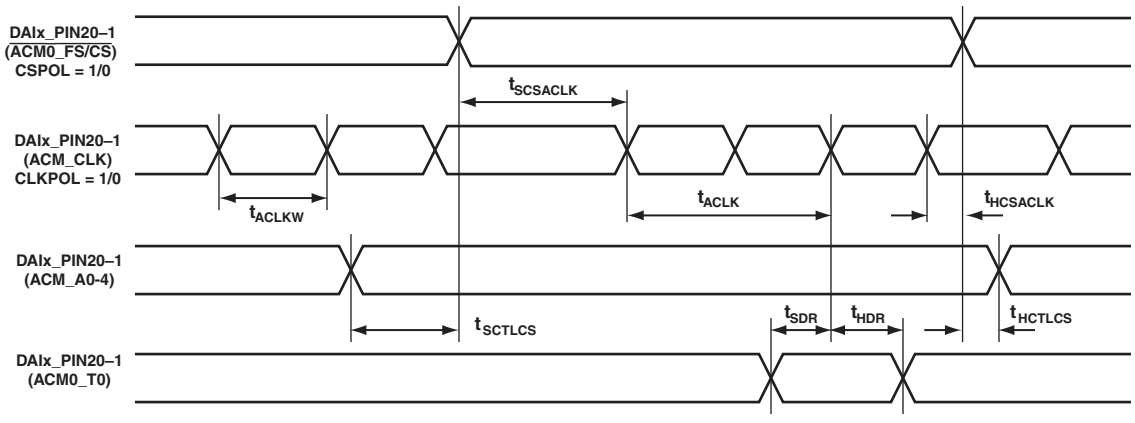


Figure 51. ACM Timing

Universal Asynchronous Receiver-Transmitter (UART) Ports—Receive and Transmit Timing

The universal asynchronous receiver-transmitter (UART) ports receive and transmit operations are described in the hardware reference manual.

Controller Area Network (CAN) Interface

The controller area network (CAN) interface timing is described in the hardware reference manual.

Universal Serial Bus (USB) On-The-Go—Receive and Transmit Timing

Table 74 describes the universal serial bus (USB) On-The-Go receive and transmit operations.

Table 74. USB On-The-Go—Receive and Transmit Timing

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
f_{USBS}	USB_XI Frequency	24	24	MHz
fs_{USB}	USB_XI Clock Frequency Stability	–50	+50	ppm

10/100/1000 Gigabit Ethernet MAC Controller (EMAC) Timing

Table 75 through Table 79 and Figure 52 through Figure 56 describe the Ethernet MAC Controller (EMAC) operations.

Table 75. Ethernet MAC Controller (EMAC) Timing: RMII Receive Signal

All specifications are based on simulation data and are subject to change without notice.				
Parameter[1]		**Min**	**Max**	**Unit**
Timing Requirements				
$t_{REFCLKF}$	ETH0_REFCLK Frequency (f_{SCLK0} = SCLK0 Frequency)	None	50 + 1%	MHz
$t_{REFCLKW}$	ETH0_REFCLK Width (t_{REFCLK} = ETH0_REFCLK Period)	t_{REFCLK} × 35%	t_{REFCLK} × 65%	ns
$t_{REFCLKIS}$	Rx Input Valid to RMII ETH0_REFCLK Rising Edge (Data In Setup)	1.75		ns
$t_{REFCLKIH}$	RMII ETH0_REFCLK Rising Edge to Rx Input Invalid (Data In Hold)	1.6		ns

[1] RMII inputs synchronous to RMII REF_CLK are ERxD1–0, RMII CRS_DV, and ERxER.

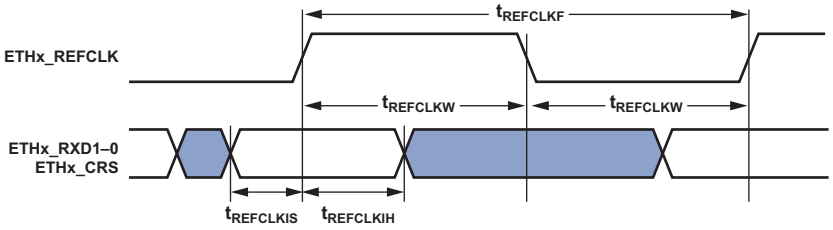


Figure 52. Ethernet MAC Controller Timing: RMII Receive Signal

Table 76. Ethernet MAC Controller (EMAC) Timing: RMII Transmit Signal

All specifications are based on simulation data and are subject to change without notice.				
Parameter[1]		**Min**	**Max**	**Unit**
Switching Characteristics				
$t_{REFCLKOV}$	RMII ETH0_REFCLK Rising Edge to Transmit Output Valid (Data Out Valid)		11.9	ns
$t_{REFCLKOH}$	RMII ETH0_REFCLK Rising Edge to Transmit Output Invalid (Data Out Hold)	2		ns

[1] RMII outputs synchronous to RMII REF_CLK are ETxD1–0.

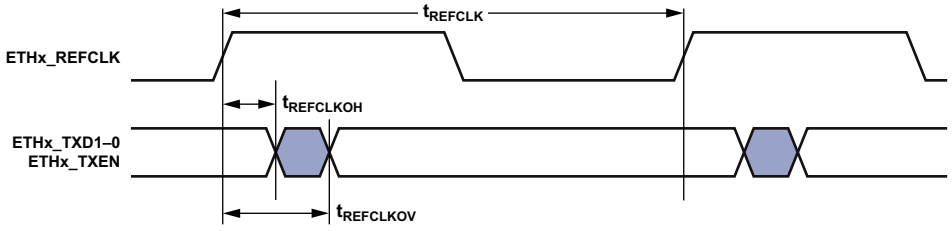


Figure 53. Ethernet MAC Controller Timing: RMII Transmit Signal

Table 77. Ethernet MAC Controller (EMAC) Timing: RMII Station Management

All specifications are based on simulation data and are subject to change without notice.				
Parameter[1]		**Min**	**Max**	**Unit**
Timing Requirements				
t_{MDIOS}	ETH0_MDIO Input Valid to ETHx_MDC Rising Edge (Setup)	12.6		ns
t_{MDCIH}	ETH0_MDC Rising Edge to ETHx_MDIO Input Invalid (Hold)	0		ns
Switching Characteristics				
t_{MDCOV}	ETH0_MDC Falling Edge to ETHx_MDIO Output Valid		t_{SCLK0} + 2	ns
t_{MDCOH}	ETH0_MDC Falling Edge to ETHx_MDIO Output Invalid (Hold)	t_{SCLK0} −2.9		ns

[1] ETH0_MDC/ETH0_MDIO is a 2-wire serial bidirectional port for controlling one or more external PHYs. ETH0_MDC is an output clock whose minimum period is programmable as a multiple of the system clock SCLK0. ETH0_MDIO is a bidirectional data line.

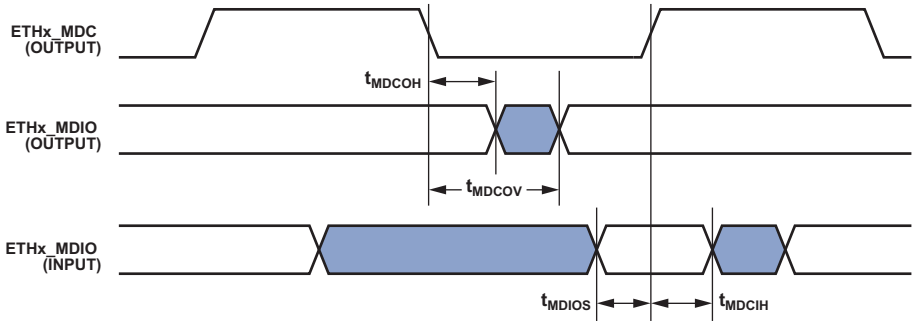


Figure 54. Ethernet MAC Controller Timing: RMII Station Management

Table 78. Ethernet MAC Controller (EMAC) Timing: MII Receive Signal

Parameter[1]		V_{DDEXT} 1.8 V Nominal		V_{DDEXT} 2.5 V or 3.3 V Nominal		Unit
		Min	Max	Min	Max	
Timing Requirements						
$t_{ERXCLKF}$	ERxCLK Frequency (f_{SCLK} = SCLK Frequency)	None	25 + 1%	None	25 + 1%	MHz
$t_{ERXCLKW}$	ERxCLK Width (t_{ERxCLK} = ERxCLK Period)	$t_{ERxCLK} \times 40\%$	$t_{ERxCLK} \times 60\%$	$t_{ERxCLK} \times 35\%$	$t_{ERxCLK} \times 65\%$	ns
$t_{ERXCLKIS}$	Rx Input Valid to ERxCLK Rising Edge (Data In Setup)	7.5		7.5		ns
$t_{ERXCLKIH}$	ERxCLK Rising Edge to Rx Input Invalid (Data In Hold)	7.5		7.5		ns

[1] MII inputs synchronous to ERxCLK are ERxD3–0, ERxDV, and ERxER.

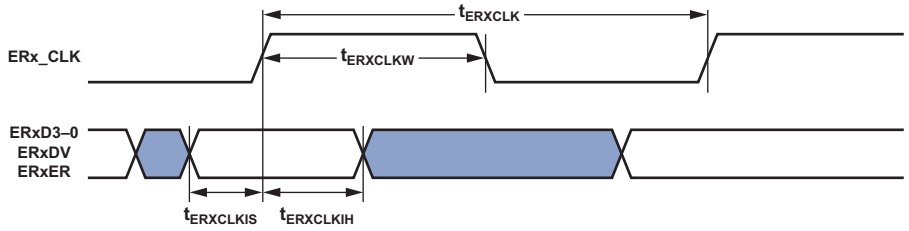


Figure 55. Ethernet MAC Controller Timing: MII Receive Signal

Table 79. Ethernet MAC Controller (EMAC) Timing: MII Transmit Signal

Parameter[1]		V_{DDEXT} 1.8 V Nominal		V_{DDEXT} 2.5 V or 3.3 V Nominal		Unit
		Min	Max	Min	Max	
Switching Characteristics						
$t_{ETXCLKF}$	ETxCLK Frequency (f_{SCLK} = SCLK Frequency)	None	25 + 1%	None	25 + 1%	MHz
$t_{ETXCLKW}$	ETxCLK Width (t_{ETxCLK} = ETxCLK Period)	$t_{ETxCLK} \times 40\%$	$t_{ETxCLK} \times 60\%$	$t_{ETxCLK} \times 35\%$	$t_{ETxCLK} \times 65\%$	ns
$t_{ETXCLKOV}$	ETxCLK Rising Edge to Tx Output Valid (Data Out Valid)		20		20	ns
$t_{ETXCLKOH}$	ETxCLK Rising Edge to Tx Output Invalid (Data Out Hold)	0		0		ns

[1] MII outputs synchronous to ETxCLK are ETxD3–0.

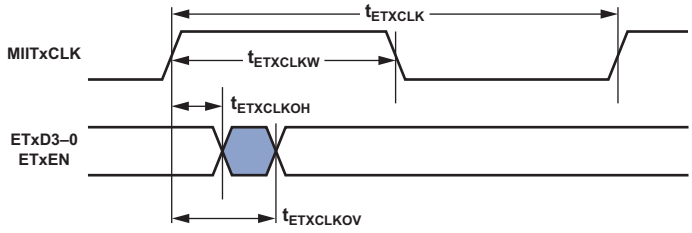


Figure 56. Ethernet MAC Controller Timing: MII Transmit Signal

Gigabit Ethernet MAC Controller (EMAC) Timing

Table 80 and Figure 57 describe the Gigabit Ethernet MAC Controller (EMAC) timing.

Table 80. Gigabit Ethernet MAC Controller (EMAC) Timing: RGMII

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
TsetupR	Data to Clock Input Setup at Receiver	1		ns
TholdR	Data to Clock Input Hold at Receiver	1		ns
$t_{GREFCLKF}$	RGMII Receive Clock Period	8		ns
$t_{GREFCLKW}$	RGMII Receive Clock Pulse Width	4		ns
Switching Characteristics				
TskewT_min	Data to Clock Output Skew at Transmitter MIN	– 0.5		ns
TskewT_max	Data to Clock Output Skew at Transmitter MAX		0.5	ns
Tcyc	Clock Cycle Duration	7.2	8.8	ns
Duty_G	Duty Cycle for Gigabit MIN	0.45 × 8	0.55 × 8	ns

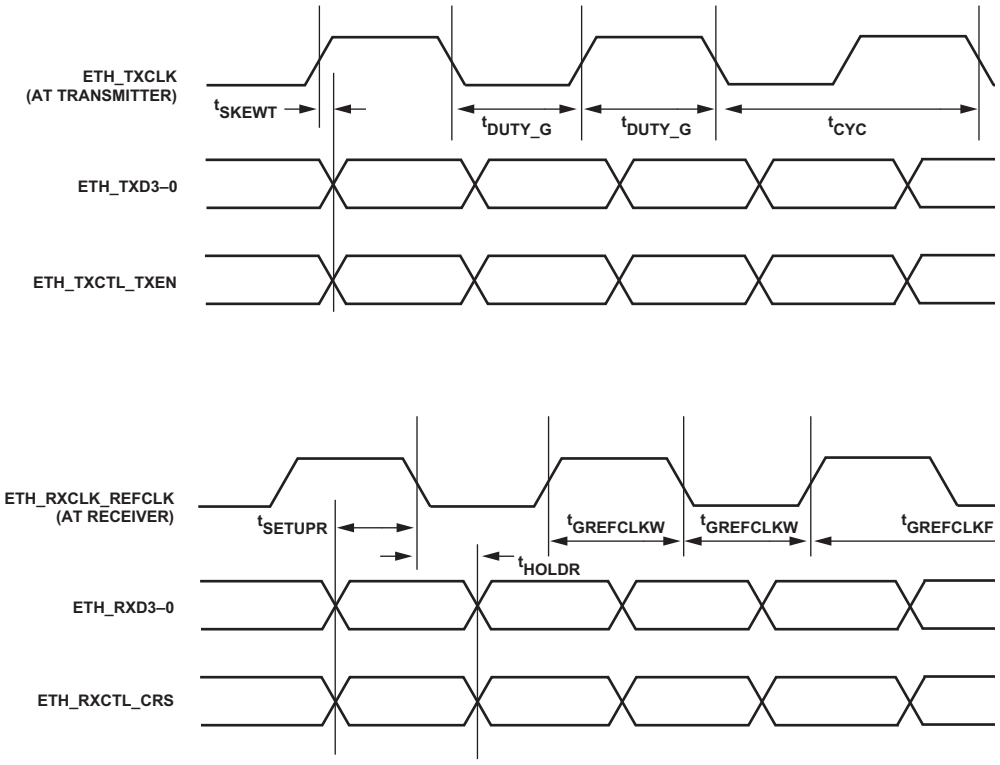


Figure 57. Gigabit Ethernet MAC Controller Timing: RGMII

Sony/Philips Digital Interface (S/PDIF) Transmitter

Serial data input to the Sony/Philips Digital Interface (S/PDIF) transmitter can be formatted as left-justified, I²S, or right-justified with word widths of 16, 18, 20, or 24 bits. The following sections provide timing for the transmitter.

S/PDIF Transmitter-Serial Input Waveforms

Figure 58 shows the right-justified mode. Frame sync is high for the left channel and low for the right channel. Data is valid on the rising edge of serial clock. The MSB is delayed the minimum in 24-bit output mode or the maximum in 16-bit output mode from a frame sync transition, so that when there are 64 serial clock periods per frame sync period, the LSB of the data is right-justified to the next frame sync transition.

Table 81. S/PDIF Transmitter Right-Justified Mode

All specifications are based on simulation data and are subject to change without notice.			
Parameter		**Nominal**	**Unit**
Timing Requirement			
t_{RJD}	Frame Sync to MSB Delay in Right-Justified Mode		
	16-Bit Word Mode	16	SCLK
	18-Bit Word Mode	14	SCLK
	20-Bit Word Mode	12	SCLK
	24-Bit Word Mode	8	SCLK

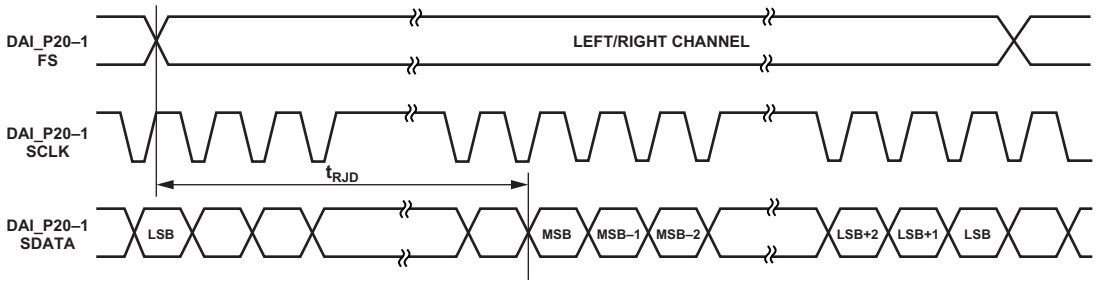


Figure 58. Right-Justified Mode

Figure 59 shows the default I²S-justified mode. The frame sync is low for the left channel and HI for the right channel. Data is valid on the rising edge of serial clock. The MSB is left-justified to the frame sync transition but with a delay.

Table 82. S/PDIF Transmitter I²S Mode

All specifications are based on simulation data and are subject to change without notice.			
Parameter		**Nominal**	**Unit**
Timing Requirement			
t_{I2SD}	Frame Sync to MSB Delay in I²S Mode	1	SCLK

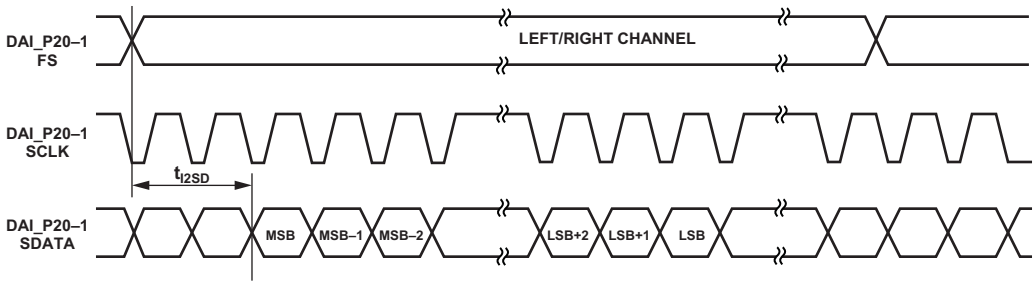


Figure 59. I²S-Justified Mode

Figure 60 shows the left-justified mode. The frame sync is high for the left channel and low for the right channel. Data is valid on the rising edge of serial clock. The MSB is left-justified to the frame sync transition with no delay.

Table 83. S/PDIF Transmitter Left-Justified Mode

All specifications are based on simulation data and are subject to change without notice.			
Parameter		**Nominal**	**Unit**
Timing Requirement			
t_{LJD}	Frame Sync to MSB Delay in Left-Justified Mode	0	SCLK

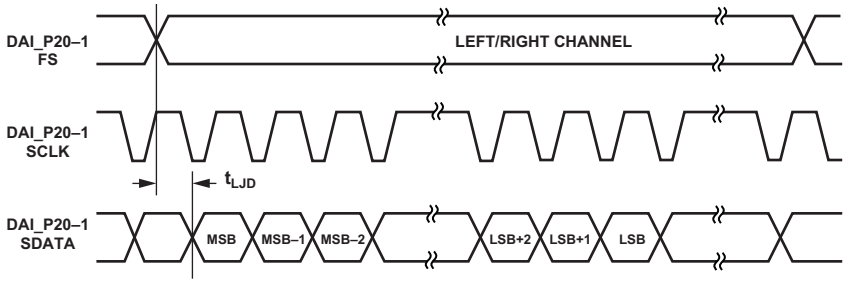


Figure 60. Left-Justified Mode

S/PDIF Transmitter Input Data Timing

The timing requirements for the S/PDIF transmitter are given in Table 84. Input signals are routed to the DAI0_Pin20–1 pins using the SRU. Therefore, the timing specifications provided below are valid at the DAI0_Pin20–1 pins.

Table 84. S/PDIF Transmitter Input Data Timing

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{SISFS}[1]	Frame Sync Setup Before Serial Clock Rising Edge	3		ns
t_{SIHFS}[1]	Frame Sync Hold After Serial Clock Rising Edge	3		ns
t_{SISD}[1]	Data Setup Before Serial Clock Rising Edge	3		ns
t_{SIHD}[1]	Data Hold After Serial Clock Rising Edge	3		ns
$t_{SITXCLKW}$	Transmit Clock Width	9		ns
$t_{SITXCLK}$	Transmit Clock Period	20		ns
$t_{SISCLKW}$	Clock Width	36		ns
t_{SISCLK}	Clock Period	80		ns

[1] The serial clock, data, and frame sync signals can come from any of the DAI pins. The serial clock and frame sync signals can also come via PCG or SPORTs. PCG's input can be either CLKIN or any of the DAI pins.

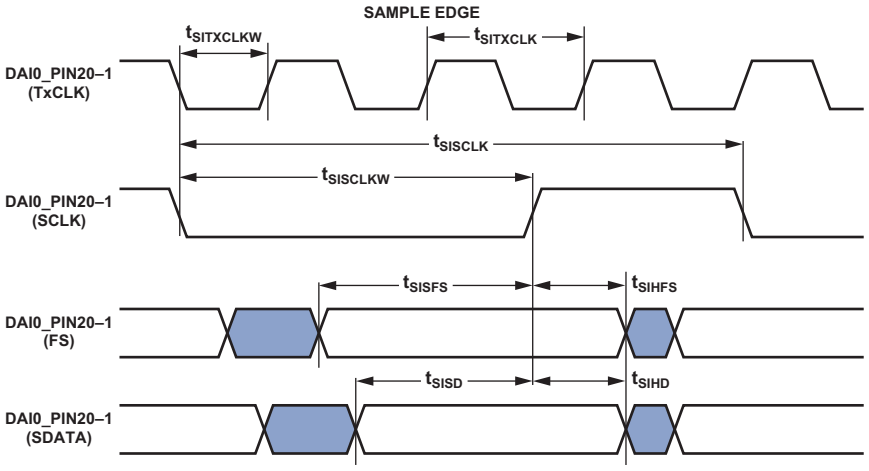


Figure 61. S/PDIF Transmitter Input Timing

Sony/Philips Digital Interface (S/PDIF) Receiver

The following section describes timing as it relates to the Sony/Philips Digital Interface (S/PDIF) receiver.

Internal Digital PLL Mode

In the internal digital phase-locked loop mode the internal PLL (digital PLL) generates the 512 × FS clock.

Table 85. S/PDIF Receiver Internal Digital PLL Mode Timing

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Switching Characteristics				
t_{DFSI}	Frame Sync Delay After Serial Clock		5	ns
t_{HOFSI}	Frame Sync Hold After Serial Clock	–2		ns
t_{DDTI}	Transmit Data Delay After Serial Clock		5	ns
t_{HDTI}	Transmit Data Hold After Serial Clock	–2		ns

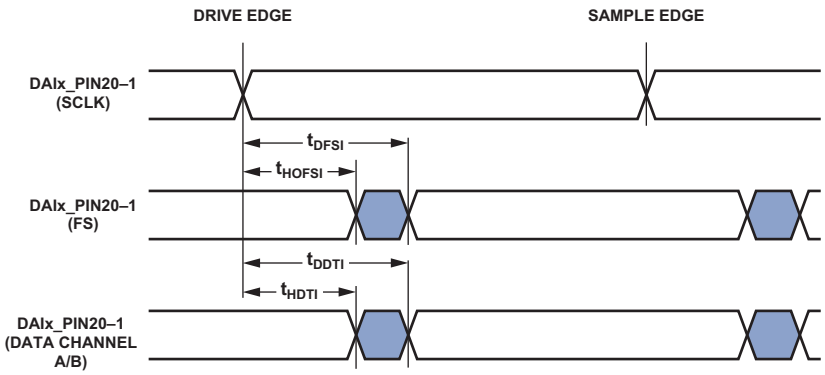


Figure 62. S/PDIF Receiver Internal Digital PLL Mode Timing

Media Local Bus (MLB)

All the numbers given are applicable for all media local bus (MLB) speed modes (1024 FS, 512 FS, and 256 FS) for 3-pin, unless otherwise specified. Please refer to the Media Local Bus Specification version 4.2 for more details.

Table 86. MLB Interface, 3-Pin Specifications

All specifications are based on simulation data and are subject to change without notice.					
Parameter		**Min**	**Typ**	**Max**	**Unit**
t_{MLBCLK}	MLB Clock Period				
	1024 FS		20.3		ns
	512 FS		40		ns
	256 FS		81		ns
t_{MCKL}	MLBCLK Low Time				
	1024 FS	6.1			ns
	512 FS	14			ns
	256 FS	30			ns
t_{MCKH}	MLBCLK High Time				
	1024 FS	9.3			ns
	512 FS	14			ns
	256 FS	30			ns
t_{MCKR}	MLBCLK Rise Time (V_{IL} to V_{IH})				
	1024 FS			1	ns
	512 FS/256 FS			3	ns
t_{MCKF}	MLBCLK Fall Time (V_{IH} to V_{IL})				
	1024 FS			1	ns
	512 FS/256 FS			3	ns
t_{MPWV}[1]	MLBCLK Pulse Width Variation				
	1024 FS			0.7	nspp
	512 FS/256			2.0	nspp
t_{DSMCF}	DAT/SIG Input Setup Time	1			ns
t_{DHMCF}	DAT/SIG Input Hold Time	2			ns
t_{MCFDZ}	DAT/SIG Output Time to Three-state	0		15	ns
t_{MCDRV}	DAT/SIG Output Data Delay From MLBCLK Rising Edge			8	ns
t_{MDZH}[2]	Bus Hold Time				
	1024 FS	2			ns
	512 FS/256	4			ns
C_{MLB}	DAT/SIG Pin Load				
	1024 FS			40	pf
	512 FS/256			60	pf

[1] Pulse width variation is measured at 1.25 V by triggering on one edge of MLBCLK and measuring the spread on the other edge, measured in ns peak-to-peak (pp).

[2] The board must be designed to ensure that the high-impedance bus does not leave the logic state of the final driven bit for this time period. Therefore, coupling must be minimized while meeting the maximum capacitive load listed.

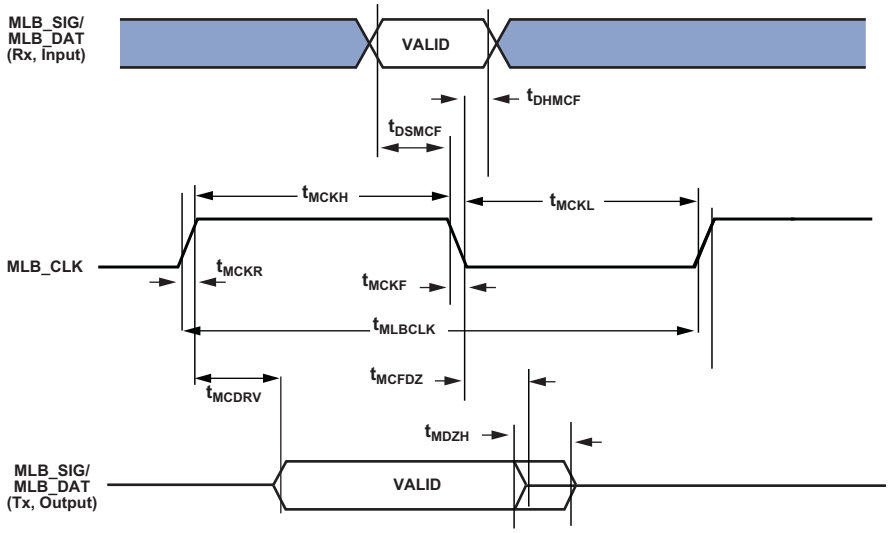


Figure 63. Media Local Bus Timing (3-Pin Interface)

The ac timing specifications of MLB 6-pin interface is detailed below. Please refer to the Media Local Bus Specification version 4.2 for more details.

Table 87. MLB Interface, 6-Pin Specifications

All specifications are based on simulation data and are subject to change without notice.						
Parameter		**Min**	**Typ**	**Max**	**Unit**	**Comment**
t_{MT}	Differential transition time at the input pin (Figure 64)			1	ns	20% to 80% V_{IN}+/– 80% to 20% V_{IN}+/–
f_{MCKE}	MLBCP/N external clock operating frequency[1] (Figure 65)	67.584		102.4	MHz	1536 × FS at 44.0 kHz 2048 × FS at 50.0 kHz
f_{MCKR}	Recovered clock operating frequency (internal, not observable at pins, only for timing references) (Figure 65)	90.112 135.168		102.4 204.8	MHz MHz	2048 × FS at 50.0 kHz 3072 × FS at 44.0 kHz 4096 × FS at 50.0 kHz
t_{DELAY}	Transmitter MLBSP/N (MLBDP/N) output valid from transition of MLBCP/N (low-to-high)[2] (Figure 66)	0.6 0.6		5 2.5	ns ns	when f_{MCKR} = 2048 × FS when f_{MCKR} = 3072 × FS or 4096 × FS
t_{PHZ}	Disable turnaround time from transition of MLBCP/N (low-to-high)[2] (Figure 67)	0.6 0.6		7 3.5	ns ns	when f_{MCKR} = 2048 × FS when f_{MCKR} = 3072 × FS or 4096 × FS
t_{PLZ}	Enable turnaround time from transition of MLBCP/N (low-to-high)[2] (Figure 67)	0.6 0.6		11.2 5.6	ns ns	when f_{MCKR} = 2048 × FS when f_{MCKR} = 3072 × FS or 4096 × FS
t_{SU}	MLBSP/N (MLBDP/N) valid to transition of MLBCP/N (low-to-high)[2] (Figure 66)	1 0.5			ns ns	when f_{MCKR} = 2048 × FS when f_{MCKR} = 3072 × FS or 4096 × FS
t_{HD}	MLBSP/N (MLBDP/N) hold from transition of MLBCP/N (low-to-high)[2, 3] (Figure 66)	0.6			ns	

[1] f_{MCKE} (max) and f_{MCKR} (max) include maximum cycle-to-cycle system jitter (t_{JITTER}) of 600ps for a bit error rate of 10E-9.
[2] t_{DELAY}, t_{PHZ}, t_{PLZ}, t_{SU}, t_{HD} may also be referenced from a low-to-high transition of the recovered clock for 2:1 recovered to external clock ratios.
[3] Receivers must latch MLBSP/N (MLBDP/N) data within t_{HD} (min) of the rising edge of MLBCP/N.

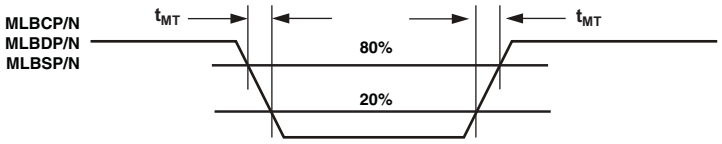


Figure 64. Media Local Bus 6-Pin Transition Time

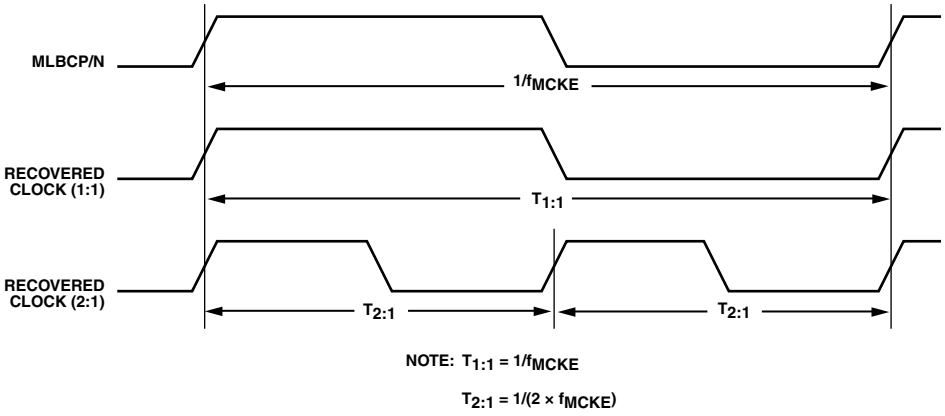


Figure 65. Media Local Bus 6-Pin Clock Definitions

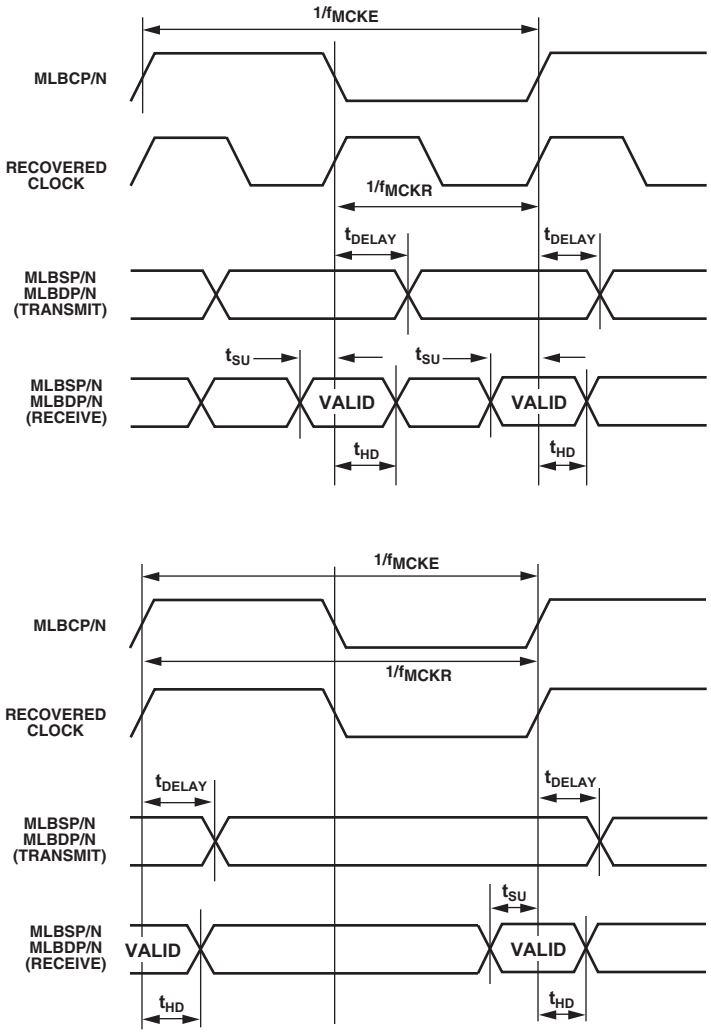


Figure 66. Media Local Bus 6-Pin Delay, Setup, and Hold Times

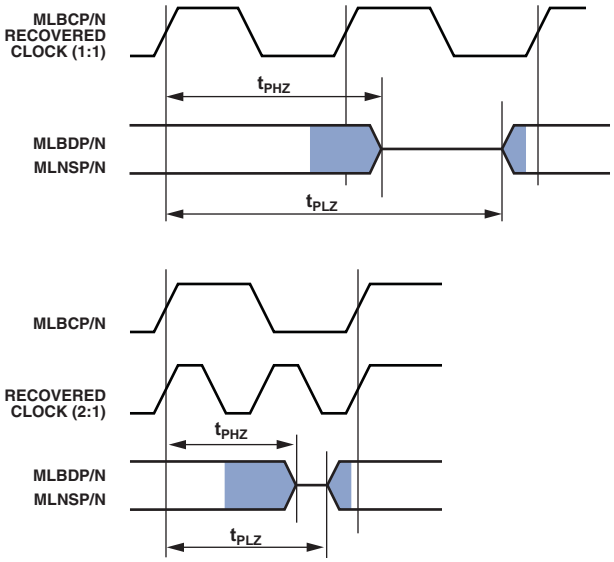


Figure 67. Media Local Bus 6-Pin Disable and Enable Turnaround Times

Mobile Storage Interface (MSI) Controller Timing

Table 88 and Figure 68 show I/O timing, related to the mobile storage interface (MSI).

Table 88. MSI Controller Timing

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{ISU}	Input Setup Time	4.8		ns
t_{IH}	Input Hold Time	–0.5		ns
Switching Characteristics				
f_{PP}	Clock Frequency Data Transfer Mode[1]		50	MHz
t_{WL}	Clock Low Time	8		ns
t_{WH}	Clock High Time	8		ns
t_{TLH}	Clock Rise Time		3	ns
t_{THL}	Clock Fall Time		3	ns
t_{ODLY}	Output Delay Time During Data Transfer Mode		2	ns
t_{OH}	Output Hold Time	–1.8		ns

[1] $t_{PP} = 1/f_{PP}$

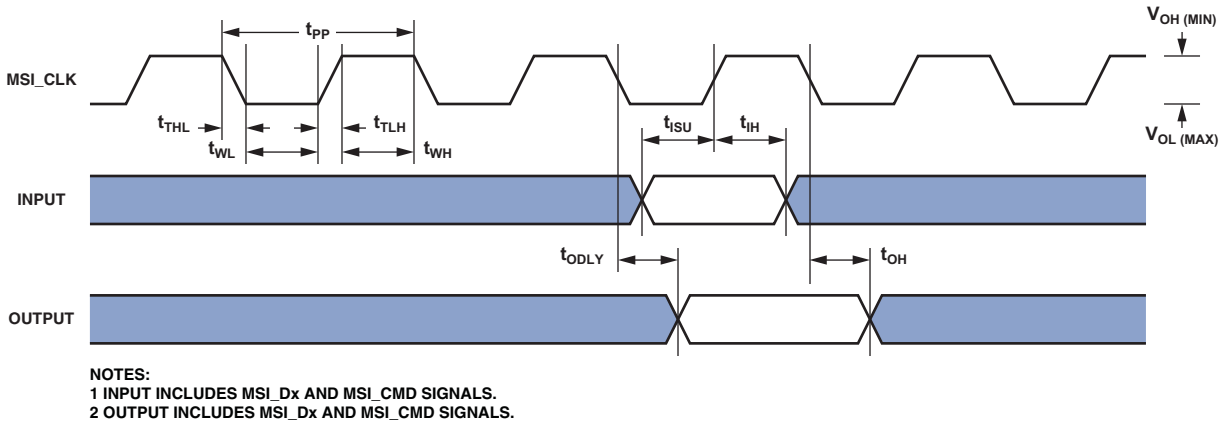


Figure 68. MSI Controller Timing

Program Trace Macrocell (PTM) Timing

Table 89 and Figure 69 provide I/O timing, related to the program trace macrocell (PTM).

Table 89. Trace Timing

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Switching Characteristics				
t_{DTRD}	TRACE Data Delay From Trace Clock MAX		5	ns
t_{HTRD}	TRACE Data Hold From Trace Clock MIN	2		ns
t_{PTRCK}	TRACE Clock Period MIN	12.32		ns

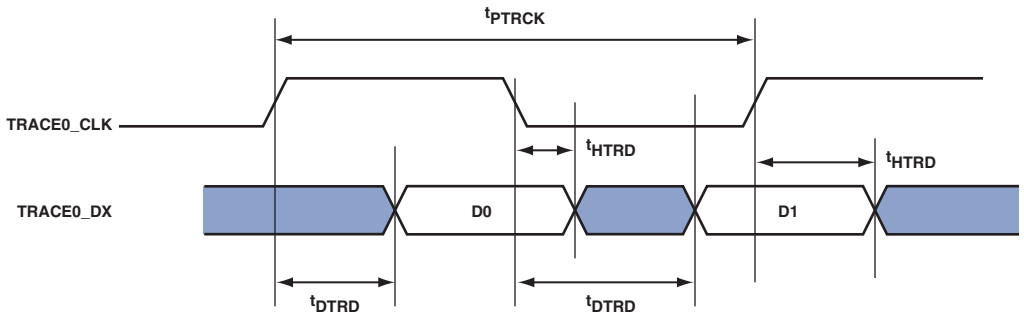


Figure 69. Trace Timing

Debug Interface (JTAG Emulation Port) Timing

Table 90 and Figure 70 provide I/O timing, related to the debug interface (JTAG Emulator Port).

Table 90. JTAG Port Timing

All specifications are based on simulation data and are subject to change without notice.				
Parameter		**Min**	**Max**	**Unit**
Timing Requirements				
t_{TCK}	JTG_TCK Period	20		ns
t_{STAP}	JTG_TDI, JTG_TMS Setup Before JTG_TCK High	4		ns
t_{HTAP}	JTG_TDI, JTG_TMS Hold After JTG_TCK High	4		ns
t_{SSYS}	System Inputs Setup Before JTG_TCK High[1]	4		ns
t_{HSYS}	System Inputs Hold After JTG_TCK High[1]	4		ns
t_{TRSTW}	$\overline{\text{JTG\_TRST}}$ Pulse Width (measured in JTG_TCK cycles)[2]	4		T_{CK}
Switching Characteristics				
t_{DTDO}	JTG_TDO Delay from JTG_TCK Low		12	ns
t_{DSYS}	System Outputs Delay After JTG_TCK Low[3]		17	ns

[1] System Inputs = MLB0_CLKP, MLB0_DATP, MLB0_SIGP, DAI0_PIN20-01, DMC0_A15-0, DMC0_DQ15-0, $\overline{\text{DMC0\_RESET}}$, PA_15-0, PB_15-0, PC_15-0, PD_15-0, PE_15-0, PF_15-0, PG_5-0, SYS_BMODE2-0, SYS_FAULT, $\overline{\text{SYS\_FAULT}}$, $\overline{\text{SYS\_RESOUT}}$, TWI2-0_SCL, TWI2-0_SDA2.

[2] 50 MHz Maximum.

[3] System Outputs = DMC0_A15-0, DMC0_BA2-0, $\overline{\text{DMC0\_CAS}}$, DMC0_CK, DMC0_CKE, $\overline{\text{DMC0\_CS0}}$, DMC0_DQ15-0, DMC0_LDM, DMC0_LDQS, DMC0_ODT, $\overline{\text{DMC0\_RAS}}$, $\overline{\text{DMC0\_RESET}}$, DMC0_UDM, DMC0_UDQS, $\overline{\text{DMC0\_WE}}$, MLB0_DATP, MLB0_SIGP, PA_15-0, PB_15-0, PC_15-0, PCIE_TXP, PD_15-0, PE_15-0, PF_15-0, PG_5-0, SYS_BMODE2-0, SYS_CLKOUT, SYS_FAULT, $\overline{\text{SYS\_FAULT}}$, $\overline{\text{SYS\_RESOUT}}$, TWI2-0_SCL, TWI2-0_SDA.

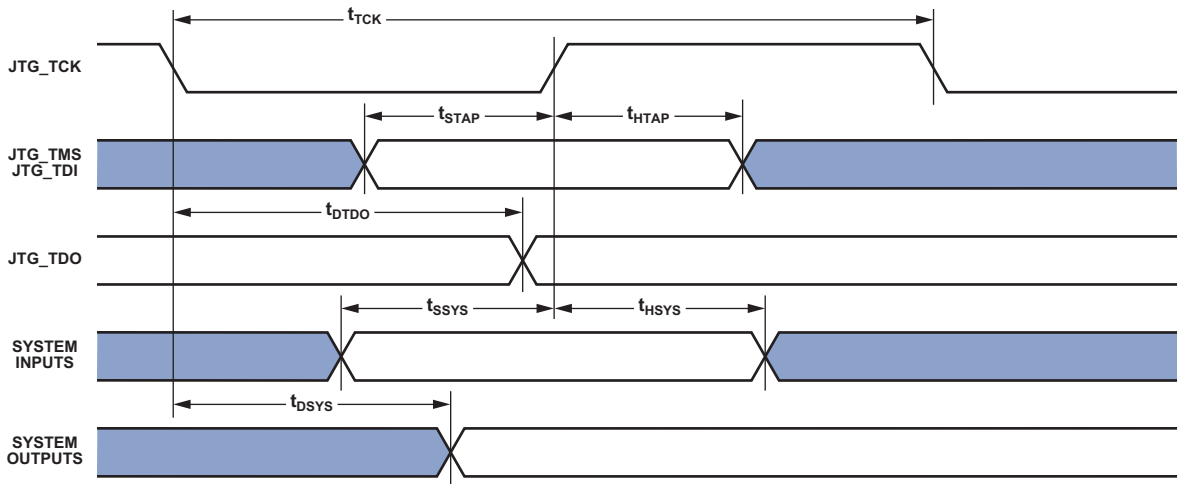


Figure 70. JTAG Port Timing

ENVIRONMENTAL CONDITIONS

To determine the junction temperature on the application printed circuit board, use the following equation:

$$T_J = T_{CASE} + (\Psi_{JT} \times P_D)$$

where:

T_J = Junction temperature (°C).

T_{CASE} = Case temperature (°C) measured by customer at top center of package.

Ψ_{JT} = From Table 91 and Table 92.

P_D = Power dissipation (see Total Internal Power Dissipation for the method to calculate P_D).

Values of θ_{JA} are provided for package comparison and printed circuit board design considerations. θ_{JA} can be used for a first order approximation of T_J by the equation:

$$T_J = T_A + (\theta_{JA} \times P_D)$$

where:

T_A = Ambient temperature (°C).

Values of θ_{JC} are provided for package comparison and printed circuit board design considerations when an external heat sink is required.

In Table 91 and Table 92, airflow measurements comply with JEDEC standards JESD51-2 and JESD51-6. The junction-to-case measurement complies with MIL-STD-883 (Method 1012.1). All measurements use a 6 layer PCB with 101.6 mm × 152.4 mm dimensions.

Table 91. Thermal Characteristics for ADSP-SC573 17 mm × 17 mm 400 CSP_BGA

Parameter	Conditions	Typical	Unit
θ_{JA}	0 linear m/s air flow	14.24	°C/W
θ_{JA}	1 linear m/s air flow	12.61	°C/W
θ_{JA}	2 linear m/s air flow	12.09	°C/W
θ_{JC}		5.71	°C/W
Ψ_{JT}	0 linear m/s air flow	0.08	°C/W
Ψ_{JT}	1 linear m/s air flow	0.14	°C/W
Ψ_{JT}	2 linear m/s air flow	0.17	°C/W

Table 92. Thermal Characteristics for ADSP-SC571 24 mm × 24 mm 176 LQFP-EP

Parameter	Conditions	Typical	Unit
θ_{JA}	0 linear m/s air flow	11.95	°C/W
θ_{JA}	1 linear m/s air flow	10.43	°C/W
θ_{JA}	2 linear m/s air flow	9.98	°C/W
θ_{JC}		11.10	°C/W
Ψ_{JT}	0 linear m/s air flow	0.15	°C/W
Ψ_{JT}	1 linear m/s air flow	0.24	°C/W
Ψ_{JT}	2 linear m/s air flow	0.29	°C/W

ADSP-SC57x/ADSP-2157x 400-BALL BGA BALL ASSIGNMENTS

ADSP-SC57x/ADSP-2157x 400-Ball BGA Ball Assignments (Numerical by Ball Number) lists the 400-ball BGA package by ball number.

ADSP-SC57x/ADSP-2157x 400-Ball BGA Ball Assignments (Alphabetical by Pin Name) lists the 400-ball BGA package by pin name.

ADSP-SC57x/ADSP-2157x 400-BALL BGA BALL ASSIGNMENTS (NUMERICAL BY BALL NUMBER)

Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name
A01	GND	C02	PC_13	E03	PE_03	G04	VDD_EXT
A02	PA_10	C03	GND	E04	PE_02	G05	VDD_INT
A03	PA_09	C04	PA_12	E05	GND	G06	GND
A04	PA_11	C05	PA_14	E06	PB_00	G07	GND
A05	PE_07	C06	PB_03	E07	VDD_EXT	G08	GND
A06	MLB0_CLKN	C07	PB_02	E08	VDD_EXT	G09	GND
A07	MLB0_CLKP	C08	PE_10	E09	VDD_EXT	G10	GND
A08	MLB0_SIGN	C09	PB_06	E10	VDD_EXT	G11	GND
A09	GND	C10	PB_05	E11	VDD_EXT	G12	GND
A10	SYS_XTAL0	C11	$\overline{\text{SYS\_HWRST}}$	E12	VDD_EXT	G13	GND
A11	SYS_CLKIN0	C12	USB0_ID	E13	VDD_USB	G14	GND
A12	GND	C13	USB_CLKIN	E14	JTG_TCK	G15	GND
A13	SYS_XTAL1	C14	PB_12	E15	PE_15	G16	VDD_INT
A14	SYS_CLKIN1	C15	PB_13	E16	GND	G17	PB_15
A15	GND	C16	JTG_TDI	E17	VDD_EXT	G18	DAI0_PIN08
A16	USB0_DP	C17	PE_14	E18	PF_04	G19	DAI0_PIN10
A17	USB0_DM	C18	GND	E19	DAI0_PIN07	G20	DAI0_PIN09
A18	PF_03	C19	PF_08	E20	DAI0_PIN03	H01	PE_01
A19	PF_05	C20	PF_11	F01	PC_02	H02	PC_09
A20	GND	D01	PC_06	F02	PC_03	H03	PC_15
B01	PC_12	D02	PC_08	F03	PC_04	H04	VDD_EXT
B02	GND	D03	PE_04	F04	PE_06	H05	VDD_INT
B03	PA_13	D04	GND	F05	VDD_INT	H06	GND
B04	PA_15	D05	PE_08	F06	GND	H07	GND
B05	PB_01	D06	PE_11	F07	VDD_INT	H08	GND
B06	PB_04	D07	PE_09	F08	VDD_INT	H09	GND
B07	MLB0_DATN	D08	PB_08	F09	VDD_INT	H10	GND
B08	MLB0_DATP	D09	PB_07	F10	VDD_INT	H11	GND
B09	MLB0_SIGP	D10	PB_09	F11	VDD_INT	H12	GND
B10	$\overline{\text{JTG\_TRST}}$	D11	SYS_CLKOUT	F12	VDD_INT	H13	GND
B11	USB0_VBUS	D12	PB_11	F13	VDD_INT	H14	GND
B12	USB_XTAL	D13	USB0_VBC	F14	VDD_INT	H15	GND
B13	PB_10	D14	PB_14	F15	GND	H16	VDD_INT
B14	JTG_TDO	D15	PE_13	F16	VDD_INT	H17	VDD_EXT
B15	JTG_TMS	D16	PE_12	F17	PF_02	H18	DAI0_PIN05
B16	PF_00	D17	GND	F18	PF_09	H19	DAI0_PIN14
B17	PF_01	D18	PF_10	F19	DAI0_PIN02	H20	DAI0_PIN11
B18	PF_06	D19	DAI0_PIN01	F20	DAI0_PIN06	J01	PE_00
B19	GND	D20	DAI0_PIN04	G01	PC_00	J02	PC_07
B20	PF_07	E01	PC_05	G02	PC_14	J03	PC_10
C01	PC_11	E02	PE_05	G03	PC_01	J04	VDD_EXT

Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name
J05	VDD_INT	L13	GND	P01	DMC0_VREF	T09	VDD_DMC
J06	GND	L14	GND	P02	DMC0_RZQ	T10	VDD_DMC
J07	GND	L15	GND	P03	DMC0_A09	T11	VDD_DMC
J08	GND	L16	VDD_INT	P04	DMC0_A10	T12	VDD_DMC
J09	GND	L17	VDD_EXT	P05	VDD_INT	T13	VDD_DMC
J10	GND	L18	DAI0_PIN17	P06	VDD_INT	T14	VDD_DMC
J11	GND	L19	GND	P07	GND	T15	VDD_DMC
J12	GND	L20	DAI0_PIN20	P08	VDD_INT	T16	GND
J13	GND	M01	DMC0_A14	P09	VDD_INT	T17	VDD_DMC
J14	GND	M02	DMC0_A15	P10	VDD_INT	T18	DMC0_A02
J15	GND	M03	PD_09	P11	VDD_INT	T19	DMC0_A01
J16	VDD_INT	M04	VDD_EXT	P12	VDD_INT	T20	$\overline{\text{DMC0\_RESET}}$
J17	VDD_EXT	M05	VDD_INT	P13	VDD_INT	U01	DMC0_DQ15
J18	DAI0_PIN12	M06	GND	P14	GND	U02	DMC0_DQ14
J19	DAI0_PIN13	M07	GND	P15	VDD_INT	U03	TWI1_SDA
J20	DAI0_PIN16	M08	GND	P16	VDD_INT	U04	GND
K01	PD_14	M09	GND	P17	HADC0_VIN1	U05	TWI1_SCL
K02	PD_13	M10	GND	P18	HADC0_VIN0	U06	VDD_EXT
K03	PD_15	M11	GND	P19	HADC0_VREFP	U07	VDD_DMC
K04	VDD_EXT	M12	GND	P20	HADC0_VREFN	U08	VDD_DMC
K05	VDD_INT	M13	GND	R01	PD_06	U09	VDD_DMC
K06	GND	M14	GND	R02	PD_07	U10	VDD_DMC
K07	GND	M15	GND	R03	PD_08	U11	VDD_DMC
K08	GND	M16	VDD_INT	R04	VDD_INT	U12	VDD_DMC
K09	GND	M17	VDD_EXT	R05	VDD_INT	U13	VDD_DMC
K10	GND	M18	HADC0_VIN7	R06	GND	U14	VDD_DMC
K11	GND	M19	HADC0_VIN5	R07	VDD_INT	U15	DMC0_BA2
K12	GND	M20	HADC0_VIN6	R08	VDD_INT	U16	$\overline{\text{DMC0\_WE}}$
K13	GND	N01	DMC0_A11	R09	VDD_INT	U17	GND
K14	GND	N02	DMC0_A13	R10	VDD_INT	U18	DMC0_A06
K15	GND	N03	DMC0_A12	R11	VDD_INT	U19	DMC0_A03
K16	VDD_INT	N04	VDD_DMC	R12	VDD_INT	U20	DMC0_A00
K17	VDD_EXT	N05	VDD_INT	R13	VDD_INT	V01	TWI2_SDA
K18	DAI0_PIN15	N06	GND	R14	VDD_INT	V02	DMC0_DQ13
K19	DAI0_PIN19	N07	GND	R15	GND	V03	GND
K20	DAI0_PIN18	N08	GND	R16	VDD_INT	V04	PD_03
L01	PD_11	N09	GND	R17	VDD_INT	V05	PD_04
L02	PD_10	N10	GND	R18	DMC0_BA0	V06	PD_01
L03	PD_12	N11	GND	R19	HADC0_VIN2	V07	PA_08
L04	VDD_EXT	N12	GND	R20	VDD_HADC	V08	PA_05
L05	VDD_INT	N13	GND	T01	PD_05	V09	PA_03
L06	GND	N14	GND	T02	TWI0_SDA	V10	PA_02
L07	GND	N15	GND	T03	TWI0_SCL	V11	PA_01
L08	GND	N16	VDD_INT	T04	VDD_EXT	V12	PA_00
L09	GND	N17	VDD_EXT	T05	GND	V13	$\overline{\text{SYS\_RESOUT}}$
L10	GND	N18	HADC0_VIN4	T06	VDD_DMC	V14	SYS_FAULT
L11	GND	N19	HADC0_VIN3	T07	VDD_DMC	V15	$\overline{\text{DMC0\_CAS}}$
L12	GND	N20	GND	T08	VDD_DMC	V16	$\overline{\text{DMC0\_RAS}}$

Ball No.	Pin Name
V17	DMC0_BA1
V18	GND
V19	DMC0_A04
V20	DMC0_A05
W01	TWI2_SCL
W02	GND
W03	DMC0_DQ12
W04	DMC0_DQ11
W05	DMC0_DQ09
W06	PD_02
W07	PD_00
W08	PA_07
W09	PA_06
W10	PA_04
W11	DMC0_DQ05
W12	DMC0_DQ04
W13	DMC0_DQ03
W14	DMC0_DQ02
W15	$\overline{\text{SYS\_FAULT}}$
W16	DMC0_ODT
W17	DMC0_A08
W18	SYS_BMODE1
W19	GND
W20	DMC0_A07
Y01	GND
Y02	$\overline{\text{DMC0\_UDQS}}$
Y03	DMC0_UDQS
Y04	DMC0_DQ10
Y05	DMC0_DQ08
Y06	DMC0_UDM
Y07	DMC0_LDM
Y08	DMC0_CK
Y09	$\overline{\text{DMC0\_CK}}$
Y10	DMC0_DQ07
Y11	DMC0_DQ06
Y12	DMC0_LDQS
Y13	$\overline{\text{DMC0\_LDQS}}$
Y14	DMC0_DQ01
Y15	DMC0_DQ00
Y16	DMC0_CKE
Y17	$\overline{\text{DMC0\_CS0}}$
Y18	SYS_BMODE0
Y19	SYS_BMODE2
Y20	GND

ADSP-SC57x/ADSP-2157x 400-BALL BGA BALL ASSIGNMENTS (ALPHABETICAL BY PIN NAME)

Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.
DAI0_PIN01	D19	DMC0_DQ02	W14	GND	G11	GND	M06
DAI0_PIN02	F19	DMC0_DQ03	W13	GND	G12	GND	M07
DAI0_PIN03	E20	DMC0_DQ04	W12	GND	G13	GND	M08
DAI0_PIN04	D20	DMC0_DQ05	W11	GND	G14	GND	M09
DAI0_PIN05	H18	DMC0_DQ06	Y11	GND	G15	GND	M10
DAI0_PIN06	F20	DMC0_DQ07	Y10	GND	H06	GND	M11
DAI0_PIN07	E19	DMC0_DQ08	Y05	GND	H07	GND	M12
DAI0_PIN08	G18	DMC0_DQ09	W05	GND	H08	GND	M13
DAI0_PIN09	G20	DMC0_DQ10	Y04	GND	H09	GND	M14
DAI0_PIN10	G19	DMC0_DQ11	W04	GND	H10	GND	M15
DAI0_PIN11	H20	DMC0_DQ12	W03	GND	H11	GND	N06
DAI0_PIN12	J18	DMC0_DQ13	V02	GND	H12	GND	N07
DAI0_PIN13	J19	DMC0_DQ14	U02	GND	H13	GND	N08
DAI0_PIN14	H19	DMC0_DQ15	U01	GND	H14	GND	N09
DAI0_PIN15	K18	DMC0_LDM	Y07	GND	H15	GND	N10
DAI0_PIN16	J20	DMC0_LDQS	Y12	GND	J06	GND	N11
DAI0_PIN17	L18	$\overline{\text{DMC0\_LDQS}}$	Y13	GND	J07	GND	N12
DAI0_PIN18	K20	DMC0_ODT	W16	GND	J08	GND	N13
DAI0_PIN19	K19	$\overline{\text{DMC0\_RAS}}$	V16	GND	J09	GND	N14
DAI0_PIN20	L20	$\overline{\text{DMC0\_RESET}}$	T20	GND	J10	GND	N15
DMC0_A00	U20	DMC0_RZQ	P02	GND	J11	GND	N20
DMC0_A01	T19	DMC0_UDM	Y06	GND	J12	GND	P07
DMC0_A02	T18	DMC0_UDQS	Y03	GND	J13	GND	P14
DMC0_A03	U19	$\overline{\text{DMC0\_UDQS}}$	Y02	GND	J14	GND	R06
DMC0_A04	V19	DMC0_VREF	P01	GND	J15	GND	R15
DMC0_A05	V20	$\overline{\text{DMC0\_WE}}$	U16	GND	K06	GND	T05
DMC0_A06	U18	GND	A01	GND	K07	GND	T16
DMC0_A07	W20	GND	A09	GND	K08	GND	U04
DMC0_A08	W17	GND	A12	GND	K09	GND	U17
DMC0_A09	P03	GND	A15	GND	K10	GND	V03
DMC0_A10	P04	GND	A20	GND	K11	GND	V18
DMC0_A11	N01	GND	B02	GND	K12	GND	W02
DMC0_A12	N03	GND	B19	GND	K13	GND	W19
DMC0_A13	N02	GND	C03	GND	K14	GND	Y01
DMC0_A14	M01	GND	C18	GND	K15	GND	Y20
DMC0_A15	M02	GND	D04	GND	L06	HADC0_VIN0	P18
DMC0_BA0	R18	GND	D17	GND	L07	HADC0_VIN1	P17
DMC0_BA1	V17	GND	E05	GND	L08	HADC0_VIN2	R19
DMC0_BA2	U15	GND	E16	GND	L09	HADC0_VIN3	N19
$\overline{\text{DMC0\_CAS}}$	V15	GND	F06	GND	L10	HADC0_VIN4	N18
DMC0_CK	Y08	GND	F15	GND	L11	HADC0_VIN5	M19
DMC0_CKE	Y16	GND	G06	GND	L12	HADC0_VIN6	M20
$\overline{\text{DMC0\_CK}}$	Y09	GND	G07	GND	L13	HADC0_VIN7	M18
$\overline{\text{DMC0\_CS0}}$	Y17	GND	G08	GND	L14	HADC0_VREFN	P20
DMC0_DQ00	Y15	GND	G09	GND	L15	HADC0_VREFP	P19
DMC0_DQ01	Y14	GND	G10	GND	L19	JTG_TCK	E14

Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.	Pin Name	Ball No.
JTG_TDI	C16	PC_06	D01	PF_06	B18	VDD_DMC	U12
JTG_TDO	B14	PC_07	J02	PF_07	B20	VDD_DMC	U13
JTG_TMS	B15	PC_08	D02	PF_08	C19	VDD_DMC	U14
$\overline{\text{JTG\_TRST}}$	B10	PC_09	H02	PF_09	F18	VDD_EXT	E07
MLB0_CLKN	A06	PC_10	J03	PF_10	D18	VDD_EXT	E08
MLB0_CLKP	A07	PC_11	C01	PF_11	C20	VDD_EXT	E09
MLB0_DATN	B07	PC_12	B01	SYS_BMODE0	Y18	VDD_EXT	E10
MLB0_DATP	B08	PC_13	C02	SYS_BMODE1	W18	VDD_EXT	E11
MLB0_SIGN	A08	PC_14	G02	SYS_BMODE2	Y19	VDD_EXT	E12
MLB0_SIGP	B09	PC_15	H03	SYS_CLKIN0	A11	VDD_EXT	E17
PA_00	V12	PD_00	W07	SYS_CLKIN1	A14	VDD_EXT	G04
PA_01	V11	PD_01	V06	SYS_CLKOUT	D11	VDD_EXT	H04
PA_02	V10	PD_02	W06	SYS_FAULT	V14	VDD_EXT	H17
PA_03	V09	PD_03	V04	$\overline{\text{SYS\_FAULT}}$	W15	VDD_EXT	J04
PA_04	W10	PD_04	V05	$\overline{\text{SYS\_HWRST}}$	C11	VDD_EXT	J17
PA_05	V08	PD_05	T01	$\overline{\text{SYS\_RESOUT}}$	V13	VDD_EXT	K04
PA_06	W09	PD_06	R01	SYS_XTAL0	A10	VDD_EXT	K17
PA_07	W08	PD_07	R02	SYS_XTAL1	A13	VDD_EXT	L04
PA_08	V07	PD_08	R03	TWI0_SCL	T03	VDD_EXT	L17
PA_09	A03	PD_09	M03	TWI0_SDA	T02	VDD_EXT	M04
PA_10	A02	PD_10	L02	TWI1_SCL	U05	VDD_EXT	M17
PA_11	A04	PD_11	L01	TWI1_SDA	U03	VDD_EXT	N17
PA_12	C04	PD_12	L03	TWI2_SCL	W01	VDD_EXT	T04
PA_13	B03	PD_13	K02	TWI2_SDA	V01	VDD_EXT	U06
PA_14	C05	PD_14	K01	USB0_DM	A17	VDD_HADC	R20
PA_15	B04	PD_15	K03	USB0_DP	A16	VDD_INT	F05
PB_00	E06	PE_00	J01	USB0_ID	C12	VDD_INT	F07
PB_01	B05	PE_01	H01	USB0_VBC	D13	VDD_INT	F08
PB_02	C07	PE_02	E04	USB0_VBUS	B11	VDD_INT	F09
PB_03	C06	PE_03	E03	USB_CLKIN	C13	VDD_INT	F10
PB_04	B06	PE_04	D03	USB_XTAL	B12	VDD_INT	F11
PB_05	C10	PE_05	E02	VDD_DMC	N04	VDD_INT	F12
PB_06	C09	PE_06	F04	VDD_DMC	T06	VDD_INT	F13
PB_07	D09	PE_07	A05	VDD_DMC	T07	VDD_INT	F14
PB_08	D08	PE_08	D05	VDD_DMC	T08	VDD_INT	F16
PB_09	D10	PE_09	D07	VDD_DMC	T09	VDD_INT	G05
PB_10	B13	PE_10	C08	VDD_DMC	T10	VDD_INT	G16
PB_11	D12	PE_11	D06	VDD_DMC	T11	VDD_INT	H05
PB_12	C14	PE_12	D16	VDD_DMC	T12	VDD_INT	H16
PB_13	C15	PE_13	D15	VDD_DMC	T13	VDD_INT	J05
PB_14	D14	PE_14	C17	VDD_DMC	T14	VDD_INT	J16
PB_15	G17	PE_15	E15	VDD_DMC	T15	VDD_INT	K05
PC_00	G01	PF_00	B16	VDD_DMC	T17	VDD_INT	K16
PC_01	G03	PF_01	B17	VDD_DMC	U07	VDD_INT	L05
PC_02	F01	PF_02	F17	VDD_DMC	U08	VDD_INT	L16
PC_03	F02	PF_03	A18	VDD_DMC	U09	VDD_INT	M05
PC_04	F03	PF_04	E18	VDD_DMC	U10	VDD_INT	M16
PC_05	E01	PF_05	A19	VDD_DMC	U11	VDD_INT	N05

Pin Name	Ball No.
VDD_INT	N16
VDD_INT	P05
VDD_INT	P06
VDD_INT	P08
VDD_INT	P09
VDD_INT	P10
VDD_INT	P11
VDD_INT	P12
VDD_INT	P13
VDD_INT	P15
VDD_INT	P16
VDD_INT	R04
VDD_INT	R05
VDD_INT	R07
VDD_INT	R08
VDD_INT	R09
VDD_INT	R10
VDD_INT	R11
VDD_INT	R12
VDD_INT	R13
VDD_INT	R14
VDD_INT	R16
VDD_INT	R17
VDD_USB	E13

CONFIGURATION OF THE 400-BALL CSP_BGA

Figure 71 shows an overview of signal placement on the 400-ball CSP_BGA.

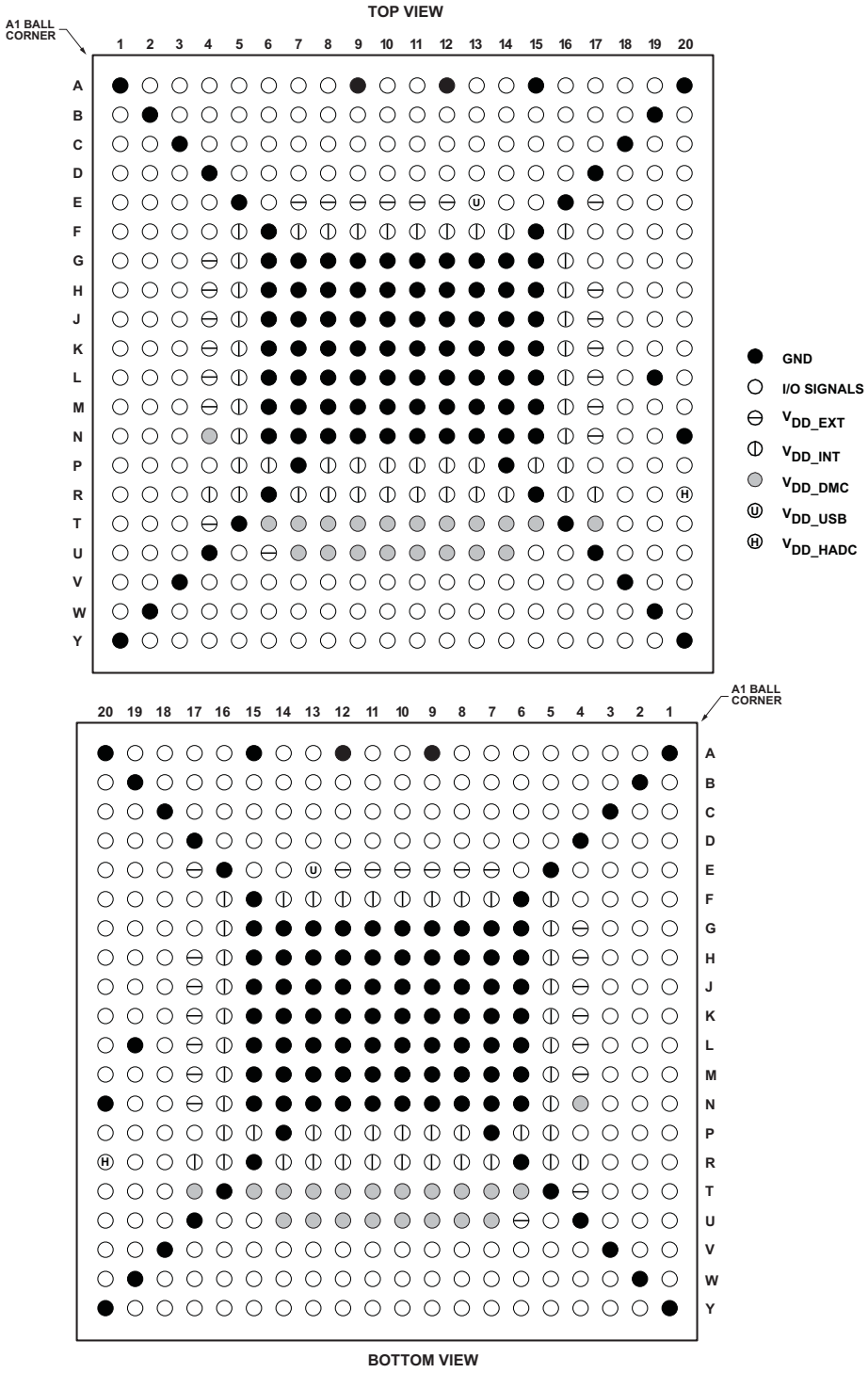


Figure 71. 400-Ball CSP_BGA Configuration

ADSP-SC57x/ADSP-2157x 176-LEAD LQFP LEAD ASSIGNMENTS

ADSP-SC57x/ADSP-2157x 176-Lead LQFP Lead Assignments (Numerical by Lead Number) lists the 176-lead LQFP package by lead number.

ADSP-SC57x/ADSP-2157x 176-Lead LQFP Lead Assignments (Alphabetical by Pin Name) lists the 176-lead LQFP package by pin name.

ADSP-SC57x/ADSP-2157x 176-LEAD LQFP LEAD ASSIGNMENTS (NUMERICAL BY LEAD NUMBER)

Lead No.	Pin Name	Lead No.	Pin Name	Lead No.	Pin Name	Lead No.	Pin Name
01	VDD_INT	41	PD_11	81	SYS_RESOUT	121	DAI0_PIN03
02	GND	42	PD_10	82	VDD_INT	122	DAI0_PIN04
03	VDD_INT	43	PD_09	83	GND	123	DAI0_PIN01
04	PA_15	44	GND	84	VDD_EXT	124	VDD_INT
05	PA_14	45	GND	85	SYS_FAULT	125	GND
06	PA_13	46	VDD_EXT	86	SYS_BMODE0	126	VDD_EXT
07	VDD_INT	47	VDD_INT	87	SYS_BMODE1	127	DAI0_PIN02
08	PA_12	48	PD_08	88	VDD_INT	128	PB_15
09	VDD_EXT	49	PD_07	89	GND	129	VDD_INT
10	PA_10	50	PD_06	90	VDD_HADC	130	VDD_INT
11	PA_11	51	PD_05	91	HADC0_VIN0	131	GND
12	PC_15	52	VDD_INT	92	HADC0_VIN1	132	VDD_INT
13	PA_09	53	TWI0_SDA	93	HADC0_VREFN	133	GND
14	VDD_INT	54	TWI0_SCL	94	HADC0_VIN2	134	VDD_INT
15	GND	55	TWI1_SDA	95	HADC0_VIN3	135	JTG_TCK
16	VDD_INT	56	TWI1_SCL	96	HADC0_VREFP	136	JTG_TDO
17	PC_14	57	TWI2_SDA	97	GND	137	JTG_TDI
18	PC_13	58	TWI2_SCL	98	VDD_INT	138	JTG_TMS
19	PC_12	59	VDD_INT	99	GND	139	VDD_INT
20	PC_11	60	VDD_EXT	100	DAI0_PIN20	140	VDD_EXT
21	VDD_EXT	61	PD_04	101	DAI0_PIN19	141	PB_14
22	PC_10	62	PD_03	102	DAI0_PIN18	142	PB_13
23	PC_09	63	PD_02	103	VDD_INT	143	VDD_EXT
24	PC_08	64	PD_01	104	VDD_EXT	144	PB_12
25	PC_07	65	GND	105	DAI0_PIN17	145	VDD_INT
26	PC_06	66	VDD_INT	106	DAI0_PIN16	146	PB_11
27	PC_05	67	PD_00	107	DAI0_PIN15	147	VDD_EXT
28	PC_04	68	PA_08	108	DAI0_PIN14	148	PB_10
29	PC_03	69	PA_07	109	VDD_INT	149	VDD_EXT
30	VDD_INT	70	PA_06	110	DAI0_PIN13	150	VDD_INT
31	VDD_EXT	71	VDD_EXT	111	DAI0_PIN12	151	SYS_HWRST
32	PC_02	72	VDD_INT	112	DAI0_PIN11	152	VDD_EXT
33	PC_01	73	VDD_INT	113	DAI0_PIN10	153	JTG_TRST
34	PC_00	74	PA_05	114	VDD_INT	154	SYS_CLKIN0
35	PD_15	75	PA_04	115	VDD_EXT	155	SYS_XTAL0
36	PD_14	76	PA_03	116	DAI0_PIN09	156	VDD_INT
37	PD_13	77	PA_02	117	DAI0_PIN08	157	SYS_CLKOUT
38	VDD_EXT	78	VDD_EXT	118	DAI0_PIN06	158	VDD_EXT
39	VDD_INT	79	PA_01	119	DAI0_PIN07	159	PB_09
40	PD_12	80	PA_00	120	DAI0_PIN05	160	VDD_EXT

Lead No.	Pin Name
161	PB_08
162	PB_07
163	VDD_INT
164	VDD_EXT
165	PB_06
166	PB_05
167	VDD_EXT
168	PB_04
169	PB_03
170	VDD_INT
171	VDD_EXT
172	PB_02
173	PB_01
174	PB_00
175	VDD_INT
176	GND
177[1]	GND

[1] Pin no. 177 is the GND supply (see Figure 73) for the processor; this pad must connect to GND.

ADSP-SC57X/ADSP-2157X 176-LEAD LQFP LEAD ASSIGNMENTS (ALPHABETICAL BY PIN NAME)

Pin Name	Lead No.	Pin Name	Lead No.	Pin Name	Lead No.	Pin Name	Lead No.
DAI0_PIN01	123	PA_01	79	PC_15	12	VDD_EXT	149
DAI0_PIN02	127	PA_02	77	PD_00	67	VDD_EXT	152
DAI0_PIN03	121	PA_03	76	PD_01	64	VDD_EXT	158
DAI0_PIN04	122	PA_04	75	PD_02	63	VDD_EXT	160
DAI0_PIN05	120	PA_05	74	PD_03	62	VDD_EXT	164
DAI0_PIN06	118	PA_06	70	PD_04	61	VDD_EXT	167
DAI0_PIN07	119	PA_07	69	PD_05	51	VDD_EXT	171
DAI0_PIN08	117	PA_08	68	PD_06	50	VDD_HADC	90
DAI0_PIN09	116	PA_09	13	PD_07	49	VDD_INT	01
DAI0_PIN10	113	PA_10	10	PD_08	48	VDD_INT	03
DAI0_PIN11	112	PA_11	11	PD_09	43	VDD_INT	07
DAI0_PIN12	111	PA_12	08	PD_10	42	VDD_INT	14
DAI0_PIN13	110	PA_13	06	PD_11	41	VDD_INT	16
DAI0_PIN14	108	PA_14	05	PD_12	40	VDD_INT	30
DAI0_PIN15	107	PA_15	04	PD_13	37	VDD_INT	39
DAI0_PIN16	106	PB_00	174	PD_14	36	VDD_INT	47
DAI0_PIN17	105	PB_01	173	PD_15	35	VDD_INT	52
DAI0_PIN18	102	PB_02	172	SYS_BMODE0	86	VDD_INT	59
DAI0_PIN19	101	PB_03	169	SYS_BMODE1	87	VDD_INT	66
DAI0_PIN20	100	PB_04	168	SYS_CLKIN0	154	VDD_INT	72
GND	02	PB_05	166	SYS_CLKOUT	157	VDD_INT	73
GND	15	PB_06	165	$\overline{\text{SYS\_FAULT}}$	85	VDD_INT	82
GND	44	PB_07	162	$\overline{\text{SYS\_HWRST}}$	151	VDD_INT	88
GND	45	PB_08	161	$\overline{\text{SYS\_RESOUT}}$	81	VDD_INT	98
GND	65	PB_09	159	SYS_XTAL0	155	VDD_INT	103
GND	83	PB_10	148	TWI0_SCL	54	VDD_INT	109
GND	89	PB_11	146	TWI0_SDA	53	VDD_INT	114
GND	97	PB_12	144	TWI1_SCL	56	VDD_INT	124
GND	99	PB_13	142	TWI1_SDA	55	VDD_INT	129
GND	125	PB_14	141	TWI2_SCL	58	VDD_INT	130
GND	131	PB_15	128	TWI2_SDA	57	VDD_INT	132
GND	133	PC_00	34	VDD_EXT	09	VDD_INT	134
GND	176	PC_01	33	VDD_EXT	21	VDD_INT	139
GND	177[1]	PC_02	32	VDD_EXT	31	VDD_INT	145
HADC0_VIN0	91	PC_03	29	VDD_EXT	38	VDD_INT	150
HADC0_VIN1	92	PC_04	28	VDD_EXT	46	VDD_INT	156
HADC0_VIN2	94	PC_05	27	VDD_EXT	60	VDD_INT	163
HADC0_VIN3	95	PC_06	26	VDD_EXT	71	VDD_INT	170
HADC0_VREFN	93	PC_07	25	VDD_EXT	78	VDD_INT	175
HADC0_VREFP	96	PC_08	24	VDD_EXT	84		
JTG_TCK	135	PC_09	23	VDD_EXT	104		
JTG_TDI	137	PC_10	22	VDD_EXT	115		
JTG_TDO	136	PC_11	20	VDD_EXT	126		
JTG_TMS	138	PC_12	19	VDD_EXT	140		
$\overline{\text{JTG\_TRST}}$	153	PC_13	18	VDD_EXT	143		
PA_00	80	PC_14	17	VDD_EXT	147		

[1] Pin no. 177 is the GND supply (see Figure 73) for the processor; this pad must connect to GND.

CONFIGURATION OF THE 176-LEAD LQFP LEAD CONFIGURATION

Figure 72 shows the top view of the 176-lead LQFP lead configuration and Figure 73 shows the bottom view of the 176-lead LQFP lead configuration.

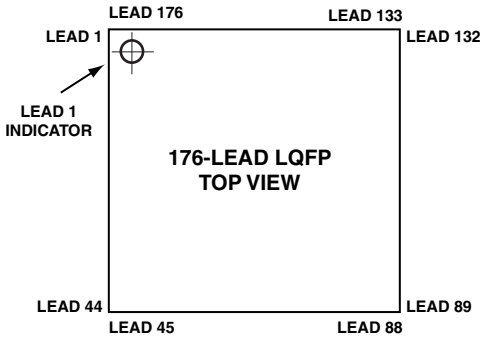


Figure 72. 176-Lead LQFP Lead Configuration (Top View)

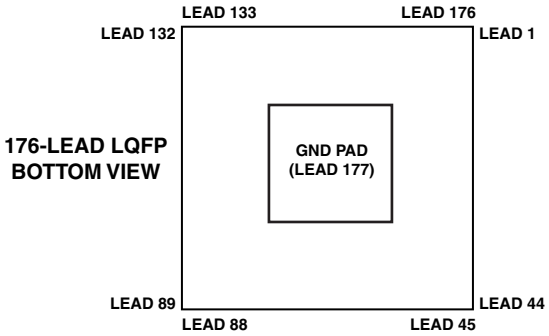


Figure 73. 176-Lead LQFP Lead Configuration (Bottom View)

OUTLINE DIMENSIONS

Dimensions in Figure 74 (for the 400-ball BGA) and Figure 75 (for the 176-lead LQFP) are shown in millimeters.

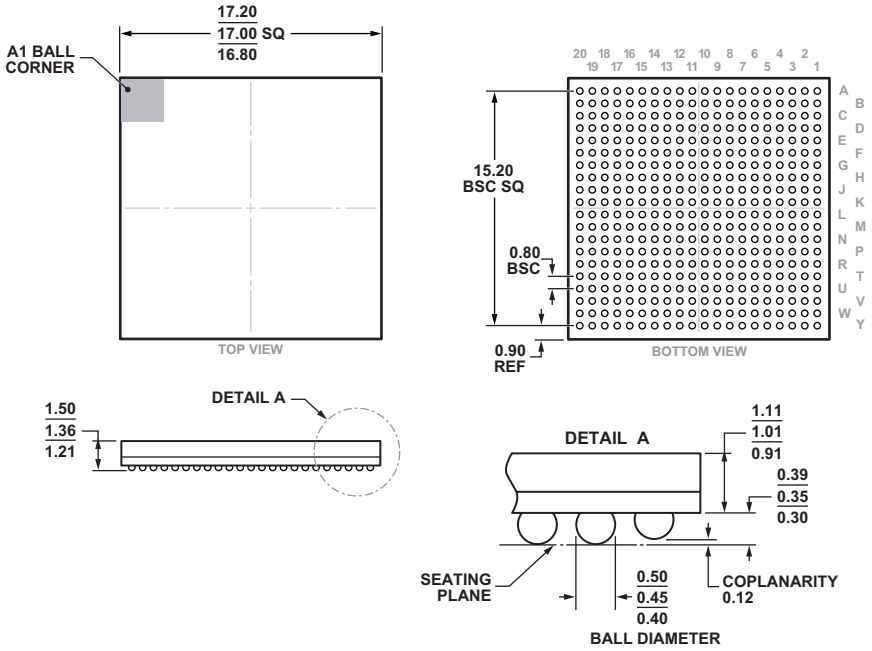


Figure 74. 400-Ball Chip Scale Package Ball Grid Array [CSP_BGA]
(BC-400-2)
Dimensions shown in millimeters

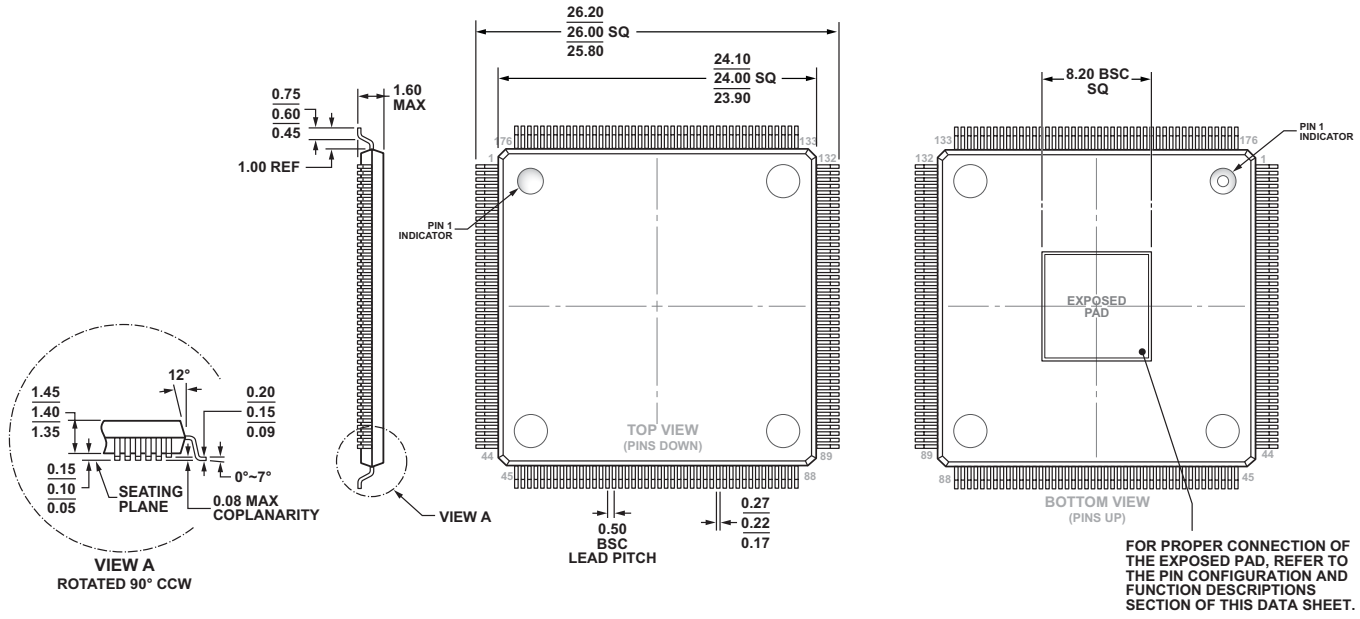


Figure 75. 176-Lead Low Profile Quad Flat Package, Exposed Pad [LQFP]
(SW-176-5)
Dimensions shown in millimeters

SURFACE-MOUNT DESIGN

Table 93 is provided as an aid to PCB design. For industry-standard design recommendations, refer to IPC-7351, *Generic Requirements for Surface-Mount Design and Land Pattern Standard*.

Table 93. CSP_BGA Data for Use with Surface-Mount Design

Package	Package Ball Attach Type	Package Solder Mask Opening	Package Ball Pad Size
BC-400-2	Solder Mask Defined	0.4 mm Diameter	0.5 mm Diameter

PRE RELEASE PRODUCTS

Model[1]	Processor Instruction Rate (Max)	Temperature Range[2, 3]	Package Description	Package Option
ADSP-SC571-SWZENG	450 MHz	NA	176-Lead LQFP, Exposed Pad	SW-176-5
ADSP-21571-SWZENG	450 MHz	NA	176-Lead LQFP, Exposed Pad	SW-176-5

[1] Z =RoHS compliant part.
[2] Referenced temperature is ambient temperature. The ambient temperature is not a specification. Please see Operating Conditions for the junction temperature (T_J) specification which is the only temperature specification.
[3] These are pre production parts. See ENG-Grade agreement for details.

PR14704-0-6/17(PrD)